

Техническое задание на разработку новостного сайта

Оглавление

1. Термины и определения
2. Общие положения
3. Цели проекта
4. Задачи проекта
5. Структура сайта
6. Требования к функциональному назначению модулей сайта
7. Типы данных
8. Требования к платформе и архитектуре
9. Требования к административной части
10. Условия приёмки работ и требования к передаче сайта
11. Гарантийные обязательства
12. Требования к ведению проекта, согласованию и предоставлению результатов работ
13. Ожидаемые результаты исполнения Договора
14. Права на результаты интеллектуальной деятельности
1. Термины и определения
 - 1.1 Сайт — программный комплекс, состоящий из клиентской (веб-интерфейс) и серверной части, предназначенный для публикации, управления и просмотра новостных постов.
 - 1.2 Посетитель (Неавторизованный пользователь) — конечный пользователь, имеющий доступ к публичной части сайта без регистрации и авторизации.
 - 1.3 Пользователь (Авторизованный пользователь) — зарегистрированный и авторизованный пользователь с ролью "user".
 - 1.4 Администратор (Admin) — пользователь с ролью "admin", обладающий полными правами на управление контентом, пользователями и системой.
 - 1.5 Пост — информационная единица, новость или статья, размещенная на сайте.

1.6 Тег (Ключевое слово) — метка, присваиваемая посту для категоризации и облегчения поиска.

1.7 Область знаний — тематическая категория, к которой может принадлежать пост (например, "Технологии", "Наука", "Искусство").

1.8 Лайк — отметка пользователя, выражающая положительную оценку поста. Один пользователь может поставить только один лайк на один пост.

1.9 Рекомендательная система — механизм, предлагающий пользователям список постов на основе их популярности (количества лайков и просмотров).

1.10 Административная часть (Админ-панель) — веб-интерфейс с ограничением доступа по роли, предназначенный для управления данными сайта.

2. Общие положения

2.1 Заказчик желает реализовать современный новостной сайт с интерактивными функциями для повышения вовлеченности аудитории.

2.2 Сайт должен быть доступен для просмотра всем пользователям сети Интернет. Часть функционала доступна только авторизованным пользователям.

2.3 Финансирование проекта осуществляется за счёт средств Заказчика.

2.4 Оплата этапов работ производится на основании выставленных Исполнителем счёт-фактур.

3. Цели проекта

3.1 Разработка информационной системы, позволяющей:

3.2 Обеспечить удобную публикацию и структурирование новостного контента.

3.3 Предоставить пользователям простой и эффективный поиск и фильтрацию постов.

3.4 Повысить вовлеченность пользователей за счет системы лайков и просмотров.

3.5 Автоматизировать модерацию контента и управление пользователями.

3.6 Внедрить механизм рекомендаций для удержания аудитории на сайте.

4. Задачи проекта

В рамках проекта Исполнитель обязан:

- 4.1 Аналитика и проектирование
 - 4.1.1 Провести сбор требований и составить сценарии использования.
 - 4.1.2 Разработать информационную модель базы данных.
 - 4.1.3 Спроектировать архитектуру клиент-серверного взаимодействия.
- 4.2 Дизайн-концепция
 - 4.2.1 Разработать дизайн-концепцию главной страницы и ключевых модулей сайта (лента постов, страница поста, личный кабинет, админ-панель).
 - 4.2.2 Подготовить интерактивные прототипы в Figma или аналогичном инструменте.
- 4.3 Разработка клиентской части (Frontend)
 - 4.3.1 Реализовать адаптивный пользовательский интерфейс.
 - 4.3.2 Разработать модуль регистрации и авторизации с валидацией данных.
 - 4.3.3 Реализовать ленту постов с сортировкой и пагинацией.
 - 4.3.4 Обеспечить функционал постановки и снятия лайка.
 - 4.3.5 Реализовать страницу поста с учетом количества просмотров.
 - 4.3.6 Разработать модуль поиска по тегам, заголовку и тексту.
- 4.4 Разработка серверной части (Backend) и API
 - 4.4.1 Реализовать REST API с необходимыми эндпоинтами:
 - /api/auth/register — регистрация пользователя.
 - /api/auth/login — авторизация пользователя.
 - /api/posts — получение списка постов, создание поста (admin).
 - /api/posts/{id} — получение, обновление, удаление поста.
 - /api/posts/{id}/like — постановка/снятие лайка.
 - /api/tags — управление тегами (admin).
 - /api/users — управление пользователями (admin).
 - /api/statistics — получение статистики (admin).
 - 4.4.2 Реализовать систему ролевого доступа (RBAC).
 - 4.4.3 Реализовать механизм рекомендаций (подборка популярных постов).

- 4.5 Административная часть
- 4.5.1 Реализовать веб-интерфейс для:
- * управления постами (CRUD);
 - * управления пользователями (просмотр, блокировка/разблокировка);
 - * управления тегами и областями знаний (CRUD);
 - * просмотра статистики по постам (лайки, просмотры).
- 4.6 Тестирование и выпуск
- 4.6.1 Провести функциональное, интеграционное и нагрузочное тестирование.
- 4.6.2 Подготовить инструкции по эксплуатации для администраторов.
- 4.6.3 Развернуть сайт на продакшн-сервере Заказчика.

