Министерство науки и высшего образования РФ

Федеральное государственное автономное

образовательное учреждение высшего образования

**«СИБИРСКИЙ ФЕДЕРАЛЬНЫЙ УНИВЕРСИТЕТ»**

|  |
| --- |
| Институт космических и информационных технологий |
| институт |
| Программная инженерия |
| кафедра |

**КУРСОВОЙ ПРОЕКТ**

|  |
| --- |
| Разработка лендинга для продвижения услуг DESC-Studio |
| тема проекта |
|  |
|  |
|  |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Руководитель | |  |  |  | П.В.Пересунько |
|  | |  | подпись, дата |  | инициалы, фамилия |
| Студент | КИ22-17/1Б, 032214416 |  |  |  | И. А. Богачев |
|  | номер группы, зачётной книжки |  | подпись, дата |  | инициалы, фамилия |
| Студент | КИ22-17/1Б, 032217462 |  |  |  | Ю. В. Восканян |
|  | номер группы, зачётной книжки |  | подпись, дата |  | инициалы, фамилия |
| Студент | КИ22-17/1Б, |  |  |  | А. В. Балденко |
|  | номер группы, зачётной книжки |  | подпись, дата |  | инициалы, фамилия |

Красноярск 2024

**СОДЕРЖАНИЕ**

[ВВЕДЕНИЕ 3](#_Toc166503246)

[1 Проектирование 4](#_Toc166503247)

[2 Разработка минимального стоящего продукта 4](#_Toc166503248)

[2.1 Первый спринт 4](#_Toc166503249)

[2.2 Второй спринт 4](#_Toc166503250)

[2.3 Третий спринт 4](#_Toc166503251)

[2.4 Четвертый спринт 5](#_Toc166503252)

[3 Подготовка релиза 5](#_Toc166503253)

[3.1 Пятый спринт 5](#_Toc166503254)

[3.2 Шестой спринт 5](#_Toc166503255)

[3.3 Седьмой спринт 5](#_Toc166503256)

[3.4 Восьмой спринт 5](#_Toc166503257)

[4 Инструкция по использованию 5](#_Toc166503258)

[ЗАКЛЮЧЕНИЕ 6](#_Toc166503259)

[СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ 7](#_Toc166503260)

[ПРИЛОЖЕНИЕ А (рекомендуемое) Отрисовка UserFlow 8](#_Toc166503261)

[ПРИЛОЖЕНИЕ Б (рекомендуемое) Диаграмма окон 9](#_Toc166503262)

[ПРИЛОЖЕНИЕ В (рекомендуемое) Диаграмма последовательности 10](#_Toc166503263)

# ВВЕДЕНИЕ

В современных школах и учебных заведениях отслеживание выполнения домашних заданий остается в большей степени ручным процессом, требующим значительного времени и усилий со стороны как учителей, так и учеников. Традиционные методы, такие как записи в тетрадях или отдельные электронные таблицы, могут быть неэффективными и недостаточно удобными для массового использования. Это может приводить к недопониманиям, упущенным срокам выполнения заданий и недостаточной обратной связи между учителем и учеником.

В рамках данной работы поставлена цель разработать сервис для управления домашними заданиями, посещениями для школы .

Для достижения этой цели были сформулированы следующие задачи:

1. Создание удобного дизайна для сервиса;
2. Разработка гибкой базы данных;
3. Разработка сервиса по дизайну;
4. Тестирование сервиса.

Используемые методы и средства решения обеспечат эффективность проекта, делая его адаптированным к современным требованиям и ожиданиям заказчика. Новизна данной проблемы заключается в стремлении к созданию программного решения, которое упростит и улучшит процесс отслеживания и контроля выполнения домашних заданий для учеников и учителей. Такое программное обеспечение должно предоставлять возможности для более эффективного взаимодействия между учениками и учителями, автоматизации процесса сбора и анализа данных о выполненных заданиях, а также обеспечивать более надежную и точную обратную связь. Новаторство заключается в том, что это программное обеспечение должно быть специально адаптировано к образовательной среде и учитывать ее специфические потребности и требования.

# Проектирование

При проектировании сервиса был проведен анализ целей и задач, в ходе которого были прописаны сценарии использования системы.

Сценарий использования системы для ученика состоит из следующих шагов:

1. Ученик заходит на сайт.
2. Ученик входит в аккаунт по выданным данным.
3. Ученик просматривает отсортированный по дате сдачи, предмету и группе список запланированных домашних заданий.
4. Ученик выбирает одно из домашних заданий, и переходит на его страницу. На ней он может ознакомиться с полным описанием, скачать прикрепленные файлы.
5. После ознакомления с домашним заданием ученик может оставить ответ на него, прикрепив к ответу файлы по необходимости.
6. После оставления ответа на домашнее задание ученик дожидается, когда учитель его проверит, и после этого может увидеть оценку и комментарий учителя. В случае необходимости, ученик может повторно оставить ответ и повторно увидеть комментарий учителя.

