Integrated and Automated Demographic Surveillance Data Quality Systems for Rural Areas

by

Joseph Tlouyamma

A thesis submitted in fulfilment of the requirements for the degree of

Doctor of Philosophy

in the

FACULTY OF SCIENCE AND AGRICULTURE

(School of Mathematical and Computer Sciences)

at the

UNIVERSITY OF LIMPOPO

Supervisor : Prof. SN Mokwena

April 2024

Declaration

I declare the Integrated and Automated Demographic Surveillance Data Quality Systems for Rural Areas hereby submitted to the University of Limpopo, for the Doctor of Philosophy (Computer Science) has not previously been submitted by me for a degree at this or any other university; that it is my work in design and in execution, and that all material contained herein has been duly acknowledged.

Surname, Initial (title)	Date

Acknowledgements

I wish to express my sincere thanks to the director of DIMAMO Population Health Research Centre, Prof. E Maimela for providing resources and infrastructure to make this study a success. I further extend my gratitude to my colleagues at DIMAMO and the Department of Computer Science for their constructive feedback, which helped shape this research.

I wish to thank the Africa Health Research Institute (AHRI) team, especially Mr E. Ehlers for providing an introduction to the Survey Solutions data collection platform and assisting with technical details in setting up the Survey Solutions. Eugene was instrumental in supporting us in the development of a high-quality electronic data collection system with effective data quality assurance algorithms. Dr. TB Darikwa played a vital role in conceptualizing the study and providing some hints on how data quality can be measured statistically. He also helped review the models in this study.

Lastly, on behalf of DIMAMO PHRC, I would like to thank the South African Population Research Institute Network (SAPRIN) for providing funding that enabled the efficient running of the IT server infrastructure. That positively and directly influenced the outcomes of this study.

Dedications

This work is dedicated to my late grandparents Phineas and Mokgaetsi Tlouamma. They both nurtured me and implanted knowledge in me to maintain work amid bad situations and made me feel that everything is achievable. In addition, I would like to dedicate this to my two boys, Amogelang and Gracious, who never failed to brighten my day, and gave me the strength and hope I needed to venture into the unknown. My mother has always been my rock of support and source of encouragement when times have been difficult, and she has never failed to deliver either of these things in adequate amounts.

Abstract

The Health and Demographic Surveillance System (HDSS) is a data collection system that can track crucial events such as births, deaths, and migrations in well-defined geographic areas, particularly in low- and middle-income households. HDSS tracks the life events of approximately three million people in 18 low- and middle-income African, Asian, and Oceanian nations. Having HDSSs strategically located within a country can provide a more complete picture of health-related and other social problems affecting the public. The HDSS keeps tabs on vital demographic and health indicators as well as other metrics to help shape national policies and programmes for departments of basic education, home affairs, social development, and health. However, their establishment was plagued by several difficulties, including the difficulty of obtaining high-quality data because of the use of antiquated methods or systems. The cornerstone of a well-functioning HDSS is high-quality, and timely health data, which is often lacking in low- and middle-income countries. There is a paucity of high-quality, disaggregated data to monitor health inequities and promote the equitable delivery of health services. HDSSs are confronted with data quality-related problems due to how data is acquired and managed. This study addresses these problems by building a data system that integrates a novel framework known as the 3-Tier Total Data Quality Management Framework (3TTDQMF). The framework manages the quality of data from the point of collection through to the storage in the database. At the core of the framework, is an automated data quality control methodology to autonomously validate and control the quality of data. Open source technologies such as Pentaho data integration (PDI), R application programming interface (R-API), Windows task scheduler, Bash and Python programming languages were used to automate and quality control the data. The experiment was set up in Hyper-converged IT infrastructure running the Windows 2016 server operating system. The results have shown that the proposed approach greatly improved the overall efficiency of the system and the quality of data. The efficiency in dealing with data quality issues was ensured through the implementation of an automated system. The research evaluated the system's capacity to generate high-quality data using measures such as data accuracy, completeness, consistency, timeliness, and validity. All quality metrics exhibited an increasing trend, indicating that the proposed approach led to a substantial improvement in data quality. The results further demonstrated that the use of Pareto analysis and Process control techniques in data quality management can greatly improve the quality of data by identifying and monitoring the causes of data quality issues.

Keywords: *Application programming interface, Automated data quality management, Data collection, Data collection platforms, Data quality, Data quality metrics, Data quality management framework, Electronic data collection, Robotic process automation, Survey Solutions, Total data quality management framework, Pentaho data integration, Windows task scheduler*

Contents

List of Figures

List of Tables

List of Abbreviations

3NF – Third Normal Form

3TTDQM – 3-Tier Total Data Quality Management

4NF – Fourth Normal Form

AHRI – Africa Health Research Institute

AI – Artificial Intelligence

API – Application Programming Interface

BI – Business Intelligence

BPA– Business Process Automation

BPM – Business Process Management

CAPI – Computer-Assisted Personal Interviewing

CL – Centre Line

CRUD – Create, Read, Update and Delete

CSPro – Census and Survey Processing System

CSS – Cascading Style Sheet

DIMAMO – Dikgale, Mamabolo and Mothiba

DQF – Data Quality Frameworks

DSS – Decision Support System

DW – Data Warehouse

EAI – Enterprise Application Integration

EDC – Electronic Data Collection

ELTA – Extract, Load, Transform and Analyse

ERP – Enterprise Resource Planning

ESB – Enterprise Service Bus

ETL – Extract, Transform and Load

FAST – Field Adapted Survey Toolkit

GPS – Geographic Positioning System

GUI – Graphical User Interface

HCI – Hyper-Converged Infrastructure

HDSS – Health and Demographic Surveillance System

HOLAP – Hybrid Online Analytical Processing

HTML – Hyper-Text Mark-up Language

IDE – Interactive Development Environment

IoT – Internet of Things

IPMAP – Information Product Map

IT – Information technology

KPI – Key Performance Indicators

LCL – Lower Control Limit

ML – Machine Learning

MOLAP – Multidimensional Online Analytical Processing

ODK – Open Data kit

OLAP – Online Analytical Processing

PDC – Paper-Based Data Collection

PDI – Pentaho Data Integration

RDBM – Relational Data Management System

ROLAP – Relational Online Analytical Processing

RPA – Robotic process automation

SAPRIN – South African Population Research Infrastructure Network

SOA – Service–Oriented Architecture

SPC – Statistical Process Control

SQL – Structured Query Language

SWRL – Semantic Web Rule Language

UCL – Upper Control Limit

UML – Unified Modelling Language

VA – Verbal Autopsy

VM – Virtual Machine

VPN – Virtual Private Network

WHO – World Health Organisation

XML – Extensible Mark-up Language

Chapter 1

Introduction and Study Background

1.1 Introduction

Health and Demographic Surveillance Systems (HDSSs) are established to collect data on a dynamic cohort's health and demography across time. Data collected tracks crucial events like births, deaths, and migrations in a geographically defined region (Sankoh and Byass, 2012, Agarwal et al., 2017). HDSS in developed countries are well-established (Kenyon, 2005, Cohen et al., 2006) to track, monitor, and manage the burden of diseases and other health issues affecting the masses of people. In developing countries, such systems lack the infrastructure and resources essential to maintain good data quality and offer appropriate coverage of demographics and health issues. HDSS tracks the life events of approximately three million people in 18 low- and middle-income African, Asian, and Oceanian nations (Sankoh and Byass, 2012). HDSSs are being established in growing numbers in Africa, with over thirty-seven (37) of them spread across different countries including South Africa. HDSSs in South Africa (see Figure 1.1) are strategically located within different provinces to provide good demographic and health coverage. Having HDSSs strategically located within a country can provide a more complete picture of health-related and other social problems affecting the public (Ye et al., 2012).

HDSS offers scientific exploration of the challenges faced by selected populations in a particular geographic area. In addition, provides crucial information in situations where no data or complete information is available. The HDSS keeps tabs on vital demographic and health indicators as well as other metrics to help shape national policies and programmes for departments of basic education, home affairs, social development, and

health. However, HDSSs often lack the infrastructure necessary to adequately monitor demographic and health events, especially in under-resourced rural areas. Rural regions have difficulty collecting dependable health-related data (Homan et al., 2015). Dependable evidence collected at the local level is essential for evidence-based policymaking and disease control. The World Health Organisation (WHO) identifies broad rural regions in Sub-Saharan Africa as a reservoir for a range of mostly avoidable communicable diseases, such as malaria, TB, and HIV/AIDS (World Health Statistics, 2014). HDSSs have the ability to address the burden of diseases by proposing intervention strategies to monitor and curb the spread. They are being established at a growing number of sites to study a variety of health indicators and illnesses (Homan et al., 2015). However, their establishment was plagued by several difficulties, including the difficulty of obtaining high-quality data because of the use of antiquated methods or systems. The cornerstone of a well-functioning HDSS is high-quality, and timely health data (Sorenson and Chalkidou, 2012), which is often lacking in low- and middle-income countries.

There is a paucity of high-quality, disaggregated data to monitor health inequities and promote the equitable delivery of health services. Although there have been strides made to improve information systems, data quality is still poor, particularly in HDSSs that are based in developing countries (Ali et al., 2018). The incapacity of the information system in most HDSSs to identify and avoid mistakes is the root cause of poor data quality. Moreover, these HDSSs do not routinely use context-specific data quality monitoring embedded in their information systems. Poor data quality leads to distorted findings, inaccurate models, loss of revenue, and possibly the collapse of the organisation. It was reported in the literature that organisations lose billions of dollars yearly due to data quality issues (Ilyas and Chu, 2015). To understand population health trends, accurate, timely, and consistent health data is necessary (World Health Organisation, 2020). Reliable data is necessary for decision-makers to design suitable policies, allocate resources, and rank initiatives.

The reliability of data often depends on data collection methods. Due to a lack of resources in areas where HDSSs are established, paper-based data collection (PDC) methods are prominent. PDC has been a reliable method of data collection for many years, despite being susceptible to data quality violations. Such methods are error-prone and serve as breeding grounds for data quality issues (Njuguna et al., 2014). Most HDSSs still use PDC methods, while sites with experience implementing electronic data collection (EDC) have rarely publicly stated their experience or the comparative effect of PDC and EDC systems (Homan et al., 2015; Zeleke et al., 2021). Despite evidence from pilot projects suggesting otherwise, EDC technologies have not been widely adopted regardless of their potential to improve data quality, boost productivity, and reduce survey

costs. EDC is a viable alternative to PDC methods for addressing the latter's inherent shortcomings. The proliferation of electronic devices makes it possible to collect high-quality data, which resulted in the generation of an enormous amount of data in recent years. Electronic devices provide an adaptable and robust infrastructure for the acquisition of superior data in a wide range of sectors and applications. To get optimal outcomes, it is critical to select the appropriate electronic data collection technologies and develop data collection methods that are in line with one's particular requirements and goals.

To identify health gaps and inequities and inform targeted, effective, and cost-effective decision-making, we urgently need robust health information systems that provide access to accurate, timely, and accurate disaggregated data (World Health Organisation, 2021). The sustainability of HDSSs depends on their ability to autonomously deal with data quality problems. This research work proposes a data system, synonymous with an information system, which employs various methodologies to deal with issues affecting the data quality in the HDSS domain. The system incorporates a novel 3-Tier Total Data Quality Management (3TTDQM) framework to validate the input data and produce high-quality output data. The framework takes into account the complexity of the longitudinal nature of HDSS data by implementing the necessary measures and user-friendly interfaces to maintain efficiency and improve data quality. This work also took into account the impact of operational practices that can lead to compromised data quality. Consequently, an automated system was proposed to integrate the workflow and eliminate previously employed manual processes. We also incorporate statistical techniques into the system to identify, monitor, and manage processes that may exacerbate issues related to data quality. This holistic approach to data quality management has the potential to enhance efficiency while improving the quality of data.

The structure of this chapter is as follows: Section 1.2 outlines the background of the study area and manual data collection processes. Section 1.3 introduces data collection platforms for electronic data collection while section 1.4 discusses the system for data quality improvement. Section 1.5 gives an overview of data quality management. Presented in section 1.6 are the significance of data integration and system automation in HDSS and section 1.7 provides the motivation for data quality in HDSS and further presents the problem statement. Section 1.8 discusses the major contributions and implications of this research. Section 1.9 provides a thesis outline while section 1.10 concludes the chapter.

1.2 Background: Study Area

1.2.1 Geographic Location and Population Coverage

South Africa is divided into nine provinces and had an estimated population of about 60.7 million in mid-year 2022, according to the United Nations. The country has seven major health and demographic surveillance systems (HDSS) in different provinces under the South African Population Research Infrastructure Network (SAPRIN), including two HDSSs that are yet to be established (See Figure 1.1). These are Agincourt (Kahn et al., 2012) in Mpumalanga, Africa Health Research Institute (AHRI) (Collinson et al., 2022) and Ethekwini in KwaZulu-Natal, C-Sharp[1] in Western Cape, DIMAMO (Alberts et al., 2015) in Limpopo, the GRT-Inspired[2] in Gauteng. Soweto and Thembelihle (Adedini et al., 2021) HDSSs in Gauteng (not shown in Figure 1.1) also contribute to the total number of HDSSs in South Africa, bringing the total to nine (9).

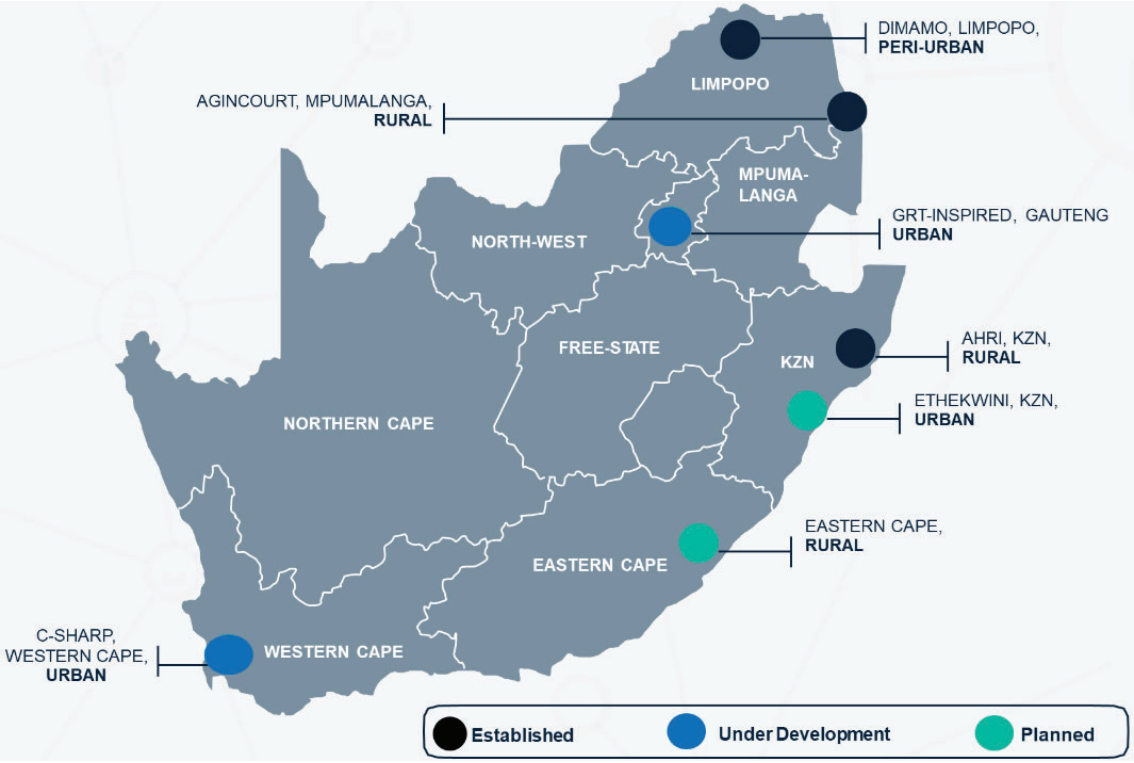


FIGURE 1.1: The distribution of HDSS across different provinces of South Africa (https://saprin.mrc.ac.za/nodes.html)

The Limpopo province, located in the northern part of South Africa, is known for being mostly rural with a population of approximately 5.8 million. Due to the province's rural nature and a lack of services and infrastructure, many of its people face significant health

[1] https://www.thehealthfoundation.org.za/cape-town-surveillance-through-healthcare-action-research-project-c-sharp/

[2] https://grt.ac.za/

and economic challenges. That necessitated the establishment of then DikgaleHDSS (Alberts et al., 2015), now DIMAMO HDSS. DIMAMO HDSS is situated approximately 40 km northeast of Polokwane, the capital city of Limpopo. It was established in 1995 and had a baseline census of about 1,036 households. Due to the necessity to collect more data and explore various demographic and health-related concerns affecting rural communities, the surveillance area was expanded in 2010 to include 5,000 households with approximately 33,954 individuals under surveillance. With more scientists or researchers interested in DIMAMO datasets for exploring the challenges facing poorly resourced communities, another expansion was made in 2018 to include more households. With expansion, the number of households in the surveillance area increased to 21,459 and that resulted in an exponential increase in population to over 100,000.

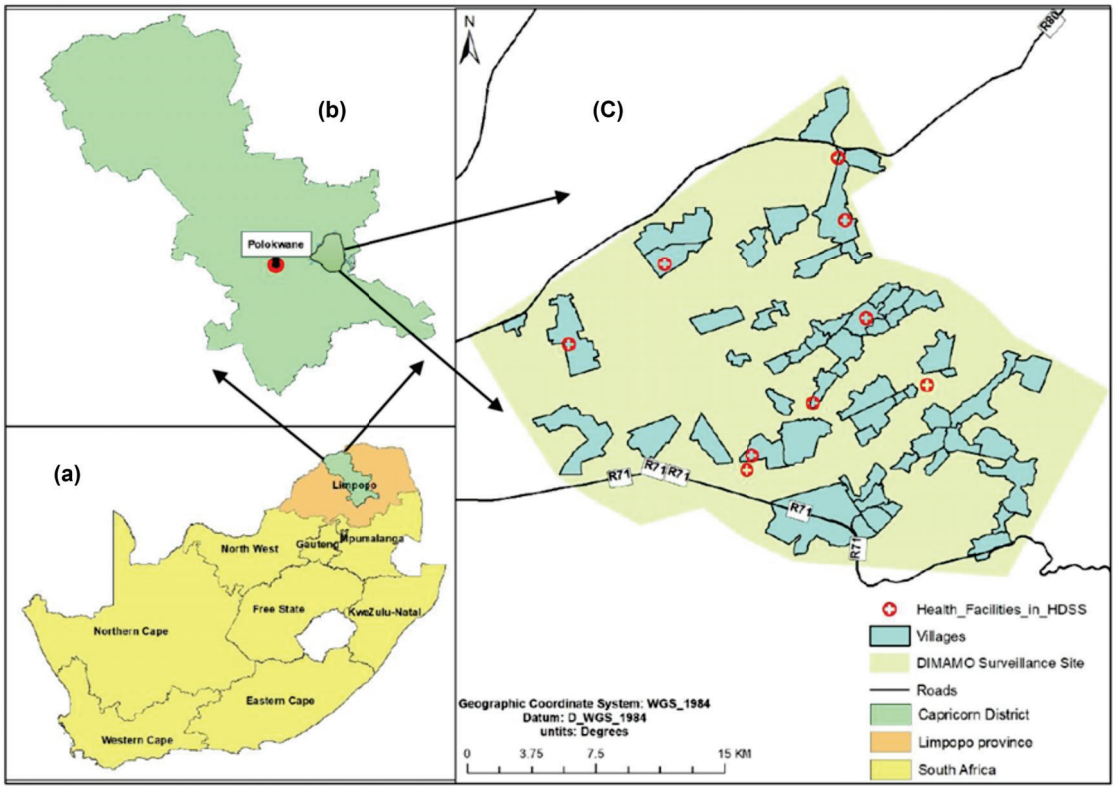


FIGURE 1.2: Eye-bird view of DIMAMO surveillance area

The maps in Figure 1.2 show the location of the HDSS, with villages scattered sporadically throughout the surveillance area. Figure 1.2 (a) shows a South African map with nine provinces, including Limpopo Province, where HDSS is located. The map presented in Figure 1.2 (b) depicts a district in which HDSS is found, while Figure 1.2 (c) shows the DIMAMO surveillance area with villages, from which data is collected. The surveillance area comprises 57 geographically dispersed villages in low and high-lying areas and has an area coverage of approximately 545.175 km^2. It is bounded by two major roads (R71 and R81), which connect Polokwane and Phalaborwa towns.

1.2.2 Current Paper-based Data Collection Procedures

1.2.2.1 Data Collection Processes

Figure 1.3 presents a workflow for paper-based data collection. In preparation for data collection, barcoded forms are printed and arranged for scanning. During scanning, the barcode on each form is electronically entered into the database. Scanning is performed to keep track of which forms are allocated to the field. Once allocated, the forms are manually distributed and dispatched to the field for data collection. It is at this stage that a team of fieldworkers goes out to start collecting data. Much of data quality issues occur at this first point of data entry due fallibility of human nature (Kovacs, Hoekstra and Aczel, 2021). Issues such as erroneous or invalid data entries, inconsistencies, missing values, and inaccurate capturing of data values may be introduced at this stage. Such issues are difficult to detect due to the lack of automated validation mechanisms in PDC.

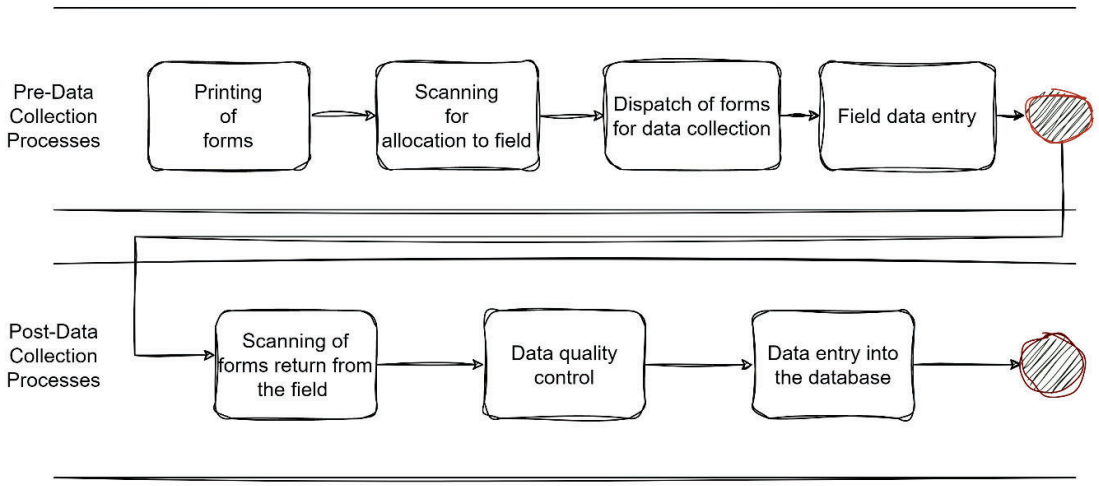


FIGURE 1.3: Paper-based data collection workflow

The forms are scanned again when they are returned from the field. This process ascertains that all forms allocated to the field are returned. Scanning is preceded by data quality control, in which quality controllers manually check if the data on completed forms has some form of quality violations. If the collected data meets quality standards, then the forms are transferred to the data capturing station; otherwise, they are returned to the field with a clearly handwritten rejection note. Manually checking the quality of data has some major drawbacks: difficulty in detecting some inconsistencies, miss detecting some issues due to lapse of concentration, inefficient and time-consuming activity. The intermediate capturing of data into the database is performed using the ADO.Net application. The application only allows authorised users to log into the application and perform data capture and manipulation. It provides low-level reporting services to allow users to track the number of forms printed, assigned to the field, and

captured into the database. The application has data validation mechanisms in place, but some typos can go undetected and be entered into the database. The obvious errors may be misspelled names, dates, open narratives, etc. Such errors are very difficult or even impossible to detect by using the current application. The major drawback of the current system is that these errors are introduced when the data are collected from the field and again during the intermediate capturing (Reczek et al., 2023) into the database.

1.2.2.2 Sources of Data Quality Issues

The effectiveness and relevancy of HDSSs depend on their ability to maintain high data quality. However, achieving high-quality data is difficult in DIMAMO HDSS due to inherent manual processes and a lack of a proper data system (information system) to identify, monitor, and mitigate issues related to data quality. Manual data collection has a detrimental impact on data quality due to its inability to validate data at the first point of entry (Kupzyk and Cohen, 2015). Neglecting quality validation mechanisms at the first point of data entry may result in lengthy and costly data management processes (Thindwa et al., 2020). Resolving data quality problems at the first point of entry is cost-effective and less time-consuming compared to dealing with issues that crawl into the database (Azeroual et al., 2019). Most HDSSs are still using paper questionnaires as a means of data collection, despite the many potential issues associated with this method (Zeleke et al., 2021). This method is time-consuming, and error-prone and often results in longer turnaround times, which delay timely data analysis (Flaxman et al., 2018; Ahmed et al., 2018). A large number of inconsistent, erroneous, incomplete, and invalid data entries were discovered in the current database, perhaps as a result of the data collection and management practices used in the past.

A number of challenges are evident in HDSS when manually capturing and quality controlling data: 1) Data entry errors: When manually inputting data, human error is a typical obstacle. Inaccurate and incorrect data may result from typos, transcription problems, and other flaws. 2) Time-consuming activity: Especially when working with huge datasets, manual data input is sometimes a time-consuming operation. It can be resource- and labour-intensive. 3) Costly: Depending on the size and complexity of the dataset, it may be prohibitively expensive to hire a team of people to handle data input and quality assurance. 4) Inefficiency: When compared to electronic or automated data collection technologies, manual data input is more time-consuming and error-prone. Data processing and analysis times may increase as a result. 5) Data integrity: Maintaining data integrity may be difficult. Errors might go undetected and lead to data discrepancies if sufficient validation and quality control methods are not in place. 6)

Security and privacy: The manual manipulation of sensitive health and demographic information might pose security and privacy problems. To secure data from unauthorized access or breaches, adequate precautions must be implemented. 7) Scale and volume: Manual data input may become impractical when working with a large-scale HDSS, causing additional data quality issues. 8) Data loss: PDC is susceptible to loss or damage, and sloppy human data input might result in data loss. 9) Audit trail: Maintaining an audit trail of modifications and additions might be difficult with PDC, but it is critical for ensuring data openness and accountability. These challenges can be addressed by implementing EDC systems and automated data quality control measures. These solutions can help eliminate mistakes, boost productivity, and strengthen data security. In addition, performing frequent data quality audits and utilizing data validation checks might help offset some of the difficulties associated with human data entry in HDSS.

1.2.2.3 Transcriptions and Inaccurate Capturing of Data

Paper-based data collection often introduces a lot of data quality issues, especially due to transcription processes (e.g., transcribing GPS coordinates, measurements, etc.) and multiple points of data entry (field data entry and recapturing data into the database using an application). Due to the transcription of GPS coordinates, HDSS had a large number of coordinates that were out of range, resulting in difficulties in locating affected dwellings. Incorrectly, geocoded dwellings are difficult to locate because their precise position cannot be accurately shown on the map. Nearly 30% of the dwellings in the system have inaccurate coordinates and are untraceable. Additionally, a preliminary evaluation of the data in the database has shown the existence of more problems concerning data quality. A typical example is the date of death coming before their date of birth, which means that the individual died before their birth. Furthermore, there were invalid date entries for events (”3013-03-02,” ”2985-04-16,” ”2974-09-09,” ”2818-03-17,” and more), and these may have been inadvertently entered into the system as a result of human error and defective validation mechanisms. In certain instances, inaccurate village names were recorded (e.g., “Ga = -Mokgopo” instead of “Ga-Mokgopo”). In the surveillance area, individuals move in and out through a process called ”migration.” In a practical sense, an individual cannot immigrate and out-migrate to or from the same household on the same day. Having mentioned that, 3.29% of individuals in the current database have the same dates of migrations (in-migration and out-migration) in the same household. In a different case, 10.74% of the individuals had duplicate events, and 12.1% of errors accounted for individuals’ event dates were greater than their observation dates. Apart from human error in data capture, the quality of data may be

impacted by manual processes involved, including printing, scanning, dispatching, processing, managing, and capturing data from questionnaires into the database. Manually performing these tasks often results in errors or lapses in attention owing to the inherent fallibility of humans, particularly when volumes reach an alarming level. There may also be monetary repercussions because of the potential requirement for more human resources to undertake assigned duties manually, as well as any additional equipment. Many problems that require immediate attention are evident in this HDSS, and if not properly addressed, these problems may accumulate over some time and detrimentally affect data quality.

1.3 Data Collection Platforms

To address data quality issues, platforms such as Open Data kit® (ODK) (Loola Bokonda et al., 2020), Redcap® (Harris et al., 2019), Survey Solutions® (World Bank, 2018), etc. are available to provide data collection and management tools. These platforms enable the hosting of EDC and data management tools for the collection of high-quality data. The robustness of mechanisms used to validate data quality differs with each platform. For example, ODK offers a complex validation mechanism, which prevents users from skipping to the next question unless the current mandatory question is answered. However, lacks a feature to perform validation of the data based on historic or previously collected data (Brunette et al., 2017). This feature is well incorporated in Survey Solutions® to validate the consistency between the collected and historical data. The major drawback of Survey Solutions® is its inability to disallow skipping to the next question until the mandatory question is answered, which is well handled in ODK. Survey Solutions® only allows the throwing of an error message when the mandatory question is skipped. Unlike, paper-based data collection, EDC offers some mechanisms to improve data quality. That includes skipping patterns, validation algorithms, or macros. According to Sheikhali et al. (2016), EDC generally improves data quality, but only when skipping patterns and validation algorithms are incorporated. Zeleke et al. (2019), Njuguna et al. (2014), Thriemer et al. (2012), and Were et al. (2010), also made this assertion in the literature.

1.4 System for Data Quality Improvement

We define the system for data quality improvement as a system that takes data as input from a data source (e.g., data collection, data file, databases) and improves its quality through various mechanisms. Such systems are often referred to as information systems.

A broad definition of an information system is "a network of computerised devices and computer programs used to gather, process, store, and disseminate data in support of organisational goal achievement and management" (Laudon and Laudon, 2000). Setyowati et al. (2021) also define an information system as a collection of linked pieces that operate as a single component to integrate data, process and store data, and disseminate data; or a blend of data, policies, data sources, networks, people, hardware, software, and information management processes in an organisation's storage, processing, and dissemination of data. It is anticipated that the state-of-the-art information system will give organisations reliable and convenient data, allowing them to gain a competitive edge (Beatrix, 2022). The lack of data quality in information systems creates significant dangers to organisation decision-making and operations (Liu et al., 2020). Data quality refers to the discrepancies between the perspective of the world offered by an information system and the actual state of the world (Parssian et al., 2004). This indicates that there is a strong correlation between the quality of the data and the information system. Enhancing the quality of the information system can positively influence data quality. The ability of the information system to provide efficient management of data quality depends on the provision of appropriate features, cutting-edge design, and acceptable ergonomics (Ibrahim et al., 2021). Such provisions are essential to identify, validate, and control data quality concerns, as well as continuously measuring data quality.

Data quality measurement testing and simulation must be possible inside the system and a dashboard-style interface for data quality monitoring must be available (Schäffer and Leyh, 2017). Depending on the methodologies implemented, data quality enhancement research for information systems could be classified into two streams (Liu et al., 2020). The first employs a data-driven enhancement strategy that substitutes low-quality values with high-quality ones using specific means or procedures. This strategy uses an assessment of data quality at the data item level to determine completeness, timeliness, accuracy, relevancy, validity, consistency, etc. However, the processes that lead to data collection and processing may affect the quality of the underlying data. That leads to the second improvement strategy, which is the process-driven technique that is used to manage and enhance the data processing flow.

Our study takes both approaches into account to ensure optimal data quality. A novel 3TTDQM framework is proposed to cater for data-driven improvement strategies. This framework accepts data of compromised quality and runs them through a data-cleaning engine to produce better-quality data. For process-driven strategies, we propose process automation and statistical process control methods to automate, identify, monitor, and control processes that may lead to poor data quality. The growing importance of data quality necessitates adequate measurements, and its quantification is vital for measuring

the quality of data in a cost-effective manner (Heinrich et al., 2007). According to Gitzel et al. (2015), a data quality dashboard may identify and flag issues, which ultimately necessitates data quality improvement. Sebastian-Coleman (2012) indicated that additional research is necessary to understand how data quality and monitoring may be measured. Implementing a system for data quality checks makes it easier to identify the sources of data quality issues, thereby improving quality (Azeroual et al., 2018).

1.5 An Overview of Data Quality Management

In the last 10 years, data quality research has grown significantly and has become a popular field of study (Liu et al., 2020). From the point of view of the user, data quality can be broken down into four main categories: accuracy, relevance, representation, and access (Wang and Strong, 1996). These categories must not limit the organisation's data quality assessment strategies. The selection of data quality dimensions must be tailored to local circumstances (Tayi and Ballou, 1998). Data quality is a multidimensional concept, so the quality measurements that are deemed relevant in one organisation may not necessarily be suitable for another (Borek et al., 2013; DAMA, 2013). Hence, HDSSs must adopt universal approaches to data quality management and monitoring.

The quality of data in HDSS is critical for monitoring the health status and evaluating the effectiveness of interventions. Some of the challenges of data quality in HDSS include: 1) Missingness of data due to incomplete reporting or loss to follow-up. This may influence the precision of the data and the ability to draw conclusions. 2) During manual data entering, data entry errors may occur, resulting in erroneous and inconsistent data. Human error, inadequate data, or conflicting data definitions may cause these problems. 3) Due to the length of time over which HDSS data is typically gathered, longitudinal data management presents unique challenges. Ineffective data management procedures may result in data inconsistencies, loss of data, and diminished data quality. 4) To guarantee the correctness and completeness of HDSS data, rigorous data quality control is required. Nonetheless, the process of quality control may be intricate and time-consuming, introducing the chance of inconsistencies and errors. 5) The analysis of HDSS data demands particular knowledge and abilities. Inadequate analysis may lead to biased findings and inaccurate conclusions.

Generally, guaranteeing data quality in HDSS needs a systematic and exhaustive strategy including efficient data administration, quality control procedures, training, standardized data collecting techniques, and data quality assessments. Assessing the quality of data involves contrasting its actual worth with some standard against which it may be

judged (Batini et al., 2009). For example, the assessment of data timeliness takes into account the comparison of the data collection date (dt) and the current date (ct). Depending on how the organisation or HDSS defines the data timeliness, the data is timely if and only if dt-ct=0. In terms of data accuracy, there should be no difference between the real-world representation of data and the data captured in the system. Regarding data completeness, the complexity may differ depending on whether the data collection is cross-sectional or longitudinal. In a cross-sectional approach, data are collected at a one-time point (Kesmodel, 2018) and data completeness is evaluated for compulsory attributes with missing values. Similarly, the case with the longitudinal approach. The longitudinal data collection, in contrast to the cross-sectional, gathers data from the same target population over some time (Ployhart and MacKenzie, 2014). That is, the data subjects sampled in the first round of data collection must be followed throughout the project to ensure complete coverage. Data completeness is compromised in cases where some data subjects are lost to follow for various reasons. On the other hand, data consistency considers how data about the same data subject is collected over a period of time. For example, if a data subject is currently recorded as “female” but was recorded as ”male” in the previous round, then this shows some form of inconsistency. The measure of consistency in this regard may be determined by comparing current data items with historic data. Lastly, the validity of the data determines the quality of decision-making and the relevancy of the data in the information system. To determine validity, there must be a standard to measure against. For example, a date variable must contain values that conform to a legitimate date format. Before initiating any data quality enhancement effort, it is necessary to conduct a data quality assessment (Baškarada and Koronios, 2014). Multiple influential authors have underlined that ”only what can be assessed can be enhanced” (Wand and Wang, 1996; Wang and Strong, 1996; English, 1999). Therefore, data must be measured to assess its quality over time.

The four primary steps of data quality assessment are definition, measurement, analysis, and improvement, and include multiple stakeholders such as data managers, data consumers, and data collectors (Wang, 1998; Strong et al., 2002). The definition step involves the identification and analysis of the current data quality problems. At this step data quality metrics and requirements are identified and it is decided to continuously monitor the data quality (Silvola et al., 2019). The quality of data can be measured in a variety of ways, both subjectively and objectively, depending on the metrics used. Measurements such as completeness, accuracy, consistency, timeliness, etc. may be measured objectively (Ayele et al. 2021), while user perceptions of data quality may be assessed subjectively. The analysis step takes a deep dive into the level of data quality measurement obtained and compares it to the earlier data quality requirement defined in the initial step. The national data quality standard, according to Haftu et al. (2021), is 95%

for data completeness and 90% for accuracy. The quality of the measured data must be compared to the established benchmark and appropriate action must be taken if the quality falls below the predefined target value. Improvement strategies must be implemented and monitored to constantly enhance the data quality. Data quality indicators serve as basic pillars for accurate data-driven decisions (Ehrlinger et al., 2019).

1.6 Integration and Automation of Processes

1.6.1 Data Integration

The term "data integration" refers to the method through which disparate datasets are unified into a single cohesive whole. The ultimate goal of data integration is to create a more comprehensive database and to enhance the quality of the information by combining data from a variety of sources (Song et al., 2019). Data integration in the HDSS may improve the research process in several ways: 1) Integrating data may enhance data quality by minimizing mistakes and inconsistencies. By merging data from numerous sources, researchers may evaluate the data's correctness and find errors, assuring the data's consistency and reliability. 2) Integration of data may offer a more complete picture of the population's health state and requirements. Researchers may acquire a more comprehensive picture of health trends and patterns by merging data from many sources to fill in data gaps. 3) Integrating data may improve decision-making by giving a more accurate and complete picture of the population's health condition and requirements. Policymakers and other stakeholders may make better-informed choices regarding resource allocation and intervention planning if they have access to high-quality, integrated data. 4) Data integration may enhance productivity by decreasing the time and resources necessary to acquire and handle data. By integrating data from diverse sources, researchers may improve data management operations and decrease effort duplication.

Integrating data from multiple sources, researchers can gain a more complete understanding of the health status and needs of the population, leading to better health outcomes. In addition, provides a more universal data database to improve efficiency and data quality (Song et al., 2019). Operating many systems simultaneously is expensive, error-prone, and may result in conflicting data and divergent outcomes (Balkan and Goul, 2010). Integration not only reduces mistakes and improves reaction times, but also promotes the sharing of data throughout business operations and departments (Booth et al. 2000; Gattiker and Goodhue 2004), and promotes the incorporation of

new knowledge. Hence, some level of data integration is necessary to maintain the quality of data (Ibrahim et al., 2021). Integration of data is required to maintain quality, particularly when data comes from multiple sources.

1.6.2 System Automation

Automation is the process of employing technology to undertake humanly performed jobs. Automation may be utilized within the framework of the Health and Demographic Surveillance System (HDSS) to enhance data quality, decrease mistakes, and boost efficiency. Data collection, data management, and data analysis are just a few of the HDSS areas where automation may be used. 1) Automated data collecting technologies, such as EDC or mobile data collection applications may be utilized to eliminate mistakes and enhance data gathering accuracy. 2) Automation may speed up and simplify the processes of data entering, data cleansing, and validation. 3) To evaluate big datasets and find trends and patterns that may not be immediately evident via human examination, automated data analysis methods like machine learning algorithms or statistical software may be utilized. 4) Through automated data visualization tools, dashboards, or reports, automation may be utilized to ease the distribution of data to stakeholders.

With HDSS automation, researchers may save time and money, make fewer mistakes, and improve the quality of their data. Automation may also release the workforce, which can be used toward other priorities like outreach and preparation for interventions. Nonetheless, it is essential to deploy automation in a manner that is sensitive to the local environment and takes into consideration the particular demands and preferences of the HDSS concerned. Automating data management processes in HDSS may provide several benefits: 1) Automating processes in HDSS may boost the efficacy of data gathering and management, decreasing the time and resources necessary to execute activities. This may result in cost savings and free up time for researchers to concentrate on other essential tasks. 2) The automation of systems may assist in eliminating mistakes and enhance the precision of data, resulting in better data quality. Automated methods may help guarantee the consistency and standardization of data collection and management, so lowering the likelihood of bias and enhancing the dependability of the findings. 3) Automating systems enables researchers to gather data in real-time, enabling them to react rapidly to new patterns and occurrences. This may be especially crucial during public health crises and disease epidemics. 4) Automated methods are readily scalable, enabling researchers to adjust to changes in project size or scope. This may be especially essential in the context of HDSS, which often includes the collection of data over an extended time period and a vast geographical region.

Automating systems can help researchers collect, manage, and analyse data more effectively, leading to better health outcomes for the population. Automation offers faster computation times, fewer mistakes, and enhanced database coordination and integration in accordance with system-wide standards (Häkkinen and Hilmola, 2008), from which to draw timely, relevant, and prospective data (Scapens and Jazayeri, 2003). Automation is a core feature of the information system and should thus be taken into account when conceptualizing the data quality (Knauer et al., 2020).

1.7 Motivation for the Research

HDSS is a reliable method for obtaining longitudinal data from well-defined enumeration zones. HDSS data assist governments in establishing priorities for health and demographic changes and for making forecasts and investment in education, health, and social well-being of the population (Kwei, 2006). Policymakers, academics, and governments need high quality data to identify and prioritise societal needs (Arikpo et al., 2019). High-quality data from HDSS may serve as a foundation for good quality decisions making, and high-quality research outputs. In HDSS, data quality is critical since it directly influences the correctness and dependability of the collected information. High-quality data is essential for informed decision-making, policy formulation, and effective public health interventions. Listed below are a few HDSS data quality drivers:

1. The HDSS is utilized to track illness trends and assess the efficacy of therapies. High-quality data ensures that the obtained information is accurate, resulting in improved decision-making and healthcare services for the community.

2. The HDSS data are used to assess the population's health requirements and distribute resources accordingly. Accurate and trustworthy data are crucial for efficient resource allocation, ensuring that funds are allocated to areas with the greatest need.

3. HDSS data are used to track the achievement of health-related goals and objectives. High-quality data ensures that progress is correctly monitored and that actions are modified as required to reach the desired goals.

4. The HDSS's high-quality data promotes data exchange and collaboration with other institutions and organizations, resulting in expanded knowledge and enhanced public health outcomes.

Ensuring data quality in HDSS is essential for enhancing healthcare, allocating resources effectively, monitoring progress, and facilitating collaboration amongst organisations and data sharing. Population health outcomes and interventions may be enhanced due to improved decision-making emanating from the data that is of good quality.

1.7.1 Problem Statement

Data has become the world's best resource, and its explosive generation has envisioned the operational efficiency and potential growth of a number of organisations. Organizations strategise on how to use their data to gain a competitive edge and increase revenue (McCausland, 2020). Data must be of high quality for organisations to thrive and gain a competitive advantage. Although good data may be a true business enabler, bad data can impede research, destroy or reduce competitiveness, and thwart innovation (Gowda, 2020). Data quality concerns had been on the research subject line for many years and were increasingly evident in the databases of the organisations (Strong et al., 1997). Strong et al. highlighted that 50% to 80% of records in the systems are ambiguous, incomplete, or inaccurate. The economic and social impact of data quality issues cost organisations billions of dollars (Ballou et al., 2003; Liepins and Uppuluri, 1990; Wang, 1995) and the United States (US) experienced a loss of about $3 trillion yearly (Redman, 2016).

Poor data quality in HDSSs is the number one opponent of machine learning's widespread and profitable adoption. Machine learning has high-quality requirements, and bad data can appear twice: once in the historical data for training the model and then in the new data deployed by that model for making future data-driven decisions. The model designed on poor-quality data is susceptible to disastrous misjudgments and may lead to bad tactical decision-making. Three potential problems arise from poor-quality data: failed projects, failed business processes, and bad decisions (Fürber, 2016). These problems lead to loss of revenue, high costs, poor product quality, and dissatisfaction among stakeholders. Much time and organisational resources are spent on identifying and correcting erroneous data, hence the sharply increasing operational cost. Other implications of poor-quality data are lower employee job satisfaction, poor performance, and less customer satisfaction (Pipino et al., 2002; Kahn et al., 2002).

Research interest in data quality has increased dramatically over the last decades (Ehrlinger and Wöß, 2017). Data analytics and data science projects received more attention in contrast to data quality assurance and data management, specifically in HDSSs. Proper data management and quality assurance are critical requirements for data analytics since quality data analysis is highly dependent on the quality of the underlying data. On the

other hand, the quality of the data depends on the quality of the underlying information system (Knauer et al., 2020). Although information systems are essential to HDSSs, several problems have been observed in sub-Saharan Africa, including insufficient use of the system's capabilities and problems with data quality that affect its consistency, timeliness, completeness, and correctness (Kimaro and Sahay, 2007; Sychareun et al., 2017; Teklegiorgis et al., 2016; Xiao et al., 2017). For HDSSs to be effective in their role of surveillance and monitoring of health-related concerns, the WHO (2015) accentuated the need for substantial investment in high-quality routine information systems. However, many information systems in resource-constrained contexts currently do not conform to WHO quality standards (Puttkammer et al., 2016; Glèlè Ahanhanzo et al., 2015). Thus compromising the quality of the underlying data. Organisational gains from improved data quality highlight the need to learn more about the elements that contribute to data quality (Knauer et al., 2020). To understand such elements that contribute to data quality problems, an automatic quality assessment and monitoring framework is needed (Reiche et al., 2014).

1.7.2 Aims and Objectives

This study's primary aim is to develop a system to improve and manage the data quality by integrating and automating workflows in rural-based demographic surveillance system.

The study seeks to achieve the following objectives:

1. Development of an electronic data collection system that incorporates automated data quality control algorithms to validate the data at the point of entry
2. Automating data export and processing to eliminate inherent manual and error-prone processes
3. Designing and deploying data quality framework to autonomously check and validate the quality of data immediately after synced to the database server
4. Development of a data reporting utility or dashboard to measure and monitor the quality of data in the database
5. Incorporating into the dashboard, the statistical techniques to identify and monitor processes that lead to data quality violations

1.8 Major Contributions and Implications of the Study

There is a steady emergence of innovative technology to address the issues faced by modern organisations. Thus, the adoption of new technology is seen as a prerequisite for competing in the modern economy (Peppard and Ward, 2004). Innovative technologies that make their way into an organisation's information system enhance data assemblage, storage, extraction, processing, and transfer, accordingly enhancing analytical proficiencies (Bhimani and Willocks, 2014). Our study uses modern technologies to solve data quality-related problems in the HDSS domain and contributes to the body of knowledge in several ways. First, the study incorporates a novel 3-Tier Total Data Quality Management Framework (3TTDQMF) into the proposed information system (data system) to identify, monitor, and manage the quality of data. The framework serves as an engine that takes as input, compromised data and runs corrective measures to produce the output of high quality. The study also integrated statistical quality control techniques (Pareto principle and Process control techniques) into the information system to identify and control the data quality issues in HDSS. At the time this research was conducted, no HDSS-related literature reported the use of Pareto and statistical process control approaches in improving the quality of data.

Second, our study uses open-source technologies to automate processes and integrate data from various sources to enable a seamless flow of data across different systems. The systems were configured to enable interoperability between different third-party applications to make possible the autonomy of the data management system. Lastly, the proposed work has crucial practical implications, which involve the deployment of a new automated system, replacing the obsolete systems within an existing entity. This was performed in an effort to improve the data quality and lay a solid foundation for other HDSSs that are willing to migrate from manual systems to electronic and automated systems. This is drawn from the fact that the vast majority of HDSSs are still using antiquated approaches, whereas sites with experience deploying modern systems have seldom shared their experience (Homan et al., 2015; Zeleke et al., 2021). The blueprints, models, algorithms, and configuration details shared in this work will provide the basis for the rapid deployment of a modern integrated, and automated information system that is suitable for HDSSs. The contributions of this study are classified as empirical, theoretical, and methodological. An empirical contribution results from an experiment performed in an existing organisation (HDSS) with the objective of enhancing both operations and data quality. This contribution expands the body of knowledge by elucidating the elements that affect data quality and the processes to consider in boosting data quality. From a theoretical standpoint, the study's findings corroborate various theoretical ideas presented in the past literature. Our research proposes a framework

that extends the notion of total data quality management (TDQM) by including three stages to improve data quality. Methodologically, this research adds to the existing literature in HDSS by explicating the methods used in integrating and automating systems to improve organisational performance and profitability. This work has important implications not only for HDSSs but also for other data-driven organizations that seek to manage and improve the quality of data. The findings of this work posit that an integrated and automated information system offers organizations the opportunity to manage and report on data quality more effectively and efficiently.

Integrating data from disparate sources eliminates difficulties in managing data distributed across various data sources, reduces data security risks, and maintains the integrity and consistency of the data. Automating data integration and other manual processes provides a plethora of benefits to the organization, as it reduces the costs associated with human resources, eliminates errors, improves performance, and improves productivity and competitiveness. HDSSs or data-driven organisations attract funding through the upkeep of high-quality data. The proposed framework incorporates mechanisms to allow organisations to effectively manage the quality of input data and produce data of better quality. The use of a dashboard enables organisations to continuously monitor and control the quality of their data. In addition, integrating statistical approaches into the dashboard ensures that factors influencing the quality of data are mitigated. In doing so, organisations are able to identify sources of issues affecting data quality and eliminate them for ultimate data quality.

1.9 Thesis Outline

Chapter 2 presents related works on data collection platforms, data quality assessment and management, information systems, process automation, and data analytics. Our research examines the literature in these domains to determine the degree to which these topics have been explored.

Chapter 3 generally provides the models, blueprints, and algorithms that allow researchers to gain a deeper understanding of the situation and plan the solution to the research problem. The chapter also presents some theories underpinning the study. Data quality measurements are also formulated and presented to provide the basis for data quality assessment.

Chapter 4 outlines the methodological and analytical procedures adopted to conduct the study. This study addressed data quality issues from a positivist point of view, and the researcher believes that the current problem could be addressed objectively. The chapter goes on at length to discuss techniques or specific procedures for identifying,

selecting, processing, and analyzing information about the proposed topic. This chapter provides an integrated view of the proposed information system by presenting data collection platforms, data stores, data integration and automation, data quality assurance processes, and algorithms.
Chapter 5 presents the interpretations of the results. The results are categorised into application interfaces, robotic process automation, application of Pareto principles and process control mechanisms, and data quality measurements. The application interfaces incorporate measures to validate data quality during capture, while automation eliminates inherent manual processes. Both the Pareto principles and the process control mechanisms apply a statistical approach to identify, monitor, and manage the quality of data. Data quality is measured by its accuracy, completeness, consistency, timeliness, and validity.
Chapter 6 provides the discussion and highlights the implications and limitations of the study while Chapter 7 presents the future work and conclusion.

1.10 Chapter Summary

Data from HDSS is essential for comprehending and resolving diverse health and demographic concerns in communities, especially in low- and middle-income nations. Data from the HDSS provide longitudinal information on populations, enabling policymakers and scientists to make observations on changes in health outcomes and vital events through time. This longitudinal component is essential to comprehending trends, patterns, and the influence of interventions. Good data quality offers the groundwork to accurately monitor the trends and patterns in health and social issues affecting the communities. This study utilizes a combination of techniques to comprehensively address challenges that contribute to poor data quality in the HDSS domain.

