**Министерство науки и высшего образования Российской Федерации**

**ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ**

**«НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ ИТМО»**

**(Университет ИТМО)**

Факультет **Прикладной информатики**

Направление подготовки **09.03.03 Прикладная информатика**

Образовательная программа **Мобильные и сетевые технологии**

**К У Р С О В О Й   П Р О Е К Т**

Тема: «Разработка веб-приложения для обмена видеопотоками (ВКС)»

Обучающийся: Останин Андрей Сергеевич, K3140

Санкт-Петербург 2024

СОДЕРЖАНИЕ

Титульный лист …………………………………………………………………..1

Содержание …………………………………………………………………….....2

1 Введение ……………………………………………………………………...…3

* 1. Актуальность рассматриваемой темы……………………………………….3
  2. Цель проекта…………………………………………………………………..4
  3. Задачи проекта………………………………………………………………...4

1. Работа над проектом…………………………………………………………..4
   1. Суть проекта…………………………………………………………………..4
   2. Процессы работы над проектом……………………………………………..4
   3. Суть задачи, поставленной передо мной……………………………………5
   4. Описание решения проблемы………………………………………………..6
   5. Анализ своей работы………………………………………………………….7
      1. Что получилось……………………………………………………………..7
      2. Основные трудности……………………………………………………….8
      3. Чему научился……………………………………………………………...8
   6. Взаимодействие с командой………………………………………………….9
   7. Взаимодействие с руководителем проекта………………………………….9
   8. Оценка работы руководителя……………………………………………….10
2. Заключение…………………………………………………………………...12
   1. Оценка выполнения всего проекта…………………………………………12
   2. Мой вклад в выполнение проекта…………………………………………..13

Список использованных источников…………………………………………...15

1. ВВЕДЕНИЕ
   1. Актуальность рассматриваемой темы

В современном мире обмен данными играет ключевую роль в жизни людей, деятельности компаний и функционировании государств. Ежедневно миллиарды людей обмениваются информацией, будь то личное общение, деловая переписка или международное сотрудничество. Важнейшим аспектом этого процесса является независимость от расстояния между участниками, что стало возможным благодаря изобретению и развитию интернета. Интернет охватывает практически всю планету, предоставляя людям возможность общаться, работать и делиться данными независимо от их местоположения.

Однако, несмотря на глобальность интернета, геополитические конфликты и экономические санкции могут создавать преграды для свободного обмена информацией. Сервисы, которые предоставляют возможность передавать данные, нередко оказываются заблокированными в одних странах или недоступными в других. Это создает серьёзные трудности для пользователей и компаний, которые зависят от этих сервисов для общения или ведения бизнеса.

Именно в таких условиях возникает необходимость создания отечественных платформ и приложений для обмена данными. Такие решения помогают обеспечить независимость от международных ограничений, гарантируя доступ к важным услугам внутри страны. Одним из примеров может быть разработка веб-приложений для обмена видеопотоками, которые позволяют людям связываться друг с другом по видеосвязи. Эта технология предоставляет десяткам миллионов людей возможность не только обмениваться данными, но и видеть друг друга в реальном времени, что особенно важно для личного общения, образовательных процессов и бизнес-коммуникации.

Создание подобных сервисов становится не просто технической задачей, но и важным шагом в обеспечении цифрового суверенитета, поддержки пользователей и развития национальной инфраструктуры. Такие проекты помогают решить актуальные задачи и предоставляют удобные инструменты для общения в современном мире.

* 1. Цель проекта

Целью проекта является создание веб-приложения для обмена видеопотоками (ВКС)

* 1. Задачи проекта

Задачи проекта следующие:

* сервис должен обеспечивать обмен видеопотоками между пользователями
* сервис должен иметь авторизацию пользователей с сохранением информации о никнеймах
* сервис должен обеспечивать возможность создания комнаты-конференции и возможности пригласить других

2 Работа над проектом

2.1 Суть проекта

Суть проекта заключается в том, чтобы создать веб-приложение для организации видеоконференций, которое позволяет пользователям обмениваться видеопотоками в режиме реального времени. Приложение должно обеспечивать стабильную связь, удобный интерфейс и базовые функции, такие как авторизация, создание комнат для конференций и управление подключениями участников.

