

2022-02

Mobile application for gate pass management system enhancement

Hilary, Rambo

NM-AIST

<https://doi.org/10.58694/20.500.12479/1608>

Provided with love from The Nelson Mandela African Institution of Science and Technology

**MOBILE APPLICATION FOR GATE PASS MANAGEMENT SYSTEM
ENHANCEMENT**

Rambo Hilary

**A Project Report Submitted in Partial Fulfillment of the Requirements for the Degree of
Master of Science in Embedded and Mobile Systems of the Nelson Mandela African
Institution of Science and Technology**

Arusha, Tanzania

February, 2022

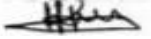
ABSTRACT

Gate pass management is an essential measure to keep records of people's entrance and exit of company premises. Technological improvement steered gate pass management from paper-based logbooks to a web-based system that relies on the internet. The web-based system contains forms to input personal information, exit and entry time, and reason for requesting a gate pass. For a large organization with a diverse number of employees who do not have access to computers for their daily routine activities, it's extremely challenging to request a gate pass through an existing web-based gate pass management system. Thus, users turn to smartphones to access the web-based gate pass management system. A responsive web system is a considerable approach to resolve screen size and input method challenge of the smartphone; however, the study shows many users are not interested to use a mobile browser for accessing web services. Hence necessitate the implementation of a mobile application for enhancing the existing web-based gate pass management system. Web-based gate pass management system requires printers for printing gate pass permission, and barcode scanners are required for scanning gate pass permission paper at the security gate. The cost of hardware and printing materials is waived through the utilization of mobile phones to create shareable digital gate passes, as well as utilizing a camera as a barcode scanner. The camera can be used for photographing gate pass users for extra records. Seventy (70) respondents were selected through the non-probability purposive sampling technique. The mobile application was developed, tested, and validated through hundred potential users.

DECLARATION

I, Rambo Hilary do hereby declare to the Senate of the Nelson Mandela African Institution of Science and Technology that this Project Report is my original work and that it has neither been submitted nor being currently submitted for degree award in any other institution.

Rambo Hilary



05/04/2022

Name and Signature of Candidate

Date

The declaration is confirmed by the following:

Dr. Dina Machuve



05/04/2022

Name and Signature of Supervisor 1

Date

Dr. Silas Mirau



05/04/2022

Name and Signature of Supervisor 2

Date

COPYRIGHT

This Project Report is copyright material protected under the Berne Convention, the Copyright Act of 1999, and other international and national enactments, in that behalf, on intellectual property. It must not be reproduced by any means, in full or in part, except for short extracts in fair dealing; for researcher's private study, critical scholarly review or discourse with an acknowledgement, without a written permission of the Deputy Vice Chancellor for Academic, Research and Innovation, on behalf of both the author and the Nelson Mandela African Institution of Science and Technology (NM-AIST).

CERTIFICATION

The undersigned certify that have read and hereby recommend for acceptance by the Senate of the Nelson Mandela African Institution of Science and Technology, a project report titled "*Mobile Application for Gate Pass Management System Enhancement*" in partial fulfillment of the requirements for the award of the degree of Master of Science in Embedded and Mobile Systems.

Dr. Dina Machuve



05/04/2022

Name and Signature of Supervisor 1

Date

Dr. Silas Mirau



05/04/2022

Name and Signature of Supervisor 2

Date

ACKNOWLEDGMENTS

Firstly, I would like to thank my almighty God for strength, knowledge and wisdom he gave me during the whole project period.

I would like to express my heartfelt appreciation to my supervisors: Dr. Dina Machuve and Dr. Silas Mirau, for their support and guidance since the beginning of this project. Their endless support has been foundation for successfully completion of this project.

Lastly, I thank A to Z Textile Mills Limited for their willingness to host me and providing internship opportunity that yielded this project. It's hard to mention everyone who supported me during the period of project at A to Z Textile Mills, but my appreciation goes to my host supervisor Mr. Srinivasan Kumar.

DEDICATION

I dedicate this project to my family and everyone who supported me spiritually and financially.

TABLE OF CONTENTS

ABSTRACT	i
DECLARATION.....	ii
COPYRIGHT	iii
CERTIFICATION.....	iv
ACKNOWLEDGMENTS.....	v
DEDICATION	vi
TABLE OF CONTENTS	vii
LIST OF TABLES	x
LIST OF FIGURES.....	xi
LIST OF APPENDICES	xii
LIST OF ABBREVIATIONS AND SYMBOLS.....	xiii
CHAPTER ONE.....	1
INTRODUCTION.....	1
1.1 Background of the Problem.....	1
1.2 Statement of the Problem	2
1.3 Rationale of the Project	3
1.4 Project Objectives.....	4
1.4.1 General Objective.....	4
1.4.2 Specific Objectives.....	4
1.5 Project Questions.....	4
1.6 Significance of the Project.....	5
1.7 Delineation of the Project.....	5
CHAPTER TWO.....	7
LITERATURE REVIEW	7
2.1 Introduction	7

2.2	Gate Pass Management.....	7
2.2.1	Traditional Gate Pass Management.....	7
2.2.2	Automated Gate Pass Management Systems	9
2.3	Related Studies	9
2.3.1	Responsive Web-based Gate Pass Systems.....	9
2.3.2	Radio Frequency Identification	10
2.3.3	Facial Recognition Gate Pass Management System	11
2.3.4	Mobile Applications	11
CHAPTER THREE.....		12
MATERIALS AND METHODS		12
3.1	Study Area and Scope of the Project.....	12
3.2	Sampling Technique.....	12
3.3	Data Collection Methods	13
3.3.1	Questionnaire.....	13
3.3.2	Interviews	13
3.3.3	Observation	14
3.4	Data Analysis.....	14
3.5	Mobile Application Development	15
CHAPTER FOUR		17
RESULTS AND DISCUSSION		17
4.1	Results	17
4.1.1	Findings from Web-Based System Users.....	17
4.1.2	Mobile Application Requirements	20
4.1.3	Application Design.....	22
4.1.4	Mobile Application Prototyping.....	28
4.1.5	Mobile Application Testing.....	43

4.1.6	Mobile Application Validation.....	45
4.2	Discussion.....	45
4.2.1	Portability.....	46
4.2.2	Camera.....	46
4.2.3	Notification.....	46
	CHAPTER FIVE.....	48
	CONCLUSION AND RECOMMENDATIONS.....	48
5.1	Conclusion.....	48
5.2	Recommendations.....	49
	REFERENCES.....	50
	APPENDICES.....	53
	POSTER PRESENTATION.....	70

LIST OF TABLES

Table 1:	Demographic information of existing system users	18
Table 2:	Mobile Application functional requirements.....	21
Table 3:	Mobile application non-functional requirements	22
Table 4:	Description of use cases	26
Table 5:	Description of use cases	45
Table 6:	Mobile application acceptance testing.....	45

LIST OF FIGURES

Figure 1: Gate pass management Log Book (Amazon, n.d).....	8
Figure 2: Software development life cycle prototype model.....	16
Figure 3: Respondent computer Access (a), Respondent smartphone ownership (b)	19
Figure 4: Respondent smartphone usage behavior	19
Figure 5: Respondent's perception of existing system enhancement.....	20
Figure 6: Application Architecture	24
Figure 7: Use case diagram	25
Figure 8: Database E-R diagram.....	28
Figure 9: Market Share of mobile operating systems (Statcounter, n.d)	29
Figure 10: Login screen	32
Figure 11: Users and Admins screen	32
Figure 12: Gate pass information	33
Figure 13: Employee gate pass detail	34
Figure 14: Visitor gate pass information	35
Figure 15: Visitor details	35
Figure 16: Admin gate pass request view.....	36
Figure 17: Staff gate pass request view	37
Figure 18: Unapproved gate pass	38
Figure 19: Approved gate pass details.....	38
Figure 20: Security screen	39
Figure 21: Gate pass barcode scanner.....	40
Figure 22: Security permission.....	41
Figure 23: Camera feature to take pictures	42
Figure 24: Sample gate pass picture	43

LIST OF APPENDICES

Appendix 1: The Sample Codes for Mobile Application Development	53
Appendix 2: The Sample Codes for Application Programming Interface	65

LIST OF ABBREVIATIONS AND SYMBOLS

API	Application Programming Interface
CCTV	Closed-Circuit Television
DB	Database
DFD	Data Flow Diagram
ERD	Entity Relationship Diagram
HTTP	Hyper Text Transfer Protocol
ICT	Information Communication Technology
ID	Identification Number
IDE	Integrated Development Environment
IIS	Internet Information Services
JSON	JavaScript Object Notation
PC	Personal Computer
RDBMS	relational database management system
REST	Representational State Transfer
SDK	Software Development Kit
SDLC	System development life cycle
UI	User Interface

CHAPTER ONE

INTRODUCTION

1.1 Background of the Problem

The gate pass management systems have been practiced in multiple ways to manage exit and entry to the industry or institute, this includes a manual register where visitors sign in and sign out to the register book on the entrance area (Venkatesa & Sanju, 2018). Usually, employees request gate passes if they have to exit the company premises during working hours. Nevertheless, visitors may need to enter the company premises for different purposes such as official meetings, consultations, or other official and unofficial activities. All these gates pass management processes were handled manually, over the last few years; has been improved from paper-based gate pass logbooks to computerized systems (Gunawan & Kadir, 2017).

Computerized systems increase the efficiency and effectiveness of managing information employees and visitors entry or exit the company or organization cases (Sunico *et al.*, 2020). Likewise, it allows the user to create, approve and authorize entrance and exiting of people, vehicles, goods, and items to the company premises (Rapartiwar *et al.*, 2017). Computerized gate pass management systems are widely adopted and used by many companies worldwide for security purposes, to track employees' movement in and out of the company premises during working hours and visitors (Mason, 2005). Digital gate pass for visitors and employees are generated automatically using the computerized systems and for some advanced systems visitors' image are captured through the camera (Rodrigues *et al.*, 2021).

Additionally, computerized gate pass management systems are favored by large organizations or companies such as A to Z Textile Mills Ltd. It comprises multiple companies working on the same premises, due to its large size, it is essential to have measures to manage visitors and employees' entrance and exit to the organization premises (Shree *et al.*, 2019). Security measure involves physical security control of entrance through gate pass management systems (Mason, 2005). The company receives approximately 300 visitors per day. In addition to 9000 employees. The best way to manage the gate pass process was through a computerized system. The A to Z Textile Mills Ltd implemented a web-based gate pass management system for easing management of incoming visitors as well as employees who leave during working hours.

The existing gate pass management system provides access to all approved gate passes to the security guard. A security guard is responsible to print gate pass from the system. Gate pass printout contains a barcode that is scanned by a barcode scanner to record the entrance and exiting time. Printed gate pass is handed to gate pass user as proof of permission to enter or exit the company premises.

Due to a large number of employees, there is a shortage of computer access to the existing gate pass management system. Moreover, users turned to their smartphones to have access to the existing web-based gate pass management system. Multiple users prefer to access the existing system through a smartphone but they face multiple challenges. Like smartphones have small screen size which makes it impossible to view text and graphics like a computer screen, smartphone lack windows which enables a user to open multiple pages at a time, the smartphone does not have a mouse pointer that can be used to navigate through a web-based system designed for computer (Hussain *et al.*, 2015).

The developed mobile application can leverage hardware capabilities to eliminate the need for extra hardware components (Ma *et al.*, 2018). Mobile phone camera hardware is leveraged by the native mobile application to scan gate pass barcodes, also taking pictures of gate pass users. Thus the developed mobile application is more convenient for gate pass management compared to the use of web-based computerized gate pass management systems (Islam & Mazumder, 2010). The mobile application focus to provide user-friendly and simple to use application. The application simplifies the process of gate pass creation, sharing of digital gate pass, scanning of the digital gate pass and lastly providing information of gate pass usage.

1.2 Statement of the Problem

To access the existing web-based gate pass management system, users need to have direct access to the computer. Due to a large number of employees (9000 employees) the company faces challenges to provide computer access to every employee because not all employees require a computer for their daily routine activities. This challenge leads to an extreme hindrance to accessing the existing web-based gate pass management systems.

Furthermore, the existing web-based gate pass management system requires printers and barcode scanners (Prerana *et al.*, 2021). The printer is used to print a gate pass as proof of permission, while the barcode scanner is used at the gate to scan the printed gate pass for verifying the gate pass. The requirement of other hardware components such as printer and barcode scanner increases the cost

of using the gate pass management system. Furthermore, it is challenging for gate passes users to keep the printed paper for a long period. Digital gate passes can help to overcome handing permission papers to the gate pass users.

The need for extra hardware components can be resolved through the mobile application that natively utilizes smartphone built-in hardware (Bessghaier & Souii, 2018). The printout gate pass paper can be replaced by a shareable digital gate pass through the mobile application.

Solutions such as responsive web design can provide a better user experience to mobile devices users. This is achieved through the ability of responsive web elements to change in response to device screen size (Hussain & Mkpojiogu, 2015). However, responsive web design requires extra hardware devices such as cameras and printers for accomplishing the process of gate pass management, because smartphone hardware such as cameras are restricted to the native application only. Thus it is challenging to utilize a smartphone camera as a barcode for scanning gate pass barcodes. Also, many users prefer mobile applications compared to the usage of a web browser to access mobile resources.

Finally, the integrated mobile application to the existing web-based gate pass management systems simplify the process of creating gate pass for staff and visitors, eliminates manual entries, utilizing smartphones hardware for scanning barcodes, quick notification, quick authorization by security guards. The mobile application enhances the existing web-based system by increasing its efficiency, effectiveness, and flexibility (Hussain *et al.*, 2017).

1.3 Rationale of the Project

Although many means of gate pass management have changed from physical register to web-based information systems. Smartphones have brought new ways for people to interact with the internet. For the case of simple computing tasks such as sending an email, browsing the internet, texting, and video call, users prefer their smartphones. Mobile applications have become inevitable for providing a better user experience. This is contributed to the fact that many users interact with their smartphones more often compared to computers.

Many companies have benefited from mobile applications through the enhancement of web-based systems. Therefore, existing mobile platforms can be used to simplify the process of gate pass management.

The mobile application has added advantages such as eliminating the need for paper and printer for printing gate passes as the generated barcode can be scanned on the gate for permission. Leveraging mobile application features such as a camera for taking photos of visitors.

The need for mobile gate pass management applications can be traced through several users currently using smartphones to access the internet, also how different users interact with social media mobile applications versus websites of these social media.

The mobile application proved to be a preference to many users compared to a web-based system. Some companies took the steps further to develop a responsive web app that is a hybrid between the native mobile application and web-based systems.

1.4 Project Objectives

1.4.1 General Objective

The main objective of this project is to develop a gate pass mobile application that is integrated into a gate pass management system for staff and visitors (customers). The application will be used to request (create) gate passes, approve or reject gate pass and security-tracking visitors and employees in and out of organization premises.

1.4.2 Specific Objectives

- (i) Review and Analysis of existing gate pass management system.
- (ii) Mobile application requirements identification for enhancement of gate pass management system.
- (iii) Design and implementation of a mobile application for gate pass management system enhancement.
- (iv) Testing and validation of a mobile application for gate pass management system enhancement.

1.5 Project Questions

- (i) What are the shortcomings of the existing web-based gate pass management system?

- (ii) What are the requirements for the enhancement of a web-based gate pass management system?
- (iii) What mobile application can be implemented for enhancing the existing web-based gate pass management system?
- (iv) Does the implemented mobile application enhance the existing gate pass management system?

1.6 Significance of the Project

Many gate pass management systems are designed to provide maximum tracking and recording of visitor access to the company premises to safeguard the organization from unwanted visitors, these systems provide control over employees who moves out of company premises during working hours. The mobile application provides a suitable way to utilize these systems for recording, tracking, and authorization of movement in and outside of the company premises. For a big company, such A to Z Textile Mills Limited which have more than 9000 employees, it is challenging to provide a computer to every employee especially not all employee do not require computers to accomplish their daily tasks. Therefore, a mobile application is required to reduce the digital divide, because many people possess smartphones.

