

Факультет информационных технологий и анализа
больших данных
Кафедра информационных технологий

Разработка веб-приложения для организации совместной работы над проектами

Выполнил студент группы ПИ21-3
Балашкин Андрей Михайлович

Руководитель к.т.н., доцент
Хасанов Ильнур Ильдарович

Актуальность исследования обусловлена необходимостью создания удобного и доступного инструмента для командной работы

Задачи путаются, сроки летят... Надо что-то придумать.

20:24

Знаю, знаю. Только вот всегда что-то нужно настраивать, изучать...

20:39 ✓✓

Ну, может, не обязательно что-то глобальное. Просто, чтобы задачи удобно ставить, быстро обмениваться информацией.

20:40

Цель - повышение эффективности и упрощение процесса организации совместной работы над проектами

- 1 анализ существующих решений
- 2 определение потребностей целевой аудитории
- 3 изучение принципов проектирования веб-приложений
- 4 разработка веб-приложения
- 5 тестирование разработанного решения

Разработанное веб-приложение должно позволяет пользователям:

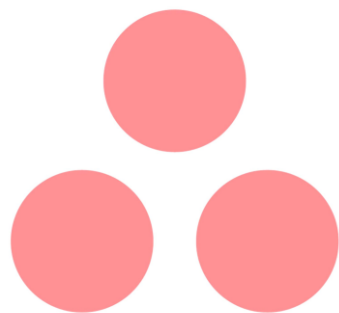
- создавать и управлять задачами с разделением по статусам;
- использовать канбан-доску для наглядного отображения рабочего процесса;
- вести текстовые заметки и комментарии;
- приглашать участников для совместной работы без сложной настройки ролей;
- работать с разных устройств благодаря адаптивному интерфейсу.

Объект исследования — процессы организации совместной работы с использованием цифровых инструментов

Предмет исследования — методы и технологии разработки веб-приложений



Существующие решения

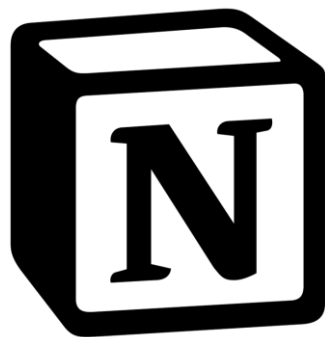


Asana



Jira

Основные проблемы: перегруженный интерфейс,
сложность настройки и сильно ограниченная
бесплатная версия



