МИНИСТЕРСТВО ОБРАЗОВАНИЯ РЕСПУБЛИКИ БЕЛАРУСЬ

УЧРЕЖДЕНИЕ ОБРАЗОВАНИЯ

«ГОМЕЛЬСКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ ИМЕНИ П.О. СУХОГО»

Факультет автоматизированных и информационных систем

Кафедра «Информатика»

Специальность 1-40 04 01 «Информатика и технологии программирования»

Отчет по преддипломной практике

на тему: «Программный комплекс для автоматизации   
менеджмента фитнес-центра»

Подготовил: студент гр. ИП-42

Чечко А.Д.

Руководитель от предприятия:

зам. начальника отдела рекламы и информации

Краснов А.А.

Руководитель:

к.т.н., доцент

Прокопенко Д.В.

Дата проверки:

Дата допуска к защите:

Дата защиты:

Оценка работы:

Подписи членов комиссии: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Гомель 2023

СОДЕРЖАНИЕ

[Введение 4](file:///C:\Users\User\Desktop\отчёт_преддипломная_Земчёнок12345678_edit.docx#_Toc69773318)

[1 Анализ предметной области 5](file:///C:\Users\User\Desktop\отчёт_преддипломная_Земчёнок12345678_edit.docx#_Toc69773319)

[1.1 Анализ предметной области 5](file:///C:\Users\User\Desktop\отчёт_преддипломная_Земчёнок12345678_edit.docx#_Toc69773320)

[1.2 Аналитический обзор существующих аналогов 5](file:///C:\Users\User\Desktop\отчёт_преддипломная_Земчёнок12345678_edit.docx#_Toc69773321)

[1.3 Обзор технологий для реализации программного обеспечения 8](file:///C:\Users\User\Desktop\отчёт_преддипломная_Земчёнок12345678_edit.docx#_Toc69773322)

[1.4 Постановка задачи на дипломное проектирование 12](file:///C:\Users\User\Desktop\отчёт_преддипломная_Земчёнок12345678_edit.docx#_Toc69773323)

[2 Архитектура программного обеспечения 13](file:///C:\Users\User\Desktop\отчёт_преддипломная_Земчёнок12345678_edit.docx#_Toc69773324)

[2.1 Общая структурная прилодения 14](file:///C:\Users\User\Desktop\отчёт_преддипломная_Земчёнок12345678_edit.docx#_Toc69773325)

[2.2 Функциональная схема 16](file:///C:\Users\User\Desktop\отчёт_преддипломная_Земчёнок12345678_edit.docx#_Toc69773326)

[2.3 Требования к архитектуре приложения 19](file:///C:\Users\User\Desktop\отчёт_преддипломная_Земчёнок12345678_edit.docx#_Toc69773327)

[Заключение 21](file:///C:\Users\User\Desktop\отчёт_преддипломная_Земчёнок12345678_edit.docx#_Toc69773329)

[Список использованных источников 22](file:///C:\Users\User\Desktop\отчёт_преддипломная_Земчёнок12345678_edit.docx#_Toc69773330)

**ВВЕДЕНИЕ**

В данном отчёте представлена информация о прохождении преддипломной практики.

Целью практики являлся анализ предметной области и создание архитектуры программного обеспечения приложения.

Я, Чечко Артур Дмитриевич, в период с 23.03.2023 г. по 19.04.2023 г. проходил преддипломную практику в государственном учреждении «Администрация свободной экономической зоны «Гомель-Ратон» (далее – администрация СЭЗ). Руководителем практики был назначен заместитель начальника отдела рекламы и информации Краснов Вячеслав Алексеевич.

Администрация СЭЗ является органом управления свободной экономической зоны «Гомель-Ратон», администрация подотчетна Правительству Республики Беларусь, а по вопросам выполнения программ социально-экономического развития Гомельской области подчиняется Гомельскому областному исполнительному комитету.

Свободная экономическая зона – это часть территории Республики Беларусь с определенными границами, в пределах которой в отношении резидентов этой свободной экономической зоны устанавливается и действует специальный правовой режим для осуществления ими инвестиционной и предпринимательской деятельности.

Специальный правовой режим – это совокупность правовых норм, устанавливающих особый режим налогообложения и иного регулирования и предусматривающих более благоприятные, чем общеустановленные, условия для осуществления инвестиционной и предпринимательской деятельности. Специальный правовой режим в СЭЗ устанавливается, в том числе, и в целях создания благоприятных инвестиционных условий и развития отдельных регионов и экономики Республики Беларусь.

Управление СЭЗ «Гомель-Ратон» для достижения целей ее создания и решения поставленных задач осуществляется администрацией СЭЗ «Гомель-Ратон».

Администрацию свободной экономической зоны «Гомель-Ратон» возглавляет глава администрации свободной экономической зоны, назначаемая на должность (освобождаемая от должности) Советом Министров Республики Беларусь по представлению Гомельского областного исполнительного комитета. Полномочия главы администрации свободной экономической зоны определяются Законом от 7 декабря 1998 г. № 213-З «О свободных экономических зонах», уставом администрации свободной экономической зоны «Гомель-Ратон» и иным законодательством.

Главой администрации СЭЗ «Гомель-Ратон» является Ежова Антонина Изотовна.

В структуру администрации свободной экономической зоны «Гомель-Ратон» входят: управление по привлечению и обслуживанию инвесторов, отдел экономического анализа и планирования, отдел рекламы и информации, отдел по развитию и функционированию инфраструктуры СЭЗ, сектор организационно-правовой работы, сектор бухгалтерского учета.

