**Техническое задание на разработку проекта “Сайт агентства недвижимости”**

### **Введение**

Цель проекта – создать веб-приложение с информацией о компании, а также для размещения объектов недвижимости клиентов компании. Платформа должна предоставлять сотрудникам удобные инструменты для создания и редактирования объектов недвижимости, а пользователям – комфортные условия для поиска недвижимости. В проекте будет использоваться фреймворк Django, Docker для контейнеризации и PostgreSQL в качестве базы данных.

### **Требования к проекту**

### **1. Архитектура и технологии**

1. Фреймворк: Django (Python).
2. База данных: PostgreSQL.
3. Контейнеризация: Docker и Docker Compose.
4. Обратный прокси: Nginx.
5. WSGI сервер: Gunicorn.
6. Фронтенд: HTML, CSS, JavaScript (используются Django-шаблоны).

### **2. Функциональные требования**

### **2.1. Управление контентом**

* **Объекты недвижимости:**
  + Создание, редактирование и удаление объектов через административную панель Django.
  + Отображение фото объекта, заголовка, информации об объекте (тип услуги, тип недвижимости, населенный пункт, улица, номер дома, номер квартиры, общая площадь, жилая площадь, площадь кухни, этаж, этажность, площадь участка, тип участка), полного текста описания, даты публикации.
  + Создание каталога со списком объектов и возможностью их фильтрации по заданным критериям.
  + Создание отдельной страницы для каждого объекта с информацией о нем и его фото.
* **Заявки:**
  + Создание заявки через форму “Отправка заявки”
  + Редактирование и удаление заявок через административную панель Django.
  + Изменение статуса заявки через административную панель Django.
  + Отображение ФИО, телефона и статуса заявки.

### **2.2. Навигация и отображение контента**

* **Главная страница:**
  + Отображение краткой информации о компании.
* **Страница каталог недвижимости:**
  + Отображение списка объектов с заголовком, краткой информацией, основной фотографией и стоимостью.
  + Возможность фильтрации объектов по заданным параметрам: типу услуги, типу недвижимости, диапазону цены, комнатности району и улице.
* **Страница объекта недвижимости:**
  + Отображение полной информации о объекте недвижимости с его фотографиями.
* **Страница о компании:**
  + Информация о компании.

### 

### **3. Нефункциональные требования**

### **3.1. Производительность**

* Сжатие и кеширование: Использование механизмов сжатия и кеширования для ускорения загрузки статических ресурсов.

### **3.2. Масштабируемость**

* Контейнеризация: Приложение должно быть контейнеризировано с использованием Docker для легкой масштабируемости и развертывания.

### **3.3. Безопасность**

* SSL/TLS: Все данные должны передаваться по защищенному протоколу.
* Защита от атак: Внедрение защиты от XSS, CSRF и SQL-инъекций.

### **3.4. Удобство использования**

* Интуитивно понятный интерфейс: Приложение должно быть удобным и понятным для пользователя.
* Адаптивный дизайн: Сайт должен корректно отображаться на всех устройствах (десктопы, планшеты, смартфоны).

### **4. Процесс разработки**

### **4.1. Система контроля версий**: Использование Git для управления кодом.

### **4.2. Развертывание**

* Разработка Dockerfile: Для создания Docker-образа приложения.
* Настройка Docker Compose: Для управления несколькими сервисами.
* Настройка Nginx: Для обратного проксирования запросов и обслуживания статических файлов.
* Развертывание на сервере: Установка Docker и Docker Compose на сервере, запуск приложения.