Software Requirements Specification (SRS)

Standard: IEEE 830 Format

Project Title: AUTOMATED CAMPUS FLOW

Team Members / Roll Numbers: HAMID KHAN-23PWBCS0967 , ABDULLAH-23PWBCS0997 , M SHAHEER-UL-ISLAM-23PWBCS1007 , MOTASIM BILLAH-23PWBCS0989.

Instructor: KANWAL ANEEQ

Date: 26-OCT-202

---

1 Introduction

1.1 Purpose:

This Software Requirements Specification (SRS) document defines the functional and non-functional requirements for the project **Automated Campus Flow**.
It serves as a guide for developers, testers, faculty clients, and university administrators involved in the design, development, and deployment of the system. The document will ensure all stakeholders share a clear understanding of the system's objectives and expected functionalities.

1.2 Scope:

Automated Campus Flow is a centralized mobile application designed to manage class schedules, room allocations, and real-time availability of university facilities.
It aims to:

- Eliminate manual scheduling errors.
- Provide real-time visibility of room/lab occupancy.
- Enable teachers to book rooms efficiently.
- Allow students to easily view personalized timetables.
- Improve overall utilization of campus resources.

The system will primarily target **university students, faculty members, and administrators**, accessible via **mobile devices** using **FlutterFlow** and **Firebase** technologies.

1.3 Definitions, Acronyms, and Abbreviations:

Term	Definition
UI	User Interface
DB	Database
DBMS	Database Management System
CRUD	Create, Read, Update, Delete
SRS	Software Requirements Specification
UCF	Automated Campus Flow (System Name)

1.4 References:

IEEE Std 830-1998 – Recommended Practice for Software Requirements Specifications.

FlutterFlow documentation: https://flutterflow.io

Firebase documentation: https://firebase.google.com

Course handouts and project brief from CS AND IT , UET Peshawar.

1.5 Overview:

This SRS contains:

- **Section 2:** Overall system description and environment.
- **Section 3:** Specific functional and non-functional requirements.
- **Section 4:** System models and diagrams.
- **Section 5:** Other requirements such as backup and security.
- **Section 6:** Appendices with supporting details and visuals.

2 Overall Description

2.1 Product Perspective

The system is a **standalone mobile application** with cloud-based backend integration via **Firebase**. It is not a replacement for an existing system but rather a **new, centralized solution** that automates campus scheduling and room allocation.

2.2 Product Functions:

Real-time room/lab availability display.

Faculty login for scheduling and room booking.

Student login to view class timetables.
Admin dashboard for managing users, rooms, and departments.

Notification system for schedule updates or room changes.

2.3 User Characteristics

User Type	Description	Technical Expertise
Administrator	Manages rooms, schedules, and accounts	Intermediate
Faculty Member	Books rooms, manages schedules	Basic to Intermediate
Student	Views timetables and notifications	Basic

2.4 Constraints:

Requires a stable **internet connection** for real-time updates.

Limited to **Android/iOS mobile devices**.

Firebase free tier may impose **storage or query limits**.

Must comply with **UET data protection and privacy policies**.

2.5 Assumptions and Dependencies

Users have valid university credentials.

System depends on Firebase for authentication and data storage.

Users will have access to smartphones with updated browsers or app stores.

3 Specific Requirement:

3.1 Functional Requirements

ID	Requirement Description	Priority
1	The system shall allow administrators to add, update, and delete room and lab details.	High
2	The system shall allow faculty to view available rooms/labs and book them for classes.	High
3	The system shall display real-time room/lab occupancy status to users.	High
4	The system shall allow students to view their semester-wise timetable.	High
5	The system shall send notifications about schedule changes or room updates.	Medium
6	The system shall authenticate users via Firebase Authentication.	High
7	The system shall allow administrators to generate reports on room usage.	Medium

3.2 Non-Functional Requirements

Type	Description
Performance	The app must load room/timetable data within 3 seconds.
Reliability	The system must maintain 99% uptime for Firebase services.
Security	All data communication must be encrypted using HTTPS.
Usability	Simple and intuitive UI designed in FlutterFlow for non-technical users.
Scalability	Should handle growth to multiple campuses without code overhaul.
Maintainability	Modular design to allow easy feature updates.
Availability	Cloud-based system accessible 24/7.

3.3 External Interface Requirements

User Interface:
Mobile UI built in FlutterFlow with dropdowns for semester/section, timetable images, and booking forms.

