University of Zululand

**OVERALL DOCUMMENTATION**

**OF**

**UIRAMS**

**BY**

|  |  |
| --- | --- |
| **MKHIZE MANQOBA** | **PROJECT DEVELOPER** |
| **MRS IN EZEJI** | **Project Supervisor** |

Table of Contents

[*CHAPTER 1* 4](#_Toc86306460)

[**1.1Problem statement** 5](#_Toc86306467)

[**CHAPTER TWO** 8](#_Toc86306471)

[**2.0REQUIREMENT SPECIFICATION** 8](#_Toc86306472)

[User Characteristics 9](#_Toc86306473)

[General Constrains 9](#_Toc86306474)

[Functional Requirements Specification 9](#_Toc86306475)

[2.7.1Sign Up Use Case 9](#_Toc86306476)

[2.7.2Sign-in Use 10](#_Toc86306477)

[2.7.3 Report Use Case 11](#_Toc86306478)

[2.7.4 Warden Use Case 12](#_Toc86306479)

[2.8Non- Functional Requirement 12](#_Toc86306480)

[2.9Functional Requirements 13](#_Toc86306481)

[2.9.1Sign up 13](#_Toc86306482)

[2.9.2Sign in 13](#_Toc86306483)

[2.9.3Report 14](#_Toc86306484)

[2.9.4Call 15](#_Toc86306485)

[2.10System Environment 16](#_Toc86306486)

[16](#_Toc86306487)

[16](#_Toc86306488)

[16](#_Toc86306489)

[16](#_Toc86306490)

[16](#_Toc86306491)

[Figure 1 System Environment 16](#_Toc86306492)

[2.11.0Detailed Non-Functional Requirements 16](#_Toc86306493)

[Figure 2 logical structure 17](#_Toc86306494)

[2 17](#_Toc86306495)

[2.13Report Data Entity 17](#_Toc86306496)

[Send Response Data Entity 18](#_Toc86306497)

[Call Data entity 18](#_Toc86306498)

[CHAPTER 3 19](#_Toc86306499)

[***3.0SYSTEM*** DESIGN 19](#_Toc86306500)

[3.1Overview 19](#_Toc86306501)

[3.2Purpose 19](#_Toc86306502)

[3.3Scope 19](#_Toc86306503)

[3.4Design Diagrams 20](#_Toc86306504)

[3.4.1Deployment Diagram 20](#_Toc86306505)

[Figure 3: Deployment 20](#_Toc86306506)

[3.5Architecture design 21](#_Toc86306507)

[Figure 4 : System architecture 21](#_Toc86306508)

[Figure 6 Data fields 24](#_Toc86306509)

[3.7.0 Use Case Realizations 24](#_Toc86306510)

[Figure 7 System Sequence Diagram 25](#_Toc86306511)

[3.7.1Use case Report 26](#_Toc86306512)

[Figure 8 Report Sequence Diagram 27](#_Toc86306513)

[3.7.2. Use Case: Send Feedback 27](#_Toc86306514)

[Figure 9 Send feedback Sequence Diagram 28](#_Toc86306515)

[3.8. Interface design 28](#_Toc86306516)

[29](#_Toc86306517)

[***Figure 10 Sign in page*** 29](#_Toc86306518)

[29](#_Toc86306519)

[***Figure 11 Sign up page*** 29](#_Toc86306520)

[figure 12 forgot password page 30](#_Toc86306521)

[Figure 13 home page 31](#_Toc86306523)

[32](#_Toc86306524)

[***Figure 14 Report page*** 32](#_Toc86306525)

[***Figure 15 Submitted page*** 32](#_Toc86306526)

[***Figure 16 Reports made*** 32](#_Toc86306527)

[***Figure 17 Call page*** 33](#_Toc86306529)

[CHAPTER 4 34](#_Toc86306530)

[**SYSTEM TESTING** 34](#_Toc86306531)

[4.1OBJECTIVE 34](#_Toc86306532)

[4.2Test Approach 34](#_Toc86306533)

[4.3Functional Testing 34](#_Toc86306534)

[4.4Integration Testing 35](#_Toc86306535)

[4.5Interface testing 35](#_Toc86306536)

[4.6Security testing 35](#_Toc86306537)

[4.7Performance testing 35](#_Toc86306538)

[4.8Response time: 35](#_Toc86306539)

[4.10Portability 36](#_Toc86306540)

[4.11Acceptance testing 36](#_Toc86306541)

[4.12Beta Testing 36](#_Toc86306542)

[4.13Pass/Fail Criteria 36](#_Toc86306543)

[4.15Schedule 36](#_Toc86306544)

[CHAPTER 5 38](#_Toc86306545)

[5.0. CONCLUSION 38](#_Toc86306546)

[***References*** 39](#_Toc86306547)

Glossary

|  |  |  |
| --- | --- | --- |
| **Term** | **Acronym** | **Definition** |
| Unizulu Infrastructure Reporting And Maintenance System | UIRAMS | It is the name of the Application |
| University of Zululand | Unizulu | The name of the University that is in kwa-Dlangezwa |
| on-campus | oncamp | The area that contains the main buildings of a university or collage |
| Physical Planning and Operation Department | PPO | The University company that specialises on fixing and maintaining infrastructure |
| SoupUI | null | Tool used for testing android application |
| Java | null | is a programming language and computing platform |
| Jmeter | null | Design to load test functional behaviour and measure performance. |
|  |  |  |
| Integrated Development Environment | IDE | Software that built application and combines developers tool into single GUI |
| Application | app | The type of software that can be installed an run on a computer, tablet or smartphones |

CHAPTER 1

1.0Introduction

**1.1.0Overview**

Infrastructure are recognized to have a significant impact on the environment and the community. (Renard Y, et.al, 2013). According to (Lee-kuo Lin et.al, 2005). Maintenance management is good or not will make a big influence during the school building using age for supplying a safety environment of student learning and facility teaching effectiveness.

According to (Maurino et.al, 1995) Maintaining at an early stage provide two important factors

1. To protect people and environment from injury and damage.
2. To enable the potential victims to escape out of hazards.

