

Санкт-Петербургский политехнический университет  
Петра Великого  
Физико-механический институт  
Высшая школа прикладной математики и вычислительной физики

Отчет по лабораторной работе №3  
по дисциплине  
"Автоматизация научных исследований"

**Генерация UML-диаграмм по описанию**

Выполнил студент:

Бабахина Софья Александровна

Группа: 5040102/50201

Санкт-Петербург  
2025

# Содержание

<b>1</b>	<b>Цель работы</b>	<b>2</b>
<b>2</b>	<b>Задание</b>	<b>2</b>
2.1	Часть 1. Диаграмма вариантов использования . . . . .	2
2.2	Часть 2. Диаграмма классов . . . . .	2
2.3	Требования к выполнению . . . . .	2
2.4	Критерии оценки . . . . .	2
<b>3</b>	<b>Результаты</b>	<b>3</b>
3.1	Результаты промпта №1 . . . . .	4
3.2	Результаты промпта №2 . . . . .	9
3.3	Результаты промпта №3 . . . . .	13
3.4	Результаты промпта №4 (личный) . . . . .	18
<b>4</b>	<b>Сравнение результатов и выводы</b>	<b>24</b>
4.1	Сравнительный анализ результатов . . . . .	24

# 1 Цель работы

Разработать и проанализировать диаграммы UML для информационной системы по выбору студента.

## 2 Задание

### 2.1 Часть 1. Диаграмма вариантов использования

- Выбрать предметную область для проектирования информационной системы
- Определить основных действующих лиц системы
- Выявить основные функциональные требования
- Построить диаграмму вариантов использования с учетом следующих требований:
  1. Отобразить все действующие лица
  2. Определить основные и вспомогательные варианты использования
  3. Установить отношения между вариантами использования (include, extend)
  4. Добавить границу системы
  5. Обеспечить логическую целостность диаграммы

### 2.2 Часть 2. Диаграмма классов

- На основе выбранной предметной области разработать диаграмму классов
- Определить основные классы системы
- Установить иерархию наследования
- Определить типы связей между классами
- Указать атрибуты классов с типами данных
- Добавить кратности ассоциаций

### 2.3 Требования к выполнению

- Использовать PlantUML для генерации диаграмм
- Соблюдать стандарты UML
- Обеспечить логическую целостность диаграмм
- Предоставить комментарии к ключевым элементам диаграмм
- Проверить работоспособность сгенерированного кода
- Провести исследование для 1-2 моделей ИИ

### 2.4 Критерии оценки

- Корректность синтаксиса PlantUML
- Соответствие стандартам UML
- Полнота отображения функциональных требований
- Качество визуализации
- Логическая целостность диаграмм

### 3 Результаты

Для исследования была выбрана диаграмма вариантов использования. Для демонстрации результатов исследования обратимся к анализу системы поиска мест и мероприятий. Система для генерации UML диаграммы вариантов использования включает следующие действующие лица:

- **Пользователь** – Человек использующий сервис для поиска мест и мероприятий. Имеет общие права зарегистрированного и незарегистрированного пользователей: доступ в каталог и поиск по сайту.
- **Незарегистрированный пользователь** – пользователь, которому доступна регистрация на сайте (помимо действий, разрешенных пользователю п. 1.1)
- **Зарегистрированный пользователь** – пользователь, создавший личный кабинет (не доступна регистрация). Доступны права пользователя, а также аутентификация, добавления места/мероприятия в избранное, просмотр персональных рекомендаций и получение напоминаний.
- **Владелец места / организатор мероприятия** – зарегистрированный пользователь, который имеет право на добавление информации о месте/событии в базу сервиса для продвижения и привлечения аудитории.
- **Редакция (администраторы/модераторы)** – команда людей, не являющихся формальными пользователями, формирующая подборки, проверяющая контент, поступающие в сервис заявки, и следящая за качеством информации в сервисе.

Исходный вариант диаграммы, сгенерированный вручную:

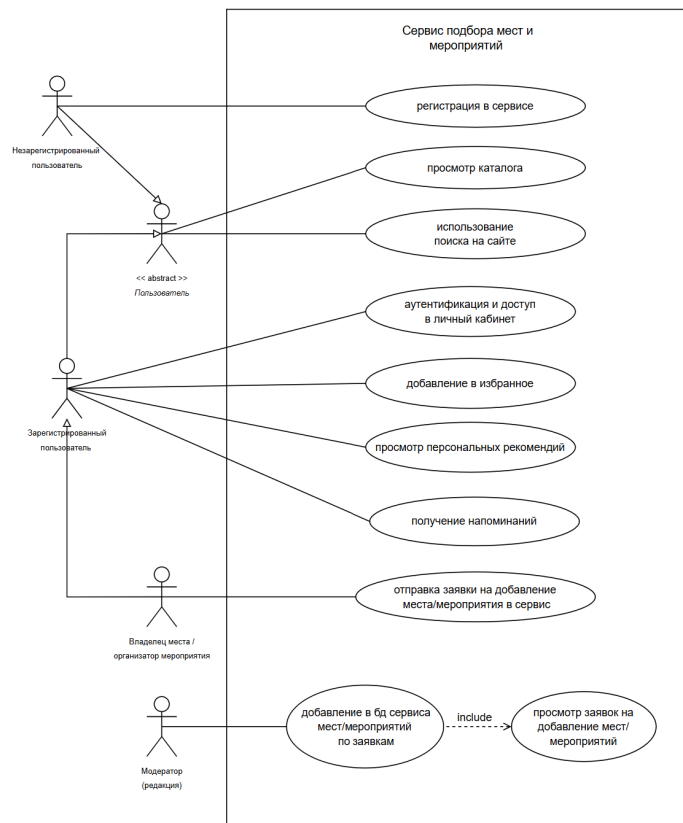


Рис. 1: Диаграмма вариантов использования

### 3.1 Результаты промпта №1

Сгенерируй PlantUML код для диаграммы вариантов использования системы поиска мест и мероприятий. Укажи действующие лица, перечисли основные случаи использования.

