



МИНОБРНАУКИ РОССИИ

Федеральное государственное бюджетное образовательное учреждение
высшего образования
«МИРЭА – Российский технологический университет»

РТУ МИРЭА

Институт информационных технологий (ИИТ)
 Кафедра математического обеспечения и стандартизации информационных технологий
(МОСИТ)

ОТЧЕТ ПО ПРАКТИЧЕСКОЙ РАБОТЕ
по дисциплине «Обоснование и разработка требований к программным
системам»

Практическое занятие № 4

Вариант № 13. Обоснование и разработка требований к программной
системе управления разработками цифровой компании

Студент группы *ИКБО-65-23, Олефиров ГГ*

(подпись)

Преподаватель *Бирюкова А.А.*

(подпись)

Отчет представлен «____ » _____ 202 ____ г.

Москва 2025 г.

1. Цель занятия

Выявить и описать внутреннюю архитектуру программной системы с помощью диаграммы классов анализа, а также описать взаимодействие элементов во времени.

2. Постановка задачи

Задачами работы являются:

- определить на концептуальном уровне состав элементов системы в виде классов анализа;
- описать взаимодействие элементов системы между собой и с внешним пользователем во времени.

В третьей практической работе была разработана следующая диаграмма вариантов использования:

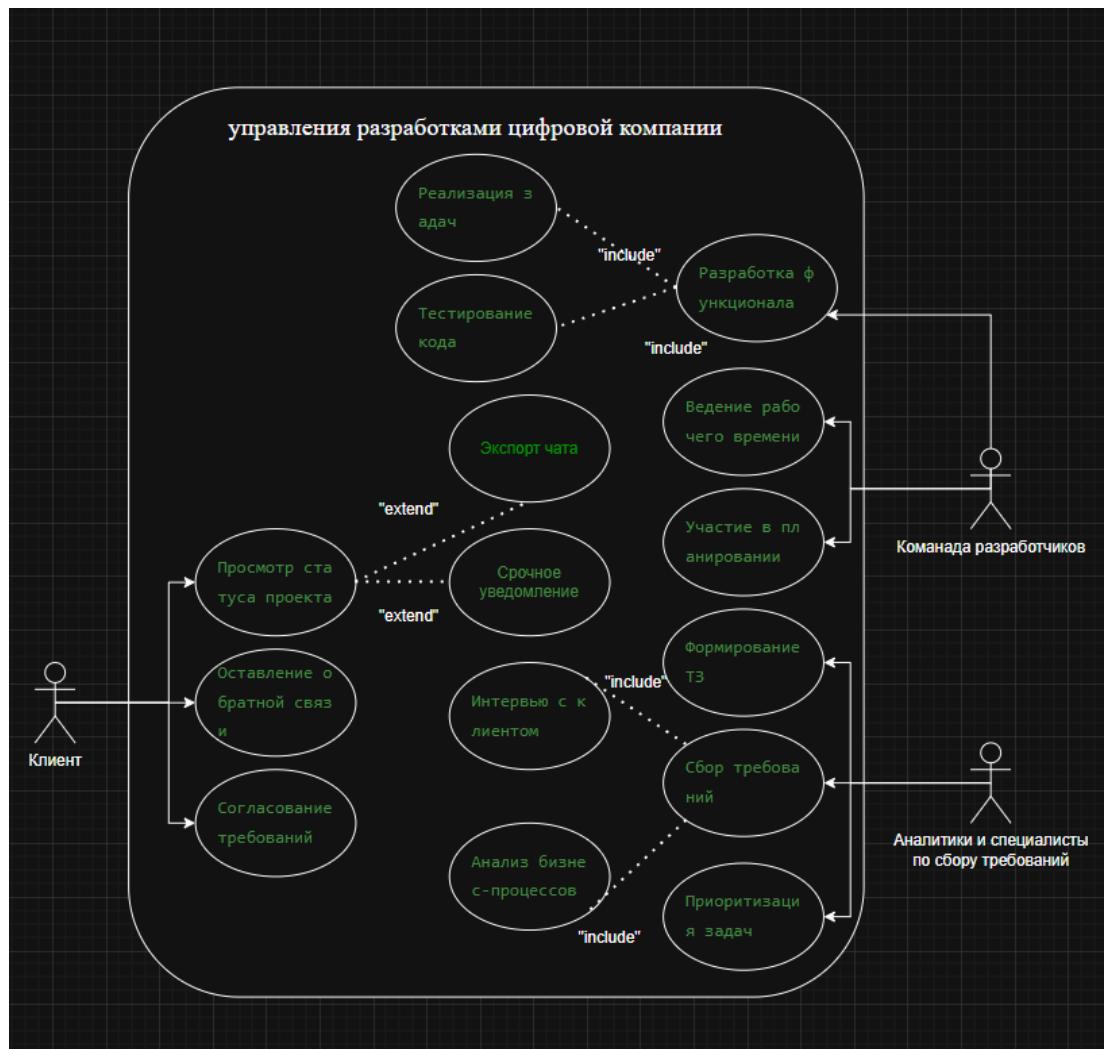


Рисунок 1 – Диаграмма вариантов использования для управления разработками цифровой компании