Infrastructures such as buildings, classrooms, laboratories and equipment education are crucial elements of learning environments in schools and universities. There is a strong evidence that high-quality infrastructure facilitate better instruction, improves student outcomes, and reduces dropout rate. (Janssen Teixeira et.al, 2017). The proposed project aims to fill the gap of scare research done on reporting and maintaining system in universities. Thus the project attempt to fill the vacuum, by creating a functional reporting system for unizulu students and staff members that will report any issue about infrastructure damage or faulty such as office light, broke desk, chair or leaking pipes etc.

**1.1Problem statement**

Unmaintained infrastructure has been a serious problem in Universities amongst those varsities Unizulu came to the top of the list of unmaintained infrastructure. It has come to my close attention that the varsities needs a Reporting system. Students and staff members fail to report every broken infrastructure to the warden or PPO due to there is a vast number of students and staff member that leads to more problems being addressed and quarter of those being attended through the manual system of which it increases the risk of injuries and unsafe environment. We need to have a clear way of reporting infrastructure damage that is fast and reliable to the University. The UIRAMS will help students and staff members to make it easy to report remotely rather than walking a long distance to report the issue thus it will save time and lives since we are leaving in an era of covid19

**1.3OBJECTIVES**

The main goal of this project is to design a functional and effective Unizulu infrastructure reporting and maintenance system that will allow staff members and students on campus to communicate with warden and PPMW for reporting any infrastructure that needs to be maintained such as faulty electrical failure, office light, street light, broken desk and many other related infrastructure issues that are on Unizulu campus.

This project will focus mainly on this objective:

* To develop a user friendly system.
* Build a database that will store Unizulu students and staff member data only.
* To provide a reliable service to students and staff members.
* To make things easier for staff and students rather than wasting study time/working time.
* To develop a functional system.

**1.4METHODOLOYGY**

The iteration between the students, staff members and warden or PPMW they will share data through the reports from students and staff members to perform their required duties. Tools required to develop the system is Android studio IDE and using Java programming language.

**1.5.0DESIGN**

**1.5.1 FRIENDLY SYSTEM**

Iwill design a user interface that is user friendly to use.

**1.5.2**. **FUNCTIONAL SYSTEM**

To create a functional application, I will need to follow object- oriented principles design, break each task to single unit test them and combine the components.

**1.6. DESIGN A DATA BASE**

The data base of the system will only allow Unizulu students and staff members only. However, to design an efficient database I will use entity- relationship schema (ER diagram)

**1.7SIGNIFICANCE**

* This project will help the University to save money. When it comes to renewal of an infrastructure it better to do maintenance earlier to prolong the lifespan and to avoid expensive renewals of infrastructure.
* This project will help to students and staff members to be safe on hazardous infrastructure.
* The project will be easy and an effective way to report issues to warden and PPMW through remotely.

**1.8LIMITATION OF PROJECT**

* Only Unizulu students and staff members will use the system
* In other to use the system you must have internet data.
* Only on-campus maintenance issues will be considered.

**1.9Evaluation**

Evaluating the system after finishing the testing to check if it meets the objectives.

* Is the system user friendly to the users?
* Does the database reliable and consistent?
* Is the system smoothly functional?
* Does the system provide reliable services to students and staff members?
* The proposed system does it add value to staff and students lives.

**The University of Zululand unmaintained infrastructure**

****

Broken Window

.

****

Broken light

Leaking pipe

****

**CHAPTER TWO**

**2.0REQUIREMENT SPECIFICATION**

**2.1The Purpose**

The purpose of this section is to collect and analyse all assorted ideas that have come up to define the system, its requirements with respect to the consumer (supervisor). Also, we shall predict and sort out how we hope this product will be used in order to gain a better understanding of the project**.** The software requirement specification document is to present a detailed description of the Unizulu Infrastructure Reporting and Maintenance System (UIRAMS). The intended audience of this document includes University of Zululand staff members, students and warden. The document will be proposed to the University of Zululand for its approval.

* 1. **Scope of project**

This application system will be a Unizulu infrastructure reporting and maintenance system for students and staff members of the University of Zululand. The software will be an application that allows students and staff members to report any faulty property that is on-campus. The system will be designed to reduce the number of broken infrastructure by bridging the gap of communication between staff, students and warden that will enable the PPO to fix broken infrastructure immediately after the report being sent by student or a staff member without any further harm.

The software will be solely based communication between students, staff members, and warden via online. Reporting with smartphones in the application. More specifically the system is designed to allow students and staff to report any faulty infrastructure to warden thus the warden will call the PPO help desk to inform them about a damage infrastructure on-campus. This system will also have a database containing a list of University of Zululand staff members and students that have registered to use this application.

* 1. **Production Perspective**

The UIRAMS is an android –based application. The system interface has two other systems, the warden system and student, staff system. The system provides a secure environment for all report and for storing and retrieving of confidential member information.

* 1. **Product functions**

Staff and students can use the UIRAMS to report defective infrastructure. To submit a report through UIRAMS, one must first register as a student or employee of the University of Zululand using the sign-up interface. Once enrolled, the user need to sign-in to access the main page. Staff and students should complete the form, which contains all of the information needed to fill up and finish the problem statement on what needs to be maintained. The UIRAMS has a feature that allows you to report a failure infrastructure. When the warden receives a report from the system, he or she calls the PPO to brief them on the report, In this application, the warden serves as an administrator

# **User Characteristics**

The two main actors of UIRAMS users are staff members, students and warden. Student and staff member are anyone registered in University of Zululand. Students and staff members can only submit the report based on where they reside. The time it will take to reach the warden is unknown because of network connectivity. The only skill needed by staff and student is the ability to be able to fill the form correctly. Warden is someone who is in charge of viewing the reports and call the PPO thus acts as an administrator of the application. The system does not need any background training since it is a user friendly system.

# General Constrains

The technology will provide all students and staff members access to an android smartphone, allowing the warden to operate. The user interface will be simple to comprehend, thus there will be no need for training. To send faulty infrastructure, students and staff members will need to be connected to the internet. The user's personal information will be safeguarded and maintained in accordance with the law.