Сценарий использования системы для учиетля состоит из следующих шагов:

1. Учитель заходит на сайт.
2. Учитель регистрируется, ввода
3. Учитель просматривает все
4. Администратор переходит на специализированную страницу, открывается форма для авторизации (пароль и логин известны заранее). Вводит корректные данные, происходит переход в админ-панель, в которой находится форма для изменения информации на сайте. Для изменения информации нужно нажать на элемент сайта, ввести в специальное окно html-код или текст, на который будет происходить замена. Для сохранения изменений необходимо нажать кнопку «Сохранить».

Также были построены диаграмма вариантов использования и диаграмма деятельности (рисунки 1 и 2).

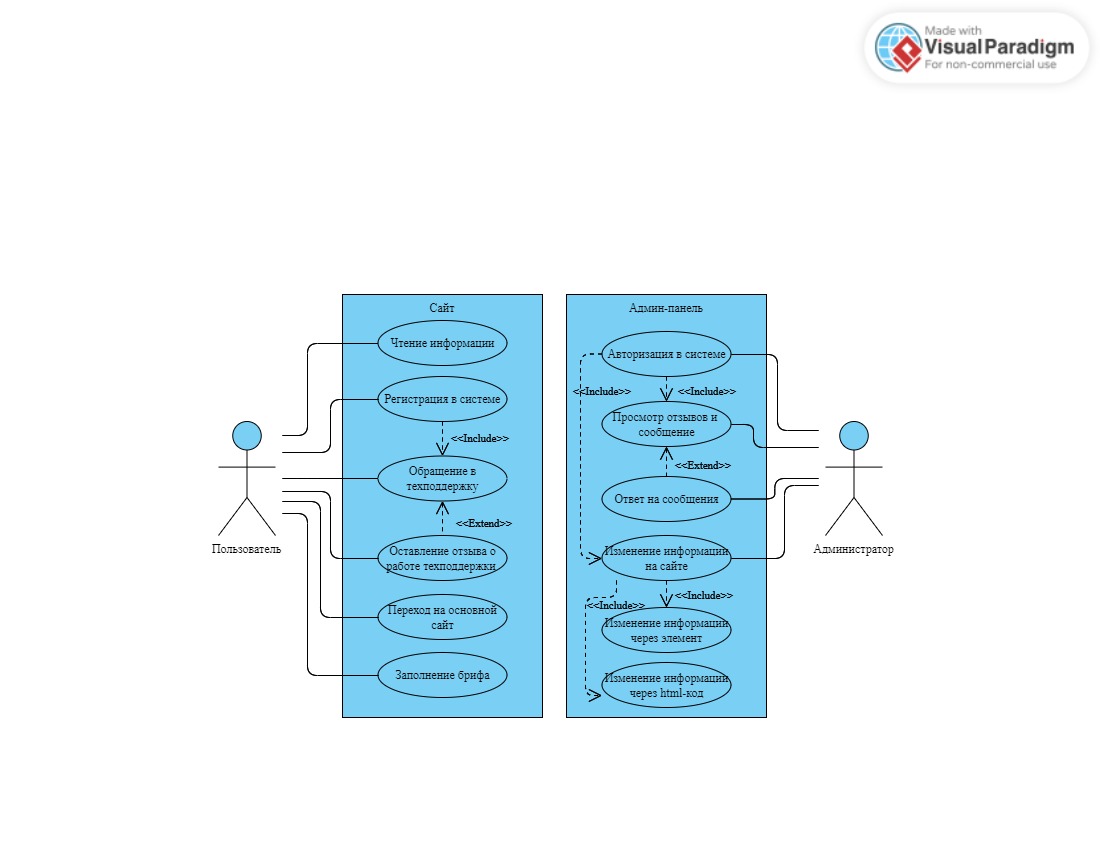


Рисунок 1 – Диаграмма вариантов использования

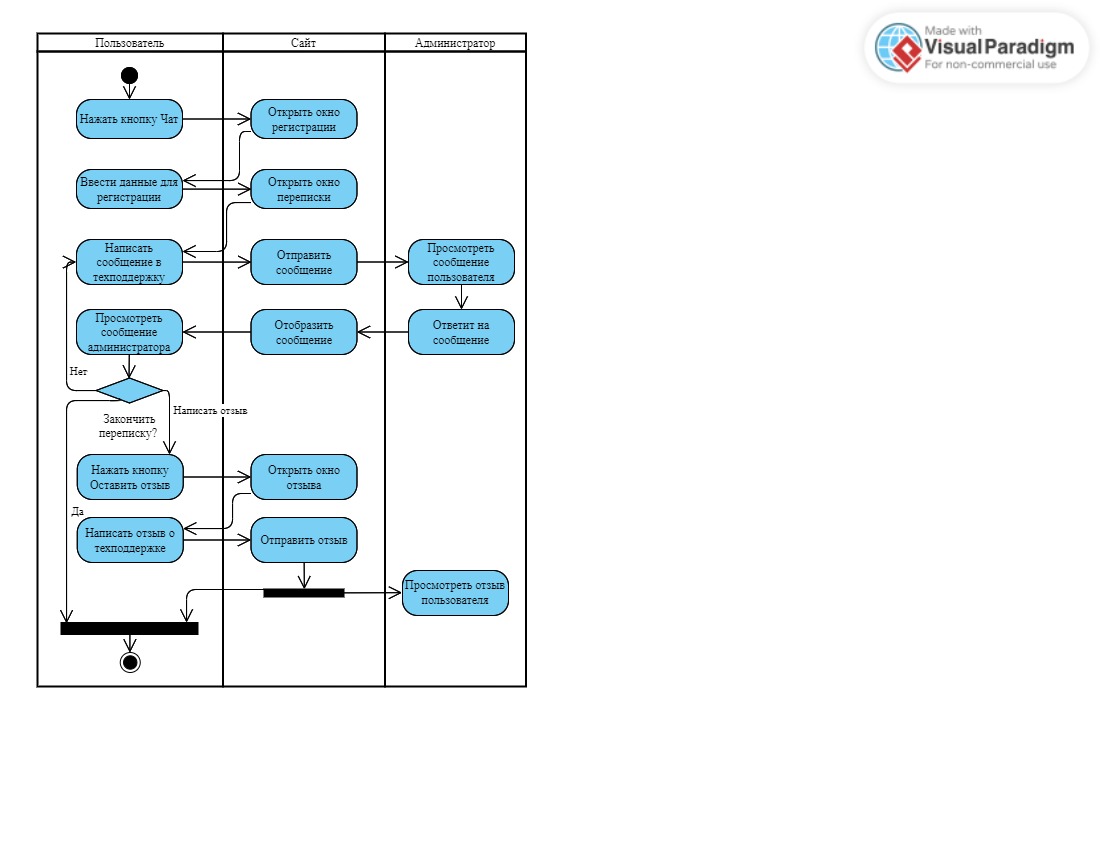


Рисунок 2 – Диаграмма деятельности

При создании сайта используется класс, у которого есть данные. Так как они не должны потеряться при повторном создании экземпляра класса, было принято решение реализовать паттерн Singleton, так как он гарантирует, что класс имеет только один экземпляр, и в таком случае данные с активными websocket соединениями не будут утеряны.

# Разработка минимального стоящего продукта

Разработка минимального стоящего продукта является одной из главных задач данной работы, так как он позволит быстро запустить продукт на рынок, начать получать обратную связь от пользователей, сэкономить время и ресурсы на разработку функционала, который не приносит реальной ценности пользователям и проверить гипотезы и предположения о продукте на практике.

