

Федеральное государственное бюджетное образовательное учреждение
высшего образования
«МИРЭА – Российский технологический университет
Институт информационных технологий (ИИТ)
Кафедра практической и прикладной информатики (ППИ)

Курсовая работа по дисциплине «Обоснование и
разработка требований к программным системам»

Тема: Обоснование и разработка требований к
программной системе управления разработками
цифровой компании

Выполнил студент группы ИКБО-65-23
Олефиров Г.Г.

Принял:
Бирюкова А.А.

Москва 2025 г.

Введение

Документирование требований к программным системам является одним из ключевых этапов разработки, определяющим успешность всего проекта. Требования служат источником истины, формируя общее понимание между заказчиком и разработчиком, что предотвращает создание продукта, не соответствующего актуальным потребностям бизнеса. Чётко сформулированные требования позволяют реалистично оценить сроки и бюджет, а также являются основой для проектирования архитектуры системы. В итоге такая проработка требований минимизирует количество дорогостоящих переработок, обеспечивает соответствие результата ожиданиям и закладывает фундамент для эффективного развития программного продукта.

Объект исследования

Объектом исследования являются процессы управления проектами и задачами в цифровой компании, а также практики сбора, анализа и формализации требований к программным системам.

Предмет исследования

Предметом исследования является процесс разработки требований к программной системе, поддерживающей управление разработками и обеспечивающей прозрачность и эффективность работы цифровой компании.

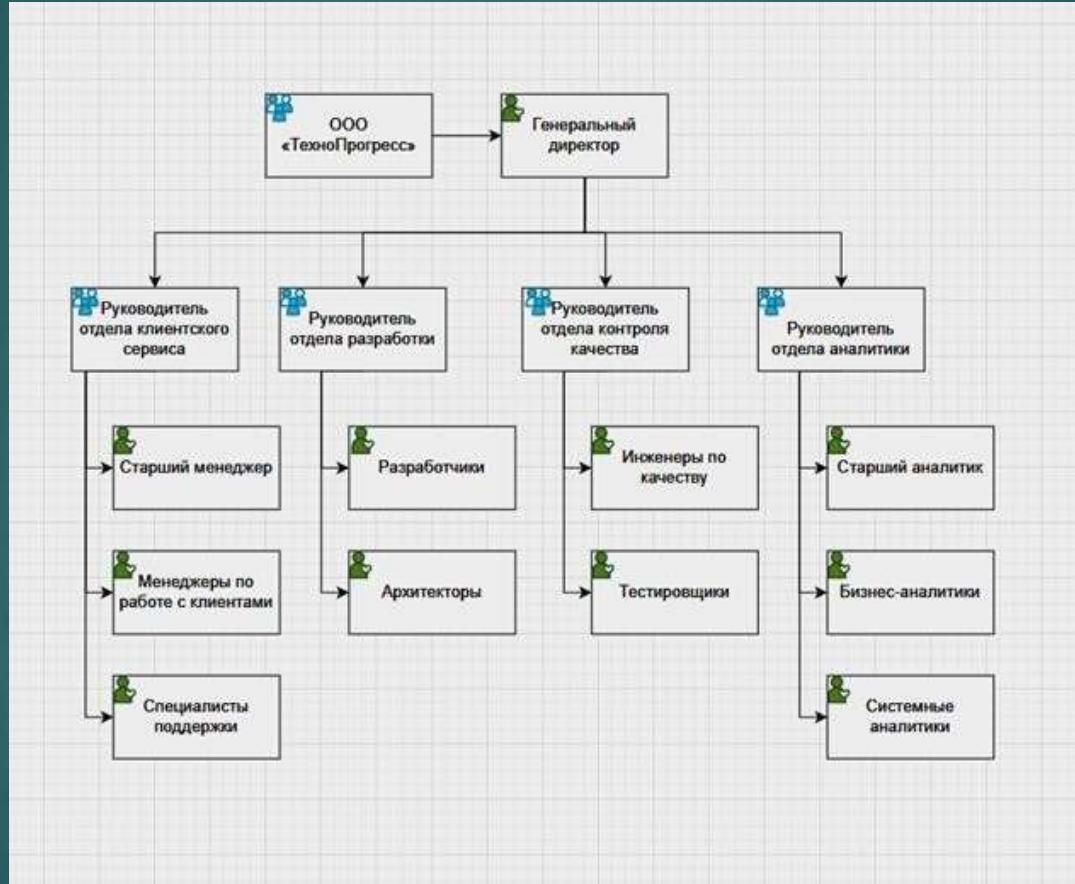
Цель работы

Целью курсовой работы является освоение методологии разработки и оформления требований к программной системе управления разработками на примере цифровой компании.

Задачи работы

- Изучить теоретические основы разработки требований к программным системам.
- Провести анализ предметной области управления проектами и задачами в цифровой компании.
- Сформулировать и документально оформить пользовательские, функциональные и нефункциональные требования к системе.
- Определить границы проекта и разработать концепцию программного продукта управления разработками.
- Смоделировать бизнес-процессы, связанные с управлением проектами и задачами, и подготовить диаграммы вариантов использования.