Chapter 2

State of the Art

2.1 Introduction

The literature review is an integral component of academic research that provides a detailed evaluation and synthesis of current theories, knowledge, and findings associated with a certain topic. This chapter will examine the published literature on data quality, data collection platforms, method of data collection, mode of data collection, automation, and business intelligent systems. The main objective is to investigate and assess the existing research works in order to get a greater grasp of the topic, uncover knowledge gaps, and lay the groundwork for our own research. By analysing prior studies critically, we hope to add to the existing body of knowledge, provide new insights, and resolve unaddressed problems.

Our literature evaluation is organized topically, combining studies and sources that address comparable themes or research objectives. This method permits us to study the current literature from many perspectives and to find trends, patterns, and controversies in the area. Our literature review is intended to be exhaustive; however, we acknowledge that it may have some limitations. There may have been limitations on the selection of sources, such as language limitations or restricted access to certain publications. Furthermore, the analysis and interpretation of the literature are susceptible to prejudice. Nevertheless, we have taken steps to reduce these limitations and ensure the validity and reliability of our findings.

The rest of the chapter is organised as follows: Section 2.2 presents data quality management approaches with emphasis on the data quality metrics and the methods of data quality control. The section further explores business intelligent systems with emphasis on the processes of building business intelligent systems as well as the four major components of business intelligent systems. Section 2.3 discusses data quality improvement

and data collection platforms. Section 2.4 discusses system automation with emphasis on the categories of automation, the definition and positioning of robotic process automation (RPA), and the major benefits of automation. Section 2.5 concludes the chapter.

2.2 Data Quality Management

2.2.1 Data Quality Assurance

The Enormous amount of data generated by organisations on a daily basis brings with it some data quality quandaries. According to (Moges et al., 2013), data quality problems increase as the data becomes complex, and that requires diligent data management processes. The study considered data management as a valuable activity, especially in the current information age where organizations generate enormous amounts of data. The enormous amount of data may result in poor quality data if not properly managed. Hence, having a proper system to manage data quality ensures understandable, usable, well-defined, secure, available, and consistent data (Couture, 2013). Dealing with poor-quality data can threaten the existence of any organization relying on data (Fisher and Kingma, 2001). Utilising poor-quality data tempers an organization's ability to properly plan and execute tasks (Redman, 2001). Wang and Strong (1996) highlighted the importance of data accuracy if decision-makers were to work effectively. According to (Marev, 2018), more research work in the field of data science focused on numeric data paying less attention to non-numeric data types. It is necessary to consider all types of data for quality management.

Authors (CorralesMúñoz et al., 2018) proposed data cleaning ontology to model data quality issues in classification tasks and data cleaning algorithms to solve data quality problems. The study evaluated the scope of data quality problems and recommended a data-cleaning algorithm. The authors created a data cleaning ontology using Semantic Web Rule Language (SWRL), which has been shown to improve the data quality. Data with high quality meet higher expectations compared to low-quality data (Sebastian-Coleman, 2012). Azeroual et al. (2018) presented possible methods of data profiling in information systems. According to the authors, data issues such as duplicates, incorrect formatting and contradictions, missing values, and data errors may be because of an increasing number of data sources and the growing volumes of data. The amount of data generated every single day is over 2.5 quintillion[1]. As the data grows, it becomes vital to focus on the quality in order to draw some meaningful insight from the data and

[1]https://www.domo.com/assets/downloads/18$_{d}omo_{d}ata - never - sleeps - 6 + verticals.pdf$

for better decision-making (Ramasamy and Chowdhury, 2020). Azeroual et al. (2018) highlighted that the quality of the data in the source directly influences the quality of the information system. In order to avoid content and structural data quality problems, the authors introduced data cleansing and analysis that took place in the process of data integration.

Data-driven and process-driven approaches are considered two major types of strategies to improve data quality (Batini et al., 2009). The data-driven approach improves data quality by directly changing the value of data whereas the process-driven approach involves redesigning of the processes that modify the data. Examples of some of the data-driven techniques considered were data normalisation or standardisation, error identification and correction, and data integration, which provided a combined view of the data from multiple sources. The process-driven techniques comprised process control to insert control actions and verification into the data production process; and the process redesign removed the major cause of data quality problems. According to Nikiforova (2020), data is considered to be of high quality if it is fit for its intended purpose in relation to planning, operation, analytics, and decision-making. Batini and Scannapieco (2016) also highlighted the importance of data in decision-making and operational processes. The only time data is fit for its intended use is when it is without defects (Redman, 2001; Wang and Strong, 1996). General processes of data cleaning are mostly time-consuming and difficult and must be performed before data analysis (Pyle, 1999). Gabernet and Limburn (2017) indicated that 80% of the valuable amount of time for data scientists was spent in cleansing and organizing data, leaving about 20% of the time for data analysis. That is particularly because the model that is based on data of poor quality is misleading and susceptible to inaccuracies. Having a good data quality framework that serves as a guideline for organisations to improve the quality of data is critical.

2.2.2 Data Quality Measurements

The essential feature of data quality management according to (Ramasamy and Chowdhury, 2020) is the assessment of data quality. Assessing the quality of real-time data is challenging and requires efficient data quality control mechanisms. Authors identified data quality metrics such as completeness, accuracy, timeliness, and consistency as widely used (Table 2.1). The studies in Table 2.1 discussed key measures of data quality. Depending on the context, studies explored different data quality dimensions. Borek et al. (2013) stated that data quality is a multidimensional concept and strongly depends on the user and the usage context. The authors recommended that every company

should select the most suitable data quality dimensions as per its specific goals. The selection of data quality dimensions should be done based on organisational requirements, levels of risk, and business context, amongst other things (DAMA, 2013).

Table 2.1: Matrix table referencing data quality metrics

	Quality Metric			
Reference	Completeness	Accuracy	Timeliness	Consistency
(Ehrlinger et al., 2019)	✓	✓	✓	✓
(Sánchez et al., 2019)	✓	✓	-	-
(Marev, 2018)	✓	✓	✓	✓
(Moyano et al., 2017)	✓	-	-	-
(Batini and Scannapieco, 2016)	✓	✓	-	-
(Cai et al., 2015)	-	✓	✓	✓
(Neumaier, 2015)	✓	✓	-	-
(Todoran, 2015)	✓	✓	✓	✓
(Umbrich et al., 2014)	✓	✓	-	-
(Dama, 2013)	✓	-	✓	✓
(Rahman, 2013)	-	✓	-	-
(Weiskopf et al., 2013)	✓	✓	✓	✓
(Sattler, 2009)	✓	✓	✓	✓
(Heinrich et al., 2007)	-	✓	✓	-
(De Amicis, 2006)	✓	✓	✓	-
(Scannapieco, 2006)	✓	✓	✓	✓
(Jarke et al.,2002)	✓	✓	✓	✓
(Leo And Pipino, 2002)	✓	-	✓	-
(Redman, 2001)	✓	✓	-	✓
(Loshin, 2001)	✓	✓	✓	✓

Ehrlinger et al. (2019) performed a systematic review and identified 667 software tools used for data quality measurements. The study considered the selection of thirteen (13) data quality tools for deeper investigation with considerations on data profiling, data quality measurements, and continuous data quality monitoring. The authors proposed a survey on a comprehensive overview of state-of-the-art data quality tools with functionalities to monitor and measure data quality. The study seeks to answer a question; "how can data quality be monitored and measured automatically". The findings of the study were relevant to data quality professionals in selecting a suitable tool and bringing to light the capabilities of the current state-of-the-art data quality tools. Continuous data quality monitoring is required to ensure data quality over time. Ehrlinger et al. (2019) and Sebastian-Coleman (2012) suggested the need for more research to investigate a question: "how could data quality be measured and monitored for data quality improvement". It is argued that the question was insufficiently answered in the literature.

2.2.3 Data Quality Management Framework

A data quality framework provides companies with a set of tools and methods to assess data quality levels and prioritise relevant service data (Camera et al., 2020). According to (Corrales Múñoz et al., 2018), Data Quality Frameworks (DQF) are designed to assess, analyse and use clean data with good quality in the data management system. DQF must present a general structure to analyse and solve data quality issues (Eppler and Wittig, 2000). In their proposed study (Camera et al., 2020), the data cleansing processes involved mapping the most important data available within the company and its external and internal flows and applying multiple criteria for decision-making techniques. The maturity model assessed the quality of the prioritised dataset and defined the strategy to address data quality issues. The findings have shown the potential benefits of the proposed process and the industrial case study validated the results. It was argued that, despite the presence of extensive literature on data quality and data management, very few published researches addressed data quality issues and data management in the context of industrial maintenance or Through-life engineering services. We share the same sentiment but in a different context, that, much was not published in addressing data management and data quality issues in HDSS tasked with collecting longitudinal data from remote and under-resourced rural areas. There is a need for strategies to improve the quality of data in HDSS. A decision support framework was developed to allow decision-makers to assess the quality of data in both context-dependent and context-independent manner (Shankaranarayanan and Cai, 2006). The framework used an Information Product Map (IPMAP) and evaluated data quality using a completeness metric. It is not clear how the framework will perform when evaluated on other data quality metrics excluded from the study. Heinrich et al. (2007) discussed how metrics for selected data quality attributes could be designed to measure the quality of data. The aim of the study was to enable the measurement and the analysis of economic consequences of data quality attributes in campaign management. Authors argued that their metrics were relevant and measured data quality more efficiently and accurately estimated the economic benefits. Moreover, the benefits of data in decision-making depend heavily on the correctness, timeliness, and completeness of such data. Many organisations according to (Strong et al., 1997) have problems in ensuring high data quality. Redman (1998) indicated that the total cost due to low data quality was between 8% and 12% of loss in revenues. Insufficient data quality has a devastating impact and according to (Meta Group, 1999), 41% of projects in data warehouses fail due to data quality issues. Adequate measurements are required for the growing relevance of data quality and the quantification of data quality is crucial to measure data quality in an economic manner (Heinrich et al., 2007). With relevant tools, the quality of data may be measured effectively and in an economic manner.

2.2.4 Information System for Data Quality Control

The significance of information systems in data quality control is critical. By enabling the discovery and resolution of data quality issues and offering insights into data-related activities, information systems help organisations to enhance the quality of their data (Papiorek and Hiebl, 2023; Edris et al., 2022; Liu et al., 2020). Organisations have the ability to evaluate the quality of their data, detect problems such as duplicate entries, missing values, improper formatting, and inconsistencies, and apply methods such as data profiling, standardization, and data cleansing to improve data quality (Azeroual and Abuosba, 2019). In addition to facilitating the extraction of knowledge from data quality outcomes, information systems enable researchers to develop logical frameworks and get a thorough grasp of their data. Moreover, information systems play a crucial role in streamlining resource allocation for data quality management, hence reducing the potential hazards that may arise from substandard data affecting company activities and decision-making.

Sánchez et al. (2019) proposed a web-based system that provide a better understanding of the structure, distribution, and content of the dataset. The system provided improvement utilities and data visualization for the data quality attributes such as accuracy, completeness, readability, and relevancy. The study reported 53.39% of missing values. The system enabled users to assess and improve data quality. Lower levels of data quality have a detrimental impact on the overall efficiency of data applications (Fu and Easton, 2017). The quality of the system may positively influence the quality of data. A system's quality can be identified by features such as user-friendliness, meeting user requirements, ease of learning and navigation, flexibility, integration, customization, and accuracy (DeLone and McLean, 2003; Sedera et al., 2004). How the system was designed determines the quality of data (Krippendorff, 2009). Gorla (2010) argues that there is a constant relationship between system and data quality. A system that employs recent technologies, integrated, conveniently, and properly handles user activities, may improve data quality and decision-making. According to Törn (1990), poor system quality leads to poor data quality. Alrayes (2015) suggested a need for an effective information system to improve the quality of data.

2.2.5 Data Integration and Visualisation

Due variety of business intelligent solutions present on the market; organisations must choose a solution that best suits their needs in relation to costs and benefits (Ghilic-Micu, 2021). TÂRNĂVEANU (2012) estimated that more than 60% of organisations and governments opted for open-source platforms to achieve their objective. The open-source

business intelligence solutions were considered the best option given budget constraints and sprawling requirements. According to (Musa et al., 2018), Business Intelligent Enterprise Edition, free software, Pentaho business intelligent, Oracle, SAP, BusinessObjects BI, IBM and business overview, Cognos, and Microsoft are some of the tools (open-sources and enterprise editions) available for developing business intelligent systems. The enterprise edition platforms offer several advantages and disadvantages compared to open-source platforms or tools. The business intelligent software is composed of a range of other tools to assist and support decision-making. Moreover, according to (Runtuwene et al., 2018), the entire business intelligence management process consists of three steps: Data Warehouse and Data Mart, Online Analytical Processing (OLAP) as well as Data Mining. In addition, (Butuza et al., 2011) viewed the business intelligence system as consisting of four major components. Those components were data warehouse (database derived from integrating data from different sources), business analytics (consisting of tools for data manipulation, analyses, and mining), business performance management (for evaluation and monitoring the performance), and user interface (allows the connection between the browser and system engine).

DivyaYadav and Choudhary (2021) considered online OLAP as a prerequisite for the creation of a decision support system (DSS). The authors briefly discussed some variations of OLAP extensions such as Relational Online Analytical Processing (ROLAP), Multidimensional Online Analytical Processing (MOLAP), and Hybrid Online Analytical Processing (HOLAP), and highlighted how they fit into BI systems. In their study, authors used ETL processes and IN-MEMORY OLAP to process and manipulate data in the database into easy to analyse star schema. Power BI was used for ETL processes and the design of star schema-based fact and dimension tables. That resulted in the development of easy-to-understand reports and dashboards. Eckerson (2006) defines a dashboard as "a multilayer application built on a business intelligence and data integration infrastructure that enables organizations to measure, monitor, and manage business performance more effectively". Dashboards are useful in that they assist in monitoring daily operations (Pestana et al., 2020) and allow the team to make well-informed decisions (Al-Hajj et al., 2013). A properly designed and built dashboard could, according to (Ward et al., 2014), improve the efficiency of any institution or organisation.

Most of the studies considered the Pentaho software suite to analyse, visualize, extract, transform, and load data into data warehouses. According to Pentaho Corporation [2], the software provided a wide range of business intelligence capabilities including dashboards, query processing and reporting, data mining, data integration, interactive analysis, and business intelligence platform. Pentaho business analytics allows rapid development

[2]http://www.pentaho.com/

and deployment of a secure, scalable, flexible, and easy-to-manage data analytic system. TÂRNĂVEANU (2012) implemented a practical solution using the Pentaho software suite to provide data analytics and integration, data mining, OLAP services, dashboard services, reporting, and ETL features. The study regarded open-source platforms as the best alternative to enterprise platforms within a business intelligent system context. In another study, in order to accomplish their main objectives, authors (Tuncer and van den Berg, 2010) put together Pentaho ETL (for data extraction, transformation, and loading. This is known to provide intuitive, graphical, and drag-and-drop features), Mondrian OLAP engine (for enabling an interactive analysis of data from SQL database and creation of multidimensional complex analysis queries) and Pentaho dashboard model (for the purposes of visualisation). The open-source database platform (MySQL) was selected for the design and development of data warehouses and data marts. The ETL processes were designed to extract data from different sources, transforming the data to meet its operational needs and loading it into a target database or data warehouse (Kakish and Kraft, 2012). Lokaadinugroho et al. (2021) used nine (9) steps of Kimball's data warehouse and ETL process in building a Tableau-based business intelligence system. The study considered Microsoft SQL Server 2014, Pentaho Data Integration (PDI), Tableau 10, and other supporting tools in designing data warehouses and developing BI systems. The data was extracted from four data sources, transformed, and loaded into the data warehouse for analysis. The integrated data system reduced the time needed to produce some complex reports and according to authors, it previously took 2-3 weeks to generate such reports. The proposed BI system allowed complex reports to be generated within 77 minutes and the data was easier to understand. The decision-making became more faster and competitive advantages were gained.

Marín-Ortega et al. (2014) presented a new approach to designing business intelligent systems. The approach aimed at reducing time spent in designing business intelligent solutions and preparation of BI solutions to be used with big data. The authors extended the ELT (Extract, Load, and Transform) concept to ELTA (Extract, Load, Transform, and Analyse). According to the authors, data management was the foundation of the BI system and the most stressful and time-consuming activity. Data management systems with SQL were significantly faster according to (Pavlo et al., 2009; Chen and Hsu, 2013), and required less code for extracting information and analytical tasks. According to Wang and Liu (2009), BI system must have the features such as data management (includes data cleaning, data extraction, data integration, storage and management of enormous amounts of data), data analysis (includes report generation, information queries, and data visualization functions) and knowledge discovery (the extraction of knowledge from rapidly growing and changing volumes of data in the data warehouse).

Authors (Mussa et al., 2018) presented the application of Pentaho in an educational institution. The transformation, extraction, and loading processes were performed using Pentaho data integration (PDI). PDI provided necessary support in transforming and loading data into SQL server-based dimension tables. The major difference between Pentaho and other BI tools was its ability to offer customisation options. The transformation of an operational database into a data warehouse could be useful for decision-makers to perform data analysis, forecasting, and predictions (Bassil, 2012). In addition to that, (Miranda, 2015) stated that data warehouses can be used to monitor organisational performance and support the processes of making decisions. According to (Mircea et al., 2011), it is necessary to build an agile business solution incorporating modern technologies such as cloud computing and master data management, business rules, business process management, and service-orientated architectures. Authors (Maury et al., 2021) built an interactive dashboard using Shiny based on the R programming language implemented on RStudio Interactive Development Environment (IDE). JavaScript and CSS were used for interactivity and, the look and feel of the user interfaces. The dashboard relied on a number of packages: shinydash for page structure, ggplot2 and plotly for interactive charts and shinyjs for custom interaction. Other packages such as dplyr, plyr, lubridate, tidyverse, tidyr, psych, stringr and forcats were used for data wrangling. The dashboard greatly improved the functioning of the laboratory in operations management and quality. Real-time interpretation of key performance indicators (KPI) has led to good decision-making and reactivity.

2.3 Data Quality Improvement and Collection Platforms

2.3.1 Open Data Kit (ODK) Data Collection Plaform

ODK was widely used by the research community for programming and hosting electronic data collection (EDC) systems (Kenny et al., 2020; Thindwa et al., 2020; Maleghemi et al., 2019; Ahmed et al., 2018; Flaxman et al., 2018; Sheikhali et al., 2016; Tom-Aba et al., 2015). The platform has a built-in module to allow offline and online data collection. The authors argued that ODK is the most suitable platform to collect general and health-related data in developing countries with poor network infrastructure. According to Maleghemi et al. (2019), ODK provided an out-of-box solution to allow users to design and create data collection forms. It facilitated and strengthened South Sudan's Acute flaccid paralysis surveillance performance. The study reported that ODK bridged the gaps in the mode of data collection (online and offline), data quality, and accountability. Kenny et al. (2020) used an ODK-based Java EDC solution deployed to Android-based devices. The system was designed for remote regions with minimal or

no network connectivity. The Bluetooth was used to transfer the acquired data to/from community health workers and their supervisors. Kenny et al. considered offline data collection to be the most cost-effective alternative to online data collection. However, the non-existence of a pre-loading capability to facilitate speedier data collecting and significant delays in transmitting data from data collectors to the database were cited as significant limitations. It was possible for users other than data collectors to alter data, which violated data integrity standards.

The EDC system built using Extensible Mark-up Language (XML) was hosted on an ODK-aggregate server for real-time data collection (Sheikhali et al., 2016). Skipping logic algorithms enhanced the overall quality of data. A similar approach was considered by (Njuguna et al., 2014; Thriemer et al., 2012; Were et al., 2010; Yu et al., 2009). Data anonymization and aggregation were done using a custom-designed application based on HTML and JavaScript, Python application programming interface, and PostgreSQL database. Innovative principles of health surveillance were introduced and included amongst other things; cloud-based data collection, management, and reporting. The study further recommended the comparison of cloud-based data collection surveillance and traditional public health surveillance with an emphasis on cost-effectiveness, completeness, accuracy, and data sharing.

Thindwa et al. (2020) designed an EDC tool using ODK and data synced to the MySQL database. The system integrated validation rules to validate data quality with SQL queries guarding against errors and inconsistencies undetected by validation rules. The study designed a system to overcome data quality complexities and evaluated the cost of EDC by measuring accuracy, timeliness, efficiency, volume, and material cost. The synchronization of data from the client to the server was performed at the office due to secured network connectivity. The approach was also recommended in other studies (Meyer et al., 2013; Baguiya, 2016). The study achieved far fewer errors (approximately 0.2% which is about 21.7 errors per 1,000 data points) in data collected compared to (Medhanyie et al., 2017; Thriemer et al., 2012; Yu et al.,2009; King et al., 2013; Dillon et al., 2014; Patnaik et al., 2009). The findings revealed that EDC has the potential to reduce error rates and improve data quality as opposed to traditional paper-based data collection.

The study to manage the Ebola outbreak was conducted in Nigeria aiming at effectively tracing contacts (Tom-Aba et al., 2015). The data collected using ODK Collect was synced to form a hub server in real time. The Cron Job script extracted data from the server and displayed Ebola symptomatic patients with temperatures above 27 degrees Celsius on the dashboard. ArcGIS software mapped the location of those individuals

who were in contact with Ebola patients. The system allowed data for those who had contacts with Ebola patients to be sent to the central server in real time. The turnaround time between identifying a contact and isolating such contact was greatly reduced. Complete and timely information was readily available at the time the decisions needed to be made. The system also allowed data to be entered simultaneously into the database and generated daily reports from different team members. Authors reported that the initial cost of setting up the system for EDC was very high and this was in line with the findings from other studies (Flaxman et al., 2018; Njuguna et al., 2014; Weber et al., 2005). Ahmed et al. (2018) carried out a study investigating the efficiency of EDC and PDC in health research surveillance in Sudan. Authors designed and developed EDC in Microsoft Excel and converted XLS file into XLSform, loaded and hosted it in ODK aggregate server. Data was collected and stored locally on Android-based smartphones and synced to the server pending the availability of a wireless connection. To enhance data quality, especially in cases where questionnaires were too long with many nested rosters and questions, complex skipping patterns and validation rules were implemented. It was found that EDC provided high-quality data with fewer errors (17%) and was less inconsistent compared to PDC, which recorded 83% of errors. The findings were in concord with other studies (Thriemer et al., 2012; Njuguna et al., 2014; Le Jeannic et al., 2014; Isara et al., 2013; Dillon et al., 2014). One major limitation of the study was that fewer participants were considered and no idea was provided on how the system will perform given a larger sample. Ahmed et al. also pointed out that ODK-based applications would close occasionally and unpredictably, in which case the data collector would have to restart the application and that would sometimes result in loss of data.

Flaxman et al. (2018) carried out Verbal Autopsy (VA) surveillance in real time to collect data in Bangladesh and the Philippines. The authors designed the EDC tool and hosted it in the ODK aggregate server. The study measured the time and cost required to process surveys for analysis. It took approximately 3 months to process data (from data collection to entering data into the central database) using PDC. On the other hand, the same process took less than two days for EDC. The upfront cost of setting up EDC was higher compared to that of PDC. However, EDC became cheaper over a period of six years compared to PDC due to various factors; tablets and other tools were bought once and some data capturers were no longer necessary for EDC.

2.3.2 Survey Solutions® Data Collection Platform

By providing a quality service platform for storing and exploiting quality data, Survey Solutions® helps to boost data quality in the surveying and mapping industries (Ge

et al., 2020). By resolving issues with surveying and mapping quality data management and the requirement for quality data utilization, the platform aids in scientific decision-making and enhances service levels (Zhang et al., 2018). The benefits of Survey solutions® for data collection are discussed in the articles by Schoenherr et al. (2015) and Miswar et al. (2018). The benefits and risks of using Survey Solutions® for empirical data collection were emphasised by Schoenherr et al., especially for the supply chain management. While it is a viable alternative to self-administered surveys, they advocate using it with caution. A comprehensive literature study was undertaken by Miswar et al in order to comprehend the technologies, characteristics, and procedures utilized in data-gathering systems. Several approaches, such as Computer-Assisted Personal Interviewing (CAPI), have been applied extensively in previous research. These publications provide essential insights and suggestions to academics researching the use of survey technologies for data collection.

Open-data Kit (ODK) and Survey Solutions® are both systems for data collecting, however, not identical. Survey Solutions® is a solution for questionnaire rendering that extends ODK's toolbox to handle complicated and highly adaptive workflows (Brunette et al., 2013). It features interactive, non-linear navigation and JavaScript/HTML-defined question widgets, enabling runtime customization of navigation and question data types (Farooq et al., 2023). ODK is a successful data-gathering tool that adheres to the JavaRosa XForm standard (Boros, 2020). While both systems promise to enhance an organization's capacity to create customized domain-specific applications, Survey Solutions® provides more extensive capabilities for complicated workflows (Lin et al., 2017). On the other hand, ODK is renowned for its simplicity and usability (Horne et al., 2020). Overall, Survey Solutions® extends ODK's capabilities and offers extra tools for developing bespoke apps with intricate workflows. In addition, Survey Solutions® generates paradata for survey data analysis.

Using paradata in Survey Solutions® offers a variety of benefits. To make better decisions for future surveys (Hasanbasri et al., 2023), the platform's paradata can reveal objective insights on the respondent burden, survey expenses, and interviewer impacts. In terms of quality control, it may be used to evaluate the accuracy of methodological guidelines and to spot and deter interviewers from engaging in falsification or fabrication (Daniil, 2022). Paradata may also be used to detect enumerators engaging in anomalous field practices and provide them with feedback to help them improve (Goel et al., 2022), therefore decreasing interviewer-induced measurement error. Paradata may also be used to assess and enhance questionnaire development, facilitating data-driven redesign of existing surveys (Rahman and Sjöström, 2021). The use of paradata in Survey Solutions® may increase respondent satisfaction, shorten interview times, and better quantify and

characterize digital data (Gordeev et al., 2021). Moreover, paradata analysis in business surveys can increase data quality and reveal areas for operational efficiency (Claveau et al., 2010).

2.3.3 Redcap Data Collection Platform

REDCap was created at Vanderbilt University to streamline the process of collecting clinical data online. It enables the development of databases and surveys, including validations and branching logic for enhanced data quality. After data gathering is complete, the information may be sent to statistical software (Morrison, 2013). The technology eliminates the need for subjective human judgment to evaluate data quality, making it more secure and less likely that sensitive information would be compromised (Ke et al., 2018). A document reveals a system and technique for collecting and aggregating data from members of a community, with procedures for identifying data incompleteness and incorrectness and the capacity to automatically update or complete the data (Kain et al., 2020). Qualitative data gathered online can offer the same level of depth and richness as that collected by pen and paper, according to research comparing the two (Horr et al., 2018). In order to improve data quality and administration capability, (Wang and Liu, 2017) researchers have developed a data resource management platform based on Internet data collecting.

2.3.4 Electronic Versus Paper-Based Data Collection Methods

PDC has the potential to introduce human error due to multiple data entries (with the first point of data entry being in the field and the use of computer software to capture data into the database) (Weber et al., 2005). To alleviate these problems, the authors used Hyper-Text Mark-up Language (HTML) to create web-based forms using Adobe Accelio Capture Designer to collect data in real time. SQL scripts were designed and run on the server to validate entered data to ensure accuracy and minimise errors. To simplify form completion and improve accuracy, custom macros, database lookups, and derived calculations were implemented. Secure socket layer technology encrypted data in transit and servers secured through password authentication. Cost, accuracy, and efficiency were performance metrics selected to measure the performance of EDC and PDC. The processes of data collection and management took less time with fewer errors in EDC than in PDC. According to Weber et al., the initial cost of setting up EDC was higher than PDC, however, as the number of participants increased over a period of time, an exponential cost increase was evident in PDC while the cost of EDC dropped. In addition, the initial cost of EDC can substantially drop to the lowest within a few

years of data collection (Njuguna et al., 2014). King et al. (2013) also found that the cost of collecting data in EDC was lower than that of PDC and the accuracy of data in electronic forms was better with fewer errors. On the contrary, the cost of setting up the EDC system was higher and Zhang et al. (2012) reckoned that it was due to the survey being smaller. Other studies also confirmed the cost-effectiveness of EDC for larger surveys (Weber et al., 2005; Thriemer et al., 2012; Koop and Mosges, 2002; Madder et al., 2012; Karimuribo et al., 2012). However, there is a lack of systematically analysed evidence to support the claim that EDC is cost-efficient and improves quality in interviewer-administered surveys (Zeleke et al., 2019). EDC tools "can be programmed to provide determinate responses, date stamped to document times of data entry, restrict times of data entry, prevent retroactive data entry, limit 'look back' to previous data, prevent omissions of data entry, and can save considerable time and labour incurred in data handling" (Lane et al., 2006). Lane et al. revealed that EDC was more efficient than PDC in terms of timeliness of data capturing, data accuracy, and adherence to data collection protocols. Authors also reported that EDC reduced time for collecting, handling, and transferring data by 23% PDC.

Hospital-based sentinel surveillance was carried out by Njuguna et al. (2014) in Kenya where PDC was replaced with EDC. The authors used the Field Adapted Survey Toolkit (FAST) for programming EDC. Data was collected, saved to the smartphone, and sent to the server using the internal 3G connection. Demographic, epidemiological, and administrative variables were used for quality assurance through the implementation of programmed validation checks, range-value restrictions, and skipping patterns. The tool was evaluated on data timeliness, data quality, and cost. The incidence of errors and inconsistent answers was higher for PDC with about 8.3% than 0.4% for EDC. Data uploaded into the database using EDC took a median time of about 7 days, which was less compared to the median duration of 21 days obtained using PDC. Data took less time to be available for analysis in EDC (Shirima et al., 2007; Lal et al., 2000). However, a number of challenges were encountered by Njuguna et al. (2014). For instance, an occasional server communication breakdown, which led to a delayed upload of the data into the server and the termination of the production of FAST platforms meant that no new updates were available. FAST was unable to handle complex or large surveillance.

King et al. (2013) developed an Android-based application for an offline data collection and data management system. The authors compared EDC to PDC by evaluating data accuracy, cost of data collection, and error rate. The study has reported that an Android-based technology improved the accuracy of global positioning system (GPS) coordinates and saved a considerable amount of time in data collection. Moreover, most

importantly, EDC was more appealing and user-friendly to the data collectors in contrary to PDC. The offline mode of data collection was considered on the basis of security and adherence to local data management guidelines. This approach dominated the literature (Thindwa et al., 2020; Kenny et al., 2020; Ahmed et al., 2018; Njuguna et al., 2014; Thriemer et al., 2012) and was found suitable for data collection in rural settings.

Setting up a real-time EDC system for health surveillance in Nepal required open-source software called AKVO (Ley et al., 2019). The authors investigated the efficiency of EDC in collecting high-quality data compared to traditional methods of data collection. A check for completeness, internal logic, values out of realistic range, and date variables within reasonable periods were considered the most effective evaluation metrics for a data collection tool. The variables were grouped into five categories and the numbers of discordant data entries were compared between EDC and PDC. The results obtained showed that the omissions in PDC were twice as frequent as in EDC. The study found that EDC is the most efficient alternative to PDC and the findings aligned well with that of (Flaxman et al., 2018; Ahmed et al., 2018; Njuguna et al., 2014). The study further set up an online data collection system to allow data to be synced in real-time. Since real-time data collection heavily depends on the network, the study showed no evidence of how the system performs under poor network connectivity, especially in rural settings. Electronic devices such as tablet computers and smartphones can greatly reduce costs and improve data quality in public health surveillance (Zeleke et al., 2019). The authors conducted the study and aimed to inform technologists, policymakers, researchers, and other stakeholders about the impact of EDC systems on cost reduction, data quality, and work efficiency. Studies in literature from 2007 to mid-2018 were reviewed with a focus on data quality, cost efficiency, usability, and user experience. The analysis of results showed that EDC yielded good quality data compared to PDC.

Like most of the other tools for creating EDC solutions, the study (Seebregts et al., 2009; Yu et al., 2009; Were et al. 2010) used Pendragon forms which has a built-in designer for creating customized data collection forms or questionnaires. The system had a built-in scripting capability for data validation, branching logic, form navigation control, and calculations. The data was synced to a Microsoft-based database and the system had the functionality to export data to Excel. From End-user's perspective, EDC was faster and easier to work with and generally produced higher-quality data. Another study (Lane et al., 2006) also reported that End-users generally preferred EDC over PDC. EDC system introduced fewer errors, with more complete data, and required less time for data cleaning (Galliher et al., 2008; Were et al., 2010). Pendragon forms had no built-in GPS capabilities, as a result, GPS coordinates were captured manually into the EDC tool.

The study by Thriemer et al. (2012) assessed the acceptability and feasibility of EDC and PDC in resource-limited settings. An EDC system was programmed in Visual Basic.Net and included skipping patterns and validation rules to enhance data quality. The data management tool was developed using Microsoft FoxPro. The data was stored on the memory card during data collection and later loaded into the central database. Authors found that directly capturing data in EDC was faster and 25% cheaper than PDC methods. Data collected through EDC had a lower error rate with no omissions compared to PDC. The study further revealed that PDC had longer data turnaround times with errors detected very late and that made error corrections very difficult. Zhang et al. (2012) compared PDC and EDC for household surveillance in rural China. The interrater reliability differences between the two methods were analysed using a chi-square test and the average time duration was compared using an independent-random t-test. The tests proved that EDC was an attractive method of data collection with regard to data quality, time consumption, and interviewer perception. Data collected using EDC was readily available immediately after collection from the field for review, status evaluation, and real-time analysis (Isara et al., 2013). Isara et al. designed and developed questionnaires using Episurveyor and SPSS statistical software was used for analysis. The authors used Kappa statistics to test agreement in two data collection methods (EDC and PDC) and found that there was no significant difference between the two methods. On the other hand, the tests to examine if there was a correlation and significant difference between EDC and PDC were conducted by (Carlbring et al., 2007). The correlations were calculated using Pearson correlations (Two-tailed) and the significance test of the differences in the two questionnaire types was done using 2X2 mixed ANOVA. Unlike the study by Isara et al., the findings of this study showed high and significant correlations between the two data collection methods. The study (Carlbring et al.) did not clearly show which method was the best while Isara et al. recommended EDC for larger surveys.

To determine which methods of data collection (PDC and EDC) were more effective, Yu et al. (2009) conducted a study on public health surveillance in Fiji. The study followed an object-oriented component-based approach in developing an EDC tool. The authors highlighted that EDC eliminated some steps performed in PDC; for example, data acquisition and capturing into the central database was done simultaneously in EDC. Statistical analysis for error rates was made between the two data collection methods and a p-value of less than 0.05 indicated a significant difference. The study reported 20.8% errors in the categories of logical range errors, skipping errors, missing values, and incorrect data types in PDC questionnaires. There were no errors reported for EDC because of properly implemented validation rules and skipping patterns. It took less

time to clean EDC data than PDC because of fewer problems in the collected data. About 71.4% of data collectors considered EDC as the best method of data collection. Healthcare providers favoured EDC over PDC (Ammenwerth et al., 2000).

Pentagon Forms was used as a platform for the development of EDC tools for household surveillance in Southern Tanzania (Shirima et al., 2007). The platform consisted of components to design the EDC tool: the synchronization conduit to transfer data to/from the centralised database (Microsoft access) and an application running on the data collection device for data entry. The data transfer between the data collection device and the laptop computer was done using a USB cable and data synchronization software. The system allowed scripts containing logical and validation statements to be attached to the questions and evaluated when data was captured. The completeness of data reported was over 99%. The proposed methods eliminated problems inherent in PDC such as error-prone processes, time-consuming methods, and cumbersomeness. Tomlinson et al. (2009) investigated the ease of implementation and feasibility of EDC in poor research infrastructure settings. The authors developed a web-based data collection tool using the ASP.Net framework. The system incorporated built-in reports and real-time data visualization for survey responses. It allowed CSV or Excel files to be exported for data analysis. Data quality checks were done in real-time. Errors and inconsistencies were identified and cleaned in a timely manner. Unlike the study by Shirima et al., Tomlinson et al. ensured the security and confidentiality of data in transit using a 128-bit encryption mechanism. In addition, the servers were configured behind a firewall to ensure total protection and prevention of unauthorized access.

2.4 Robotic Process Automation and Data Integration

2.4.1 Automation: Lightweight IT and Heavyweight IT

Information technology (IT) provides numerous options for enhancing tasks through process automation. Bygstad (2016) outlined two primary approaches that process automation may be used to enhance tasks, which were heavyweight IT and lightweight IT. Heavyweight IT is the knowledge regime involved with the creation and management of complex systems. Lightweight IT, on the other hand, is characterised as a new knowledge practice that is strongly tied to the consumerisation of IT. The traditional back-end system automation is directly linked to Heavyweight IT and requires the integration of the Application Programming Interface (API) (Bygstad, 2016; Lacity and Willcocks, 2016). In this approach, more advanced solutions are sometimes created through intricate integration. Heavyweight IT is characterized by expanding size and integration,

which often results in rising costs and complexity (Sommerville et al., 2012). Software engineering and the use of technologies like enterprise service bus (ESB), service-oriented architecture (SOA), and enterprise resource planning (ERP) software are common practices for achieving successful back-end integration (Penttinen et al., 2018). Penttinen et al. (2018) define back-end system automation as invasive automation, implemented by means of system development and/or data or application-layer system integration. Penttinen et al. (2018) describe back-end system automation as intrusive automation accomplished through "system development and/or data or application layer system integration". Back-end system automation encompasses the following approaches in the implementation of process automation: 1) Middleware solution, 2) Business Process Management (BPM) solution with business process automation (BPA) extension, and 3) Extension to the current system (Mohapatra, 2009). The word "middleware" refers to software used to link two or more independent applications. They are a central element of IT infrastructures, as they allow the joining of heterogeneous systems together in one framework. Due to middleware' or APIs' ability to integrate disparate, non-interoperable systems into a unified whole, they play a crucial role in information technology infrastructures. Enterprise Application Integration (EAI) is an example of middleware that provides a framework for integrating many technologies into a unified whole (Sabooniha et al., 2012). Serrano et al. (2014) provided a comparative analysis of EAI solutions such as Integration suite, ESB, and Integration framework. The selection of the optimal solution is significantly influenced by the system's requirements and complexity. An alternative to the Middleware solution is a BPM solution with BPA extension. BPM solution with BPA extension offers a basis for mapping all business processes; this slows down the automation of specific operations. Extension of the present system enables the fulfilment of new business requirements. Extending the present system may be adequate when process automation is accomplished utilizing a single system.

In contrast to Heavyweight IT, applications that assist business operations without altering the existing IT infrastructure are indicative of lightweight IT. Lightweight IT is developed mostly by users and providers who are not experts in IT but who can construct basic, task-specific applications. A lightweight IT environment may foster new innovative ideas (Bygstad, 2016). However, privacy and security concerns might be more difficult to manage with lightweight technology, and lightweight solutions can lead to disconnected apps and devices. Robotic process automation (RPA) is a prototypical form of lightweight IT since it mostly functions on the presentation layer and the pre-existing functionality of applications (Lacity and Willcocks, 2016; Primer, 2015). Since RPA uses a presentation layer to interact with systems, no changes to the underlying programming logic of such systems are required. RPA is further discussed in the sections below.

Penttinen et al. (2018) compare Heavyweight IT and Lightweight IT in automating processes in the telecom industry. The decision on whether to implement Heavyweight IT or Lightweight IT depends on the stability of the environment, the need to access multiple environments, and the high volume of transactions (Fung, 2014). Additional research should examine automation initiatives across sectors to see if the selection is influenced by various variables in different circumstances. Alternatively, explore instances in which Lightweight IT automation was considered, and rejected in favour of Heavyweight IT automation and compare selection factors to other works in literature.

2.4.2 Robotic Process Automation (RPA) Definition and Positioning

RPA is sometimes referred to as an instance of software (Leopold et al., 2018; Gejke, 2018; Ratia et al., 2018) or the software configuration (Lacity et al., 2016) that automatically performs certain tasks (Leopold et al., 2018). According to Van der Aalst et al. (2018), RPA is a catchall term for software that mimics human interaction with a graphical user interface (GUI). Various techniques are considered for a robot configuration, which includes capturing the workflows, the development of process flowcharts, and script development (Penttinen et al., 2018; Leno et al., 2018). RPA tools extract anchors using HTML code and APIs for capturing procedures on GUI (Van der Aalst et al., 2018). In flowcharts, configurable process components are graphically displayed to depict the execution of the process. Components may be bundled into packages and made available in process libraries so that modelled sub-processes can be reused and updated with little effort (Penttinen et al., 2018). The third method for configuring robots is script development, which requires programming (Leno et al., 2018). RPA is built on a foundation of robotics, cognitive computing, and artificial intelligence (AI), allowing for autonomous learning and decision-making by robots (Kaya et al., 2019; Van der Aalst et al., 2018).

There has been a widespread usage of RPA tools to automate workflows in various domains. UiPath, Automation Anywhere, WinAutomation, AssitEdge, and Automatica were among the RPA tools investigated by Ribeiro et al. (2021). UiPath allows the creation of RPA features and capabilities in its framework in order to build and execute programming scripts (Tripathi, 2018). The platform running UiPath is subdivided into three components: UiPath Orchestrator, UiPath Robot, and UiStudio. The latter permits the modeling, designing, and execution of workflows. It further transfers packages, creates and maintains connections between the robots while the UiPath Orchestrator orchestrates the robots. UIAutomation offers some Artificial Intelligence (AI)

to UiPath tools in information extraction, classification, optimization, and recognition (UiPath, 2022). Automation Anywhere RPA tool integrates analytical data analysis and cognitive automation features applied to RPA processes (Ribeiro et al., 2021). Using IQ Bot, Automation Anywhere executes algorithms such as Natural Language Processing, Artificial Neural networks, and Fuzzy logic to extract and validate the information. Thus improving the efficiency of document validation. Automation Anywhere has its application in healthcare and pharma (12%), high-tech and telecommunications (12%), and in 30% of capital markets and banking (Le Clair et al., 2018). The choice of RPA tool was based on overall ease and cost of deployment. WinAutomation delivers a set of functionalities such as email automation, automation of files in various formats, and other features incorporated into RPA processes (WinAutomation, 2021). The tool can further perform the automation of file transfer protocol, file, and folder automation, excel automation, mouse and keyboard automation, automated task execution, web and desktop automation, scripting, automated database, and SQL, etc. AssistEdge improves operational productivity and business processes while modernizing customer service (AssistEdge, 2021). It uses algorithms such as Artificial Neural Networks to automate data analysis and data capture. Automagica, a tool developed in Python exploits AI techniques to provide RPA services (Automagica, 2021). RPA features are compatible with Google Tensorflow for text and image recognition. The features included the capability to automate information from the browser, automate information from Word and Excel files, extract texts from PDF files, read optical character recognition, etc. The duplication and programming of the RPA processes can be performed differently to execute different tasks.

2.4.3 Benefits and Drawbacks Of RPA

RPA presents a lot of benefits, which outweigh the drawbacks (Penttinen et al., 2018). The most frequently discussed benefit in literature is cost saving (Kaya et al., 2019; Anagnoste, 2017; Asatiani and Penttinen, 2016; Aguirre and Rodriguez, 2017; Geyer-Klingeberg et al., 2018; Lacity et al., 2016; Madakam et al., 2019; Fernandez and Aman, 2018). An RPA solution may cost as little as one-fifth as much as a full-time worker would to do the same work (Kaya et al., 2019). The savings may be ascribed to the low-cost deployment (Anagnoste, 2017; Hallikainen et al., 2018; Lacity et al., 2016) and RPA's integration into an organisation's infrastructure because of its non-intrusive nature (Madakam et al., 2019; Osmundsen et al., 2019). When assessing the performance of RPA, the examined literature emphasises increased productivity and improved quality of work (Ratia et al. 2018) concerning accuracy (Leshob et al., 2018; Kaya et al., 2019; Geyer-Klingeberg et al., 2018; Madakam et al., 2019; Fernandez and Aman, 2018).

In reporting the performance of RPA, the literature highlighted improved productivity and work quality (Ratia et al. 2018) in terms of efficiency (Leshob et al., 2018; Kaya et al., 2019) and accuracy (Leshob et al., 2018; Kaya et al., 2019; Geyer-Klingeberg et al., 2018; Madakam et al.,2019; Gejke, 2018). In addition, minimizing human inattention (Fernandez and Aman, 2018) reduces errors (Kaya et al., 2019; Penttinen et al., 2018), which improves consistency (Geyer-Klingeberg et al., 2018) and dependability (Madakam et al., 2019) of operations. Additionally, substantial time savings are cited as an additional advantage of RPA (Gejke, 2018). This element enables workers to be focused on more engaging and high-value jobs (Leshob et al., 2018; Asatiani and Penttinen, 2016; Lacity et al., 2016; Fernandez and Aman, 2018; Leopold et al., 2018). For example, customer service (Kaya et al., 2019; Ratia et al. 2018) and innovation (Kaya et al., 2019). RPA has the potential to reduce human labour while ensuring scalability (Kaya et al., 2019; Geyer-Klingeberg et al., 2018; Osmundsen et al., 2019) and high flexibility (Leopold et al., 2018). One major advantage of RPA is the quick deployment (Asatiani and Penttinen, 2016; Lacity et al., 2016) and 24/7 availability (Kaya et al., 2019; Anagnoste, 2017; Madakam et al., 2019; Fernandez and Aman, 2018) to carry out routine tasks. RPA may provide a substantial return on investment due to its simplicity and inexpensive configurations (Anagnoste, 2017; Penttinen et al., 2018).

2.5 Chapter Summary

Data quality management plays a fundamental role in any data-collecting organization. Hence, much of the research was conducted to investigate the impact of poor data quality on the advancement of data-driven organizations. Several methods of data quality management were deployed in the literature to enhance data quality. These included the designing of data quality frameworks, validation algorithms, data quality measurements, and electronic data collection systems. The emphasis on the significance of data quality management resulted in the creation of several data models and frameworks. However, the majority of these frameworks were neither designed nor evaluated for HDSS-based longitudinal data nor were they validated in actual environments.

On the other hand, previous studies provided enough evidence to suggest that EDC improves data quality compared to PDC. The findings showed that PDC was prone to errors, cumbersome, time-consuming, and had longer turnaround times. EDC was regarded as the optimal method for collecting high-quality data. ODK-based EDC dominated the literature due to its flexibility in validating the data and support for real-time and offline data collection. Furthermore, the platform had advanced features to cater for most of data collection needs and data quality assurance. However, the platform lacks

some crucial features such as complete data preloading, provision of paradata, workspace for task separation, and user-friendly interfaces for survey management. Such features are necessary for improving the data quality and speeding up the processes of data collection. Quality controlling numeric data became a focal point for most of the studies despite the multidimensionality of data types in given a dataset. Quality controlling textual data remains a challenge with most of the approaches proposed in the literature. In an effort to manage and control data quality, a wide variety of tools for building business intelligent systems were discussed. Reporting data quality issues was considered the foremost important step and some studies suggested the deployment of business intelligent systems. Such systems enabled users to identify and fix data quality issues and most importantly, allowed decision-makers to make well-informed data-driven decisions. Several tools such as Pentaho business intelligence solution, Rstudio with Shiny, Tableau, Oracle, SAP, BusinessObjects BI, Microsoft BI solution, etc., were considered. Some of these tools have built-in features to extract, transform, and load (ETL) data into the data warehouse for the development of multidimensional arrays called cubes. Numerous studies adopted this approach in integrating data from multiple repositories. Although most of the studies considered some form of data integration, it is not clear whether the integration of data was performed manually or scheduled to run automatically. Automating processes has received attention in literature due to its benefits in improving productivity and reducing errors. However, far less attention was provided to data quality improvement in the HDSS domain. Automating data quality management in HDSS remains a “grey area”, requiring further exploration.

Chapter 3

Theories and System Models

3.1 Introduction

This chapter discusses fundamental design concepts and models to enable the researcher to better understand the phenomenon under investigation and effectively plan the solution. The approach paves the way for the design of a high-quality data system. Reports insinuate that the quality of system design greatly affects the quality of the collected data (Krippendorff, 2009). Developing design models before the development of the system "is as essential as having a blueprint for a large building" (Wu, 2006). Models are built before building a robust system to help the researcher deal with the complexities associated with multiple components of the system. Understanding the connection and communication amongst various components is of paramount importance in building a good quality system. Therefore, the researcher pays special attention to the design aspects of the system and his philosophical stance.

The chapter is outlined as follows: Section 3.2 presents the design of the electronic data collection (EDC) system components. Section 3.3 presents theoretical background while Section 3.4 discusses a novel 3-Tier Total Data Quality Management (3TTDQM) framework with automated data quality control mechanism. Data quality measurements with considerations on data accuracy, completeness, consistency, timeliness, and validity are presented in Section 3.5 while Section 3.6 takes a dive into Pareto analysis and statistical process control technique. In Section 3.7, the flow of data in the EDC is presented and section 3.8 presents a work flow for the proposed automated data quality control system. Section 3.9 discusses the workflow of the data quality reporting utility built to continuously monitor data quality. Section 3.10 presents a conceptual model for the database used for reporting data quality issues, while Section 3.11 illustrates and

discusses the communication model that shows data synchronisation over the communication network. Section 3.12 shows a blueprint or conceptual model of the proposed study, which provides a step-by-step guide on how the entire system was constructed. The Chapter is concluded in Section 3.13.

3.2 Electronic Data Collection System Component Design

"Component represents a modular, deployable, and replaceable part of a system that encapsulates implementation and exposes a set of interfaces" (Booch et al., 2001b). Component-level design describes the interface characteristics, algorithms, data structures, and communication processes allocated to every component for system development. The component design details can be modelled at various levels of abstraction with a unified modelling language (UML) activity diagram representing processing logic (Roger and Bruce, 2015). Figure 3.1 presents the UML diagram representing a component-level design of the proposed electronic data collection system. The components are linked using pointed arrows that show the relationships among them. Each component encapsulates a set of questions asked during data collection.