5. Структура сайта

1. Публичная часть:
 - a. Главная страница (Лента постов)
 - b. Страница поста
 - c. Страница регистрации
 - d. Страница авторизации
 - e. Страница результатов поиска
2. Личный кабинет пользователя (после авторизации):
 - a. Настройки профиля

Административная часть:

Панель управления постами

Панель управления пользователями

Панель управления тегами и областями знаний

Страница статистики

6. Требования к функциональному назначению

- 6.1 Неавторизованный пользователь
- 6.1.1 Просматривает ленту постов.
- 6.1.2 Сортирует посты по дате публикации и количеству лайков.
- 6.1.3 Переходит на страницу поста, увеличивается счетчик просмотров.
- 6.1.4 Видит заголовок, текст, автора, дату, количество лайков и просмотров

поста.

6.1.5 Регистрируется на сайте с соблюдением ограничений (пароль от 8 символов, латиница и т.д.).

6.1.6 Осуществляет поиск постов по ключевым словам (тегам), заголовку и тексту.

6.2 Авторизованный пользователь

6.2.1 Имеет весь функционал неавторизованного пользователя.

6.2.2 Ставит лайк на пост (только один раз).

6.2.3 Снимает свой лайк с поста.

6.2.4 Получает рекомендации (блок "Самые популярные посты").

6.2.5 Осуществляет поиск постов.

6.3 Администратор

6.3.1 Входит через логин/пароль в админ-панель.

6.3.2 Выполняет полный CRUD (Создание, Чтение, Обновление, Удаление) для постов.

6.3.3 Управляет пользователями: просматривает список, блокирует/разблокирует учетные записи.

6.3.4 Управляет тегами и областями знаний: добавляет, редактирует, удаляет.

6.3.5 Просматривает статистику: количество лайков и просмотров для каждого поста, список самых популярных постов.

7. Типы данных

7.1 Пользователь (User)

- `user_id` (уникальный идентификатор, первичный ключ)
- `username` (уникальный логин, используется для входа и связи)
- `email` (уникальный)
- `password_hash`
- `created_at` (дата регистрации)
- `role` ('user', 'admin')
- `is_active` (флаг активности)

7.2 Пост (Post)

- `post_id` (уникальный идентификатор, первичный ключ)
- `title`
- `content` (текст поста)
- `created_at` (дата публикации)
- `author_username` (связь с пользователем через `User.username`)
- `views_count`

7.3 Тег (Tag)

- `tag_id` (уникальный идентификатор, первичный ключ)
- `name` (уникальное название тега, используется для связи и поиска)

7.4 Область знаний (Category)

- `category_id` (уникальный идентификатор, первичный ключ)
- `name` (уникальное название категории)

7.5 Связь Пост-Тег (Post_Tag)

- `id`
- `post_id` (связь с `Post.post_id`)
- `tag_name` (связь с `Tag.name`)

7.6 Связь Пост-Область знаний (Post_Category)

- `id`
- `post_id` (связь с `Post.post_id`)
- `category_name` (связь с `Category.name`)

7.7 Лайк (Like)

- `like_id`
- `username` (связь с `User.username`)
- `post_id` (связь с `Post.post_id`)
- `created_at`

8. Требования к платформе и архитектуре

1. Клиентская часть: Адаптивный веб-сайт, работающий в современных браузерах (Chrome, Firefox, Safari, Edge). Рекомендуется использовать React, Vue.js или аналогичный фреймворк.

2. Серверная часть: REST API на Node.js (Express), Python (Django/FastAPI), PHP (Laravel) или аналогичной технологии.
3. База данных: Реляционная СУБД (например, PostgreSQL или MySQL).
4. Аутентификация: На основе JWT (JSON Web Tokens).

9. Требования к административной части

1. Админ-панель реализуется как защищенная часть веб-приложения.
2. URL: `https://[domain]/admin`
3. Доступ имеют только пользователи с ролью `admin`.
4. Все списки сущностей (посты, пользователи, теги) поддерживают пагинацию, поиск и фильтрацию.
5. Интерфейс должен быть интуитивно понятным и удобным для выполнения задач, описанных в п. 6.3.

10. Условия приёмки работ и требования к передаче сайта

- 10.1 Приёмка осуществляется по этапам с подписанием Актов.
- 10.2 Заказчик предоставляет тестовое и продакшн окружение (хостинг/VPS).
- 10.3 Исполнитель передаёт:
 - 10.3.1 Исходный код клиентской и серверной части.
 - 10.3.2 Инструкции по развертыванию и эксплуатации.
 - 10.3.3 Документацию по API.
 - 10.3.4 Доступ к админ-панели (учётные данные администратора).
 - 10.3.5 Дамп базы данных с тестовыми данными (опционально).

11. Гарантийные обязательства

Исполнитель предоставляет гарантийную поддержку на срок [6] месяцев с момента подписания акта сдачи-приёмки, в рамках которой устраняются выявленные ошибки и недоработки.

12. Требования к ведению проекта

Исполнитель обязуется предоставлять еженедельные отчеты о ходе работ, использовать систему контроля версий (Git) и систему управления проектами (например, Jira, Trello), доступные Заказчику.

13. Ожидаемые результаты исполнения Договора

- 13.1 Заказчик получает:
- 13.1.1 Работающее веб-приложение (новостной сайт).
- 13.1.2 Бэкенд с REST API.
- 13.1.3 Админ-панель для управления контентом и пользователями.
- 13.1.4 Полный комплект документации и инструкций.
- 13.1.5 Развернутый и настроенный проект на инфраструктуре Заказчика.

14. Права на результаты интеллектуальной деятельности

- 14.1 Все исключительные права на созданный программный продукт, исходный код, дизайн и документацию передаются Заказчику после полной оплаты работ.
- 14.2 Исполнитель не вправе использовать переданные Заказчику материалы без его письменного согласия.
- 14.3 Исполнитель гарантирует, что используемые в проекте библиотеки, шрифты и иконки имеют соответствующие лицензии, не нарушающие права третьих лиц.