

WELCOME TO

PIGGY BANK

EFFORTLESS BANKING APPLICATION

**BY- BATCH 1
COHORT 401**

Maanvitha Allampalli
Ashish Asnani
Ritika Kar

PROBLEM STATEMENT

A background image showing several stacks of silver coins, such as quarters or一角 coins, arranged in a grid pattern across the slide.

As banking needs evolve, traditional banking websites often remain complex, with cluttered interfaces and multi-step processes that can overwhelm users. Many people, especially those with limited digital literacy, find it challenging to navigate these systems to perform simple tasks like transferring money or checking their balance. This complexity has created a gap where users, frustrated by difficult-to-use banking portals, turn to alternative payment apps like PhonePe and Google Pay. These apps offer simplified, one-click transactions and intuitive interfaces, making financial transactions accessible to a broader audience, including those with minimal technical skills.

To address these challenges, our project aims to bridge this accessibility gap by building a streamlined banking application that combines the security of traditional banking with the simplicity of modern payment apps, ensuring an easy, user-friendly experience for everyone.

LOGO



Our logo embodies the image of a piggy bank, symbolizing simplicity and ease in managing finances. It reflects our project's core goal: to create an intuitive, user-friendly banking platform where transactions and saving money are as effortless as using a piggy bank. This visual ties into our mission to make financial management accessible for everyone, reinforcing the values of simplicity and convenience in every transaction.

CORE PROJECT FEATURES

The table below provides a summary of the PiggyBank application's core features, offering a snapshot of implemented functionalities. This overview highlights the application's essential components, from secure transactions to intuitive user interfaces, all designed to streamline the user experience and simplify financial management. The table distinguishes completed features list, allowing a clear view of the project's progress and next steps.

FEATURE	DESCRIPTION
User Authentication	Secure signup and login
Account Management	View details and create account
Transactions	Deposit, Withdraw, Transfer
Account Deletion	Delete accounts with cascade
Transaction History	Log transaction records

SYSTEM SETUP

Backend Setup:

- Clone the repository and navigate to the backend directory.
- Configure the application.properties file with the database connection details and server configurations.
- Build the backend using Maven with the command mvn clean install.
- Run the backend application with mvn spring-boot:run to start the server.

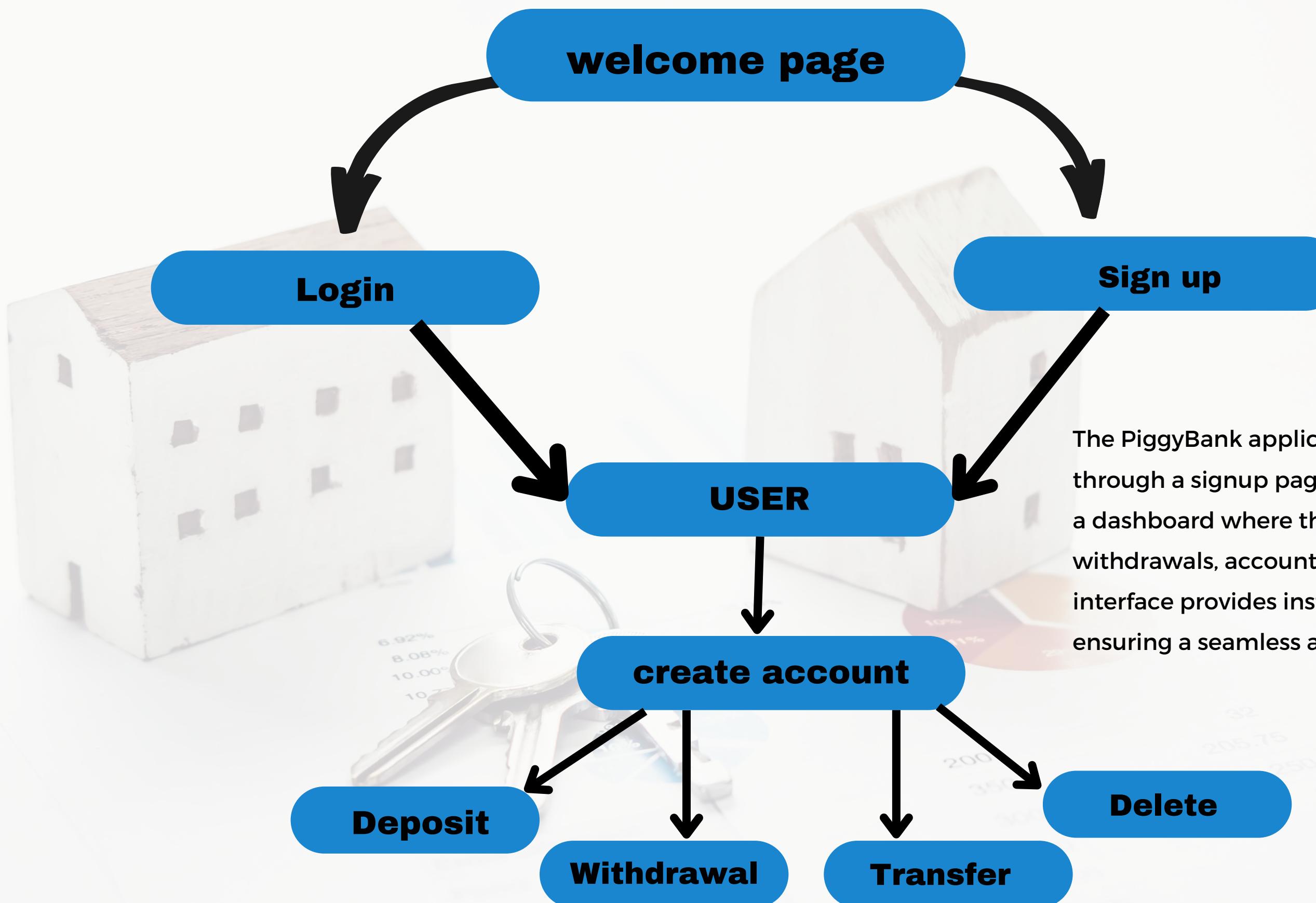
Prerequisites:

- Ensure Java JDK, Node.js, and your database (MySQL/PostgreSQL) are installed and compatible with the project requirements.

Frontend Setup:

- Navigate to the frontend directory.
- Install dependencies using npm install.
- Start the Angular development server with ng serve, making the frontend available at <http://localhost:4200>.

APPLICATION FLOW



USER FRIENDLY INTERFACE

Notifications:

- The application sends notifications similar to alert messages. Upon login, users receive a message with their name and relevant transaction details, such as deposits, withdrawals, or transfers. These notifications include the amount specified, ensuring users stay informed about their account activities in real-time.

Dashboard:

- After logging in, users are directed to a dashboard that provides easy access to all the functionalities. The dashboard is designed to be intuitive and user-friendly, presenting all available features in a clear and organized way, without overwhelming the user.

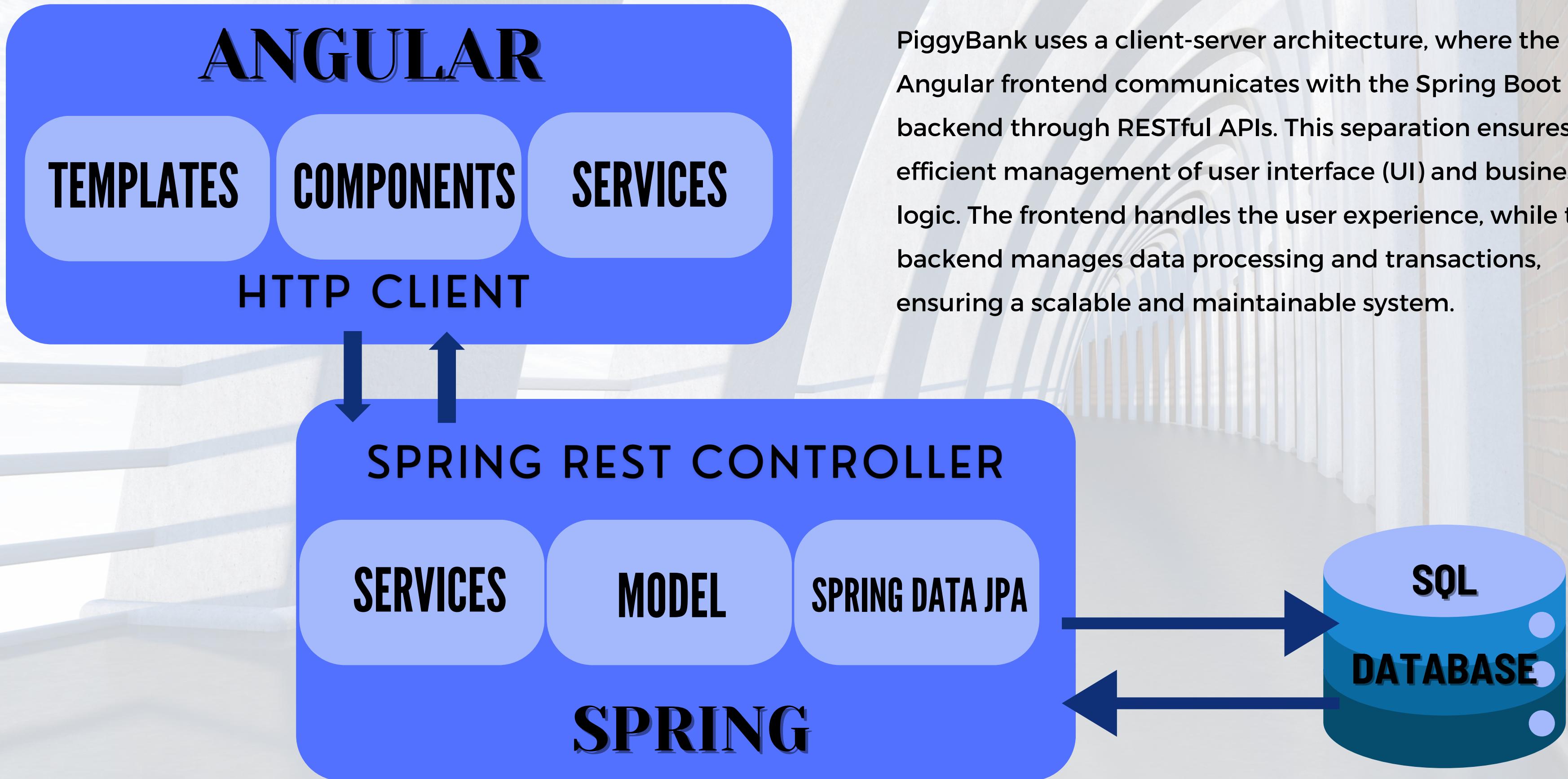
Account Balance:

- The total account balance is prominently displayed on the top of the dashboard, providing users with a quick overview of their available funds. This balance updates automatically after every transaction, such as transfers, deposits, or withdrawals.

Account Number Verification:

- When transferring funds to another person's account, users enter the account number. To ensure the transaction is accurate, the application features a verification function that fetches and displays the account holder's name. This allows users to cross-verify the recipient's name before completing the transaction, similar to how UPI transactions work, ensuring added security and reducing errors.

ARCHITECTURE OVERVIEW



API ENDPOINTS