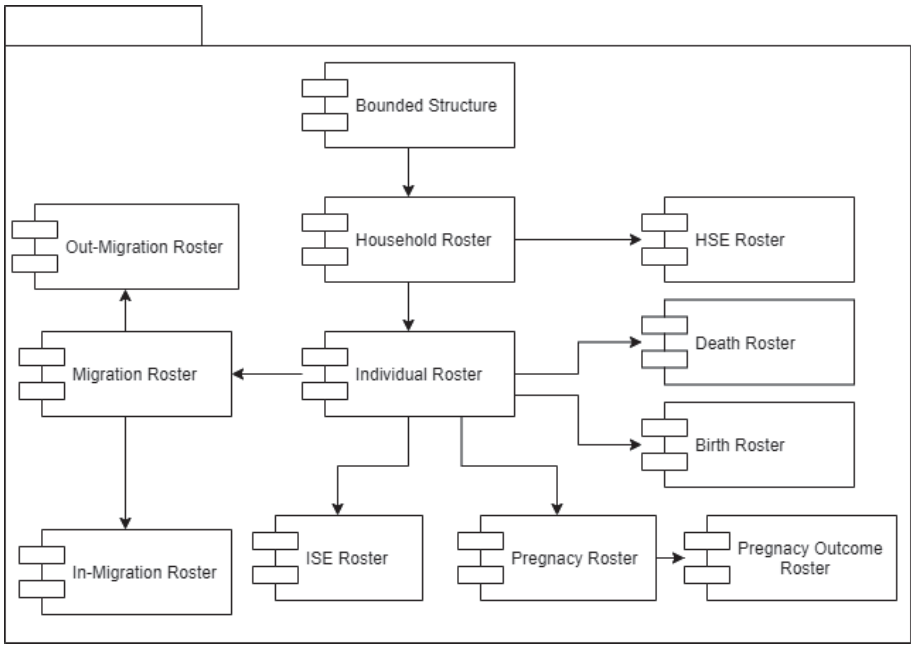


FIGURE 3.1: Component level design of an electronic data collection system

The components are defined as follows:

1. Bounded Structure – This component encapsulates questions related to the dwelling and defines the location of one or more households. It provides important information such as Geographic Information System details (Longitudes, Latitudes,

Altitudes, and Accuracy) and dwelling details (Dwelling Id, Dwelling start date, Owner, State, etc.). This information allows for the identification of the dwelling in which a household is resident.

2. Household Roster – This component captures household-related questions. A household is defined as a group of individuals who share the same meal, and the group reports to one household head. The basic details such as household members, household head, migration, etc. are considered in this component. In addition to that, a subcomponent HSE Roster is embedded within the Household Roster component to capture household economic statuses. There exists a strong correlation between the two components.

   (a) HSE Roster – HSE is an acronym derived from Household Socio-Economic. This subcomponent has a bearing on household socioeconomic status questions and records details such as household financial status, power sources, food security, water supply, etc.

3. Individual Roster – Unlike the household roster component that captures details at the household level, the individual roster records details at an individual level. The focus at this point is on the collection of individual details. Basic details captured: surname, names, date of birth, gender, contact details, parental details, marital status, conjugal relationships, etc. Due to the voluminous details collected at the individual level, the individual component was further restructured into subcomponents. These sub-components are the death roster, births roster, ISE roster, pregnancy roster, and migration roster.

   (a) Death Roster – As individuals are continually being observed, some exit the surveillance system due to death. It remains critical to record details related to death, and this subcomponent was designed specifically for that purpose. The details such as date of death, place of death, causes of death, etc. are considered.

   (b) Birth Roster – Individuals are born at some point during surveillance. The birth component was designed to keep track of all births in the surveillance system and record details such as date of birth, birth settings, birth attendant, live birth, stillborn, etc.

   (c) ISE Roster – ISE is short for Individual Socio-Economic. This component captures the socioeconomic status of an individual. Such statuses, amongst other things, include employment, education and finances, grants, religion, marital, and health status.

   (d) Pregnancy Roster - The component records details of those who were observed pregnant during surveillance. If an individual is observed to be pregnant,

then it is expected that such an individual will have a pregnancy outcome. The details of the pregnancy outcome are recorded under the subcomponent named Pregnancy Outcome Roster.

i. Pregnancy Outcome Roster – Captures the pregnancy outcome events that result from pregnancy. The component captures details such as pregnancy outcome (abortion, miscarriage, and live birth), method of delivery (normal, cesarean, etc.), birth attendant (nurse, traditional healer, etc.), and birth settings (home, health care facility, etc.).

(e) Migration Roster - Migration is defined as the movement of people in and out of the surveillance area. Therefore, the migration roster subcomponent keeps track of these movements and decides on the direction. The direction refers to moving in or out of the surveillance area. Moving out triggers the out-migration roster subcomponent, while moving-in activates the in-migration roster subcomponent.

i. Out-Migration Roster – This subcomponent captures details to do with outmigration, and these are migration date, place migrating to, the reason for migration, etc.

ii. In-Migration Roster – This subcomponent considers gathering information about individuals moving into the surveillance area. Details such as migration date, place migrating to, reason for migration, etc. are captured at this level.

The components are connected to show the level of dependencies and associations. The connectedness of the components shows how the underlying tables are represented in the database. The tables use the same naming conversation as the components. The conceptual model of the database takes the structure of the components. The links between the components represent the relationship between the entities or tables. Components encapsulate questions, and each question is given a unique variable name for identification and mapping. By default, according to our system design, the questionnaire variable names become column names in the database. This enables the ease of variable mappings between the electronic data collection (EDC) system and the database tables. The components and questions were designed with the quality of the system and the data in mind. To enforce the quality of the system and data, the skipping logic algorithms and validation rules are built into the system to assist the user in navigating the system with ease. The validation rules enable the correct capture of data and notify the user of any violations. A well-designed EDC system that incorporates the necessary algorithm may prevent potential data problems from 'crawling' into organisational archives. EDC can significantly minimise errors in transcription, improve efficiency and facilitate the

flow of information and data timeliness, resulting in improved data quality (Alschuler et al., 2004). The EDC system may minimise the possibility of data loss (Tomlinson et al., 2009), error rates (Thriemer et al., 2012), the time required for data collection (Avilés et al., 2008), improve the completeness of data (Zeleke et al., 2019) and the automatic gathering of geolocation data and timestamps. According to Ley et al. (2019), EDC is the most efficient method of data collection and positively contributes to improving data quality. From this discussion, we hypothesise that
H1: Skipping and validation algorithms may significantly improve the effectiveness of EDC and the quality of data at the point of capture.

3.3 Theoretical Background

Decision makers adopt the information generated through methodical and heuristic processes. The processes are not superior to each other. The best cognition processes determine the optimal strategies to improve data quality for decision-making. The theoretical model in Figure 3.2 was developed to understand both the perceptual and the objectives of data quality assessment. The users meticulously and heuristically process the received data. Data processing can influence the assessment of data quality, and this processing depends on the availability of cognitive resources. The deployment of more cognitive resources to the problem would mean that systematic processes are dominant and, on the other hand, fewer cognitive resources for the assessment of data quality imply that heuristic processing prevails. Therefore, the highest level of user experience on a particular topic and involvement increases cognitive resources. These factors may result in systematic processing and increase the role of data quality dimensions in determining both quality and relevance. For a less skilled and uninvolved user, heuristic processing will dominate the assessment of data quality and relevancy. The assessed data quality was measured based on how users evaluated objective data quality and data believability. Data quality attributes inform how relevant the data is. However, irrelevant data cannot be of high quality, no matter how timely, complete, and accurate the data is.

To ensure high-quality data, we adopted the data quality work cycle from (Canadian Institute for Health Information, 2009). The data quality work cycle comprises three major activities:

1. Planning, which involves data quality activities (these activities include best practices on data entry, monitoring incoming data records, data management, and reporting).

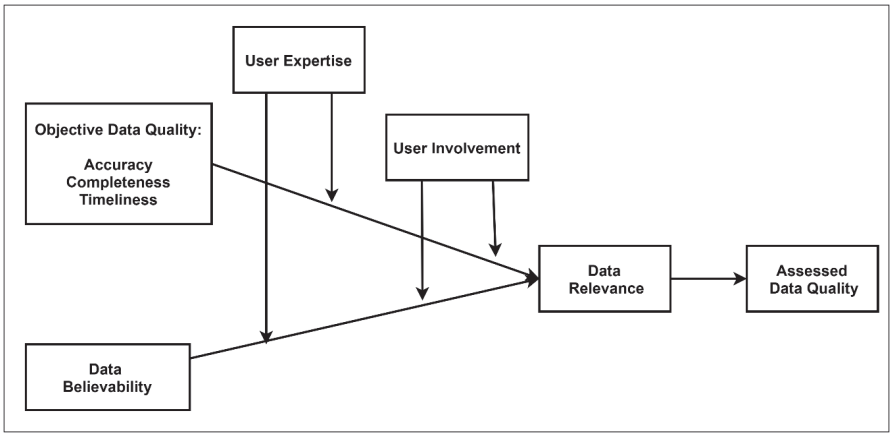


FIGURE 3.2: Theoretical Model of Data Quality Assessment (Shankar and Watts, 2003)

2. Implementation of data quality measures (deployment of an electronic data quality system with features to validate and report data quality issues).

3. Assessment of data quality (reporting the level of data quality using data quality attributes such as accuracy, completeness, consistency, timeliness, and validity).

These three activities must be repeated as frequently as possible to achieve the high-quality data-quality objective. As shown in Figure 3.3, planning, implementation, and assessment directly influence the outcomes of data quality assessment.

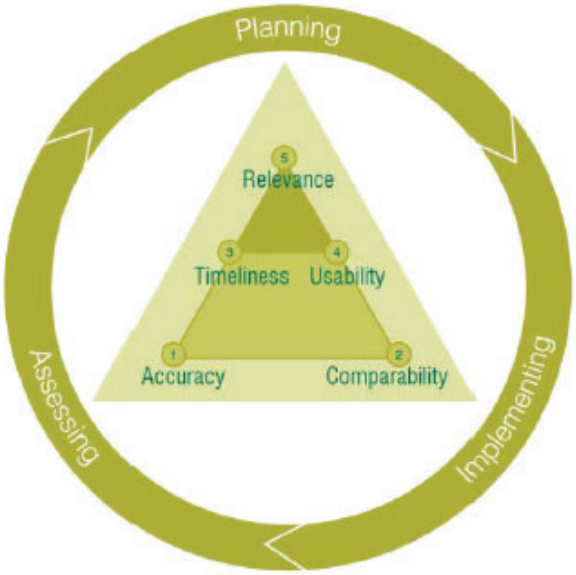


FIGURE 3.3: Hierarchical Approach to data quality control (Canadian Institute for Health Information, 2009)

These activities serve as the building block for high-quality data assessed using accuracy, comparability, timeliness, usability, and relevance. These dimensions can be viewed together as interrelated and, to some extent, interdependent. For example, the relevance

of the data affects other dimensions. If the data is irrelevant, its value declines substantially to the extent of not being fit for the purpose, even if the other four data quality dimensions are met.

To compare different data elements, they have to be accurate and consistent across the systems, in which case the data becomes usable. The usability of the data comes from the ability to access the data and the ease with which it can be understood. The data must be as timely as possible to reflect the current state of a real-life entity. These data quality dimensions, if properly planned and achieved, enable users to make well-informed decisions based on comparable, relevant, accurate, timely, and usable data.

Figure 3.4 presents a data quality framework for classification tasks. The guides the user through the identification and dealing with the data quality issues in classification tasks and presents an ontology to epitomise knowledge in cleaning data and recommend a possible plan of action. The framework provides a stepwise approach to resolving data quality issues in the classification of machine learning and data mining projects. The framework suggests the imputation in the existence of missing values. Otherwise, the imputation step is skipped to detect outliers. The outliers are selected based on algorithms for high dimensional spaces such as density algorithms such as Local Outlier Factor (LOF) or Angle-Based Outlier Degree (ABOD), etc. (Breunig et al., 2000; Kriegel et al., 2008).

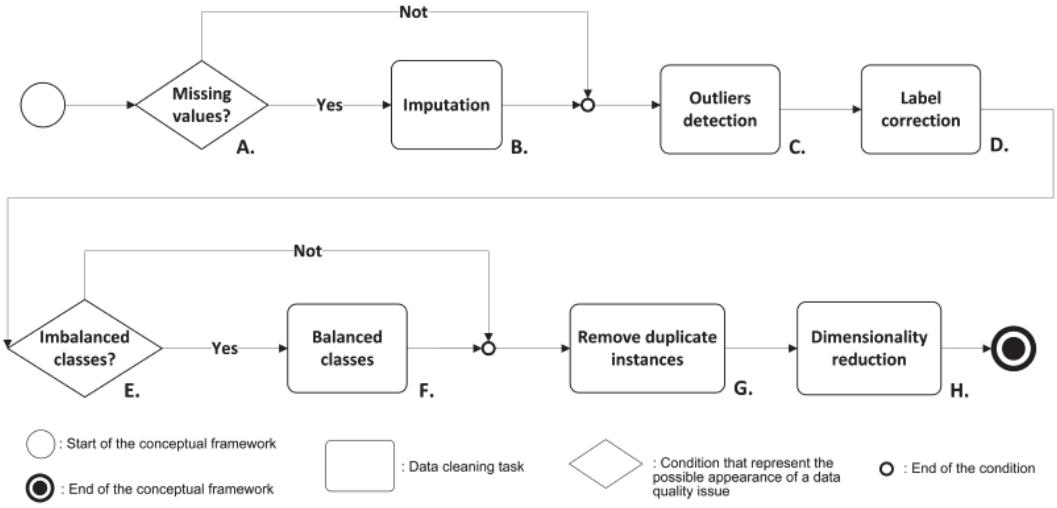


FIGURE 3.4: Conceptual model for data cleaning in task classification (Corrales Múñoz et al., 2018)

Label correction identifies cases with the same values. But in the case of different classes, the label is corrected or the occurrence is removed. The label correction algorithm runs to detect mislabelled instances from the incoming data or data generated through the imputation processes. For imbalanced classes, an imbalanced ratio is applied to measure

the distribution of binary classes. The class is considered balanced when a normalised entropy is close to 1. Remove duplicates step checks the duplicate instances and runs an appropriate algorithm to ensure uniqueness of data. The final step of the framework reduces the dimensionality of the data.

3.4 3-Tier Total Data Quality Management Framework

Data Quality Frameworks (DQF) are incorporated into the data management system to assess, analyse, and clean data (Corrales Múñoz et al., 2018) and according to (Kerr and Norris, 2004; Wang and Strong. 1996), data management systems drive organisational profitability tasks and processes. DQF must present a general structure to analyse and solve data quality issues (Eppler and Wittig, 2000). Based on this premise, a novel 3-Tier Total Data Quality Management Framework (3TTDQM) was designed and implemented (Figure 3.5). The 3TTDQM establishes the structure of data quality management systems with the primary purpose of producing high-quality data. The 3TTDQM framework comprises three tiers of data quality control: 1) Electronic data collection (EDC) system with data validation algorithms, 2) automated data quality control system, and 3) SQL triggers and stored procedures. The framework provides a general model for analysing and solving data quality issues (Eppler and Wittig, 2000). Tier 1: The EDC system was developed to enforce the accuracy, consistency, completeness, and validity of the data. The system incorporated algorithms and validation rules to detect and flag instances of data quality breaches. The descriptive error message that outlines the nature of the violation is presented to the field worker to allow immediate correction at data entry. Data preloading adds an additional layer of data quality control by validating the presently gathered data with historical data. This step enforces consistency and validity between two datasets.

In tier 2: Human intervention is completely eliminated by automating the data quality control process. Three levels of automation are considered: data export, loading and validation, and rejection/approval automation. An automated data export using R Application Programming Interface (R-API) feeds data into Pentaho Data Integration (PDI) for loading into the database. PDI is also automated to enable an efficient flow of data streams. SQL scripts validate the data before being loaded into the database. Validations determine whether the data is rejected or not. If rejected, another R-API automation interfaces with the Survey Solutions® server to autonomously reject the data else approves. Automates are time-triggered by the Windows task scheduler to run tasks at various times. Section 3.7 unpacks the workflow at this tier and describes the automation processes at the lowest level of abstraction.

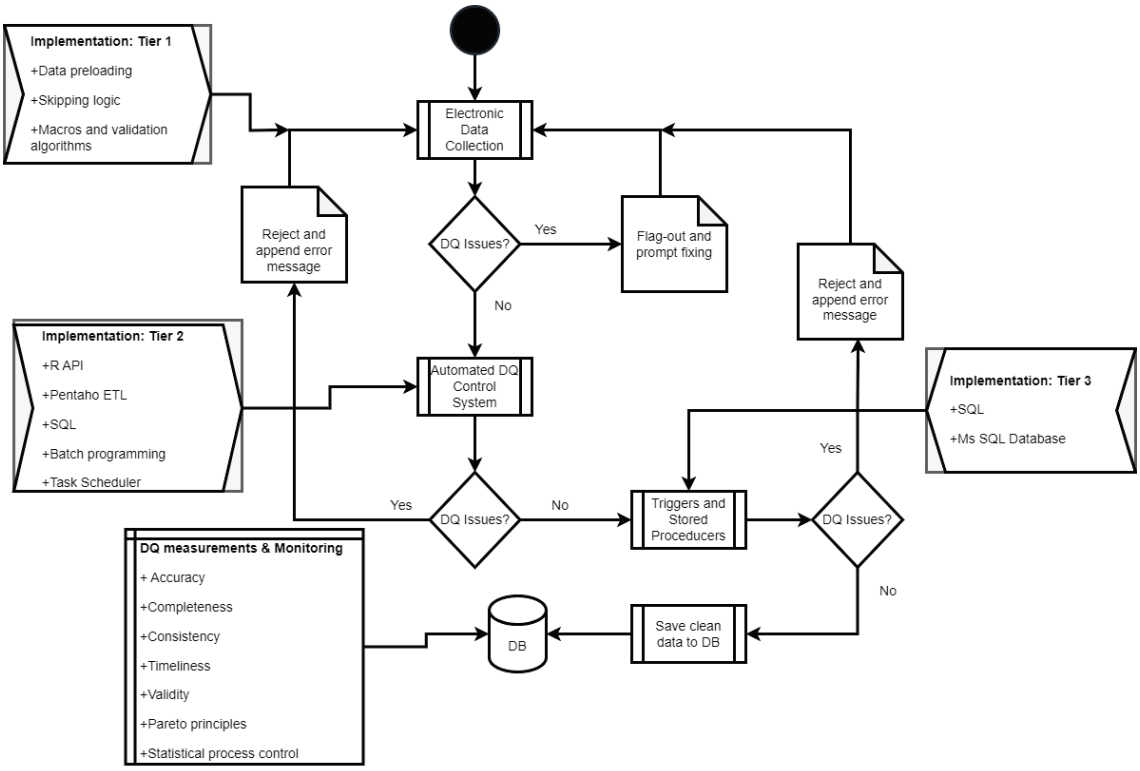


FIGURE 3.5: Representation of 3-tiered total data quality management framework

Tier 3 performs the final data cleaning by passing the data through SQL triggers or stored procedures to further validate its quality. This preserves data integrity and consistency by comparing the incoming data with the database data. In the event of a data quality violation, an error report that describes the nature of the error is generated with suggestions of how it must be resolved. This tier ensures thorough data cleaning before loading it into the production database. Continuous monitoring of the level of data quality in the database is crucial (Shah et al., 2010). As a result, the level of data quality in the database is regularly measured and monitored using data quality metrics and statistical methods. This work uses statistical methods (Pareto and Process control technique) to identify, monitor, and report problems in the data. Data quality is a multidimensional concept and context-dependent. According to (Borek et al., 2013), organisations must select suitable data quality dimensions to evaluate the extent to which data meet their objectives. For that reason, our study considered five dimensions or attributes to evaluate the quality of the data in the database (refer to Figure 3.5). We set a threshold of 90% and consider a data quality level of at least 90% to be acceptable. Any measure below 90% indicates the need for improvement.

A robust data management system must be able to report errors and their sources, recommend corrective measures and have an interface to resolve data-related issues.

This serves as an important component of the proposed DQF. DQF offers businesses a collection of instruments and methodologies to assess data quality levels and prioritise service data (Camera et al., 2020). According to (Corrales Múñoz et al., 2018), DQF is intended to assess, analyse, and improve data of low quality in data management systems. The framework provides an all-encompassing paradigm for analysing and resolving data quality problems (Eppler and Wittig, 2000). The DQF is implemented through data cleansing routines, data integration procedures, and verification of data accuracy.
We therefore hypothesise that:

H2: By implementing a DQF, data quality may be monitored systematically, with the possibility of identifying areas for improvement and guaranteeing the availability of trustworthy and accurate data.

3.5 Data Quality Measurements

Data quality is an integral component of good data management. Organisations rely on high-quality data to make well-informed decisions, drive business operations and profitability, and gain valuable insights. Data quality metrics or measurements are essential for the assessment of data quality, the identification of areas for improvement, and ensuring that the data is appropriate for its intended use. Data quality metrics are the techniques and measurements used to assess and quantify data quality. These metrics give a systematic method for evaluating: 1) accuracy, 2) completeness, 3) consistency, 4) timeliness, and 5) validity.

These metrics are used to measure the level of data quality in the database and are discussed in full in the sub-sections below. In addition to data quality metrics, we consider process control techniques and Pareto analysis in the identification of potential problems and quality analysis. Before diving into metrics and data quality analysis, we derive mathematical formulations to model the data in the database. We consider database D_x as consisting of data values $x_1, x_2, x_3 \ldots\ldots\ldots\ldots x_n$. The dirty data ($D_i$) may be represented as $D_i \in \{x_1, x_2, x_3 \ldots\ldots\ldots\ldots x_n\}$ and clean data, denoted by C_j for $C_j \in \{x_1, x_2, x_3 \ldots\ldots\ldots\ldots x_n\}$ where $i, j = 1, 2, 3, \ldots\ldots\ldots n \; \forall n \in \mathbb{R}$. All data values in the database can be represented as:

$$D_x = D_i U C_j \tag{3.1}$$

We seek to eliminate D_i while increasing C_j in D_x. To achieve this objective, we measure the level of data quality in D_x using the selected data quality metrics. In addition, we perform data quality analysis to identify and monitor potential causes of data quality issues. This is done using Pareto's principle and Process control approaches. After measuring the extent to which data meets the set organisational data quality standards, data managers or users must take action to resolve identified data quality issues.

3.5.1 Data Accuracy

DAMA UK (2013) defined accuracy as "the degree to which data correctly describe the "real world" object or event being described". Redman (1997) defines accuracy in terms of how near the value A is close to A'. Value A' is considered to be consistent with real-world entities. The data is accurate if and only if two values are equal. The data accuracy can be represented semantically and syntactically, and Mecella et al. (2002) distinguished between the two. Syntactic precision defines the nearness of value A to value A' where the syntax of value A' is considered correct. On the other hand, semantic accuracy defines the proximity of value A to value A' where A' is perceived semantically correct. The accuracy of data in the database depends on several factors such as measurement error – introduced when a data element is answered incorrectly. This may be a result of unclear definitions or a lack of training for data collectors. A key question is what measures to employ to minimise errors or issues in the incoming data. We define the accuracy of the data, A_c by

$$A_c = \frac{\sum_{i=1}^{n} x_i - \sum_{i=1}^{n} e_i}{\sum_{i=1}^{n} x_i} \tag{3.2}$$

$$= 1 - \frac{\sum_{i=1}^{n} e_i}{\sum_{i=1}^{n} x_i}$$

where e_i is the erroneous values and x_i represents all data values in the system. The main objective here is to make $\sum_{i=1}^{n} e_i \to 0$ such that $\mathrm{Ac} \to 1$.

H3: Data validation and quality control processes, if well implemented, can improve the accuracy of data

3.5.2 Data Completeness

The completeness of data is defined by (Bove et al., 2003) as "information having all required parts of an entity's description". Data completeness refers to the degree to which data coverage is achieved and it is context-dependent according to (Sebastian-Coleman,

2013; Tomic et al., 2015; Orme et al., 2007). In longitudinal data systems, maintaining all participants of interest throughout all observation periods is essential. The check must be made to ascertain if there is a difference between the population of interest and the population of reference. The difference must be zero if all participants of interest were observed. In that case, all participants would have been observed and the data would be complete. The incompleteness of the data may also be a result of missing values. All mandatory data values must be captured for the data to be complete. Data collection serves the sole purpose of bringing data into the database. At that point, a wide range of issues are introduced, such as typos, missing values, and inconsistent data entries. The completeness of data can be measured at three levels: schema, column, and population completeness (Pipino et al., 2002). The completeness of the schema is the most abstract level and checks the degree to which attributes and entities are not missing from the schema. This ensures that all objects assigned to a specific schema are present. Column completeness assesses the degree to which data values are not missing from the table attributes, while population completeness is concerned with the degree to which the reference population consists of non-missing values. The assessment of schema and population completeness is beyond the scope of this work. Much attention is paid to assessing completeness at the attribute level. It is important to evaluate NULL values in relational databases to establish the reasons why it occurred (Sampaio et al., 2015). There are several reasons NULLs are found in the data and according to (Atzeni et al.,1993); they occur due to 1) the value is inexistent, 2) the value exists but is currently unavailable, 3) it is not known if the value exists. So cases 1) and 3) may not necessarily be considered incomplete during the assessment. For case 1), the reason is that if the value does not exist, means it cannot be mapped to a real-world entity. That value will never exist. For example, in the column pregnancy_duration, male participants must always have NULL values, since it cannot be possible for males to be pregnant. In case 3), using the same example, a woman may be in the early stages of pregnancy and not aware of her pregnancy status. So at that point, no one knows if pregnancy exists, and as a result, the NULL value can be captured. So, the two cases cannot be deemed incomplete; however, Case 2) is a typical example of incomplete data values. For better clarity, we consider the variable Date_of_birth. Having a date of birth set as a NULL value would mean that an individual does not have a date of birth. In reality, every individual must have a date of birth, otherwise it is missing. In that case, the value of the database does not correspond to the real entity represented.

Data completeness seeks to answer the question; "is all the relevant data at an attribute level present". In an attempt to answer the question, we consider the completeness of data in D_x as denoted by C_x and missing values represented by m_i.

Then C_x can be computed by

$$C_x = \frac{\sum_{i=1}^{n} x_i - \sum_{i=1}^{n} m_i}{\sum_{i=1}^{n} x_i} = 1 - \frac{\sum_{i=1}^{n} m_i}{\sum_{i=1}^{n} x_i} \tag{3.3}$$

where $0 \leq \frac{\sum_{i=1}^{n} m_i}{\sum_{i=1}^{n} x_i} \leq 1$. We need to make the value of $\frac{\sum_{i=1}^{n} m_i}{\sum_{i=1}^{n} x_i}$ as small as possible to maximize the output C_x.

H4: Implementing validation checks or pre-defined rules on the required fields in EDC can greatly improve data completeness.

3.5.3 Data Consistency

Data consistency means that the "representation of data values remains the same at multiple data items in multiple locations" (Almutiry et al., 2013). The percentage of values that match across different platforms. In a database environment, data consistency means that the data between two databases have the same meaning and structure. Data consistency is essential in a data management system to ensure that data is correct and current. When the same data is stored in various forms or data types across separate systems or databases, or when the data is not synced between systems, data inconsistency can emerge. This might result in inaccurate or misleading conclusions when data are used for decision-making or analysis. The consistency of data is essential to ensure that the data is reliable and dependable and those decisions based on that data are accurate. By creating appropriate data management procedures and standards, companies can ensure that their data is consistent and of good quality, leading to better operational outcomes.

The check of theoretical consistency or plausibility is a useful step in detecting errors in data (Schwarz, 2018). In a data collection environment, respondents' responses are not always theoretically consistent and may at times directly contradict one another. Such situations are most likely to occur as a result of respondents misinterpreting the survey question, misreporting for other reasons such as social desirability, or simply an incorrect recall of information. Schwarz (2018) recommends looking at cross-tabulations of variables to see if there are any implausible combinations of answers.

H5: Preloading historic data along with the primary data collected can improve consistency during the data collection.

3.5.4 Data Timeliness

Timeliness means that "shared data should be as near to real-time as possible. Thus, the data should be timely, in that it relates to the present" (Orfanidis et al., 2004). Data timeliness has gained relevance in time-critical applications such as large-scale sensor data management and real-time information systems (Quin et al., 2014). The frequency with which the data is updated plays a critical role in such systems. Data update frequency can be divided into two main categories: static and dynamic data (Pernici and Scannapieco, 2003; Bouzeghoub, 2004). The former refers to the data that may not be updated during its lifetime. For example, data representing planet names, mathematical equations, etc. may not change during its lifetime. On the other hand, the latter means that the data is updated frequently. The frequency at which dynamic data is updated may be user-driven or automated. The design consideration is how frequently the data can be updated. The dynamic data's frequency time interval is divided into two main categories: seldom-update data and frequency-update data. With seldom-update, the data is updated less frequently and this may be monthly, quarterly, yearly, and so on. Even for non-real-time applications, such as reporting and business intelligence, timeliness can be crucial. In such instances, it may be necessary to frequently update the data to ensure that it is current and relevant. On the other hand, the frequency update allows high-frequency data updates to fit information into the time-critical application. Various factors can impact timeliness, including the frequency of data acquisition, the precision of data entry, and the speed of data processing. The choice of frequency depends on the kind of application to deploy. The relevant attributes for the assessment of timeliness are those containing time, date, timestamps, etc. Timeliness is a crucial feature of data quality, as outdated data can result in less accurate analysis and decision-making. Generally, ensuring data timeliness involves efficient data management procedures, including automated data quality checks and data processing, and frequent data updates. In assessing data timeliness, we consider all fields containing the dates. These are mainly event dates (migration dates, date of death, observation dates, etc.). The freshness of the data is a fundamental part of the data quality. The timeliness of data (T_x) can be modelled as $S_i \leq C_i \leq E_i$. The current date value C_i is bounded between S_i and E_i where S_i is the start observation date and E_i is the end observation date value. So, data timeliness can be computed as

$$T_x = \frac{\sum_{i=1}^{n} x_i}{\sum_{i=1}^{n} C_i + \sum_{i=1}^{n} E_i} - \frac{\sum_{i=1}^{n} E_i}{\sum_{i=1}^{n} C_i + \sum_{i=1}^{n} E_i} \tag{3.4}$$

Since C_i and E_i are disjoint subsets of x_i then, it implies that $\sum_{i=1}^{n} C_i + \sum_{i=1}^{n} E_i =$

$\sum_{i=1}^{n} x_i$. So equation (3-4) can be simplified as

$$= 1 - \frac{\sum_{i=1}^{n} E_i}{\sum_{i=1}^{n} C_i + \sum_{i=1}^{n} E_i} \tag{3.5}$$

$$= 1 - \frac{\sum_{i=1}^{n} E_i}{\sum_{i=1}^{n} x_i}$$

The highest percentage of data timeliness can be achieved if and only if $\sum_{i=1}^{n} E_i$ is made as small as possible, which influences the final output of T_x.

H6: Measuring the timeliness of data can improve its validity, ultimately improving its overall quality.

3.5.5 Data Validity

According to (DAMA UK, 2013), validity refers to "whether or not a data item conforms to the syntax (format, type, range) of its definition". We seek to find the percentage of data with values within the domain of acceptable values. Data values that fall outside the acceptable domain are regarded as invalid data entries and must be isolated for further scrutiny. For example, an observation event date is valid only if it falls within the start event and end event dates. If an individual has died, then the date of death is out of bounds if it does not occur between the date of birth and the current observation date. We define validity, V_d such that

$$V_d = l_b \leq x_i \leq u_b \tag{3.6}$$

where l_b is the lower bound value or the lowest expected value and u_b the upper bound value or the highest expected value and x_i is the actual date value in the database.

H7: Implementing validation algorithms on date variables can potentially improve the validity of data.

3.5.6 Overall Data Quality Measurement Hypothesis

Sub-section 3.5.1 to 3.5.5 modelled and defined five crucial attributes that were used in this study to measure the quality of data in the database. Data quality measurements contribute to data quality improvement in several ways: assessing and reporting levels of data accuracy, completeness, consistency, validity, and timeliness. Measuring data quality may also help identify data abnormalities, develop data quality benchmarks, and

enable data quality improvement initiatives. Numeric assessment of quality attributes has been highlighted as a crucial element of effective data quality management (Pipino et al., 2002). The data quality assessment framework demonstrates how to quantify and monitor data quality to ensure its consistency over time (Sebastian-Coleman, 2012). Measuring and assessing the quality of data involves contrasting its actual worth with some standard against which it may be judged (Batini et al., 2009). This standard can serve as a baseline for data quality improvement. From the discussions on the data quality measurement, we posit the following hypothesis.

H8: Collectively measuring the levels of data quality can serve as a baseline for the identification of data anomalies and the data quality improvement strategy, which will continuously improve overall data quality (see Figure 3.6).

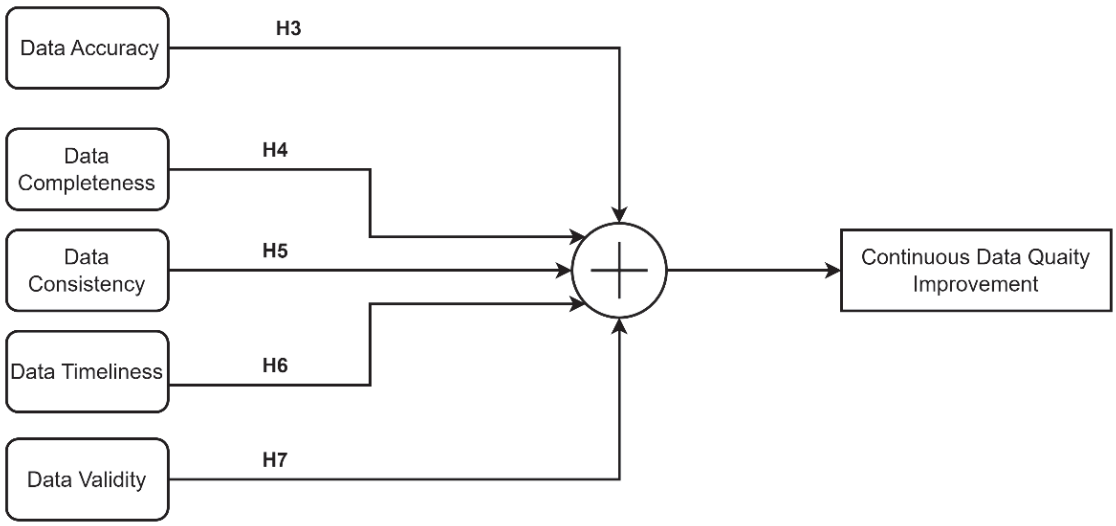


FIGURE 3.6: Overarching Hypothesis on data quality measurements

3.6 Data Quality Analysis

The quality of the data in the surveillance may be affected by several factors. For example, errors, refusals, participants noncontact, premature completion of interviews, etc. The control and monitoring of such factors remain vital to improving data quality. Statistically, process control techniques and the Pareto principle are tools known to control quality and identify major sources of problems. The application of these tools in the context of data quality is discussed in a sub-section below.

3.6.1 Pareto Distribution

In the twentieth century, Vilfredo Pareto observed that 20% of the population own 80% of Italy's wealth (Grosfeld-Nir et al., 2007). The idea was then applied in the quality context by quality expert Joseph M. Juran, who proposed the 80/20 rule, also known as the Pareto principle. He generalised the principle's application and attached Pareto's name to it (Baudin et al., 2012). The idea behind this principle is that for everything we are trying to improve, 80% of the effects lie with 20% of the reasons or causes. So, defining this in the context of data quality, we say that 80% of data quality errors or defects are mostly the result of 20% of the causes. 20% of causes are called a "vital few", and effectively dealing with them may result in 80% improvement. In this case, the quality of the data may drastically improve.

The Pareto principle is a subset of the larger Pareto distribution phenomenon. In his book (Pareto, 1897), Pareto claimed that there is a simple law that determines income distribution in all countries and in all eras. The claim was proven correct and applied in different disciplines to improve the quality of products. We apply this claim to data quality to determine the distribution of error amongst users and causes. So, if we let N represent the number of users with data quality issues larger than a certain limit x, with K and δ as constants, then $N = \frac{K}{X^{\delta}}$ such that

$$\log N = \log K - \delta \log X \tag{3.7}$$

So, this means that if the logarithm of a number of users with data quality issues greater than a definite number is plotted against the logarithm of these errors, the resultant graph is a straight line. The line will have a slope δ known as the Pareto index. The Pareto distribution provides a more general description of the preceding statement and is defined by its cumulative distribution function.

$$f(x) = \left\{ \begin{array}{ll} 1 - (\frac{X_{min}}{X})^{\delta}, & \text{if } X \geq X_{min} \\ 0, & \text{if } X < X_{min} \end{array} \right\} \tag{3.8}$$

Where δ is a Pareto index and X_{min} is a scale denoting the minimum value of X. The density function, which is derived from (3-8) is given by

$$f(x) = \left\{ \begin{array}{ll} 1 - \delta(\frac{X^{\delta}_{min}}{X^{\delta+1}}), & \text{if } X \geq X_{min} \\ 0, & \text{if } X < X_{min} \end{array} \right\} \tag{3.9}$$

If we set the Pareto index to $\delta_0 = log_4 5 \approx 1.16$ then we have the 80/20 rule, which states that 80 percent of the effects result from 20 percent of the causes.

3.6.2 Statistical Process Control

The concepts and methods of Statistical Process Control (SPC) have grown in importance in the manufacturing and process industries (MacGregor and Kourti, 1995). Their goal is to track the performance of a process over a period of time to ensure that it remains in a "state of statistical control." A process remains in a state of control if its values are closer to the desired values and the variations from the mean are within the upper and lower control limits. We apply statistical process control to data to monitor factors that affect its quality. Data quality issues are within control if the mean variations are closer to the center line and their mean values remain within the upper and lower limits. The statement can be represented mathematically by defining the mean or center line (CL) as

$$\bar{x} = \frac{\sum_{i=1}^{n} x_i}{N} \tag{3.10}$$

where x_i denotes the data quality issue and N is the sample size. Data quality issues that are in statistical control must be bounded between the lower control limit (LCL) and the upper control limit (UCL). The two limits are defined by

$$(LCL, UCL) = (\bar{x} - 3\delta\sqrt{N}, \bar{x} + 3\delta\sqrt{N}) \tag{3.11}$$

Since we seek to minimise data errors or defects as much as possible, we set LCL to zero so that (3.11) becomes

$$(LCL, UCL) = (0, \bar{x} + 3\delta\sqrt{N}) \tag{3.12}$$

Daily average errors or defects above the UCL are considered out of control and require intervention. The highest number of errors in the system has an adverse impact on the quality of the data. Ideally, we would want to maximise data quality throughput thereby limiting factors badly influencing its quality. Factors influencing data quality can be plotted on a process control chart for continuous monitoring. The control chart represents the process variation (see Figure 3.7)

The process owner (data managers) identifies causes of variation, especially when the erroneous data average exceeds the limits (LCL and UCL), and must immediately apply

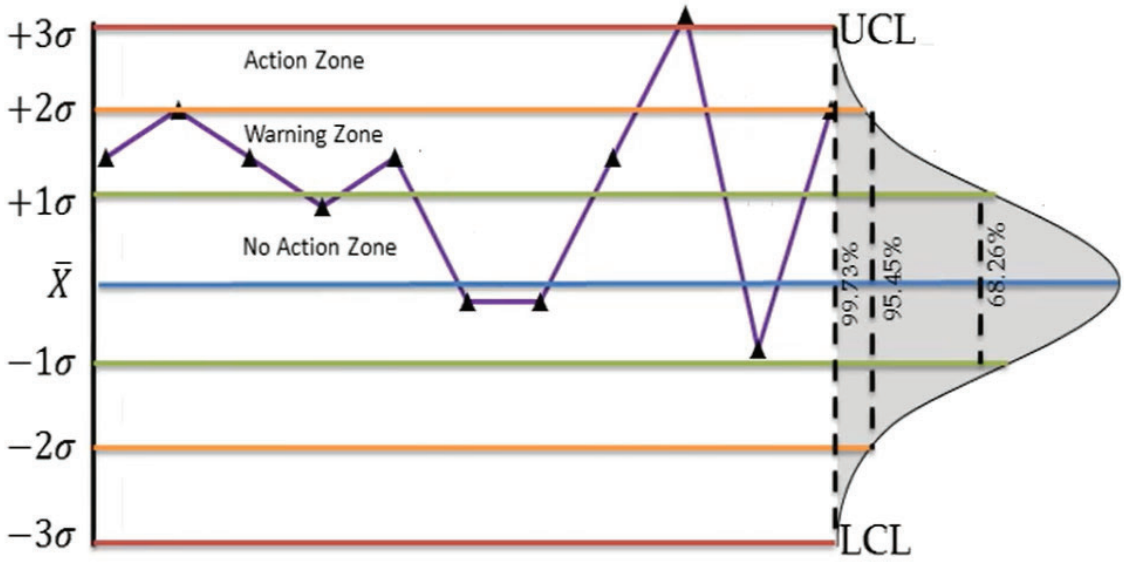


FIGURE 3.7: Statistical control chart displaying the amount of variation in the investigated data

corrective measures to circumvent the situation. The objective is to make the average data variations from the mean as small as possible, in which case no action is required. This is particularly true because the process is under control. The process owner must be on high alert when data values enter the warning zone. This means that things may get out of hand and some interventions are required. The process is said to be out of statistical control if it goes beyond the action zone. This indicates a major problem in data quality, and some corrective actions must be taken to restore the process to a normal state.

Both Pareto analysis and process control approaches play an important role in quality management systems. Pareto analysis assists in identifying significant issues and prioritising improvement efforts, whereas process control approaches provide continuous monitoring and enhancement of process performance. These strategies collectively help to identify and monitor significant data quality issues and deliver higher data quality. Continuous monitoring of data quality is critical to verify that data continues to adhere to specifications and to discover unanticipated changes in data (Ehrlinger et al., 2018). Furthermore, continuous data quality measurement identifies opportunities for future improvements in data quality, hence enhancing the reliability of data-driven decisions. On the other hand, Vancauwenbergh (2019) emphasised that the assessment of data quality must be carried out frequently to continuously improve the quality of data in terms of examining the underlying cause of errors. Organisations must have data quality standards and a continuous monitoring approach to ensure their application improves data quality (Nascimento et al., 2015). From this discussion, we draw the hypothesis

that:

H9: Continuously identifying significant data issues and monitoring data quality enables early detection of unanticipated trends and the application of corrective measures to improve data quality.

3.7 The Flow of Data in the Electronic Data Collection System

The flow of a data collection system consists of a set of interrelated processes that facilitate the smooth progression of the data from the source to the destination. While the specifics may vary depending on the purpose and requirements of the system, Figure 3.8 depicts the data flow from fieldworkers to the operational database.

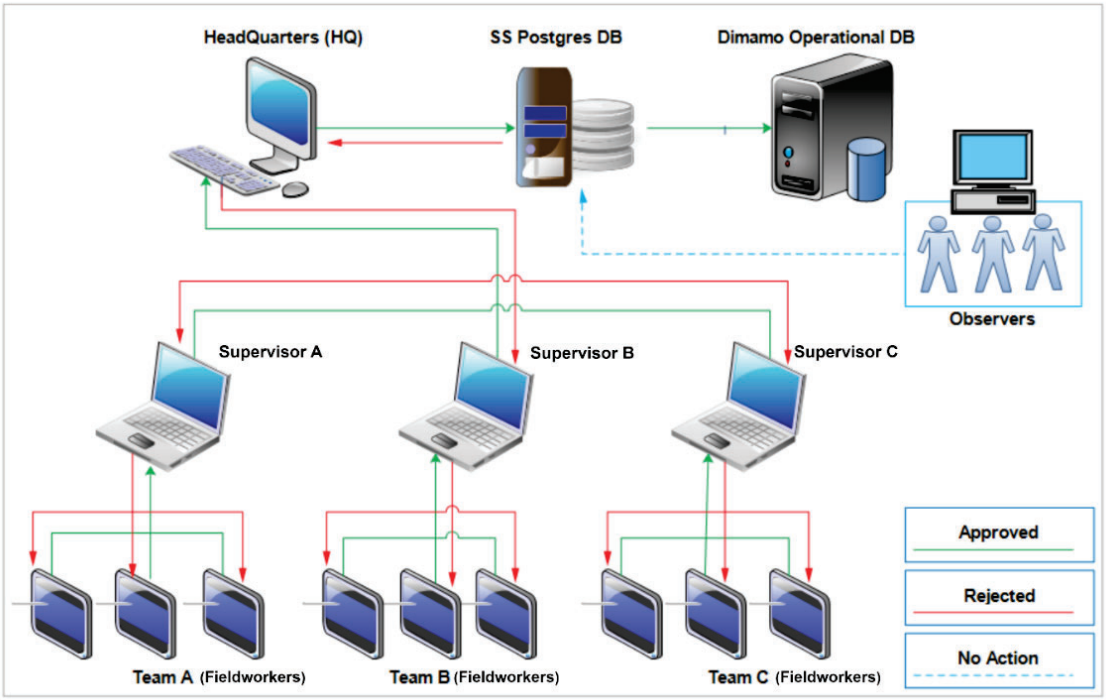


FIGURE 3.8: The representation data movement between data collectors, supervisors, and the server

Data transfer involves three levels: fieldworkers (lowest level), supervisors (intermediate level), and headquarters (highest level). At the fieldworker's level, data is collected from the field and transmitted via Bluetooth to the supervisor. Once at the supervisor level, basic data quality checks are performed and may be rejected and transferred back to the fieldworker if there are data quality violations. Otherwise, the supervisor synchronises the data to the Survey Solutions® based PostgreSQL database. At the headquarters

level, validations are performed by the automated data quality control system and reject the data back to the supervisor in the event of data quality issues. Transferring data between the supervisor and headquarters levels requires a virtual private network (VPN) connection. In the absence of network connectivity, the supervisor requires Wi-Fi access for data transfer. Once all stages of the data quality check have been completed, the data is permanently stored in the operational database. By adhering to a well-designed flow, we can leverage the power of data to make well-informed decisions and generate significant outcomes. Each step in the flow plays a vital role in guaranteeing the quality, dependability, and usefulness of acquired data.

3.8 Automated Data Quality Control System Workflow

An automated data quality control system (Figure 3.9) is a complex solution that employs cutting-edge technologies such as R-API, Bash programming, task scheduler, SQL, and PDI to ensure data quality throughout its lifecycle. It entails a variety of methods and strategies that detect and resolve errors, abnormalities, and inconsistencies in the data, to improve the overall quality. The system operates by applying predefined rules and algorithms to automatically execute a number of data quality checks. It can validate, among other things, data consistency, timeliness, completeness, accuracy, and validity.

The first step in the process is the electronic collection of data from the field by fieldworkers. The data shared is with supervisors as explained in section 3.7. Using virtual private networks (VPN), supervisors transfer data from the field to the Survey Solutions® database. The application programming interface (API) developed in the R programming language interfaces with Survey Solutions® to export recently collected data. The R-API is automated to autonomously export and load text data files into a central folder. PDI is used to read and transform data into acceptable formats ready to load into the MS SQL database. The second level of automation is done using a PDI job. This moves data streams into the MS SQL database, from which thorough data quality checks and validations are performed. The triggers and stored procedures programmed using SQL are configured in the database to validate the quality of the incoming data.

The quality of the data is checked to ensure consistency, validity, completeness, and accuracy. If any of these is compromised, an error message highlighting the nature violation is generated. This allows the fieldworkers to understand the nature of the problem in the data and fix it. The error messages, statuses, and other details generated during data quality checks are inserted into a database table (SSInterviewReport). The

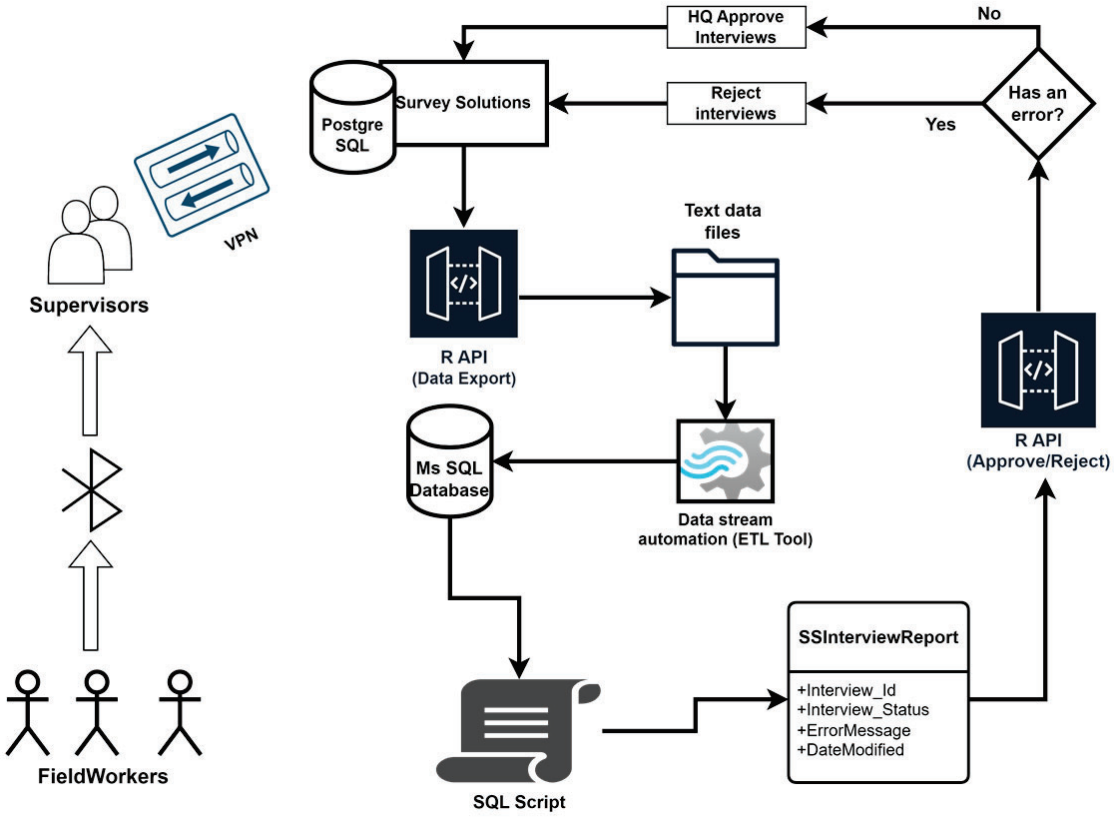


FIGURE 3.9: System workflow for a fully integrated and automated data quality control

status may be rejected or approved. The data is approved only if it meets predefined data quality standards, else rejected back to the fieldworker for resolving the issues. Using status, an R script autonomously rejects or approves the collected data. Data may be sent back and forward between fieldworkers and the Survey Solutions® database in real time for validation and quality checks. The data quality checks require no human involvement. By automating data quality checks, time and effort that would otherwise be spent on tedious, error-prone human data validation and cleaning operations are saved.

3.9 Data Quality Reporting Utility (Dashboard) Workflow

Data quality reporting comprises the systematic evaluation, monitoring, and communication of the quality of an organisation's data assets. It provides a comprehensive perspective on the data quality status, highlighting improvement areas, identifying problems, and monitoring progress over time. Data quality reporting promotes informed decision-making and ensures the dependability and credibility of data-driven efforts by

providing stakeholders with transparent and actionable information. The reporting process begins with the definition and establishment of data quality metrics and standards that are consistent with organisational objectives and requirements. These metrics encompass dimensions such as data consistency, timeliness, completeness, accuracy, and validity. Once the metrics are developed, data quality checks are performed to determine how well the data conforms to these standards.

