Software Requirements Specification (SRS)

1. Introduction

This app is a comprehensive platform designed to assist teachers and educators in efficiently managing their teaching responsibilities. The app aims to provide a user-friendly interface for creating and managing syllabi, tracking deadlines and events, and personalizing app settings. This document outlines the high-level requirements for the app.

2. Functional Requirements

2.1 Syllabus Page

- Teachers can log in to their accounts to access the app.
- Teachers can create, edit, and delete syllabi for various subjects they are teaching.
- Syllabi should include course objectives, topics, reading materials, assignments, and other relevant information.
- Syllabi can be organized by subject and term for easy retrieval.
- Teachers can easily navigate and search for specific syllabi.

2.2 Calendar Page

- Teachers can view a visual calendar that displays upcoming deadlines and events.
- Teachers can input due dates for assignments, exams, and other important events.
- The calendar should support different views (day, week, month) for flexible event tracking.
- Pop-up reminders and alerts are generated to notify teachers of approaching deadlines.

2.3 Landing Page

- Upon login, teachers are directed to the landing page, which serves as the app's main navigation hub.
- The landing page provides options to access the syllabus page, calendar page, and settings.
- Teachers can customize their app experience by adjusting settings and personal details.

3. Non-Functional Requirements

3.1 Usability

- The app's user interface (UI) should be intuitive and user-friendly, catering to users with varying levels of technical expertise.
- Navigational elements should be clearly labeled and easy to understand.

3.2 Performance

- The app should load and respond quickly, even when handling large amounts of syllabus data and calendar events.
- The calendar should update in real-time to reflect any changes made by teachers.

3.3 Security

- User authentication and authorization mechanisms should ensure secure access to the app's features.
- Teacher accounts should be protected from unauthorized access.

4. Constraints

- The app will be developed using React.js for the front-end and may require integration with backend services for user authentication and data storage.
- The app's development timeline is constrained by the academic semester schedule.

5. Assumptions

- Users are assumed to have basic computer literacy and internet connectivity.
- The app will be primarily used on web browsers, with mobile responsiveness as a secondary consideration.

6. Dependencies

- The app may depend on external libraries or APIs for features such as the calendar and user authentication.

7. Conclusion

This React App aims to revolutionize how teachers manage their teaching responsibilities by providing a streamlined platform for syllabus management, event tracking, and personalization.

By adhering to the functional and non-functional requirements outlined in this document, the app will contribute to enhancing educators' efficiency and productivity in their roles.