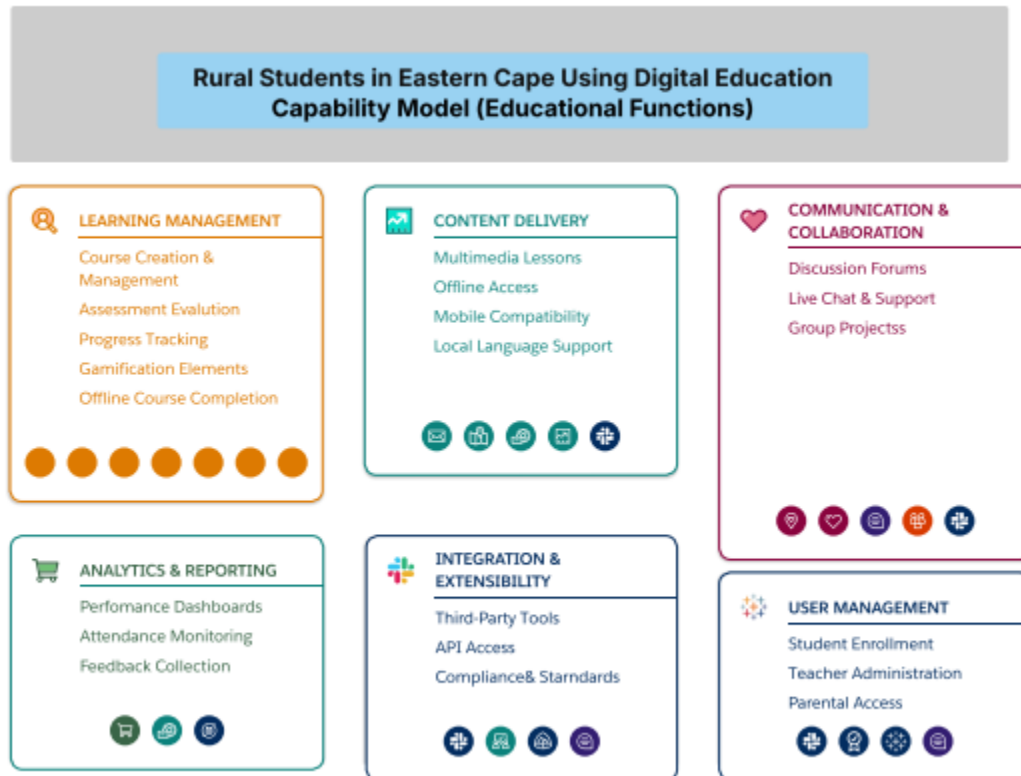


1. Capability model



A. Learning Management

- **Course Creation and Management:** Developing and organizing course materials.
- **Assessment and Evaluation:** Quizzes, assignments, and exams.
- **Progress Tracking:** Monitoring student performance and milestones.
- **Gamification Elements:** Reward systems, badges, and leaderboards to boost engagement.
- **Offline Course Completion:** Ability to complete lessons and quizzes without internet access.

B. User Management

- **Role-Based Access:** Students, teachers, and administrators.

- **Profile Management:** User information and preferences.

C. Communication and Collaboration

- **Discussion Forums:** Peer-to-peer and student-teacher interactions.
- **Messaging Systems:** Direct and group messaging.
- **Virtual Classrooms:** Live interactive sessions.

D. Content Management

- **Resource Repository:** Storage and retrieval of learning materials.
- **Multimedia Support:** Videos, audios, and interactive content.

E. Analytics and Reporting

- **Performance Dashboards:** Visual representation of progress.
- **Custom Reports:** Detailed insights for stakeholders.