# Functional Requirements Specification

The use cases for students and staff members are outlined in this section. Each student and staff member has their own use case, while the warden serves as a system administrator.

# 2.7.1Sign Up Use Case

Use Case Diagram: Sign Up

Student or staff member

Sign Up

**Short Description**

Student or staff member access the online UIRAMS application, reports any infrastructure that needs to be maintained.

**Initial Step-By–Step Description**

Before this use case can be initiated, student or staff member will have to have internet data to connect to UIRAMS android application.

1. The system first display a sign in page that has sign up text view.
2. The user clicks the sign up button.
3. The system displays a Sign up page.
4. The user fills in all the fields that are in the page.
5. After the user completing to fill in the fields clicks on the submit button.
6. The system displays a pop up massage stating you have successfully registered all the details are store in the database
7. The system will go back to the sign-in page.

2.7.2Sign-in UseCase

Use Case: Sign In

Student or staff member

Sign in

**Short Description**

Student or staff member access the mobile app to report any infrastructure damage.

**Initial Step-By–Step Description**

Before this use case can be initiated, student or staff member has already registered through the UIRAMS android application by signing-up in the fields that are specify.

1. The system first display a sign in page that has sign up text view.
2. The user fills in the fields that are specified using the details that he/she entered when registering to the application.
3. The user clicks on sign in button.
4. The system pass to the home page if he/she entered a valid information.

# 2.7.3 Report Use Case

Use case: **Report**

**Diagram:**

Student or staff member

Report

**Brief Description**

Student or staff member access the online UIRAMS application, reports any infrastructure that needs to be maintained.

**Initial Step-By–Step Description**

Before this use case can be initiated, student or staff member has already registered through the UIRAMS android application by signing-up in the fields that are specify.

1. The system display picture of Kwa-Dlangezwa campus with contact details an address.

2. The student or staff member choose which campus they reside in by clicking a report button.

3. The system display new page that have a report form.

4. The form has the following fields for students and staff members to fill in:

* Building name
* Flour number
* Report statement

1. After the user completed to fill the form, click the submit button to send the report to warden.
2. The system displays a completed form that have date and time of the report.
3. The user clicks the back button that will direct him/her back to the report page.
4. If the user want to see all report that he/she has made, clicks on the reports made button to view them.
5. The user may make multiple report.

# 2.7.4 Warden Use Case

Use case: **call**

**Diagram**

warden

call

PPO

**Brief Description**

The warden access online UIRAMS system using an android smartphone. Warden will act as an administrator of this application

**Initial Step-By-Step Description**

Before this use case can be initiated, the warden has already connected online and sign-in using their unique password and email to the UIRAMS System.

1. The system displays a scrollable list of all reports.
2. The warden selects the first reports and read it.
3. The warden clicks on the contact button.
4. The system goes to the next page.
5. The system displays a green call picture in the center.
6. The warden clicks to the picture to call the PPO help desk.

# 2.8Non- Functional Requirement

The UIRAMS android application will be on server with high speed internet capability. The physical machine to be used are smartphones supported by android. The software developed use a tool such as android studio and firebase for connection between the smartphones and the database. The speed of student and staff member will depend on good network coverage and must have internet data rather than distinctive of the system. The warden system manager will run on the smartphones. Things that may affect the performance, or quality of the system. Namely:

* Computer Processing power
* Ram

The disk space will not affect the functionality.

# 2.9Functional Requirements

# 2.9.1Sign up

|  |  |
| --- | --- |
| **Use Case Name** | Sign Up |
| **Trigger** | The staff or student assesses the UIRAMS android application |
| **Precondition** | The mobile app start on the sign in page before the user clicks on sign up text view. |
| **Basic Path** | 1.The user clicks on the sign up text View.  2. The system displays a sign up page that have fields that needs to be filled.  3. After the user have inputted the correct details he/ she is obligated to click a sign up button.  4. The system displays a pop up message show that you are now a registered user.  5. All details of the user are now stored in the database. |
| **Alternative Paths** | There are no alternative paths all the paths stated above are compulsory since one needs to sign up in other to user this app. |
| **Post condition** | The system returns to the sign in page. |
| **Exception Paths** | The user may cancel the sign up at any time. |
| **Other** | The user information includes Email address ,student number and password |

# 2.9.2Sign in

|  |  |
| --- | --- |
| **Use Case Name** | Sign in |
| **Trigger** | The staff or student assesses the UIRAMS android application |
| **Precondition** | The user has already registered in a sign up page. |
| **Basic Path** | 1. The user input the email that was used when registering the app. 2. The user input the password he/she entered when registering that is more than eight characters. 3. The user clicks on the sign in button. 4. The system goes to the next page. |
| **Alternative Paths** | There is no alternative paths. |
| **Post condition** | The system displays the home screen. |
| **Exception Paths** | The user may cancel to sign in at any time. |

# 2.9.3Report

|  |  |
| --- | --- |
| **Use Case Name** | Report |
| **Trigger** | The student or staff member access the UIRAMS mobile application. |
| **Precondition** | The app displays the sign-in page |
| **Basic Path** | 1. The user chooses to report infrastructure damage on their campuses they reside in.  2. The user select a report button.  3. The system will display a new page that have report form.  4. The user writes on the spaces provided to complete the report.  5. The user clicks the submit button to send the report to the warden.  6. The system goes back to the main page. |
| **Alternative Path** | In steps 4, if the student or staff member did not fill all the required details specified in the form. The system goes back to step 1. |
| **Post condition** | The warden have received the report from students or staff members. |
| **Exception path** | The student and staff member may retrieve the report at any time. |

# 2.9.4Call

|  |  |
| --- | --- |
| **Use case Name** | call |
| **Trigger** | The warden select check report of the UIRAMS in the database |
| **Precondition** | The warden has accessed the UIRAMS homepage and the list is already in the database |
| **Basic Path** | 1.The system create and display list of all active report  2. The warden clicks the view button to see the first report.  3. The warden clicks the contact button.  4. The system goes to the call page.  5. The warden clicks on the green call button and calls the PPO to inform them about a faulty infrastructure. |
| **Alternative Path** | There is no alternative path |
| **Post condition** | The PPO has received the call and attend to the problem stated. |
| **Exception Path** | The warden may reject or abort the operation at any time. |

# 2.10System Environment



DATABASE

INTERNET OR NETWORK

Application server



UIRAMS application

Warden

Student or staff member

# Figure 1 System Environment

The UIRAMS System has 2 active actors student or staff and the second one is the warden 2 cooperating subsystems (camera & database). The camera is capturing everything that happening on the road, the database is storing the cars and driver’s details, and the operator is handling the cars with mismatching or missing details .