2.2 Процессы работы над проектом

Процесс работы над проектом включает несколько ключевых этапов, которые мы выполняли в соответствии с планом, который утвердил руководитель проекта. Работа над проектом была организована таким образом, что к каждой онлайн встрече с руководителем нам необходимо было выполнить задание для обсуждения текущего прогресса и получения обратной связи. Каждый участник команды был ответственен за определённую часть работы, что позволило эффективно распределить задачи и минимизировать пересечения. Процесс разработки был разделён на этапы, которые завершались демонстрацией результатов и корректировкой дальнейших шагов на основе полученных замечаний.

2.3 Суть задачи, поставленной передо мной

Мне поставили задачу написать серверную часть проекта на Node.js с использованием фреймворка Express.js. Основной целью было создание API для обработки запросов от фронтенда, обеспечивающего корректное взаимодействие между клиентской частью приложения и сервером.

Для достижения этой цели мне нужно было выполнить несколько ключевых задач. Во-первых, инициализировать серверное приложение, настроив его базовые параметры, такие как маршрутизация и обработка HTTP-запросов. Во-вторых, разработать сигнальный сервер для поддержки видеосвязи. Этот компонент отвечал за передачу сигнальных данных между пользователями через протокол WebRTC, включая такие сообщения, как offer, answer и ICE candidates. В-третьих, мне предстояло реализовать серверные эндпоинты, которые обеспечивали бы авторизацию пользователей, создание комнат (лобби) для видеозвонков и управление их сессиями.

Эти задачи требуют не только разработки серверного кода, но и взаимодействия с другими компонентами приложения, такими как база данных и клиентская часть, а также учёт особенностей работы с WebSocket для реального времени.

2.4 Описание решения проблемы

Для решения этой задачи я начал с инициализации серверного приложения на Node.js с использованием фреймворка Express.js. Я настроил базовую структуру приложения, создав необходимые файлы и директории для организации кода. После этого был реализован сигнальный сервер, который обеспечивал обмен сообщениями между клиентами для установления соединений через WebRTC. Сигнальный сервер использует WebSocket для отправки и получения сообщений в реальном времени, что позволяет синхронизировать действия пользователей при обмене видеопотоками.

Одним из важнейших шагов было создание серверных эндпоинтов для работы с различными аспектами веб-приложения. Я создал RESTful API, которое включало следующие маршруты:

* **аутентификация пользователей**: для регистрации и авторизации был создан набор эндпоинтов, позволяющих пользователю зарегистрироваться, войти в систему и поддерживать активную сессию.
* **управление сессиями видеосвязи**: для организации видеоконференций был реализован механизм создания, начала и завершения сессий видеосвязи. Эндпоинты обрабатывали запросы на подключение пользователей к сессиям, а также управление состоянием этих сессий.
* **интеграция с WebRTC**: для поддержки обмена видеопотоками и обмена сигналами между клиентами были разработаны маршруты, позволяющие отправлять и получать сигнальные сообщения в процессе установки соединения.
* **взаимодействие с базой данных**: были настроены эндпоинты для работы с базой данных, чтобы обеспечить хранение информации о пользователях и сессиях, включая создание записей, обновление и удаление данных.

Для всех эндпоинтов была учтена асинхронность операций, чтобы минимизировать время отклика и обеспечить стабильную работу сервера при высокой нагрузке. Также было принято решение об использовании middleware для обработки ошибок и выполнения дополнительных проверок на безопасность и валидацию данных.

Таким образом, задача была решена путём реализации комплексного API для взаимодействия фронтенда и бэкенда, обеспечив стабильное и безопасное функционирование веб-приложения для обмена видеопотоками.

2.5 Анализ своей работы

Проанализирую свою работу, рассматривая, что удалось реализовать успешно, а что, возможно, не получилось достичь в полном объёме.

2.5.1 Что получилось

В рамках проекта удалось успешно реализовать ключевые элементы серверной части. Одной из важных функций стала авторизация, которая позволяет идентифицировать пользователей и обеспечивать безопасный доступ к приложению. Это решение обеспечивает изолированность информации и стабильность работы сессий для каждого пользователя.

