

## RevPay – Digital Wallet System

- A secure Java-based digital payment application.
- Supports both personal users and business users.
- Implements wallet, cards, loans, invoices and analytics.
- Designed using layered architecture principles.

*Real-Time Example: Similar to Paytm or Phonepe wallet system.*

# Project Introduction

- RevPay allows users to manage digital money securely.
- Users can send, receive and request payments.
- Business users can generate invoices and analyze revenue.
- System is console-based but follows real-world workflows.

*Real-Time Example: User logs in and sends money to another user instantly.*

# Project Objectives

- Provide secure authentication and authorization.
- Enable wallet-based transactions instead of cash.
- Support business-level financial operations.
- Maintain clean, modular and testable code.

*Real-Time Example: Replacing cash payments with digital wallet transfers.*

## Technologies Used

- Java 17 for application logic.
- JDBC for database communication.
- MySQL for structured data storage.
- Log4j2 for logging and monitoring.
- JUnit for unit testing.
- BCrypt and AES for security.

*Real-Time Example: BCrypt protects passwords even if database is compromised.*

# System Architecture

- UI Layer handles user interaction.
- Service Layer contains business rules.
- DAO Layer performs database operations.
- Database stores persistent data securely.

*Real-Time Example: If business rules change, only Service Layer is updated.*

# Package Structure

- App – application entry and menu flow.
- Services – core business logic.
- DAO – database access logic.
- MODEL – entity representation.
- Security – encryption and hashing.
- DBconfig – centralized DB connection.

*Real-Time Example: Clear separation improves maintainability.*

## Database & ER Design

- APP\_USERS acts as the central table.
- Wallet is linked one-to-one with users.
- Transactions and requests link sender and receiver.
- Loans and invoices support business operations.

*Real-Time Example: One user can have many transactions but one wallet.*

## Core Features

- User registration and secure login.
- Wallet deposit and withdrawal.
- Direct money transfer and requests.
- Card management with encryption.
- Loan and invoice management.

*Real-Time Example: User sends money to friend using wallet balance.*



# Security Implementation

- Passwords hashed using Bcrypt.
- Cards encrypted using AES algorithm.
- Transaction PIN for sensitive actions.
- Account locked after multiple failures.

*Real-Time Example: Even admin cannot see original password or card number.*

# Logging & Testing

- Log4j2 used for application logging.
- INFO for normal flow, WARN for unusual cases.
- ERROR for failures and exceptions.
- JUnit tests validate service behavior.

*Real-Time Example: Logs help debug production issues quickly.*

## Advantages of RevPay

- Secure and reliable transactions
- Scalable and modular design
- Easy to extend to web or mobile
- Follows industry best practices

*Real-Time Example: Can be extended into a full fintech product.*

## Conclusion

- RevPay demonstrates real-world payment workflows.
- Strong focus on security and architecture.
- Suitable for academic and real-world use.
- Foundation for future enhancements.

*Real-Time Example: Can evolve into full-stack payment application.*