DeepSeek

```
1 @startuml
2 !theme plain
3
4 left to right direction
5
6 actor "Турист" as User
7 actor "Зарегистрированный Пользователь" as AuthUser
8 actor "Администратор" as Admin
9 actor "Владелец Места" as Owner
10
11 rectangle "Система поиска мест и мероприятий" {
12
13     (Поиск мест и мероприятий) as Search
14     (Просмотр деталей места) as ViewDetails
15     (Просмотр отзывов и рейтингов) as ViewReviews
16     (Фильтрация и сортировка результатов) as Filter
17
18     (Регистрация) as Register
19     (Авторизация) as Login
20
21     (Добавление места/мероприятия) as AddPlace
22     (Редактирование места/мероприятия) as EditPlace
23     (Управление своими местами) as ManagePlaces
24
25     (Добавление отзыва) as AddReview
26     (Добавление в избранное) as AddToFav
27     (Просмотр избранного) as ViewFav
28
29     (Модерация контента) as Moderate
30     (Управление пользователями) as ManageUsers
31     (Управление категориями) as ManageCategories
32
33 }
34
35 ' Связи акторов с use case
36 User --> Search
37 User --> ViewDetails
38 User --> ViewReviews
39 User --> Filter
40
41 User --> Register
42 User --> Login
43
44 AuthUser --> AddReview
45 AuthUser --> AddToFav
46 AuthUser --> ViewFav
47 AuthUser --> AddPlace
48 AuthUser --> ManagePlaces
49
50 Owner --|> AuthUser
51 Owner --> EditPlace
52
53 Admin --> Moderate
```

```

54 Admin --> ManageUsers
55 Admin --> ManageCategories
56
57 ' Связи между use case (include/extend)
58 Search .> ViewDetails : include
59 ViewDetails .> ViewReviews : include
60 Search .> Filter : include
61
62 @enduml

```

Листинг 1: Код диаграммы вариантов использования (сгенерирован DeepSeek по промпту 1)



Действующие лица (**Акторы**):

- **Турист** — неавторизованный пользователь, который ищет информацию.

- **Зарегистрированный Пользователь** — пользователь с учетной записью, который может взаимодействовать с системой более активно.
- **Владелец Места** — частный случай Зарегистрированного пользователя, который может управлять информацией о своих местах.
- **Администратор** — отвечает за модерацию и общее управление системой.

#### Основные случаи использования (Use Case):

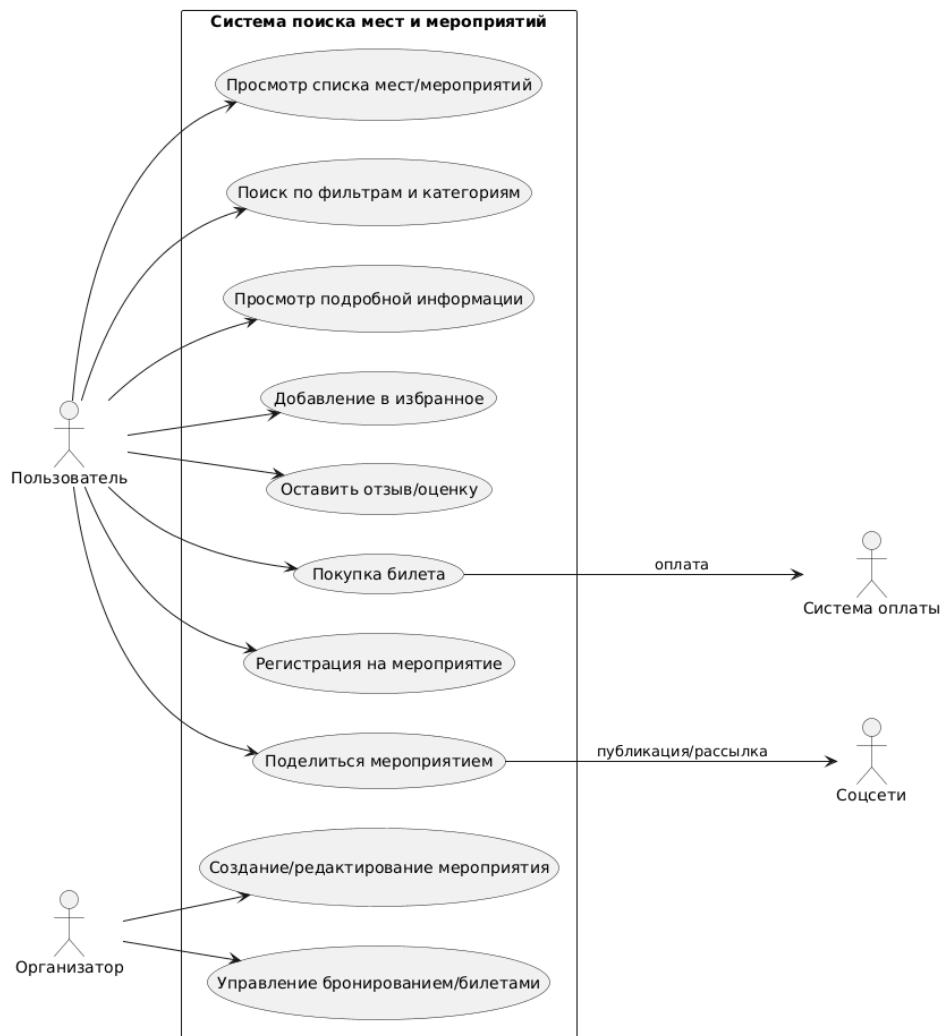
- **Для всех пользователей (Турист +):**
  - Поиск мест и мероприятий (основная функция системы)
  - Просмотр деталей места (адрес, описание, фото, контакты)
  - Просмотр отзывов и рейтингов
  - Фильтрация и сортировка результатов (по местоположению, типу, рейтингу и т.д.)
  - Регистрация и Авторизация
- **Для Зарегистрированных пользователей:**
  - Добавление отзыва и оценки к месту
  - Добавление в избранное для быстрого доступа позже
  - Просмотр избранного
  - Добавление нового места/мероприятия в систему
- **Для Владельцев мест:**
  - Управление своими местами (просмотр списка)
  - Редактирование места/мероприятия (обновление информации, добавление фото)
- **Для Администратора:**
  - Модерация контента (проверка и одобрение новых мест, удаление отзывов)
  - Управление пользователями (блокировка, удаление)
  - Управление категориями (добавление, редактирование категорий и тегов для мест)

Данная диаграмма создаёт четкую и наглядную структуру, показывающую взаимодействие между пользователями и системой.

## ChatGPT

```
1 @startuml
2 left to right direction
3
4 actor "Пользователь" as User
5 actor "Организатор" as Organizer
6 actor "Система оплаты" as Payment
7 actor "Соцсети" as Social
8
9 rectangle "Система поиска мест и мероприятий" as System {
10     usecase "Просмотр списка мест/мероприятий" as UC1
11     usecase "Поиск по фильтрам и категориям" as UC2
12     usecase "Просмотр подробной информации" as UC3
13     usecase "Добавление в избранное" as UC4
14     usecase "Оставить отзыв/оценку" as UC5
15     usecase "Покупка билета" as UC6
16     usecase "Регистрация на мероприятие" as UC7
17     usecase "Поделиться мероприятием" as UC8
18
19     usecase "Создание/редактирование мероприятия" as UC9
20     usecase "Управление бронированием/билетами" as UC10
21 }
22
23 User --> UC1
24 User --> UC2
25 User --> UC3
26 User --> UC4
27 User --> UC5
28 User --> UC6
29 User --> UC7
30 User --> UC8
31
32 Organizer --> UC9
33 Organizer --> UC10
34
35 UC6 --> Payment : "оплата"
36
37 UC8 --> Social : "публикация/рассылка"
38
39 @enduml
```

Листинг 2: Код диаграммы вариантов использования (сгенерирован ChatGPT по промπτу 1)



#### Основные действующие лица:

- **Пользователь** — ищет мероприятия и места.
- **Организатор** — добавляет и управляет событиями.
- **Система оплаты** — внешняя система.
- **Соцсети** — внешний сервис для публикаций.