Figure 3.10 presents essential steps from the development to the deployment of the data quality reporting system or dashboard. The collected data is synced into the Survey Solutions® server and exported to a zipped folder. To eliminate manual data export, an automated R script exports data from the PostgreSQL database using the Survey Solutions® API. The data is loaded into the central file data repository (folder) for further processing. The data is loaded as tab-delimited files which require further processing before loading into the relational database. Modern tools have the ability to accomplish such tasks with less effort. One such tool is PDI, which is the business intelligence suite that provides out-of-box solutions to transform, extract, and load data into databases and data warehouses (Mussa et al., 2018). The suite offers a wide range of services such as data integration, reporting, and analysis. PDI allows the creation of transformations and jobs. A transformation can be defined as a network or collection of logical tasks that perform a specific action. These logical tasks are often called steps. The steps are connected using arrowed links which give the direction of data flow. Jobs coordinate the activities performed in the transformation. A job defines the dependencies and flows for the order in which transformations can be run. One job may be used to run multiple transformations to activate data streams. One major advantage of this approach is the ability to automate jobs. It is possible to set a job to run automatically to ingest data into data sources. The PDI job was used to integrate and automate data streams from different data sources (text files and two databases). The job was scheduled to run automatically, twice daily at 7:00 in the morning and 12:00 midnight. The batch files were programmed to launch R Studio and PDI and configured to be executed by the task scheduler at scheduled times (7:00 and 12:00). Automation processes are explained in Chapter 4. Automation ensured that databases were well coordinated and updated with recently synced data. The recently collected data is mapped to historical data to generate data summaries from which reports or analytics can be drawn. The process ensures the validity and consistency of the data and also provides additional information needed for cube development.

A reporting database (data warehouse) is a specialised type of database that contains highly summarised data. The summarised data is contained in the dimension and fact tables that organise the data in an easy way for analysis. These tables can further be

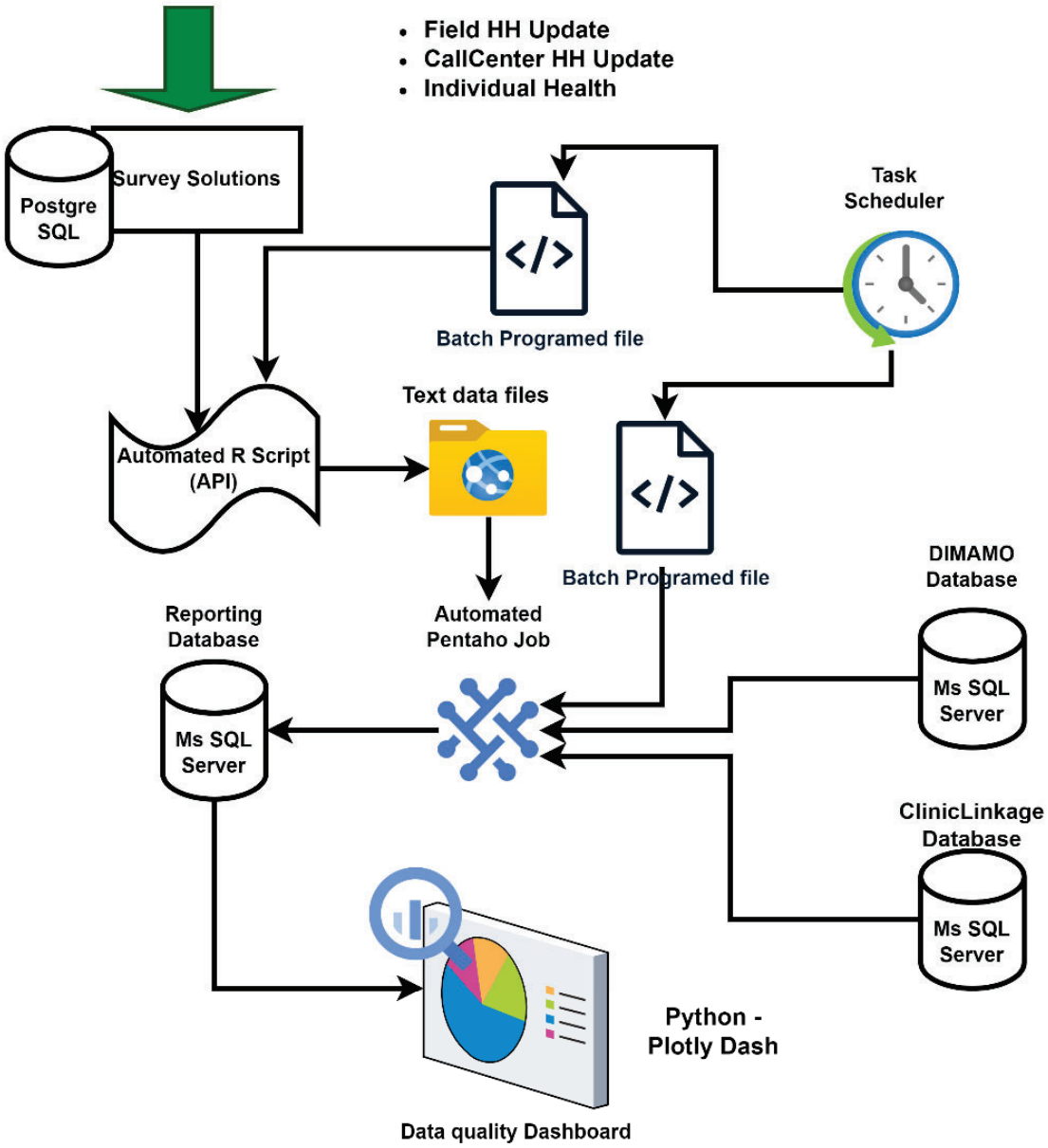


FIGURE 3.10: Diagrammatic representation of data reporting system work-flow

restructured and linked together using either star or snowflake schemas (the readers are referred to section 3.10 for more details).

The client-server communication approach (Sallow et al., 2020) is adopted to enable data communication between the server and the reporting or business intelligence system. The client (reporting system) and the server (data warehouse) must be in sync using the connection string for seamless communication. The connection string defines the IP address and a port through which data can be sent and specifies parameters to be passed to the database. To render data visuals, Python uses libraries such as

Plotly Dash, Seaborn, etc. Seaborn is built on top of Matplotlib to render data visualisation and is integrated with pandas. Like seaborn, Plotly Dash generates interactive, publication-quality charts and allows charts to be rendered on the browser with auto-refresh. The Panda data structure allows easy data manipulation and analysis. Numpy is considered one of the most powerful Python-based machine-learning libraries for performing high-intensity mathematical computations (Ziogas et al., 2021). Many libraries exist for data visualisation in Python. All these libraries are packaged together using anaconda distribution.

The Anaconda distribution is a tool that equips developers with thousands of libraries and open-source packages. It has a provision for an open-source integrated development environment (IDE) to build software and machine learning models. These include amongst others; Spider, Pycharm, Jupyter Notebook, etc. Anaconda's individual edition has a large user base of more than 25 million worldwide [1]. The Python web-based dashboard can be published and hosted on Heroku. Heroku is a cloud-based platform as a service that is compatible with several programming languages such as Python, PHP, Ruby, Java, Noje.js, etc.

The Heroku cloud platform manages software and hardware, helping programmers focus on perfecting the design of their applications. It has a powerful ecosystem designed for the deployment and running of modern applications. For a reporting or business intelligence system to be effective and serve its purpose, it must be hosted locally or on the cloud. Hosting permits the sharing of the system's contents with the intended users. The sole purpose of a reporting or business intelligence system is to transform data into information and present that information in a "language" that is easy to understand. Understanding the information leads to better and well-informed decisions.

3.10 Conceptual Model - Reporting Database

The reporting database is purely designed to maintain highly summarised data for analysis, from which high-quality decisions may emanate. The quality of data in both the operational and production databases leads to high-quality decisions (these databases are discussed in Chapter 4). This is particularly true because the data summaries in the reporting database originate from the operational and production databases. The reporting database "is a subject-oriented, integrated, time-variant, non-volatile collection of data in support of management's decisions" (Coronel et al., 2020). The data in the analytical database is integrated from the operational and production databases

[1] https://www.anaconda.com/products/individual

and optimised for faster retrievals. Integrating data from both data sources (operational and production databases) into the reporting database involved ETL processes. ETL extracts data from heterogeneous data sources, transforms and loads it into the analytical database (Luján-Mora and Trujillo, 2003). Data must be transformed or processed before loading into the reporting database. For example, convert values stored as characters (1, 0, etc.) into integer values for aggregation purposes, clean data through imputation processes, and normalising data. We follow the multidimensional paradigm (Luján-Mora et al., 2002) in designing a reporting database and divide the data to derive dimension and fact tables. Fact tables are mainly made up of measures, while dimension tables contain variables on which measures are based. These tables are organised into star schemas.

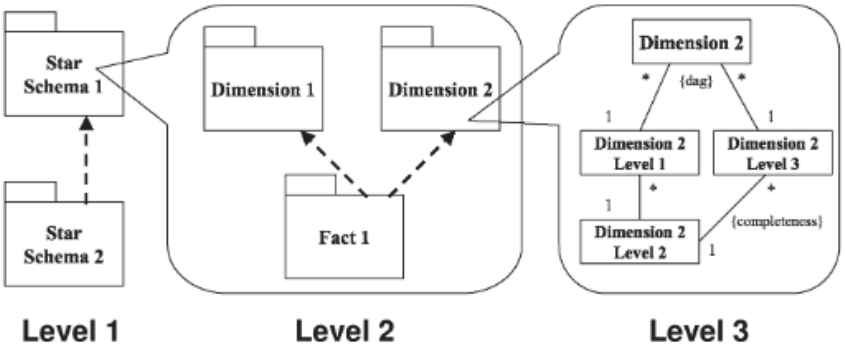


FIGURE 3.11: The three levels of a multi-dimensional model represented by means of UML packages (Luján-Mora and Trujillo Mondéjar, 2003)

In a complex data warehousing environment, multiple-star schemas may depend on each other (Figure 3.11). Level 1 looks at the dependencies between star-schema packages. Each package encapsulates facts and dimensions (level 2). Depending on the level of dimensionality or hierarchy, the star schema may extend into snowflakes. This is the case when the dimension extends into the other (level 3). Dimensions must be designed in such a way that many-to-many relationships are eliminated. Following this approach, Figure 3.12 presents a conceptual database model for the proposed reporting database. In designing this model, four techniques for optimising the data warehouse were considered. Such techniques are de-normalisation of fact tables, replicating and partitioning of tables, keeping multiple fact tables to represent diverse levels of aggregation, and normalisation of dimension tables (Coronel et al., 2020). Multiple fact tables according to Coronel et al. are created for semantic and performance reasons. The SQL codes running against such tables are likely to retrieve decision-support data faster. The figure presents three fact tables with a schema named "fact". Prefixed naming conversions (OPS and PROD) denote the source database from which the data is extracted. For example, table names prefixed with OPS denote that the data comes from

the operational database, while PROD suggests that the data comes from the production database. These facts tables contain the organisation's key interests so far as data collection progress and quality are concerned. All variables in the fact tables, except 'InterviewKey', were quantified for easier aggregation. This quantification is performed at run-time and updated periodically by the automated PDI Job to keep data current.

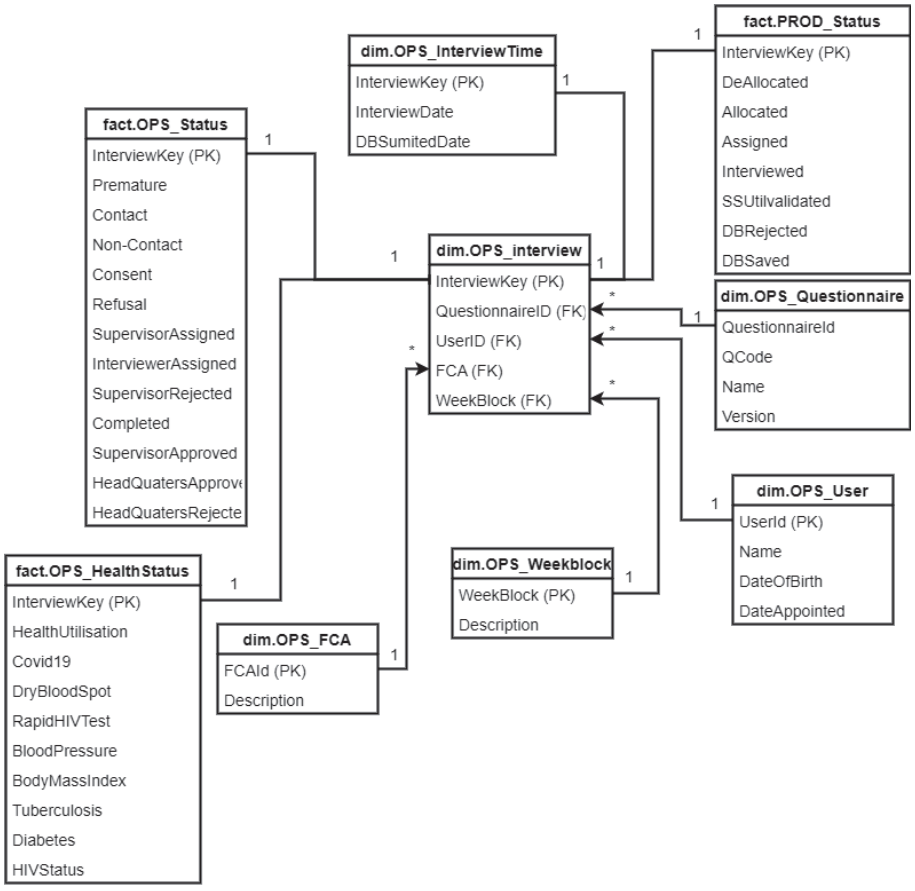


FIGURE 3.12: Conceptual model representing multi-fact tables implemented in the proposed analytical database

Another essential step is the normalisation of dimension tables. It permits easier navigation through the dimensions by end-users and achieves semantic simplicity. All dimension tables in Figure 3.12 are normalised to 3NF and have no transitive dependencies. 3NF produces relatively fewer data duplications in the database (Coronel et al., 2020). The dimension tables are grouped under a dim schema and consist of variables that are mainly of interest to the business analyst. These variables serve as a magnifying glass through which facts can be studied. They provide descriptive characteristics in relation to the facts. Common attributes such as questionnaire name, FCA, weekblock, and user name were identified to enable the data analyst to filter through the data for analysis. This multidimensional data view provides an opportunity for the researcher to analyse data from different angles or views. The multidimensionality presents a view

of interview data by questionnaire name, FCA, weekblock, and user name. In this way, the decision on progress and data quality may be made.

3.11 System and Data Communication Model

Data communication models provide a framework for analysing and understanding the flow of data between various components in a communication system. These models simplify the design and development of complicated communication networks, in addition to enabling efficient and reliable data transmission. Data communication involves the transmission of information or data between two or more devices connected via a medium, such as optical fibers, wireless networks, and cables. Figure 3.13 presents the communication model showing the interconnection between various devices and servers. Field communication is arranged such that fieldworkers share data with their respective supervisors via Bluetooth. Using upload/download links, the supervisor syncs the data to/from the Survey Solutions® server. This may be done only if there is 3G or higher network connectivity.

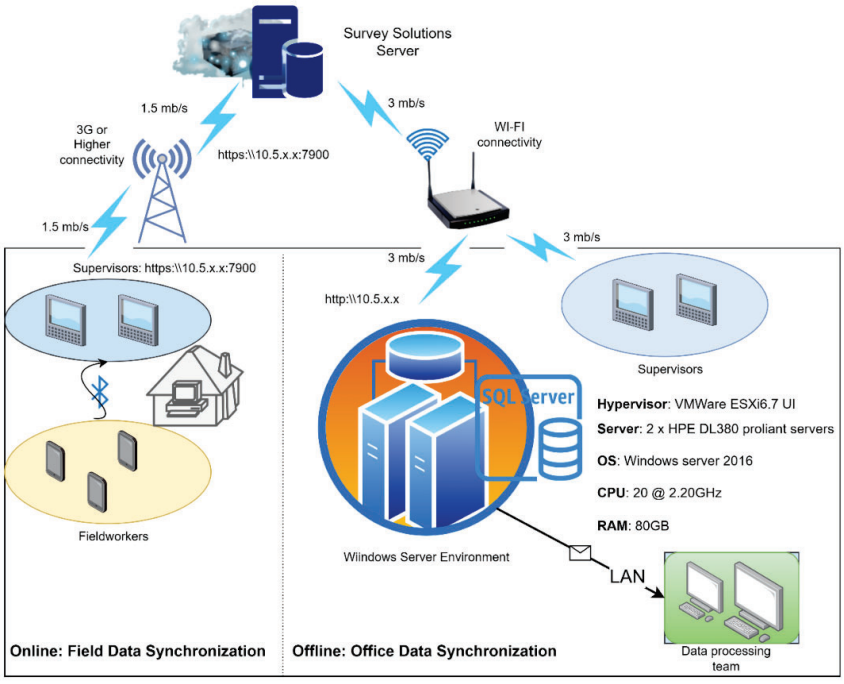


FIGURE 3.13: Data collection system's network communication model

The advantage of syncing data directly from the field is that the automated data quality control system can immediately validate the quality of data in real-time. The system can then make a decision on whether to reject or approve the data. The rejected data can be received immediately by the supervisor for correction of errors before resubmitting. The approach further strengthens the data quality assurance process and has shorter

data turnaround times. In the case where a connection is below the predefined network connectivity threshold ($3G$), data can be saved on tablets and later synced to the server. The minimum upload or download speed for a mobile network may not be below 1.5 Mb/s. Supervisors are required to connect to the Wi-Fi network upon arrival at the office to sync the data saved on the tablets to the server. The predefined minimum upload or download speed is 3Mb/s. The approach is crucial for areas with poor network connectivity or poor network infrastructure. The data processing team connects to Survey Solutions® and MS SQL servers using a local area network (LAN) to further process the data.

3.12 The Conceptual Framework of the Proposed System

This section discusses the conceptual framework for our study. According to Miles & Huberman (1994), "A conceptual framework explains, either graphically or in a narrative form, the main things to be studied; the key factors, constructs, or variables, and the presumed relationships among them. Frameworks can be rudimentary or elaborate, theory-driven or commonsensical, descriptive, or casual". Rudestam & Newton (1992) also define a conceptual framework as statements linking abstract ideas or concepts to empirical data. With reference to these definitions, we present the conceptual framework (Figure 3.14) for the study and outline how the components of the system are constructed and connected. The framework presents the roadmap for our study and depicts four main components of the system integrated to improve the quality of data. The design and development of the system were driven by the desire to achieve high data quality. A well-designed system facilitates the collection and management of good-quality data (Couture, 2013). The proposed system comprises IT infrastructure, an electronic data collection (EDC) system, automated data quality management, and data quality reporting systems or dashboards. The systems were designed and configured to effortlessly transfer, manage, process, and automate data quality checks. There exists a strong correlation between these four components. The EDC system and automated data management system are both dependent on the underlying IT infrastructure, which is critical for their hosting. Without properly configured IT infrastructure, it becomes impossible to automate and electronically collect the data. Electronically collecting data increases the chances of collecting data of good quality. However, that depends on the quality of the EDC system. A poorly developed EDC system may result in the collection of low-quality data, which may be a result of a lack of validation mechanisms. In ensuring high-quality data, robust and adaptive data management is required. The quality of the data management system can be evaluated using data quality metrics. The lower levels of data quality mean the inability of the data management system to

effectively deal with anomalies in the data. Summarizing from Figure 3.14, X → Y and X → W), therefore Y → W. On the other hand, W → Z and Y → Z, which means that W → Y. We can conclude that W and Y have effects on Z but both depend on X. The functionalities and key features of these components (Y, W, and Z) are explained in the following sub-sections. The configurations of IT infrastructure (X) are discussed in Chapter 4.

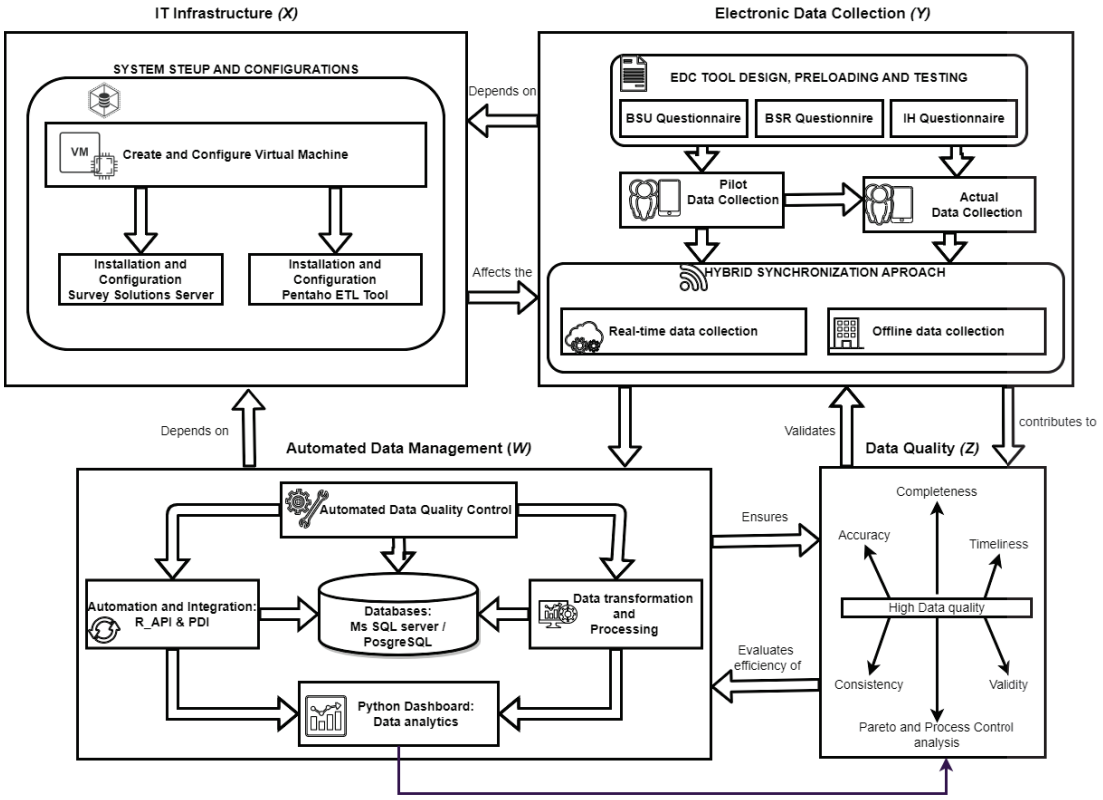


FIGURE 3.14: Detailed diagrammatic representation of the conceptual framework

3.12.1 Electronic Data Collection System (Y)

An EDC system handles the first level of data entry from the field. Data entry is performed using electronically developed questionnaires with about 1 366 questions validated using validation algorithms or C macros. Skipping patterns or logic algorithms give users the ability to navigate through the EDC system with ease. Both the validation and the skipping logic algorithms contribute to high data quality. With validation algorithms, data issues such as invalid data entries, missing values, inconsistencies, and general errors are flagged and reported for fixing. Data validation can also be performed through the design of the preloading feature. This can be achieved through the generation of an SQL script to extract data from the database and preload it to the EDC interface. The existing data can then be used to validate the currently collected data.

This guarantees the consistency and validity of the data between the two systems. Once the EDC system is fully developed, it must be piloted to check if the system meets the requirements. Conducting a pilot study in the context of HDSS is an essential measure to ascertain the functionality, dependability, security, and user-friendliness of the EDC system. This eventually aids the HDSS in its mission to gather and administer high-quality demographic and health data. Subsequent to the system's piloting phase, the system is deployed to the field for actual data collection. It is necessary to determine how the data obtained electronically should be transmitted to the server. There are two modes of data collection to consider: Real-time (Online) and offline. With online data collection, the data is synced to the database server in real time. This approach requires a seamless mobile network connection and offers great benefits with the provision of shorter data turnaround times. On the other hand, offline data collection uses a Wi-Fi network connection, and takes at least a day for the data to be available for analysis. The data can be saved to the device's local memory and synced to the database server when Wi-Fi connectivity is available. This approach works well for data collection in areas without stronger network connectivity. Both approaches offer immense benefits depending on the context under which they are deployed.

3.12.2 Data Quality Management System (W)

The data quality management system is a crucial element of any company's data strategy. It ensures that the data is of high quality, allowing companies to gain valuable insights, make informed decisions, and drive profitability. Organisations can improve their data capabilities and realise the full potential of their data assets by developing comprehensive data quality management processes and employing the relevant technologies. Technologies such as R-API and PDI can be deployed to automate data quality assurance processes and data transfer across systems. Automating data management processes has a number of benefits, which include the elimination of manual handling of data, consistency in performing routine data management tasks, minimisation of errors, reliability, and cost-effectiveness, improved efficiency, and minimisation of data turnaround times. The automation also involves the transfer and coordination of data between various data sources (e.g., MS SQL server database, PostgreSQL, CVS, or text files). Coordinating data across databases or data systems may be a tedious and time-consuming task, but automation makes this process much easier. Another important feature of data management is the integration of data from multiple sources. This process can introduce errors and inconsistencies if not properly handled (Akoka et al., 2007). Several tools can be used for integration (Biswas et al., 2020), and it may be necessary to automate the movement of data between the systems.

3.12.3 DATA QUALITY MEASUREMENT AND REPORTING (Z)

Data integration and automation permit an easy update of the data warehouse (DW). The DW contains the summarised data in dimension and facts tables (DivyaYadav and Choudhary, 2021). The facts table contains all historical data as facts that are of interest to the analysts, whereas the dimension table holds attributes supporting the facts. Tables can be designed in a way to present data in multidimensional arrays called cubes (Prevedello et al., 2010). The multidimensionality allows data analysis to be performed from different points of view. The specialised form of DW called data mart can be used to further aggregate data per division or section within an organisation. The data analytic system reads data from the data warehouse into data quality dashboards or systems. The system incorporates reports on the level of data quality. Data quality dimensions measure the extent to which the data represent the real-world entity. When investigating the extent to which data meets user data quality objectives, metrics such as completeness, accuracy, timeliness, validity, and consistency were considered. The selected dimensions contribute positively to the evaluation of data quality and data management systems. A good data management system accurately identifies data quality issues. This uncovers dependencies and patterns in the data and recommends an action plan. Python libraries were used to graphically represent data quality issues and measurements. The system was designed to identify, monitor, and measure the quality of data in the database.

3.13 Chapter Summary

System models are essential in the field of science and technology due to the provision of a simplified picture of actual systems, processes, or occurrences. The value of system models resides in their capacity to facilitate the comprehension, analysis, design, and optimization of complex systems. Modeling data quality systems enables the designing and building of a good data quality system. With a robust data quality system in place, the quality of data is more likely to improve. This chapter presented the models to understand; the system to be built, automated data quality framework, data quality metrics, and statistical approaches to identify and control data quality concerns. The conceptual model was created to give a step-by-step depiction of the system construction processes. The communication model was subsequently developed to illustrate how the data is synchronised between the field and the database server. These models were used for data quality control processes and allowed the identification of system inefficiencies and suggestions for improvement. Sharing the models with the research community enables the understanding of the system construction and the assessment for validity

and reliability of the current study. The models can also serve as guides for organizations intending to implement a comparable system for data quality management.

Chapter 4

Philosophical Positioning and Analytical Procedures

4.1 Introduction

The chapter on research methodology is an essential part of empirical research since it provides a detailed framework for conducting rigorous and methodical investigations. Empirical research involves the collection and exploration of data to answer research questions and develop empirical insights. This chapter describes the strategies, methods, and procedures used to ensure the reliability, validity, and generalisability of the study findings. In addition to establishing the reliability and validity of the empirical study, the research methodology chapter acts as an invaluable resource for other researchers on the subject. By providing a thorough explanation of the research methods, the reader may evaluate the study's rigor and quality with greater objectivity. This chapter also serves as a guide for new researchers, introducing them to various research procedures in the current area of research, and supporting them in planning their own empirical investigations.

The rest of the chapter is organised as follows. Section 4.2 discusses paradigms and the philosophical position with special emphasis on research methodologies, theoretical perspectives, epistemological approaches, and analytical and research methods. Section 4.3 presents the experimental Layout while Section 4.4 provides an overview of the IT infrastructure used to set up the study experiment. In Section 4.5, we discuss the data collection platforms and related technologies used to host the electronic data collection (EDC) system. Section 4.6 presents the steps involved in system automation and integration, while Section 4.7 discusses data quality assurance and procedures, which detail

the data cleaning engine for the proposed 3TTDQM framework. Data stores used for this study are presented in Section 4.8 while Section 4.9 concludes the chapter.

4.2 Paradigms and Philosophical Position

Scientific research philosophy is the general belief system of how the researcher can obtain new knowledge. Saunders et al. (2003) relate research philosophy to the development of knowledge and its nature. It is essential to follow proven methods in the development of knowledge. The philosophy informs the choice of research approach, research strategy, problem formulation, data gathering and processing, and analysis (Žukauskas et al., 2018). Embedded within the research philosophy are important assumptions about the world-view point. These assumptions underlie the research methods and strategies to apply in conducting research. Research emanates from the assumptions a researcher makes (Hitchcock and Hughes, 2002). Therefore, researchers may make different assumptions about knowledge acquisition and the nature of truth (Cohen et al., 2007). The assumptions about how knowledge should be acquired inform the choice of the research paradigm. That is, the scientific research paradigm contributes to the definition of scientific research philosophy.

The paradigms of scientific research consist of ontology, epistemology, methodology, and methods (Saunders et al., 2003). The choice of methodology according to Holden and Lynch (2004) should align well with the philosophical position. The research paradigms are considered by Cohen et al. (2007) as a support structure embodying beliefs, perceptions, and consciousness of practises and theories used to conduct scientific research. The research paradigm is the way of thinking about research, the accomplishment of processes, and implementation methods (Gliner et al., 2016). In other words, it is not necessarily methodology that details the processes of conducting research but rather a philosophy. Paradigm is the methodological and theoretical regulations adopted by the scientific community at a particular stage of the development of research to model and solve scientific research problems. Figure 4.1 shows the links between epistemology, theoretical perspective, research strategy, and methods.

These research elements according to (Crotty, 1998) inform one another. The choice of epistemology determines the theoretical perspective, which in turn leads to the selection of the research strategy. The research strategy, on the other hand, informs the choice of methods. The subsections below discuss these important aspects of research and present the philosophical position of this research.

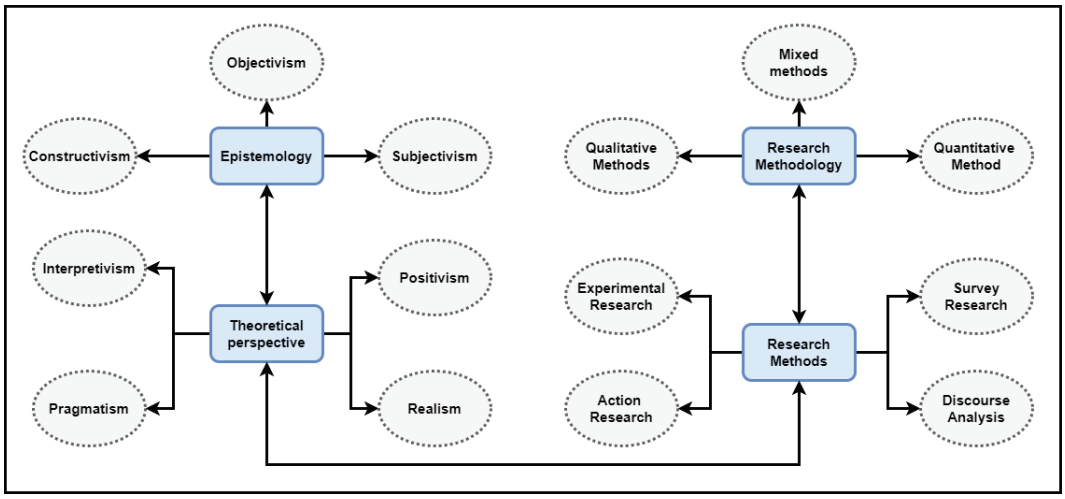


FIGURE 4.1: Linking epistemology, theoretical perspective, research strategy, and research methods

4.2.1 Research Methodology

The research methodology presents a systematic and organised approach to conducting research. It describes the rationale for the choice of research methods. Easterby-Smith et al. (2008) described research methodology as the exploration of different situations by combining different techniques in solving a specific research problem. It serves as a support structure for research and is influenced by theoretical perspectives (Walter, 2006). Some of the major types of research methodologies according to (Al-Ababneh, 2020) are survey research, ethnography, discourse, experimental research, feminist standpoint research, analysis, phenomenological research, action research, grounded theory, heuristic enquiry, etc. We choose to discuss survey and experimental research methodologies particularly because they have a bearing on our study. Survey research is a type of non-experimental quantitative research that includes personal interviews, phone surveys, normative surveys, and questionnaires (Mohajan, 2020). It is the systematic collection of data from respondents to predict and understand some attributes of the population of interest's behaviour. Survey research explores the relative incidences, distributions, and interrelationships (Kerlinger and Lee, 2000) and provides a valuable source of fundamental scientific knowledge. On the other hand, experimental research attempts to establish a link between dependent and independent variables (Polit and Beck, 2008). It is the process of organising a study to achieve specific objectives. Experiments involve making an observable and measurable change in one variable and then observing how that changes other variables (Chen, 2011). Its purpose is to test a hypothesis to establish cause-and-effect relationships (Ary et al., 2010). Experimental research offers a wide range of advantages (Mildner, 2019): the possibility of research replicability and repeatability, causal relationships can be easily determined, and the ability to create

conditions that are difficult to observe in natural settings or would necessitate an inordinate amount of time to observe.
Our study designs electronic data collection (EDC) and automated data quality control systems and establishes an experimental environment to investigate the quality of data. EDC system was used to collect survey data that will be used for data quality evaluation. The survey was carried out by a group of fieldworkers who physically collected data from the households in the surveillance area. The data was collected electronically and synced to the database using a WIFI network connection (This is extensively discussed in Section 4.5). In evaluating the data quality, We seek to find factors that affect the completeness, accuracy, timeliness, consistency, and validity of the data.

4.2.2 Theoretical Perspectives

Crotty (1998) defines the theoretical perspective as "the philosophical stance informing the methodology and thus provides a context for the process and grounding its logic and criteria". Johnson and Clark (2006) highlighted that the philosophical commitment made through the choice of research strategy significantly impacts what is to be understood and investigated. In other words, the practical consideration is influenced by the philosophy adopted. The major influence may possibly be the researcher's view of the relationship between knowledge development processes and knowledge itself. The researcher who gathers facts about the phenomena quantitatively may have a different view from the other who generates knowledge qualitatively. Saunder et al. (2009) argued that no philosophy is better than the other, but depends on the questions to be answered or hypothesis to be tested. The pragmatist view also states that the choice of philosophy is determined by the research question.

The theoretical perspective represents the philosophical position that substantiates the main assumption of the chosen methodology. It informs and determines the research methodology (Crotty, 1998). Collis and Hussey (2003) classified theoretical perspectives or paradigms into positivism and interpretivism. On the other hand, Žukauskas et al. (2018) categorized paradigms into positivism, interpretivism, realism, and pragmatism. Positivism states that observation of the phenomenon leads to the production of credible data from which knowledge can be generated. With positivism, the assumption made by Remenyi et al. (1998) was that the researcher is independent and does not impact the research subject. The positivist viewpoint is that the world can be understood objectively, in which the researcher dissociates himself from what is studied. According to Easterby-Smith et al. (2008), positivism is closely associated with methods that collect numeric data such as experiments and observations. The positivist believes that the

truth can be observed, measured, and described (Park et al., 2020). Inverse to Positivism is interpretivism research philosophy, which states that one way to understand the social world is to subjectively interpret it (Žukauskas et al., 2018). In this case, the researcher should be seen playing a specific role in that which is researched. Therefore, the researcher and researched are somehow interactive. With interpretivism, the researcher uses small data samples and carefully assesses them to fully understand the attitudes of the larger population (Kasi, 2009) while on the other hand, positivist generates knowledge from larger samples (Mohajan, 2020; Park et al., 2020). According to Lancaster (2005), realism is rooted in the principles of interpretivism and positivism. Pragmatism considers practical results and facts (Lancaster, 2005). Alghamdi and Li (2013) argued that pragmatism belongs to no philosophical system and reality. So researchers have a wealth of choices to make depending on the research question. Choices include but are not limited to suitable methods, procedures, and techniques that best meet the researcher's needs. Pragmatists claim that the research philosophical choice is mainly determined by the research problem or question. Wiersma and Jurs (2008) stated that the researcher's position can significantly impact the outcomes of the research. People's perceptions of the world have an impact on research procedures and design (Collis and Hussey, 2013).

Our viewpoint is well aligned with that of (Kuhn, 1970) who believes paradigm models problem and provide solutions to the community of researcher and it asks questions such as: 1) what is to be studied and scrutinised, 2) the kind of questions to be asked and probed for answers in relation to the subject, 3) how these questions are to be structured, 4) how the results of scientific enquiry should be interpreted, 5) how experiment should be conducted, and 6) what tools and equipment are available for setting up the experiment. These questions are fundamental to conducting successful research and are addressed in this research from the positivist point of view. The paradigm aligns well with the researcher's beliefs or principles about how the phenomenon can be investigated and the knowledge derived from it. The positivist gathers knowledge through data quality measurements to evaluate the phenomenon of interest.

4.2.3 Epistemological Approaches

Epistemology is the branch of philosophy that constitutes what is acceptable knowledge in the field of study (Saunders et al., 2003). Brewerton and Millward (2001) defined epistemology as an assessment of what sets apart a reasonable assurance from an opinion. Its impact on the methodology and data collection methods is enormous (Hitchcock and Hughes, 2002). Muchanga (2020) argues that there cannot be a methodology without a

research philosophy. The choice of methodology is informed by the philosophical stance. Epistemology is concerned with the creation of knowledge, the distinction between bad and good knowledge, and the description and representation of reality (Hatch and Cunlife, 2006). Knowledge can be created constructively, subjectively, and objectively. The underlying foundation for knowledge creation is the research paradigm. The research paradigm is the common set of viewpoints and agreements shared amongst scientists on how best research problems should be understood and addressed (Kuhn, 1970). The choice of paradigm determines how knowledge should be created. The constructivist emphasizes the significance of beliefs, skills, and knowledge in learning and research (Garbett, 2011). It suggests that people construct their knowledge and understanding of the world by encountering things and reflecting on those encounters. In constructivism, researchers create knowledge through the understanding and interpretation of phenomena in historical and social contexts, which is commonly regarded as a method for conducting qualitative research. In subjective knowledge creation, the researcher chooses to interact with what is researched to gain knowledge (Žukauskas et al., 2018). From a subjectivist point of view, "social phenomena are created from the perceptions and consequent actions of social actors" (Saunders et al., 2009). Other researchers may choose to believe that a researcher should not have a direct influence on what is researched. That is, a meaningful truth exists in reality apart from consciousness (Crotty, 1998). This means that the attitude of the researcher does not affect the research subject and in that case, the researcher is objectivist. Saunders et al. (2003) describe objectivism as portraying "the position that the social entity exists, in reality, external to social actors concerned with their existence". Tashakkori and Teddie (1998) differentiated between subjectivism and objectivism by stating that, "the knower and the known must be interactive" while another case may be that "one may be more easily apart from what is studied". Here, the former is a typical example of subjective reality and the latter is the representation of objective knowledge creation.

From the discussion, we choose an objective path to uncover knowledge about what is to be studied. This is in line with our view that data quality exists in reality independently of the researcher. Our view is that if we fine-tune data quality systems, then the data quality will improve over a period of time. Thus, data quality becomes a dependent variable as it depends on other factors such as the quality of the data collection system, data management system and business intelligence system. In investigating the relationship between data quality and data systems, the researcher remains objective and will not influence the results of the experiment in any way. This aligns well with our philosophical stance or theoretical perspective in finding the truth about the phenomenon. Our stance is based on a set of beliefs about reality that influence the questions we ask and the answers we get as a result.

4.2.4 Positivist View of Data Quality Management Processes

We view data management processes as measurable objects for assessing data quality. Data quality management is a comprehensive practice that begins with an initial evaluation of data quality and has become increasingly crucial (Williams and Tang, 2020). Data quality management takes into account people, data, technology, and processes, with data at the core. All these elements must work in an integrated manner to achieve high-quality data (Mahanti, 2019). Poor data quality arises from inconsistencies, missing data values, errors, and other related issues. These may have a detrimental impact on organisational growth. Even minor data quality issues can accumulate over a period of time, causing process inefficiency, a loss in revenue, and compliance failure (Kiel et al., 2020). To address such issues, theories, and frameworks were developed to effectively counteract their impact.

4.2.5 Analytical and Research Methods

Analytical and research methods are techniques used in data collection and analysis to gain a better understanding of the topic. According to Mackenzie and Knipe (2006), the research question and paradigm determine the methods of data collection and analysis. These methods include qualitative, quantitative, and mixed methods. Given the research question and paradigm, the researcher can adapt the most suitable methods to answer the research question. Qualitative research is more reliable compared to quantitative and mixed methods (Mohajan, 2020). The reliability, validity, and generalisability of quantitative methodology are used to assess its rigor and strength (Choy, L.T., 2014; Morris and Burkett, 2011). This work considers a quantitative research approach to test objective theories by exploring interrelation among variables in order to advance the scientific knowledge base. The choice of this approach was informed by other studies in the literature that addressed problems in similar settings (Ahmed et al., 2018; Njuguna et al., 2014; Ley et al., 2019; Carlbring et al., 2007; Thindwa et al., 2020). We perform measures to define reality, thus investigating the quality of data with respect to completeness, accuracy, timeliness, validity, and consistency. Analyses are performed on a larger sample (more than 21,000 households and 125,000 individuals) to measure the relationship between variables and the effects they have on data quality. One of the advantages of quantitative analysis is that it has the potential to better generalise the results of the investigation in a larger sample to the broader population (Leedy and Ormrod, 2010). Quantitative research findings can be explanatory, confirmatory, and predictive (Creswell and Creswell, 2017), and its goal is to create and apply theories, mathematical models, and hypotheses/propositions about phenomena (Given, 2008).

4.3 Experimental Layout

Figure 4.2 provides an overview of the steps taken in building the entire system for data quality management. One component of the system is EDC, which was built using Survey Solutions Designer that incorporated C programming language for data validation. EDC applications were installed on Samsung Galaxy Tab A10 tablets and allowed users to log in and access the data and modules or tabs for data collection. The New Interview tab allowed the users to access questionnaires to proceed with data collection. After data collection, the data in the completed questionnaires moves to the Completed tab. The Rejected tab contains the data that failed the validation test, requiring user intervention to resolve the issues. These modules are connected to the PostgreSQL database through the synchronization point or database server IP address. This is to allow data to be sent to the database over a WIFI communication network.

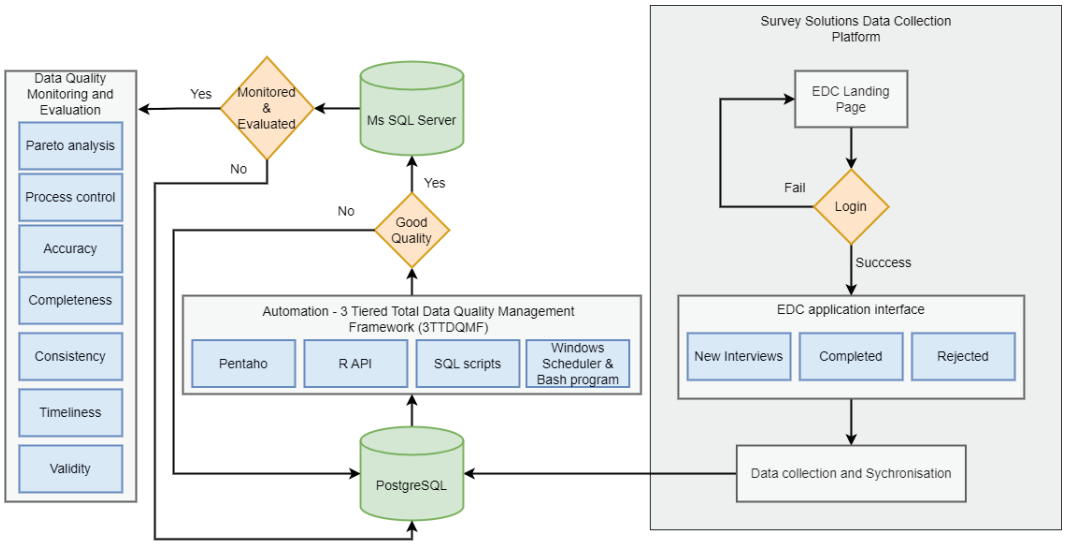


FIGURE 4.2: A Process diagram illustrating the sequential actions undertaken to conduct an experiment

The PostgreSQL database contains operational data that must be subjected to thorough quality checks. The quality checks were performed through 3TTDQMF (refer to 4.7 for more details), which was automated to eliminate issues associated with manual data quality procedures. The automation was carried out using tools such as R-API, SQL, Pentaho, Windows task scheduler, and Bash Programs (refer to Section 4.6 for finer details). The tools were systematically configured to autonomously export, validate, and clean the data. Data cleansing was performed before to importing it into the Ms. SQL Server database version 14.0.2037.2. There was a need to continuously monitor and evaluate the quality of data in the database, hence the system was developed. The system was developed using Python-based Plotly Dash, which enabled the use of statistical analysis techniques such as Pareto analysis and Process control charts. In addition, the

system included data quality indicators (accuracy, completeness, consistency, timeliness, and validity) to illustrate the extent of data quality concerns present in the database. The experiment was carried out in the IT infrastructure described in Section 4.4.

4.4 Information Technology (IT) Server Infrastructure

The current HDSS in which the experiment for this study was set up has a state-of-the-art hyperconverged infrastructure (HCI) with 2 x HPE DL380 Gen10 ProLiant servers and 1 x HPE DL360 Gen10 ProLiant server. All hosts run VMware ESXi 6 with the integration of VMware vSAN Academic 6.7. Storage devices, both local and direct-attached, are pooled together in a VMware vSphere cluster so that all hosts in the cluster have access to the same unified data store through VMware vSAN. A 20 TB vSAN storage pool was configured on RAID 5 technology for scalability, high availability, and data security. On the other hand, VMware vSphere Enterprise Plus 6.7 was configured to provide high availability regardless of the hardware or software system layer and allowed users to remotely run their virtual machines (VMs), even with sophisticated operations. The virtualized environment is made up of multiple VMs that provide different specialised services. One of the VMs is devoted to data services and hosts Microsoft SQL Database Server 2017 Enterprise Edition. This is the database in which the manually collected data is permanently stored.

The database is accessible to data capturers and data managers, who are responsible for manually processing and quality control data. This IT server infrastructure provides a solid foundation for the proposed information system or data system. Different VMs were dedicated to various data services. For example, DBsrver is configured to host MS SQL server-based operational and analytical databases, SSsrver is used to manage data collected through the proposed Survey Solutions® platform with a PostgreSQL database, and SSsrver-Dev is particularly used for piloting the proposed EDC systems. The unified view of data across these VMs is essential for accurate reporting and decision-making. The proposed system consists of several data sources, suggesting that some level of data integration is required to maintain the quality of the data (Ibrahim et al., 2021). Data integration, process automation, and data quality assurance are crucial components of the proposed data system. These components facilitated the integration and automation of data and operations across several VMs. The processes were carried out to guarantee optimal data quality.

4.5 Electronic Data Collection Platform and Technologies

A data collection platform refers to an integrated set of software tools for managing survey data. The platform offers a wide range of benefits such as data quality control features, capturing of all types of data, integration with third-party applications using application programming interface (API), customisation of the dashboard for reporting, users and survey data management features, support for relational databases, scalability, and advanced security features, support for online and offline data collection. Several data collection platforms, such as Red Cap, Census and Survey Processing System (CSPro), Qualtrics, Open Data Kit (ODK), and Survey Solutions® exist to host EDC systems (Baschieri et al., 2019). Each platform is characterized by ease of use, data protection, and advanced data management features. The Google App Engine allows ODK to scale up in the event of increased traffic (Sharif at el., 2016). ODK is freely available and has built-in data encryption features for data protection. The strength of ODK is in its ability to enforce data quality by not allowing progression to the next section if all required questions in the current section are not answered.

4.5.1 The Choice Electronic Data Collection (EDC) Platform

Selecting an appropriate platform is one step to ensuring high-quality data, and the selection criteria must be based on (Zhongming et al., 2019): 1) server and data security, 2) pricing, 3) adequate case management, 4) support for complex skipping and validation algorithms, and 5) Android operating system.

1. Server and data security – several platforms offer an option for both online and offline hosting services. Hosting data online is susceptible to security threats and discourages most organizations from collecting confidential data. The platform must ensure a secure connection between the tablets and the server for data exchange. With offline or local data hosting, minimal server and data security is required as the systems are not exposed to the Internet, hidden from cyber-attacks. Some organisations prefer local servers for data storage and transfer (Zhongming et al., 2019) and to comply with the Protection of Personal Information Act (POPIA).

2. Pricing – Some open-source platforms are freely available and developed by a multilateral organisation to support high-quality data systems. Some studies have shown that paid version platforms offer no substantial benefits compared to open-source versions. Open-source platforms relieve the financial burden of an organisation running on a limited budget while ensuring sustainability and quality of data.

3. Adequate case management – an essential feature of any platform is its ability to manage the collected data and users. The ability of interviewers, supervisors, and data managers to complete, reject, and approve interviews ensures a collaborative approach to data quality management. In assessing data quality, the platform must have a reporting feature to track progress and report erroneous measurements.

4. Support for complex skipping and validation algorithms – Skipping and validation algorithms play a vital role in data quality management. Skipping questions that are irrelevant reduces the possibility of human error, thus enhancing data quality. Validation algorithms, on the other hand, filter and examine the data in accordance with the established quality standard. This enables the cleaning of data at the point of capture.

5. Android operating system – The proliferation of Android-based devices makes them cheaper alternatives to the iOS operating system. Android provides greater customisation choices and flexibility than iOS. This enables the customisations that are tailored to a specific data collection needs. One advantage of Android for data collection is its larger ecosystem of third-party applications and services, which can be beneficial in collecting high-quality data.

Baschieri et al. stated that Survey Solutions® offers a wide range of benefits such as the ability to link questions, features for reporting and aggregation of data, integration of validation algorithms, user-friendly questionnaire designer software, an interface for data management and survey administration, technical support, freely available for use, and its support for Android-based devices. It also caters for the integration of third-party applications (R applications, Python, and *C#*) for data processing and analysis.

Survey Solutions® provides support for both real-time and offline data collection. The former requires good network connectivity and has shorter turnaround times compared to the latter. The latter works well in areas with unstable or no network connectivity. Using both offers great advantages when considering areas with/without network connectivity.

4.5.2 Survey Solutions® Configuration Requirements on Standalone Server

Survey Solutions® is a free software platform designed and developed by the Data Group of the World Bank. The software was developed as a platform to host data collection systems and enables easy data capture and management. The Survey Solutions® server

is available and downloadable from the World-Bank website[1] for standalone installations. The prerequisite for configuring the server is the installation and configuration of the PostgreSQL database server[2]. The minimum requirements for the installation of both servers (Survey Solutions® and PostgreSQL) on the virtual machine (VM) or physical server are as follows:

- Survey Solutions® version 22.06
  - CPU: 4 physical cores, 64-bit
  - RAM: 16GB RAM
  - DISK: 500GB SSD
  - OS Microsoft Windows Server 2016 or newer
  - Microsoft.Net Core 3.1 runtime
- PostgreSQL version 10
  - 64-bit Windows Platforms : 2016, 2012 R2 & R1, 7, 8, 10
  - 32-bit Windows Platforms : 2008 R1, 7, 8, 10
- Technical Requirement for Tables
  - Android 8.0 or higher
  - RAM: Minimum 1.5GB or higher
  - Storage: 8GB of flash memory storage (for offline data storage) or more
  - Wi-Fi module (for software setup, upgrades, and synchronisation)
  - The 3G/4G connectivity module is required for synchronisation from the field.