3. Результат работы

3.1. выявление и описание внутренней архитектуры программной системы (Диаграмма классов анализа)

Используя диаграмму вариантов использования, построенную на предыдущей практике, определим основные классы анализа:

Границные классы:

- Интерфейс подачи требований
- Интерфейс разработки
- Интерфейс анализа требований

Управляющие классы:

- Контроллер сбора требований
- Контроллер разработки
- Контроллер анализа требований

Классы-сущности:

- Требование
- Проект
- Отчет о версиях продукта
- Отчет о задачах
- Список требований заказчика
- Документ с оценкой рисков

Взаимосвязи между классами:

Система реализует трехуровневую архитектуру, в которой:

- Акторы взаимодействуют с системой через граничные классы
- Управляющие классы координируют бизнес-процессы
- Классы-сущности обеспечивают хранение и управление данными

Основные отношения представлены ассоциациями между граничными и управляющими классами, а также зависимостями между управляющими классами и сущностями, что соответствует принципам проектирования программных систем, изложенным в учебных пособиях по данной дисциплине [1]. Диаграмма демонстрирует четкое разделение ответственности между компонентами системы и обеспечивает

масштабируемость архитектуры для дальнейшего развития функциональности, что является важным аспектом при построении диаграмм UML для анализа и проектирования информационных систем [2].

Разработаем модель классов анализа и установим между классами соответствующие отношения.

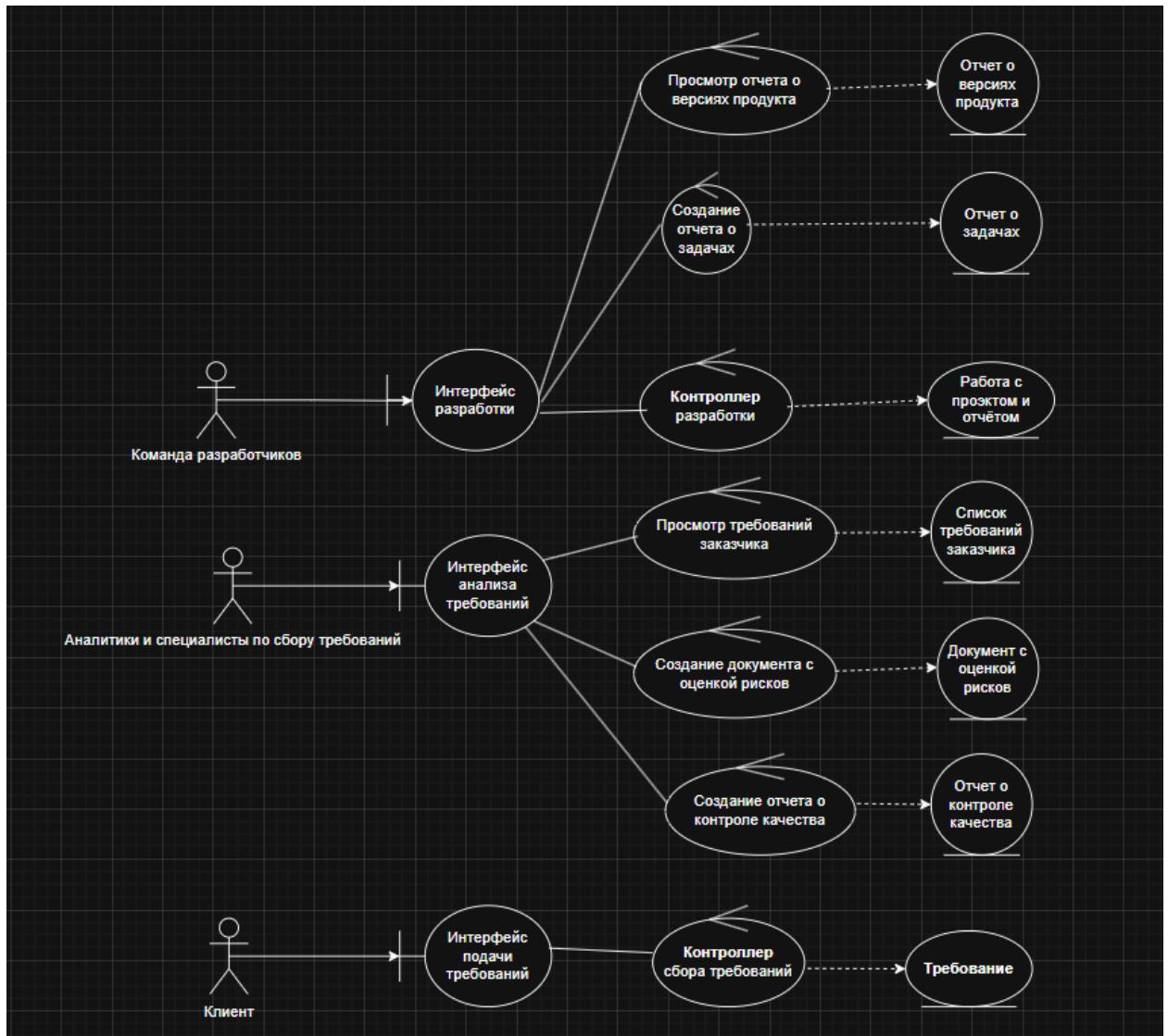


Рисунок 2 - Итоговая единая диаграмма классов анализа

3.2. Описание взаимодействия элементов во времени (диаграмма последовательности)