Для качественной работы над проектом был создан список задач, который был разделен по спринтам.

## Первый спринт

Первый спринт является основой для сайта, так как в нём создаются макеты баз данных и дизайна.

Задачи на первый спринт представлены в таблице 1.

Таблица 1 – Задачи первого спринта

|  |  |  |  |
| --- | --- | --- | --- |
| **Задача** | **Описание задачи** | **Вес** | **ФИО исполнителя** |
| 1. Разработать дизайн | Задача состоит из разработки концепции сайта и прототипа, а также из создания итогового варианта. | 3 | Смышляева С.А. |
| 2. Сверстать макет  сайта | Сверстать сайт с помощью html, css, js | 5 | Балденко А.В. |
| 3. Реализовать техподдержку без cookie | Реализовать техподдержку без cookie для пользователей | 3 | Восканян Ю.В. |
| 4. Спроектировать БД | Реализовать схему базы данных в графическом виде | 1 | Восканян. Ю.В. |
| 5. Создать таблицы в PGADMIN | Создать таблицы в СУБД Pgadmin 4 | 1 | Восканян. Ю.В. |
| 6. Проверить адаптацию под различные устройства | С помощью режима разработчика проверить адаптацию под различные устройства | 1 | Балденко А.В. |

Смышляева С.А. разработала концепцию, прототип и итоговый вариант сайта. Основой для внешнего вида сайта послужил брендбук студии DESK, являющейся заказчиком данного сайта. Инструментом для создания служила онлайн-платформа Figma. Результат работы представлен на рисунках 1 и 2.



Рисунок 1 – Мобильная версия сайта

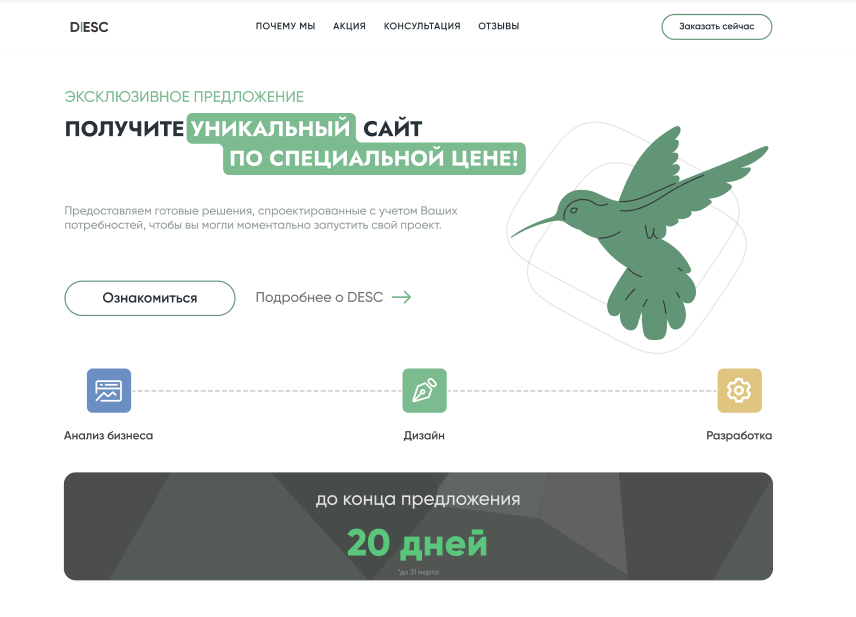


Рисунок 2 – Компьютерная версия сайта

Восканян Ю. В. реализовал техподдержка без cookie (рисунок 3), спроектировал базу данных (рисунок 4) и создал таблицы в PgAdmin (PostgreSQL) (рисунок 5).

