

Санкт-Петербургский политехнический университет Петра Великого
Физико-механический институт

ОТЧЕТ ПО ЛАБОРАТОРНОЙ РАБОТЕ № 3
по дисциплине «Автоматизация научных исследований»
«Генерация диаграмм UML»

Работу выполнила
студентка группы 5040102/50201
Казакевич Анна Юрьевна

Преподаватель: Новиков Ф.А.

Санкт-Петербург
2025 г.

Содержание

1	Цель работы	2
2	Задание	2
3	Выбор темы и системы	2
3.1	Промпт для выбора предметной области и системы	2
3.2	Ответ ChatGPT 5	3
4	Промпты и ответы (ChatGPT 5)	3
4.1	ДИ — Диаграмма вариантов использования	3
4.1.1	Промпт 1 — ДИ	3
4.1.2	Промпт 2 — ДИ	5
4.1.3	Промпт 3 — ДИ	7
4.1.4	Промпт 4 — ДИ	9
5	Анализ результатов проектирования	12
6	Заключение	14

1 Цель работы

Разработать и проанализировать диаграмму UML (варианты использования и/или классов) для выбранной информационной системы. Использовать PlantUML для генерации, предоставить исходный код, изображения диаграмм и пояснения.

2 Задание

На основе требований из файла Task3.md:

- Часть 1а: Диаграмма вариантов использования — выбрать предметную область, определить действующих лиц, основные/вспомогательные варианты, отношения include/extend, границу системы, логическую целостность.
- Часть 1б: Диаграмма классов — определить основные классы, наследование, типы связей, атрибуты с типами данных, кратности ассоциаций.
- Требования: PlantUML, соответствие стандартам UML, логическая целостность, комментарии к ключевым элементам, проверка работоспособности кода.
- Форма отчёта: исходный код PlantUML, изображения диаграмм, пояснительная записка (предметная область, акторы, требования, структура классов), анализ результатов.

3 Выбор темы и системы

Тема и система будут выбраны с помощью запроса к нейросети (только ChatGPT 5), по аналогии с примером.

3.1 Промпт для выбора предметной области и системы

Запрос (ChatGPT 5):

Выбери информационную систему для проектирования - она должна соответствовать теме для научной работы (тоже выбери сам) для направления "01.04.02 «Математические методы анализа и визуализации данных»". Пропиши абзацем кратко про выбранную тему и абзацем про выбранную систему для проектирования. Под выбранной системой для проектирования понимается: что мы будем проектировать и для чего создавать диаграммы (например: Система управления библиотекой; Действующие лица: читатель, библиотекарь, администратор). Учти, что для этой системы необходимо будет в дальнейшем создать диаграмму классов и диаграмму вариантов использования, используя PlantUML. Пока просто напиши кратко выбранную тему, выбранную систему, которую проектируем и её действующие лица.

3.2 Ответ ChatGPT 5

Выбранная тема научной работы.

Тема научной работы: Математические методы анализа и визуализации многомерных данных в задачах обнаружения аномалий. В рамках темы рассматриваются методы статистического анализа, снижения размерности (PCA, t-SNE, UMAP), кластеризации и оценки плотности распределений для выявления аномалий в многомерных наборах данных. Особое внимание уделяется визуализации результатов анализа, позволяющей интерпретировать сложные структуры данных и закономерности, скрытые в высоких размерностях, что актуально для прикладных задач в инженерии, биомедицине и обработке изображений.

Выбранная информационная система для проектирования.

Проектируемая система — Информационная система анализа и визуализации многомерных данных. Система предназначена для загрузки наборов данных, применения математических методов анализа (предобработка, снижение размерности, кластеризация, поиск аномалий) и интерактивной визуализации результатов в виде графиков, проекций и тепловых карт. Диаграммы используются для формализации структуры системы и сценариев её использования. Основные действующие лица системы: исследователь, аналитик данных, администратор системы.