Практику я непосредственно проходил в отделе рекламы и информации.

Во время прохождения практики передо мной были поставлены следующие задачи:

* анализ предметной области;
* анализ архитектуры программного комплекса;
* обзор и анализ информации по разработке веб-приложений;
* разработка функциональной модели приложения;
* разработка структуры и состава информационного обеспечения приложения;
* подготовка отчёта по преддипломной практике.

# 1 АНАЛИЗ ПРЕДМЕТНОЙ ОБЛАСТИ

## 1.1 Анализ предметной области

Автоматизация администрации свободных экономических зон (СЭЗ) является важным направлением для повышения эффективности работы органов государственной власти, управляющих СЭЗ. Это позволяет значительно сократить время на обработку документов и упростить процессы взаимодействия между участниками инвестпроектов в СЭЗ.

Использование современных информационных технологий и автоматизированных систем в управлении СЭЗ дает возможность ускорить процессы кредитования, регистрации организаций и лицензирования. Это особенно актуально в условиях глобализации рынков, когда привлечение иностранных инвестиций становится необходимым для развития экономики страны.

Кроме того, автоматизация администрации СЭЗ способствует повышению качества предоставляемых услуг, улучшению условий инвестирования и созданию благоприятной инвестиционной среды в стране. Это в свою очередь приводит к увеличению количества предприятий, созданию новых рабочих мест, расширению производства и повышению экспортных возможностей.

В целом, автоматизация администрации СЭЗ является необходимой мерой для развития экономики страны и повышения ее конкурентоспособности на мировом рынке. Она позволяет обеспечить прозрачность, эффективность и быстроту работы органов государственной власти в управлении СЭЗ, что способствует дальнейшему развитию экономики в целом.

## 1.2 Аналитический обзор существующих аналогов

Аналоги в Беларуси существуют, но разнообразием по функциональности выделиться не могут, поэтому мы будем рассматривать только некоторые из них. Как известно свободная экономическая зона(СЭЗ) есть в каждой крупной области РБ (Гомель, Витебск, Гродно, Могилев, Брест, Минск), соответственно у каждой СЭЗ имеется свой личный сайт, который имеет схожую структуру с остальными, а именно ограничивается только предоставлением определенного перечня информации необходимой для потенциального резидента, формой обратной связи и страницей новостей.

Из всех шести свободных экономических зон Беларуси можно выделить пару достойных внимания, а именно: СЭЗ «Брест» и СЭЗ «Гродноинвест». Далее рассмотрим их подробнее.

Сайт СЭЗ «Гродноинвест"» имеет ряд преимуществ, которые делают его удобным и полезным для пользователей:

1. Информативность: Сайт предоставляет информацию о СЭЗ «Гродноинвест», ее возможностях и бенефитах для инвесторов. На сайте представлены подробные описания услуг, которые предоставляет СЭЗ, а также характеристики зон и перечень инвестпроектов.

2. Обновляемость: Сайт регулярно обновляется и информация на нем всегда актуальна. Он содержит последние новости о деятельности СЭЗ и важных событиях, которые могут быть полезны для инвесторов.

3. Легкость использования: Сайт легок в использовании и навигации, что позволяет пользователям находить нужную информацию без труда. Ваши пользователи быстро найдут информацию о зоне, услугах и условиях ее использования.

4. Оформление: Сайт имеет привлекательный и современный дизайн, что сделает его привлекательным для своей аудитории. Это создает хорошее впечатление на пользователей и делает работу с сайтом еще более приятной.

5. Мобильная версия: Сайт имеет адаптивный дизайн, что позволяет отображать его на мобильных устройствах. Это дает возможность пользователям получить доступ к сайту и информации, когда им удобно и без проблем.

6. Доступность: Сайт СЭЗ «Гродноинвест» доступен на русском и английском языках, что позволяет привлекать инвесторов со всего мира.

В целом, сайт СЭЗ «Гродноинвест» является качественным и информативным ресурсом для инвесторов, который поможет им получить всю необходимую информацию о зоне и условиях работы в ней [5].