# 2.11.0Detailed Non-Functional Requirements

**2.11.1Logical** Structure of the Data

The logical structure is to be stored in the internal UIRAMS manager database. Represents how the system operate in background when it comes to sharing of data to server its purpose and fulfil all requirement, starting from the beginning (source of input) all to the final output

Database

warden

Staff member

student

writes

sent to

PPO

**call**

# Figure 2 logical structure

The data entities describing the data is as follows.

2**.12Student and Staff** member Data entity

|  |  |  |  |
| --- | --- | --- | --- |
| **Data Item** | **Type** | **Description** | **Comment** |
| Name | Text | Name of the student or staff |  |
| Email address | Text | Internet address |  |
| Password | Text | Password of the student or staff |  |
| Username | Integer | Username of The student or staff |  |

# 2.13Report Data Entity

|  |  |  |  |
| --- | --- | --- | --- |
| **Data Item** | **Type** | **Description** | **comment** |
| Name | Text | Name of The reported person |  |
| Building Name | Text | Building name |  |
| Flour | Integer | Flour of the Building |  |
| Problem statement | Text | Short description of the infrastructure that needs to be maintained. |  |

# Send Response Data Entity

|  |  |  |  |
| --- | --- | --- | --- |
| **Data Item** | **Type** | **Description** | **comment** |
| Report | Pointer | Student and staff member entity | May be several |
| message sent | Text | Message sent to student and staff member |  |
| Send Feedback | Text | Pass information |  |

# Call Data entity

|  |  |  |  |
| --- | --- | --- | --- |
| **Data Item** | **Type** | **Description** | **Comment** |
| **call** | **Pointer** | **Warden call the PPO** |  |

# CHAPTER 3

# ***3.0SYSTEM*** DESIGN

# 3.1Overview

The Software Design chapter provides a documentation which will be used to aid in software development by providing the details for how the software should be built. This section consists of narrative and graphical documentation of the software design for the project including use case models, sequence diagrams, collaboration models, object behaviour models, and other supporting requirement information as well as how all this collaborate to form a system.

# 3.2Purpose

This Software Design Document provides the design details of University of Zululand infrastructure reporting and maintaining application system (UIRAMS). The intended audience of this document are University of Zululand staff members and students including the warden.

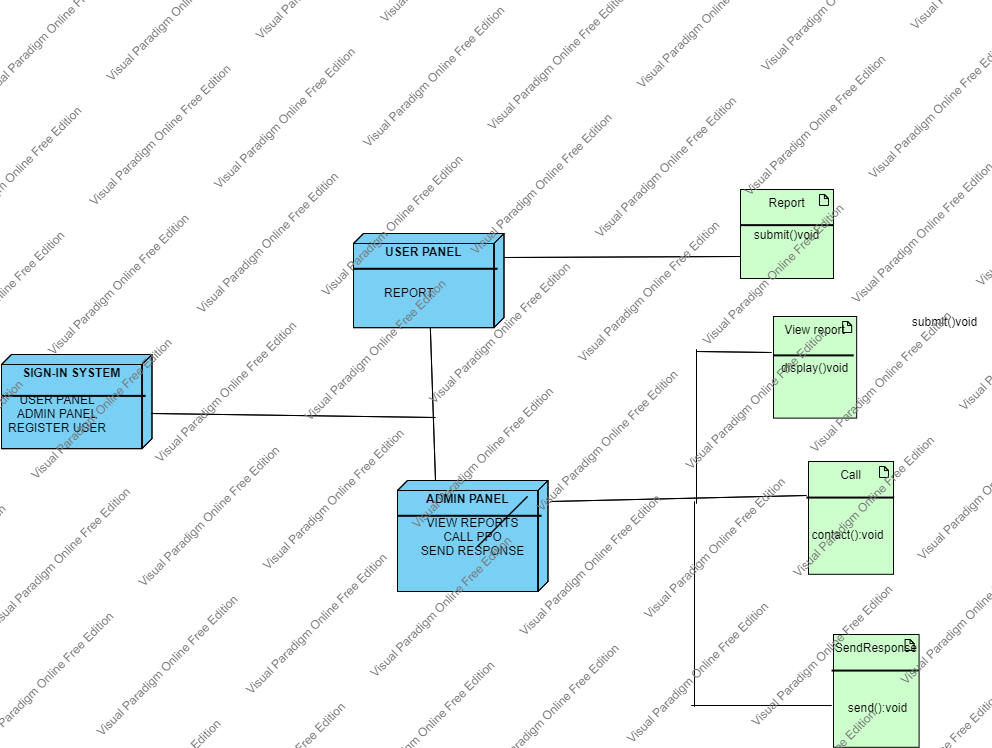
# 3.3Scope

This document contains a perfect illustration of the design of UIRAMS. The basic architecture is an android server from a client paradigm. The basic pages will be in android smartphones. The students, staff member and warden will access the UIRAMS thus the warden acting as an administrator doing all the necessary update and passing the reports to the PPO. This does not limit the client to send multiple reports and there are no restrictions on how many report he or she can submit.

