

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ

ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ  
ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ «ДОНСКОЙ ГОСУДАРСТВЕННЫЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ»

(ДГТУ)

Факультет «Информатика и вычислительная техника»

Кафедра «Кибербезопасность информационных систем»

|  |  |  |
| --- | --- | --- |
|  | | |
| Зав. кафедрой | |  |
| \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ | | О.А.Сафарьян |
| (подпись) | |  |
| «\_\_\_» | \_\_\_\_\_\_\_\_\_\_\_\_\_ 2024 г. | |

ОТЧЕТ

по производственной (преддипломной) практике

на предприятии ООО «1С-Корпоративные системы управления»

Обучающаяся \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ А.П. Бойкова

подпись, дата

Обозначение отчета ПП.850000.000 Группа ВИ41

Направление 09.03.01 Информатика и вычислительная техника

Направленность (профиль) Информатика и вычислительная техника

Руководитель практики:

|  |  |
| --- | --- |
| от предприятия | Руководитель |

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Антон Федорович Котов

подпись, дата

от кафедры старший преподаватель \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Николай Валерьевич Гапон

подпись, дата

Оценка \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

дата подпись

Ростов-на-Дону  
2024



МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ

ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ  
ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ «ДОНСКОЙ ГОСУДАРСТВЕННЫЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ»

(ДГТУ)

Факультет «Информатика и вычислительная техника»

Кафедра «Кибербезопасность информационных систем»

ЗАДАНИЕ

по производственной (преддипломной) практике

на предприятии ООО «1С-Корпоративные системы управления»

в период с «22» апреля 2024 г. по «18» мая 2024 г.

Обучающаяся Бойкова Арина Павловна

Обозначение отчета ПП.850000.000 Группа ВИ41

Срок представления отчета на кафедру «18» мая 2024 г.

Содержание индивидуального задания:

Разработка WEB - приложения для информационного сопровождения задач, выполняемых группой сотрудников

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Руководитель от предприятия | | \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  подпись, дата | | А.Ф. Котов |
| Руководитель практики от кафедры | \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  подпись, дата | | Н.В. Гапон | |
| Задание принял к исполнению | \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  подпись, дата | | А.П. Бойкова | |



МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ

ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ  
ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ «ДОНСКОЙ ГОСУДАРСТВЕННЫЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ»

(ДГТУ)

Факультет «Информатика и вычислительная техника»

Кафедра «Кибербезопасность информационных систем»

|  |  |  |
| --- | --- | --- |
| Зав. кафедрой | | «КБИС» |
| \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ | | О.А. Сафарьян |
| (подпись) | |  |
| «\_\_\_» | \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_2024 г. | |

Рабочий график (план) проведения практики

|  |  |  |
| --- | --- | --- |
| № | Мероприятие | Срок выполнения |
| 1 | Прохождение вводного и первичного инструктажа по охране труда на рабочем месте, и инструктажа по пожарной безопасности на объекте. | 22.04.2024 |
| 2 | Аналитический обзор | 24.04.2024 |
| 3 | Алгоритмическое конструирование | 30.05.2024 |
| 4 | Программная реализация | 01.05.2024 |
| 5 | Тестирование приложения | 14.05.2024 |
| 6 | Подготовка итогового отчета | 16.05.2024 |
| 7 | Защита итогового отчета | 18.05.2024 |

Руководитель практики:

|  |  |
| --- | --- |
| от предприятия | Руководитель |

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Антон Федорович Котов

подпись, дата

от кафедры старший преподаватель \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Николай Валерьевич Гапон

подпись, дата

Ростов-на-Дону  
2024

ДНЕВНИК ПРОХОЖДЕНИЯ ПРАКТИКИ

|  |  |  |  |
| --- | --- | --- | --- |
| Дата | Место работы | Выполняемые работы | Оценка руководителя |
| 1 | 2 | 3 | 4 |
| 22.04.2024 | ООО «1С-Корпоративные системы управления» | Знакомство с предприятием, прохождение вводного инструктажа. |  |
| 22.04.2024 | ООО «1С-Корпоративные системы управления» | Ознакомление с территорией предприятия, прохождение первичного инструктажа по ТБ, ПБ. |  |
| 23.04.2024 | ООО «1С-Корпоративные системы управления» | Получение индивидуального задания. |  |
| 24.04.2024-29.04.2024 | ООО «1С-Корпоративные системы управления» | Анализ предметной области. |  |
| 30.04.2024-02.05.2024 | ООО «1С-Корпоративные системы управления» | Анализ существующих систем. |  |
| 03.05.2024-12.05.2024 | ООО «1С-Корпоративные системы управления» | Разработка системы |  |
| 13.05.2024-17.05.2024 | ООО «1С-Корпоративные системы управления» | Оформление отчета практики. |  |
| 18.05.2024 | ООО «1С-Корпоративные системы управления» | Защита отчета по практике на предприятии. |  |
| 18.05.2024 | Кафедра «КБИС» | Защита отчета по практике на кафедре. |  |

|  |  |
| --- | --- |
| от предприятия | Руководитель |

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Антон Федорович Котов

подпись, дата

от кафедры старший преподаватель \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Николай Валерьевич Гапон

подпись, дата

ОТЗЫВ - ХАРАКТЕРИСТИКА

Обучающаяся Бойкова Арина Павловна

4 курса группы ВИ41

кафедры «Кибербезопасность информационных систем»

Вид практики: производственная (преддипломная) практика

Наименование места практики: ООО «1С-Корпоративные системы управления»

Обучающаяся выполнила в полном объёме задания по программе производственной(преддипломной) практики, справилась с разработкой WEB - приложения для информационного сопровождения задач, выполняемых группой сотрудников

Дополнительно ознакомился/изучил

При разработке WEB - приложения для информационного сопровождения задач, выполняемых группой сотрудников, изучила следующие аспекты и технологии: HTML и CSS для создания и стилизации WEB-страниц, JavaScript для добавления интерактивности и PHP для разработки серверной логики.

Заслуживает оценки \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

|  |  |
| --- | --- |
|  | Руководитель практики от предприятия  Антон Федорович Котов  «\_\_\_\_» \_\_\_\_\_\_\_\_\_\_\_\_\_\_2024 г. |

Содержание

Изм.

Лист

№ докум.

Подпись

Дата

Лист

6

ПП.850000.000

Разраб.

А.П. Бойкова

Провер.

Н.В. Гапон

.

Н.контр.

Утв.

Сафарьян О.А.

Разработка WEB-приложения для информационного сопровождения задач, выполняемых группой сотрудников

Лист.

Листов

33

ДГТУ

Кафедра «КБИС»

Введение 7

1 Описание предметной области 9

1.1 Основные понятия 9

1.2 Задачи разработки 10

1.3 Обзор современных программных средств разработки 11

2 Разработка WEB - приложения 12

2.1 Создание базы данных 12

2.2 Создание карты и макета WEB - приложения 14

2.3 Создание шаблона дизайна страницы 17

2.5 Верстка WEB - приложения 19

3 Тестирование WEB - приложения 25

Заключение 28

Перечень использованных информационных ресурсов 29

Приложение А Листинг программы 30

Введение

Преддипломная практика проходила на предприятии ООО «1С-Корпоративные системы управления», которое специализируется на разработке программного обеспечения для управления корпоративными системами. Это предприятие предоставляет широкий спектр решений, направленных на улучшение бизнес-процессов, автоматизацию и оптимизацию управления данными. Во время прохождения практики была поставлена задача разработки WEB - приложения для информационного сопровождения задач, выполняемых группой сотрудников.

Целью данной работы является разработка WEB - приложения для информационного сопровождения задач, выполняемых группой сотрудников, с целью обеспечения более эффективной организации и управления рабочими процессами.

Для достижения этой цели были поставлены следующие задачи:

* изучение основных теоретических аспектов, касающихся информационного сопровождения задач и принципов работы WEB - приложений в данной области;
* исследование основных доступных сервисов и инструментов для организации и управления задачами и проектами в командной среде;
* описание основных программных функций, которые должны быть реализованы в разрабатываемом WEB - приложении для обеспечения информационного сопровождения задач;
* проведение тестирования разработанного программного средства с целью обеспечения его корректной работы и удобства использования.