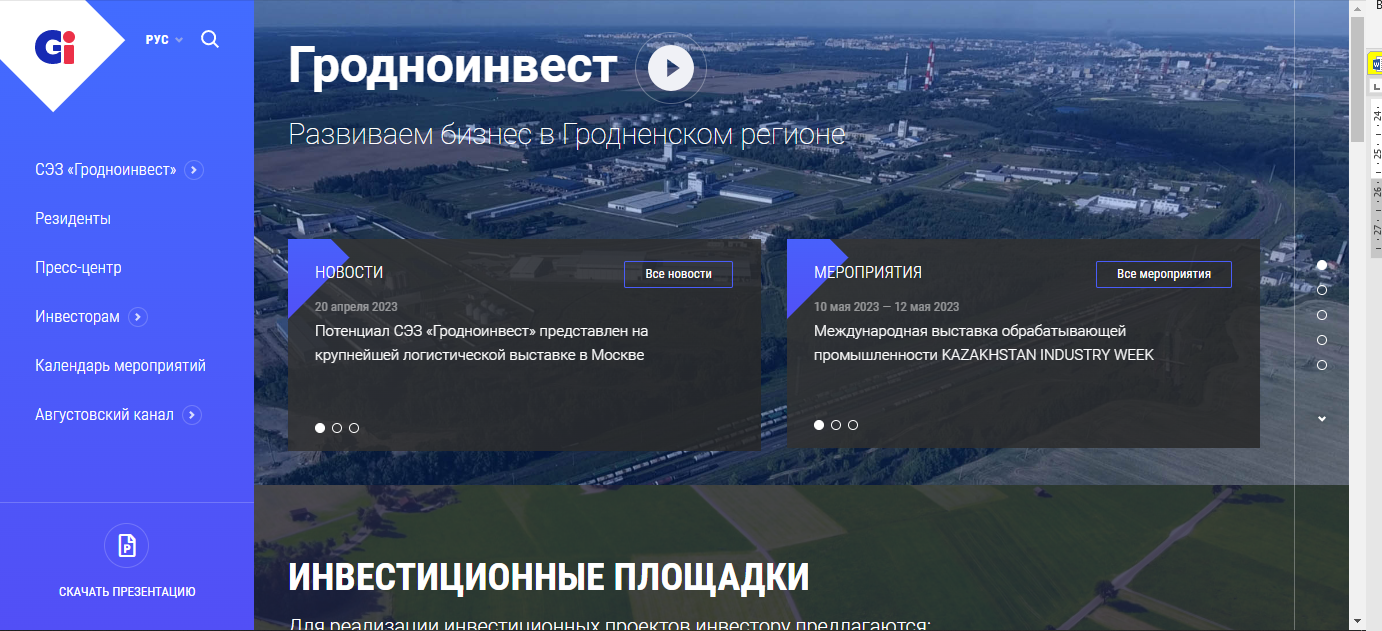


Рисунок 1.1 ­– Главная страница сайта СЭЗ «Гродноинвест»

Сайт СЭЗ «Брест» – это официальный информационный портал Специальной экономической зоны в Брестской области, который предоставляет сотни тысяч пользователей в Международной площадке доступ к важным и актуальным новостям из мира бизнеса, инвестиций, экономики и права.

Среди главных преимуществ сайта можно выделить следующие:

1. Информативность и актуальность: сайт постоянно обновляется в соответствии с последними новостями из СЭЗ «Брест» и других связанных с ней областей, что обеспечивает пользователям свежую и актуальную информацию.

2. Удобный интерфейс и простота использования: сайт имеет интуитивно понятный интерфейс, который помогает пользователям быстро и легко найти нужную информацию.

3. Поддержка мультиязычности: сайт доступен на нескольких языках, что делает его удобным для использования для международных партнеров и инвесторов.

4. Разнообразие информации: на сайте можно найти подробные сведения о правовых аспектах деятельности в СЭЗ, инвестиционные возможности и услуги, документы и образцы договоров и многое другое.

5. Функционал онлайн-консультаций: на сайте существуют разделы для онлайн-консультаций, где пользователи смогут задать вопросы экспертам и получить квалифицированные ответы.

Суммируя вышеизложенное, можно утверждать, что сайт СЭЗ «Брест» является необходимым ресурсом для всех, кто занимается или желает заниматься бизнесом в Беларуси, и предоставляет все необходимые инструменты и ресурсы для успешного развития бизнеса в рамках данной экономической зоны [6].

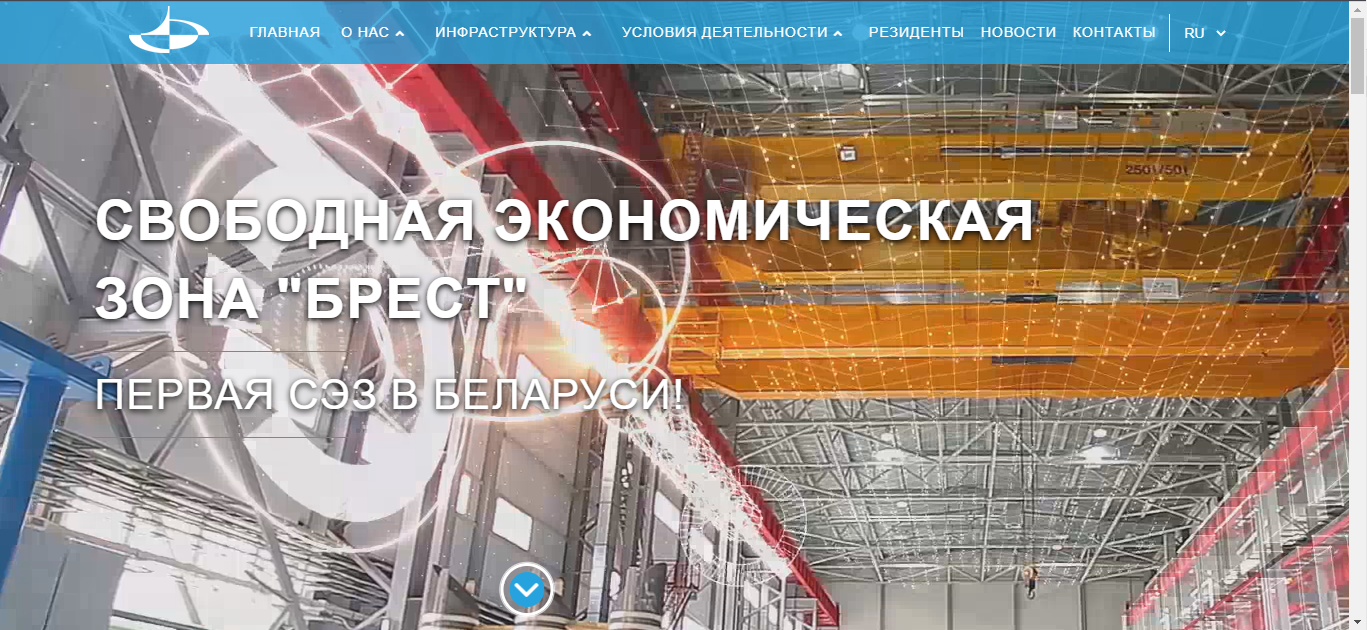


Рисунок 1.2 ­– Главная страница сайта СЭЗ «Брест»

**1.3 Обзор технологий для реализации программного обеспечения**

***1.3.1*** JavaScript – это динамический язык программирования, который используется для разработки интерактивных веб-страниц и приложений. Перед тем как мы перейдем к его приемуществам, давайте рассмотрим историю создания языка программирования JavaScript.