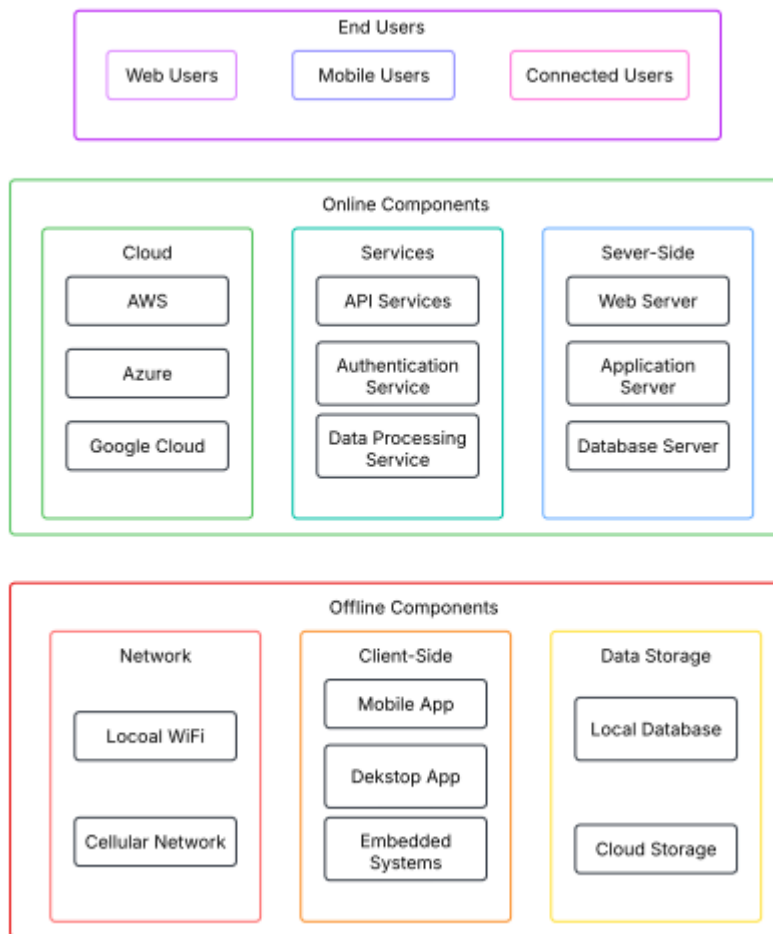
F. Accessibility and Inclusivity

- **Multilingual Support:** Catering to diverse linguistic backgrounds.
- **Offline Access:** Functionality without continuous internet.

G. Integration and Extensibility

- **Third-Party Tools:** Seamless integration with other educational tools.
- **API Support:** For custom functionalities.

2. Technology Architecture: Offline/Online Components



End Users

- Students, Teachers, and Administrators who interact with the system

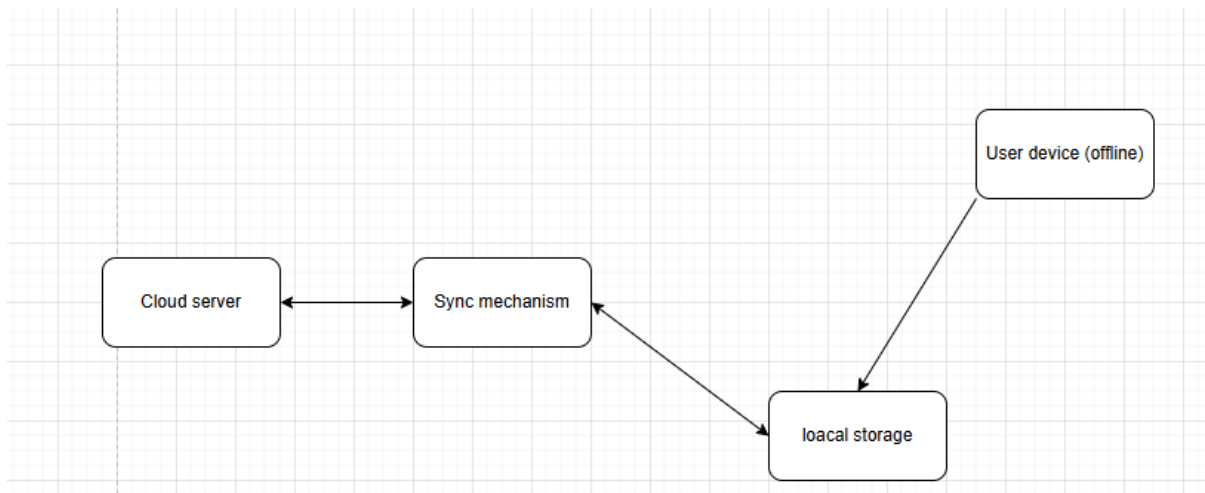
Offline Components

- **Progressive Web App/Hybrid Mobile App:** The main interface accessible across devices without heavy installation
- **Local Storage:** Enables offline access to educational content
- **Service Workers:** Handle background synchronization when connectivity returns
- **Data Compression:** Optimizes media for low-bandwidth environments
- **Offline Data Storage:** Preserves student progress locally
- **User Interface:** Simplified design with accessibility features including dark mode, high-contrast mode, voice commands, and gesture controls

Online Components

- **Cloud Infrastructure:** Scalable backend (AWS/Azure/Google Cloud/Firebase)
- **Load Balancing & CDN:** Ensures global content delivery
- **Role-Based Access Control:** Manages permissions and data access
- **Smart Data Syncing:** Coordinates updates between offline and online systems
- **Conflict Resolution:** Prevents data loss during synchronization
- **Content Management System:** With AI recommendations and moderation tools
- **Security Measures:** Including encryption, multi-factor authentication, and compliance audits

3. Data architecture: cloud vs local storage



The data Architecture diagram consists of 4 layers, which are known as:

- 1) **User device (offline):** This device allows data or content to be stored locally.
- 2) **Local storage:** It allows web applications to be accessible within the user's browser. The data will not get damaged or deleted. It is also accessed when offline.

