

Дипломный проект на тему: «Bank (Financial processing system)»

Слушатели:

Прощаев Сергей Федорович

Васильев Сергей Дмитриевич

Губинский Олег Алексеевич

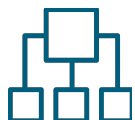
Ночкин Евгений Игоревич

Филин Константин Георгиевич

Актуальность темы и её проблематика:



Financial processing (FP) является составной частью современных систем



Навык проектирования архитектуры FP является важным навыком разработчика



В дипломном проекте рассмотрена реализация FP в банковской системе



При реализации архитектуры FP используется функционал, изученный в курсе



В качестве СУБД для реализации FB используется PostgreSQL

Используемые технологии



Spring Framework (Spring Boot, Spring Data, Spring Security, Spring AOP)



PostgreSQL



Apache Tomcat



Apache POI



JavaServer Pages










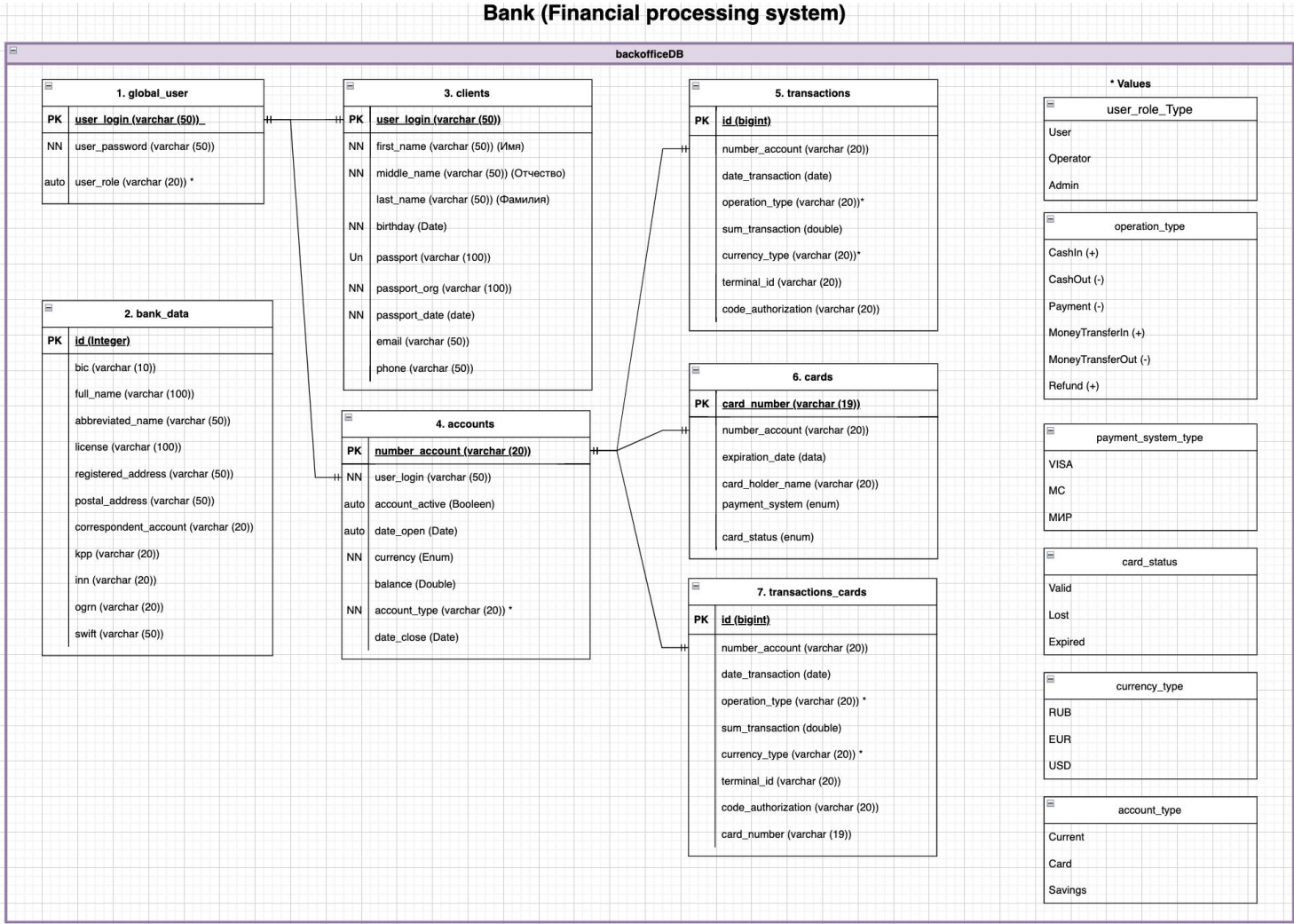
Архитектурная схема:



Структура базы данных

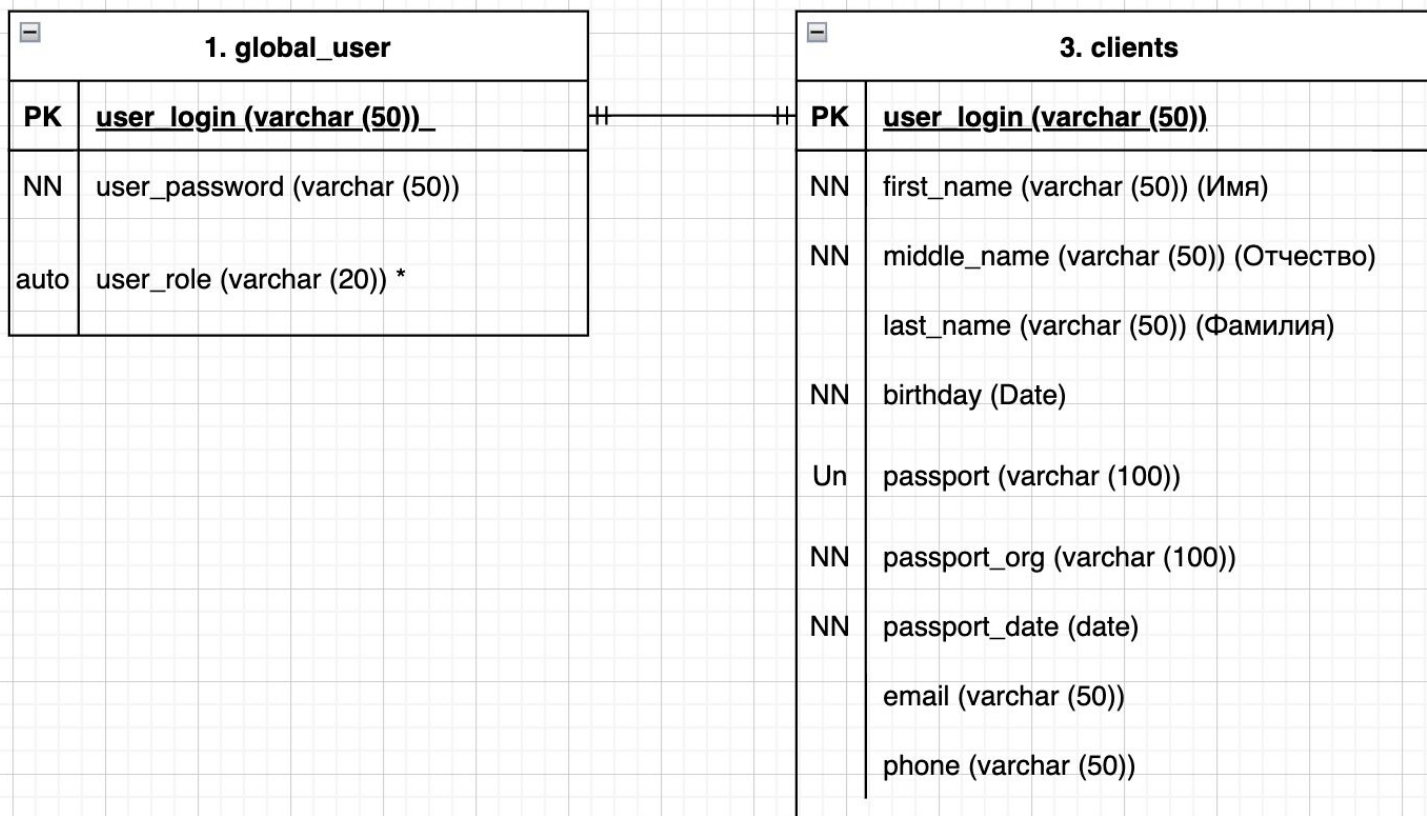
Табличные пространства СУБД:

-  Регистрационные данные:
таблица global_user
-  Персональные данные:
таблица clients
-  Счета клиентов:
таблица accounts
-  Движение по счетам:
таблица transactions
-  Карточные продукты:
таблица cards
-  Транзакции по картам:
таблица transactions_cards
-  Реквизиты банка-эмитента:
таблица bank_data



Филин Константин | Регистрация пользователей

Реализованная архитектура:



Используемые технологии:

Для разделения функционала по ролям используется фреймворк Spring Security

Хранение зарегистрированных пользователей в системе осуществляется в СУБД PostgreSQL в таблице global user

Доступ из приложения к таблицам PostgreSQL осуществляется через использование Spring Data

Для вывода данных (UI) используются JSP-страницы

Сквозное логирование осуществляется через Spring AOP и slf4j - logback

Филин Константин | Регистрация пользователей

Разработанный функционал системы:



Регистрация пользователей в системе с ролями user, admin



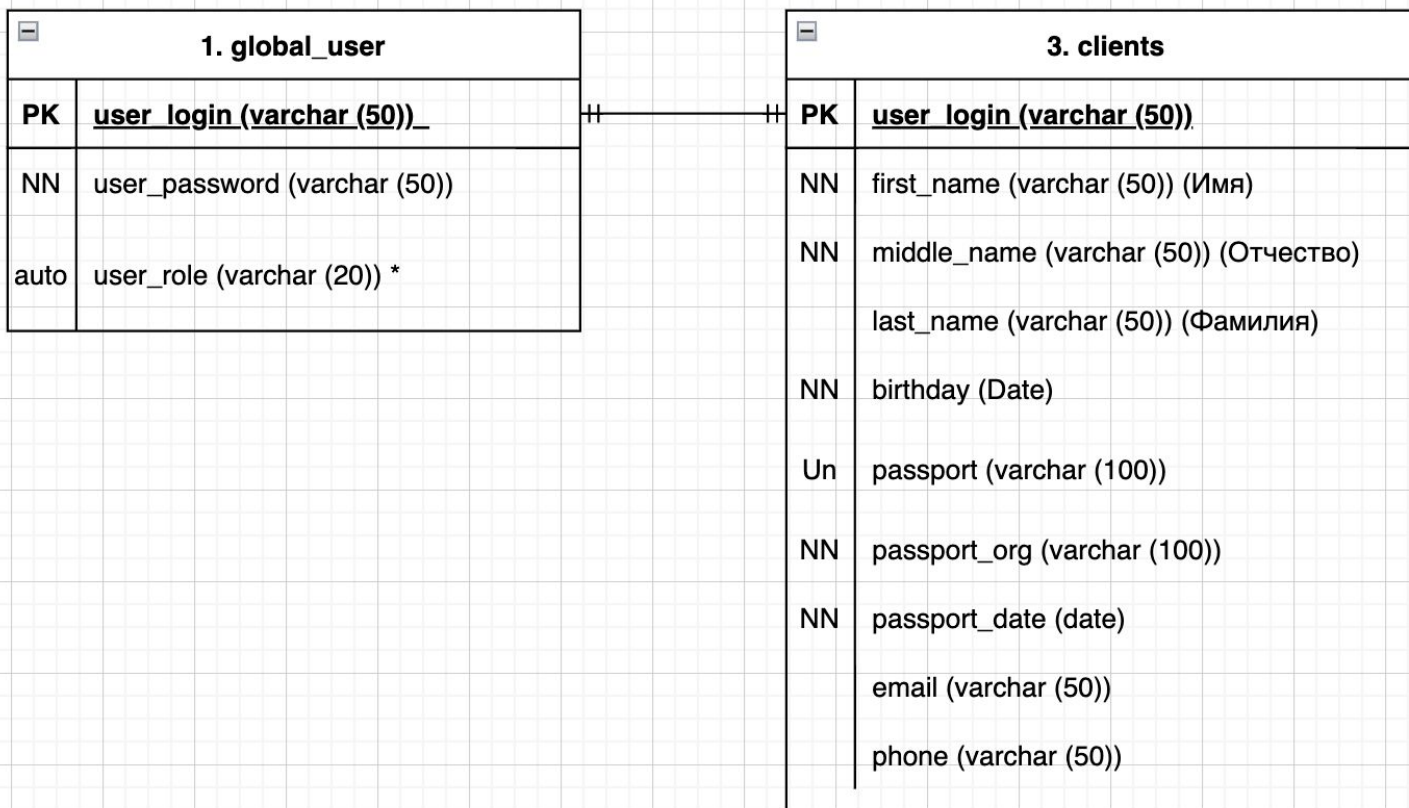
Вывод информации по ранее зарегистрированным пользователям



