2017

A Cryptography-based system for offline collection, verification and accounting of revenue in County governments

Carlton H. Wanga
Faculty of Information Technology (FIT)
Strathmore University

A Cryptography-Based System for Offline Collection, Verification and Accounting of Revenue in County Governments

Carlton H Wanga

Master of Science in Mobile Telecommunications and Innovation.

2017

Declaration

I declare that this work has not been previously submitted and approved for the award of a degree by this or any other University. To the best of my knowledge and belief, the thesis contains no material previously published or written by another person except where due reference is made in the thesis itself.

Carlton H Wanga

…….....

June 2017

Approval

The thesis of Carlton Wanga was reviewed and approved by the following:

Dr. Bernard Shibwabo,
Senior Lecturer, Faculty of Information Technology,
Strathmore University

Dr. Joseph Orero,
Dean, Faculty of Information Technology,
Strathmore University

Professor Ruth Kiraka,
Dean, School of Graduate Studies,
Strathmore University

Abstract

An integrated revenue collection system is primarily a cash and receipt management system that incorporates a highly-distributed payment processing system. It enables county governments to consolidate revenue across their operating branches at the end of the day. In the current context of county governments in Kenya, efficient tax collection is highly dependent on validation of payment documents. This has significantly been challenged by the fact that revenue collection has traditionally employed paper-based collection receipts. The research targets to address the challenges of validation of payment receipts in offline revenue collection systems. It supports automation attempts that have been made through the introduction of electronic mobile point of sale terminals. The solution is based on providing an offline model that supports the distributed nature of payment stations. This approach focuses on using cryptography-based techniques to enable offline validation of receipts even in cases of unreliable network connectivity. The aim is to provide a solution that affords ease of both revenue collections for the county governments and payments for the Citizenry while stopping revenue leakages, ensuring reliable verification of payment receipts and maximising of revenue collection by providing reliable accounting reports. The research applies the use of the Waterfall methodology. Waterfall methodology was preferred because it provides better management of timelines and project deliverables. The research lead to the development of a reliable revenue collection system that enables offline receipting and verification of payment receipts in integrated mobile point of sale terminals.

Keywords: Revenue collection, Receipt verification, Offline payment processing, Electronic mobile point of sale terminals

Table of Contents

List of Figures

List of Tables

List of Abbreviations/ Acronyms

LAIFOMS - Local Authorities Integrated Financial Operations Management System

HTTP - Hypertext Transfer Protocol

HTTPS - Hypertext Transfer Protocol Secure

POS - Point of Sale

CRBR - Counter Receipt Book Register

USSD - Unstructured Supplementary Service Data

ETR - Electronic Tax Register

QR - Quick Response

SQL - Structured Query Language

API - Application Programming Interface

SDK - Software Development Kit

JSON - JavaScript Object Notation

REST - Representational State Transfer

URL - Universal Resource Locator

MVC - Model View Controller

CSV - Comma Separated Values

J2ME - Java 2 Platform, Micro Edition

Chapter 1 : Introduction

1.1 Background

The need for effective tax collection systems for collection of parking fees, permit payments and market fees and the underlying processes of verification of payments have become more pronounced in the current context of devolved governments in Kenya (Khaunya, Wawire, & Chepng'eno, 2015). Officers at the County Level now have greater autonomy in providing robust tax collection processes that reduce revenue leakage. It is in this context that there is need to ensure adequate systems are put in place for effective collection, verification and accounting of revenue (Society for International Development, 2012; Osoro, Atambo, & Abuga, 2015).

In addition, as a new system of governance, the counties in Kenya have to create the necessary revenue collection and accounting structures to deliver on their constitutional mandate (Khaunya, Wawire, & Chepng'eno, 2015; Buluma & M.Obande, 2015). County governments, for this reason, need a revenue collection and automated revenue management systems which will provide sound revenue collection processes for better financial management. A reliable system will cement sustainability of the counties on the basis of their own income (Mwenda, 2010).

Collection of revenue in county governments involves setting up collection stations that provide a means of receiving payments for defined revenue streams (Nduda, 2015). Upon receipt of service fees, collectors need to offer tickets or permits showing proof of payment. The collection receipts not only serve as proof of payment but also provide a way of verification before granting access to specific services (Nduda, 2015). Collection of parking fees is one of the most prominent modes of revenue collection. This process involves customers reserving parking slots by paying a fee. Parking attendants need to validate actual receipt of payment by verifying the payment ticket before allowing the vehicle to park in a specific area.

Devolution of revenue collection and management to the counties and subsequently to the sub-counties comes with enormous challenges especially lack of network and communication infrastructure needed to facilitate Internet connectivity. There is therefore a need to employ alternative technologies that would enable offline collection and verification of revenue (Khaunya, Wawire, & Chepng'eno, 2015).

Cryptography-based techniques have provided reliable methods for validation of electronic data (Alwyn & Albertus, 2013; Christin & Safavi-Naini, 2014). The authentication capabilities made possible by public key cryptography using standard algorithms make it possible to develop robust data validation applications. This is made conceivable by their ability to generate unique identifiers whose sequence of generation cannot be easily predicted. The techniques also make it possible to encrypt information blocks that can be decrypted by intended parties to enable viewing and validation of the actual data.

1.2 Problem Statement

Collection of revenue by county governments in Kenya has been plagued with numerous cases of fraudulent and counterfeit receipts that are used to avoid payment of fees (Ataro, Muturi, & Wandera, 2016). In the current setting, verification teams have no way of differentiating between legal and illegal receipts. This is because receipt validation would typically require a centralised database receipting system that can be queried to verify receipt of payment (Alwyn & Albertus, 2013; Centeno, 2008). This is extremely difficult to achieve due to lack of reliable network connectivity across all stations of revenue collection.

The consequence of poor revenue collection and payment validation structures is that county governments are not able to meet their revenue collection targets. The problem is further fuelled by the absence of a reliable tracking process or audit trails. It is very difficult to present a solid case against defaulters who do not pay service fees (Khaunya, Wawire, & Chepng'eno, 2015).

Individuals and small scale enterprise engaged in economic activities also avoid payment of taxes to county governments due to the lack of a proper debt recovery mechanisms (Karori & Abuga, 2016). The end result is that revenue collection administration is often inefficient and large amounts of funds are left uncollected while that collected is sometimes inappropriately managed.

1.3 Aim

The aim of this research is to develop an electronic receipting system that will support offline collection of revenue by county governments. The system targets to provide a reliable model that will work even in cases of extreme network connectivity challenges. The offline receipting model needs to provide a way of accurately validating payment receipts. This will

enable county governments to improve their revenue collection targets by reducing the sources of revenue leakage.

1.4 Specific Objectives

(i) To investigate the causes of prevalent use of counterfeit receipts in revenue management systems.

(ii) To examine challenges that face implementation of offline electronic revenue collection systems in county governments.

(iii) To analyse the methods, models, techniques and technologies that can be used to verify the validity of payment receipts generated offline.

(iv) To develop an integrated system to support receipting, receipt verification, accounting and reporting of revenue collection.

(v) To test and validate the solution.

1.5 Research Questions

(i) What are the causes of prevalent use of counterfeit receipts in revenue management systems?

(ii) What challenges face implementation of offline electronic revenue collection systems in county governments?

(iii) Which methods, models, techniques and technologies can be used to verify the validity of payment receipts that have been generated offline?

(iv) How can an integrated system that supports receipting, receipt verification, accounting and reporting of revenue collection be developed?

(v) How does the system provide a robust solution for revenue collection and receipt validation?

1.6 Justification

County governments serve as the pillar of development and economic growth in Kenya today, and it is essential to ensure that they have the capacity to raise enough revenue to support development projects (Society for International Development, 2012; Buluma & M.Obande, 2015). Cases of loss of revenue have been rampant since the inception of devolution. This is mainly due to poor revenue collection and receipt validation techniques (Nduda, 2015).

Lack of proper payment verification mechanism is one of the major challenges facing revenue collection according to the Collection of Revenue Allocation Report (2015). The commission of revenue allocation further issued a directive for counties to stop borrowing money to service their operations until they are able to prove their ability to pay debts based on their performance on local revenue collection.

Counties are however still suffering from unrecoverable debts due to poor record keeping as well as improper accounting procedures as a result of the use of paper-based receipt books (Ataro, Muturi, & Wandera, 2016). It is estimated by the Commission of Revenue Allocation in their 2015 annual report that county revenues are reduced by half due to Revenue leakage as a result of inadequate procedures in the reconciliation of revenues collected and an over-reliance on erroneous paper documentation provided by revenue collection officers. To achieve revenue growth, it is important to ensure that county governments adopt best practices and conform to the legal requirements as enshrined in the Public Finance Management Act 2012 and Chapter 12 of the Constitution of Kenya.

1.7 Scope and Limitation

Various county governments have adopted different finance bills that that define distinct underlying business processes to collect revenue. The research does not provide a solution that would capture the detailed process flow of each of the forty-seven counties in Kenya. However, it focuses on ensuring that all the high-level processes such as revenue collection, billing, receipting, verification and accounting are well designed to handle transparent collection and verification of Revenue.

The revenue streams supported by the application are limited to unstructured sources of revenue in the County Government of Bungoma. This is because it serves as a mid-level county that is partly urban and partly rural. It therefore offers sub counties with both reliable and unreliable Internet connections. The county has also been faced by allegations over misappropriation of funds in the recent past and will therefore provide a good case study for challenges facing revenue management.

The application is also limited to running on hardware technologies that are able perform tasks such as Mobile Point of Sale Terminal collection or thermal Bluetooth printing. This is because various product vendors offer different Software Development Kits to achieve a particular

task. The research, therefore, determined the most cost friendly and efficient devices that can be adopted as standard revenue collection devices.

Comprehensive revenue accounting would also require integration with bank to enable automatic reconciliation. However, counties are served by different banking providers. It might not be possible to integrate with all banks. The solution, therefore, did not focus on providing a module for bank reconciliations.

Chapter 2 : Literature Review

2.1 Introduction

County governments in Kenya have been tasked with the objective of ensuring that they provide well defined systems that will guarantee transparent collection of revenue (Commission of Revenue Allocation, 2015). This directive by the Commission of Revenue Allocation comes with multiple implementation challenges because of the varying technological capacities in different counties. This section analyses various components to revenue automation and linked findings of past studies to provide knowledge that might serve as possible solutions to current issues facing automation of revenue collection and verification of receipts in county governments as well as open avenues for possible studies in the future.

2.2 Automation of Revenue by County Governments

The main objective of revenue automation by county governments is to ensure that service delivery to revenue payers is convenient and that it supports modern payment solutions. Convenient service delivery is that which avoids the need of tax payers to have to go to authorities in order to access services of tax payment (Basu, 2010).

2.2.1 Causes of Revenue Leakage in County Governments

ICT serves as the best possible solution to solving inefficient service delivery by Local authorities in Kenya (Ochieng, Gichoya, & Odini, 2011; Adera & Waema, 2011). According to the Commission of Revenue Allocation Report (2015), most counties in Kenya are still using paper-based Receipt books to collect revenue.

Revenue books are controlled by the store's manager who is responsible for placing orders for new books as well issuing books to the Revenue Department (Muthama, 2013). Reconciliation is done by the revenue accountant who is responsible for writing off used books that are returned to the County by the Revenue Officers. The revenue books are entered into LAIFOMS and the Counter Receipt Book Register (CRBR) on receipt from where they are issued to the Sub County Revenue officers (ROs). The Revenue officers also maintain the counter receipt book registers from which they issue receipt books to the revenue collection clerks (Muthama, 2013).

Revenue officers track the issued receipt books through control sheets (Mutysia, 2014). It is noteworthy that the revenue books have no security features and are prone to forgery hence loss

of revenue by counties (Siehl, 2010). The control of revenue books through the counter receipt book register is cumbersome and prone to errors (Mutysia, 2014; Karori & Abuga, 2016).

Figure 2.1, provides an overview of the current revenue collection and receipting process in most county governments. It tracks the process of management of payment receipt books and the accounting process that follows.

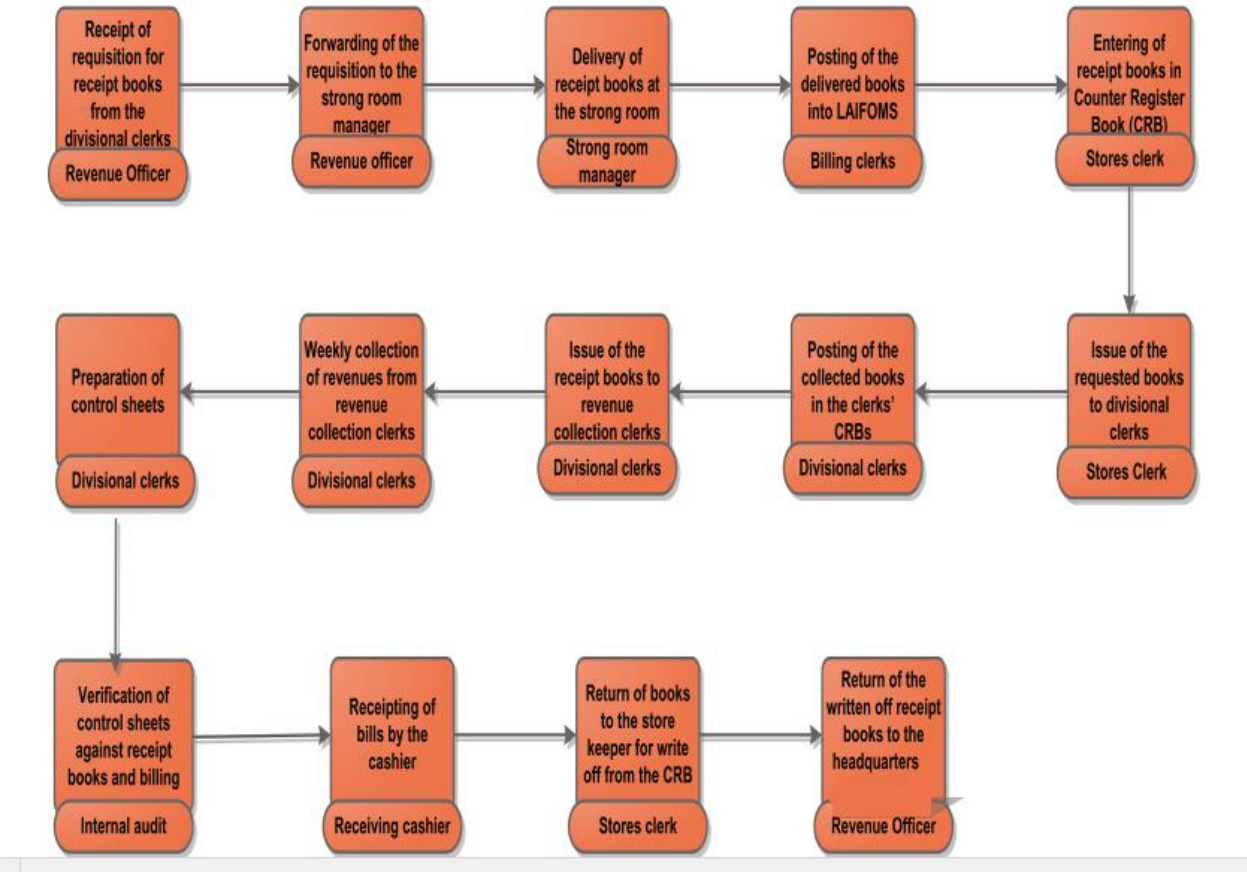


Figure 2.1: Overview of the Paper-Based Revenue Collection Process

2.3 Challenges Facing Automation of Revenue Collection

There is an urgent need for county governments to automate collection of revenue in order to seal the gaps that cause leakage of revenue (Commision of Revenue Allocation, 2015). However, counties are still facing massive connectivity challenges that hamper the automation process. It is imperative that alternate automated solutions that can robustly handle connectivity challenges are developed.

This problem of network connectivity could easily be solved by developing offline mobile application solutions. However, an integrated revenue management system would require a mechanism to perform verification checks such as validation of parking receipts. It would otherwise be very difficult to carry out such checks without some form of network communication (Mburugu, 2016).

The infrastructure that is required to handle the demands of the traditional paper-based way of revenue collection is also very expensive to set up and cumbersome to maintain (Ngotho & Kerongo, 2016; Mutysia, 2014). Despite the technological potential that can be released by leveraging on the power of the Internet to improve revenue growth in county governments, most of the areas are remote, and the network coverage can be very unreliable (Nduda, 2015).

2.3.1 Internet Connectivity Challenges

African countries are still being challenged by lack of reliable Internet connection required to sustain governance and commercial activities. This possess a great challenge as governments need to harness the power of information technology to provide better and convenient services (Adera & Waema, 2011; Ataro, Muturi, & Wandera, 2016).

Lack of reliable Internet connection is the root cause of lack of plausible advancements in development (Souter & Kerretts, 2012). Internet connectivity challenges have to be overcome in order to achieve greater public access to information that would make a distinctive, positive contribution to the democratic and socio-economic future of the continent (Diallo & Calland, 2013).

Evidently, inadequate Internet connectivity is a great challenge facing implementation of information systems in Kenya. There is therefore a great need to come up with alternate solutions that can overcome impediments caused by poor connection.

2.3.2 Need to Implementing Offline Capabilities in Financial Systems

The greatest impediment to implementing systems which involve validation of payments is network communication (Lee, 2009). While current technological advances can comfortably provide reliable offline systems with data synchronisation capabilities, it becomes very difficult to confirm payments by verifying receipts when not all the data has been synchronised (Birgit, 2002). In county governments' revenue collection, it is important that an enforcement team is placed at strategic points to verify the validity of receipts. In cases of revenue collection on streams such as

parking fees, there is a need for validation of vehicle number plates against a database of payments through verification checks to maximise on collection (Mutysia, 2014).

2.4 Impacts of Revenue Automation

Automation of revenue management mainly involves transition of revenue collection from use of paper-based receipts to some form of electronic system that facilitates data entry, processing and reporting of revenue (Mutysia, 2014). The possible solutions vary from USSD wallets to self-service portals. The important aspect of automation is ensuring that convenient and effective service delivery is availed to customers while at the same time provide an efficient monitoring and control structure that will avoid revenue leakage (Muthama, 2013).

Automation not only makes it possible for accurate data capturing but also provides constraints to ensure that tax payers are charged the correct amount on the finance bills. The power of network connectivity and computational storage makes it possible to integrate various applications to provide accurate reports (Mutysia, 2014).

In Kenya, automation is a result of tax administration reforms to better improve the reporting of revenue (Muthama, 2013). It has been common for counties to give understatements of revenue they collected or sometimes fail to account for the collection performance (Commision of Revenue Allocation, 2015). Keeping track of all collection streams has been faulty. With inaccurate data, it has not been possible for counties to do predictive analysis or effective planning (Mutysia, 2014). This is therefore a key challenge that affects Kenya as a country at large and needs improvements so as to foster development.

2.5 Models, Techniques and Technologies for Improvement of Revenue Collection

Receipting is an unavoidable component of every revenue collection system. The need for printing is inevitable in most payment solutions. Electronic Tax Register (ETR) receipts have served an important role in servicing payments in most developing countries (Lumumba, 2010). Efficient collection of revenue requires mobile capabilities that can enable provision of services in any location even those where setting up of physical computers can be a challenge (Omwansa, 2014). Electronic receipting offers many advantages over paper-based receipting by providing a mechanism for efficient record keeping as well as well as effective reporting (Lumumba, 2010). This provides the advantage of non-repudiation where the service clerk and the customer cannot deny processing of a particular transaction (Lumumba, 2010).

Effective mobile payments should ensure that the synchronised figures are correct and that it can be guaranteed that the transaction details have not been modified. Important narration details should be captured as well as corresponding audit trails to ensure that both parties can be identified so as to guarantee transparency. Mobile transactions should also need restricted access and security mechanisms to ensure that the receipts are accessible only to those who are authorised to view the corresponding transactions (ISACA, 2011).

The primary objective of a point of sale terminal services is not only to guarantee reliable electronic transactions of billing and receipting payment, but also to ensure that communication between cashiers, managers administrators and other stakeholders is easier and more reliable as compared to the physical receipting. Due to the possibilities of user authentication and authorisation brought about by the computational abilities of mobile Point of Sale terminals, it can be guaranteed that transactions have been performed by users who have actually been delegated to perform those operations (ISACA, 2011).

2.5.1 Mobile Point of Sale Terminals

Ecommerce and retailing operations have been revolutionised by the advent of mobile point of sale terminals (Godbole, 2006). Mobile point of sale devices have the ability to support development of custom payment solutions since they are embedded with smart operating systems and powerful software development kits (Sharma, 2016). According to Sharma, these devices provide a great potential for additional access services. This is because, operating systems such as Android provide additional sensory capabilities such as geolocation which can serve as an important tool to obtaining accurate transaction location for control mechanisms as well as provide reliable geolocation reports (Tanai, 2014).

In additional to geolocation capabilities, mobile point of sale terminals are built with revolutionary device capabilities that are not available in conventional computers. Use of payment cards has become a convention in the processing of mobile payments (Jih, 2009). Most mobile point of sale terminals come prebuilt with readers that can read both magnetic and smart cards making it possible to process payments through swiping of cards or tap as you go services (Yamamoto, 2012). Since mobile terminals are built with smart phone technologies, they offer tools such as RFID, and QR code scanners that can be used to perform verification checks.

If well designed applications running on mobile point of sale terminals can be well designed to handle offline data storage and supported by their abilities of cellular connections as

well as Wi-Fi to handle data synchronisation with offline servers (István, 2008). Modern terminals also have out of the box data storage capabilities that makes it easy to perform caching of data dictionaries as well as application preferences. Apart from the portability that is brought about by using mobile terminals, they also come with inbuilt thermal printers that can meet all receipt issuance needs.

These devices have already been employed in other avenues such as shipping deliveries. If well utilised these devices can serve as effective point of sale devices for revenue collection. Figure 2.2 shows a mobile point of sale terminal with an in-built thermal printer. The device is powered by Android and provides the ability to print receipts as well as read bar codes.

Figure 2.2: Point of Sale Terminal with a Thermal Printer (Adapted from Deloitte, 2017)

2.5.2 Receipting with Bluetooth Thermal Printers

Bluetooth thermal printers are a new technology that have made it possible to process payments through the use of conventional mobile phones (István, 2008). They serve as an add-on that provides the possibility to print directly from mobile phones thereby leveraging on the computational abilities of smart phones (Leonardo, Beniamino, & Marcello, 2014). Figure 2.3 shows a Bluetooth thermal printer. These devices are light weight, have a long battery life and also offer high printing speeds of around 60 mm/s (480 dots/sec).

Figure 2.3: Bluetooth Thermal Printer (Adapted from Datecs, 2017)

2.6 Verification and Validation of Receipts

Automation of Revenue has proven to be a great success in improving revenue collection in a number of counties (Mutysia, 2014; Khaunya, Wawire, & Chepng'eno, 2015). However, there has still been an inevitable challenge of being able to validate authentic payments and to detect the validity of payment receipts. Typical validation procedures involve some sort of network communication that transmits payment narrations such as the receipt number to a remote server that servers the purpose of validating the receipt against payment records (Raina, 2014).

While the approach of remote database checks can be effective in areas where network connectivity is guaranteed, the County environments offer a more unfriendly environment where network connectivity for the verification process is not always guaranteed (Adera & Waema, 2011). In addition, the collection centres might not have synched all transactions by the time verification is required. Even with presence of stable network connectivity validating some revenue such as parking fees can pose a challenge since the time of validation might not be enough to accommodate the network request-response life cycle for example when the vehicle in question is being driven.

In an offline system, there are also plenty of challenges that come about which might affect the generation of unique receipt numbers (Johri, 2008; Christin & Safavi-Naini, 2014). Conventional receipting involves using a sequential auto increment key in the database to easily generate distinct receipt numbers. In cases of multiple offline device terminals, there is risk of collision of the serial identity keys since every device has a different offline database (Johri, 2008). To ensure uniqueness of fields from various offline databases that are synchronised to a central database connection hash algorithms can be used with the input function using the unique identification details of the offline databases to generate unique records.

Offline verification in practice can also be achieved by the use of cryptography (Pedro, 2015). In Kenya, the prepayment electricity meter system uses tokens to identify the value and authenticity of purchased tokens without network communications to verify that the payment was actually made.

2.6.1 QR Codes for Receipt Verification

The issue of receipt verification is also still challenged by the possible length of receipt numbers that may require typing into a verification system. If the receipt number is as a result of a cryptographic function, this could be a very long string that could take a lot of time to input (Simon, 2013). There is need to have some form of optical recognition system to the process of data input easy. QR code have the advantage of being able to store huge amounts of information. QR codes are versatile and can accommodate information of this nature (Raina, 2014).

2.7 Conceptual Framework

Figure 2.4, the conceptual framework, links the different components that meet implementation objectives the revenue management system. It shows how various components

interact and the protocols that bind them together to provide an integrated receipting and receipt verification model. The model comprises of mobile point of sale collection devices that are connected to a Bluetooth thermal printer. These collection devices have the ability to work offline. They serve as cashier units that receive payment from customers. Once collection transactions have been processed a data synchronization process is initiated. This process involves sending transactional data to a central web server over HTTPS. The central server is connected to a number of service components that contain monitoring and data storage sub systems. Web client applications then provide real time reporting provisions through email and SMS notification channels.

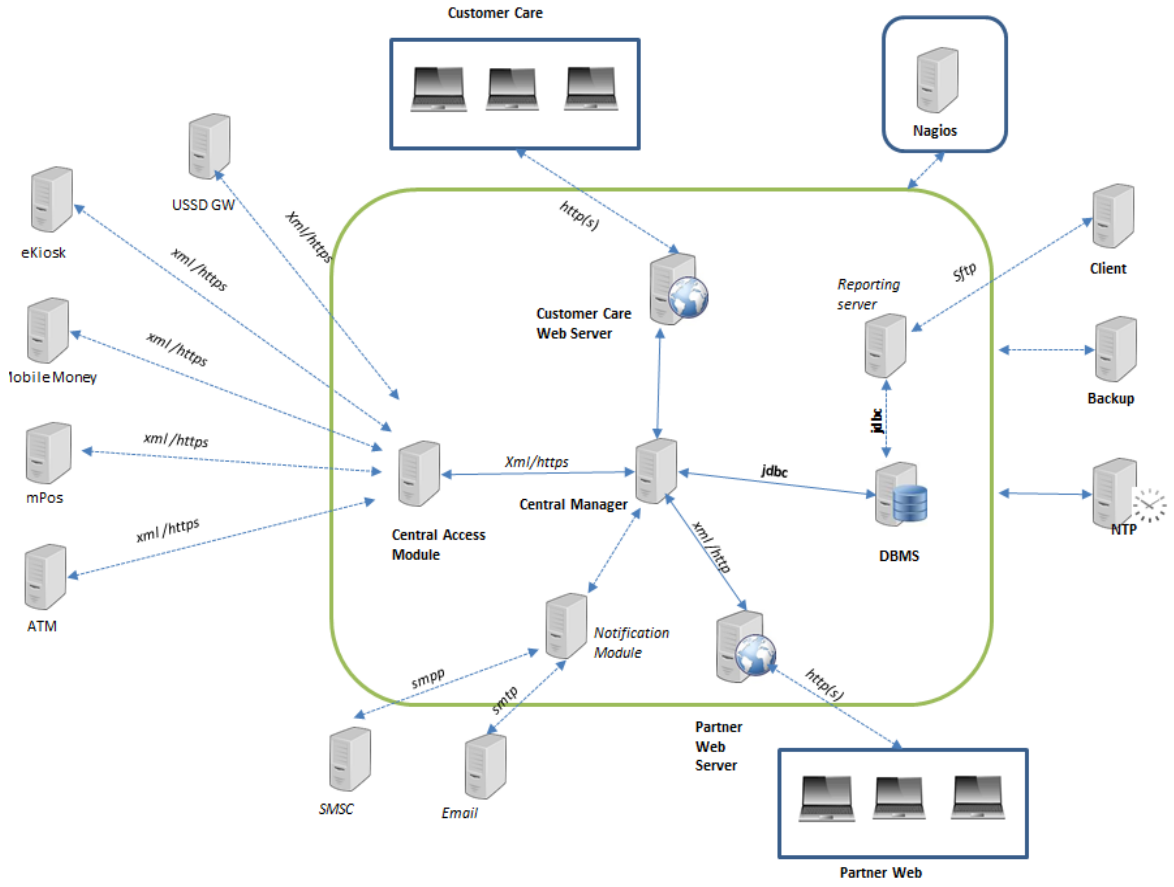


Figure 2.4: Conceptual Framework

2.8 Conclusions

The implementation of a robust revenue management system in the Kenyan setting needs to be hitched on a mobile platform that will be used for the collection, validation, updating,

reporting, management and reconciliation of all county revenues. The solution should ameliorate transparency and accountability of the revenue collection processes, embedded control mechanism, data security and reliability.

Development of such a system should be well designed with multiple technologies that can support long hours of network unavailability as well as shortage or lack of electricity. These factors bring the need to ensure that all components of the solution are well integrated in a way that can enable operations to work offline.

Previous research findings show that revenue collection applications have been dependent on network connectivity. There is a research gap on the possible use of cryptography as an alternative to providing offline verification of receipts. There is therefore a need to develop a standard algorithm that can be used to enable reliable verification of receipts. Such an algorithm needs to enable unique generation of receipt numbers that can be decrypted to provide relevant information that can be used to verify the validity of receipts.

Chapter 3 : Methodology

This chapter entails the detailed step-by-step procedures that the research went through. The selected software development methodology was based on one that delivers an end product faster in one iteration of development. The aim was to follow best software development and research practices to develop a high-quality product within the scheduled project duration.

3.1 Research Design

The research followed a collaborative approach to themed reviews of the current revenue management processes. The first step was to determine metrics for evaluation of counties revenue collection potential through consultative meetings with identified county staff. Followed by data collection and analysis procedures aimed at understanding the current system.

The phases that this research went through are:

a) Planning and mobilisation: This will be the inception phase and will be facilitated by initial meetings with a few identified county staff and people with knowledge of county revenue collection operations. The objective of this research phase was to gain an understanding of current county operations that are tied to revenue collection and accounting.

b) Review status of the current revenue streams: This stage involved an analysis of the prevailing streams, processes and systems in county revenue collection operations. This analysis was done by analysing technical and user documentations of the current system as well as reports from the Commission of Revenue Allocation.

c) Preparation of reports: Following the above processes, the research prepared a draft report for review by a member of the county revenue collection staff.

3.1.1 Research Tasks

Task 1.1: Consultation Meeting

A consultation meeting with identified county staff kicked off the research assignment. It involved selecting main revenue collection user roles for interviewing. The key considerations at the consultation meeting included the following: -

a) Clarification of the key processes in the revenue management process.

b) Discussing and identifying loop holes of the current revenue management and receipt verification system.

c) Identifying possible improvements to the current revenue collection structure.

d) Request to obtain relevant documents and other information materials to be used for reference.

The following are the outputs of the consultative meetings:

a) The research gained an understanding of challenges facing revenue collection and automation in county governments.
b) The study identified causes of revenue leakages and prevalence of counterfeit receipts in the current system.
c) The study identified ways of improving revenue collection and receipt verification in county governments.

Task 1.2: Preparation of a Findings Report

Following the deliberations at the consultation meeting, the research prepared a report of the findings. The findings report entailed the following:

i) System requirements specifications.
ii) Work plan and system design.
iii) System components.

Task 1.3: Documents Review

The Research conducted desk review of the relevant documents to extract pertinent information for use in the system review and development process. The purpose of the desk research of the documents was to: -

a) Gain in-depth understanding of county revenue management process.

b) Understand the internal operation and control environment of county governments in relation to revenue collection and payment verification.

c) Understand the counties' external operating environment with a focus on technological advancements.

The following are the documents reviewed at this stage:

a) Documentation of the current system
b) Revenue Reports
c) Paper-based Receipts
d) Control Sheets
e) Baseline survey reports for the meetings.

3.2 System Development Methodology

The system development methodology applied in this research is the Waterfall methodology. The Waterfall model includes seven stages as shown in Figure 3.1. Each stage has its own processes which must be complete before development can proceed to the subsequent stages (William, 2009).

