

Title of ADR

Choosing the development approach for the mobile app development

Status

Proposed

Context

The team is responsible for developing a new mobile app for a retail company. The app's primary functionality includes allowing customers to browse and purchase products, view order history, track deliveries, and participate in a loyalty program. Several key requirements, such as offline mode support, push notifications, payment gateways integration, user behavior tracking, image optimization, and internationalization, need to be addressed during the app's development.

Options Considered • Native • Web • Hybrid

Decision

The team has decided to develop the mobile app using a hybrid approach.

Consequences

The decision to use a hybrid approach has the following consequences:

Pros of Hybrid Approach:

- **Faster development:** A hybrid approach allows us to write code once and deploy it on multiple platforms, reducing development time and effort.
- **Cost-effective:** Developing a single codebase for both iOS and Android can be more cost-effective compared to maintaining separate native apps.
- **Cross-platform consistency:** The hybrid approach ensures a consistent user experience across different platforms.
- **Access to native features:** Hybrid frameworks like React Native or Flutter provide access to native device features, fulfilling the requirement for push notifications, payment gateways, and more.
- **Offline support:** Hybrid apps can be designed to support offline mode through local data storage and synchronization.

Cons of Hybrid Approach:

- **Slightly reduced performance:** While hybrid apps offer native-like performance, they may not match the performance of fully native apps in some cases.
- **Framework dependencies:** Choosing a hybrid framework (e.g., React Native, Flutter) introduces a dependency on that framework's ecosystem and community support.
- **Given the need for rapid development, cost-effectiveness, and access to native features, the hybrid approach aligns with the project's objectives and requirements.**

Rationale

The choice of a hybrid approach provides a balance between development speed, cost-effectiveness, and access to native features. It allows us to efficiently address the retail company's requirements, such as offline

mode support, push notifications, and payment gateways integration. Additionally, the decision aligns with the team's familiarity with hybrid development frameworks, ensuring a smoother development process.

Follow-Up Actions

- Begin the development process using the chosen hybrid framework (e.g., React Native or Flutter).
- Explore and select appropriate libraries and plugins for implementing offline mode, push notifications, and payment gateways integration within the hybrid app.
- Establish a localization strategy and select the necessary localization libraries or tools to support multiple languages and cultural preferences.
- Continuously monitor the app's performance and user experience to ensure it meets the retail company's expectations and requirements.