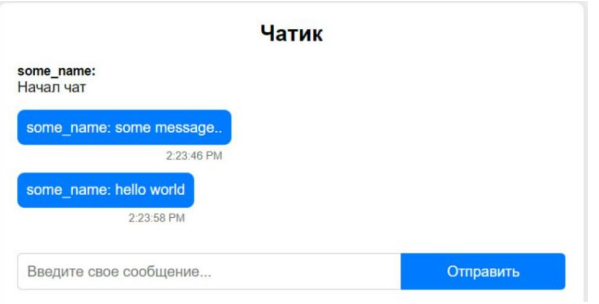
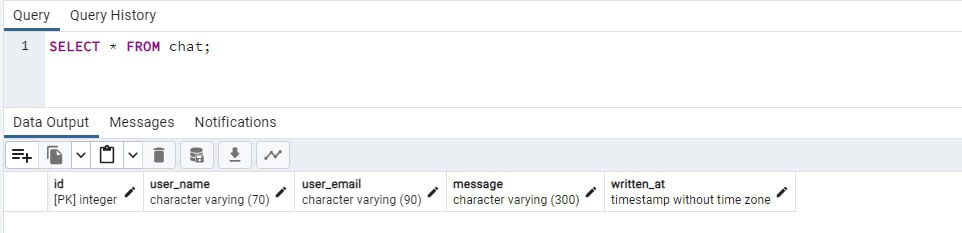
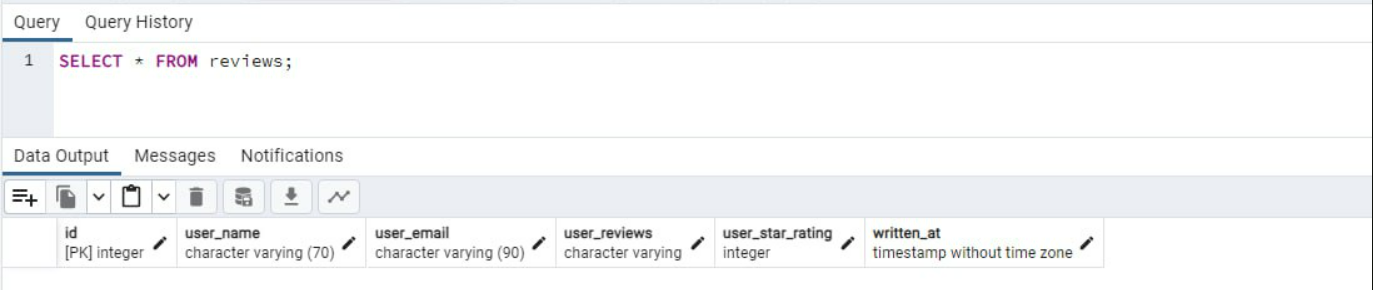


Рисунок 3 – Техподдержка



Рисунок 4 – Спроектированная база данных





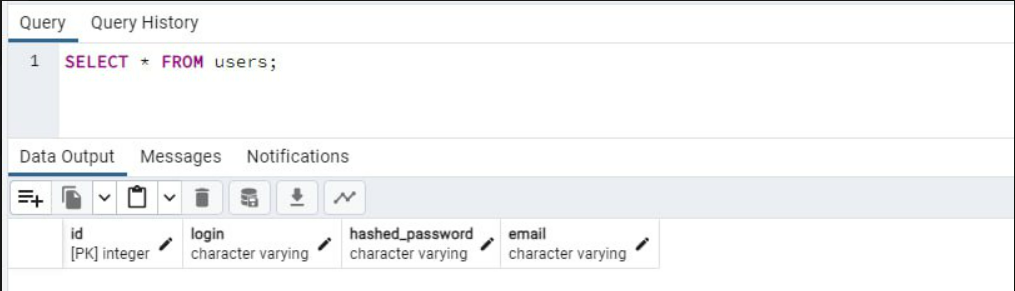


Рисунок 5 – Созданные таблицы в Pgadmin

## Второй спринт

Задачи на второй спринт представлены в таблице 2.

Таблица 1 – Задачи первого спринта

|  |  |  |  |
| --- | --- | --- | --- |
| **Задача** | **Описание задачи** | **Вес** | **ФИО исполнителя** |
| 1. Проверить адаптацию под различные устройства | С помощью режима разработчика проверить адаптацию под различные устройства | 1 | Балденко А.В. |
| 1. Реализовать паттерны | Создание паттернов Singleton | 3 | Восканян Ю. В. |
| 1. Подключить таблицы к python через SQLAlchemy | Создать модель таблиц в Python | 1 | Восканян Ю. В. |
| 1. Сделать макет админ-панели | Создать прототип и готовый вариант админ-панели | 3 | Смышляева С.А. |
| 1. Реализовать техподдержку с cookie | Реализовать техподдержку с cookie для пользователей | 3 | Восканян Ю.В. |

Смышляева С.А. создала прототип дизайна админ-панели (рисунок).