Объектом исследования являются процессы организации и управления задачами, выполняемыми группой сотрудников. Предметом исследования является WEB - приложение, предназначенное для информационного сопровождения задач и управления рабочими процессами.

Методологическую основу научной исследовательской работы составляют такие научные методы, как анализ, дедукция, сравнение и описание. Эти методы позволяют провести всесторонний анализ требований к WEB-приложению и разработать эффективное программное средство, соответствующее потребностям пользователей.

1 Описание предметной области

* 1. Основные понятия

Предметная область WEB - приложения для информационного сопровождения задач, выполняемых группой сотрудников, охватывает процессы организации и управления рабочими процессами в командной среде. Основной целью такого приложения является обеспечение эффективного управления задачами и проектами, координация деятельности сотрудников, а также обеспечение прозрачности и отчетности в выполнении работ.

Основные понятия в этой предметной области включают в себя задачи и проекты, которые представляют собой конкретные работы, подлежащие выполнению. Задачи могут быть разделены на этапы или подзадачи, а проекты объединяют в себе набор задач, направленных на достижение общей цели.

Сотрудники, участвующие в выполнении задач и проектов, играют ключевую роль. Каждый сотрудник может быть назначен на определенные задачи, иметь определенные роли и обладать определенными правами доступа к информации и функционалу приложения.

Важным аспектом является коммуникация и совместная работа между сотрудниками. Приложение должно обеспечивать возможность обмена сообщениями, комментариями, файлами и другой информацией, необходимой для успешного выполнения задач и проектов.

Отчетность и аналитика также играют важную роль в управлении задачами. Приложение должно предоставлять возможность генерировать отчеты о выполненных задачах, оценивать прогресс проектов, анализировать производительность команды и принимать управленческие решения на основе полученных данных.

Все эти аспекты составляют основу для разработки WEB - приложения, которое будет эффективно поддерживать информационное сопровождение задач и управление рабочими процессами в командной среде.

* 1. Задачи разработки

При разработке WEB - приложения для информационного сопровождения задач, выполняемых группой сотрудников, основной целью является создание инструмента, который обеспечит эффективную организацию и управление рабочими процессами в командной среде. Для достижения этой цели необходимо решить ряд задач.

Во-первых, требуется провести анализ основных теоретических аспектов, связанных с информационным сопровождением задач и принципами работы WEB - приложений в данной области. Это позволит понять особенности процессов управления задачами и определить функциональные требования к разрабатываемому приложению.

Во-вторых, необходимо изучить основные доступные сервисы и инструменты для организации и управления задачами и проектами в командной среде. Это позволит определить существующие решения, их преимущества и недостатки, а также выявить потребности пользователей, которые должны быть удовлетворены в рамках разработки нового приложения.

Далее, следует описать основные программные функции, которые должны быть реализованы в разрабатываемом WEB - приложении. Это включает в себя функционал по созданию и назначению задач, управлению проектами, коммуникации и совместной работе сотрудников, а также генерации отчетов и аналитики по выполненным задачам.

После этого необходимо провести тестирование программного средства для обеспечения его корректной работы и удобства использования. Это позволит выявить и исправить возможные ошибки и недочеты в функционале приложения, а также удостовериться в его соответствии требованиям пользователей и задачам бизнеса.

1.3 Обзор современных программных средств разработки

Обзор современных программных средств разработки и языков программирования для создания WEB - приложений, предназначенных для информационной поддержки задач, выполняемых группой сотрудников, является важным этапом при выборе технологического стека для проекта. В современном мире существует множество инструментов и языков программирования, которые способны обеспечить функциональность, производительность и удобство использования WEB - приложения.

При выборе программных средств разработки необходимо учитывать такие факторы, как масштабируемость проекта, требования к производительности, уровень сложности интерфейса, а также опыт команды разработчиков. Среди популярных языков программирования для WEB-разработки можно выделить JavaScript, Python, Ruby, PHP и Java. Каждый из этих языков имеет свои особенности и преимущества, которые следует учитывать при выборе.

В сфере программных средств разработки для WEB - приложений также существует множество инструментов, облегчающих процесс создания, отладки и развертывания приложений. К ним относятся фреймворки, библиотеки, среды разработки и инструменты автоматизации. Некоторые из наиболее популярных фреймворков включают в себя React, Angular, Vue.js, Django, Flask, Ruby on Rails, Laravel и Spring.

Выбор конкретных программных средств и языков программирования зависит от требований проекта, предпочтений разработчиков и существующих технологических решений в компании. При правильном подборе технологического стека можно обеспечить эффективную разработку, поддержку и масштабирование WEB - приложения для информационного сопровождения задач, выполняемых группой сотрудников.

2 Разработка WEB - приложения

* 1. Создание базы данных

Для создания WEB - приложения, база данных играет важную роль, поскольку она служит для хранения всех данных, необходимых для функционирования приложения и обеспечения пользовательского опыта. Вот несколько причин, почему база данных необходима для такого приложения:

* хранение пользовательских данных. База данных позволяет хранить информацию о пользователях, включая их учетные записи, личную информацию и предпочтения;
* хранение данных досок. Для каждой созданной доски необходимо хранить информацию о её содержимом, такую как заметки, изображения, видео и комментарии. База данных позволяет структурированно хранить эту информацию и обеспечивать доступ к ней;
* управление правами доступа. База данных позволяет определять права доступа пользователей к различным функциям приложения и контенту. Например, определение администраторов досок или управление приватностью контента;
* отслеживание активности пользователей. База данных может использоваться для отслеживания активности пользователей, такой как создание, редактирование или удаление заметок, комментариев и досок, что позволяет улучшить аналитику и персонализировать пользовательский опыт;
* обеспечение целостности данных. Использование базы данных позволяет обеспечить целостность данных путем предотвращения потери или искажения информации.

Для создания базы данных для такого приложения необходимо определить её структуру и используемые таблицы, а также установить отношения между ними. Даталогическая модель для информационного сопровождения задачи выполняемой группой работников представлена на рисунке 1.

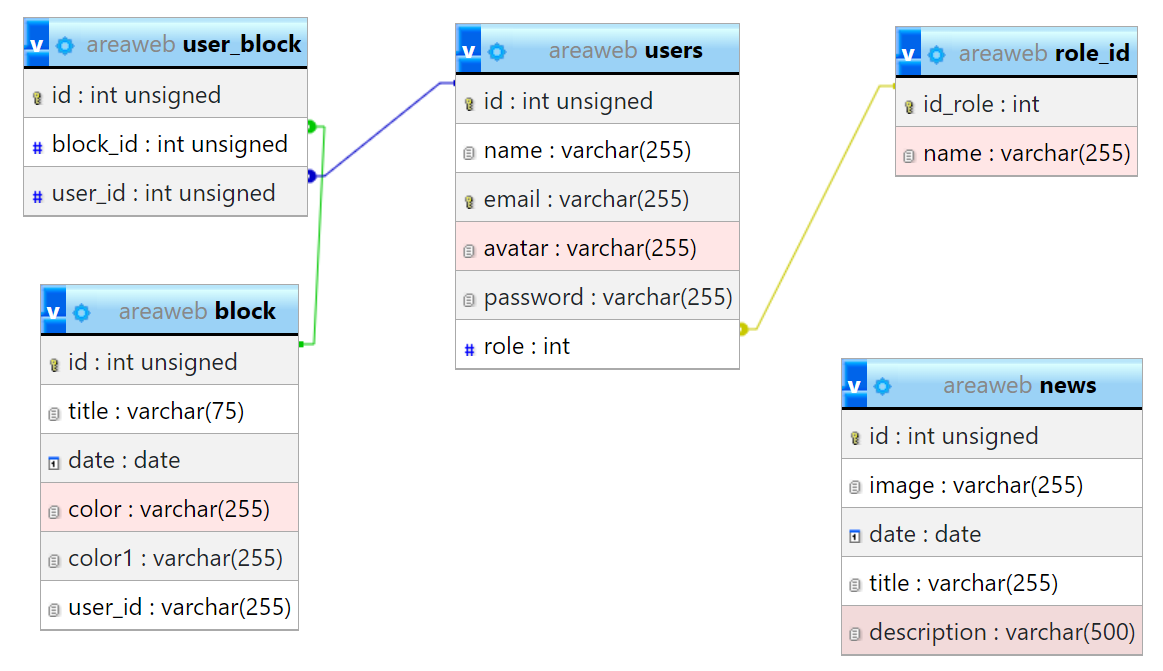


Рисунок 1 – Даталогическая модель

Рассмотрим каждую таблицу:

Таблица 1 - Пользователи

|  |  |  |
| --- | --- | --- |
| Столбец | Тип данных | Описание |
| id | int | Уникальный идентификатор пользователя |
| name | varchar (255) | Уникальный логин пользователя |
| email | varchar (255) | Уникальный email пользователя |
| avatar | varchar (255) | Фотография пользователя |
| password | varchar (255) | Хэшированный пароль пользователя |
| role | int | Роль пользователя |

Таблица 2 - Роль пользователя

|  |  |  |
| --- | --- | --- |
| Столбец | Тип данных | Описание |
| id\_role | int | Уникальный идентификатор роли |
| name | varchar (255) | Наименование роли (админ, пользователь) |

В таблице блоки содержится информация о созданных блоках доски, название, цвет блока, цвет текста, дата создания и распределение доступности для определенного пользователя, каждый из блоков будет отображаться на странице у администратора, руководителя и пользователя.

Таблица 3 - Блоки

|  |  |  |
| --- | --- | --- |
| Столбец | Тип данных | Описание |
| id | int | Уникальный идентификатор блока |
| title | varchar (255) | Тема блока |
| date | date | Дата создания блока |
| color | varchar (255) | Цвет блока |
| color1 | varchar (255) | Цвет текста блока |
| user\_id | int | Доступность блока пользователям |

В таблице пользовательские блоки, будет информация о том, какой блок с темой доступен определенному пользователю.

Таблица 4 - Блоки для пользователя

|  |  |  |
| --- | --- | --- |
| Столбец | Тип данных | Описание |
| id | int | Уникальный идентификатор |
| block\_id | int | Уникальный идентификатор блока |
| user\_id | int | Уникальный идентификатор пользователя |

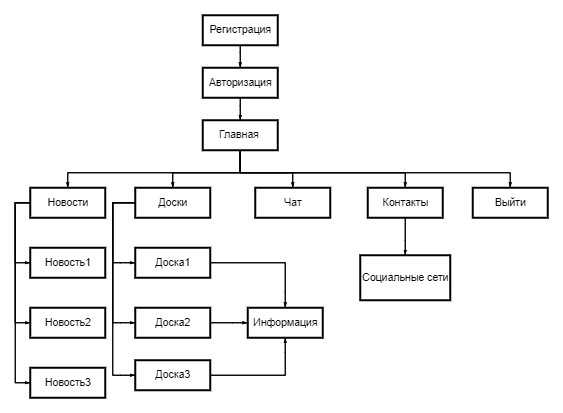
В таблице новости будет представлена информация о созданных новостях.

Таблица 5 - Новости

|  |  |  |
| --- | --- | --- |
| Столбец | Тип данных | Описание |
| id | int | Уникальный идентификатор новости |
| image | varchar (255) | Картинка новости |
| date | date | Дата создания новости |
| title | varchar (255) | Тема новости |
| description | varchar (255) | Описание новости |

* 1. Создание карты и макета WEB - приложения

Карта сайта содержит список всех доступных на WEB-сайте страниц или разделов, обычно представленных в иерархическом порядке. Карта сайта облегчает поиск информации по сайту для поисковых роботов и пользователей. Рассмотрим карту сайта на рисунке 2.

Рисунок 2 – Карта сайта

Основные разделы включают:

* главная страница: ссылки на основные разделы сайта, информационные блоки;
* блоки новостей, которые включают в себя новостную информацию;
* страница чата, содержит список пользователей и взаимодействие между ними;
* страница доски, которая включает в себя информацию о созданном блоке, содержит фотографии, документы, комментарии и реакции пользователей.

Каждый раздел сайта логически связан с другими, обеспечивая удобную навигацию и удобный интерфейс для пользователей. Рассмотрим макет страницы приложения на рисунке 3.

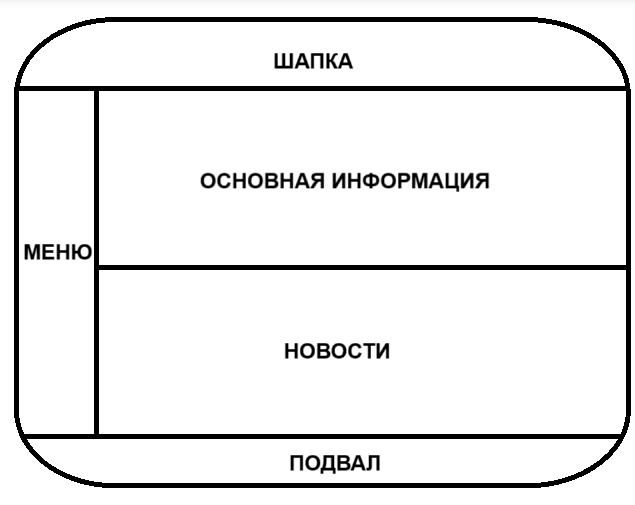


Рисунок 3 – Макет страницы

На данном макете представлена структура WEB - приложения, которая состоит из следующих разделов:

* шапка - верхняя часть страницы, обычно содержащая логотип, название сайта и основное меню навигации. Может также включать контактные данные, кнопки для входа в систему, поиск и другие элементы;
* навигационное меню - содержит ссылки на основные разделы сайта или страницы, облегчая пользователям навигацию по сайту. Обычно размещается в шапке страницы, но также может быть расположено в боковой панели или в подвале;
* основное содержимое - основная область страницы, содержащая основной контент, такой как текст, изображения, здесь размещается информация, которая предназначена для привлечения внимания пользователя и передачи основного сообщения страницы;
* подвал - нижняя часть страницы, содержащая дополнительные ссылки, контактную информацию и другие вспомогательные элементы. Обычно включает ссылки на разделы сайта, контактные данные, ссылки на социальные сети и другие;

Макет страницы определяет расположение и структуру контента на WEB-сайте. Он позволяет разработчикам и дизайнерам организовать информацию на странице таким образом, чтобы пользователи могли легко ориентироваться и быстро находить нужную им информацию.

* 1. Создание шаблона дизайна страницы

Шаблон страницы играет ключевую роль в определении общего стиля и внешнего вида веб-сайта. Основные элементы дизайна включают цветовую палитру, читаемые шрифты, четкую структуру страниц, и использование высококачественных изображений.

В данном случае, цветовая палитра состоит из белых, синих и голубых оттенков, с более яркими цветами для выделения важной информации. Читаемые и современные шрифты облегчают восприятие информации, а четкая структура страницы сделает навигацию пользователя интуитивно понятной. Использование качественных изображений, отражающих деятельность организации или проектов, поможет создать привлекательное визуальное впечатление.

Для создания привлекательного, функционального и удобного для пользователей дизайна страницы также важно учитывать адаптивность под различные устройства и экраны. Это обеспечит оптимальный пользовательский опыт независимо от устройства, с которого пользователь заходит на сайт.

Кроме того, важно провести тестирование дизайна с целью выявления и

исправления возможных проблем или несоответствий, чтобы обеспечить высокий уровень удовлетворенности пользователей. Пример шаблона дизайна главной страницы сайта можно увидеть на рисунке 4.