История JavaScript.

JavaScript был создан в 1995 году программистом Бренданом Айком в компании Netscape под названием Mocha. Впоследствии его переименовали в LiveScript и выпустили в 1996 году вместе с Netscape Navigator 2.0. В том же году компания Microsoft начала разработку языка программирования под названием JScript, который был совместим с JavaScript.

В 1997 году был выпущен стандарт ECMAScript, который определяет язык JavaScript. С тех пор язык продолжает развиваться, вносятся изменения, включаются новые функциональности.

Преимущества JavaScript:

**Широкое применение** – JavaScript используется на почти всех веб-страницах и веб-приложениях.

**Простота** – язык довольно прост в освоении, благодаря чему новички могут быстро понять его синтаксис и начать создавать приложения.

**Интерактивность** – JavaScript позволяет создавать интерактивные веб-страницы, которые могут реагировать на действия пользователя.

**Обработка форм** – язык позволяет легко обрабатывать данные, вводимые пользователями в формы на веб-страницах.

**Библиотеки и фреймворки** – существует множество библиотек и фреймворков, расширяющих функциональность JavaScript.

**Кросс-платформенность** – язык работает на всех основных операционных системах (Windows, Mac OS, Linux) и позволяет создавать приложения, которые будут выглядеть одинаково на всех устройствах.

**Быстрота** – JavaScript выполняется непосредственно в браузере, благодаря чему он быстро обрабатывает запросы.

**Совместимость** – JavaScript может взаимодействовать с другими языками, такими как HTML и CSS, благодаря чему можно создавать более сложные приложения.

JavaScript – это один из важнейших языков программирования на сегодняшний день. Он позволяет создавать интерактивные, функциональные и привлекательные веб-приложения для пользователей со всего мира [3].

***1.3.2*** React – это JavaScript-библиотека для создания пользовательских интерфейсов. Библиотека разработана Facebook и поддерживает множество компонентов, из которых можно создавать сложные пользовательские интерфейсы. Первоначально библиотека была создана для веб-разработки, но в настоящее время React может использоваться для разработки как веб-приложений, так и мобильных приложений.

Центральной концепцией React является использование компонентов, которые могут быть повторно использованы в разных местах приложения. React использует JSX (расширенный синтаксис JavaScript), который позволяет встраивать HTML-код в JavaScript-код, что упрощает написание кода и повышает его читаемость.

Для работы с React нужно установить Node.js и пакетный менеджер npm. Затем нужно создать новый проект, установив React с помощью команды npm install react.

Каждый компонент React может иметь состояние, который может изменяться в зависимости от действий пользователя или других факторов. Это позволяет компонентам быть интерактивными.

React также имеет свою систему маршрутизации, которая позволяет создавать мультистраничные веб-приложения, а также поддерживается большим сообществом разработчиков, что облегчает поиск решения при возникновении проблем.

В целом, React – это эффективный выбор для создания масштабируемых и быстрых веб-приложений [1].

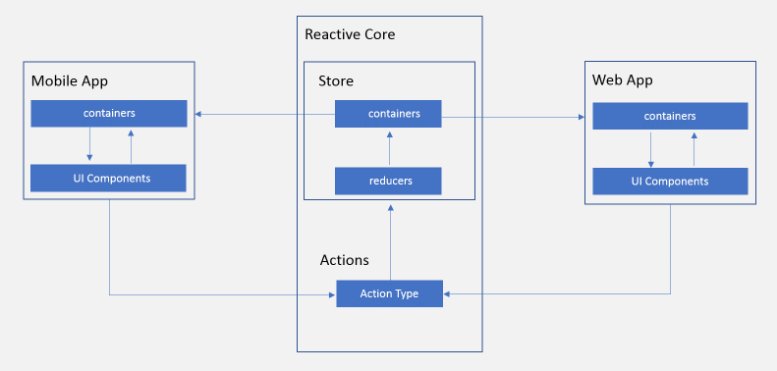


Рисунок 1.3 – архитектура React приложения

***1.3.3*** Firebase – это платформа для разработки мобильных и веб-приложений, которая разрабатывается компанией Google. Она предоставляет сервисы для хранения, анализа и автоматической обработки данных, а также инструменты для создания эффективных пользовательских интерфейсов.

Firebase включает в себя следующие сервисы:

1. Realtime Database – это база данных в режиме реального времени, которая позволяет создавать приложения, которые могут синхронизировать данные мгновенно. Этот сервис использует протокол WebSocket, и это очень удобно для создания приложений, которые обновляются автоматически, например, мессенджеры или социальные сети.

2. Cloud Firestore – еще одна база данных от Firebase, которая является документо-ориентированной и предоставляет возможность хранить, подключать и извлекать данные. Firestore использует API-интерфейс REST, что делает его более гибким и позволяет использовать его в любом языке программирования.

3. Authentication – сервис для авторизации пользователей и контроля доступа. Он поддерживает различные методы аутентификации, включая электронную почту, телефон, социальные сети и веб-провайдеров.