To configure communication between the Survey Solutions® server and the PostgreSQL database server, port configuration is required. The installation of the PostgreSQL database server version 10 precedes that of the Survey Solutions® server. The PostgreSQL server version is selected only on the basis of its compatibility with the Survey Solutions® server version 22.06. The Survey Solutions® server was hosted as a web application using internet information services (IIS). The access to the server is solely through the synchronisation point (server IP address) using the internet browser. Tablets are configured to send and receive data to and from the Survey Solutions® server that runs the PostgreSQL database as the back-end. The action is accomplished by configuring the synchronisation point on each tablet. The synchronisation point enables the downloading and upload of data to and from the EDC application.

[1] https://mysurvey.solutions/download
[2] https://www.postgresql.org/download

4.5.3 Development of EDC System and Data Quality Assurance in Survey Solutions®

The World Bank provides a separate development environment or designer for building EDC applications. The designer is exclusively designed to build EDC applications and provides capabilities to program multi-select, single-select, open-ended, geographic positioning system (GPS) points capturing, date, and numeric questions (Thysen et al., 2021). It subsumes a built-in data quality assurance feature to write C#-based macros or validation and skipping pattern algorithms. Validation algorithms ensure the completeness, accuracy, consistency, and validity of the data. The designer allows the system to be designed and built in a way that error messages are thrown if inconsistent or inaccurate data values are captured. This alerts the user of any data quality violation at the point of data entry. Skipping patterns also add another layer of quality control. Patterns play a crucial role in ensuring proper navigation between different sections of the application. In a typical data collection environment, there may be certain questions that may be unnecessary to ask specific participants. For example, pregnancy-related questions are irrelevant to male participants. Hiding such questions greatly enhances data quality, thus circumventing erroneously capturing pregnancy details for male participants.

Another exciting feature of Survey Solutions® is the capability to pre-load the data. Data preloading allows existing or historic data to be loaded onto the EDC application prior to field deployment. Basic details such as GPS readings, names, last names, date of birth, gender, civilian IDs, parent details, etc., may be prepopulated on the application to hint data collectors of eligible participants. Data preloading may also enforce data quality. For example, consistency checks can be made between the currently collected data and the known information about the participants. The known information must be hidden from the enumerator to ensure validity and trustworthiness. For example, if the participant's known gender is male and a different gender is captured in the current data collection, then an error message reporting inconsistency must be thrown. The reader is referred to (Appendix A) for an example of an SQL script used to generate data for preloading. The script extracts and formats existing data from the database. The extracted data is then loaded into text files and uploaded to the EDC system hosted on the Survey Solutions® server to preload the desired fields.

Data preloading, which is the process of preloading certain information into an EDC system prior to data collection, provides various benefits in relation to data quality management; 1) Time-saving: Preloading data saves time by preventing data collectors from entering the same information repeatedly for each data collection round. This

minimises the likelihood of manual data input errors and saves time. Instead of duplicating the entry of existing data, data collectors might focus on collecting new or unique data points. 2) Standardisation: Data preloading guarantees the standardisation and consistency of the collected data. Organisations can impose predetermined formats, classifications, or alternatives by preloading certain data fields. This minimises data discrepancies caused by human data entry and improves the quality and dependability of the data obtained. 3) Contextual guidance: Context and guidance can be provided to data collectors throughout the data collection process by preloading pertinent data or instructions. This can contain reference materials, prepopulated instructions, and even past responses to the same or comparable data collection situations. Such an approach can help data collectors in collecting consistent and accurate data. 4) Complex data structures: Preloading can ease data collection processes that involve intricate or interrelated data structures. For instance, if many data fields are interdependent or have predetermined relationships, preloading can populate the relevant fields with the supplied data. This guarantees data integrity and minimises the likelihood of data missingness or inconsistency.

4.6 Automation and Integration

Automation is the process of arranging a system to operate independently and in a timely fashion without user participation (Tussyadiah, 2020). The procedure enables the transfer of data in real-time and database updates. This research aims to automate data quality control processes, data export, download, and migration from a Survey Solutions® -based database to the MS SQL Server database. This section discusses automation procedures and is organised as follows: Subsection 4.6.1 presents an overview of the automation process and discusses steps in the automation procedures, Subsection 4.6.2 discusses automation processes using R scripts, while Subsection 4.6.3 presents automation and integration procedures using the Pentaho data integration tool. Lastly, Subsection 4.6.4 presents the automated scheduling of batch files in the Windows environment.

4.6.1 Eye-Bird View of the Automation Processes

System automation offers a wide range of benefits such as the elimination of human-generated errors (Kaya et al., 2019; Penttinen et al., 2018), improvement of user productivity and quality of work (Ratia et al., 2018), saving of labour and time (Gejke, 2018), and improving the precision and accuracy (Leshob et al., 2018; Kaya et al., 2019) of the tasks carried out. Automating processes increases the performance, availability, and

reliability of the services rendered. Figure 4.3 shows the processes of automating export, download, and loading data from Survey Solutions®based data into the MS SQL server destination database.

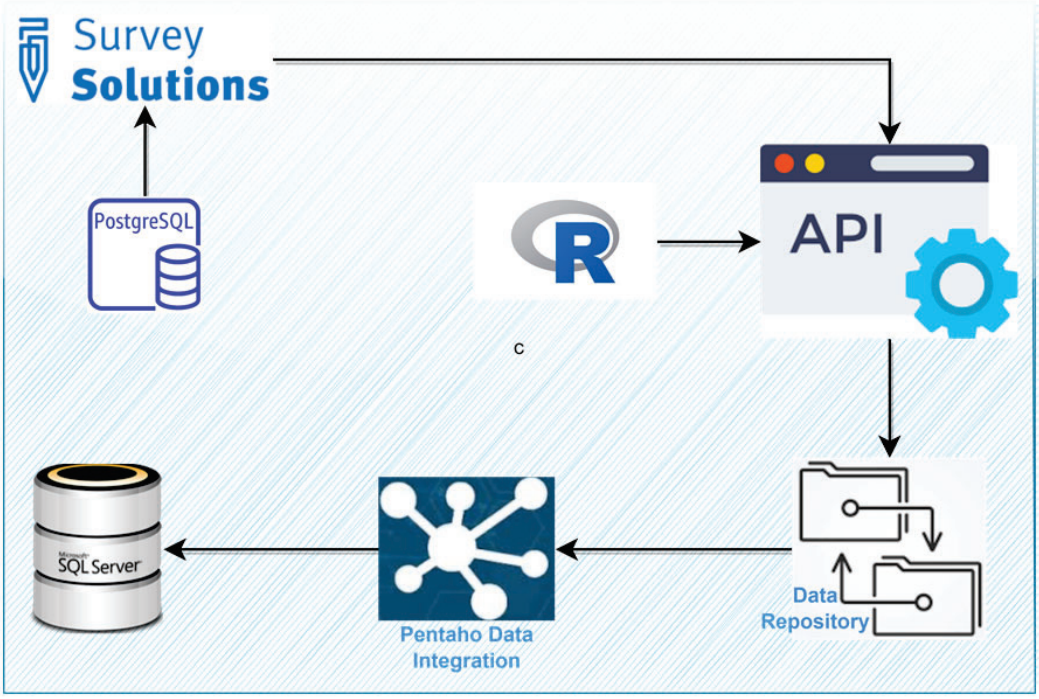


FIGURE 4.3: Diagrammatic representation of the proposed data flow and automated processes

Survey Solutions® uses PostgreSQL database as its main data source. The data in the PostgreSQL database cannot be accessed directly but through the data export feature. The supported file formats for export are SPSS, tab-delimited, and Stata. The Application Programming Interface (API) developed in the R programming language facilitates the exporting, downloading, and loading of data into a data repository. Refer to Subsection 4.6.2 for the application of R API and its automation process.

The data repository in Figure 4.3 holds the data files exported from Survey Solutions® which are ready for processing and integration. The Pentaho Data Integration (PDI) tool accelerates data management activities such as cleaning, transformation, processing, integration, extraction, and data loading (Pulla et al., 2016). Automating PDI ensures effortless movement of data from the data repository into the MS SQL server database (see subsection 4.6.3). This is the main database server for hosting operational, production, and analytical data. The primary goal here is to autonomously transfer data from Survey Solutions®ś PostgreSQL database to Microsoft's SQL server database. This relieves users from the burden of having to manually export, download, process, and

load the data. Automation operations are seamless and reduce the enormity of labor-intensive tasks.

Two phased automation processes are proposed: 1) Automate data extraction and export from the Survey Solutions® server (PostgreSQL database) into the data repository using R-API and 2) extraction of data from the data repository into operational and analytical databases (MS SQL server) using PDI. These two stages are further discussed in Subsections 4.6.2 and 4.6.3.

4.6.2 The Automation of R Script

R is a programming language that allows the automation of processes and the implementation of statistical techniques (R Development Core Team, 2005). It offers a wide range of services such as data manipulation, reporting, and visualisation, automation, statistical analysis, and supports some machine learning libraries. This study uses the R-API to accelerate the processes of exporting and downloading data from the PostgreSQL database (Survey Solutions® – based database). The data are exported in tab-delimited file format and packed into zipped folders. Data export is facilitated by the R-API developed for Survey Solutions®Ṙ-API packages are freely available for use[3]. The R functions in Figure 4.4 make use of the API included in the Susoapi library to export, download, and load data into the central repository.

Exporting and loading data using the R functions involves three steps:
1) Use a password and username to log in and connect to the server using the specified IP address. These parameters are passed to a function set_credentails and permanently stored for future reuse. A successful connection makes possible the start of data export, and that is facilitated by the function called start_export.
2) The start_export function takes four parameters, which are questionnaire Id, export type, status, and metadata. Because there may be multiple questionnaires containing data on the server, it is necessary to specify a questionnaire from which the data is to be exported. The parameter export type specifies the data format to export. The supported export file formats are SPSS, Tabular (tab-delimited data), and Stata. The interview_status parameter assigns the status of an interview. The status of the interviews may be interview-assigned, completed, supervisor-assigned, supervisor-approved, supervisor-rejected, headquarters-approved, and headquarters-rejected. The data export may be performed with any of the listed statuses. The last parameter, include_meta, is set to true to allow the inclusion of the metadata when the data is exported. Metadata describes the exported data and provides useful information for better interpretation of

[3]https://github.com/arthur-shaw/susoapi

```
1
2  library(susoapi)
3
4  #STEP 1: CONNECT TO THE SERVER
5
6  set_credentials(
7    server = "http://172.6.12.1:8080",
8    user = "api_user",
9    password = "api_user@2022#@#"
10 )
11
12 #STEP 2: START DATA EXPORT PROCESSES
13
14 start_export(
15   qnr_id = "9898d5f60b3b4d1fbd50fa0a96dbeaed$3",
16   export_type = "Tabular",
17   interview_status = "ApprovedByHeadquarters",
18   include_meta = TRUE
19 ) -> started_job_Id1
20
21
22 # STEP 3: DOWNLOAD THE EXPORT DATA TO DATA REPOSITORY, ONCE THE JOB IS COMPLETE
23
24 get_export_job_details(job_id = started_job_Id1)
25
26 get_export_file(
27   job_id = started_job_Id1,
28   path = "F:/Data/Data Repository"
29 )
```

FIGURE 4.4: R functions definition for export data from PostgreSQL database

the data.

In the last step, 3) the get_export_file function allows the downloading of data into the data repository. These three steps accomplish the goal of exporting data from the Survey Solutions® database and loading them into the data repository. By default, these steps are destined to be run manually. However, the automation layer was incorporated to allow seamless export and loading of data into the data repository.

To automate data export and downloading, a batch file was developed using Bash programming language. The Bash programming can be done using any lightweight development environment such as Notepad++, Notepad, Visual Studio code, or any supported integrated development environment. Batch files have .bat extensions and are vital for running programs in a Windows environment. Figure 4.5 shows the Bash programming code that executes R functions to start the automation of data export and download. The batch file launches the command prompt (cmd) in minimised mode. The program first locates the R executable file on the C drive and changes the directory to point to the R script named AutomatedR_SS_Export_All.R. This is the actual script containing the R functions (shown in Figure 4.4) to export and download the data. The last line of the Bash code instructs the cmd to terminate, subsequent to the data export and download. The batch file is mainly created to run the R script and is scheduled to be executed by the Windows task scheduler.

The Windows task scheduler is a program in the Windows environment that is mainly

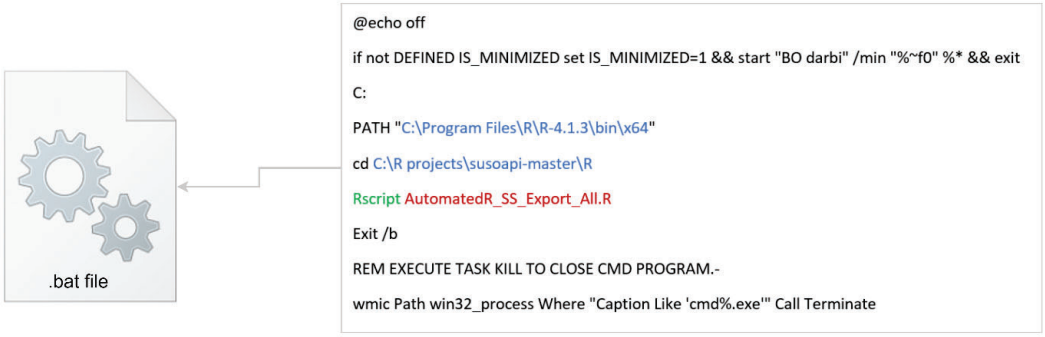


FIGURE 4.5: Bash programming code for launching R Script in Windows environment's task scheduler

used to schedule tasks or programs (details are provided in Subsection 4.6.4). The task scheduler can be configured to run the batch file at specific times. The scheduling enables automated export and download of data into a specific data repository at regular intervals. The time interval determines the frequency of data export, download, and loading into the data repository. PDI reads the data from the data repository and loads it into the database from which the dashboard or reporting system reads the data. PDI automation processes are detailed in the next section.

4.6.3 PDI Job Automation and Data Integration

PDI is a piece of software that provides ETL (Extract, Load, and Transform) capabilities to manipulate, clean, and move data through a consistent and uniform format that is relevant and accessible to end users and Internet of Things (IoT) technologies [4]. PDI presents a workflow as a building block for data transformation. The workflow consists of steps that are linked together using hops. Steps and hops are used to build transformations and jobs. A transformation "is a network of logical tasks called steps" (see example in Figure 4.6). The transformation extracts data from the databases, joins the tables together, and loads selected data values into the analytical database. The transformation creates data streams to move data from the source to the destination. In situations where multiple transformations exist, a PDI job may be used to batch and run them as a single unit of work. A job is used to coordinate the execution, resources, and dependencies of ETL activities. It allows the batch execution of transformations and other tasks. A job can perform tasks such as emailing the error log files in case of execution failure, executing transformation to retrieve FTP or web server files, checking the existence of target database tables, executing ETL transformations to update the analytical database, executing ordering control, etc.

[4] https://help.hitachivantara.com/Documentation/Pentaho/9.0/Products/Pentaho$_{D}ata_{I}ntegration$

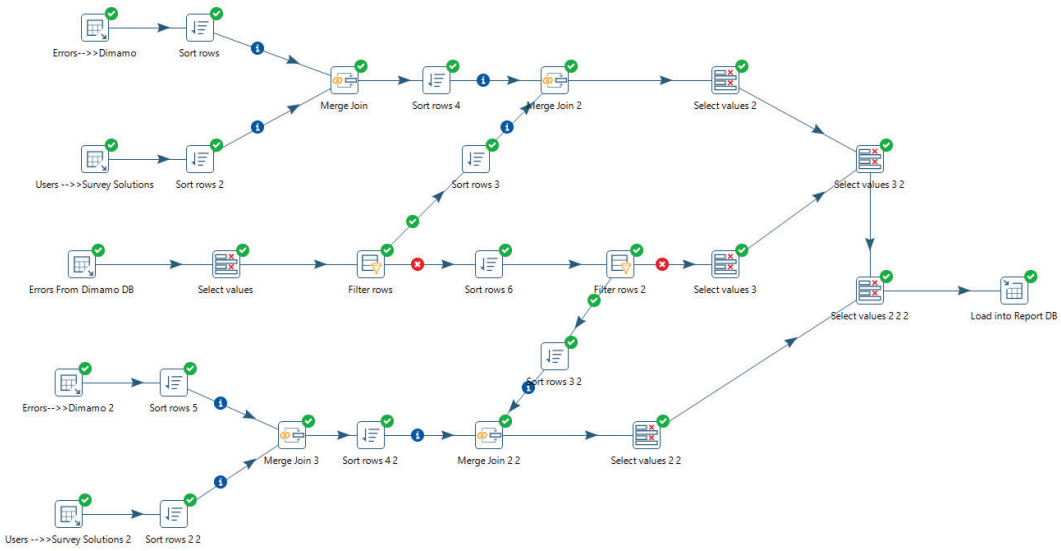


FIGURE 4.6: Transformation moving data from two databases into an analytical database

The job can be configured to execute these tasks automatically. A job allows batching of tasks and serves as a unit of automation. For example, Figure 4.7 shows a job that batched together 11 transformations to extract data from different data sources into the analytical database. The job first executes an SQL query to check if tables exist in the database. It also performs a simple evaluation and aborts the job if the execution of one transformation fails. The transformation steps prefixed with DTi for $i = 0, 1, \ldots, 4$ fetch data from multiple tab-delimited files and other databases, transform, and load it into different operational database tables from which an analytical database is built.

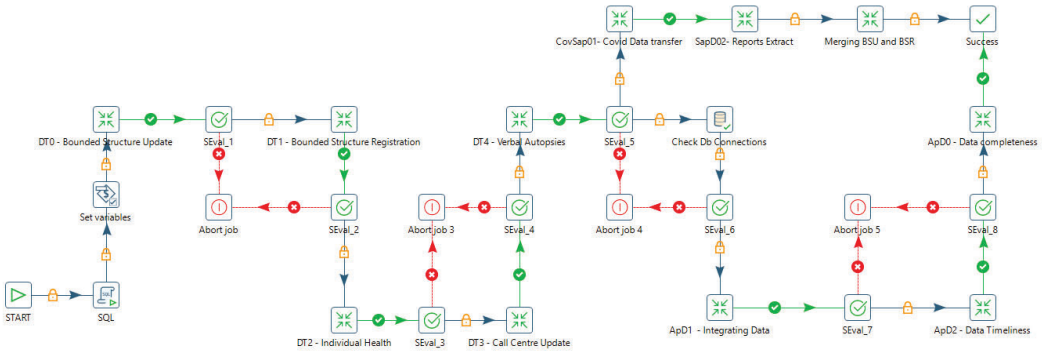


FIGURE 4.7: A unit of automation - integrating data from multiple data systems

Other steps with names prefixed with ApDi for $i = 0, 1, \ldots, 2$, reference transformations that aggregate and load data into the analytical database. The process allows analysis and reporting of data quality. To ensure near real-time analysis and reporting, the job is scheduled to automatically execute transformations 2 to 3 times a day. The automation process is initiated by executing a batch file that contains a path to the PDI job (see Figure 4.8). The batch file (.bat) starts cmd in a minimised mode and

launches kitchen.bat, which runs the PDI job in a silent mode. Kitchen.bat launches the ApD0001-Master.kjb job, which, in turn, runs the multiple transformations to start data transfers. The .bat file is scheduled in the Windows-based task scheduler to automate data extraction, transfer, transformation, integration, and loading.

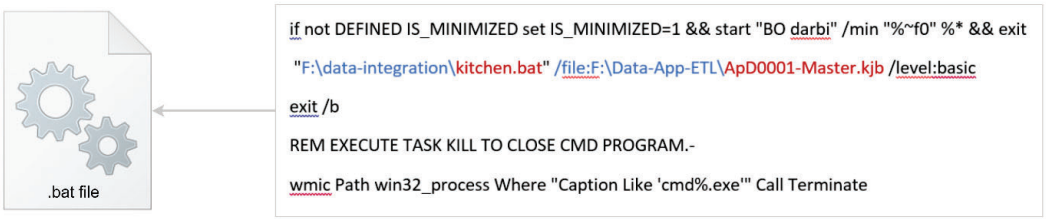


FIGURE 4.8: Bash programming code for launching PDI in Windows environment's task scheduler

4.6.4 Configuring Batch Files to Run PDI and R Script from the Windows Task Scheduler

The Windows operating system has a built-in scheduler (see Figure 4.9) to facilitate task scheduling. The scheduler can be used by the system and other apps to automate and maintain tasks such as updates, app start-up, disk clean-up, and fragmentations[5]. The task scheduler was set up to automate the PDI job and the R script. When scheduling both the PDI job and R script, separate batch files were created. The task scheduler was used to configure and time-trigger the execution of the batch files. The task scheduler initiates batch files to execute the PDI job and the R script to transform data, export, download, transfer, and load data. The launching is triggered by the frequency (hourly, daily, weekly, monthly, etc.) at which the schedule is set to run. For example, the system can be set to execute batch files every day at 12:00 a.m. or any time of the day. The system can be executed as frequently as necessary to ensure that the database has the most recent data. However, the frequency at which the system is run may have a detrimental impact on the overall system or server performance. For example, in this study, automation is configured to run on the server where other tasks and services are executed. If the frequency is exceedingly high, such as every 30 minutes or hourly, a single job may monopolise the system and prevent other processes from running efficiently. Hence, the execution frequency must be decided with diligence.

The task scheduler was set on Windows Server 2016 to enable high availability and reliable updating and transfer of data from the source to the destination database. The server runs 24 hours a day, seven days a week, making it perfect for setting up a schedule to execute bash-programmed code that automatically executes both PDI

[5]https://www.windowscentral.com/how-create-automated-task-using-task-scheduler-windows-10

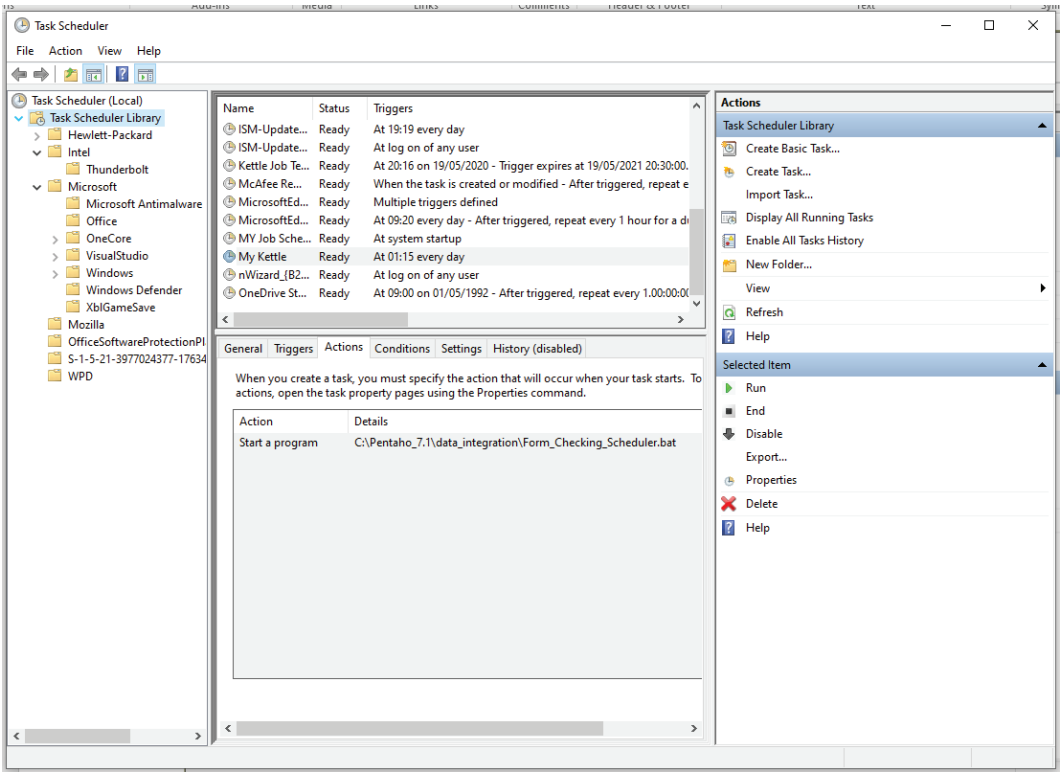


FIGURE 4.9: Microsoft Windows task scheduler on database server

jobs and R scripts. The automation procedure is particularly essential since it relieves users of the burden of manually operating the system. The selection of R and Pentaho automation was influenced by the accessibility of Survey Solutions®ś R-API packages, the simplicity of automation and data manipulation capabilities, the author's experience with the tools, and the open-source nature of both platforms.

4.7 Data Quality Assurance and Procedures

Data quality assurance is the process of ensuring that collected data meet high-quality standards. These processes start at the initial point of data entry in the field when primary data is collected. Most data quality issues arise at this stage and, if not properly managed, may crawl into the database. It is less expensive to quality control data at the early point of data entry compared to data already saved in the database (Ross, 2017). The total data quality management rule (1-10-100) states that, for every dollar used for preventing data quality issues, organisations can save 10 dollars to correct such data errors and prevent 100 dollars in loss of revenue due to organisational collapse. In light of this, our research proposed a data cleaning engine or framework called the 3TTDQM (3-Tier Total Data Quality Management) framework that transforms bad quality data into good and usable data. 3TTDQM is the proposed framework that applies preventive

and corrective measures to incoming data to enhance and report issues related to data quality. These measures are further discussed in Subsections 4.7.1 and 4.7.2.

4.7.1 Data Cleaning Engine for the 3TTDQM Framework

Data cleaning is one of the essential steps in data management processes and involves the identification and correction of data errors. It helps improve data quality and provides more reliable, consistent, and accurate data for better decision-making within an organisation. Uncleaned data can lead to defective decisions, operational problems, and misguided strategies which may ultimately increase costs while revenue and profits fall. When done properly, data cleaning has the potential to improve operational performance, improve decision-making, increase data usage for research or other purposes, and reduce cost due to fewer data management activities needed in the long run. Having a good data cleaning system is one giant leap toward ensuring the success and sustainability of any data-driven organisation. Figure 4.10 presents a data-cleaning engine that forms an integral part of the 3TTDQM framework. Consider input dataset x consisting of dirty data elements$(d_1, d_2, d_3 \ldots . d_n)$. Dirty data is data comprising of missing values, typing errors, invalid data entries, inconsistent values, incomplete datasets, and temporal or transitional violations.

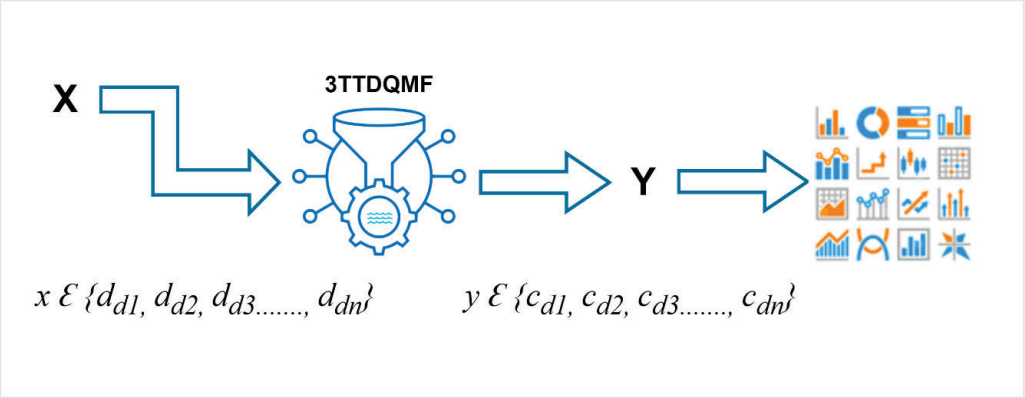


FIGURE 4.10: Data cleaning engine for 3-tier total data quality management framework

The 3TTDQM takes dirty data, applies preventative and corrective measures, and produces clean data elements $(c_1, c_2, c_3 \ldots . c_n)$. The framework incorporates three levels of data quality assurance (refer to Chapter 3 for activities performed at each level). These levels collectively are synonymous with a data-cleaning engine. This engine receives dirty data, performs automated data quality checks, applies corrective measures, and produces a better version of the data received (good quality data).

4.7.2 Validation and Skipping Logic Algorithms

The primary distinction between a paper-based data collection system (PDC) and an electronic data collection (EDC) application is the ability to incorporate data quality assurance methods. In PDC, the data cannot be validated at the time of capture, which introduces a multitude of problems. For example, invalid data entry may not be detected until very late in the analysis. Therefore, PDC is susceptible to erroneous data entries (King et al., 2013) and requires extensive data quality management activities. It has longer data turnaround times (Okwaraji et al., 2012) due to intermediate processing of data and transcriptions. The inability to implement data quality assurance algorithms in PDC makes it a less efficient method of data collection. On the other hand, EDC offers better mechanisms for data quality control. The importance of enforcing data quality during primary data collection cannot be underestimated. This is the first line of defense against data anomalies. In EDC, data values such as dates, geocodes, numeric, and text can be checked for data quality violations. In case of any violation, a prompting message can be displayed requiring correction. The check for data quality issues ensures consistency, completeness, and overall accuracy of the collected data. Figures 4.11, 4.12, 4.13, and 4.14 present algorithms implemented on EDC for quality assurance.

```
$validate_day

self.Substring(0, 2) == "17" ?
self.Substring(8, 2) == "01" :
(self.Substring(5, 1).ToUpper().ConsistsOf("SQ") ?
((self.Substring(8, 1).ConsistsOf("0") &&
self.Substring(9, 1).ConsistsOf("123456789")) ||
(self.Substring(8, 1).ConsistsOf("12") &&
self.Substring(9, 1).ConsistsOf("1234567890")) ||
(self.Substring(8, 1).ConsistsOf("3") &&
self.Substring(9, 1).ConsistsOf("10")) ||
(self.Substring(8, 2).ToUpper() == "UU") ||
(self.Substring(8, 1).ToUpper().ConsistsOf("F") &&
self.Substring(9, 1).ConsistsOf("12")) ||
(self.Substring(8, 1).ToUpper().ConsistsOf("W") &&
self.Substring(9, 1).ConsistsOf("1234"))) :
((self.Substring(8, 1).ToUpper().ConsistsOf("0U") &&
self.Substring(9, 1).ToUpper().ConsistsOf("123456789U")) ||
(self.Substring(8, 1).ToUpper().ConsistsOf("12U") &&
self.Substring(9, 1).ToUpper().ConsistsOf("1234567890U")) ||
(self.Substring(8, 1).ToUpper().ConsistsOf("3U") &&
self.Substring(9, 1).ToUpper().ConsistsOf("10U")) ||
(self.Substring(8, 1).ToUpper().ConsistsOf("F") &&
self.Substring(9, 1).ConsistsOf("12")) ||
(self.Substring(8, 1).ToUpper().ConsistsOf("W") &&
self.Substring(9, 1).ConsistsOf("1234"))))
```

FIGURE 4.11: *C#*-based macros for validating a day of the week on the electronic data collection system

Figure 4.11 shows an algorithm that validates the day in the captured date. This is used mainly for estimated dates. Here, the rule is that, if a day on a date is unknown during an interview, then "UU" must be captured instead. For example, if an interviewee only

remembers the year and month, then the date must be captured as "2021-10-UU'. In the case where the month and day are unknown, a date can be captured as "2021-UU-UU' (The algorithm for validating the month is not included here). For an estimated date, the estimated values must be captured as "UU". If not, an algorithm throws an error with an appropriate error message. The algorithm guards against capturing an estimated value that is not "UU". This allows a universal way of capturing estimated values. Imputations are performed during data saving to replace "UU" with the random day or month value and flagged appropriately as the estimated value. Unlike the algorithm presented in Figure 4.11, the algorithm in Figure 4.12 validates all date values. The algorithm ensures that the correct length is captured. In the case of an unknown date, the default date must be set to "9999-99-99". For the date captured as "2021-10" or "202-01-01", an algorithm throws an error prompting the entry of a valid date. This is particularly true because a valid date must be of length 10.

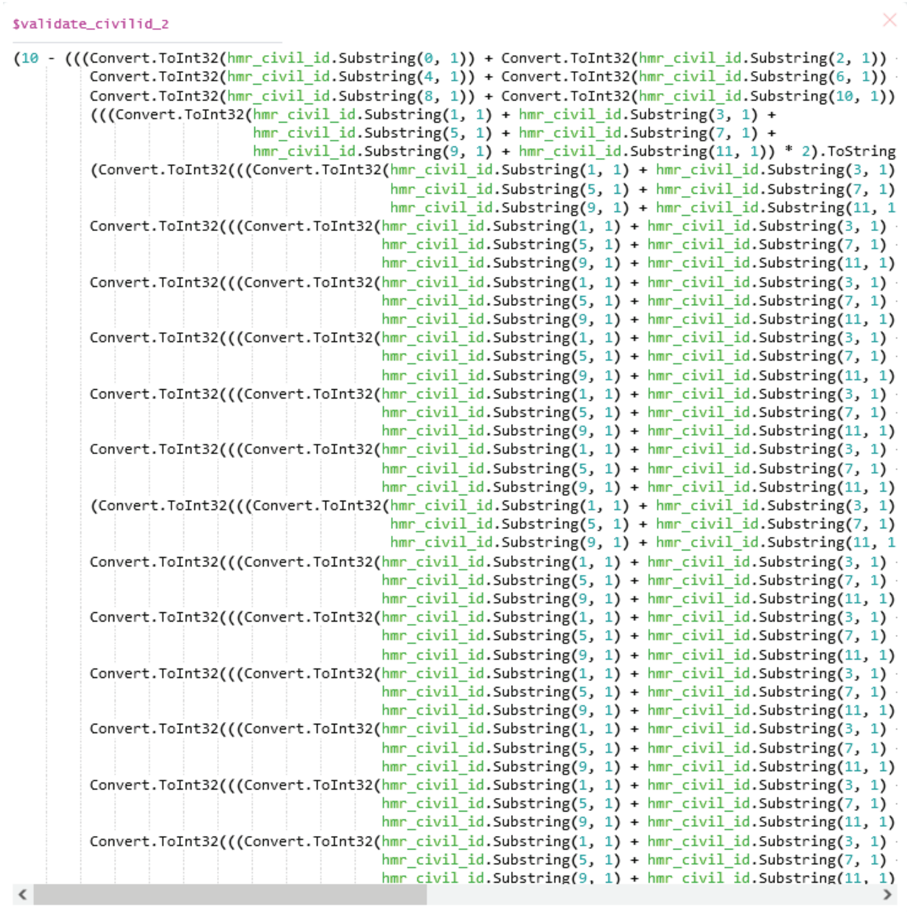
```
$validate_date

self.Length != 10 ? false : |
(self.Substring(0, 4).IsNumber() &&
self.Substring(5, 2).IsNumber() &&
self.Substring(8, 2).IsNumber() ?
(self == "9999-99-99" ? true : IsDate(Convert.ToInt32(self.Substring(0, 4)),
Convert.ToInt32(self.Substring(5, 2)),
Convert.ToInt32(self.Substring(8, 2)))) :
true)
```

FIGURE 4.12: An algorithm for validating any date captured in the system

The algorithms or macros are designed and referenced by the name. For example, in Figure 4.12, "$validate_date" is an algorithm name used to reference all date fields. The algorithm name is attached to all date-specific fields to validate the date entries. Referencing by name allows the algorithms to be called and executed multiple times for different questions. An obvious reason behind the approach is maintainability. It is easier to maintain one algorithm that is referenced by multiple fields than having different algorithms for each field. In the latter case, should there be a bug in the algorithms, all the algorithms for each field must be manually updated to fix the bug. For example, if there are 20 fields each with embedded algorithms, to fix a bug, one will have to go through each algorithm, locate an error, and fix it. So, maintaining similar algorithms for each field is cumbersome and time-consuming. But in the case of one global algorithm, a bug in the algorithm may be easily fixed in one place. This speeds up the system development and maintenance processes.

The nature of South African identity number makes it prone to mistakes. This is especially due to a long sequence of numbers. An algorithm in Figure 4.13 validates the accuracy of the captured ID number. The length must be 13 with the first 6 digits representing the year, month, and day. These digits must correspond to the date of birth. If not, then an algorithm throws an error, prompting the correction since there



```
$validate_civilid_2

(10 - (((Convert.ToInt32(hmr_civil_id.Substring(0, 1)) + Convert.ToInt32(hmr_civil_id.Substring(2, 1))
        Convert.ToInt32(hmr_civil_id.Substring(4, 1)) + Convert.ToInt32(hmr_civil_id.Substring(6, 1))
        Convert.ToInt32(hmr_civil_id.Substring(8, 1)) + Convert.ToInt32(hmr_civil_id.Substring(10, 1))
        (((Convert.ToInt32(hmr_civil_id.Substring(1, 1) + hmr_civil_id.Substring(3, 1) +
                          hmr_civil_id.Substring(5, 1) + hmr_civil_id.Substring(7, 1) +
                          hmr_civil_id.Substring(9, 1) + hmr_civil_id.Substring(11, 1)) * 2).ToString
        (Convert.ToInt32(((Convert.ToInt32(hmr_civil_id.Substring(1, 1) + hmr_civil_id.Substring(3, 1)
                                       hmr_civil_id.Substring(5, 1) + hmr_civil_id.Substring(7, 1)
                                       hmr_civil_id.Substring(9, 1) + hmr_civil_id.Substring(11, 1
        Convert.ToInt32(((Convert.ToInt32(hmr_civil_id.Substring(1, 1) + hmr_civil_id.Substring(3, 1)
                                      hmr_civil_id.Substring(5, 1) + hmr_civil_id.Substring(7, 1)
                                      hmr_civil_id.Substring(9, 1) + hmr_civil_id.Substring(11, 1)
        Convert.ToInt32(((Convert.ToInt32(hmr_civil_id.Substring(1, 1) + hmr_civil_id.Substring(3, 1)
                                      hmr_civil_id.Substring(5, 1) + hmr_civil_id.Substring(7, 1)
                                      hmr_civil_id.Substring(9, 1) + hmr_civil_id.Substring(11, 1)
        Convert.ToInt32(((Convert.ToInt32(hmr_civil_id.Substring(1, 1) + hmr_civil_id.Substring(3, 1)
                                      hmr_civil_id.Substring(5, 1) + hmr_civil_id.Substring(7, 1)
                                      hmr_civil_id.Substring(9, 1) + hmr_civil_id.Substring(11, 1)
        Convert.ToInt32(((Convert.ToInt32(hmr_civil_id.Substring(1, 1) + hmr_civil_id.Substring(3, 1)
                                      hmr_civil_id.Substring(5, 1) + hmr_civil_id.Substring(7, 1)
                                      hmr_civil_id.Substring(9, 1) + hmr_civil_id.Substring(11, 1)
        Convert.ToInt32(((Convert.ToInt32(hmr_civil_id.Substring(1, 1) + hmr_civil_id.Substring(3, 1)
                                      hmr_civil_id.Substring(5, 1) + hmr_civil_id.Substring(7, 1)
                                      hmr_civil_id.Substring(9, 1) + hmr_civil_id.Substring(11, 1)
        (Convert.ToInt32(((Convert.ToInt32(hmr_civil_id.Substring(1, 1) + hmr_civil_id.Substring(3, 1)
                                       hmr_civil_id.Substring(5, 1) + hmr_civil_id.Substring(7, 1)
                                       hmr_civil_id.Substring(9, 1) + hmr_civil_id.Substring(11, 1
        Convert.ToInt32(((Convert.ToInt32(hmr_civil_id.Substring(1, 1) + hmr_civil_id.Substring(3, 1)
                                      hmr_civil_id.Substring(5, 1) + hmr_civil_id.Substring(7, 1)
                                      hmr_civil_id.Substring(9, 1) + hmr_civil_id.Substring(11, 1)
        Convert.ToInt32(((Convert.ToInt32(hmr_civil_id.Substring(1, 1) + hmr_civil_id.Substring(3, 1)
                                      hmr_civil_id.Substring(5, 1) + hmr_civil_id.Substring(7, 1)
                                      hmr_civil_id.Substring(9, 1) + hmr_civil_id.Substring(11, 1)
        Convert.ToInt32(((Convert.ToInt32(hmr_civil_id.Substring(1, 1) + hmr_civil_id.Substring(3, 1)
                                      hmr_civil_id.Substring(5, 1) + hmr_civil_id.Substring(7, 1)
                                      hmr_civil_id.Substring(9, 1) + hmr_civil_id.Substring(11, 1)
        Convert.ToInt32(((Convert.ToInt32(hmr_civil_id.Substring(1, 1) + hmr_civil_id.Substring(3, 1)
                                      hmr_civil_id.Substring(5, 1) + hmr_civil_id.Substring(7, 1)
                                      hmr_civil_id.Substring(9, 1) + hmr_civil_id.Substring(11, 1)
        Convert.ToInt32(((Convert.ToInt32(hmr_civil_id.Substring(1, 1) + hmr_civil_id.Substring(3, 1)
                                      hmr_civil_id.Substring(5, 1) + hmr_civil_id.Substring(7, 1)
                                      hmr_civil_id.Substring(9, 1) + hmr_civil_id.Substring(11, 1)
        Convert.ToInt32(((Convert.ToInt32(hmr_civil_id.Substring(1, 1) + hmr_civil_id.Substring(3, 1)
                                      hmr_civil_id.Substring(5, 1) + hmr_civil_id.Substring(7, 1)
                                      hmr_civil_id.Substring(9, 1) + hmr_civil_id.Substring(11, 1)
```

FIGURE 4.13: Validation algorithm for checking consistency and validity of civilian IDs

are inconsistencies between the date of birth and ID number. The next four digits after the first six digits define gender. The ranges 0000 – 4999 and 5000 – 9999 represent females and males, respectively. These 4 digits must be validated against the captured gender. If the captured gender is “Female” and the next 4 digits are “5885”, then clearly the digits represent a “Male” since it falls within 5000 – 9999. So the validation fails and an error message is thrown. The eleventh digit classifies citizenship. For example, 0 represents South African citizenship while 1 defines permanent residency status. This can be validated against any digit that is not 0 or 1. Alternatively, it can be validated against an answer to a question such as “Are you a South African citizen?” If the answer is yes, then the expected answer would be 0 else error if any other digit. The validation algorithm can validate the consistencies between gender, date of birth, and ID number. In a complex data collection environment, questionnaires are often lengthy and comprise multiple rosters. Some questions in such questionnaires may not be asked of some participants. For example, a male participant must not be asked pregnancy-related questions. This can be achieved by implementing skipping logic algorithms (Figure 4.14).

Skipping logic algorithms allow questions to be hidden if certain criteria are not met.

```
var_mem_age >= 12 && var_mem_age <= 49 && (hm_current_gender == 2 || hm_gender == "Female") &&
(hm_new_female_member == 1 || hm_has_preg_history == "N") && hm_now_member == 1 && hm_current_mem_resident == 2
```

FIGURE 4.14: Skipping logic algorithm

In the figure, the question is enabled if the participant is a female aged between 12 and 49. She must be a new member or have no pregnancy history. Finally, the question is only administered to participants who are currently resident members. The skipping logic algorithm controls how the questions should be administered. It enforces data quality by hiding irrelevant questions. In a typical environment where PDC is deployed, it is impossible to hide on question-based conditions. All questions are available, and given the type of participant, an interviewer has to decide which questions to skip. At times, the interviewer may mistakenly provide answers to irrelevant questions, and that may greatly affect the quality of data. For example, the question "is the participant pregnant" may have the answer "yes" even if the participant is a male. Since such questions are not necessary to ask male participants, it may be a good idea to have them hidden to avoid erroneous entries. The skipping logic algorithms preserve the structure of the questionnaire and save a lot of time by filtering out irrelevant questions during interviews.

4.8 Data Stores

The process of building the database goes through several stages of the database life cycle (DBLC) (Teorey et al., 2010). These stages according to (coronel et al., 2020), are database initial study, database design, implementation and loading, testing and evaluation, operation and maintenance: Database Initial Study - At this stage, the main goal is to define the problem, company situation, business rules and constraints, boundaries and scope, and the objectives of designing the database. These allow the designer to gather as much information as possible to better understand the company's operational environment and systems prior to the actual design of the database. Some of the activities performed at this level involve meeting with the relevant stakeholders and users of the existing systems, analysis of legacy systems, etc.

1. Database design - It is at this stage that the developers carry out activities to ensure that the design meets the system and user requirements. It is arguably the most crucial phase of DBLC. Much attention must be paid to the finer details of database model development and validation. At this point, the designer critically analyses two views of the system: the business and the designer's view. The two-mindedness approach grants the designer the opportunity to design the

database from different perspectives. From a business point of view, the designer sees data as the main source of information from which knowledge is derived. On the contrary, the designer view examines data access, the structure of data, and activities required to manipulate and convert data into meaningful information. The database design process is broadly linked to the design and analysis of the larger system. Hence, understanding how different components or pieces fit into the whole system is of utmost importance. The design of the database involves three stages: Conceptual, Logical, and Physical designs.

(a) Conceptual design - At this stage, an abstract database is created in a way to represents a real-world entity as realistically as possible. This representation must represent a pure insight into the business and its core activities. It is not necessary to identify software and hardware at this level of abstraction, as this may divert the designer's attention from the actual task of designing the database. Therefore, the design must be independent of software and hardware.

(b) Logical design – Intermediary stage that creates the mappings between conceptual model and logical model to deploy on the relational DBMS. Logical design validates and assigns integrity constraints, uses normalisation to validate logical models, and reviews the model with the user. The access rights to the database are defined at this stage. Unlike conceptual design which is software-independent, logical design depends on the underlying software to define appropriate domains and necessary access constraints. The stage also defines the physical requirements needed for the optimal functioning of the database system in its operational hardware environment.

(c) Physical design – deals with the data access characteristics and the selection of the data storage. It is at this stage that the logical models are translated into actual tables, the database usage and volume are analysed, data storage requirements are estimated, and database security is determined.

2. Implementation and loading – takes into account details or blueprint outline in the design phase. These are the sequence of guidelines describing the creation of the database, tables, security constraints, indexes, views, domains, attributes, etc. The choice of which DBMS to install must be made prior to creating the database. The DBMS may be installed on the physical or virtual server, depending on the design of the server environment.

3. Testing and Evaluation - The design must be tested and evaluated to validate the user requirements. Bugs and imperfections must be identified and corrected at this stage to meet specific requirements as outlined in the database initial study

phase. It is at this stage that the database is tested and fine-tuned for better performance, security constraints, consistency, integrity, and multiaccess. This may be done concurrently with application programming. Programmers may use an entity framework to map database tables to object-orientated classes on the application layer. An entity framework is a collection of ADO.net technologies that aid in the development of data-driven software applications (Castro et al., 2007). Programmers can use a conceptual schema to model and access their data, which is mapped to a database table via a versatile mapping. Data manipulation can be accomplished using object-relational mappers, iterator APIs, or SQL language. The data-driven applications depend on the underlying database. Therefore, a poorly designed database has an adverse effect on the application itself and reporting. The interoperability and interdependency between the application and database systems must be thoroughly tested and evaluated.

4. Operation – the database is said to be operational once it has passed the evaluation stage. Full-scale operation is performed on the system by connecting applications programs and users to perform operational activities. That may put a burden on the system and cause it to ultimately fail. Because the system evolves over a period of time, the designer must always monitor its use, identify bottlenecks and causes of failure, and apply corrective measures.

5. Maintenance – some of the problems identified during operation may need to be resolved at this stage. Routine maintenance activities must be performed to optimise database performance and keep the system current. Other activities that may be performed are corrective maintenance, preventative maintenance, assignments and maintenance of access rights for old and new users, use system generated statistics to perform security audits, system monitoring, and performance tuning. The change in modern systems (applications and databases) is inevitable due to the high demand for advanced data and reporting features. Adding more data may require a change in database structure, and that may ultimately cascade down to the application program. Therefore, at some point, system maintenance will be necessary.

A good database design eliminates data redundancy and data inconsistency and improves operational speed (Suraya and Sholeh, 2022). Properly designed databases go through the normalisation processes (Lee, 2008). Normalisation is the process of breaking down the relationship of the database into simpler and more manageable tables. The process ensures a flawless design with minimal data redundancy and improved data integrity. The normalisation of the database tables may be performed up to the third normal form (3NF) or even the Fourth Normal Form (4NF). We choose to discuss only

3NF, and the reader is referred to (Wu, 1992; Date and Date, 2019) for details on 4NF. 3NF enforces the application of normalisation principles and dictates that the relational database schema be designed in a way to eliminate data redundancies, guarantee referential integrity, and circumvent data anomalies, thus simplifying data management (Demba, 2013). The main objective of normalisation is to remove transitive and partial dependencies. To define transitive dependency, let X, Y and Z be variables in relation R such that the two dependencies $X \rightarrow Y$ and $X \rightarrow Z$ hold in R. Then Z is said to be transitively dependent on X. For partial dependency, the dependency $X \rightarrow Y$ is partially dependent if and only if for some variables $Z \in X$ removed from X, the dependency still remains (Thomas and Carolyn, 2005).

Our study considered relational databases for data storage. Relational databases use software called a relational data management system (RDBM) to manage data access (roles and access rights) and data retrievals. With the aid of RDBMS, relational databases organise data into tables and create relationships between them (Sumathi and Esakkirajan, 2007). The relationships are established by creating a primary key in the parent table and a foreign key in the child table. The relationships or linking of the tables make possible the reconstruction of the data and extraction of the useful data. Data extraction from the database may be performed in various ways. For example, through diverse ETL tools or by means of a structured query language (SQL). SQL is the most powerful and yet simple programming language for manipulating data from the database (Jamison, 2003). SQL gained popularity over the years in the operationalisation of data definition, manipulation, and transaction control languages in relational databases. SQL works well in a transactional environment and is not suitable for most modern databases such as MongoDB, Cassandra, Hbase, Neo4J, etc. These databases use noSQL to define and manipulate data. NoSQL simply means that there is no SQL required to manipulate data. These databases are used in big data environments due to better performance, scalability, availability, and ability to handle structured, semi-structured, and unstructured data (Acharya et al., 2019). However, NoSQL databases are not suitable for CRUD (create, read, update, delete) operations or transactional data processing (González-Aparicio et al., 2017). The nature of our study dictates the use of a relational database to perform the CRUD operation. In a quest to achieve the study objectives, we consider the following databases to investigate data quality issues: PostgreSQL (Operational Database) and MS SQL server (Production database), which are discussed in the next Subsections.