Notion

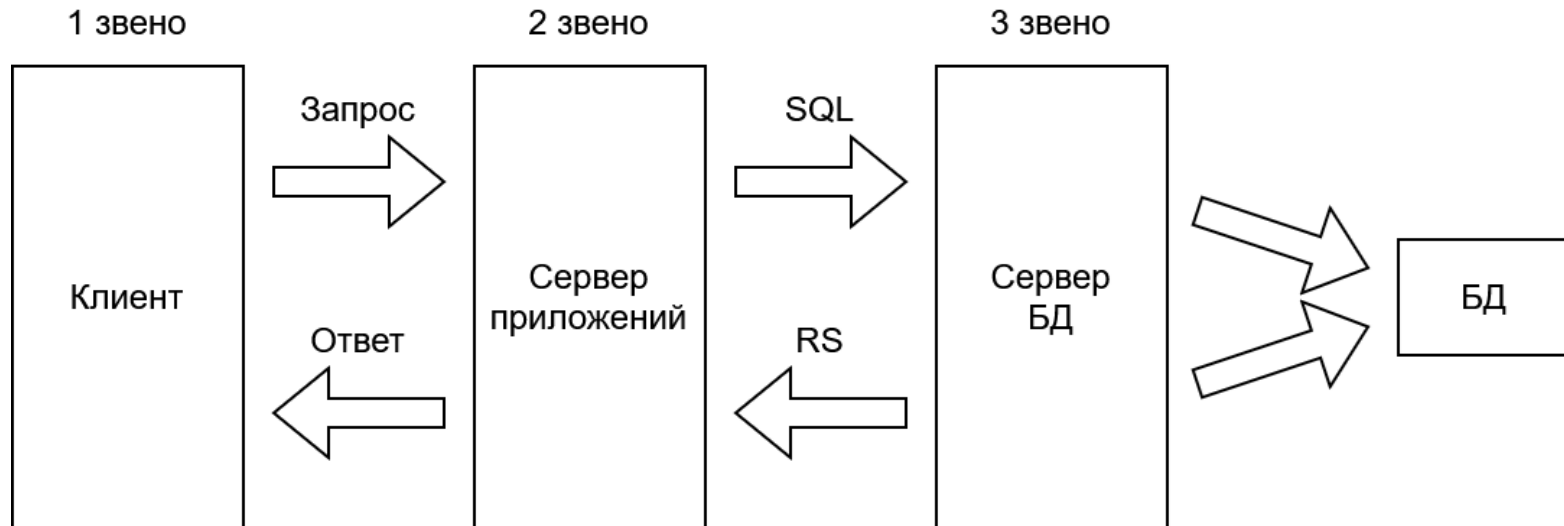


Trello

	Управление задачами	Канбан	Текстовые записки	Простое управление ролями	Бесплатная версия
Trello	+	+	-	+	10 канбан-досок
Asana	+	-	-	-	15 человек
Jira	+	+	-	-	10 человек, ограниченные настройки
Notion	+	-	+	-	10 человек

Таблица сравнения возможностей сервисов

Современные приложения построены по клиент-серверной модели взаимодействия

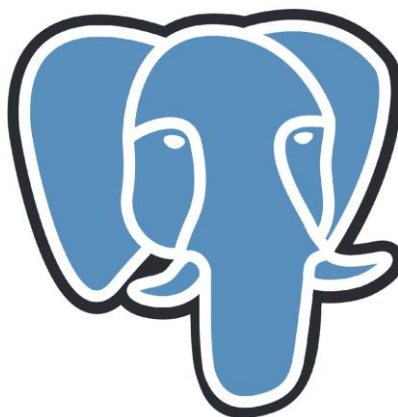
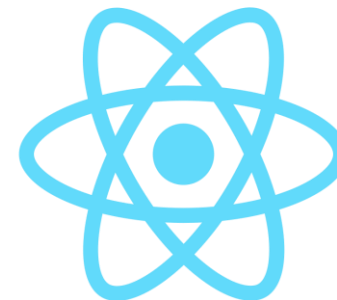


	Django	Spring	Express	Flask
Язык	Python	Java	JavaScript	Python
Концепция	MVT (MVC)	MVC	MVC	MVT (MVC)
Совместимость с PostgreSQL	+	+	+	+
Сложность	Средне	Сложно	Легко	Легко
Особенности	ORM Админ-панель Миграции Аутентификация Безопасность	Аутентификация Аннотации	Настройка вручную Зависимости	Базовые шаблоны

Таблица сравнения фреймворков для разработки серверной части

	Vue	React	Angular
Производительность	Высокая	Высокая	Высокая
Гибкость	Высокая	Высокая	Низкая
Поддержка	Сообщество	Facebook	Google
Рендеринг	VDOM	VDOM	RDOM
Масштабируемость	Для небольших проектов	Для средних и больших проектов	Для крупных проектов