# 3.4Design Diagrams

# 

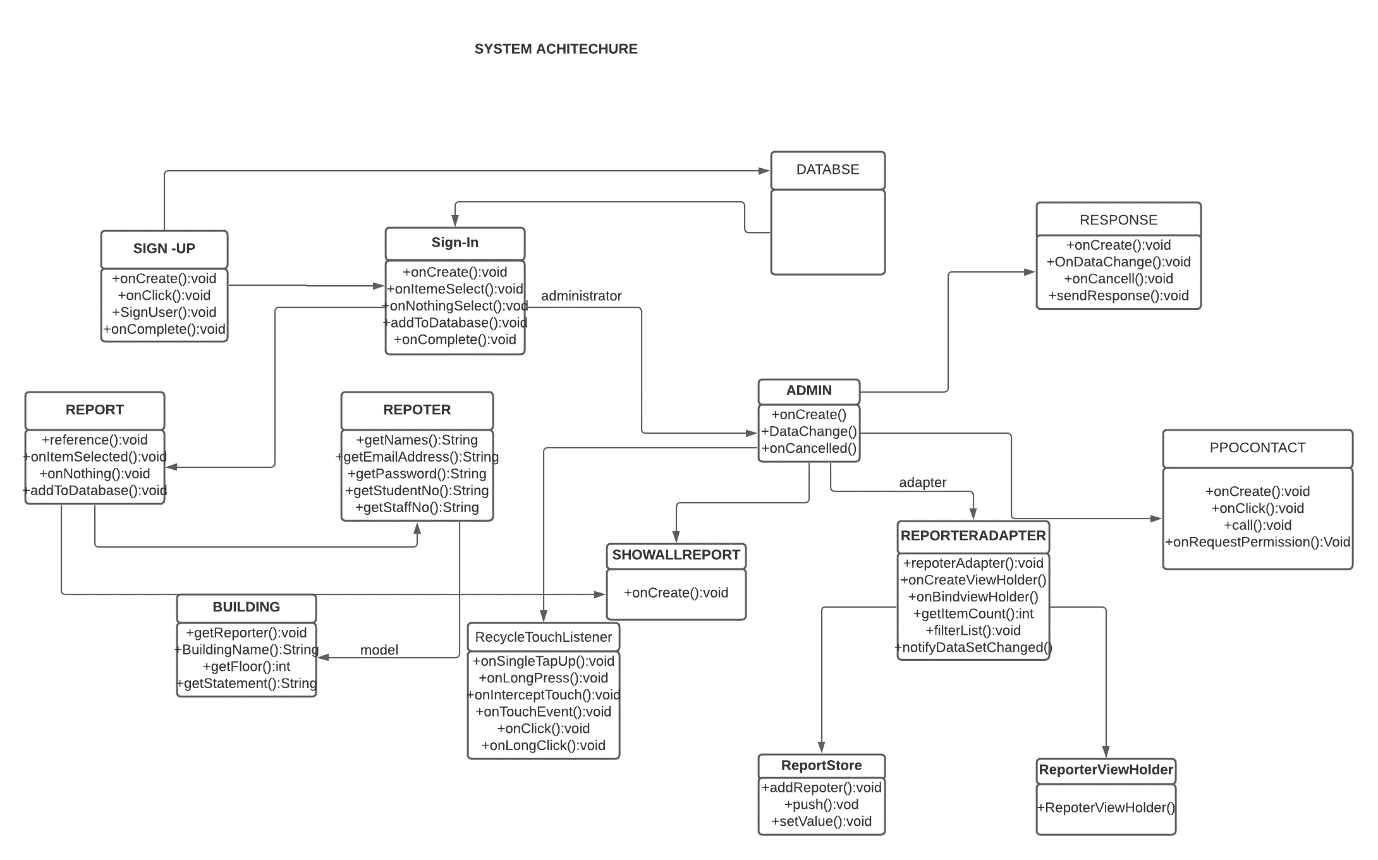
# 3.4.1Deployment Diagram



# Figure 3: Deployment

# 3.5Architecture design

**3.5.1** **Application** System Architecture

****

# Figure 4 : System architecture

**Sign In Screen**

**Name:** Sign In

Type: Android Page

Description: It is used by three actors namely students, staff members and the warden, they are required to fill their personal information in other to pass to the next page, it is the first page that appears onscreen.

Attributes: None

Resources: None

Operations:

Name: Sign-in()

Argument: None

Pre-Condition: There user must sign-up to the system.

Post Condition: Home Page

Exception: None

Flow of Events:

1. The Unizulu student and staff members access the application.

2. The Sign-in form is being displayed on screen

3. The alternative path, if the user is not registered he or she must click sign-up button to register and fill in their personal details.

4. The user clicks the sign-in button.

5. The user is connected to Home page.

UIRAMS Home Page

Name: UIRAMS Home Page

Type: android Page

Description: This initial home page presented to students and staff members upon connection with firebase. There are two buttons that does the same function report button that enables the user to go to the next page.

Attributes: None

Resources: None

Operations:

Name: Select()

Argument: None

Pre-Condition: Connected to internet

Post-Condition: On another page

Exception: None

Flow of Event:

1. Student and staff are presented with the report home page.
2. Student or staff clicks a button
3. Student or staff members is connected to another page.

Report Faulty infrastructure

Name: Report faulty infrastructure

Type: Android Page

Description: When the students clicks the report button he or she will be presented with a form. The form. The form will a blank field to be completed when the form is completed the submit the form.

Attributes: Text

Resources: None

Operations:

Name: Submit()

Arguments: None

Pre-Condition: Connected to internet.

Post-Condition: Form saved and the warden receives the report.

Flow of Event:

1. Student is presented with the form.

2. Student fills the form.

3. Student clicks submit

4. Firebase server save data in a text file.

5. Firebase sends the report to the warden.

Report Database

Name: Report Database

Descriptions: The UIRAMS database will reside on the firebase server.

Attributes:

Last Name: String

Student No: String

Password: String

Building Name: String

Floor Number: int

Operations:

Name: Database()

Arguments: None

Pre-Condition: None

Post-Condition: A new record is added to the UIRAMS database Table.

Floor of Event:

1. The student is on the add an entry form page.

2. The student fills in the fields.

3. The student clicks submit.

4. The firebase server adds the new information to the database.

Send Feedback

Name: Send Feedback

Type: Android Page

Description: This page has two buttons one to view report and the second one to send feedback .On the view button the warden of which in this case acts as administrator views the reports all recent reports will be displayed onscreen and are scrollable.