Доступ к JSP-страницам всего приложения в зависимости от роли

Губинский Олег | Регистрация персональных данных

Реализованная архитектура:



Используемые технологии:

Для разделения функционала по ролям используется фреймворк Spring Security

Хранение зарегистрированных пользователей в системе осуществляется в СУБД PostgreSQL в таблице clients

Доступ из приложения к таблицам PostgreSQL осуществляется через использование Spring Data

Для вывода данных (UI) используются JSP-страницы

Сквозное логирование осуществляется через Spring AOP и slf4j - logback

Губинский Олег | Регистрация персональных данных

Разработанный функционал системы:



Регистрация нового клиента, ввод персональных данных



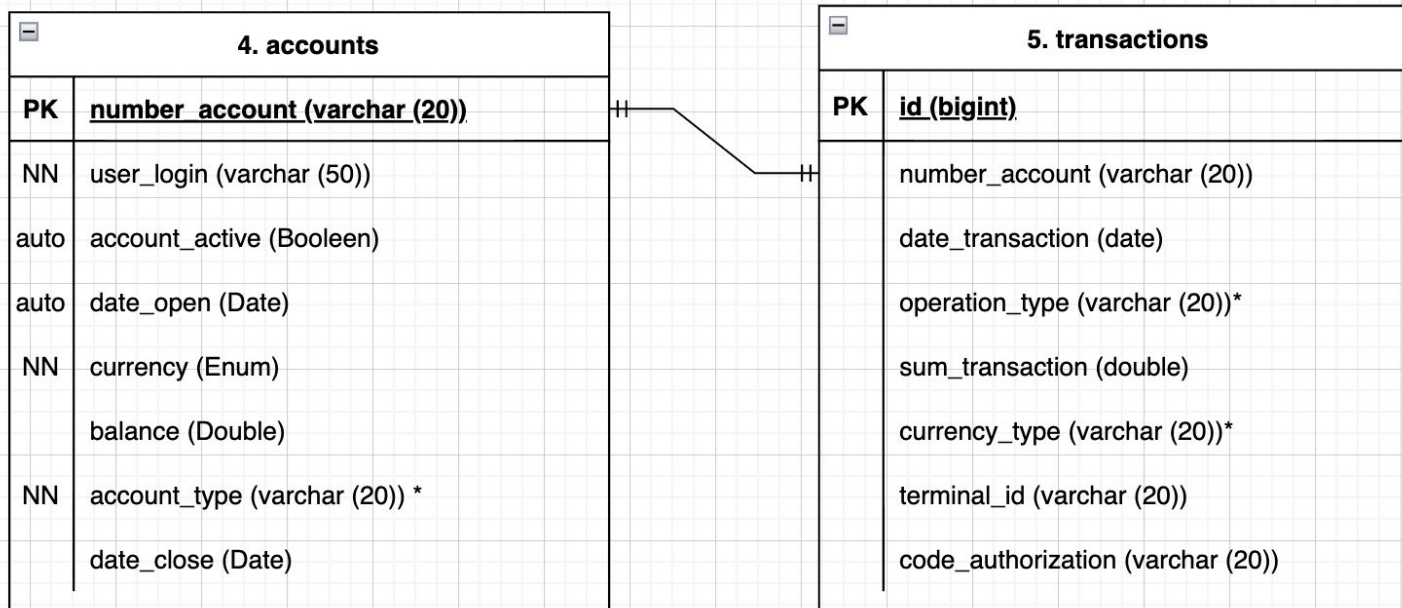
Операции поиска, изменения, просмотра информации по клиенту