#### Основные варианты использования:

- Поиск и просмотр мест/мероприятий.
- Просмотр подробностей.
- Добавление в избранное.
- Покупка билетов / регистрация.
- Оставление отзывов.
- Создание и управление мероприятиями (для организатора).
- Поделиться событием.

## 3.2 Результаты промпта №2

Сгенерируй PlantUML код для диаграммы вариантов использования системы подбора мест и мероприятий. Укажи действующие лица: Пользователь, Незарегистрированный пользователь, Зарегистрированный пользователь, Владелец места / организатор мероприятия, Редакция (администраторы). Перечисли основные случаи использования такие, как регистрация в сервисе, просмотр каталога, использование поиска на сайте, аутентификация и доступ в личный кабинет, добавление в избранное, просмотр персональных рекомендаций, получение напоминаний, отправка заявки на добавление места/мероприятия в сервис, добавление в бд сервиса мест/мероприятий по заявкам, просмотр заявок на добавление мест/мероприятий др.

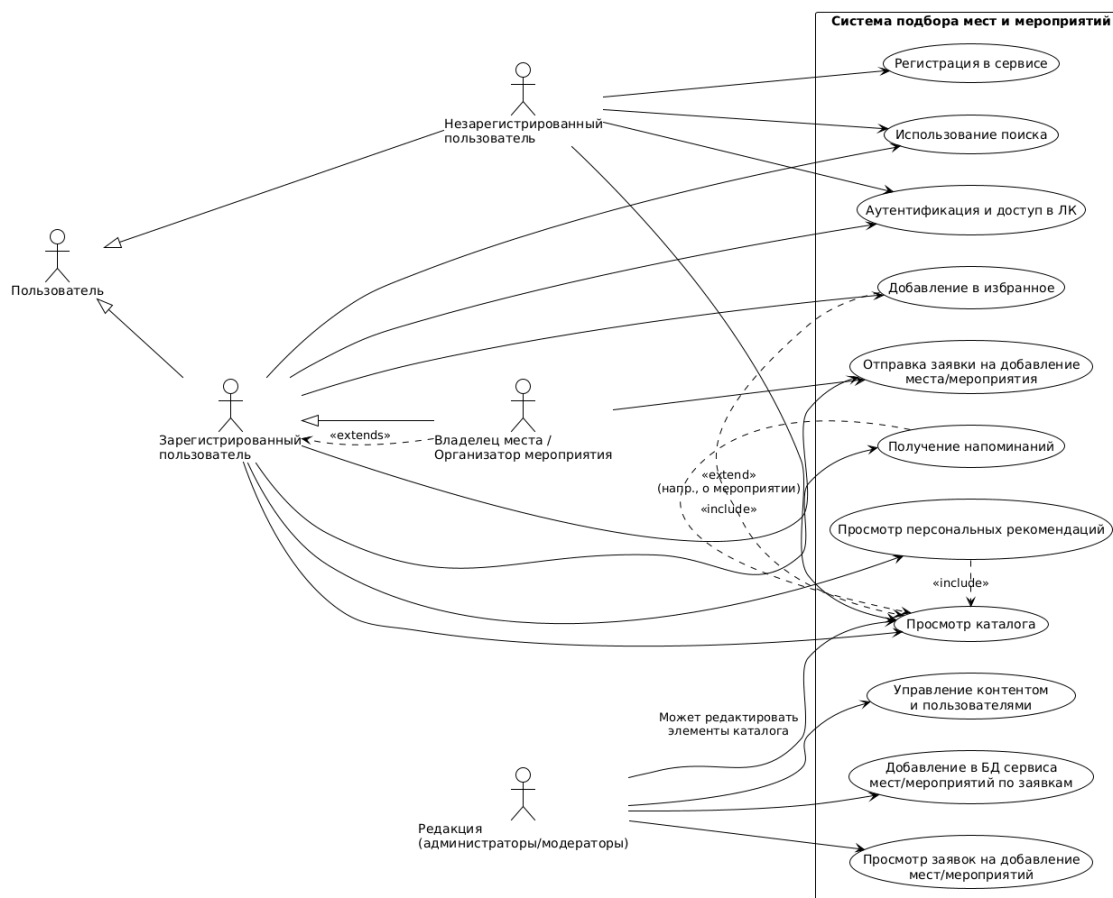
### DeepSeek

```
1      @startuml
2      !theme plain
3
4      left to right direction
5
6      actor "Пользователь" as User
7      actor "Незарегистрированный\пользователь" as Guest
8      actor "Зарегистрированный\пользователь" as RegUser
9      actor "Владелец места /\nОрганизатор мероприятия" as Owner
10     actor "Редакция\n(администраторы/модераторы)" as Editor
11
12     User <|-- Guest
13     User <|-- RegUser
14     RegUser <|-- Owner
15
16     rectangle "Система подбора мест и мероприятий" {
17         (Просмотр каталога) as ViewCatalog
18         (Использование поиска) as Search
19         (Аутентификация и доступ в ЛК) as Auth
20         (Регистрация в сервисе) as Register
21         (Добавление в избранное) as AddToFav
22         (Просмотр персональны рекомендаций) as ViewRecs
23         (Получение напоминаний) as GetReminders
24         (Отправка заявки на добавление\nместа/мероприятия) as SubmitRequest
25         (Добавление в БД сервиса\nмест/мероприятий по заявкам) as AddToDB
26         (Просмотр заявок на добавление\nмест/мероприятий) as ReviewRequests
27         (Управление контентом\nи пользователями) as Manage
28     }
29
30     Guest --> ViewCatalog
31     Guest --> Search
32     Guest --> Register
33     Guest --> Auth
34
35     RegUser --> Auth
36     RegUser --> ViewCatalog
37     RegUser --> Search
38     RegUser --> AddToFav
39     RegUser --> ViewRecs
40     RegUser --> GetReminders
41     RegUser --> SubmitRequest
42
43     Owner --> SubmitRequest
44     Owner ..> RegUser : <<extends>>
45
46     Editor --> ReviewRequests
47     Editor --> AddToDB
```

```

48 Editor --> Manage
49 Editor --> ViewCatalog : Может редактировать\пэлементы каталога
50
51 ' Связи для включения (include)
52 ViewRecs .> ViewCatalog : <<include>>
53 AddToFav .> ViewCatalog : <<include>>
54
55 ' Связи для расширения (extend)
56 GetReminders .> ViewCatalog : <<extend>>\n(напр., о мероприятии)
57
58 @enduml