4 Промпты и ответы (ChatGPT 5)

4.1 ДИ — Диаграмма вариантов использования

4.1.1 Промпт 1 — ДИ

Запрос:

Сгенерируй PlantUML код для диаграммы вариантов использования выбранной системы. Укажи действующие лица, перечисли основные случаи использования. Добавь границу системы, связи акторов с вариантами использования, и отношения include/extend там, где это логично.

Ответ ChatGPT 5:

PlantUML код приведён ниже.

```
@startuml
left to right direction

actor "" as Researcher
actor " " as Analyst
actor " " as Admin

rectangle " \n " {

    usecase " " as UC_Load
    usecase " " as UC_Preprocess
    usecase " " as UC_Method
    usecase " " as UC_DimRed
    usecase " " as UC_Cluster
    usecase " " as UC_Anomaly
```

```

usecase " " as UC_Visual
usecase " \n" as UC_Params
usecase " \n" as UC_Compare
usecase " \n" as UC_Save

usecase " " as UC_Users
usecase " \n" as UC_Resources
}

Researcher --> UC_Load
Researcher --> UC_Preprocess
Researcher --> UC_Method
Researcher --> UC_DimRed
Researcher --> UC_Anomaly
Researcher --> UC_Visual
Researcher --> UC_Save

Analyst --> UC_Load
Analyst --> UC_Preprocess
Analyst --> UC_Method
Analyst --> UC_Cluster
Analyst --> UC_Compare
Analyst --> UC_Visual
Analyst --> UC_Save

Admin --> UC_Users
Admin --> UC_Resources

UC_Method --> UC_Params : <<include>>
UC_DimRed --> UC_Preprocess : <<include>>
UC_Cluster --> UC_Preprocess : <<include>>
UC_Anomaly --> UC_Preprocess : <<include>>

UC_Visual ..> UC_Compare : <<extend>>
UC_Visual ..> UC_Anomaly : <<extend>>

@enduml

```

Диаграмма:

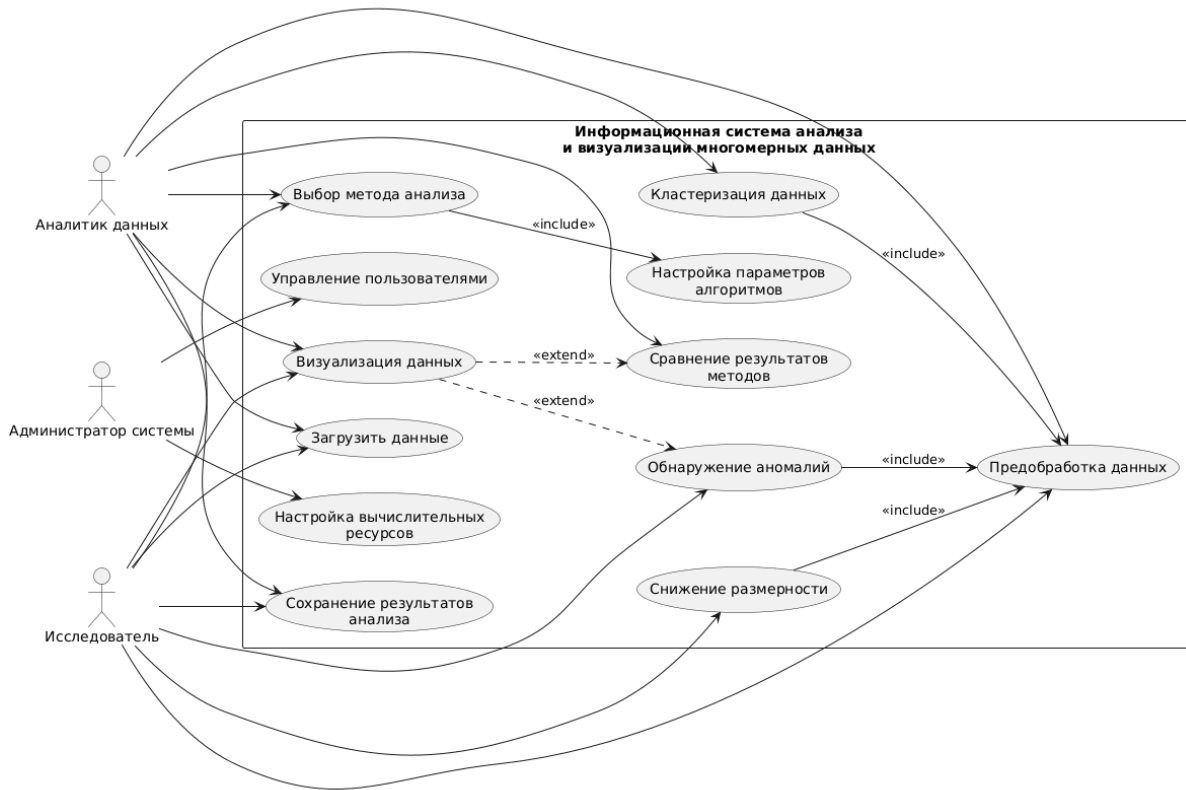


Рис. 1: Диаграмма вариантов использования — ДИ1

4.1.2 Промпт 2 — ДИ

Запрос:

Сгенерируй PlantUML код для диаграммы вариантов использования выбранной системы. Перечисли случаи использования по ролям (акторам). Покажи связи между действующими лицами и вариантами использования. Включи границу системы и отношения include между зависимыми случаями.

Ответ ChatGPT 5:

PlantUML код приведён ниже.

```
@startuml
left to right direction

actor "" as Researcher
actor "" as Analyst
actor "" as Admin

rectangle " \n " {

    package " " {
        usecase "" as R_Load
        usecase "" as R_Preprocess
        usecase "" as R_Method
        usecase "" as R_DimRed
    }
}
```

```

    usecase " " as R_Anomaly
    usecase " " as R_Visual
    usecase " " as R_Save
}

package " " {
    usecase " " as A_Load
    usecase " " as A_Preprocess
    usecase " \n " as A_Method
    usecase " " as A_Cluster
    usecase " \n" as A_Compare
    usecase " " as A_Visual
    usecase " " as A_Save
}

package " " {
    usecase " " as Ad_Users
    usecase " \n " as Ad_Roles
    usecase " \n" as Ad_Resources
}
}

Researcher --> R_Load
Researcher --> R_Preprocess
Researcher --> R_Method
Researcher --> R_DimRed
Researcher --> R_Anomaly
Researcher --> R_Visual
Researcher --> R_Save

Analyst --> A_Load
Analyst --> A_Preprocess
Analyst --> A_Method
Analyst --> A_Cluster
Analyst --> A_Compare
Analyst --> A_Visual
Analyst --> A_Save

Admin --> Ad_Users
Admin --> Ad_Roles
Admin --> Ad_Resources

R_Method --> R_Preprocess : <<include>>
R_DimRed --> R_Preprocess : <<include>>
R_Anomaly --> R_Preprocess : <<include>>

A_Method --> A_Preprocess : <<include>>
A_Cluster --> A_Preprocess : <<include>>
A_Compare --> A_Method : <<include>>

@enduml

```

Диаграмма:

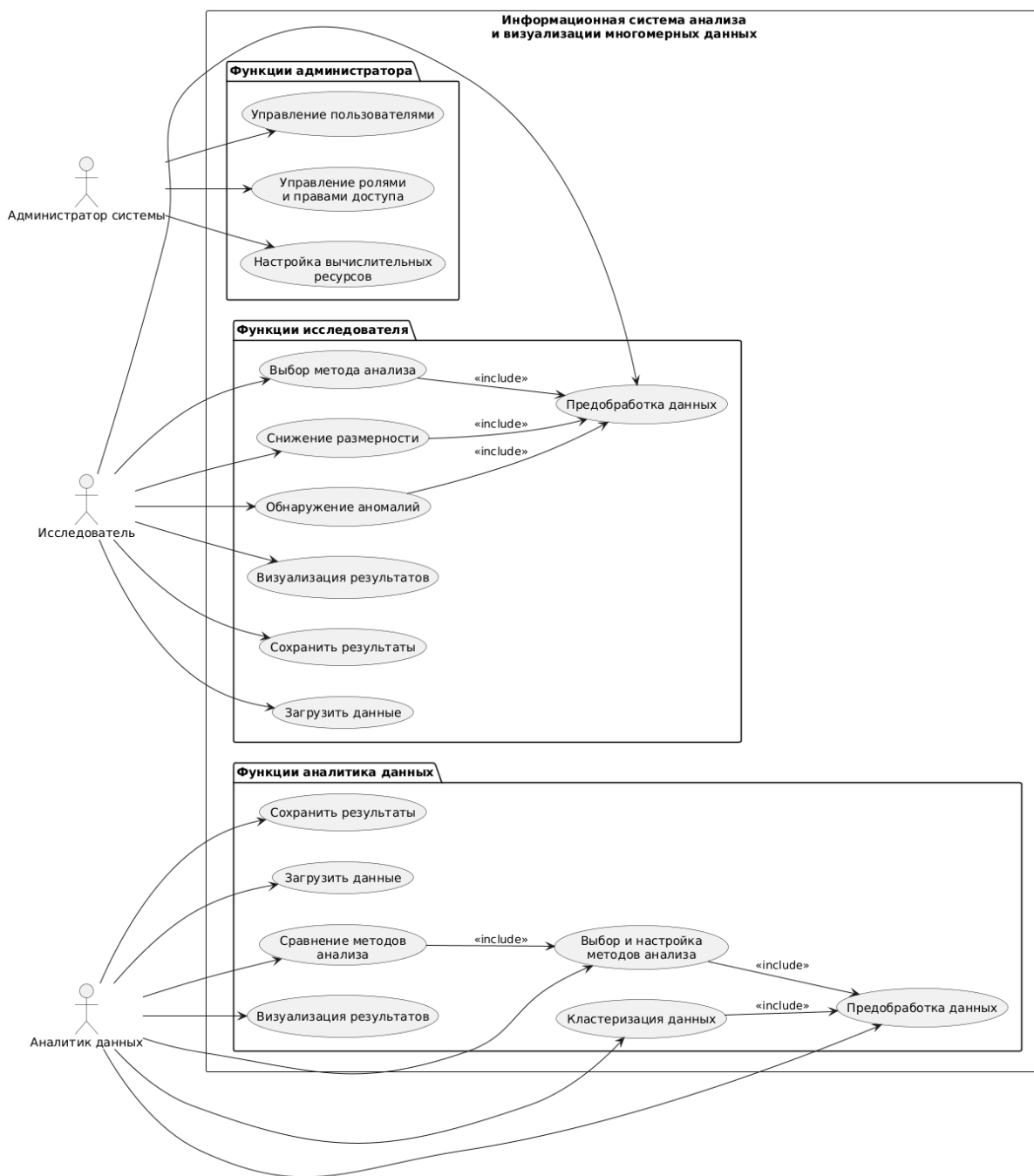


Рис. 2: Диаграмма вариантов использования — ДИ2

4.1.3 Промпт 3 — ДИ

Запрос:

Сгенерируй PlantUML код для диаграммы вариантов использования. Укажи действующие лица и перечисли случаи использования по ролям: для каждого актора — соответствующие действия. Отобрази все связи и границу системы.