Advantage: It is easier to use.

Disadvantage: Its security is not secure.

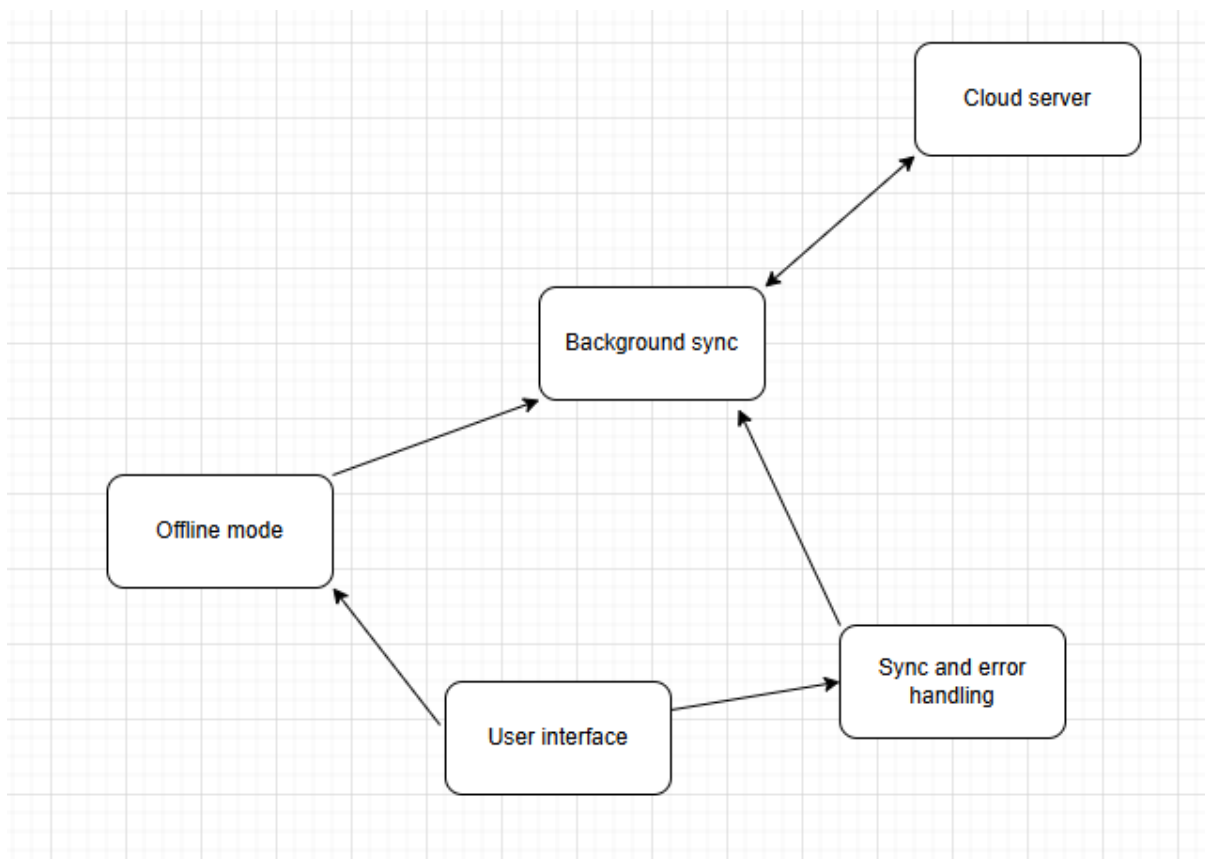
- 3) **Sync Mechanism:** This handles the data transfer when connectivity is available. The data transfer is synchronized with a common signal that is generated by the sending device. It is used by both the sending and receiving devices to ensure there is sync in between them.

Advantage: The design procedure is easy.

Disadvantage: If a slow-speed unit is connected, it can downgrade the overall rate of transfer in the system.

- 4) **Cloud Server:** Stores data centrally for access across devices.

4. Application architecture



- 1) **User interface:** Allows users to interact with the device or a system.
- 2) **Offline mode:** It stores temporary data locally when there is no connectivity or internet.

- 3) **Sync and Error Handling:** It manages certain synchronizations and connectivity issues.
- 4) **Background Sync:** It functions by transferring data in the background when the internet is available.
- 5) **Cloud Server:** It stores and updates data for a device or a system.

These diagrams illustrate how the data is stored locally and when or how it syncs into the cloud.