4. Hosting – сервис для хостинга веб-страниц и статических файлов. Firebase позволяет размещать сайты на высокопроизводительных серверах Google и обеспечивает быструю загрузку контента для пользователей.

5. Cloud Functions – сервис для создания и развертывания настраиваемого кода на стороне сервера с помощью Node.js. Этот сервис позволяет создавать скрипты для автоматизации задач и обработки данных, например, отправки уведомлений.

6. Cloud Messaging – сервис для отправки сообщений на устройства, которые имеют приложение на Firebase. Это удобно для отправки уведомлений, напоминаний и других сообщений.

Все эти сервисы интегрируются в одну платформу и могут использоваться совместно, что делает Firebase мощным инструментом для разработки мобильных и веб-приложений. Он также имеет широкое сообщество и большое количество документации, что делает его доступным для новичков и профессионалов [4].

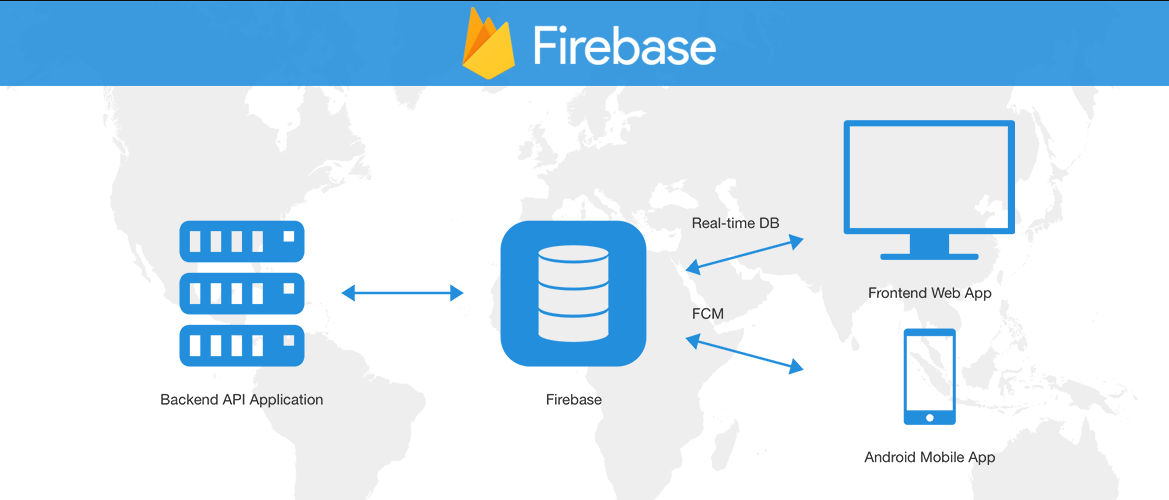


Рисунок 1.4 – архитектура веб приложения с Firebase

***1.3.4*** React-Bootstrap – это набор компонентов Bootstrap, которые были переписаны на React. Библиотека предлагает множество готовых компонентов, которые разработчики могут использовать для создания пользовательских интерфейсов. Они содержат те же ключевые функции Bootstrap, включая сетку, кнопки, формы, навигационные элементы и т.д.

Использование React-Bootstrap имеет несколько преимуществ:

Он предлагает модульный подход к созданию пользовательского интерфейса: Разработчики могут легко использовать отдельные компоненты, которые им нужны, вместо вынужденного использования всего набора Bootstrap.

React-Bootstrap обеспечивает простоту разработки: Разработчики могут сконцентрироваться на пользовательском интерфейсе, не тратя время на решение проблем с css и поддержкой браузера. Кроме того, React-Bootstrap предлагает поддержку различных устройств, так что пользовательский интерфейс будет одинаково отображаться как на настольных, так и на мобильных устройствах.

React-Bootstrap предоставляет готовые решения: Разработчики могут использовать готовые компоненты и изменять их на свой вкус, что значительно ускоряет процесс разработки и улучшает производительность.

Разработчики могут использовать React-Bootstrap для создания сложных пользовательских интерфейсов с помощью уже готовых компонентов. Библиотека легко интегрируется с React. Она также поддерживает множество стилей, которые пользователи могут использовать в своих проектах [2].

## 1.4 Постановка задачи на дипломное проектирование

Цель дипломного проектирования – разработать Web-приложение на React, JS и Firebase, которое автоматизирует процессы свободной экономической зоны (СЭЗ).

Задачи проекта:

1. Разработать прототип приложения, определить функциональные и нефункциональные требования;

2. Разработать дизайн и интерфейс пользователя;

3. Интегрировать Firebase в приложение для хранения данных и обеспечения безопасности;

4. Реализовать возможность создания и управления аккаунтами пользователей приложения;

5. Добавить функциональность для создания заявок на становление резидентом СЭЗ;

6. Реализовать функциональность для контроля кураторами процесса регистрации и утверждения заявок;

7. Добавить механизм генерации отчетов по выполнению работы в СЭЗ для пользователей и кураторов;

В результате выполнения дипломного проекта должно быть разработано Web-приложение, которое позволяет автоматизировать работу в свободной экономической зоне и облегчает взаимодействие между пользователями и кураторами. Приложение должно обеспечивать надежность и безопасность хранения и обработки данных при использовании Firebase.