Определение прав клиента для доступа к функционалу из «Домашнего банка»

Васильев Сергей | Открытие счета клиента

Реализованная архитектура:



Используемые технологии:

Для разделения функционала по ролям используется фреймворк Spring Security

Хранение счетов клиентов в системе осуществляется в СУБД PostgreSQL в таблице accounts, движение по счетам в таблице transactions

Доступ из приложения к таблицам PostgreSQL осуществляется через использование Spring Data

Для вывода данных (UI) используются JSP-страницы

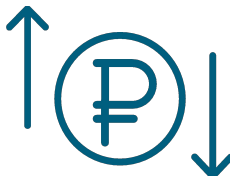
Сквозное логирование осуществляется через Spring AOP и slf4j - logback

Васильев Сергей | Открытие счета клиента

Разработанный функционал системы:



Открытие нового счета клиенту: текущий, накопительный, карточный



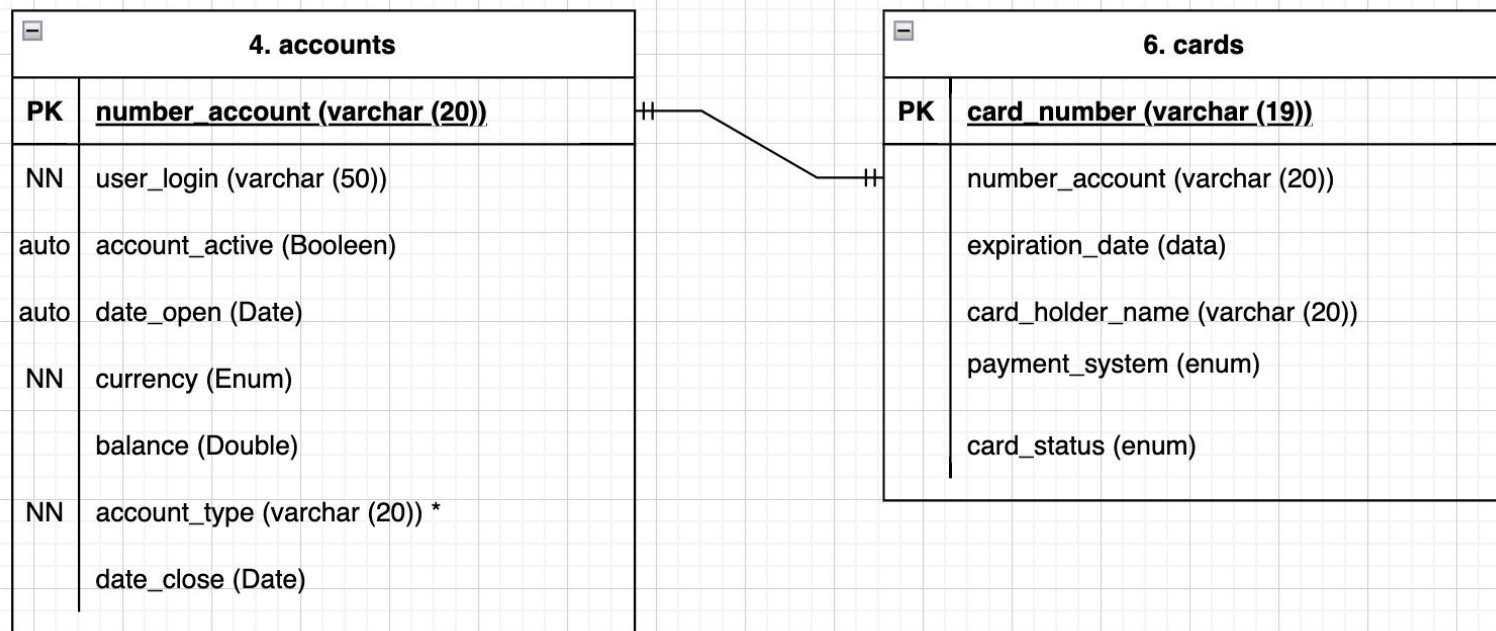
Операции внесения и получения средств со счета (CashIn, CashOut)



