МИНОБРНАУКИ РФ

ФГБОУ ВО «ИжГТУ имени М.Т. Калашникова»

Кафедра «Программное обеспечение»

Отчет

по лабораторной работе №1

по дисциплине «Проектирование и конструирование

программного обеспечения»

|  |  |  |
| --- | --- | --- |
| Выполнил  студент гр. Б20-191-2: |  | Вотинцев Я.А. |
| Принял: |  | Еланцев М.О. |

Ижевск 2023

**Интернет-площадка для размещения авторского контента**

**Суть проекта.** Hidepost - интернет-площадка для размещения авторского контента (блогов, статей), призванная заместить некоторые подобные зарубежные площадки, которые недоступны на территории России, где создатели могут представлять свои произведения широкой аудитории, а пользователи получать доступ к уникальным и качественным материалам.

**Цель**. Предоставление площадки для продвижения и монетизации авторского контента.

Аналитический обзор.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Характеристика | Medium | Patreon | Substack | DeviantArt | Hidepost |
| API доступ | + | + | + | + | + |
| Монетизация | Партнерская программа | Подписочная модель | Подписочная модель | Продажа искусства | Подписочная модель |
| Возможность оплаты из России | - | - | - | - | + |
| Мобильное приложение | + | + | + | + | - |
| Тип контента | Статьи, блоги | Разнообразный контент | Электронные письма, блоги, курсы | Художественные произведения | Статьи, блоги |
| Push-уведомления | + | + | + | + | + |
| Мультиязычность сайта | + | + | + | + | + |
| Редактор для создания постов | + | + | + | + | + |

**Результат проекта** – сайт.

**Функциональные требования.**

1. Наличие frontend приложения.
   * Этот модуль работает и получает информацию напрямую с API системы.
   * Должны быть такие интерфейсы, как: авторизация/регистрация пользователей, просмотр профилей пользователей, поиска/создания/просмотра постов.
2. Наличие API Gateway:
   * Единая точка входа для всех микросервисов.
   * Балансировка нагрузки и маршрутизация запросов.
   * Обработка и валидация входящих запросов.
3. Наличие сервиса авторизации:
   * Реализация механизма регистрации и авторизации.
   * Генерация и валидация токенов доступа.
   * Защита от попыток несанкционированного доступа.
4. Наличие сервиса пользователей:
   * Сохранение и извлечение профилей пользователей.
   * Механизм редактирования данных пользователя.
   * Поддержка загрузки фото профиля пользователя.
5. Наличие сервиса постов:
   * Создание, редактирование и удаление постов.
   * Сортировка и фильтрация постов по различным параметрам.
   * Поддержка мультимедийного контента в постах.
6. Наличие сервиса комментариев:
   * Добавление, редактирование и удаление комментариев к постам.
   * Фильтрация комментариев для конкретного поста.
7. Наличие сервиса лайков:
   * Отслеживание и сохранение лайков к постам и комментариям.
   * Подсчет общего числа лайков для каждого поста и комментария.
8. Наличие сервиса друзей:
   * Механизм добавления в друзья и принятия заявок в друзья.
   * Отображение списка друзей для каждого пользователя.
   * Система сообщений между друзьями
9. Наличие сервиса платежей:
   * Интеграция с платежными системами.
   * Обработка платежей, связанных с платными подписками.
10. Наличие сервиса нотификаций:
    * Генерация уведомлений для пользователей о различных событиях (новые комментарии, лайки, подписки и т.д.).
    * Отправка уведомлений в реальном времени на почту.
11. Наличие сервиса для логирования и мониторинга всей системы:
    * Отслеживание и сохранение логов всех микросервисов.
12. Наличие сервиса для хранения конфигураций и настроек других сервисов
    * Возможность сменить/создать любые настройки сервисов, не прекращая их работу
13. Наличие сервиса для хранения файлов
    * Другие сервисы могут использовать данные сервис для хранения и получения файлов. Пример файла – фото профиля пользователя.

**Допущения и ограничения**.

* Данный проект не является моей идеей, и я не работаю над ним один. В связи с этим, я не могу точно определить временные рамки. Некоторые модули проекта, в частности – frontend приложение и дизайн сайта могут быть не готовы до сдачи диплома в июне.
* Каждый модуль может иметь собственную базу данных по необходимости, а значит может иметь независимую ER модель.