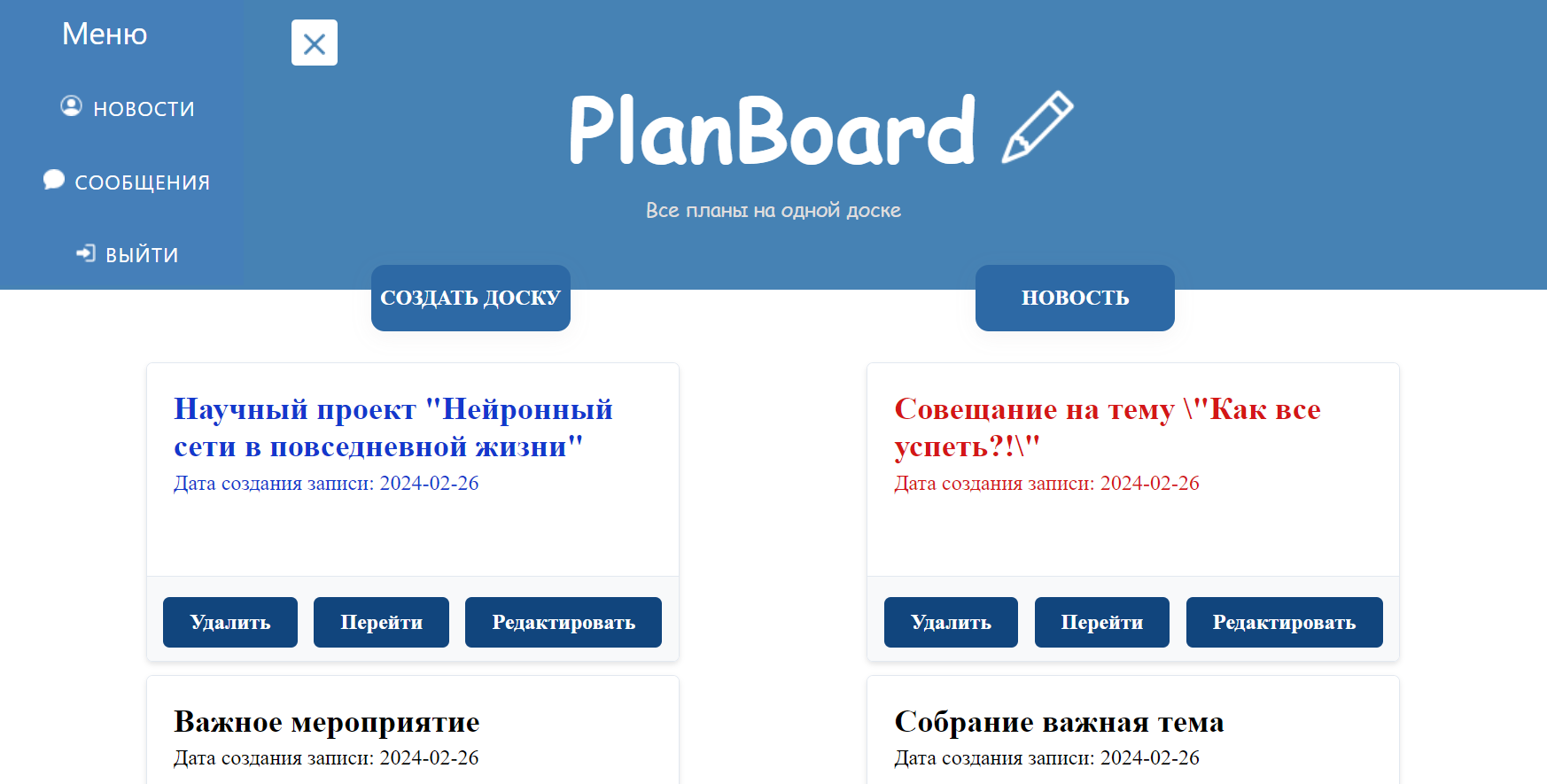


Рисунок 4 – Пример дизайна страницы

* 1. Формирование контента

Новостная страница:

На новостной странице сайта обычно должна быть представлена актуальная информация о событиях, новостях, анонсах и других важных событиях, которые могут быть интересны целевой аудитории.

Страница регистрации:

Это страница, на которой пользователи могут создать учетную запись для доступа к WEB-приложению или сервису. На странице представлены поля для ввода имени пользователя, адреса электронной почты, изображение профиля и пароля. Пользователи должны заполнить все обязательные поля и нажать кнопку "Зарегистрироваться", чтобы завершить процесс регистрации. Также на странице может быть ссылка для перехода на страницу авторизации, если пользователь уже имеет учетную запись.

Страница авторизации:

Это страница, на которой пользователи могут войти в свою учетную запись, используя ранее созданные учетные данные. На странице присутствуют поля для ввода имени пользователя (или адреса электронной почты) и пароля. Пользователи должны правильно заполнить эти поля и нажать кнопку "Войти", чтобы получить доступ к своему аккаунту.

Главная страница:

Это основная страница WEB - приложения, на которой пользователи оказываются после успешной авторизации. На главной странице представлено навигационное меню, чат пользователей, основная информация из созданных досок и новостная лента, также представлен подвал, в котором располагается контактная информация и ссылки на сторонние сайты.

Страница информации по каждой доске:

На данной странице представлена информация для каждого блока доски, документы, файлы, изображения, которые создаются администратором или руководителем WEB – приложения. Данные на этой странице пользователи не только видят, но также комментируют и отвечают реакциями.

Страница чата:

На данной странице пользователи могут взаимодействовать друг с другом и так же с руководителем.

Страница администратора:

На данной странице администратор видит всю информацию, но также может добавлять новую, добавлять новостные блоки, удалять элементы и редактировать их.

Страница руководителя:

На данной странице руководитель может создавать новые доски и приглашать туда пользователей.

* 1. Верстка WEB - приложения

Для разработки сайта использовался локальный сервер Open Server, обеспечивающий среду для работы с PHP и базой данных MySQL.

Содержание папки «plan-board» с проектом включает следующие файлы:

* register.php: этот файл представляет собой HTML-страницу для регистрации пользователя на веб-сайте. В нем используется PHP для проверки статуса пользователя и отображения сообщений об ошибках валидации. Форма регистрации содержит поля для ввода имени, адреса электронной почты, пароля и изображения профиля. Пользователь должен заполнить обязательные поля и отправить форму, чтобы зарегистрироваться на сайте. Если введенные данные не проходят валидацию, пользователю отображаются сообщения об ошибках. В конце страницы есть ссылка для перехода на страницу авторизации, если у пользователя уже есть аккаунт;
* index.php: этот файл представляет собой HTML-страницу для авторизации пользователей на веб-сайте "PlanBoard". Она содержит форму входа, позволяющую пользователям ввести свой email и пароль для входа в

свою учетную запись. В верхней части страницы используется PHP для проверки статуса пользователя: если пользователь уже авторизован, он будет перенаправлен на другую страницу;

* home.php: этот файл является главной страницей веб-приложения "PlanBoard" для администратора. В файле присутствуют формы для добавления новых досок и новостей, а также списки уже созданных досок и новостей. Страница также содержит шапку с логотипом и навигационным меню, а также подвал с ссылками на социальные сети и информацией об авторском праве. Для динамического обновления контента используются технологии JavaScript и библиотека Vue.js;
* homeruk.php: этот файл является главной страницей веб-приложения "PlanBoard" для руководителя. В файле также присутствуют формы для добавления новых досок;
* homeuser.php: этот файл является главной страницей веб-приложения "PlanBoard". Он предоставляет пользователю интерфейс для просмотра новостей и создания досок с информацией;
* helpers.php: этот файл содержит набор функций и настроек для работы с сессиями, базой данных и другими аспектами веб-приложения. Он предоставляет удобные методы для реализации функций, таких как перенаправление пользователя, установка и получение сообщений, обработка ошибок валидации, загрузка файлов на сервер и другие;

Рассмотрим реализацию WEB – приложения и его взаимодействие с базой данных. На рисунке 5 представлена форма регистрации. В форме регистрации пользователь вводит свои уникальные данные. Для отправки в базу данных, они все должны быть заполнены. Когда пользователь нажимает на кнопку продолжить, происходит проверка на заполнение данных и запрос отправляется на сервер, в базе данных появляется информация о пользователе, если запрос был корректно сформирован.

