**Структура проекта**

src

├── main

│ ├── java

│ │ └── com

│ │ └── example

│ │ └── banking

│ │ ├── config

│ │ │ └── SecurityConfig.java

│ │ ├── controller

│ │ │ ├── BankController.java

│ │ │ ├── CurrencyExchangeController.java

│ │ │ ├── CashServiceController.java

│ │ │ └── ...

│ │ ├── dto

│ │ │ ├── BankDto.java

│ │ │ ├── CurrencyExchangeDto.java

│ │ │ ├── CashServiceDto.java

│ │ │ └── ...

│ │ ├── enums

│ │ │ ├── AccountType.java

│ │ │ ├── BankServiceType.java

│ │ │ └── ...

│ │ ├── exception

│ │ │ ├── CustomExceptionHandler.java

│ │ │ └── ...

│ │ ├── mapper

│ │ │ ├── BankMapper.java

│ │ │ ├── CurrencyExchangeMapper.java

│ │ │ ├── CashServiceMapper.java

│ │ │ └── ...

│ │ ├── model

│ │ │ ├── entity

│ │ │ │ ├── Bank.java

│ │ │ │ ├── CurrencyExchange.java

│ │ │ │ ├── CashService.java

│ │ │ │ └── ...

│ │ │ ├── repository

│ │ │ │ ├── BankRepository.java

│ │ │ │ ├── CurrencyExchangeRepository.java

│ │ │ │ ├── CashServiceRepository.java

│ │ │ │ └── ...

│ │ │ └── ...

│ │ ├── service

│ │ │ ├── BankService.java

│ │ │ ├── CurrencyExchangeService.java

│ │ │ ├── CashServiceService.java

│ │ │ └── ...

│ │ ├── BankExchangeApplication.java

│ │ └── ...

│ └── resources

│ ├── application.properties

│ ├── db

│ │ └── changelog

│ │ └── changes

│ │ └── v0.0.1-master.xml

│ └── ...

└── test

└── java

└── com

└── example

└── banking

└── ...

Объяснения:

**config:** Здесь размещаются конфигурационные классы, такие как SecurityConfig.

**controller:** Контроллеры, обрабатывающие веб-запросы. Каждый контроллер отвечает за определенную сущность или функциональность.

**dto:** Классы DTO, отражающие структуру данных, передаваемых между клиентом и сервером.

**enums:** Перечисления, такие как AccountType, BankServiceType, и т.д.

**exception:** Классы обработки исключений, например, CustomExceptionHandler.

**mapper:** Классы мапперов для преобразования между сущностями и DTO.

**model:** Пакет для хранения сущностей, таких как Bank, CurrencyExchange, и др.

**repository:** Интерфейсы репозиториев для работы с базой данных.

**service:** Сервисы, реализующие бизнес-логику.

**resources:** Ресурсы приложения, такие как файлы конфигурации (application.properties) и миграции для Liquibase.

**test:** Директория для юнит-тестов.