

**САНКТ-ПЕТЕРБУРГСКИЙ НАЦИОНАЛЬНЫЙ
ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ ИТМО**

Дисциплина: Фронт-энд разработка

Отчет

Лабораторная работа 1

Выполнил:

Берулава Леон Алхасович

К3429

Проверил:

Добряков Д. И.

Санкт-Петербург

2026 г.

1. Общее описание проекта

«Кулинарный Портал» представляет собой веб-приложение для публикации, поиска и управления кулинарными рецептами. Проект реализован в виде статического веб-сайта с использованием современных веб-технологий и предоставляет пользователям интуитивный интерфейс для работы с кулинарными рецептами.

2. Цель и назначение

Основные цели проекта:

- Предоставление пользователям удобного инструмента для хранения и поиска рецептов
- Создание сообщества пользователей для обмена кулинарными идеями
- Демонстрация современных подходов к веб-дизайну и пользовательскому интерфейсу
- Обучение и практика в области frontend-разработки

3. Технологический стек

3.1. Frontend-технологии

- HTML5 — структура и семантическая разметка страниц
- CSS3 — стилизация с использованием современных техник
- JavaScript (ES6+) — интерактивность и динамическое поведение
- Bootstrap 5.3.0 — адаптивный layout и компоненты

3.2. Библиотеки и зависимости

- **Font Awesome 6.4.0** — иконки
- **Bootstrap Bundle** — JavaScript компоненты
- **LocalStorage API** — локальное хранение данных

4. Структура проекта

4.1. Файловая организация

- **index.html** — страница входа в систему
- **register.html** — страница регистрации
- **search.html** — страница поиска рецептов
- **recipe.html** — детальная страница рецепта
- **profile.html** — личный кабинет пользователя
- **css/style.css** — основные стили
- **js/main.js** — основная логика приложения

5. Функциональные возможности

5.1. Система аутентификации

Реализованы функции входа и регистрации пользователей с сохранением состояния в LocalStorage:

- Валидация email и пароля
- Функция «Запомнить меня»
- Интеграция с социальными сетями (Google, Facebook) — UI готов
- Восстановление пароля — интерфейс реализован

5.2. Поиск и фильтрация рецептов

Многоуровневая система фильтрации:

- По сложности приготовления (легко, средне, сложно)
- По типу кухни (итальянская, азиатская, американская, русская, французская)
- Динамическое обновление списка рецептов при выборе фильтров

5.3. Работа с рецептами

Пользователи могут:

- Просматривать детальную информацию о рецепте

- Сохранять рецепты в избранное
- Ставить лайки рецептам
- Делиться рецептами в социальных сетях
- Комментировать рецепты
- Копировать список ингредиентов в буфер обмена

5.4. Вспомогательные функции

- Плавная прокрутка по якорям
- Анимация появления элементов (Intersection Observer API)
- Отложенная загрузка изображений (lazy loading)
- Toast-уведомления о действиях пользователя
- Debounce для оптимизации событий

6. Дизайн и пользовательский интерфейс

6.1. Концепция дизайна «Glass Morphism»

Проект использует современный тренд glassmorphism (эффект матового стекла):

- Полупрозрачные контейнеры с размытием фона (backdrop-filter)
- Градиентный фон с радиальными эффектами
- Тонкие светлые границы для выделения элементов
- Плавные тени и переходы

6.2. Цветовая палитра

- Основной градиент: фиолетовый (#667eea) → пурпурный (#764ba2)
- Дополнительные акценты: розовый и голубой
- Белый текст для контраста на темном фоне

6.4. Интерактивность

- Hover-эффекты на карточках рецептов
- Анимация при появлении элементов
- Плавные переходы между состояниями
- Визуальная обратная связь при взаимодействии

7. Архитектура кода

7.1. JavaScript модули

Код организован в логические модули:

- **Storage** — работа с LocalStorage
- **RecipeData** — управление данными рецептов
- **Auth** — аутентификация и авторизация
- **RecipeApp** — глобальный объект с утилитами

7.2. Утилитарные функции

- `validateForm()` — валидация форм
- `showToast()` — уведомления пользователя
- `formatTime()` — форматирование времени
- `formatNumber()` — форматирование чисел
- `copyToClipboard()` — копирование в буфер

8. Особенности реализации

8.1. Производительность

- Использование Intersection Observer для ленивой загрузки
- Debounce для оптимизации частых событий
- Минимизация DOM-операций

8.2. Пользовательский опыт

- Плавные анимации для улучшения восприятия
- Информативные уведомления о действиях

- Визуальная обратная связь на все действия
- Адаптивный интерфейс для всех устройств

8.3. Безопасность

- Обработка ошибок в try-catch блоках
- Валидация пользовательского ввода
- Безопасное использование LocalStorage

9. Возможности для развития

9.1. Backend-интеграция

- Разработка REST API для работы с данными
- Реализация системы аутентификации на сервере
- Интеграция с базой данных (PostgreSQL)
- Реализация загрузки изображений

9.2. Функциональные улучшения

- Добавление видео-рецептов
- Система рейтингов и отзывов
- Создание собственных коллекций рецептов
- Расширенный поиск с использованием фильтров
- Персонализированные рекомендации

9.3. Технические улучшения

- Миграция на современный фреймворк (Vue.js, React)
- Внедрение системы сборки (Webpack, Vite)
- Оптимизация производительности
- Добавление unit-тестов
- PWA-функциональность (offline режим)

10. Выводы

Проект «Кулинарный Портал» демонстрирует современный подход к разработке веб-приложений с акцентом на пользовательский опыт и визуальную привлекательность. Основные достижения проекта:

- Реализован привлекательный и современный интерфейс с использованием glassmorphism дизайна
- Создана базовая функциональность для работы с рецептами (поиск, фильтрация, сохранение)
- Применены современные веб-технологии и best practices фронтенд-разработки
- Обеспечена полная адаптивность для различных устройств
- Заложена основа для дальнейшего развития и масштабирования

Проект может служить отличной основой для создания полноценного веб-приложения с backend-частью. Для продолжения разработки рекомендуется:

- Разработать серверную часть на Golang с использованием REST API
- Интегрировать PostgreSQL для хранения данных
- Мигрировать frontend на Vue.js для улучшения архитектуры
- Внедрить систему тестирования
- Оптимизировать производительность и добавить PWA-функциональность