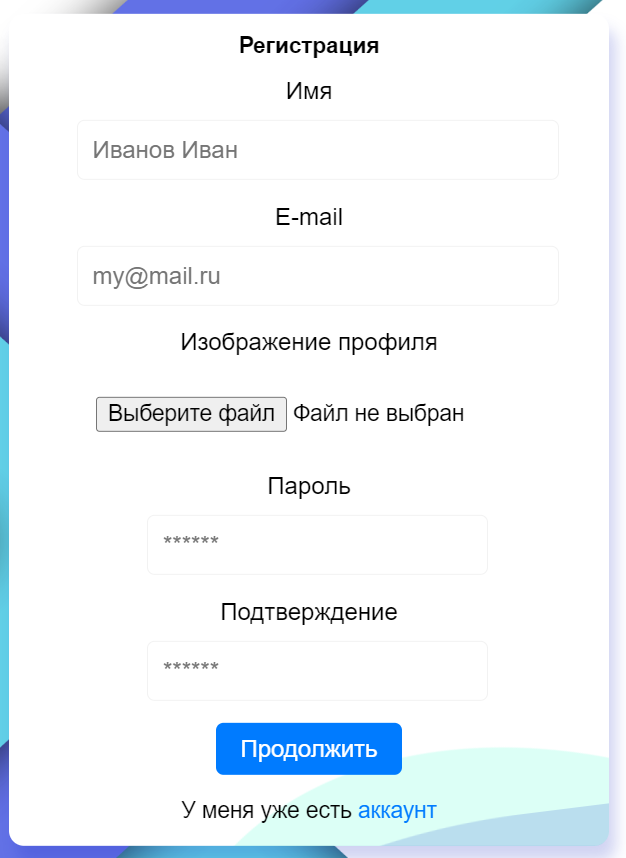


Рисунок 5 – Форма регистрации

Теперь пользователь может пройти авторизацию, представлена на рисунке 6.

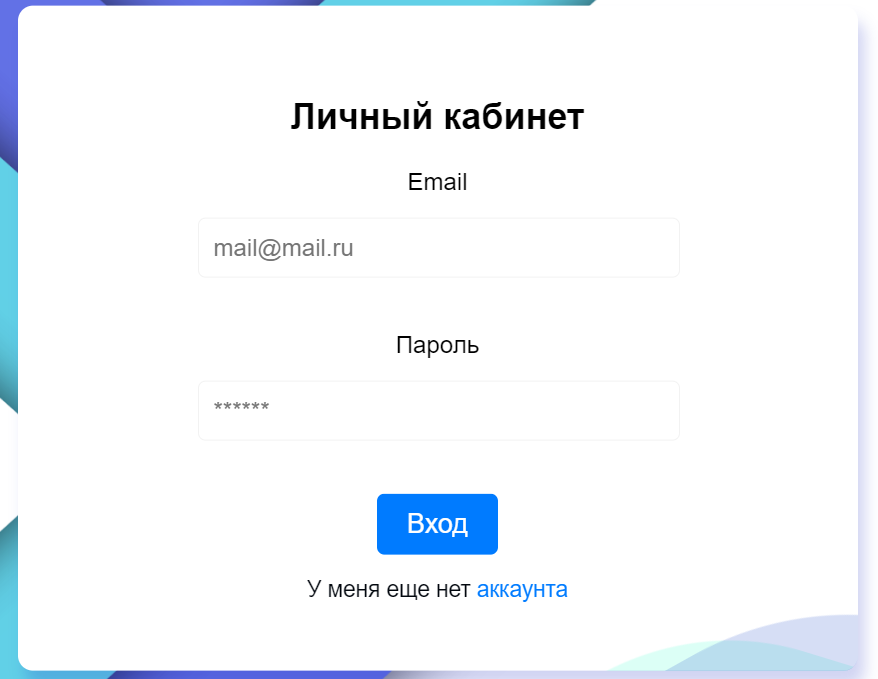


Рисунок 6 – Форма авторизации

В форму авторизации пользователь, администратор или руководитель вводят данные, которые были указаны при регистрации. Они отправляются на сервер, где происходит проверка данных, если данных не будет в базе данных, то вернется ошибка. Во время отправки данных устанавливается идентификатор пользователя, по сессии происходит проверка роли пользователя и открывается страница в зависимости от роли. Рассмотрим WEB – приложение со стороны работы администратора на рисунке 7.

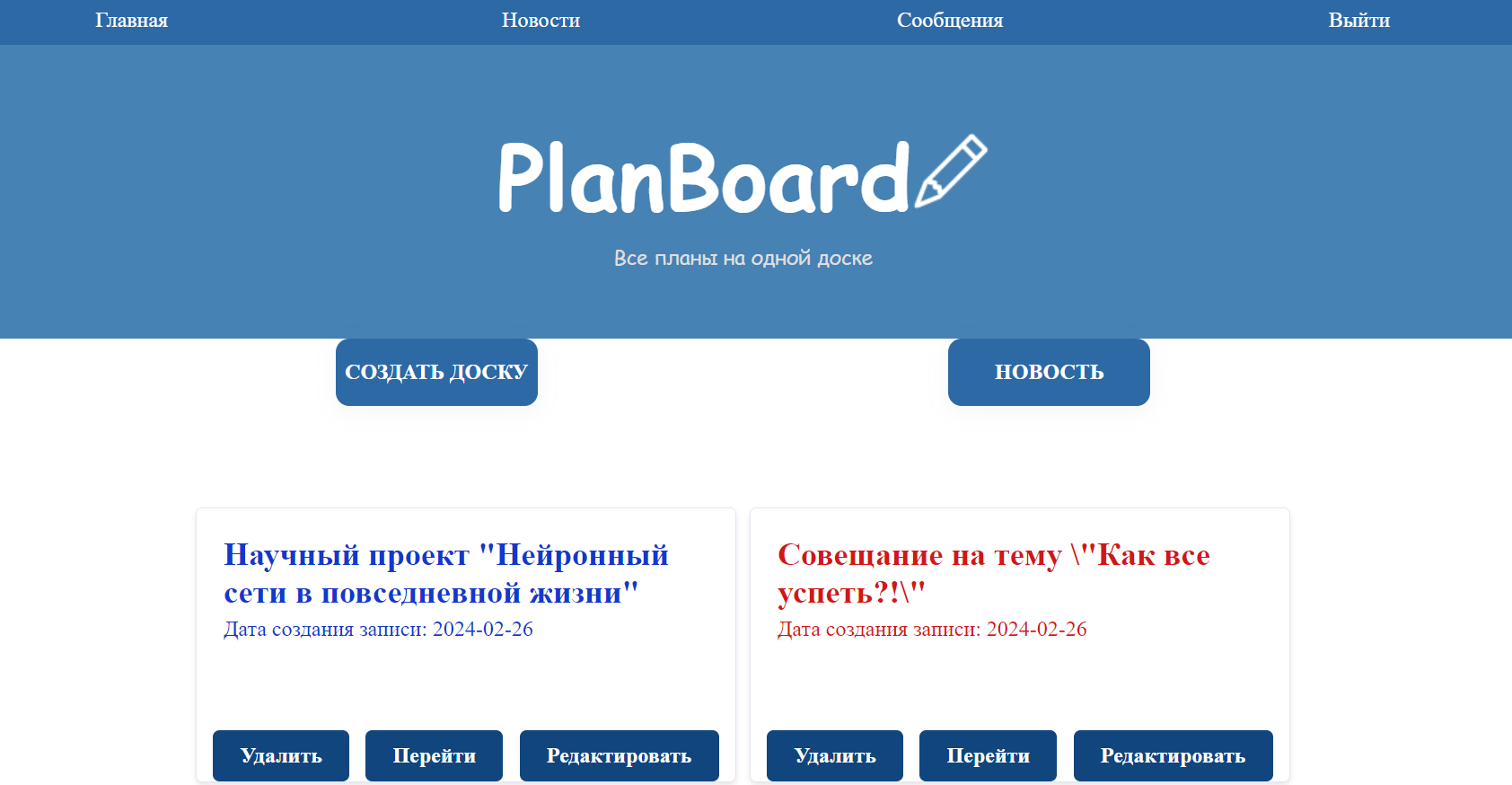


Рисунок 7 – Страница администратора

На этой странице администратор с помощью динамических форм «Создать доску» и «Новость» может добавить данные на страницу приложения, работа данных форм представлена на рисунке 8.