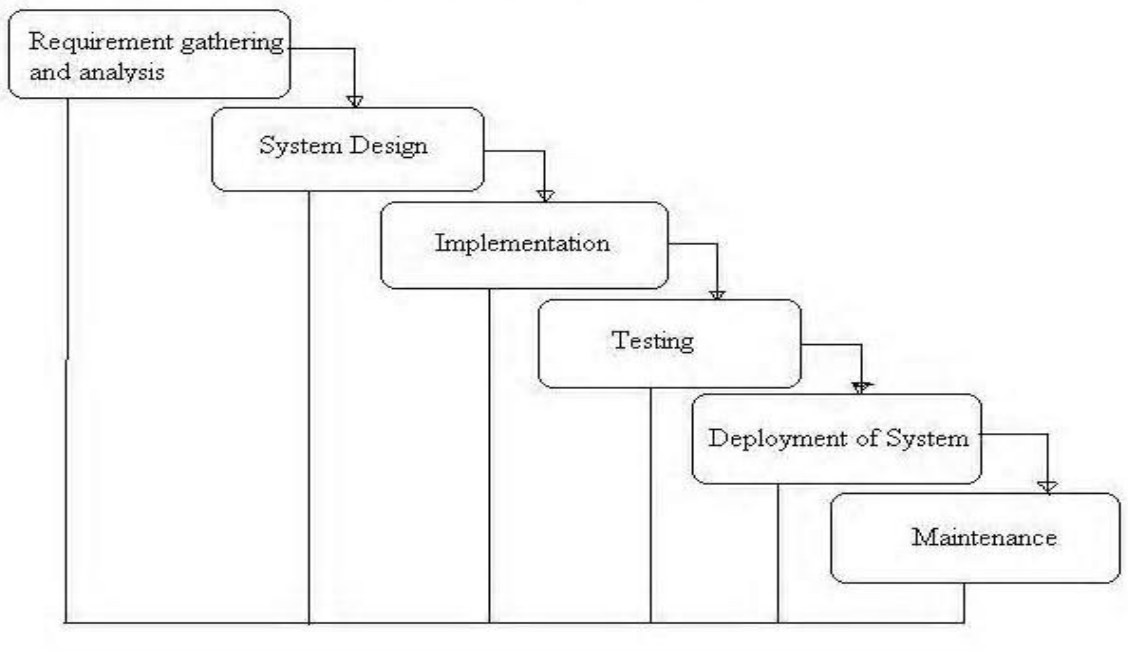


Figure 3.1: System Development Life Cycle Phases (Adapted from William,2009)

3.2.1 Requirements Gathering and Analysis

This task involved identifying the project specifications. All possible requirements of the developed system were captured at this phase and documented in a requirement specification document. In this stage, the research clearly identified and investigated the type of project to be developed and the ways of developing the whole project in order to ensure the overall processes are running smoothly and successfully.

This development phase was also used to analyse the business processes for the county revenue management system that served as a use case for the proposed system. The study investigated the receipting process as well gained an understanding of the sources of counterfeit receipts.

The Research used user stories and context diagrams to model requirements. These tools were also used to analyse how automation can impact on collection of revenue and to find out what could be done to improve the current existing system. The research used the common data collection techniques in recording the data needed for development and improvement of the system. The techniques included interviews and observation.

3.2.1.1 Interviews

The managers, staff and stakeholders of the revenue collection division in the County Government of Bungoma were interviewed. Questions addressed to them asked for their view on how the current system works and their opinion on whether introduction the proposed system will be of any value to them.

3.2.1.2 Observation

Observation is the systematic recording, description, analysis and interpretation of people's behaviour (Roger & Jupp, 2006). It is concerned with collecting primary data through looking and taking note. It is mostly used if research concerns what people do and how they do them. Observation was used to observe how users interact with existing revenue collection systems, so as to note the difficulties they encounter and suggest ways to improve on them.

3.2.2 System Design

The system design phase involved a number of software modelling and design techniques which were, use case diagrams, data flow diagrams and sequence diagrams. The requirement specifications from the first phase were studied in this phase and the system design prepared. System design helped in specifying hardware and system requirements and also in defining the overall system architecture.

3.2.3 Implementation

The system was designed to be an all–in–one enterprise revenue collection and management system with single secure database powered by PostgreSQL (Obe & Hsu, 2015) and supported by Java Enterprise Edition (Farley, Crawford, & Malani, 2006). It was designed to integrate to a wide range revenue collection components that will enable county governments to perform efficient collection and to get up-to-the minute prudent decision making reports.

3.2.3.1 Database Development Tools

Database design of the system employed the use of Entity Relationship Diagrams as well as Schema Design Tools. E-draw max tool was used to capture the database sketches because it contains numerous components that conform to database design standards. Navicat (Ozar, 2012) was employed for the schema design because of its capabilities to handle Model to Schema automation as well as data and structure synchronisation. Due to the reporting requirements of the enterprise application PostgreSQL (Obe & Hsu, 2015) was the preferable database management engine. Postgres offers a reliable database engine that can handle transactional processes gracefully. Database management is therefore serviced by PgAdmin (Obe & Hsu, 2015).

3.2.3.2 Programming Environment

The main programming tool of development that was employed is Java. This is because it offers an intuitive Object Oriented environment that makes code easily maintainable. In addition, Java offers a set of standard libraries that can support the cryptographic algorithms that are required to meet the data encryption and security standards of the system. To make database access functions easier, the application used the Java Persistent API (McKenzie, 2008) along with Groovy (Barclay & Savage, 2010) and the Spring Framework (Gutierrez, 2014).

3.2.4 Testing

All the units developed in the implementation phase were integrated into the production system after testing of each unit. The entire integrated system was also tested for any faults or failures.

3.2.4.1 Level of Testing

The system was subjected to testing on different levels to ensure that all components of the system worked coherently. Both the mobile application and web-based system were tested independently through a number of test phases that are outlined in this section.

Unit Testing

Unit testing was used to test the minimal software components. Unit test was done by writing tests using the JUnit Test Framework (Appel, 2015). The tests used dummy data as parameters to low level functions and the return value matched against test results. Each unit (basic component) of the system was tested to verify that the detailed design for the unit had been correctly implemented. Since the system was implemented in an Object-oriented environment, this

was done at the class level. The minimal unit tests included testing of constructors, destructors and member functions.

Integration Testing

This test level helped to expose defects in the interfaces and interaction between integrated components (modules). This test was done by inspecting communication messages such as HTTP message bodies and headers against expected request and response parameters. Progressively larger groups of tested software components corresponding to elements of the architectural design were integrated and tested until the software worked as a system.

Functional Testing

Functional testing was used to conduct tests at all levels. Functional testing began at the class level where member functions were tested against a predefined set of parameters and return values. Interfaces were also tested by implementing them on dummy classes and testing them for proper functionality as defined in the specification.

System Testing

This test was used to analyse the completely integrated system to verify that it met its requirements specifications. The test was done by developing a list of expected system behaviours based on user actions and comparing them with the observed system behaviour after invoking the action on the test environment.

Acceptance Testing

This test needs to be conducted by the end-user, customer, or client to validate whether or not to accept the product. Acceptance testing was performed by conducting a pilot test at the county government of Bungoma. Revenue collectors tested the system by performing actual revenue collection transactions and receipt verification checks.

3.2.4.2 Test Plan

The testing phase required completion of various tests to ensure the accuracy of programmed code, the inclusion of expected functionality, and the interoperability of application and other network components. Thorough testing was critical to ensuring the system met county governments' revenue collection requirements.

The first testing approach was to employ a bottom-up approach which tests smaller components first and progressively adds and tests additional components and systems. The other

approach would be top-down which first tests major components and connections and progressively tests smaller components and connections.

The bottom-up tests began with functional (requirements based) testing. Functional tests were conducted to ensure that expected functional, security, and internal control features are present and operating properly. Testers then completed integration and end-to-end testing to ensure application and system components interact properly. Users then conduct acceptance tests to ensure systems meet defined acceptance criteria. Any county of deployment will be expected to review and complete user, operator, and maintenance during the testing phase. Additionally, they should finalise conversion, implementation, and training plans.

Once all the features of the system had been tested by running each function available in the system. The results of the tests conducted on the system were analysed. Only after getting satisfactory results of testing was the system be put into use.

3.2.5 Deployment of the System

Once the functional and non-functional testing were complete, the product was deployed to the production environment. The deployment phase involved installing supporting applications into production environments. Additionally, the users were requested to input and verify data, configure and test system and security parameters, and conduct post-implementation reviews. This phase was conducted as a pilot in the county government of Bungoma and required the help of the county's management to circulate implementation schedules to all affected parties and to notify users of any implementation responsibilities.

3.2.5.1 System Deployment Strategies

The research considered a number of system deployment strategies. The first approach of consideration was phased conversion. Phased conversion is an incremental approach where individual components of the new system would be implemented one by one. Different modules of the new and old system would be used together until the whole system is installed.

Another option would be direct cut-over, where, the old system would be directly replaced by the new system. It is the most risky conversion because there may still be major problems that will not be uncovered until the system has been in operation. Direct cut over may be necessary when the implementation time is very short. It is also called crash conversion. An alternative to this is pilot conversion. Here, one part of the county government would use the new system as the

rest of the revenue collection division uses the old system. When one part of the revenue collection division is satisfied with the new system, the rest of the organisation can start using it.

After considered a number of options, the research found parallel conversion most suitable. This is a type of conversion in which both new and old systems operate together for a period of time**.** It is the safest type of conversion since the results of both systems are compared. The old system can be used until all problems in the new system are removed.

Parallel conversion was the most preferable way of implementing this system. This is because of the simple reason that it is safe for the users of the system as t0hey will still be using the old system until they all learn to use the new system. It will also make change easier and if the users cannot adapt to the new system it can be put aside. Other reasons that drove preference to this approach are:

i. If anything goes wrong with the new system, the old system will act as a back-up. The old paper-based system should be used in parallel with the new system as errors encountered in the early stages of development should not halt the operations of the county. The old system should be used as maintenance is carried out on the new system.
ii. The outputs from the old and new systems can be compared to check that the new system is running correctly. The records stored in the county revenue management system should be consistent in ensuring service delivery to the general public. Therefore, keeping the old system to countercheck records when need arises is very crucial to enhance data integrity.

3.2.6 Maintenance

There are some issues which are expected to come up in the client environment. This phase is to be used to fix those issues and realise updates. Also, to enhance the product some better versions will be released at this stage. Maintenance will mainly be done to deliver changes to the customer environment.

3.3. Proposed System Modules

The proposed approach was to develop a consolidated mobile collection terminal solution built on an electronic payment framework and designed to be a centralised payment system/ database for county governments to receive all types of revenue, be they Cess, Land Rates, Business Permits and parking fees. The solution targets to provide an integrated reporting component for county governments to manage revenue collection in real time regardless of the

type of customer, geographical location, or payment method. The System should have both offline and online support.

3.4 Time Plan

Appendix A outlines the project development time in accordance to the Waterfall methodology. The Gannt chart in Appendix A, Figure A1 shows the research and development phases that the research went through in development of the system.

Chapter 4 : System Analysis and Design

4.1 Introduction

This chapter focuses on presenting details of the system architecture and design based on requirements specifications. The design revolves around developing a system that enables offline collection of revenue and verification of receipts. The system targets two main platforms. The first is a mobile based application that runs on a mobile point of sale terminal. The mobile application will be served by a web based application that will support a RESTful API.

4.2 Functional Requirements

i) The system should provide a user authentication module. This module must ensure that users are properly authenticated before they are authorised to perform any operations on the system. In addition, the system needs to provide role based access control and users should only have access to features that are tied to their roles. This is achieved by showing menus and icons relevant to the current logged in users. The system should also support low level functional access controls. Function level access control will ensure that access to all endpoints and resources are also locked to user sessions to prevent possible escalation of privileges by direct URL traversal.

   Based on the objectives of the system, the following are the identified user roles.

   a. System Administrator: Performs administrative functions related to setting up users and configurations.
   b. Revenue Collector: A cashier handling point of sale terminals with access to generate receipts. This user will also be responsible for handling verification of receipts using QR code scanners.
   c. Revenue Officer: Managerial user with access to operational reports.
   d. Accountant: The accountant has access to accounting reports.

Authentication should also use device serial numbers to give collectors access. This will help lock access to devices they have been assigned to. All users of the system should therefore be able to Log in using a username and password. In addition, they should be able to log out as well as manage their own passwords. System administrators should have rights to manage users access by performing functions such as deactivating users or resetting passwords.

ii) The system should generate all standard financial statements and revenue collection reports in real time. The reports include: daily transactions; collections by revenue stream; detailed collector reports; cashier transactional reports and accounts breakdown; It should be flexible to allow the County authorised officers to add additional revenue sources with minimal customisation.

iii) The system components should have a built-in ticketing system for receipt generation that has an offline interface to enable collections in rural areas. The system should also have a real-time reporting module that reports revenue collection in real time.

iv) The system should be able to map the counties revenue sources including revenue groups, ministries and streams.

v) The system should support both cash and other payment options including mobile money.

vi) The system needs to facilitate collection, monitoring, enforcement and reconciliation of revenue at all levels. It should support handheld devices that will enable officers to make receipt verification checks. Verification should be performed even in cases of network unavailability.

vii) The system must be able to provide portable, offline revenue collection capabilities: By employing lightweight devices with attached printers that can enable collection on a handheld computer and have receipts printed.

viii) The application collection components need to work in both online and offline modes as a measure against network failures.

4.3 System Architecture

The system mainly provides a mobile based application that enables county governments to collect payments and generate receipts using portable devices. The application needs to first of all cache data dictionaries and authenticate the current user before it is able to collect revenue. The mobile application therefore has its own data storage solution that is powered by a SQLite database. The local database keeps track of all transactions that a particular collector has collected based on payments received. The mobile application not only serves as a cashier station but also offers an interface to enable verification of payment receipts. To enable accurate reporting the application has automatic data synchronisation capabilities. This is achieved by automatically detecting when the device is online and synchronising data with a central database.

The centralised database also serves as a store for all configurations and data dictionaries. The main database is interfaced by a web based application that enables system administrators to configure users, devices as well revenue charges. The web based system also serves as a reporting component that gives accountant and revenue officers access to operational and accounting reports.

Figure 4.1 illustrates the main components that make up the system and how they interact to enable offline receipting and verification. The mobile application mainly communicates with the backend using a JSON API. Each mobile point of sale terminal will be serviced by an SQLite database that will handle storage of transactions. The point of sale terminal communicates with the system backend over HTTP. This communication is mainly used to cache data definitions on the terminals as well as synchronise data between the collection devices and main database.

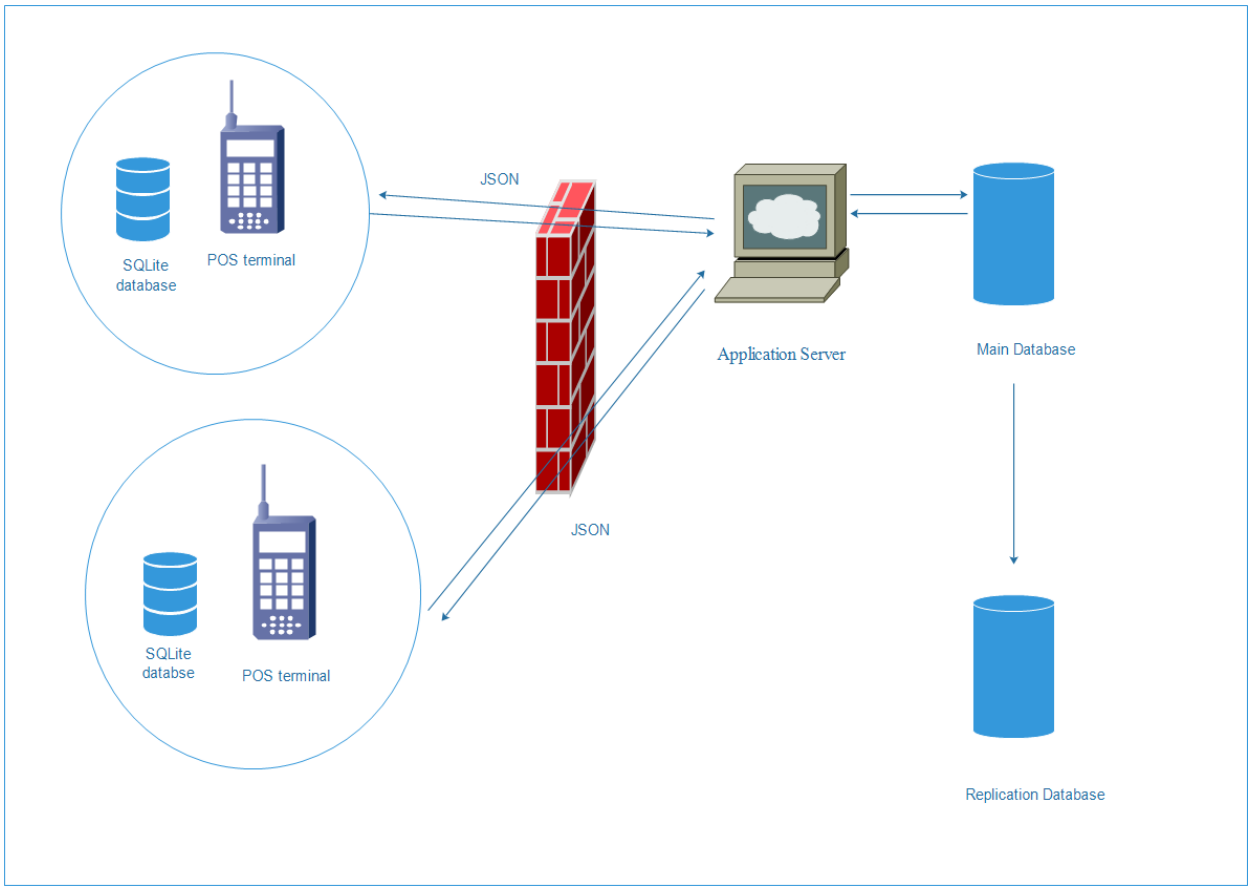


Figure 4.1: System Architecture

4.4 System Design

The system design phase employed the use of various software modelling tools and techniques to present the desired system architecture. The design tools target to provide a clear definition of a solution that meets the functional and non-functional requirements. The following is a list of identified design techniques that have been employed:

i. Use case diagram
ii. Class diagram
iii. Context diagram
iv. Entity relationship diagram
v. Sequence diagram
vi. Wireframes

4.4.1 Use Case Diagram

The system has a role based access control structure that categorises various user groups based on the functions they are expected to perform in the system. The most critical role is that of revenue collection and receipt validation which is supported by an administrative user role group that enables configurations. The other users primarily serve a monitoring role and therefore mainly have access to reports, audit logs and analytics. The following are the identified user roles and actors that perform main system functions.

i. System Administrator: These actors are mainly responsible for configuring users as well as devices and various fees and charges to be levied on customers. The system administrator configures the locations and grouping of charges. Once all the charges have been set up, collectors will need to cache them on their devices through a data synchronization process.
ii. Revenue Collector: Revenue collectors are the handlers of mobile point of sale terminals and serve as mobile cashiers. Collectors are also responsible for receipt verifications by QR code scanning. The revenue collection process involves using a point of sale terminal to receive a payment from a customer which the generates a unique receipt number that can be verified offline.
iii. Revenue Accountants: This user group has access to accounting reports as well as ensure that all the monies collected are properly banked. The revenue accountant need to verify

end of day collections pulling the synchronized collections of a given collector on a specific date. One the actual amount has been obtained, the accountant will need to check the banking slip of that corresponding date and post the banking to a collector ledger that tracks all collection verses banking amounts.

iv. Revenue Officers: These actors represent the managerial group that has access to various operational reports as well as analytics. These reports are collector ledgers, collection by group, zones and subgroups as we as detailed transactional reports for every mobile point of sale terminal.

In order to meet the functional requirements of the system the following are the process that are expected in the system:

i. User Management: This is where a system actor creates a user and assigns appropriate role based privileges to enable a user to perform defined functions.
ii. Device Management: This process encompasses management functions that enable receipting devices to be assigned to revenue collectors.
iii. Revenue Configuration: This process facilitates grouping of charges to ensure that customers are correctly charged according to the finance bill.
iv. Receipting: This process entails generation of receipts after a customer pays.
v. Verification: This process uses a specific narration to confirm that a customer has actually paid for a given fee. Verification should give detailed information on the validity period of a specific payment.
vi. Reporting: Actors are given access to view reports on various operations that are undertaken in the system.

4.4.2 Use Case Description

Table 4.1: Use Case Description

Actor	Use Case	Description
System Admin	Crete user account	**Primary Actors:** ICT officer **Stakeholders:**

		Revenue officers, Revenue collectors, Accountants **Preconditions:** Need to add a user to the system **Postcondition:** Request to add a user to the system submitted **Success Scenario:** User created and a username generated along with a password
	Log in	**Primary Actors:** System administrator, Accountants, Revenue officers, Revenue collectors **Stakeholders:** System administrator, Accountants, Revenue officers, Revenue collectors **Preconditions:**

		Need gain access to the system **Postcondition:** User authenticated **Success Scenario:** A session created to enable and user granted access to defined functions
	Revenue Configuration	**Primary Actors:** ICT officer **Stakeholders:** Revenue collectors **Preconditions:** Need for amendments to the current revenue configurations **Postcondition:** Request to add, update or deactivate a revenue group, sub group or charge is submitted **Success Scenario:** Revenue configuration successfully updated
	Device Management	**Primary Actors:** ICT officer

		Stakeholders: Revenue collectors **Preconditions:** Request for allocation or deallocation of a device to a user The device has been added to the system Current user is not able to log in using the device **Postcondition:** Update requests to change device allocation status received **Success Scenario:** Device allocated to a user. User is able to log in using the device
Revenue Collector	Cache data dictionary	**Primary Actors:** Revenue collector **Stakeholders:** Customer **Preconditions:** Collector is logged in Device is online

		User needs have latest charge configurations **Postcondition:** Request to download latest data dictionary definitions **Success Scenario:** Data dictionaries updated and saved on the local database
	Process receipting transaction	**Primary Actors:** Revenue Collector **Stakeholders:** Customer **Preconditions:** Collector is logged in Bluetooth is enabled Thermal printer is connected **Postcondition:** Charge corresponding service fees Submit command to process receipt **Success Scenario:** Payment details saved and receipt generated

	Perform receipt verification check	**Primary Actors:** Revenue Collector **Stakeholders:** Customer **Preconditions:** Collector is logged in Collector has transaction narration or receipt to check **Postcondition:** Receipt QR code scanned and submitted for verification **Success Scenario:** Transaction details displayed
	Print end of day summary report	**Primary Actors:** Revenue collector **Stakeholders:** Revenue accountant **Preconditions:** Collector is logged in Collector has performed several transactions on that day **Postcondition:**

		Application commanded to compute and print the end of day summary report **Success Scenario:** Summary report printed from the mobile device
	Synchronise offline transactions	**Primary Actors:** Revenue collector **Stakeholders:** Revenue officer **Preconditions:** Collector is logged in Device connected to the Internet **Postcondition:** Request to synchronise a record submitted **Success Scenario:** Transaction saved on the main database and record marked as synchronised
Revenue Officer	View operational reports	**Primary Actors:** Revenue officer

		Stakeholders: Revenue collector **Preconditions:** Officer is logged in Collectors have synced their collections **Postcondition:** Report requested **Success Scenario:** Report is loaded and displayed
Revenue Accountant	View accounting reports	**Primary Actors:** Revenue accountant **Stakeholders:** Revenue collector **Preconditions:** The accountant is logged in Collectors have synchronized their collections Accountant applies report filters **Postcondition:** Report requested **Success Scenario:**

		Report is loaded and displayed

4.4.3 Use Case Diagram

Figure 4.2 shows the system use case diagram. It outlines the system actors as well as the actions they perform. All users need to log in to the system in order to perform any action. All administrative functions are performed by the system administrator while revenue officers and accountants perform a supervisory role. The core tasks of revenue collection and receipt validation are performed by revenue collectors.

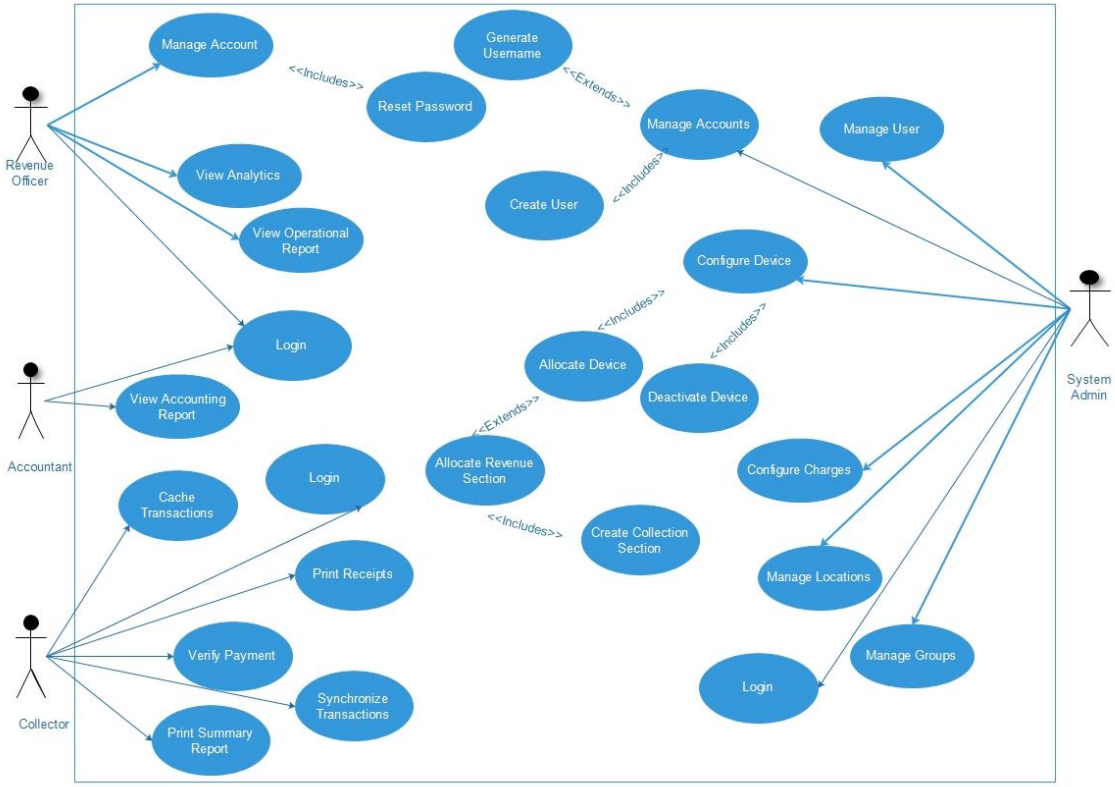


Figure 4.2: Use Case Diagram

4.4.4 Design Class Diagram

The system uses an object-oriented design pattern in both the mobile application and web based system. Figure 4.3 shows the breakdown of the classes implemented in the client mobile application.

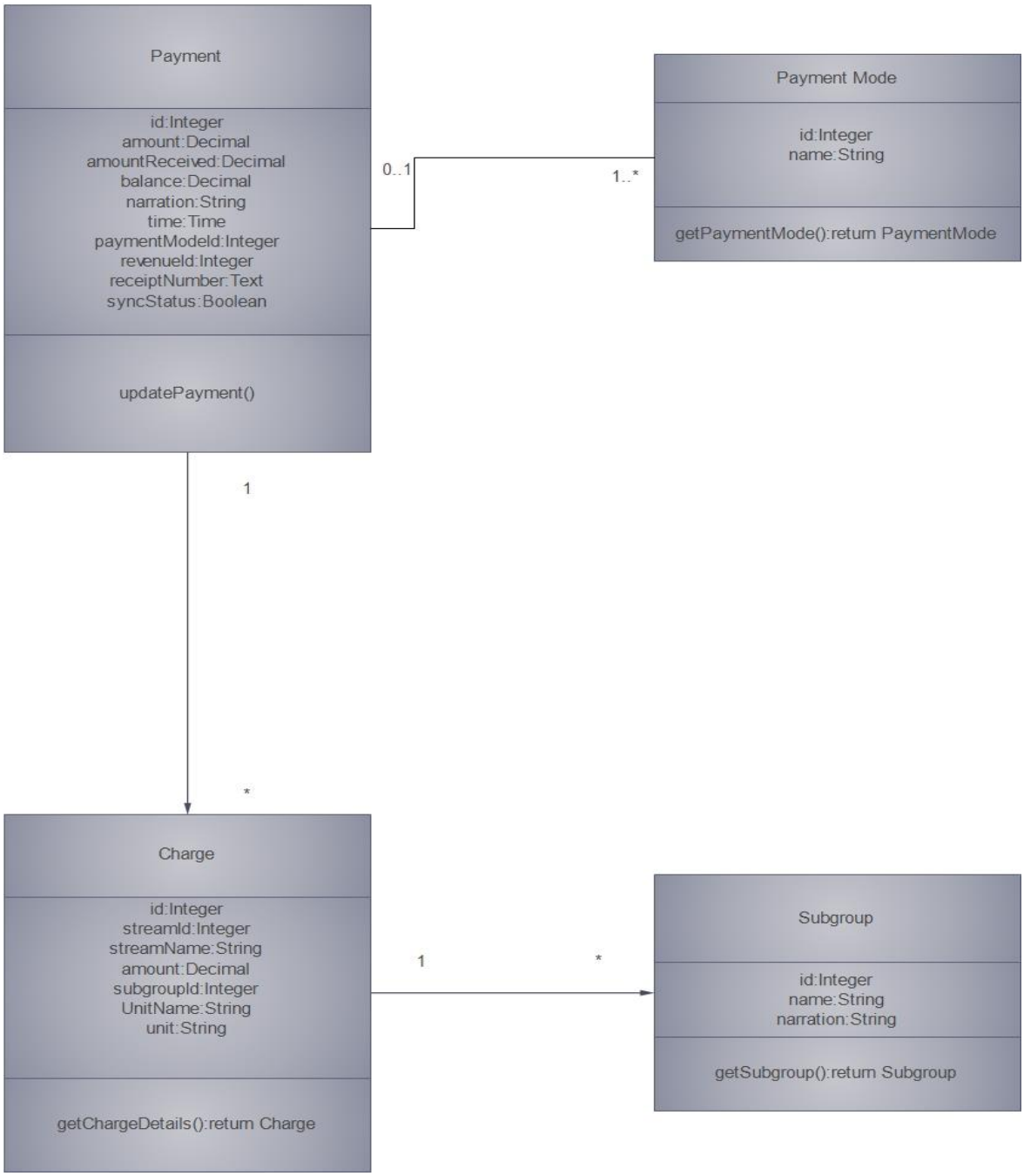


Figure 4.3: Client Application Class Diagram

Figure 4.4 below shows the web application's class diagram. The web application follows an MVC web structure. The classes interact to facilitate receipt and verification of payments.