Furthermore, mobile applications compared to Desktop computers or other computing devices, the mobile application offers flexibility, quick management, and control of gate passes, minimizing manual input such as entrance time and existing time. Mobile phone hardware components such as a camera can be utilized to scan gate pass bar codes and taking pictures of visitors for more security purposes.

Therefore, it is crucial to enhance web-based gate management systems to achieve maximum use user experience, reduce a digital divide, flexibility, customizability, it can be accessed anywhere at any time, as well as keeping up with technology that is highly used and acceptable to many users. The mobile application requires fewer resources compared to web-based systems.

1.7 Delineation of the Project

Web-based gate pass management systems have been used to automate the process of gate pass management. The purpose of this study is not to redevelop or restructuring the existing systems through a mobile application. This project aims to enhance the existing system to deliver access to

multiple users through smartphones, improve the process of gate pass management, simplify user interaction with the existing system. Therefore, the mobile application communicates and accesses the database of the existing system through an Application Programming Interface (API). The business logic of the existing system is used as the direction for mobile application implementation.

CHAPTER TWO

LITERATURE REVIEW

2.1 Introduction

This Chapter discusses a review of literature on various studies used to establish requirements for enhancement of existing web-based gate pass management system. The Chapter begins with an evaluation of gate pass management processes, the evolution of gate pass management processes, related work for gate pass management. Finally, identifying strength and weakness of existing solutions which leads to the development of a mobile application for enhancing web-based gate pass management system.

2.2 Gate Pass Management

Security has been an important measure to counteract any malpractices by visitors or company employees (Gunawan & Kadir, 2017). Gate pass management through keeping records of all visitors and employee movements is one of the security measures to control and prevent unauthorized people or malicious activities (Rodrigues *et al.*, 2021). Gate pass management has been practiced for years to keep records and track of people's entrance and exit to the company premises by using a different method. Passage management is managed by security personnel who allows entrance and exit from the company premises. The process of get pass management has been influenced by computing technology. The evolution started from traditional log book to automated gate pass management through various Information Communication Technologies (ICT).

2.2.1 Traditional Gate Pass Management

Traditional gate pass management involves recording visitors' information and employee movement to the register book (log book) which is handled by security personnel or receptionist (Liu, 2016). The designated personnel question the visitor through various questions such as the host's name, reason for visitation to determine the authenticity of a visitor. Traditional gate pass management is practiced by many small companies which have few employees, as well as experience little visitor traffic. Other companies with poor ICT infrastructure use traditional gate pass management. The main shortcoming of traditional gate pass management is not an efficient way to keep and track records for a long time, difficult to produce a report of visitor and employee movement (Rapartiwar *et al.*, 2017).

Visitor Register

DATE	NAME	FIRM	ADDRESS	TO SEE
			STREET	
			CITY STATE	TIME IN TIME OUT
			STREET	
			CITY STATE	TIME IN TIME OUT
			STREET	
			CITY STATE	TIME IN TIME OUT
			STREET	
			CITY STATE	TIME IN TIME OUT
			STREET	
			CITY STATE	TIME IN TIME OUT
			STREET	
			CITY STATE	TIME IN TIME OUT
			STREET	
			CITY STATE	TIME IN TIME OUT
			STREET	
			CITY STATE	TIME IN TIME OUT
			STREET	
			CITY STATE	TIME IN TIME OUT
			STREET	
			CITY STATE	TIME IN TIME OUT
			STREET	
			CITY STATE	TIME IN TIME OUT
			STREET	
			CITY STATE	TIME IN TIME OUT

Visitor Register

DATE	NAME	FIRM	ADDRESS	TO SEE
			STREET	
			CITY STATE	TIME IN TIME OUT
			STREET	
			CITY STATE	TIME IN TIME OUT
			STREET	
			CITY STATE	TIME IN TIME OUT
			STREET	
			CITY STATE	TIME IN TIME OUT
			STREET	
			CITY STATE	TIME IN TIME OUT
			STREET	
			CITY STATE	TIME IN TIME OUT
			STREET	
			CITY STATE	TIME IN TIME OUT
			STREET	
			CITY STATE	TIME IN TIME OUT
			STREET	
			CITY STATE	TIME IN TIME OUT
			STREET	
			CITY STATE	TIME IN TIME OUT
			STREET	
			CITY STATE	TIME IN TIME OUT

AT-A-GLANCE

Figure 1: Gate pass management Log Book (Amazon, n.d)

2.2.2 Automated Gate Pass Management Systems

The growth of computer technology enables many companies to migrate from using manual gate management systems to automated gate pass management which is more superior to traditional means of management of visitors and employee movement in and out of the organization, especially for a large organization. Compared to traditional means the automated gate pass management adds value to the practice of gate pass management (Sunico *et al.*, 2020). The automated systems can keep records for a very long time, simplify retrieval and access of all records regardless of time, automated systems provide efficient means to manage and track all gate passes (Hussain *et al.*, 2017). Automated gate pass systems have been developed in various forms such as web-based gate pass systems, Biometric systems, and closed-circuit television (CCTV) cameras.

A computerized gate-pass management system is intended to replace the traditional method of manual registration of visitors and management of visitor's information activities in the premises which sometimes resulted in poor record-keeping of visitor's information. This will ensure smooth functioning of work, safety, and company security through computerized means to keep records of all people visiting and leaving premises. Employee movement can be managed during work hours while visitors need special permission to enter the organization's premises (Hussain *et al.*, 2017).

2.3 Related Studies

2.3.1 Responsive Web-based Gate Pass Systems

Responsive web design is a common technology for keeping records and track of incoming and outgoing visitors from the company premises. The technology uses a web server that interacts with the database to record all traffic of employees and visitors. Usually, elements of the responsive web change in the response to screen size. This ensures a better user experience for mobile devices users.

In one study, Hussain and Mkpjojiogu (2015) focused on the effect of responsive web design on the user experience with laptop and smartphone devices. The metrics of the study were task completion rate, task error rate, task time, task difficulty, perceived usability, perceived learnability, and loyalty. The study recommended the use of responsive web design to counteract the challenge of device screen size.

In another study, Gowtham *et al.* (2019) proposed automation of visitors gate management system. The system employs a responsive web design to accommodate computer and mobile devices users.

The proposed web portal provides an interface to host employees to create gate passes that need approval from the administrator or department head. After approval the gate pass becomes visible to the security personnel, the security personnel set time of entrance, print the gate pass, and hand it over to the visitor as proof of authorization. The gate pass paper contains various information such as visitor's information and visitor's host information. The study proposed photographing of the visitor through a camera interfaced to the personal computer (PC) through ethernet.

Based on a discussion of the studies above, responsive web design can provide a better user experience to mobile devices users. However, responsive web design requires extra hardware devices such as a camera and printer for gate pass management. This is contributed by the limitation of responsive web access to mobile resources. Therefore, the responsive web cannot utilize a camera for scanning and photographing gate pass users. Also, from user usability experience, users tend to prefer mobile applications to accomplish various tasks, if there is a mobile application and responsive of the same service (Hussain *et al.*, 2017). This is observable to many users who use Gmail mobile application to access google mail services compared to users who access mail service through smartphone web browsers.

2.3.2 Radio Frequency Identification

Gunawan and Kadir (2017) proposed Integration Protocol Student Academic Information to Campus Radio Frequency Identification (RFID) Gate Pass System. Through the study solution implemented through integration with student system. By using the RFID, it provides real-time information capturing. The proposed solution requires every user to have an RFID card where its data will be collected at the gate by an RFID reader. The RFID reader updates the database on the entrance or existing. Therefore, the company is required to have a database of all potential gate users.

The limitation of the proposed solution relies on prerecorded data of each potential gate pass user. This process requires security personnel to pre-enter the information of each incoming visitor before handing over the RFID card. It's an exhausting task for security personnel especially for a company that receives multiple visitors in a single day. Also, the RFID card holds information of a single user, which means a single RFID card for a single user. Thus, the card will require updating the RFID card for every new visitor. The previous visitors will require a new RFID card for the next visitation because the previous card has been updated for the new visitor.

2.3.3 Facial Recognition Gate Pass Management System

In another study, Rikshit *et al.* (2008) proposed a face recognition visitors management system. The study proposed an integration of the system into the camera for facial recognition. Facial recognition technology is essential to track employee attendance. However, facial recognition systems require pre-recorded visitors' faces, thus the main limitation and challenge of facial recognition are to identify new incoming visitors.

2.3.4 Mobile Applications

Smartphones brought a revolution in how people interact with automated systems (Prerana *et al.*, 2021). The study proposed standalone mobile applications for gate pass management. The study implemented mobile applications for college students. The mobile application focuses on reducing paperwork by using a digital gate pass generated by mobile applications. However, some limitations were identified from the gate pass mobile application. The limitations include; lack of a barcode scanner for scanning gate pass at the entrance gate, lack of a camera feature for photographing visitors.

Therefore, this study addresses the above limitation of the existing web-based gate pass management system through the utilization of smartphone features. Potential limitations of other studies are resolved by enhancing the web-based system through a mobile application. Compared to other studies, the mobile application utilizes smartphones for scanning barcodes as well as photographing gate pass users. Also, the mobile application allows one user to create a gate pass on behalf of another user by using an employee identification number (ID).

CHAPTER THREE

MATERIALS AND METHODS

3.1 Study Area and Scope of the Project

This project was carried out at A to Z Textile Mills Limited which is located in Arusha, Tanzania. It has a group of companies with over 9000 employees, they receive multiple visitors per day (around 300). At the company there is a web-based gate pass management system which is the main measure to keep records of visitors of the company. Staff is required to request a gate pass to move out of company premises during working hours. This makes the company suitable for implementing a mobile application for the enhancement of the existing gate passes management system. The systems will provide an API (application programming interface) which will be used as the main channel of communication between the application and the main system.

Considering the number of employees and visitors to the company, this makes A to Z Textile Mills Limited a suitable company to implement gate pass mobile applications. The project can be extended to other company institution which needs to keep track of visitors and employee pass.

3.2 Sampling Technique

In this study, the target population sample was selected amongst A to Z Textile Mills Limited employees to determine the extent of the existing system problem and identify the requirements needed to develop a mobile application for gate pass management system enhancement. The non-probability purposive sampling technique was used to collect information from users who access the existing system frequently (Liu, 2016).

The non-probability purposive technique was used because the study was formulated to enhance the existing gate pass management system. Therefore, it was crucial to pick employees with a wide understanding of the existing system (Taherdoost, 2018).

A total of 70 respondents were involved in this study, of which four were system administrators, 10 were security guards, and 53 were normal system users. Also, three system developers were involved in data collection, application design, and application testing.

3.3 Data Collection Methods

The data collection activity for this study was conducted for four weeks from 7th September to 9th October, 2020. The data collection method applied qualitative and quantitative approaches to collect sufficient information for the development of a mobile application. The qualitative approach involved unstructured interviews and observation to collect technical information for the development of the mobile application. The quantitative approach involved structured questionnaires for collecting existing system users' awareness and perception of mobile applications.

3.3.1 Questionnaire

The questionnaire for existing web-based system users comprises three sections: Demographic information, access to a smartphone, and awareness of mobile applications. Demographic information was collected to determine the correlation between the usefulness of the mobile application and users' acceptance of the mobile application. Information of respondent owning smartphones collected to assess utilization of mobile application since mobile application depends on smartphones.

Furthermore, information on respondents' awareness of mobile applications aimed to determine potential mobile application users' knowledge and their perception towards having mobile applications.

3.3.2 Interviews

The interview was conducted through semi-structured interviews with the gate pass management system users, existing system developers, and system administrators to determine their opinion on the gate pass management system. Also, to get an insight into the proposed mobile application requirements.

(i) Interview for Existing Users

Existing system users were interviewed to determine their opinion on the existing system as well as their attitude toward mobile applications. The interview focused on understanding what features are most preferred to mobile application compared to web-based system.

(ii) Interview for Gate Pass Management System Administrators and System Developers

Gate pass management system developers were interviewed to determine the business logic of the existing system. The developers provided much insight on what is required for the mobile to communicate with the existing system.

(iii) Interview for the Gate Pass Management System Administrator

System administrators are employees designated for approving gate pass requests from other employees. The interview was conducted to determine mobile application properties which will enhance the management of gate pass requests.

3.3.3 Observation

The process of requirement gathering involved the observation of the existing gate pass management system that require enhancement. The observation focused on how users interact with the web-based gate pass management systems, inputs, and outputs of the existing system, understanding the underlying business logic of the web-based system. The result of observation shows system users are categorized into three groups, normal users, security personnel, and system administrators. Normal users can create gate pass requests for their visitors or self gate pass request to move out of the company premises during working hours. The administrators are the ones responsible to approve or reject the gate pass request from normal users. The Head of the department is the administrator with the role of granted permission to the gate pass request. The security personnel at the entrance company gate they use the gate pass management system to view approved gate pass, print the gate pass and allow to employee or visitor to move in or out of the company premises.

3.4 Data Analysis

Data collected at A to Z Textile Mills Limited were categorized into quantitative data and qualitative data. Quantitative data was analyzed by using Python, a programming language suitable for data analytics. Since the purpose of the questionnaires was to assess users' awareness and perception of having a mobile application, the frequencies and percentages were interpreted and inferences were drawn.

Qualitative data were analyzed by using a generic inductive approach, in which data collected from unstructured interviews and observation were organized, categorized, and analyzed using the

qualitative data analysis package NVivo. The NVivo data analysis package tools was useful to analyze all the interview text (Liu, 2016). Interview findings were compared to establish application requirements.

3.5 Mobile Application Development

The evolutionary prototype model system development life cycle (SDLC) model was used for gate pass management mobile application development. This model was selected for mobile application development because mobile application development requires more user involvement and higher user interaction. The model helps to visualizes components of the mobile application during implementation. Also, the model helps users to provide better and more complete feedback and specification.

The purpose of using the evolutionary prototype model is to focus on creating the actual software instead of concentrating on documentation (Sabale, 2012). The prototyping model enables releasing of the mobile application in advance before the full implementation of all features.

Additionally, of evolutionary prototype model includes user involvement in the product even before its implementation which gives the user a better understanding of the software being developed, reduces time and cost as the defects can be detected much earlier, quicker user feedback is available leading to better solutions, missing functionality can be identified easily, confusing functions can be identified. Therefore, the final product is more likely to satisfy the user's desire for a look, feel, and performance (Sabale, 2012).

Finally, evolutionary prototype model allows users to evaluate the application during development and try them out before implementation. Evaluating application underdevelopment helps to identify requirements that may not have been considered during mobile application development. In this model, the following stages were involved as depicted in Fig. 2.