4.8.1 Operational Database

Survey Solutions® hosts the PostgreSQL database that is used to store the collected data. The data in this database is a time-variant due to fast-changing statuses, hence called operational database. Survey Solutions® is linked to the EDC application through a synchronisation point to load data into the PostgreSQL database. The data collected using the EDC application is synced on a daily basis to update the operational database. PostgreSQL was selected as the operational database due to its compatibility with the Survey Solutions® platform. By default, the fields or variables in the EDC application are mapped to database tables in the PostgreSQL database. This enables mapping between application variables and the database fields or attributes. When the data is synced from the application, the mappings ensure the correct loading of data in the relevant fields in the operational database. The operational database is designed to store fast-changing data updates. Consequently, the state of the data changes rapidly as data collection and processing occur. For example, recently collected data will have a completed status by the fieldworker; after being reviewed by the field supervisor, it will have a supervisor-approved or supervisor-rejected status, depending on whether the interview data was rejected or accepted. The automated data quality system can further approve or reject interview data and attach headquarters-approved or headquarters-rejected statuses. All these changes must be recorded in the database and are crucial for tracking the data collection progress. As data collection and updates take place, para-data is generated. Para-data are the data records that provide additional information about the conducted interviews such as interview start-time, end-time, duration, statuses, etc. It gives the details of how long and when each interview was conducted. It also links all user activities to interviews. For example, for all completed, approved, or rejected interview data, it provides details of the users involved. The details are mainly user-Id, status, date and time, and also the reason for rejection if rejected. Para-data provides rich analytical data and is very useful for general survey analysis. Data in the operational database must have a status headquarters approved for permanent storage in the production database. Headquarters is the highest stage of quality control, and data is presumed to be of high quality once it has passed this stage. At this point, data can be transferred and permanently saved to the production database.

The major difference between the operational and production databases is that the operational database has fast-changing data statuses and keeps track of current data collection processes. The 3TTDQM framework is implemented on this database to validate the quality of the data. On the other hand, the production database contains the finalised data with high quality and its state does not change frequently. That is, it contains data that has passed through all levels of quality control and is presumed to be of high quality. The last level of quality control is implemented in the production database

to ensure the highest possible data quality. The main objective is to thoroughly clean the data whilst in the operational database and move it into the production database when in a state of good quality. Once stored in the production database, no further changes are necessary.

4.8.2 Production Database

The production database stores both historical and current data. The data must have gone through all stages of quality control before being archived in the database. The database has rules configured to further validate the quality of the data. These rules guard against inconsistencies and data violations. Violations include, among other things, missing values, duplicate events, and invalid data entries. Database rules are precautionary measures to improve data quality. The data entry not meeting quality standards is rejected back to the operational database for fixing. The detailed error message is attached and sent to the relevant user for correction. Example of an error message from the production database: "*Unable to save Call Centre Document Id 345331 Interview key 36-29-05-91. An error occurred while updating the entries. See the inner exception for details. ERROR RI320: Individual PHTP-H is too old to be the mother of RXMD-B because she was born on 06 Oct 1957, over 55 years before RXMD-B was born on 01 Nov 2016! The transaction ended in the trigger. The batch has been aborted*". Error messages are thrown to point out possible errors in the initial point of the data entry point. From the error message, it may be true that PHTP-H is the mother of RXMD-B but the mother's date of birth is incorrectly captured, making her too old to conceive. Or RXMD-B is not a child of PHTP-H pointing to a possible mistake in the selection of the mother. Note that PHTP-H and RXMD-B are individual identifiers that uniquely identify individuals in the database. Given these individual identifiers, we are able to obtain details of specific individuals. Some error messages are constructed by comparing the incoming data with the existing data in the database. While others are thrown on the basis that the incoming data has some violations. The combination of these cases ensures the elimination of inconsistencies and invalid data entries in the production database. Both operational and production databases make provision for data quality assurance but do not present a straightforward analytical state of data quality. Hence, it is necessary to have an analytical database to report on the progress and the state of data quality as more issues are resolved. The analytical database was modelled and discussed in Chapter 3.

4.8.3 Business Analytics Approaches

Business analytics is a subset of a business intelligent system and a data management solution that involves data transformation and analysis, identifying trends, and application of complex machine learning algorithms to gain insight into organisational operations using data. Business analytics provides valuable insights that support organisational performance decisions. Business analytics employs methods from the field of information systems, machine learning, operational research, and data science (Mortenson et al., 2015). Akerkar (2013) and Krumeich et al. (2016) classified business analytics into three main categories ordered by their level of complexity, intelligence, and value: descriptive, predictive, and prescriptive analytics. These categories raise crucial questions in finding root causes and solutions to organisational problems. For example, descriptive analytics seeks to find answers to questions such as: "What happened?", "Why did it happen?", "What is happening currently?". These are investigative questions to understand the current state of things and identification of issues through trends analysis. To understand what happened, data is analysed quantitatively to understand the trends in the data. Trends will suggest possible data quality issues or problems in data collection processes that must be scrutinised to understand why they occurred. Understanding why it occurred takes into account the relationships linking various kinds of data (Soltanpoor and Sellis, 2016). Once the sources of data quality issues are ascertained, issues can be dealt with and prevented in the future. The current state of data quality may be good, but it is necessary to look further into the future trends for possible issues. Doing so requires predictive analytics; answering questions such as "What will happen in the near future?", "When will it happen?" and "Why is it happening?" Understanding what will happen in the future is crucial in mitigating the effects of events. Descriptive and predictive analytics identify trends in the data, however, do not provide actionable interventions or suggestions to resolve the problems. Prescriptive analytics suggests how an actor must be involved in resolving data quality issues identified in descriptive and predictive analytics phases. It has been regarded as the next step in improving business performance by "increasing data analytics maturity and leading to optimised decision making ahead of time" (den Hertog and Postek, 2016). Crucial questions asked in the prescriptive analytics phase: "What must I do to resolve data quality issues?", "Why should I resolve these issues", and "What are the implications of not resolving data quality issues?" These questions are fundamental to achieving our study objectives.

Figure 4.15 depicts the business value of descriptive, predictive, and prescriptive analytics. The goal of descriptive analytics is to establish what is happening now by gathering and analysing factors that contribute to the root causes of the event that needs to be

corrected or adequately addressed. Descriptive analytics can search for patterns that pose a significant challenge or a potential organisational prospect.

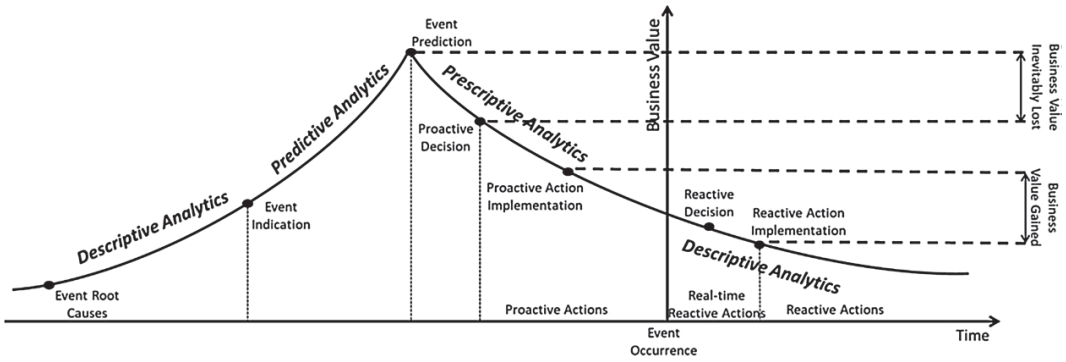


FIGURE 4.15: The business value of analytics with respect to time (Lepenioti et al., 2020)

Given the output of descriptive analytics, events can be predictively investigated to contribute to business value. Contribution to business value depends on the action and decision implemented. In making decisions that drive profitability, human expertise and knowledge play a crucial role. To achieve optimal decision-making, prescriptive analytics may be implemented after predictive analytics. Unfortunately, there is a time interval between the prediction and proactive decision, which results in inevitable business value. Greatly reducing the time interval will result in a minimised loss in business value, leading to organisational success. Process automation has the potential to minimise the time interval and bring the system to near-real-time. The outcomes of the event predictions are fed into prescriptive analysis to produce preemptive decisions. Prescriptive analytics closely monitors the system and checks if an event has occurred. Insights into what has happened and why it has occurred are derived whenever events take place. These factors that influence business value have the potential for organisational growth through well-informed reactive and proactive decision-making.

4.9 Chapter Summary

The chapter explained the methodological approaches used to investigate the phenomenon of interest. Emphasis was placed on paradigms and the philosopher's position as a building block to understanding the nature of truth and its derivation. The paradigms and philosophies made possible the selection of suitable methods to investigate data quality issues. Equally important, the study considered quantitative methods to investigate the relationship between dependent and independent variables. We seek to investigate whether there is a positive relationship between EDC systems, data management systems, and data quality. To achieve that, the researcher thoroughly discussed the

processes of building an EDC system and data management system. Additionally, the study incorporated integration and automation to enhance the functionalities of EDC and data management systems. The R automated API facilitated the export, download, and loading of data into the central data repository. The next step of automation involved PDI in the integration of data from the central data repository and the production database into the operational database. All these steps played a crucial role in data quality assurance. For example, EDC systems included data quality assurance mechanisms such as validation algorithms, skipping patterns, etc. while data management systems factored in data quality metrics, for instance, completeness, accuracy, validity, timeliness, etc. The methods selected for this study distinguishably focused on three parts; EDC system, data management system, and data analytics to holistically assure high data quality through error identification and reporting.

Chapter 5

Analysis and Interpretation of Results

5.1 Introduction

Previous chapters have provided the groundwork by describing the study objectives, system models, methodology, and a detailed evaluation of the pertinent literature. This chapter focuses on the results and analysis derived from the experimental setup in Chapter 4, offering a comprehensive discussion of the hypotheses findings. The primary objective of this study was to build a data quality framework to effectively manage the quality of data in HDSS. To achieve this, open-source technologies were considered in the development of electronic data collection (EDC) and automated data quality control systems. The use of the system generated an enormous amount of data, which was rigorously analysed and interpreted to answer research questions and contribute to existing knowledge on the subject.

The findings of this study emanated from the implemented framework that was aimed at monitoring, managing, and improving the quality of the data. The framework included algorithms embedded in the EDC system to improve data quality at the data entry point. Part of the analysis seeks to evaluate the effectiveness of algorithms in validating and improving data quality. Furthermore, we evaluated the efficiency with which the automated system improved data quality and productivity. Moreover, the proposed framework incorporated the Pareto principle and Process control techniques to identify, monitor, and manage the processes that are involved in curation and data assemblage. In addition to improving the quality of the data, metrics were used to evaluate the quality of the data over one year. These metrics were built into the dashboard, which aims

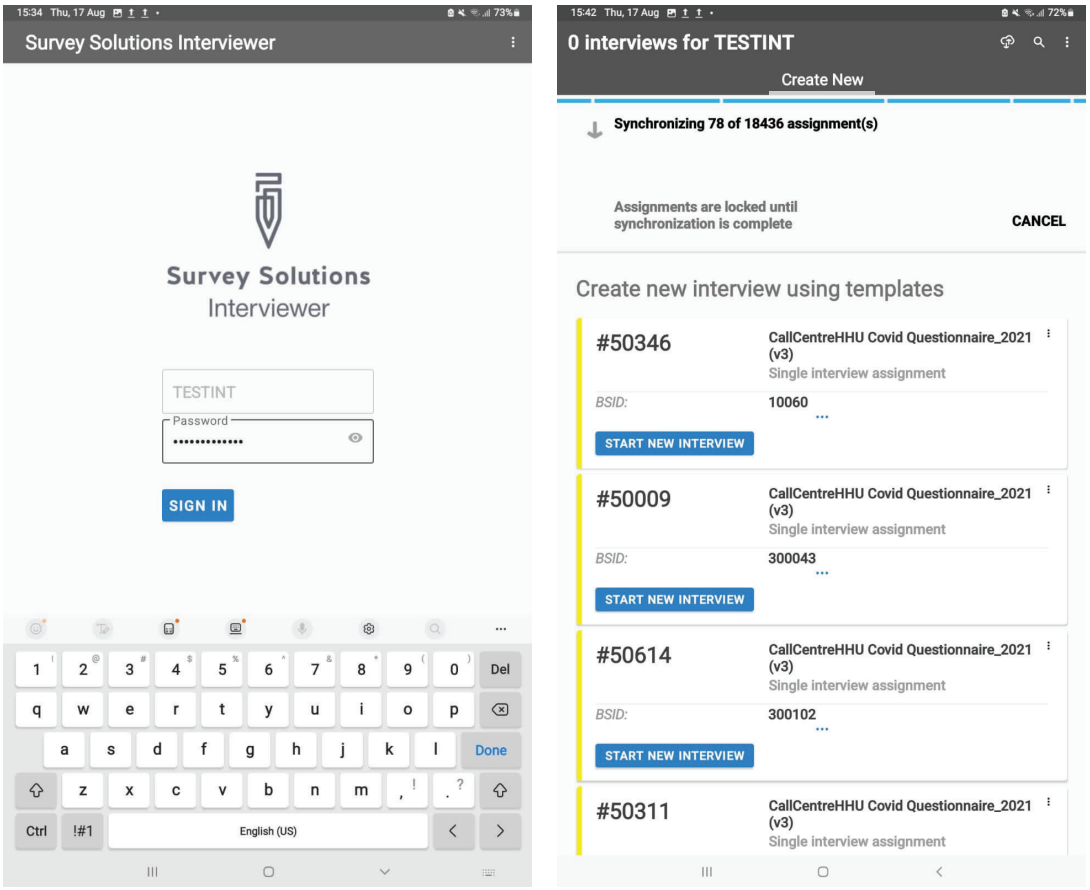
to monitor and improve the quality of data. The statistics on data quality issues were gathered from the database server, aggregated, and analysed using Python programming language. The graphing library in Python Dash called plotly.graph_objects was used to visualize data graphically on the web-based dashboard. The library was also used to plot widgets to present the levels of data quality in the database. The dashboard was built using Python programming language and deployed to Heroku® for ease of access for users.

The remainder of the chapter is structured as follows. Section 5.2 examines data quality assurance at the application or EDC system level, whereas Section 5.3 discusses the output of system automation. In Section 5.4, the Pareto analysis was used to identify and analyse the significant contributors to data quality problems. The section further presents the results of how the Process control mechanism monitored and maintained the error rate within acceptable boundaries. Section 5.5 compares the data quality levels before and after the deployment of the proposed data quality framework. User acceptance test results are presented in section 5.6. Section 5.7 discusses the hypotheses and section 5.8 concludes the chapter.

5.2 Data Quality Assurance at the Application Level

Survey Solutions® is an EDC platform used in this study to host and manage data. The platform incorporated an integrated environment to build an EDC system aimed at improving the quality of data. This study takes advantage of this aspect and uses the platform's capabilities to include C#-based algorithms to validate the quality of data. The approach made it easier to implement data quality improvement strategies such as skipping logic, macros, and validation algorithms. Data quality assurance forms an integral part of this work and was enforced at an application level to prevent data quality issues. In protecting the integrity and confidentiality of data, necessary security measures must be employed as shown in Figure 5.1 (a). Successful login allows the synchronisation of data between the tablets and the Survey Solutions® server (refer to Figure 5.1 (b)). This is the actual data collected from the field and transferred to the PostgreSQL database.

EDC system interfaces in Figure 5.1 allow fieldworkers to login and access questionnaires for data collection. The interfaces were designed and built for ease of use and featured algorithms to guarantee good-quality data. The algorithms validate the data when captured and alert the fieldworkers of any data quality violation by throwing error messages. Various approaches to data quality validations are presented in Figure 5.2.



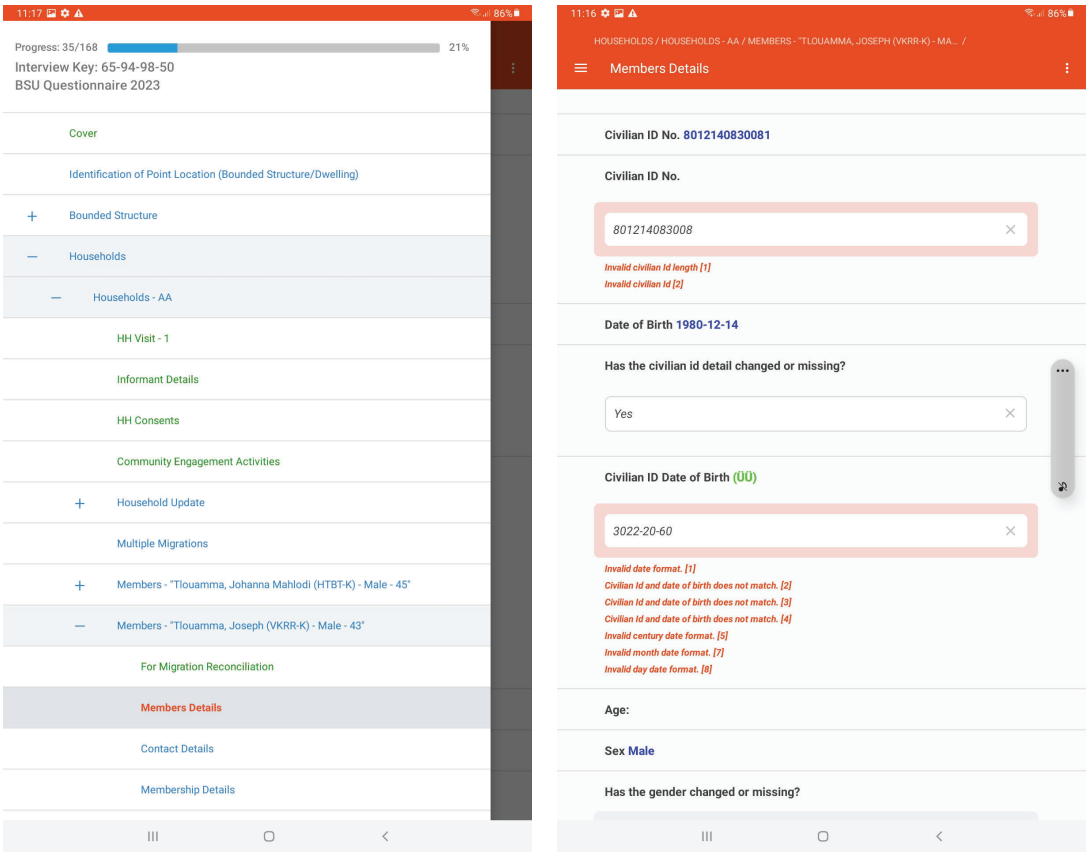
(a) Electronic data collection system login screen

(b) Synchronisation of data between Survey Solutions® Server and the tablets

FIGURE 5.1: Electronic data collection system user interfaces

The system is organised into sections encapsulating interview questions. The navigation panel in Figure 5.2 (a) allows fieldworkers to navigate through various sections with ease to access questions to be asked during data collection. The colour coding in the sections serves as an important measure of data quality. The colour provides a crucial hint to the fieldworker by suggesting that some questions in a particular section are incomplete or have errors.

Sections with blue-colour coding, for instance, indicate the presence of one or more unanswered questions. This is a crucial attribute to satisfy the data completeness criterion. Thus, the data is complete if a value is provided for every required attribute or field. Incomplete data values significantly degrade data quality, necessitating remedial actions to maintain quality. On the other hand, the green colour scheme shows that all required data elements are provided, which guarantees completeness. However, this does not guarantee the validity, consistency, or accuracy of the data. Diverse quality measures were implemented at multiple levels to identify and eliminate all potential problems that degrade data quality. Figure 5.2 (b) depicts an application interface that includes

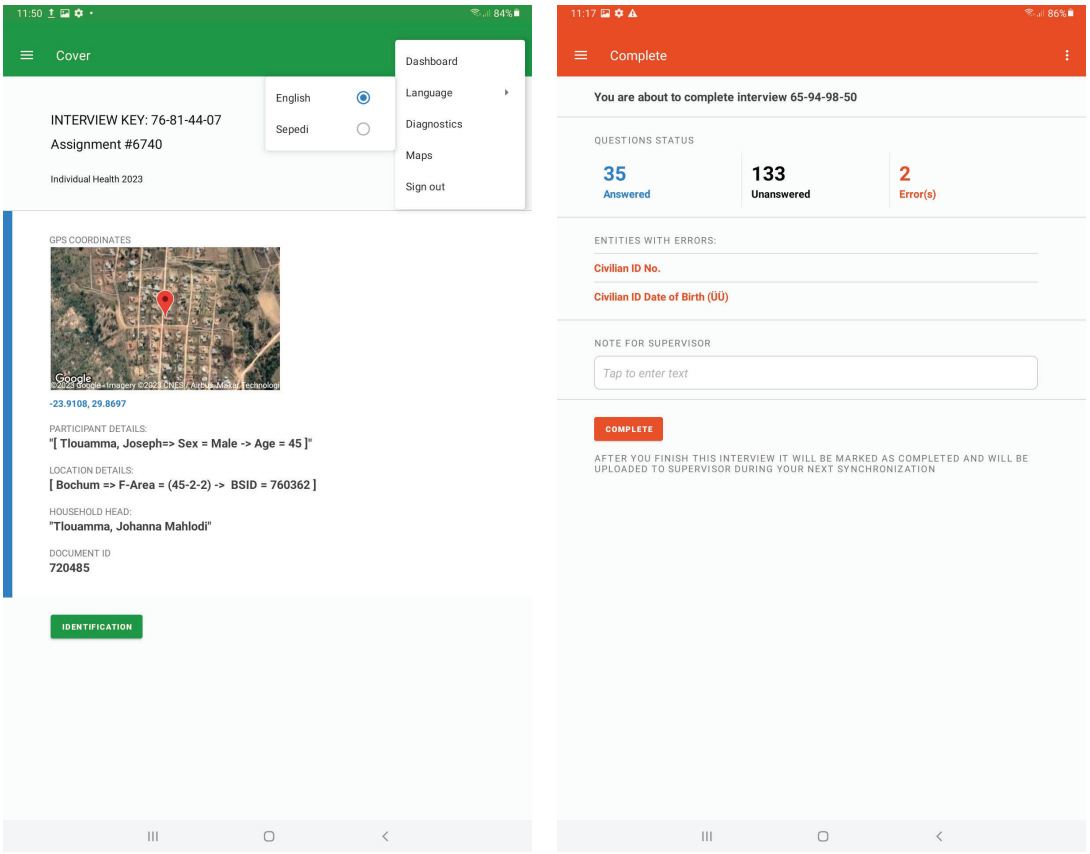


(a) Electronic data collection system organised in sections with different colour coding schemes

(b) Electronic data collection system with validation error message

FIGURE 5.2: Data quality control measures deployed in the electronic data collection system

a module to guarantee validity, consistency, and accuracy. The module includes the skip patterns, validation rules, and macros necessary to improve the quality of the data. The red-coloured section or text in the figure reports the existence of error(s). Error messages are thrown when erroneous data entries are performed. When an inaccurate, inconsistent, or invalid data entry is made, a red-coloured text message is displayed to alert the user of any potential problems. An error message describes the nature of the violation and, in some instances, suggests a possible solution to fix the problem. In Figure 5.2 (b), for instance, the ID number contains twelve digits rather than thirteen (13), which triggered the error messages "Invalid civilian ID length" and "Invalid civilian ID". Validation algorithms were designed to guard against inconsistencies in capturing South African identification numbers and dates of birth. The date of birth must match the first six digits of the South African identification number. The date of birth with an invalid year, month, or day is the first element to be pointed out. Errors associated with each scenario were thrown, indicating the root cause of the problem. Additionally, the system examines the date of birth and the first six South African identification numbers for uniformity. The error message "Civilian Id and Date of Birth do not match"



(a) Data preloading for verification and identification (b) Verification of data completeness and accuracy

FIGURE 5.3: Electronic data collection system user interfaces

was generated due to discrepancies. Validation algorithms are vital for ensuring the consistency, precision, and completeness of these variables. While the use of validation algorithms and macros is an essential component of data quality assurance, they can introduce problems if they are not properly designed and implemented. This study also considered skipping patterns or algorithms as a measure to improve data quality at the point of data entry. This is crucial for irrelevant questions that may be mistakenly asked. For example, the questions that are specific to females. Opening such questions to all interviewees irrespective of gender can potentially introduce problems. The macros and validation algorithms enforced the quality of the data by providing hints to fieldworkers and reporting the nature of the violation. This measure improved accuracy and completeness. Skipping patterns and data preloading effectively enforced the validity and consistency of the data. The data consistency was enforced through data pre-loading.

Figure 5.3 (a) shows historically collected data loaded onto the EDC system before data collection. This allows validation algorithms to check if the current data is consistent with previously collected data. The prepopulated data served two purposes; validating currently collected data against previously collected information. And checking the

consistency of the currently collected data with the historical data. The previously collected data served as a baseline for ensuring the validity and consistency of data. In addition, Figure 5.3 (a) further shows another important feature called language translation. This feature allows fieldworkers to use either Sepedi or English languages to administer questionnaires. Administering questionnaires in the local language enhances understanding and effective communication during the data collection, which may yield better-quality data. As shown in Figure 5.3 (b), the proposed EDC system provides some statistics on the number of questions answered, unanswered, and with errors. This feature is an essential measure of data quality since it allows fieldworkers to review what was done. Since all questions are mandatory, the fieldworkers have to revisit unanswered questions and resolve all the errors. This approach ensured completeness, accuracy, and consistency of data.

5.3 Automation of Manual Processes

On a server operating 24 hours a day, seven days a week, the automation of processes was implemented to facilitate the export, download, transfer, and loading of data into the production database from the PostgreSQL database. Automation was programmed to run daily at midnight to initiate the export of data from the Survey Solutions® server. The R-script was launched through the Windows task scheduler to initiate the data export and transfer processes. Figure 5.4 depicts the data export process in action. The process involves the sequential export of three datasets. The first set of data was exported successfully and ready for download, while the second set was still in the running state. The third dataset was queued, waiting for the completion of the second. Once the data export is completed, the next step is to download the data to the desired or target folder. It should be noted that the data export must be performed prior to the download, transfer, and loading of data. This process was timed to ensure the complete exportation of all datasets before downloading and transferring data to the Microsoft SQL server database.

The automation procedure allowed the transfer of data from the Survey Solutions® based database (PostgreSQL) to the server hosting production database (MS SQL). The automated system was proposed to replace manual operations previously used to export, transform, download, and transfer data between servers. The automation eliminated human errors inherent in manual processing, eliminated extraneous activities leading to shorter turnaround times, optimised system performance, and increased productivity. Users were able to focus on other more important tasks than exporting, downloading, and transferring data. Since the implementation of the automated system, there have

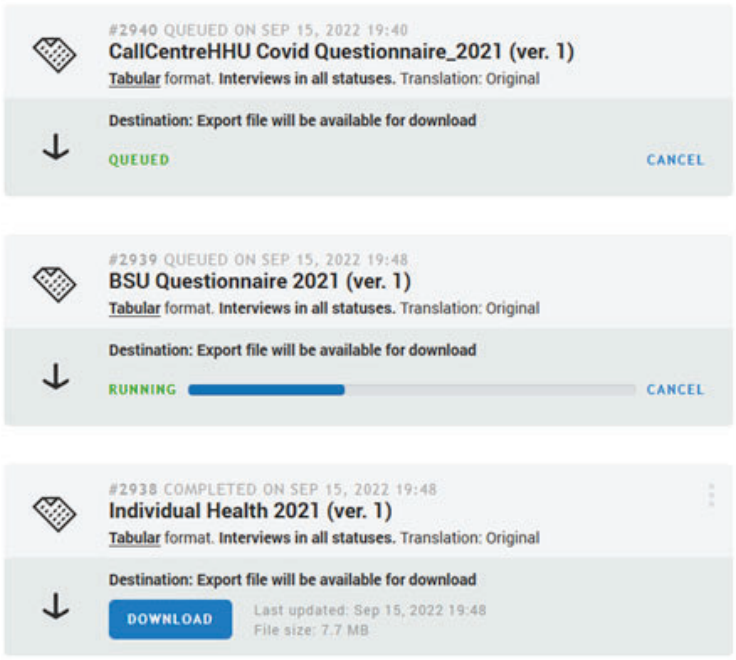


FIGURE 5.4: Automated data export, download, and transfer from the Survey Solutions®based PostgreSQL database

been no reports on the types of errors that inevitably occurred when users processed the data. The most frequent mistakes that occurred when manually processing data involved exporting the incorrect data, passing wrong parameters (questionnaire ID, interview status, data export type, etc.), transferring data to a wrong folder, and passing wrong arguments to the data processing PDI Job. Such mistakes would result in an error in running the PDI job or incorrectly generated reports. Consequently, users would have to sequentially re-perform all time-consuming activities. The implementation of an automated system completely eliminated the inherent mistakes in the manual processing of data and execution of data systems. Proper configuration of automation parameters such as automation time, server IP address, authentication details, questionnaire Id, status, workspace, path, etc., allowed a series of routine actions to be carried out repeatedly and reliably. The automation effectively eliminated error-prone tasks and waiting times. The term "waiting time" is used to describe the amount of time a user must wait for an action to complete, such as the completion of data export, download, transfer, and processing operations using PDI Job. The waiting time or delay can take anywhere from a few minutes to an hour, depending on how much data is being processed. For example, manually exporting and downloading all datasets shown in Figure 5.4 would take approximately 40 to 60 minutes. This is the amount of time users had to wait before the data could be processed. However, with the automation of these procedures (export, transform, download, and transfer of data), waiting times were eliminated since activities were performed autonomously at midnight. Running processes at night has

a number of benefits, including improved server performance due to less traffic to and from the server, the availability of up-to-date data the following day, and a reduction in peak-period processing burden.
In addition to process automation, this study considered the automation of data quality control procedures. An automated data quality control system is the core of the proposed framework and has demonstrated its effectiveness in improving the quality of data and eliminating human-based quality control. This automated system improved the quality of the data by approximately 18% and provided 0% human involvement. Other results that are directly related to the deployment of the automated system are presented in the following sections.

5.4 Application of Pareto Principle and Process Control Mechanism in Data Quality Management

The study applied Pareto analysis and the Process control technique in evaluating the efficiency of the proposed framework and enhancing the quality of data. According to Pareto, 20 percent of the causes contribute to 80 percent of the problems. Adopting this principle to data quality, we analysed how data quality was affected by 20 percent of the causes and why was it necessary to pay closer attention to this vital few in resolving 80 percent of the issues. When utilised properly, the Pareto principle may aid in prioritising tasks, optimising resources, and increasing overall productivity. It offers a beneficial foundation for understanding complicated systems and finding major development opportunities (Halog and Manik, 2011). Section 5.4.1 provides analyses of how the Pareto principle was applied to improve data quality. Although the Pareto principle was effective in identifying the main contributors to the data quality issue, it did not account for special and common cause variations. In the process of monitoring, controlling, and improving quality, it may be necessary to account for deviations of processes or errors from the mean. Taking that into account, the process control technique was applied to analyse daily average errors and this is discussed in Section 5.4.2.

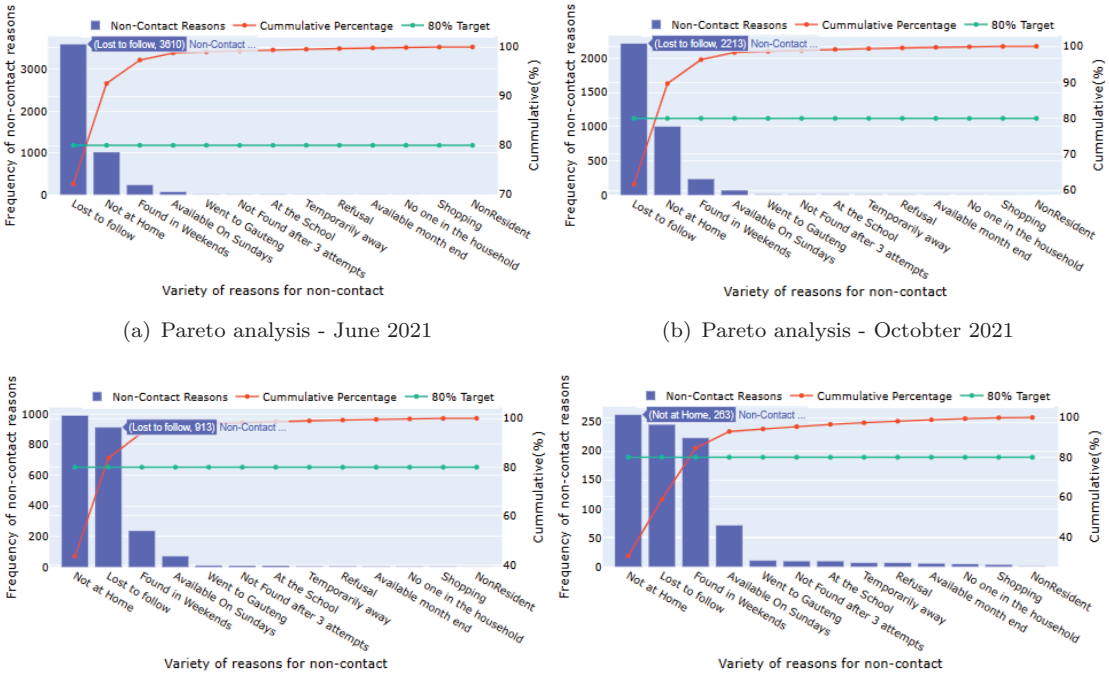
5.4.1 Data Quality Improvement using Pareto analysis

Wilfred Pareto observed that “20 percent of people own 80 percent of the world’s wealth or resources”. The observation was later applied to total quality management and was called the Pareto principle. The Pareto principle states that 20% of causes result in 80% of issues. The 20 percent of causes is also referred to as “vital few”. That means that few things result in most of the problems. This research adopted and applied the Pareto

principle to HDSS data quality management due to its efficacy in other settings (Serdar and Syuhaida, 2012; Mineva, 2020; Kadiri et al., 2020; Sunadi et al., 2021). The Pareto principle played a key role in identifying a few non-contact reasons that contributed to most households not being located during data collection (refer to Figure 5.5). The principle was also used to identify a few fieldworkers who produced most of the errors (see Figure 5.6). We seek to identify 20 percent of fieldworkers or non-contact reasons contributing to 80 percent of data quality issues.

In Figure 5.5, the Pareto charts were used to categorise the reasons that led to households not being found during data collection. We generally call these non-contact reasons. Non-contact of the household has a bearing on the data quality, specifically data completeness, and timelines. Idealistically, the HDSS would want to visit and collect data from all households' under-surveillance, however, this may be difficult due to various reasons. For example, participants may not always be home when data is collected or GPS coordinates may be inaccurate, making it difficult to determine the actual location of the household. Completeness and timeliness would be 100 percent if all households could be located and data collected. It should be highlighted that completeness in this context does not refer to the completeness of data elements, but rather to the successful collection of data from all households (complete coverage). On the other hand, timeliness refers to the "freshness of data." The data in the database must reflect the current state of entities in the real world at a specific point in time. To achieve data quality criteria for both completeness and timeliness dimensions, Pareto charts were used to determine the most prevalent causes of data quality violations. From Figure 5.5 (a), "Lost to follow" contributed the largest proportion of data quality issues. 'Lost to follow' is defined as the reason for not finding the household due to its uncertain location. Inaccurate geo-codes or coordinates are the most significant contributors to uncertain or unknown locations. It should be mentioned that each location from which a household resides is geo-coded in order to facilitate navigation by GPS device or map. So, if coordinates are erroneous or inexact, it becomes harder to locate households, hence owing to the largest number of "Lost to follow".

As may be observed from Figure 5.5 (a), "Lost to follow" accounts for over 90% of attempts of not finding the households. Indeed, one non-contact reason accounts for a bigger chunk of data quality violations. The Pareto charts are attractive in that they identify the main cause(s) of the problem and suggest the starting point for resolving such issues. Putting measures in place to resolve the underlying causes of "Lost to follow" may result in resolving approximately 90% of the problems, thus improving the quality of the data. As shown in Figure 5.5 (b), three months after identifying the major non-contact problem, interventions were taken to increase coordinate precision, resulting in a 39% decrease in "Lost to follow". A pattern change may be observed for



(a) Pareto analysis - June 2021

(b) Pareto analysis - Octobter 2021

(c) Pareto analysis - February 2022

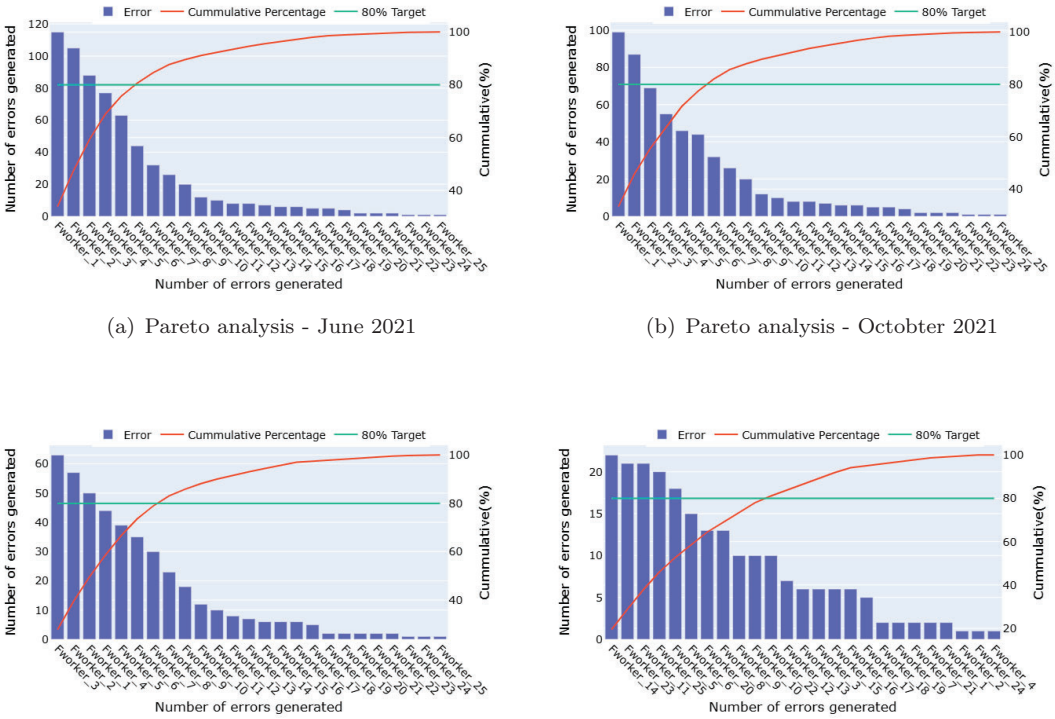
(d) Pareto analysis - June 2022

FIGURE 5.5: Application of Pareto principle in identifying major reasons for non-contact

February 2022, as more than 70% of "Lost to follow" issues were fixed between June 2021 and February 2022. The frequency of "Lost to follow" was lower compared to "Not at home". This occurred as a result of actions taken to address "Lost to follow". The actions involved isolating all locations with the tag "Lost to follow" and correcting GPS coordinates.

Figure 5.5 (c) shows two noncontact reasons that were put in the spotlight; i.e., "Not at home" and "Lost to Follow". The Pareto principle suggests that much attention must be put on resolving these two reasons, as they contribute to 80% of the issues. When dealing with "Not at home", special task teams were employed to visit affected households in the evening to track those who are not available during the day due to work, school, or other reasons. This exercise contributed to improving completeness and timeliness. A year after the implementation of the proposed system, as shown in Figure 5.5 (d), even though there were still noncontact reasons, there has been a substantial reduction in the number of non-contact reasons compared to June 2021. The results show that approximately 96% of the households were correctly located, with data successfully collected. This has, without reasonable doubt, improved the timeliness and completeness of data. The continued use of the Pareto principle may lead to improved data quality.

Figure 5.6 shows a second set of Pareto charts designed to identify the fieldworkers who are mostly error-prone. The initial stage of data capture generates the majority of errors,



(a) Pareto analysis - June 2021

(b) Pareto analysis - Octobter 2021

(c) Pareto analysis - February 2022

(d) Pareto analysis - June 2022

FIGURE 5.6: Applying Process Control Charts to monitor and control processes that lead generation of errors

which violates data integrity requirements or rules. The application and database-level rules protect against inconsistent data input, incorrect data items, or values. In the event of a rule violation, an error message is returned detailing the specifics of the violation. Errors occur for several reasons, including careless typing, inconsistent data input, inexperience with the data collection system, inadequate training, etc. Using the Pareto principle permits the identification of fieldworkers responsible for most errors. The intersection of the cumulative percentage and the 80% target line graphs in Figure 5.6 (a) identifies the vital few that contributed to most of the errors. The vital few are found from the intersection of two graphs to the left, which provides 20% of the fieldworkers that generated 80% of the errors. Focussing only on a vital few may have a tremendous impact on data quality improvements. To guarantee the effectiveness of research on quality improvement, planning, or control, it is essential to identify the vital few (Juran, 1954). The relevance of the vital few lies in the fact that nothing significant can occur unless the vital few are identified and resolved. This is especially true when a small number of factors account for most of the problems.

Understanding the cause of errors is essential to improve the quality of the data. By isolating 20% of fieldworkers (vital few) and analysing their particular mistakes, it may

be possible to decrease future error occurrences. For example, the vital few (Fworkers 1 to 5) from Figure 5.6 (a) were isolated and their errors were critically analysed. The critical analysis of errors led to the discovery of a variety of factors that led to errors. With necessary interventions, the rate at which errors are generated was greatly reduced in 3 months, as shown in Figure 5.6 (b). Although the same fieldworkers remain under scrutiny, there was a discernible decline in errors. Further interventions reduced errors by more than 50% in February 2022. Figure 5.6 (d) presents an interesting trend showing improved data quality. Quality improvement is a result of the successful identification of problematic areas and the implementation of the action plan.
We expect that the continued use of the Pareto principle will result in improved data quality by lowering data concerns to negligible proportions. It is important to remember that the Pareto principle does not really fix data quality problems; however, pinpoints the components that are most likely to be the root cause of those problems. The HDSS data management team is still responsible for carrying out the necessary steps and executing plans to address the data concerns. By applying the Pareto principle, the HDSS data management team can quickly narrow down to the most critical issues, from which they can begin to implement solutions. Incorporating the Pareto principle into the proposed data quality framework served as an important measure to improve data completeness and timelines.

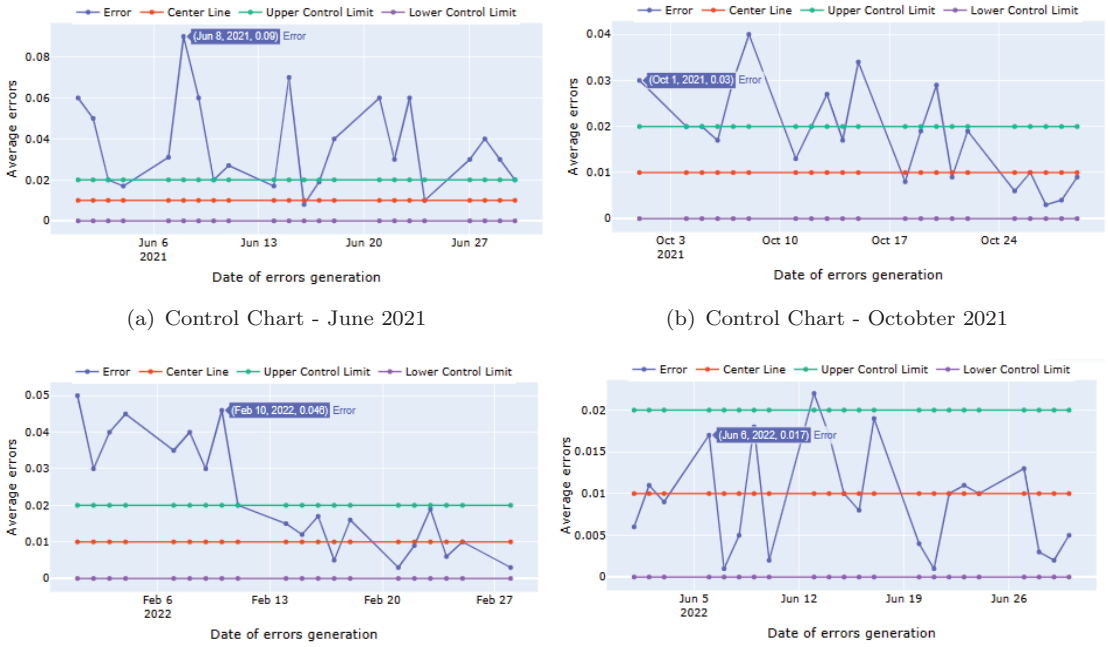
5.4.2 Application of Process Control Chart in Monitoring and Controlling Data Quality Processes

Statistical process control (SPC) is a technique that uses statistical analysis as the basis for monitoring, managing, and ultimately improving a process. The process control chart is an essential SPC tool that can be used, with statistical confidence to quickly identify changes in production processes (De Vries and Reneau, 2010). SPC has several advantages, including its simplicity, forecast of future process performance, investigation of the effect of process modifications, identification of areas for improvement, process improvement, etc. (Suman and Prajapati, 2018). This study applied a process control chart to monitor and manage processes that may potentially affect the quality of data. If these processes are effectively managed, data quality is likely to increase.

Presented in Figure 5.7 is a set of process control charts to monitor and manage error variations over a period of time. They highlight critical areas that require further scrutiny in so far as data quality is concerned. The graphs comprise three crucial horizontal lines that determine whether the process under investigation is in a stable state or not. These are center line, lower (LCL), and upper (UCL) control limits. The center

line indicates the statistical mean of the data. Data points falling within LCL and UCL are said to have a common cause of variation, showing stability and normality. The limits determine to what extent the process is out of control or unpredictable. The unpredictability of the process signals major data problems that require immediate attention. These problems may be inherent in data collection processes, resulting in the generation of errors. Monitoring and controlling processes that lead to the generation of errors is a critical step to ensure high data quality. We established our control limits (LCL and UCL) as being three standard deviations below and above the mean of the process or error. This is because control charts are based on a principle developed by Walter Shewhart (Shewhart, 1931), a modern quality movement pioneer, in the early twentieth century: if a single measurement is within three standard deviations of the mean of processes (center line), we may confidently call that an "anticipated" result. Observing only common-cause variation is what you would expect to see in a stable and well-controlled process. Variation due to common causes occurs naturally in any process or system and is expected from the process's design, routine operations, and other factors. When the data point deviates from the control bounds (LCL and UCL), an unexpected event would have occurred (see Figure 5.7(a), Process control chart: June 2021). Something extraordinary has led the process to spiral out of control. This is a specific cause variation example. It suggests that it is highly probable that the deviation is a result of data quality issues. We see that the standard deviation of the average error from the mean (center line) exceeds UCL. The trend indicates that the error rate for June 2021 was not under control.

Understanding the processes that led to the generation of errors could mitigate the effect of errors on data quality. Errors may be kept within a manageable range by using process control charts, which allow users to quickly and easily pinpoint and remedy any data problems. Preventive measures were applied to resolve issues identified in June 2021 and resulted in a minimised error rate as shown in Figure 5.7 (b). It remained difficult to maintain errors within the limits in the first 19 days of October 2021; however, these were reduced to acceptable limits for the remainder of the month. As may be observed in Figure 5.7 (c), the error rate skyrocketed uncontrollably in the first few days of February, despite the expectation that the trend of decreasing error rates from October 2021 would continue for the foreseeable future. The error averages beyond UCL were examined, and it was discovered that newly hired fieldworkers introduced greater errors due to unfamiliarity with the operational procedures and the systems. By analysing the nature of errors, the ideal strategy for enhancing data quality was identified and applied. Consequently, after the first ten days of January 2022, the average daily errors steadily decreased to acceptable levels. Furthermore, errors were significantly reduced one year after the implementation of the proposed framework, as



(a) Control Chart - June 2021

(b) Control Chart - Octobter 2021

(c) Control Chart - February 2022

(d) Control Chart - June 2022

FIGURE 5.7: Applying the Pareto principle in identifying fieldworkers who are generating more errors

shown in Figure 5.7 (d). The figure showed that errors were kept under control for most of the month, thus improving the quality of the data. However, at one point in mid-June 2022, errors exceeded the UCL, necessitating corrective efforts to bring errors back under control. It should be noted that even if all of the data points (average errors) fall within the LCL and UCL, potential problems may still exist. For example, according to (Suman and Prajapati, 2018), the existence of eight runs of consecutive data points on either side of the center line signifies a special cause of variation. This trend does not appear in any of the graphs in Figure 5-6, indicating that all processes within the UCL and LCL ranges have been effectively stabilised. Additionally, if the data points plot looks non-random with the points showing some kind of systematic pattern; that may also signal potential problems requiring attention. Taking a closer look at Figure 5.7 (d), we observe irregular data plot patterns, suggesting common-cause variation. The data points oscillation around the center line shows normal operations, which may include minimally acceptable errors. In such cases, there would be no need for intervention, as the rate at which errors are produced is low. Nonetheless, continuous monitoring of the errors is required since things may occasionally spiral out of control, resulting in a special cause of variation that may necessitate intervention. Usually, it is rather impossible to discern intuitively, without control charts, what sort of variation occurs and whether direct intervention would ultimately be beneficial or detrimental to process results (Benneyan, 1998). We identified several issues that led to special

causes of variation and applied corrective measures. For example, the 3TTDQMF was designed to account for consistent, accurate, and validated data entry through the use of validation and logic algorithms. The approach minimised the prevalence of data quality concerns, which ultimately allowed the process control mechanism to maintain errors under control.

5.5 Data Quality Measurements

This section analyses the results that were generated by quantifying the instances of data that were in violation of the agreed data quality standards. The MS SQL database server served as the primary data source from which the data was extracted and analysed. The data in the database were subjected to extensive data quality checks using the 3-Tier Total Data Quality Management Framework(3TTDQMF). The study analysed the quality of the data before and after the implementation of 3TTDQMF to assess its impact. The evaluations were performed every five months for a period of one year from June 2021 to June 2022. June 2021 served as the baseline and presented to results prior to the implementation of 3TTDQMF. The goal was to perform the analyses at different time points and assess the effectiveness of the proposed framework in improving data quality. To evaluate the impact of the proposed information system or framework on data quality, metrics such as completeness, timeliness, validity, accuracy, and consistency were considered. The metrics were modelled and programmed into the dashboard using Python programming language. The dashboard was deployed locally and also published to Heroku® cloud services for ease of access for users to continuously evaluate and monitor data quality. The dashboard is periodically re-updated to reflect the current state of data quality in the system. Figures 5.8 and 5.9 show dashboards illustrating four of the metrics assessed to report the level of data quality in the database. These metrics serve as the mirror of the database and report the state of data quality. Correct measurement and assessment of data quality can improve the value and usability of organisational data, which are important resources for making quality decisions that drive profitability. Figure 5.8 presents data completeness and timeliness. The data is complete if and only if all the data values from compulsory variables are present in the database. However, from the Figure, we observe that 78% of data elements were present in June 2021, meaning that 22% of the data values were missing. This assessment was made prior to the implementation of the proposed data quality framework that assesses the level of data quality in the database. This baseline analysis was necessary to evaluate the impact of the proposed system on data quality. The main objective of designing the framework was to monitor and improve the overall quality of the data by implementing the appropriate techniques to curb data anomalies.

