

Система управления библиотекой

Проектная работа по дисциплине «Системный анализ и системное проектирование»

Михайлов Дмитрий Андреевич (Р3306)
Малышев Никита Александрович (Р3312)

Университет ИТМО

2025

Проблема и Решение

Что было не так?

- Ручная обработка операций выдачи и возврата книг
- Отсутствие централизованного каталога
- Сложность отслеживания состояния библиотечных материалов
- Неэффективное управление библиотечными процессами

Что мы предлагаем?

- Автоматизация процессов выдачи и возврата
- Централизованный каталог библиотечных материалов
- Онлайн-бронирование и продление сроков выдачи
- Удобный доступ читателей к библиотечным ресурсам

Для кого и зачем?

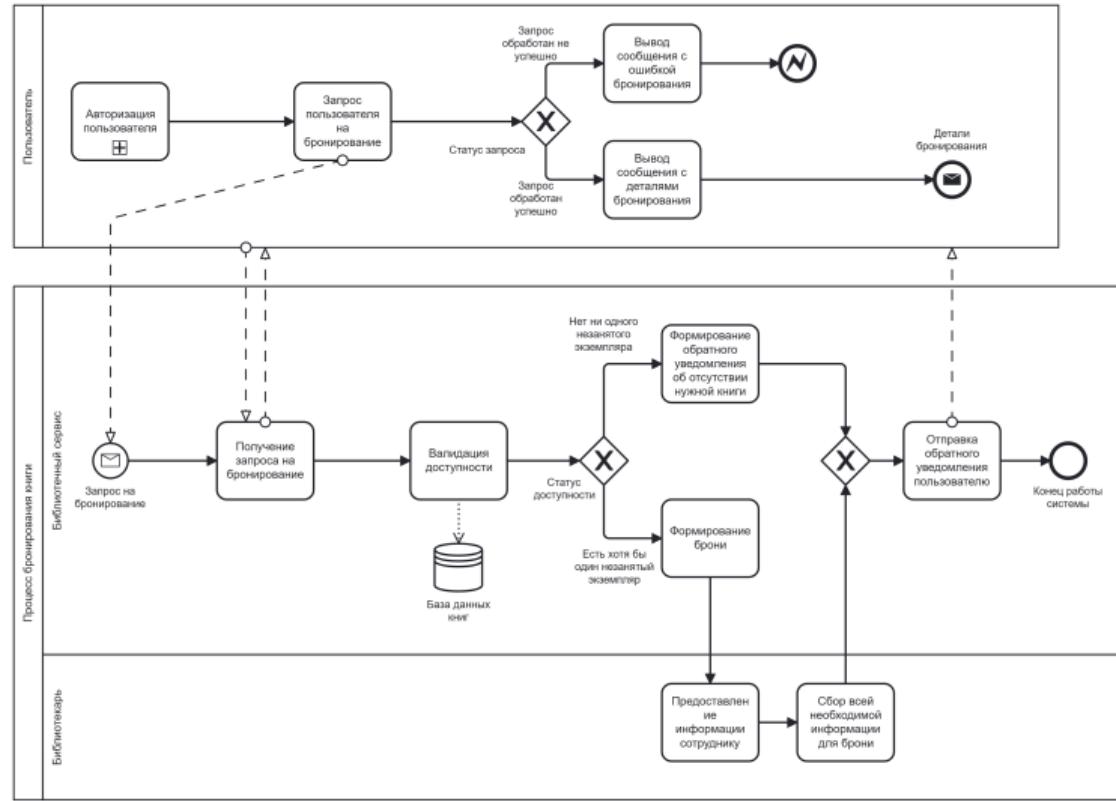
Ключевые стейкхолдеры:

- **Читатели** — пользователи библиотеки, которые могут искать книги, бронировать и продлевать сроки выдачи
- **Библиотекари** — сотрудники библиотеки, которые управляют каталогом и обрабатывают операции выдачи/возврата
- **Директор библиотеки** — заказчик системы, определяет стратегические цели