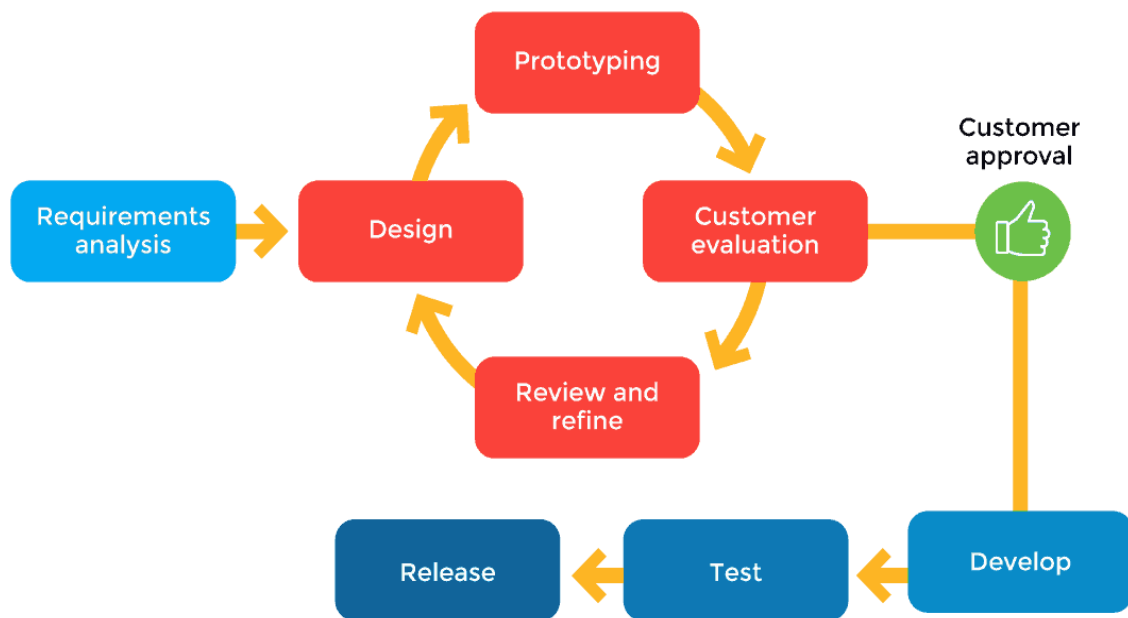


Figure 2: Software development life cycle prototype model

CHAPTER FOUR

RESULTS AND DISCUSSION

4.1 Results

This section presents findings from data collection, mobile application requirements as well as results of the developed mobile application for gate pass management system enhancement according to the objectives. The main objective was to develop a mobile application that would enhance gate pass management system management. The management of the gate passes includes employee request a gate passes, administrator approval of gate passes, and security guard responses based on the approved gate passes. During the mobile application implementation, the mobile application was developed and integrated into the existing web-based gate pass management system. Therefore, the mobile application can provide gate pass management services.

4.1.1 Findings from Web-Based System Users

Existing system users' data were collected to determine the extent of the problem and whether the mobile application would be a relevant solution. A total of 70 respondents participated in this study.

(i) Demographic Information

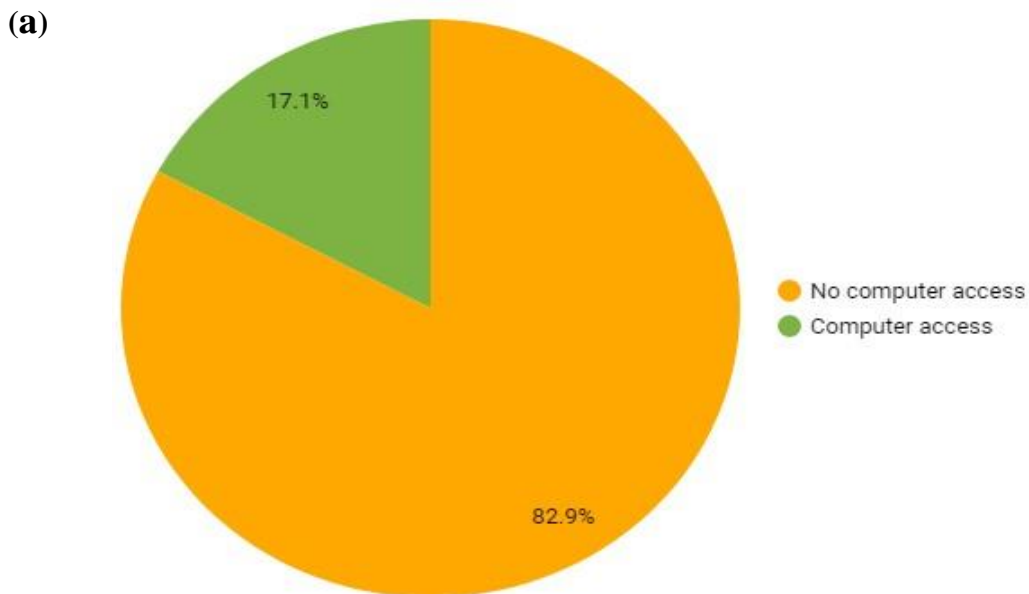
Since the study focuses primarily on users' awareness of mobile applications, the benefit of using smartphones for gate pass management and perception towards mobile applications, therefore, it is important to consider demographic characteristics among the potential users of the mobile application. The demographic characteristics which were considered in this study are: Gender, age, and education level. Out of 70 respondents, 45 were males while 25 were females. The majority of the respondents were aged in between 25 and 50 years, which represents 66.43% of the sample size. Approximately, 93% of respondents' have reached the college education level. Demographic information are illustrated in Table 1.

Table 1: Demographic information of existing system users

Demographic characteristic		Respondents	Percentage (%)
Gender	Female	25	35.71
	Male	45	64.28
Age (Years)	Less than 20	5	7.1
	20 – 29	33	47.1
	30 – 49	22	31.4
	50 – 59	8	11.4
	60 and above	2	2.9
Education	Secondary education	5	7.1
	College education	65	92.9

(ii) Users' Access to Computers and Smartphones

The 70 respondents involved during data collection are people who interact frequently with existing web-based gate pass management system. Among the 70 respondents, only 17.1% had access to computers for their daily activities, while 82.9% needs to ask other employees help for requesting a gate pass. 87.1% poses smartphones which contains multiple mobile applications, while 12.9% possess feature phones which cannot use mobile application. Responds access to a computer, and smartphone ownership results are illustrated in Fig. 3 (a) and (b), respectively.



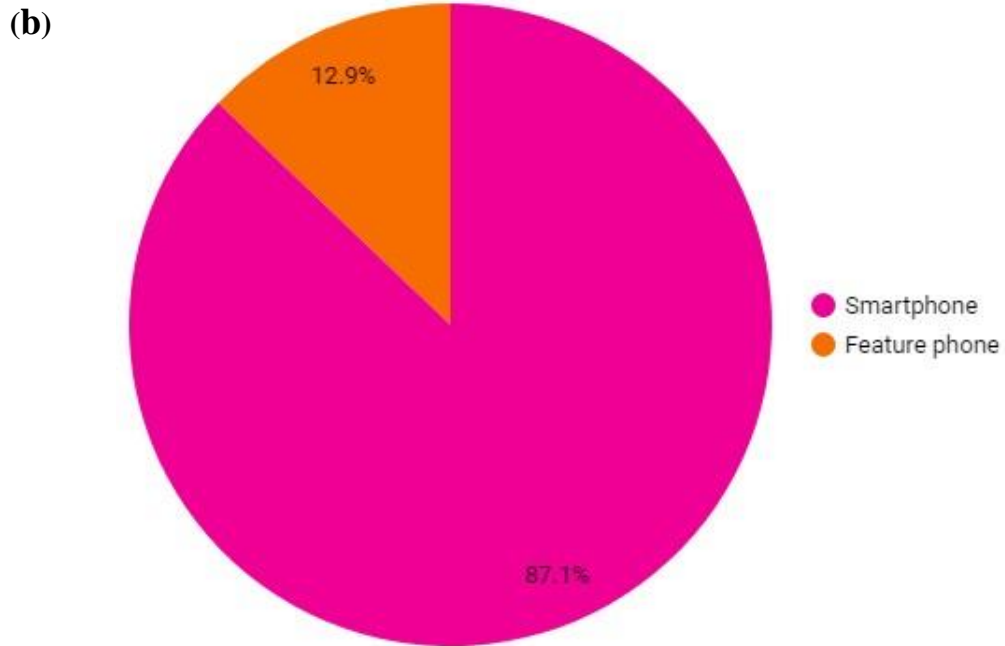


Figure 3: Respondent computer Access (a), Respondent smartphone ownership (b)

(iii) Smartphone Usage Behaviors

Out of 61 respondents who owned a smartphone 91.4% used various mobile application for accomplishing various tasks. Mobile application is the first choice for accessing internet resources. The remaining 8.6% used smartphone for normal communication such as calling and sending short messages. Smartphone usage behavior is illustrated in Fig. 4.

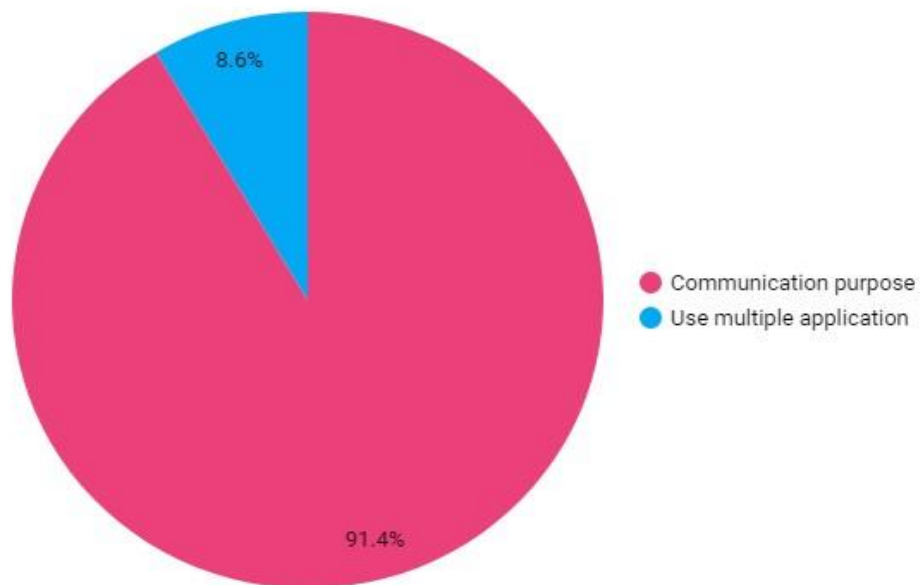


Figure 4: Respondent smartphone usage behavior

(iv) Respondent Perception on the Mobile Application for Web-Based System Enhancement

The 70% of respondents agreed that it would be helpful to have a mobile application for enhancing the existing gate pass management system. The 22.9% suggested a responsive web system to replace the existing system. The remaining 7.1% suggested purchasing of extra computer for all employees. Respondents' perception is illustrated in Fig. 5.

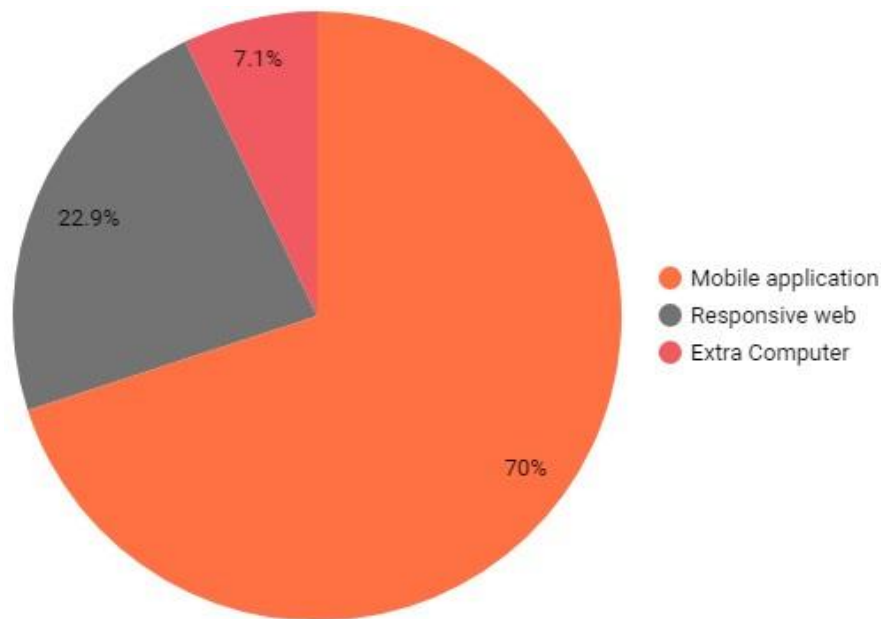


Figure 5: Respondent's perception of existing system enhancement

4.1.2 Mobile Application Requirements

The requirement gathered for mobile application development defines the specification of required mobile application to meet user needs. The mobile application requirements were gathered from the different stakeholders at A to Z Textile Mills Limited. Stakeholders include existing system users, existing system developers, and existing system administrators. Also, requirements were gathered through reviewing and observing the existing gate pass management system. The following are functional and non-functional requirements of gate pass management mobile application.

(i) Functional and Non-Functional Requirements

The defined requirements of mobile applications were grouped into functional requirements that focus on the functional aspect of the required mobile application and non-functional requirements

that focus on the operation of the mobile application. Table 2 and Table 3 show the identified functional and non-functional requirements respectively.

Table 2: Mobile Application functional requirements

Requirement category	Requirement description	System actor
Users account management	Each user should be able to set their account profile information	All Employees
Request visitor or employee gate pass	The mobile application should allow employees to create gate pass request	All Employees
View and edit gate pass request	Users should be able to view/edit her/his unapproved gate pass request	User requesting the gate pass
Notifications	User should get notification of gate pass approval	User requesting the gate pass
View and approve gate pass request	Head of department should be able to view and approve or reject gate pass request from other users	Administrator
Gate pass request Notifications	Heads of departments should receive notification of gate pass request	Administrator
View approved gate pass	A security guard should be able to view approved gate passes	Security guard
Allow exist or entrance	A security guard should be able to specify the entrance or exiting of employees or visitors to the company premises	Security guard
Scanning gate pass barcode	A security guard should be able to scan the barcode of the approved gate pass	Security guard
Taking picture	A security guard should be able to take pictures of permitted visitor or employee	Security guard

Table 3: Mobile application non-functional requirements

Requirement category	Requirement description
Security	The mobile application should authenticate all users before their interaction with the systems' functionalities
Reliability	The mobile application should be available to users all the time.
Usability	The mobile application should be simple, easy, and self-intuitive to use
Scalability	The mobile application should be able to accommodate new features and functions with fewer re-design issues.
Response time	The mobile application should have a short response time to users' requests
Operating System	The mobile application should run in any android device ranging from API level 23 (Marshmallow) and above.

4.1.3 Application Design

The gathered mobile application requirements were used to design the mobile application. The design involved defining mobile application layouts, mobile applications, and existing system interaction architecture. Use case diagrams and Entity Relationship Diagrams (ERD) were used to design the required mobile application.

(i) Application Architecture

The developed mobile application designed considered mobile application components, API for communication with existing gate pass management system, the structure of existing system database.

Mobile Application Components

The mobile application is divided into three components which are staff component, administrator components, and security guard component.

- **Staff Component**

The staff component comprises all employees with no administrative privileges. The component allows staff to request a personal gate pass, create gate pass for a visitor, view his/her gate passes, and editing gate pass which is still waiting for approval.

- **Admin Component**

The admin component authorizes staff designated as admins to add new application users (staff), to define and manage user rights, approve or disapprove gate pass requests, view all gate pass requests, and create visitor gate passes.

- **Security Guard Component**

This is designed for security guards at the entrance gate. The components will provide access to the approved gate passes, the gate passes are grouped as visitor response and employee response. Also, security personnel will gate access to the barcode scanner which will be used to scan the generated barcode for exit or entrance permission. The security component is integrated into the Barcode scanner, the scanner uses a smartphone's camera to scan a barcode from a printed gate pass. Lastly, the application will provide the option to capture pictures of permitted visitors.

(ii) Application Programming Interface

An API is used to enable data transmission between an existing gate pass management system and a mobile application. Communication between a web-based system and mobile application is interfaced by API that developed using the slim framework. For web-based gate pass management system enhancement, the private API was used to integrate the mobile application and the system, this allows the company to fully control the API usage. Figure 6 illustrates the architecture of the developed mobile application for gate pass management system enhancement.