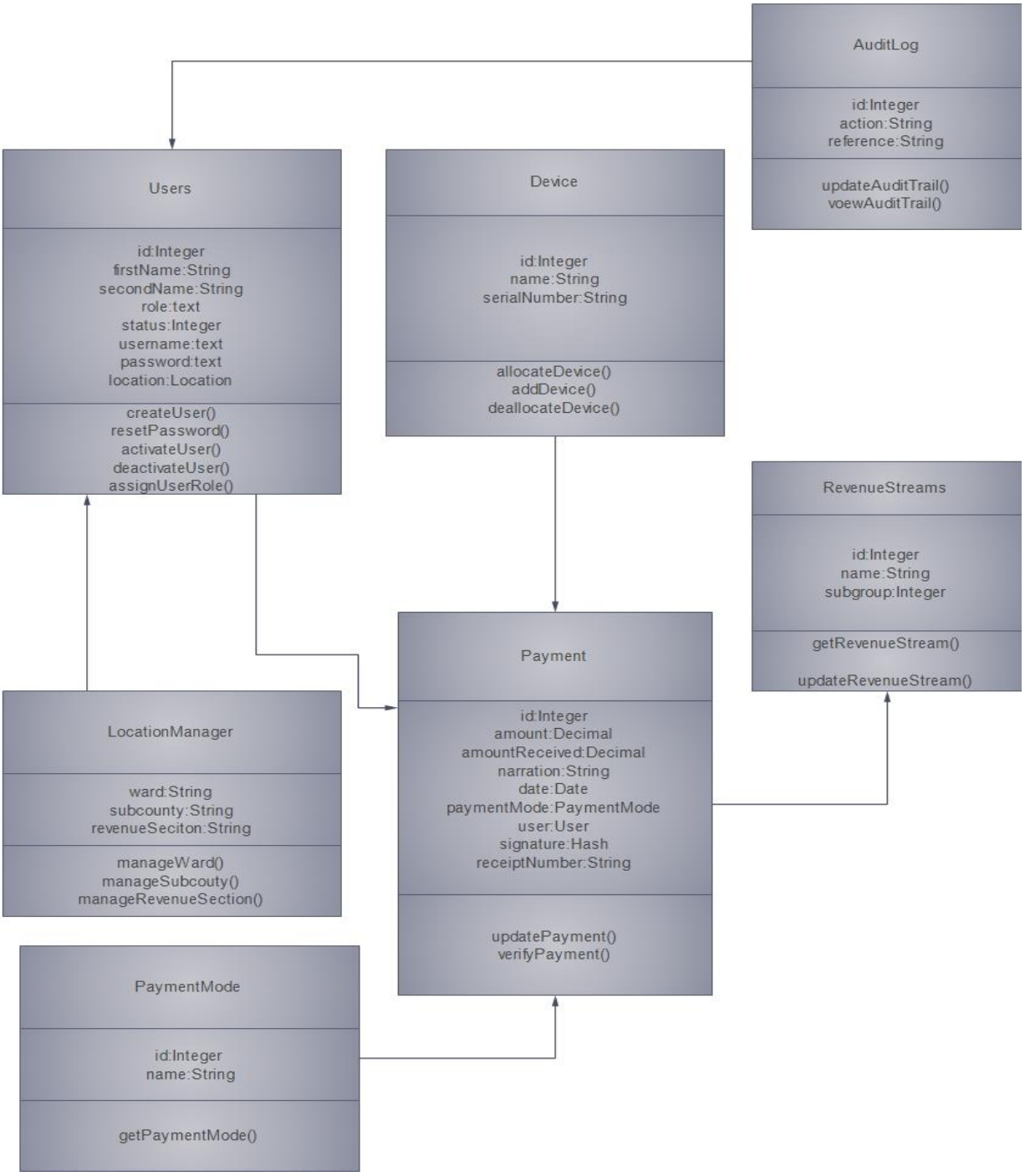


Figure 4.4: Web Application Class Diagram

Table 4.2 gives a detailed description of main functions performed by every class to facilitate collection of revenue and support other system operations.

Table 4.2: Class Descriptions

Class name	**Description**
User	This class is responsible for managing all user related operations. It processes requests to create user accounts and offers helper functions that enable administrators to perform user support operations such as password resets and access control.
Device	The device class is responsible for managing point of sale terminals. It provides methods to enable management of device allocations.
Location Manager	This class provides functions to enable mapping of revenue by location. It contains helper functions that will enable configurations of points of revenue collections for accurate charging of fees as well us correct classification of reports.
Payment	The class Payment manages all payment related transactions and provides security mechanisms to ensure that all payments are done securely. It also provides helper functions to enable validation of payment receipts.
Revenue Streams	The class revenue stream is responsible for grouping of various fees and charges according to appropriate reporting groups.

Audit Log	The class keeps track of all user actions to facilitate auditing operations.

4.4.5 Context Diagram

The context diagram presented in Figure 4.5 shows a high-level model of how various entities interact with the system to enable collection and verification of revenue. The system administrator is the primary entity who provides various configurations to support other user operations. Major system operations of payment processing and receipt verification are carried out by revenue collectors. All operations performed by the collectors are then monitored by accountants and revenue officers through generation of receipts.

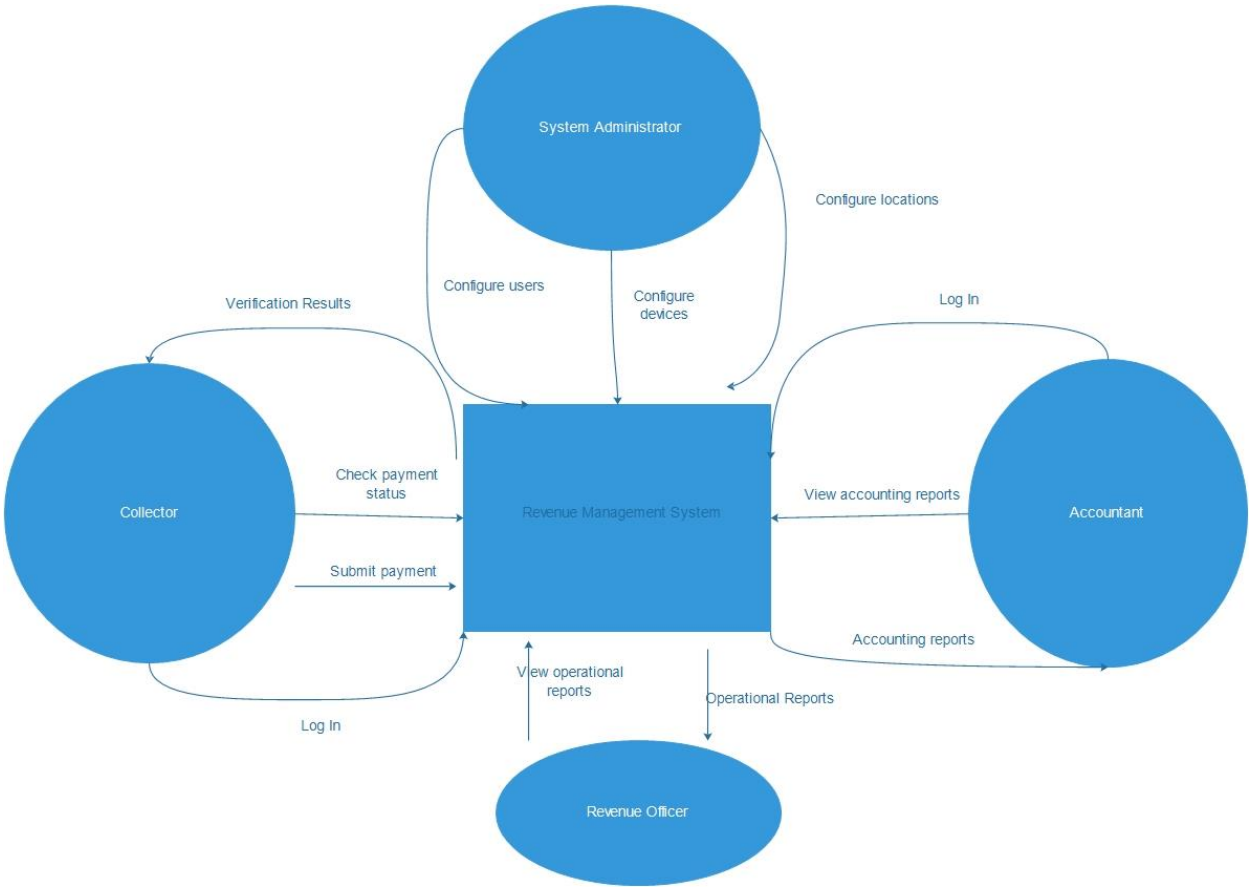


Figure 4.5: Context Diagram

4.4.6 Sequence Diagram

The sequence diagram presented in Figure 4.6 illustrates the sequence of information flow in the revenue system. Before system operations commence, the administrator needs to configure the various fees according to revenue streams. Revenue stream configurations are followed by configuration of point of sale terminals that will be used by the collectors. Once a point of sale terminal is correctly configured, it is allocated to a collector. The collector can therefore proceed with collection of revenue. Revenue collection begins when a client makes a payment. The payment is received the transaction is saved on the device before a receipt is generated.

Once a receipt has been generated and issued to the customer verification of the transaction can be performed. The receipt verification process should be possible even in cases where both devices that generated the receipt are offline. In the course of the collection process collectors can generate summary reports with details of how they are faring on in the collection process.

After collection, reports are made available to accountants and other officers. The reports are well classified according to various revenue streams in order give a comprehensive analysis of how various revenue streams of collection are performing. Although reporting in the web application might not be in real time, collectors will have up to date reports on collections through a dashboard on the mobile application that will support offline reporting.

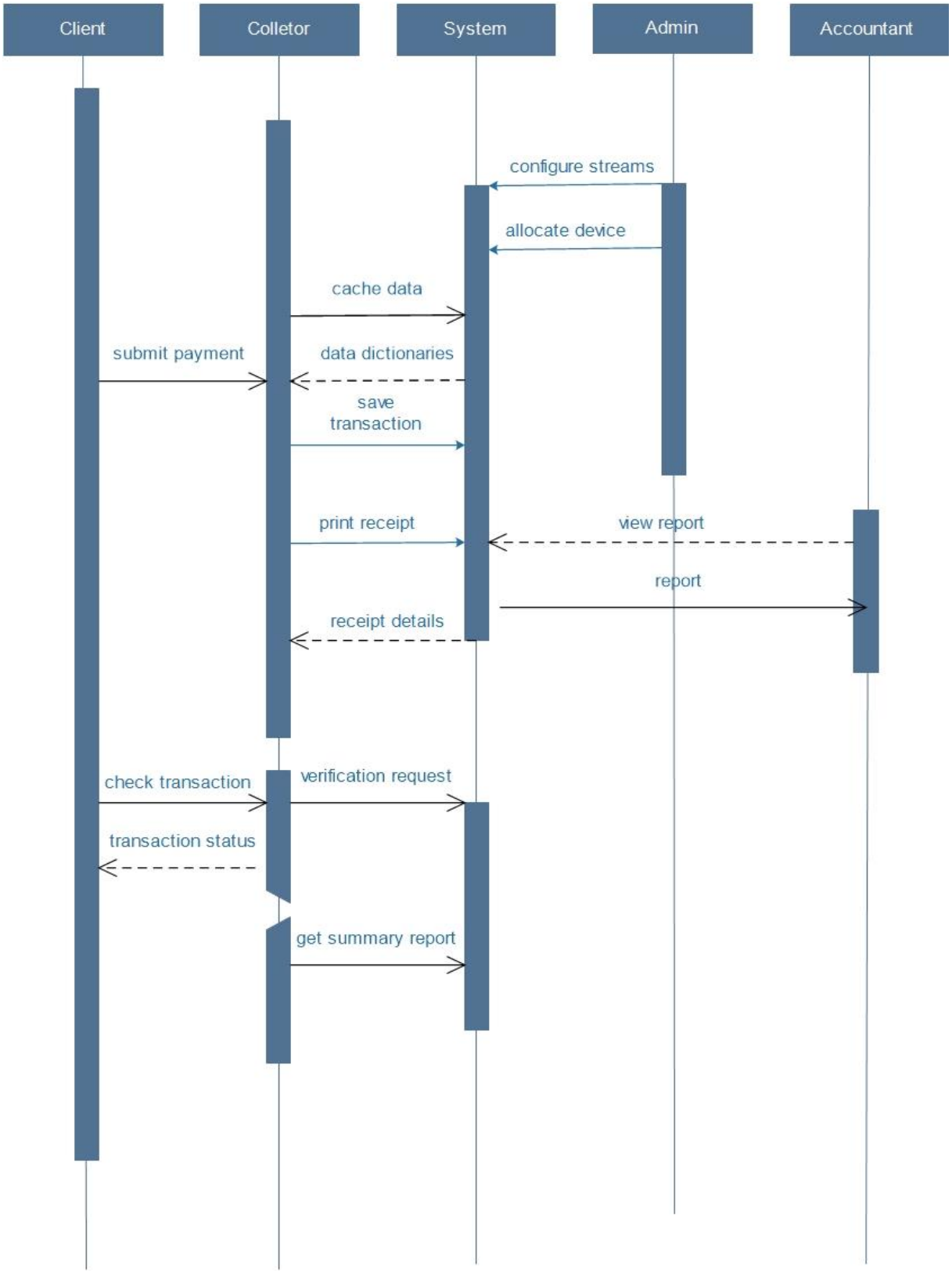


Figure 4.6: Sequence Diagram

4.4.7 Entity Relationship Diagram

Figure 4.7 shows the entity relationship diagram used to model the mobile client application data layer. The storage engine of the application uses a SQLite database to keep track of payment transactions. It consists of four tables. The payment table is the primary table which tracks all payments processed by the application.

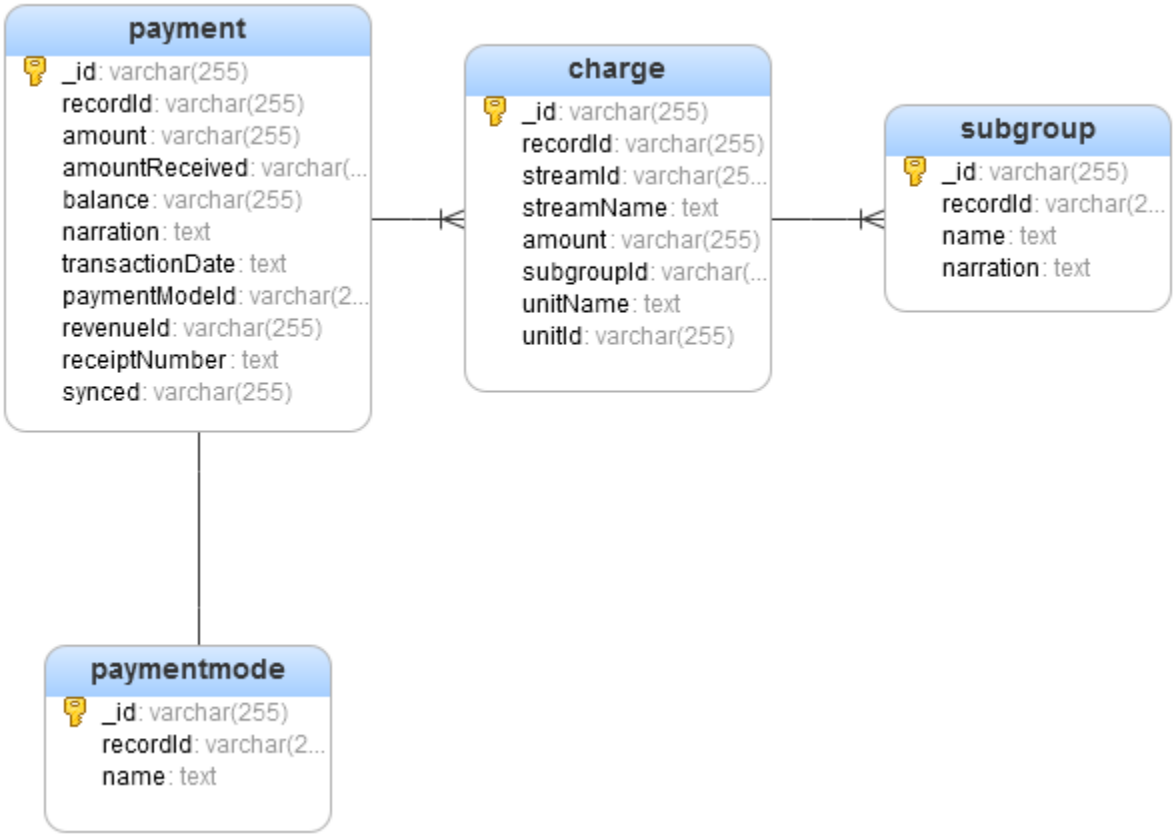


Figure 4.7: Mobile Application Entity Relationship Diagram

Figure 4.8 shows the entity relationship diagram used in the web application. It provides a storage model that is able to meet the operational requirements of the web based application. The tables hold user related data as well as track all payments processed to facilitate accurate reporting.

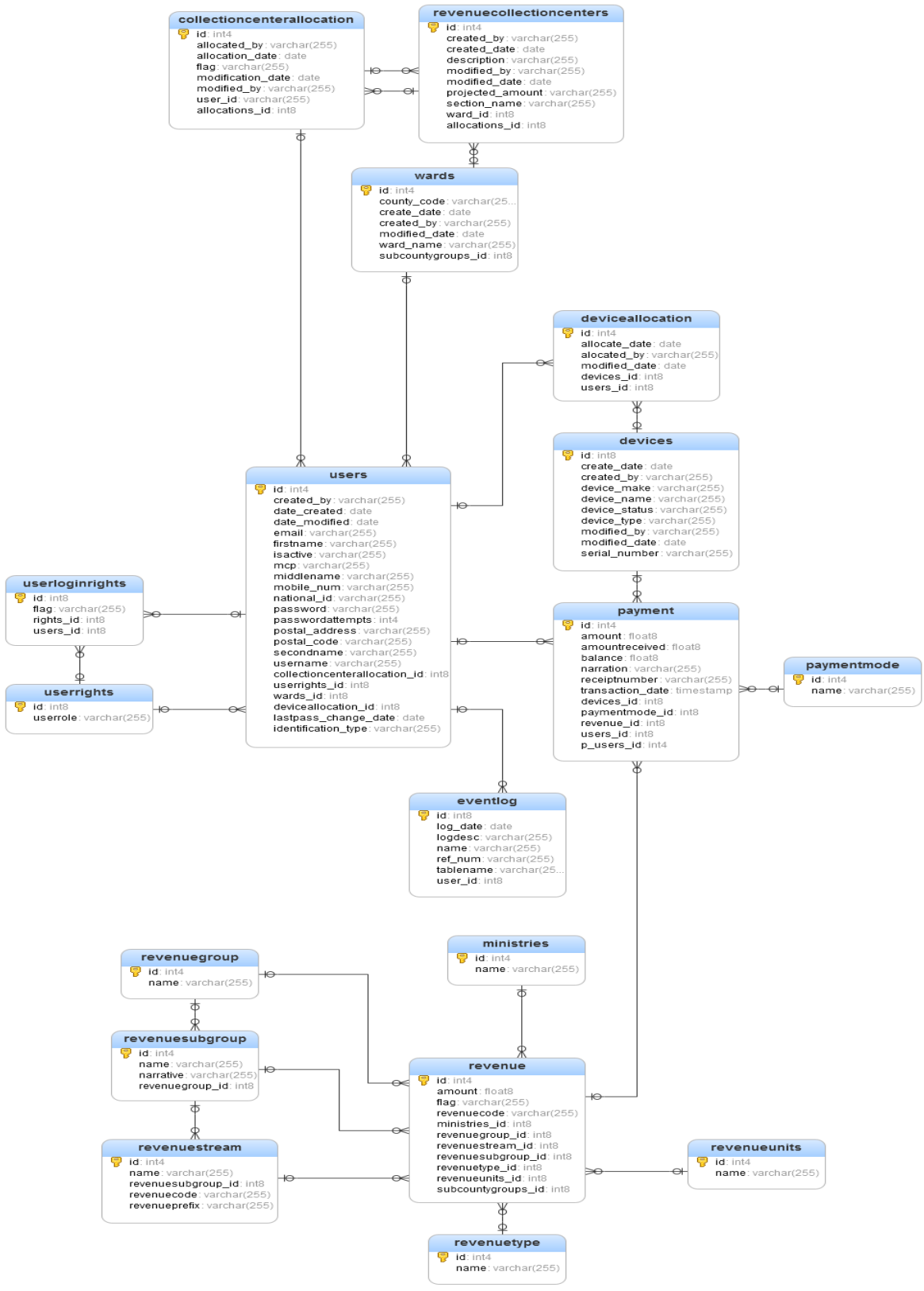


Figure 4.8: Web Application Entity Relationship Diagram

Table 4.3 shows an overview of the roles performed by tables in the database schema. It gives a short description of the table structures and key relationships that enable effective storage of data stored in the web application.

Table 4.3: Database Tables Description

Table	**Description**
Users	Holds user identification information as well as access control rights details. The table contains foreign keys that tie a user to the ward as well as the role.
Userloginrights	Contains information on the linking users to the corresponding role.
Userrights	The table has details of identifying possible roles that a user in the system can be allocated.
Devices	This table has information that identifies a particular collection point of sale collection terminal.
Deviceallocation	This is a link table that contains foreign keys track which user is allocated to a particular device
Wards	A data table that stores all wards
Revenuecollectioncenters	Lists names of all locations where revenue is collected
revenuecollectioncenters	The table is a link table that contains foreign keys to the user and revenuecollectioncenters

	table. It holds information on which section a user is allocated to
Payment	This payment contains transaction details of receipting transactions. It contains a foreign key to identify the user that performed a specific transaction as well as foreign key linking the mode of payment used in the transaction.
Paymentmode	This is a data dictionary that stores accepted payment modes.
Eventlog	This table is responsible for recording an audit trail of all the events performed by users.
Revenuegroup	Contains all revenue classifications categorised at the highest reporting level.
Revenuesubgroup	The table holds a finer classification of revenue sources.
Revenuestream	Contains the small quantifiable unit of revenue sources
Revenueunit	The table contains a list of acceptable quantity units of payment.
Revenue	The table contains actual charges that are to be levied on a customer upon payment

4.4.8 User Interface Flow Diagram

This section outlines the mock-ups that illustrate the process flow of the application. The application aim to achieve a simplified user interface that makes it easy for collectors to receive payments, generate receipts and verify the validity of receipts. The first screen of access as shown in Figure 4.9 is the log in screen that enable uses to be authenticated before they have access to the application.

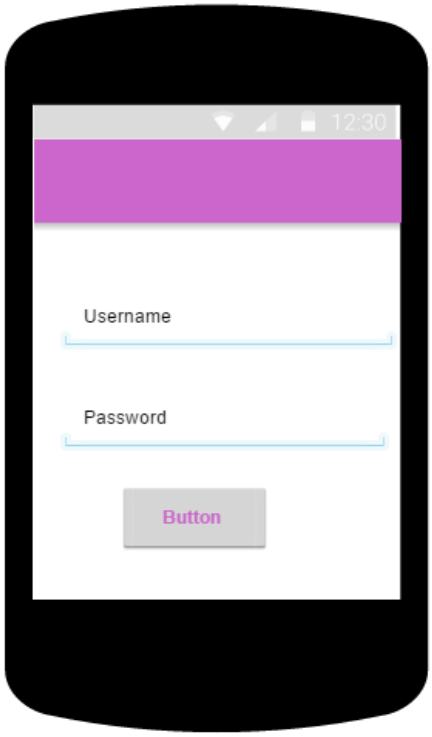


Figure 4.9: Mobile Application Log in Screen

Once users have been properly authenticated, they can start collection by selecting a revenue subgroup followed by a revenue stream then subsequently enter payment details as shown in figures 4.10,4.11 and 4.12. All collections can then be monitored through a dashboard that is designed to give a summary of the collections as shown in the mock-up screen 4.13. Users also

have access to a quick navigation menu that provides an interface which provides a way to quickly get to various parts of the application. Figure 4.14 illustrates the menu drawer.

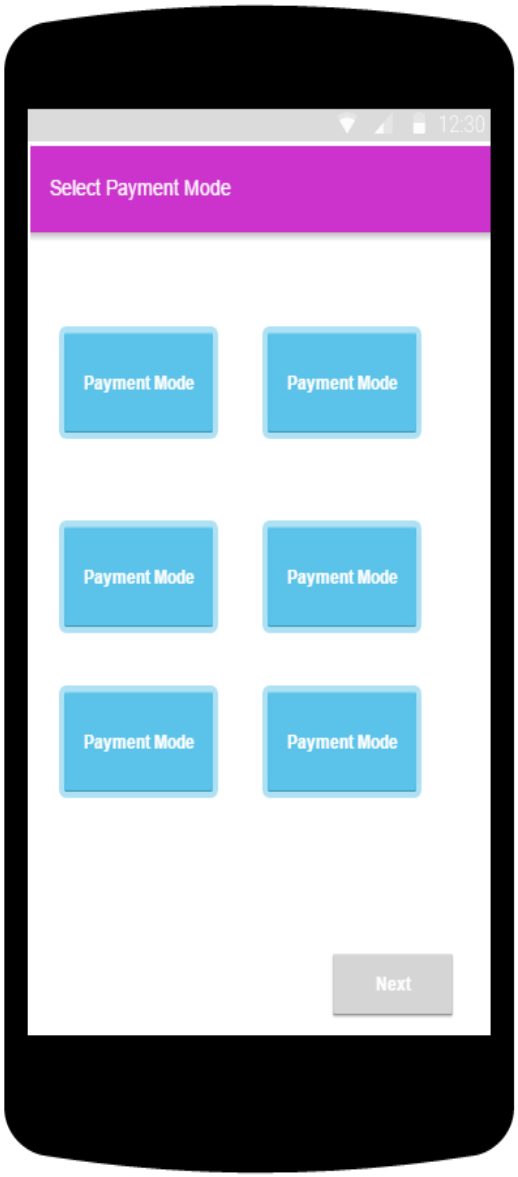


Figure 4.10: Stream Selection Screens

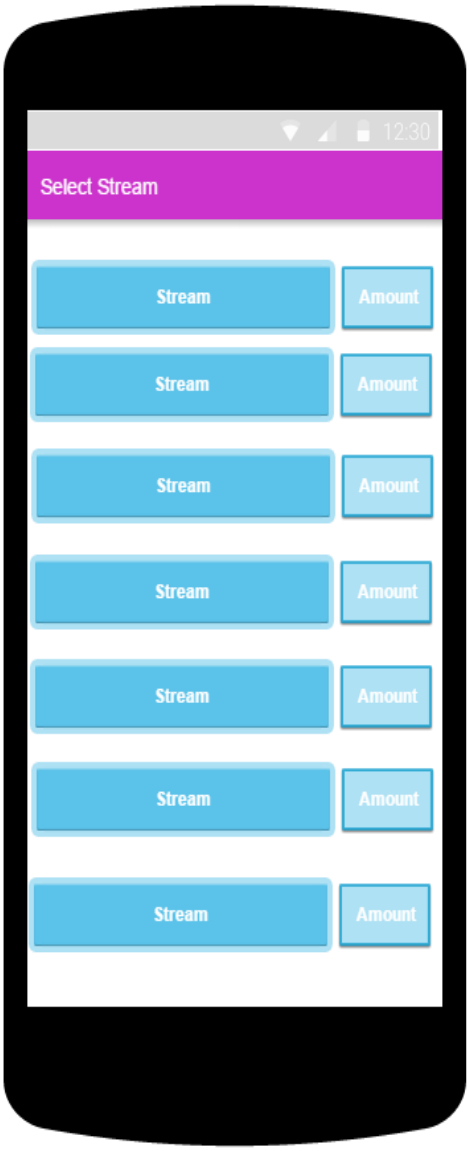


Figure 4.11: Stream Selection Screen

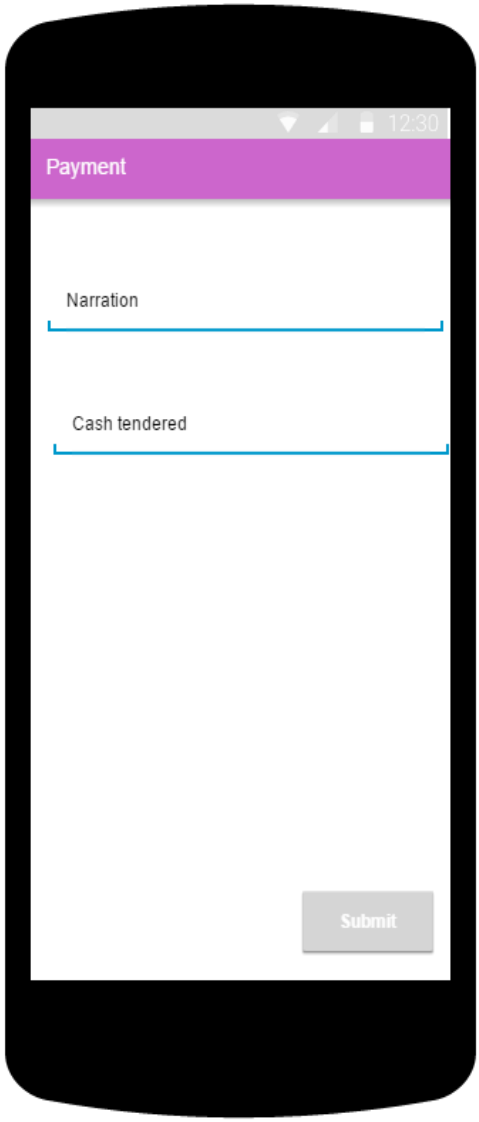


Figure 4.12: Payment Details Collection Screen

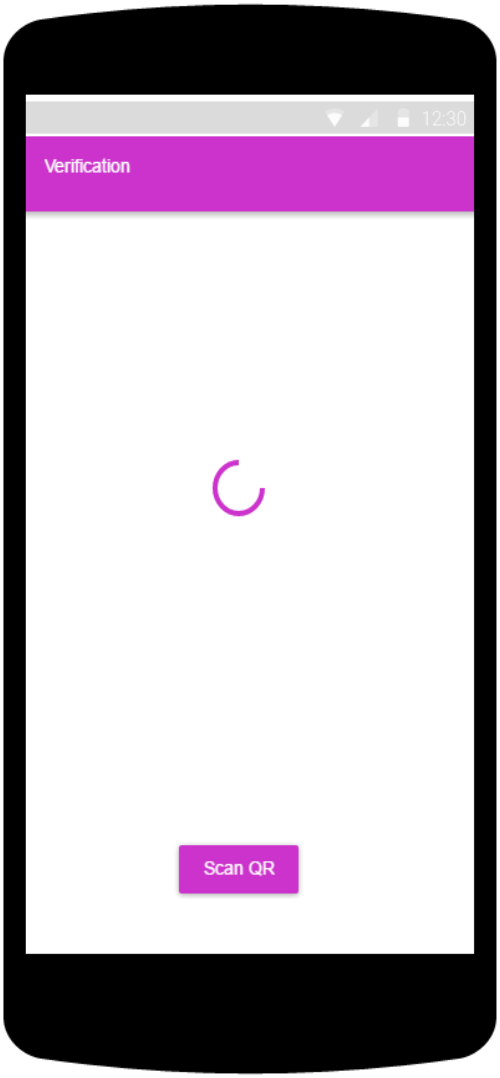


Figure 4.13: Receipt Verification Screen

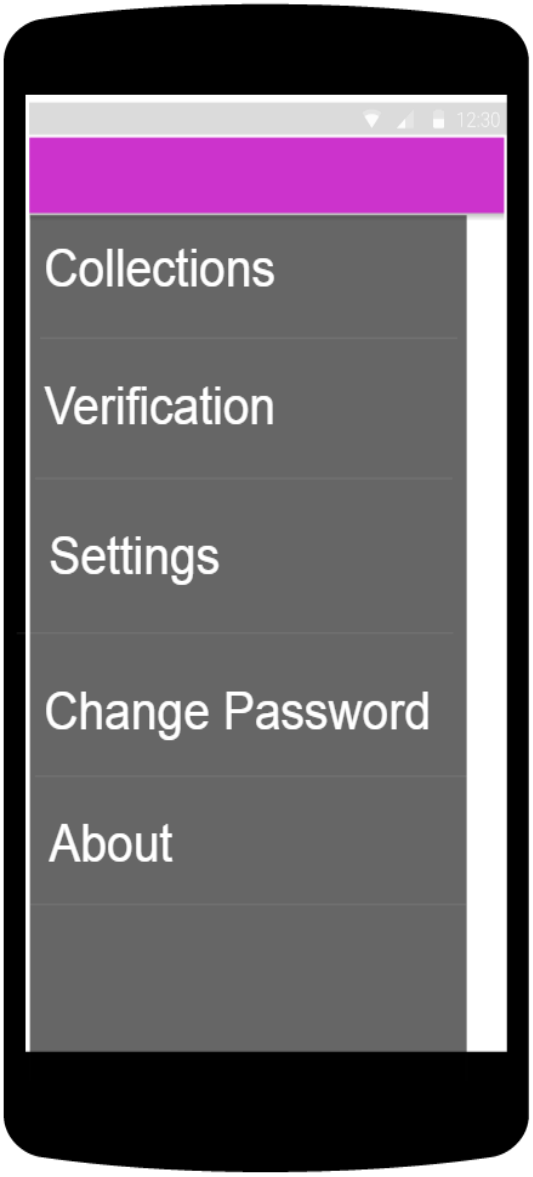


Figure 4.14: Application Menu Screen

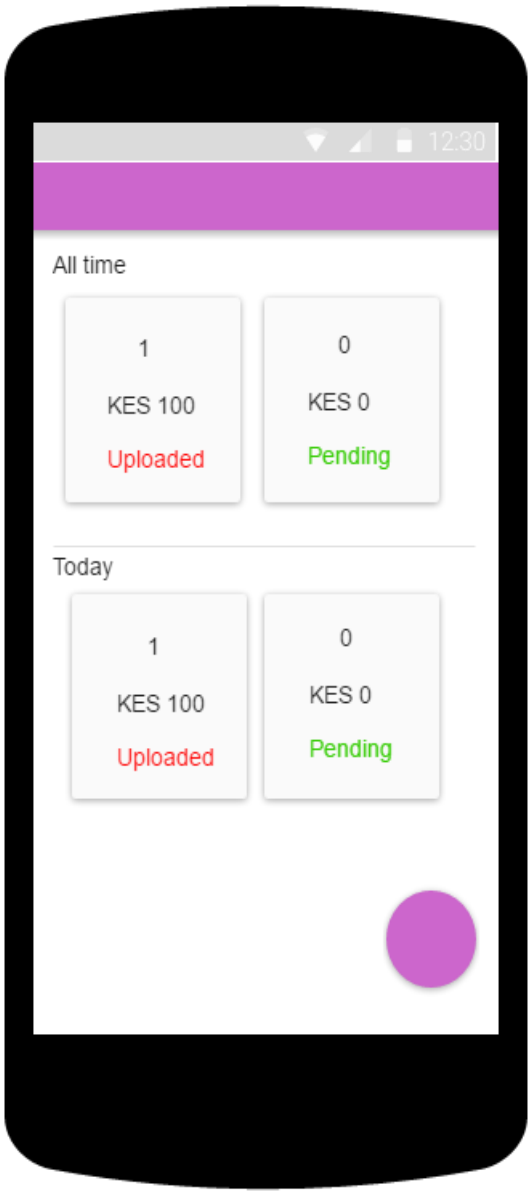


Figure 4.15: Application Dashboard Screen

Chapter 5 : System Implementation and Testing

5.1 Introduction

The systems developed targets to enable offline collection of revenue and verification of receipts in integrated point of sale terminals. It consists of two main applications with the main one running on the Android platform and supported by a web based application. The Android application was strictly developed in accordance to the design specifications in order to meet the functional requirements.

