

Министерство науки и высшего образования Российской Федерации
ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ
ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ
«Национальный исследовательский университет ИТМО»
ITMO University

Факультет Прикладной информатики

Программирование в инфокоммуникационных системах

ОТЧЕТ

о практике «Учебная, ознакомительная практика»

Тема задания: Разработка базы данных для веб-приложение для управления игровыми сессиями D&D с системой приоритетов и балансировки групп.

Обучающийся: Сафронов Иван Сергеевич К3221,
Сакулин Иван Михайлович К3221, Мануковская Дарья Михайловна К3221,
Пухарев Сергей Валерьевич К3220, Мартенс Дмитрий Павлович К3220.

Согласовано:

Руководитель практики от университета: Казанова Полина Петровна

Практика пройдена с оценкой _____

Дата _____

Санкт-Петербург

2025

СОДЕРЖАНИЕ

Стр.

ВВЕДЕНИЕ	3
1 ВЫПОЛНЕНИЕ РАБОТЫ	4
1.1 Анализ и разработка требований.....	4
1.1.1 Анализ компании	4
1.1.2 Аналогии.....	5
1.1.3 Проблемы, решаемые продуктом	6
1.2 Разработка проекта программной системы.....	7
1.2.1 Выбор инструментов: PostgreSQL	7
1.2.2 Обоснование структуры и архитектуры базы данных	8
1.3 Выводы.....	11
ЗАКЛЮЧЕНИЕ	12

ВВЕДЕНИЕ

В условиях активного развития веб-технологий и роста популярности настольных ролевых игр актуальной становится задача автоматизации процессов организации и управления игровыми сессиями. Одной из наиболее распространённых систем является Dungeons & Dragons (далее D&D), требующая координации игроков, мастеров, расписаний, а также соблюдения баланса игровых групп. Отсутствие специализированных инструментов управления приводит к перегрузке ведущих, снижению качества игрового процесса и неэффективному распределению участников по сессиям.

Необходимо спроектировать веб-приложение, обеспечивающее централизованное управление игровыми сессиями D&D с использованием системы приоритетов и механизмов балансировки групп.

Цель: проектирование веб-приложения для управления игровыми сессиями D&D с поддержкой приоритетного распределения участников и балансировки игровых групп.

Задачи:

- 1) анализ предметной области и существующих решений для управления игровыми сессиями;
- 2) формализация функциональных и не функциональных требований;
- 3) моделирование ключевых бизнес-процессов системы;
- 4) разработка базы данных;
- 5) создание прототипа интерфейса веб-приложения.

В качестве планируемых результатов практики предусматривается получение формализованного описания системы для управления игровыми сессиями D&D с системой приоритетов и балансировки групп, которое может быть использовано в дальнейшем при реализации программного продукта. Итоговые материалы могут служить основой для разработки прототипа или полноценной информационной системы.

1 ВЫПОЛНЕНИЕ РАБОТЫ

В процессе прохождения практики были пройдены этапы организации команды, анализа предметной области

1.1 Анализ и разработка требований

Первым шагом практики стал анализ предметной области, компании и её требований, затем было проведено исследование аналогов платформы.

1.1.1 Анализ компании

В качестве компании было выбрано подразделение ролевых игр «D&D» клуба настольных игр «GEEKMO», которое несколько лет успешно проводит игры. Запись на игры в изучаемой компании проводится с помощью различных сторонних инструментов (публикации в социальной сети «VK», записи через «google»-таблицы). Отбор и бизнес-логика выполняются вручную, отсутствует защита некоторых данных в таблицах.

Примерное число сотрудников компании (организаторы игр) в разное время составляет от 15 до 20 человек, только от 2 до 5 из них используют персональный компьютер для организации процесса и модерации заявок. На администраторской роли был четко обозначен лишь 1 сотрудник. Отсюда следует, что размер компании можно определить как «очень малая».