**2 АРХИТЕКТУРА ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ**

**2.1 Общая структура приложения**

***2.1.1*** Архитектура программного обеспечения (АПО) сайта, написанного на React зависит от многих факторов, включая масштаб проекта, цели разработки, требования к функциональности и многих других. Однако, в целом, можно выделить несколько основных составляющих АПО сайта.  
 React строится на базе компонентов, каждый из которых отвечает за отдельные части интерфейса. Компоненты React обычно создаются в файле с расширением .jsx и представляют собой синтаксический сахар над обычным JavaScript.  
 Для начала, на первой стадии проектирования программа разбивается на множество компонентов. Однако, чтобы компоненты могли работать вместе, необходимо реализовать механизм, позволяющий им взаимодействовать друг с другом. В React для этого используется свойства (props) и состояния (state) компонентов, которые определяют данные, которые компоненту нужно отобразить и изменять [1].



Рисунок 2 – Разбиенине на компоненты в React-приложении

JavaScript – это язык программирования, который можно использовать как для написания фронт-энда, так и для написания бэк-энда приложений. В контексте разработки сайта на React и Firebase, JS используется для взаимодействия с веб-сервером и обменом данными [3].  
 Firebase предоставляет множество инструментов для разработки мобильных и веб-приложений, включая базу данных в реальном времени, аутентификацию, хостинг, хранилище и многие другие. Чтобы интегрировать Firebase в приложение на React, необходимо создать проект Firebase и подключить его API. Затем, используя JavaScript, можно записывать данные в базу данных Firebase и получать данные из нее.  
 Программное обеспечение сайта на React и Firebase обычно имеет следующую архитектуру:  
 Компоненты. Как уже упоминалось ранее, React построен на базе компонентов, каждый из которых отвечает за отдельные части пользовательского интерфейса. Компоненты могут быть классами или функциями, которые возвращают JSX элементы.  
 Сервисы. Сервисы – это классы или функции, которые представляют собой дополнительную логику приложения, которая не связана с отображением интерфейса. Например, сервис авторизации может содержать методы для регистрации и входа пользователей.  
 База данных. Firebase предоставляет базу данных в реальном времени, которая может использоваться для хранения пользовательских данных. База данных Firebase имеет структуру древовидной структуры. Для доступа к базе данных в React используется Firebase API.  
 Хранилище. Хранилище – это объект, который содержит в себе состояние приложения. Хранение состояния в объекте позволяет контролировать изменения состояния и обновлять интерфейс пользователя соответствующим образом.  
 Роутер. Роутер – это компонент, который управляет навигацией на сайте. Роутер часто используется в React-приложениях для создания маршрутов веб-страниц.  
 Однако, архитектура программного обеспечения сайта на React, JS и Firebase может быть очень различной, в зависимости от требований к функциональности и масштаба проекта. Но в любом случае, важно организовать приложение таким образом, чтобы оно было масштабируемым, легко поддерживаемым и масштабируемым для изменений и улучшений в будущем [4].



Рисунок 2.1 – архитектура Firebase

***2.2.2*** Приложение написанное на JS и React с использованием Firebase для авторизации, хранения файлов и FirestoreDatabase для хранения данных представляет собой современное приложение с большим набором функционала. Оно позволяет пользователям создавать свои аккаунты, загружать файлы и хранить их на сервере, а также вводить и получать данные, которые будут храниться в базе данных.

Общая схема приложения будет следующей: пользователь сначала попадает на страницу входа, где он может воспользоваться своим аккаунтом Firebase или зарегистрироваться, используя свой адрес электронной почты и пароль. В случае успешной авторизации пользователь попадает на главную страницу приложения, где он может работать со своими файлами и данными.

Далее происходит взаимодействие с Firebase для загрузки, хранения и отображения файлов. Firebase предоставляет удобный API для загрузки файлов на сервер, а также возможность сохранения метаданных о каждом загруженном файле (например, имя, тип и размер файла). Кроме того, Firebase имеет возможность автоматической генерации ссылок для скачивания или просмотра файлов, что позволяет облегчить работу с файлами в приложении.

FirestoreDatabase используется для хранения и обработки данных. Для работы с FirestoreDatabase используется набор API, который позволяет выполнять операции чтения и записи данных, а также отслеживать изменения в реальном времени. Приложение может хранить такие данные, как названия файлов, описания, теги и другие параметры, которые пользователь может использовать для отображения и фильтрации файлов в своих списках.

Также можно использовать FirestoreDatabase для хранения информации о пользователе и его аккаунте. Например, можно хранить список загруженных пользователем файлов и устанавливать соответствующие права доступа для других пользователей. Таким образом, FirestoreDatabase позволяет создавать многопользовательские приложения с различными уровнями доступа к файлам и данным.

Взаимодействие между компонентами приложения может быть организовано с помощью технологий реактивного программирования, таких как RxJS. RxJS предоставляет мощные инструменты для работы с потоками данных, что позволяет более эффективно управлять изменениями в данных и обновлять пользовательский интерфейс приложения.

В целом, приложение, написанное на JS и React с использованием Firebase для авторизации, хранения файлов и FirestoreDatabase для хранения данных, представляет собой многофункциональное приложение с большим количеством возможностей и широким набором функционала, которые позволяют удобно и эффективно управлять своими файлами и данными в облаке.