5.2 Programming Tools Details

This section outlines the tools that were used in the development phase of the system. It provides an overview of the software development practices that were followed. The section also provides a description of the technologies that helped in the development of the system.

5.2.1 Database

The web based database was implemented using the PostgreSQL database application server. This is mainly because it provides better query processing performance as compared to other open source alternatives such as MySQL. Primary keys were uniquely identified and generated using autoincrement sequences natively supported by PostgreSQL. All linked tables were also constrained using foreign key indexes. To enhance data integrity, fields such as receipt numbers that need to be unique also have database constraints to ensure uniqueness.

5.2.2 Mobile Application

The mobile application was implemented using the native Android development kit. It provides an interface that enables collectors to receive payment. The application offers capability that enable printing of receipts. Receipt generation is automatically supported when the user is using a device with an inbuilt thermal printer. However, in cases where the user is using a mobile phone the application also has support for receipt printing on a Bluetooth printer. The most important aspect of the mobile application is that it is able to perform the two main functions of receipting and receipt verification when it is offline. The application therefore has an internal storage engine developed using a SQLite database.

Data synchronisation in the mobile application is achieved through synch manager that automatically detects when the device is offline to synchronise all transactions that have not been

committed to the central database. Users also have the ability to set application preferences on how often they would like the device to attempt synchronisation with the web application.

5.2.3 Web Application

The web application is implemented using Java. It follows object oriented design patterns to structure the low-level functions. The application is designed to be accessed on any web browser. The web application runs on a tomcat server. The application design also follows the MVC pattern to enable better code reuse. The back end is also composed of a restful web service that interacts with the mobile application. The web service uses JSON data format to enable communication with third parties.

5.3 Implementation Details

This section explains details of how various system processes were developed to enable the application meet functional requirements. The section provides an overview of concepts that were followed during the development phase and the thinking behind the system's implementation details. Table 5.1 illustrates technical components of the proposed system:

Table 5.1: Technical System Components

Application server for the backend	Apache Tomcat
Database server	PostgreSQL
Hosting Environment	Google Cloud
Point of Sale Terminals Operating System	Android
Server Operating system	Debian
Language and framework	JDK 1.8, J2EE, Spring Framework, Groovy, JPA
Reporting component	Jasper Report

5.3.1 Cryptographic Offline Verification

The system achieves the main objective of enabling offline verification of receipts by using cryptographic tokens. This is achieved using a cryptographic algorithm that takes important fields of a transaction which are the date, narration and amount collected to generate a receipt number upon receipt of payment. The algorithm ensures that the generated receipt numbers do not follow a generated sequence but instead are a random combination of unreadable text. All linked point of sale terminals have a shared key that is able to decipher the receipt number generated by other point of sale terminals.

It was essential to ensure that the generated receipt number tokens have a unique identifying token and that the identifier was human friendly in terms of length and character encoding. To make this possible, the research used Base 32 encoding specified in RFC 4648. The encoding would ensure that; the overall length of the token is reduced since it applies compact encoding and also avoid similar looking characters therefore minimising transcription errors. This is for example achieved by avoiding the use of the number one due its similarity with the letter I as well as Zero due it its resemblance to the letter O. The table below shows the base 32 character set that was employed.

A	B	C	D
E	F	G	H
I	J	K	L
M	N	O	P
Q	R	S	T
U	V	W	X
Y	Z	2	3
4	5	6	7
=			

The application generates Base 32 identifiers using the following algorithm.

i. The number of IDs to last the application accounting cycle is estimated using an 8 bytes long integer which was selected as 2^64. This served as the database key column.
ii. A secure random generator is used to ensure the receipt numbers are unpredictable in this case the java class java.security.SecureRandom.
iii. Upon receipt generation, the ID is run through a base 32 encoder.

The following are the achieved results that were achieved by using base 32 encoding over the standard UUID:

UUID (Standard format): 38503690-0475-4c48-95ed-a3c9eaa2ac3a

UUID (Base 32): HBIDNEAEOVGERFPNUPE6VIVMHI

The application uses Digital signatures as the preferred encryption algorithm. This is because digital signatures provide Integrity, Authentication and non-repudiation which meet the required specification for the encryption algorithm needed to generate a unique identifiable receipt number token. Digital signatures were preferred over Hash-based Message Authentication Code (HMAC) because they offer non-repudiation which was important in the receipting processes to ensure that collectors take responsibility for all transactions.

The research settled on use of RSA over elliptic curve algorithm because RSA offers a broader support in terms of programming libraries available. RSA256 was preferred because it is based on RSA PKCS # 1 which is the most widely used private/public key cryptographic technique. The signature also requires minimal CPU time to perform a verification check and would therefore be most appropriate for mobile devices.

Validation is achieved by a counter algorithm that is able to decipher the cyphertext using a secret key. If it is not able to decipher the receipt number using the agreed key then that receipt number will be flagged as invalid. The validity of the payment is confirmed after the receipt number has been decrypted to obtain the date of payment, amount and narration details of the transaction. Since the algorithm generates long receipt numbers that may be quite difficult to

manual input into a search field, the token is represented using a QR code that is scanned during the verification stage. The verification engine then attempts to use the secret key to obtain transaction details and displays corresponding results.

5.3.2 Data Synchronization

Data synchronization in the system is achieved by maintaining two databases. A central database on a server that is responsible for holding all charge configurations and client databases that hold temporary payment data. One a collector logs in to the application, a background service a sends a HTTP data check request with the current timestamp as a request parameter. If there is need for synchronization the web service will return a JSON array containing data dictionaries that will then be parsed by the application to generate database insertion queries to save the newly downloaded data dictionaries.

Upon collection, the collection data is saved on a SQLite database on the device and marked with a flag to show that it has not been synchronized. The application contains a sync adapter that checks whether the internet connection is available. If the internet connection is determined to be available the application will select all data records in the SQLite database that have not synchronized and send a corresponding HTTP request to the server with the payment details as the request body. The server then saves all the corresponding transactions and returns a success message.

5.4 Web Application Components

This section provides the main system modules that make up the web application. It provides implementation details of the interfaces. The section outlines the various components that enable system users to perform operations on a web browser.

5.4.1 Authentication

Access to the main backend is controlled by challenging the user for a username and password by displaying a login page. Figure 5.1 shows the login screen that is presented to the user upon loading the main page. If the user supplies the wrong username and password they are informed of the authentication failure. Every log in attempt it recorded and users have to get a password reset from the system administrator once they exceed five log in attempts. Upon successful authentication users are presented with a session key that expires upon ten minutes of

inactivity. The session key however has a life ling expiration period of six hours and users will be automatically logged out if a request to regenerate the session key is not requested.

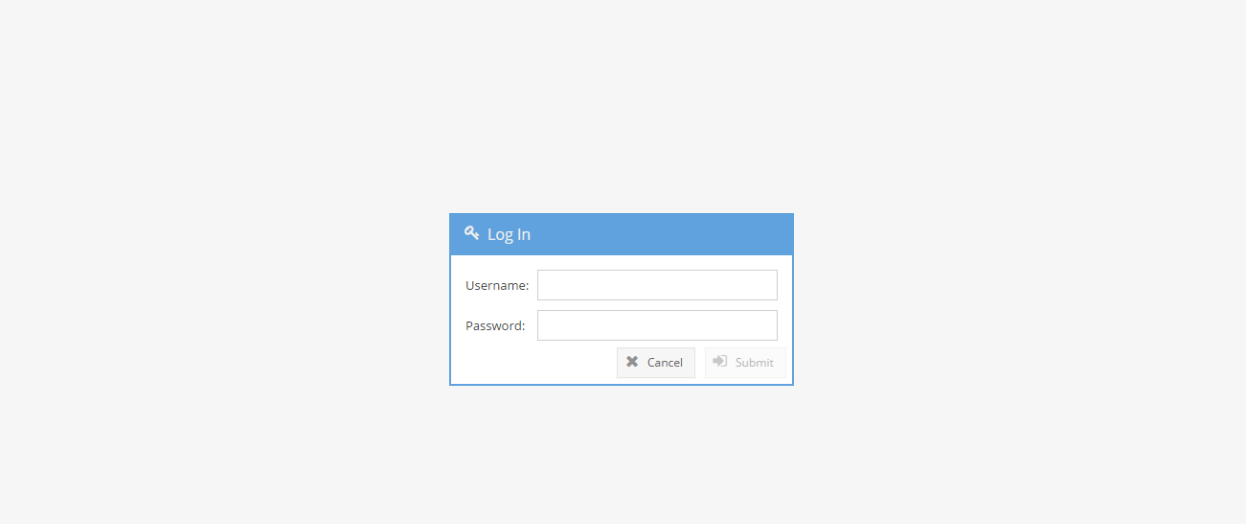


Figure 5.1: Web Application Login Page

If a user is logged in for the first time after creation of the count they will automatically be prompted to change the password after being logged in. Password also expire after every thirty days and users are therefore usually prompted to reset the password after every expiration cycle before they are able to gain access to the system. The system also keeps track of previous password to ensure that they do not recycle old passwords. The password change process also evaluates the quality of the supplied password before a user is able to set a password and only accepts passwords that are considered strong. Figure 5.2 shows the password change screen.

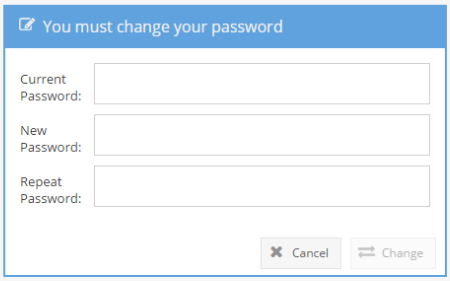


Figure 5.2: Change Password Screen

During password resets users are also prompted to retype the selected password before they are able to proceed with the reset process. They are also asked to private the previous password to prevent substantial account damage in case of session hijacking which is handled through the use of authentication tokens. Figure 5.3 shows how messages are displayed on the change password form to flag any errors.

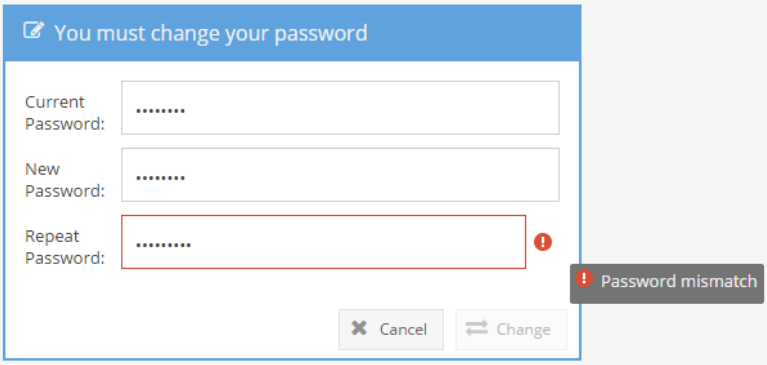


Figure 5.3: Change Password Mismatch Messages

5.4.2 Role Selection

After successful authorisation, the user is prompted to select a role. Only roles assigned to a particular user are displayed at this stage. The system supports multi role based user allocation where one user can be assigned more than one role. Users are redirected to the appropriate menu controls upon successful authentication. Figure 5.4 shows the role selection screen that is presented to the user after they log in.

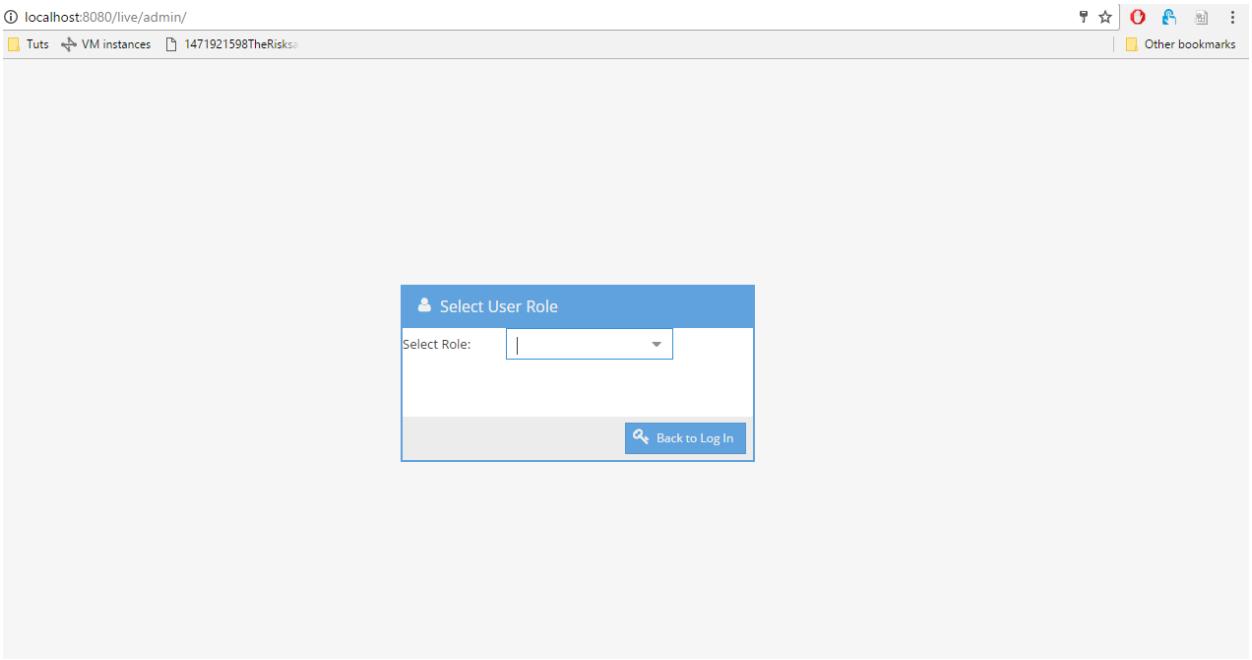


Figure 5.4: Role Selection Screen

5.4.3 System Administration

This section outlines the various features that system administrators have access to. It outlines the user interface panels that enables administrators to perform system configurations and other management tasks.

5.4.3.1 Location Management

This module enables administrators to manage locations of revenue collection based on geographical boundaries. Since the application use case was based on county governments, it enables configuration and management of sub counties and wards. Figure 5.5 shows the grid that enables system administrators to configure sub counties. A detail view of the location management panel displays a ward configuration module as shown in Figure 5.6.

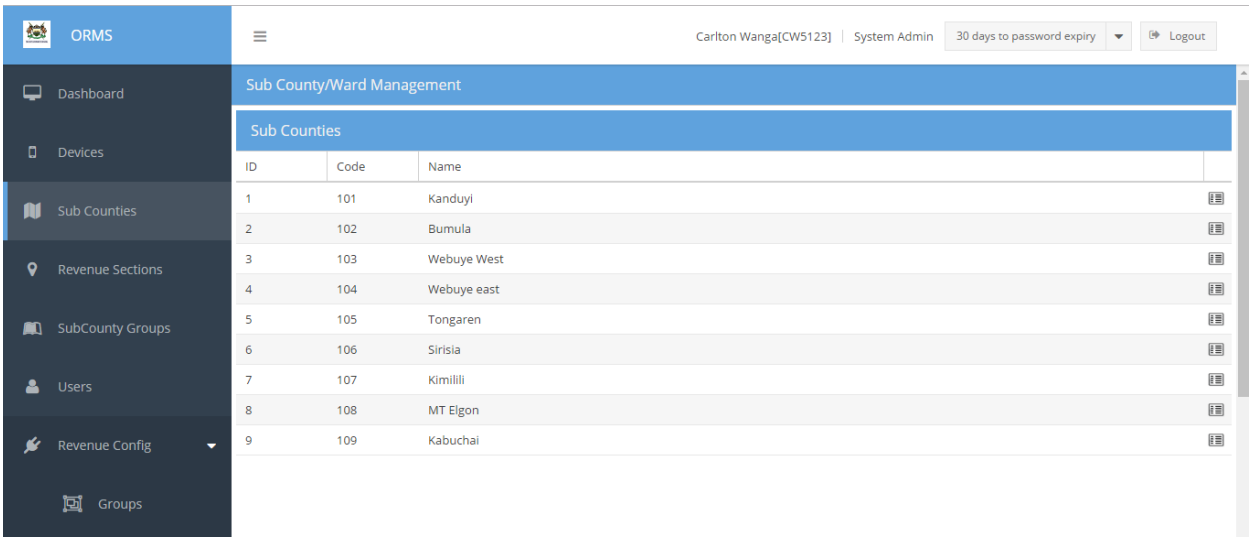


Figure 5.5: Location Management

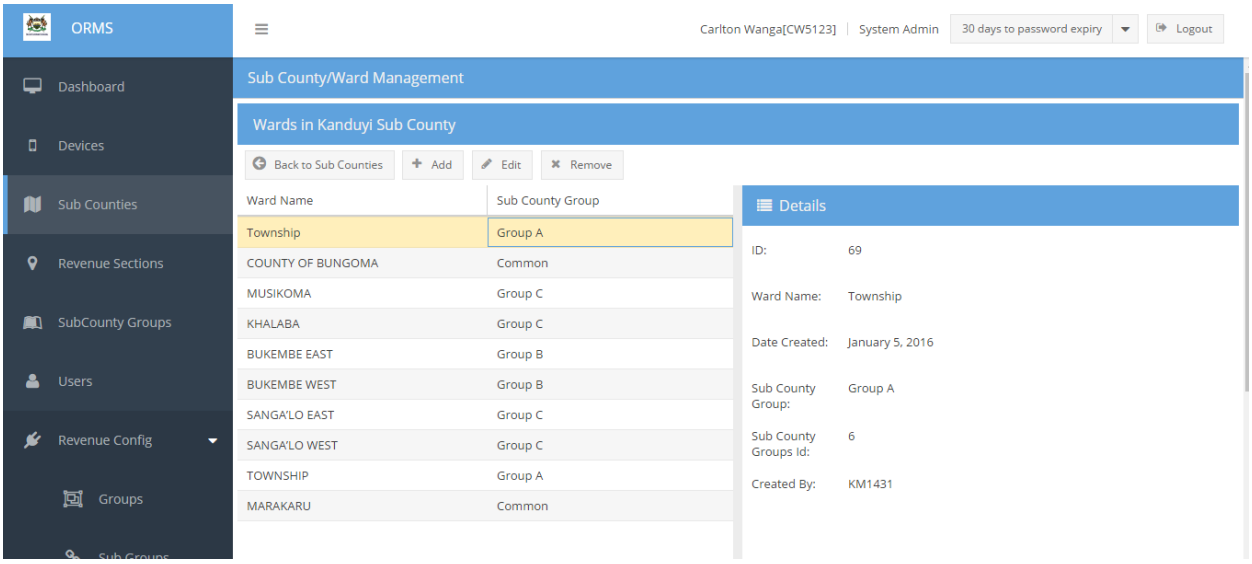


Figure 5.6: Ward Management Panel

5.4.3.2 Device Management

The system offers administrators a device management panel where they are able to configure devices so that they are easily identifiable. Figure 5.7 shows the device management panel that enables administrators to create, edit and deactivate devices.

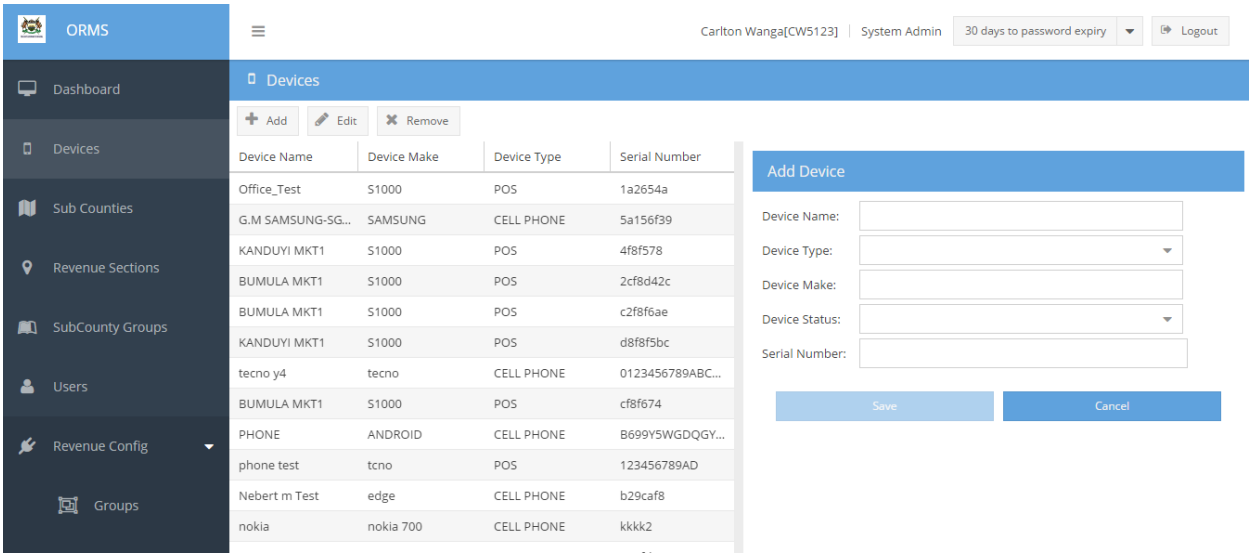


Figure 5.7: Device Management Panel

5.4.3.3 Revenue Sections

For finer classification of revenue, the system uses revenue sections to identify a virtual location where collectors are assigned to. A revenue section is given a projected target to enable setting of collector targets based on the section they are allocated to. Figure 5.8 shows how revenue sections are managed by administrators.

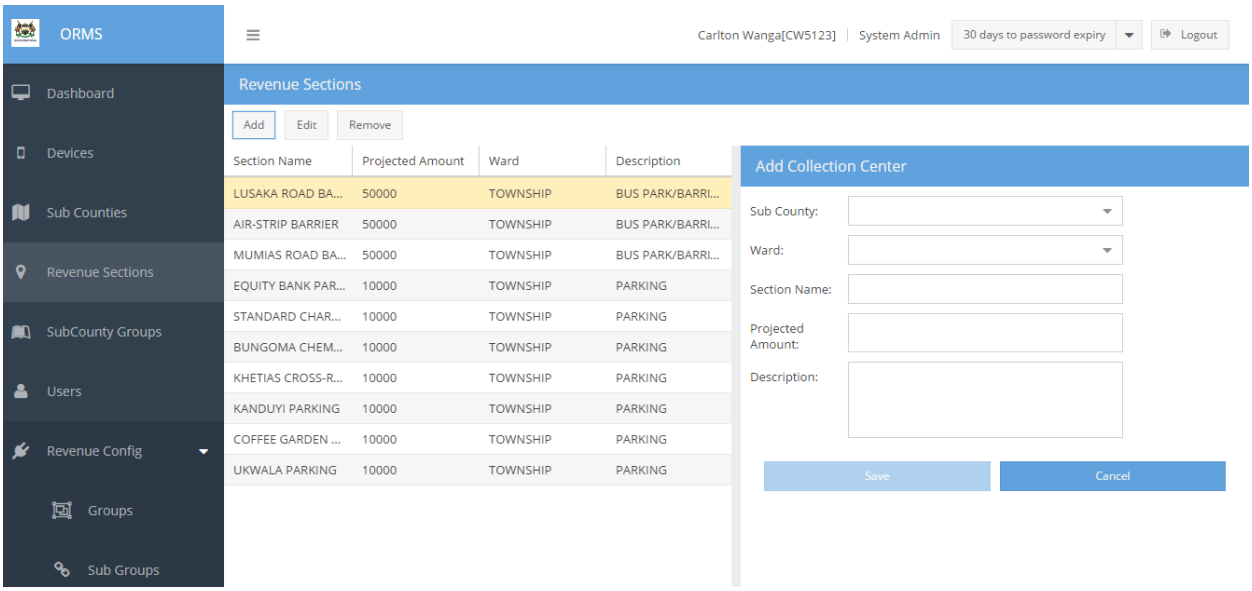


Figure 5.8: Revenue section Management Panel

5.4.3.4 Sub County Groups

To enable easy classification of revenue charges on areas that do not have any geographical boundary connection but share common charge properties the application uses revenue groups as shown in Figure 5.9.

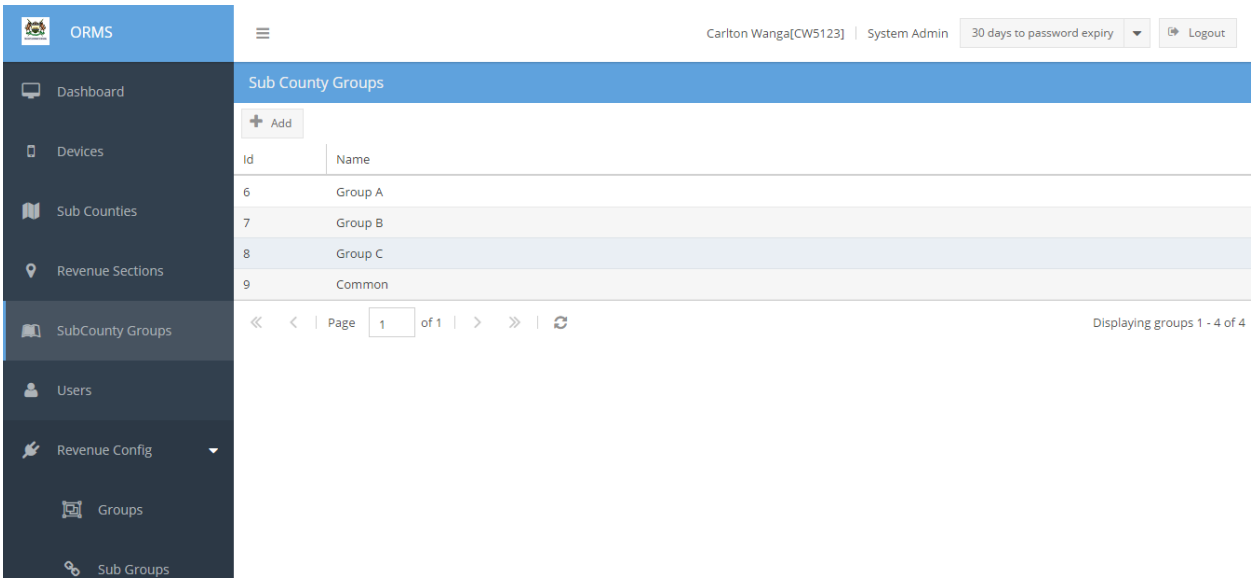


Figure 5.9: Sub County Group Management Panel

5.4.3.5 Revenue Configuration

Revenue Group

Figure 5.10 shows the configuration table that enables system administrators to manage revenue group data. The interface allows them to add new data dictionaries as well us perform editing operations on the data grid.

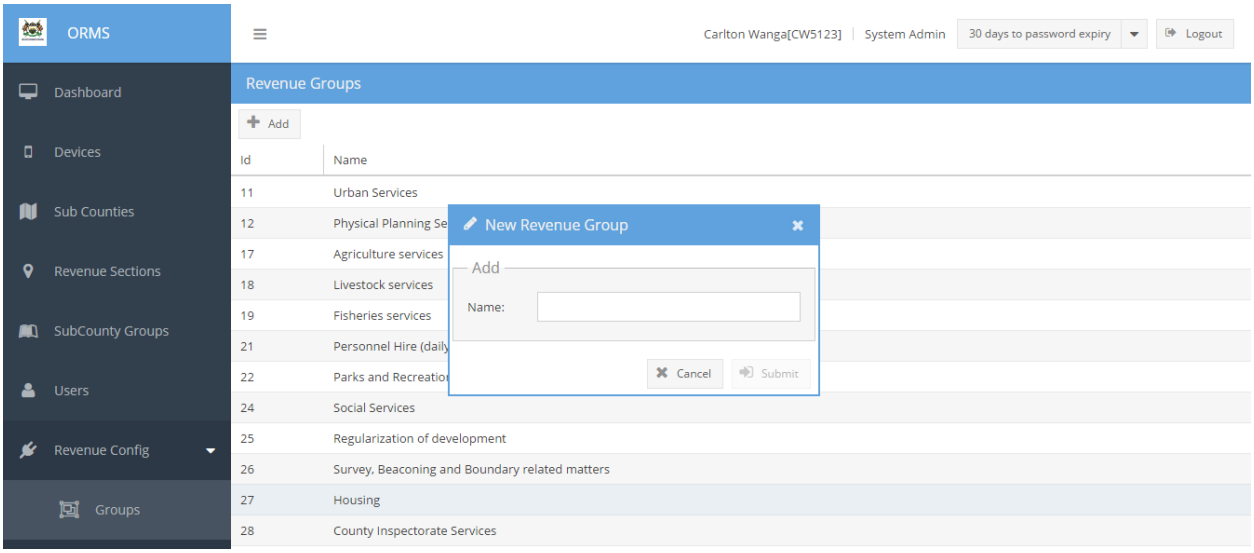


Figure 5.10: Revenue Group Management Panel

Revenue Subgroups Management

The revenue subgroup module offers an interface to enable definitions of revenue sub groups and link them to corresponding groups as shown in figure 5.11.

