

LOST IDENTIFIABLE ITEMS RECOVERY SYSTEM

^{1*} **Gidraph Danford Maina** *danfordmaina@gmail.com* ^{2**} Josphat Karani Mwai jkaraniw@gmail.com ^{3***} **Mwangi Njomo** mwanginjomo2018@gmail.com

¹ Student, Bachelor of Science in Information Technology, Kirinyaga University, Kenya ^{2, 3} Tutorial Fellows and Researchers, Kirinyaga University, Kenya

Abstract: In Kenya, many agree that the current system that deals with recovery of found lost items is very inefficient and largely inexistent. Lost property including identity documents require one to follow all the painstakingly slow and corrupt procedures to recover their property, if at all possible. In most countries, identification cards are a necessary document to hold a formal job, own property, receive basic social services, benefit from social transfers and carry out bank transactions. Passports and several other identification documents are also key entities that a Kenyan citizen will need to live a fulfilling life. However, there are numerous complaints that the process of identity cards and other key identity documents issuance or replacement is plagued with unbearable delays and corruption. Delaying citizens their lost items and documents has serious repercussions in economic, social and political spheres of their lives. The period between the time of losing said items and finding them has been shown to last anywhere from a few months to never actually getting them. This research project therefore aims to develop an integrated web and mobile-based system that facilitates significantly easier recovery mechanisms of all lost identifiable items that happen to be found by strangers.

Keywords: Identifiable, Lost, Recovery, Identity Documents

INTRODUCTION

The project aims to create a web and mobile based system that makes it much easier to recover lost identifiable documents such as national, work, school and other IDs, Passports, medical cards, ATM cards among others. Such a system has a very broad significance and importance especially as recent times have proven just how crucial and necessary it is for an individual to possess these documents. According to Liquid Telecom Kenya CEO Kenya demonstrates the multiple benefits gained from the broad connectivity in the country, where each investment in this sector accelerates the next, with an ever-increasing footprint of beneficiaries. [1]. This research project outlines the design, analysis, implementation and testing that such a system will call for.

The general view is that there are hundreds if not thousands lost personally identifiable documents and items in every town within the country. This poses a clear failure in the systems in place that are meant to tackle this or lack off. Lack of a central "lost" database means the owners of any of these lost documents and items are forced to instead replace them entirely. For national documents such as IDs and Passports, one must go to a police station and report, upon which you receive the police abstract that kickstart the tiresome process of renewal at a Huduma Center. Also, there are malicious people who can use your lost IDs and such crucial

documents to commit illegalities in classic cases of identity theft. The entire renewal process is resource consuming, tiresome and eventually very stressful.

Losing a passport is even more of a headache. Any planned flights or abroad trips regardless of how crucial and important they are will be delayed and postponed for not less than 3 weeks. Even the replacement process would cost the victim nothing short of Ksh. 10,000/=. These are just a few of the reasons why developing and implementing a centralized well maintained and user friendly ubiquitous system to aid in quick recovery of said lost items and documents is crucial. Although it may not solve the problem entirely, this project has provided an excellent solution that is scalable and very easy to implement on the large scale.

1.1 BACKGROUND

The explosion in use and access to the web throughout the world and the country specifically has opened avenues which can be exploited to realize my system efficiently and easily. Also, the pervasive possession and widespread use of smartphones in Kenya has gone go a long way to enabling my system achieve its major objectives. Further, in a recent statement by the ministry of Technology, Cabinet Secretary Joe Mucheru stated that Kenya ranks 14th in the list of countries with fastest internet as of 2017.

As at the end of the fourth quarter of 2016/17 financial year, the number of mobile subscriptions stood at 40.2 million. The mobile penetration level rose by 2.5 percentage points to stand at 88.7 per cent. The total number of internet subscriptions amounted to 29.6 million. The growth was driven by 15.3 per cent increase of mobile data subscriptions to 29.4 million at the end of the quarter which accounted for approximately 99.0 per cent of total subscriptions. As at the end of the fourth quarter of the financial year 2016/17, the number of mobile subscriptions stood at 40.2 million. [7]

Most of the companies in the country have adopted a mobile first approach to marketing as a result of the statistics by CCK. This has created a feedback system where even more and more clients now prefer to reach these companies using the mobile platforms. Because of this, the technology in this area has been fine-tuned by the leading mobile service providers in Kenya and my project has taken take advantage of such technologies as USSD, text, mobile Apps and QR code scanning. My project aims to revolutionize the process of recovering lost personally identifiable items and documents by leveraging both the mobile and web platform to become truly ubiquitous.

1.2.CURRENT SYSTEM

As at current, businesses usually recover valuables that customers leave behind in their premises. However, in most cases they don't even bother to find the owners of said lost items using the available platforms such as phone calls or online posting. This is mainly due to both lack of an easy, user-friendly system that would make such undertakings a breeze and also due to the fear of being prosecuted by the owners wrongfully. This way, these items end up being put into heaps of lost or stuck on windows, power poles or just thrown away.

Most of the current systems mostly focuses on lost national identity cards. There's numerous documents one national can lose besides identity cards: license, Permits 1) Passports and immigration Driver's Job papers, certificates such even scholarly certificates 3) Important as birth, death. marriage or 4) Vehicle logbooks: proof of ownership for automobiles and also to identify one's vehicle 5) Military Id, Title deeds, Voter's cards

Furthermore, one can lose even other items which are not necessarily identifiable such as bags, electronic devices, clothing, merchandise etc. there is currently no available formal system to assist the common civilian retrieve these things. Such items are either lost for good or in the rarest of cases get found and returned to a common place in the hope the owner might come back at one time.

There are numerous drawbacks to all the current systems as most of them are either underutilized due to their business model approach or lack of widespread knowledge and an ease-of-use approach to their design and implementation. Also, only one has a mobile app which was last updated on 2016 and only has 50 installs according to Google Play Store. [3].

1.3.PROBLEM STATEMENT

Anyone who has at one time lost their identity card, for example, would acknowledge that having to replace the same is a tedious process in Kenya. This problem doesn't however only apply to identity and documents that are personally identifying. In general, losing anything at all is both psychologically and financially demanding. The whole process of re-application and collection of such ID-able documents is impeded by unnecessary delays. The problems that come with misplacing such documents are such as Legal barriers, Delayed or cancelled, Voided land ownership claims, no proof of identity, Incapacity to undertake certain civil obligations like voting...etc. Losing other personal items also have huge implications on our livelihood. Consider losing a phone or a laptop which contains a lot of important information that we may require promptly. The key problem is that there is yet to be a well-maintained, effective and most importantly organized, easily accessible centralized system that can help a person in the recovery of their lost items hassle free.

1.4.SYSTEM

The system is a lost items recovery system that works on both web and mobile platforms to rapidly improve and ease the recovery of lost identifiable items in the country. The system allows users to search for their lost items in the central database as well as report lost items details to get notified if so said items are found. It also provides the ability to post details of found identifiable items which are stored in the database and indexed for searching.

The system helps individuals evade long queues at immigration departments, embassies, registration of person's offices and other such departments re-applying for the same documents. With a convenient way to allow Kenyans who lose their items and those who find the same to interact and provide information of where to find the same, considerable time is saved contrary to making countless trips to renew the papers. The national registration process in Kenya leaves minorities on the edge of statelessness. This is especially in the case of immigrants, say, from Sudan, they mostly have a hard time proving their nationality without the necessary papers. [4]

It exposes a web portal on its app as well as WebApp and a Mobile App where a user/victim can post the details of their lost identifiable documents, search the database for said items and also post the details of found identifiable items. The items which a person intends to have found in case of loss are tagged using a sticker that has a QR Code printed onto it. This widen the application base for the system to cover anything that the system's user might want to tag using a configurable QR Code sticker which links to the profile of the item.

