

Краткая презентация вариантов тем дипломного проекта

8 сентября 2025 г.

Содержание

Введение	2
1 Идея 1: Интеллектуальный PWA-конструктор портфолио для креативных специалистов с использованием AI	2
2 Идея 2: Визуальный инструмент для проектирования и прототипирования UI-анимаций и микровзаимодействий	4
3 Идея 3: Сервис для «шаринга» секретов с истекающим сроком действия и ограничением числа просмотров	5
4 Идея 4: Анализ и визуализация утечек данных на основе публичных баз	6

Введение

Этот документ содержит четыре продуманных и «продающих» идеи для дипломного проекта в области веб-разработки и прикладной безопасности. Для каждой идеи описаны суть, целевая аудитория, ценность, отличия от аналогов, примерный стек технологий, план работ и аргументы в пользу выбора темы.

1 Идея 1: Интеллектуальный PWA-конструктор портфолио для креативных специалистов с использованием AI

Портал/приложение, которое за 10–15 минут превращает набор работ дизайнера/фотографа/иллюстратора в уникальное, адаптивное и оффлайн-доступное PWA-портфолио с интеллектуальным AI-ассистентом, который генерирует тексты, предлагает порядок проектов и подбирает цветовую палитру.

Суть проекта

Разработать PWA с интерфейсом "загрузил работы — получил сайт":

- AI-ассистент генерирует описания проектов, теги и рекомендации по структуре;
- алгоритм предлагает композицию и порядок проектов на основе метрик (вес работ, стиль, хронология);
- генерация стилистики (цвета, акценты) на основе анализа изображений;
- оффлайн-режим и установка как приложение (PWA).

Целевая аудитория

Креативные специалисты: дизайнеры, фотографы, иллюстраторы, motion-дизайнеры, которые не хотят тратить время на верстку, но хотят выразительный, персонализированный сайт-портфолио.

Ценность для пользователя

- Экономия времени (AI пишет тексты и структуру);
- Уникальный дизайн вместо шаблонов;
- Работоспособность оффлайн и возможность показать портфолио без интернета;
- Простота использования без навыков верстки.

Аналоги и отличия

Аналоги: Tilda, Readymag, Wix, Behance.

Отличия: фокус на AI-ассистенте для генерации контента и внешнего вида, PWA-first

подход (оффлайн), генерация уникальной палитры и порядка работ, а не набор готовых шаблонов.

Технологии и стек

- Фронтенд: React + Vite, PWA manifest + service workers;
- AI: OpenAI API (или локальная модель), prompts + постобработка;
- Анализ изображений: TensorFlow.js / ML-алгоритмы для извлечения доминантных цветов/стилей;
- Бэкенд: Node.js + Express (опционально) или serverless для хранения метаданных;
- Хранение: CDN / объектное хранилище; база данных: PostgreSQL / Firebase;
- Деплой: Vercel/Netlify и контейнеры для API.

Ожидаемый результат

Рабочий прототип PWA с возможностью загрузки работ, AI-генерации описаний и экспорта итогового сайта/шаблона; демонстрация нескольких реальных портфолио примеров.

Это практический проект с сильной коммерческой ценностью: сочетает AI, современный фронтенд и PWA. Результат — демонстрабельный продукт, портфолио само по себе станет кейсом и может продолжиться как стартап.

2 Идея 2: Визуальный инструмент для проектирования и прототипирования UI-анимаций и микровзаимодействий

Инструмент, который позволяет дизайнеру "нарисовать" сложную анимацию интерфейса и получить готовый к использованию CSS/JS-код (например, для GSAP или Framer Motion) — мост между дизайнером и разработчиком.

Суть проекта

Разработка веб-инструмента с визуальным редактором таймингов, кривых перехода, вложенных анимаций и состояний. На выходе — экспорт готового кода и примеров использования.

Целевая аудитория

UI/UX-дизайнеры и frontend-разработчики.

Ценность для пользователя

- Ускорение передачи дизайна в продуктовую разработку;
- Уменьшение недопониманий между дизайнерами и разработчиками;
- Возможность получить точный, оптимизированный код микровзаимодействий.

Аналоги и отличия

Аналоги: Framer (частично), Principle, After Effects (с плагинами).

Отличия: полноценный экспорт production-ready кода; фокус только на микровзаимодействиях и простота для дизайнеров.

Технологии и стек

- Фронтенд: React, Typescript, Canvas/WebGL или SVG для визуального редактора;
- Библиотеки анимации: GSAP и/или Framer Motion;
- Экспорт: генератор кода (JS/CSS), примеры интеграции;
- Бэкенд: минимален (авторизация, сохранение сцен) — Node.js / serverless;

Ожидаемый результат

Интерактивный прототип редактора + набор экспортируемых примеров (кнопка с анимацией, меню, переходы), инструкция по встраиванию в проект.

