

САНКТ-ПЕТЕРБУРГСКИЙ НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ
УНИВЕРСИТЕТ ИТМО

Отчет по домашней работе №1 по курсу “Бэкенд разработка”

Выполнили:

Бахарева М. А., К3342

Привалов К.А., К3342

Проверил:

Добряков Д.И.

Санкт-Петербург

2025 г.

1. Описание темы

Общее описание:

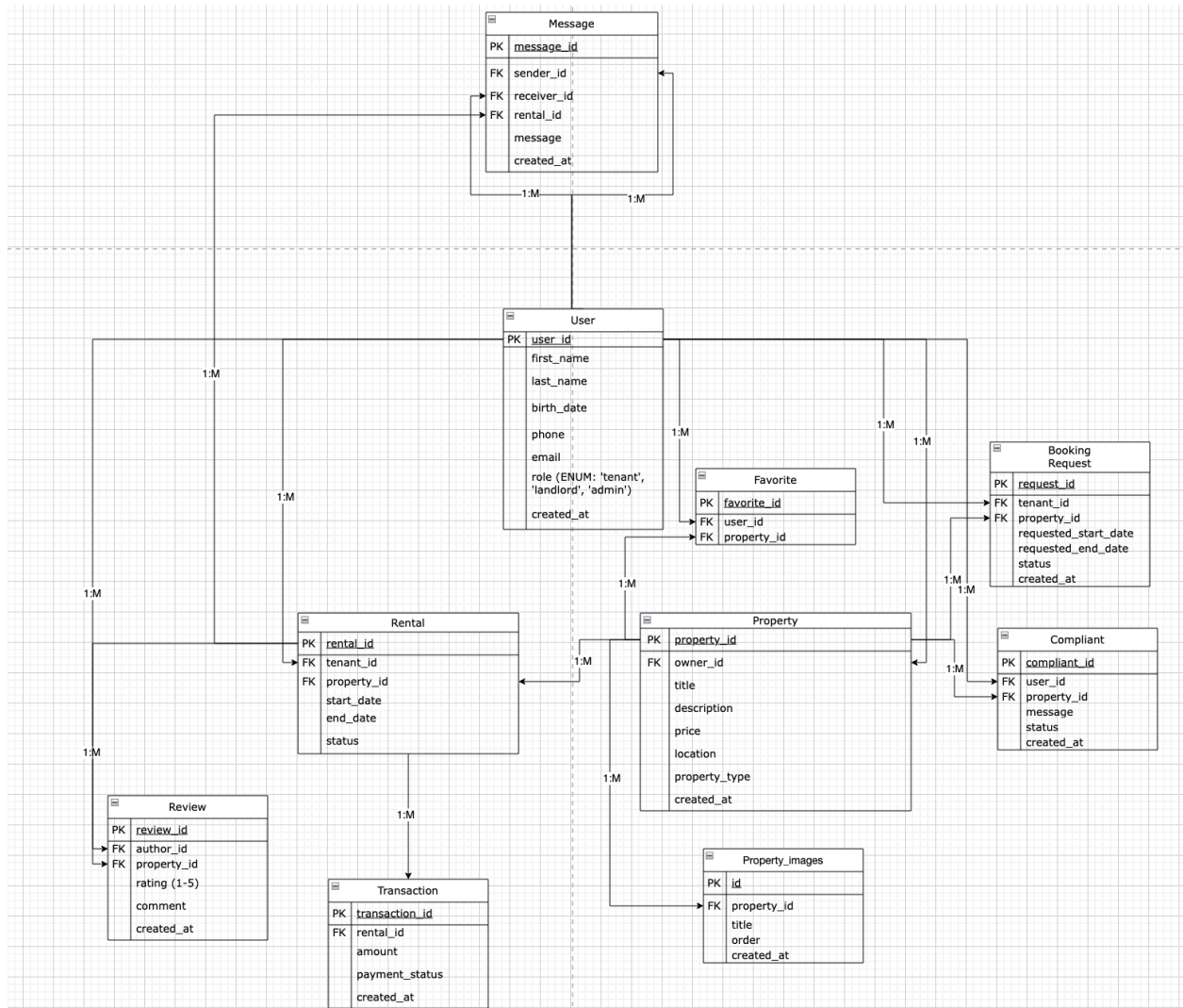
Приложение предназначено для поиска и аренды недвижимости. Пользователи могут просматривать и искать объявления, фильтровать варианты по параметрам (по типу, цене, расположению), общаться с арендодателями и бронировать жилье. У пользователей есть личный кабинет, в котором хранится информация об арендованных и арендуемых объектах, а также “любимых”, добавленных в избранное объектах недвижимости. Есть страницы объектов недвижимости с основной информацией: фото, описание, условия аренды, цена.

