**Пример Технического Задания на Разработку Веб-приложения «Социальная сеть»**

**1. Введение**

1.1 **Цель документа**

Этот документ описывает требования к разработке веб-приложения «Социальная сеть» для повышения эффективности работы команды за счет автоматизации управления задачами и сроками их выполнения.

1.2 **Актуальность и назначение проекта**

Ведение блогов становится важным инструментом самовыражения и общения с аудиторией. Однако многие пользователи сталкиваются с трудностями в управлении контентом и продвижении своих материалов. Цель проекта — создать платформу для удобного создания и управления блогами, с возможностью интеграции с социальными сетями для распространения контента и взаимодействия с читателями. Это поможет блогерам повысить вовлеченность и упростить управление своим контентом.

1.3 **Основные пользователи системы**

Система предназначена для пользователей социальной сети:

* **Пользователи** — создание и управление постами, взаимодействие через комментарии и оценки.
* **Администраторы** — модерация контента.

**2. Цели и задачи проекта**

2.1 **Основные цели**

* Создание удобной платформы для введения и управления блогами.
* Повышение эффективности взаимодействия между пользователями через комментарии, оценки и интеграцию с социальными сетями.

2.2 **Конкретные задачи**

* Разработка функциональности для создания, редактирования и удаления постов.
* Реализация системы комментирования и оценки постов.
* Интеграция с социальными сетями для авторизации
* Разработка системы подписки на пользователей и уведомлений о новых постах.
* Создание интерфейса для просмотра постов с фильтрацией по дате и популярности.

*Пример:* Система должна позволять пользователям создавать и редактировать посты, добавлять изображения и медиафайлы. Читатели должны иметь возможность оставлять комментарии и оценки, а также подписываться на авторов для получения уведомлений о новых материалах.

**3. Требования к системе**

3.1 **Функциональные требования**

* **Управление постами:** возможность добавления, редактирования и удаления постов. Посты могут быть как публичными так и приватными**.**
* **Комментарии и оценки:** пользователи могут оставлять комментарии и оценивать посты**.**
* **Подписка и уведомления:** пользователи могут подписываться на авторов и получать уведомления о новых постах**.**
* **Авторизация через социальные сети:** возможность авторизовываться через социальные сети**.**
* **Поиск и фильтрация:** система должна поддерживать поиск и фильтрацию постов по дате и популярности.

*Пример*: Система должна позволять пользователям создавать и редактировать посты, а также иметь возможность оставлять комментарии и оценки. Также могут видеть количество лайков и комментариев под своими постами.

3.2 **Нефункциональные требования**

* **Производительность:** отклик интерфейса не более 2 секунд при загрузке задач.
* **Масштабируемость:** возможность расширения на 5000+ пользователей без ухудшения производительности.
* **Безопасность:** поддержка HTTPS и защита данных через SSL.

**4. Требования к пользовательскому интерфейсу**

4.1 **Основные экраны**

* **Главная страница:** Отображение постов и кнопка для создания поста.
* **Страница поста:** Подробное описание поста, имя автора, комментарии, дата выкладки и оценки.
* **Панель администратора:** Интерфейс для модерации пользователей и постов.

4.2 **Навигация**

Система должна включать меню навигации с разделами: “Главная”, “Категории”, “Авторизация”, “Регистрация” и поисковая строка.

4.3 **Юзабилити**

* Простой и понятный интерфейс для быстрого создания постов.

*Пример*: На главной странице отображаются посты.

**5. Требования к технической реализации**

5.1 **Языки и технологии**

* **Backend**: Laravel
* **База данных**: MySQL.

5.2 **Архитектура системы**

* Хранение данных в реляционной базе данных.

5.3 **Интеграции**

* Интеграция с системами авторизации, такими как Google OAuth.

*Пример*: Каждый компонент архитектуры должен быть независимым, чтобы можно было масштабировать сервер приложений и базу данных по мере роста нагрузки.

**6. Требования к безопасности**

6.1 **Аутентификация и авторизация**

* **Аутентификация**: OAuth для авторизации через корпоративные учетные записи Google.
* **Авторизация**: разграничение прав доступа — администраторы, пользователи.

6.2 **Шифрование данных**

* **HTTPS** для всех передаваемых данных.
* **Хранение паролей**: хэширование через bcrypt.

6.3 **Политики доступа**

* Защита от SQL-инъекций и XSS.
* Ограничение по IP для административного доступа.

*Пример*: Пользователи не могут редактировать чужие посты.

**7. Ограничения и допущения**

7.1 **Технические ограничения**

* Серверная часть должна работать на облачных платформах (например, AWS или Azure).

7.2 **Финансовые ограничения**

* Бюджет проекта: до 150000 ₽, включая тестирование и документацию.

7.3 **Сроки выполнения**

* Полная реализация проекта — 6 месяцев с начала разработки.

*Пример*:В рамках ограничения по бюджету допустимо использовать только open-source решения для разработки.

**8. Требования к тестированию и приемке**

8.1 **Типы тестирования**

* **Функциональное тестирование**: проверка всех реализованных функций.
* **Нагрузочное тестирование**: .
* **Тестирование безопасности**: проверка на уязвимости.

8.2 **Критерии приемки**

* Полное выполнение всех функциональных и нефункциональных требований.
* Прохождение не менее 95% тестов по результатам приемочных испытаний.

*Пример*: если в процессе тестирования будет выявлено более 5 критических ошибок, система не может быть принята в эксплуатацию.

**9. Требования к документации**

9.1 **Пользовательская документация**

* Руководство пользователя с инструкциями по созданию, редактированию и управлению постами.

9.2 **Техническая документация**

* Описание архитектуры, структура базы данных, API.

*Пример*: Руководство пользователя должно содержать разделы «Управление постами», «Взаимодействие с пользователями», «Фильтрация и сортировка» и быть структурировано для быстрого поиска.

**10. План реализации**

10.1 **Этапы разработки**

1. Анализ и проектирование — 1 месяц.
2. Разработка прототипа — 1 месяц.
3. Полноценная разработка и тестирование — 3 месяца.
4. Внедрение и обучение — 1 месяц.

10.2 **Сроки выполнения этапов**

* Дата начала: 1 марта 2024 года.
* Дата завершения: 31 августа 2024 года.

10.3 **Ответственные лица**

* Менеджер проекта: Батталов А.Б.
* Ведущий разработчик: Батталов А.Б.

*Пример*: Этап «Полноценная разработка и тестирование» включает разработку всех функциональных модулей и последующее тестирование каждого из них.

**11. Приложения**

* **Приложение A**: Детализированные схемы архитектуры.
* **Приложение B**: Примеры интерфейсов.
* **Приложение C**: Технические спецификации для интеграции с API других систем.