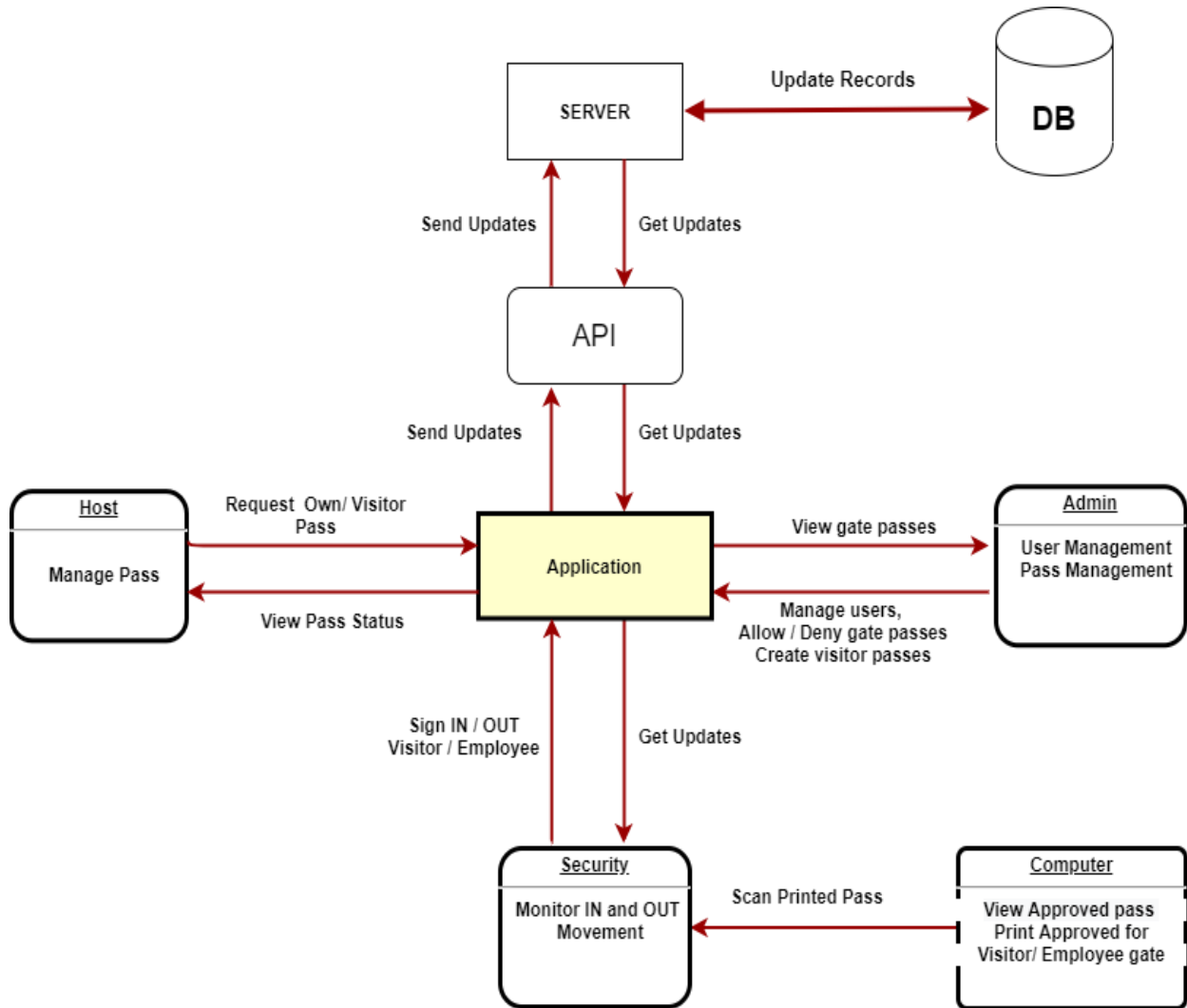


Figure 6: Application Architecture

(iii) Use Case Diagram

The use case diagrams are used to illustrate the interaction between mobile application functionalities and actors. The main actors of the mobile application were staff, designated administrators, and security guards. Figure 7 represents a mapping of a mobile application use case and its interaction with actors.

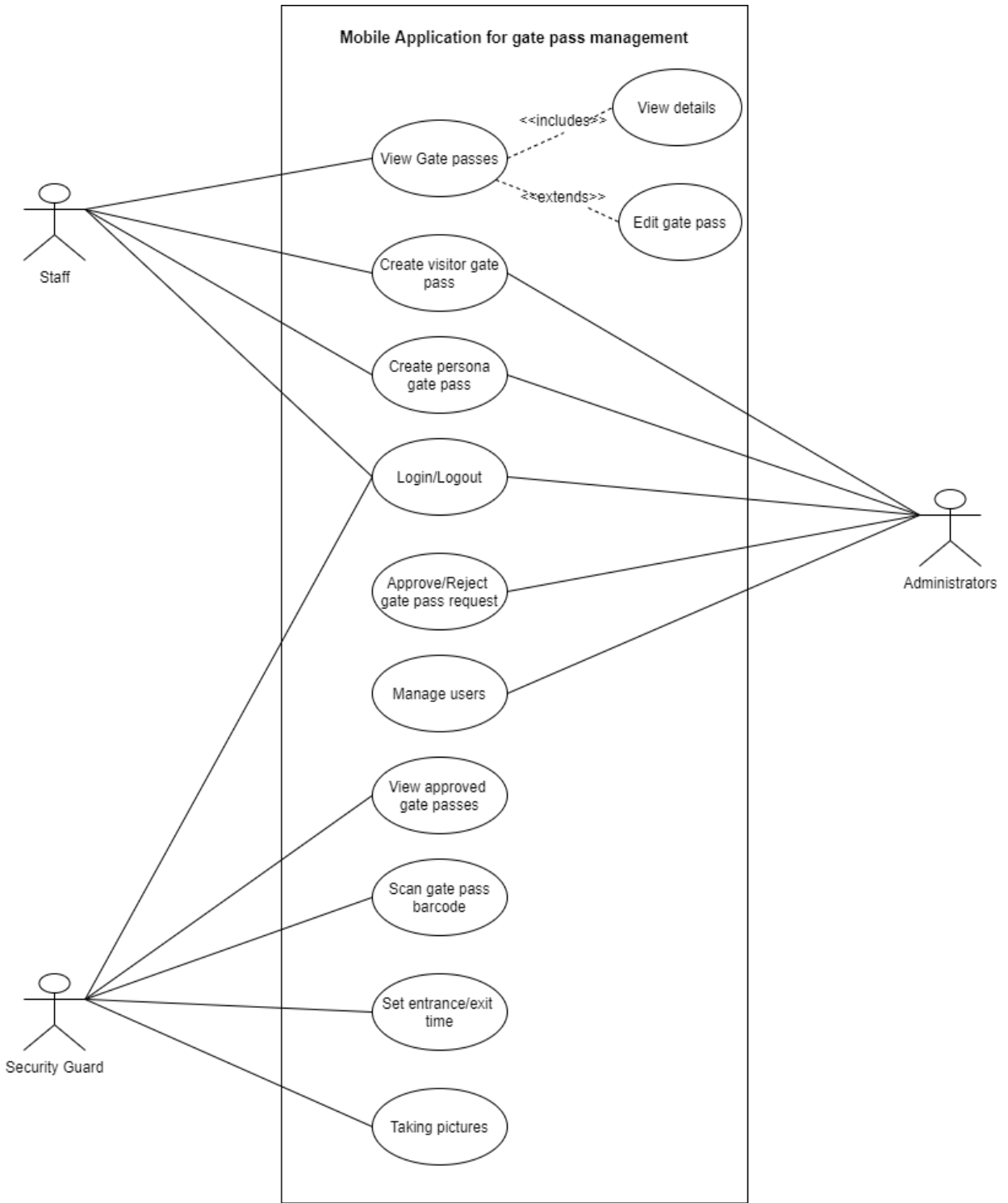


Figure 7: Use case diagram

The use case shown in the previous Fig. 7 has been described in detail in a Table 4.

Table 4: Description of use cases

Use case	Description	Actor
View gate passes	User can view her/his unapproved or unused gate pass	Staff
Create visitor gate pass	User can create gate pass for visitors	Staff and administrators
Create personal gate pass	Users can request gate pass to move out of company during working hours	Staff and administrators
Login and logout	Users can log in to the system by entering their credentials	Staff, security guards, administrators
Approve or reject gate pass request	User can approve the requested gate pass or user can reject the request	Administrator
Manage user	User can add, remove or deactivate other users	Administrator
View approved gate pass	User can view approved gate passes	Security guard
Scan gate pass barcode	User can scan gate pass barcode for providing permission	Security guard
Taking picture	User can take pictures of visitors	Security guard

(iv) Gate Pass Management System Database Structure

Microsoft SQL Server was used for existing gate pass management system implementation. The developed API provides the abstraction for a mobile application to store and retrieve data from this database. Analysis was conducted to understand the schema of the existing system database. The database contains four Tables follows:

Gate Pass Header Table

This Table is used to keep all information of gate pass requests. The Table contains gate pass reference number, the date of gate pass creation, type gate pass if it is either visitor or employee gate pass, the purpose for a gate pass, the status of the gate pass, validity time of the gate pass, employee information and administrator response.

Employee Gate Pass Details Table

The employee gate pass details Table is linked to the header table through the gate reference number. The Table holds detailed information of the employee gate pass. The information includes gate pass reference number from gate pass header Table, employee card number, mode of transport, materials he/she have, time in and time out, name of a security guard who allowed the employee to exit or enter and gate pass status.

Visitor Gate Pass Details Table

The visitor gate pass Table is linked to the gate pass header through the gate pass reference number. The visitor Table holds detailed information of the incoming visitor. The information includes gate pass reference number, the number of visitors, the material they have, phone number, in and out time, visitor photo, and name of security guards who allowed the visitor to enter or exit company premises, gate pass status.

User Login Information Table

User login information contains information of registered employees. The employee information details include employee card number, login username, password, user role, mobile number, user status, and user-created date.

(v) Gate Pass Management Database Schema

The database schema is used to show a relationship between Tables of the existing database management system. The database schema is shown to Fig. 8.

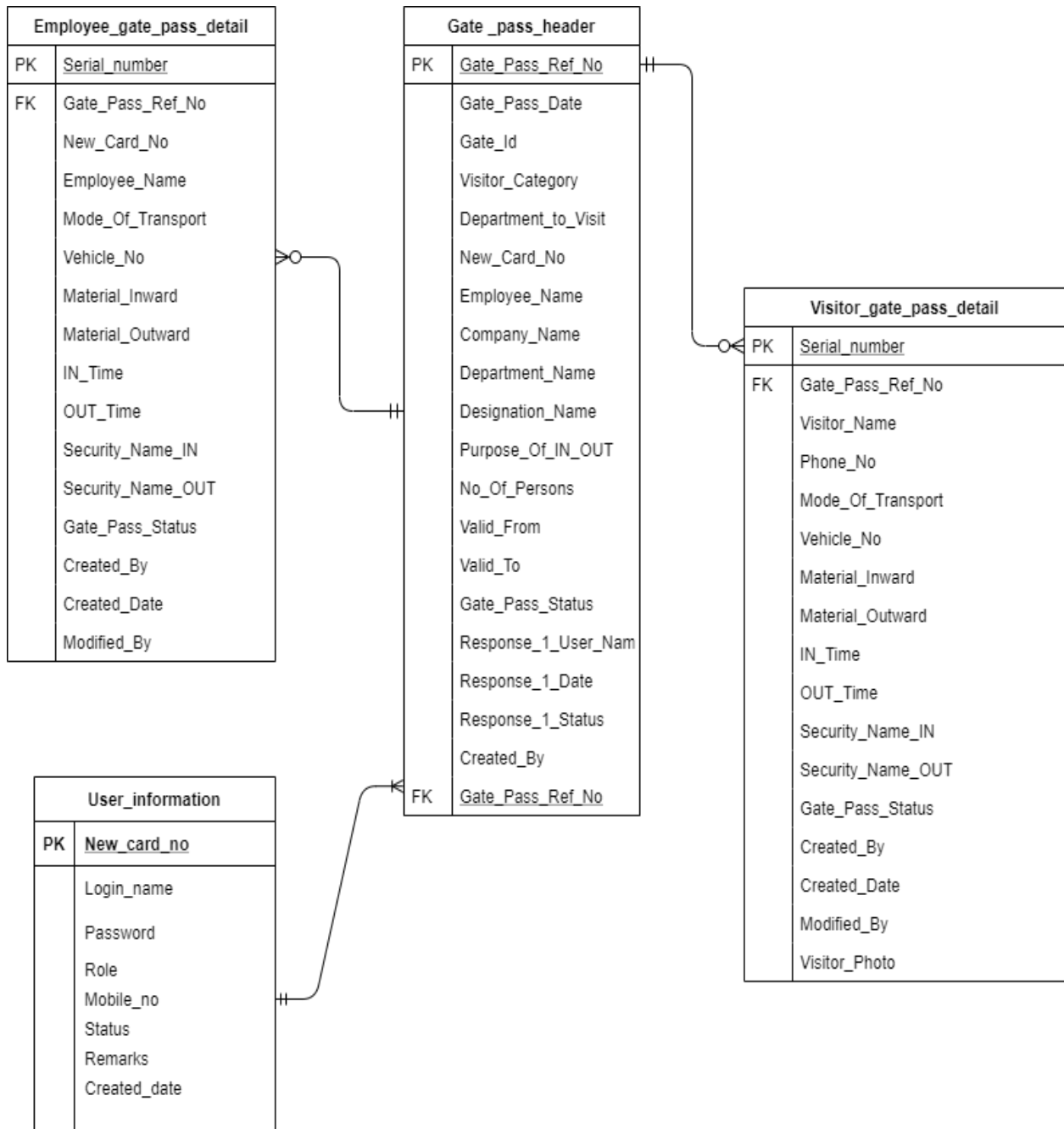


Figure 8: Database E-R diagram

4.1.4 Mobile Application Prototyping

This stage involved the development of a mobile application prototype, prototype evaluation, and refining of the application through user feedback. The prototyping phase started with user interface designing. The second phase involved the implementation of mobile application functionality which uses SQLite database to implement all basic application functionality before integration to gate pass management system. The prototype holds all logical functionality of the original software but is limited to the mobile phone local database. Users were provided a chance to interact and

evaluate the implemented prototype. The prototype was refined and reviewed based on user feedback. The refined prototype was integrated into the web-based gate pass management system through API. Users were provided an opportunity to interact and evaluate the integrated prototype. The integrated prototype was reviewed and refined according to the user feedback. The approved prototype was used to develop the final mobile application.

(i) Mobile Application Development

This stage involved the development of the final mobile application by using the approved prototype. During the development stage, the final mobile application was fully integrated into the web-based gate pass management system. The developed mobile application comprises all functional and non-functional requirements as well as user feedback from the prototype. The following tools and technology were used during the prototyping and development of the gate pass management mobile application.

Android Platform

The mobile application was developed to work in android based smartphones. The android platform allows developers to develop a mobile application through Java, Kotlin, or Dart programming language. Android platform has more user base because of its large market share. Currently android stands to 71.81% of the market share compared to other platforms. The market share of different mobile operating systems is shown in Fig. 9.