Предпроектное исследование

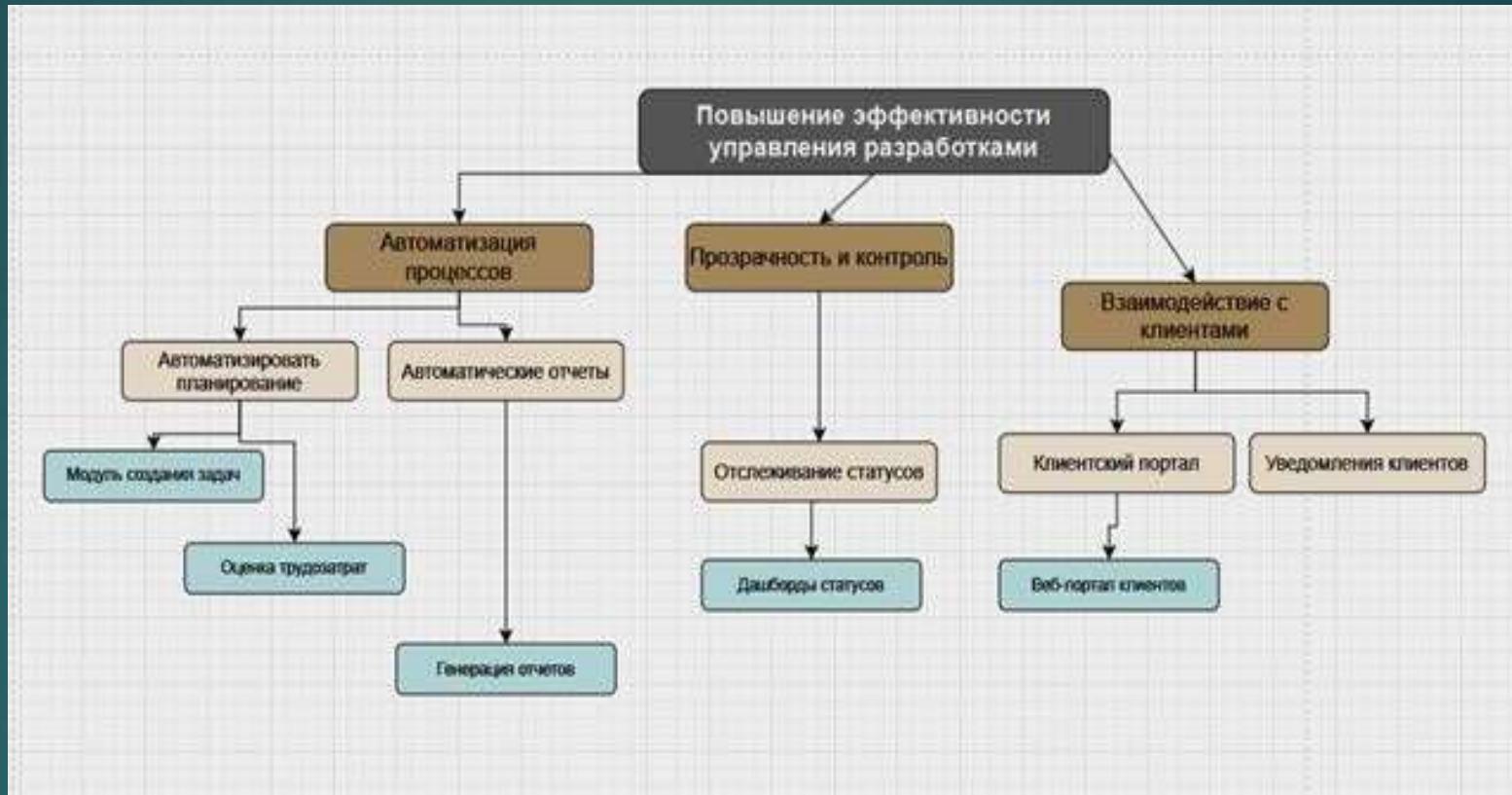


Организационная модель организации

Предпроектное исследование

Критерий	Jira Software	YouTrack
Функциональность управления проектами и задачами	Поддержка бэклогов, спринтов, канбан-досок, эпиков, связей задач, SLA по задачам	Поддержка agile-досок, спринтов, эпиков, гибкая система связей задач
Поддержка ролей и совместной работы	Гранулярные права, роли, комментарии, упоминания, наблюдатели, workflow-уведомления	Роли и группы, комментарии, упоминания, командные доски, уведомления
Отчетность и аналитика	Богатый набор отчётов: burndown, velocity, контроль соблюдения сроков, дашборды	Встроенные отчёты и дашборды, возможность кастомных отчётных виджетов
Интеграции с инструментами разработки	Глубокие интеграции с Git, CI/CD, Confluence, системой код-ревью и др.	Интеграции с Git, CI/CD, IDE JetBrains, мессенджерами
Удобство использования и настраиваемость процессов	Гибкая настройка workflow, полей, экранов, автоматизаций; интерфейс может быть перегружен для новичков	Гибкая настройка workflow через правила и скрипты, лаконичный интерфейс
Стоимость и модель распространения	Подписка (облачно) или серверная лицензия; стоимость растёт с числом пользователей	Подписка или on-premises-лицензия; более мягкая ценовая политика для малых команд