On the feedback button the warden will be presented with a report page and fields.

Attribute: None

Resources: None

Operations:

Name: Response()

Argument: None

Return: None

Pre-Condition: Connection to internet.

Post-Condition: Staff and students receives a response.

Flow of Event:

1. Warden is on the Home page.

2. Warden clicks view report.

3. The warden is presented with the reports made by students and staff.

4. The warden clicks on the report and new page is displayed with the fields.

5. The warden complete the form and clicks the send response.

**3.6Data Structure** Design

Firebase is used to store the data in the UIRAMS database. The fields for sending data to and from the database are listed in the table below.

Data field types and sizes.

|  |  |  |
| --- | --- | --- |
| **Attribute Name** | **Attribute Type** | **Attribute Size** |
| Initials and last Name | String | 30 |
| Building Name\*# | String | 50 |
| Floor Number\*# | Integer | 30 |
| Problem statement\*# | String | 50 |
| Staff Number\*# | String | 5 |
| Student Number\*# | String | 9 |
| Email Address# | String | 20 |
| Password\*# | String | 10 |
| Confirm Password\*# | Boolean | 1 |

# Figure 6 Data fields

# 3.7.0 Use Case Realizations

# Figure 7 System Sequence Diagram

# 3.7.1Use case Report

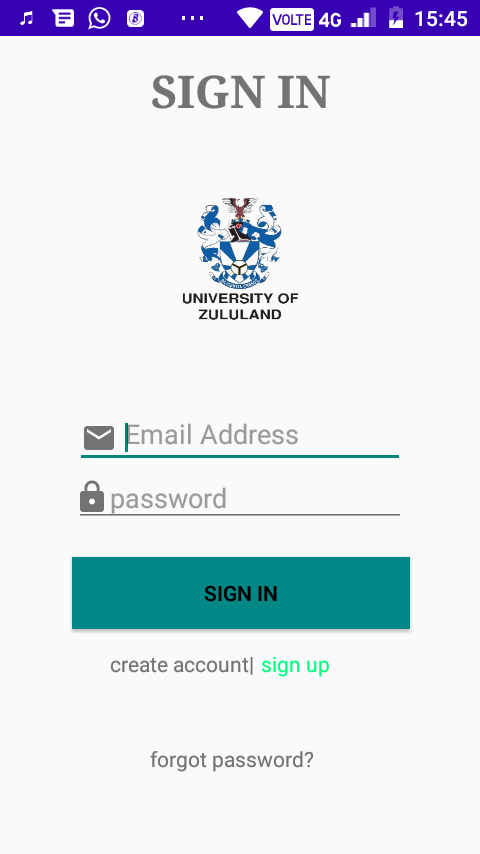
# Figure 8 Report Sequence Diagram

# 3.7.2. Use Case: Send Feedback

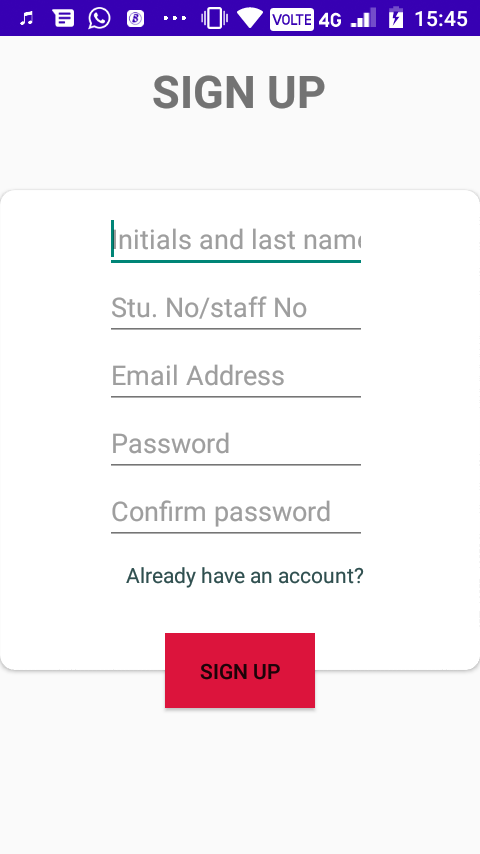
# Figure 9 Send feedback Sequence Diagram

# 3.8. Interface design

The interface will use same colours, design schemes that are presented on Unizulu infrastructure reporting and maintenance system.

