Министерство образования Новосибирской области

ГБПОУ НСО «Новосибирский авиационный технический колледж имени Б.С. Галущака»

Руководство программиста

Выполнил: Конюк И.М.

Группа: ПР-22.106  
Проверил: Козлов М.Д.

2025

**Аннотация**  
В настоящем документе описаны особенности разработанного веб-приложения, реализованного на JavaScript с использованием React и Tailwind CSS. Приложение состоит из двух основных модулей: страницы авторизации, где пользователь вводит логин и пароль, и страницы с историей колледжа, содержащей текстовое описание и иллюстрирующую информацию (изображения). Руководство предназначено для разработчиков, сопровождающих и модернизирующих систему

**Список принятых терминов и сокращений**

**- JS** – JavaScript

**- React** – библиотека для построения пользовательских интерфейсов

**- Tailwind CSS** – фреймворк для быстрого создания адаптивного дизайна

**- API** – интерфейс программирования приложений

**- UI** – пользовательский интерфейс

**- UX** – удобство пользования

- localhost - локальный адрес компьютера

Оглавление

[1. Общие положения 5](#_Toc9444)

[1.1 Наименование программы 5](#_Toc13346)

[2. Назначение и условия применения программы 6](#_Toc16949)

[2.1 Назначение программы 6](#_Toc11975)

[2.2 Функции, выполняемые программой 6](#_Toc1572)

[2.3 Сведения о технических и программных средствах, обеспечивающих выполнение программы 6](#_Toc17539)

[3. Характеристика программных средств программы 8](#_Toc10484)

[3.1 Описание основных характеристик программы 8](#_Toc5675)

[4. Обращение к программе 9](#_Toc21334)

[4.1 Запуск процедур для интеграции с внешними системами 9](#_Toc20498)

[4.2 Создание скриптов 9](#_Toc13590)

[5. Входные и выходные данные 10](#_Toc19600)

[5.1 Характер и организация входных данных 10](#_Toc27133)

[5.2 Характер и организация выходных данных 10](#_Toc32727)

[6. Сообщения 11](#_Toc26329)

# Общие положения

## 1.1 Наименование программы

Полное наименование системы: College History WebApp  
(В дальнейшем – Приложение)

# Назначение и условия применения программы

## 2.1 Назначение программы

Приложение предназначено для:

- Предоставления пользователю возможности аутентификации посредством ввода логина и пароля.

- Демонстрации информационной страницы с историей колледжа, включающей текстовое описание и изображения.

## 2.2 Функции, выполняемые программой

Основные функции Приложения включают:

**- Аутентификация пользователя:** Валидация учетных данных при входе в систему.

**- Отображение контента:** Вывод страницы с историей колледжа, содержащей описательный текст и прикрепленные изображения.

**- Интеграция:** Возможность подключения к внешним информационным системам через REST API для обновления или получения данных.

## 2.3 Сведения о технических и программных средствах, обеспечивающих выполнение программы

##### 2.3.1 Требования к составу и параметрам комплекса технических средств

- Серверное оборудование для хостинга Node.js-приложения (минимум: 2 ядра CPU, 4 ГБ ОЗУ).

- Сетевое оборудование, обеспечивающее стабильное подключение к Интернету.

- Клиентские устройства: персональные компьютеры, планшеты, смартфоны с современными браузерами.

##### 2.3.2 Требования к общесистемному программному обеспечению

- Операционные системы: Linux, Windows или macOS (совместимость с современными ОС).

- Веб-сервер (например, Nginx или Apache) для организации работы приложения.

- Node.js версии 14 и выше.

- Современные браузеры (Google Chrome, Mozilla Firefox, Microsoft Edge).

##### 2.3.3 Требования к персоналу

- Разработчики с опытом работы с JavaScript, React и Tailwind CSS.

- Специалисты по обеспечению качества (QA) для проведения функционального и интеграционного тестирования.

- Системные администраторы для настройки серверного окружения и поддержки работоспособности приложения.

# Характеристика программных средств программы

## 3.1 Описание основных характеристик программы

##### 3.1.1 Временные характеристики программы

Приложение не содержит специальных оптимизаций для сокращения времени отклика. Оно предназначено для проверки работы Node.js, поэтому производительность соответствует базовым возможностям среды выполнения без дополнительной оптимизации.

##### 3.1.2 Режим работы программы

Приложение функционирует как простой веб-сервер на Node.js, обслуживая страницу авторизации и страницу с историей колледжа. Приложение не рассчитано на работу в условиях высокой нагрузки и не включает расширенных механизмов управления ресурсами.

##### 3.1.3 Средства контроля правильности выполнения программы

В реализации отсутствуют специализированные средства логирования или мониторинга. Корректность работы приложения проверялась посредством ручного тестирования и использования стандартных обработчиков ошибок Node.js.

##### 3.1.4 Средства контроля самовосстанавливаемости программы

Отсутствуют механизмы автоматического восстановления и резервного копирования. Работа приложения обеспечивается за счёт стандартных возможностей операционной системы и среды выполнения Node.js без дополнительных мер по самовосстановлению.

## Обращение к программе

## 4.1 Запуск процедур для интеграции с внешними системами

- Приложение предоставляет REST API для обмена данными с внешними информационными системами.

- Интеграция осуществляется через установленные веб-сервисы, что позволяет динамически обновлять контент страницы истории колледжа.

## 4.2 Создание скриптов

Для сборки проекта используется npm-скрипты:

- npm start – запуск сервера разработки;

- npm run build – сборка оптимизированной версии приложения;

- npm test – запуск тестов.

Дополнительные скрипты могут использоваться для автоматизации деплоя и мониторинга.

## Входные и выходные данные

## 5.1 Характер и организация входных данных

**Аутентификация:**

- Входными данными являются логин и пароль, введённые пользователем.

- Данные передаются через защищённое соединение (HTTPS) на сервер для проверки корректности.

**Запросы к API:**

- Входные данные для получения информации о колледже передаются в виде HTTP-запросов с параметрами (например, идентификатор страницы или фильтры).

## 5.2 Характер и организация выходных данных

**Ответы сервера:**

- Сообщения об успешной аутентификации или возникших ошибках (например, неверный пароль).

- HTML-страница, содержащая текстовую информацию и изображения, отображаемые пользователю.

- Ответы API в формате JSON для интеграционных сценариев.

## Сообщения

Приложение генерирует следующие типы сообщений:.

**Предупреждения:** Сообщения о необходимости обновления данных или временных ограничениях в работе.

**Ошибки:** Сообщения о некорректном вводе данных, проблемах с сервером или отсутствии соединения с внешними системами (например, «Ошибка авторизации»)