Коммерчески востребована: сокращение времени разработки UI, тесная связь с индустрией. Тема технически интересна (визуализация, алгоритмы интерполяции кривых, генерация кода) и даёт сильные практические результаты.

3 Идея 3: Сервис для «шаринга» секретов с истекающим сроком действия и ограничением числа просмотров

Простой и безопасный сервис для отправки одноразовых секретов (пароли, ключи, токены) — шифрование на стороне клиента, уничтожение после первого просмотра и гибкие настройки времени жизни.

Суть проекта

Веб-приложение, где пользователь шифрует текст в браузере (Web Crypto API), получает короткую ссылку. Сервер хранит только зашифрованный контейнер, а при первом успешном просмотре удаляет данные или помечает как использованные.

Целевая аудитория

IT-специалисты, команды разработки, пользователи, которые хотят безопасно передать чувствительные данные.

Ценность для пользователя

- Безопасная передача краткосрочных секретов;
- Сервер не видит незашифрованные данные (zero-knowledge подход);
- Понятный UX и контроль над временем/количеством просмотров.

Аналоги и отличия

Аналоги: privatebin.net, некоторые функциональности password managers.

Отличия: акцент на UX, строгая архитектура «сервер никогда не знает секрет», гибкая политика просмотров и интеграция с мобильными PWA/уведомлениями.

Технологии и стек

- Фронтенд: React / Svelte, Web Crypto API (AES-GCM);
- Бэкенд: Node.js + Express / Go, хранение только ciphertext;
- БД: PostgreSQL / Redis (для TTL и счётчиков просмотров);
- Дополнительно: Rate limiting, защита от брутфорса, audit-log.

Ожидаемый результат

Рабочий сервис с безопасным клиентским шифрованием, политиками истечения и отчётом об использовании. Документация по безопасности и анализ угроз.

Тема сочетает практическую значимость (реальная потребность в безопасной передаче секретов) и глубокую инженерную составляющую (криптография, архитектура). Отлично подходит для демонстрации навыков безопасности и продакшн-разработки.

4 Идея 4: Анализ и визуализация утечек данных на основе публичных баз

Сервис, который позволяет безопасно проверить, присутствует ли email в утечках, и визуализировать историю компрометации — когда, из какой утечки и какие типы данных были затронуты.

Суть проекта

Сбор (или использование готовых) публичных дампов утечек, нормализация, индексирование и создание интерфейса для безопасной проверки через хеширование email на клиенте. Подробные визуализации: временные шкалы, типы данных, статистика.

Целевая аудитория

Широкая: пользователи с email, компании, security-исследователи и образовательные проекты.

Ценность для пользователя

- Не просто "да/нет" а подробная аналитика о каждой утечке;
- Понимание, какие данные могли быть украдены и когда;
- Образовательная ценность — повышение осведомлённости о цифровой гигиене.

Аналоги и отличия

Аналоги: Have I Been Pwned (HIBP).

Отличия: углублённая визуализация истории пользователя, образовательный интерфейс, фокус на UI-аналитике и объяснениях, а также локальная клиентская логика хеширования для приватности.

Технологии и стек

- БД/хранилище: Elasticsearch/ClickHouse/PostgreSQL для быстрых поисков;
- ETL: Python/Go для парсинга и нормализации дампов;
- Фронтенд: React + D3.js / Recharts для визуализаций;
- Приватность: клиентское хеширование (SHA-1/sha256) и минимизация логирования.

Ожидаемый результат

Рабочий прототип поиска по хешам с набором визуализаций утечек и удобным интерфейсом объяснения риска. Отчёт по производительности и предложения по оптимизации (индексация, sharding).

Баланс между инженерией больших данных и UI/UX: отличная демонстрация навыков обработки массивных данных, аналитики и визуализации. Практическая ценность для общества и бизнеса.

Сравнение и рекомендации

- **Для коммерческой и стартап-ориентированной работы:** Идея 1 (PWA-конструктор) — яркий продукт, который можно монетизировать и масштабировать.
- **Для глубокой технической специализации (алгоритмы/визуализация):** Идея 2 (инструмент анимаций) — интересные технические задачи по визуализации и генерации кода.
- **Для демонстрации security/crypto компетенций:** Идея 3 (одноразовые секреты) — сильный академический и практический кейс по безопасности.
- **Для работы с большими данными и аналитикой:** Идея 4 (утечки данных) — масштабируемая система, хороший повод показать навыки ETL/индексации/визуализации.

Каждая из предложенных тем даёт заметный практический результат — работающий прототип, демонстрация профессиональных навыков и материалы для публикации/демонстрации на защите: код, документация, результаты тестов и презентация.