```



## ChatGPT

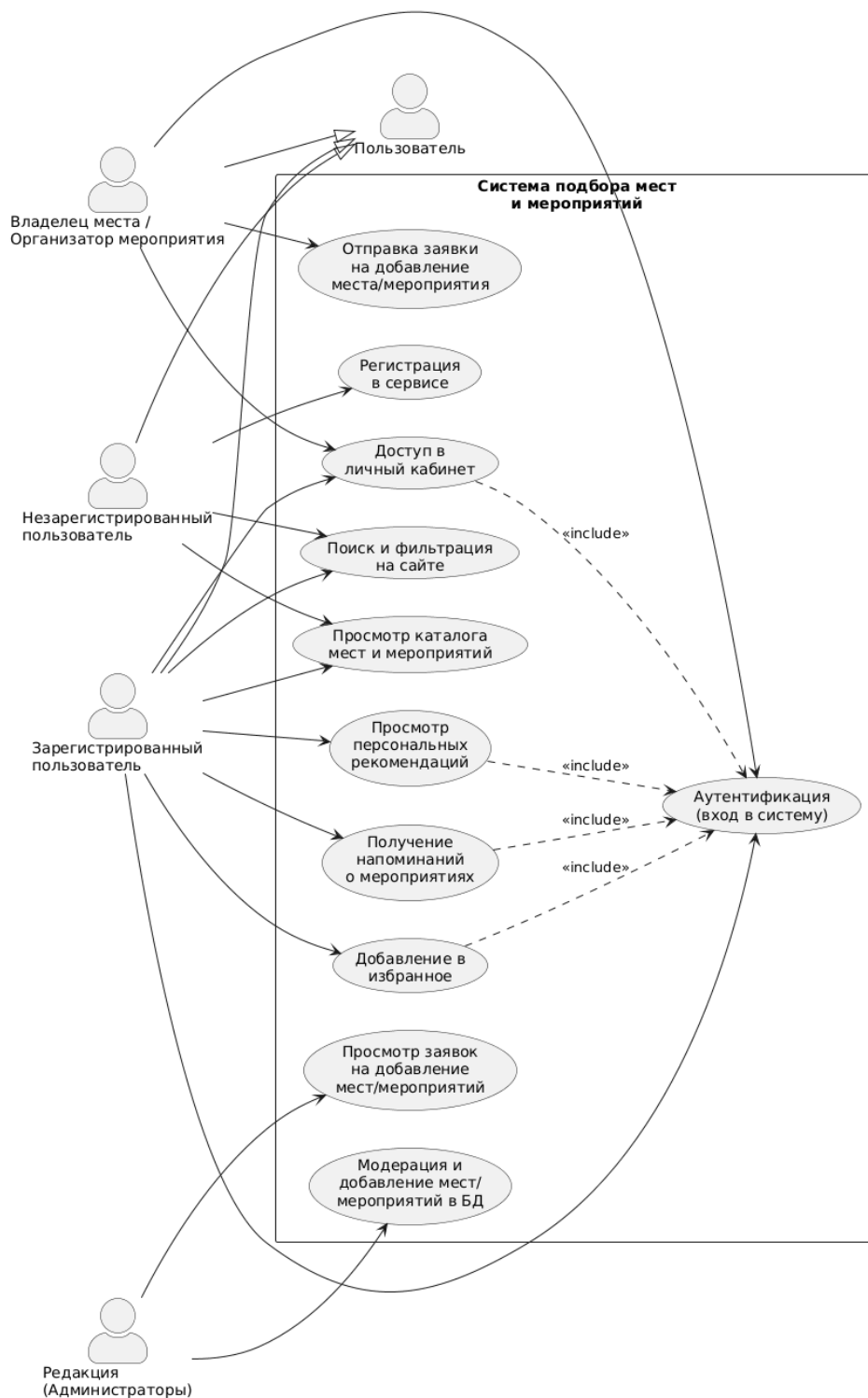
```
1 @startuml
2 left to right direction
3 skinparam actorStyle awesome
4
5 ' ==== Акторы ====
6 actor "Пользователь" as User
7 actor "Незарегистрированный\пользователь" as Guest
8 actor "Зарегистрированный\пользователь" as Registered
9 actor "Владелец места /\nОрганизатор мероприятия" as Owner
10 actor "Редакция\n(Администраторы)" as Admin
11
12 Guest --|> User
13 Registered --|> User
14 Owner --|> User
15
16 ' ==== Варианты использования ====
17 rectangle "Система подбора мест\ни мероприятий" {
18
19     usecase "Просмотр каталога\nмест и мероприятий" as UC_Catalog
20     usecase "Поиск и фильтрация\nна сайте" as UC_Search
21     usecase "Регистрация\nв сервисе" as UC_Register
22     usecase "Аутентификация\n(в од в систему)" as UC_Login
23     usecase "Доступ в\nличный кабинет" as UC_Profile
24     usecase "Добавление в\nизбранное" as UC_Favorites
25     usecase "Просмотр\nперсональны \nрекомендаций" as UC_Recommendations
26     usecase "Получение\nнапоминаний\nпо мероприятия " as UC_Notifications
27
28     usecase "Отправка заявки\nна добавление\nместа/мероприятия" as
        UC_SubmitRequest
29     usecase "Просмотр заявок\nна добавление\nмест/мероприятий" as
        UC_ViewRequests
30     usecase "Модерация и\nдобавление мест/\nмероприятий в БД" as UC_AddToDB
31 }
32
33 ' ==== Связи акторов с вариантами использования ====
34
35 Guest --> UC_Catalog
36 Guest --> UC_Search
37 Guest --> UC_Register
38
39 Registered --> UC_Login
40 Registered --> UC_Profile
41 Registered --> UC_Catalog
42 Registered --> UC_Search
43 Registered --> UC_Favorites
44 Registered --> UC_Recommendations
45 Registered --> UC_Notifications
46
47 Owner --> UC_Login
48 Owner --> UC_Profile
49 Owner --> UC_SubmitRequest
50
51 Admin --> UC_ViewRequests
52 Admin --> UC_AddToDB
53
54 ' ==== Зависимости ====
55 UC_Profile ..> UC_Login : <<include>>
56 UC_Favorites ..> UC_Login : <<include>>
57 UC_Recommendations ..> UC_Login : <<include>>
```

```

58 UC_Notifications ..> UC_Login : <<include>>
59
60 @enduml

```

Листинг 3: Код диаграммы вариантов использования (сгенерирован DeepSeek по промπτу 2)



### 3.3 Результаты промпта №3

Сгенерируй PlantUML код для диаграммы вариантов использования системы поиска мест и мероприятий. Укажи действующие лица и перечисли случаи использования по ролям. Для Пользователя перечисли действия, учитывая, то это человек, использующий сервис для поиска мест и мероприятий. Имеет общие права зарегистрированного и незарегистрированного пользователей: доступ в каталог и поиск по сайту. Для Незарегистрированного пользователя перечисли действия, учитывая, то это Пользователь, которому доступна регистрация на сайте (помимо действий, разрешенных пользователю) Для Зарегистрированного пользователя перечисли действия, учитывая, то это Пользователь, создавший личный кабинет (не доступна регистрация). Доступны права пользователя, а также аутентификация, добавления места/мероприятия в избранное, просмотр персональных рекомендаций и получение напоминаний. Для Владельца места / организатора мероприятия перечисли действия, учитывая, то это Зарегистрированный пользователь, который имеет право на добавление информации о месте/событии в базу сервиса для продвижения и привлечения аудитории. Для Редакции (администраторов/модераторов) перечисли действия, учитывая, то это Команда людей, не являющихся формальными пользователями, формирующая подборки, проверяющая контент, поступающие в сервис заявки, и следящая за качеством информации в сервисе. Отобрази все связи между действующими лицами и соответствующими вариантами использования. Включи границу системы.