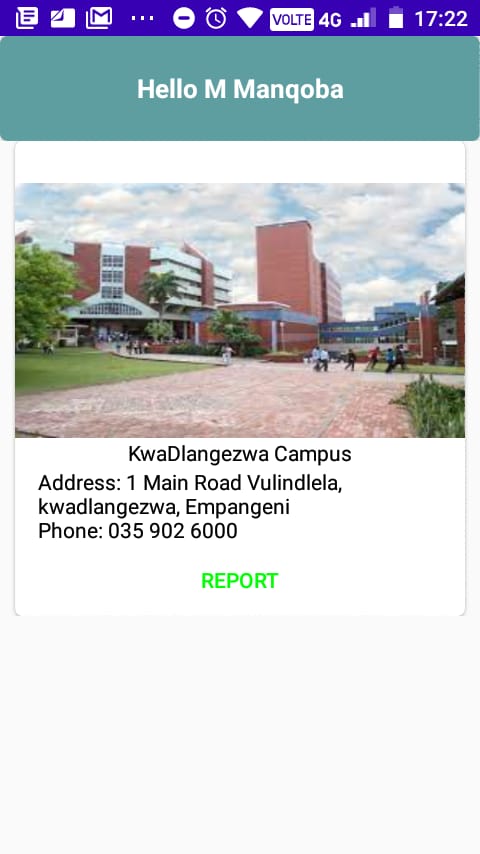


***Figure 10 Sign in page***

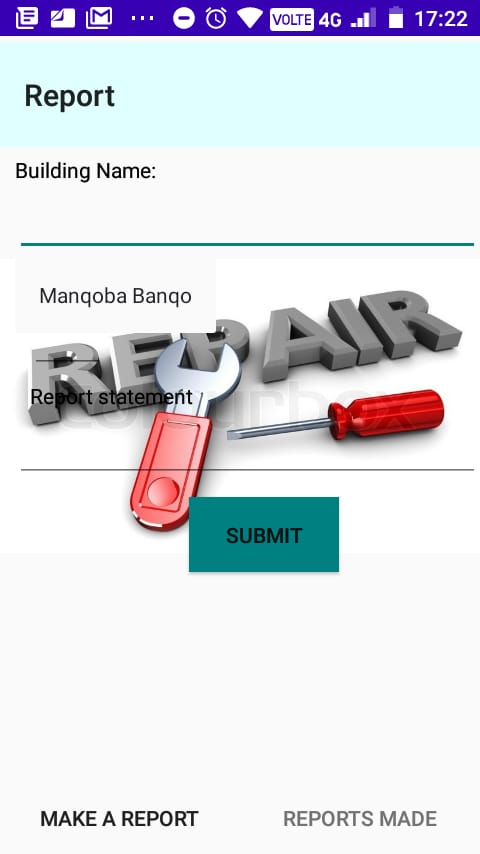


***Figure 11 Sign up page***

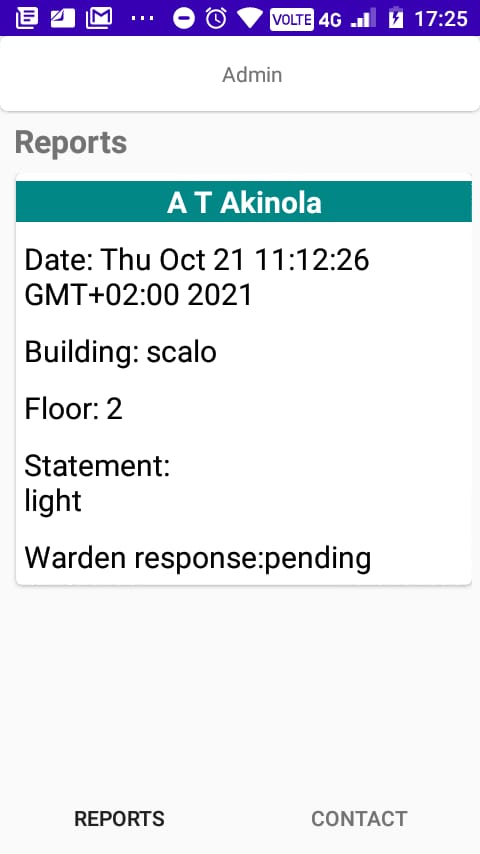
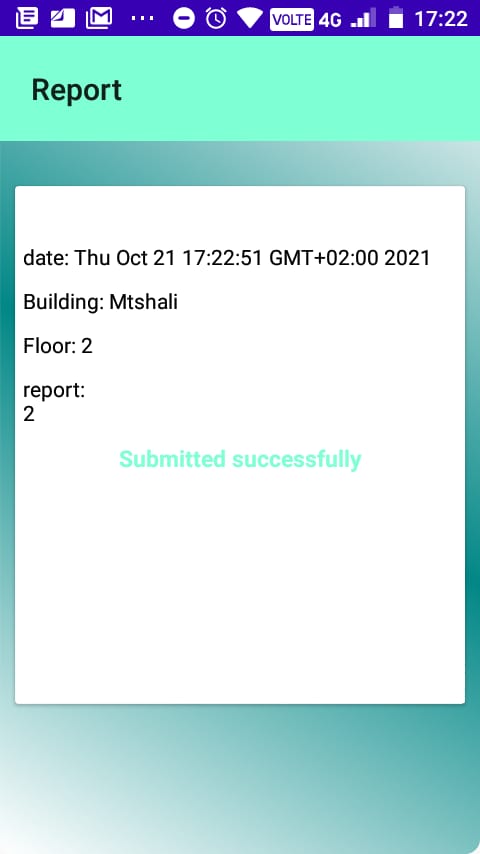
# figure 12 forgot password page

******

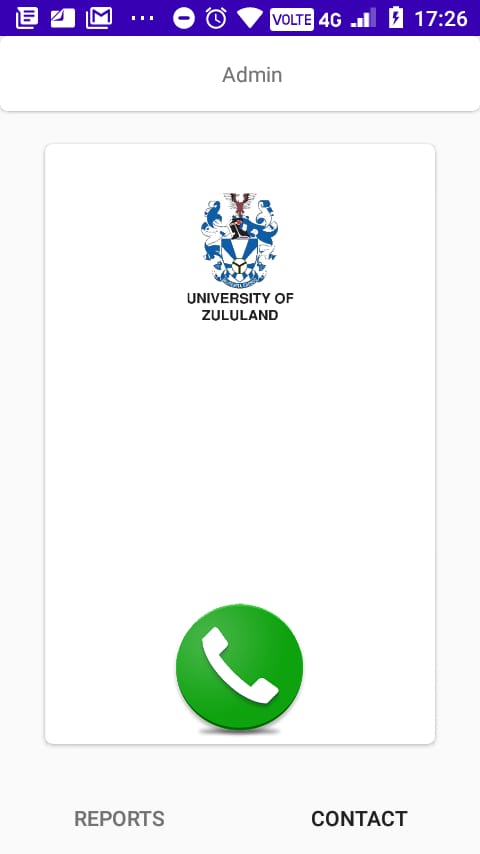
# Figure 13 home page

******

***Figure 14 Report page***

***Figure 15 Submitted page***

***Figure 16 Reports made***

******

***Figure 17 Call page***

# CHAPTER 4

# **SYSTEM TESTING**

# 4.1OBJECTIVE

The aim of system testing is to test the if the reporting system is working hence to be able to achieve the goal. Testing should be implemented at a high-level in both functional component and non- functional component. The purpose of this document is to be able to determine success and failure of this project. Thus to be able to validate the system, being able to find defect with quality analysis that will produce quality system.