The portal provide security by obscuring the photos if any using a blurring effect to curb any malicious use of the information in the database. There have been only three such systems implemented or attempted in the country of which only one is currently active which is a paid service. This idea was inspired by seeing the

immense number of lost IDs, ATM cards, Medical cards and many other items strewn all over. Also, an article about one Kamuzu Banda on the standard which stated "there are hundreds of lost national identification cards lying around the country and we don't have a central 'lost' location where their owners can visit to collect them" - was a source of inspiration. [2]

For mobile users on other platforms and those who don't have access to smartphones, there is a USSD/SMS based approach which make the system usable by just about anyone. The intended system's main approach to the problem on mobile however, is the use of QR Code printed stickers. This is mainly because all the previous systems under case study have had the same point of failure, that is, assumption that people are inherently good and willing to help. But after prolonged discussions with colleagues on the subject, it became evident that most people want to apply the least amount of effort in helping strangers.

Therefore, using QR Code stickers is a psychological approach that I believe are effective. It has been shown in studies that the human brain receives its input at over 90% visually and also processes this 60000x faster than text. [5] QR Code printed stickers register faster to a stranger passing by and significantly increase the chances that the person investigates and eventually scans the QR. Upon scanning the QR Code, the finder's smartphone opens a browser window automatically and the system pick this as a ping and collect the GPS location (if available and allowed) or collect the device's IP and try and trace its location through online free APIs in the backend. In the case where the finder uses the inbuilt App function of scanning lost items QR Codes, the App notify the system and give detailed information of the finder's location and contact details as provided by the finder during app setup. When the system gets a notification, it declares the item in the database found and pending further verification and then send out a notification to the owner of the lost item.

The search function presented by the system was keyword based where the person can search for any unique keywords or numbers that their documents had to get the results. The results shall also indicate the location found and the pick-up point. These pick-up points are public places such as hotels, shops, police stations, chief's camps and other public offices. The mobile app reporting interface have a location tracking system inbuilt to give exact coordinates or place IDs from Google maps.

The system raises some identity theft concerns by being public and therefore I have implement an obfuscation technique such as blurring images of the lost personal identifying documents if they are of a sensitive nature. These include alien IDs, national IDs, passports and ATM cards.

1.5.OBJECTIVES

The system has one major objective and a few minor objectives as indicated below

1.5.1. Main Objective

To design, develop and implement an efficient, user-friendly, mobile and web-based lost identifiable items recovery system.

1.5.2. Specific Objectives

The project aims to achieve the objectives as follows:

- \checkmark To analyze how the current systems, solve lost items recovery
- ✓ To develop a system that will allow users to upload lost identifiable item
- ✓ To provide a secure mobile and web-based platform that allows owners of lost item easily search and recovery them.

1.6.RESEARCH QUESTIONS

Achieving my objectives demands the following research questions to be raised and answered:

These questions need to match with the objectives above i.e. answer to question one should be objective one

- ✓ How does the current systems help users in recovering their lost identifiable items?
- ✓ What system would enable users to upload details of their lost identifiable item?
- ✓ What system would provide a secure mobile and web-based platform that allows owners of lost item easily search and recovery them?

1.7.JUSTIFICATION

The benefits brought about by the successful implementation of the system go a long way to justify the system development and deployment. They include:

- **Increased efficiency** by eliminating paper work such as re-application and printouts/photocopies of the required corresponding documents e.g. forms through online operations.
- **Rampant Identity** theft and mugging cases in an area can be reduced by the system aiding the police in **identification of hotspot areas** where they can plant more security to catch offenders in the act
- Significantly **reduced time and capital wastage** through re-application, transit and waiting.

1.8.LIMITATIONS

The system has these limitations arising from the current state of affairs

- The system demands that user must be either computer literate or savvy with use of mobile technologies such as Apps, SMS and USSD & also have access to the internet on their device to use the mobile and web apps.
- The system only helps individuals recover their found items i.e. it only works for those whose items were identifiable (have a QR sticker) and have been recovered since it won't have

1.9.SCOPE

The system works only on the following two platforms to the extent of its use.

1.9.1. Web Platform

The WebApp present a simplistic responsive user-friendly UI that has easy controls for searching, reporting and posting details about lost items. It also gives some statistics on the current number of lost and recovered items and their categories. The reporting form allow a user to enter the details of a lost item that is identifiable and the user's phone number which was used to notify them when it is reported as found. Links to the more convenient mobile app also be presented.

1.9.2. Mobile Platform

On the mobile, users were able to utilize USSD to query the database and receive notifications on their phone. These cost the user the standard operating fees set by service providers. But if a user has the mobile app installed the notifications and features are accessible all from the app. Also, users with smartphones can use the far more efficient QR Code approach to registering and reporting found items.

1.10 CONCEPTUAL FRAMEWORK



Figure 1 Conceptual Framework

LITERATURE REVIW

2.1.The Traditional Approach

Case Study: The Kenyan Government Lost Identity Documents Renewal System

Losing one's items especially identity cards leads to a scrupulous process of going to the issuer of identity cards and acquiring new ones. This process is still plagued by rampant corruption and proves to be a headache for the victim of such losses. The personnel at these issuance offices openly request for a bribe in order to expedite the process of your renewal after re-application. This system is not only very resource intensive but also criminal at its core. It takes advantage of the individuals who have lost their identity documents.

Sometimes, the lost items and cards may have even been found by strangers somewhere but without a means to reach the owner and notify them, they end up being tossed aside again. The National Registration Board in Kenya – who are in charge of the provision and re-provision of lost national documents claim that it takes approximately 4 to 6weeks on the average to produce an identity card. [6] The process as they outline requires the following:

- A police report copies stating the nature of the lost document
- A copy of the applicant's passport where one is present

- An original copy of the birth certificate of said applicant
- A copy of the ID that was lost if any

Then, with this at hand, the applicant pays up a further Ksh. 300. A series of very tantalizing questions and forms to be filled follow to act as proof of identity. As is now evident, there is a huge failure in the system in terms of streamlining.

2.2. Current Approaches

Case Study 1: Web-Based Identity Documents Recovery Systems

There are currently only two active online systems in the country for recovery of such items. The first company, called LostAndFoundDC, has a major downside in that their services are not free as they are a private company. They charge storage and transport fees for your recovered documents. Also, only documents can be recovered by their system [7]. The second company is called Patacom Kenya and due to poor marketing and exposure their database has less than 20 reported lost items that are not secured at all. [8] This makes Patacom an especially good target for identity thieves and serial criminals who can lure an unknowing person to themselves in the guise of giving them back their lost items.

Case Study 2: Kennedy Kamanja Stickers System

Another system in place is the innovation by one Ken Kamanja who invented a sticker printer which prints out small yellow stickers with the person's details that one then sticks onto the personally identifiable documents. [9] The main drawback of this system is that it exposes your phone number to strangers and also can be used by the malicious people to ask for ransom for your lost items recovery

Case Study 3: Kahunyo's Lost Reported National Identity Centralized Database

Yet another ambitious innovator attempted to tackle this in 2012 by creating a central database fed by data from all police stations, chief camps and government institutions about these lost IDs. Back then it was using Safaricom's interface and in in its beta version costed Ksh. 15 per SMS. The company was in partnership with Safaricom and Kenyaweb, and Kahunyo, owner confirmed that the majority of their profits would be generated by selling advertising directly on their SMS text messages. [10]. This one too failed due to the expensive and profit oriented nature it was presenting.

Case Study 4: Kamuzu Banda GSM Technology Approach

Kamuzu Banda, a young inventor who even managed to attract the attention of the press [2] was able to employee the use of mobile USSD which is a feature of all GSM devices. Back when it was functional, the application allowed you to enter a USSD code on your mobile, like *320#, upon which you would receive prompts for your name, phone number and town where the ID was lost. The information was then transmitted back to a central database which would either return a 'found' notification or 'report' your item as lost for future notification when it would hopefully be found.

Case Study 5: Zipate Lost Identity Documents Recovery Business

Yet another ambitious project funded by KYDF in 2014 a group of three Kenyan youth who used the funding to create a website http://zipate.com/. This had major drawbacks to its interface and presentation as it did not offer any clear methods of posting or reporting the items. Similarly, the way in which it presented results for a search allowed anyone to claim to be the owner of the item. All details including clear photos of the person who posted and the posted item would be presented and this led to a huge security leak as noted by Stanley

Githinjii's Designing Lost and Found Web Applications Based on Trusted Third Party (TTP) Model paper. [11]. The three also employed a business mindset where they would pay people a fee for bringing in lost identifiable items and then recoup by charging the owners a hefty fee to give them back their items. Zipate has since died out and their website is now deregistered.

