

P2P Banking System

Author: Matěj Červenka C4c 7.2.2025

School Project: Střední průmyslová škola elektrotechnická, Praha 2, Ječná 30

Contact: matej.cervenka1106@gmail.com

Introduction

The **P2P Banking System** is a distributed banking application where multiple bank nodes communicate over a network using predefined commands. Each node represents an independent bank, capable of managing customer accounts, processing transactions, and interacting with other banks in the system.

Key Features

- Account creation, deposit, withdrawal, and balance inquiries.
- Distributed banking with interbank communication via TCP/IP.
- Unique bank and account identifiers.
- Supports proxy-based command forwarding for interbank transactions.

Database Structure

The system uses a **MySQL** database with the following tables:

account

Column	Type	Description
id	INT (PK, AUTO_INCREMENT)	Unique account ID
account_number	INT (UNIQUE)	Account number (10000–99999)
balance	BIGINT	Account balance (in smallest currency unit)

Technologies Used

- **Backend:** Java
 - **Networking:** TCP/IP (Socket Programming)
 - **Database:** MySQL
 - **Build Tool:** Maven
 - **Deployment:** Command Prompt
 - **Application** PuTTY for remote operating
-

How to Run and Operate the Application

1. Prerequisites

Ensure you have the following installed:

- **Java 17+**
- **Maven**
- **MySQL**
- **PuTTY (for network testing)**

2. Installation and Deployment

- **Clone the repository**
- **Find the /resources directory**
- **Copy the application.properties into the /target directory**
- **Open the /target directory in command prompt**
- **Enter** `java -jar bank.jar`

3. Available Commands

- There is number of commands you can use to integrate with the bank server
- **BC (Bank Code)** Returns the IP address of the bank as its unique code.
- **AC (Account Create)** Creates a new bank account with a unique account number within the specified bank.
- **AD (Account Deposit)** Deposits a specified amount of money into a given account.
- **AW (Account Withdrawal)** Withdraws a specified amount from a given account, provided sufficient funds are available.
- **AB (Account Balance)** Returns the current balance of a specified account.
- **AR (Account Remove)** Deletes a specified account, only if its balance is zero.
- **AS (Account Show)** Shows all the accounts existing in the bank.

- **BA (Bank Amount)** Returns the total funds across all accounts in the bank.
- **BN (Bank Number)** Returns the total number of clients (accounts) in the bank.

Name	Code	Call	Response on success
Bank code	BC	BC	BC [ip]
Account create	AC	AC	AC [account] / [ip]
Account deposit	AD	AD [account] / [ip] [balance]	AD [account] / [ip] +[balance]
Account withdrawal	AW	AW [account] / [ip] [balance]	AW [account] / [ip] -[balance]
Account balance	AB	AB [account] / [ip]	AB [balance]
Account remove	AR	AR [account] / [ip]	AR [account] / [ip] Removed
Account show	AS	AS	AS [account] / [ip]
Bank (total) amount	BA	BA	BA [balance]
Bank number of clients	BN	BN	BN [number]

4. Communication between Banks

- Three commands are available for between banks communication.
- If you enter either command **AD** **AW** **AB** and as parameters **[account]** / **[ip]** enter **ip** of another bank in current LAN and **account number** existing in the entered bank, the command will be send to the bank application running on the **ip** entered, processes the command and returns appropriate answer.

- **AD (Account Deposit)** Deposits a specified amount of money into a given account.
- **AW (Account Withdrawal)** Withdraws a specified amount from a given account, provided sufficient funds are available.
- **AB (Account Balance)** Returns the current balance of a specified account.

Name	Code	Call	Response on success
Account deposit	AD	AD [account] / [ip] [balance]	AD [account] / [ip] +[balance]
Account withdrawal	AW	AW [account] / [ip] [balance]	AW [account] / [ip] -[balance]
Account balance	AB	AB [account] / [ip]	AB [balance]

Reused code

- Active Record Database Pattern - [ActiveRecordDatabase - Entities Directory](#)

Links

[Java Socket](#)

[Maven Dependencies](#)

[ExecutorService](#)

[ThreadPool](#)

[Java Networking](#)

[Properties](#)

[ChatGPT](#)