Предпроектное исследование



Предпроектное исследование

Параметр	Частичное решение 1	Частичное решение 2	Частичное решение 3	Частичное решение 4
Платформа системы	Веб-приложение (SPA на React + Tailwind)	Веб-приложение (классический MVC)	Десктопное приложение	Мобильное приложение
Управление задачами и проектами	Канбан-доска	Иерархический список задач	Scrum-спринты	Гибридный режим (Kanban + спринты)
Информирование пользователей	E-mail-уведомления	Push-уведомления в веб-интерфейсе	Интеграция с мессенджером (Telegram/Slack)	SMS-уведомления
Взаимодействие ролей (клиент/разраб/аналитик)	Единая панель с переключением ролей без перезагрузки	Отдельные интерфейсы для каждой роли	Общие экраны + роль-специфичные вкладки	Только интерфейсы разработчика и аналитика

Морфологическая карта

Разработка требований к системе

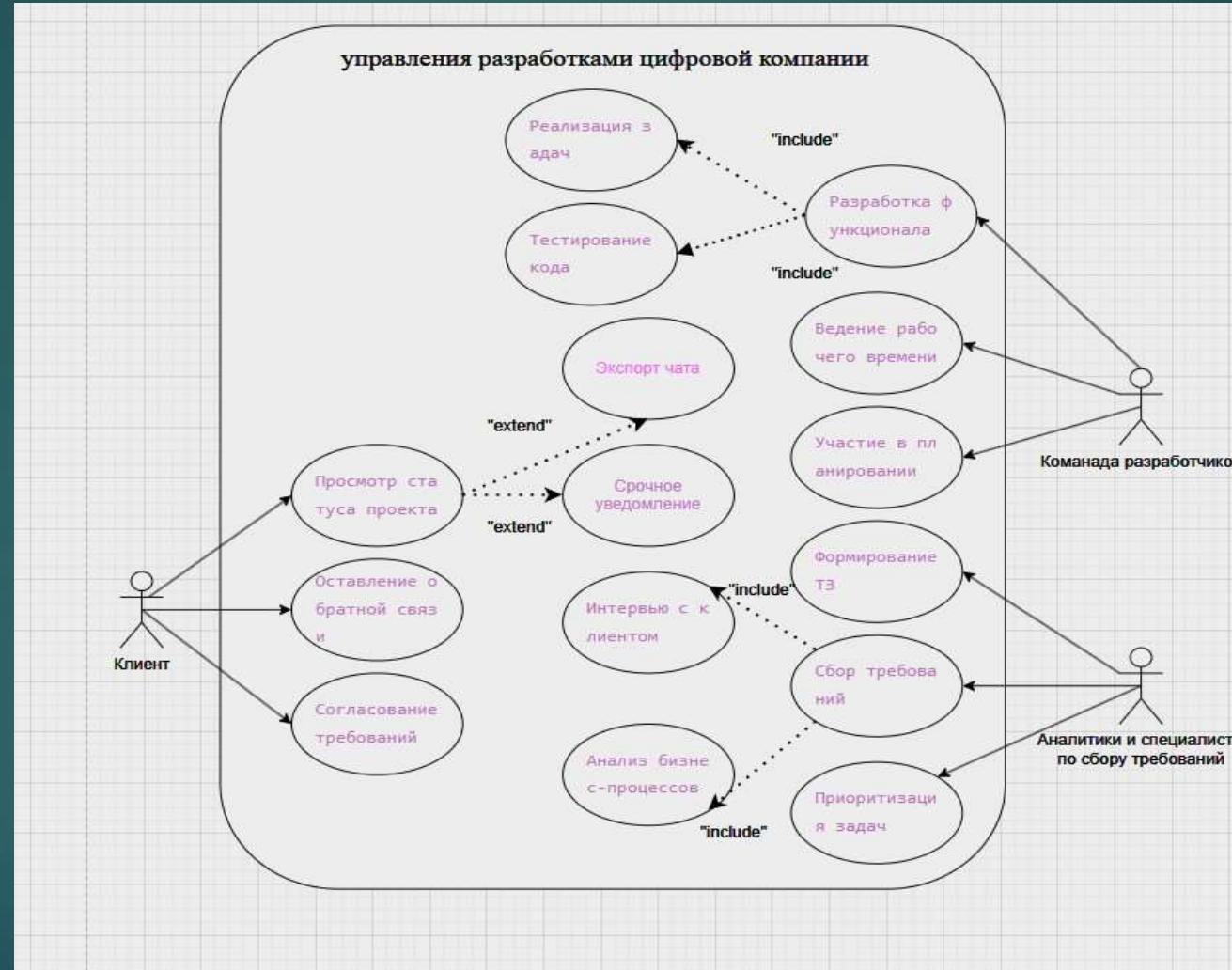


Диаграмма вариантов использования

Разработка требований к системе

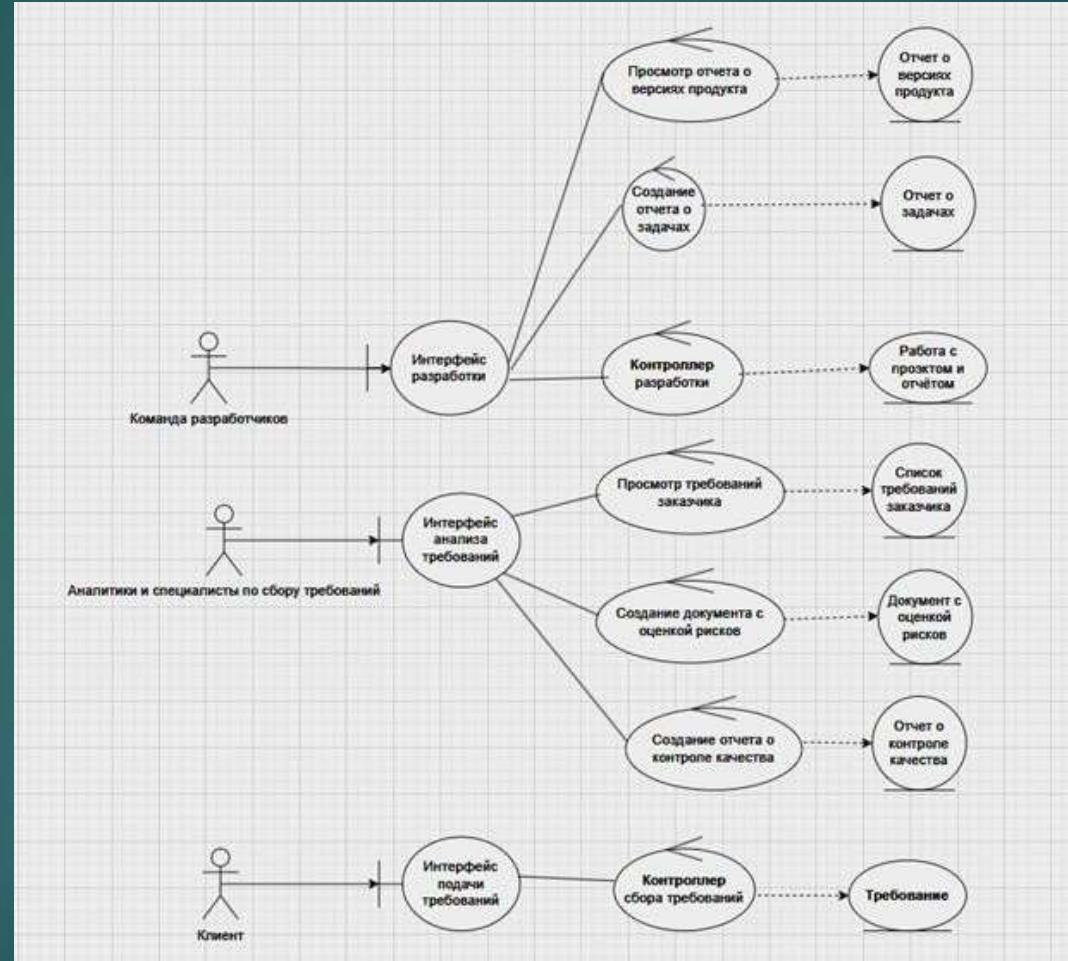


Диаграмма классов анализа

Разработка требований к системе

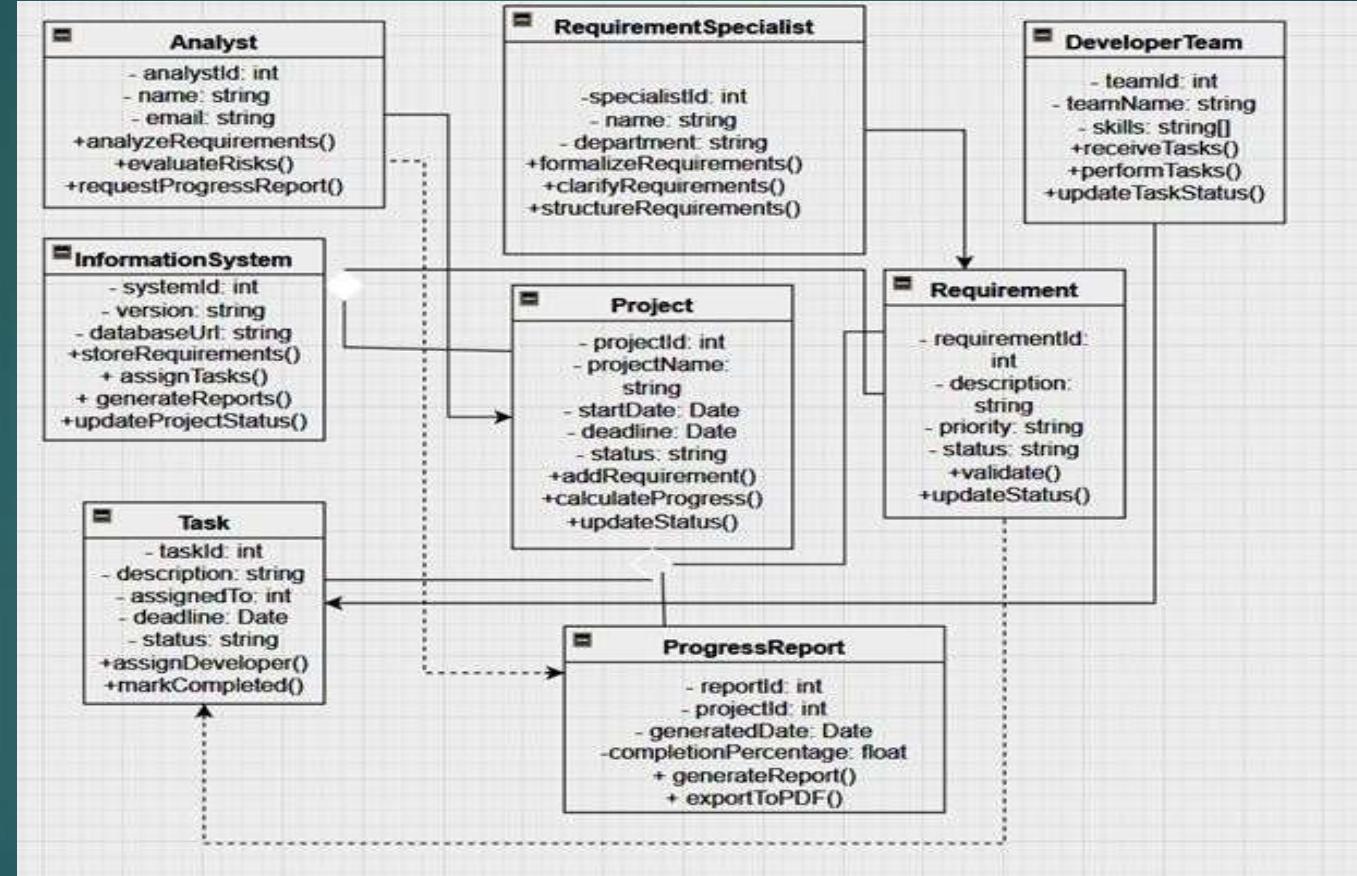


Диаграмма классов

Разработка требований к системе

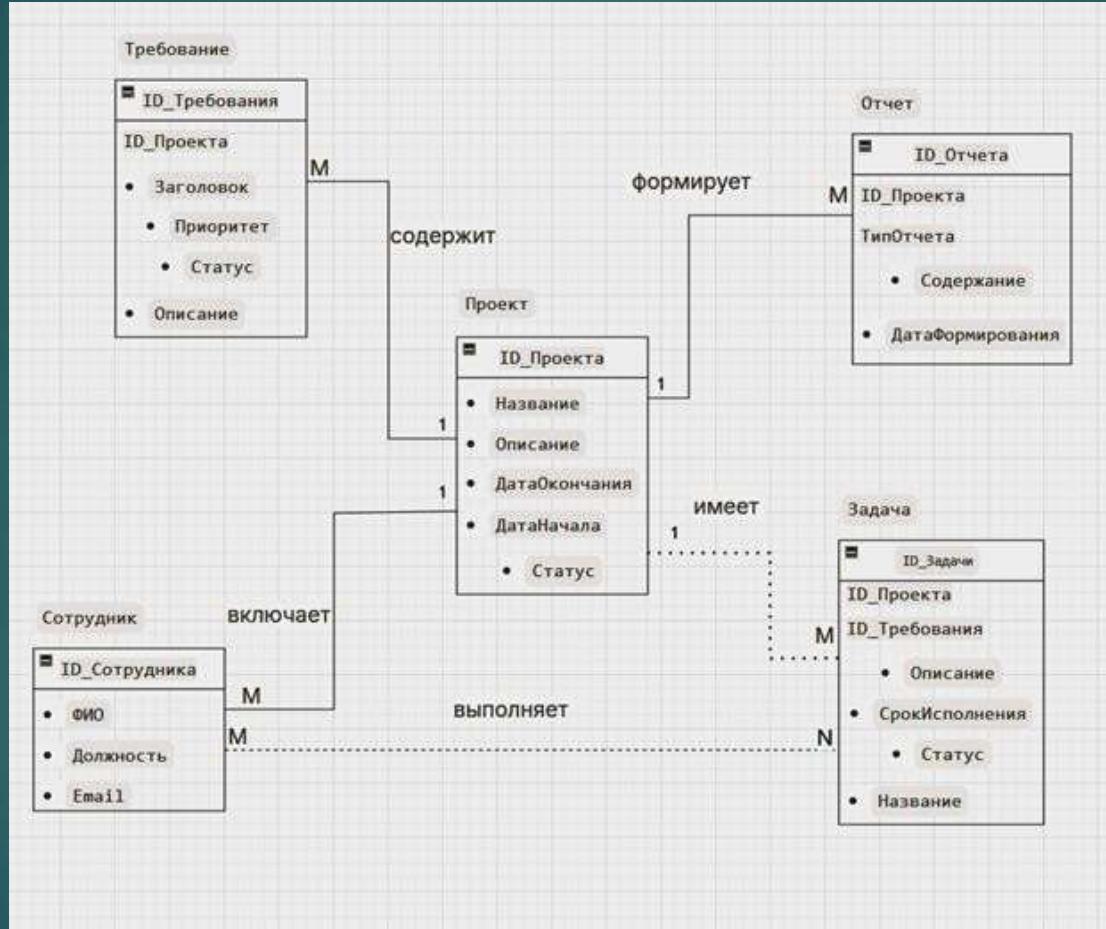


Диаграмма «сущность-связь»