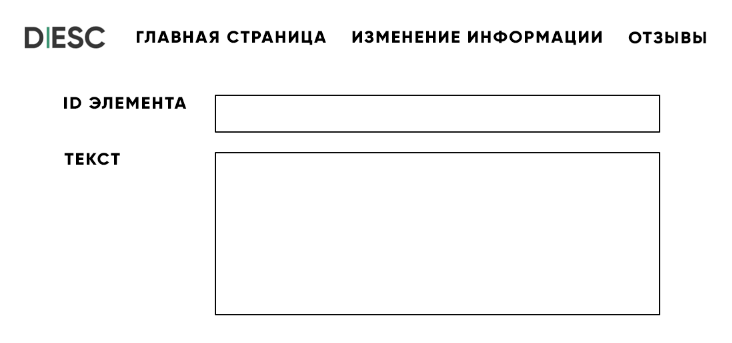


Рисунок – Админ-панель

В дальнейшем наша команда пришла к другому виду и типу админ панели.

Восканян Ю. В. реолизовал паттерн «Одиночка» (Singleton), подключил таблицы к python через SQLAlchemy, реализовал техподдержку с cookie (рисунок).



Рисунок – Реализация техподдержки с cookie

Паттерн Singleton был создан для класса ConnectionManager (рисунок), который хранит активные websocket-соединения, и для того, чтобы данные не были потеряны при повторном создании экземпляра класса.



Рисунок – Реализация паттерна Singleton

Для работы с таблицами через Python, была использована библиотека SQLALchemy, которая способна взаимодействовать с таблицами баз данных, используя классы языка программирования.



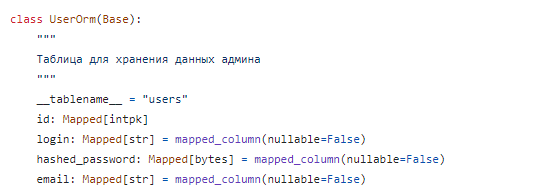


Рисунок – Модели таблиц

## Третий спринт

\*Текст\*

## Четвертый спринт

\*Текст\*

# Подготовка релиза

\*Текст\*

## Пятый спринт

\*Текст\*

## Шестой спринт

\*Текст\*

# Инструкция по использованию

Для работы с сайтом были разработаны инструкции для пользователя и администратора.

Пользователь заходит на сайт https://desc-studio.online/index.html и читает информацию (рисунок 3).

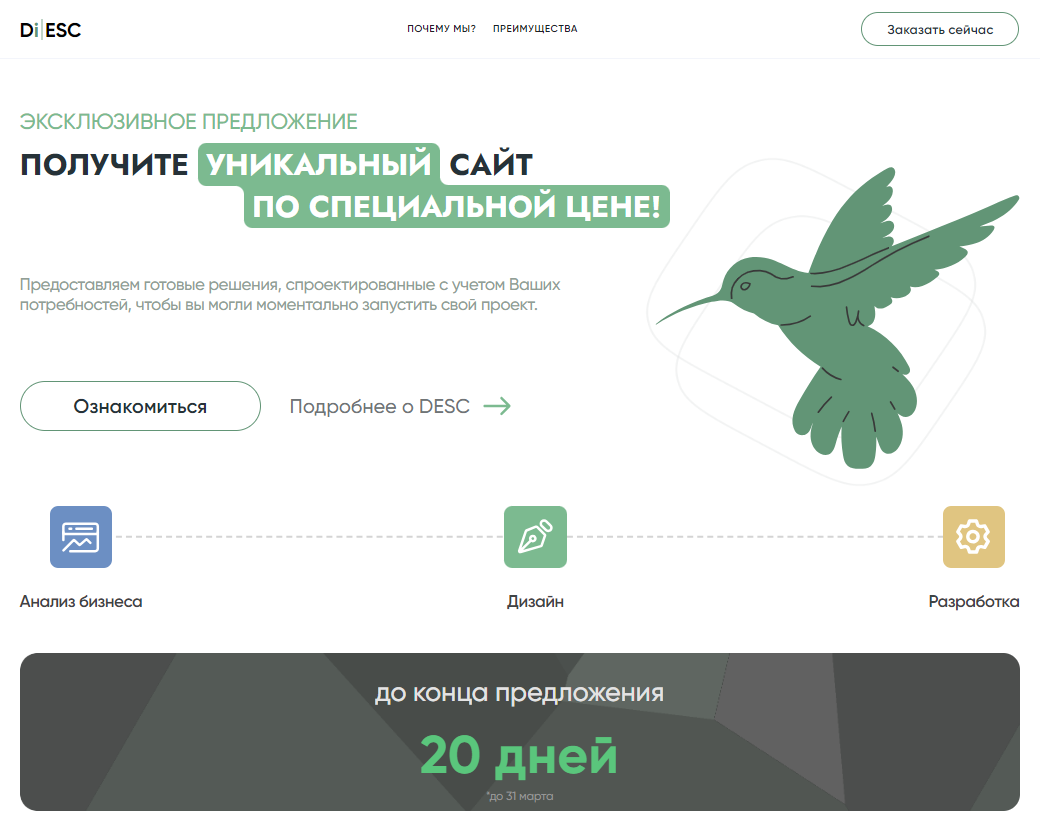


Рисунок 3 – Страница сайта

Если у пользователя возникают вопросы, то он может написать в техподдержку, для этого в правом нижнем углу есть кнопка «Чат» (рисунок 4).