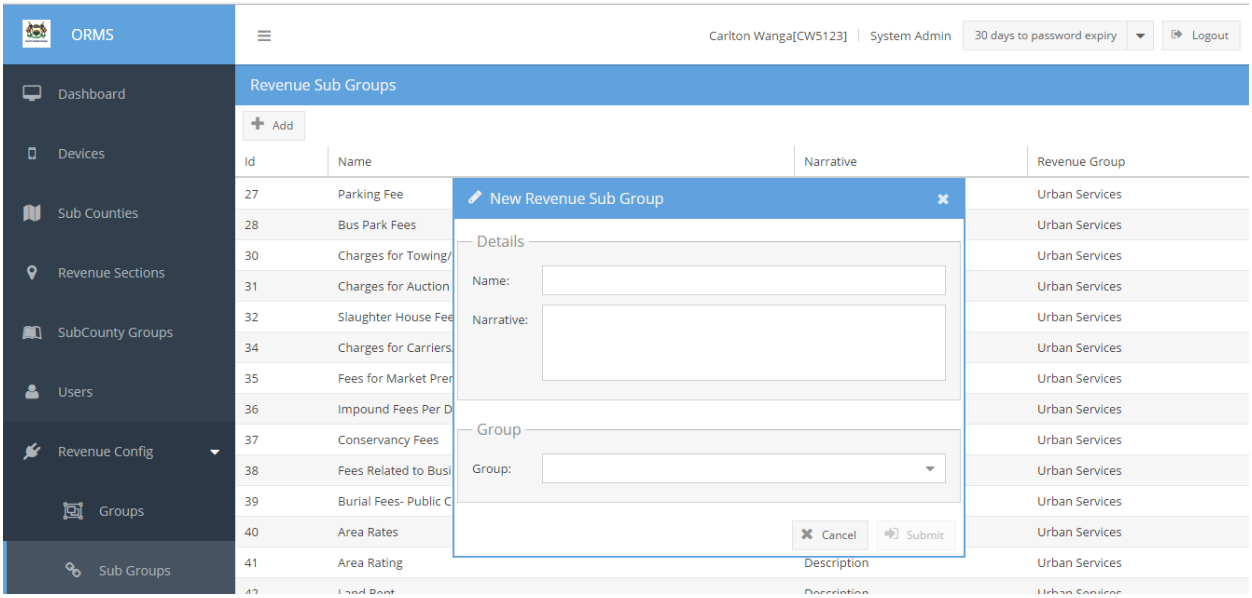


Figure 5.11: Revenue Subgroup Management Panel

Revenue Stream

Figure 5.12 shows the web interface that enables system administrators to manage revenue streams.

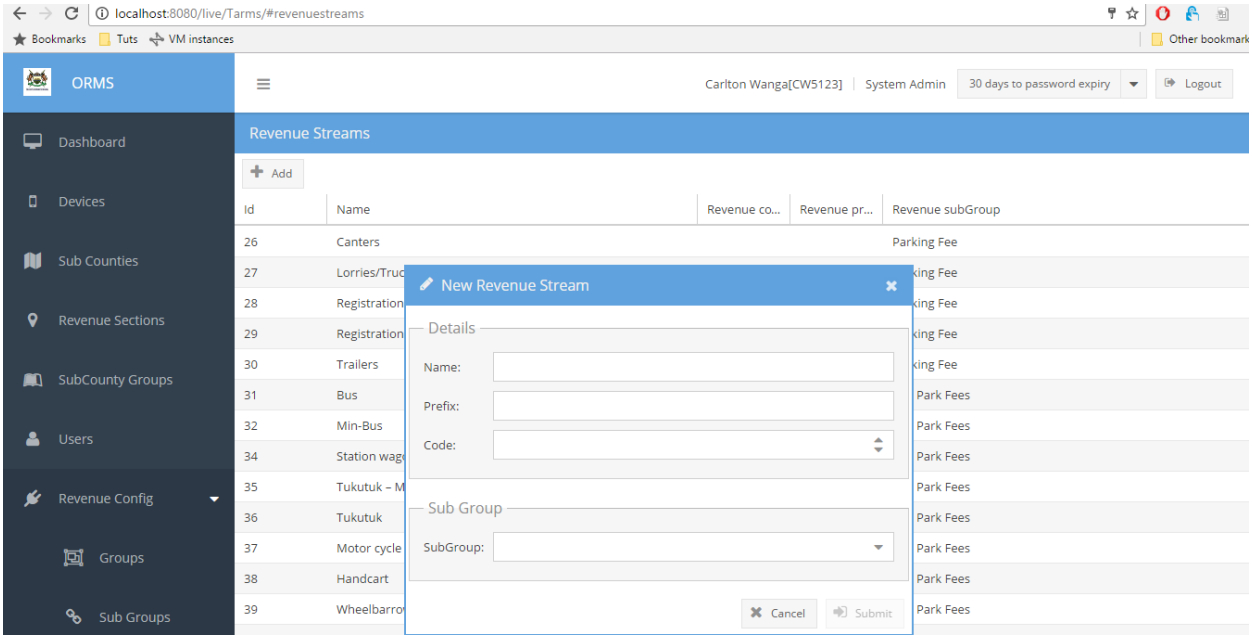


Figure 5.12: Revenue Sub Groups Management Panel

Revenue units

Figure 5.13 shows the data grid that allows system administrators to managed units that enable quantifying of received payments.

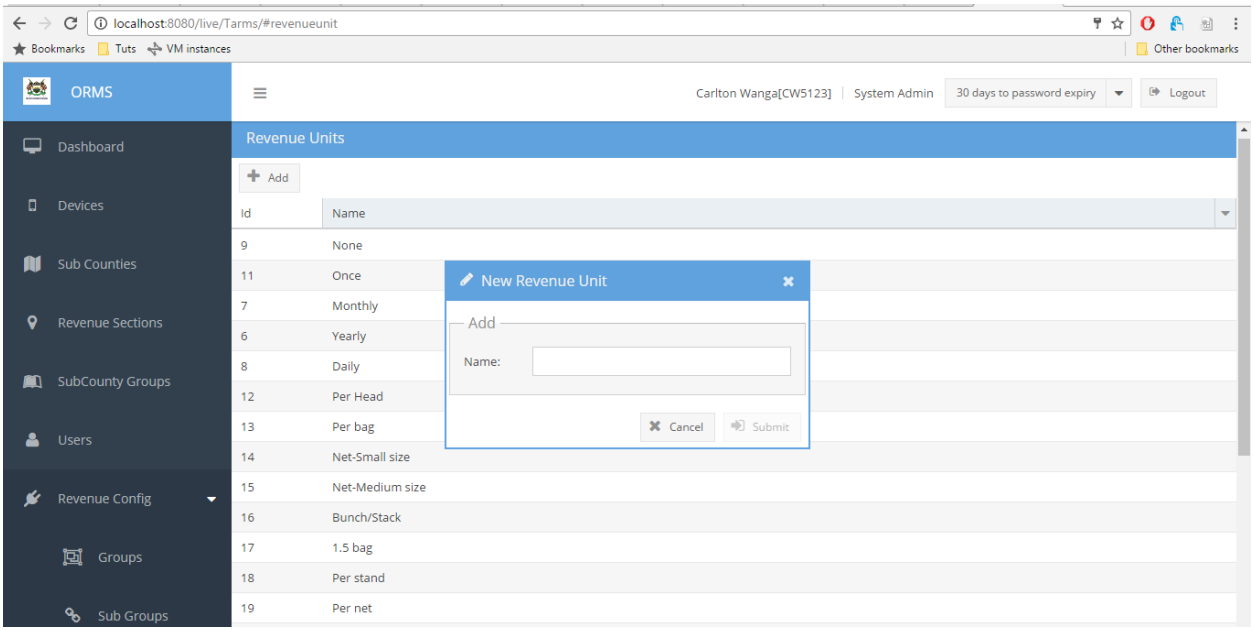


Figure 5.13: Revenue Units Management Panel

Revenue Charges

Figures 5.14 and 5.15 show the interfaces that enable configuration of revenue charges

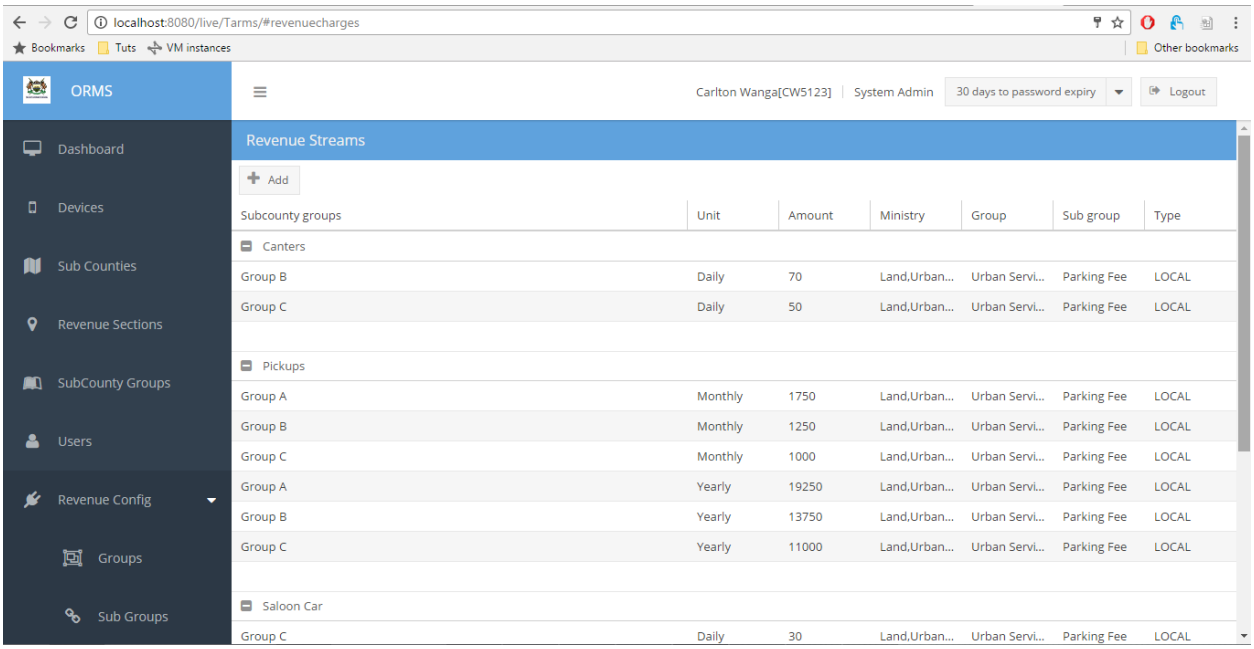


Figure 5.14: Revenue Charges Management Grid

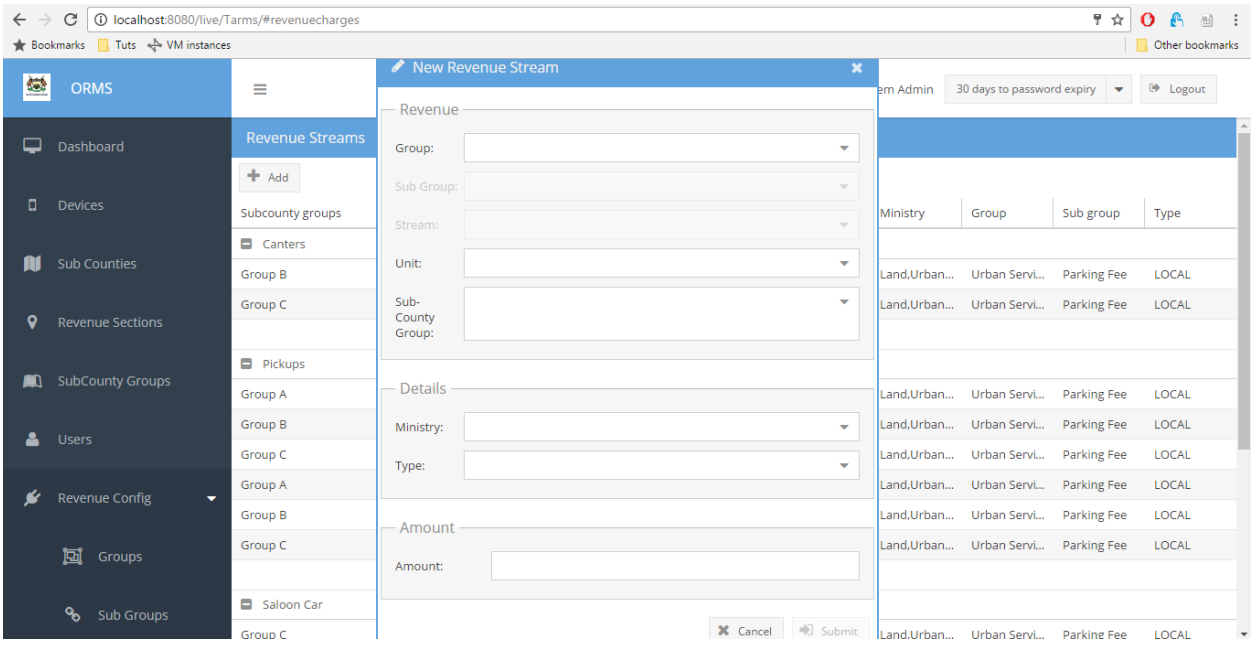


Figure 5.15: Add Revenue Charge Panel

User Management

The user management panel provides the all user related functions. It also serves as a data grid for displaying all user data as well provides searching capabilities that make it users to find users. Once a user is identified and selected all their corresponding details are displayed on the details panel as shown in Figure 5.16.

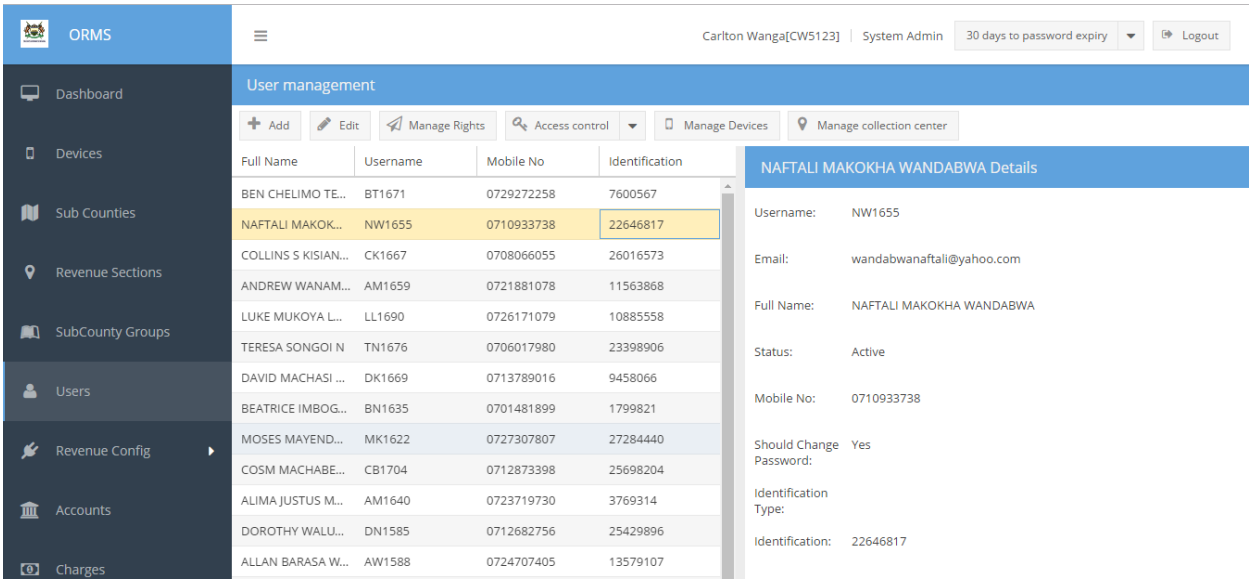


Figure 5.16: User Management Panel

User management involves creation of user accounts where the administrator adds new users to the system. The process involves an administrator supplying identification details as shown in Figure 5.17. Once an administrator has submitted the details, the system automatically generates a corresponding username and auto generated password. The user will be prompted to change the password upon the first log in attempt.

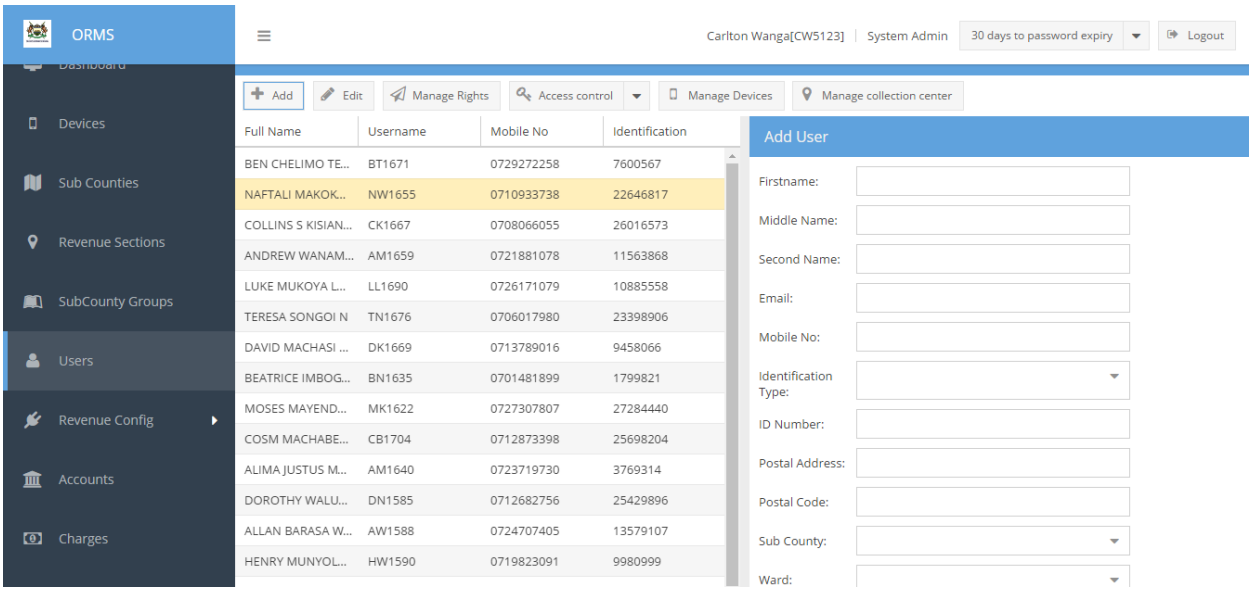


Figure 5.17: Add User Panel

Creation of user accounts is followed by the process of role assignment. Users need to be assigned a role before they can have access to the system. The system also allows users to have more than one user role. Administrators therefore have access to allocate and deallocate user roles as shown in Figure 5.18.

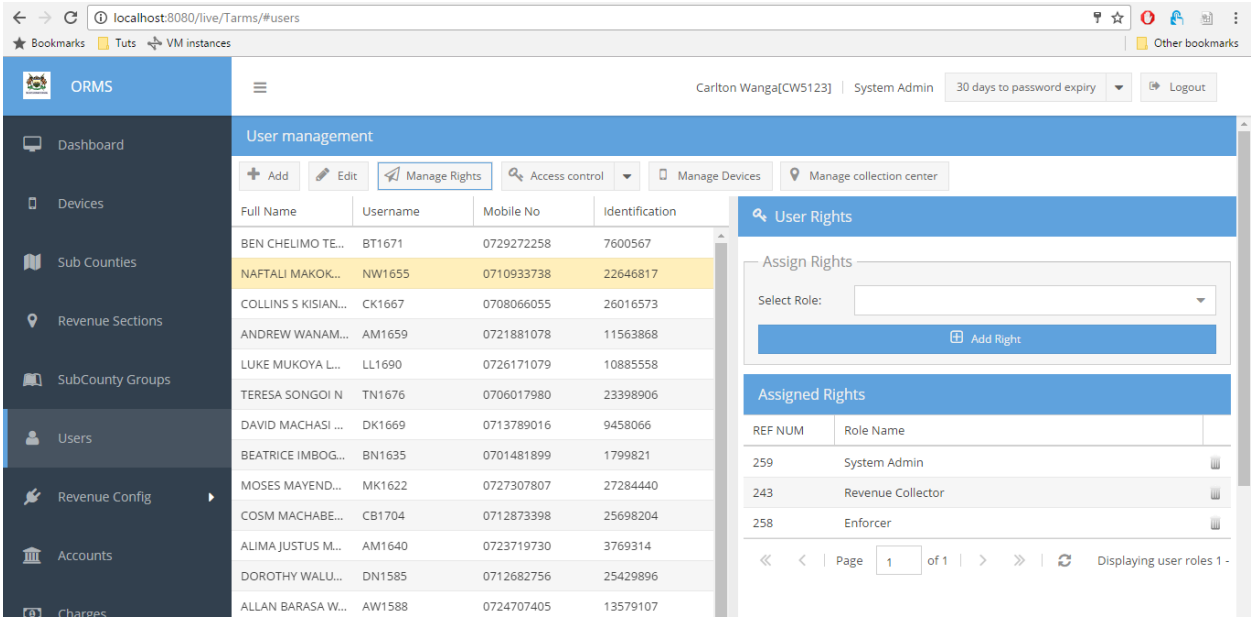


Figure 5.18: Role Assignment Panel

When a user is allocated revenue collection rights they need to be allocated a device before they can collect. The device's serial number is used as an authentication factor when a collector attempts to log in to a point of sale terminal. Figure 5.19 shows the device allocation panel.

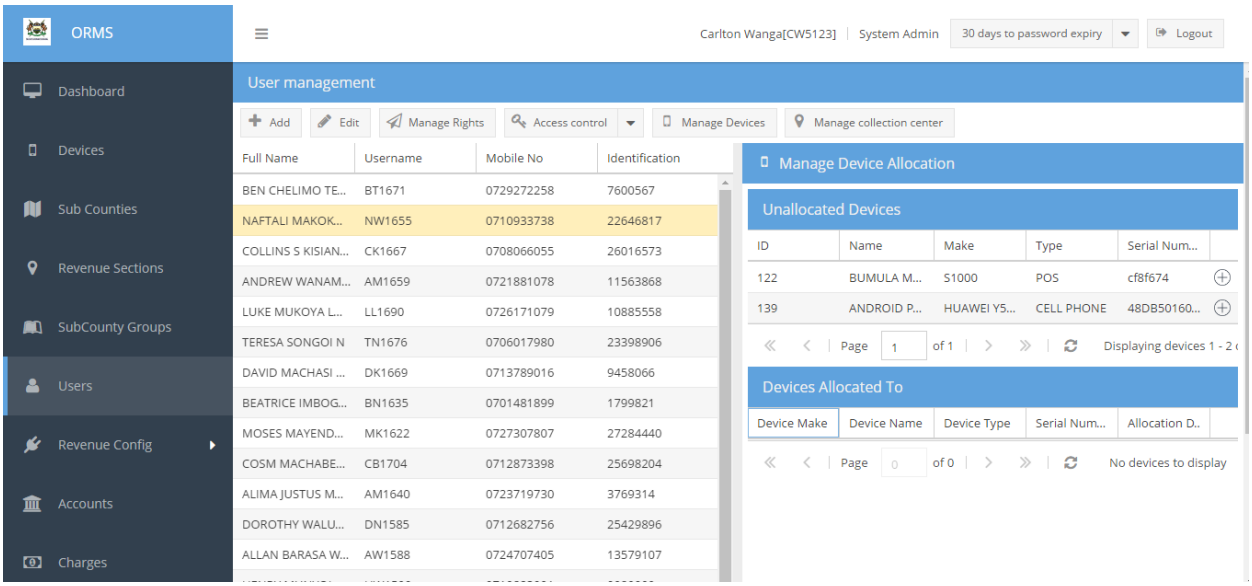


Figure 5.19: Device Allocation Panel

After a user is allocated a device, they need to be allocated a revenue section. The revenue section determines the projected amount that the user is expected to collect. Figure 5.20 shows the device allocation panel that is displayed when the administrator allocates a revenue section to a collector.

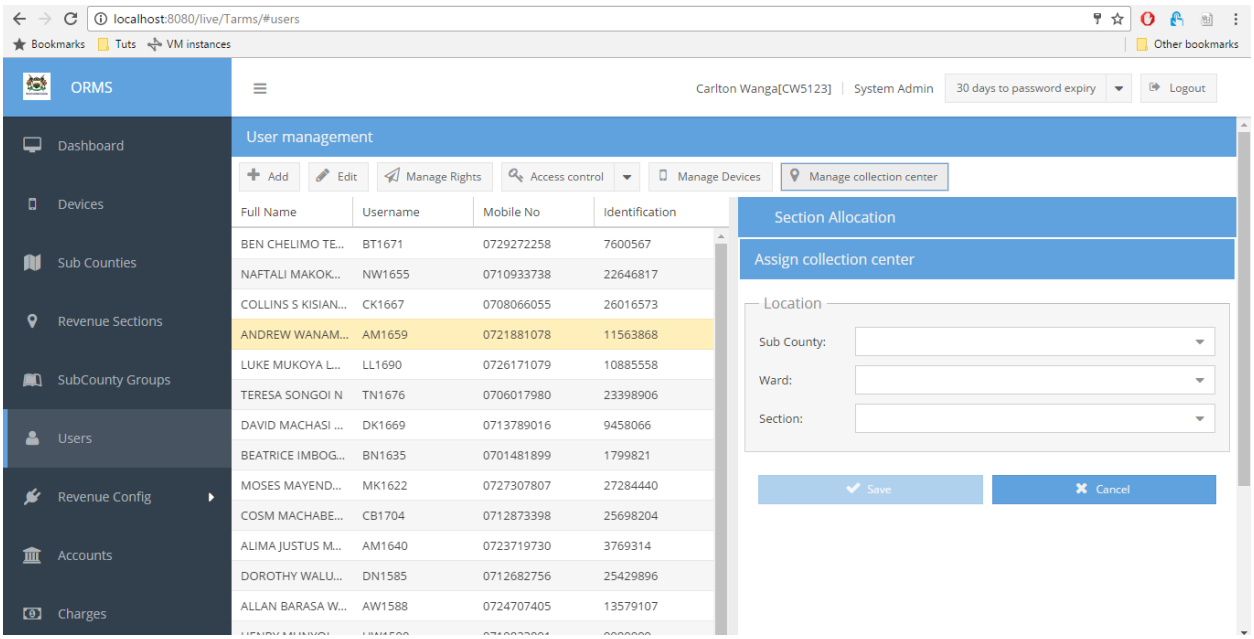


Figure 5.20: Collection Section Allocation

5.5 Point of Sale Terminal

This section outlines components of the mobile application. It gives screenshots of the interface that will be displayed on the point of sale terminals. The section further provides descriptions of the application processes. It follows a step by step description of the receipting and receipt verification processes.

5.5.1 User Authentication

All collectors using the point of sale terminals need to be authenticated before they are granted access to the devices. Authentication begins with checking the username and password. After a correct username and password is supplied the system validate whether that collector has actually been assigned to that device.

5.5.2 Home screen

After authentication users are presented with a home screen showing a dashboard with a summary of number of transactions performed by the current logged in user detailing those that are have not synched as well as the transactions that have been committed to the main backend as shown in Figure 5.21.

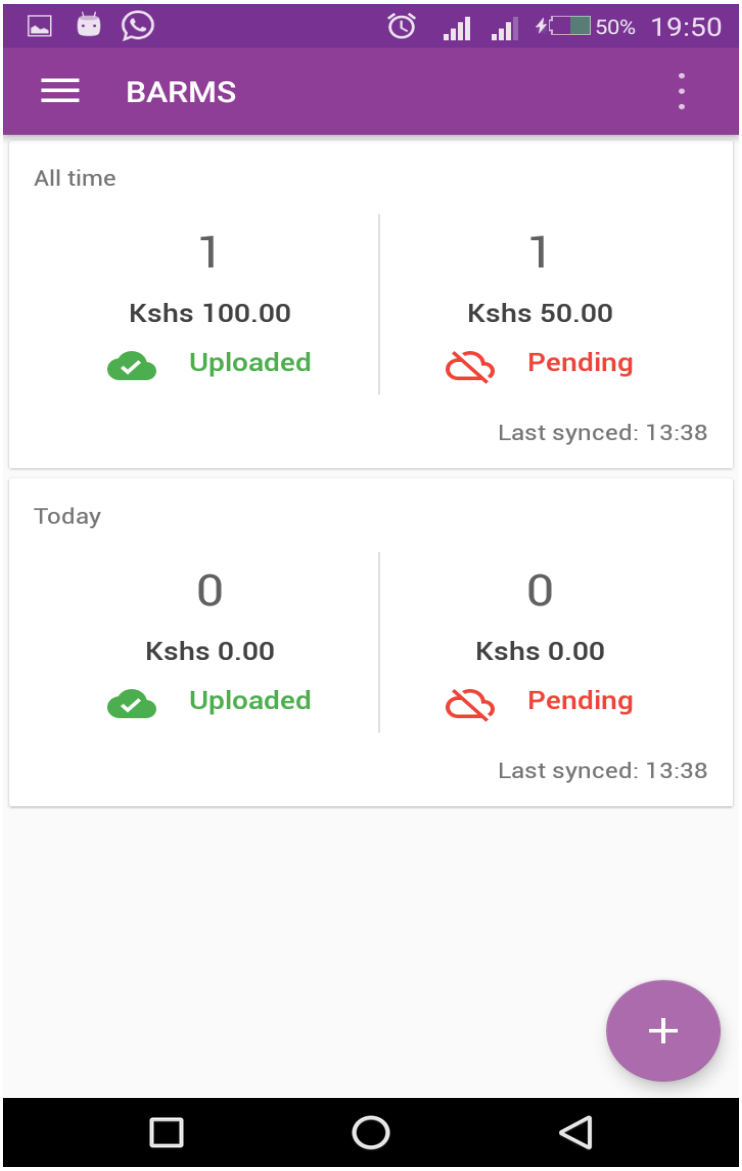


Figure 5.21: Mobile Application Dashboard

The Android application provides a navigation menu that is housed by a sliding drawer. The menu contains quick access options that enable collectors to navigate main application controls. The first option labeled as collections allows them to go to the home screen to enable quick access to the dashboard.The menu also offers options for navigation to the verification panel, settings and passoword change options as shown in Figure 5.22.

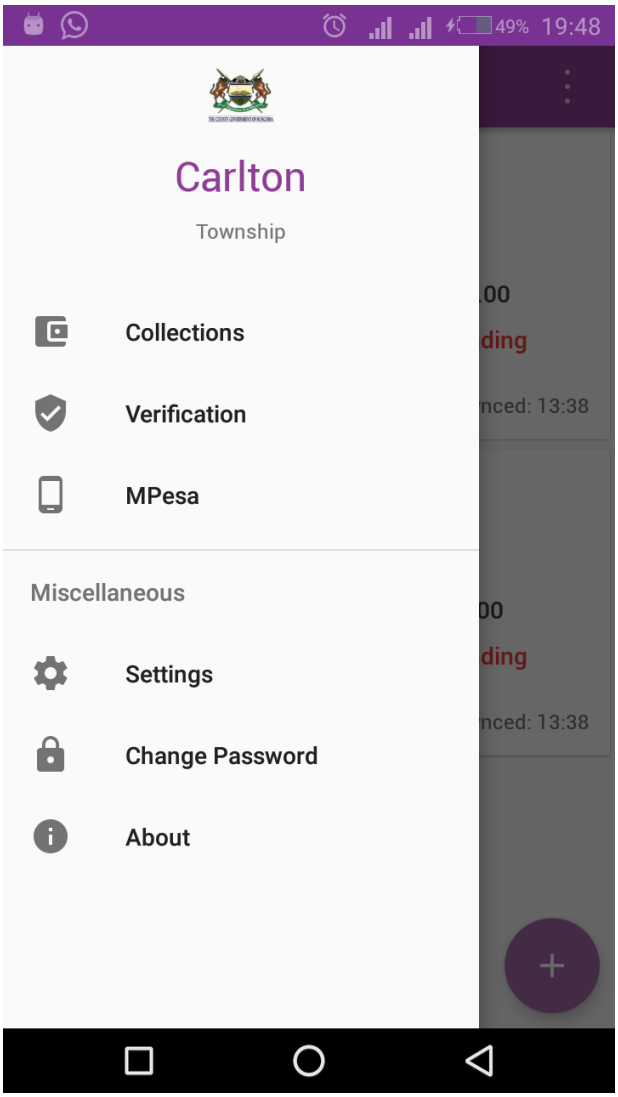


Figure 5.22: Application Navigation Menu

Collection of revenue begins with the user selecting the plus action button on the home screen. The user is then direct to filter based on a list of subgroups as shown in Figure 5.24. Thereafter the system filters a list of revenue streams in that subgroup as shown in Figure 5.25.

