

Software Requirements Specification (SRS)

University Timetabling App

Version: 1.0

Date: September 04, 2025

Prepared by: Soon Phei Tin, Requirements Engineering Team

1. Introduction

1.1 Purpose

This Software Requirements Specification (SRS) defines the functional and non-functional requirements for the University Timetabling App, a mobile and web application to manage class schedules, room bookings, and notifications at the University of Spacetra vel. The document serves as a contract between stakeholders (students, lecturers, administrators, IT staff) and the development team, ensuring all requirements are clear, traceable, and verifiable. It is designed for use in an in-house development project with a 6-month timeline.

1.2 Scope

The University Timetabling App will:

- Allow students to view their class schedules and receive notifications for changes.
- Enable lecturers to request room changes and receive confirmations.
- Support administrators in managing room assignments and generating usage reports.
- Integrate with the university's SQL-based student database.
- Ensure accessibility, security, and cross-platform compatibility (iOS, Android, web).

The app aims to replace the current manual system (spreadsheets and emails), reducing errors like room conflicts and improving efficiency. It does not include virtual reality interfaces or advanced AI features due to budget and timeline constraints.

1.3 Definitions, Acronyms, and Abbreviations

- **SRS:** Software Requirements Specification.
- **WCAG:** Web Content Accessibility Guidelines.
- **MoSCoW:** Must Have, Should Have, Could Have, Won't Have prioritization.
- **FR:** Functional Requirement.
- **NFR:** Non-Functional Requirement.
- **SQL:** Structured Query Language (for database integration).

1.4 References

- Stakeholder interviews with Sarah (student), Dr. Li (lecturer), Ms. Lee (administrator).
 - Scenarios: View Schedule, Request Room Change, Generate Report.
 - IEEE 830-1998 Standard for Software Requirements Specifications.
-

2. Overall Description

2.1 Product Perspective

The University Timetabling App is a standalone system that integrates with the university's SQL-based student database for user and course data. It replaces the manual timetabling process, providing a mobile and web interface for stakeholders. The system interacts with external notification services (e.g., push notifications, email) and must comply with accessibility standards (WCAG 2.1).

2.2 Product Functions

- **Students:** View weekly timetables, receive push notifications for schedule changes.
- **Lecturers:** View schedules, request room changes, receive email confirmations.
- **Administrators:** Approve room changes, generate room usage reports, receive capacity alerts.
- **General:** Secure login, cross-platform access, accessible interface.

2.3 User Classes and Characteristics

- **Students:** Tech-savvy, need mobile access, some require accessibility features (e.g., text-to-speech).
- **Lecturers:** Use web or mobile, require reliable scheduling and notifications.
- **Administrators:** Use web interface, focus on room management and compliance.
- **IT Staff:** Maintain system, ensure database integration and uptime.

2.4 Operating Environment

- Mobile: iOS (version 15+), Android (version 10+).
- Web: Modern browsers (Chrome, Firefox, Safari).
- Backend: SQL database integration, hosted on university servers.

2.5 Constraints

- Development: In-house, 6-month timeline.
- Budget: Limited to existing resources.
- Compliance: WCAG 2.1 accessibility standards.
- Integration: Must use existing SQL database.

2.6 Assumptions and Dependencies

- **Assumption:** Stakeholders have access to compatible devices.
- **Dependency:** University provides SQL database schema and API access.

3. Specific Requirements

3.1 Functional Requirements

Each requirement includes its ID, description, MoSCoW priority, source, and rationale for traceability and prioritization.

ID	Requirement	Priority	Source	Rationale
FR1	The system shall allow students to view their weekly timetable on mobile devices, displaying class details (e.g., course, room, lecturer, time).	Must	Sarah (Interview), Scenario 1	Core functionality for students to access schedules on the go, addressing current issues with PDF-based schedules.
FR2	The system shall allow lecturers to request a room change for conflicting slots, selecting from available rooms.	Should	Dr. Li (Interview), Scenario 2	Reduces conflicts but less frequent than viewing schedules; supports lecturer efficiency.
FR3	The system shall generate monthly room usage reports for administrators in PDF format, including occupancy rates.	Should	Ms. Lee (Interview), Scenario 3	Supports administrative planning but not daily use; enhances efficiency.
FR4	The system shall send push notifications to students for schedule changes or cancellations within 5 minutes.	Should	Sarah (Interview), Scenario 1	Enhances student experience by ensuring timely updates; not critical for core function.
FR5	The system shall send email notifications to lecturers for room change confirmations after admin approval.	Could	Dr. Li (Interview), Scenario 2	Useful but lecturers can check manually if needed; lower priority.
FR6	The system shall allow administrators to approve or reject room change requests via a web interface.	Should	Ms. Lee (Interview), Scenario 2	Critical for room management workflow, supports FR2.
FR7	The system shall alert administrators if a room's capacity is exceeded during scheduling.	Must	Ms. Lee (Interview), Scenario 3	Prevents operational issues and ensures compliance with safety regulations.
FR8	The system shall authenticate users via university ID for	Must	Scenario 1	Essential for access control and data security.

ID	Requirement	Priority	Source	Rationale
	secure login.			