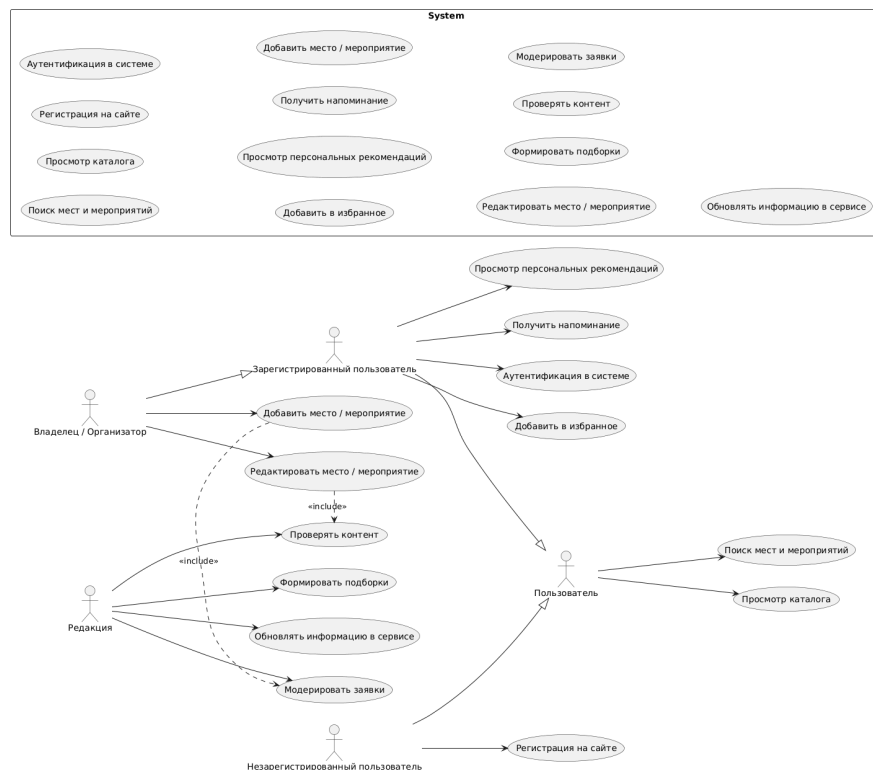
#### DeepSeek

```
1      @startuml
2
3  left to right direction
4
5  actor "Пользователь" as User
6  actor "Незарегистрированный пользователь" as Guest
7  actor "Зарегистрированный пользователь" as AuthUser
8  actor "Владелец / Организатор" as Owner
9  actor "Редакция" as Editor
10
11 usecase "Поиск мест и мероприятий" as UC1
12 usecase "Просмотр каталога" as UC2
13
14 usecase "Регистрация на сайте" as UC3
15
16 usecase "Аутентификация в системе" as UC4
17 usecase "Добавить в избранное" as UC5
18 usecase "Просмотр персональны рекомендаций" as UC6
19 usecase "Получить напоминание" as UC7
20
21 usecase "Добавить место / мероприятие" as UC8
22 usecase "Редактировать место / мероприятие" as UC9
23
24 usecase "Формировать подборки" as UC10
25 usecase "Проверять контент" as UC11
26 usecase "Модерировать заявки" as UC12
27 usecase "Обновлять информацию в сервисе" as UC13
28
29 User --> UC1
30 User --> UC2
31
32 Guest --|> User
33 Guest --> UC3
34
35 AuthUser --|> User
36 AuthUser --> UC4
```

```

37 AuthUser --> UC5
38 AuthUser --> UC6
39 AuthUser --> UC7
40
41 Owner --|> AuthUser
42 Owner --> UC8
43 Owner --> UC9
44
45 Editor --> UC10
46 Editor --> UC11
47 Editor --> UC12
48 Editor --> UC13
49
50 UC8 .> UC12 : <<include>>
51 UC9 .> UC11 : <<include>>
52
53 rectangle System {
54     (Поиск мест и мероприятий)
55     (Просмотр каталога)
56     (Регистрация на сайте)
57     (Аутентификация в системе)
58     (Добавить в избранное)
59     (Просмотр персональные рекомендаций)
60     (Получить напоминание)
61     (Добавить место / мероприятие)
62     (Редактировать место / мероприятие)
63     (Формировать подборки)
64     (Проверять контент)
65     (Модерировать заявки)
66     (Обновлять информацию в сервисе)
67 }
68
69 @enduml

```



- Пояснения к диаграмме:

- Действующие лица (Actors):

- \* *Пользователь*: Базовая роль, от которой наследуются все остальные. Имеет доступ к каталогу и поиску.
    - \* *Незарегистрированный пользователь*: Наследует права Пользователя и может зарегистрироваться.
    - \* *Зарегистрированный пользователь*: Наследует права Пользователя и имеет доступ к персональным функциям (ЛК, избранное, рекомендации).
    - \* *Владелец / Организатор*: Наследует права Зарегистрированного пользователя и может управлять информацией о своих местах/мероприятиях.
    - \* *Редакция*: Отдельная роль с административными и модераторскими функциями.

- Связи:

- \* Стрелка -|> обозначает наследование (например, “Незарегистрированный пользователь” является частным случаем “Пользователя”).
    - \* Стрелка -> связывает действующее лицо с вариантом использования.
    - \* Пунктирная стрелка .> с пометкой «include» показывает, что один вариант использования обязательно включает в себя другой (например, “Добавить место/мероприятие” включает в себя “Модерировать заявки”, так как новая информация должна быть проверена).