Hardware Interface:
Android or iOS mobile devices.

Software Interface:
Firebase (Authentication, Firestore, Cloud Storage).

Communication Interface:
Internet connectivity using REST APIs provided by Firebase.

3.4 System Features

Feature 1: Room and Lab Management

- **Description:** Admin adds and manages rooms/labs.
- **Inputs:** Room ID, Capacity, Type, Availability.
- **Processing:** Store and update records in Firebase.
- **Outputs:** Room list and status.

Feature 2: Timetable Display

- **Description:** Students view semester and section-wise timetables.
- **Inputs:** Semester, Section selection.
- **Processing:** Retrieve and display corresponding timetable image.
- **Outputs:** Visual timetable.

Feature 3: Booking and Scheduling

- **Description:** Teachers can book rooms for lectures/labs.
- **Inputs:** Date, Time, Room selection.
- **Processing:** Check conflicts and confirm booking.
- **Outputs:** Booking confirmation.

4 System Models

Use Case Diagram

Actors: Admin, Teacher, Student

Main Use Cases:

- Admin → Manage rooms, view reports
- Teacher → Book rooms, view schedules
- Student → View timetable

Data Flow Diagram (DFD – Level 0)

1. **Input:** User credentials, room details, booking info
2. **Process:** Validate, store, and fetch data
3. **Output:** Timetable display, booking confirmation, status updates

5 Other Requirements

Backup & Recovery:
Firebase automatic backup and recovery ensure data persistence.

Security:
Role-based access control (Admin, Faculty, Student).

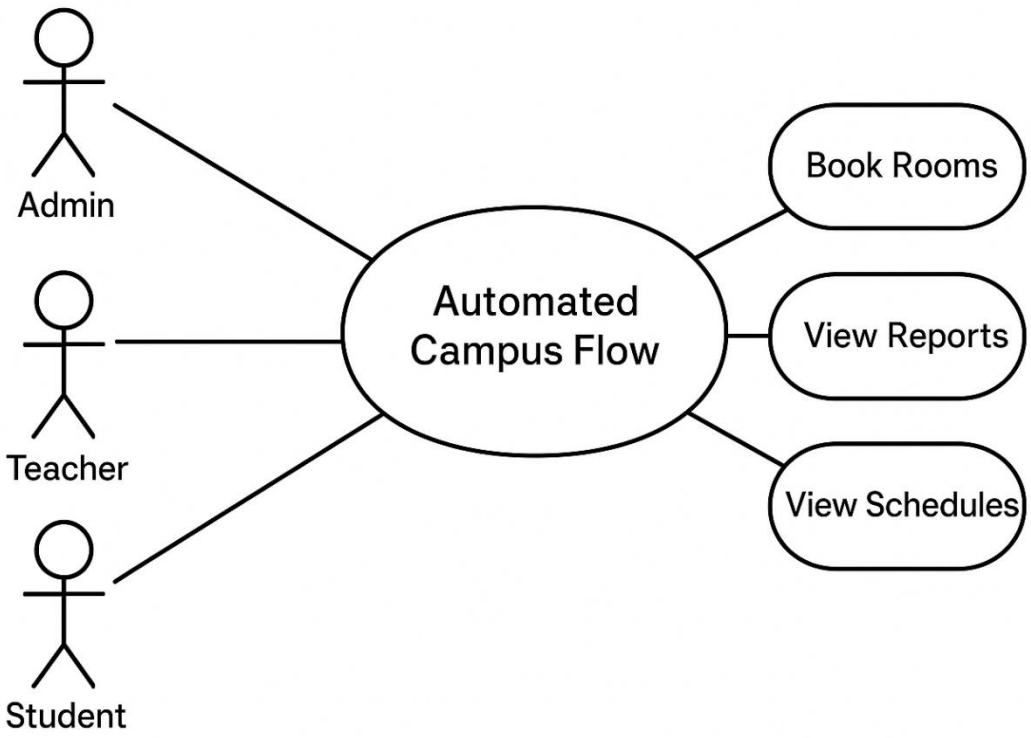
Legal:
Must comply with UET policies and not misuse personal data.

Standards Compliance:
Follow IEEE 830 and Google Material Design guidelines.

6 Appendices

Glossary

Term	Meaning
Timetable	Weekly class schedule for students
Booking	Reserving a room/lab for class
Availability	Current status of a room (occupied/free)


Admin
Teacher
Student
Automated
Campus Flow
Book Rooms
View Reports
View Schedules