Основные функции:

- Регистрация и авторизация (по email)
- Просмотр объявлений недвижимости (фото, описание, цена)
- Фильтрация объектов (цена, район, тип жилья)
- Чат с владельцем (встроенный мессенджер)
- Оформление бронирования (без сложных юридических механизмов и привязки оплаты)

2. Проектирование базы данных

ERD



Описание базы данных

Сущности и их описание

1. user – хранит информацию о пользователях сервиса

- Основные поля: id, email, password_hash, full_name, phone, role (ENUM: tenant, landlord, admin), created_at

- Связи:

- 1:M с property (один владелец может сдавать несколько объектов)
 - 1:M с rental (один арендатор может арендовать несколько объектов)
 - 1:M с message (пользователь может отправлять и получать сообщения)
 - 1:M с review (один пользователь может оставлять несколько отзывов)
 - 1:M с favorite (пользователь может добавить в избранное несколько объектов)
 - 1:M с booking_request (пользователь может отправлять запросы на аренду)
 - 1:M с complaint (пользователь может подавать жалобы)
2. property – хранит информацию об объектах недвижимости
- Основные поля: id, owner_id (FK → user), title, description, price, location, property_type, created_at
 - Связи:
 - M:1 с user (каждый объект принадлежит одному владельцу)
 - 1:M с rental (один объект может быть арендован многократно)
 - 1:M с review (на объект могут оставлять отзывы)
 - 1:M с favorite (объект может быть в избранном у нескольких пользователей)
 - 1:M с booking_request (на объект могут поступать запросы на аренду)
 - 1:M с complaint (на объект могут подавать жалобы)
3. rental – хранит информацию об аренде объектов
- Основные поля: id, tenant_id (FK → user), property_id (FK → property), start_date, end_date, status
 - Связи:
 - M:1 с user (каждая аренда связана с одним арендатором)
 - M:1 с property (каждая аренда связана с одним объектом)
 - 1:M с transaction (на одну аренду может приходиться несколько платежей)
 - 1:M с message (аренда может иметь историю переписки)
4. message – хранит переписку пользователей
- Основные поля: id, sender_id (FK → user), receiver_id (FK → user), rental_id (FK → rental, nullable), message, created_at
 - Связи:
 - M:1 с user (каждое сообщение отправлено одним пользователем)
 - M:1 с rental (может быть привязано к аренде)
5. transaction – хранит платежи, связанные с арендой
- Основные поля: id, rental_id (FK → rental), amount, payment_status, created_at
 - Связи:
 - M:1 с rental (одна аренда может иметь несколько платежей)
6. review – хранит отзывы о недвижимости
- Основные поля: id, author_id (FK → user), property_id (FK → property), rating, comment, created_at
 - Связи:
 - M:1 с user (каждый отзыв оставлен одним пользователем)

- M:1 с property (каждый отзыв относится к одному объекту)
- 7. favorite – хранит избранные объекты пользователей
 - Основные поля: id, user_id (FK → user), property_id (FK → property)
 - Связи:
 - M:1 с user (пользователь может добавить в избранное несколько объектов)
 - M:1 с property (объект может быть в избранном у нескольких пользователей)
- 8. booking_request – хранит запросы на аренду
 - Основные поля: id, tenant_id (FK → user), property_id (FK → property), requested_start_date, requested_end_date, status, created_at
 - Связи:
 - M:1 с user (запрос на аренду отправляет один пользователь)
 - M:1 с property (запрос связан с одним объектом)
- 9. complaint – хранит жалобы пользователей
 - Основные поля: id, user_id (FK → user), property_id (FK → property, nullable), message, status, created_at
 - Связи:
 - M:1 с user (жалоба подается одним пользователем)
- 10. Property_images - хранит фотографии недвижимости
 - Основные поля: id, user_id (FK → user), property_id (FK → property, nullable), title, order, created_at
 - Связи:
 - M:1 с user
 - M:1 с property

Вывод

В ходе выполнения лабораторной работы была разработана концептуальная модель базы данных для сервиса аренды недвижимости. Были определены ключевые сущности системы (пользователи, объекты недвижимости, аренды, сообщения, платежи и т.д.), их атрибуты, а также взаимосвязи между ними. Использование реляционной модели и типизации, включая перечисления (ENUM), обеспечивает целостность данных и логичную организацию информации.