- Граница системы:

- \* Прямоугольник **rectangle System** обводит все варианты использования, показывая функциональность, которую предоставляет сама система. Действующие лица находятся за пределами этой границы.

```

1      @startuml
2  left to right direction
3
4  ' ===== Действующие лица =====
5  actor "Пользователь" as User
6  actor "Незарегистрированный\пользователь" as Guest
7  actor "Зарегистрированный\пользователь" as Registered
8  actor "Владелец места /\nОрганизатор мероприятия" as Owner
9  actor "Редакция\n(администраторы / модераторы)" as Admin
10
11 ' ===== Наследование ролей =====
12 Guest --|> User
13 Registered --|> User
14 Owner --|> Registered
15
16 ' ===== Граница системы =====
17 rectangle "Система поиска мест и мероприятий" {
18
19   ' --- Общие варианты использования ---
20   usecase "Просмотр каталога\nмест и мероприятий" as UC_Catalog
21   usecase "Поиск по сайту\n(фильтры, категории)" as UC_Search
22   usecase "Просмотр карточки\nместа / мероприятия" as UC_View
23
24   ' --- Для незарегистрированного пользователя ---
25   usecase "Регистрация\nна сайте" as UC_Register
26
27   ' --- Для зарегистрированного пользователя ---
28   usecase "Аутентификация\n(в од / вы од)" as UC_Login
29   usecase "Управление\nличным кабинетом" as UC_Profile
30   usecase "Добавление в\nизбранное" as UC_Favorite
31   usecase "Просмотр\nперсональны рекомендаций" as UC_Recommend
32   usecase "Получение\nнапоминаний\nпо событиям " as UC_Notify
33
34   ' --- Для владельца / организатора ---
35   usecase "Добавление информации\nпо месте" as UC_AddPlace
36   usecase "Добавление информации\nпо мероприятии" as UC_AddEvent
37   usecase "Редактирование\nсобственны данные \nпо места и события " as
    UC_EditOwn
38   usecase "Просмотр статистики\nинтереса аудитории" as UC_Stats
39
40   ' --- Для редакции ---
41   usecase "Модерация контента\n(проверка и публикация)" as UC_Moderate
42   usecase "Обработка заявок\nот владельцев / организаторов" as UC_Requests
43   usecase "Формирование\nподборок и рекомендаций" as UC_Collections
44   usecase "Контроль качества\nинформации" as UC_Quality
45 }
46
47 ' ===== Связи Пользователя =====
48 User --> UC_Catalog
49 User --> UC_Search
50 User --> UC_View
51
52 ' ===== Связи Незарегистрированного пользователя =====
53 Guest --> UC_Register
54
55 ' ===== Связи Зарегистрированного пользователя =====
56 Registered --> UC_Login
57 Registered --> UC_Profile
58 Registered --> UC_Favorite

```

```

59 Registered --> UC_Recommend
60 Registered --> UC_Notify
61
62 ' ===== Связи Владельца / организатора =====
63 Owner --> UC_AddPlace
64 Owner --> UC_AddEvent
65 Owner --> UC_EditOwn
66 Owner --> UC_Stats
67
68 ' ===== Связи Редакции =====
69 Admin --> UC_Moderate
70 Admin --> UC_Requests
71 Admin --> UC_Collections
72 Admin --> UC_Quality
73
74 @enduml

```



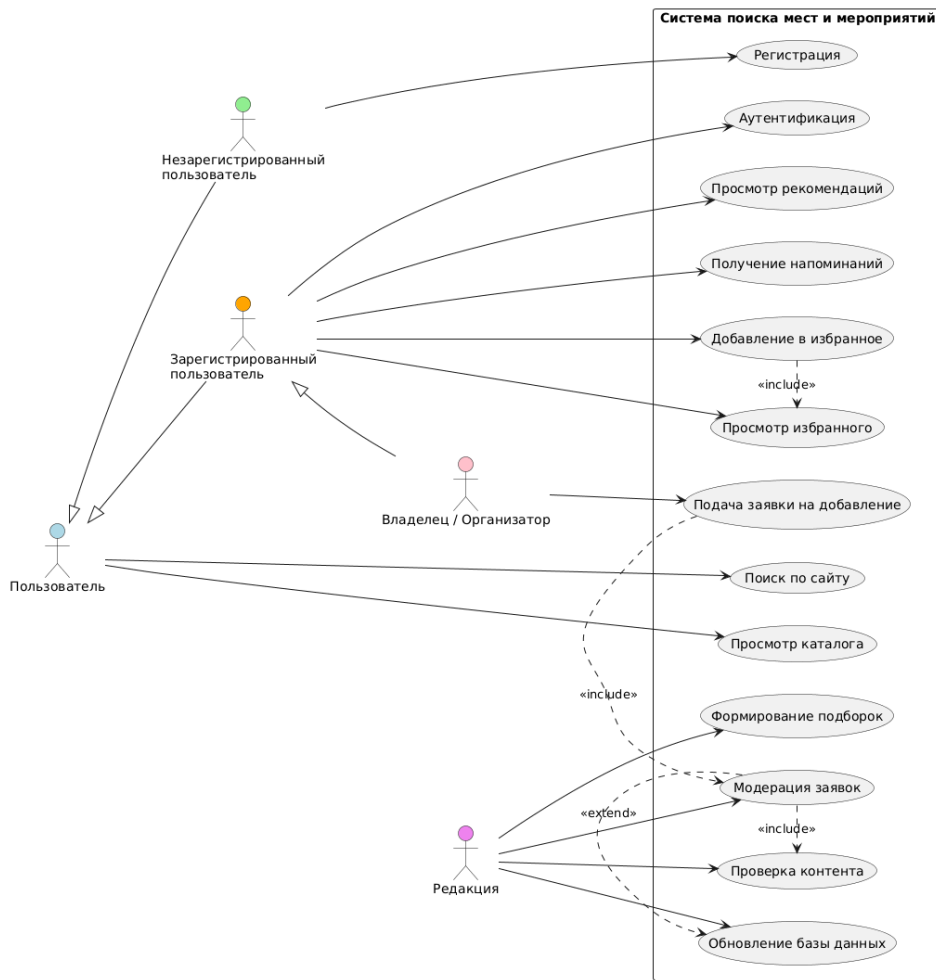
### 3.4 Результаты промпта №4 (личный)

На основе промптов, предложенных в презентации, был придуман свой промпт, который будет использоваться в качестве "Продвинутого":