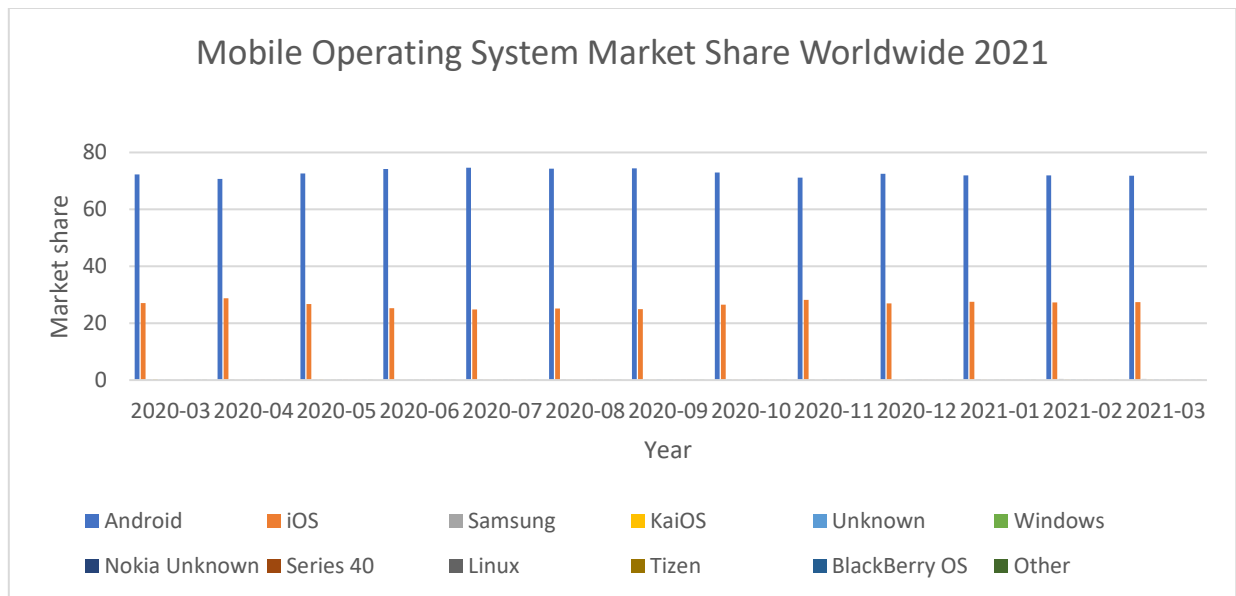


Figure 9: Market Share of mobile operating systems (Statcounter, n.d)

Java Programming Language

Java is an object-oriented programming language released in 1995 as a core component of Sun Microsystems. Compared to other languages for android application development Java is the most popular language used for Android application development. Java was selected to develop a gate pass management mobile application.

Android Studio

Android Studio is an Integrated Development Environment (IDE) designed for mobile application development. Gate pass mobile application was implemented using the Android Studio development environment. Android studio is bundled with Android Software Development Kit (SDK) which provides all requirements and libraries for android application development.

SLIM PHP Framework

Slim is a PHP lightweight framework that is used to create APIs endpoints for a mobile application. The slim framework was used to create Representational State Transfer (REST) API for a mobile application to communicate with a web-based gate pass management system. The slim framework supports all HyperText Transfer Protocol (HTTP) including GET, POST, PUT, DELETE.

Internet Information Services

Internet Information Services (IIS) is a flexible, general-purpose web server from Microsoft that can be installed in Windows Operating System to respond to requests from remote computers. The existing gate pass management system which requires enhancement through a mobile application is hosted in the Internet Information Services web server, therefore all images captured through a mobile application will be uploaded to this web server.

Microsoft SQL Server Database

Microsoft SQL Server is a relational database management system (RDBMS) developed by Microsoft. The primary function of the Microsoft SQL database server is for storing and retrieving data as requested by applications. The existing system was implemented through Microsoft SQL Server. Therefore, the mobile application uses the API to store and retrieve data from the same database as the existing system.

JavaScript Object Notation

JavaScript Object Notation (JSON) is a lightweight data-interchange format used to transfer data between the mobile application and the existing web-based gate pass management system. The JSON is a language-independent data format, thus was preferred for transferring data from mobile application to webserver as well as from web server to the mobile application.

(ii) Mobile Application Implementation Results

The mobile application developed is based on android which leverages essential features offered by the Android operating system. Android platform was selected due to its large market share and many users in the Africa market. Therefore, Android ensures a large user base for the gate pass mobile application. The mobile application provides various features to enhance existing gate pass management. The following are detailed features provided by mobile application for enhancement of gate pass management system.

Authentication

All users should be registered in the existing gate pass management system database before using the mobile application. The mobile application uses the existing logins accounts. The main system database contains username, password as well as user roles for authorizing a user to perform different activities through a mobile application. The roles involve staff, security guards, and administrators. During authentication, a user is required to enter a username and password to log in as shown in Fig. 10. The credentials will be checked to the database. If user credentials are wrong, the user will be prompted to enter the correct credentials or contact the system administrator.

For the correct credential, the API will return user information such as user names, department, designation as well as user role. The returned user role will be used to determine the authorization of users. Staff and administrators will be directed to the main screen that provides an option to create visitor gate pass, employee gate pass as well as viewing gate pass requests (Fig. 11). Security will be directed to the security control screen.

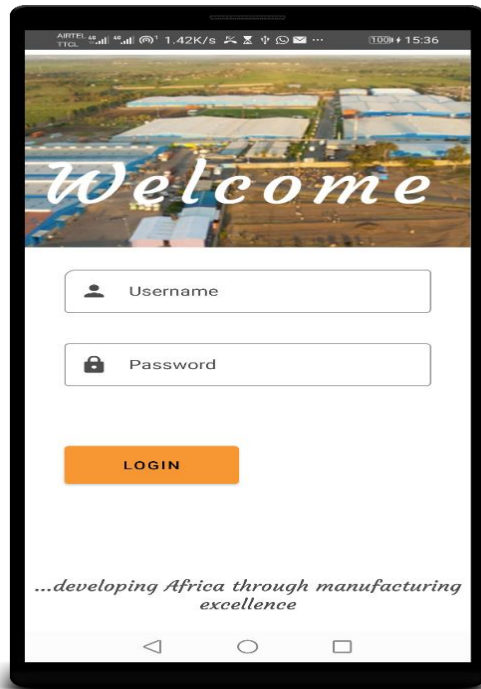


Figure 10: Login screen



Figure 11: Users and Admins screen

Creating Gate Pass Request

The mobile application enables a user to create a gate pass request. The gate pass requests are categorized into employee gate pass and visitor gate pass. If an employee wants to move out of the company premises during working hours, he/she will create an employee request. For the case of

a visitor, the employee will use a visitor gate pass. The following are stages to create employee or visitor gate pass.

- **Creating Employee Gate Pass**

Creating an employee gate pass is accomplished through two stages. The first stage requires general information of the gate pass request (Fig. 12). The information required includes gate pass date, type of the gate pass if is temporary or permanent, the purpose of the gate pass, number of vehicles, any remarks, lastly expected out time and in time. The second step requires detailed information of the outgoing employees as shown in Fig. 13. An employee will enter the card number to search and retrieve other employee details such as names, departments, and designation. Thereafter, employee is required to fill in the materials he/she will have when going out and coming back, vehicle number, and the number of employees who will share the same gate pass.

The screenshot shows a mobile application interface for creating an employee gate pass. The screen is titled "CREATE EMPLOYEE GATE PASS" and has a "FRESH" status. The form includes the following fields and buttons:

- Gate Pass Date:** A date picker field showing "21-Apr-2021 14:38:53".
- Gate Pass Type:** A dropdown menu with "Temporary" selected.
- Gate Name:** A text input field with "Gate Name" as a placeholder.
- Purpose:** A text input field.
- No of Vehicles:** A text input field with a car icon and "No of Vehicles" as a placeholder.
- Remarks:** A text input field.
- Expected Out Time:** A date picker field with "Pick Date" as a placeholder.
- Expected In Time:** A date picker field with "Pick Date" as a placeholder.
- Buttons:** A red "CLEAR" button and an orange "CONTINUE" button.

Figure 12: Gate pass information

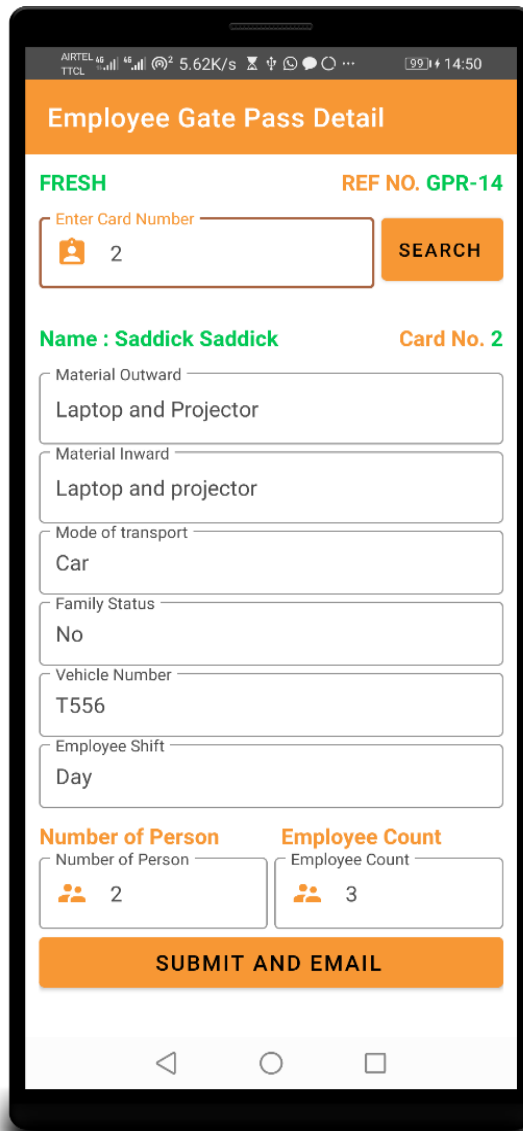


Figure 13: Employee gate pass detail

- **Creating Visitor Gate Pass**

Also, the creation of visitors' gate pass requests is divided into two steps. The first step requires information on the gate pass and information of the hosting employee. The host information identifies the responsible person for the incoming visitor. Therefore, an employee should enter a card number and then retrieve the required information from the employee database (Fig. 14). Employee information includes employee name, department, designation, and phone number. Also hosting employees should specify the number of incoming visitors, visitation purpose, and time validity of the gate pass. The second step requires detailed information of the coming visitor (Fig. 15). Visitor information includes visitor name, visitor company, visitor phone number, visitor email, department to visit, mode of transport, and vehicle number.

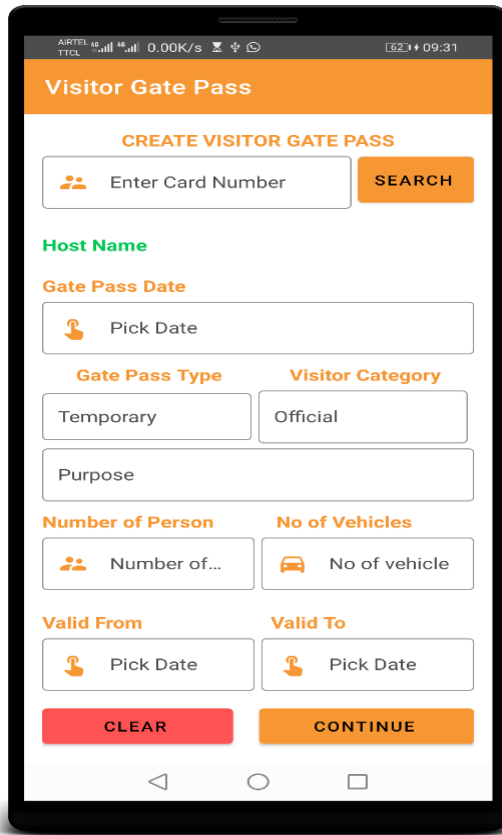


Figure 14: Visitor gate pass information

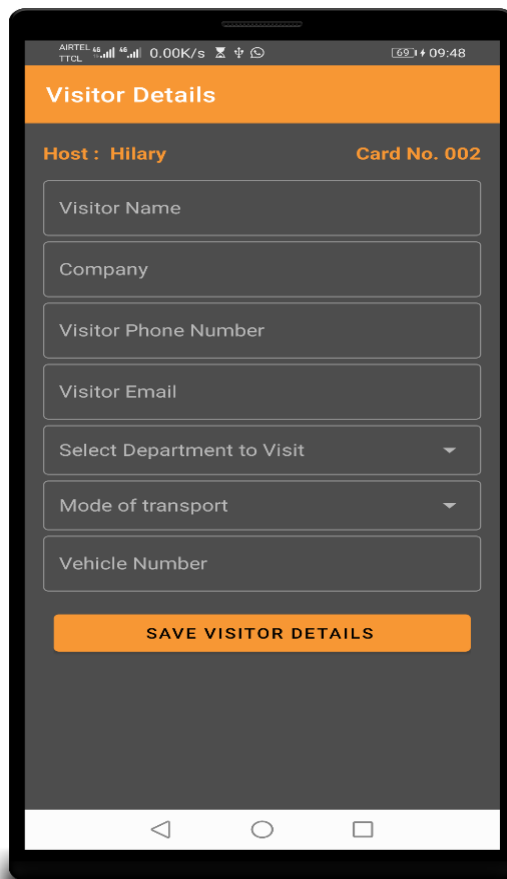


Figure 15: Visitor details

Viewing Gate Passes

The gate pass request can be viewed by the administrator or other staff. Administrators view all the gate pass requests while staff views personal gate pass requests. On viewing the gate pass administrator gates options to approve or reject the gate pass request as shown in Fig. 16.



Figure 16: Admin gate pass request view

On the other side, staff can view or edit personal gate pass requests before approval as shown in Fig. 17. After approval of the gate pass, the gate pass will be accessible to the security guards at the entrance gate.



Figure 17: Staff gate pass request view

Updating and Viewing Gate Pass Detail

The mobile application allows users to update unapproved or rejected gate passes, the gate pass will not show a barcode and cannot be shared as shown in Fig. 18.



Figure 18: Unapproved gate pass

User cannot change, rejected or approved gate pass, the approved gate pass contain gate pass details as well as a shareable barcode as shown in Fig. 19.



Figure 19: Approved gate pass details

Security Control and Permission

The mobile application at the main gate provides three functionalities to the security guards. These functionalities are a barcode scanner, taking a picture, and setting the entrance and exit time of the gate pass user. During authentication, security personnel is directed to the security main screen as shown in Fig. 20. The security main screen provides two options namely visitor response and employee response. Visitor response is used to provide permission to the visitors while employee response is for employees only. When opening either of the response, a security guard will gate access to the barcode scanner which will be used to scan the barcode generated after gate pass approval.

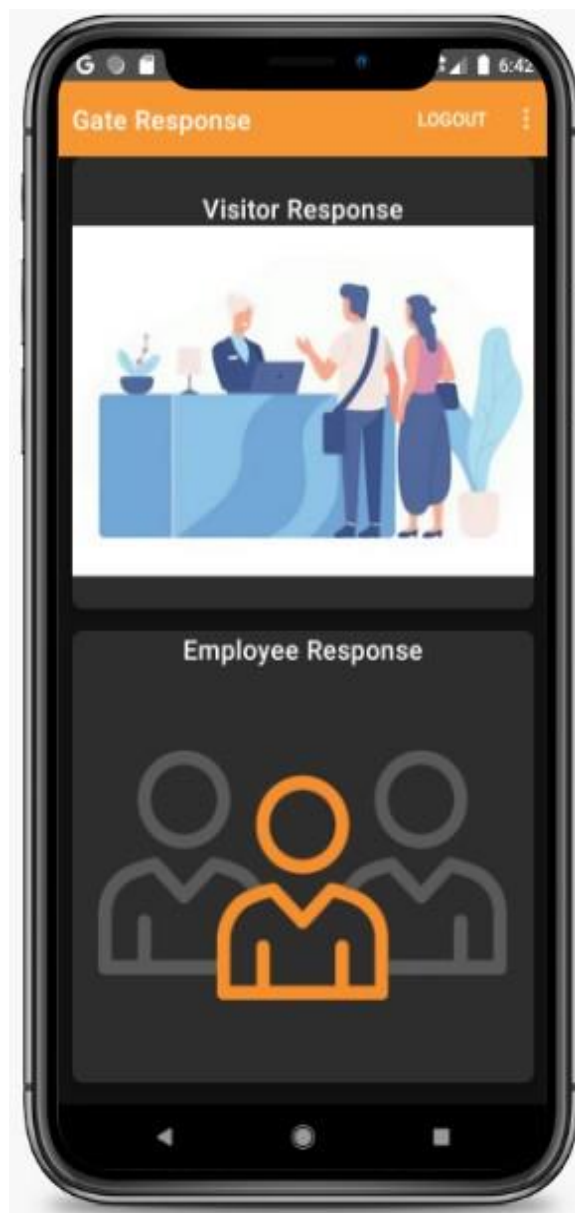


Figure 20: Security screen

- **Scanning Gate Pass Barcode**

The scanner functionality enables the security guard to scan the generated barcode to get the gate pass reference number (Fig. 21).