Case Study 6: The Informal Social Media Approach

Finally, the more informal approach to this problem has been presented on social media websites such as Facebook and Twitter. Here, individuals create groups where they can post details of their lost items as well as items they found in the hope that someone notice something familiar and make a claim. This approach is however plagued by high insecurity as posting personal details and items on social media can lead to identity theft and fraudulent acts. This is also tackled by Stanley Githinji in his assessment. [11] . the approach also fails in having no centralized datastore/repository for said items as well as no standardized means of reporting, searching and retrieving one's items.

2.3.Identified Gaps

- ✓ Lack of capabilities to aid in recovery of more items other than just identity documents
- ✓ Lack of support for preemptive measures such as QR Code Sticker item tagging
- ✓ Lack of a good mass information strategy to ensure the people the system is intended to benefit get to use the system in the first place
- ✓ Lack of a multi-platform approach to tackling the problem of lost items recovery to ensure everyone is included and can use the system

SYSTEM DEVELOPMENT METHODOLOGY

3.1.System Development Life Cycle: Waterfall

3.1.1. Analysis

The developer analyzes the requirements, and fully understands the problems. This is a research phase that includes no building. For the project, analysis was conducted on the current systems failures and strengths. This way, the required improvements are better understood and captured for later implementation. Further analysis also be conducted on the problem definitions to clearly understand what domains to target most. Finally, the analysis of platform dependent requirements was undertaken. This includes a proper definition of the intended system's target platforms and dependencies such as Android version, browser support, licensing, and standards.

3.1.2. Design

The developer designs a technical solution to the problems set out by the product requirements, including scenarios, layouts and data models. This phase is usually accompanied by documentation for each requirement, which enables other members of the team to review it for validation. The project demands that two design sub phases be applied.

3.1.2.1.Logical Design

Logical design characteristically looks at the intended system from a logical perspective without considering physical requirement. The project need a logical design that models the flow of data and information through

the system from input to output. All along this data needs to undergo thorough validation and refining to protect both the system as well as its users. Logical design also models the security checks that the system is using as well as the formats for all data items in the system.

3.1.2.2.Physical Design

The physical design is concerned with how the physical architecture of the entire system will interact to achieve its objectives. It models the user interfaces, the server architecture and the database models and architecture too. The system spans two popular platforms and must be designed in such a way that it works seamlessly across both mobile and web. Transparency schemes are of uttermost concern here.

3.1.3. Implementation

Once the designs are deemed to be viable, technical implementation begins. Implementing the project is the toughest part as all the coding is done in this phase. Being that the project serves both mobile and web platforms, coding took take place in three phases two of which can run concurrently.

3.1.4. Database coding phase

The backend relies on a robust implementation of MySQL database. The database is relational in architecture and host tables which can be abstracted into views for the front end as needed and by access level specifications. All the system's data is stored and processed here

3.1.5. Web and Mobile Coding

This are the two key areas of coding that consume majority of the coding time. The project is coded in HTML + JavaScript + PHP for the web platform and React JavaScript XML (JSX) + Java. Android for the mobile platform app. Coding the two platforms was done in parallel to ensure transparency in design which translate to a uniform UI.

3.1.6. Testing

Upon completion of full implementation, testing needs to occur before the system can be launched for public consumption. As the developer and tester, I use the design documents, personas and user case scenarios to run comprehensive tests on all builds both during coding (Components testing) and on the finished applications. I also solicit help and assistance from my fellow colleagues whenever I require user input for my scenarios and user response to aid in the fine tuning of the system.

3.2. PRIMARY DATA COLLECTION

3.2.1. Interviews

Interview allow me to collect data from individuals who are currently the users of existing systems. These users include students and general public who were the ones affected by the system. Interviews can be time and resource intensive and incur costs sometimes. However, they provide excellent qualitative information. They allow me to better understand key areas of misunderstanding that may bring along hindrances to the project implementation. The key advantage of these interviews is their flexibility and the interviewer can use open ended questions to gain deeper insight into the problem domain.

3.2.2. Direct Observation

Observing the case on the ground is the second approach I aim to employee in my primary data collection. This entail visiting popular joints where most lost items are left by well-meaning strangers. I also intend to do

a count and estimation of the number of items reported or posted as lost in these places. Many individuals who have lost their items either do so by careless misplacement, mugging, neglect, accidental drop etc. all these are actions that are hard to observe first hand but can be inferred from the repercussions.

3.3. SECONDDARY DATA COLLECTION

3.3.1. Online Resources and Reference Materials

The internet is among the best sources of widely ranging information. It is especially suited for my project because the system utilizes the internet as a platform to be realized. Many materials on the subject have been posted on blogs, social media and reports. With regards to the proposed project, I intend to acquire quantitative as well as qualitative information on the subject matter which I use for statistical reference in the process of building the system. Such information is key in determining the traffic requirements, HCI specifications that are befitting, feasibility and other requirement specifications for making the system a reality.

SYSTEM DESIGN

4.1 Data flow diagram



Figure 1 Data Flow Diagram of the system

4.2. SYSTEM REQUIREMENTS SPECIFICATIONS

The system is running across two platforms as indicated in the scope of this document. Therefore, for each platform there were some default minimums that must be met in order for the system to perform as expected. These are outlined below in order of platform.

4.2.1 Mobile Platform

Minimum System Requirements

CATEGORY	REQUIRED	DESCRIPTION
OS	Android Operating System	Platform OS
Web Engine	Android System WebView	Displaying Webpages
FRAMEWORK	Google Play Services	Native API Library Access
SoC	Single/Multi Core + 1GHz and	System on a Chip
	Above	(Microprocessor)
RAM	512MB and Above	Operating System Memory
CAMERA	5MP and Above	Scanning QR Codes
GSM	SMS and USSD	Data Free System Access
MOBILE DATA	EDGE, WCDMA (HSPA or	WAN Online Functionality
	HSPA+), LTE	

4.2.2 Web Platform

4.2.2.1 Minimum System Requirements

CATEGORY	REQUIRED	DESCRIPTION
OS	Apple iOS, Android, MS Windows,	Web browser host Platform
	Linux, Mac	OS
Web Browser	Above Chrome 12, Firefox 14, IE 7,	Displaying Webpages
	Safari 2	
JavaScript	Supported and Enabled	Front end Logic execution
Processing	Single/Multi Core + 1GHz and	Microprocessor
	Above	
RAM	512MB and Above	Host Operating System
		Memory
Internet Access	1MBps Downlink, 512KBps Uplink	WAN Online Functionality

4.3 USER REQUIREMENTS

The system has no special user requirements and can be used by anyone with access to the technologies as specified in the system requirements section above. However, the user should be ethical in utilizing the system and at least be literate in the English language.