To make collection easier, the system keeps track of the most used collection streams collectors are presented with an option of accessing revenue streams through a history tab. The history tab lists charges based on the most frequently used as illustrated in 5.23.

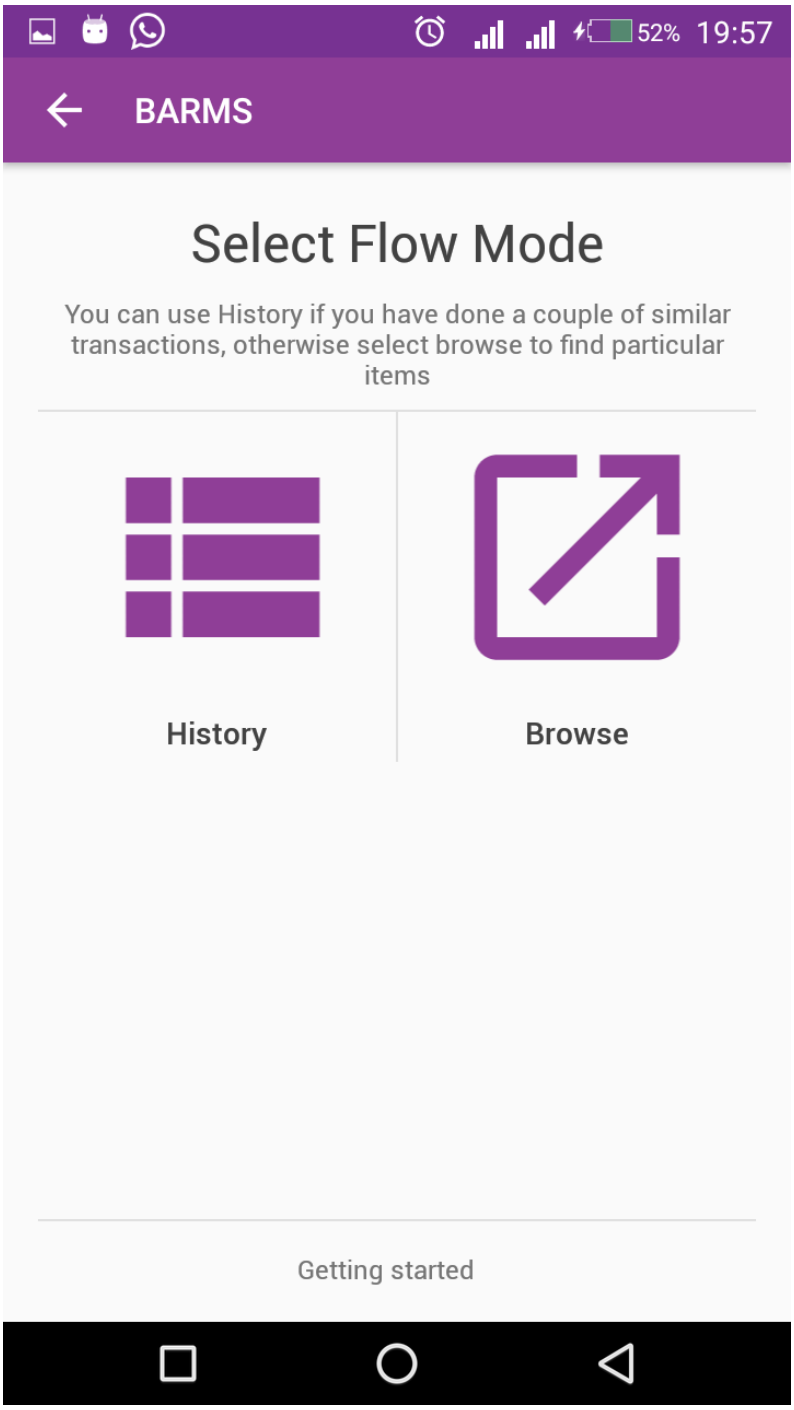


Figure 5.23: Flow Selection Screen

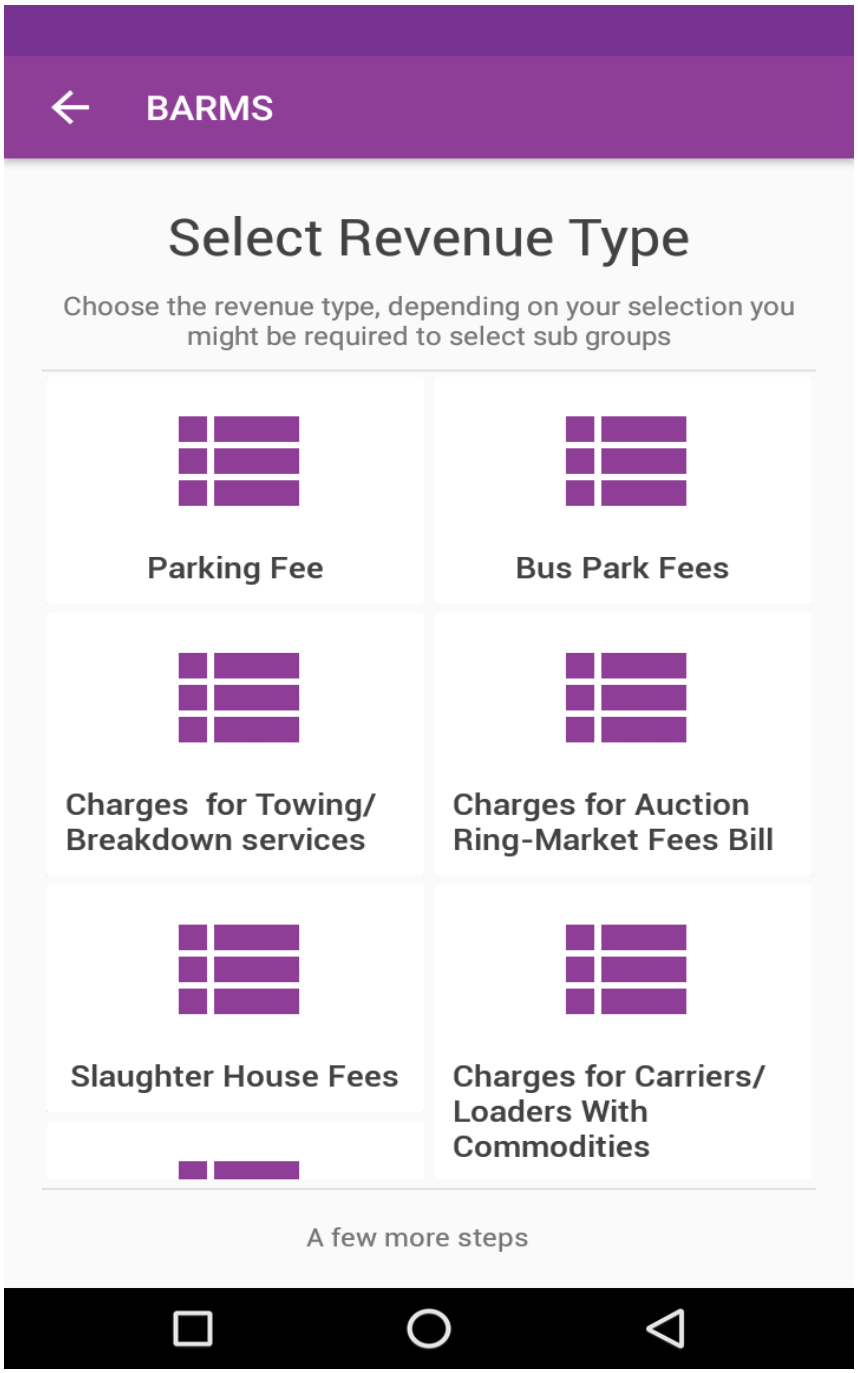


Figure 5.24: Revenue Type Selection

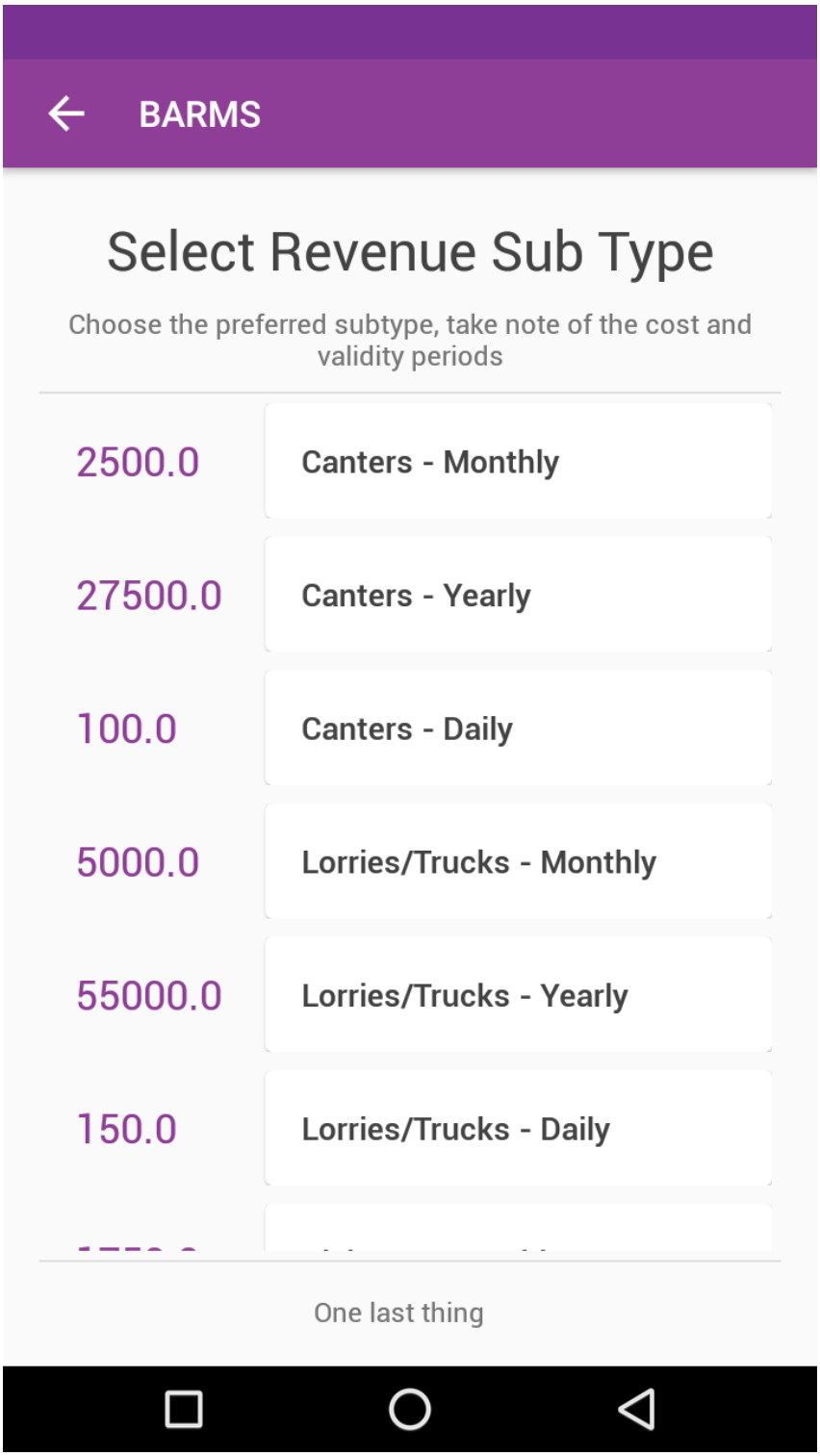


Figure 5.25: Revenue Sub Type Selection

When a collector has correctly selected, a revenue stream the application will prompt them to provide a narration of the transaction as shown in Figure 5.26. This will serve as a customer identification field for example a vehicle number plate in cases of parking fees.

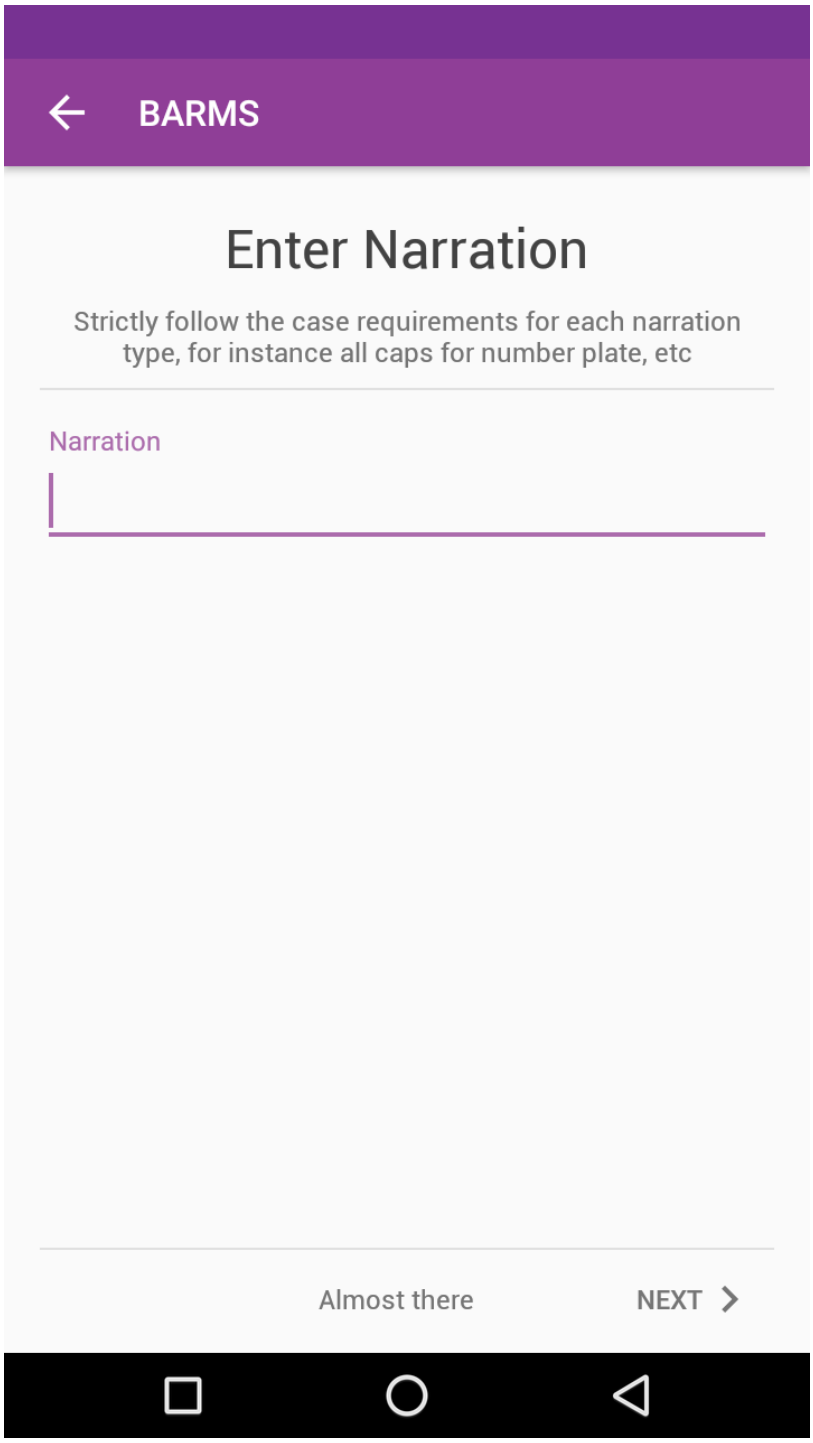


Figure 5.26: Narration Screen

Once a narration has been correctly entered the user is presented with a list of possible payment modes as illustrated in Figure 5.27.

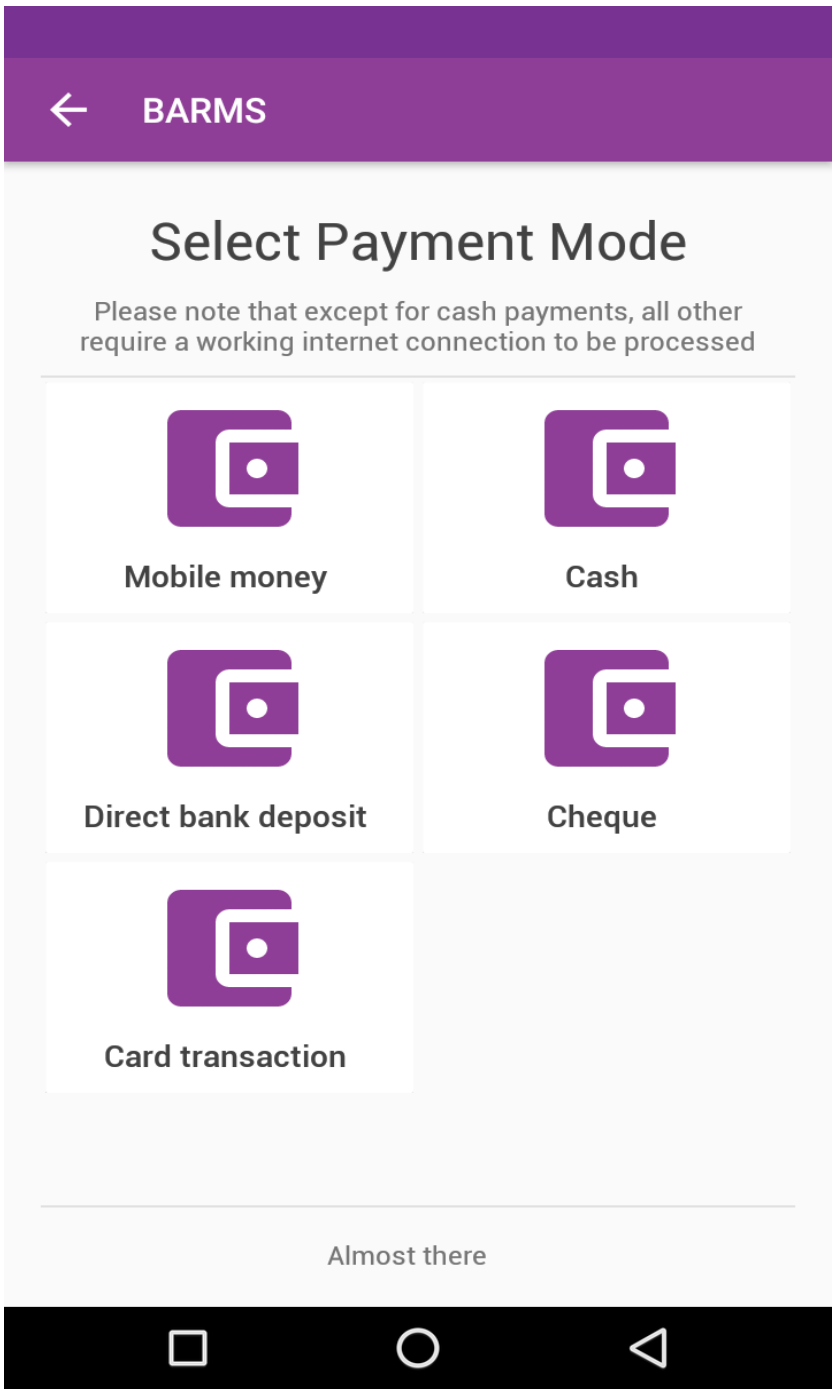


Figure 5.27: Payment Selection Screen

After selection of the payment mode the user is asked to enter details of the corresponding payment mode. This process is shown in Figure 5.28.

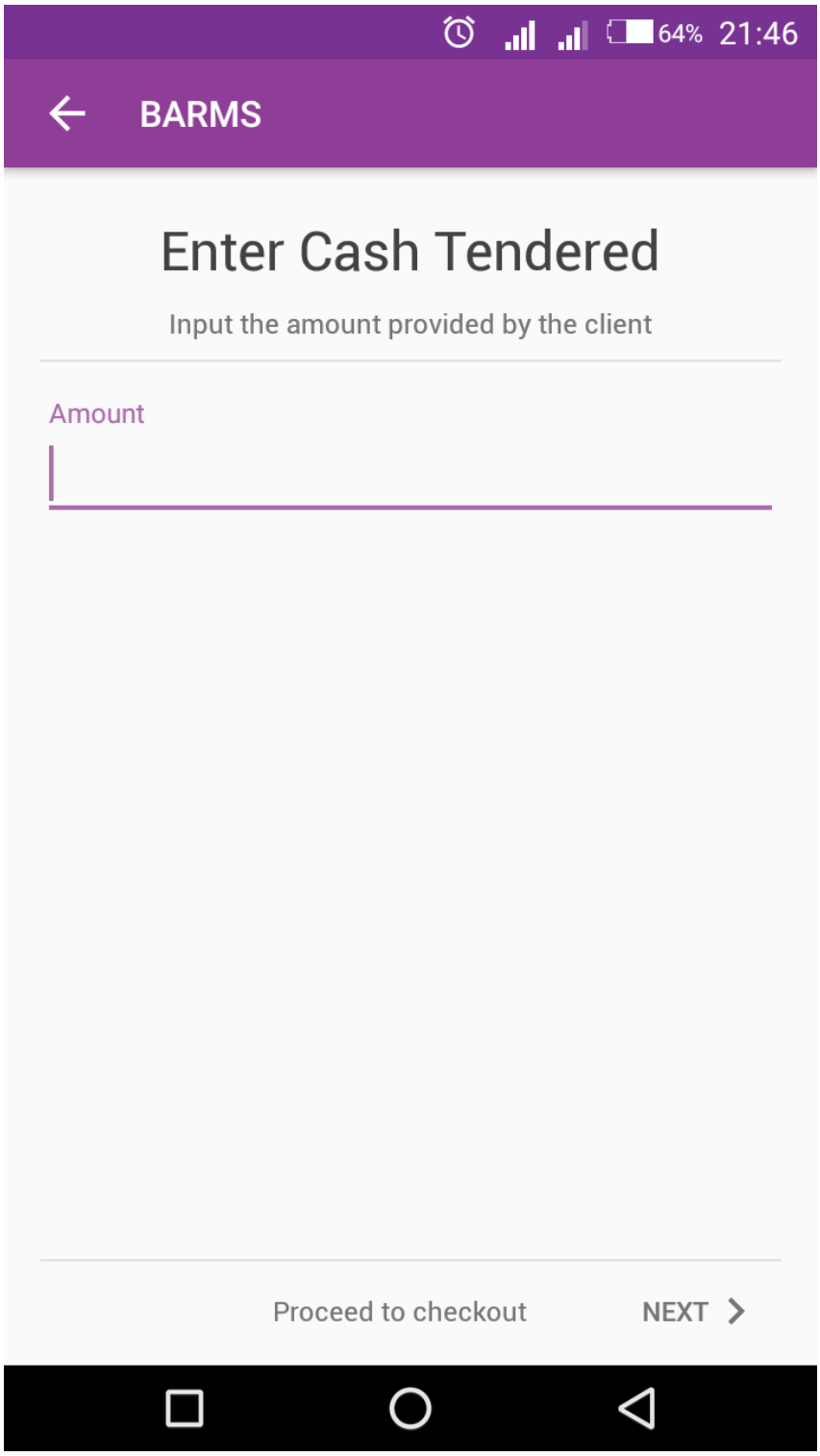


Figure 5.28: Cash Tendered Screen

The system validates to check whether the cash tendered meets the required charge then prompts the user to confirm the transaction. Once a user confirms a transaction they are taken to a receipt printing panel as shown in Figure 5.29.

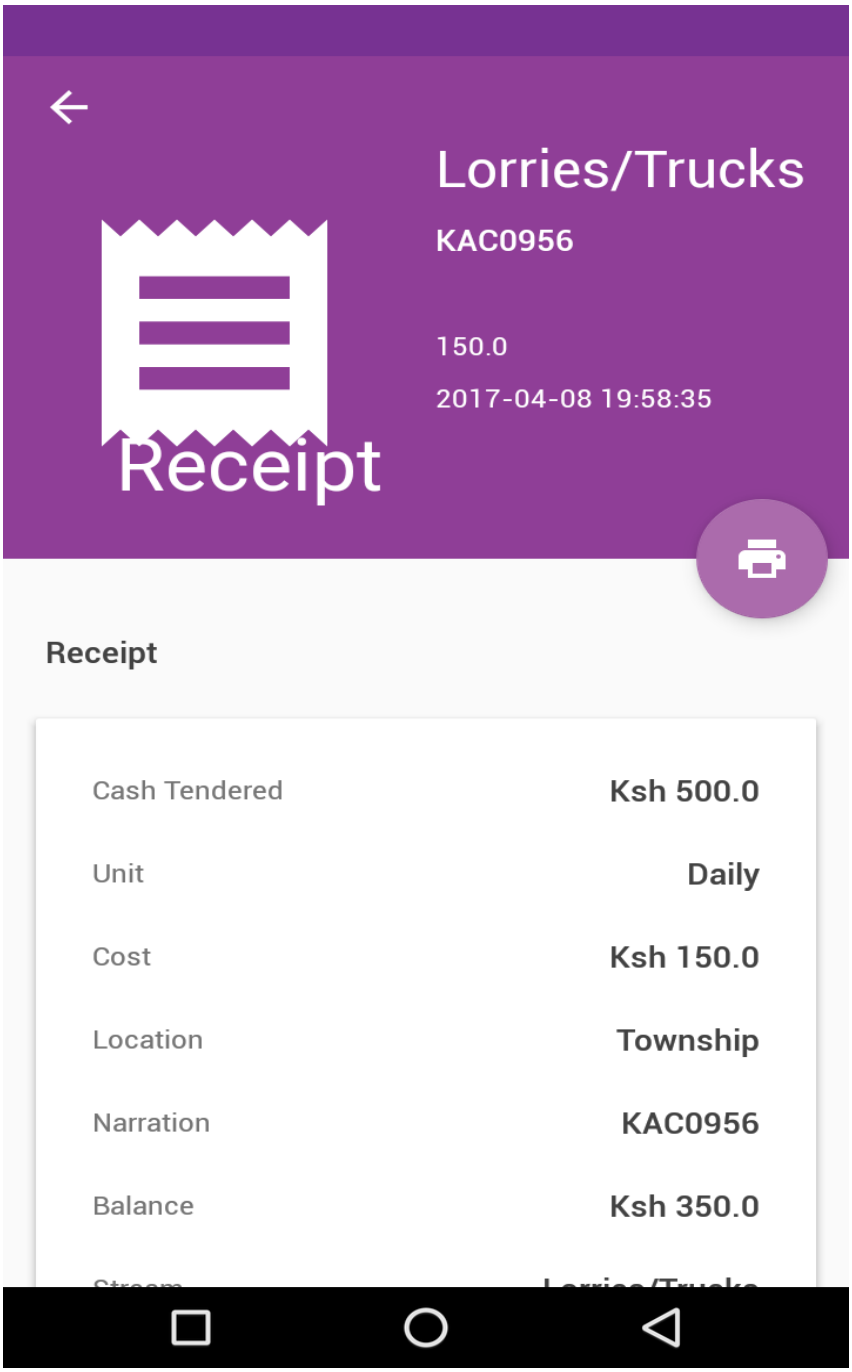


Figure 5.29: Print Receipt Screen

After receipting, collectors have panel for verifying receipts where they can scan a QR code or enter the narration details as shown in figure 5.30.

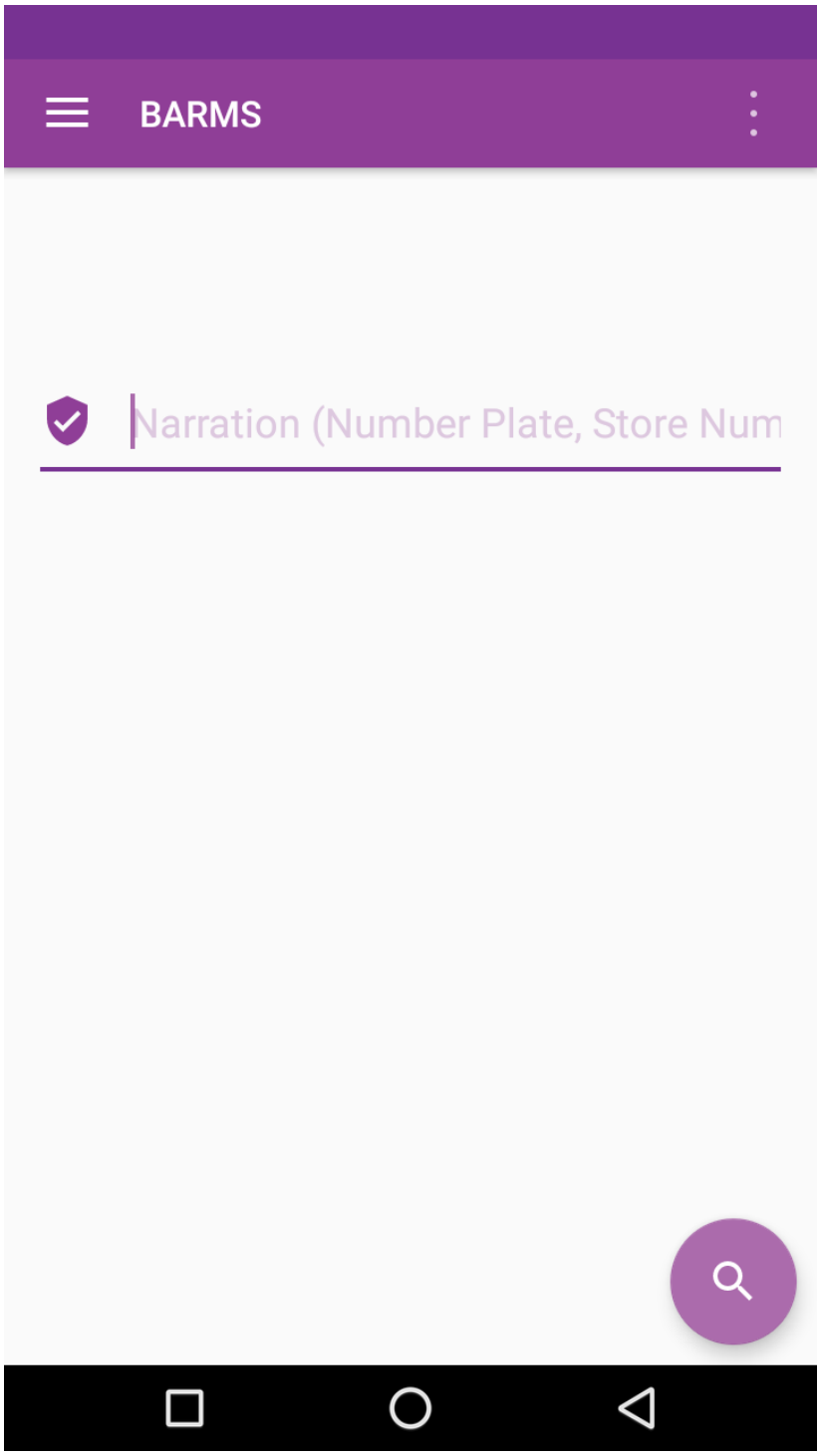


Figure 5.30: Receipt Verification Screen

The application also offers additional utility functions that support collectors in their day to day operations. Collectors are able to account for their collections and monitor their current status through a provision that enables them to print an end of day summary report. The report gives a summary of subgroups collected on a given day by a collector as shown in Figure 5.31.