  
Рисунок 4 – Кнопка «Чат»

Пользователь кликает на неё, вводит свои данные и начинает беседу (рисунок 5 и 6), так же можно оставить отзыв, нажав кнопку «Оставить отзыв» (рисунок 6). В отзыве пишется текст и оценка работы техподдержки. После чего нажимается кнопка «Отправить».

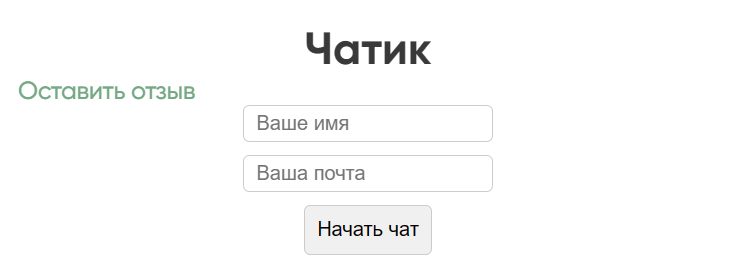


Рисунок 5 – Регистрация пользователя

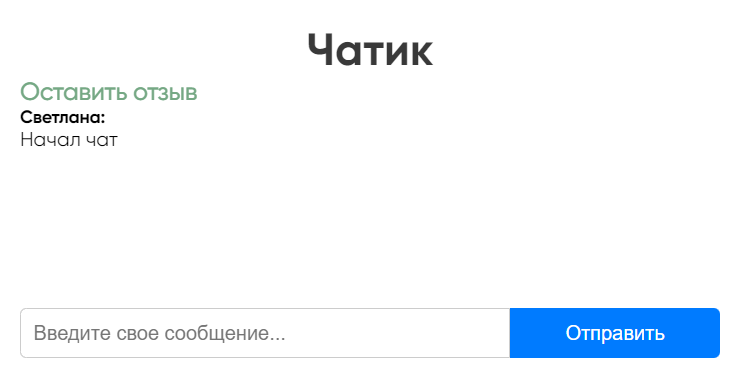


Рисунок 6 – Окно чата

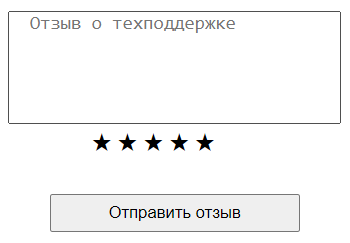


Рисунок – Окно отправки отзыва

Если пользователь хочет познакомиться с digital-студией DESC-Studio, то он может перейти на основной сайт, нажав кнопку «Подробнее о DESK» (рисунок).



Рисунок – Кнопка перехода на основной сайт

Если пользователь хочет заказать услугу, то он может перейти на заполнение брифа, нажав кнопку «Заказать сейчас» (рисунок).



Рисунок – Кнопка для заполнения брифа

Администратор переходит на специализированную страницу, открывается форма для авторизации (рисунок). Вводит данные, происходит переход в админ-панель, в которой находятся отзывы пользователей о техподдержке и их сообщения (рисунок).

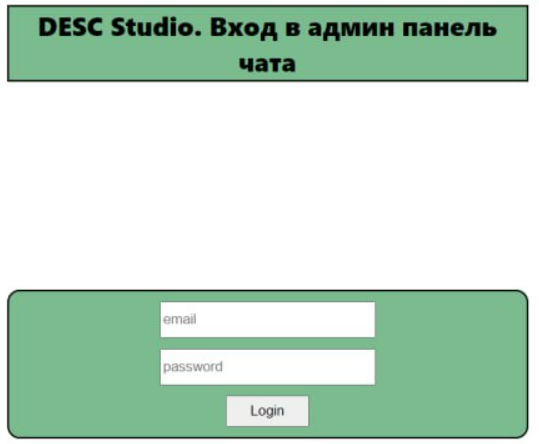


Рисунок – Форма авторизации

Для ответа на сообщение пользователя нужно ввести текст в специальное окно и нажать кнопку «Отправить» (рисунок).

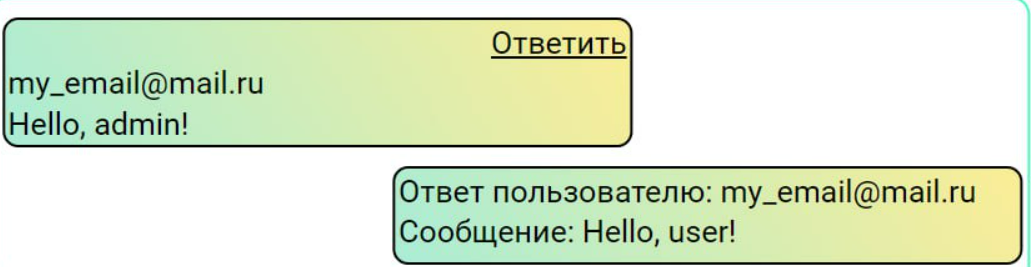


Рисунок – Ответ на сообщение пользователя

Для изменения информации на сайте администратор переходит на специализированную страницу, открывается форма для авторизации (рисунок).