# 4.2Test Approach

This system is used to report faulty infrastructure. The system will not only depends report forms because some cases may be emergency so testing should not stick to one or two components they are varies activities that will be tested and testing should be completed within time and budget in other to meet the user needs.

# 4.3Functional Testing

* White Box (Unit) Testing:

This system is designed to be modular, this is done to make it easy to maintain and modify. We are going to use the modularity advantage to test the subsystems of the whole system to see if the inputs provided gives expected results.

There are eight subsystems to be tested, namely:

* 1. Sign up
  2. Sign in
  3. Forget password
  4. Home page
  5. Report
  6. View report
  7. Send response
  8. Call

# 4.4Integration Testing

These component modules will have to combine and exchange their data as the system functions, the output of one subsystem will become the input of another subsystem, we will also test subsystems that collaborate. Sign up Sign in + Forgot password. If these system works accordingly they will produce the output of home page+ report + view reports+ call

# 4.5Interface testing

The interface are the crucial part of this project to ensure we get a functional system. Testing all interfaces that are properly displayed thus it should be user friendly to someone who have no background knowledge about this application. Five people around campus will be given the app to use thus their feedback will be highly taken into consideration. What is lacking and not operating efficiently. The tools that were used to test the interface are Jmeter, Postmat, SoupUI.

# 4.6Security testing

The database testing to ensure that only unizulu employees and students can access to this application. It was done by restricting the database. If the user signing up is not in the database the user cannot use the app. This will improve the UIRAMS system security and help vendors from outside cannot mess around with the application.

# 4.7Performance testing

This will be done by 5n to six time to ensure that the customer is certified with the application on the performance testing we performed three categories namely response time, availability, portability

# 4.8Response time:

The system is tested and operate perfectly fine there is no stack up on the system when it try to pass another activity.

**4.9Availability:**

This testing have not been made since it still under construction it is not found on app store for availability but soon as the stakeholders have approved the application.

4.10Portability:

The system have been tested positive for good portability four smartphones that are supported by android have been used to debug the application. Smartphone that is not supported by android it cannot work and the aim that the system is for android user only.

# 4.11Acceptance testing

Determined by six people on unizulu campus that the system is acceptable and the system meet the requirement of the proposed project and component operate hand in hand to produce the customer needs.

# 4.12Beta Testing

Computer Science students played a big part in achieving the goal by using a beta testing ,each student were given the application and they performed all commands of the apps their task was to determine if the system have failures and have no defect.

# 4.13Pass/Fail Criteria

Sign-in, sign-up, home, repots, call have pass the standards that was set by the requirement but the response to the user failed the criteria thus the shortest path was taken to achieve the goal and the requirement of the system.

# 4.15Schedule

Pipeline approach was achieved for testing, seven groups have been separated to do their specific task once the other one is done the other group will pass the result to the next one .The formula used is k = n-1

|  |  |  |
| --- | --- | --- |
| **Features** | Date | Comment |
| Sign-up | 03/10/2021 | pass |
| Sign-in | 05/10/2021 | pass |
| Home Page | 08/10/2021 | pass |
| Reports | 10/10/2021 | pass |
| Response | 12/10/2021 | pass |
| Display Report | 14/10/2021 | pass |
| Call | 19/10/2021 | pass |

The team that tested the system.

1. Cebo Dlomo – Group leader of team A that will be responsible for sign-up testing

2. Lungelo Ngiba - Group leader of team B that will be responsible for sign-in testing

3. Amahle Base­ ­­- Group leader of team C that will be responsible for home page testing

4. Lungelo Mkhize- Group leader of team D that will be responsible reports testing

5. Ntibane Mnguni- Group leader of team E that will be responsible for response testing

6. Sphamandla Langa- Group leader of team F that will be responsible for display report testing

7. Lizwi Zungu- Group leader of team G that will be responsible for call testing

# CHAPTER 5

# 5.0. CONCLUSION

From the beginning, right from requirement phase and documenting those requirements, planning the management of project to design, then implementing the design and closing at testing. It is with no hesitation that the idea behind this project is indeed what is needed, not by UIRAMS, Students will be able to live in a safer environment reporting have been made easy to all that will be using the system will be able to live in a safer environment reporting have been made easy to all that will be using the system.

**Findings**

Java – As everything has their limit but java does not have any limit only anything is possible since it has produce many systems there said to be impossible but they became possible.

Android Studio- is the best working tool even though it is difficult to understand but features they produce amazing. Through my workings with the IDE that I rate it as the best working environment.

**Future scope**

In future purposes this application can serve to the country and to the rest of the world and it is open for extension but closed for modification .This application does not only for unizulu but I took a small community to work with any varsity and big companies can make use of this application. Thus on my own opinion I see this application meet with the forth revolution

***References***

1. Software Requirement Specification of web publishing system. April 15 2004 by Bobbie Baker
2. [IEEE] The applicable IEEE standards are published in “IEEE Standards Collection,”
3. .
4. [Bruade] The principal source of textbook material is “Software Engineering: An Object-Oriented Perspective” by Eric J. Bruade (Wiley 2001).
5. Lee-Kou lin ‘A study 4. of campus facilities maintenance system with building information modelling application’(2005)http:/www.maintenace system.com
6. [www.whyeducationinfrastracturematters@learning.com](http://www.whyeducationinfrastracturematters@learning.com)
7. [www.youtube/onlinemaintenancesystem.com](http://www.youtube/onlinemaintenancesystem.com)
8. [www.online/maintenace/system.com](http://www.online/maintenace/system.com)
9. [www.softwaresrsdocumentexamples.com](http://www.softwaresrsdocumentexamples.com)
10. [www.utubesrsfunctionalrequirements.com](http://www.utubesrsfunctionalrequirements.com)
11. [www.utubesrsNon-functionalrequirement.com](http://www.utubesrsNon-functionalrequirement.com)
12. [www.utubesrsusecase.com](http://www.utubesrsusecase.com)
13. [www.systemAchitectureexamples.com](http://www.systemAchitectureexamples.com)