4.4 DATABASE DESIGN

J	Structure	SQL	Search	n 🗍 Quer	y 🛃 Ex	port	📑 Import	🥜 Ор	erations	ැසි Routi	nes			
	Table 🔺		Action						Rows 😡	Туре	Collat	tion	Size	Overhead
	Items_Bank	Card	Browse	M Structure	🖎 Search	📑 i Inse	rt 扁 Empty	Drop	1	MyISAM	latin1_	swedish_ci	1 KiB	-
	Items_Certif	icate	Browse	Y Structure	👒 Search	🛃 inse	rt 扁 Empty	\ominus Drop		MyISAM	latin1_	swedish_ci	1 KiB	-
	Items_Drivir	ng_Licence	Browse	<u>Structure</u>	🍕 Search	🛃 inse	rt 🚍 Empty	\ominus Drop		MyISAM	latin1_	swedish_ci	1 KiB	-
	Items_Gadg	et	Browse	🖌 Structure	👒 Search	👫 Inse	rt 💻 Empty	Orop		MyISAM	latin1_	swedish_ci	1 KiB	-
	Items_Identi	ity_Card	Browse	M Structure	👒 Search	🕌 Inse	rt 💻 Empty	Drop		MyISAM	latin1_	swedish_ci	1 KiB	-
	Items_Log_I	Book	Browse	🖌 Structure	👒 Search	🕌 Inse	rt 💻 Empty	🔵 Drop	1	MyISAM	latin1_	swedish_ci	1 KiB	-
	Items_NHIF		Browse	M Structure	👒 Search	🕌 Inse	rt 🗮 Empty	Drop		MyISAM	latin1_	swedish_ci	1 KiB	-
	Items_NSSF	:	Browse	🖌 Structure	👒 Search	👫 Inse	rt 💻 Empty	Orop	1	MyISAM	latin1_	swedish_ci	1 KiB	-
	Items_Other	r	Browse	M Structure	👒 Search	👫 Inse	rt 💻 Empty	Drop		MyISAM	latin1_	swedish_ci	1 KiB	-
	Items_Pass	port	Browse	🖌 Structure	👒 Search	👫 Inse	rt 💻 Empty	Drop	1	MyISAM	latin1_	swedish_ci	1 KiB	-
	Items_Textil	е	Browse	M Structure	👒 Search	🕌 Inse	rt 💻 Empty	Drop		MyISAM	latin1_	swedish_ci	1 KiB	-
	Items_Title_	Deed	Browse	V Structure	👒 Search	👍 Inse	rt 💻 Empty	🔵 Drop	1	MyISAM	latin1_	swedish_ci	1 KiB	-
	QR_Store		Browse	M Structure	👒 Search	🕌 Inse	rt 🗮 Empty	Drop		MyISAM	latin1_	swedish_ci	1 KiB	-
	recovR_Use	rs	Browse	🖌 Structure	👒 Search	👫 Inse	rt 💻 Empty	Orop	1	MyISAM	latin1_	swedish_ci	1 KiB	-
	14 tables		Sum							MyISAM	latin1	_swedish_ci	14 KiB	0 B

Figure 2 Whole Database View

Figure 3 Bank Cards Table

SELECT * FROM `Items_Bank_Card` LIMIT 0, 30

🗌 Pro

#	Name	Туре	Collation	Attributes	Null	Default	Extra	Action			
1	counter	int(4)			No	None	AUTO_INCREMENT	🥜 Change	Drop	Browse	distinct values
2	item_name	varchar(30)	latin1_swedish_ci		No	None		🥜 Change	Drop	Browse	distinct values
3	owner_phone	varchar(20)	latin1_swedish_ci		No	None		🥜 Change	Drop	Browse	distinct values
4	owner_token	varchar(100)	latin1_swedish_ci		No	None		🥜 Change	Drop	Browse	distinct values
5	item_unique_number	varchar(50)	latin1_swedish_ci		No	None		🥜 Change	Drop	Browse	distinct values
6	item_descrition	varchar(2000)	latin1_swedish_ci		No	None		🥜 Change	Drop	Browse	distinct values
7	item_pic	varchar(1500)	latin1_swedish_ci		No	None		🥜 Change	Drop	Browse	distinct values
8	item_type	varchar(20)	latin1_swedish_ci		No	None		🥜 Change	Drop	Browse	distinct values
9	item_status	varchar(20)	latin1_swedish_ci		No	None		🥜 Change	Drop	Browse	distinct values
10	item_qr	varchar(250)	latin1_swedish_ci		No	None		🥜 Change	Drop	Browse	distinct values

Figure 4 certificates table

SELECT * FROM `Items_Certificate` LIMIT 0 , 30

Prof

#	Name	Туре	Collation	Attributes	Null	Default	Extra	Action				
1	<u>counter</u>	int(4)			No	None	AUTO_INCREMENT	🥜 Change	Drop	Browse	distinct value	as _é
2	item_name	varchar(30)	latin1_swedish_ci		No	None		🥜 Change	Drop	Browse	distinct value	es 🧃
3	owner_phone	varchar(20)	latin1_swedish_ci		No	None		🥜 Change	Drop	Browse	distinct value	es
4	owner_token	varchar(100)	latin1_swedish_ci		No	None		🥜 Change	Drop	Browse	distinct value	es 🧃
5	item_unique_number	varchar(50)	latin1_swedish_ci		No	None		🥜 Change	Drop	Browse	distinct value	es
6	item_descrition	varchar(2000)	latin1_swedish_ci		No	None		🥜 Change	Drop	Browse	distinct value	es 🧃
7	item_pic	varchar(1500)	latin1_swedish_ci		No	None		🥜 Change	Drop	Browse	distinct value	es 🧃
8	item_type	varchar(20)	latin1_swedish_ci		No	None		🥜 Change	Drop	Browse	distinct value	es 🧃
9	item_status	varchar(20)	latin1_swedish_ci		No	None		🥜 Change	Drop	Browse	distinct value	es
10	item_qr	varchar(250)	latin1_swedish_ci		No	None		🥜 Change	Orop	Browse	distinct value	es 🕜

Figure 5 Driving License Table

SELECT * FROM `Items_Driving_Licence` LIMIT 0 , 30

#	Name	Туре	Collation	Attributes	Null	Default	Extra	Action				
1	<u>counter</u>	int(4)			No	None		🥜 Change	Drop	Browse	distinct value	s
2	item_name	varchar(30)	latin1_swedish_ci		No	None		🥜 Change	Drop	Browse	distinct value	s
3	owner_phone	varchar(20)	latin1_swedish_ci		No	None		🥜 Change	Drop	Browse	distinct value	s
4	owner_token	varchar(100)	latin1_swedish_ci		No	None		🥜 Change	Drop	Browse	distinct value	s
5	item_unique_number	varchar(50)	latin1_swedish_ci		No	None		🥜 Change	Drop	Browse	distinct value	s
6	item_descrition	varchar(2000)	latin1_swedish_ci		No	None		🥜 Change	Drop	Browse	distinct value	;s
7	item_pic	varchar(1500)	latin1_swedish_ci		No	None		🥜 Change	Drop	Browse	distinct value	;s
8	item_type	varchar(20)	latin1_swedish_ci		No	None		🥜 Change	Drop	Browse	distinct value	s
9	item_status	varchar(20)	latin1_swedish_ci		No	None		🥜 Change	Drop	Browse	distinct value	s
10	item_qr	varchar(250)	latin1_swedish_ci		No	None		🥜 Change	Orop	Browse	distinct value	s

Figure 6 Gadgets Table

SELECT * FROM `Items_Gadget` LIMIT 0, 30

#	Name	Туре	Collation	Attributes	Null	Default	Extra	Action
1	counter	int(4)			No	None	AUTO_INCREMENT	🥜 Change 🤤 Drop 📺 Browse distinct values
2	item_name	varchar(30)	latin1_swedish_ci		No	None		🖉 Change 🤤 Drop 📊 Browse distinct values
3	owner_phone	varchar(20)	latin1_swedish_ci		No	None		🖉 Change 🤤 Drop 📺 Browse distinct values
4	owner_token	varchar(100)	latin1_swedish_ci		No	None		🖉 Change 🤤 Drop 📺 Browse distinct values
5	item_unique_number	varchar(50)	latin1_swedish_ci		No	None		🖉 Change 🤤 Drop 📺 Browse distinct values
6	item_descrition	varchar(2000)	latin1_swedish_ci		No	None		🖉 Change 🤤 Drop 📊 Browse distinct values
7	item_pic	varchar(1500)	latin1_swedish_ci		No	None		🖉 Change 🤤 Drop 📺 Browse distinct values
8	item_type	varchar(20)	latin1_swedish_ci		No	None		🖉 Change 🤤 Drop 📺 Browse distinct values
9	item_status	varchar(20)	latin1_swedish_ci		No	None		🖉 Change 🤤 Drop 📺 Browse distinct values
10	item_qr	varchar(250)	latin1_swedish_ci		No	None		🖉 Change 🥥 Drop 📊 Browse distinct values

Figure 7 Identity Cards Table

SELECT * FROM `Items_Identity_Card` LIMIT 0, 30

Name Collation Attributes Null Default Extra Туре Action No AUTO_INCREMENT & Change O Drop I Browse distinct values 1 counter int(4) None 2 item_name varchar(30) latin1_swedish_ci None No 3 owner_phone 🖉 Change 🥥 Drop 📰 Browse distinct values varchar(20) latin1_swedish_ci No None 4 owner_token varchar(100) latin1_swedish_ci No None 5 item_unique_number varchar(20) None latin1_swedish_ci No 6 item_descrition varchar(6000) latin1_swedish_ci No None 🖉 Change 🥥 Drop 📰 Browse distinct values 7 item pic varchar(1500) latin1 swedish ci No None 🖉 Change 🥥 Drop 📄 Browse distinct values 8 item_type varchar(20) latin1 swedish ci No None 9 item_status varchar(20) latin1_swedish_ci No None 🖉 Change 🥥 Drop 📰 Browse distinct values 10 item_qr varchar(250) latin1_swedish_ci No None Change Orop Browse distinct values