Сгенерируй PlantUML код для диаграммы вариантов использования системы поиска мест и мероприятий. Укажи действующие лица и перечисли случаи использования по ролям. Для Пользователя перечисли действия, учитывая, то это человек, использующий сервис для поиска мест и мероприятий. Имеет общие права зарегистрированного и незарегистрированного пользователей: доступ в каталог и поиск по сайту. Для Незарегистрированного пользователя перечисли действия, учитывая, то это Пользователь, которому доступна регистрация на сайте (помимо действий, разрешенных пользователю) Для Зарегистрированного пользователя перечисли действия, учитывая, то это Пользователь, создавший личный кабинет (не доступна регистрация). Доступны права пользователя, а также аутентификация, добавления места/мероприятия в избранное, просмотр персональных рекомендаций и получение напоминаний. Для Владельца места / организатора мероприятия перечисли действия, учитывая, то это Зарегистрированный пользователь, который имеет право на добавление информации о месте/событии в базу сервиса для продвижения и привлечения аудитории. Для Редакции (администраторов/модераторов) перечисли действия, учитывая, то это Команда людей, не являющихся формальными пользователями, формирующая подборки, проверяющая контент, поступающие в сервис заявки, и следящая за качеством информации в сервисе. Для понимания структуры сервиса проанализируй текстовое описание диаграммы использования: Сервис по подбору мест и мероприятий представляет из себя платформу для подбора мест и мероприятий разным категориям пользователей. Основными участниками системы являются Абстрактный пользователь, Незарегистрированный пользователь, Зарегистрированный пользователь, Владелец места/организатор мероприятия и Модератор (администратор/редактор). Абстрактный пользователь в сервисе имеет доступ в каталог и может воспользоваться поиском по сайту. Он обладает некоторыми общими правами зарегистрированного и незарегистрированного пользователей: пользователь может пользоваться каталогом мест и мероприятий, а также поисковой строкой. Незарегистрированный пользователь является пользователем (обладает его правами) с доступом к регистрации. После регистрации в сервисе пользователь получает права зарегистрированного пользователя. Зарегистрированный пользователь является пользователем (обладает его правами) и может пройти аутентификацию на сайте (имеет личный кабинет), просматривать персональные рекомендации, сформированные системой на основе его предпочтений, а также получать напоминания от системы. Владелец места/организатор мероприятия является зарегистрированным пользователем (обладает его правами), а также может отправить заявку на добавление новой позиции в сервис на модерацию в редакцию сервиса. Модератор не является формальным пользователем сервиса. Модераторы (=редакторы/администраторы) сервиса осуществляют контроль каталога, т.е. просматривают все поступающие заявки на добавление мест и мероприятий. После проверки заявки модераторы добавляют в базу сервиса новые места и мероприятия. Отобрази все связи между действующими лицами и соответствующими вариантами использования. Включи границу системы.

## DeepSeek

```
1 @startuml
2
3 left to right direction
4
5 actor "Пользователь" as User #lightblue
6 actor "Незарегистрированный\нпользователь" as Unreg #lightgreen
7 actor "Зарегистрированный\нпользователь" as Reg #orange
8 actor "Владелец / Организатор" as Owner #pink
9 actor "Редакция" as Editor #violet
10
11 User <|-- Unreg
12 User <|-- Reg
13 Reg <|-- Owner
14
15 rectangle "Система поиска мест и мероприятий" {
16     (Поиск по сайту) as Search
17     (Просмотр каталога) as Catalog
18     (Регистрация) as Register
19     (Аутентификация) as Auth
20     (Просмотр избранного) as Favorites
21     (Добавление в избранное) as AddFavorite
22     (Просмотр рекомендаций) as Recommendations
23     (Получение напоминаний) as Notifications
24     (Подача заявки на добавление) as SubmitRequest
25     (Формирование подборок) as CurateCollections
26     (Проверка контента) as VerifyContent
27     (Модерация заявок) as ModerateRequests
28     (Обновление базы данны ) as UpdateDB
29 }
30 ' Связи для Пользователя
31 User --> Search
32 User --> Catalog
33 ' Связи для Незарегистрированного пользователя
34 Unreg --> Register
35
36 ' Связи для Зарегистрированного пользователя
37 Reg --> Auth
38 Reg --> Favorites
39 Reg --> AddFavorite
40 Reg --> Recommendations
41 Reg --> Notifications
42
43 ' Связи для Владельца/Организатора
44 Owner --> SubmitRequest
45
46 ' Связи для Редакции
47 Editor --> CurateCollections
48 Editor --> VerifyContent
49 Editor --> ModerateRequests
50 Editor --> UpdateDB
51
52 ' Внутренние связи use cases
53 SubmitRequest .> ModerateRequests : <<include>>
54 ModerateRequests .> VerifyContent : <<include>>
55 ModerateRequests .> UpdateDB : <<extend>>
56 AddFavorite .> Favorites : <<include>>
57
58 @enduml
```



## • Описание диаграммы:

### – Действующие лица (Actors):

- \* Пользователь (абстрактный) - базовый пользователь с общими правами
- \* Незарегистрированный пользователь - наследует права Пользователя + регистрация
- \* Зарегистрированный пользователь - наследует права Пользователя + расширенный функционал
- \* Владелец/Организатор - наследует права Зарегистрированного пользователя + управление контентом
- \* Редакция - административная команда (не формальные пользователи)

### – Варианты использования по ролям:

#### \* Для Пользователя:

- Поиск по сайту
- Просмотр каталога

#### \* Для Незарегистрированного пользователя:

- Все возможности Пользователя +
- Регистрация

#### \* Для Зарегистрированного пользователя:

- Все возможности Пользователя +
- Аутентификация
- Добавление в избранное

- Просмотр избранного
- Просмотр рекомендаций
- Получение напоминаний
- \* **Для Владельца/Организатора:**
  - Все возможности Зарегистрированного пользователя +
  - Подача заявки на добавление места/мероприятия
- \* **Для Редакции:**
  - Формирование подборок
  - Проверка контента
  - Модерация заявок
  - Обновление базы данных

Диаграмма отображает границы системы и все связи между действующими лицами и вариантами использования, включая наследование ролей и зависимости между use cases.

