

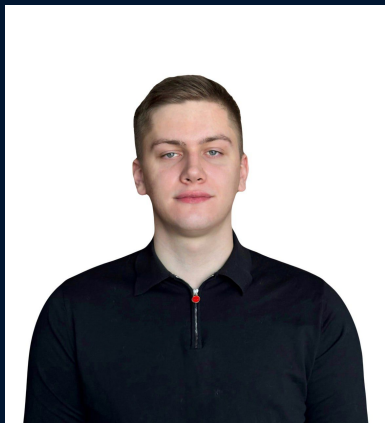


# Team 4

---

GPN LLM Hackathon

# Команда



Максим  
**Юрлов**

Настройка БД  
Реализация бота



Алина  
**Абдушкурова**

Парсинг данных  
Реализация  
рекомендательной  
системы



Максим  
**Шугаев**

Обработка данных  
Реализация  
рекомендательной  
системы



Илья  
**Ореховский**

Настройка бота

# Проблематика

- **Переизбыток информации:** Пользователи сталкиваются с огромным количеством заведений (ресторанов, кафе, баров и т.д.), что затрудняет выбор. Поиск подходящего места становится сложным и требует значительных временных затрат.
- **Плоскость рекомендаций:** Существующие платформы (например, Google Maps, TripAdvisor) часто предлагают рекомендации только на основе общих рейтингов и популярности, которые не всегда учитывают индивидуальные предпочтения пользователя.
- **Отсутствие персонализации:** Большинство сервисов не адаптируют свои рекомендации под конкретные запросы пользователя, такие как предпочтения в кухне, бюджет, местоположение и другие.

Решение – **DORCIA**



# Презентация решения



**Dorcia — это телеграм-бот, который помогает пользователям находить наиболее подходящие заведения (рестораны, кафе и т.д.) на основе их вкусовых предпочтений.** Бот использует рекомендательную систему, основанную на правилах и взвешивании параметров, чтобы предложить пользователю топ-5 заведений, которые лучше всего соответствуют его запросам.

Как Dorcia способствует решению выделенных проблем?

1. Анализ базы заведений и предложение топ-5 вариантов.
2. Использование персонализированной рекомендательной системы, учитывающей популярность, рейтинг заведения другие параметры.
3. Бот задает пользователю вопросы о его предпочтениях (например, любимая кухня, желаемый уровень цен, предпочтения по атмосфере) и на основе этих данных формирует персонализированную подборку заведений.

# Основной функционал

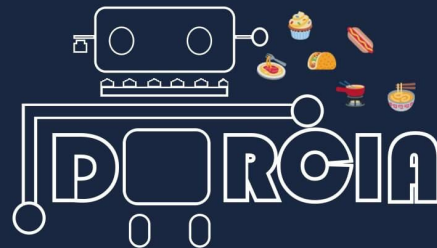
## Опрос пользователя

## Вывод списка рекомендованных заведений

- Для одного пользователя
- Для группы пользователей

## Как использовали LLM

- Написание кода
- Написание текста



сервис по подбору ресторанов



# Firestore Realtime Database



Firestore

Какие вопросы стояли перед нами:

**Как хранить данные о пользователе?**

**Как обойти повторную регистрацию / ввод предпочтений для пользователя?**

Почему именно **Firestore**:

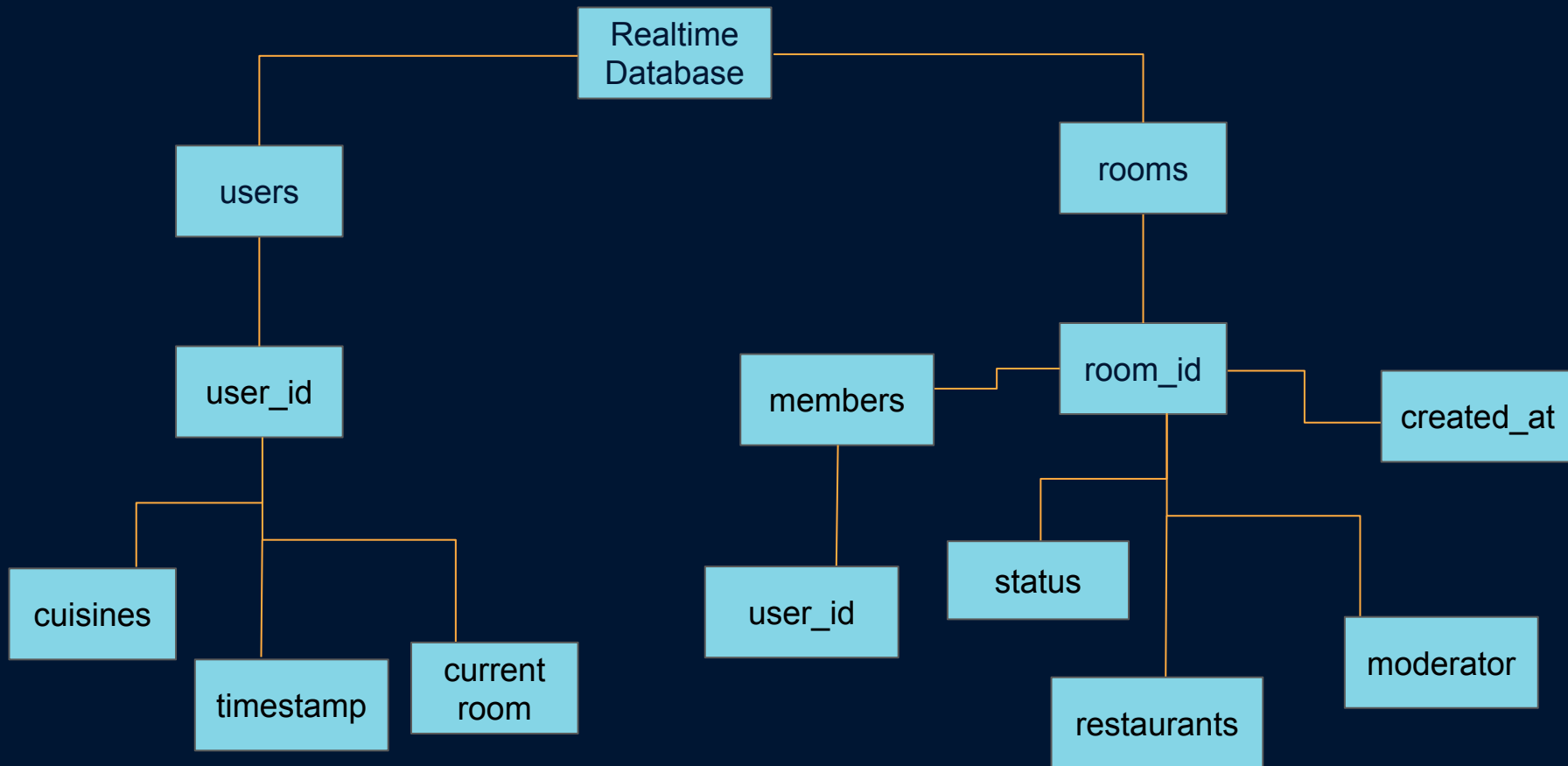
- Надежное хранение данных
- Бесплатный старт
- Синхронизация данных в реальном времени
- Данные хранятся не локально
- Простая интеграция

Как использовали LLM (DeepSeek, GigaChat):

Помог выбрать

Объяснил, как работать с БД

# Структура БД



---

# Парсинг данных

Выполненные задачи:

С помощью get-запросов к API 2GIS собрали данные о заведениях:

- название, адрес, категории, ассортимент, ссылки на фото заведения/блюд (для вывода пользователю)
- кухни, рейтинг, средний чек и время пути (для создания рекомендаций)
- координаты (для построения оптимального маршрута до выбранного ресторана)

Использованные сервисы: Places API и Routing API

Как использовали LLM:

- Оптимизация кода
- Подход к увеличению объема собираемых данных (пагинация)



# Обработка данных и рекомендательная система

Выполненные задачи:

- Реализация алгоритма для обработки данных парсера
- Создание опроса для уточнения пользовательских предпочтений
- Интеграция результатов опроса в рекомендательную систему
- Ранжирование заведений по релевантности, формирования списка топ-5

Как использовали LLM:

1. Формулировка вопросов для опроса
2. Сопоставление типов кухонь и предпочтений пользователя
3. Обработка данных, извлечение с помощью библиотеки ast

---

## Улучшения проекта

- Интеграция реальных данных
- Реализация голосования в комнатах
- Добавление фидбека для ресторана
- Добавление маршрута до выбранного ресторана
- Учёт времени на обед, погодных условий

# Инсайты о командной работе

- Необходимость плана созвона (цель, продолжительность)
- Сложности в организации коммуникации
- Неэффективное распределение сил и возможностей
- Больше внимание на мвр, а не доп фичи
- Технические проблемы (опыт с гит)
- Необходимость в связующем между командами
- ЛЛМки могут сильно подвести

# Спасибо за внимание



Выбрали рестик и идём кушать!