StudyBuddy
System Requirements Specification (SRS)

Ait Lahsen Mohamed, Ait Aadi Youssef
Course: Software Engineering

November, 2025

Contents

1 Introduction

1.1 Purpose

This document defines the Software Requirements Specification (SRS) for StudyBuddy, a university Learning Management System. It outlines the system's purpose, high-level description, functional and non-functional requirements, and technical specifications.

1.2 Scope

StudyBuddy enables academic management through:

- Courses and class assignment
- Modules and learning materials
- Assignments, submissions, and grading
- Real-time discussion chat
- Dashboard notifications
- Student to-do list

The system supports three roles: Student, Teacher, and Administrator.

1.3 Definitions, Acronyms, and Abbreviations

- **LMS:** Learning Management System
- **JWT:** JSON Web Token
- **STOMP:** Simple Text Oriented Messaging Protocol
- **Module:** A content section of a course

2 Overall Description

2.1 Product Perspective

StudyBuddy is a standalone web platform developed using React (frontend) and Spring Boot (backend). PostgreSQL manages persistent data, and WebSocket STOMP ensures real-time communication.

2.2 Product Functions

The main system functions include:

- Role-based authentication using JWT
- Course management by administrators

- Module and assignment creation by teachers
- Assignment submissions by students
- Grading workflow for teachers
- Real-time discussion chat per course
- Dashboard notifications for major events
- To-do list management for students

2.3 User Characteristics

- **Students:** Access to courses of their assigned class.
- **Teachers:** Manage content, assignments, discussions, and grades.
- **Administrators:** Configure courses, teachers, and classes.

2.4 Assumptions and Dependencies

- Stable internet connection
- PostgreSQL and Spring Boot services operational
- Supported modern browsers
- WebSocket server running for discussion features

3 Functional Requirements

3.1 Authentication

- FR1: Users shall register with a selected role.
- FR2: Admin registration shall require the password “@dmin”.
- FR3: The system shall authenticate users using JWT.

3.2 Course Management

- FR4: Admin shall create courses with title, description, class, and teacher.
- FR5: Students shall view only courses assigned to their class.
- FR6: Teachers shall view only courses they teach.

3.3 Modules & Items

- FR7: Teachers shall create and edit course modules.
- FR8: Teachers shall add items of type TEXT or FILE.
- FR9: Students shall view modules of their enrolled courses.

3.4 Assignments & Submissions

- FR10: Teachers shall create assignments with due dates and points.
- FR11: Students shall submit responses as text or file URLs.
- FR12: Teachers shall grade submissions.
- FR13: Students shall view their grades.

3.5 Discussion System

- FR14: The system shall support real-time course discussions via WebSocket.
- FR15: Messages shall be saved in the database.
- FR16: Users shall retrieve discussion history.

3.6 People Management

- FR17: Admin shall view teacher and students of each course.
- FR18: Teachers shall view enrolled students.
- FR19: Students shall view classmates and their teacher.

3.7 Dashboard Events

- FR20: The system shall generate events for assignments, submissions, and grades.
- FR21: Users shall retrieve their latest 10 events.

3.8 To-Do List

- FR22: Students shall create, update, delete, and complete tasks.

4 Non-Functional Requirements

- NFR1: The system shall be responsive across devices.
- NFR2: Page load time shall not exceed 2 seconds.
- NFR3: All APIs shall enforce JWT authentication.
- NFR4: WebSocket latency shall remain minimal.
- NFR5: The UI shall maintain consistent design.

5 Tools and Technologies

- React.js, Vite
- Spring Boot, JPA
- PostgreSQL
- WebSocket + STOMP
- GitHub

6 Technical Challenges

- Real-time message synchronization
- Role-based access enforcement
- Database consistency across modules, assignments, and events

7 Validation and Testing

- Unit testing for controllers, services, and UI components
- Integration testing for authentication and real-time chat
- Database testing for submissions and grading workflows

8 Conclusion

This SRS defines all requirements of the updated StudyBuddy LMS. It provides the foundation for system design, implementation, and testing phases while ensuring clarity and consistency for all stakeholders.