📃 Pro

🗌 Pro

Figure 8 Log Books Table

SELECT * FROM `Items_Log_Book` LIMIT 0 , 30

#	Name	Туре	Collation	Attributes	Null	Default	Extra	Action
1	<u>counter</u>	int(4)			No	None	AUTO_INCREMENT	🖉 Change 🥥 Drop 📄 Browse distinct values
2	item_name	varchar(30)	latin1_swedish_ci		No	None		🖉 Change 🥥 Drop 🔲 Browse distinct values
3	owner_phone	varchar(20)	latin1_swedish_ci		No	None		
4	owner_token	varchar(100)	latin1_swedish_ci		No	None		🖉 Change 🥥 Drop 📄 Browse distinct values
5	item_unique_number	varchar(50)	latin1_swedish_ci		No	None		
6	item_descrition	varchar(2000)	latin1_swedish_ci		No	None		
7	item_pic	varchar(1500)	latin1_swedish_ci		No	None		🖉 Change 🥥 Drop 📄 Browse distinct values
8	item_type	varchar(20)	latin1_swedish_ci		No	None		
9	item_status	varchar(20)	latin1_swedish_ci		No	None		
10	item gr	varchar(250)	latin1 swedish ci		No	None		Change Conception Change Conception Concepti Conception Conception Conception Concept

Figure 9 NHIF Table

SELECT * FROM 'Items_NHIF' LIMIT 0 , 30

#	Name	Туре	Collation	Attributes	Null	Default	Extra	Action
1	<u>counter</u>	int(4)			No	None	AUTO_INCREMENT	🥜 Change 🤤 Drop 📺 Browse distinct values
2	item_name	varchar(30)	latin1_swedish_ci		No	None		🖉 Change 🥥 Drop 📺 Browse distinct values
3	owner_phone	varchar(20)	latin1_swedish_ci		No	None		
4	owner_token	varchar(100)	latin1_swedish_ci		No	None		🖉 Change 🥥 Drop 📄 Browse distinct values
5	item_unique_number	varchar(50)	latin1_swedish_ci		No	None		🖉 Change 🥥 Drop 📺 Browse distinct values
6	item_descrition	varchar(2000)	latin1_swedish_ci		No	None		🖉 Change 🥥 Drop 📺 Browse distinct values
7	item_pic	varchar(1500)	latin1_swedish_ci		No	None		🖉 Change 🥥 Drop 📺 Browse distinct values
8	item_type	varchar(20)	latin1_swedish_ci		No	None		🖉 Change 🥥 Drop 📺 Browse distinct values
9	item_status	varchar(20)	latin1_swedish_ci		No	None		🖉 Change 🥥 Drop 📺 Browse distinct values
10	item_qr	varchar(250)	latin1_swedish_ci		No	None		🔗 Change 🥥 Drop 📺 Browse distinct values

🗌 Pro

🗌 Pr

Figure 10 NSSF Table

SELECT * FROM `Items_NSSF` LIMIT 0 , 30

#	Name	Туре	Collation	Attributes	Null	Default	Extra	Action	
1	<u>counter</u>	int(4)			No	None	AUTO_INCREMENT	🖉 Change 🥥 Drop	Browse distinct values
2	item_name	varchar(30)	latin1_swedish_ci		No	None		🖉 Change 🥥 Drop	Browse distinct values
3	owner_phone	varchar(20)	latin1_swedish_ci		No	None		🖉 Change 🥥 Drop	Browse distinct values
4	owner_token	varchar(100)	latin1_swedish_ci		No	None		🖉 Change 🥥 Drop	Browse distinct values
5	item_unique_number	varchar(50)	latin1_swedish_ci		No	None		🖉 Change 🥥 Drop	Browse distinct values
6	item_descrition	varchar(2000)	latin1_swedish_ci		No	None		🖉 Change 🥥 Drop	Browse distinct values
7	item_pic	varchar(1500)	latin1_swedish_ci		No	None		🖉 Change 🥥 Drop	Browse distinct values
8	item_type	varchar(20)	latin1_swedish_ci		No	None		🖉 Change 🥥 Drop	Browse distinct values
9	item_status	varchar(20)	latin1_swedish_ci		No	None		🖉 Change 🥥 Drop	Browse distinct values
10	item gr	varchar(250)	latin1 swedish ci		No	None		🧷 Change 🖨 Drop	Browse distinct values

Figure 11 Other Items Table

SELECT * FROM `Items_Other` LIMIT 0 , 30

Prot

Prot

#	Name	Туре	Collation	Attributes	Null	Default	Extra	Action	
1	counter	int(4)			No	None	AUTO_INCREMENT	🥜 Change 🥥 Drop 📻 Brow	se distinct values ₍
2	item_name	varchar(30)	latin1_swedish_ci		No	None		🥜 Change 🥥 Drop 📄 Brow	vse distinct values _t
3	owner_phone	varchar(20)	latin1_swedish_ci		No	None		🥜 Change 🥥 Drop 📻 Brow	vse distinct values _u
4	owner_token	varchar(100)	latin1_swedish_ci		No	None		🥜 Change 🥥 Drop 📄 Brow	vse distinct values _c
5	item_unique_number	varchar(50)	latin1_swedish_ci		No	None		🥜 Change 🥥 Drop 📺 Brow	vse distinct values _u
6	item_descrition	varchar(2000)	latin1_swedish_ci		No	None		🥜 Change 🥥 Drop 📄 Brow	vse distinct values _c
7	item_pic	varchar(1500)	latin1_swedish_ci		No	None		🥜 Change 🥥 Drop 📺 Brow	vse distinct values _u
8	item_type	varchar(20)	latin1_swedish_ci		No	None		🥜 Change 🥥 Drop 📄 Brow	vse distinct values _c
9	item_status	varchar(20)	latin1_swedish_ci		No	None		🥜 Change 🥥 Drop 📰 Brow	se distinct values _t
10	item_qr	varchar(250)	latin1_swedish_ci		No	None		🥜 Change 🥥 Drop 🔲 Brow	se distinct values

© Maina, Mwai, Njomo

Figure 12 Passports Table

S F L	ele Roi Mi1	CT* 1 `Items_Passport` 1 0 , 30												
													(🗌 Pro
	#	Name	Туре	Collation	Attributes	Null	Default	Extra		Action				
	1	counter	int(4)			No	None	AUTO	INCREMENT	🥜 Change	Drop	Brows	e distinct va	alues
	2	item_name	varchar(30)	latin1_swedish_ci		No	None			🥜 Change	Drop	Brows	e distinct va	alues
	3	owner_phone	varchar(20)	latin1_swedish_ci		No	None			🥜 Change	Drop	Brows	e distinct va	alues
	4	owner_token	varchar(100)	latin1_swedish_ci		No	None			🥜 Change	Drop	Brows	e distinct va	alues
	5	item_unique_number	varchar(50)	latin1_swedish_ci		No	None			🥜 Change	Drop	Brows	e distinct va	alues
	6	item_descrition	varchar(2000)	latin1_swedish_ci		No	None			🥜 Change	Drop	Brows	e distinct va	alues
	7	item_pic	varchar(1500)	latin1_swedish_ci		No	None			🥜 Change	Drop	Brows	e distinct va	alues
	8	item_type	varchar(20)	latin1_swedish_ci		No	None			🥜 Change	Drop	Brows	e distinct va	alues
	9	item_status	varchar(20)	latin1_swedish_ci		No	None			🥜 Change	Drop	Brows	e distinct va	alues
	10	item_qr	varchar(250)	latin1_swedish_ci		No	None			🥜 Change	Orop	Brows	e distinct va	alues

