

Федеральное государственное автономное
образовательное учреждение высшего образования

Университет ИТМО

Дисциплина: Рефакторинг баз данных и приложений

Дизайн документ проекта

Работу выполнили студенты группы Р34101:

Антипин А.А.

Кондрашов К.Ю.

Преподаватель:

Логинов Иван Павлович

2022 г.

Санкт-Петербург

Описание проекта

Мы хотели попробовать реализовать модель клиринговой системы - безналичные расчёты между банками на основе взаимного зачёта, исходя из условий баланса платежей.

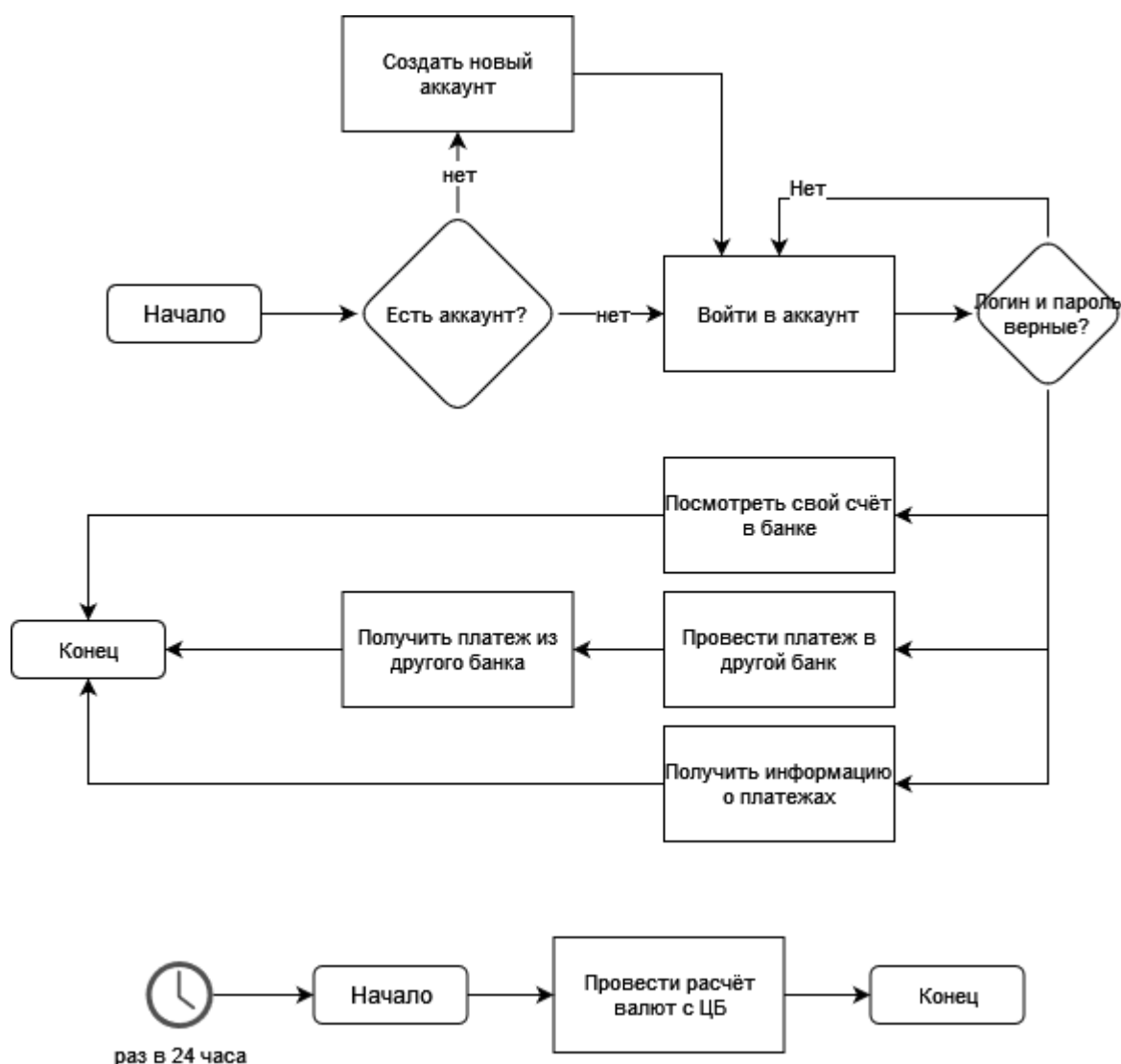
Пользовательские сценарии

Для банков:

Отправка/получение платежей

Получение информации от Центрального банка информации о выплатах другим банкам

Пользовательский сценарий

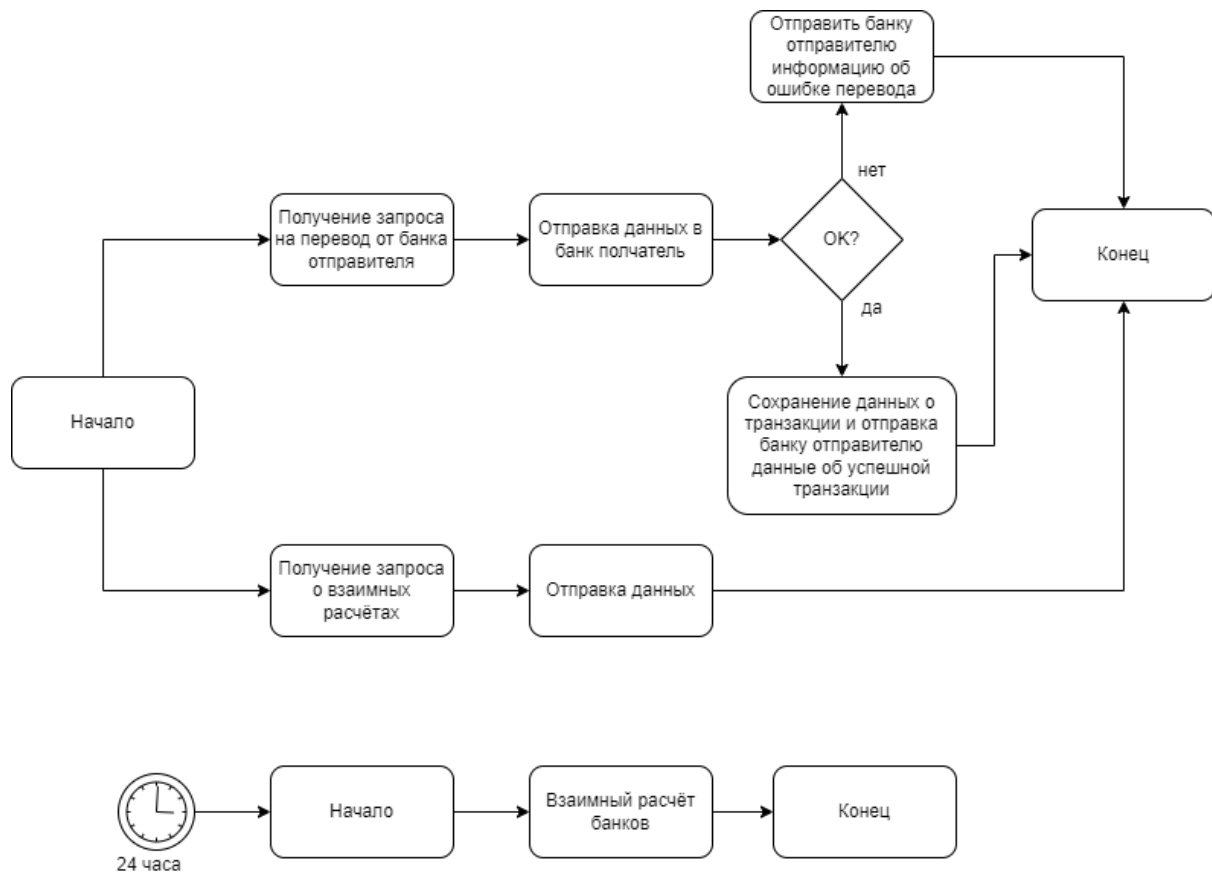


По данной диаграмме пользовательских сценариев можно выделить 3 основных функциональных группы:

1. Регистрация и авторизация пользователей
2. Перевод/получение платежей через ЦБ
3. Информационная группа: Посмотреть свой счёт (внутри банка) и получить информацию о платежах (через ЦБ)

*не относиться к пользовательскому сценарию, но раз в день банки будут проводить синхронизацию с ЦБ чтобы установить денежную разницу между банковскими переводами.

Для ЦБ:



По данной диаграмме пользовательских сценариев можно выделить 3 основных функциональных группы:

1. Провод платежей между пользователями
2. Статистика платежей
3. Расчет взаимных зачетов для банков

Этапы разработки (дерево задач)

Сервис клиринга:

- Первый спринт
 - Postgres
 - Создать таблицы базы данных
 - Http4s Application
 - Сконфигурировать проект

- Создать сущности базы данных
 - Создать crud-репозитории и сервисы к ним
- Второй спринт
 - Http4s Application
 - Создать сервисы
 - Создать контроллеры
- Третий спринт
 - Http4s Application
 - Связать приложение с банками
 - Протестировать полный функционал

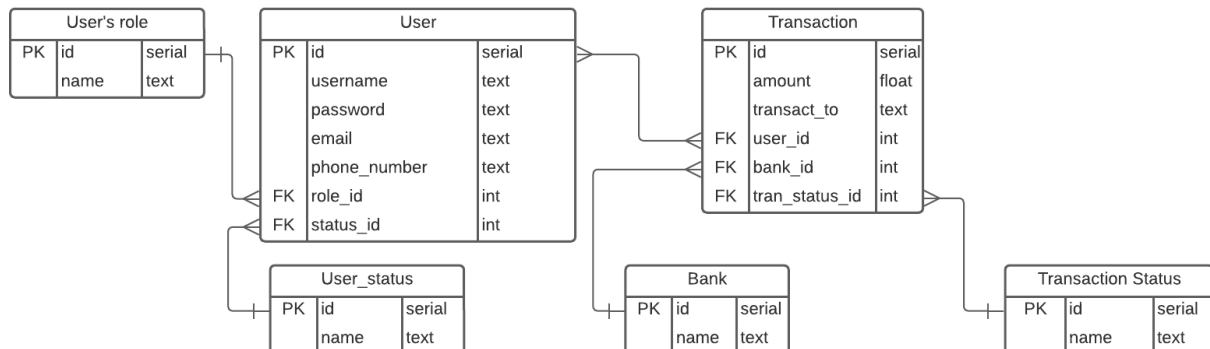
Сервис банка:

- Первый спринт
 - Postgres
 - Создать таблицы базы данных
 - Spring Boot Application
 - Создать сущности базы данных
 - Объединить бэкенд-приложение с базой данных
 - Создать crud-репозитории и сервисы к ним
 - Создать контроллеры
- Второй спринт
 - Spring Boot Application
 - Добавить Spring Security
 - Расширить функционал репозиторий/сервисов/контроллеров
 - Создать контроллер для обработки ошибок
- Третий спринт
 - Spring Boot Application
 - Связать приложение с приложением ЦБ

- Добавить таймер для синхронизации (установления разницы) транзакций с ЦБ

Архитектурные решения

ER модель базы данных для сервиса банка:



ER модель базы данных для сервиса ЦБ:

