| **МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ** |
| --- |
| **ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ «МОСКОВСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ ТЕХНОЛОГИЙ И УПРАВЛЕНИЯ ИМЕНИ К.Г. РАЗУМОВСКОГО (ПЕРВЫЙ КАЗАЧИЙ УНИВЕРСИТЕТ)»**  **(ФГБОУ ВО «МГУТУ ИМ. К.Г. РАЗУМОВСКОГО (ПКУ)»)** |
|  |
| **УНИВЕРСИТЕТСКИЙ КОЛЛЕДЖ ИНФОРМАЦИОННЫХ ТЕХНОЛОГИЙ** |

| **КУРСОВОЙ ПРОЕКТ** |
| --- |
|  |
| по междисциплинарному курсу: МДК.03.01. Технология разработки программного обеспечения. |
|  |
| на тему: Разработка веб-сайта для проведений видеоконференций. |
|  |
| студента группы 090203-9o-19/1  специальности 09.02.03 Программирование в компьютерных системах |
| Каплуна Олега Денисовича |

| Студент |  | О.Д. Каплун |
| --- | --- | --- |
| Руководитель курсового проекта |  | С.Ю. Кузьменко |
| Председатель ПЦК специальности 09.02.03 Программирование в компьютерных системах |  | А.И. Глускер |

| Дата защиты «\_\_\_\_» \_\_\_\_\_\_\_\_\_\_\_ 2022 г. | | |
| --- | --- | --- |
| Оценка:\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ | | |
| Заведующий отделением № 3 |  | И.А. Миланова |

Москва

2022

[**ВВЕДЕНИЕ**](#_3oaadeohbyj9) **3**

[**1 ОБЩАЯ ЧАСТЬ**](#_504hxf5omr76) **4**

[1.1 Исследование предметной области](#_iiilb6j5rewr) 4

[1.1.1 Категории видеоконференцсвязи](#_48lgid9qz6nq) 4

[1.1.2 Оборудование для видеоконференцсвязи](#_iu4d71dtbhmu) 4

[1.1.3 Протоколы для видеопередачи](#_mh3c8pwy7ya7) 6

[1.1.4 Обзор аналогов](#_obpulx6528bq) 7

[1.2 Анализ и выбор редактора кода](#_sqgvym7qb8am) 10

[1.2.1 Visual Studio Code](#_rtkogbn808j0) 10

[1.2.2 Sublime Text](#_2bdt6u892koe) 11

[1.2.3 Atom](#_cva6tqu49kbj) 12

[1.3 Обзор используемых инструментов разработки](#_4fzx3uz724z4) 14

[1.3.1 JSX](#_z3mlw2hv225s) 14

[1.3.2 Node.js](#_1xmnp01cz30) 15

[1.3.3 Sass](#_azokajleilz3) 16

[1.4 Обзор используемых библиотек](#_2khsw399dm4o) 17

[1.4.1 React.js](#_tlnjpnmad7vs) 17

[1.4.2 WebRTC](#_xh4njwsoraz5) 18

[**2 СПЕЦИАЛЬНАЯ ЧАСТЬ**](#_sttv3baxlzbi) **20**

[**ЗАКЛЮЧЕНИЕ**](#_y7n1dtq97ig6) **21**

[**СПИСОК ИСПОЛЬЗУЕМЫХ ИСТОЧНИКОВ**](#_9253zd5e10r4) **23**

# ВВЕДЕНИЕ

Актуальность данной темы заключается в том, что видеоконференции за короткое время вынужденного карантина в 2020 году стали новою реальностью. В условиях пандемии заниматься деятельностью подразумевающую в себе контакт с людьми был невозможен и был исключительно онлайн. За несколько недель люди привыкли к такому образу жизни, за счет чего платформы для видеосвязи стали бить все рекорды по скачиваниям.

Целью проекта является закрепление теоретических знаний и приобретение практических навыков в области разработки программного продукта и его описания на основе аудиторного и самостоятельного изучения специальной литературы.

Объектом исследования является разработка веб-сайта.

Предметом исследования является разработка функциональных возможностей веб-сайта.

Сообразно с этим в работе поставлены следующие задачи:

* проанализировать предметную область;
* спроектировать диаграмму вариантов использования;
* изготовить макет веб-сайта под любое устройство средствами Figma;
* подобрать подходящие технологии и фреймворки для написания веб-сайта;
* разработать веб-сайт средствами Microsoft Visual Studio Code;
* протестировать данный проект в различных браузерах.

Для решения задач выделены следующие методы исследования: изучение специальной литературы по теме курсового проекта и анализ данных, полученных из различных источников.

# 1 ОБЩАЯ ЧАСТЬ

## 1.1 Исследование предметной области

### 1.1.1 Категории видеоконференцсвязи

Учитывая функции и цели применения, оборудование видеоконференцсвязи систематизируется на категории и классы.

Индивидуальные системы - обеспечивают возможность индивидуального видеообщения пользователя в режиме реального времени, не покидая своего рабочего места.

Групповые системы - предназначены для проведения групповых сеансов видеоконференцсвязи в переговорных (совещательных) комнатах.

Отраслевые системы - применяются непосредственно в определенной отрасли. Например, в медицинской отрасли очень часто применяют системы для проведения операций (телемедицина).

Мобильные системы - позволяют за короткое время организовать сеанс видеоконференцсвязи в нестандартных условиях. Данные системы обычно используются государственными органами, принимающими оперативные решения.

Административные системы - представляют собой совокупность аппаратно-программных средств администрирования/управления многоточечными сеансами видеоконференцсвязи с использованием различного оконечного оборудования.

### 1.1.2 Оборудование для видеоконференцсвязи

Оборудование для видеоконференций разделяется на три основных вида:

* Персональные системы видеоконференций, использующие возможности персонального компьютера и веб-камеры;
* Системы видеоконференций для переговорных комнат, предназначенные для установки в конференц-залах малых и средних размеров;
* Системы видеоконференций для залов заседания, используемые в больших конференц-залах и поддерживающие максимально возможное количество дополнительных сервисов.

Для организации видеоконференций используются следующие устройства:

1. Терминалы абонентов с поддержкой аудио и видеосвязи - индивидуальные или групповые видеосистемы или IP-телефоны. Терминал состоит из следующих компонентов:
   1. кодек (основное обрабатывающее устройство);
   2. камера;
   3. микрофон;
   4. отображающее устройство.
2. Серверы многоточечной связи (MCU). MCU H.323 совмещает в себе обязательный многоточечный контроллер, управляющий соединениями, и один или несколько опциональных мультимедийных процессоров, назначение которых - микширование аудио и видеосигналов, поступающих от многих участников;
3. Шлюзы (gateways) соединяют коммутируемые ISDN-сети с пакетными IP-сетями. В функции шлюза входит преобразование форматов передачи данных и коммуникационных процедур (H.225/H.221 и H.245/H.242);
4. Контроллеры зоны (gatekeepers) - это программные модули, которые авторизуют подключения, транслируют используемые в системе имена терминалов и шлюзов в IP-адреса, маршрутизируют запросы через шлюзы;
5. Сетевые экраны и прокси-серверы (firewalls и proxies) предотвращают несанкционированный доступ к конференции в случае связи через Интернет.

### 1.1.3 Протоколы для видеопередачи

Стремительный рост передачи мультимедиа-информации предъявляет новые требования к скорости и объемам передачи данных. И для того чтобы удовлетворить все эти запросы, одного увеличения емкости сети недостаточно, необходимы разумные и эффективные методы управления трафиком и контролем загруженности линий передачи.

В приложениях реального времени отправитель генерирует поток данных с постоянной скоростью, а получатель (или получатели) должен предоставлять эти данные приложению с той же самой скоростью. Такие приложения включают, например, аудио- и видеоконференции, живое видео, удаленную диагностику в медицине, компьютерную телефонию, распределенное интерактивное моделирование, игры, мониторинг в реальном времени и др.

Наиболее широко используемый протокол транспортного уровня - это TCP. Несмотря на то, что TCP позволяет поддерживать множество разнообразных распределенных приложений, он не подходит для приложений реального времени. TCP обеспечивает гарантированную доставку пакетов и проверку их целостности и отлично подходит для задач, связанных с передачей важной информации, не привязанной к времени. Протокол TCP осуществляет проверку доставки пакета и в случае неудачной доставки осуществляет повторную передачу. Такой механизм недопустим для передачи в режиме реального времени, так как в случае потери пакета и его повторной передачи остальные пакеты будут вынуждены ждать доставки потерянного пакета, что приведет к существенному разрыву во времени передачи и приема пакетов - что является недопустимым для передачи информации в реальном времени.

Эту задачу решает дейтаграммный протокол UDP, задача которого - быстрая доставка дейтаграмм без установления соединения, повторной передачи и гарантии доставки пакета. Несмотря на эти недостатки, протокол UDP более предпочтителен для передачи данных реального времени, так как обеспечивает более быструю передачу данных.

В типичной среде реального времени отправитель генерирует пакеты с постоянной скоростью. Они отправляются через одинаковые интервалы времени, проходят через сеть и принимаются получателем, воспроизводящим данные в реальном времени по их получении. Однако ввиду изменения времени задержки при передаче пакетов по сети, они могут прибывать через нерегулярные интервалы времени. Для компенсации этого эффекта поступающие пакеты буферизуются и затем предоставляются с постоянной скоростью программному обеспечению, генерирующему вывод.

### 1.1.4 Обзор аналогов

Обзор аналогов может быть довольно полезным процессом, поскольку это не только позволяет определять общую тенденцию данной сферы, но и не допустить тех же ошибок, которые присутствуют в аналогичном продукте.

Рассмотрим программу Zoom - это программа для организации видеоконференций, разработанная компанией Zoom Video Communications.

Она предоставляет сервис видеотелефонии, который позволяет подключать одновременно до 100 устройств бесплатно, с 40-минутным ограничением для бесплатных аккаунтов. Пользователи имеют возможность повысить уровень обслуживания, используя один из тарифных планов, с максимальным числом подключений до 500 человек одновременно, без ограничений по времени.



Рисунок 1 - Внешний вид программы Zoom

Хочется сразу подметить, что у данной платформы существуют достаточно много преимуществ и недостатков, но приведу лишь только основные, на мое мнение:

Преимущества:

* удобное контролирование администратором всеми участниками беседы с помощью панели управления;
* существует интерактивная доска, где можно что-то нарисовать и показать наглядно;
* каждый участник конференции может настраивать под себя отображение конференции — например, сфокусироваться на докладчике или видеть всех участников беседы;
* существует запись звонка;
* предусмотрена функция размытия заднего фона или его замены, для тех кто не хочет показывать пространство вокруг себя.

Недостатки:

* максимальная продолжительность конференции - 40 минут в бесплатной версии;
* довольно сложный интерфейс;
* не так хорошо защищен от «пранкеров», которые могли сорвать школьные уроки или важные собеседования.

Далее рассмотрим платформу Microsoft Teams - это корпоративная платформа, объединяющая в рабочем пространстве чат, встречи, заметки и вложения. Разработана компанией Microsoft и является частью их пакета Office 365.



Рисунок 2 - Внешний вид программы Microsoft Teams

Преимущества:

* пользоваться программой можно бесплатно, кроме того, пользователи бесплатно получают 10 Гб облачного хранилища, куда будут сохраняться совместные проекты, созданные с помощью приложений из пакета Microsoft Office;
* если у одного из собеседников нет доступа к Сети, его можно подключить по номеру телефона;
* также как и в Zoom организатор имеет возможность планировать начало конференции и управлять остальными участниками конференции.

Недостатки:

* у сервиса есть один глобальный минус. Если приходится общаться с людьми, работающими в других компаниях. Участники конференции, которым необходимо высказываться или делиться экраном, должны быть авторизованы под учетными записями той компании, которая начала конференцию.

Проведя исследование предметной области можно сделать вывод, о том что создание сайтов для видеоконференций создают большое кол-во трудностей. А также автору удалось сделать обзор аналогов, благодаря которым, можно задуматься о том, чтобы в дальнейшем при разработке нашего веб-сайта создать отличительные признаки, которых не существуют у конкурентов и сделать их недостатки нашим преимуществом.

## 1.2 Анализ и выбор редактора кода

В настоящее время существуют большое кол-во редакторов кода для написания программного продукта.

При выборе подходящего редактора кода разработчики ориентируются на следующие факторы:

* существует ли поддержка необходимых языков программирования?
* существует ли возможность совместной разработки для командной работы?
* какие дополнительные функции доступны разработчикам?
* можно ли устанавливать дополнительные плагины?
* нужна ли подписка или можно обойтись бесплатной версией?

На основании вышеуказанных вопросов, проведем анализ популярных редакторов кода на сегодняшний день, выделив основные их преимущества и недостатки, а также сделаем свой выбор в пользу того редактора, который будет удовлетворять нашим требованиям и желаниям.

### 1.2.1 Visual Studio Code

Visual Studio Code – это бесплатный редактор кода, который включает в себя инструменты запуска и отладки, интеграцию с системой контроля версий и дает возможность писать код на многих языках программирования: C++, CSS, JavaScript, JSON, PHP, Python и другие.

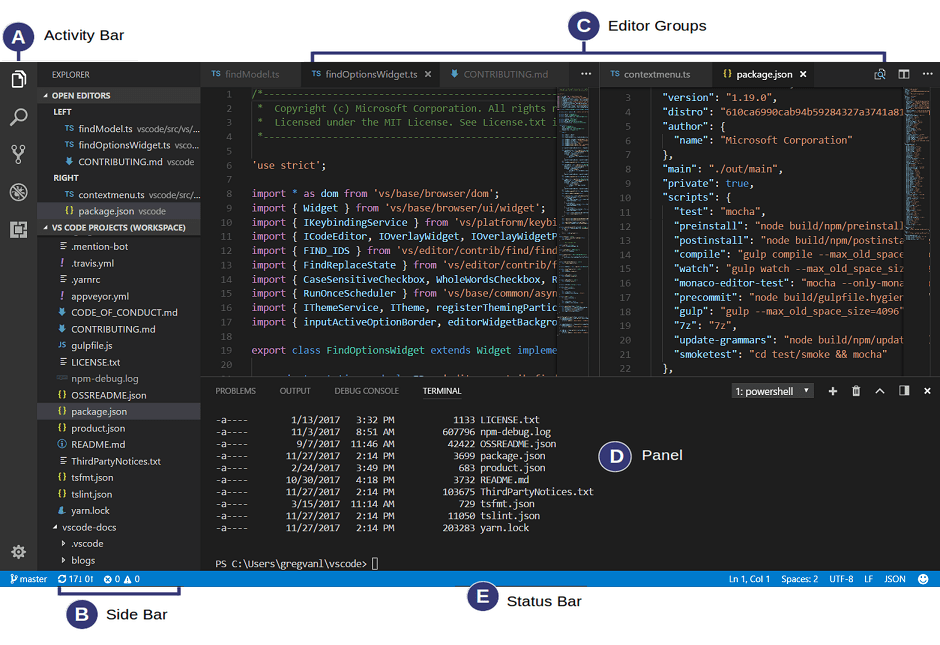


Рисунок 3 - Интерфейс Visual Studio Code

Преимущества:

* кроссплатформенность: поддержка на Windows, macOS и Linux;
* встроенная технология автодополнения кода IntelliSense;
* множество бесплатных расширений в маркетплейсе;
* полностью бесплатный редактор.

Недостатки:

* в Visual Studio Code доступно меньше инструментов, чем в полноценной IDE от Microsoft;
* отсутствует поддержка split screen.

### 1.2.2 Sublime Text

Sublime Text - это редактор кода, хорошо поддерживаемый комьюнити разработчиков. Он включает возможности для упрощения редактирования кода: Goto Anything, сопоставление скобок, множественное выделение и Python API. Поддерживает языки: C++, Python, JavaScript, PHP, SQL и другие.

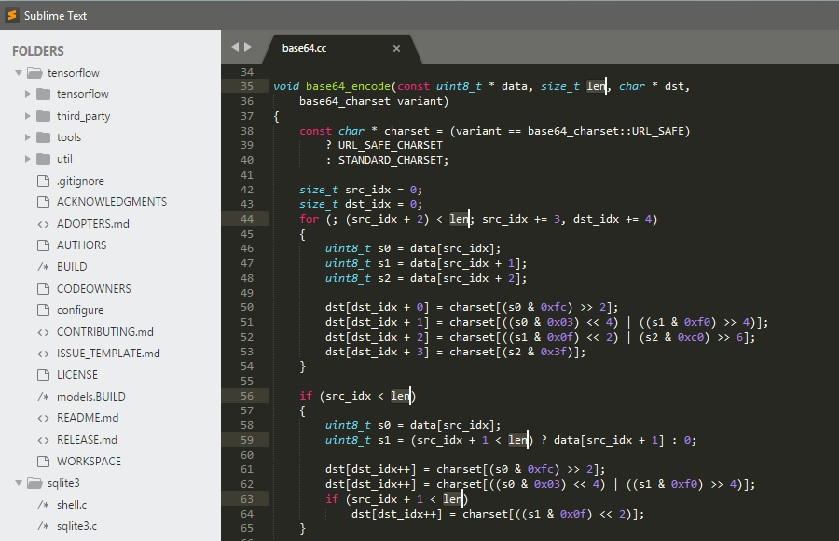


Рисунок 4 - Интерфейс Sublime Text

Преимущества:

* большой выбор дополнений для поддержки синтаксиса и редактирования кода;
* навигация Goto Anything дает возможность быстро получать доступ к файлам;
* позволяет быстро перемещаться между проектами с сохранением изменений;
* множественное выделение позволяет вносить изменения в строки кода сразу в нескольких местах.

Недостатки:

* для полного использования всех функций необходимо приобрести лицензии;
* отсутствует функция автоматической отладки и боковая панель, как, например, в Visual Studio Code.

### 1.2.3 Atom

Atom - это бесплатный опенсорсный редактор кода, позволяющий работать в любой ОС. Поддерживает такие языки, как Python, CSS, JavaScript, Ruby, Java и другие. Позволяет устанавливать большое количество расширений.

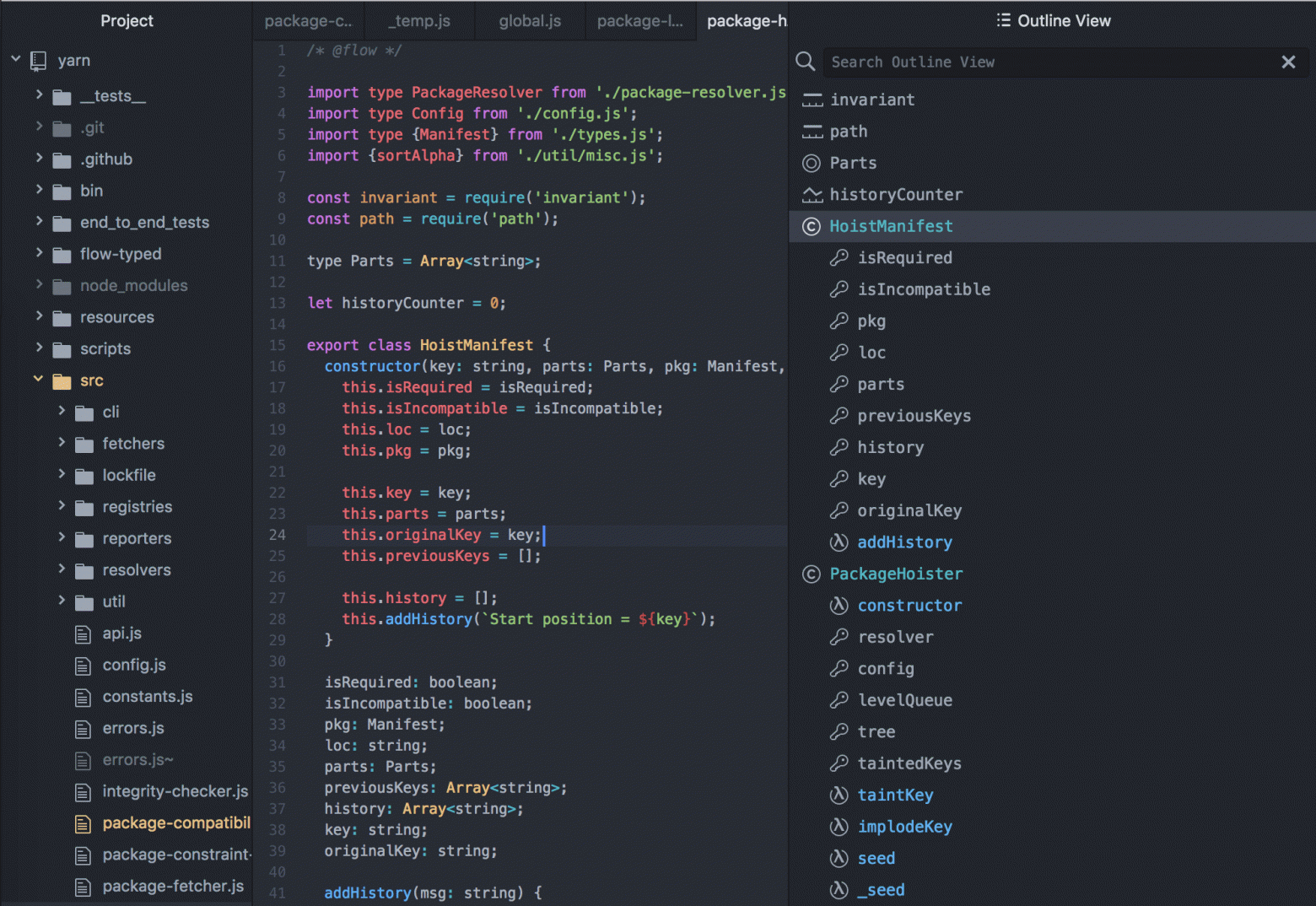


Рисунок 5 - Интерфейс Atom

Преимущества:

* подходит для начинающих;
* редактор можно настроить под свои требования;
* возможность установки новых плагинов;
* мультиязычность;
* интеграция с Node.js, включая запуск веб-сервера прямо из редактора.

Недостатки:

* сильная нагрузка на мощности из-за встроенного браузера, внутри которого постоянно выполняется JavaScript-код.

Проведя анализ популярных редакторов кода, автор сделал свой выбор в использовании Visual Studio Code, т.к данная среда является простой в использовании за счет своего интерфейса, а также из-за большой системы плагинов, написанных разными разработчиками.

## 1.3 Обзор используемых инструментов разработки

Прежде чем приступить к разработке программного продукта, разработчик должен задаться вопросом: Какие инструменты для разработки будут использованы в проекте?

В этой главе будут кратко рассмотрены: React.js и WebRTC, т.к. автор использовал их в своем проекте.

Для того чтобы ответить на этот вопрос, рассмотрим характеристики тех инструментов, которые были использованы автором в этом проекте и выделим основные их преимущества и недостатки.

### **1.**3**.1 JSX**

В React.js используется JSX — специальное расширение синтаксиса для создания объектов JavaScript, но через синтаксис HTML. Такое расширение упрощает модификацию DOM и улучшает читабельность кода.

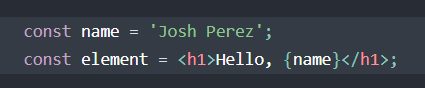


Рисунок 6 - Синтаксис JSX

Преимущества:

* можно внедрять любые допустимые выражения JavaScript внутри фигурных скобок JSX;
* JSX предотвращает инъекционные атаки;
* JSX представляет сжатые объекты элементов страницы, за счет чего их перерисовка происходит быстрее и они занимают меньше ресурсов.

Недостатки:

* синтаксис HTML посреди JavaScript-кода может запутать молодых разработчиков.

### 1.3.2 Node.js

Node.js — среда выполнения, которая базируется на движке Chrome V8 JavaScript.

Технология была впервые представлена в 2009 году Райаном Далем на ежегодном европейском форуме JSConf и сразу же была признана самой значимой частью программного обеспечения в современной вселенной JavaScript.

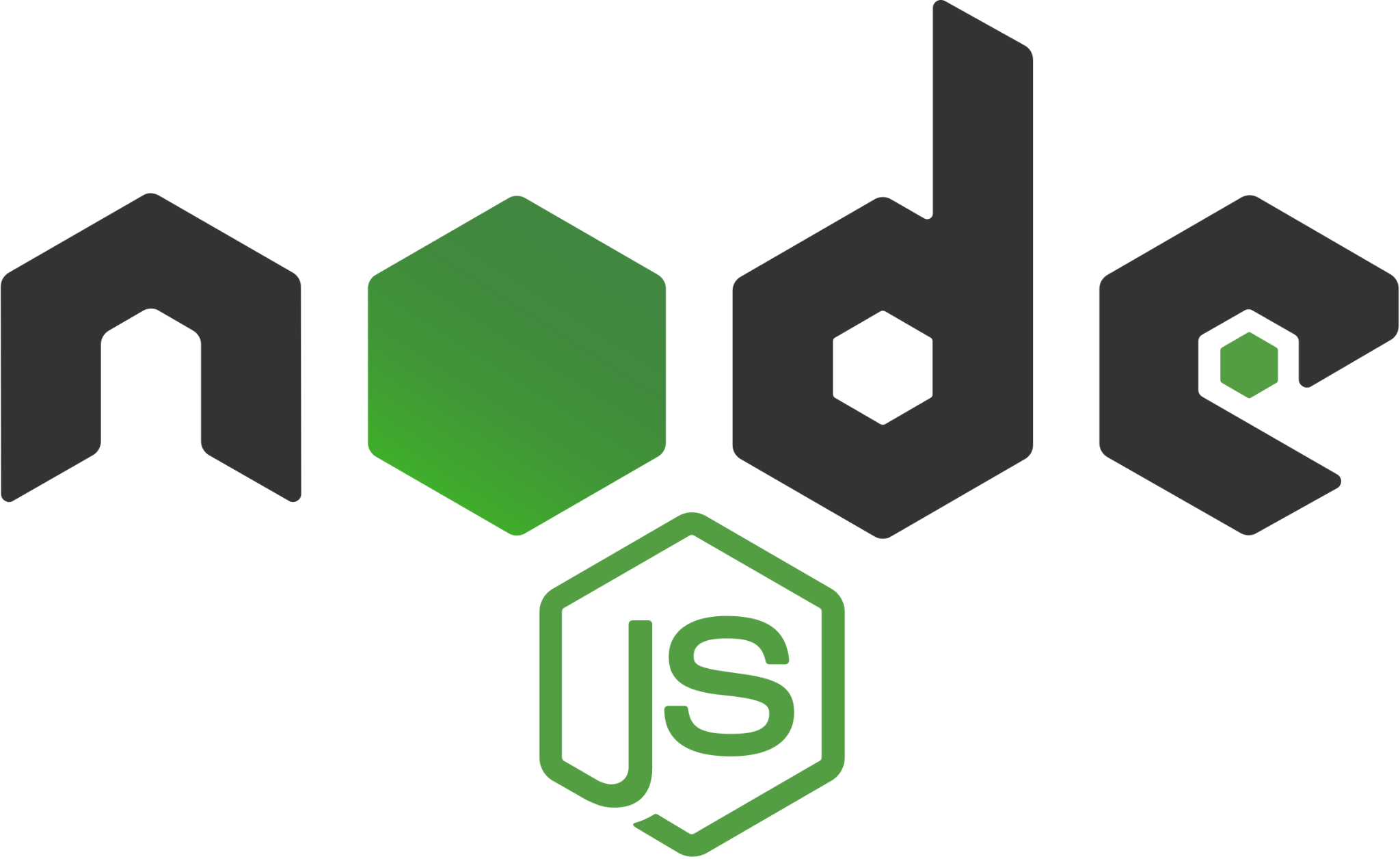


Рисунок 7 - Логотип Node.js

Сильные и слабые стороны Node.js остаются предметом жарких дискуссий среди разработчиков. Для того чтобы внести ясность, стоит проанализировать как плюсы, так и минусы этой среды:

Преимущества:

* надежная технология на базе JavaScript;
* быстрая обработка запросов и эффективная событийная модель;
* идеальный выбор для микро сервисной архитектуры;
* богатая экосистема;
* поддержка IT-гигантов;
* полная поддержка JSON.

Недостатки:

* низкая производительность при работе с тяжёлыми вычислительными задачами;
* зависимость от обратного вызова;
* в реестре npm есть много инструментов, которые имеют низкое качество, либо не имеют четкой документации и не были протестированы должным образом.

### **1.**3**.**3 **S**ass

Sass является одним из наиболее широко используемых CSS-препроцессоров. Он имеет различные функции, помогающие разработчикам писать CSS-код лучше и чище.