**2.2 Функциональная схема**

На рисунках 2.2-2.5 представлена функциональная схема пользователей, составленная для данной системы. Функциональная схема необходима для того, чтобы показать взаимодействия между пользователями системы

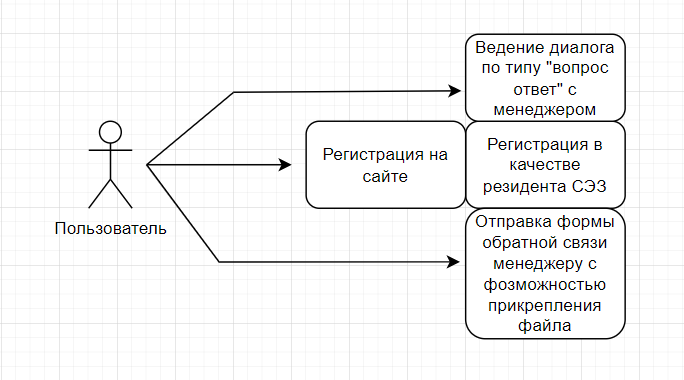


Рисунок 2.2 – Функциональная схема пользователя

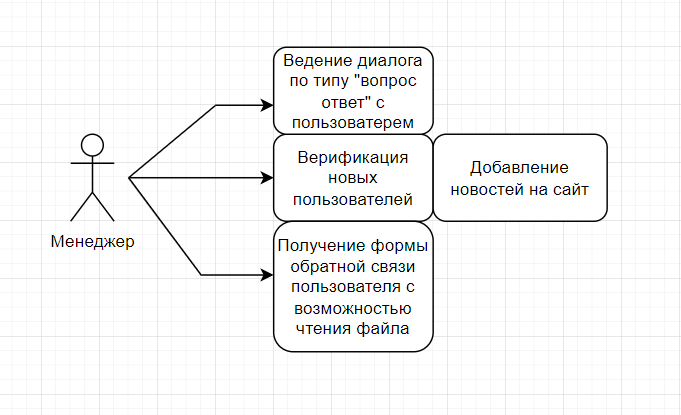


Рисунок 2.3 – Функциональная схема менеджера

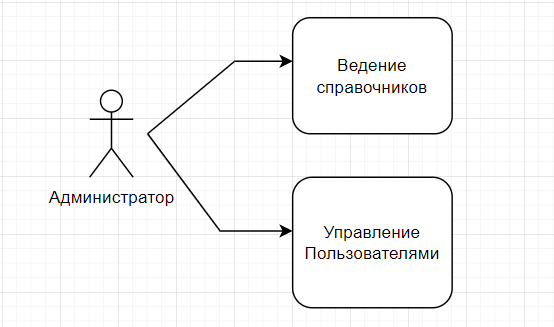
****

Рисунок 2.4 – Функциональная схема администратора

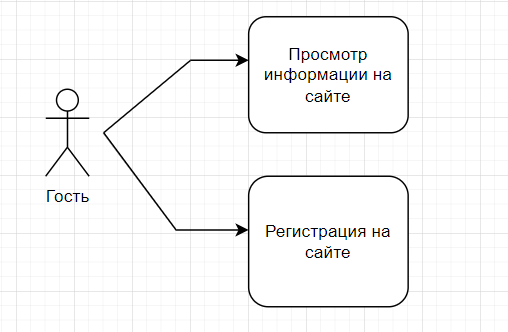
****

Рисунок 2.5 – Функциональная схема гостя

В данной системе есть 4 роли, которые могут взаимодействовать с системой.

У админа есть возможность ведения нескольких справочников, которые нужны системе, а также другим членам. Также имеет возможность распоряжаться правами пользователей, которые находятся в данной системе. Смотреть статистику по резиденту или пользователю (количество обращений, степень доверия и тд.). И самая основная функция – это абсолютная власть над всеми пользователями, а именно регистрация или удаление.

Менеджер – это одна из ключевых ролей данной системы. Данный человек администрации следит за всеми пользователями и их обращениями, присваивает им степень доверия, верифицирует зарегистрированные аккаунты, а также ведет новостную ленту сайта

Функционал гостя (кроме просмотра требуемой информации на сайте) исходя из функций менеджера соответственно может регистрироваться на сайте и отправлять данные менеджеру, который проверяет их и если человек действительно является тем кто он есть отправляет ему на электронную почту временный логин и пароль который нужно будет сменить на действующий в личном кабинете пользователя.

Пользователь в свою очередь заполняет формы для регистрации в качестве резидента и отправляет их менеджеру на «Одобрение», а также ведет диалог с ним по принципу «вопрос – ответ» по интересующим пользователя вопросам.

**2.3 Разработка информационной модели**

***2.2.1*** Одна из главных задач множества существующих приложений – хранение данных в каком-либо виде. Базы данных – неотъемлемая часть работы подавляющего большинства программного обеспечения, существующего на сегодняшний день. Использование баз данных является достаточно эффективным способом хранить данные.

Как уже упоминалось ранее, Firebase предоставляет множество инструментов для разработки веб-приложений, включая базу данных в реальном времени, аутентификацию, хостинг, хранилище, что достаточно удобно для своевременного реагирования членам администрации СЭЗ для регистрации новых потенциальных инвесторов.

Для заполнения базы данных были использованы данные личной электронной почты для проверки авторизации с помощью сервисов Google, а также данные, внесённые в базу данных во время тестирования и работы с программой.

