Enterprise Mobility

SE5070



*Assignment I*

**Name**: B. R. M. S. R. B. Rathnayake

**Student ID**:  MS21911958

**Application Name**: *SLIITLIFY Masters*

**Technologies**: *React Native/Firebase*

Contents

[Selection of Project Area 3](#_Toc84433625)

[Overview 3](#_Toc84433626)

[Application architecture 3](#_Toc84433627)

[Offline access technology 3](#_Toc84433628)

[Design constraints and principles followed 3](#_Toc84433629)

[Integration hardware 3](#_Toc84433630)

[UI/UX flow 3](#_Toc84433631)

[Usage of custom components 3](#_Toc84433632)

## Selection of Project Area

**MS21911958** = 21911958

= 2 + 1 + 9 + 1 + 1 + 9 + 5 + 8

= 36 = 3 + 6

= 9 modulo 3

= 0 -> EduTech.

## Overview

“SLIITLIFY Masters” is an react-native application built for students who are enrolled to SLIIT master’s degree.

### Project idea

*User - Student*

May it be MSc, MBA, MPhil, in whichever stream this app serves the generic purposes for students and lecturers, to ease following up with attendance for lecturers, whilst making it easier for students to attend lectures and mark attendance through the app itself via a QR (quick response) code scanner. The student has direct access to Moodle (courseweb), student profile, SLIIT portal.

Students can login to the system using their MS Number as the username, and NIC as password (the same login is used to view results/grades in official SLIIT result board). This login will be validated by the backend servers. Authenticated students will login, invalid credentials will be rejected with a toast message. Student can slide the dashboard to the right, and view/select the menu options.

Main objective behind the project is to capture student’s attendance and to allow the student to view their attendance progress. ***How?*** The lecturers project a QR code during the lecture session, embedded in the middle of the lecture slide. (To make things interesting, students must pay attention to the lectures because the QR can be projected on any random slide.) For any reason if the student was unable to use the app/smartphone, the student needs to inform the lecturer regarding the technical difficulty during the lecture itself. The scanned QR code will directly update the backend, which maintains the attended lectures, total lectures (3 lectures attended out 10 total lectures) along with the enrolled module(s) details.

*User – Guest*

This mobile app is not limited for just SLIIT masters students. If any outsider from the university or any student who doesn’t follow a master’s degree wants to use the app, they can simply use the Guest login. Guest login is authenticated with Google authentication with Firebase Auth. The guests can simply login to the app and browse through limited functionality shown on the left-hand side navigator pane. The reason for this login is to ensure that the app has a broad audience, and it could help outsiders to learn more about SLIIT Culture, learn about the offered master’s degrees and explore SLIIT vicinities.

### Project features

Student authentication against Firebase backend.  
Dedicated backend REST API, using cloud functions to perform CRUD operations.  
Easy user Navigation.  
User preferences (dark theme/light theme)  
Native hardware used: QR code scanner, camera, vibrate, speakers.

guest users can view faculty vicinities and classroom interiors in 360 mode with the aid of Gyro scope. a graphical view for students to view current attendance status along with modules enrolled with a given semester/year. depending on the user login ( guest /user) the left hand side navigator pane will show/ hide certain features and functionalities available for the respective user

Better yet, guests could view Faculty Classrooms at 360 degrees using their smartphone Gyroscope.

## Application architecture

* + Major decisions
  + How the source code is organized
    - Introduce modularized firebase code
  + Data Flow
    - Sequence diagram
  + Restructure the file structure

## Offline access technology

## Design constraints and principles followed

* + Hamburger menu

## Integration hardware

* + QR code scanner, sound, vibration torch

## UI/UX flow

* + Low fidelity diagrams and high-fidelity diagrams

## Usage of custom components

* + Tag completely developed by the student himself