Figure 13 QR Codes Store table

SELECT * FROM 'QR_Store' LIMIT 0 , 30

#	Name	Туре	Collation	Attributes	Null	Default	Extra	Action
1	<u>qr_counter</u>	int(4)			No	None	AUTO_INCREMENT	🖉 Change 🤤 Drop 📊 Browse distinct values 🔊 I
2	item_id	varchar(50)	latin1_swedish_ci		Yes	NULL		🖉 Change 🥥 Drop 📊 Browse distinct values 🔑 I
3	item_table	varchar(30)	latin1_swedish_ci		No	None		🖉 Change 🥥 Drop 📺 Browse distinct values 🔑 I
4	scan_counter	int(4)			No	None		🖉 Change 🥥 Drop 📺 Browse distinct values 🔑 I

Figure 14 Textiles Table

SELECT * FROM `Items_Textile` LIMIT 0 , 30

📃 Profilin

#	Name	Туре	Collation	Attributes	Null	Default	Extra	Action
1	counter	int(4)			No	None	AUTO_INCREMENT	🥜 Change 🤤 Drop 📃 Browse distinct values 🔌
2	item_name	varchar(30)	latin1_swedish_ci		No	None		${\mathscr O}$ Change \bigcirc Drop 📰 Browse distinct values ${\mathscr O}$
3	owner_phone	varchar(20)	latin1_swedish_ci		No	None		🖉 Change 🥥 Drop 📺 Browse distinct values 🔌
4	owner_token	varchar(100)	latin1_swedish_ci		No	None		${\mathscr O}$ Change \bigcirc Drop 📄 Browse distinct values ${\mathscr O}$
5	item_unique_number	varchar(50)	latin1_swedish_ci		No	None		🖉 Change 🥥 Drop 📰 Browse distinct values 🔌
6	item_descrition	varchar(2000)	latin1_swedish_ci		No	None		${\mathscr O}$ Change \bigcirc Drop 📰 Browse distinct values ${\mathscr O}$
7	item_pic	varchar(1500)	latin1_swedish_ci		No	None		🖉 Change 🥥 Drop 📺 Browse distinct values 🔌
8	item_type	varchar(20)	latin1_swedish_ci		No	None		${\mathscr O}$ Change \bigcirc Drop 🔲 Browse distinct values ${\mathscr O}$
9	item_status	varchar(20)	latin1_swedish_ci		No	None		🖉 Change 🥥 Drop 📰 Browse distinct values 🔌
10	item_qr	varchar(250)	latin1_swedish_ci		No	None		🖉 Change 🥥 Drop 📺 Browse distinct values 🔑

Figure 15 Title Deeds Table

SELECT * FROM `Items_Title_Deed` LIMIT 0, 30

📃 Profilin

#	Name	Туре	Collation	Attributes	Null	Default	Extra	Action
1	counter	int(4)			No	None	AUTO_INCREMENT	🥜 Change 🥥 Drop 📺 Browse distinct values 🔊 I
2	item_name	varchar(30)	latin1_swedish_ci		No	None		🖉 Change 🥥 Drop 📺 Browse distinct values 🔑 I
3	owner_phone	varchar(20)	latin1_swedish_ci		No	None		🖉 Change 🥥 Drop 📺 Browse distinct values 🔑 I
4	owner_token	varchar(100)	latin1_swedish_ci		No	None		🖉 Change 🥥 Drop 📺 Browse distinct values 🔑 I
5	item_unique_number	varchar(50)	latin1_swedish_ci		No	None		🖉 Change 🥥 Drop 📺 Browse distinct values 🔑 I
6	item_descrition	varchar(2000)	latin1_swedish_ci		No	None		🖉 Change 🥥 Drop 📺 Browse distinct values 🔑 I
7	item_pic	varchar(1500)	latin1_swedish_ci		No	None		🖉 Change 🥥 Drop 📺 Browse distinct values 🔑 I
8	item_type	varchar(20)	latin1_swedish_ci		No	None		🖉 Change 🥥 Drop 📺 Browse distinct values 🔑 I
9	item_status	varchar(20)	latin1_swedish_ci		No	None		🖉 Change 🥥 Drop 📺 Browse distinct values 🔑
10	item_qr	varchar(250)	latin1_swedish_ci		No	None		🖉 Change 🥥 Drop 📄 Browse distinct values 🔑 I

Figure 16 Users Table

SFL	EL RO MI	ECT MirecovR_U T0:30	Users'	ĸ					
	#	Name	Туре	Collation	Attributes	Null	Default	Extra	Action
8	1	<u>counter</u>	int(4)			No	None	AUTO_INCREMENT	🥜 Change 🤤 Drop 🔟 Browse distinct values 🔊 F
D	2	firebas_key	varchar(100)	latin1_swedish_ci		No	None		🥜 Change 🤤 Drop 🔄 Browse distinct values 🄑 F
	3	user_token	int(100)			No	None		🥜 Change 🤤 Drop 📰 Browse distinct values 🔑 F

4.5 INPUT DESIGN

4.5.1 Login User Interfaces



4.5.2. Registration Forms





Figure 28 Contact Us Page for the web app	Home Download Report About Us Contact Us Login or Register
Name Email Message	Phone Contact Details If you have any ideas or questions, please feel free to contact us using our contact data or this form. Phone : ↓ +254 701 933 075 Email : ☑ info@recovR.com
Send Message	Address :143-10300 Kerugoya



DASHBOARD		Your Profile					
		Company	Username	Email address			
PROFILE		Stackstech Inc	DunMa	danmainor@outlook.com			
ITEMS		First Name	Last Nar	ne			
		Dunford	Maina	t			
<	Ounford Mainor	r Address					
	If you are going through hell, going.	1234, WestStreet	1234, WestStreet				
	Sir Winston Churchill	County	City	Postal Code			
	5 0	23 Kirinyaga	Kirinyaga	00253			
	items Lost	Scons About Me					
BACK TO HOME		A biography, or sin the basic facts like	nply bio, is a detailed description of a p education, work, relationships, and de	erson's life. It involves more than just ath: it portrays a person's experience			





SYSTEM TESTING AND IMPLEMENTATION

5.1.UNIT TESTING

Unit involves testing software with a small piece of source code (unit, component, and/or function) of the same software. During performing tests, some hypotheses would be made, and the testing would then determine if true or false. This way, the developer was able to check whether a unit behaves as intended or whether a unit corresponds to the design specifications.

All the sources used in unit testing were created by the developer as a part of software development. The following unit tests were performed to ascertain functionality.



Table 1Test Cases for Unit Testing

5.2.INTERFACE TESTING

Interface Testing was performed to evaluate whether systems or components pass data and control correctly to one another. It was also used to verify if all the interactions between these modules are working properly and errors are handled properly. To perform the interface tests, the developer created a checklist that outlined all the functional requirements of the system and the various test cases to assess them.

Functional requirement	Description
FR01	Registering and Authentication for users of the application
FR02	Adding Items into the system
FR03	Searching the database for items
FR04	Reporting an item as lost

Table 2 Functional requirements descriptions

5.2.1. TEST CASES

The table below shows how each of the functional requirements were assessed using Test Cases.