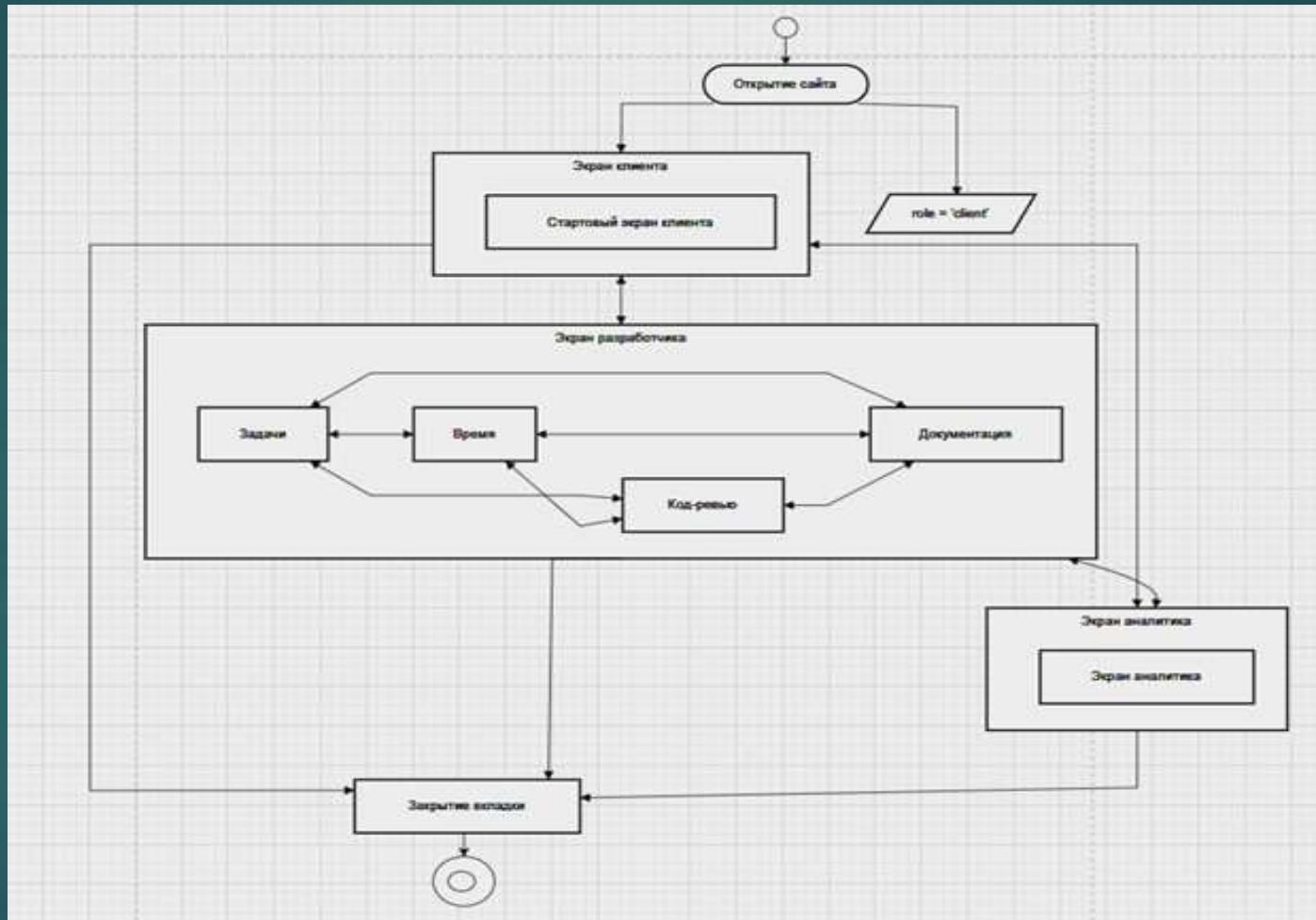
Разработка требований к системе

The image displays four side-by-side screenshots of a software application interface, likely a web-based tool for project management or requirement tracing. The interface is designed for three main user roles: Клиент (Client), Разработчик (Developer), and Аналитик (Analyst). The tabs at the top of each screenshot correspond to these roles.