3.2 Non-Functional Requirements

ID	Requirement	Priority	Source	Rationale
NFR1	The system shall load timetables within 3 seconds on mobile devices with standard network conditions.	Should	Sarah (Interview, clarified "quick")	Improves user experience; slightly slower load acceptable if secure.
NFR2	The system shall comply with WCAG 2.1 accessibility standards, including text-to-speech support.	Must	Sarah, Ms. Lee (Interview)	Legal requirement; ensures inclusivity for all users.
NFR3	The system shall achieve 99.9% uptime for reliability.	Must	Ms. Lee (Interview, implied "reliable")	Critical for user trust and continuous access.
NFR4	The system shall support iOS (version 15+) and Android (version 10+) mobile platforms.	Must	Sarah (Interview)	Ensures accessibility for primary student devices.
NFR5	The system shall integrate with the university's SQL-based student database for user and course data.	Must	Ms. Lee (Interview)	Required for core functionality and data accuracy.
NFR6	The system shall ensure secure login using lightweight OAuth 2.0 to prevent unauthorized access.	Must	Dr. Li (Interview)	Protects sensitive schedule data; OAuth balances security and performance.
NFR7	The system shall deliver notifications within 5 minutes of schedule changes.	Should	Sarah (Interview, clarified "timely")	Timely updates improve UX but slight delays are tolerable.
NFR8	The system shall be usable on modern web browsers (Chrome, Firefox, Safari) for desktop access.	Should	Ms. Lee, Dr. Patel (Interview)	Supports admin/lecturer access but secondary to mobile.

3.3 Conflict Resolutions

Conflicts identified during analysis (from Stage 2) are resolved to ensure consistency:

- **Performance vs. Security (NFR1 vs. NFR6):** Lightweight OAuth 2.0 is used for secure login to balance load time (<3s). Tested on average devices to ensure compliance.
- **Timely Notifications vs. System Load (NFR7 vs. NFR3):** Batch processing for notifications prioritizes critical alerts to maintain 99.9% uptime.
- **Accessibility vs. Development Time (NFR2 vs. 6-month timeline):** Core WCAG features (e.g., high-contrast UI, screen reader support) implemented in first release; text-to-speech deferred to later iterations if needed.
- **Room Change Speed vs. Approval (FR2 vs. FR6):** Temporary room swaps allowed pending admin approval within 1 hour to balance speed and control.

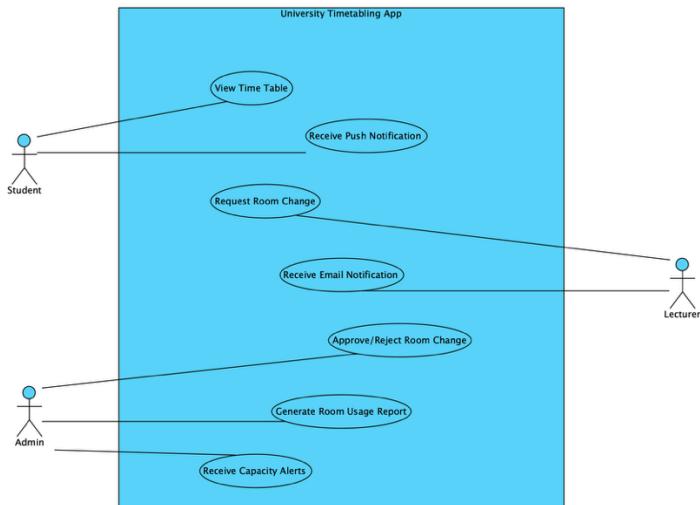
4. Supporting Information

4.1 Assumptions

- All users have access to compatible devices (iOS 15+, Android 10+, modern browsers).
- The university provides SQL database schema and API documentation.

4.2 Use Case Diagram

Below is a simplified use case diagram for key interactions.



4.3 Verification Plan

- **FR1:** Test timetable display on iOS/Android; verify details shown within 3s.
- **FR7:** Simulate scheduling with over-capacity rooms; verify alert triggers.
- **NFR2:** Use accessibility tools (e.g., WAVE) to confirm WCAG 2.1 compliance.
- **NFR3:** Monitor system uptime over 30 days to ensure 99.9% reliability.

4.4 Traceability Matrix

Requirement ID	Source	Stakeholder	Priority
FR1	Sarah, Scenario 1	Student	Must
FR2	Dr. Patel, Scenario 2	Lecturer	Should
FR3	Ms. Lee, Scenario 3	Administrator	Should
FR4	Sarah, Scenario 1	Student	Should
FR5	Dr. Patel, Scenario 2	Lecturer	Could
FR6	Ms. Lee, Scenario 2	Administrator	Should
FR7	Ms. Lee, Scenario 3	Administrator	Must
FR8	Scenario 1	All	Must
NFR1	Sarah	Student	Should
NFR2	Sarah, Ms. Lee	All	Must
NFR3	Ms. Lee	All	Must
NFR4	Sarah	Student	Must
NFR5	Ms. Lee	Administrator	Must
NFR6	Dr. Patel	All	Must
NFR7	Sarah	Student	Should
NFR8	Ms. Lee, Dr. Patel	Administrator, Lecturer	Should