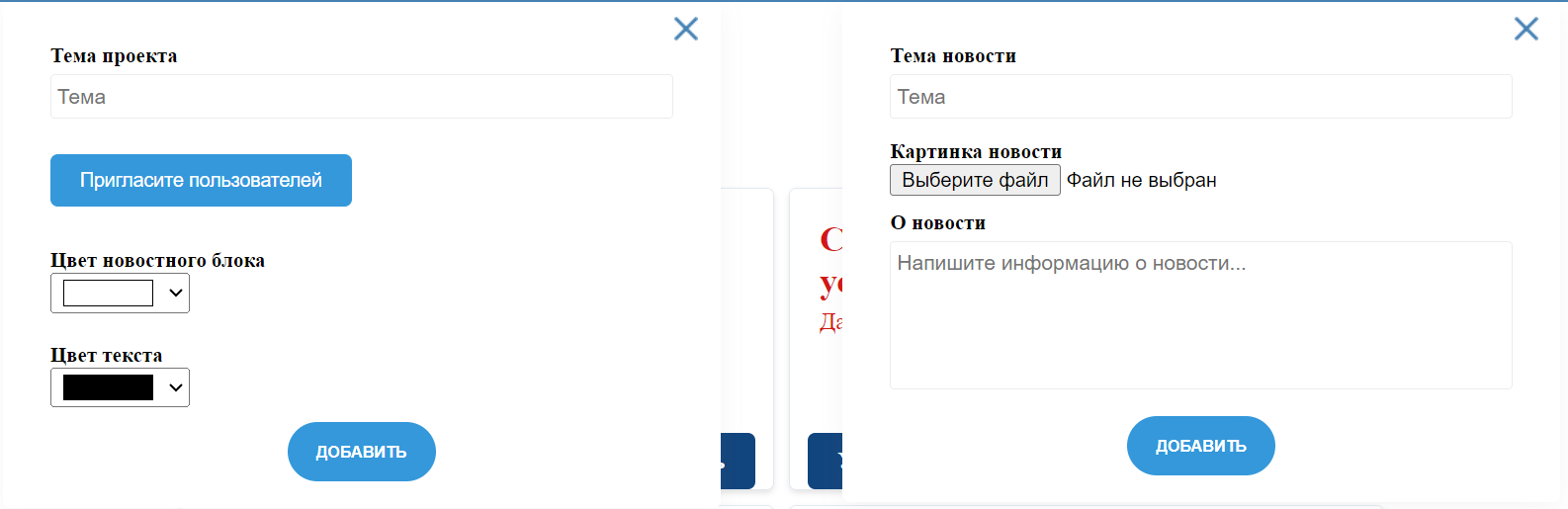


Рисунок 8 – Динамические формы отправки

Администратор заполняет форму данными и отправляет их на сервер, если данные указаны корректно они добавляются в базу данных и затем отображаются на странице,отображение блоков на рисунке 9.

Администратор может удалять и редактировать данные в блоке, выбирается блок на удаление и происходит запрос по идентификатору блока, если он корректный, блок удаляется. Так же формируется запрос на редактирование, данные отправляются на сервер и перезаписываются в базе данных.

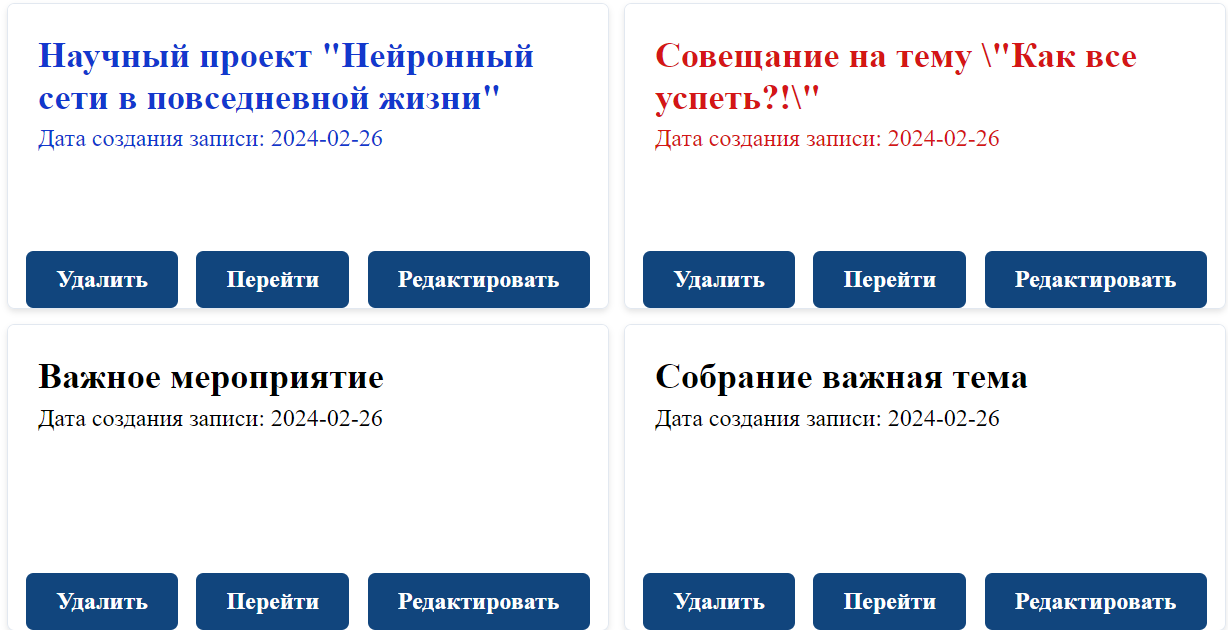


Рисунок 9 – Блоки досок

Если рассматривать работу приложения со стороны руководителя,то руководитель может добавлять только новостные блоки.

Данные заполняются в форме добавления новостных блоков и отправляются на сервер.Корректные данные добавляются в базу данных и отображаются на странице, рисунок 10.

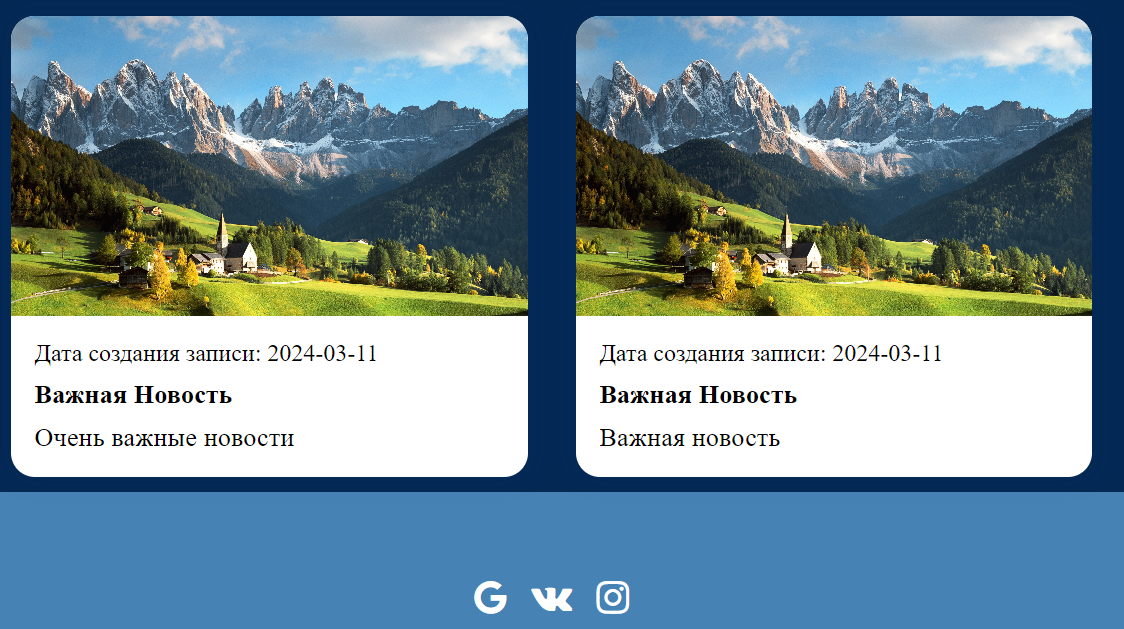


Рисунок 10 – Новостные блоки

Работа WEB – приложения со стороны пользователя представлена на рисунке 11.

