Logo, company name

Description automatically generated

**PUSL3190 Computing Individual Project**

**Project Proposal**

<Project Title Here>

Supervisor: <Supervisor Name Here>

Name: <Your Name Here According to PU ID>

Plymouth Index Number: 10819486

Degree Program: <Your Degree Program Here>

Table of Contents

[**Chapter 01 Introduction** 3](#_Toc160062523)

[1.1 Introduction 3](#_Toc160062524)

[1.2 Problem Definition 3](#_Toc160062525)

[1.3 Project Objectives 3](#_Toc160062526)

[**Chapter 02 System Analysis** 4](#_Toc160062527)

[2.1 Facts Gathering Techniques 4](#_Toc160062528)

[2.2 Existing System 4](#_Toc160062529)

[2.3 Use case diagram 4](#_Toc160062530)

[2.4 Drawbacks of the existing system 4](#_Toc160062531)

[**Chapter 03 Requirements Specification** 5](#_Toc160062532)

[3.1 Functional Requirements 5](#_Toc160062533)

[3.2 Non-Functional Requirements 5](#_Toc160062534)

[3.3 Hardware / Software Requirements 5](#_Toc160062535)

[3.4 Networking Requirements (Optional) 5](#_Toc160062536)

[**Chapter 04 Feasibility Study** 6](#_Toc160062537)

[4.1 Operational Feasibility 6](#_Toc160062538)

[4.2 Technical Feasibility 6](#_Toc160062539)

[4.3 Outline Budget 6](#_Toc160062540)

[**Chapter 06 Development Tools and Technologies** 8](#_Toc160062541)

[6.1 Development Methodology 8](#_Toc160062542)

[6.2 Programming Languages and Tools 8](#_Toc160062543)

[6.3 Third Party Components and Libraries 10](#_Toc160062544)

[6.4 Algorithms 11](#_Toc160062545)

[References 13](#_Toc160062546)

# **Chapter 01 Introduction**

## 1.1 Introduction

## 1.2 Problem Definition

## 1.3 Project Objectives

# **Chapter 02 System Analysis**

## 2.1 Facts Gathering Techniques

## 2.2 Existing System

## 2.3 Use case diagram

## 2.4 Drawbacks of the existing system

# **Chapter 03 Requirements Specification**

## 3.1 Functional Requirements

## 3.2 Non-Functional Requirements

* Security:

A screen shot of a computer screen

Description automatically generated

Have also used getters and setters in java encapsulation as a security measurement. Which helps to improve maintainability of the code structure and the security.

## 3.3 Hardware / Software Requirements

## 3.4 Networking Requirements (Optional)

# **Chapter 04 Feasibility Study**

## 4.1 Operational Feasibility

## 4.2 Technical Feasibility

This application is running on react-native. And react-native is a cross-platform. By running ‘npm start’ on the command prompt or the terminal, can select the platform that should be running this application. Can run this system on Android, IOS or Web (reactnative.dev, n.d.).

When it comes to running and connecting to the server, cannot use localhost to make requests to the server through development machine. Only using the development machine’s local IP address (ex: IPv4 - Internet Protocol version 4) we can access the server. Mobile application can also access this IP address since it is also connected to the same network as the development machine.

## 4.3 Outline Budget

Chapter 05 System Architecture

5.1 Class Diagram of Proposed System

5.2 ER Diagram

5.3 High-level Architectural Diagram

5.4 Networking Diagram (Optional)

# **Chapter 06 Development Tools and Technologies**

## 6.1 Development Methodology

## 6.2 Programming Languages and Tools

Using postman tool, have tested the HTTP requests and its responds. Under collections have created a new collection called ‘Sizerecom’. There a new POST request is made to the ‘http://localhost:8080/user’ endpoint and that request would be sent in a JSON format as shows in the figure 1 below,

A screenshot of a computer

Description automatically generatedFigure 1: Postman API is being used to test the HTTP requests, send to MySQL database.

When the request is sent, it would update the MySQL database tables as in the figure 2 below. So, then it could test the APIs in this application as uses PostMapping annotation to handle POST requests.

A screenshot of a computer

Description automatically generated

Figure 2: How the MySQL database is being updated from the test post requests.

A new GET request is made to get (retrieve) the data and view all users as shown in the figure 3 using the endpoint of ‘/allusers’. Can see in the figure 4, the ID is auto incrementing by one using the ‘@GeneratedValue’ annotation with ‘@Id’ annotation. The Id then used as the primary key.

A close-up of a computer screen

Description automatically generated

Figure 3: Endpoint of the GET request.

A screenshot of a computer screen

Description automatically generated

Figure 4: Shows all the added users.

## 6.3 Third Party Components and Libraries

1. npm install @react-navigation/native
2. npm install @react-navigation/stack
3. npx react-native link react-native-gesture-handler react-native-screens react-native-svg

* Since the ‘Button’ component in react-native does not include many options when it comes to editing or styling the component, in here it uses a component called ‘TouchableOpacity’ which can easily be styled and act as a button.

1. npm install react-native-screens

* uses to navigate between the screens.

1. npm install @react-navigation/native @react-navigation/bottom-tabs

* to create the bottom navigation bar in most of the pages

1. npm install @react-native-picker/picker

* To create the dropdown list in the ‘Material.jsx’ page

1. npm install axios

* uses this library to make http requests from the react-native app to the server.

## 6.4 Algorithms

Chapter 07 Discussion [Max of 1 Page]

Overview of the Interim Report

Summary of the Report

Challenges Faced

Future Plans / Upcoming Work

# References

reactnative.dev. (n.d.). *Introduction · React Native*. [online] Available at: https://reactnative.dev/docs/getting-started.

‌

‌