**Сайт для редакции изображений**

**Руководство пользователя**

**Содержание**

[1 Введение 2](#_Toc181120925)

[1.1 Наименование системы 2](#_Toc181120926)

[1.2 Область применения 2](#_Toc181120927)

[1.3 Условные обозначения системы 2](#_Toc181120928)

[2 Установка системы 3](#_Toc181120929)

[2.1 Требование для установки 3](#_Toc181120930)

[2.2 Установка зависимостей 3](#_Toc181120931)

[2.3 Настройка базы данных 3](#_Toc181120932)

[2.4 Запуск сервера 3](#_Toc181120933)

[3 Внутреняя структура системы 4](#_Toc181120934)

[3.1 Используемый технологический стек 4](#_Toc181120935)

[3.2 Функции и их описание 4](#_Toc181120936)

[3.3 Структура данных 4](#_Toc181120937)

[3.4 Описание API 6](#_Toc181120938)

# Введение

## Наименование системы

Сайт для редакции изображений.

## Область применения

Сайт для редакции изображений представляет собой веб-приложение, предназначенное для быстрого и просто редактирования изображений с помощью мощного инструментария. Основная цель сайта для редакции изображений – предоставить пользователям быстрый, простой и бесплатный способ редактирования изображений прямо в браузере, без необходимости обязательной регистрации и загрузки дополнительного ПО на устройство.

## Условные обозначения системы

Условное обозначение сайта для редакции изображений – Сайт.

# Установка системы

## Требование для установки

Для развертывания Сайта нужен сервер, с поддержкой Node.js с фреймворком Express.js и PostgreSQL.

Сайт должен поддерживать все современные браузеры, такие как: Chrome, Firefox, Edge, Opera, Safari и т.д.

## Установка зависимостей

Фронтенд: установите зависимости для фронтенда (если есть), выполнив следующую команду:

npm install имя\_библиотеки

или через тег <script> в HTML-файле, используя ссылку на CDN. Например, для подключения Konva.js:

<script src="https://cdn.jsdelivr.net/npm/konva@8.3.5/konva.min.js"></script>

Бекенд: установите необходимые библиотеки node.js (если применимо), например, через Composer:

npm install express

## Настройка базы данных

Чтоб установить базу данных на сервер нужно создать новую базу данных в PostgreSQL и импортировать готовую структуру из уже существующей базы данных.

## Запуск сервера

Чтобы запустить сервер, в папке проекта нужно открыть терминал и вписать следующую команду:

npm start

или

npm run

# Внутреняя структура системы

## Используемый технологический стек

Сайт использует HTML, CSS и JS с библиотеками Jimp, Kaman.js и Konva.js для фронтенда, и Node.js с фреймворком Express.js для бекенда. В качестве СУБД используется PostgreSQL.

## Функции и их описание

У Сайта имеются следующие функции:

* Регистрация – создание аккаунта для пользователя по введённым логину и паролю. Логины пользователей должны быть уникальными. При создании аккаунты соответствующие данные об этом аккаунте вносятся в базу данных;
* Авторизация – авторизация пользователя на сайте по его логину и паролю. Происходит проверка логина и пароля в базе данных на соответствие для корректной авторизации;
* Загрузка изображения для редактирования – загрузка изображения с устройства пользователя на сайт для его дальнейшего редактирования;
* Загрузка отредактированного изображения на устройство пользователя – загрузка изменённого изображения на устройство пользователя;
* Изменение уровней изображения – изменение таких параметров изображения, как: яркость, контраст, насыщенность и т.д.;
* Обрезка изображения – обрезка изображения в соответствии с установленными пользователем параметрами;
* Наложение визуальных эффектов на изображение – наложение на изображение заранее заготовленных шаблонов с настроенными параметрами уровней в определённых пропорциях, для достижения визуального эффекта (например: чёрно белый фильтр, монохром и т.д.);
* Вставка текста в изображение – вставка текста в изображение в соответствии с параметрами, установленными пользователем (положение текста, шрифт, размер шрифта, цвет и т.д.);
* Рисование на изображении – нанесение рисунка на изображение при помощи мыши в соответствии с установленными параметрами пользователя;
* Помещение изображения в рамку – помещение изображения в рамку в соответствии с параметрами, установленными пользователем (форма рамки, цвет, толщина и т.п.);
* Удаление заднего фона изображения – удаление заднего фона главного объекта изображения (кадра) (Изменение заднего фона таким образом, чтобы он стал полностью прозрачный);
* Создание шаблона – создание заготовки с именем для дальнейшего переиспользования, представляющей собой заранее установленные значения для различных инструментов в соответствии с параметрами, установленными пользователем.