Ответ ChatGPT 5:

PlantUML код приведён ниже.

```
@startuml
left to right direction

actor "" as Researcher
actor "" as Analyst
actor "" as Admin

rectangle "" \n "" {

    usecase "" as UC_Load
    usecase "" as UC_Preprocess
    usecase "" as UC_Method
    usecase "" \n "" as UC_Params
    usecase "" as UC_DimRed
    usecase "" as UC_Cluster
    usecase "" as UC_Anomaly
    usecase "" as UC_Visual
    usecase "" \n "" as UC_Compare
    usecase "" as UC_Save

    usecase "" as UC_Users
    usecase "" \n "" as UC_Roles
    usecase "" \n "" as UC_Resources
}

Researcher --> UC_Load
Researcher --> UC_Preprocess
Researcher --> UC_Method
Researcher --> UC_DimRed
Researcher --> UC_Anomaly
Researcher --> UC_Visual
Researcher --> UC_Save

Analyst --> UC_Load
Analyst --> UC_Preprocess
Analyst --> UC_Method
Analyst --> UC_Params
Analyst --> UC_Cluster
Analyst --> UC_Compare
Analyst --> UC_Visual
Analyst --> UC_Save

Admin --> UC_Users
Admin --> UC_Roles
Admin --> UC_Resources

UC_Method --> UC_Params : <<include>>
UC_DimRed --> UC_Preprocess : <<include>>
UC_Cluster --> UC_Preprocess : <<include>>
UC_Anomaly --> UC_Preprocess : <<include>>
UC_Compare --> UC_Method : <<include>>

@enduml
```

Диаграмма:

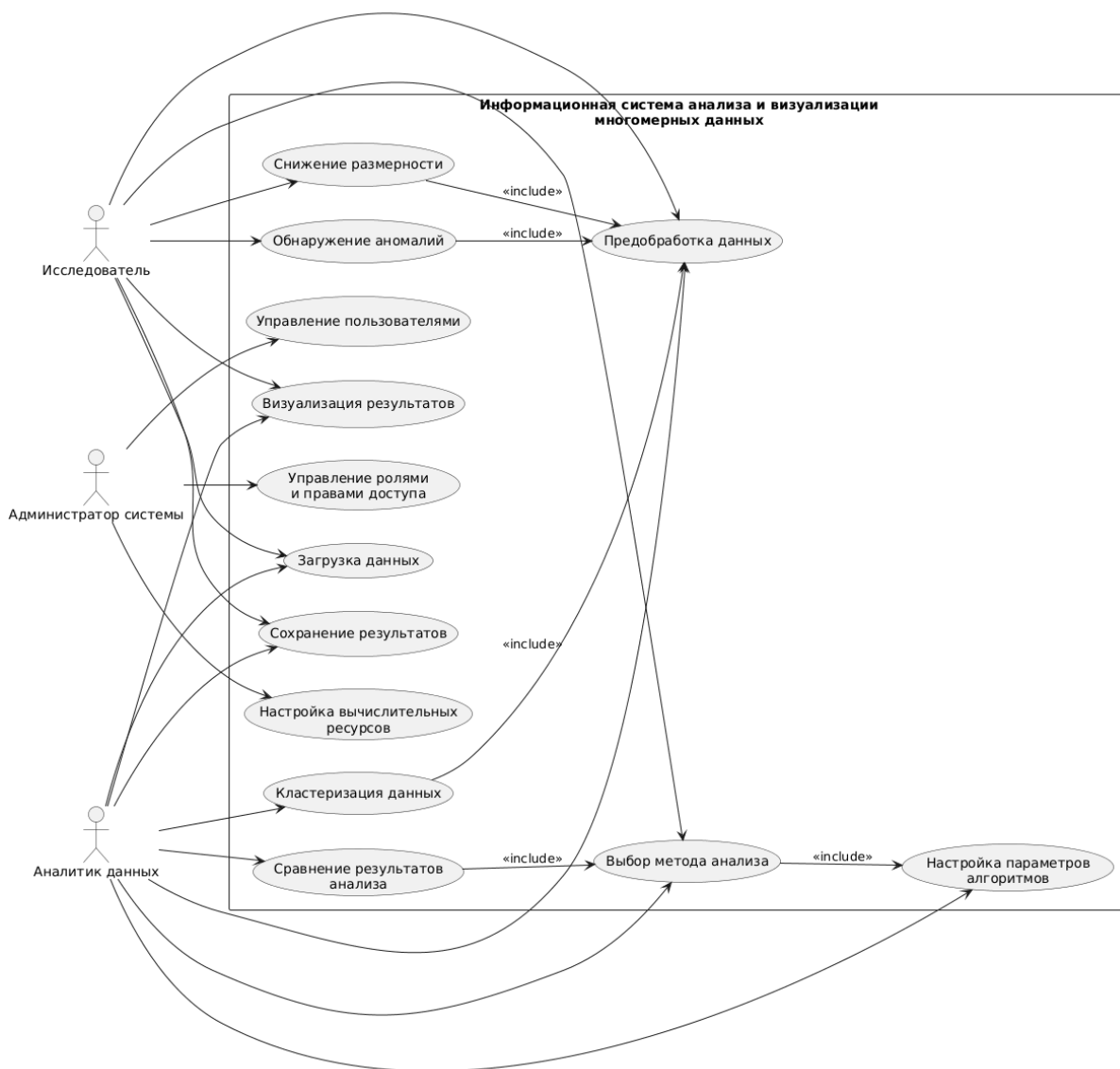


Рис. 3: Диаграмма вариантов использования — ДИЗ

4.1.4 Промпт 4 — ДИ

Запрос:

Сгенерируй PlantUML код для диаграммы вариантов использования выбранной системы, по аналогии с заданием для библиотеки: распиши варианты для каждой роли; включи include/extend там, где уместно; отобрази наследование ролей (если уместно), границу системы и все связи.

Ответ ChatGPT 5:

PlantUML код приведён ниже.

@startuml
left to right direction

```

' =====
actor " " as User
actor "" as Researcher
actor " " as Analyst
actor " " as Admin

User <|-- Researcher
User <|-- Analyst
User <|-- Admin

' =====
rectangle " \n " {

' --- ---
usecase "\n" as UC_Login
usecase " " as UC_Load
usecase " " as UC_View
usecase " " as UC_Save

' --- ---
usecase " " as UC_Preprocess
usecase " " as UC_Method
usecase " \n" as UC_Params
usecase " " as UC_DimRed
usecase " " as UC_Cluster
usecase " " as UC_Anomaly

' --- ---
usecase " " as UC_Visual
usecase " \n" as UC_Compare

' --- ---
usecase " " as UC_Users
usecase " \n " as UC_Roles
usecase " \n" as UC_Resources
}

' =====
User --> UC_Login
User --> UC_Load
User --> UC_View
User --> UC_Save

Researcher --> UC_Preprocess
Researcher --> UC_Method
Researcher --> UC_DimRed
Researcher --> UC_Anomaly
Researcher --> UC_Visual

Analyst --> UC_Preprocess
Analyst --> UC_Method
Analyst --> UC_Params
Analyst --> UC_Cluster
Analyst --> UC_Compare
Analyst --> UC_Visual

```

```
Admin --> UC_Users
Admin --> UC_Roles
Admin --> UC_Resources

' ===== include / extend =====
UC_Method --> UC_Params : <<include>>
UC_DimRed --> UC_Preprocess : <<include>>
UC_Cluster --> UC_Preprocess : <<include>>
UC_Anomaly --> UC_Preprocess : <<include>>

UC_Compare ..> UC_Visual : <<extend>>

@enduml
```

Диаграмма:



Рис. 4: Диаграмма вариантов использования — ДИ4

5 Анализ результатов проектирования

Целью анализа является проверка качества построенных диаграмм с точки зрения синтаксиса PlantUML, соответствия стандартам UML, полноты требований и логической целостности модели.

Корректность синтаксиса PlantUML

Код диаграмм успешно валидируется в редакторе PlantUML (онлайн/локально), ключевые конструкции распознаны корректно. Использованы базовые элементы: actor, usecase, прямоугольник границы системы для ДИ; class, ассоциации, агрегация/композиция, наследование для ДК. Длинные строки в коде корректно переносятся за счёт настроек окружения листингов.