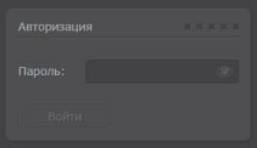


Рисунок – Форма авторизации

Вводит данные, происходит переход в админ-панель, в которой находится форма для изменения информации на сайте. Для изменения информации нужно нажать на элемент сайта, ввести в специальное окно html-код или текст, на который будет происходить замена. Для сохранения изменений необходимо нажать кнопку «Сохранить» (рисунок).

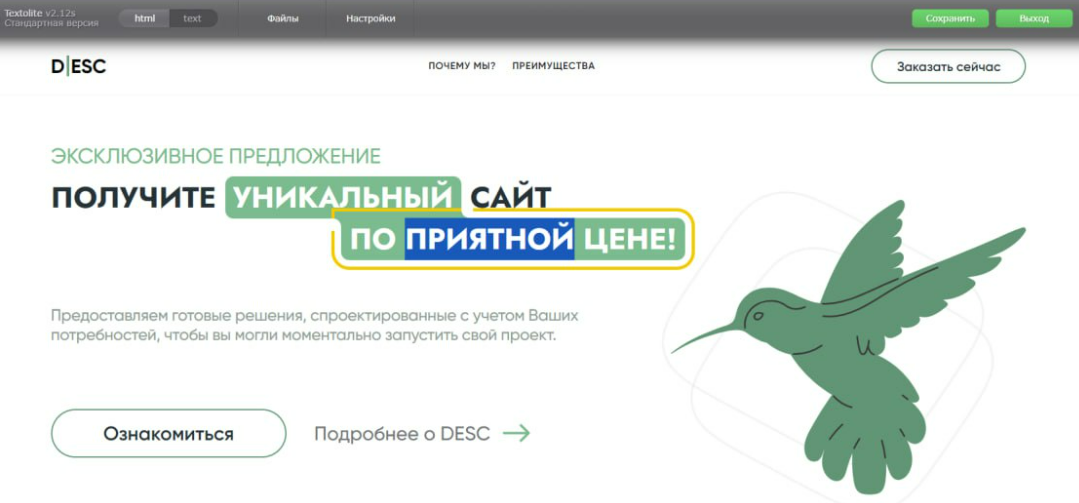


Рисунок – Изменение информации на сайте

# ЗАКЛЮЧЕНИЕ

В рамках данной работы был разработан одностраничный web-сайт для digital-студии.

Для создания сайта использовались технологии HTML, CSS, JavaScript и jQuery. Для возможности редактирования содержимого сайта был интегрирован плагин Textolite. Особое внимание было уделено административной панели, для которой использовался framework FastAPI, обеспечивающий эффективную обработку и управление сообщениями, получаемыми через форму технической поддержки. Сайт был создан полностью адаптивным, что гарантирует его корректное отображение на различных устройствах, включая мобильные телефоны и планшеты.

Как итог, данные инструменты и подходы позволили создать надежную и функциональную платформу, отвечающую современным требованиям и потребностям заказчика.

# СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

Иванов А. Современные методики веб-разработки / А. Иванов // Журнал Web-разработчиков – 2023. – Т. 28, № 2. – С. 45–60.

Петрова Е. Значение дизайна пользовательского интерфейса в веб-разработке / Е. Петрова // Журнал UX-дизайнеров – 2022. – Т. 14, № 3. – С. 18–25.

Смирнов Д. Введение в HTML5 и CSS3 / Д. Смирнов // Руководство по Web-технологиям – 2-е изд. – Москва : Издательство XYZ – 2021. – С. 110–125.

Гаврилова М. Лучшие практики JavaScript для одностраничных приложений / М. Гаврилова // Журнал JavaScript – 2023. – Т. 40, № 6. – С. 70–85.

Белов А. jQuery: упрощение разработки на JavaScript / А. Белов // Инсайты jQuery – 2022. – Т. 12, № 4. – С. 30–35.

Козлов П. Интеграция плагина Textolite для редактирования контента / П. Козлов // Форум по интеграции плагинов – 2023. – Т. 7, № 1. – С. 12–17.

Лебедева С. Создание адаптивных веб-сайтов с помощью медиа-запросов CSS / С. Лебедева // Мастера CSS – 2022. – Т. 18, № 2. – С. 55–68.

Соколов Д. Введение в FastAPI для веб-разработки на Python / Д. Соколов // Материалы конференции Python Developers – Москва, 2023. – С. 220– 235.