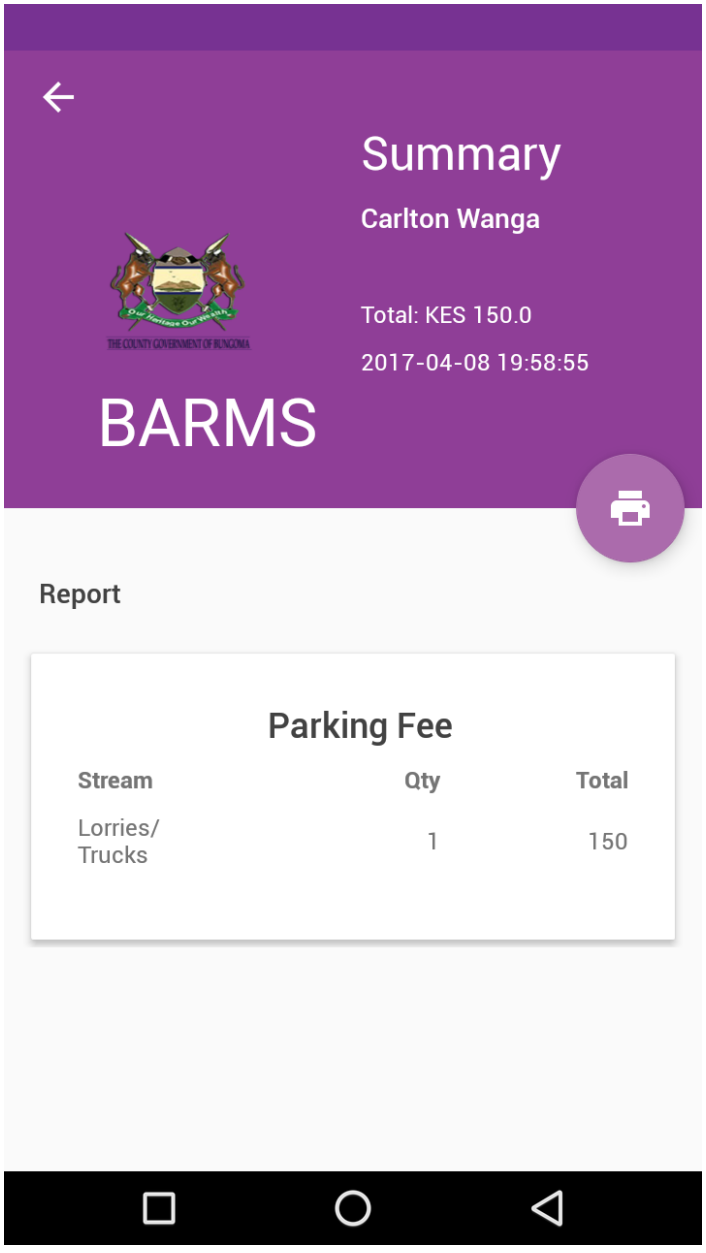


Figure 5.31: End of Day Summary

Other utility functions that collectors have access to includes a password management panel that enables collectors to reset their passwords. The process of password rest is illustrated in Figure 5.32.

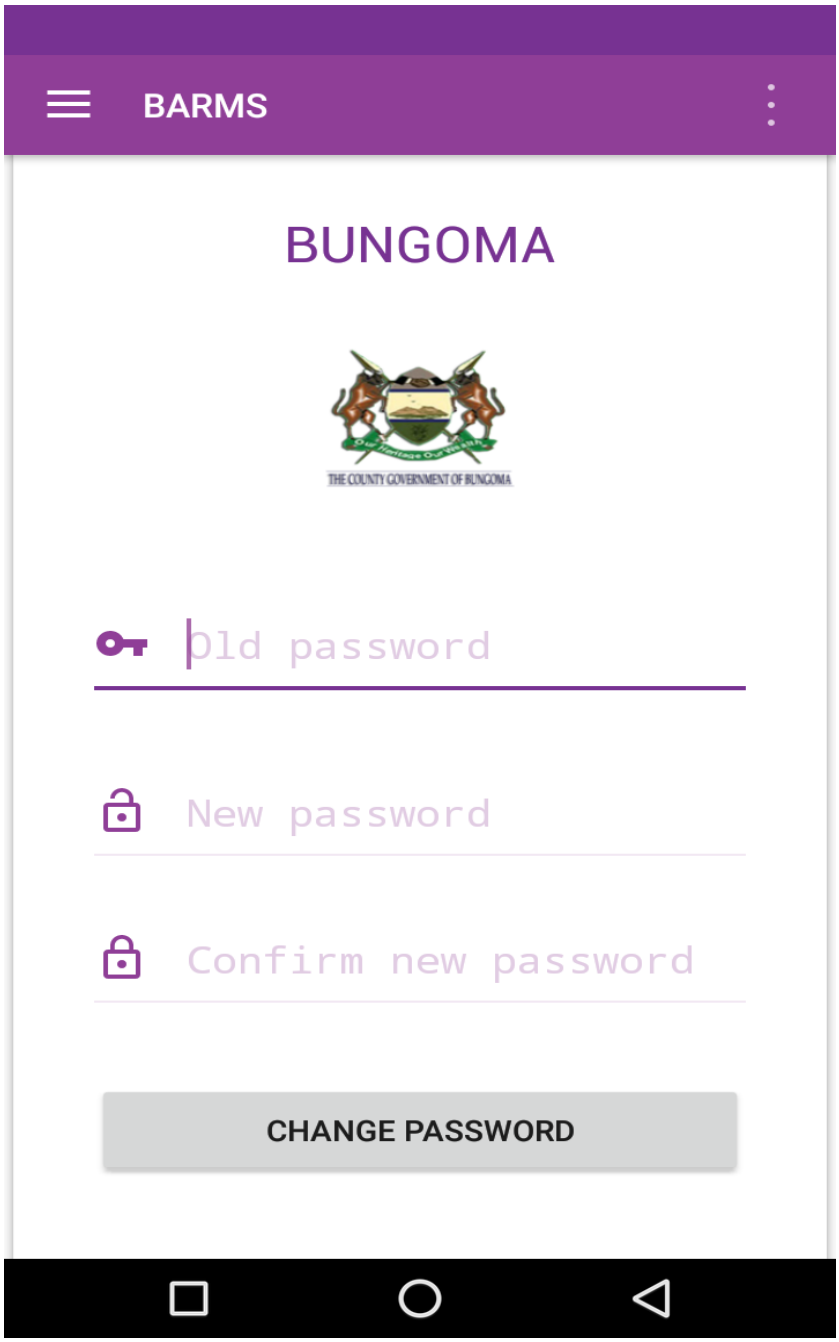


Figure 5.32: Change Password Screen

In addition to password resets, collectors also have access to configuring application preferences to determine the frequency of synching transactions as illustrated in Figure 5.33.

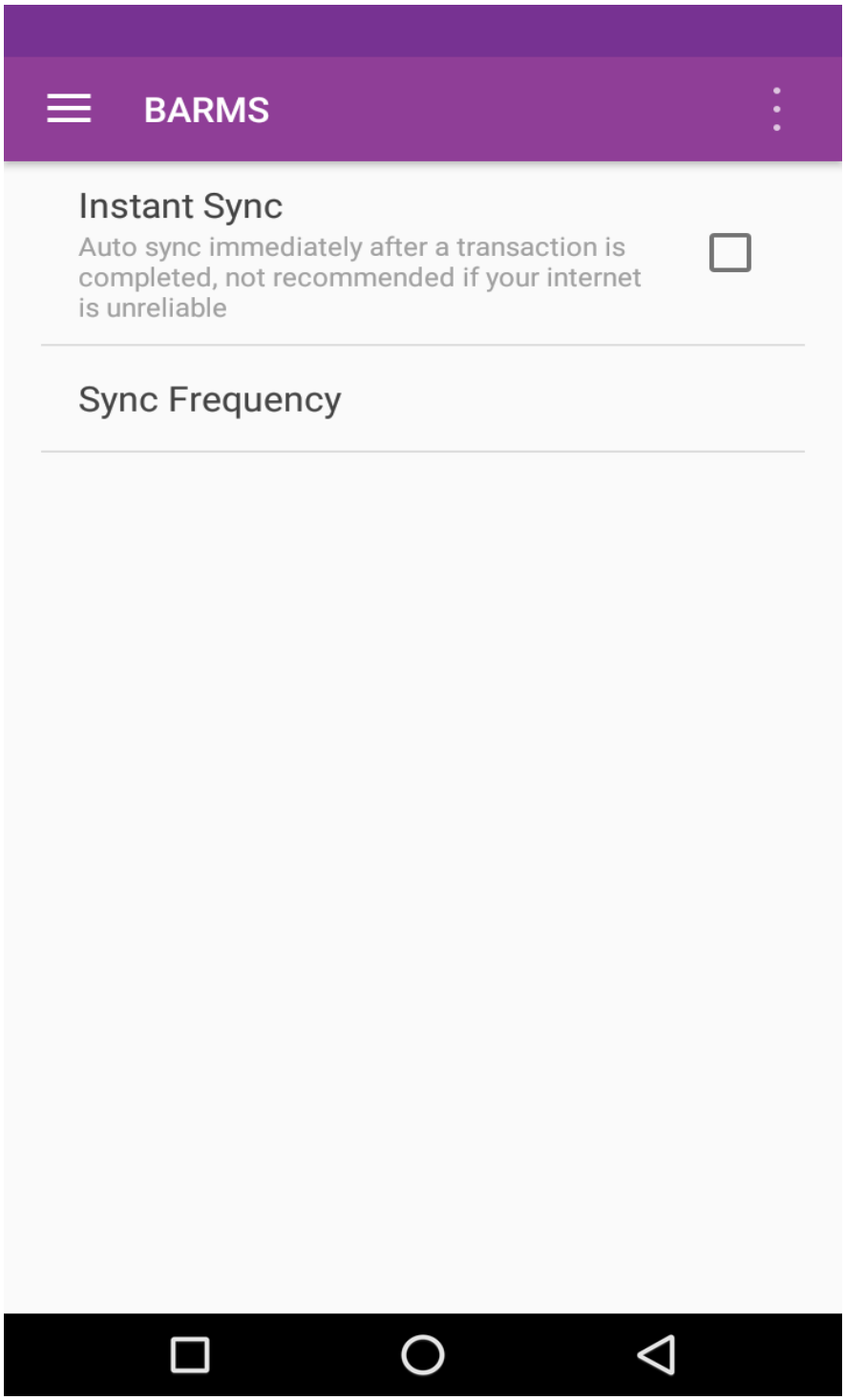


Figure 5.33: Application Preference Settings

5.6 Reporting

Accountants and revenue officers using the application have access to reports that enable them to make more informed management decisions. Reports are presented in a web based grid with pagination mechanisms. Users have the option of exporting data to various data formats of excel, word, pdf and csv.

5.6.1 Accounts Breakdown

The first report that users have access to is the accounts breakdown shown in Figure 5.34. This report gives a summary of the total amount collected as well as a breakdown of the corresponding collections made. Figure 5.34 shows a screenshot of the tabular report.

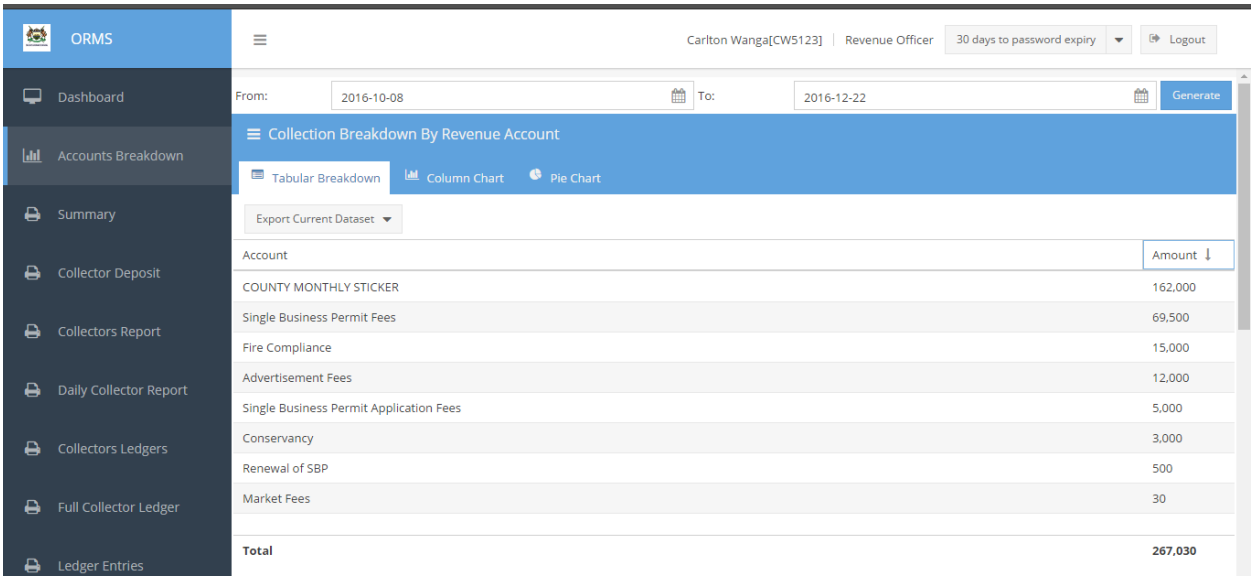


Figure 5.34: Accounts Breakdown Report

The report also gives visual representations of the breakdown by displaying a collection bar graph shown in Figure 5.35 and a pie chart to enable a better visual comparison of the revenue accounts. Figure 5.36 shows a screenshot of the accounts breakdown report represented on a pie chart.

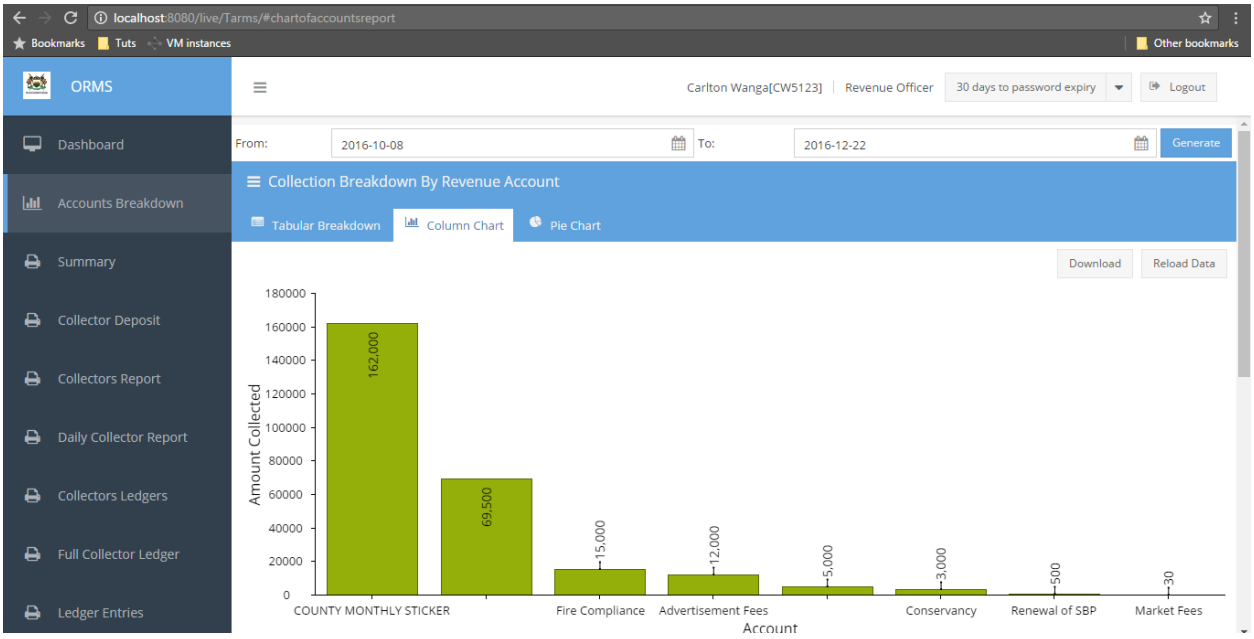


Figure 5.35: Accounts Breakdown Bar Graph

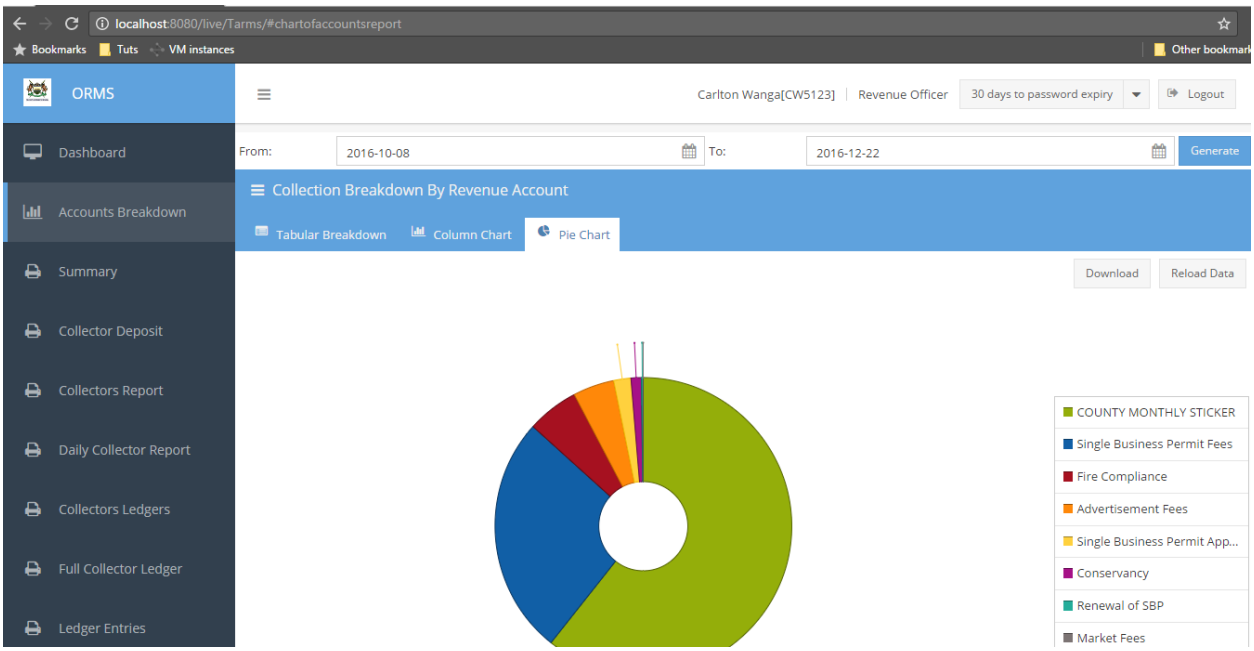


Figure 5.36: Accounts Breakdown Pie Chart

5.6.2 Collector's Report

To give a detailed summary of all the collection summary the system provides a report showing total amounts collected by each collector in a given date range. Th bne report also provides filtering capabilities that narrow the data set based on the location of collection as shown in Figure 5.37.

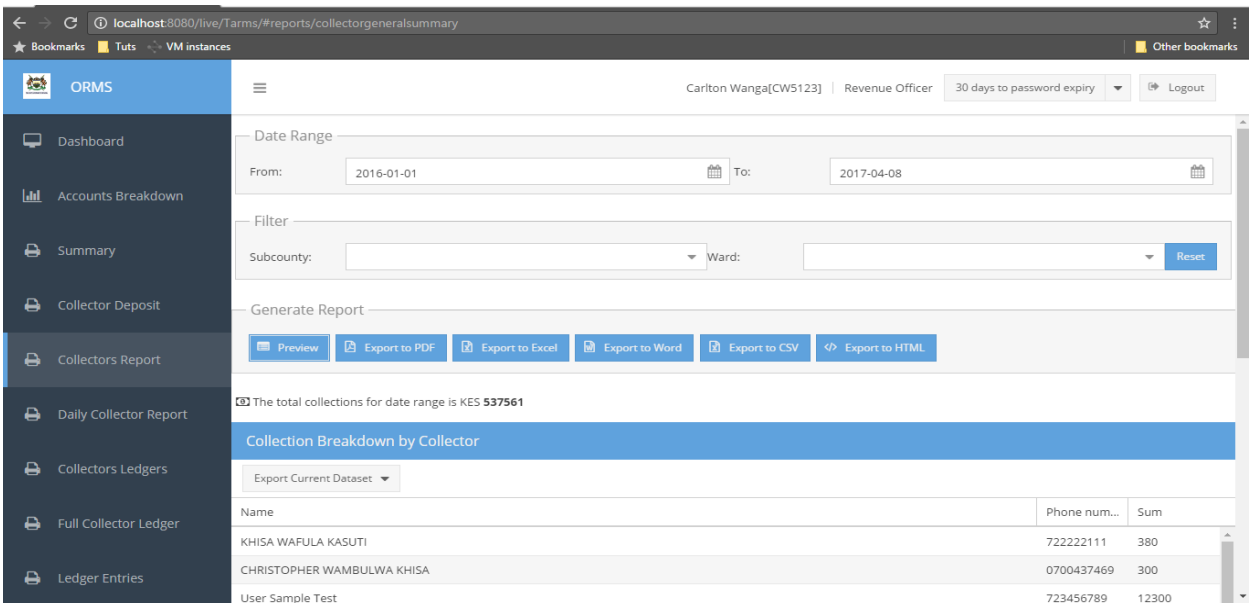


Figure 5.37: Collectors Report

5.6.3 Daily Collector's Report

The daily collector's report gives a detailed overview of every transaction performed by a collector on a given day. It provides a search criteria that makes it easy for officers to identify a collector. Once a collector has been identified, the system allows the user to select a date of collection and load corresponding transactions as shown in Figure 5.38.

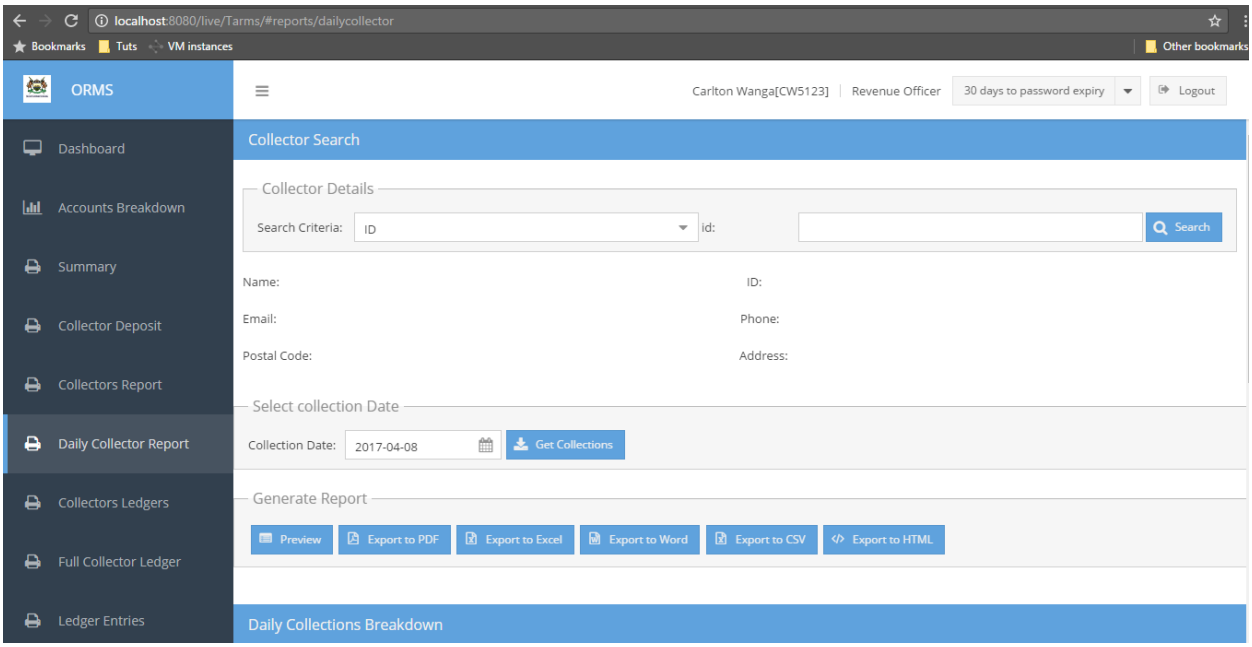


Figure 5.38: Daily Collector's Report

5.6.4 Device Monitoring

Figure 5.39 show the panel that enables officers to monitor device activities on a given date range. The panel shows details of collections made by every device. It also includes a tab for monitoring inactive devices.

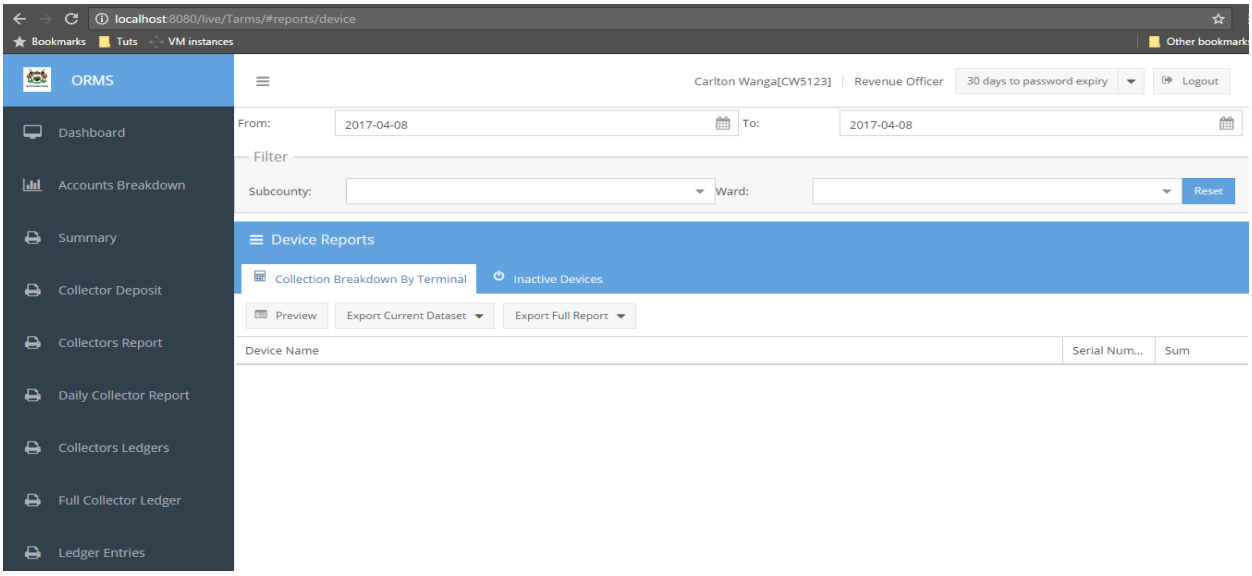


Figure 5.39: Device Report

5.6.5 Collectors Ledgers

Figure 5.40 shows ledgers that accounts have access to. The ledgers provide accounting reports of money collected by a collector verses the amounts banked.

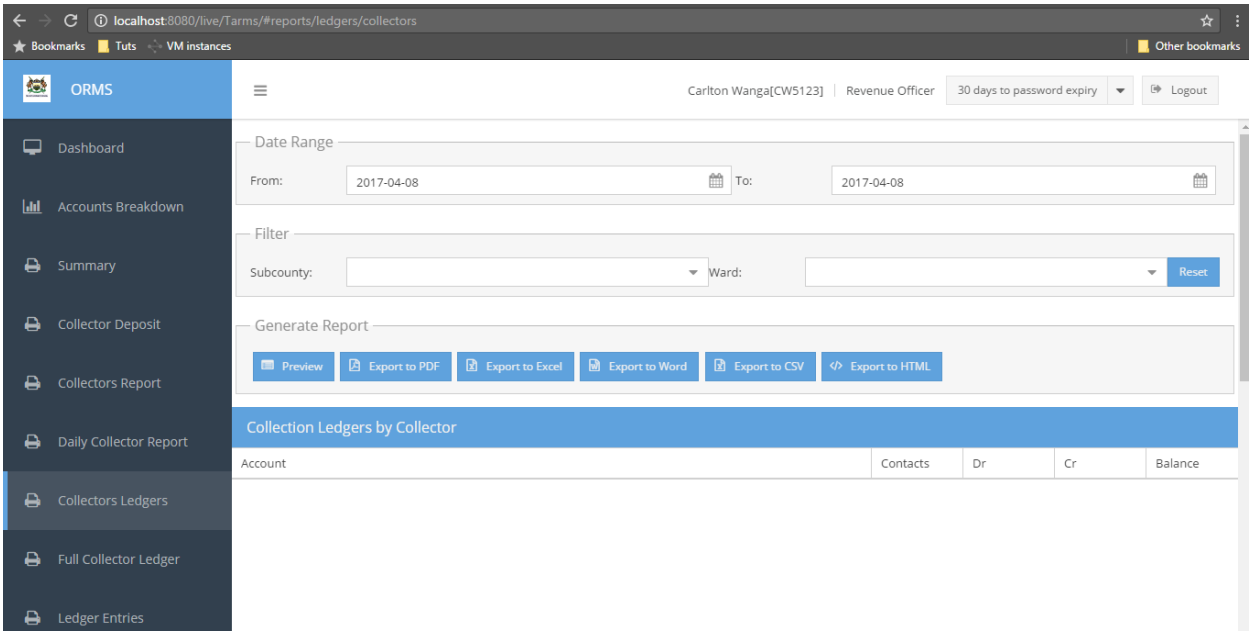


Figure 5.40: Ledgers Report

5.6.6 Collection by Revenue Stream

The web application provides a detailed overview of collections made on every revenue stream. This report is helpful in helping officers to determine which revenue sources raise better revenues so as to allocate more resources.

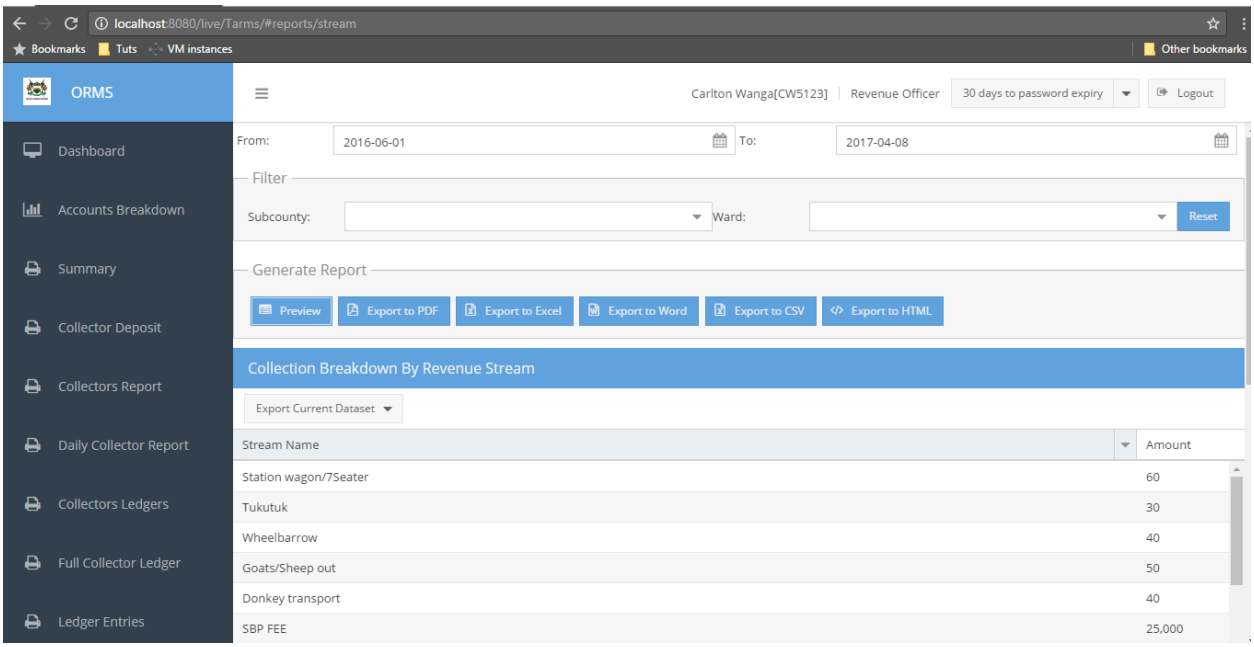


Figure 5.41: Collection by Revenue Stream Report

5.6.7 Collection by Revenue Subgroup

Figure 5.42 shows collections by revenue subgroup. This is a more generalised report that summarises collections of related revenue streams.