Table 3 Test Case One Details and Results

TEST CASE (TC#) FUNCTIONAL TEST REQUIREMENT		TEST NAME	AME TEST DESCRIPTION		E TEST ENVIRO	NMENT	
TC1	FR1	Authentication	Verify and authenticate user using email and password	recovR	Windows 10 Pro Wamp(Apache) Android 6.0 Mar	Windows 10 Pro, 1TB HDD, 8GB RAM, Wamp(Apache) Server, MySQL Server Android 6.0 Marshmallow, 3GB RAM	
Action Performed	Action's output	Valio	l Input	Inv	alid Input	Result	
Enter email addressNavigate to user's all itemsand passwordsection		all items Well	formatted email address and	password Inc	orrect Password	Pass Test	

Table 4 Test Case Two Details and Results

TEST CASE (TC#) FUNCTIONAL TEST N REQUIREMENT		TEST NAME	TEST DESCRIPTION	SOFTWARE	TEST ENVIRONMENT	
TC2	FR2	Insertion	Add Items into the database	recovR	Android 6.0 Marshmallow, 3GB RAM	
Action Performed	Action's output	Valio	l Input	Inval	id Input	Result
Added a Travelling	Navigate to user's	all items All it	em details including name, de	scription, None		Pass Test
bag item	section and display item on the list	added photo	o, QR Code, Category			

Table 5 Te	est Case Th	ree Details an	d Results
------------	-------------	----------------	-----------

TEST CASE (TC#)	FUNCTIONAL REQUIREMENT	TEST NAM	IE TEST DESCRIPTION	SOFTWARE	TEST ENVIRONM	IENT
TC3	FR3	Search	Search an Item either by	recovR	Windows 10 Pro, 11	B HDD, 8GB RAM,
			unique key(text) or by Ql	2	Wamp(Apache) Serv	ver, MySQL Server
			Scan		Android 6.0 Marshm	allow, 3GB RAM
		·	· ·		·	
Action Performed Action's output		Va	Valid Input		alid Input	Result
Searched for the Navigate to Details screen		screen Al	ll item details such as name, un	ique ID, A n	on recovR QR code or	Pass Test
traveling bag by its	traveling bag by its of the item		R Code	an i	nexistent unique ID	
name and scanned						
QR						

Table 6 Test Case Four Details and Results

TC4 FR4 Reporting Reporting an item as lost recovR Windows 10 Pro, 1TB HDD, 8GB RAM, Wamp(Apache) Server, MySQL Server Android 6.0 Marshmallow, 3GB RAM Action Performed Action's output Valid Input Invalid Input Result Reported the travelling bag as lost Pop up alert displaying success in reporting All item details such as name, unique ID, description, phone number Illegibe bogus text Pass Test	TEST CASE (TC#)	FUNCTIONAL REQUIREMENT	TEST NAME	TEST DESCRIPTION	SOFTWARE	TEST ENVIRO	NMENT
Action PerformedAction's outputValid InputInvalid InputResultReported the travelling bag as lostPop up alert displaying success in reportingAll item details such as name, unique ID, description, phone numberIllegible bogus textPass Test	TC4	FR4	Reporting	Reporting an item as lost	recovR	Windows 10 Pro Wamp(Apache) S Android 6.0 Mar	, 1TB HDD, 8GB RAM, Server, MySQL Server shmallow, 3GB RAM
Reported the Pop up alert displaying All item details such as name, unique ID, Illegible bogus text Pass Test travelling bag as lost success in reporting description, phone number Entertion Entertion	Action Performed	Action's output	Valid	l Input	Inval	lid Input	Result
	Reported the travelling bag as los	Pop up alert display t success in reporting	ying All its g descri	em details such as name, uniq iption, phone number	ue ID, Illegi	ble bogus text	Pass Test

5.3. USABILITY TESTING

The table below summarized tests that were performed to ascertain the usability and experience of users while interacting with the system.

Table 7 Usability Testing Results

Element	Output
Flow from start to finish	Yes
Feedback from Actions performed	Instant Feedback
Seamless Navigation	Yes
Performance	Optimal
Failure or crashes	None
Runtime error messages	None
Slow or delayed loading	Acceptable

5.4. INTEGRATION TESTING

This check whether the various components of the system are integrated and working in sync. All the screens, functions, stores, data tables and other modules were connected with seamless interfacing. All the required outputs were produced successfully as expected from the systems and all inputs were validated and stored in the correct formats.

CONCLUSIONS AND RECOMMENDATION

6.1. Conclusions

The project has produced a novel intelligent system for quick and efficient management of lost items that were recovered. The system was tested and implemented in a development environment and multiple test case scenarios achieved. The system may seem simple but it has wide range of practical applications in our daily lives. The system brings about quick and easy management of queues with very little cost and can be successfully implemented in medium crowd environment and thus help in the elimination of physical lines and waiting time all over the country in service-based institutions and organizations.

This project has awarded me the opportunity to learn and a plethora of new technologies and integration of all the knowledge that is acquired in the four-year course. Just to mention a few, the system has enabled me put in practice Software development life cycle from requirements specification to documentation. It has also enabled me to interact and use the various CASE tools that are available to make the whole process easy and efficient.

Among the tools that I have been able to interact with is

- MySQL and Microsoft Visio which assisted in database design
- SMS gateways from AfricasTalking API
- Google products such as firebase, Google Cloud Functions
- Apache web server
- Visual Studio Code

6.2. Recommendation for future work

Even as fulfilled as it is, there is still plenty of work to be done to make the system much better than I have made it so far. I can therefore recommend the following two courses of action to be done in the future:

- 1. Incorporate more categories of items in a dynamic manner into the app
- 2. Allow users to add items from the web platform after searching

Works Cited

- [1] B. Roberts, "Lifting Barriers to Internet Development in Africa," Nairobi, 2014.
- K. Banda, "An app to find your lost ID," 17th September 2014. [Online]. Available: https://www.standardmedia.co.ke/lifestyle/article/2000135242/an-app-to-find-your-lost-id. [Accessed 10th October 2017].

International Journal of Social Sciences and Information Technology ISSN 2412-0294

Vol IV Issue VII, July 2018

- [3] Google Play Store, October 2017. [Online]. Available: https://play.google.com/store/apps/details?id=com.lostandfounddocumentscentre.lnf.
- [4] L. M. Katherine Southwick, "National Registration Process Leave Minorities on the Edge of," Nairobi, 2008.
- [5] Amanda Sibley, "19 Reasons You Should Include Visual Content Your Marketing," hubspot.com, 10 November 2017. [Online]. Available: https://blog.hubspot.com/blog/tabid/6307/bid/33423/19-Reasons-You-Should-Include-Visual-Content-Your-Marketing.html. [Accessed 10 November 2017].
- [6] National Registration Bureau, "National Registration Bureau Standard Operating," Government Printers, Nairobi, 2007.
- [7] Lostandfounddc, "Home," LostAndFoundDc, 19` February 2014. [Online]. Available: http://lostandfounddc.co.ke/. [Accessed 12 October 2017].
- [8] Patacom Kenya, "Home," Patacom.com, 6 March 2014. [Online]. Available: http://patacom.co.ke. [Accessed 25 November 2017].
- [9] Phillip Etemesei, August 2017. [Online]. Available: https://omgvoice.com/technology/recover-lost-cards.
- [10] Charles Gichane, February 2012. [Online]. Available: https://www.capitalfm.co.ke/news/2012/02/track-your-lost-id-using-sms/.
- [11] S. Githinji, "Designing Lost and Found Web Application," Kemu.Academia.Edu, Nairobi, 2015.
- [12] Communications Authority of Kenya, "FOURTH QUARTER SECTOR STATISTICS REPORT FOR THE," The Government Press, Nairobi, 2017.

APPENDICES

SAMPLE CODE

Login Script Web

```
<script src="https://www.gstatic.com/firebasejs/4.12.1/firebase.js"></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></
                 <script> // Initialize Firebase
                                  var config = {
                                  apiKey: "AIzaSyBMbSNheWTyDk3slRCxO5diLxTynX7Q64Q",
                                  authDomain: "recovr-app.firebaseapp.com",
                                  databaseURL: "https://recovr-app.firebaseio.com",
                                  projectId: "recovr-app",
                                  storageBucket: "recovr-app.appspot.com",
                                  messagingSenderId: "327082146500" };
                                  firebase.initializeApp(config);
                                  function signInUser(){
                                                   var email = $('#email').val() || ''
                                                   var password = $('#password').val() || ''
                                                   if(email == ''){
                                                                     alert('Login Error: Missing Email Address')
                                                   }else if(password == ''){
```

```
alert('Login Error: Missing Password')
           }else{
               firebase.auth().setPersistence(firebase.auth.Auth.Persistence.SESSION)
                .then(function() {
                   firebase.auth().signInWithEmailAndPassword(email, password).then(()=>{
                       // window.location.href = 'Index.php'
                       var hashed_email = b64_sha1(email.toUpperCase())
firebase.database().ref('users/'+hashed_email+'/').once('value',(snap)=>{
                           if(snap.val()){
                               var request;
                               var user = snap.val()
                               var user_name = user.user_name
                               var user_phone = user.user_phone
                               var user_token = firebase.auth().currentUser.uid
                               var user_firebase_key = hashed_email
                               var serializedData = [
                                   { name: 'user name', value: user name },
                                   { name: 'user_email', value: email },
                                   { name: 'user_phone', value: user_phone },
                                   { name: 'user_token', value: user_token },
                                   { name: 'user_firebase_key', value: user_firebase_key }
                               1
                               request = $.ajax({
                                   url: 'php/set session.php',
                                   contentType: 'application/x-www-form-urlencoded',
                                   type: 'POST',
                                   data: serializedData
                               })
                               request.done((response, textStatus, jqXHR) => {
                                   console.log(response)
                                   response = JSON.parse(response)
                                   console.log('=========');
                                   console.log(response);
                                   console.log('========');
                                   if (response.success == 'true') {
                                       // alert('tuko true')
                                       window.location.href = 'dashboard/dashboard.php'
                                   } else {
                                       notifier.showNotification('Oooops! The request failed
and we are working on the issue right away. Try again in a while', 'top', 'right', 'danger', 4000)
                                   }
                               })
                               request.fail((jqXHR, textStatus, errorThrown) => {
                                   notifier.showNotification('Oooops! The request failed and
we are working on the issue right away. Try again in a while', 'top', 'right', 'danger', 4000)
                               })
                           }else{
                               notifier.showNotification('Oooops! The User doesnt
exist','top','left','info',4000)
                           }
                       })
```

