

Evaluation

Code and Interface Development Process

The development process of the FindJ application followed a linear approach, since the requirements and the design was clearly documented. Source code was regularly reviewed, and interface development was conducted concurrently. However, there were instances where meeting milestones and deadlines posed challenges due to unforeseen technical issues and the integration of complex features.

Meeting Milestones and Deadlines

While the initial planning effectively set milestones, the dynamic nature of mobile development introduced challenges. Issues such as third-party library compatibility and debugging took longer than expected, impacting delays in milestones. In future improvements of FindJ, a more detailed risk assessment and contingency plan should be implemented.

Analysis of the Chosen App Development Framework

The selection of React Native and Expo significantly contributed to an effective development process. Expo streamlined the build and deployment process, simplifying tasks and reducing the need for native development expertise. The hot-reloading feature in React Native fastens the debugging process, enhancing developer productivity.

A significant reason for the selection of React Native and Expo for FindJ is because the technologies are beginner friendly .

Critical Evaluation of an Alternative Framework

A critical evaluation of an alternative framework, such as Flutter, suggests potential benefits and drawbacks. Flutter's "write once, run anywhere" philosophy might provide a consistent user experience across platforms, but the learning

curve and potential limitations in accessing native modules could impact project flexibility.

While Flutter offers advantages, transitioning from React Native to Flutter may introduce challenges in terms of retraining the development team and migrating existing codebases. Future projects should carefully weigh the benefits of a new framework against the potential disruptions and resource requirements.

Strategies for Improvement

- **Thorough Project Planning:** Enhance project planning by conducting a comprehensive risk assessment and outlining contingency plans for potential roadblocks.
- **Continuous Integration and Testing:** Implement robust continuous integration practices to identify and rectify issues early in the development cycle, reducing the impact on milestones.
- **Skill Diversification:** Encourage skill diversification among the development team to ensure adaptability to different frameworks, facilitating smoother transitions if framework changes become necessary.
- **Frameworks Evaluation:** Regularly assess emerging frameworks to stay informed about advancements in the mobile development landscape, enabling informed decisions for future projects.

Conclusion

The mobile application development process using React Native and Expo was successful, albeit with challenges in meeting deadlines. The chosen framework played a pivotal role in the project's efficiency. While alternative frameworks like Flutter offer intriguing possibilities, careful consideration is crucial due to potential challenges in migration and team adaptation. Implementing the outlined strategies for improvement can enhance the overall success of future development projects.

Student Details

Name: Oshan Madhushika

Student ID: iAMDT.2797

Course: UX Design