Exchange Store Application

- Student: Kuzey Arda Bulut
- Date: 23.09.2025

Purpose

- Simulates daily operations of a currency exchange office
- Helps cashiers and managers:
- - Perform transactions
- - Check/update exchange rates
- - Issue receipts & daily reports
- No internet dependency → works offline

Scope & Features

- Currency exchange (full & partial)
- Update exchange rates dynamically
- Manage reserves (deposit/withdraw funds)
- Set minimum reserve levels
- Generate receipts for each transaction
- End-of-day profit & balance reports

System Structure

- main.cpp → Entry point, controls program flow
- currency_manager.cpp/hpp → Core logic for managing currencies, rates, and reserves
- utils.cpp/hpp → Input validation, formatting, and helper functions
- Heap objects used for secure memory management

Execution Flow

- Program starts → Main menu
- User selects an action (exchange, show rates, manage reserves)
- System validates input (via utils)
- CurrencyManager updates data structures
- Receipt generated → transaction recorded
- At exit → end-of-day report printed

Design & Requirements

- Use-Case: Cashier exchanges money → system updates reserves → receipt generated
- CRC Cards:
- - CurrencyManager: handles rates/reserves
- - Utils: handles validation, input/output formatting
- Constraints:
- - Language: C++
- - Modular, portable, offline
- Non-functional goals: Reliability, error handling, maintainability

Conclusion

- A complete simulation of an exchange office
- Modular design: easy to extend with new currencies or features
- Clear educational value:
- - Combines business logic + programming practices
- - Demonstrates input validation, memory safety, and reporting