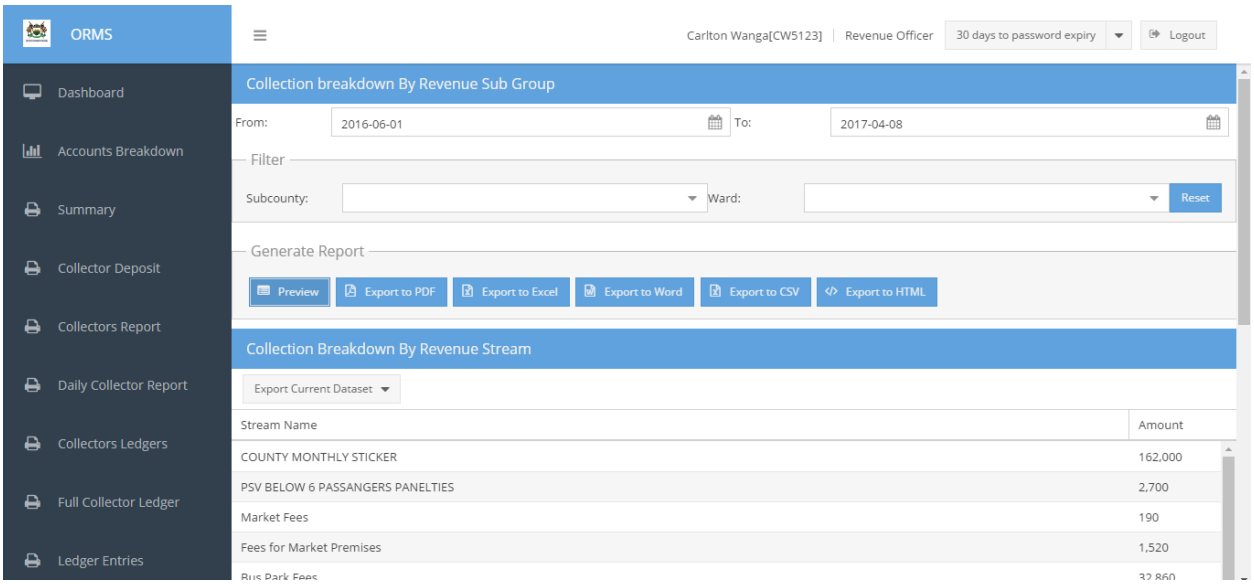


Figure 5.42: Collection by Revenue Sub Group Report

5.6.8 Collection Summary Report

A dashboard with important collection summaries is also presented to the revenue officers to show important collection information as presented in the screenshot on Figure 5.43.

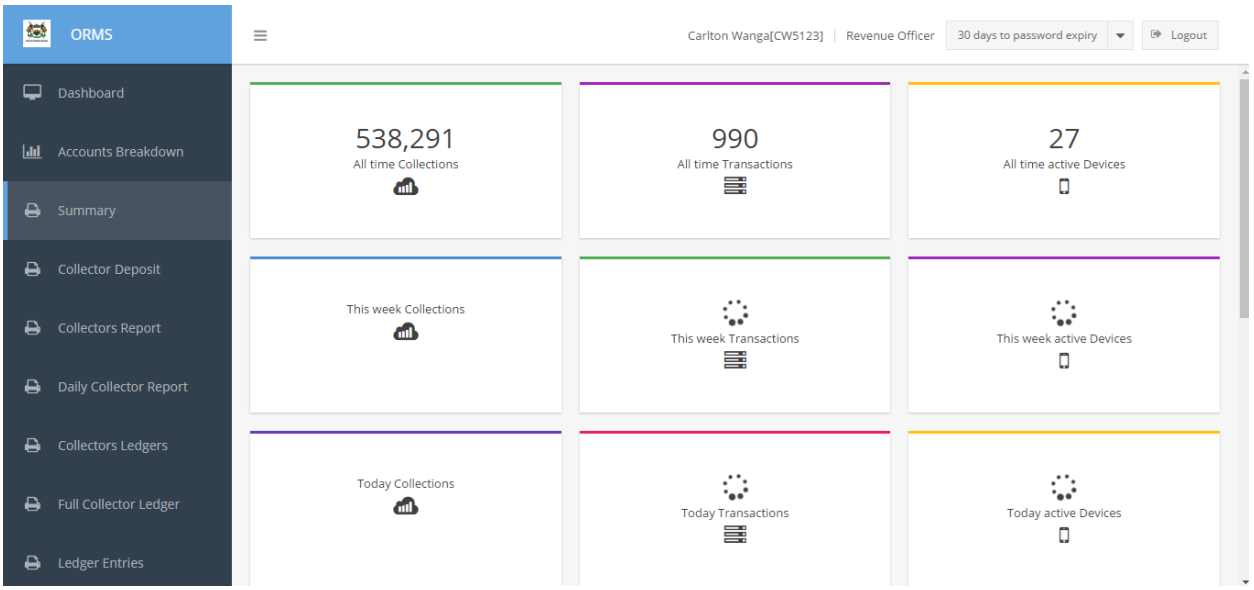


Figure 5.43: Collection Summary

5.6.8 Trial Balance Report

The system has a trial balance report that gives a summary of all accounts from an accounting perspective. The trial balance illustrated in Figure 5.44 generates the trial balance in accordance to accounting principles based on a given date range. The report is highly summarised and it mainly explains all transactions performed in the system.

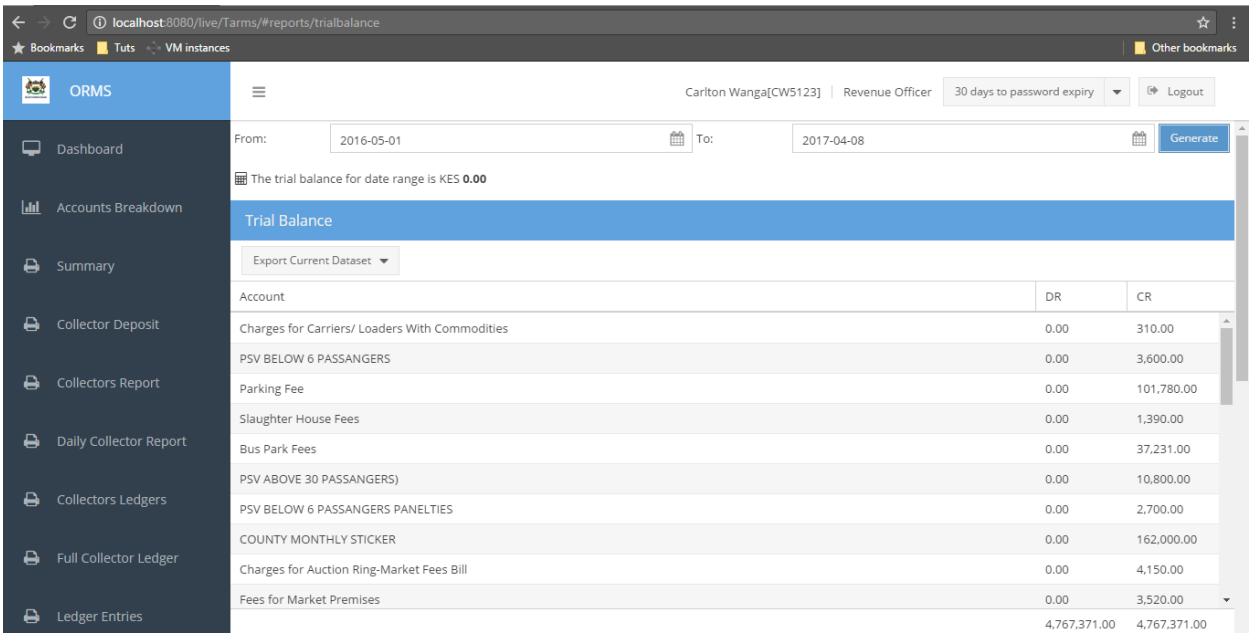


Figure 5.44: Trial Balance Report

5.7 Testing and Evaluation of Test Results

The application was subjected to a number of tests that were targeted to ensuring that the system met its main objectives. The first test case was based on determining whether the system meets all the functional requirements. In that case, a step by step test was undertaken to test the functionality of each component based on a test plan. The test plan included testing every unit of the application to verify that the expected results were obtained.

The tests were undertaken by using a module to module approach starting with the low-level processes that serve as a basis for other high-level requirements. Apart from testing to investigate whether the application met a specific requirement and that processes worked as expected, the tests targeted to verify the accuracy of the processes. This was achieved by checking whether a given input generated an expected system output.

Due to the many application components, it was also imperative to ensure that all components worked coherently. An integrated test was used to verify that the overall system worked as expected. The research also checked the compatibility of the application with platforms

that it was expected to support. This section provides a breakdown of the tests undertaken and the results that were obtained.

5.7.2 Functional Testing

Functional testing was conducted on every module of the system to test how various components behaved in comparison to the expected behaviour. The first approach involved supplying the system with predetermined parameters and analysing the processed results. To further confirm whether a component passed a functional test, an action was performed and the system behaviour observed as the operation executed. Table 5.2 outlines the results that were obtained when the application was subjected to functional tests.

Table 5.2: Functional Test Results

Component	**Description**	**Observed Behaviour**	**Error**	**Conclusion**
User authentication module	Validating whether users are correctly able to log in to the application	Users are able to gain access to the application only after supplying the correct username and password. Resources can also only be accessed when a user has the authorised user role and is also fully authenticated.	No errors	Working as expected
Device management module	Verifying that users can only access the system through	Collectors are only able to access the system through point of sale	No errors	Working as expected

	configured devices.	terminals with serial numbers that matches the devices that they have been allocated to		
User management module	Testing to check that user management operations are functional	Administrators are able to create new users. The system should generate usernames enable operations such as resetting of passwords as well as activation and deactivation of user accounts.	No errors	Working as expected
Data caching	Verifying that the application is able to download and save data dictionaries	The application accurately caches all data configuration items locally on the device	No errors	Working as expected
Offline collection	Checking whether payments can be received when the device is offline	Application operations should work comfortably even when the device is offline	No errors	Working as expected

Receipting module	Testing to check whether the application can generate payment receipt	The point of sale devices should be able to print a well formatted receipt after processing a payment	No errors	Working as expected
Data synchronisation	Verifying the ability of the application to synchronise data with the main backend	Application should upload all pending transactions then there is an active Internet connection	No errors	Working as expected
Offline receipt verification	Testing the application's ability to validate receipts when offline	The application should be able to check the validity of receipts generated by other terminals even when offline by scanning the receipt QR code	No errors	Working as expected
Reporting module	Checking whether the application is able to generate collection reports	The system should be able to correctly generate accurate receipts of all collection components	No errors	Working as expected

5.7.3 Usability Testing

Usability testing of the application was done to determine how friendly the application was to the end user. This test analysed the user experience when interacting with various components of the application. The test also analysed the user interface design and the ability of the application to correctly provide feedback on background operations. Table 5.3 shows a summary of results that were obtained in the usability tests.

Table 5.3: Usability Testing Results

Component	**Description**	**Observed Behaviour**	**Errors**	**Conclusion**
Input validation	Check to see whether the application validates input before users can submit	The application correctly validates input fields and allows submission only after a user types an input that matches the expected pattern. Users also have to fill in mandatory fields before they can proceed with an operation	None	Working as expected
Error messages	Testing to check whether the application displays correct and informative error messages	The system displays informative display messages for all errors and flags fields with wrong input values.	None	Working as expected
Loading indicators	A test to check whether the application shows loading status when	The application shows a spinning loader when performing background operations	None	Working as expected

	performing background tasks			
Data exports	Test to see whether reports can be exported to various data formats.	System correctly exports all reports to word, excel, pdf and csv formats	None	Working as expected

5.7.4 User Testing and Validation

A pilot exercise was at the county government of Bungoma and the application was used by 81 revenue collectors to validate the functionality of the application. The tests aimed to evaluate the overall user experience of the application as well as verify the functionality to ensure that all components worked as expected. The main test components were:

i. Functionality testing
ii. Usability testing

5.7.4.1 Functionality Testing

Functionality validation was done by users through comparing the application features and the requirement specification document. After all the features were found optimal for revenue collection through continuous revision of the prototype the application was put to use. The pilot tested the functionality of the application by subjecting it to actual usage. The main features of revenue collection were tested to ensure that the application would support all features without failing.

The application passed the main test criterial of ensuring that collectors were able to download data dictionaries. The second test was to confirm that the application was able to collect revenue offline and in that case successfully synchronize all the transactions. This test was passed as all the 81 devices were able to process payments and synchronize transactions were accurately reported on the backend.

5.7.4.2 Usability Testing

This test was used to validate that the application was user friendly. Users were request to give feedback. This was mainly about the theme, ease of use and overall learnability of the application. Out of the 81 users 72 of the collectors easily navigated though the application without the need for training. Out of the 9 users who experience difficulties 4 were of an age of above 55 years. This led to the conclusion that the application was friendly as users would comfortably navigate through with ease.

5.8 Conclusions

Implementation and testing of the system were targeted at meeting the overall objective of developing a revenue management system that was capable of generating receipts that can be verified offline. The tests reveal that the system is able to handle these requirements by providing a robust solution that is able to handle all aspects of offline revenue collection while considering security aspects.

Chapter 6 : Discussions

6.1 Introduction

Revenue collection and management is an important aspect of county government operations. The process is important in ensuring that counties are able to raise enough revenues to support their development agendas. To tap on the full revenue potential of county government, it is critical to ensure that the process id digitised. Having an electronic form of revenue collection will be very helpful in sealing revenue leakages. This is because it will provide a way of ensuring customers are correctly charged the appropriate fees. It is also clear that the current paper based process is not adequate to keep records for accounting purposes.

Reporting is an important area in every sector of governance. Lack proper reporting has greatly affected revenue growth in county governments. This is because there is no way of properly monitoring collectors and tracking their collection operations to ensure that every process is done in accordance to the laid down procedure and with high integrity. Having a digital means of receipting and managing revenue collection will helpful in solving the challenges counties face. This is because it will provide a reliable way of ensuring that officers are able to access reports and take corrective actions on collectors that do not meet their collection targets.

Despite the great improvement that can be realised by providing an electronic form of revenue collection, the challenges of its implementation cannot be underestimated. The first challenge that needs to be overcome is the provision of a mobile way of printing receipts. This is brought about by the fact that revenue collection requires a high degree of mobility. Printing of receipts would typically require a stand by form of electrical power to run the printer. The research has however discovered alternative approaches that can provide a mechanism of printing digital receipts on mobile devices through Bluetooth printing gadgets that have a long battery life and are highly portable.

Solving the process of mobile receipt generation to provide a digital receipt does not provide a solution to the challenge of providing monitoring capabilities of the collection process. Monitoring can only be realised by having system components that provide reliable reports. The reporting components can only be realised by providing a means of sending collection data to a central location. Network communication in the current setting is however a great challenge. Not all places of revenue collection are expected to have enough network coverage to support real-

time network communication. In addition, communication will be greatly challenged by the fact that the receipting duration needs to be as short as possible and making network connections will greatly affect the service time.

Offline collection is a challenging process. It is complicated by the fact that in cases of revenue collection, multiple devices are expected to store data that is ultimately expected to be mapped to a central database while ensuring that the integrity of the data is not compromised. It is also important to ensure that the unique record identifiers on local device databases do not conflict. In cases of financial systems such as receipting the generated receipt numbers should not only be unique but also unpredictable. The algorithm of receipt generation should have way of generating identification tokens that are not predicable and are also unique.

The research settled on the use of device serial numbers along with autoincremented database sequence as part of table primary keys to ensure uniqueness when data is synchronised. This can be a reliable way of design databases for offline mobile applications that need to synchronise with a central database server.

An offline application with syncing abilities would also require a way of storage of data dictionaries that have a direct mapping to the central database. The research adopted an approach of also developing local data tables for data dictionaries. The dictionaries are subsequently updated by the client performing polling operations to check for any updates on the server side and updating the local data store. It is also critical to provide a mechanism of recording both the device's collection timestamp as well as the time when the transaction has synchronised with the main backend.

A challenge to implementation of an offline collection model was the issue of managing data after synchronisation. The first approach would be to delete each record after it has successfully been synchronised with the server records. However, this would be an impediment to performing rollbacks. The research thereby adopted a mechanism as marking fields in the local device databases as synchronised after the synchronisation process. To ensure that the device is not affected by memory constrains the application automatically archives records at the end of the moth where they are purged after on financial year.

6.2 Review of the Research objectives in Relation to the Developed System

This section targets to analyse whether the research was able to meet the objectives set. It attempts to draw a relationship between the developed system and the objectives while discussing any findings. It also outlines any limitations or challenges that may have been experienced in the process of developing the application.

The first objective was to investigate the causes of prevalent use of counterfeit receipts in revenue management system. This objective was approached by analysing the current revenue collection system and process flow used to transact with paper based receipts. The research identified that the main reason of widespread use of counterfeit receipts was because of lack of a proper receipt verification mechanism. The solution to this problem would be developing an electronic receipting system that would make it easy to query the validity of specific receipts.

The research also discovered that lack of proper monitoring mechanisms is one of the main causes of prevalent use of counterfeit receipts. Administrators have no way of analysing and monitoring collectors in the process of revenue collection. This problem needs to be solved by developing robust reporting components that will make it possible for officers to monitor the current state of affair.

The second objective was to investigate the challenges that would be faced in implementing an offline electronic revenue management system. The main challenge was devising a portable way of generating receipts. The research discovered that the most ideal solution would be using point of sale terminals with in-built receipt printers. The research developed a solution that runs on terminals with inbuilt point of sale terminals. The developed application can also run on mobile phone applications since it can generate receipts using connected portable Bluetooth printers.

The research also targeted to find ways in which the offline application would be used to verify the validity of receipts. This was tackled by analysing the possibility of using cryptography as an alternative to making network connection in order to verify the validity of receipts. The application developed an algorithm of encrypting payment details into a token of random string characters which can only be decrypted to obtain payment details by a terminal with the decryption key and knowledge of the algorithm.

To make the verification process easier the application developed an approach of using QR codes on receipts to save time that would otherwise have been spent typing receipt numbers. The application therefore is able to tell the difference between a genuine receipt and a counterfeit one even if transactions from the terminal that generated the receipt have not been synched with the central database.

The research also delved into developing an integrated system that would enable offline receipting and receipt verification. The solution was built on a mobile application with the ability to generate receipts, verify receipts generated by other point of sale terminals and produce reports to monitor the revenue collection process.

6.3 Merits of the developed system

i. The application has the ability withstand network connectivity problems because of its ability to work offline.
ii. The system curbs the problem of prevalent use of counterfeit receipts by providing a robust receipt verification model.
iii. The application provides an electronic way of printing payment receipts make it easier to track collections.
iv. The system provides access to informative reports thereby making it easy for county officers to monitor collections.

6.4 Limitations of the approved system

i. Some operations such initial user authentication and data caching require Internet connection.
ii. Currently limited to running in point of sale terminals running on the Android platform.
iii. Verification receipt number relatively long making verification input by typing hectic. The limitation was however overcome by using QR codes to make verification inputs easier to scan.

Chapter 7 : Conclusions, Recommendations and Future Work

7.1 Conclusions

Effective tax collection is very important aspect of county government operations. It is evident that there are so many challenges facing the current paper based way of tax collection. The main challenge is that most of the money collected is paid to the back to the county since loophole enable revenue collectors to pocket a great share of collection fees. The greatest solution to the problem is implementation of an electronic way of revenue collection.

The research has concluded that the current network infrastructural development status of county governments is not stable enough to support real-time receipting that involves server side communication. This is because network coverage is not always available. In addition, the time taken to post transaction details to a server side application will greatly delay the transaction time.

Offline receipting and receipt verification would greatly improve the process of tax collection since it solves the major challenges facing revenue collection in county governments. Implementing the solution will therefore improve the amount of money that is currently being collected and in the long run empower counties to promote development.

7.2 Recommendations

An offline revenue automation model is a new concept that also provides a number of implementation challenges. The research has identified a number of findings that can greatly enhance automation of revenue collection and at the same time function in other application arears that require payments and receiving of payments. The algorithms and models used in the development of the system are not limited to the use case of revenue automation. The same model can also apply in other industries such as shipping and delivery, hotel bookings and payments of event tickets.

Due to the many advantages that come with electronic receipting I greatly recommend that this application is adopted by stakeholders in fields that handle processing of payments. If more research is put into improving the algorithms that make it possible to perform offline validation of receipts, the technology offers great potential in improving sales and the overall customer experience when making payments.

The research also recommends adoption of cashless payment methods as a way of complementing offline receipting to provide customers with more payment alternatives. Cashless payments would also minimise cash handling by revenue staff thereby reducing cases of mishandling of money by revenue collectors.

7.3 Future Work

The study aims to improve on the limitations outlined in the future. The first step would involve improving the algorithm to ensure that there are no chances of collision while at the same time reduce the length of the of the receipt number. This will make it possible for the application to run on platforms that do not have a QR code scanner.

The study also targets to develop solutions that can run on other platforms apart from Android. The application will be developed to run on the iOS platform in addition to windows phone. Since most point of sale terminals also run the j2ME platforms the study also plans to develop support for this platform in future. In addition, the research also targets to develop additional payment modes for the application including mobile money and card payments.

References

Adera, E. O., & Waema, T. M. (2011). *Local governance and ICTs in Africa.* Cape Town, South Africa: Pambazuka Press.

Alwyn, H., & Albertus, P. (2013). Securing Number plates based on Digital Signatures and RFID. 21-27.

Appel, F. (2015). *Testing with JUnit.* Packt Publishing.

Ataro, P. O., Muturi, W., & Wandera, W. R. (2016). Factors Affecting Revenue Collection Efficiency in County Governments in Kenya. *A Case Study of Trans-Nzoia County*.

Barclay, K., & Savage, J. (2010). *Groovy Programming.* Elsevier Science.

Basu, S. (2010). Possibility and Desirability. *Direct Taxation and E-Commerce*, 37-63.

Birgit, P. (2002). *Payment Systems.* Saarland University.

Buluma, F. C., & M.Obande, M. N. (2015). Justification for a Devolved Systems of Government: Corporate Governance and Financial Management Issues in Kenya.

Centeno, C. (2008). Securing Internet payments the potential of public key cryptography, public key infrastructure and digital signatures.

Christin, N., & Safavi-Naini, R. (2014). Financial Cryptography and Data Security. *18th International Conference.* Berlin: Heidelberg.

Commision of Revenue Allocation. (2015). *CRA 2015 Conference Action Points.* Nairobi: Comission of Revenue Allocation.

Commission of Revenue Allocation. (2015). *CRA Annual Report for financial Yera 2014/2015.* Nairobi: Commission of Revenue Allocation.

Datecs. (2017, Febuary 02). *DATECS LTD*. Retrieved from DATECS LTD: http://www.datecs.bg/en/products/61

Deloitte. (2017, January 07). *Smart device, smart pay*. Retrieved from DU Press: https://dupress.deloitte.com/dup-us-en/industry/retail-distribution/mpayments-mobile-pos-system-in-retail.html

Diallo, F., & Calland, R. (2013). *Access to information in Africa : law, culture and practice.* Leiden.

Farley, J., Crawford, W., & Malani, P. (2006). *Java enterprise in a nutshell.* Beijing: O'Reilly.

Godbole, N. (2006). Relating Mobile Computing to Mobile Commerce.

Gutierrez, F. (2014). *Introducing Spring Framework.* Berkeley: Apress.

ISACA. (2011). Mobile Payments. *Risk, Security and Assurance Issues*, 5-9.

István, M. (2008). New Challenges for Smart Organizations.

Jih, W.-J. K. (2009). Service Features, Customer Convenience, and Shopping Intention in the Context of Mobile Commerce.

Johri, A. (2008). *Online, Offline, and In-Between.*

Karori, N., & Abuga, V. M. (2016). Influence of Revenue Colletion Efficiency on the Operational Performance of Kisii County Government, Kenya.

Khaunya, F. M., Wawire, P. B., & Chepng'eno, V. (2015). Devolved Governance in Kenya. *Is it a False Start in Democratic*, 27-29.

Lee, I. (2009). *Triangular Strategic Analysis for Hybrid E-Retailers.* InfoSci-Books.

Leonardo, C., Beniamino, D. M., & Marcello, M. (2014). *Smart organizations and smart artifacts : fostering interaction between people, technologies and processes.* New York: Springer.

Lumumba, O. M. (2010). The Effectiveness of Electronic Tax Registers in processing of Value Added Tax Returns. *AIBUMA Publishing*, 45-46.

Mburugu, K. P. (2016). Determinants Influencing Revenue Collection on the performance of Kenya Revenue Authority.

McKenzie, C. (2008). *Hibernate made easy : simplified data persistence with Hibernate and JPA (Java persistence API) annotations.* Hiberbook.

Muthama, J. (2013). The Effects of Revenue System Modernization on Revenue Collection At Kenya Revenue Authority.

Mutysia, J. (2014). Effects of an Interfreated Revenue Management System and Challenges facing its Implementation in Machakos county .

Mwenda, K. A. (2010). Economic and Administrative Implications of the Devolution Framework Established by the Constitution of Kenya.

Nduda, J. M. (2015). Analysis of Factors Influencing Optimal Revenue Collection By County Governments in Kenya.

Ngotho, J., & Kerongo, F. (2016). Determinants of Revenue Collection in Developing Countries. *Kenya's Tax Collection Perspective*.

Obe, R. O., & Hsu, L. S. (2015). *PostgreSQL : up and running.* Beijin: O'Reilly.

Ochieng, D., Gichoya, D., & Odini, C. (2011). Proposed ICT-enabled services model for local authorities in Kenya. 1-9.

Omwansa, T. K. (2014). Introduction of VAT on ICT Equipment in Kenya. *Special focus on Mobile Phones*, 30-33.

Osoro, S. K., Atambo, W. N., & Abuga, V. V. (2015). *Effects of Revenue Collection on the Relationship between Deficit Budget Financing and Operational Performance of Kisii County Government, Kenya*.

Ozar, G. (2012). *MySQL management and administration with Navicat.* Birmingham: Packt Pub.

Pedro, F. (2015). *Understanding Bitcoin : Cryptography, Engineering and Economics.* West Sussex: Wiley.

Raina, V. K. (2014). *Overview of Mobile Payment.*

Roger, S., & Jupp, V. (2006). *Data collection and analysis.* London: SAGE Publications.

Sharma, M. (2016). Services of Mobile Commerce.

Siehl, E. (2010). *Addressing tax evasion in developing countries.* schborn: Deutsche Gesellschaft fur .

Simon, M. G. (2013). Mobile-based verification in anti-counterfeit commodity supply chain management systems. 379-388.

Society for International Development. (2012). Public Finance Reforms in Kenya. *Issue and Relevance Under the Context of Devolution*, 12-16.

Souter, D., & Kerretts, M. (2012). *Internet Governance in Kenya - An Assesment for the Internet Society.* Nairobi: Ict Development Associates.

Tanai, K. (2014). Monetary and Capital Markets . *Oversight Issues in Mobile Payments*, 5-19.

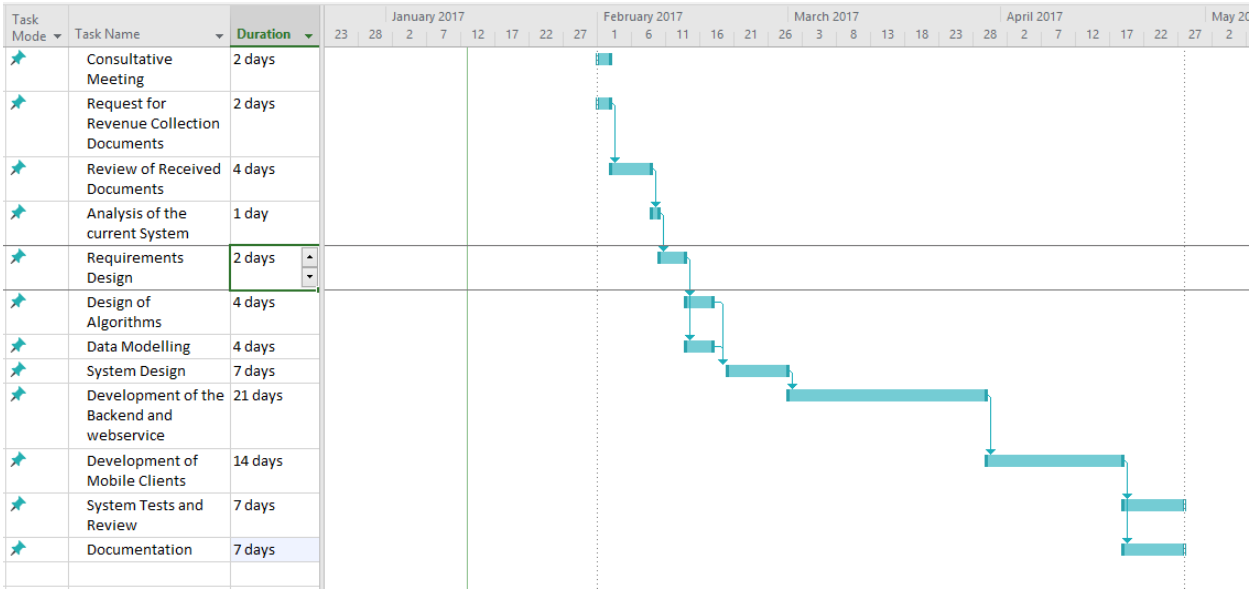
Twum, A. I. (2015). The Effective Use of Quick Response (QR) Code as a Marketing Tool.

William, E. L. (2009). *Software testing and continuous quality improvement.* Boca Raton.

Yamamoto, Y. (2012). Open Service Field-Point of Service. *A Method to Continuously Observe Tourist Behavior in Sightseeing Areas*, 86-98.

Appendices

Appendix A: Project Development Time Plan



Task Mode	Task Name	Duration
	Consultative Meeting	2 days
	Request for Revenue Collection Documents	2 days
	Review of Received Documents	4 days
	Analysis of the current System	1 day
	Requirements Design	2 days
	Design of Algorithms	4 days
	Data Modelling	4 days
	System Design	7 days
	Development of the Backend and webservice	21 days
	Development of Mobile Clients	14 days
	System Tests and Review	7 days
	Documentation	7 days

Figure A 1 Project Gant Chart

Appendix B: Turnitin Report

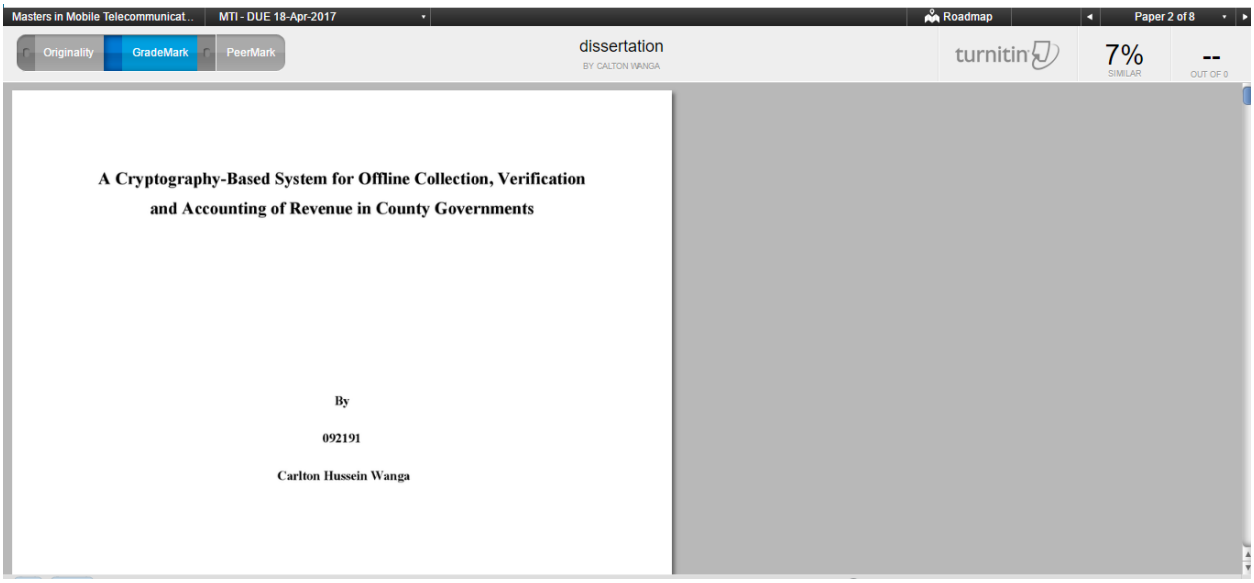


Figure A 2 Turnitin Report