Соответствие стандартам UML

Диаграмма вариантов использования содержит явные акторы, границу системы, основные и вспомогательные случаи использования, а также отношения include/extend там, где это уместно. Диаграмма классов отражает основные сущности домена, выделяет устойчивые связи (ассоциации), композиционные зависимости для неотделимых частей и иерархии наследования для ролей/обобщений. Обозначения кратностей применены для ключевых связей.

Полнота отображения функциональных требований

Сценарии ДИ покрывают ключевые пользовательские цели и опорные подпроцессы. Для ДК представлены атрибуты и отношения, достаточные для реализации базовой логики домена. При необходимости детализация может быть расширена (например, уточнение типов полей, добавление сервисных классов, перечислений статусов и ассоциативных классов для отношений многие-к-многим).

Качество визуализации

Диаграммы читаемы: акторы и случаи логически сгруппированы, связи не перегружены, подписи отношений лаконичны. Для ДК соблюден баланс между детализацией атрибутов и визуальной компактностью; где необходимо, применены кратности и типы связей, повышающие выразительность.

Логическая целостность и обоснованность домена

Выбранная предметная область и роли акторов взаимно согласованы: границы ответственности акторов на ДИ соответствуют зонам ответственности классов на ДК. Уточняющие отношения (include/extend) не противоречат основной логике процессов. Структура классов поддерживает основные сценарии использования.

Вывод по анализу

По совокупности критериев диаграммы соответствуют целям лабораторной работы: корректный синтаксис PlantUML, соблюдение основных нотаций UML, покрытие ключевых требований и логическая непротиворечивость модели. Рекомендуемая область точечной доработки — расширение атрибутов доменных сущностей и уточнение кратностей по мере детализации требований.

Анализ по промптам (ДИ)

Ниже приведена оценка для каждой диаграммы вариантов использования (ДИ1–ДИ4) по критериям из задания.

ДИ1. Хорошая базовая полнота (загрузка, предобработка, методы, снижение размерности, кластеризация, аномалии, визуализация, сохранение), присутствуют include/extend. Возможна двусмысленность в расширении визуализации через сравнение/аномалии. Итог: высокая корректность синтаксиса и покрытия, небольшие замечания к семантике отношений extend.

ДИ2. Структурирование по пакетам ролей повышает читаемость и прослеживаемость ответственности. Include-отношения логичны (предобработка как зависимость, сравнение зависит от выбора методов). Итог: сбалансированная диаграмма, хорошая визуальная организация.

ДИ3. Компактная диаграмма без пакетирования, но с чётким набором зависимостей. Добавлена явная настройка параметров для аналитика. Итог: корректная, чуть менее структурирована визуально по ролям по сравнению с ДИ2.

ДИ4. Добавлено наследование акторов от базового пользователя, чётко разделены общие и специфические варианты использования, корректно применены include/extend. Итог: наибольшая выразительность и полнота из представленных ДИ.

Сводная таблица оценок (ДИ)

Баллы по шкале 0–10 (0 — полное несоответствие, 10 — отличное соответствие).

extbfДиаграмма	Кр.1 (синтакс.)	Кр.2 (UML)	Кр.3 (полнота)	Кр.4 (визуал.)	Кр.5 (логика)	Кр.6 (обосн.)	Средн.
ДИ1	9	8	8	8	8	9	8.3
ДИ2	9	9	8	9	8	9	8.7
ДИ3	9	8	8	8	8	9	8.3
ДИ4	9	9	9	9	9	9	9.0

Примечание: оценки носят обоснованный характер исходя из текущего кода PlantUML и структуры диаграмм; при изменении кода/домена целесообразно пересчитать баллы.

6 Заключение

В ходе работы подготовлены исходные коды PlantUML и визуализации для диаграммы вариантов использования и диаграммы классов выбранной информационной системы. Диаграммы продемонстрировали соответствие базовым стандартам UML, логическую согласованность между пользовательскими сценариями и структурой данных, а также достаточную полноту для дальнейшей детализации.