Таблица сравнения фреймворков для разработки клиентской части



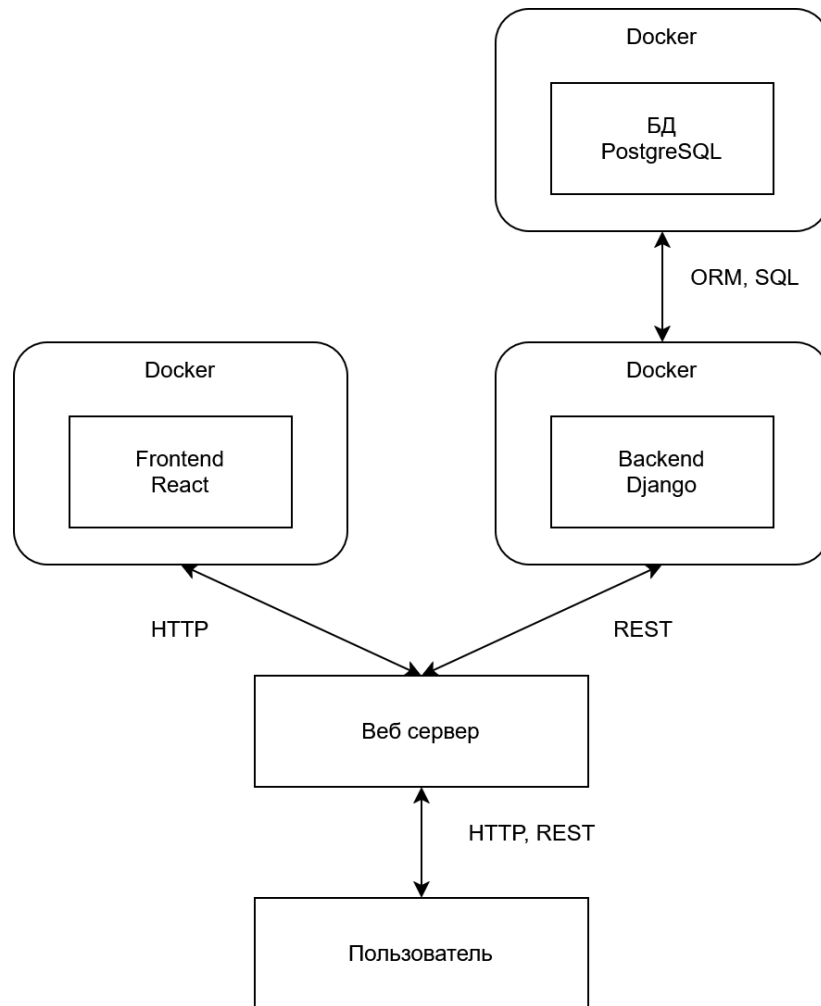
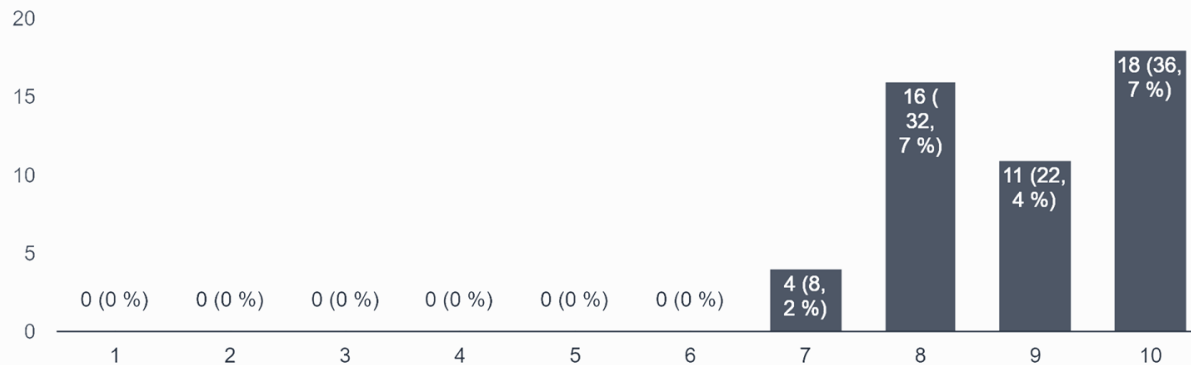


Схема архитектуры
(взаимодействия клиента с
приложениями)

Было проведено тестирование с использованием Google Forms. Средняя оценка - 8.88

Общее впечатление от сервиса

49 ответов



В результате в ходе выполнения работы были решены все поставленные задачи:

- проведен анализ существующих решений и определены основные потребности целевой аудитории;
- сформулированы и обоснованы функциональные и нефункциональные требования к приложению;
- выбраны оптимальные архитектурные и технологические решения;
- реализованы серверная и клиентская части приложения;
- проведено тестирование готового продукта и дана оценка его эффективности.



Факультет информационных технологий и анализа
больших данных
Кафедра информационных технологий

Спасибо за внимание!

Выполнил студент группы ПИ21-3
Балашкин Андрей Михайлович

Руководитель к.т.н., доцент
Хасанов Ильнур Ильдарович