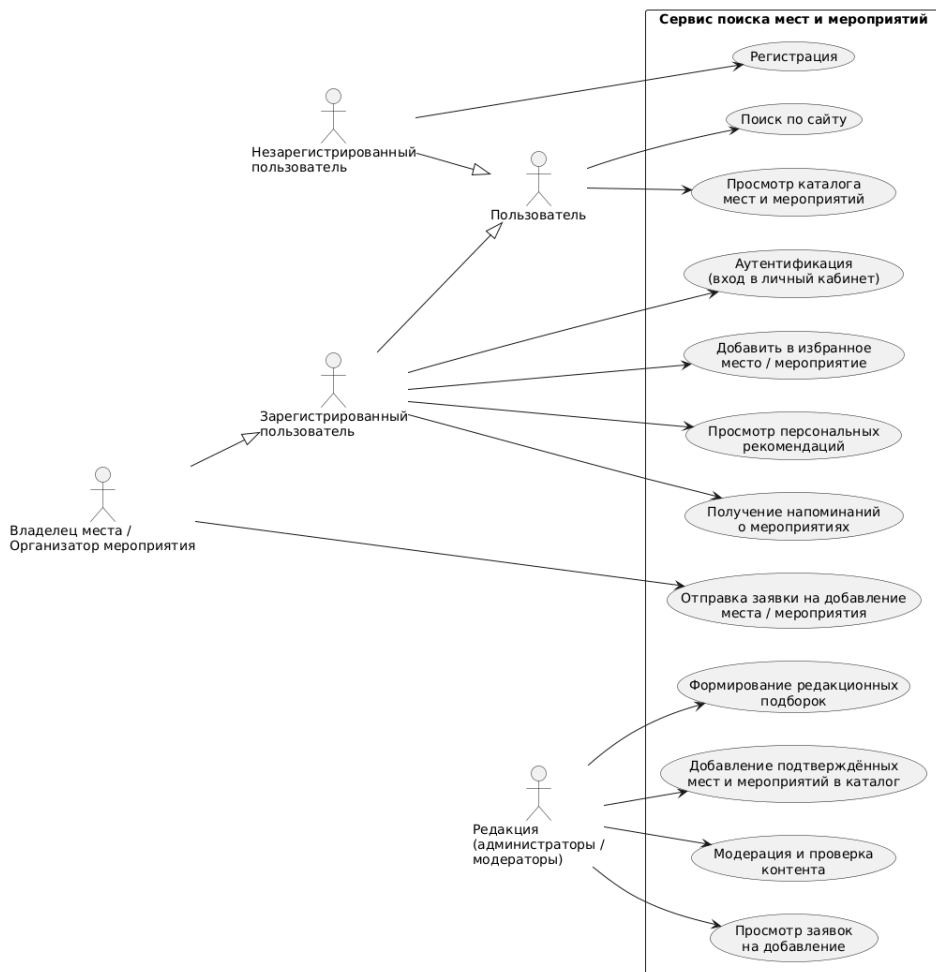
## ChatGPT

```
1      @startuml
2  left to right direction
3
4  ' ==== Граница системы ====
5  rectangle "Сервис поиска мест и мероприятий" {
6
7      ' --- Общие варианты использования ---
8      usecase "Просмотр каталога\nмест и мероприятий" as UC_Catalog
9      usecase "Поиск по сайту" as UC_Search
10
11     ' --- Незарегистрированный пользователь ---
12     usecase "Регистрация" as UC_Register
13
14     ' --- Зарегистрированный пользователь ---
15     usecase "Аутентификация\n(в од в личный кабинет)" as UC_Login
16     usecase "Добавить в избранное\nместо / мероприятие" as UC_Favorite
17     usecase "Просмотр персональны \nрекомендаций" as UC_Recommendations
18     usecase "Получение напоминаний\nпо мероприятия " as UC_Notifications
19
20     ' --- Владелец места / организатор ---
21     usecase "Отправка заявки на добавление\nместа / мероприятия" as
        UC_SubmitRequest
22
23     ' --- Редакция / модераторы ---
24     usecase "Просмотр заявок\nна добавление" as UC_ReviewRequests
25     usecase "Модерация и проверка\nконтента" as UC_Moderation
26     usecase "Добавление подтверждённые \nмест и мероприятий в каталог" as
        UC_AddToCatalog
27     usecase "Формирование редакционны \nподборки" as UC_Collections
28 }
29
30 ' ==== Действующие лица ====
31 actor "Пользователь" as User
32 actor "Незарегистрированный\nпользователь" as Guest
33 actor "Зарегистрированный\nпользователь" as Registered
34 actor "Владелец места /\nОрганизатор мероприятия" as Owner
35 actor "Редакция\n(администраторы /\nmодераторы)" as Moderator
36
37 ' ==== Наследование ролей ====
38 Guest --|> User
39 Registered --|> User
40 Owner --|> Registered
41
42 ' ==== Связи Пользователя ====
43 User --> UC_Catalog
44 User --> UC_Search
45
46 ' ==== Связи Незарегистрированного пользователя ====
47 Guest --> UC_Register
48
49 ' ==== Связи Зарегистрированного пользователя ====
50 Registered --> UC_Login
51 Registered --> UC_Favorite
52 Registered --> UC_Recommendations
53 Registered --> UC_Notifications
54
55 ' ==== Связи Владельца / организатора ====
56 Owner --> UC_SubmitRequest
57
```

```

58 ' ==== Связи Редакции ====
59 Moderator --> UC_ReviewRequests
60 Moderator --> UC_Moderation
61 Moderator --> UC_AddToCatalog
62 Moderator --> UC_Collections
63
64 @enduml

```



### Краткое пояснение структуры

- Пользователь — абстрактная роль с базовыми возможностями: каталог и поиск.
- Незарегистрированный пользователь — наследует Пользователя и может зарегистрироваться.
- Зарегистрированный пользователь — имеет личный кабинет, рекомендации, избранное и напоминания.
- Владелец места / организатор — расширяет права зарегистрированного пользователя возможностью подачи заявок.
- Редакция — внешняя по отношению к пользователям роль, отвечающая за модерацию и наполнение каталога.

## 4 Сравнение результатов и выводы

Все диаграммы были сгенерированы через <https://editor.plantuml.com/> с первого раза (т.е. ИИ был предложен код на PlantUML без ошибок). Для каждого столбца применена десятибалльная шкала.

### 4.1 Сравнительный анализ результатов

Модель	Корректность синтаксиса	Соответствие стандарту UML	Полнота отображения ф-х. треб-й	Кач-во визуализации	Логическая целостность диаграмм
Промпт 1					
DeepSeek	10	7	10	8	6
ChatGPT	10	6	5	10	6
Промпт 2					
DeepSeek	10	9	10	2	8
ChatGPT	10	5	10	2	6
Промпт 3					
DeepSeek	10	5	7	3	4
ChatGPT	10	9	9	10	7
Промпт 4					
DeepSeek	10	7	10	9	10
ChatGPT	10	9	10	10	7

#### Выводы

- Зависимость качества от детализации промта:** Результаты демонстрируют четкую корреляцию между уровнем детализации промта и качеством сгенерированной диаграммы. Например, для обеих моделей наблюдается значительный рост показателей «Полнота отображения функциональных требований» и «Логическая целостность диаграмм» при переходе от промтов 1–3 к промту 4. Особенно наглядно это видно на примере DeepSeek, где оценка логической целостности выросла с 4–6 до 10 баллов, а качество визуализации — с 2–8 до 9.
- Сравнительное превосходство GPT-5:** В общем случае модель GPT-5 показала более стабильные и высокие результаты, особенно в ключевых для практического использования аспектах. Она лидирует или делит первое место по «Корректности синтаксиса» (стабильно 10), «Качеству визуализации» (в трёх случаях из четырёх оценка 10 против максимума 9 у DeepSeek) и «Соответствию стандарту UML» (в двух случаях из четырёх имеет лучший показатель). DeepSeek показала выдающийся результат только в одной категории («Логическая целостность» на промте 4), в то время как GPT-5 демонстрирует лидерство в нескольких категориях одновременно.
- Оптимальная комбинация (GPT-5 + Промт 4):** Наивысший совокупный результат действительно продемонстрирован комбинацией модели GPT-5 и наиболее детализированного Промта 4. Данная связка обеспечила идеальные оценки (10 баллов) по трём из пяти критериев: «Корректность синтаксиса», «Полнота отображения функциональных требований» и «Качество визуализации». При этом по оставшимся двум критериям («Соответствие стандарту UML» — 9, «Логическая целостность» — 7) результат также является высоким, что подтверждает данный вывод о максимальной эффективности.

Таким образом, для получения наиболее качественного кода PlantUML и, как следствие, UML-диаграммы, рекомендуется использовать максимально подробное и структурированное техническое задание (промт) в сочетании с наиболее совершенной из доступных языковых моделей.