Project ID: 29

Project Title

Design And Prototype Of A Fintech App With Payto Integration Via Apis For Real-Time Bank Payments In Australia

Client Name

Muhammad Raheel Raza

Group Capacity

6 groups

Project Background

The rapid evolution of financial technology (fintech) has transformed how individuals manage money, make payments, and interact with banking systems. One notable innovation is Turkey's Papara, a digital wallet platform that enables users to transfer money, pay bills, earn cashback, and make online purchases with minimal reliance on traditional banks.
In the Australian context, regulatory and technological advancements such as PayTo and the New Payments Platform (NPP) have paved the way for similar real-time, account-to-account payment solutions. This project aims to explore and replicate the core features of Papara within the Australian financial ecosystem, focusing on usability, regulatory awareness, and integration with mock or sandbox payment infrastructure.
The aims of the project are:
1. Analyze Papara's key features and the Australian payment landscape (PayTo, NPP, BECS).
2. Design a mobile/web fintech application offering core wallet functions such as fund transfers, bill payments, and balance management.
3. Simulate PayTo-style automatic payment authorization using mock APIs or sandbox environments.
4. Develop a secure backend and frontend using appropriate technology stacks.
5. Incorporate simulated KYC (Know Your Customer) and transaction logging features.
6. Address regulatory and security considerations relevant to fintech development in Australia.
7. Deliver a working prototype, technical documentation, and a project presentation showcasing the app's functionality and compliance logic.

Project Scope

This project involves designing and prototyping a Papara-inspired digital wallet app tailored for the Australian payment ecosystem. It includes user registration with simulated KYC, wallet management, peer-to-peer transfers, and bill payments using mock or sandbox APIs. The app will simulate PayTo-style automatic payment authorization. Security and compliance features will be incorporated at a basic level. Real bank integrations and actual

money transfers are out of scope. The final deliverable is a working prototype with supporting documentation.

Project Requirements

1. User Registration and Authentication
2. Wallet management: Each user has a digital wallet showing current balance and transaction history.
3. Pay-to-Simulation: Use mock or sandbox APIs to simulate bank-to-wallet interactions.
4. UI Interface: Clean, intuitive UI/UX design suitable for mobile or web platforms.
5. System Architecture and Integration: Backend API to handle business logic, user management, and transactions. Frontend client app interacting with backend and mock banking APIs. Use of mock or sandbox APIs to replicate banking/payment network behaviors (PayTo, NPP).

Required Skills

For frontend, React Native or Flutter (for mobile), or React for web. For backend, Node.js / Django / Flask. For database, Firebase / PostgreSQL / MongoDB. For Auth, Firebase Auth, JWT. For Mock Payto API, use simulated PayTo workflows or mock payment gateway. For devops, GitHub, Docker, Vercel/Render for deployment.

Expected Outcomes

A functional prototype of a Papara-inspired digital wallet app tailored to the Australian payment system context.

Demonstrated simulation of real-time payment authorization (PayTo-style) via mock or sandbox APIs.

Source Code: Complete and well-commented codebase for frontend and backend components, hosted in a version-controlled repository (e.g., GitHub).

Technical Documentation: Detailed documents covering system architecture, API design, data flow diagrams, and security considerations.

User Guide: Instructions for installing, running, and using the application, including screenshots or demo walkthroughs.

Project Report: Comprehensive report summarizing objectives, development process, challenges faced, and lessons learned.

Presentation/Demo: Final presentation and live demo showcasing the application's features and project outcomes.

Disciplines

Web Application Development;Mobile Application Development;Software Development;Computer Science and Algorithms;Security/Cyber Security;

Other Resources

None