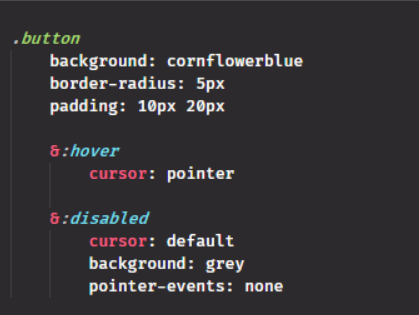


Рисунок 8 - Пример кода с использованием Sass

Преимущества:

* Он позволяет писать чистый CSS-код в стиле языков программирования;
* Это расширение CSS, с помощью которого дизайнеры и разработчики могут работать намного эффективнее и быстрее;
* Sass совместим со всеми версиями CSS, использовать можно любые доступные CSS-библиотеки;
* В нем можно использовать синтаксис вложенности и такие полезные функции, как манипуляция с цветами, математические и другие значения.

Недостатки:

* Разработчику понадобится некоторое время на изучение возможностей препроцессора;
* Если над одним сайтом работает много человек, то им нужно использовать один препроцессор. Некоторые используют Sass, другие обычный CSS для редактирования файлов напрямую. Это усложняет работу над сайтом;
* Есть шанс потерять возможность использовать встроенный в браузер инспектор элементов;

Проанализировав данную информацию о приведенных в данном разделе инструментов для разработки. Можно сделать вывод, что благодаря JSX, Node.js и Sass можно написать работающий и читаемый код за короткий промежуток времени, что играет большую роль в разработке программного продукта.

## 1.4 Обзор используемых библиотек

Перед началом разработки каждого нового сайта необходимо убедиться, что выбранный JavaScript-фреймворк или сторонняя библиотека идеально подходит для проекта, для выполнения поставленных задач.

В этой главе будут кратко рассмотрены: React.js и WebRTC, т.к. автор использовал их в своем проекте.

### **1.**4**.1 React.js**

React.js - это JavaScript-библиотека с открытым исходным кодом для разработки пользовательских интерфейсов.

React разрабатывается и поддерживается Facebook, Instagram и сообществом отдельных разработчиков и корпораций.