- Клиент (Client) View:** Shows a progress bar for the project at 33%, a list of tasks (Design of the main page, Integration of payment API, Configuration of notification settings), and a feedback form.
- Разработчик (Developer) View:** Displays a dashboard with tabs for Tasks, Time, Code Review, and Documentation. It shows a list of tasks under 'My tasks' (Integration of payment API, Configuration of notification settings) and a list of code reviews (Add form validation, Database query optimization, Component refactoring).
- Аналитик (Analyst) View:** Features a 'Code review' tab. It lists three tasks: 'Добавить валидацию форм' (Author: Иван), 'Оптимизация запросов БД' (Author: Мария), and 'Рефакторинг компонентов' (Author: Петр). Each task has buttons for 'Отклонить' (Reject), 'Одобрить' (Approve), and 'Просмотр' (View).
- General View (Client, Developer, Analyst):** Shows a 'New requirement' section with fields for 'Name of requirement', 'Description of requirement', 'Priority' (dropdown with 'Добавить'), and a 'Create' button. It also includes a 'Prioritization of tasks' section showing three tasks with priority levels: 'Высокий' (High) for the first two, and 'Низкий' (Low) for the third. A summary at the bottom indicates 3 requirements total, 2 of which are of high priority.

Некоторые элементы пользовательского интерфейса

Разработка требований к системе



Взаимодействие диалоговых окон

Разработка требований к системе

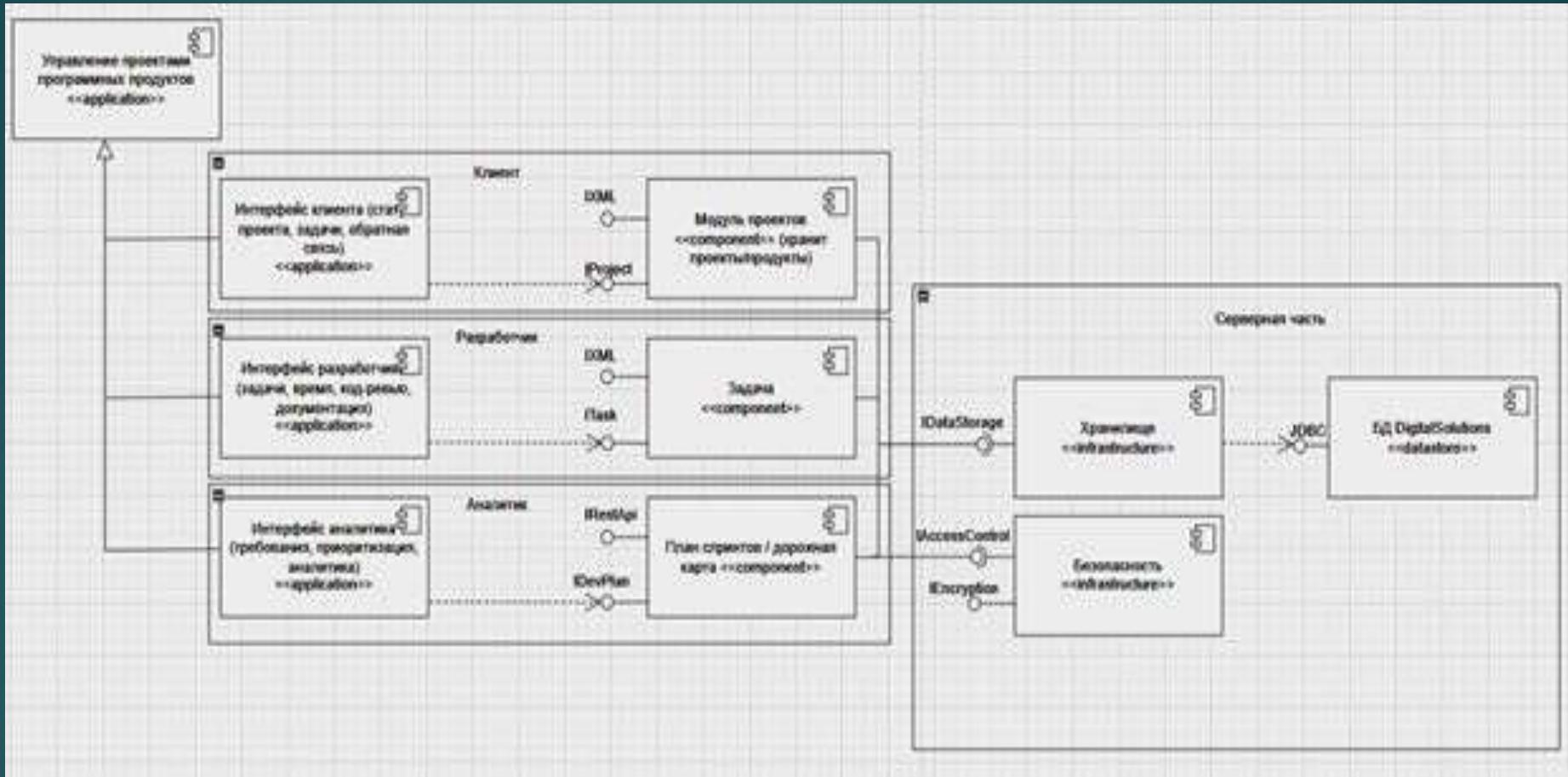


Диаграмма компонентов

Разработка требований к системе

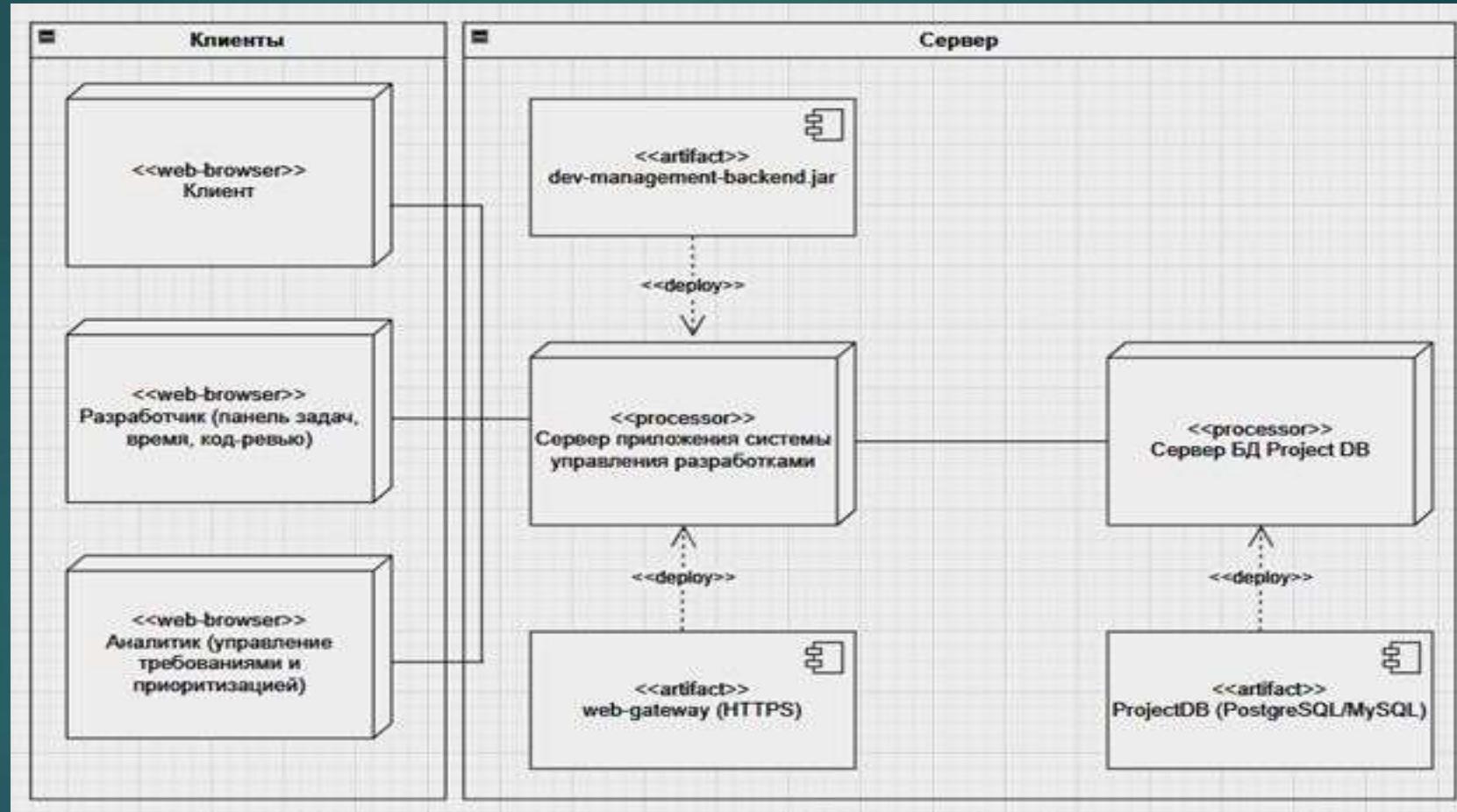


Диаграмма развертывания

ЗАКЛЮЧЕНИЕ

В ходе выполнения данной курсовой работы была достигнута поставленная цель — освоены подходы к разработке требований к программной системе управления разработками цифровой компании. На примере условной компании

