Multi-Threaded Banking System in C++

Project Overview:
This is a multi-threaded banking system written in C++ that demonstrates advanced concepts like object-oriented programming, multi-threading, design patterns, and memory management. The system simulates real-world banking operations such as deposit, withdrawal, transfer, and balance inquiry with concurrent users.

Features

- Multi-threaded transactions with thread-safe operations.
- Design Patterns: Singleton and Factory patterns.
- Memory Management: Smart pointers to manage dynamic memory.
- Performance Benchmarks: Measure transaction times with std::chrono.

Performance Benchmarks

- Deposit: The deposit operation took an average of 0.004 seconds across 1000 operations.
- Transfer: Transferring funds between accounts took an average of 0.006 seconds under heavy load.

Setup

1. Install required dependencies (e.g., C++14, cpprestsdk).
2. Compile the program using the provided Makefile.
3. Run the program with the ./bank command.

API Documentation (if extended to REST/gRPC)

/deposit
- POST /deposit

- Request: {"amount": 100}
- Response: { "status": "success", "balance": 1500 }

/withdraw
- POST /withdraw
- Request: {"amount": 50}
- Response: { "status": "success", "balance": 1450 }