Figure 21: Gate pass barcode scanner

The reference number is used to find other gates pass information from the database. Depending on the current status of the gate pass. If the gate pass has been used for out movement, a security guard will get in button and option to update materials carried by the gate pass user. If clicking in button the application automatically will pick the current time as in time. If it is fresh, then the security guard will get out button and option to update materials carried by gate pass user as shown in Fig. 22.



Figure 22: Security permission

- **Taking Pictures**

Taking pictures of incoming or outgoing visitor functionality is embedded in the mobile application as shown in Fig. 23. The feature utilizes a mobile phone camera to take pictures for extra security records. The security guard can use this feature to take up to five pictures of the gate pass user.

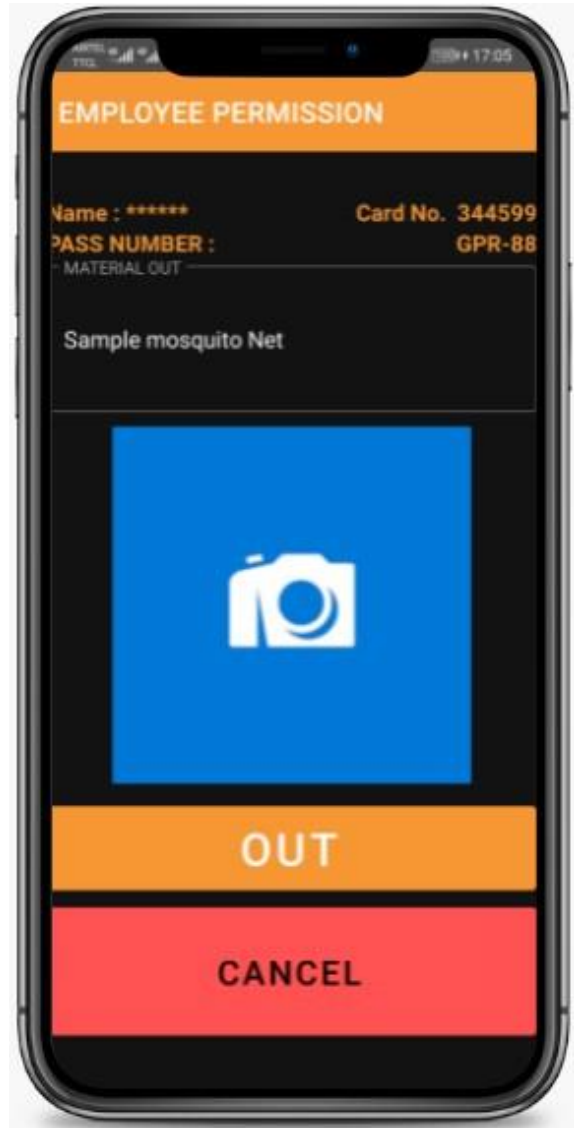


Figure 23: Camera feature to take pictures

All pictures of gate pass users are uploaded to the main server through a mobile application. The pictures are saved by using the gate pass number for easy identification. Figure 24 shows sample pictures of gate pass users.

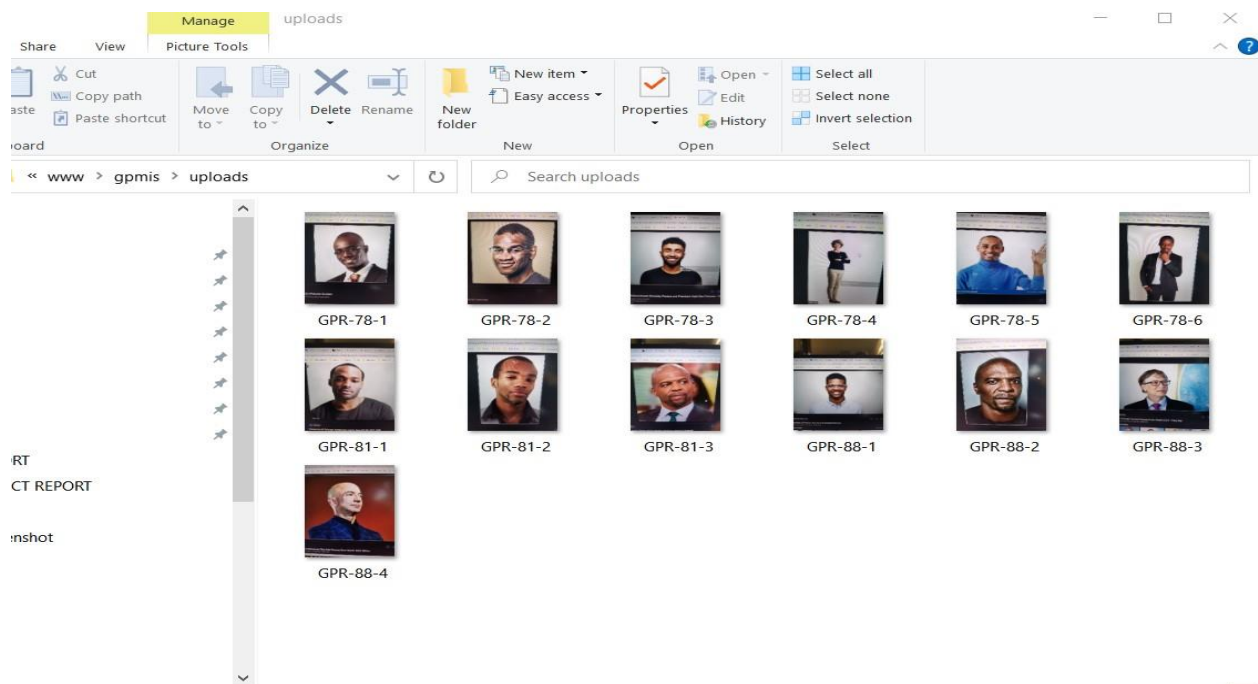


Figure 24: Sample gate pass picture

4.1.5 Mobile Application Testing

Mobile application validation involved both application performance testing and user acceptance testing. Mobile application testing was carried out to ensure the mobile application meets all identified requirements as well as user needs. The testing involved unit testing, integration testing, and system testing.

(i) Unit Testing

Unit testing was performed to test individual components of the mobile application and the API. This was performed to ensure each component meet the specified requirements. Components tested are authentication, Creating gate pass, Barcode scanner, Camera, API.

Authentication Testing

Authentication testing was performed to test if a user can log in and log out of the mobile application. The login unit was designed and implemented as the first part of the mobile application. User roles are determined through the authentication unit. The unit tested with different users of different roles.

Creating Gate Pass

The component for creating employee and visitor gate pass was implemented and tested separately. All functionalities of creating gate pass tested to ensure users can create visitors and employee gate passes. The SQLite database used to test creating gate pass unit. The SQLite is a built-in database of Android. Therefore, there was no need for API or external databases to test creation, update and approve of the gate passes.

Barcode Scanner

The Barcode scanner module was tested as a different unit from other components to ensure this unit returns information of the scanned barcode. During implementation different barcode scanner libraries such as google mobile vision library and ZXing. During unit testing the ZXing library performed well, therefore it was selected for the barcode scanner unit.

API Testing

All API links and operations were tested before integration with the mobile application. Postman API testing tool was used to test the functionalities of the developed API. Through postman the API all API operations were tested to ensure data was received and returned according to the defined formats.

(ii) Integration Testing

Integration testing was performed to test the communication and interaction of integrated components. This involved integration and the testing response of integrated components. The authentication component was integrated into the API for retrieving user information. Creating gate pass components was integrated into the API to store and retrieve user data from the database.

(iii) System Testing

The integrated components are tested as a single system to ensure the mobile application meets user requirements. The testing involved authentication to the creation of gate pass, gate pass barcode scanning, gate pass update, gate pass approval, and security gate pass control. Table 5 shows the results of mobile application testing.

Table 5: Description of use cases

System requirements	Results
The application must allow a registered user to login and logout	PASS
User can create visitor or employee gate pass	PASS
User can view, edit, approve gate pass	PASS
User can scan gate pass barcode	PASS
User can take pictures of gate pass owner	PASS

4.1.6 Mobile Application Validation

After the mobile application implementation and testing by the technical staff of A to Z Textile Mills Limited, the mobile application was taken to the company staff for usability and effectiveness validation. Hundred people participated in this activity. The survey responses were collected on three-point scale responses of Strong Agree = 3, Agree = 2, and Disagree = 1. Table 6 shows the mobile application acceptance test results.

Table 6: Mobile application acceptance testing

Acceptance test	Strong agree	Agree	Disagree
There are no compatibility issues between the smartphone and mobile application	77	19	4
I do not need further training and assistance on how to fully navigate to different application user interfaces (UI)	86	12	2
I am satisfied with the overall mobile application performance	73	26	1
The mobile application is easy to use and interact	80	17	3
Mobile application shows correct employee details	100	0	0
User can send gate pass to visitor	100	0	0
User can get timely notification	80	16	4
Users can update personal information	100	0	0

4.2 Discussion

The prevalence of smartphones and internet technology signifies that access is no longer one-way communication. Instead, users can interact with provided services anytime and anywhere (Hajiheydari & Ashkani, 2018). Due to availability and accessibility nature of mobile devices,

many users prefer mobile application rather than web-based system. Gate pass mobile application was implemented to utilize the mobile application features as well as providing easy interaction with the existing web-based system. The project is not intended to replace the existing gate pass management system rather enhance the existing system. The following are the features of mobile phones utilized for the enhancement of the gate pass management system.

4.2.1 Portability

Mobile phones can be carried and used anywhere and anytime; this enables users to access services delivered through a mobile application (Asta *et al.*, 2017). The idea of gate pass management system enhancement through a mobile application is to ensure users can interact with the existing web-based system anywhere and anytime.

4.2.2 Camera

A mobile phone camera was utilized to enhance the gate pass management system for scanning barcodes and taking pictures of gate pass users. The barcode module implemented to the gate pass mobile application uses a phone camera to scan the gate barcode for security permission. Also, the application uses a phone camera to take pictures of gate pass users.

4.2.3 Notification

Mobile application is more efficient for users to receive notification in real-time compared to web-based systems (Overview & Gate, 2020). The increases response time to the concerned user. Gate pass mobile application uses a notification feature to notify an administrator when there is a gate pass request. Also, staff can get notification of gate pass approval.

Due to the need for gate pass management system enhancement, this project implemented gate pass mobile application to increase system accessibility, usability, and flexibility. The potential mobile application users tested and verified how the mobile application improved gate pass management processes.

The gate pass application store and retrieve all gate pass data to the same Microsoft SQL server database used by existing gate pass management system. The interaction with the main system database is achieved through an API. The data includes employee and visitor gate information, gate pass user's pictures, and any other documents.

Furthermore, the mobile application helps to reduce the number of devices required for gate pass management. For the web-based gate pass management system device such as computers, printers, and barcode scanners are required to manage gate pass processes. Computers are used to create and approving of gate pass requests. This means users should have access to the computers to create gate pass requests. Printer and barcode scanner used by a security guard to print approved gate pass. The scanner was used to scan this printed gate pass. The implemented mobile application managed to integrate all these features into a single mobile device. This application greatly reduces device cost and the requirement of papers as the gate pass can be scanned directly from the user's smartphone.

Lastly taking gate pass user pictures helps to keep visible records for security purposes. These pictures can be reviewed in case of any security breach. Therefore, mobile application enhancement of the existing gate pass management system has improved the process of gate pass management.

However, the mobile application is based on the Android operating system. Mobile operating system dependence limits the number of mobile application users because the application can be installed only on the Android smartphone only.

CHAPTER FIVE

CONCLUSION AND RECOMMENDATIONS

5.1 Conclusion

This study aimed to evaluate the existing gate pass management system, and thus developing a mobile application to enhance the existing web-based gate pass management system. The implementation focused on the enhancement of the existing system to increase availability, flexibility, and usability. The study had three research questions as follows:

- (i) What are the requirements for the enhancement of a web-based gate pass management system?
- (ii) What mobile application can be implemented for enhancing the existing web-based gate pass management system?
- (iii) Does the implemented mobile application enhance the existing gate pass management system?

In the first question, a literature review and analysis were conducted to evaluate the existing gate pass management system. User and mobile application requirements were identified after a web-based gate pass management system review. The requirements identified for gate pass mobile application development.

The second question was fulfilled through the development of a mobile application prototype using prototyping methodology. Modeling techniques, tools, and technologies such as use cases and E-R diagrams were used during mobile application implementation. The study resulted in the development of gate pass mobile application that utilizes a mobile phone's camera, notifications, and portability for the enhancement of web-based gate pass management.

The developed mobile application will help to reduce the number of devices required for gate pass management. Usually, a web-based gate pass management system requires computers for users to request gate pass, printer and scanner for security guard to print gate pass permission. The scanner is used to scan the printed gate pass document for verification. The mobile application provides all the functionality in a single mobile phone with the added functionality of taking gate pass user pictures through a mobile phone camera.

The developed mobile application was tested and validate at A to Z Textile Mills Limited by users of different roles to determine if the application met all identified requirements. Users and stakeholders were satisfied with the developed mobile application on the enhancement of the

After the implementation of the mobile application, acceptance testing was conducted by involving potential stakeholders at A to Z Textile Mills Limited. The testing was performed where users from the ICT department participated during in-house testing of the application and their comments were used to improve some features before going to other users.

Finally, validation was performed where 100 potential users were involved. The application was installed on the smartphone and it was in use for 10 days. The application effectiveness was validated against the identified requirements.

5.2 Recommendations

Based on the results of gate mobile application for the enhancement of the web-based gate pass management system, it is recommended to enhance the web-based system through a mobile application for reaching many users and providing a good user experience. This will reduce the amount of time and devices required for managing gate passes. Also, the notification feature of mobile phones helps to provide real-time responses.

The scope of this study was to enhance gate pass management system through a mobile application. Android was a platform of choice due to its large customer base. To increase and reach more users the mobile application can be developed for both iOS and Android users.

This study explores techniques that could be used to utilize mobile devices for enhancing web-based systems. Utilization of camera to replace barcode scanner, notification for real-time response, creating digital gate pass for replacing printout documents as well camera for photographing gate pass users. Therefore, the study opens more possibility to reach more users through utilization of mobile devices.