Определены основные типы пользователей: помимо разработчиков и администраторов были также выделены обычные пользователи (игроки и игровые мастера) и модераторы (обычные пользователи с правом блокировки нежелательного контента).

На разработку веб-приложения было решено выделить однократные вложения в бюджет. После анализа была выбрана итерационная модель жизненного цикла (поэтапная разработка до достижения готового результата). Поддержку после создания платформы будет осуществлять обученный администратор компании. Найденные в течении определённого времени после

окончания разработки во время эксплуатации ошибки остаются на исправление разработчикам.

1.1.2 Аналоги

Настоящее исследование позволяет идентифицировать системный дефицит на рынке платформ для организации игровых сессий в настольных ролевых играх, который выражается в отсутствии механизмов, гарантирующих уравнительный доступ к игровым ресурсам. Анализ существующих решений (таблица 1) демонстрирует, что их функциональность ограничивается либо предоставлением коммуникационной инфраструктуры («Discord»), либо интеграцией с инструментарием виртуального стола («Fantasy Grounds»), либо выполнением роли пассивного агрегатора объявлений («RPGTableFinder»). Очень важным фактором также является ограниченность платформ в стране. При этом критически важный аспект – алгоритмическое обеспечение справедливого распределения игровых мест – остаётся нереализованным.

Таблица 1 — Сравнение с аналогами

Название	Преимущества	Недостатки
Fantasy Grounds LFG	1) Прямая интеграция с виртуальным столом Fantasy Grounds; 2) Целевая аудитория опытных игроков и мастеров; 3) Поддержка множества игровых систем (D&D, Pathfinder, Savage Worlds).	1) Требуется покупка лицензии (финансовый барьер); 2) Устаревший форумный интерфейс, нет удобных фильтров; 3) Отсутствие инструментов для справедливого отбора игроков.

Продолжение таблицы 1

Название	Преимущества	Недостатки
Discord	1) Гибкость и бесплатность настройки серверов под сообщества; 2) Мгновенная коммуникация через голос и чаты; 3) Огромная и активная аудитория игроков.	1) Полный хаос в организации записи («гонка за места»); 2) Отсутствие единого каталога и профилей игроков; 3) Ручная работа модераторов по управлению списками.
RPGTableFinder	1) Специализированный сервис для поиска онлайн и оффлайн игр; 2) Возможность организовать свой стол и управлять расписанием; 3) Удобный интерфейс с поиском и фильтрами.	1) Небольшая база пользователей, особенно русскоязычных; 2) Нет интеграции с популярными виртуальными столами; 3) Нет алгоритмов для справедливого распределения мест.

1.1.3 Проблемы, решаемые продуктом

Разрабатываемая платформа отличается от большинства других сервисов для поиска игр «D&D». Её основная идея заключается в справедливом распределении игроков, так, чтобы возможность попасть в игру была у каждого. Приоритет получает тот пользователь, который дольше всего не участвовал в играх. Данный подход решает несколько проблем, которые имеются у уже существующих аналогов:

- 1) Исключение случайности – у каждого пользователя одинаковые возможности для попадания на игру, причём каждый пользователь пони-

мает принцип отбора. Такая система гораздо справедливее распространённого «кто быстрее».

- 2) Меньше ручной работы – разрабатываемая платформа сама формирует состав игроков, рассылает уведомления и ведёт очередь. Мастеру не нужно договариваться с каждым участником лично.
- 3) Более устойчивое сообщество – благодаря системе приоритетов, у каждого нового пользователя равные шансы с давним попасть в игру, что повышает шансы новых пользователей остаться в сообществе.
- 4) Прозрачные правила – все решения принимаются по заранее известным для всех пользователей правилам. Пользователи понимают, почему именно они получили или не получили место, и видят своё положение в очереди. Это снижает количество конфликтов и повышает доверие к платформе.

Таким образом, главное отличие разрабатываемой платформы в том, что она не просто соединяет людей, а активно и справедливо управляет распределением игровых сессий. Это помогает решить проблему ограниченного числа мест в играх при большом количестве желающих и предлагает понятные и честные правила взаимодействия внутри игрового сообщества.