Документация фреймворка FastAPI // fastapi.com – URL: <https://fastapi.tiangolo.com/learn/> (дата обращения 20.03.2024)

Документация SQLAlchemy // sqlalchemy.org – URL: <https://docs.sqlalchemy.org/en/20/orm/> (дата обращения 15.03.2024)

Документация PyDantic // pydantic.dev – URL: <https://docs.pydantic.dev/latest/> (дата обращения 07.03.2024)

Документация PyJWT // pyjwt.io – URL: <https://pyjwt.readthedocs.io/en/stable/> (дата обращения 07.03.2024)

Курс по Docker // youtube.com – URL: <https://www.youtube.com/watch?v=_uZQtRyF6Eg> (дата обращения 28.03.2024)

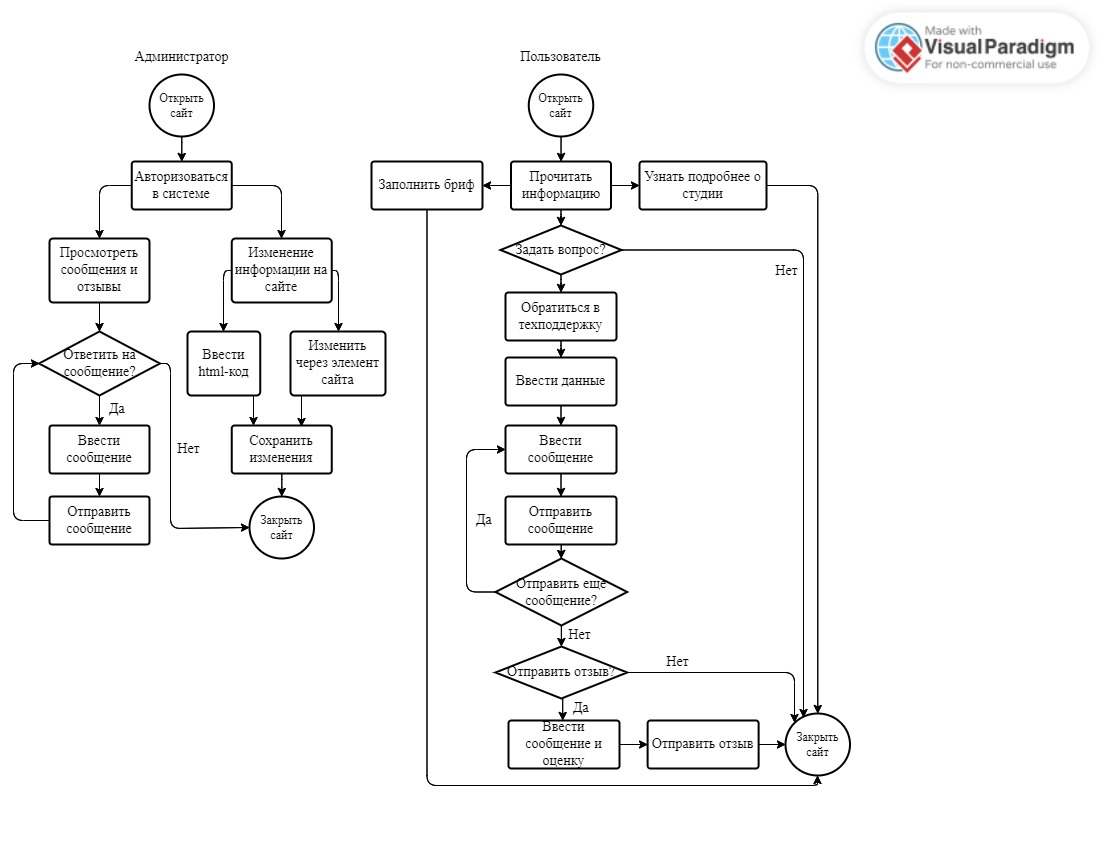
Документация PostgreSQL // postgresql.org – URL: <https://www.postgresql.org/docs/> (дата обращения 20.03.2024)

Статья про WebSocket // habr.com – URL: <https://habr.com/ru/sandbox/171066/> (дата обращения 30.03.2024)

# ПРИЛОЖЕНИЕ А

**(рекомендуемое)**

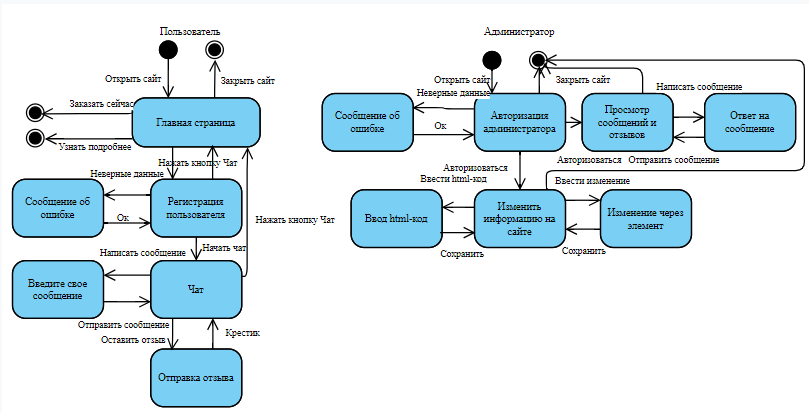
**Отрисовка UserFlow**



# ПРИЛОЖЕНИЕ Б

**(рекомендуемое)**

**Диаграмма окон**



# ПРИЛОЖЕНИЕ В

**(рекомендуемое)**

**Диаграмма последовательности**