REFERENCES

- Amazon. (n.d). *Recycled Visitor Register Book*. <https://www.amazon.com/AT-GLANCE-8058005-Recycled-Register/dp/B00006IAWT?th=1>
- Asta, T., Shahrok, N., & Rimantas, G. (2017). Mobile application driven consumer engagement. *Telematics and Informatics*, 34(4), 145–156.
- Bessghaier, N., & Souii, M. (2018). *Towards usability evaluation of hybrid mobile user interfaces. Proceedings of International Conference on Computer Systems and Applications, 2017-October*, 895–900. <https://doi.org/10.1109/AICCSA.2017.10>
- Sabale, G. R. (2012). Comparative Study of Prototype Model for Software Engineering with System Development Life Cycle. *Journal of Engineering*, 2(07), 21–24. <https://doi.org/10.9790/3021-02722124>
- Gowtham, I., Sathishkumar, T., Lakshmiprasad, S., Arumugam, P., & Rao, G. P. (2019). *Automation of Visitor Gate Pass Management System. The 2019, 2nd International Conference on Intelligent Computing, Instrumentation and Control Technologies*. <https://doi.org/10.1109/ICICICT46008.2019.8993173>
- Gunawan, H., & Abdul, K. E. (2017). *Integration protocol student academic information to campus RFID gate pass system. International Conference on Electrical Engineering, Computer Science and Informatics*. <https://doi.org/10.1109/EECSI.2017.8239197>
- Hajiheydari, N., & Ashkani, M. (2018). Mobile application user behavior in the developing countries: A survey in Iran. *Information Systems*, 77, 22–33. <https://doi.org/10.1016/j.is.2018.05.004>
- Harish, R., Pushpanjali, S., & Omkar, S. (2017). Visitor Gate Pass Management System. *Visitor Gate Pass Management System*, 6(2), 170–178.
- Hussain, A., & Mkpojiogu, E. O. C. (2015). The effect of responsive web design on the user experience with laptop and smartphone devices. *Jurnal Teknologi*, 77(4), 41–47. <https://doi.org/10.11113/jt.v77.6041>
- Hussain, A., Mkpojiogu, E. O. C., & Hussain, Z. (2015). Usability evaluation of a web-based health awareness portal on smartphone devices using ISO 9241-11 model. *Jurnal Teknologi*, 77(4),

1–5. <https://doi.org/10.11113/jt.v77.6035>

Hussain, A., Mkpojiogu, E. O. C., Musa, J., & Mortada, S. (2017). *A user experience evaluation of Amazon Kindle mobile application. AIP Conference Proceedings, 1891*. <https://doi.org/10.1063/1.5005393>

Islam, R., & Mazumder, T. (2010). Mobile application and its global impact. *International Journal of Engineering, 06*, 72–78. <http://ijens.org/107506-0909/IJET-IJENS.pdf>

Liu, L. (2016). Using Generic Inductive Approach in Qualitative Educational Research: A Case Study Analysis. *Journal of Education and Learning, 5*(2), 129-135. <https://doi.org/10.5539/jel.v5n2p129>

Ma, Y., Liu, X., Liu, Y., Liu, Y., & Huang, G. (2018). A Tale of Two Fashions: An Empirical Study on the Performance of Native Apps and Web Apps on Android. *Transactions on Mobile Computing, 17*(5), 990–1003. <https://doi.org/10.1109/TMC.2017.2756633>

Mason, P. (2005). Visitor management in protected areas: From “hard” to “soft” approaches? *Current Issues in Tourism, 8*(2–3), 181–194. <https://doi.org/10.1080/13683500508668213>

Gate Pass Management System. (2020). *Project Overview Of Gate Pass*. <https://www.google.com>

Prerana, K., Vaishnavi, P., & Vikas, S. (2021). Gate Pass Management System using Android. *International Journal of Advance Research and Innovative Ideas in Education, 7*(3),1-6.

Rikshit, M. M., Romil, N. M., Jain, M. J., Tejas, L. M., & Sabnis, M. S. (2008). VISITX: Face Recognition Visitor Management System. *International Research Journal of Engineering and Technology, 6*(3), 1-5.

Rodrigues, R., Pavate, A., Sawant, R., & Lopes, N. (2021). Smart Gate Pass Security Management System Using Random Key Generation. *International Journal of Innovative Research in Computer Science & Technology, 9*(3), 1-6. <https://doi.org/10.21276/ijircst.2021.9.3.10>

Shree, M., Chopdar, S., & Shambhavi, P. (2019). Gate Pass System. *International Journal of Engineering and Advanced Technology, 9*(2), 1-4.

Statcounter. (n.d). *Market Share of mobile operating systems*. <https://gs.statcounter.com/os-market-share/mobile/worldwide>

- Sunico, R. J. A., Argana, E. S., & Dumale, M. M. (2020). Development and evaluation of automated gate pass system. *International Journal of Advanced Trends in Computer Science and Engineering*, 9(5), 8846–8850. <https://doi.org/10.30534/ijatcse/2020/279952020>
- Taherdoost, H. (2018). Sampling Methods in Research Methodology: How to Choose a Sampling Technique for Research. *Electronic Journal*, 5(2), 18-27. <https://doi.org/10.2139/ssrn.3205035>
- Venkatesa, P., & Sanju, R. (2018). Online Gate Pass Application form for Hostel Students. *International Journal of Pure and Applied Mathematics*, 119(18), 1657-1664.
- Yogesh, S. W., Ankita, R. S., Harshala, V. G., Komal, B. G., & Apurva, R. G. (2017). Security System for Industrial Gate And Generation of Gate Pass. *International Journal on Recent and Innovation Trends in Computing and Communication*, 5(2), 260-264, doi: 10.17762/ijritcc.v5i2.211.

APPENDICES

Appendix 1: The Sample Codes for Mobile Application Development

(i) Authentication

```
package com.azpfl.gatepassv2;

import androidx.appcompat.app.AlertDialog;
import androidx.appcompat.app.AppCompatActivity;

import android.content.DialogInterface;
import android.content.Intent;
import android.database.Cursor;
import android.os.Bundle;
import android.view.View;
import android.widget.Button;
import android.widget.Toast;

import com.azpfl.gatepassv2.Database.DatabaseManager;
import com.azpfl.gatepassv2.Model.EmployeeModel;
import com.azpfl.gatepassv2.Model.UserModel;
import com.google.android.material.dialog.MaterialAlertDialogBuilder;
import com.google.android.material.textfield.TextInputEditText;

import java.util.ArrayList;
import java.util.List;

public class Authentication extends AppCompatActivity {

    DatabaseManager databaseManager;
    TextInputEditText username,password;
    Button authenticate;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_authentication);

        username = findViewById(R.id.username);
        password = findViewById(R.id.password);
        authenticate = findViewById(R.id.login_button);

        authenticate.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                String cardNumber = username.getText().toString().trim();
                String userPassword = password.getText().toString().trim();

                if (cardNumber.isEmpty()){
```

```

        username.setError("Required");
    }
    else if(userPassword.isEmpty()){
        password.setError("Required");
    }

    /* else {

        databaseManager = new DatabaseManager(Authentication.this);
        List<UserModel> userModelList =new ArrayList<UserModel>();
        userModelList = databaseManager.userAuthentication(cardNumber,
userPassword);

        if (userModelList.isEmpty()){
            Toast.makeText(Authentication.this,"Wrong Password or Card_number",
Toast.LENGTH_LONG).show();
        }

        else {
            UserModel employeeData = userModelList.get(0);
            String mode = employeeData.getMode();

            if (mode.equals("Admin")){
                Intent intent = new Intent(Authentication.this, GatePass.class);
                startActivity(intent);
                finish();
            }

            else if (mode.equals("Staff")){
                Intent intent = new Intent(Authentication.this, Staff.class);
                startActivity(intent);
                finish();
            }

            else if (mode.equals("Security")){
                Intent intent = new
Intent(Authentication.this, Security_Gates_Permission.class);
                startActivity(intent);
                finish();
            }

            else {
                AlertDialog.Builder builder = new
MaterialAlertDialogBuilder(Authentication.this, R.style.AlertDialogTheme);
                builder.setTitle("Authentication Problem");
                builder.setIcon(R.drawable.icon_error);
                builder.setPositiveButton("OK", new
DialogInterface.OnClickListener() {
                    @Override

```

```

        public void onClick(DialogInterface dialog, int which) {

            }
        });

        builder.setMessage("Contact Administrator to check your Mode");
        AlertDialog alert = builder.create();
        alert.show();
    }
}
}*/

else {
    if(cardNumber.equals("1")){
        Intent intent = new Intent(Authentication.this, GatePass.class);
        startActivity(intent);
        finish();

    }

    else if(userPassword.equals("2")){
        Intent intent = new
Intent(Authentication.this, Security_Gates_Permission.class);
        startActivity(intent);
        finish();
    }

    else {

    }

}

    /* Intent intent = new
Intent(Authentication.this, Security_Gates_Permission.class);
    startActivity(intent);
    finish();*/

}

});

}

}

```

(ii) Viewing Gate Pass Request

```
package com.azpfl.gatepassv2;

import androidx.annotation.NonNull;
import androidx.appcompat.app.AppCompatActivity;
import androidx.recyclerview.widget.LinearLayoutManager;
import androidx.recyclerview.widget.RecyclerView;

import android.database.Cursor;
import android.os.Bundle;
import android.util.Log;
import android.view.View;
import android.widget.Switch;
import android.widget.Toast;

import com.android.volley.Request;
import com.android.volley.RequestQueue;
import com.android.volley.Response;
import com.android.volley.VolleyError;
import com.android.volley.toolbox.JsonArrayRequest;
import com.android.volley.toolbox.Volley;
import com.azpfl.gatepassv2.Database.DatabaseManager;
import com.azpfl.gatepassv2.Model.EmployeeModel;
import com.azpfl.gatepassv2.Model.GatePassModel;
import com.azpfl.gatepassv2.Model.VisitorModel;
import com.azpfl.gatepassv2.View.AdapterEmployeeGatePass;
import com.azpfl.gatepassv2.utils.Constants;
import com.google.android.material.floatingactionbutton.ExtendedFloatingActionButton;
import com.google.android.material.switchmaterial.SwitchMaterial;

import org.json.JSONArray;
import org.json.JSONException;
import org.json.JSONObject;

import java.util.ArrayList;
import java.util.List;

public class Gate_Passes_List extends AppCompatActivity {

    RecyclerView recyclerView;
    ExtendedFloatingActionButton extendedFab;
    DatabaseManager databaseManager;
    EmployeeModel employeeModel;
    List<EmployeeModel> employeeModelList = new ArrayList<>();

    SwitchMaterial allPassSwitch, requestPassSwitch, approvedPassSwitch;
```

```

private ArrayList Gate_pass_ID, Gate_pass_reference_number, Gate_pass_date,
Purpose,
Material_outward, Material_Inward, Mode_of_transport, Family_status,
Vehicle_number, Shift, Expected_out_time, Expected_in_time, Gate_pass_status;

```

```

AdapterEmployeeGatePass adapterEmployeeGatePass;

```

```

@Override
protected void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
    setContentView(R.layout.activity_gate_passes_list);

    extendedFab = findViewById(R.id.refresh_gate_pass_list);

```

```

    allPassSwitch = findViewById(R.id.switch_all_pass);
    requestPassSwitch = findViewById(R.id.switch_request);
    approvedPassSwitch = findViewById(R.id.switch_approved);
    recyclerView = findViewById(R.id.gate_pass_recycle_view);

```

```

    requestPassSwitch.setChecked(true);
    approvedPassSwitch.setChecked(false);
    allPassSwitch.setChecked(false);

```

```

    adapterEmployeeGatePass = new AdapterEmployeeGatePass(Gate_Passes_List.this,
employeeModelList);
    recyclerView.setAdapter(adapterEmployeeGatePass);
    recyclerView.setLayoutManager(new
LinearLayoutManager(Gate_Passes_List.this));

```

```

    fetchData();

```

```

    extendedFab.setOnClickListener(new View.OnClickListener() {
        @Override
        public void onClick(View v) {
            finish();
            overridePendingTransition(0, 0);
            startActivity(getIntent());
            overridePendingTransition(0, 0);
        }
    });

```

```

/*
    databaseManager = new DatabaseManager(Gate_Passes_List.this);
    Gate_pass_ID = new ArrayList<>();
    Gate_pass_reference_number = new ArrayList<>();

```



```

    Gate_pass_date = new ArrayList<>();
    Purpose = new ArrayList<>();
    Expected_in_time = new ArrayList<>();

    // storeDataInArray();

    List<EmployeeModel> employeeModelList=new ArrayList<EmployeeModel>();
    employeeModelList= databaseManager.getAllApprovedEmployeePass();

    List<VisitorModel> visitorModelList=new ArrayList<VisitorModel>();
    // visitorModelList= databaseManager.read_All_Data_Visitor_Gate_Pas_Table();

    List<GatePassModel> gatePassModelList = new ArrayList<GatePassModel>();
    GatePassModel obj = new GatePassModel();
    obj.set_id(0);
    obj.set_type("EMPLOYEE");
    obj.set_employeePassList(employeeModelList);
    gatePassModelList.add(obj);

    obj = new GatePassModel();
    obj.set_id(0);
    obj.set_type("VISITOR");
    obj.set_visitorsPassList(visitorModelList);
    gatePassModelList.add(obj);

    if (requestPassSwitch.isChecked()){
        employeeModelList= databaseManager.getAllRequestEmployeePass();

        adapterEmployeeGatePass = new
        AdapterEmployeeGatePass(Gate_Passes_List.this,employeeModelList);
        recyclerView.setAdapter(adapterEmployeeGatePass);
        recyclerView.setLayoutManager(new
        LinearLayoutManager(Gate_Passes_List.this));
    }

    approvedPassSwitch.setOnClickListener(new View.OnClickListener() {
        @Override
        public void onClick(View v) {
            if (approvedPassSwitch.isChecked()) {
                List<EmployeeModel> employeeModelList=new
                ArrayList<EmployeeModel>();
                employeeModelList= databaseManager.getAllApprovedEmployeePass();

                adapterEmployeeGatePass = new
                AdapterEmployeeGatePass(Gate_Passes_List.this,employeeModelList);
                recyclerView.setAdapter(adapterEmployeeGatePass);
            }
        }
    });

```

```

        recyclerView.setLayoutManager(new
LinearLayoutManager(Gate_Passes_List.this));

        requestPassSwitch.setChecked(false);
        allPassSwitch.setChecked(false);

    } else {
        List<EmployeeModel> employeeModelList=new
ArrayList<EmployeeModel>();
        adapterEmployeeGatePass = new
AdapterEmployeeGatePass(Gate_Passes_List.this,employeeModelList);
        recyclerView.setAdapter(adapterEmployeeGatePass);
        recyclerView.setLayoutManager(new
LinearLayoutManager(Gate_Passes_List.this));
    }
}
});

requestPassSwitch.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {
        if (requestPassSwitch.isChecked() ) {
            List<EmployeeModel> employeeModelList=new
ArrayList<EmployeeModel>();
            employeeModelList= databaseManager.getAllRequestEmployeePass();
            adapterEmployeeGatePass = new
AdapterEmployeeGatePass(Gate_Passes_List.this,employeeModelList);
            recyclerView.setAdapter(adapterEmployeeGatePass);
            recyclerView.setLayoutManager(new
LinearLayoutManager(Gate_Passes_List.this));

            approvedPassSwitch.setChecked(false);
            allPassSwitch.setChecked(false);

        } else {
            List<EmployeeModel> employeeModelList=new
ArrayList<EmployeeModel>();
            adapterEmployeeGatePass = new
AdapterEmployeeGatePass(Gate_Passes_List.this,employeeModelList);
            recyclerView.setAdapter(adapterEmployeeGatePass);
            recyclerView.setLayoutManager(new
LinearLayoutManager(Gate_Passes_List.this));
        }
    }
});

allPassSwitch.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {
        if (allPassSwitch.isChecked() ) {

```

```

        List<EmployeeModel> employeeModelList=new
        ArrayList<EmployeeModel>();
        employeeModelList= databaseManager.getAllEmployeePass();
        adapterEmployeeGatePass = new
        AdapterEmployeeGatePass(Gate_Passes_List.this,employeeModelList);
        recyclerView.setAdapter(adapterEmployeeGatePass);
        recyclerView.setLayoutManager(new
        LinearLayoutManager(Gate_Passes_List.this));

        approvedPassSwitch.setChecked(false);
        requestPassSwitch.setChecked(false);

    } else {

        requestPassSwitch.setChecked(true);

        List<EmployeeModel> employeeModelList=new
        ArrayList<EmployeeModel>();
        employeeModelList= databaseManager.getAllRequestEmployeePass();

        adapterEmployeeGatePass = new
        AdapterEmployeeGatePass(Gate_Passes_List.this,employeeModelList);
        recyclerView.setAdapter(adapterEmployeeGatePass);
        recyclerView.setLayoutManager(new
        LinearLayoutManager(Gate_Passes_List.this));
    }
}
});

*/
}

private void fetchData() {
    RequestQueue requestQueue= Volley.newRequestQueue(this);
    JsonRequest jsonArrayRequest= new JsonRequest(Request.Method.GET,
    Constants.VIEWPASSES, null, new Response.Listener<JSONArray>() {
        @Override
        public void onResponse(JSONArray response) {
            Log.d("cheki", "onResponse: " + response.toString());
            for (int i = 0; i < response.length(); i++) {
                try {
                    JSONObject gatePass = response.getJSONObject(i);
                    int id = gatePass.getInt("id");
                    String gpr = gatePass.getString("gpr");
                    String purpose = gatePass.getString("purpose");
                    String card_no = gatePass.getString("card_no");
                    String expectedTimeOut = gatePass.getString("expected_OUT_time");
                    String expectedTimeIn = gatePass.getString("expected_OUT_time");