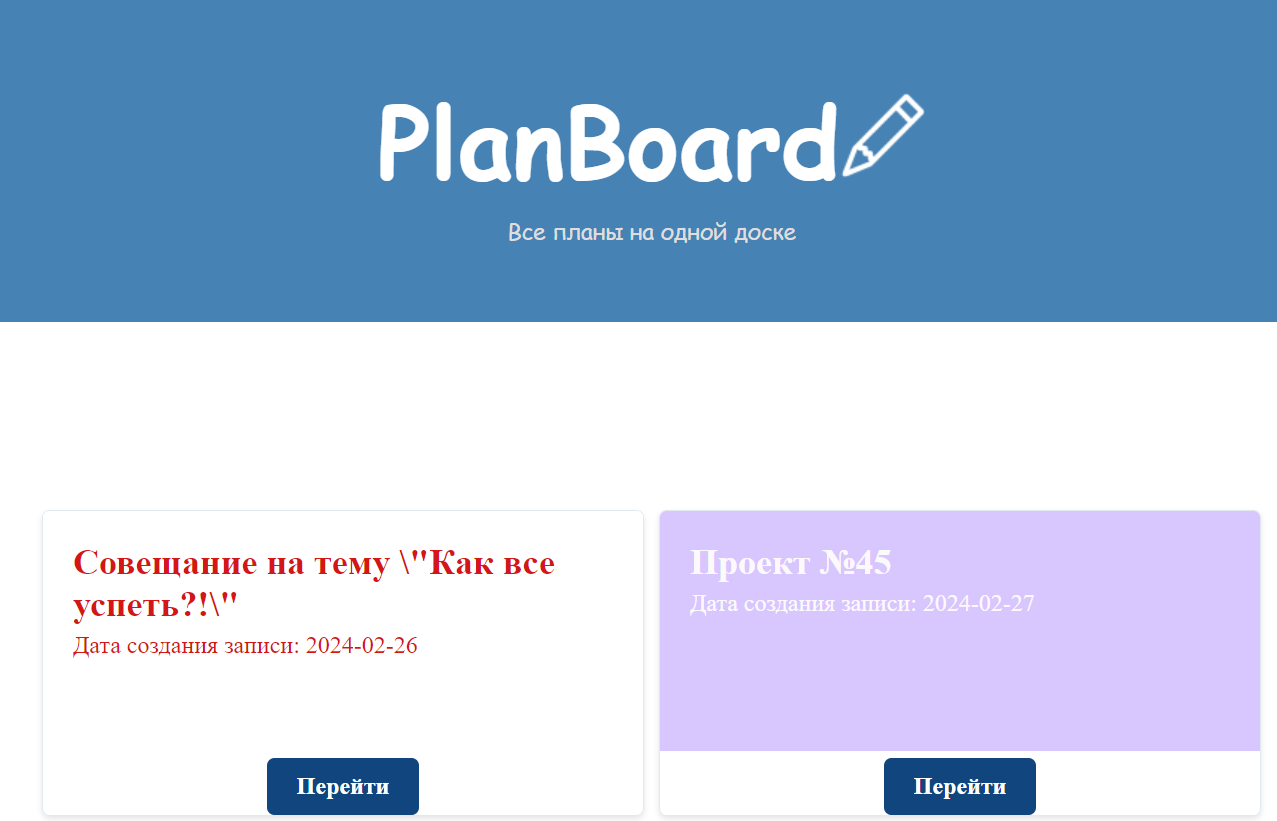


Рисунок 11 – Страница пользователя

Пользователь может просматривать только те блоки, которые ему доступны.

Все описание работы серверной части можно более детально рассмотреть с помощью алгоритма на рисунке 12.

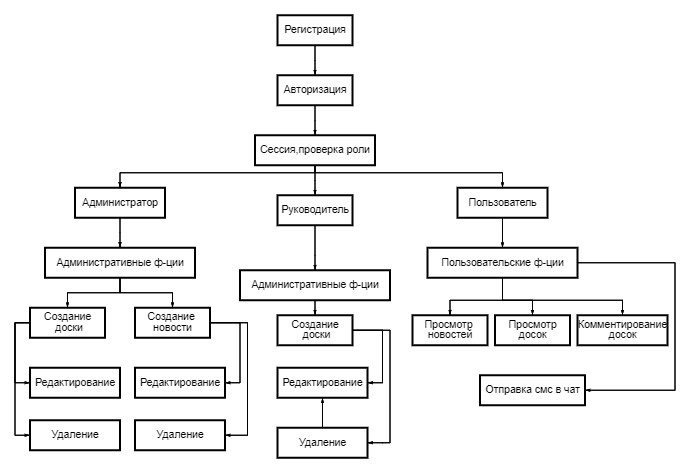


Рисунок 12 – Алгоритм работы серверной части

3 Тестирование WEB - приложения

Для обеспечения работоспособности сайта необходимо пройти через целый ряд тестовых сценариев, чтобы убедиться, что все функции работают правильно.

Регистрация нового пользователя:

* проверка ввода некорректных данных в форму регистрации и обработка ошибок;
* проверка ввода корректных данных и успешное добавление пользователя в базу данных.

Авторизация пользователя:

* проверка ввода учетных данных незарегистрированного пользователя и обработка ошибок;
* проверка ввода учетных данных зарегистрированного пользователя и успешный вход на сайт;
* проверка корректного перенаправления пользователя после успешной авторизации и доступа к функциям только авторизованным пользователям.

Создание нового блока:

* проверка ввода некорректных данных и обработка ошибок;
* проверка успешного добавления нового блока с корректными данными.

Редактирование блока:

* проверка возможности редактирования текста блока;
* проверка обработки ошибок при сохранении изменений;
* проверка уведомления пользователя об ошибке и сохранении данных блока без изменений.

Удаление блока:

* проверка подтверждения удаления блока;
* проверка обработки ошибок при удалении;
* проверка уведомления пользователя об ошибке и сохранении блока на доске.

Создание новости:

* проверка ввода некорректных данных и обработка ошибок;
* проверка успешного добавления новости с корректными данными.

Редактирование новости:

* проверка возможности редактирования текста или изображения новости;
* проверка обработки ошибок при сохранении изменений;
* проверка уведомления пользователя об ошибке и сохранении данных новости без изменений.

Удаление новости:

* проверка подтверждения удаления новости;
* проверка обработки ошибок при удалении;
* проверка уведомления пользователя об ошибке и сохранении новости на доске.

Отображение блоков:

* проверить, что все блоки корректно отображаются на доске и сохраняют свои положения и данные после перезагрузки страницы.

Обработка ошибок и отображение сообщений:

* ввести некорректные данные или выполните действие, которое может привести к ошибке;
* убедиться, что приложение корректно обрабатывает ошибки, выводит информативные сообщения об ошибках и предлагает решения или подсказки для пользователя.

Провести тестирование производительности, чтобы убедиться, что приложение работает стабильно и быстро при добавлении большого количества блоков или при многопользовательском использовании.

