

TASK:

Mobile Development Task - Figma Design Integration

Task Title: Wallet App - Design Integration and API Implementation

Task Description:

Objective:

Integrate the provided Figma design into a functional mobile application for Wallet. Implement key functionalities, including user authentication, wallet balance retrieval, and transaction history listing using the provided API.

Design Assets:

Figma Design Link:

[https://www.figma.com/file/DIGlYrte8YFTrSupdw8QB7/Mobile-UI-Kit%3A-Fintech-\(Community\)?type=design&node-id=0-1&mode=design&t=qWasG5FpJzIY91OH-0](https://www.figma.com/file/DIGlYrte8YFTrSupdw8QB7/Mobile-UI-Kit%3A-Fintech-(Community)?type=design&node-id=0-1&mode=design&t=qWasG5FpJzIY91OH-0)

API: <https://app.beeceptor.com/mock-server/crypto-wallet-server>

Screens:

- Splash Screen One
- Splash Screen Two
- Home Screen

Development Steps:

- Design Integration:

Utilize the Figma design for the specified screens.

Implement responsive layouts for different screen sizes.

- Splash Screens:

Implement Splash Screen One and Splash Screen Two as per the design.

Display relevant information or branding during the splash screens.

- Login and Register Screens:

Design and integrate the Login and Register screens based on typical mobile app conventions.

Implement validation for user input:

Integrate the `/api/v1/register` and `/api/v1/login` endpoints for user registration and login, respectively.

Home Screen:

- Design and integrate the Home screen as per the Figma design:

Display user details, wallet balance, and transaction history.
Implement the balance retrieval functionality using the `/api/v1/balance` endpoint.
Display a list of transactions using the `/api/v1/transactions` endpoint.

API Integration:

Utilize the provided API base URL: `https://crypto-wallet-server.mock.beeceptor.com`
Integrate the login and register endpoints to authenticate users.
Fetch and display the user's wallet balance using the `/api/v1/balance` endpoint.
Retrieve and display the user's transaction history using the `/api/v1/transactions` endpoint.
Error Handling and Loading States:

Implement proper error handling for API calls and user interactions.
Display loading states during API calls to provide feedback to users.
Submission Guidelines:

Submission:

- Commit your code to a repository on your GitHub and push it to the repository.
- Include a README.md file with detailed instructions on how to setup and run your application.
- Share the project link with us via email.
- Include your GitHub username and any additional instructions or notes related to your submission.

Evaluation:

- Close alignment with Figma design.

Functionality:

- Correct and complete feature implementation.
- Well-structured and organized code
- Error Handling and Loading States:
- Effective for user experience.
- Adherence to specified deadlines.

DEADLINE: 3 days