**Software Requirement Specification**

**Student - Faculty Meeting Scheduler and College Bus Booking System**

**Prepared by:**

**Team 15**

**Sidharth Ganesh 19/CS/113**

**Jerickson George 19/CS/058**

**Akhil A 19/CS/011**

**Jobin KJ 19/CS/063**

**Guided by:**

**Asst. Prof. Kishore Sebastian**

**Date : 03/06/2022**

**Table of Contents**

**1. Introduction**

**2. Overall Description** 2.1 Functional Requirements

2.2 Tools to be used

**3. Non-functional Requirements** 3.1 Performance Requirements

3.2 Safety Requirements

3.3 Security Requirements

3.4 Software Quality Attributes

**4. References**

**Revision History**

| **Name** | **Date** | **Reason For Changes** | **Version** |
| --- | --- | --- | --- |
|  |  |  |  |
|  |  |  |  |

# **1. Introduction**

*A platform where students get to interact with their faculties avoiding the wastage of time and other traditional difficulties. The purpose here is to allocate students their time of interaction with faculties according to the time they schedule through the platform.*

*The other phase grants students to book tickets and use the buses that are provided by the institution with ease. This removes their difficulties in boarding college buses with the ticket being purchased offline.*

*The purpose of this application is to create a convenient and easy to access platform for students to interact with the faculties provided for them. The scope can be further extended to enabling students to book tickets for utilizing the bus facility provided by the college.*

*The app handles the student teacher interactions and monitors those time intervals and notifies them to the advisor. The intended audience are the college personnel who are directly associated with student mentoring and students themselves.*

# **2. Overall Description**

## **2.1 Functional Requirements**

*This application provides the following components:*

***1: ADMIN MODULE***

*The attributes of this module are AdminID, AdminName, and Password.*

* ***add user ()***

*This feature enables the admin to add students to the platform and provide them a unique ID and password.*

* ***view appointment ()***

*By default, student-teacher appointments can be viewed by the concerned faculty*

* ***check payments ()***

*The details of payment (if any available) can be only accessed by the admin by default.*

* ***check payment status ()***

*The admin can also access details of students who have completed the payment.*

***2: STUDENT MODULE***

*The attributes of this module are user ID, user name, and password. This module consists of students who have registered in the app using their credentials.*

* ***login ()***

*The student user can log in to his/her account using the credentials created during the initial registration.*

* ***register appointment ()***

*Once the student registers for an appointment he/she will be considered a participant from that point onwards.*

* ***ticket payment ()***

*Users can pay for the tickets when they complete the booking process from within the app.*

* ***view ticket ()***

*Users can view the details of the bus tickets in the app.*

* ***edit profile ()***

*Users have the option to edit their profile and make changes to account/personal details. This authority is limited as some data fields such as email and phone numbers are necessary.*

*3:* ***FACULTY MODULE***

*The attributes of this module are Username, and password.This is a privileged set of users who can perform elevated functions compared to normal users.*

* *l****ogin ()***

*Faculty can log in to he/her account using the credentials provided by the admin*

* ***set appointment() and delete appointment()***

*Enables the faculty to schedule and remove the appointment slots for the students to select*

* ***check student ()***

*The faculty can see the student details and their appointment schedules with other appointments.*

**3. Non-functional Requirements**

**3.1 Performance Requirements**

*The basic objective is to reduce redundancy of the stored data. Storing of information several times leads to wastage of storage space and increase in the total size of data stored.*

**3.2 Safety Requirements**

*The app provides recovery methods in case any data is lost. The recovery method restores a past copy of the database that was backed up to archival storage and reconstructs the database to its previous state.*

**3.3 Security Requirements**

*Every password is encrypted uniquely so that incase of a data breach password would not be the compromising factor. Only the admin has the authorization to access all the appointment and payment details. The procedures are encrypted to further improve the security experience.*

**3.4 Software Quality Attributes**

**AVAILABILITY:**

*The app should be up and running.*

**CORRECTNESS:**

*Appointment details displayed, Data stored in the database and Data collected should be correctly and accurately stored in the database and is needed to be retrieved as exactly the same.*

**MAINTAINABILITY:**

*The adminis responsible for maintaining correctness of the appointments displayed in the app.*

**USABILITY:**

*The app should be able to handle and satisfy as many numbers of students as possible.*

## 

## **4. References**

1. <https://web.cs.dal.ca>
2. <https://www.tutorialspoint.com/software_testing_dictionary/software_requirement_specification.htm>