FIGURE 5.8: Data visualisation system reporting data quality measurement

Three months after the deployment of the data quality framework, we observed a 3% improvement in data completeness, taking the overall percentage to 81. The improvement was directly linked to the interventions taken. Amongst other things, these interventions were the designing of data validation algorithms or macros, skipping patterns, deploying process control systems, and the Pareto analysis. The completeness of the data increased by 6% in February 2022 and by 5% in June 2022. The quality of data with respect to completeness has increased by 14% since the introduction of the data quality framework. The primary objective of this study was to maintain data quality within the range of 95 – 100%. Data quality levels falling within the range of 95 – 100% are considered acceptable. This is drawn from the argument made by (Schafer, 1999) that less than 5% of missing data is insignificant. Conversely, Bennett (2001) argued that if the number of missing data exceeds 10 percent, the statistical analysis would provide skewed findings. Therefore, we set a 5% threshold as an acceptable limit for data quality issues. However, there is no satisfactory percentage of missing values in the data to draw an accurate statistical conclusion (Dong et al., 2013; Kalkan et al., 2018). The goal was to bind data quality within acceptable limits, but from the figure, the level of completeness remained below 95% over a period of one year of the implementation of the proposed framework. The highest percentage of data completeness achieved was 92 and on the present trajectory, data quality seemed to be on the rise, which bodes well for its eventual ascent into a tolerable range.

Figure 5.8 further presents another attribute of data quality termed timeliness. In the context of this work, the data is timely if and only if its observation date is between n and n-1 where n is the current year. The evaluation of data timeliness was performed

over a period of a year from June 2021 to June 2022. As may be observed from Figure 5.8, data timeliness was measured to be 76.7% before deployment of the proposed framework. The percentage of data timeliness steadily increased over a period of a year to 95.8%. Having an accurate and up-to-date picture of the population under surveillance requires a good representation of the data in a timely manner. In addition to data timeliness and completeness, Figure 5.9 presents other key measurements of data quality. That is, the validity and accuracy. Validity in this instance measures the legitimacy of date variables. This is measured by looking at dates that must be bounded between start and observation dates. Any dates falling outside this range must be considered an outlier and invalid. As shown in Figure 5.9, the validity of the data in June 2021 was found to be 85.1%. This validity measure was based on historical data collected prior to the implementation of the proposed data quality framework. It was discovered that 14.9% of dates were invalid or incorrectly captured. The proposed framework successfully identified and resolved 12% of such issues, consequently improving the validity of dates by June 2022 to 96.7%.

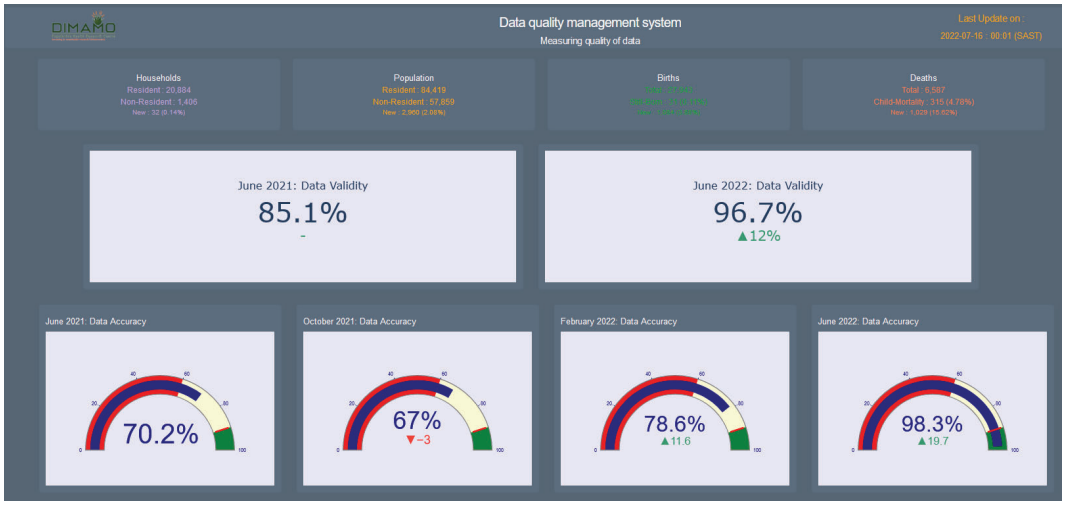


FIGURE 5.9: Dashboard measuring validity and accuracy of data

Figure 5.9 further presents data accuracy, which is the measure of how well the data represents real-world entities. The stored data had 70.2% accuracy level, indicating that 29.8% data elements are inaccurate. This was directly related to how the data was previously captured and validated. Using paper-based methods resulted in the accumulation of erroneous data entries, thus compromising the quality of data in the database. The inability to incorporate skip logic and validation algorithms makes paper-based methods susceptible to errors. Hence, necessitated the implementation of the proposed EDC system. The EDC system positively influenced the accuracy of the data. The level of data accuracy was low prior to the implementation of the proposed framework; however, three months (Figure 5.9: October 2021) after the implementation of the framework, the data accuracy decreased by 3% from 70.2%. The drop was unexpected and that necessitated

a deeper analysis into the system to investigate what could have happened. This drop was attributed to the inability to validate non-South African identity numbers, errors introduced due to non-familiarity with the system, and faulty validation rules. In February 2022, the accuracy of the data improved by 11.6% as a consequence of the effective elimination of the principal causes of errors within the algorithms. The accuracy improved further by 19.7%, making overall accuracy fall within the acceptable range (95 – 100%). It is necessary to maintain the data accuracy within the tolerable range for good representation of real-world entities, and that may be achieved with the continued use of the proposed data quality framework. Data preloading was considered an approach to improve the consistency of the data in the current study. Data preloading played a crucial role in validating and cleansing the data, thereby enabling the framework to check for mismatches or inconsistencies. Since the data was synchronized between two data sources (PostgreSQL and MS SQL server databases), there would be the possibility of inconsistent data retrievals. However, data preloading ensured data consistency between different systems. That was achieved through the establishment of predefined data formats and standards, and contextual or reference data prepopulating. Predefined data formats and standards facilitate the standardisation of data structures, variable names, and other pertinent features, thus enhancing consistency. On the other hand, having the relevant reference or contextual data readily available alongside the primary data being collected allows for better-informed judgments and ensures data consistency. The experiment demonstrated 100 percent data consistency between different systems (EDC and database systems). The consistency of data was determined to be one of the most important metrics for data quality management in data integration, and the definition of data consistency should be refined at the operational level (Yang et al., 2017).

5.6 User Acceptance Test (UAT)

User Acceptance Testing (UAT) is an essential stage in the software development lifecycle during which the system's end-users assess whether the system satisfies their expectations and requirements. UAT's principal objective is to verify that the software system operates accurately in an authentic setting and fulfills the intended business requirements. The system was assessed by a group of fieldworkers, data managers, and data quality controllers in a real work environment. The assessment identified issues such as erratic skipping patterns and validation algorithms, bugs in API, and invalid paths configured in bash files. After the resolution of the observed problems, the system functioned at its peak, satisfying the requirements and expectations of its intended users before its implementation in a production setting. The EDC system was preferred by

around 95% of fieldworkers because of its efficient mechanism for validating and providing hints for erroneous data entries. Conversely, all data managers (100 percent) embraced 3-TTDQMF as a viable solution for persistent data quality challenges. An intriguing characteristic of this system was its automation, which rendered human interaction unnecessary when it came to verifying the integrity of data. Despite the data quality controllers' team being concerned about potential job losses as a result of the automation, they all agreed that it is the most effective method for addressing data quality issues. In general, intended users have expressed that the implementation of the system has significantly enhanced productivity and provided consistency in addressing data quality concerns.

5.7 Drawing Conclusions on Hypotheses

This work designed a framework to validate and improve the quality of data in the HDSS domain. To achieve that, the researcher predicted the outcomes of the study by drawing on 9 hypotheses as shown in Figure 5.10. The hypotheses were categorised into data quality validation and assurance, data quality metrics, and identification and monitoring of data anomalies. The first category considered hypotheses H1 and H2, while the second category grouped hypotheses H3 – H8. The last category included hypothesis H9. This study looked at data quality improvement strategies from different perspectives and was hypothesised as such. The first approach was to develop an engine to quality control the data received into the system, which is covered by Hypotheses H1 and H2. Once the data has been fed into the system, it becomes necessary to measure the level of its quality for quality control interventions. Hence, Hypotheses H3 – H8. Hypothesis H9 was drawn based on the significance of the identification and monitoring of data anomalies for data quality improvement. From the presentation of the results, all 9 hypotheses were separately supported.

From the data quality measurement point of view, the hypotheses such as H3, H4, H5, H6, and H7 were each supported. Depending on the organisation's choice of data quality measures, all selected metrics must be supported for overall data quality. It should be noted that data quality is a multidimensional concept; hence organisations are at liberty to select metrics that are deemed relevant for their data quality measurements (Borek et al., 2013). Therefore, in this study, hypotheses H3 – H7 were based on the selection of the following metrics: accuracy, completeness, consistency, timeliness, and validity. All hypotheses were supported and the results showed 95.7% overall data quality. According to the choice of data quality metrics in this research, all 5 hypotheses must be supported for the ultimate data quality. This satisfied the requirements for hypothesis H8, which

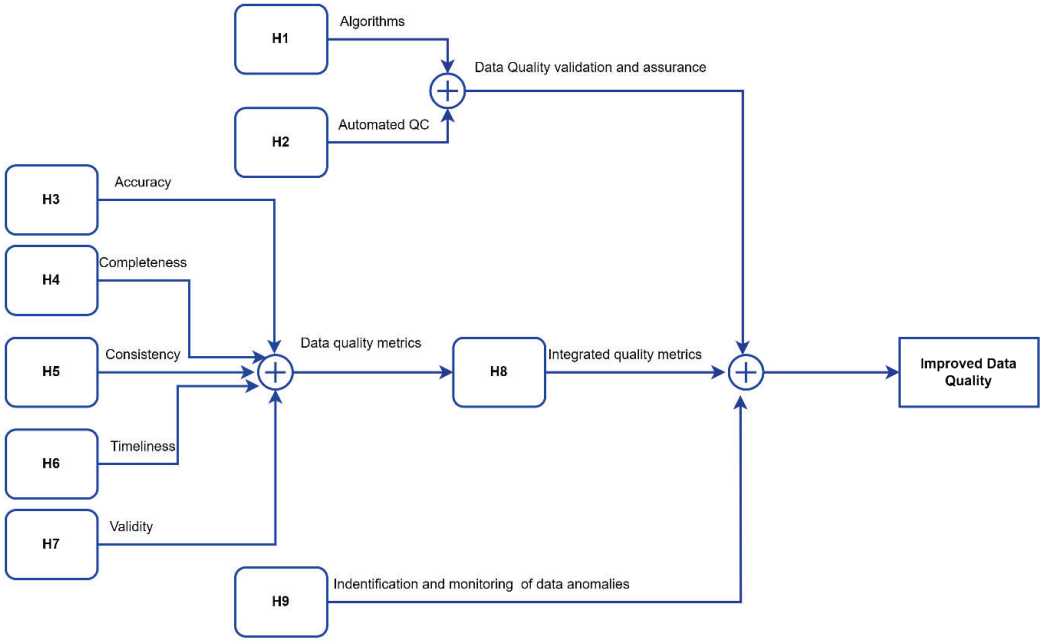


FIGURE 5.10: Categorization of hypotheses for data quality improvement

states that the quality of data can be improved if all hypotheses based on data quality measurements are supported. H8 further predicted that continuous monitoring of data quality metrics ultimately improves data quality. The deployment of a dashboard ensured continuous monitoring of data quality, helping practitioners identify and control the underlying cause of data quality issues. However, data quality metrics alone cannot guarantee data quality but merely provide a measure of quality. Therefore, other hypotheses were drawn. H1 and H2 were drawn based on the framework's ability to autonomously control the quality of the data. Both hypotheses were supported. H1 was based on the system's ability to flag data quality issues and suggest a plan of action to resolve such issues. The results have demonstrated that the implementation of skipping patterns and validation algorithms vastly improved the quality of the data. H2 on the other hand, complimented H1 in predicting efficiency in the deployment of automated data quality control techniques. The findings demonstrated that the automation of processes greatly improved the data quality by eliminating errors, monitoring occurrences of anomalies, and identification of sources of data issues. Hypothesis H9 draws its prediction of high-quality data based on the system's ability to identify and continuously monitor anomalies. The use of Pareto analysis and statistical control techniques in identifying data anomalies and monitoring process behaviour has been shown to improve the overall quality of data. The findings of this study validated the hypotheses made and have shown a significant improvement in data quality. The overall quality of the data before the implementation of the proposed framework was 77.3%, which improved by 18.4% after one year of the deployment of 3TTDQMF. The overall quality of the data

remained at 95.7%, one year after the implementation of the proposed framework, and the quality of the data is expected to continuously improve with the use of the system.

5.8 Chapter Summary

This chapter described the findings resulting from the strategies employed to enhance the data quality in HDSS. These strategies included the validation and skipping logic algorithms at the application level, automation of data quality control systems, continuous monitoring of data quality, and the application of statistical data quality control techniques. A major increase in data quality was observed, with trends indicating more improvements in the future. Furthermore, the automated data quality control strategies not only improved data quality but also enhanced system efficiency. The system autonomously validated data quality every midnight, relieving users of the burden of manually running the system. The effectiveness of the proposed system in improving the data quality was evaluated using metrics such as data accuracy, completeness, consistency, timeliness, and validity. Additionally, the Pareto analysis and Process control technique enhanced the data quality by identifying and monitoring the underlying cause of data quality issues that contributed to 80% of the problems. The quality of data significantly improved after one year of deploying the system. The proposed approach positively influenced the quality of data and has the potential to resolve various data quality problems encountered by data-driven organisations.

Chapter 6

Discussions and Implications of the study

6.1 Introduction

In this chapter, we conduct an in-depth discussion of the findings reported in the previous chapter. We discuss the implications of our findings, make comparisons to the prior literature, and acknowledge the limitations of our investigation.

This chapter is organised as follows: Section 6.2 briefly discusses the findings of the current study and Section 6.3 compares our findings with those in the existing literature. Section 6.4 presents the implications of the study and the application of the proposed system. Section 6.5 discusses the contributions of our work to the body of knowledge and Section 6.6 presents the limitations of this study.

6.2 Interpretation of Findings

The purpose of the study was to develop a system that incorporates the 3TTDQMF framework to manage and control the quality of the data collected. From the analysis of the results, the proposed framework produced positive results in improving data quality. The evaluation of data quality was carried out four times over a period of one year (June 2021 – June 2022) to assess the impact of the proposed framework. The results demonstrated that the quality of the data has improved since the implementation of the proposed framework. Several strategies, including the EDC system with quality assurance algorithms, automation, Pareto analysis, process control methodologies, and data quality assessments, were collectively used to improve data quality.

With the adoption of data validation algorithms at the application level and an automated data quality control system, data quality increased by 15% to 20%. Data completeness increased from 78% to 92%, while data accuracy improved from 70.2% to 98.2%. The timeliness of the data and validity improved by 19.1% and 11.6% to 95.8% and 96.7% respectively. These improvements were a result of the implementation of automated data quality control, Pareto analysis, and Process Control methods, which identified and mitigated data quality issues. Pareto analysis is commonly used in many areas of business, including marketing, sales, customer service, quality assurance, and manufacturing deficits (Craft and Leake, 2002). While analysis is applied to improve productivity and make well-informed decisions (Dunford and Tamang, 2014), the dearth of literature on the application of Pareto analysis and process control approaches to data quality management in HDSSs is evident. This is despite its potential benefits in quality management and improvement. Our study revealed that the adoption of Pareto analysis in HDSS increased productivity because a great deal of focus was placed on a few causes (a few vital ones) in resolving the majority of issues and because high-quality judgments were made regarding the causes of data quality concerns. Although the Pareto analysis provides a starting point for fixing problems, it does not account for common- and specific-cause variations. Correct identification of these variations is essential for ongoing monitoring, management, and improvement of data quality assurance systems. In order to account for common- and specific-cause variations, this work considered a Process control technique for monitoring and managing issues relating to data quality. Both variations provide a benchmark for future enhancements and evaluate the performance of a process over time. The process control approach offers information on the capacity and efficacy of the process to prevent defects or data errors. This method has been shown to increase productivity; however, to be effective, the process must be continuously monitored to identify any deviations as quickly as possible (Shah et al., 2010). Monitoring processes over time enables users or data managers to establish a benchmark against which future data quality performance may be measured and acted upon as appropriate. Although the approach offers a clear and universal vocabulary for discussing process performance and behaviour, data quality assurance ultimately depends on the data managers' plan of action (Suman and Prajapati, 2018). Data managers are responsible for identifying the special causes of variance and rectifying them. Correctly identifying and addressing special causes of variation may eventually result in enhanced data quality, which will make HDSSs more relevant entities for accurately addressing health and social concerns impacting communities.

The findings reported in the previous chapter align well with the five objectives proposed in this study. To achieve objective one, an EDC solution was proposed with

features to validate the quality of data at the point of entry. The algorithms used at the application level validated the data quality and notified fieldworkers of probable data quality violations, triggering an error message indicating the severity of the violation. A number of measures to deal with data quality issues such as skipping logic, data preloading, and C#-based validation algorithms were deployed. The approach improved not only the quality of data but, also the efficiency of data collection. In a pursuit to further improve data quality, the second objective was drawn. This objective was mainly achieved through the use of open-source technologies such as R-API, PDI, Bash programming language, SQL, and Windows task scheduler. These technologies were integrated and configured to automate data processing (export, extraction, transformation, loading, and cleaning). Data processing was previously performed manually, hence had longer turnaround times, and has been proven to be erroneous. However, the automated system improved overall system efficiency and eliminated the mistakes involved in manually processing the data. Thus, further improving the data quality.

In achieving the third objective, a novel framework named 3TTDQMF was implemented to autonomously receive the data as input, apply corrective measures, and allow only good-quality data to pass through to the database. The framework incorporates algorithms to check if the received data meets the agreed data quality standards and rejects it if not. Unlike in previous cases where users manually validate all mandatory variables, the automated data quality control system diligently performs such activities without human intervention.

Achieving the fourth objective required the implementation of a Python-based dashboard. The dashboard incorporated algorithms that measure the level of data quality in the database. The data quality metrics such as data accuracy, completeness, consistency, timeliness, and validity were considered to evaluate the effectiveness of the proposed framework. The comparison of data quality measurements was made before and after the implementation of the proposed framework. As the findings suggest, the 3TTDQMF greatly improved the quality of the data. In addition, the dashboard further enhanced the quality by including features to continuously monitor the levels of data quality. The fifth objective was accomplished by integrating into the dashboard, the Pareto analysis and Process control strategies. With Pareto analysis, the main contributors to data quality issues were identified and dealt with. Identifying data quality issues is the main step towards the implementation of corrective and future preventive measures. While it is necessary to identify data quality issues, accounting for common causes of variation in how errors are generated is also of paramount importance. Hence, the dashboard also incorporated Process control measures. This is to identify if the rate at which errors are generated is out of control and to plan the measures to keep

errors minimal. The dashboard allows data quality issues to be continuously identified, monitored, and controlled through the inclusion of data quality metrics, Pareto analysis, and Process control methods.

All the objectives achieved in this study, collectively, have a positive contribution to the data quality of data. Data quality assurance starts in the field during the data collection and is further applied as the data enters the system. At these stages, validation algorithms at the application level and intermediate, automated data quality control and processing systems play a crucial role in data cleaning. These stages complement each other and are integrated in a way that an error missed at one level can be detected at the next. Data from these levels will further be processed at the third level, where data quality issues are reported. The third level refers to the dashboard, which provides information on the level of data quality from the system. Continuous monitoring of the quality of data can result in much-improved data quality.

6.3 Comparison with Existing Literature

6.3.1 Data Quality Improvement

There is a need for robust health information systems that provide access to high-quality, timely, and accurate disaggregated data (World Health Organisation, 2021), especially in developing countries (Ali et al., 2018). Our solution takes this into account by including the validation, control, and monitoring algorithms required to ensure the integrity of the input and stored data. Automated routing of a complicated data system, error detection through range and consistency checks, and timely warnings to users about erroneous inputs are just a few of the ways our system ensures data completeness and accuracy (Zeleke et al., 2021). Similarly to accuracy, data completeness is a crucial aspect of data quality. It plays a crucial role in guaranteeing the completeness of the query responses (Nutt et al., 2012; Hannou et al., 2019) and in ensuring the accuracy of the analyses (Emran et al., 2013; Emran et al., 2014). Data completeness is calculated as the ratio of all missing values to the total values in the system subtracted from one. Keeping the ratio as small as possible results in good data quality in terms of completeness. The issue of missing values in the data is quite widespread and may arise at several levels of data processing, from data collection to data storage. Although allowing missing data into the database might result in a costly and time-consuming process, the necessary precautions must be taken to prevent such occurrences. Under possible circumstances, the software error prevention mechanism must be implemented to its full extent (Zeleke et al., 2021) to ensure the accuracy of the data and prevent

missing values. The assessment of data completeness is important not only to determine the data quality but also to verify the accuracy of results obtained from queries performed on insufficient datasets. In many decision-making systems, missing values pose a serious challenge, as making an informed choice relies on having all relevant data at hand (Azimi et al., 2019). Our research investigated data quality, and the results indicated that the proposed framework enhanced data completeness. With a good system for managing data, the quality of data increases gradually over time (Yoshihisa et al., 2018).

For noticeable improvement, data quality must be consistently checked quantitatively and/or qualitatively against data quality indicators in order to catch up with the organisation's information revolution objective (Endriyas et al., 2019). However, Hartzema et al., 2013 asserted that obtaining complete and trustworthy data is challenging; therefore, systems to review data quality should be ongoing and transparent, especially given the frequency with which data is updated. The transparent and ongoing data quality monitoring and assurance system takes into account the corrective measures that must be applied to the design defects and bugs in the system's algorithms. Gradually removing such defects can result in much-improved data quality. However, ideal data quality should not be the objective; rather, data quality should be improved to a predetermined degree (Haug et al., 2011). The preset level of data quality in the current study was 95 – 100%. The goal was to maintain the measure of data quality within this interval.

The authors (Van den Berghe and Van Gaeveren, 2017) advanced their knowledge in the subject of data quality management by presenting a thorough framework and evaluation of proactive data quality management, as well as a decision model for determining the appropriate data quality improvement strategy. Compliance and stakeholder satisfaction were also included in the evaluation, which prioritised cleaning activities along many characteristics, such as reusability and complexity. On the contrary, Sebastian-Coleman, (2012) presented a data quality assessment framework that aids in measuring and observing data quality across time. It consisted of about three dozen measurement types associated with five objective quality dimensions: consistency, validity, timeliness, integrity, and completeness. The framework helps organisations prioritize measurements and effectively report the outcomes, as well as providing techniques to use data measurements to manage and enhance data quality. Unlike in our study, the article does not include step-by-step guidance for adopting the framework; therefore, organizations may need to tailor the framework to their own requirements and resources. In addition, the study presupposes a certain degree of expertise and familiarity with data quality concepts; therefore, it may not be appropriate for novices in the subject.

To evaluate data error rates and enumerator performance during electronic data collection, the research presents an approach dubbed "validation relaxation" (Kenny et al., 2017). The method involves the omission of data validation elements for some questions so that mistakes can be made when capturing data and then tracked and corrected. Validation relaxation can detect individual enumerators or systemic data issues that are amenable to enumerator training and should be addressed as part of a comprehensive data quality assurance approach. Instead of intentionally omitting validation rules through validation relaxation, our study used a different approach. Pareto analysis was used to identify fieldworkers responsible for a disproportionate share of data quality concerns, while process control measures maintained error rates within acceptable limits. We argue that validation relaxation is more likely to compromise data quality than to improve it. Finally, their research lacks a comprehensive analysis of the sorts of mistakes produced by the enumerators, which would have offered additional insight into the efficacy of the validation relaxation technique. The research by Wang et al. (2019) provides a unique technique for improving data quality based on the greedy algorithm, which has been shown to enhance data quality while lowering time and computing costs. This method has resulted in more reliable and accurate results. Although greedy algorithms have been widely used for data quality improvement, their major limitation is the longer running time, especially for a larger dataset. This is because, for each greedy step, the algorithm must update a model or calculate a function using the previously selected options and the new candidate (Khanna et al., 2017). Given the longitudinal nature of HDSS data, the longer the running times during data quality assessment, the longer the data turnaround time.

Musu et al., 2020 presented an overview of quality control techniques, including the preparation of standardised manuals, national and international quality control programs, and the monitoring of data collection activities to ensure high-quality data and permit cross-country comparisons of research outcomes. While such documentation is an essential part of data quality management, it is important to note that the quality control approaches described by Musu et al. are based on a specific area, which may not necessarily be applicable to the HDSS context. The study gives an overview of the quality control techniques but does not evaluate their performance or potential downsides in depth.

The findings of our study are in line with the findings of other studies in the literature. Gains in data quality improvements were reported; however, some studies did not explicitly test the practicality of their experiments in real-world settings. Although the literature reported an improvement in data quality, our work stands out by implementing an automated data quality control engine. The engine improves not only data

quality and productivity but also the efficiency of the system.

6.3.2 Automation of Processes for Quality Improvement

Automating operations greatly minimised turnaround times and relieved the user of the burden of manually running the system. While previous research on automation in management accounting (Knauer et al., 2020; Gorla et al., 2010), manufacturing industries (Adrita et al., 2021; Madakam et al., 2019), equipment maintenance (Wang et al., 2012; Zhao et al., 2012), railway services (Balfe et al., 2015), mass production industries (Jämsä-Jounela, 2007), digital forensics (Asquith and Horsman, 2019), and auditing (Moffitt et al., 2018) have yielded good results, the dearth of literature on HDSS process automation remains evident. Due to HDSS's distinctive operational environment, industrial automation methods are not immediately applicable. Therefore, new strategies are necessary to take advantage of automation in HDSS domains. The numerous advantages of automation make it a particularly attractive option for firms seeking to improve productivity and data quality while reducing costs. To bridge the knowledge gap in HDSS literature, this study used open-source technologies (R-API, PDI, and Windows Bash programming languages) and other methodologies to integrate and design an automated system that seamlessly performed tasks that previously required human interventions. The finding of the current study indicated that the automated system reduced turnaround times and time-wasting activities, eliminated errors in data processing, eased workflow integration, and enabled near-real-time data processing and management. Process automation provides a multitude of benefits.

Automated processes provide systematic procedures and advanced technology to efficiently handle and label data, culminating in the regularisation and augmentation of data quality (Lorenz Simoes et al., 2021). Herrick and Tyndall (2013) showed that automation decreased effort and time wasting while enhancing deployments, and Balfe et al. (2015) found that automation led to a decrease in workload and consistently maintained peak system performance. Automation, on the other hand, has been shown to increase both the efficiency and quality of IT services (Krishnan and Ravindran, 2017), while Brown and Keller (2006) found that automation could speed up the deployment of complicated Java-based business applications. Generally, automation of processes improves the efficacy and efficiency of operations (Grover et al., 2018). In addition, automation facilitates expedited calculations, less human error, and better system-wide database coordination and integration (Häkkinen and Hilmola, 2008). Without user interaction, information can be stored, accessed, processed, and disseminated. Ideally, the resultant operations are more efficient, less prone to mistakes, simpler to execute, and

more consistent (Bravo et al., 2016). Furthermore, data input automation minimises data errors (Covington et al., 2020), significantly reduces delays in entering data into the system, and enhances work satisfaction (Bauer et al., 2019). Data transfer durations per measurement event were reduced from 5 minutes to 2 hours, and data error rates fell from around 20% to 0%. In our study, automation minimised waiting times from approximately 40 minutes to 0. This was particularly true because the automation was scheduled to run overnight and users had no waiting times the next day.

The automation system is capable of classifying both good and bad data and relies on human judgment to categorise "grey area" instances (Borisyak et al., 2017). The study reveals that the suggested workflow is capable of autonomously processing at least 20 percent of samples without observable result deterioration. Only 20% of the samples can be processed automatically using the suggested workflow; therefore, a substantial proportion of the data must still be categorised manually by human specialists. The findings of the study by (Bauer et al., 2020) showed that automating data input improved efficiency, productivity, and employee satisfaction without increasing costs. The study also reported that automation minimised data omission and eliminated data input mistakes. Therefore, it is necessary to automate previously manually operated tasks to improve the quality of operations that subsequently result in the production of high-quality data.

Automation has the capability to improve data quality, however, it may also pose some challenges. One difficulty is the necessity for constant enhancement of database and data processing quality (Borisyak et al., 2017). This involves examining the timeliness and value-added characteristics of the data and correcting disparities in the quality of data collected by different data collectors. Another difficulty is the segmentation of data into distinct categories, such as good, poor, and "grey area" instances, which may need human expert judgment (Harahap, 2019).

6.4 Implications and Applications

This work has important implications not only for HDSSs but also for other data-driven organisations that seek to manage and improve the quality of their data. The findings of this work posit that an integrated and automated information system offers organisations the opportunity to manage and report data quality effectively. Integrating data from disparate sources eliminates difficulties in managing data distributed across various data sources, reduces data security risks, and maintains the integrity and consistency of the data. In addition, automating data integration and other manual processes provides a multitude of benefits to the organisation, as it reduces the costs associated with human

resources, eliminates errors, improves performance, productivity, and competitiveness. It should be noted that HDSSs or data-driven organisations attract funding through the upkeep of high-quality data. Hence, the proposed framework incorporates mechanisms to allow organisations to effectively manage the quality of input data and produce data of better quality. The use of reporting systems enables organisations to continuously monitor and control the quality of their data. Furthermore, the use of statistical approaches in the reporting system ensures effective mitigation of factors that negatively influence data quality. In doing so, organisations are able to identify sources of issues that affect data quality and eliminate them for the ultimate quality of the data.

6.5 Contribution to New Knowledge

6.5.1 Methodological and Practical Contribution

This work contributes to the body of knowledge by drawing attention to the relevance of automation and integration, data quality assurance, the Pareto principle, and the Process control approach for HDSS practise. The information system that incorporates these features has various repercussions on the quality of data in HDSS. First, the system incorporates a novel 3TTDQMF, which takes the input data, cleans, processes, and generates data of superior quality. High-quality data in HDSS has many benefits including attracting funding (since most of the HDSSs rely on external funding), producing high-quality research outputs, making well-informed decisions, drafting accurate governmental policies and strategies to improve the livelihood of communities, drawing an accurate picture of health and social issues affecting communities. Second, process automation was proposed to eliminate previously applicable human data manipulation, transfer, and processing procedures. Manual handling contributes to prolonged processing times, poor system performance, and erroneous entries. Having an automated HDSS system allows the user to concentrate more on tasks that are difficult to automate, resulting in increased productivity. Furthermore, the automated data quality control system replaced human data quality control operations, which dramatically improved turnaround times, efficiency, and data quality. Third, while techniques applied for continuous monitoring of data quality in HDSS have received less attention in the literature, this study considered a unique approach by incorporating the dashboard, the Pareto analysis, and Process control techniques to identify and manage the factors that contribute to data quality concerns. These approaches improved the quality of the data by identifying vital few and accounting for variations in error generation. Additionally, five data quality metrics (accuracy, completeness, consistency, timeliness, and validity)

were examined to evaluate the effectiveness of the proposed system. The systems' ability to improve HDSS data quality is a crucial aspect of its quality. Finally, this work provides the technical information necessary to help practitioners implement the technology utilised in comparable contexts. We further shared the details on the models, algorithms, and configurations for HDSSs that wish to apply comparable methodologies to verify data quality.

The system proposed in this study was deployed in real real-world setting to effectively deal with data quality issues that were previously encountered. Since the introduction of the system in DIMAMO HDSS, the quality of the data has dramatically improved as a result of the use of several data quality enhancement methodologies. Workflow automation not only improved the efficiency but also the quality of data by eliminating laborious and erroneous manual operations. In the contemporary HDSS, the influence and applicability of this study have had a favourable effect on data management operations.

6.5.2 Theoretical Contribution

Theoretical contributions offer the intellectual framework upon which empirical research is conducted and contribute to the advancement of theoretical knowledge of a subject. To better understand and explain phenomena, this work proposed a novel framework to autonomously manage the quality of data in the HDSS domain. The framework presents a theoretical perspective through which practitioners and academics can view and comprehend the subject of data quality management. This study also developed the models, which involved integrating and synthesizing previous theories to construct a more complete and cohesive knowledge of data quality improvement strategies.

6.5.3 Contribution to Database Practises

Contributions to the area of database systems might include a vast array of breakthroughs and inventions that improve data storage, management, and querying of data. These contributions are essential to enhancing the effectiveness, scalability, and dependability of database systems. This study improved the effectiveness of data transfer and extraction from various databases by automating data streams. This process involved integrating data from PostgreSQL and MS SQL server databases to account for data consistency and integrity.

6.6 Limitations

The data collection platform proposed for this study is incompatible with devices running the iOS operating system, limiting its applicability to various organisations. The proposed framework was deployed on the Windows operating system, and major modifications may be required for other operating systems. In this review, a limited number of metrics were considered to evaluate the performance of the proposed system. In the case of a comprehensive experiment, the use of machine learning (ML) automation provides an effective means of data quality monitoring, facilitating the automatic classification of data as either good or bad, while relying on human expert judgment for cases that are ambiguous (Borisyak et al., 2017).

Artificial intelligence (AI) and machine learning (ML) algorithms have had a substantial influence on the field of quality control by providing more efficient and accurate methods to monitor and improve product quality. There are a number of ways AI and ML algorithms, for example, anomaly detection, adaptive learning, and self-improvement, automated inspection, anomaly detection, root cause analysis, process optimisation, defect detection, and classification. These provide an extent to which AI and ML algorithms can be applied to the quality control of data. Given the magnitude of the data generated by HDSSs, it may be necessary to apply AI and ML approaches to intelligently detect and correct anomalies. The algorithms applied in our study lack the intelligence inherent in AI and ML, which makes adaptability very difficult. Although the algorithms applied in our study effectively identified and improved the data quality, adaptive learning, and self-improvement are difficult to achieve. Machine learning models can adapt and improve over time as they receive more data, leading to better accuracy in defect detection and quality assessment. With continued use, the AI and ML models can adapt and improve over time, resulting in improved defect identification and quality evaluation.

6.7 Chapter Summary

The primary emphasis of the discussion chapter was on the critical data quality issues and the approaches to deal with them. This subject is of the utmost significance since the dependability and precision of data immediately affect the decision-making for data-driven organisations. Several major insights emerged from our investigation. Firstly, briefly assessed the effect of the proposed framework with the findings suggesting a positive influence. Our efforts to improve the quality of data involved employing stringent data validation methods, guaranteeing data integrity throughout the data collection

phase, cross-checking data from numerous sources, and eliminating error-prone processes through automation. Our work highlights the need for upfront reporting of data quality parameters so that other researchers may evaluate the data's trustworthiness and duplicate the study if necessary. Our evaluation of the data quality assurance framework has far-reaching implications. By prioritizing data quality, HDSS can formulate evidence-based policies and allocate resources, accurately monitor disease outbreaks and emerging health issues, identify the patterns and changes in health and demographics, produce valid and credible research outputs, accurately identify vulnerable populations, and attract funding. The HDSS can deploy the framework proposed in this work in a real-world environment to improve efficiency and overall data quality. Guaranteeing good data quality requires rigorous data collection methodologies, validation procedures, and continuing quality control techniques, all of which contribute to the dependability and credibility of the collected data.

Chapter 7

Conclusion and Future Work

7.1 Conclusion

Data quality is an integral component of any Health and Demographic Surveillance System (HDSS). It is crucial to ensure data quality in an HDSS in order to create accurate and trustworthy information for research, policymaking, and public health initiatives. Data quality in HDSS is a complex problem requiring training, planning, administration, and regular monitoring. This work proposed a system to effectively deal with issues affecting data quality, helping practitioners to identify, monitor, and control the quality of data.

The system was set up on a Windows 2016 server and configured to allow traffic from Android-based tablets, which were used to collect data from the field. Quality controlling data involved setting up a Survey Solutions® Server to provide capacity for the development and hosting of electronic data collection (EDC). Validation algorithms or *C#*-based Macros, skipping pattern algorithms and data preloading were data quality control strategies used to manage the quality of data at the application level (EDC system). This method ensured high-quality data at the point of entry. Dealing with data quality issues at the application level had a number of benefits; Issues were identified before crawling into the database, and appropriate actions were taken. Using automated validation algorithms, data quality concerns were identified and presented to the fieldworkers, allowing for immediate resolution. It is generally less expensive to resolve issues during data entry than later in the database.

A novel framework called 3-Tier Total Data Quality Management Framework (3TTDQMF) was designed and deployed to quality control the data before saving it to the database. The framework deployed an engine that received data as input and executed algorithms

to quality control the data. The data quality control engine was automated to validate the quality of data without users' interventions. That allowed users to focus on other important tasks, thereby improving productivity and efficiency. Other automated tasks were data export, transfer, processing, and loading. The automation was performed using open-source platforms such as R studio (R-API), PDI, Bash programming, Windows task scheduler, and SQL. Automation has been shown to improve productivity and efficiency, minimised data errors or mistakes, and improve data turnaround times.

The framework was evaluated on its ability to enhance the data quality, with the following metrics used: data accuracy, completeness, consistency, timeliness, and validity. The results have shown that the data quality improved since the implementation of the proposed framework. To ensure continuous data quality improvement, this study built a dashboard to report on the level of data quality. In addition to data quality metrics, the dashboard included Pareto analysis and a Process control strategy to identify, monitor, and control the instances of data quality concerns. The approaches have been shown to improve the data quality by effectively identifying and monitoring the causes of most of the errors in the system. In the context of public health and demographic research, it is impossible to overstate the importance of HDSS data for monitoring health outcomes, assessing interventions, and comprehending demographic trends. Our research emphasizes that guaranteeing data quality in HDSS is not only a technical detail but a fundamental concept that underlies the integrity and dependability of the generated data. Our approaches to data quality control can help practitioners and academics in the pursuit of high-quality data that promotes good change in public health and demography as HDSS continues to improve and increase its influence. Our research has shed light on a number of important factors and recommended procedures for preserving data quality within the framework of an HDSS.

7.2 Future Research Directions

Data quality and integrity assurances take on increased importance as businesses increasingly rely on data-driven decision-making. Traditional data quality management approaches are typically insufficient to manage the magnitude, complexity, and dynamic nature of contemporary data sources. This provides an opportunity to explore innovative ways to leverage AI and ML approaches for more efficient and effective data quality control. The future work aims to contribute to the creation of a complete framework that can automatically detect, fix, and avoid data quality concerns across several domains and applications by combining the power of modern algorithms with domain expertise.

By leveraging AI and ML methodologies, a more robust and adaptive data quality control framework can be built, leading to more reliable, complete, timely, consistent, and accurate data.

Bibliography

Aguirre, S. and Rodriguez, A., 2017. Automation of a business process using robotic process automation (RPA): A case study. In Applied Computer Sciences in Engineering: 4th Workshop on Engineering Applications, WEA 2017, Cartagena, Colombia, September 27-29, 2017, Proceedings 4, Springer International Publishing, pp. 65-71.

Acharya, B., Jena, A.K., Chatterjee, J.M., Kumar, R. and Le, D.N., 2019. Nosql database classification: new era of databases for big data. International Journal of Knowledge-Based Organizations (IJKBO), 9(1), pp.50-65.

Adedini, S.A., Thaele, D., Sello, M., Mutevedzi, P., Hywinya, C., Ngwenya, N., Myburgh, N. and Madhi, S.A., 2021. Approaches, achievements, challenges, and lessons learned in setting up an urban-based Health and Demographic Surveillance System in South Africa. Global Health Action, 14(1), p.1874138.

Adrita, M.M., Brem, A., O'Sullivan, D., Allen, E. and Bruton, K., 2021. Methodology for Data-Informed Process Improvement to Enable Automated Manufacturing in Current Manual Processes. Applied Sciences, 11(9), p.3889.

Agarwal, D., Dhotre, D., Patil, R., Shouche, Y., Juvekar, S. and Salvi, S., 2017. Potential of health and demographic surveillance system in asthma and chronic obstructive pulmonary disease microbiome research. Frontiers in public health, 5, p.196.

Akerkar, R., 2013. Advanced data analytics for business. Big Data Computing. Boca Raton, FL: Chapman and Hall/CRC, pp.377-9.

Al-Ababneh, M., 2020. Linking ontology, epistemology and research methodology. Science & Philosophy, 8(1), pp.75-91.

Alberts, M., Dikotope, S.A., Choma, S.R., Masemola, M.L., Modjadji, S.E., Mashinya, F., Burger, S., Cook, I., Brits, S.J. and Byass, P., 2015. Health & demographic surveillance

Alghamdi, A.H. and Li, L., 2013. Adapting design-based research as a research methodology in educational settings. International Journal of Education and Research, 1(10), pp.1-12.

Al-Hajj, S., Pike, I. and Fisher, B., 2013, November. Interactive dashboards: Using visual analytics for knowledge transfer and decision support. In Proceedings of the 2013 Workshop on Visual Analytics in Healthcare, Washington, DC, USA (Vol. 16).

Ali, S.M., Naureen, F., Noor, A., Kamel Boulos, M.N., Aamir, J., Ishaq, M., Anjum, N., Ainsworth, J., Rashid, A., Majidulla, A. and Fatima, I., 2018. Data quality: a negotiator between paper-based and digital records in Pakistan's TB control program. Data, 3(3), p.27.

Anagnoste, S., 2017. Robotic Automation Process The next major revolution in terms of back office operations improvement. In: Proceedings of the International Conference on Business Excellence. De Gruyter, Bucharest. pp. 676–686.

Arikpo, I., Okoro, A., Esu, E., Aquaisua, E., Ekinya, I. and Meremikwu, M., 2019. Differences in Population Dynamics and Uptake of Reproductive Health Services in the Urban and Rural Cohorts of Cross River Health and Demographic Surveillance System of Southern Nigeria. Developing Country Studies, 9(5), pp.64-71

Ary, D., Jacobs, L.C., Sorensen, C. and Razavieh, A., 2010. Introduction to research in education . United States: Wadsworth, Cengage Learning. In International Conference on English Language Teaching (pp. 720-729).

Asatiani, A. and Penttinen, E., 2016. Turning robotic process automation into commercial success–Case OpusCapita. Journal of Information Technology Teaching Cases, 6(2), pp.67-74.

Asquith, A. and Horsman, G., 2019. Let the robots do it!–Taking a look at Robotic Process Automation and its potential application in digital forensics. Forensic Science International: Reports, 1, p.100007.

AssistEdge, 2021. AssistEdge RPA OpenSource Community. Available from: https://www.edgeverve.com/assistedge/community/

Automagica, 2021. Automagica Documentação. Available from: https://automagica.readthedocs.io/index.html and https://github.com/automagica/automagica/wiki/Documentation

Azeroual, O., Saake, G. and Schallehn, E., 2018. Analyzing data quality issues in research information systems via data profiling. International Journal of Information Management, 41, pp.50-56.

Azeroual, O. and Abuosba, M., 2019. Improving the data quality in the research information systems. arXiv preprint arXiv:1901.07388.

Azimi, I., Pahikkala, T., Rahmani, A.M., Niela-Vilén, H., Axelin, A. and Liljeberg, P., 2019. Missing data resilient decision-making for healthcare IoT through personalization: A case study on maternal health. Future Generation Computer Systems, 96, pp.297-308.

Balfe, N., Sharples, S. and Wilson, J.R., 2015. Impact of automation: Measurement of performance, workload and behaviour in a complex control environment. Applied ergonomics, 47, pp.52-64.

Baschieri, A., Gordeev, V.S., Akuze, J., Kwesiga, D., Blencowe, H., Cousens, S., Waiswa, P., Fisker, A.B., Thysen, S.M., Rodrigues, A. and Biks, G.A., 2019. "Every Newborn-INDEPTH"(EN-INDEPTH) study protocol for a randomised comparison of household survey modules for measuring stillbirths and neonatal deaths in five Health and Demographic Surveillance sites. Journal of Global Health, 9(1).

Bassil, Y., 2012. A data warehouse design for a typical university information system. arXiv preprint arXiv:1212.2071.

Batini, C. and Scannapieco, M., 2016. Data and information quality. Cham, Switzerland: Springer International Publishing.

Batini, C., Cappiello, C., Francalanci, C. and Maurino, A., 2009. Methodologies for data quality assessment and improvement. ACM computing surveys (CSUR), 41(3), pp.1-52.

Bauer, J.C., John, E., Wood, C.L., Plass, D. and Richardson, D., 2020. Data entry automation improves cost, quality, performance, and job satisfaction in a hospital nursing unit. JONA: The Journal of Nursing Administration, 50(1), pp.34-39.

Bennett, D.A., 2001. How can I deal with missing data in my study? Australian and New Zealand Journal of public health, 25(5), pp.464-469.

Benneyan, J.C., 1998. Use and interpretation of statistical quality control charts. International Journal for Quality in Health Care, 10(1), pp.69-73.

Booth, P., Matolcsy, Z. and Wieder, B., 2000. The impacts of enterprise resource planning systems on accounting practice–the Australian experience. Australian Accounting Review, 10(22), pp.4-18.

Borek, A., Parlikad, A.K., Webb, J. and Woodall, P., 2013. Total information risk management: maximizing the value of data and information assets. Newnes.

Borisyak, M., Ratnikov, F., Derkach, D. and Ustyuzhanin, A., 2017, October. Towards automation of data quality system for CERN CMS experiment. In Journal of Physics: Conference Series (Vol. 898, No. 9, p. 092041). IOP Publishing.

Boros, T., 2020. Survey of indoor navigation solutions, 2, pp.84-89, doi: 10.35925/J.MULTI.2020.2.11

Bravo, E.R., Santana, M. and Rodon, J., 2016. Automating and informating: Roles to examine technology's impact on performance. Behaviour & Information Technology, 35(7), pp.586-604.

Brewerton, P.M. and Millward, L.J., 2001. Organizational research methods: A guide for students and researchers. Sage.

Brunette, W., Sudar, S., Sundt, M., Larson, C., Beorse, J. and Anderson, R., 2017, June. Open Data Kit 2.0: A services-based application framework for disconnected data management. In Proceedings of the 15th Annual International Conference on Mobile Systems, Applications, and Services (pp. 440-452).

Brunette, W., Sundt, M., Ginsburg, A. and Borriello, G., 2013, December. Customizing and improving medical workflows using ODK survey. In Proceedings of

the 4th Annual Symposium on Computing for Development, pp. 1-2.

Butuza, A., Hauer, I., Muntean, C. and Popa, A.S., 2011. Increasing the Business Performances using Business Intelligence. Analele Universitatii" Eftimie Murgu" Resita Fascicola de Inginerie, 3(XVIII), pp.67-72.

Bygstad, B., 2017. Generative innovation: a comparison of lightweight and heavyweight IT. Journal of Information Technology, 32(2), pp.180-193.

Cai, L. and Zhu, Y., 2015. The challenges of data quality and data quality assessment in the big data era. Data science journal, 14.

Camera, F., Erkoyuncu, J.A. and Wilding, S., 2020. Service Data Quality Management Framework to Enable Through-life Engineering Services. Procedia Manufacturing, 49, pp.206-210.

Cappiello, C., Francalanci, C., Pernici, B., Plebani, P. and Scannapieco, M., 2003. Data quality assurance in cooperative information systems: a multi-dimension quality certificate. In International Workshop on Data Quality in Cooperative Information Systems (DQCIS 2003-ICDT 2003), pp. 47-54.

Castro, P., Melnik, S. and Adya, A., 2007, June. ADO. NET entity framework: Raising the level of abstraction in data programming. In Proceedings of the 2007 ACM SIGMOD international conference on Management of data (pp. 1070-1072).

Chen, C.C., 2011. Quantitative methodology: Appropriate use in research for blind baseball ergonomics and safety design. The Journal of Human Resource and Adult Learning, 7(1), p.1.

Chen, F. and Hsu, M., 2013, March. A performance comparison of parallel DBMSs and MapReduce on large-scale text analytics. In Proceedings of the 16th International Conference on Extending Database Technology, pp. 613-624.

Claveau, J., Reedman, L. and Yang, X., 2010. Business survey data collection research at statistics canada.

Choy, L.T., 2014. The strengths and weaknesses of research methodology: Comparison and complimentary between qualitative and quantitative approaches. IOSR journal of humanities and social science, 19(4), pp.99-104.

Cohen, L., Manion, L. and Morrison, K.R.B., 2007. Research Methods in Education. 6th ed. UK: Routledge, Oxon, ISBN-10: 0415368782, ISBN-13: 978-0415368780, pp. 657.

Collis, J. and Hussey, R., 2013. Business research: A practical guide for undergraduate and postgraduate students. Macmillan International Higher Education.

Connolly, T.M. and Begg, C.E., 2005. Database systems: a practical approach to design, implementation, and management. Pearson Education.

Coronel, C., Crockett, K., Morris, S. and Craig, Blewett C., 2020. Database Principles - Fundamentals of Design, Implementation, and Management, 3rd ed., Cengage Learning (Emea) Ltd, ISBN: 978-1-4737-6804-8

Corrales Múñoz, D.C., Ledezma Espino, A.I. and Corrales, J.C., 2018. From Theory to Practice: A Data Quality Framework for Classification Tasks.

Covington, E.L., Popple, R.A. and Cardan, R.A., 2020. Use of automation to eliminate shift errors. Journal of Applied Clinical Medical Physics, 21(3), pp.192-195.

Craft, R.C. and Leake, C., 2002. The Pareto principle in organizational decision making. Management Decision.

Creswell, J.W. and Creswell, J.D., 2017. Research design: Qualitative, quantitative, and mixed methods approaches. Sage publications.

Crotty, M., 1998. Introduction: The research process. The foundations of social research: Meaning and perspective in the research process, pp.1-17.

DAMA, U., 2013. The Six Primary Dimensions for Data Quality Assessment-Defining Data Quality Dimensions. Bristol: np URL: https://www. whitepapers. em360tech.com/wpcontent/files_mf/1407250286DAMAUKDQDimensionsWhitePaperR37. pdf, access date, 2021.

Daniil, L., 2022. Using GPS-Paradata to Control the Data Collection Process: Review of Existing Methods and Analysis of GPS-Paradata Quality. Sociologičeskij žurnal, doi: 10.19181/socjour.2022.28.4.9313

Date, C.J. and Date, C.J., 2019. MVDs and 4NF. Database Design and Relational Theory: Normal Forms and All That Jazz, pp.241-260.

De Amicis, F., Barone, D. and Batini, C., 2006, November. An Analytical Framework to Analyze Dependencies Among Data Quality Dimensions. In ICIQ, pp. 369-383.

De Vries, A. and Reneau, J.K., 2010. Application of statistical process control charts to monitor changes in animal production systems. Journal of Animal Science, 88(suppl_13), pp.E11-E24.

Demba, M., 2013. Algorithm for relational database normalization up to 3NF. International Journal of Database Management Systems, 5(3), p.39.