На момент написания отчета по преддипломной практике было подключено и в полной мере программно-реализовано два сервиса, а именно: сервис для аутентификации (Authentication) рисунок 2.7 и сервис для хранения файлов (Storage) рисунок 2.6

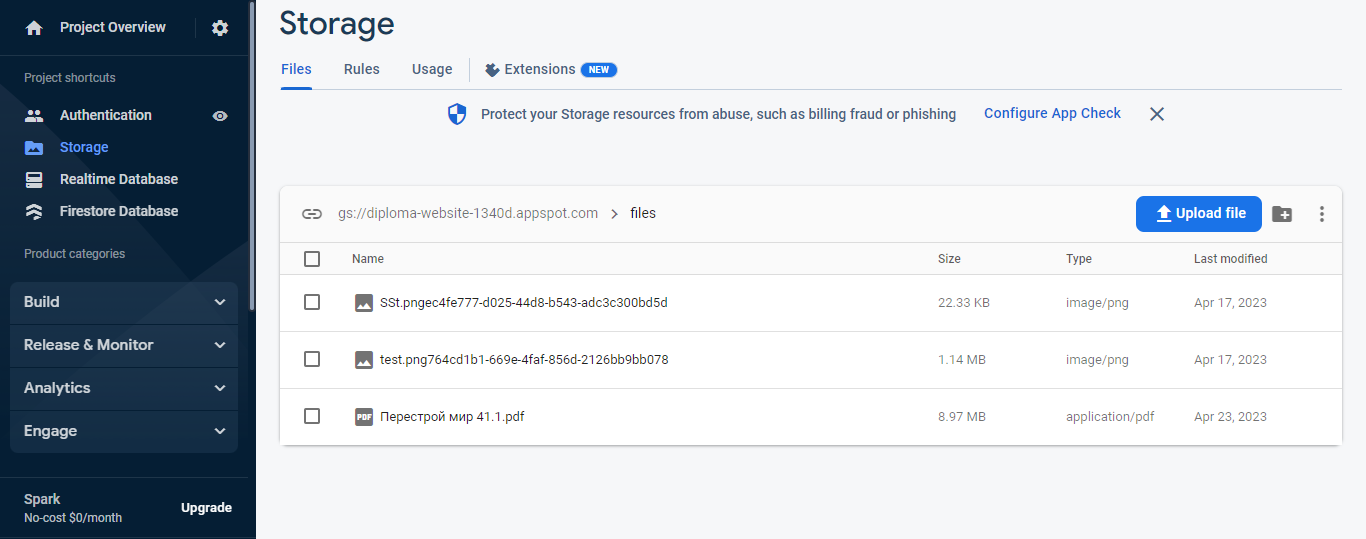
****

Рисунок 2.6 – сервис для хранения файлов в Firebase



Рисунок 2.7 – сервис для авторизации и хранения данных о пользователях в Firebase

***2.2.2*** На основе анализа предметной области можно выделить следующие сущности:

* аутентификация (*Authentication*);
* хранилище (*Storage*);
* новости (*News*);
* личный кабинет (*Office*);
* чат (*Chat*);
* обращения (Petition)
* предприятия (*Company*);

Сущность «Аутентификация» Представляет собой сервис Firebase для хранения информации о пользователе.

Сущность «Хранилище Представляет собой сервис Firebase для хранения файлов, прикрепленных к электронному обращению.

Сущность «Новости» представляет собой некоторую новость или новости предоставленную всем пользователям и гостям Web-приложения.

Сущность «Личный кабинет» Представляет собой кабинет пользователя для хранения личной информации, а также мониторинга рассмотрения заявки на регистрацию резидентом.

Сущность «Чат» служит для ведения диалогу между пользователем и менеджером.

Сущность «Обращения» хранит в себе все электронные обращения пользователей для просмотра и реагирования на них менеджеру Web-приложения.

Сущность «Предприятия» служит для регистрации предприятия в приложении (расширенная регистрация пользователя).

# ЗАКЛЮЧЕНИЕ

В ходе прохождения преддипломной практики была проанализирована предметная область разрабатываемого продукта, был произведён аналитический обзор существующих аналогов на рынке приложений. На основе результатов анализа предметной области были выделены группы основных бизнес-процессов и роли пользователей. Также был произведён обзор технологий для реализации программного обеспечения.

Было произведено функциональное моделирование, логическое и физическое проектирование базы данных, анализ информационной модели программного комплекса. Также в ходе выполнения преддипломной практики были изучены методы реализации функциональных возможностей программы.

Были подготовлены и написаны главы дипломного проекта:

* введение;
* анализ предметной области;
* архитектура программного комплекса.

В соответствии с календарно-тематическим планом были пройдены следующие этапы:

* прохождение курса по адаптации;
* обзор и анализ информации по разработке веб-приложений;
* разработка функциональной модели приложения;
* разработка структуры и состава информационного обеспечения приложения;
* подготовка отчёта по преддипломной практике.

За время прохождения преддипломной практики все поставленные задачи были выполнены в полном объёме.

**Список использованных источников**

1. React [Электронный ресурс]. – React, 2021. – Режим доступа: https://react.dev/. – Дата доступа – 21.04.2023.
2. React Bootstrap [Электронный ресурс]. – react-bootstrap, 2023. – Режим доступа: <https://react-bootstrap.github.io/> – Дата доступа – 21.04.2023.
3. Java Script Documentation [Электронный ресурс]. – js-documentation, 2023. – Режим доступа: https://js-documentation.netlify.app/ – Дата доступа – 21.04.2023.
4. Firebase – Американская компания, поставщик облачных услуг. [Электронный ресурс]. – Firebase, 2021. – Режим доступа: https://console.firebase.google.com/. – Дата доступа – 21.04.2023.
5. Свободная экономическая зона «Гродноинвест» [Электронный ресурс]. – grodnoinvest, 2023. ­ – Режим доступа: https://grodnoinvest.by/. – Дата доступа – 21.04.2023.
6. Свободная экономическая зона «Брест» [Электронный ресурс]. – fezbrest, 2023. – Режим доступа: https://fezbrest.com/ – Дата доступа – 21.04.2023.