## Структура данных

Сайт имеет базу данных, структура которой изображена в виде логической и физической ER-диаграмм на рисунках 1 и 2 соответственно.

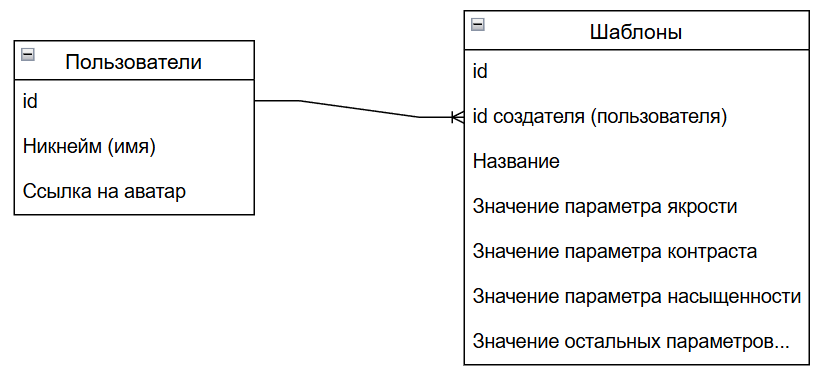


Рисунок 1 – Логическая ER-диаграмма

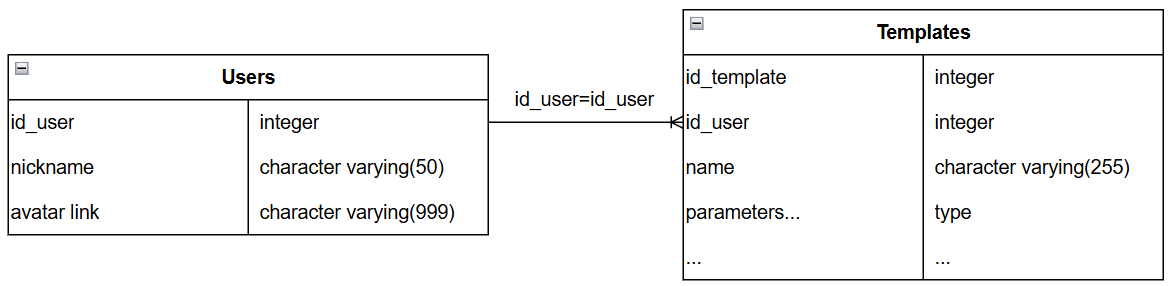


Рисунок 2 – Физическая ER-диаграмма

В данной базе данные есть две таблицы: «Пользователи» и «Шаблоны».

* Таблица «Пользователи» содержит следующие данные:
* id: идентификатор пользователя;
* Никнейм (имя): отображаемое имя пользователя;
* Ссылка на аватар: ссылка на аватар пользователя.

Таблица «Шаблоны» содержит следующие данные:

* id: идентификатор шаблона;
* id создателя (пользователя): идентификатор создателя шаблона;
* Название: название шаблона;
* Параметры: значение параметра изображения (яркость, контраст, насыщенность и т.д.).

## Описание API

На Сайте используются следующие API запросы:

* POST-запросы – используются для регистрации, авторизации, выхода из аккаунта и загрузки изображения для редактирования на Сайт;
* GET-запросы – используются для получения информации об изображении, получения информации о шаблонах и скачивания изображения на устройство пользователя;
* PUT-запрос – используется для изменения существующего шаблона;
* DELETE-запрос – используется для удаления шаблона.