**Architectural Decision Record**

CPRG-303-B

Date:

21/02/2024

Prepare for:

Nick Hamnett

Prepared by:

Andres Chacon

**Social Networking Mobile App for a University**

**Native, Web, or Hybrid App**

**Issue:**

The Social Networking Mobile App must offer good performance to all users. Allow access to IOS and Android devices no matter how old or new these are, in addition to supporting long-term maintenance that is simple and efficient so it can be performed across different platforms.

**Decision:**

It was determined to use the React Native framework for building a hybrid application. This choice was made in order to use online technologies for faster development and simpler maintenance, while also supporting the iOS and Android platforms.

**Status:**

Accepted

**Consequences:**

* React Native allows us to save time and resources as it permits us to build the app on both IOS and Android platforms with a single base code.
* React Native gives access to the hardware and functions of the device with the help of native modules and plugins, like the camera, GPS, and push notifications. Additionally allowing data synchronization using libraries like AsyncStorage and Redux, as well as an offline mode.
* React Native will provide the app with fast performance and an excellent user experience, allowing live and fast reloading, which lets users view changes made to the program without having to restart it.

**UI Framework**

**Issue:**

The Social Networking Mobile App needs a UI Framework that enables cross-platform development and maintenance, supports offline capabilities, app to app compatibility, and minimal data consumption.

**Decision:**

Material UI is the chosen framework for the application’s user interface. It provides a set of React Components that follows Google’s material design guidelines, and provides a consistent user friendly interface across all platforms.

**Status:**

Accepted

**Consequences:**

* Material UI integrates well with React. This will let us take full advantage of React features, such as its declarative programming model, component-based architecture, and state management capabilities.
* Material UI offers a large number of pre-designed components such as buttons, cards and dialogs that can be easily customized and changed to match our app.
* Material UI supports responsive design, allowing the app to adapt to different screen sizes and orientations. Improving the app compatibility and user experience, especially for users who may not have a high-end smartphone.

**Backend Language**

**Issue:**

The Social Networking Mobile App needs an scalable, dependable, and secure backend server that is also simple to maintain. And it should manage data processing, business logic, and integration with other services like Active Directory.

.

**Decision:**

It had been decided to use Node.js as the backend language, because of its performance and scalability. We believe it is a good selection for managing the mobile app data synchronization, as well as integrating with the data storage and Active Directory.

.

**Status:**

Accepted

**Consequences:**

* By using Node.js we can use web development tools and practices like npm, webpack, and testing frameworks, allowing us for a quick and effective development and deployment.
* Node.js's event-driven and non-blocking I/O model will help us handle multiple concurrent requests and heavy traffic, offering a scalable and dependable server.
* Using Node.js means we rely on a single threaded language to run the server, meaning there could be some efficiency or security risks. Taking this into account we have to plan a way to optimize and secure the server for the safety of all users.

**Permissions**

**Issue:**

The Social Networking Mobile App must guarantee proper access control and protected user data. Has to manage private data like grades, and profiles, and it has to enforce roles and permissions for both instructors and students, while also integrating with the Active Directory System.

**Decision:**

We chose to work with OAuth 2.0 and OpenID Connect (OIDC), two very known protocols for authorization and authentication. This was because they offer a safe method for the app to access the user data, for the user to log in, and because they support Active Directory. In addition we also plan on using Role-Based Access Control (RBAC), which will be used to enforce permissions and roles for students and professors.

**Status:**

Accepted

**Consequences:**

* OAuth 2.0 supports the identity provider we plan on using for the app, Active Directory. It will also let users log in constantly and safely while letting them access user data.
* RBAC is a model that can manage complicated access control needs enabling simple administration and fine-grained access control.
* It follows data protection laws and safeguards about sensitive information, so based on roles and permissions, the app can restrict access to sensitive information and audit access logs to ensure compliance.

**Data Storage**

**Issue:**

The Social Networking Mobile App has to support offline mode in addition to storing and syncing data across many different platforms and devices, as well as protecting private data for each student and instructor like grades or user profiles.

**Decision:**

We decided to use MongoDB as our main database, because of its flexibility and scalability it is perfect to store the users profiles, schedules, announcements, and almost any other relevant data in the app.

**Status:**

Accepted

**Consequences:**

* MongoDB is a documented oriented database that is adaptable and expandable. While offering a data model that can support high traffic quantities of data.
* MongoDB protects private data and sensitive information. It also has security features that allow it to evaluate data activities, control access to data and encrypt both in transit and at rest data.
* For MongoDB some configuration and integration might be necessary as the app needs to properly integrate with it and configure the database settings.

**Additional Frameworks or Technology Stacks**

**Offline Mode and Data Synchronization**

**Issue:**

The Social Networking Mobile App needs to protect the Users data and guarantee a proper access control. Additionally it should enforce permissions and roles for students and instructors, while integrating with the current Active Directory.

**Decision:**

To manage User Authentication in this app we will be using Firebase Authentication. With this users can easily and securely authenticate themselves by using different features like phone number, email, password, or even use identity providers like Apple, Google, or Facebook.

**Status:**

Accepted

**Consequences:**

* Firebase Authentication provides a safe and standardized service that can handle millions of users. And also offers protection against password cracking, or account takeover attacks.
* Firebase offers a consistent user experience by supporting both online and offline real-time data synchronization, as well as authentication on multiple platforms, such as IOS, Android, and Web platforms.
* We can improve the overall User Experience by using Firebase Authentication’s features like password recovery email verification, or account licking.