1.2 Разработка проекта программной системы

1.2.1 Выбор инструментов: PostgreSQL

Для реализации базы данных проекта был выбран PostgreSQL — реляционная СУБД с открытым исходным кодом, соответствующая требованиям проекта:

- Поддержка сложных типов данных: Наличие встроенного типа ENUM (party_status, queue_status) обеспечивает строгость и читаемость данных, ограничивая возможные значения статусов.
- Целостность данных: Механизмы внешних ключей (FOREIGN KEY), проверочные ограничения (CHECK) и каскадные операции (ON DELETE

CASCADE/SET NULL) гарантируют согласованность и непротиворечивость данных при любых манипуляциях.

- Производительность и масштабируемость: Индексирование по первичным ключам и потенциальным полям для поиска (например, username, email) обеспечит высокую скорость выполнения запросов даже при росте числа пользователей.
- Безопасность: Встроенные средства управления доступом и шифрования позволяют безопасно хранить конфиденциальные данные, такие как хэши паролей.

Данный выбор обеспечивает надежную, масштабируемую и легко сопровождаемую основу для backend-части итогового продукта.

1.2.2 Обоснование структуры и архитектуры базы данных

База данных спроектирована в нормализованной форме для минимизации избыточности и аномалий при вставке, обновлении и удалении данных. Структура состоит из четырех основных таблиц, связанных четкими отношениями.

Существующие таблицы изображены на рисунке 1:

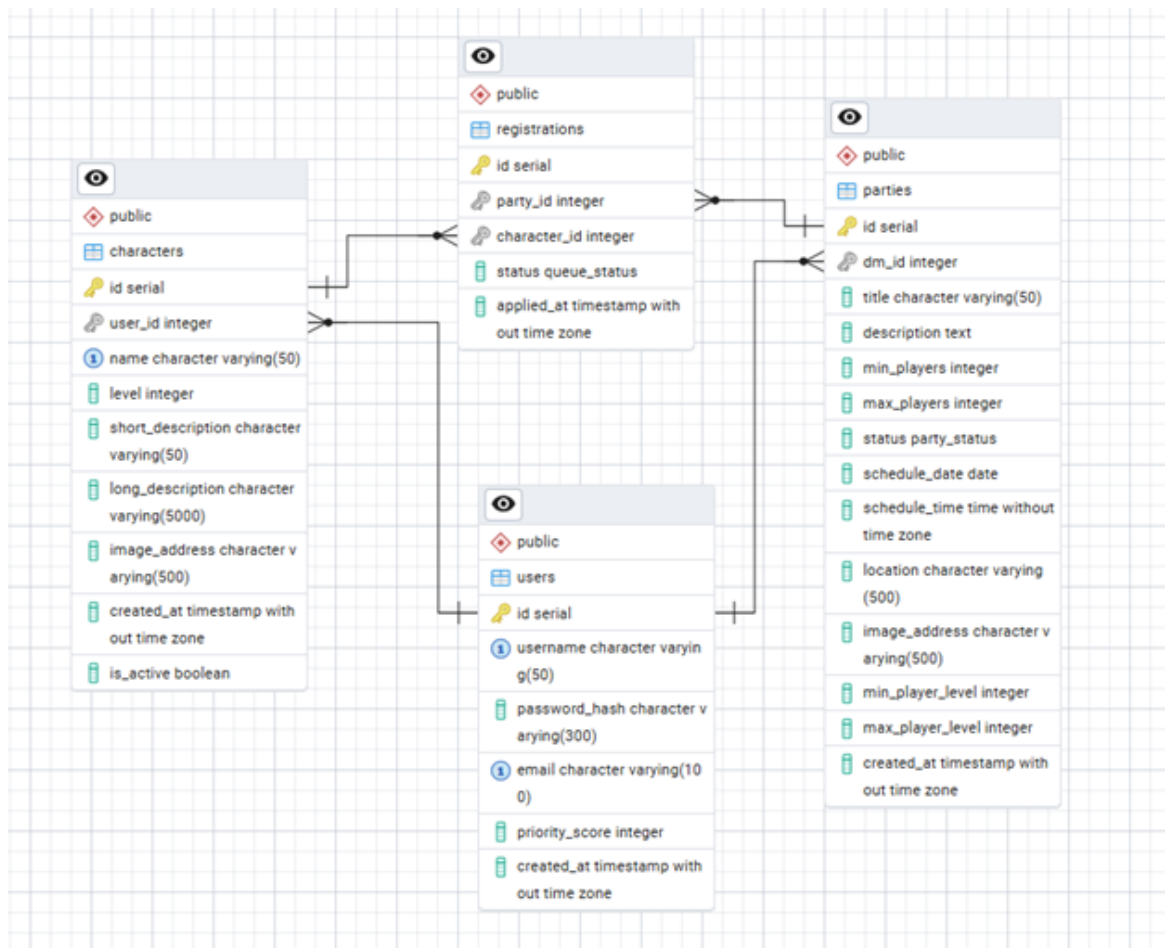


Рисунок 1 — ER-диаграмма базы данных

1. **users**: Центральная таблица для хранения данных зарегистрированных пользователей. Поля `priority_score` и `is_moder` являются ключевыми для бизнес-логики: первый определяет приоритет при записи на партию (основа системы "давности игры"), второй — права доступа для модерации контента.
2. **characters**: Хранит игровые персонажи, созданные пользователями. Связь `user_id` позволяет одному пользователю иметь несколько персонажей, что соответствует игровой практике. Поле `is_active` обеспечивает блокировку персонажа со стороны модерации, без строгого удаления.
3. **parties**: Содержит информацию об игровых партиях (сессиях). Поля `min_players/max_players` и `min_player_level/max_player_level` задают ключевые правила формирования группы: количество и уровень участников. Статус (`status`), управляемый через ENUM, четко определяет жизнен-

ный цикл партии (от модерации до завершения). Связь `dm_id` указывает на пользователя-ведущего (мастера).

4. `registrations`: Реализует связь "многие-ко-многим" между персонажем и игровой сессией. Эта таблица-посредник критически необходима для обработки заявок: несколько персонажей могут регистрироваться на одну партию, и один персонаж может подавать заявки на разные партии. Поле `status` фиксирует решение ведущего или модератора по каждой заявке.

Ключевые связи:

1. Один-ко-многим: Пользователи (`users`) могут создавать множество персонажей (`characters`). Пользователь как мастер может создать множество партий (`parties`). Каждая партия может получить множество заявок в `registrations`. Каждый персонаж (`characters`) может иметь множество заявок в `registrations`.
2. Многие-ко-многим: Персонажи (`characters`) регистрируются на партии (`parties`) через таблицу-связку `registrations`.

Гарантии целостности:

1. Первичные ключи (PRIMARY KEY): `id` во всех таблицах.
2. Внешние ключи (FOREIGN KEY): Все связи между таблицами явно объявлены с правилами обработки удаления (CASCADE для зависимых записей, SET NULL для ведущего партии).
3. Проверки (CHECK): Ограничения на уровни игроков и персонажей (BETWEEN 1 AND 20), логику количества игроков (`max_players >= min_players`).
4. Уникальность (UNIQUE): `username`, `email` в `users`; `name` в `characters` (в рамках всей системы).

Такая архитектура полностью покрывает требования к хранению данных для функционала регистрации, управления пользователями, персонажами и партиями с системой приоритетов, обеспечивая гибкость для расширения и эффективность выполнения запросов.

1.3 Выводы

ЗАКЛЮЧЕНИЕ

В ходе практики были полностью/ частично выполнено Индивидуальное задание

Планируемые результаты практики достигнуты полностью/ частично