Geekbrains

**Создание информационно-управляющего портала для членов ЖСПК "Зенитчик-4": проектирование и реализация на основе современных веб-технологий**

Программа:

Разработчик — Fullstack разработчик

Андросик Максим Михайлович

г. Фаниполь

2025

**Введение**

**Тема проекта**: «Создание информационно-управляющего портала для членов ЖСПК "Зенитчик-4": проектирование и реализация на основе современных веб-технологий».

**Актуальность**: В условиях цифровизации жилищно-коммунального сектора Республики Беларусь многие кооперативы сталкиваются с неэффективностью традиционных методов управления:

* **Ручной учёт** платежей и показаний счетчиков приводит к ошибкам.
* **Отсутствие прозрачности** в финансовых операциях снижает доверие членов кооператива.
* **Ограниченные каналы коммуникации** (офлайн-собрания, бумажные объявления) затрудняют оперативное взаимодействие.

Разработанный портал решает эти проблемы, предоставляя единую цифровую платформу для автоматизации ключевых процессов ЖСПК.

**Цель проекта**: Создать полнофункциональный веб-портал для ЖСПК «Зенитчик-4», обеспечивающий:

* Автоматизацию учёта членов кооператива и финансовых операций.
* Удобное взаимодействие между жильцами и правлением.
* Прозрачность управления жилым комплексом.

**Задачи**:

1. Провести анализ потребностей ЖСПК «Зенитчик-4» и аналоговых решений.
2. Разработать архитектуру системы (фронтенд на React + Material-UI, бэкенд на Laravel).
3. Реализовать модули:
   * Личный кабинет (платежи, показания счетчиков).
   * Система объявлений и чат.
   * Административная панель для правления.
4. Протестировать функционал на удобство и безопасность.
5. Разработать рекомендации по масштабированию системы.

**Практическая значимость**:

* Повышение эффективности управления жилым комплексом.
* Снижение нагрузки на администрацию кооператива.
* Возможность тиражирования решения для других ЖСПК.
* Перспективы интеграциис ЕРИП (онлайн-платежи) и АСКУЭ (Автоматизированная система коммерческого учёта электроэнергии).

**Используемые технологии**:

* *Frontend*: React, Material-UI, Axios.
* *Backend*: Laravel (PHP), REST API.
* *База данных*: MySQL.
* *Дополнительные инструменты*: Postman (тестирование API).

**Безопасность:**

Система реализует многоуровневую защиту данных:

* **Аутентификация** через **Laravel Sanctum** с подтверждением email, что исключает доступ неавторизованных пользователей.
* **Разграничение ролей**:
  + Пользователи (гости) — могут отправлять обращения правлению через сайт.
  + Собственники (члены кооператива) — доступ к персональным финансовым данным, подаче показаний счетчиков.
  + Управляющие — публикация объявлений, формирование квитанций, загрузка документов.
* **Шифрование** HTTPS, хеширование паролей.

**Состав команды**: Проект реализован индивидуально. Роли:

* Full-stack разработчик (проектирование, верстка, программирование).
* UX/UI-дизайнер (создание интерфейсов).
* Тестировщик (юзабилити-тесты, отладка).

**Структура работы**: Дипломный проект состоит из введения, 3 глав, заключения, списка литературы и приложений (исходный код, скриншоты интерфейсов).

### ****Глава 1. Теоретические основы разработки информационно-управляющего портала****

**1.1. Анализ предметной области: управление ЖСПК**

* **Проблемы жилищных кооперативов**:
  + Неэффективный документооборот (бумажные квитанции, протоколы собраний).
  + Отсутствие прозрачности финансовых операций.
  + Сложности коммуникации между жильцами и правлением.
* **Требования к цифровому решению**:
  + Автоматизация рутинных процессов (начисления, отчетность).
  + Многоуровневая система доступа (ролевая модель).
  + Соответствие законодательству РБ (Закон «О жилищных отношениях»).

**1.2. Обзор современных веб-технологий**

* **Frontend**:
  + **React**: компонентный подход, виртуальный DOM для быстрого рендеринга.
  + **Material-UI (MUI)**: готовые UI-компоненты, соответствие принципам Material Design.
  + **Axios**: работа с API, обработка ошибок.
* **Backend**:
  + **Laravel**: MVC-архитектура, встроенные инструменты безопасности (Sanctum).
  + **REST API**: стандартизированный обмен данными между фронтендом и бэкендом.
* **Базы данных**:
  + **MySQL**: реляционная СУБД для структурированного хранения данных (платежи, пользователи).

**1.3. Методы обеспечения безопасности**

* **Аутентификация и авторизация**:
  + **Laravel Sanctum**: токены доступа + cookie-аутентификация.
  + **Подтверждение email**: защита от ботов и нелегитимных регистраций.
* **Ролевая модель**:
  + **3 уровня доступа** (пользователь, собственник, управляющий) на основе Middleware.
  + **Шифрование данных**: HTTPS, хеширование паролей (bcrypt).

**1.4. Тестирование и оптимизация веб-приложений**

* **Юзабилити-тестирование**:
  + A/B-тестирование интерфейсов (например, формы подачи показаний).
* **Нагрузочное тестирование**:
  + Инструменты: Apache JMeter (проверка на 200+ пользователей).
* **Оптимизация**:
  + Ленивая загрузка (LazyLoad) для изображений.
  + Кэширование API-ответов (Redis).