

Summary of Proposed Solution

The proposed solution for the uLesson EdTech app involves the development of a robust React-based web platform, with optional TypeScript integration, to provide learners with seamless access to online video lessons and limited offline functionality. The primary goal is to ensure system stability, scalability, and adherence to good coding principles for future extensibility by junior team members.

System High-Level Design:

The system architecture is designed to prioritize stability and scalability. The React framework will be utilized, with the option to integrate TypeScript for enhanced type safety. Redux, though not mandatory, will be considered for state management to facilitate a centralized and predictable state container.

Prototype Overview:

1. Home Screen:

- The home screen will serve as the central hub for users to access the app's main features.
- Key elements on the home screen include:
 - Browse and stream new video lessons.
 - Shortcut to resume viewing the last unfinished video.

2. Video Streaming:

- Videos will be streamed online to optimize content loading.
- The system will cache the current video and the next one in sequence, ensuring efficient streaming.
- Whenever possible, caching for non-current videos will be performed when connected to WiFi.

3. Offline Functionality:

- Cached content will be available for offline access, allowing learners to continue their studies without an internet connection.
- If a video is cached, it will be played from the cache rather than streaming.

4. Note-Taking and Bookmarks:

- Users will have the ability to make notes and tag specific portions of the video, creating bookmarks for easy reference.

5. Daily Log-In Reward:

- A 'daily log-in' reward feature will be implemented to encourage user engagement.
- Learners who complete watching their first video of the day will collect a badge.

Stability and Scalability:

- The system design prioritizes stability through efficient state management (potentially using Redux) and proper error handling.
- Scalability is addressed by structuring the codebase with modular components, adhering to coding standards and design patterns suitable for future extension.

Documentation:

- The system design document will include UML diagrams, sequence diagrams for main flows, and contracts with the backend.
- The documentation will focus on communicating the proposed solution comprehensively, enabling less senior team members to understand and continue development.

In conclusion, the proposed solution aims to provide a well-structured, stable, and scalable foundation for the uLesson EdTech app. The emphasis on clear documentation ensures that junior team members can confidently contribute to the project's ongoing development.