Login Script on Mobile App

```
_signIn(){
       this. contentContainer. root.scrollToPosition(0, 0)
        const {auth} = this.props.screenProps.stores
        const {email,password} = this.state
       if(email && password){
            this.setState({autologin:true})
            auth.signIn({email,password}).then((user)=>{
                AsyncStorage.multiSet([
                    ['user_email',email.toString()],
                    ['user_password', password.toString()]
                ],(err)=>{
                    if(err){
                        console.log(err.toString())
                        this._showAlert('Storage Error: ', 'Trouble Saving Due To: ' +
err.toString(), 'Close')
                    }else{
                        // console.log('navigating to home')
                        // console.log(this.state)
                        // console.log(user)
                        if(user){
                        this.props.navigation.replace('Home')
                    }
                })
            }).catch((err)=>{
                // alert(err.message)
                this._showAlert('Authentication Error: ', 'Reason: ' + err.message, 'Close')
                console.log(err.message)
            })
       }else{
            this._showAlert('Missing Information: ', 'It appears you have not entered a
password or an email address, please enter each correctly to proceed', 'Close')
        }
    }
    componentDidMount() {
       AsyncStorage.multiGet(['user_email', 'user_password'], (err, results)=>{
```

```
if(err){
                console.log(err.toString())
                this._showAlert('Storage Error: ', 'Trouble Retrieving Credentials:' +
err.toString(), 'Close')
            }else if(!results){
                console.log('No Login Details in store')
            }else{
                // this.setState({autologin:true})
                // console.log(results)
                let email = results[0][1]
                let password = results[1][1]
                if(email && password){
                    this.setState({
                        email: email,
                        password: password,
                    },()=>{
                        this._signIn()
                    })
                }
            }
        })
```

Register Script on the Web

```
<script src="https://www.gstatic.com/firebasejs/4.12.1/firebase.js"></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></
<script> // Initialize Firebase
            var config = {
            apiKey: "AIzaSyBMbSNheWTyDk3slRCx05diLxTynX7Q64Q",
            authDomain: "recovr-app.firebaseapp.com",
            databaseURL: "https://recovr-app.firebaseio.com",
            projectId: "recovr-app",
            storageBucket: "recovr-app.appspot.com",
            messagingSenderId: "327082146500" };
            firebase.initializeApp(config);
            function replaceBulk( str, findArray, replaceArray ){
                        var i, regex = [], map = {};
                         for( i=0; i<findArray.length; i++ ){</pre>
                                      regex.push( findArray[i].replace(/([-[\]{}()*+?.\\^$|#,])/g,'\\$1') );
                                     map[findArray[i]] = replaceArray[i];
                         }
                         regex = regex.join('|');
                         str = str.replace( new RegExp( regex, 'g' ), function(matched){
                                     return map[matched];
                         });
                        return str;
            function signupUser(){
                        var full name = $('#name').val()
                        var email = $('#email').val()
                        var phone_no = $('#phone_no').val()
                        var password = $('#password').val()
```

```
var confirm pass = $('#confPass').val()
        // console.log(`fullname: ${full name}\nfullname: ${email}\nfullname:
${phone_no}\nfullname: ${password}\nfullname: ${confirm_pass}`)
        if(full name == ''){
            console.log('Missing Name')
        }else if(email == ''){
            console.log('Missing email')
        }else if(phone_no == ''){
            console.log('Missing phone')
        }else if(password == ''){
            console.log('Missing password')
        }else if(confirm_pass == ''){
            console.log('Missing confirmation password')
        }else if(password != confirm_pass){
            console.log('Mismatching password and confirmation password')
        }else{
            var fb_phone_number = phone_no.replace('254','0')
            $('#register_loading').append('<Small class="text text-md" >Logging In...</Small>')
            firebase.auth().createUserWithEmailAndPassword(email,password).then(()=>{
                var userhash = b64 sha1(email.toUpperCase())
                // console.log(`userhash: ${userhash}\n joined name: ${newJoinedName}`)
                // console.log(userhash);
                var userhash = userhash.split('/').join(' ')
                var userhash = userhash.split('#').join('_')
                var userhash = userhash.split('.').join('_')
var userhash = userhash.split('[').join('_')
                var userhash = userhash.split(']').join('_')
                var userhash = userhash.split('$').join('_')
                // console.log(userhash);
                firebase.database().ref('users/'+userhash+'/').update({
                    user_name: full_name.toUpperCase(),
                    user_email: email.toLowerCase(),
                    user phone: fb phone number,
                }).then(()=>{
                    window.location.href = 'Login.php'
                }).catch((err)=>{
                    console.log(err.message)
                    $('#register_loading').empty()
                    alert('user details update error: '+err.message)
                })
            }).catch((err)=>{
                console.log(err.message)
                $('#register_loading').empty()
                alert('user creation error: '+err.message)
            })
        }
   }
</script>
Register Script on Mobile
signUp(){
        const {auth} = this.props.screenProps.stores
        const {first name,last name,email,phone number,password,confirm pass} = this.state
        console.log(
            `FirstName: ${first name}\n`+`LastName: ${last name}\n`+`Email: ${email}\n`+
```

```
`Phone: ${phone number}\n`+`Password: ${password}\n`+`ConfirmPass:
${confirm_pass}\n`)
            if(email && password && first name && last name && phone number && (password ===
confirm_pass)){
                this. contentContainer. root.scrollToPosition(0, 0)
                this.setState({loading:true},()=>{
                    auth.signUp({email,password}).then((user)=>{
                        // console.log(user)
                        if(user){
                            this.setState({loading:false})
                            this.setState({logging in:true},()=>{
                                auth.signIn({email,password}).then(()=>{
                                    let full_name = first_name+' '+last_name
                                    let phone = phone_number.toString()
                                    auth.updateDetails({email,phone,full_name}).then(()=>{
                                        this.setState({logging_in:false})
                                        AsyncStorage.multiSet([
                                             ['user_email',email.toString()],
                                             ['user password', password.toString()]
                                        ],(err)=>{
                                            if(err){
                                                 this._showAlert('Storage Error: ',
err.toString(), 'Close')
                                            }
                                        })
                                        this.props.navigation.navigate('Login')
                                    }).catch((err)=>{
                                        console.log('updatedetails error:' + err.message)
                                        this._showAlert('Storage Error: ', 'updatedetails
error:' + err.message, 'Close')
                                    })
                                }).catch((err)=>{
                                    console.log('signIn error:' + err.message)
                                    this._showAlert('Storage Error: ', 'signin error:' +
err.message, 'Close')
                                })
                            })
                                // this.props.navigation.navigate('Login')
                        }else{
                            this._showAlert('Signup Error: ', 'User not created', 'Close')
                        }
                    }).catch((err)=>{
                        console.log('signup error: ' + err.message)
                        this._showAlert('Storage Error: ', 'signup error:' + err.message,
'Close')
                    })
                })
            }else{
                alert('Please fill all details correctly and ensure the passwords match')
            }
                console.log('Signup successful')
                console.log(auth.authUser)
    }
```