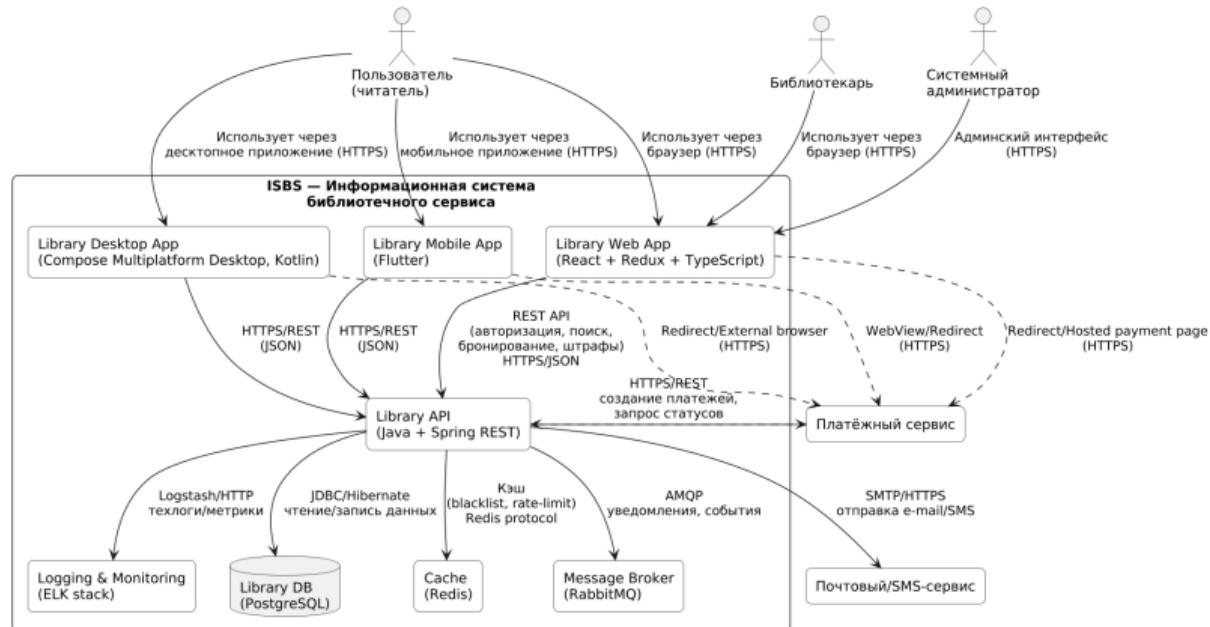
Главное бизнес-требование (SMART):

Увеличить среднее количество бронирований на пользователя на 15% через персональные рекомендации и напоминания

Как это работает? (BPMN)



Из чего это состоит? (Архитектура)



Выбор технологий

Диаграмма контейнеров показывает высокоуровневую архитектуру системы, включая основные приложения, базы данных и внешние системы.

Клиентские приложения:

- **Library Web App** (React, Redux, TypeScript) — веб-интерфейс для читателей и библиотекарей, компонентный подход, типобезопасность
- **Library Mobile App** (Flutter) — мобильное приложение, кроссплатформенность (iOS/Android), единая кодовая база
- **Library Desktop App** (Compose Multiplatform Desktop, Kotlin) — нативное десктопное приложение, современный UI, производительность

Backend и инфраструктура:

Backend: Java + Spring REST

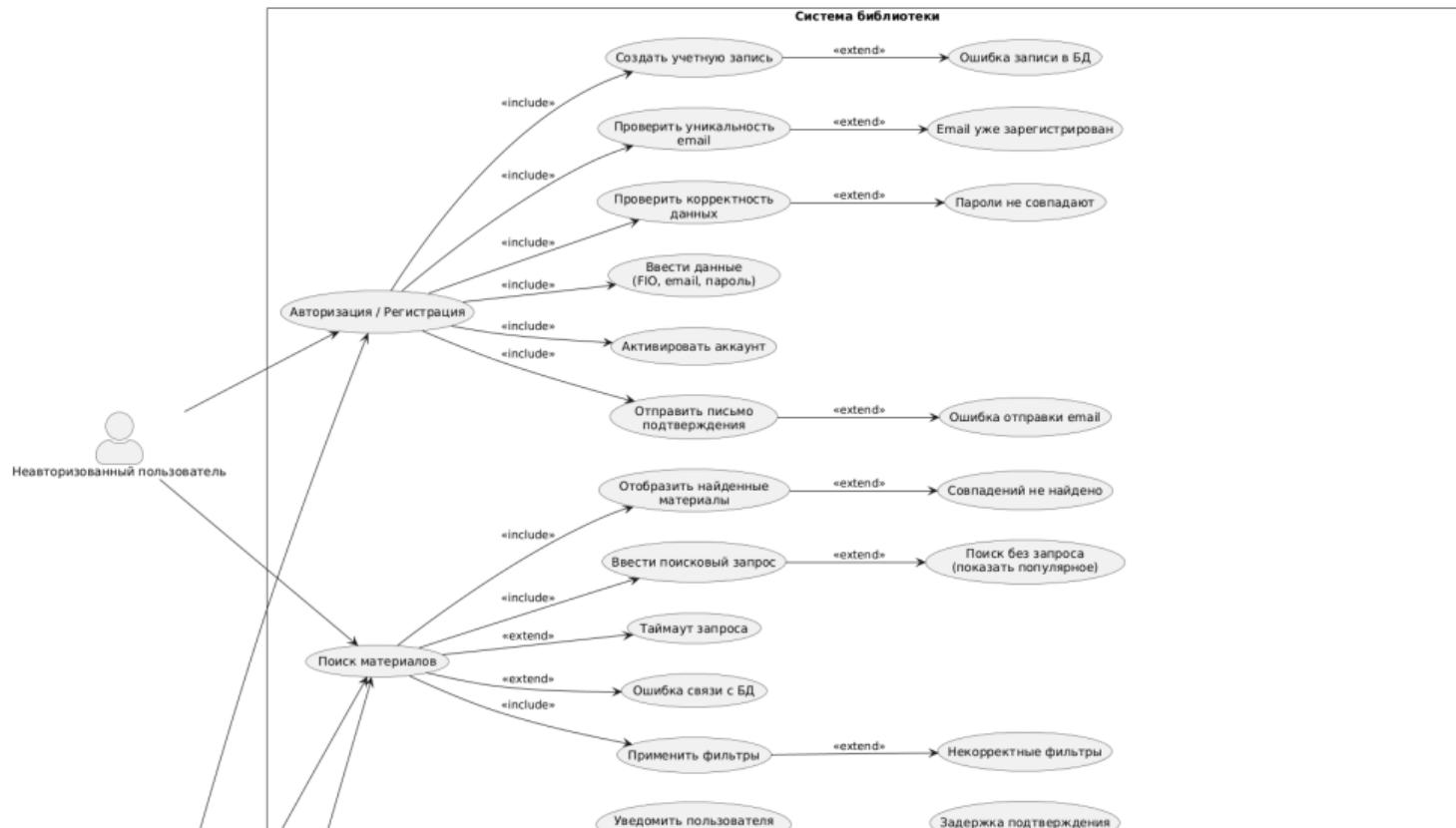
- Зрелый фреймворк, надёжность
- Отличная поддержка REST API

База данных: PostgreSQL

Инфраструктура:

- **Redis** — быстрый кэш, rate-limiting
- **RabbitMQ** — асинхронная обработка уведомлений
- **ELK stack** — централизованное логирование

Что умеет система? (Use Case Diagram)



Функциональные и нефункциональные требования

Ключевые функции:

- **Авторизация** — регистрация и вход пользователей
- **Поиск материалов** — поиск книг в каталоге
- **Оформление брони** — резервирование материалов
- **Оформление выдачи** — выдача материалов читателям
- **Возврат материала** — обработка возврата
- **Погашение штрафа** — оплата штрафов онлайн

Нефункциональные требования:

- Поисковые запросы обрабатываются за время **не более 1.5 секунд** (95% запросов)
- Доступность системы не менее **98%** в течение месяца
- Система поддерживает до **200 одновременных пользователей**
- Все данные защищены протоколом **TLS 1.2+**

**Мы спроектировали систему управления библиотекой,
которая:**

- Решает проблему неэффективного управления библиотечными процессами
- Автоматизирует операции выдачи и возврата книг
- Предоставляет удобный доступ к библиотечным ресурсам
- Обеспечивает централизованный каталог материалов

Все требования и архитектура документированы

Спасибо за внимание!

Вопросы?