This table lists the key API endpoints required for customer and account management in the PiggyBank application, providing details on their methods and functionalities. These endpoints enable seamless interaction between the front-end and back-end, supporting operations such as user registration, login, account management, transactions, and investment options. Each endpoint is designed to ensure secure, efficient handling of user data and financial operations.

METHOD	ENDPOINT	DESCRIPTION
POST	/api/bank/customers	Create a new customer.
GET	/api/bank/accounts/{customerId}	Retrieve account by customer ID.
POST	/api/bank/deposits	Deposit funds into an account.
POST	/api/bank/login	Login a customer; returns details.
DELETE	/api/bank/customers/{id}	Delete a customer by ID.

FUTURE SCOPE AND IMPLEMENTATIONS

Building on the foundation of simplified CRUD operations, the future plans for PiggyBank include expanding into comprehensive financial management options:

- 1. Investment Options:** Integrate diverse investment plans (mutual funds, savings plans) to provide customers with a range of financial growth opportunities.
- 2. Interactive Information Sharing:** Deliver investment insights and financial planning tips via engaging formats like videos, interactive tutorials, and personalized content.
- 3. Stock Market Integration:** Enable users to buy, sell, and manage stock portfolios within the app, providing a seamless investment experience.
- 4. Smart Recommendations:** Implement AI-driven recommendations for stocks and investments based on market trends and user profiles, empowering customers with informed financial choices.
- 5. Integrated Chatbot Assistance:** Add a chatbot to guide users through the application, providing personalized assistance, answering queries, and offering real-time support for smoother navigation and user experience.
- 6. Enhanced User Interaction:** Develop features that improve ease of use, with a focus on making complex investment tools accessible and engaging for users of all backgrounds.

A stack of various credit cards, including American Express, Visa, MasterCard, and Discover, fanned out in the background.

Thank You