StudyBuddy

System Requirements Specification (SRS)

Submitted by: Name1, Name2
Course: Software Engineering
Date: October 7, 2025

Contents

1 Introduction

1.1 Purpose

This document defines the software requirements for the **StudyBuddy** project. It provides a comprehensive description of the system's functionality, non-functional requirements, and technical specifications. The SRS serves as a foundation for the design, implementation, and validation phases.

1.2 Scope

StudyBuddy is a web-based productivity platform designed for students to manage their academic activities efficiently. It combines organizational tools, collaboration features, and gamification elements to enhance motivation and performance.

The main functionalities include:

- Personal **Agenda** and **Task Management**.
- Real-time **Chat** for collaboration.
- **Reminders** and notifications for pending tasks.
- **Notes** creation and organization.
- **Ranking system** based on completed tasks and competition.

1.3 Definitions, Acronyms, and Abbreviations

- **User:** Any registered student using the platform.
- **Admin:** Person responsible for managing users and maintaining the system.
- **SRS:** System Requirements Specification.

1.4 References

Based on the course "Software Engineering" requirements assignment and feasibility study of StudyBuddy (2025), chat gpt .

2 Overall Description

2.1 Product Perspective

StudyBuddy is a standalone, web-based system built with a React frontend and Firebase backend. It supports real-time data synchronization, notifications, and secure user authentication. Hosting will be done on Vercel or Firebase Hosting.

2.2 Product Functions

1. **Agenda & Tasks:** Create, view, edit, and delete study tasks with deadlines.
2. **Reminders:** Automatic push notifications for upcoming or overdue tasks.
3. **Notes:** Users can take, organize, and store personal study notes.
4. **Chat:** Real-time communication system between users or study groups.
5. **Gamification:** Points and rankings based on completed tasks.
6. **Competition:** Leaderboard displaying users' performance.
7. **Calendar Integration:** Detect schedule conflicts and display upcoming events.

2.3 User Characteristics

Users are mainly students at high school or university level, familiar with modern web interfaces and online productivity tools.

2.4 Assumptions and Dependencies

- Users must have a stable internet connection.
- The system will operate in modern browsers (Chrome, Firefox, Edge).
- Firebase services will be available and functional.

3 Functional Requirements

- **FR1:** The system shall allow users to register, log in, and log out using Firebase Authentication.
- **FR2:** The system shall allow users to add, update, and delete tasks in their agenda.
- **FR3:** The system shall automatically send push notifications for upcoming or overdue tasks.
- **FR4:** The system shall display a calendar view and detect task scheduling conflicts.
- **FR5:** The system shall allow users to take and organize personal notes.
- **FR6:** The system shall allow users to chat in real-time with other users or groups.
- **FR7:** The chat messages shall synchronize instantly across devices.
- **FR8:** The system shall assign points when a user completes a task.
- **FR9:** The system shall update and display the leaderboard dynamically.
- **FR10:** The system shall allow users to view their progress and ranking history.

4 Non-Functional Requirements

- **NFR1:** The system shall be responsive across devices (desktop, tablet, mobile).
- **NFR2:** The average page load time shall not exceed 2 seconds.
- **NFR3:** Data shall be encrypted and securely stored using Firebase Security Rules.
- **NFR4:** The system shall handle real-time data updates with minimal latency.
- **NFR5:** The user interface shall be intuitive, minimal, and consistent.
- **NFR6:** The system shall support concurrent users without significant performance loss.
- **NFR7:** All critical actions (login, delete, logout) must provide confirmation messages.

5 Tools and Technologies

- **Frontend:** React.js, HTML, CSS, JavaScript
- **Backend, Database & Authentication:** Firebase
- **Hosting:** Vercel / Firebase Hosting
- **Version Control:** GitHub

6 Technical Challenges

- Real-time synchronization for chat and agenda across multiple users.
- Implementation of a fair gamification and ranking algorithm.
- Integration of calendar conflict detection and event scheduling.
- Push notification delivery using Firebase Cloud Messaging (FCM).

7 Validation and Testing

Testing will ensure all requirements are met:

- Unit testing for all frontend components and Firebase operations.
- Integration testing for chat, authentication, and leaderboard updates.
- Usability testing to verify intuitive design and ease of navigation.

8 Conclusion

This SRS defines the full set of requirements for the **StudyBuddy** project. It ensures that all functional, non-functional, and technical specifications are clearly outlined, allowing consistent development, testing, and future improvement.