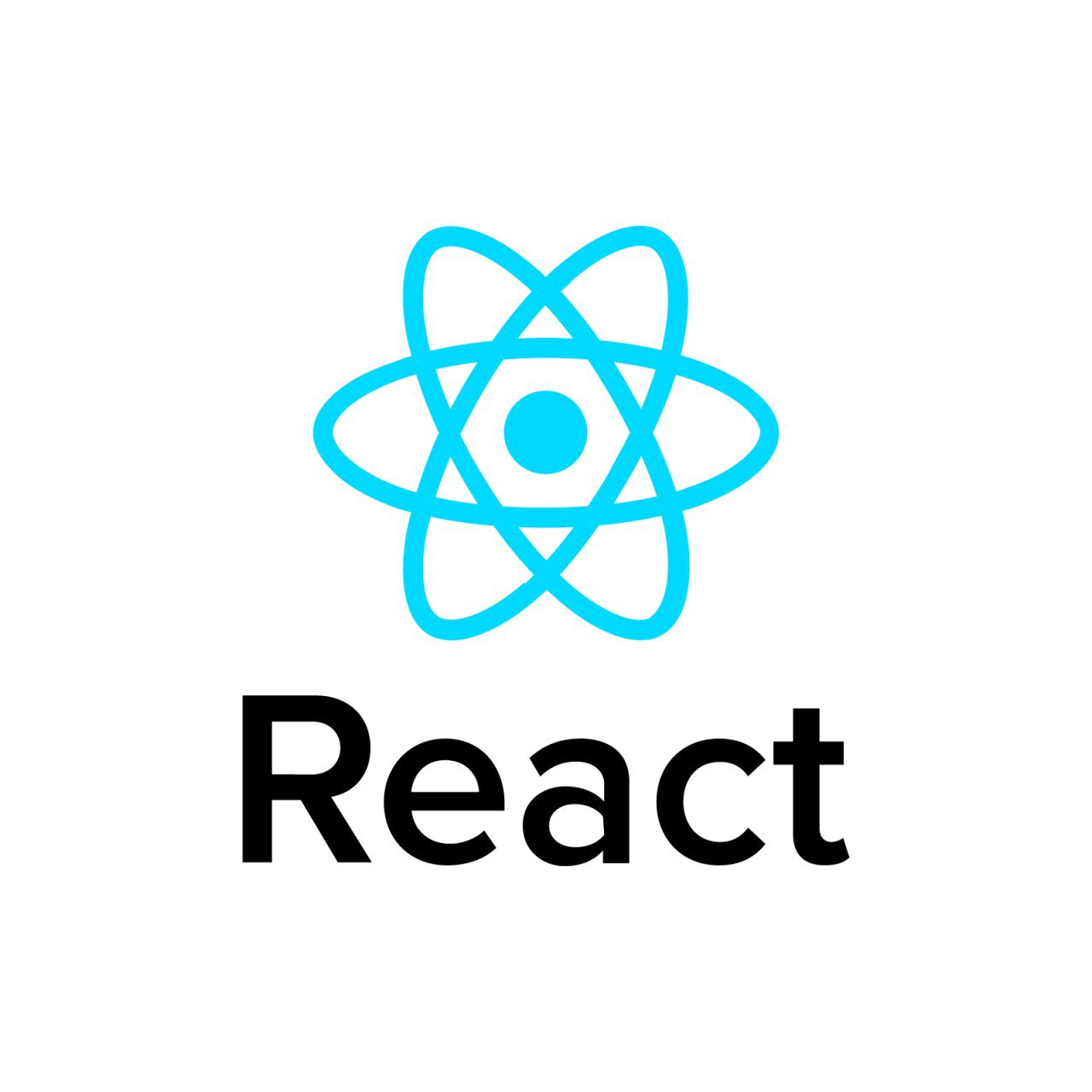


Рисунок 9 - Логотип React.js

Когда React рассматривается в контексте возможных альтернатив, то пригодится знание о его преимуществах и недостатках:

Преимущества:

* виртуальная объектная модель документа;
* повторное применение компонентов;
* нисходящий поток данных;
* огромное сообщество;
* браузерные инструменты React-разработчика;

Недостатки:

* запутанный синтаксис JSX для начинающих разработчиков;
* сложность поисковой оптимизации;
* фокусировка на пользовательском интерфейсе;

### **1**.4**.**2 **WebRTC**

### 

WebRTC (Web Real-Time Communication) — это API, который может использоваться веб-приложениями для видеочата, голосовых вызовов и обмена файлами P2P.



Рисунок 10 - логотип WebRTC

Преимущества:

* быстрая настройка;
* высокое качество связи;
* безопасное соединение;
* потоковая передача данных.

Недостатки:

* тяжело отладить;
* разные ОС ведут себя по разному с медиа-устройствами;
* есть большая несовместимость с многими браузерами.

Подводя итог этого раздела, можно сказать, что существуют и другие Java-script фреймворки для разработки пользовательского интерфейса, такими являются: Vue.js и Angular, а также многие другие библиотеки для разработки систем конференц связи.

Но выбор автора пал на React.js, т.к. большое кол-во компании на данный момент используют именно этот фреймворк, а WebRTC был использован, по той причине что благодаря ему удалось достаточно быстро создать P2P-соединение между клиентами, которое имеет достаточно высокое качество связи.

# 2 СПЕЦИАЛЬНАЯ ЧАСТЬ

## 

# ЗАКЛЮЧЕНИЕ

Курсовой проект посвящен веб-сайту для проведений видеоконференций, который помог автору усвоить новые и закрепить изученные ранее навыки.

Итогом данной работы является веб-сайт, созданный с помощью средств Microsoft Visual Studio Code, языка программирования JavaScript, библиотеки React.js и технологии предназначенной для создания прямых соединений(P2P) - WebRTC.

В процессе выполнения курсового проекта:

* проанализирована предметная область;
* спроектирована диаграмма вариантов использования;
* изготовлен макет веб-сайта под любое устройство средствами Figma;
* подобраны подходящие технологии и фреймворки для написания веб-сайта;
* разработан веб-сайт средствами Microsoft Visual Studio Code;
* протестирован данный проект в различных браузерах.

На разработку курсового проекта в соответствии с поставленными задачами, было потрачено: 14 дней (2 недели).

Дальнейшее развитие приложения предполагает: изменение сетевой топологии с Mesh на SFU путем создания отдельного сервера для соединения с клиентами и раздачи их медиа-данных другим участникам звонка, а также добавление текстового чата с помощью протокола связи WebSocket.

# СПИСОК ИСПОЛЬЗУЕМЫХ ИСТОЧНИКОВ

1. Справочная система WebRTC [Электронный ресурс]. URL: <https://webrtc.org> (дата обращения: 16.11.2022).
2. Справочная система MDN Web Docs [Электронный ресурс]. URL: <https://developer.mozilla.org/en-US> (дата обращения: 16.11.2022).
3. Справочная система React [Электронный ресурс]. URL: <https://reactjs.org/docs/getting-started.html> (дата обращения: 17.11.2022).
4. Справочная система Metanit [Электронный ресурс]. URL: <https://metanit.com> (дата обращения: 17.11.2022).
5. Статья на Хабр: И[спользование диаграммы вариантов использования UML при проектировании программного обеспечения](https://habr.com/ru/post/566218/) [Электронный ресурс]. URL: <https://habr.com/ru/post/566218> (дата обращения: 24.11.2022)
6. Справочная система Socket.IO [Электронный ресурс]. URL: <https://socket.io/docs/v4> (дата обращения: 19.11.2022).
7. Справочная система React Router [Электронный ресурс]. URL: <https://reactrouter.com/en/main/start/tutorial> (дата обращения: 20.11.2022).
8. Современный учебник JavaScript [Электронный ресурс]. URL: <https://learn.javascript.ru/?ysclid=lb21kyla6c813640185> (дата обращения: 23.11.2022).