Формирование выписки по счету клиента

Ночкин Евгений | Открытие карточного продукта

Реализованная архитектура:



Используемые технологии:

Для разделения функционала по ролям используется фреймворк Spring Security

Хранение карточных продуктов в системе осуществляется в СУБД PostgreSQL в таблице cards, связь с карточным счетом в таблице accounts

Доступ из приложения к таблицам PostgreSQL осуществляется через использование Spring Data

Для вывода данных (UI) используются JSP-страницы

Сквозное логирование осуществляется через Spring AOP и slf4j - logback

Ночкин Евгений | Открытие карточного продукта

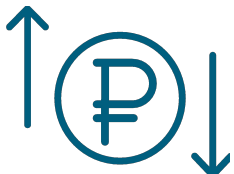
Разработанный функционал системы:



Открытие нового карточного продукта клиенту



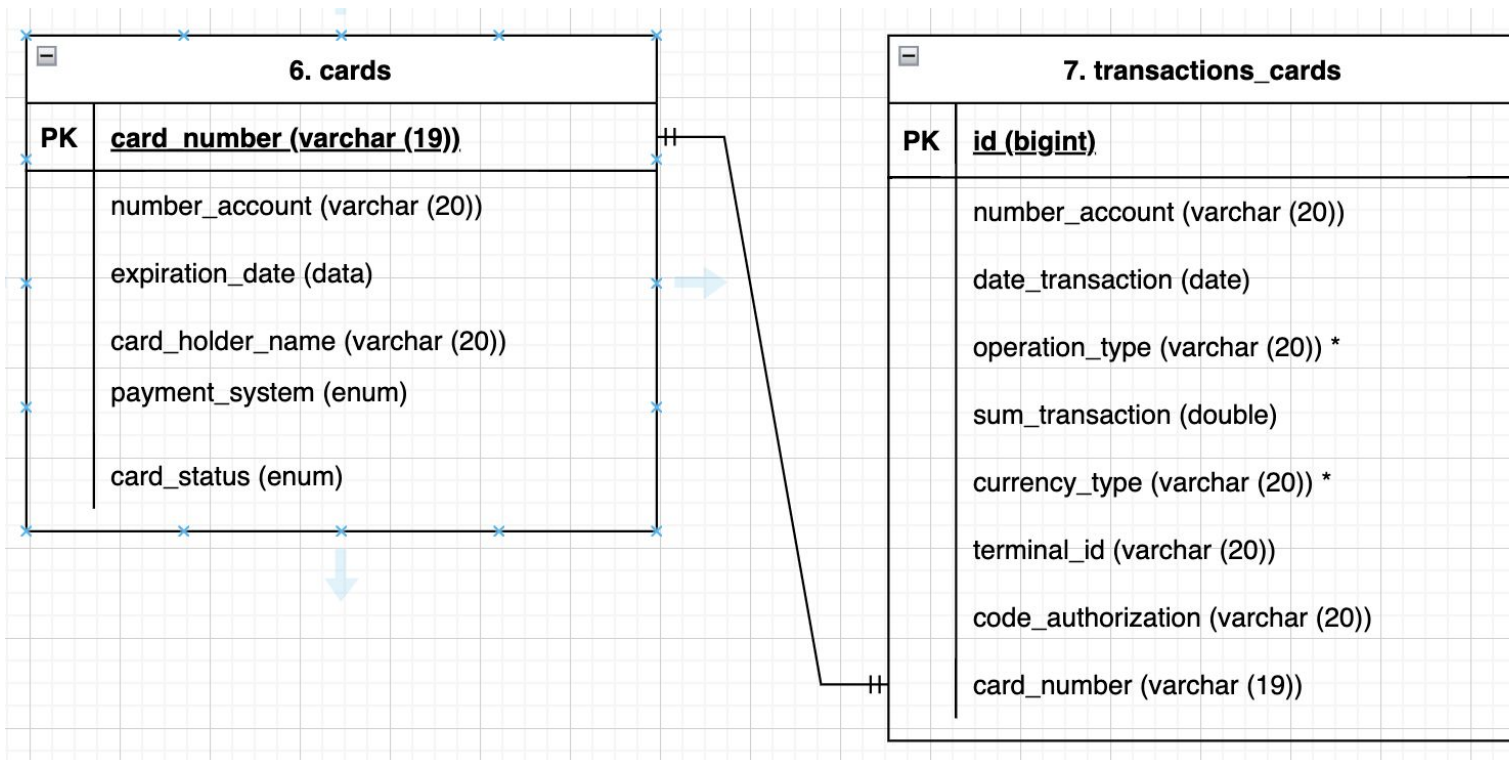
Привязка карточного продукта к счету банковской карты



Изменение доступного остатка клиента при совершении операций

Прощаев Сергей | Микросервис Point of sale (POS)

Реализованная архитектура:



Используемые технологии:

Для разделения функционала по ролям используется фреймворк Spring Security

Хранение транзакций по банковским картам в системе осуществляется в СУБД PostgreSQL в таблице transactions_cards

Доступ из приложения к таблицам PostgreSQL осуществляется через использование Spring Data

Для вывода данных (UI) используются JSP-страницы

Сквозное логирование осуществляется через Spring AOP и slf4j - logback

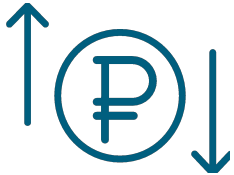
Генерация выписки по операциям в файл формата MS Excel через использование Apache POI

Прощаев Сергей | Микросервис Point of sale (POS)

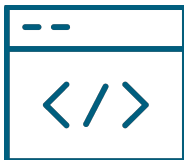
Разработанный функционал системы:



Авторизация суммы покупки по реквизитам банковской карты

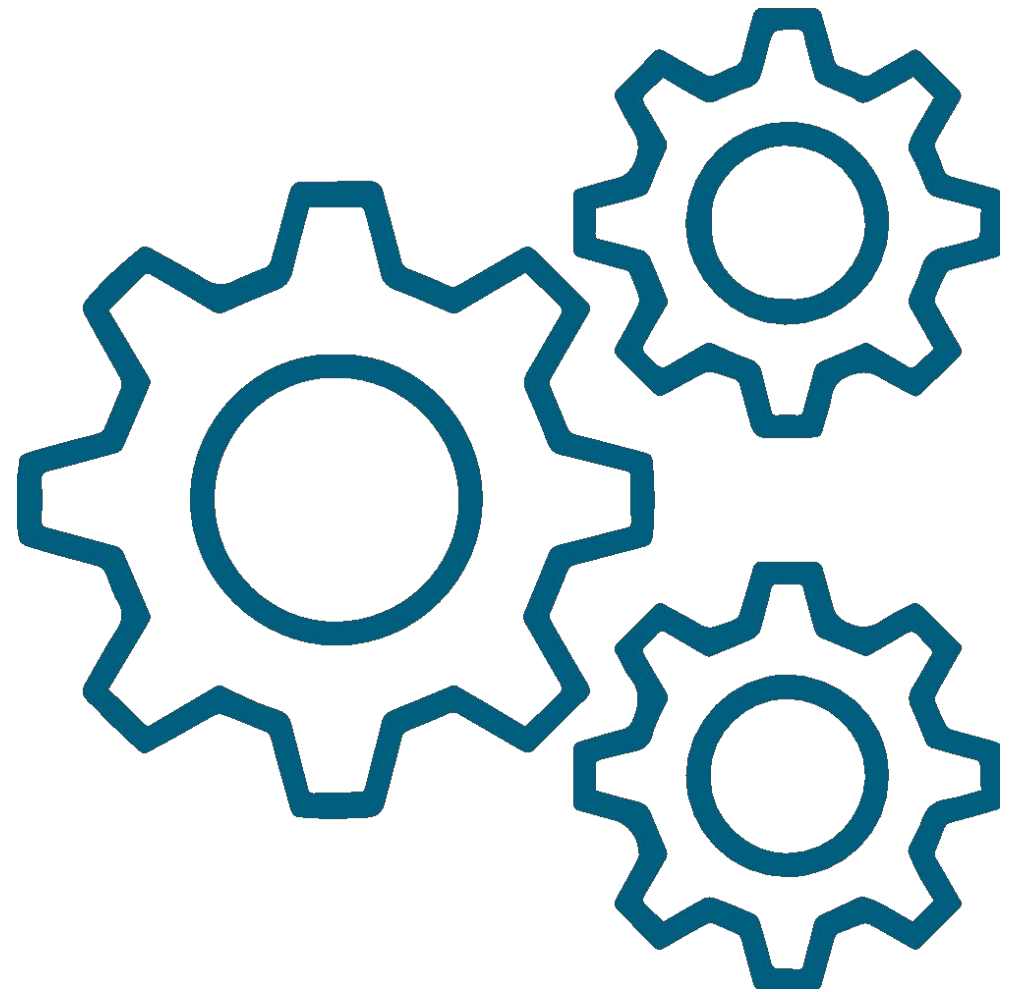


Формирование возврата средств при отказе от покупки клиентом



Формирование выписки по операциям оплаты в POS (ПСВ)

Демонстрация работы системы



Выводы:

1. По итогам реализации проекта «**Bank (Financial processing system)**» был изучен Spring

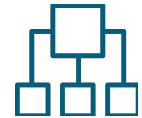
Framework (Spring Boot, Spring Data, Spring Security, Spring AOP)



2. Закреплены навыки работы с СУБД PostgreSQL



3. Применена реализация проектирования системы на основе микросервисов



4. Получен опыт декомпозиции задач и работы в команде с использованием

распределённой системы управления версиями Git



5. Сформированы требования к реализации процессинговых систем (financial processing)

Список использованных источников:

1. Microservice Patterns by Chris Richardson // Published by Manning Publications, 2018
2. Spring in Action, Fifth Edition by Craig Walls // Published by Manning Publications, 2018
3. Java Persistence with Hibernate, Second Edition by Christian Bauer, Gavin King and Gary Gregory // Published by Manning Publications, 2015
4. <https://start.spring.io>
5. <https://www.java.com>
6. <https://www.postgresql.org>

Наша команда



Васильев Сергей



Прощаев Сергей



Губинский Олег



Ночкин Евгений



Филин Константин