«ТехноПрогресс» были проанализированы процессы управления проектами, задачами и взаимодействием участников разработки, что позволило перейти от теоретических основ к практическому формированию требований.

В процессе работы были последовательно решены все поставленные задачи. Проведён анализ теоретических основ разработки требований, что позволило обосновать выбор используемых моделей и артефактов при их описании. Выполнено предпроектное исследование предметной области и организационной структуры компании, выявлены ключевые бизнес-процессы, участники и их роли. На этой основе сформулирован и задокументирован набор пользовательских, функциональных и нефункциональных требований к системе управления разработками, включающий требования к качеству, ограничениям, интерфейсу и архитектуре. Определены границы проекта и разработана концепция продукта, отражающая назначение системы, целевую аудиторию и ожидаемый эффект от внедрения.

Главным результатом курсовой работы является сформированный комплекс требований, который может служить основой для последующей разработки и прототипирования программной системы управления разработками. Чёткая формулировка требований и их структурированное представление снижают риск расхождений между ожиданиями заказчика и реализуемым функционалом, а также упрощают планирование архитектуры и этапов реализации. Полученные результаты подтверждают, что качественный сбор, анализ и документирование требований являются критически важным этапом создания программных систем, позволяющим сократить затраты времени и ресурсов на дальнейших стадиях жизненного цикла программного обеспечения.

Спасибо за внимание!