Den Hertog, D. and Postek, K., 2016. Bridging the gap between predictive and prescriptive analytics-new optimization methodology needed. Tilburg Univ, Tilburg, The Netherlands.

DivyaYadav, D. and Choudhary, N., 2021. Business intelligence for local mining company acknowledgement reporting system. Turkish Journal of Computer and Mathematics Education (TURCOMAT), 12(14), pp.732-738.

Dong, Y. and Peng, C.Y.J., 2013. Principled missing data methods for researchers. SpringerPlus, 2(1), pp.1-17.

Dunford, R., Su, Q. and Tamang, E., 2014. The pareto principle.

Easterby-Smith, M., Thorpe, R., Jacson, P. and Lowe, A., 2008. Management Research: An Introduction. 3rd ed. Los Angeles, London, New Delhi, Singapore, Washington DC: SAGE Publications, ISBN-10: 1847871771, ISBN-13: 978-1847871770, pp.368.

Eckerson, W.W., 2010. Performance dashboards: measuring, monitoring, and managing your business. John Wiley & Sons.

Edris Abadi, R., Ershadi, M.J. and Niaki, S.T.A., 2022. A clustering approach for data quality results of research information systems. Information Discovery and Delivery.

Ehrlinger, L. and Wöß, W., 2017, October. Automated data quality monitoring. In Proceedings of the 22nd MIT International Conference on Information Quality (ICIQ 2017), pp. 15-1.

Ehrlinger, L., Rusz, E. and Wöß, W., 2019. A survey of data quality measurement and monitoring tools. arXiv preprint arXiv:1907.08138.

Emran, N.A., Embury, S. and Missier, P., 2014. Measuring population-based completeness for single nucleotide polymorphism (SNP) databases. In Advanced Approaches to Intelligent Information and Database Systems (pp. 173-182). Springer, Cham.

Emran, N.A., Embury, S., Missier, P., Isa, M.N.M. and Muda, A.K., 2013, March. Measuring data completeness for microbial genomics database. In Asian conference on intelligent information and database systems (pp. 186-195). Springer, Berlin, Heidelberg.

Endriyas, M., Alano, A., Mekonnen, E., Ayele, S., Kelaye, T., Shiferaw, M., Misganaw, T., Samuel, T., Hailemariam, T. and Hailu, S., 2019. Understanding performance data: health management information system data accuracy in Southern Nations Nationalities and People's Region, Ethiopia. BMC health services research, 19(1), pp.1-6.

Epelde, G., Beristain, A., Alvarez, R., Arrúe, M., Ezkerra, I., Belar, O., Bilbao, R., Nikolic, G., Shi, X., De Moor, B. and Mulvenna, M., 2020. Quality of data measurements in the big data era: Lessons learned from MIDAS project. IEEE Instrumentation & Measurement Magazine, 23(7), pp.18-24.

Eppler, M.J. and Wittig, D., 2000. Conceptualizing Information Quality: A Review of Information Quality Frameworks from the Last Ten Years. IQ, 20(0), p.0.

Eppler, M.J. and Wittig, D., 2000. Conceptualizing Information Quality: A Review of Information Quality Frameworks from the Last Ten Years. IQ, 20(0), p.0.

Even, A. and Shankaranarayanan, G., 2005, November. Value-Driven Data Quality Assessment. In ICIQ.

Farooq, M.U., Eizad, A. and Bae, H.K., 2023. Power solutions for autonomous mobile robots: A survey. Robotics and Autonomous Systems, 159, pp.104285.

Fernandez, D. and Aman, A., 2018. Impacts of robotic process automation on global accounting services. Asian Journal of Accounting & Governance, 9.

Friedman, T. and Judah, S., 2013. The state of data quality: Current practices and evolving trends. Stamford: Gartner. https://www.gartner.com/doc/2636315/state-data-quality-current-practices

Fu, Q. and Easton, J.M., 2017, December. Understanding data quality: Ensuring data quality by design in the rail industry. In 2017 IEEE International Conference on Big Data (Big Data) (pp. 3792-3799). IEEE.

Fung, H.P., 2014. Criteria, use cases and effects of information technology process automation (ITPA). Advances in Robotics & Automation, 3.

Gabernet, A. and Limburn, J., 2017. Breaking the 80/20 rule: How data catalogs transform data scientists' productivity.

Garbett, D., 2011. Constructivism deconstructed in science teacher education. Australian Journal of Teacher Education, 36(6), pp.36-49.

Gattiker, T.F. and Goodhue, D.L., 2004. Understanding the local-level costs and benefits of ERP through organizational information processing theory. Information & management, 41(4), pp.431-443.

Ge, J., Han, W., Zhang, X. and Zhou, J., 2020, April. Research on Construction of Quality Service Platform of Survey and Mapping. In Proceedings of the 2020 3rd International Conference on Geoinformatics and Data Analysis, pp. 57-61.

Gejke, C., 2018. A new season in the risk landscape: Connecting the advancement in technology with changes in customer behaviour to enhance the way risk is measured and managed. Journal of Risk Management in Financial Institutions, 11(2), pp.148-155.

Geyer-Klingeberg, J., Nakladal, J., Baldauf, F. and Veit, F., 2018, July. Process Mining and Robotic Process Automation: A Perfect Match. In BPM (Dissertation/Demos/Industry), pp. 124-131.

Ghilic-Micu, B., Mircea, M. and Stoica, M., 2010. The audit of business intelligence solutions. Informatica Economica, 14(1), p.66.

Gitzel, R., Turring, S. and Maczey, S., 2015, July. A data quality dashboard for reliability data. In 2015 IEEE 17th Conference on Business Informatics, IEEE. (Vol. 1), pp. 90-97

Given, L.M. ed., 2008. The Sage encyclopedia of qualitative research methods. Sage publications.

Gliner, J.A., Morgan, G.A. and Leech, N.L., 2016. Research methods in applied settings. An integrated approach to design and analysis, Routledge.

Goel, D., Abraham, R. and Lahoti, R., 2022. Improving Survey Quality Using Paradata: Lessons from the India Working Survey.

González-Aparicio, M.T., Ogunyadeka, A., Younas, M., Tuya, J. and Casado, R., 2017. Transaction processing in consistency-aware user's applications deployed on NoSQL databases. Human-centric Computing and Information Sciences, 7(1), pp.1-18.

Gordeev, V.S., Akuze, J., Baschieri, A., Thysen, S.M., Dzabeng, F., Haider, M.M., Smuk, M., Wild, M., Lokshin, M.M., Yitayew, T.A. and Abebe, S.M., 2021. Paradata analyses to inform population-based survey capture of pregnancy outcomes: EN-INDEPTH study. Population health metrics, 19, pp.1-14.

Gorla, N., Somers, T. M., and Wong, B., 2010. Organizational impact of system quality, information quality, and service quality. Journal of Strategic Information Systems, 19(3), 207–228.

Grover, V., Chiang, R.H., Liang, T.P. and Zhang, D., 2018. Creating strategic business value from big data analytics: A research framework. Journal of management information systems, 35(2), pp.388-423.

Häkkinen, L. and Hilmola, O.P., 2008. Life after ERP implementation: Long-term development of user perceptions of system success in an after-sales environment. Journal of Enterprise Information Management.

Hallikainen, P., Bekkhus, R. and Pan, S.L., 2018. How OpusCapita Used Internal RPA Capabilities to Offer Services to Clients. MIS Quarterly Executive, 17(1).

Halog, A. and Manik, Y., 2011. Advancing integrated systems modelling framework for life cycle sustainability assessment. Sustainability, 3(2), pp.469-499.

Hannou, F.Z., Amann, B. and Baazizi, M.A., 2019, August. Explaining Query Answer Completeness and Correctness with Partition Patterns. In International Conference on Database and Expert Systems Applications (pp. 47-62). Springer, Cham.

Harahap, V.A., 2019. Recommendations to improve the quality and the automation of the OREDA data process (Master's thesis, NTNU).

Harris, P.A., Taylor, R., Minor, B.L., Elliott, V., Fernandez, M., O'Neal, L., McLeod, L., Delacqua, G., Delacqua, F., Kirby, J. and Duda, S.N., 2019. The REDCap consortium: building an international community of software platform partners. Journal of biomedical informatics, 95, p.103208.

Hartzema, A., Reich, C., Ryan, P., Stang, P., Madigan, D., Welebob, E. and Overhage, J., 2013. Managing Data Quality for a Drug Safety Surveillance System. Drug Safety, 36, pp.49-58. https://doi.org/10.1007/s40264-013-0098-7.

Hasanbasri, A., Kilic, T., Koolwal, G. and Moylan, H., 2023. Using Paradata to Assess Respondent Burden and Interviewer Effects in Household Surveys: Evidence from Low-and Middle-Income Countries.

Hatch, M.J. and Cunlife, A.L., 2006. Organization Theory: Modern, Symbolic, and Postmodern Perspectives. 2nd ed. UK: Oxford University Press, ISBN 9780199260218, ISBN 0199260214, pp.370.

Haug, A., Zachariassen, F. and Liempd, D., 2011. The costs of poor data quality. Journal of Industrial Engineering and Management, 4, pp.168-193. https://doi.org/10.3926/JIEM..V4N2.P168-193.

Heinrich, B., Kaiser, M. and Klier, M., 2007. How to measure data quality? A metric-based approach.

Herbst, K., Juvekar, S., Jasseh, M., Berhane, Y., Chuc, N.T.K., Seeley, J.,

Sankoh, O., Clark, S.J. and Collinson, M.A., 2021. Health and demographic surveillance systems in low-and middle-income countries: history, state of the art and future prospects. Global health action, 14(sup1), p.1974676.

Hitchcock, G. and Hughes, D., 2002. Research and the teacher: A qualitative introduction to school-based research. Routledge.

Holden, M.T. and Lynch, P., 2004. Choosing the appropriate methodology: Understanding research philosophy. The marketing review, 4(4), pp.397-409.

Homan, T., Di Pasquale, A., Kiche, I., Onoka, K., Hiscox, A., Mweresa, C., Mukabana, W.R., Takken, W. and Maire, N., 2015. Innovative tools and OpenHDS for health and demographic surveillance on Rusinga Island, Kenya. BMC research notes, 8(1), pp.1-11.

Horne, K.M., Nichols, J.E., Logsdon, D., Phipps, H., Sanders, S., Wojtyniak, M., McKnight, L.J., Vigneulle, R., Jackson, D. and Elliott, J., 2020. Survey of occupational and environmental exposure monitoring solutions. Military Medicine, 185(Supplement_1), pp. 396-403.

Horr, E.E.T. and Heimlich, J.E., 2018. Effect of platform used for data collection on open-ended response quality. Visitor Studies, 21(1), pp.121-134.

Ilyas, I.F. and Chu, X., 2015. Trends in cleaning relational data: Consistency and deduplication. Foundations and Trends® in Databases, 5(4), pp.281-393.

Jamison, D.C., 2003. Structured query language (SQL) fundamentals. Current protocols in bioinformatics, (1), pp.9-2.

Jämsä-Jounela, S.L., 2007. Future trends in process automation. Annual Reviews in Control, 31(2), pp.211-220.

Jarke, M., Lenzerini, M., Vassiliou, Y. and Vassiliadis, P., 2002. Fundamentals of data warehouses. Springer Science & Business Media.

Juran, J.M., 1954. Universals in management planning and controlling. Management Review, 43(11), pp.748-761.

Kadiri, D., Alao, A. and Onabanjo, B., 2020. Using the Pareto principle to

control building cost in Nigeria. Ethiopian journal of environmental studies & management, 13(3).

Kakish, K. and Kraft, T.A., 2012. ETL evolution for real-time data warehousing. In Proceedings of the Conference on Information Systems Applied Research ISSN, 2167, pp. 1508.

Kalkan, Ö.K., Yusuf, K.A.R.A. and Kelecioğlu, H., 2018. Evaluating performance of missing data imputation methods in IRT analyses. International Journal of Assessment Tools in Education, 5(3), pp.403-416.

Kain, R.C., Bojil, V. and Bloom, K.R., 2020. Community data aggregation platform with data quality control.

Kasi, P., 2009. Research: What, Why and How? A Treatise from Researchers to Researchers. 1st ed. Bloomington: Author House

Kaya, C.T., Türkyılmaz, M. and Birol, B., 2019. Impact of RPA technologies on accounting systems. Muhasebe ve Finansman Dergisi, (82).

Ke. Y., Lixin, D., Miao, Y. and Hu, X., 2018. Data quality evaluation platform and method.

Kelly, J., 2009. Poor data quality costing companies millions of dollars annually. SearchDataManagement. com. Aug.

Kerlinger, F. N., & Lee, H. B. (Eds.)., 2000. Foundations of Behavioral Research (4th Ed.). Orlando, FL: Harcourt College Publishers.

Kerr, K. and Norris, T., 2004, November. The Development of a Healthcare Data Quality Framework and Strategy. In ICIQ, pp. 218-233.

Khanna, R., Elenberg, E., Dimakis, A., Negahban, S. and Ghosh, J., 2017, April. Scalable greedy feature selection via weak submodularity. In Artificial Intelligence and Statistics, PMLR, pp. 1560-1568.

Kiel, D., Müller, J.M., Arnold, C. and Voigt, K.I., 2020. Sustainable industrial value creation: Benefits and challenges of industry 4.0. In Digital disruptive innovation, pp. 231-270.

Knauer, T., Nikiforow, N. and Wagener, S., 2020. Determinants of information system quality and data quality in management accounting. Journal of Management Control, 31(1), pp.97-121.

Kovacs, M., Hoekstra, R. and Aczel, B., 2021. The role of human fallibility in psychological research: a survey of mistakes in data management. Advances in methods and practices in psychological science, 4(4), p.25152459211045930.

KPMG International. Now or Never: 2016 Global CEO Outlook, 2016. https://home.kpmg/content/dam/kpmg/pdf/2016/06/2016-global-ceo-outlook.pdf (retrieved: July 2021).

Krumeich, J., Werth, D. and Loos, P., 2016. Prescriptive control of business processes. Business & Information Systems Engineering, 58(4), pp.261-280.

Kupzyk, K.A. and Cohen, M.Z., 2015. Data validation and other strategies for data entry. Western journal of nursing research, 37(4), pp.546-556.

Kurniawan, N.B., 2018, October. A systematic literature review on survey data collection system. In 2018 International Conference on Information Technology Systems and Innovation (ICITSI), IEEE, pp. 177-181.

Lacity, M. and Willcocks, L., 2016. Robotic Process Automation at Telefónica O2, MIS Quarterly Executive 15(1), pp. 21–35.

Lacity, M., Willcocks, L.P. and Craig, A., 2015. Robotic process automation at Telefonica O2.

Lancaster, G., 2005. Research Methods in Management. A Concise Introduction to Research in Management and Business Consultancy, Jordan Hill.

Le Clair, C., UiPath, A.A. and Prism, B., 2018. The Forrester wave™: robotic process automation, Q2 2018. Forrester Research.

Lee, S.U.J., Sun, J., Dobbie, G. and Groves, L., 2008, March. Verifying Semistructured Data Normalization using PVS. In 13th IEEE International Conference on Engineering of Complex Computer Systems, IEEE, (iceccs 2008), pp. 15-24.

Leno, V., Dumas, M., Maggi, F.M. and La Rosa, M., 2018, June. Multi-perspective process model discovery for robotic process automation. In Proceedings of the Doctoral Consortium Papers Presented at the 30th International Conference on Advanced Information Systems Engineering, CEUR-WS, (CAiSE 2018), (Vol. 2114), pp. 37-45.

Leo, L. and Pipino, L.Y., WL, & Wang, RY , 2002. Data quality assessment. Communications of the ACM, 45(4), pp.211-218.

Leopold, H., van Der Aa, H. and Reijers, H.A., 2018. Identifying candidate tasks for robotic process automation in textual process descriptions. In Enterprise, Business-Process and Information Systems Modeling: 19th International Conference, BPMDS 2018, 23rd International Conference, EMMSAD 2018, Held at CAiSE 2018, Tallinn, Estonia, June 11-12, 2018, Proceedings, Springer International Publishing 19, pp. 67-81.

Lepenioti, K., Bousdekis, A., Apostolou, D. and Mentzas, G., 2020. Prescriptive analytics: Literature review and research challenges. International Journal of Information Management, 50, pp.57-70.

Leshob, A., Bourgouin, A. and Renard, L., 2018, October. Towards a process analysis approach to adopt robotic process automation. In 2018 IEEE 15th international conference on e-business engineering (ICEBE), IEEE, pp. 46-53.

Lin, T., Rivano, H. and Le Mouël, F., 2017. A survey of smart parking solutions. IEEE Transactions on Intelligent Transportation Systems, 18(12), pp.3229-3253.

Liu, Q., Feng, G., Zhao, X. and Wang, W., 2020. Minimizing the data quality problem of information systems: A process-based method. Decision Support Systems, 137, p.113381.

Lokaadinugroho, I., Girsang, A.S. and Burhanudin, B., 2021. Tableau Business Intelligence Using the 9 Steps of Kimball's Data Warehouse & Extract Transform Loading of the Pentaho Data Integration Process Approach in Higher Education. Engineering, MAthematics and Computer Science (EMACS) Journal, 3(1), pp.1-11.

Loola Bokonda, P., Ouazzani-Touhami, K. and Souissi, N., 2020. Mobile Data Collection Using Open Data Kit. In Innovation in Information Systems and Technologies to Support Learning Research: Proceedings of EMENA-ISTL 2019, Springer

International Publishing, 3, pp. 543-550.

Lorenz Simoes, F., Agosti, D. and Guidoti, M., 2021. Delivering Fit-for-Use Data: Quality control.

Loshin, D., 2001. Dimensions of data quality. Enterprise Knowledge Management p. 101124

Luján-Mora, S. and Trujillo Mondéjar, J.C., 2003. A comprehensive method for data warehouse design.

Luján-Mora, S., Trujillo, J. and Song, I.Y., 2002, October. Multidimensional modeling with UML package diagrams. In International Conference on Conceptual Modeling Springer, Berlin, Heidelberg, pp. 199-213.

Mackenzie, N. and Knipe, S., 2006. Research dilemmas: Paradigms, methods and methodology. Issues in educational research, 16(2), pp.193-205.

Madakam, S., Holmukhe, R.M. and Jaiswal, D.K., 2019. The future digital work force: robotic process automation (RPA). JISTEM-Journal of Information Systems and Technology Management, 16.

Mahanti, R., 2019. Data quality and data quality dimensions. Software Quality Professional, 22(1), pp.4-8.

Marev, M.S., Compatangelo, E. and Vasconcelos, W., 2018. Towards a context-dependent numerical data quality evaluation framework. arXiv preprint arXiv:1810.09399.

Marín-Ortega, P.M., Dmitriyev, V., Abilov, M. and Gómez, J.M., 2014. ELTA: New approach in designing business intelligence solutions in era of big data. Procedia technology, 16, pp.667-674.

Maury, E., Boldi, M.O., Greub, G., Chavez, V., Jaton, K. and Opota, O., 2021. An automated Dashboard to improve laboratory COVID-19 diagnostics management. medRxiv.

McCausland, T., 2020. The Bad Data Problem. Research-Technology Management, 64(1), pp.68-71.

Mildner, V., 2019. The SAGE Encyclopedia of Human Communication Sciences and Disorders. Experimental Research, pp.728-732.

Mineva, D., 2020. Where is the focus of surgical activities in breast cancer in bulgaria?. knowledge-international Journal, 40(4), pp.713-717.

Miranda, E., 2015. Management report for marketing in higher education based on data warehouse and data mining. International Journal of Multimedia and Ubiquitous Engineering, 10(4), pp.291-302.

Mircea, M., Ghilic-Micu, B. and Stoica, M., 2011. An agile architecture framework that leverages the strengths of business intelligence, decision management and service orientation. Business Intelligence-Solution for Business Development, pp.1939-1374.

Moffitt, K.C., Rozario, A.M. and Vasarhelyi, M.A., 2018. Robotic process automation for auditing. Journal of emerging technologies in accounting, 15(1), pp.1-10.

Mohajan, H.K., 2020. Quantitative research: A successful investigation in natural and social sciences. Journal of Economic Development, Environment and People, 9(4), pp.50-79.

Mohapatra, S., 2009. Business process automation. PHI Learning Pvt. Ltd..

Moore S., 2018. How to create a business case for data quality improvement. https://www.gartner.com/smarterwithgartner/how-to-create-a-business-case-fordata-quality-improvement (retrieved: July 2021).

Morris, E. and Burkett, K., 2011. Mixed methodologies: A new research paradigm or enhanced quantitative paradigm. Online Journal of Cultural Competence in Nursing and Healthcare, 1(1), pp.27-36.

Morrison, S.M., 2013. Introduction to REDCap for Clinical Data Collection.

Mortenson, M.J., Doherty, N.F. and Robinson, S., 2015. Operational research from Taylorism to Terabytes: A research agenda for the analytics age. European Journal of Operational Research, 241(3), pp.583-595.

Muchanga, M., 2020. Reflexive Debate on Use of Philosophy in Scientific Research, International Journal of Humanities Social Sciences and Education (IJHSSE), 7(6), pp. 208-213.

Muriithi, G.M. and Kotzé, J.E., 2013, October. A conceptual framework for delivering cost effective business intelligence solutions as a service. In Proceedings of the South African Institute for Computer Scientists and Information Technologists Conference, pp. 96-100.

Mussa, M., Souza, S., Freire, E., Cordeiro, R. and Hora, H., 2018. Business intelligence in education: An application of Pentaho software. Revista Produção e Desenvolvimento, 4(3), pp.29-41.

Nikiforova, A., 2020. Open Data Quality Evaluation: A Comparative Analysis of Open Data in Latvia. arXiv preprint arXiv:2007.04697.

Nutt, W., Razniewski, S. and Vegliach, G., 2012, April. Incomplete databases: Missing records and missing values. In International Conference on Database Systems for Advanced Applications, Springer, Berlin, Heidelberg, pp. 298-310.

Osmundsen, K., Iden, J. and Bygstad, B., 2019. Organizing robotic process automation: balancing loose and tight coupling.

Otto, B. and Österle, H., 2015. Corporate data quality: Prerequisite for successful business models. epubli.

Park, Y.S., Konge, L. and Artino, A.R., 2020. The positivism paradigm of research. Academic Medicine, 95(5), pp.690-694.

Papiorek, K.L. and Hiebl, M.R., 2023. Information systems quality in management accounting and management control effectiveness. Journal of Accounting & Organizational Change.

Pavlo, A., Paulson, E., Rasin, A., Abadi, D.J., DeWitt, D.J., Madden, S. and Stonebraker, M., 2009, June. A comparison of approaches to large-scale data analysis. In Proceedings of the 2009 ACM SIGMOD International Conference on Management of data, pp. 165-178.

Penttinen, E., Kasslin, H. and Asatiani, A., 2018, June. How to choose between robotic process automation and back-end system automation? In European Conference on Information Systems 2018.

Pestana, M., Pereira, R. and Moro, S., 2020. Improving health care management in hospitals through a productivity dashboard. Journal of medical systems, 44(4), pp.1-19.

Polit, D.F. and Beck, C.T., 2008. Nursing research: Generating and assessing evidence for nursing practice. Lippincott Williams & Wilkins.

Primer, A., 2015. Introduction To Robotic Process Automation. Institute for Robotic Process Automation, pp.1-35.

Pulla, V.S.V., Varol, C. and Al, M., 2016. Open source data quality tools: Revisited. In Information Technology: New Generations (pp. 893-902). Springer, Cham.

Putra, M. and Putera, M., 2019. Analisis Perbandingan metode SOAP dan REST yang digunakan pada Framework Flask untuk membangun Web Service. SCAN-Jurnal Teknologi Informasi dan Komunikasi, 14(2), pp.1-7.

R Development Core Team., 2005. R: A Language and Environment for Statistical Computing Vienna, Austria: R Foundation for Statistical Computing.

Rahman, A., Smith, D.V. and Timms, G., 2013. A novel machine learning approach toward quality assessment of sensor data. IEEE Sensors Journal, 14(4), pp.1035-1047.

Rahman, M.H. and Sjöström, J., 2021. Respondent Behavior Logging: A Design Science Research Inquiry into Web Survey Paradata. In The Next Wave of Sociotechnical Design: 16th International Conference on Design Science Research in Information Systems and Technology, DESRIST 2021, Kristiansand, Norway, August 4–6, 2021, Proceedings 16 (pp. 248-259). Springer International Publishing.

Ramasamy, A. and Chowdhury, S., 2020. Big Data Quality Dimensions: A Systematic Literature Review. JISTEM-Journal of Information Systems and Technology Management, 17.

Ramdani, A.L. and Firmansyah, H.B., 2018. Clustering Application For UKT Determination Using Pillar K-Means Clustering Algorithm and Flask Web Framework. Indonesian Journal of Artificial Intelligence and Data Mining, 1(2), pp.53-59.

Ratia, M., Myllärniemi, J. and Helander, N., 2018, October. Robotic process automation-creating value by digitalizing work in the private healthcare. In Proceedings of the 22nd International Academic Mindtrek Conference, pp. 222-227.

Redman, T.C., 2001. Data quality: the field guide. Digital press.

Remenyi, D., Williams, B., Money, A. and Swartz, E., 1998. Doing research in business and management: an introduction to process and method. Sage.

Reczek, P., Panczyk, J., Wetula, A. and Młyniec, A., 2023, June. Data Collection Automation in Machine Learning Process Using Robotic Manipulator. In IFIP International Conference on Artificial Intelligence Applications and Innovations (pp. 505-514). Cham: Springer Nature Switzerland.

Ribeiro, J., Lima, R., Eckhardt, T. and Paiva, S., 2021. Robotic process automation and artificial intelligence in industry 4.0–a literature review. Procedia Computer Science, 181, pp.51-58.

Roger, S.P. and Bruce, R.M., 2015. Software engineering: a practitioner's approach. 8th edition, McGraw-Hill Education, pp.247

Ross, J.E., 2017. Total quality management: Text, cases, and readings. Routledge.

Runtuwene, J.P.A., Tangkawarow, I.R., Manoppo, C.T.M. and Salaki, R.J., 2018, February. A comparative analysis of extract, transformation and loading (ETL) process. In IOP Conference Series: Materials Science and Engineering, IOP Publishing, 306 (1), pp. 012066.

Sabooniha, N., Toohey, D. and Lee, K., 2012, August. An evaluation of hospital information systems integration approaches. In Proceedings of the International Conference on Advances in Computing, Communications and Informatics , pp. 498-504.

Sánchez, R.Á., Iraola, A.B., Unanue, G.E. and Carlin, P., 2019. TAQIH, a

tool for tabular data quality assessment and improvement in the context of health data. Computer methods and programs in biomedicine, 181, p.104824.

Saunders, M., Lewis, P. and Thornhill, A., 2003. Research methods for business students. Essex: Prentice Hall: Financial Times.

Saunders, M., Lewis, P. and Thornhill, A., 2009. Research methods for business students. Pearson education.

Scannapieco, M., 2006. Data Quality: Concepts, Methodologies and Techniques. Data-Centric Systems and Applications. Springer.

Schafer, J.L., 1999. Multiple imputation: a primer. Statistical methods in medical research, 8(1), pp.3-15.

Schoenherr, T., Ellram, L.M. and Tate, W.L., 2015. A note on the use of survey research firms to enable empirical data collection. Journal of Business Logistics, 36(3), pp. 288-300.

Schwarz, H., 2018. Data Consistency. Data Processing and Documentation, pp.25.

Sebastian-Coleman, L., 2012. Measuring data quality for ongoing improvement: a data quality assessment framework. Newnes.

Serdar, D. and Syuhaida, I., 2012. Pareto analysis of on-site productivity constraints and improvement techniques in construction industry. Scientific Research and Essays, 7(7), pp.824-833.

Shah, S., Shridhar, P. and Gohil, D., 2010. Control chart: A statistical process control tool in pharmacy. Asian Journal of Pharmaceutics (AJP), 4(3).

Shankaranarayanan, G. and Cai, Y., 2006. Supporting data quality management in decision-making. Decision support systems, 42(1), pp.302-317.

Sharif, B., Lundin, R.M., Morgan, P., Hall, J.E., Dhadda, A., Mann, C., Donoghue, D., Brownlow, E., Hill, F., Carr, G. and Turley, H., 2016. Developing a digital data collection platform to measure the prevalence of sepsis in Wales. Journal of the American Medical Informatics Association, 23(6), pp.1185-1189.

Shewhart, W.A., 1931. Economic control of quality of manufactured product. Macmillan And Co Ltd, London.

Soltanpoor, R. and Sellis, T., 2016, September. Prescriptive analytics for big data. In Australasian database conference, Springer, Cham, pp. 245-256.

Song, J., Hao, J., Gang, C., Suojuan, Z. and Yiping, G., 2019, August. Design and Implementation of a Universal Data Quality Management Software Based on Data Flow. In 2019 10th International Conference on Information Technology in Medicine and Education (ITME) (pp. 645-648). IEEE.

Sommerville, I., Cliff, D., Calinescu, R., Keen, J., Kelly, T., Kwiatkowska, M., McDermid, J. and Paige, R., 2012. Large-scale complex IT systems. Communications of the ACM, 55(7), pp.71-77.

Sorenson, C. and Chalkidou, K., 2012. Reflections on the evolution of health technology assessment in Europe. Health Economics, Policy and Law, 7(1), pp.25-45.

Strong, D.M., Lee, Y.W. and Wang, R.Y., 1997. Data quality in context. Communications of the ACM, 40(5), pp.103-110.

Suman, G. and Prajapati, D., 2018. Control chart applications in healthcare: a literature review. International Journal of Metrology and Quality Engineering, 9, pp.5.

Sumathi, S. and Esakkirajan, S., 2007. Fundamentals of relational database management systems, Springer, 47.

Sunadi, S., Purba, H.H. and Paulina, E., 2021. Overall Equipment Effectiveness to Increase Productivity of Injection Molding Machine: A Case Study in Plastic Manufacturing Industry. ComTech: Computer, Mathematics and Engineering Applications, 12(1), pp.53-64.

Suraya, S. and Sholeh, M., 2022. Designing and Implementing a Database for Thesis Data Management by Using the Python Flask Framework. International Journal of Engineering, Science and Information Technology, 2(1), pp.9-14.

Susanti, E. and Mailoa, E., 2020. RESTful API Implementation in Making a

Master Data Planogram Using the Flask Framework (Case Study: PT Sumber Alfaria Trijaya, Tbk). JITeCS (Journal of Information Technology and Computer Science), 5(3), pp.255-269.

TÂRNĂVEANU, D., 2012. Pentaho business analytics: a business intelligence open source alternative. Database Systems Journal, 3(3), pp.23-34.

Tashakkori, A., Teddlie, C. and Teddlie, C.B., 1998. Mixed methodology: Combining qualitative and quantitative approaches (Vol. 46). sage.

Teorey, T.J., Lightstone S.S., Nadeau T. and Jagadish H.V., 2010. Database Modeling and Design, 4th ed. Morgan Kaufmann, ISBN: 9780080470771

Thysen, S.M., Tawiah, C., Blencowe, H., Manu, G., Akuze, J., Haider, M.M., Alam, N., Yitayew, T.A., Baschieri, A., Biks, G.A. and Dzabeng, F., 2021. Electronic data collection in a multi-site population-based survey: EN-INDEPTH study. Population health metrics, 19(1), pp.1-14.

Todoran, I.G., Lecornu, L., Khenchaf, A. and Caillec, J.M.L., 2015. A methodology to evaluate important dimensions of information quality in systems. Journal of Data and Information Quality (JDIQ), 6(2-3), pp.1-23.

Tripathi, A.M., 2018. Learning Robotic Process Automation: Create Software robots and automate business processes with the leading RPA tool–UiPath. Packt Publishing Ltd.

Tuncer, O. and van den Berg, J., 2010. Implementing BI concepts with Pentaho, an evaluation. Delft University of Technology. Netherlands.

Tussyadiah, I., 2020. A review of research into automation in tourism: Launching the Annals of Tourism Research Curated Collection on Artificial Intelligence and Robotics in Tourism. Annals of Tourism Research, 81, p.102883.

UiPath, 2021. About the UI automation activities pack. [Online]. Available from: https://docs.uipath.com/activities/docs/about-the-ui-automation-activities-pack

Van der Aalst, W.M., Bichler, M. and Heinzl, A., 2018. Robotic process automation. Business & information systems engineering, 60, pp.269-272.

Walter M., 2006. Social Science Methods. An Australian Perspective. Oxford, New York, Oxford University Press, pp. 400.

Wang, R.Y. and Strong, D.M., 1996. Beyond accuracy: What data quality means to data consumers. Journal of management information systems, 12(4), pp.5-33.

Wang, X., Wang, Y. and Xu, D., 2012, June. Lean Six Sigma implementation in equipment maintenance process. In 2012 international conference on quality, reliability, risk, maintenance, and safety engineering, IEEE, pp. 1391-1395.

Wang, Y. and Liu, Z., 2009, December. Study on port business intelligence system combined with business performance management. In 2009 Second International Conference on Future Information Technology and Management Engineering. IEEE, pp. 258-260

Wang, W. and Liu, Z., 2017. Data resource management platform based on Internet data collection.

Wang, Z., Fu, Y., Song, C., Ge, W., Qiao, L. and Zhang, H., 2019. A data quality improvement method based on the greedy algorithm. In Machine Learning and Intelligent Communications: 4th International Conference, MLICOM 2019, Nanjing, China, August 24–25, 2019, Springer International Publishing, Proceedings 4, pp. 256-266.

Ward, M.J., Marsolo, K.A. and Froehle, C.M., 2014. Applications of business analytics in healthcare. Business horizons, 57(5), pp.571-582.

Wiersma, W. and Jurs, S.G., 2008. Research Methods in Education: An Introduction. 9th ed. London, UK, Pearson, ISBN-13: 978-0205581924, ISBN-10: 0205581927, pp. 493.

Williams, D. and Tang, H., 2020. Data quality management for industry 4.0: A survey. Software Quality Professional, 22(2), pp.26-35.

WinAutomation, 2021. Desktop automation https://www.winautomation.com/product/all-features/desktop-automation

World Bank, 2018. Survey solutions CAPI/CAWI platform: release 5.26.

World Health Organization, 2020. World health statistics 2020. Accessed 10 January 2023. https://digitalcommons.fiu.edu/srhreports/health/health/28/

World Health Organization, 2021. World health statistics 2021. Accessed 10 January 2023. https://www.who.int/data/gho/publications/world-health-statistics

Wu, M.S., 1992. The practical need for fourth normal form. ACM SIGCSE Bulletin, 24(1), pp.19-23.

Yang, Q., Ge, M. and Helfert, M., 2017. Guidelines of Data Quality Issues for Data Integration in the Context of the TPC-DI Benchmark. , pp.135-144. https://doi.org/10.5220/0006334301350144.

Ye, Y., Wamukoya, M., Ezeh, A., Emina, J.B. and Sankoh, O., 2012. Health and demographic surveillance systems: a step towards full civil registration and vital statistics system in sub-Sahara Africa?. BMC public health, 12(1), pp.1-11.

Yoshihisa, T., Ishi, Y., Kawakami, T., Matsumoto, S. and Teranishi, Y., 2018. Models for Stream Data Distribution with Progressive Quality Improvement. In Advances on P2P, Parallel, Grid, Cloud and Internet Computing: Proceedings of the 12th International Conference on P2P, Parallel, Grid, Cloud and Internet Computing (3PGCIC-2017), Springer International Publishing, pp. 496-505.

Zeleke, A.A., Naziyok, T., Fritz, F., Christianson, L. and Röhrig, R., 2021. Data quality and cost-effectiveness analyses of electronic and paper-based interviewer-administered public health surveys: systematic review. Journal of medical Internet research, 23(1), p.e21382.

Zhang, M., Wo, T. and Xie, T., 2018, March. A Platform Solution of Data-Quality Improvement for Internet-of-Vehicle Services. In 2018 IEEE International Conference on Pervasive Computing and Communications (PerCom), IEEE, pp. 1-7.

Zhao, D., Ye, W. and Gao, C., 2012, June. Research on process optimization for equipment maintenance based on lean Six Sigma management. In 2012 international conference on quality, reliability, risk, maintenance, and safety engineering, IEEE, pp. 1333-1337.

Zhongming, Z., Linong, L., Xiaona, Y., Wangqiang, Z. and Wei, L., 2019. The CAPI Effect: Boosting Survey Data through Mobile Technology.

Žukauskas, P., Vveinhardt, J. and Andriukaitienė, R., 2018. Philosophy and paradigm of scientific research. Management Culture and Corporate Social Responsibility, 121.

Appendix A

```
        /****** Object:  UserDefinedFunction [dbo].[fn_Exract&PreloadBSUData]
        Script Date: 2021/02/03 10:24:14 PM ******/
USE [Referenced Database]
GO

SET ANSI_NULLS ON
GO

SET QUOTED_IDENTIFIER ON
GO

CREATE FUNCTION [dbo].[fn_Exract&PreloadBSUData]()
Returns Table
AS
Return (

 WITH RPL_CTE (LocationUId, Sequence, FunctionalArea, FieldWorkArea, ParentId1,
 Team)
  AS ( SELECT
     l.LocationUid, REPLACE(LEFT(AreaName,2),'-','') Weekblock,
     LEFT (a.AreaDescription,5) FCA,
    a.AreaName,
     ROW_NUMBER() OVER(ORDER BY l.LocationUid, lt.LocationTypeId) ParentId1,
     RIGHT(LEFT(a.AreaName,4),1) Team

    FROM ops.QuestionnaireAllocations qa
      JOIN ops.Questionnaires q ON q.QuestionnaireUid = qa.QuestionnaireUid
      JOIN ops.DataCollectionRounds dr ON dr.RoundUid = qa.RoundUid
      JOIN LocationsInAreas LA on LA.LocationUid = QA.LocationUid
      JOIN Locations l ON l.LocationUid = la.LocationUid
      JOIN ops.LocationStates ls ON ls.LocationUid = l.LocationUid
      JOIN Events lse On lse.EventUid =  ls.EndEventUid
      JOIN codes.LocationStatusCodes lsc On lsc.Code = ls.[Status]
      JOIN ops.LocationTypeEpisodes lte ON l.LocationUid = lte.LocationUid
      JOIN Events Se ON se.EventUid = lte.StartEventUid
      JOIN Events ltee On ltee.EventUid =  lte.EndEventUid
```

```
      JOIN codes.LocationTypes lt ON lt.LocationTypeId = lte.LocationTypeId
      JOIN ops.Areas A ON A.AreaUid = La.AreaUid
    WHERE qa.Status = 0 AND AreaTypeId = 5  AND lse.EventTypeId IN (18,19)
    AND ltee.EventTypeId IN (18,19) AND  q.Acronym = 'BSU_2021' AND FCA = 1
    AND lt.LocationTypeId = 1 AND REPLACE(LEFT(AreaName,2),'-','') IN ('14','15')
  ),

HHR_CTE (LocationUId, hhu_list, HouseholdUId, StartEvent, EndEvent,ParentId1,
                                              ParentId2, Sequence,FieldWorkArea)
AS ( SELECT
       rpl.LocationUId,
       hr.HouseholdExtId hhu_list,
       hr.HouseholdUid HouseholdUId,
       hr.StartEventUid,
       hr.EndEventUid,
       rpl.ParentId1,
       Row_Number() OVER (PARTITION BY rpl.LocationUId ORDER BY
       hr.HouseholdExtId) ParentId2,
       rpl.Sequence,
       rpl.FieldWorkArea
    FROM RPL_CTE rpl
     LEFT JOIN HouseholdResidences hr ON hr.LocationUid = rpl.LocationUId
     LEFT JOIN Events ee ON ee.EventUid = hr.EndEventUid
     LEFT JOIN codes.EventTypes et ON et.EventTypeId =ee.EventTypeId
    WHERE et.EventTypeName IN ('OBS', 'OBL')

  ),

 CTE_Union
  As ( SELECT
         Individual1Uid,
         PS.PartnershipStatusName
       FROM Unions u
           JOIN Individuals I1 ON I1.IndividualUid = u.Individual1Uid
           JOIN IndividualObservations Io ON IO.IndividualUid = I1.IndividualUid
           JOIN Events EE ON EE.EventUid = U.EndEventUid AND
           EE.EventUid = IO.ObservationUid
           JOIN codes.PartnershipStatuses ps ON ps.PartnershipStatusId =
           io.PartnershipStatus
```

```
      WHERE EE.EventTypeId IN (18,19)
  ),

HHMem_CTE
AS  ( SELECT
      DISTINCT   hhr.LocationUId,
      hm.HouseholdUid,
      I.IndividualUid,
      P.ExternalIdentifier,
      p.Surname,
      p.FirstName1,
      p.FirstName2,
      p.AltSurname,
      p.CivilianID,
      de.EventDate,
      DATEDIFF(YY,de.EventDate,GETDATE()) Age,
      sv.SexName Sex,
      P.TelephoneNr1 TelNo1,
      P.TelephoneNr2 TelNo2,
      CONVERT(VARCHAR(8), se.EventDate, 112) MemStartDate,
      CASE WHEN iree.EventTypeId=18 THEN  'Yes - Resident'
       ELSE 'No - Not Resident' END AS  IsResident,
      CONVERT(VARCHAR(10),HRT.HHRelationshipTypeId) + ' - ' +
      HRT.HHRelationshipTypeName AS HeadRelationship,
      CASE WHEN mth.Surname IS NOT NULL THEN mth.Surname + ', ' +
      ISNULL(mth.Firstname1, '') + ' ' + ISNULL(mth.Firstname2, '')
      ELSE ISNULL(p.MotherSurname, '') + ', ' +
      ISNULL(p.MotherFirstName, '') END AS MotherName,
      mth.ExternalIdentifier AS MotherDSID,
      CASE WHEN mthe.EventTypeId = 7 THEN 'Dead'
         WHEN mthe.EventTypeId = 18 THEN 'Alive'
      ELSE 'Don''t Know' END AS MotherLastStatus,
      NULL MotherLastAge,
      CASE WHEN fth.Surname IS NOT NULL THEN fth.Surname + ', ' +
      ISNULL(fth.Firstname1, '') + ' ' + ISNULL(fth.Firstname2, '')
      ELSE ISNULL(p.FatherSurname, '') + ', ' +
      ISNULL(p.FatherFirstname, '') END AS FatherName,
      fth.ExternalIdentifier FatherDSID,
      CASE WHEN fthe.EventTypeId = 7 THEN 'Dead'
```

```
            WHEN fthe.EventTypeId = 18 THEN 'Alive'
        ELSE 'Don''t Know' END AS FatherLastStatus,
        NUll FatherLastAge,
        p.Surname + ', ' + p.FirstName1 + ' ' + ISNULL(p.FirstName2, '') +
        ' (' + p.ExternalIdentifier + ')' + ' - ' + sv.SexName
        + ' - ' + CONVERT(VARCHAR(3), DATEDIFF(YY,de.EventDate,GETDATE()))
        AS MemList,
        CONVERT(VARCHAR(2),mso.MaritalStatus) + ' - ' +
        ISNULL(ms.MaritalStatusName, '') MaritalStatus,
        CONVERT(VARCHAR(2),mso.PartnershipStatus) + ' - ' +
        ISNULL(pp.PartnershipStatusName, '')  PartnershipStatus,
        mso.PregnancyStatus,
        NULL LastPregnancyEndDate,
        NULL ConjugalRelationship,
        hm.StartEventUid HMStartEvent,
        hm.EndEventUid HMEndEvent,
        hhr.StartEvent HRStartEvent,
        hhr.EndEvent HREndEvent,
        hhr.ParentId1,
        hhr.ParentId2,
        hhr.Sequence,
        hhr.FieldWorkArea
FROM HouseholdMemberships hm
  JOIN HHR_CTE hhr ON hhr.HouseholdUId = hm.HouseholdUid
  JOIN Events ee ON ee.EventUid = hm.EndEventUid
  JOIN Events se ON se.EventUid = hm.StartEventUid
  JOIN ProtectedIndividuals p ON p.IndividualUid = hm.IndividualUid
  JOIN Individuals I ON I.IndividualUid = p.IndividualUid
  JOIN Events ie ON ie.EventUid = i.EndEventUid
  JOIN codes.SexValues sv ON sv.Sex = I.Sex
  JOIN Events de ON de.EventUid = I.BirthEventUid
  JOIN Observations O ON O.EventUid = EE.ObservationEventUid
    JOIN IndividualObservations MSO ON MSO.ObservationUid = O.EventUid AND
    P.IndividualUid = MSO.IndividualUid
    JOIN HHeadRelationships HHH ON  HH.HouseholdMembershipUid=
    HM.HouseholdMembershipUid
    AND EE.ObservationEventUid = HHH.EndEventUid
    JOIN codes.HHRelationshipTypes HRT ON HH.HHRelationshipTypeId=
    HRT.HHRelationshipTypeId
```

```
JOIN IndividualResidences ir ON ir.IndividualUid = i.IndividualUid
    AND ir.LocationUid = hr.LocationUId
JOIN Events iree ON iree.EventUid = ir.EndEventUid
LEFT JOIN ProtectedIndividuals mth ON mth.IndividualUid = I.MotherUid
LEFT JOIN Individuals mthri ON mthri.IndividualUid = I.MotherUid
LEFT JOIN Events mthe ON mthe.EventUid = mthri.EndEventUid
LEFT JOIN ProtectedIndividuals fth ON fth.IndividualUid = I.FatherUid
LEFT JOIN Individuals fthri ON fthri.IndividualUid = I.FatherUId
LEFT JOIN Events fthe ON fthe.EventUid = fthri.EndEventUid
LEFT JOIN codes.MaritalStatuses ms ON ms.MaritalStatusId = mso.MaritalStatus
LEFT JOIN codes.PartnershipStatuses pp ON pP.PartnershipStatusId =
 mso.PartnershipStatus
WHERE ie.EventTypeId IN (18, 19) AND ee.EventTypeId IN (18,19)
  ),

  CHL_CTE (RegisteredIndividual, Household, HasCHL)
    AS ( SELECT
            hm.IndividualUid,
            hm.HouseholdUId,
           CASE
         WHEN (hm.Age >= 0 AND hm.Age <= 6) THEN 1
         ELSE 0 END HasCHL
    FROM HHMem_CTE hm
          LEFT JOIN IndividualResidences ir ON ir.IndividualUid = hm.IndividualUid
          AND ir.LocationUid = hm.LocationUId
          LEFT JOIN Events iree ON iree.EventUid = ir.EndEventUid
          LEFT JOIN codes.EventTypes et ON et.EventTypeId=iree.EventTypeId
       WHERE et.EventTypeName IN ('OBS', 'OBL')
  )
SELECT
Row_Number() OVER (PARTITION BY LocationUid, HouseholdUId
ORDER BY Age DESC, mem_reg_intid) members_roster__id,*
  FROM (
     SELECT
        DISTINCT  hm.MemList AS hhu_members_list,
        hm.IndividualUid AS mem_reg_intid,
        chl.HasCHL hm_has_chl,
        IsPregnacyHistory hm_has_preg_history,
        hm.DSID hm_dsid,
```

```
NULL hmu_is_new_member,
NULL hmu_member_dsid,
hm.Surname hm_surname,
NULL hm_surname_changed,
NULL hm_current_surname,
hm.FirstName1  hm_FirstName1,
NULL hm_hasFirstname1_changed,
NULL hm_current_FirstName1,
hm.FirstName2 hm_FirstName2,
NULL hm_hasFirstname2_changed,
NULL hm_current_FirstName2,
NULL hm_maiden_name_changed,
NULL hm_current_maiden_name,
NULL hm_nationalty,
NULL hm_citizenship,
NULL hm_civil_id_detail,
ISNULL(hm.CivilianID,'999999999999999')  hm_civil_id,
SUBSTRING(hm.CivilianID, 1, 4) + '-' +
SUBSTRING(hm.CivilianID, 5, 2) + '-' +
SUBSTRING(hm.CivilianID, 7, 2) hm_civil_id,
NULL hm_current_civil_id,
CONVERT(date,hm.OffDoB) hm_civil_id_dob,
NULL hm_civil_id_changed,
NULL hm_current_civil_id_dob,
hm.Sex hm_gender,
NULL hm_gender_changed,
NULL hm_current_gender,
NULL hm_place_of_birth,
NULL hm_tel_primary,
NULL hm_tel_provider_primary,
NULL hm_tel_primary_oth_specify,
NULL hm_tel_secondary,
NULL hm_tel_provider_secondary,
NULL hm_tel_secondary_oth_specify,
SUBSTRING(hm.MemStartDate, 1, 4) + '-' +
SUBSTRING(hm.MemStartDate, 5, 2) + '-' +
SUBSTRING(hm.MemStartDate, 7, 2) hm_start_date,
LL hm_married_partner,
NULL hm_married_date,
```

```
NULL hm_partner_other_wives,
NULL hm_in_employment,
NULL hm_self_employed,
NULL hm_time_spent__1,
NULL hm_time_spent__2,
NULL hm_time_spent__3,
NULL hm_time_spent__4,
NULL hm_time_spent__5,
NULL hm_time_spent__6,
NULL hm_time_spent__7,
NULL hm_time_spent__8,
NULL hm_time_spent__9,
NULL hm_time_spent__10,
NULL hm_time_spent__11,
NULL hm_time_spent__12,
NULL hm_current_employer,
NULL hm_current_employment_sector,
NULL hm_responsible_care_person,
NULL hm_responsible_relationship,
NULL hm_responsible_rel_other,
NULL hm_chl_rth_card,
NULL hm_chl_rth_dob,
NULL hm_chl_rth_weight,
NULL hm_chl_had_vaccination,
NULL hm_chl_first_vac_date,
ParentId2 household_roster__id,
hm.ParentId1 AS interview__id,
NULL interview__key,
hm.LocationUid,
HouseholdUId,
Age,
CASE
    WHEN Team=1 THEN 'TeamA_QC1'
    WHEN Team=2 THEN 'TeamB_QC2'
    WHEN Team=3 THEN 'TeamC_QC3'
END _responsible
FROM HHMem_CTE hm
LEFT JOIN CHL_CTE chl ON chl.Household = hm.HouseholdUId AND
chl.RegisteredIndividual = hm.IndividualUid
```

```
        JOIN LOCATIONS L ON L.LocationUId = hm.LocationUId
        JOIN RPL_CTE rpl on rpl.LocationUId = L.LocationUId
        LEFT JOIN (
         SELECT DISTINCT I.IndividualUid, 'N' IsPregnacyHistory
         FROM Individuals I
         JOIN Events Be ON Be.EventUid =I.BirthEventUid
         JOIN Events DE ON DE.EventUid = I.EndEventUid AND
         DE.EventUid=DE.ObservationEventUid
         LEFT JOIN Pregnancies p on p.WomanUid = I.IndividualUid
         LEFT JOIN BirthHistories B ON B.PregnancyUid = P.PregnancyUid
        WHERE I.Sex = 2 AND B.BirthHistoryUid IS NULL AND
        DATEDIFF(YY,Be.EventDate,GETDATE()) BETWEEN 15 AND 55
        AND DE.EventTypeId IN (18,19)
            )NopregHistory ON NopregHistory.IndividualUid = hm.IndividualUid
         LEFT JOIN CTE_Union u ON u.Individual1Uid = hm.IndividualUid
)Tmp
)GO
```