Нефункциональное тестирование:

* производительность: оценить скорость загрузки страниц и выполнения запросов к базе данных. Убедиться, что сайт работает быстро и отзывчиво;
* безопасность: проверить наличие уязвимостей безопасности. Обеспечить защиту данных пользователей и аутентификацию;
* совместимость: проверить работоспособность сайта на различных браузерах (Chrome, Firefox, Safari, Edge) и устройствах (компьютеры, планшеты, мобильные телефоны);
* отзывчивый дизайн: убедиться, что сайт корректно отображается и работает на разных разрешениях экрана и устройствах;
* проверка корректного отображения сайта и его функций на различных мобильных устройствах с разными размерами экранов;
* убедиться, что элементы управления и контент легко доступны и пригодны для использования на мобильных устройствах;
* проверка корректности управления сеансами пользователей, чтобы убедиться, что авторизованные пользователи не могут получить доступ к защищенным страницам или функциям без входа в систему;
* проверка производительности базы данных при обработке запросов на чтение и запись;
* оценка времени ответа базы данных при выполнении различных операций, таких как создание, обновление и удаление записей.

После проведения тестирования необходимо внимательно проанализировать результаты, исправить обнаруженные ошибки и улучшить функциональность и удобство использования виртуальной доски.

Заключение

В процессе работы над приложением для информационного сопровождения задач группы работников прошли несколько этапов.

Сначала было выполнено планирование и проектирование, включающее определение функциональных и нефункциональных требований к приложению, создание его архитектуры и выбор используемых технологий.

Затем была разработана основная функциональность приложения, такая как добавление, редактирование и удаление блоков информации, а также функции работы с пользователями и загрузки изображений.

После этого проводилось тестирование и отладка, нацеленные на обнаружение и исправление ошибок, а также на улучшение производительности и безопасности приложения.

Наконец, приложение дорабатывалось и оптимизировалось, включая улучшение интерфейса, добавление новых функций и оптимизацию кода и базы данных для повышения производительности и улучшения пользовательского опыта.

Результатом работы стало приложение для информационного сопровождения задач группы работников, позволяющее удобно работать с блоками информации на виртуальной доске. Для его создания использовались современные технологии веб-разработки, такие как Vue.js, PHP и AJAX, обеспечивающие динамический интерфейс, серверное взаимодействие и обмен данными без перезагрузки страницы. Приложение предоставляет базовый функционал и может быть дополнено и расширено в будущем с учетом потребностей пользователей.

Перечень использованных информационных ресурсов

1. Официальная документация языка разметки HTML [Электронный ресурс], URL: https://www.w3.org/2019/04/WHATWG-W3C-MOU.html. (дата обращения: 01.05.2024 г.).
2. Официальная документация каскадных таблиц стилей [Электронный ресурс], URL: https://www.w3.org/Style/CSS/Overview.en.html. (дата обращения: 01.05.2024 г.).
3. Официальная документация языка программирования JS [Электронный ресурс], URL: https://www.javascript.com. (дата обращения: 05.06.2023 г.).
4. Официальная документация языка программирования РНР [Электронный ресурс], URL: <https://www.php.net/manual/ru/index.php> (дата обращения: 26.04.2024 г.).
5. Официальная документация MySQL [Электронный ресурс], URL: <https://www.opennet.ru/docs/RUS/mysqlrus/> (дата обращения: 05.05.2024 г.).

Приложение А Листинг программы

<?php

// Включаем файл с функциями и проверкой авторизации

require\_once \_\_DIR\_\_ . '/src/helpers.php';

// Проверяем авторизацию пользователя

checkAuth();

// Получаем текущего пользователя

$user = currentUser();

// Подключение к базе данных

require\_once \_\_DIR\_\_ . '/src/config.php';

// Создание соединения

$conn = new mysqli(DB\_HOST, DB\_USERNAME, DB\_PASSWORD, DB\_NAME);

// Проверка соединения

if ($conn->connect\_error) {

die("Ошибка подключения: " . $conn->connect\_error);

}

?>

<!doctype html>

<html lang="en" data-bs-theme="auto">

<head><script src="../assets/js/color-modes.js"></script>

<meta charset="utf-8">

<meta name="viewport" content="width=device-width, initial-scale=1">

<meta name="description" content="">

<meta name="author" content="Mark Otto, Jacob Thornton, and Bootstrap contributors">

<meta name="generator" content="Hugo 0.122.0">

<title>Страница администратора</title>

<link rel="canonical" href="https://getbootstrap.com/docs/5.3/examples/dashboard/">

<link rel="stylesheet" href="assets/dashboard.css">

<link rel="stylesheet" href="https://cdn.jsdelivr.net/npm/@docsearch/css@3">

<link href="../assets/dist/css/bootstrap.min.css" rel="stylesheet">

<!-- Custom styles for this template -->

<link href="https://cdn.jsdelivr.net/npm/bootstrap-icons@1.10.5/font/bootstrap-icons.min.css" rel="stylesheet">

<!-- Custom styles for this template -->

<link href="dashboard.css" rel="stylesheet">

</head>

<body>

<div class="loader show">

<div class="loader-inner">

<div class="loader-line-wrap">

<div class="loader-line"></div>

</div>

<div class="loader-line-wrap">

<div class="loader-line"></div>

</div>

<div class="loader-line-wrap">

<div class="loader-line"></div>

</div>

<div class="loader-line-wrap">

<div class="loader-line"></div>

</div>

<div class="loader-line-wrap">

<div class="loader-line"></div>

</div>

</div>

</div>

<header class="navbar sticky-top bg-dark flex-md-nowrap p-0 shadow" data-bs-theme="dark">

<a class="navbar-brand col-md-3 col-lg-2 me-0 px-3 fs-25 text-white" href="#">PlanBoard</a>

<ul class="navbar-nav flex-row d-md-none">

<li class="nav-item text-nowrap">

<button class="nav-link px-3 text-white" type="button" data-bs-toggle="offcanvas" data-bs-target="#sidebarMenu" aria-controls="sidebarMenu" aria-expanded="false" aria-label="Toggle navigation">

<svg class="bi"><use xlink:href="#list"/></svg>

</button>

</li>

</ul>

</header>

<div class="container-fluid">

<div class="row">

<div class="sidebar border border-right col-md-3 col-lg-2 p-0 bg-body-tertiary">

<div class="offcanvas-md offcanvas-end bg-body-tertiary" tabindex="-1" id="sidebarMenu" aria-labelledby="sidebarMenuLabel">

<div class="offcanvas-header">

<h5 class="offcanvas-title" id="sidebarMenuLabel">PlanBoard</h5>

<button type="button" id="sidebarMenuCloseButton" class="btn-close" data-bs-dismiss="offcanvas" data-bs-target="#sidebarMenu" aria-label="Close"></button>

</div>

<div class="offcanvas-body d-md-flex flex-column p-0 pt-lg-3 overflow-y-auto">

<ul class="nav flex-column">

<li class="nav-item">

<a class="nav-link d-flex align-items-center gap-2" href="#" onclick="showSection('newsSection', 'Добавление новости')">

<i class="bi bi-file-earmark-arrow-up"></i>

Добавить новость

</a>

</li>

<li class="nav-item">

<a class="nav-link d-flex align-items-center gap-2" href="#" onclick="showSection('boardSection', 'Добавление блока доски')">

<i class="bi bi-file-earmark-arrow-up"></i>

Добавить блок доски

</a>

</li>

<li class="nav-item">

<a class="nav-link d-flex align-items-center gap-2" href="#" onclick="showSection('contentSection', 'Добавление контента')">

<i class="bi bi-file-earmark-arrow-up"></i>

Добавить контент

</a>

</li>

</ul>

<ul class="nav flex-column mb-auto">

<li class="nav-item">

<form id="logout-form" action="src/actions/logout.php" method="post" style="display: none;">

<button type="submit" id="logout-button"></button>

</form>

<a href="#" class="nav-link d-flex align-items-center gap-2" onclick="document.getElementById('logout-button').click(); return false;">

<svg class="bi"><use xlink:href="#door-closed"/></svg>