Построим диаграмму последовательности процесса «Процесс согласования и реализации требований»

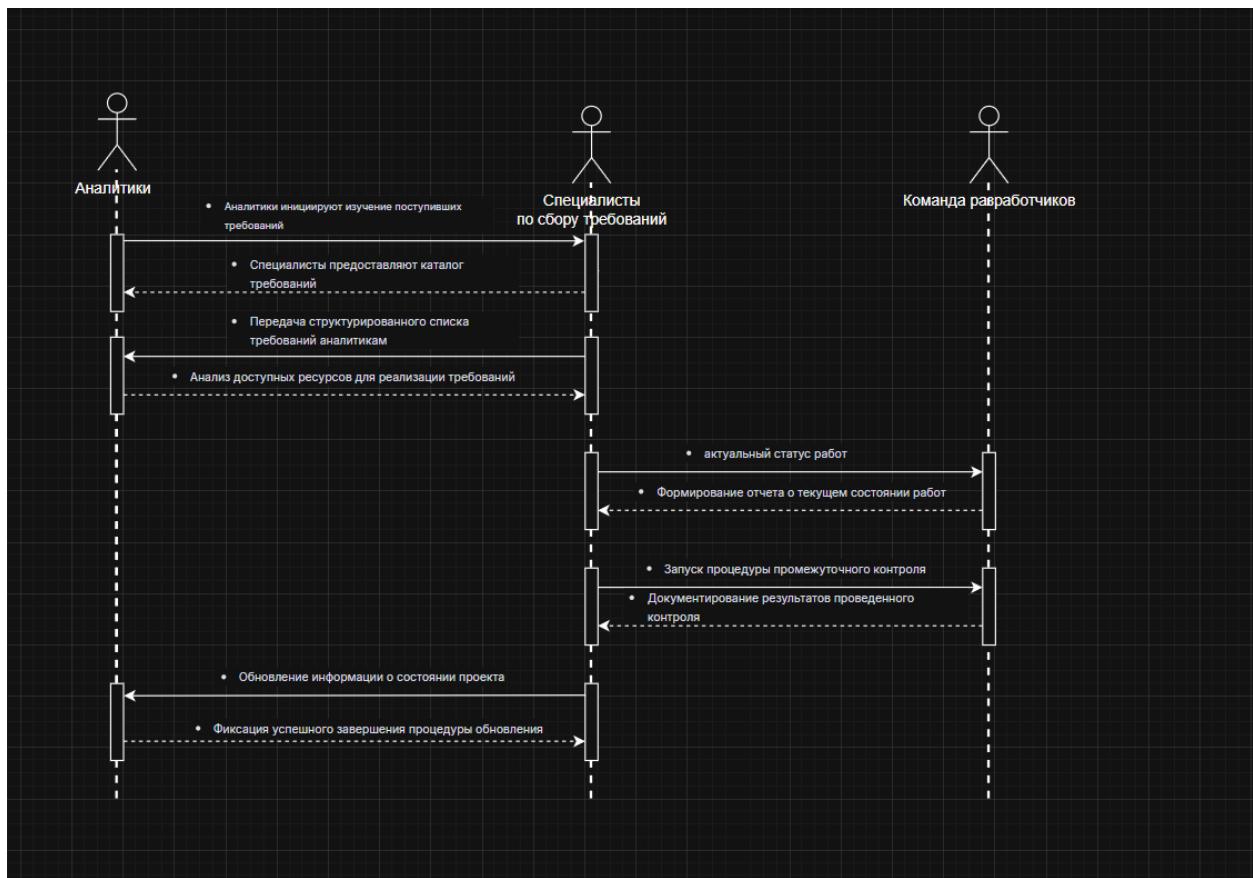


Рисунок 3 - Диаграмма последовательности

Список использованных источников и литературы:

1. Ахмедова, Х. Г. Обоснование и разработка требований к программным системам : учебное пособие / Х. Г. Ахмедова. — Москва : РТУ МИРЭА, 2023. — 104 с. — ISBN 978-5-7339-1934-8. — Текст : электронный // Лань : электронно-библиотечная система. — URL: <https://e.lanbook.com/book/382694> (дата обращения: 22.09.2025). — Режим доступа: для авториз. пользователей.
2. Завьялов, А. В. Диаграммы UML для анализа и проектирования информационных систем : учебно-методическое пособие / А. В. Завьялов. — Москва : РТУ МИРЭА, 2021. — 65 с. — Текст : электронный // Лань : электронно-библиотечная система. — URL: <https://e.lanbook.com/book/218630> (дата обращения: 22.09.2025). — Режим доступа: для авториз. пользователей.