

Software Requirements Specification (SRS) for Classroom planner for multiple batches running in parallel

Table of Contents

1. Introduction
 - 1.1 Purpose
 - 1.2 Scope
 - 1.3 Definitions, Acronyms, and Abbreviations
 - 1.4 References
2. Overall Description
 - 2.1 Product Perspective
 - 2.2 Product Functions
 - 2.3 User Classes and Characteristics
 - 2.4 Operating Environment
 - 2.5 Design and Implementation Constraints
3. Specific Requirements
 - 3.1 User Interface
 - 3.2 Authentication and Authorization
 - 3.3 News Management
 - 3.4 User Management
 - 3.5 Search and Filters
 - 3.6 Notifications
 - 3.7 Performance Requirements
 - 3.8 Security Requirements
 - 3.9 Database Requirements
 - 3.10 Reporting and Analytics
 - 3.11 Maintenance and Support
 - 3.12 Legal and Compliance
4. External Interface Requirements
 - 4.1 User Interfaces
 - 4.2 Hardware Interfaces
 - 4.3 Software Interfaces
 - 4.4 Communication Interfaces
5. Non-functional Requirements
 - 5.1 Usability
 - 5.2 Reliability
 - 5.3 Scalability
 - 5.4 Availability
 - 5.5 Response Time
 - 5.6 Data Backup and Recovery
 - 5.7 Accessibility
 - 5.8 Compatibility
6. System Models
 - 6.1 Use Case Diagram
 - 6.2 Class Diagram
 - 6.3 Entity-Relationship Diagram (ERD)
 - 6.4 Sequence Diagram

- 7. Appendices
 - 7.1 Glossary
 - 7.2 Document Revision History

Software Requirements Specification (SRS)

For: Classroom Planner for Multiple Batches Running in Parallel

Version: 1.0.0

Date: 3rd May 2025

1. Introduction

1.1 Purpose

This document outlines the software requirements for a Classroom Planner designed to manage and schedule academic courses, instructors, classrooms, and student batches efficiently, avoiding conflicts and ensuring optimal utilization of institutional resources.

1.2 Scope

The Classroom Planner is a web-based scheduling solution that automates slot management, course allocation, and conflict detection for educational institutions. The system ensures that classrooms, time slots, and instructors are assigned without overlaps and supports multiple batches running in parallel.

1.3 Definitions, Acronyms, and Abbreviations

- **Slot:** A defined unit of time allocated for a course session.
- **Session:** A specific academic term or semester.
- **Conflict Resolution:** Process to prevent scheduling overlaps.
- **JWT:** JSON Web Token used for secure authentication.

1.4 References

- IEEE Software Requirements Specification Template by Karl E. Wiegers
 - UML Diagrams
 - Institutional academic policies and scheduling guidelines
-

2. Overall Description

2.1 Product Perspective

The Classroom Planner is an independent web application with optional integration capabilities with existing university management systems. It is modular and scalable to support growing numbers of courses and students.

2.2 Product Functions

- Define and allocate time slots
- Assign instructors and manage student enrollments
- Detect and resolve scheduling conflicts
- Provide role-based dashboards and notifications
- Generate visual timetables and reports

2.3 User Classes and Characteristics

- **Administrators:** Full access to scheduling, slot creation, and user management.
- **Instructors:** Can view and manage assigned courses and schedules.
- **Students:** Can view their course timetables and enrolled sessions.

2.4 Operating Environment

- Platform: Web-based
- Operating Systems: Windows, Linux, macOS
- Backend: Node.js, Django, or Flask
- Frontend: React.js or Vue.js
- Database: PostgreSQL or MySQL

2.5 Design and Implementation Constraints

- Efficient concurrency handling for multiple users
- Secure data storage and access
- Optimization for timetable generation

3. Specific Requirements

3.1 User Interface

- Responsive web dashboards
- Interactive timetable grid
- Accessibility-compliant layout
- Dark/light mode options

3.2 Authentication and Authorization

- Secure login with JWT tokens

- Role-based access control (Admin, Instructor, Student)
- Password reset and account verification

3.3 Slot Management

- Define, edit, and merge slots
- Assign time periods to sessions
- Display availability in a grid view

3.4 Course Scheduling

- Assign courses to instructors and students
- Auto-detect and resolve conflicts
- Maintain hash maps for performance

3.5 User Management

- Create, modify, or remove users
- Assign roles and permissions
- Audit log of user activity

3.6 Search and Filters

- Search by course, instructor, or student
- Filter schedules by department, semester, or date range

3.7 Notifications

- Email alerts for slot updates or conflicts
- Real-time schedule changes via push notifications

3.8 Performance Requirements

- Schedule processing under 2 seconds
- Support for 1000+ concurrent users

3.9 Security Requirements

- Data encryption in transit and at rest
- Secure APIs and input validation

- Two-factor authentication (optional)

3.10 Database Requirements

- Relational schema for users, sessions, slots, and courses
- Indexing for fast conflict detection
- Support for audit and rollback

3.11 Reporting and Analytics

- Daily/weekly timetable reports
- Conflict frequency analysis
- Slot utilization metrics

3.12 Maintenance and Support

- Admin dashboard for issue management
- Documentation and user manuals
- Helpdesk integration (optional)

3.13 Legal and Compliance

- GDPR compliance for user data
- Institutional policy adherence
- Logging and consent records

4. External Interface Requirements

4.1 User Interfaces

- Admin dashboard
- Instructor panel
- Student view
- Accessible UI/UX

4.2 Hardware Interfaces

- Compatible with standard hardware (4GB RAM, dual-core CPU minimum)

4.3 Software Interfaces

- REST APIs
- LDAP or OAuth authentication integration

4.4 Communication Interfaces

- HTTPS communication
 - SMTP for email notifications
-

5. Non-functional Requirements

5.1 Usability

- Intuitive interface
- Tutorials and help guides
- Multi-language support (optional)

5.2 Reliability

- Uptime > 99.9%
- Regular backups and rollback support

5.3 Scalability

- Modular architecture
- Cloud deployment ready

5.4 Availability

- 24/7 access
- Load balancing and failover support

5.5 Response Time

- UI response < 1 second
- Conflict resolution < 2 seconds

5.6 Data Backup and Recovery

- Nightly full backups
- Point-in-time recovery

5.7 Accessibility

- WCAG 2.1 compliance
- Keyboard navigation support

5.8 Compatibility

- Browser support (Chrome, Firefox, Safari, Edge)
- Mobile responsive design

6. System Models

6.1 Use Case Diagram

- Depicts user interactions (Admin, Instructor, Student)

6.2 Class Diagram

- Shows entities like User, Course, Slot, Timetable

6.3 Entity-Relationship Diagram (ERD)

- Relationships among Users, Sessions, Slots, Courses

6.4 Sequence Diagram

- Timetable generation flow from slot selection to final schedule

7. Appendices

7.1 Glossary

Terms like Slot, Session, Timetable, Conflict Resolution, etc.

7.2 Document Revision History

Version	Date	Changes
1.0	03-05-2025	Initial Draft