Inter-Thread Communication

The goal of this project is to implement inter-thread communication in a simulated banking system using C++ programming language. The project involves the use of multiple C++ files to create a fully functional banking system with additional features.

The banking system simulation includes several functionalities such as account creation, deposit, withdrawal, different types of accounts, currency exchanger, draft, etc. The project utilizes inter-thread communication to ensure that multiple threads can access the shared resources of the system without causing conflicts or errors. The project uses a mutex object to protect the balance from concurrent access by multiple threads, and a condition_variable object to signal and wait on changes to the balance. The project also includes an exception handling mechanism to handle insufficient funds and overdraft fees.

Overall, the project serves as a practical and efficient solution for implementing inter-thread communication in a banking system. The project team hopes that this project will serve as a valuable resource for developers and programmers looking to implement inter-thread communication in their own projects.