```

```

        EmployeeModel gatePassModel = new EmployeeModel(id, card_no, gpr,
"", purpose, "", "", "",
        "", "", "", expectedTimeOut, expectedTimeIn, "", "",
        "", "");

        employeeModelList.add(gatePassModel);

    } catch (JSONException e) {
        e.printStackTrace();
    }
}

adapterEmployeeGatePass.setList(employeeModelList);

}
}, new Response.ErrorListener() {
    @Override
    public void onErrorResponse(VolleyError error) {
        Log.d("cheki", "errors: "+ error.getMessage());
        Toast.makeText(Gate_Passes_List.this,
error.getMessage(), Toast.LENGTH_SHORT).show();
    }
});
requestQueue.add(jsonArrayRequest);
}
}
}

```

(iii) Barcode Scanner

```

package com.azpfl.gatepassv2;

import androidx.appcompat.app.AlertDialog;
import androidx.appcompat.app.AppCompatActivity;
import androidx.core.app.ActivityCompat;
import androidx.core.content.ContextCompat;

import android.content.DialogInterface;
import android.content.Intent;
import android.content.pm.PackageManager;
import android.os.Build;
import android.os.Bundle;
import android.widget.Toast;

import com.google.android.material.dialog.MaterialAlertDialogBuilder;
import com.google.zxing.Result;

import me.dm7.barcodescanner.zxing.ZXingScannerView;

import static android.Manifest.permission.CAMERA;
public class BarCode_Scanner extends AppCompatActivity implements

```

```

ZXingScannerView.ResultHandler {

    private static final int REQUEST_CAMERA=1;
    private ZXingScannerView scannerView;

    String gate_pass_response_category;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_scanner);

        scannerView = new ZXingScannerView(this);
        setContentView(scannerView);

        Intent intent = getIntent();
        gate_pass_response_category = intent.getStringExtra("ResponseCategory");

        if (Build.VERSION.SDK_INT >= Build.VERSION_CODES.M) {
            if(checkPermission()){
                Toast.makeText(Barcode_Scanner.this,"Permission
granted",Toast.LENGTH_SHORT).show();
            }
            else {
                requestPermissions();
            }
        }
    }

    private boolean checkPermission(){
        return (ContextCompat.checkSelfPermission(Barcode_Scanner.this, CAMERA) ==
PackageManager.PERMISSION_GRANTED);
    }

    private void requestPermissions(){
        ActivityCompat.requestPermissions(this,new String[]{CAMERA},
REQUEST_CAMERA);
    }

    public void onRequestPermissionsResult(int requestCode, String permission[], int
grantResult[]){

        switch (requestCode){
            case REQUEST_CAMERA :
                if (grantResult.length>0){

```

```

        boolean cameraAccepted = grantResults[0] ==
PackageManager.PERMISSION_GRANTED;
        if (cameraAccepted){
            Toast.makeText(Barcode_Scanner.this,"Permission
granted",Toast.LENGTH_SHORT).show();

        }
        else {
            Toast.makeText(Barcode_Scanner.this,"Permission
Denied",Toast.LENGTH_SHORT).show();
            if (Build.VERSION.SDK_INT >= Build.VERSION_CODES.M){
                if (shouldShowRequestPermissionRationale(CAMERA)){
                    displayAlertMessage("Allow Access to Camera ", new
DialogInterface.OnClickListener() {
                        @Override
                        public void onClick(DialogInterface dialog, int which) {
                            requestPermissions(new String[]{CAMERA},
REQUEST_CAMERA);
                        }
                    });
                    return;
                }
            }
        }
        break;
    }
}

@Override
public void onResume(){
    super.onResume();
    if (Build.VERSION.SDK_INT >= Build.VERSION_CODES.M){
        if (checkPermission()){
            if (scannerView == null){
                scannerView = new ZXingScannerView(this);
                setContentView(scannerView);
            }
            scannerView.setResultHandler(this);
            scannerView.startCamera();
        }
        else {
            requestPermissions();
        }
    }
}

@Override
public void onDestroy(){

```

```

        super.onDestroy();
        scannerView.stopCamera();
    }

    public void displayAlertMessage(String message, DialogInterface.OnClickListener
listener){
        new AlertDialog.Builder(BarCode_Scanner.this)
            .setMessage(message)
            .setPositiveButton("OK",listener)
            .setNegativeButton("Cancel",null)
            .create()
            .show();
    }

    @Override
    public void handleResult(Result result) {
        final String scanResult = result.getText();
        AlertDialog.Builder builder = new MaterialAlertDialogBuilder(this,
R.style.AlertDialogTheme);
        builder.setTitle("Gate Pass Number");
        builder.setIcon(R.drawable.person_icon);
        builder.setPositiveButton("Continue", new DialogInterface.OnClickListener() {
            @Override
            public void onClick(DialogInterface dialog, int which) {
                // scannerView.resumeCameraPreview(MainActivity.this);
                Intent gpr_data = new Intent(BarCode_Scanner.this,Security_Control.class);
                gpr_data.putExtra("GPR_Value",scanResult);
                gpr_data.putExtra("ResponseCategory",gate_pass_response_category);
                startActivity(gpr_data);
                scannerView.stopCamera();
            }
        });
        builder.setNeutralButton("Scan Again", new DialogInterface.OnClickListener() {
            @Override
            public void onClick(DialogInterface dialog, int which) {
                scannerView.resumeCameraPreview(BarCode_Scanner.this);
                // Intent intent = new Intent(Intent.ACTION_VIEW, Uri.parse(scanResult));
                //startActivity(intent);
            }
        });
        builder.setMessage(scanResult);
        AlertDialog alert = builder.create();
        alert.show();
    }
}

```

Appendix 2: The Sample Codes for Application Programming Interface

(i) API Links

```
<?php

require_once './DbOperation.php';

//an array to display response
$response = array();

date_default_timezone_set('Africa/Nairobi');
$Current_Time = date('Y/m/d h:i:s a', time());

//if it is an api call
//that means a get parameter named api call is set in the URL
//and with this parameter we are concluding that it is an api call
if(isset($_GET['apicall'])) {

    switch ($_GET['apicall']) {

    case 'authentication':
        $Username = $_POST['Username'];
        $Password = $_POST['Password'];
        $db = new DbOperation();
        $response['User_Login_Info'] = $db->authentication($Username, $Password);
        $response['error'] = false;
        $response['message'] = 'Success';
        break;

    case 'get_employee_detail':
        $Card_number = $_GET['New_Card_No'];
        $db = new DbOperation();
        $response['response'] = $db->get_employee_detail($Card_number);
        $response['error'] = false;
        $response['message'] = 'User Exist';
        break;

    case 'creategatepass_employee_hdr':
        //creating a new dboperation object
        $db = new DbOperation();
        // $Gate_Pass_Ref_No = $_POST['Gate_Pass_Ref_No'];
        // $Card_number = $_POST['Card_number'];
        // $Employee_name = $_POST['Employee_name'];
        $Gate_name = $_POST['Gate_name'];
        $Gate_pass_type = $_POST['Gate_pass_type'];
        $No_of_Vehicle = $_POST['No_of_Vehicle'];
        $Gate_pass_date = $_POST['Gate_pass_date'];
        $Purpose = $_POST['Purpose'];
```



```

// $Company_Name= $_POST['Company_Name'];
// $Department_Name= $_POST['Department_Name'];
// $Designation_Name= $_POST['Designation_Name'];
$Remarks= $_POST['Remarks'];

```

```

//Converting Date String to Date Object for SQL Server
$Expected_time_OUT= $_POST['Expected_time_OUT'];
$Converted_time = strtotime($Expected_time_OUT);
$Valid_From =date('Y-m-d H:i:s', $Converted_time);

```

```

//Converting Date String to Date Object for SQL Server
$Expected_time_IN= $_POST['Expected_time_IN'];
$Converted_time = strtotime($Expected_time_IN);
$Valid_To =date('Y-m-d H:i:s', $Converted_time);
// $number_of_person= $_POST['number_of_person'];
$time = $Current_Time;

```

```

$response['Gate_Pass_Ref_No'] = $db->create_gate_pass_employee_hdr_data(
    $Gate_pass_date,
    $Gate_pass_type,
    1,
    "",
    "",
    "",
    "",
    "",
    "",
    $Purpose,
    "",
    $No_of_Vehicle,
    $Valid_From,
    $Valid_To,
    "FRESH",
    $Remarks,
    CF,
    "",
    $time,
    "",
    "",
    "",
    "",
    ""
    "Employee Gate Pass"
);

```

```

//if the record is created adding success to response
if ($response != false) {
    //record is created means there is no error
    $response['error'] = false;

```

```

//in message we have a success message

```

```

        $response['message'] = 'Detail Saved Successfully';

    } else {
        //if record is not added that means there is an error
        $response['error'] = true;

        //and we have the error message
        $response['message'] = 'Failed to save data';
    }

break;

case 'create_employee_gatepass':
    //creating a new dboperation object
    $db = new DbOperation();
    $Gate_Pass_Ref_No = $_POST['Gate_Pass_Ref_No'];
    $Card_number= $_POST['Card_Number'];
    $Employee_name = $_POST['Employee_Name'];
    $Material_OUT_ward = $_POST['Material_OUT_ward'];
    $Material_IN_ward = $_POST['Material_IN_ward'];
    $Family_status = $_POST['Family_status'];
    $Vehicle_number = $_POST['Vehicle_number'];
    $Employee_shift = $_POST['Employee_shift'];
    $Number_of_person = $_POST['Number_of_person'];
    $Vehicle_number = $_POST['Vehicle_number'];
    $Employee_count = $_POST['Employee_count'];
    $Mode_of_transport= $_POST['Mode_of_transport'];
    $time = $Current_Time;

    $result = $db->create_gate_pass_employee_detail(
        $Gate_Pass_Ref_No,
        $Employee_count,
        $Card_number,
        $Employee_name ,
        $Family_status,
        $Number_of_person,
        $Mode_of_transport,
        $Vehicle_number,
        $Material_IN_ward,
        $Material_OUT_ward,
        "",
        "",
        "",
        "",
        "FRESH",
        "Request",
        "",
        "",
        $time,

```

```

        """,
        """,
        """,
        """,
        """,
        $Employee_shift
    );

    //if the record is created adding success to response
    if ($result == false) {
        //record is created means there is no error
        $response['error'] = false;

        //in message we have a success message
        $response['message'] = 'Gate Pass Created Successfully';

    } else {
        //if record is not added that means there is an error
        $response['error'] = true;

        //and we have the error message
        $response['message'] = 'Failed to save data';
    }
break;

case 'get_all_gatepasses':
    $db = new DbOperation();
    $response['error'] = false;
    $response['message'] = 'Request successfully completed';
    $response['gate_passes'] = $db->get_all_gate_pass();
    break;

case 'approve-or-reject_reqquets':
    $Gate_Pass_Ref_No = $_POST['Gate_Pass_Ref_No'];
    $choice = $_POST['choice'];
    $db = new DbOperation();
    $response['error'] = false;
    $response['message'] = 'Action successfully completed applied';
    $response['action'] = $db-
>approve_or_reject_gate_pass_request($Gate_Pass_Ref_No,$choice);
    break;

case 'security_gate_permission':
    $Gate_Pass_Ref_No = $_GET['Gate_Pass_Ref_No'];
    $db = new DbOperation();
    $response['response'] = $db->security_gate_response($Gate_Pass_Ref_No);
    $response['error'] = false;
    $response['message'] = 'Status';
    break;

case 'security_employee_in_or_out':

```

```

$Gate_Pass_Status = $_POST['choice'];
$Gate_Pass_Ref_No = $_POST['Gate_Pass_Ref_No'];
$Material = $_POST['materials'];
$Security_Name = $_POST['security_name'];
$OUT_Time = $Current_Time;
$db = new DbOperation();

if($Gate_Pass_Status == "OUT"){
    $response['response'] = $db-
>security_employee_OUT_permission($Gate_Pass_Ref_No,"OUT",$Material,$Security_
Name,$OUT_Time);
    $response['error'] = false;
    $response['message'] = 'Status';
}
elseif ($Gate_Pass_Status == "IN"){
    $response['response'] = $db-
>security_employee_IN_permission($Gate_Pass_Ref_No,"Expired",$Material,$Security_
Name,$OUT_Time);
    $response['error'] = false;
    $response['message'] = 'Status';
}
else{
    $response['message'] = 'Gate pass is expired';
}

break;
}
}
else{
    //if it is not api call
    //pushing appropriate values to response array
    $response['error'] = true;
    $response['message'] = 'Invalid API Call';
}

//displaying the response in json structure
print_r(json_encode($response));
echo "<br>";
// echo $response["message"]
?>

```

POSTER PRESENTATION



Nelson Mandela African Institution of Science and Technology, Arusha - Tanzania



Mobile Application for Gate Pass Management System Enhancement
 Dr. Dina Machuve, Dr. Silas Mirau & Rambo Hilary

Background

The gate pass management systems have been practiced in multiple ways to manage exit and entry to the industry or institute. Over the last few years, many companies including A to Z Textile Mills Ltd has shifted from paper-based gate pass logbooks to computerized systems the web based gate pass system in particular (Gunawan & Kadir, 2017). Despite being so cumbersome to handle such as requires computers, printers and barcode scanners. No study on the use of smartphones in place of computers has ever done at The A to Z Textile Mills, Tanzania.

Research Objectives

Therefore, this project aimed to develop a gate pass mobile application that is integrated into a gate pass management system for staff and visitors (customers).

Methods

The evolutionary prototype model system development life cycle (SDLC) model was used for gate pass management mobile application development.

Results

The mobile application was developed to work in android based smartphones. The android platform allows developers to develop a mobile application through Java, Kotlin, or Dart programming language.



Figure 1: Staff gate pass request view



Figure 2: Unapproved gate pass



Figure 3: Security permission

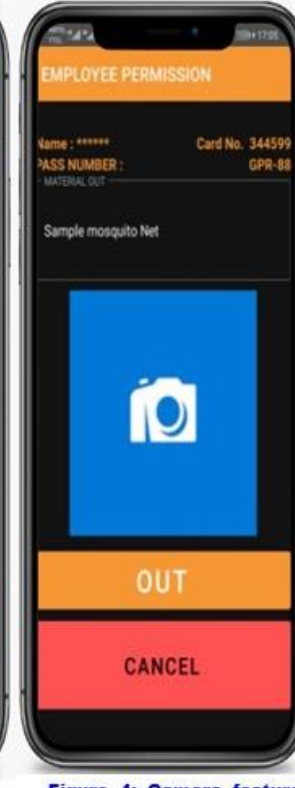


Figure 4: Camera feature to take picture

Conclusion

The application was installed on the smartphone and it was in use for 10 days. Users and stakeholders were satisfied with the accuracy of the developed mobile.

Gunawan, H., & Abdul, K. E. (2017). Integration protocol student academic information to campus RFID gate pass system. *International Conference on Electrical Engineering, Computer Science and Informatics*. <https://doi.org/10.1109/EECSI.2017.8239197>.