Также был разработан сигнальный сервер, который играет важную роль в передаче информации о пирах (пользователях) в сессии. С помощью WebRTC, сервер обрабатывает сигнальные сообщения, такие как offer, answer и ICE candidates. Это позволяет устанавливать P2P-соединения между пользователями для обмена видеопотоками.

Ещё одним достижением стало создание функционала лобби. Реализована логика, позволяющая пользователям создавать комнаты (переговорные) для обмена информацией. Эти лобби обеспечивают изолированность сессий и автоматическое управление подключениями участников.

Для реализации этих задач были использованы современные инструменты и технологии. Фреймворк Express обеспечил удобную работу с маршрутами и middleware, а ORM Prisma упростила взаимодействие с базой данных, делая управление данными более понятным и структурированным. И главное WebSocket позволил реализовать сигнальный сервер для обмена данными в реальном времени.

2.5.2 Основные трудности

В процессе работы над серверной частью проекта мне пришлось столкнуться с рядом трудностей, которые потребовали дополнительных усилий. Одной из главных сложностей стало освоение новых технологий. Многие инструменты, такие как Prisma (ORM) для взаимодействия с базой данных и библиотека ws для работы с WebSocket, были для меня новыми. Это потребовало детального изучения документации, поиска примеров и проведения тестирования. Освоение этих технологий заняло значительное время, но в конечном итоге позволило успешно реализовать необходимые функции.

Ещё одной сложностью стала настройка WebSocket-сервера. Процесс был осложнён необходимостью работы с асинхронными подключениями пользователей к лобби. На начальном этапе возникали ошибки, связанные с некорректной обработкой подключений, что потребовало анализа проблем и оптимизации кода. Однако эти трудности были преодолены благодаря внимательному подходу к разработке и поддержке со стороны команды.

Несмотря на возникающие проблемы, каждая из них стала полезным опытом, который позволил мне не только завершить проект, но и значительно расширить свои знания и навыки. Трудности помогли мне лучше разобраться в тонкостях серверной разработки и подготовили к решению более сложных задач в будущем.

2.5.3 Чему научился

В ходе выполнения этого задания, я научился, в первую очередь, работать в команде, а именно в совместной разработке продукта в команде. Также, немало важное, было повышение компетентности в разработке серверной части проекта, я научился создавать peer-to-peer соединение, используя технологию WebRTC, создавать сигнальный сервер для того, чтобы соединять пользователей между собой, а также создавать серверные эндпоинты для маршрутизации запросов.

2.6 Взаимодействие с командой

В ходе выполнения задачи мне очень много помогал более опытный товарищ, чьи советы и рекомендации значительно ускоряли процесс разработки. Он помог мне разобраться в некоторых аспектах работы с сервером, а также в настройке взаимодействия с WebRTC. Благодаря его помощи мне удалось избежать множества ошибок и быстрее освоить тонкости реализации серверной части проекта.

В команде было принято прислушиваться к мнению каждого, и это сыграло ключевую роль в успешной реализации проекта. Например, мне не понравился первоначальный дизайн интерфейса, и я выразил своё мнение. Мы обсудили это с командой, и в результате было принято решение изменить некоторые элементы дизайна, что сделало приложение более удобным и привлекательным для пользователей.

Такое сотрудничество и открытость в команде значительно способствовали успешной разработке проекта и позволили каждому участнику внести свой вклад в его улучшение.

2.7 Взаимодействие с руководителем проекта

Руководитель проекта сыграл ключевую роль в моём успешном выполнении поставленных задач. Он оказывал важную поддержку на всех этапах работы, всегда находя время для детальных и понятных объяснений. Особенно это касалось вопросов, связанных с архитектурой проекта. Руководитель помог мне разобраться в структуре приложения, указал на важные аспекты, которые необходимо учитывать при разработке серверной части, и предложил оптимальные подходы к решению поставленных задач.

Его консультации были особенно ценны, когда я сталкивался с техническими сложностями. Например, он подробно объяснял, как организовать взаимодействие между клиентской и серверной частями приложения, как правильно обрабатывать асинхронные запросы и эффективно использовать возможности WebSocket. Эти рекомендации помогли мне избежать многих ошибок и значительно ускорили процесс разработки.

Кроме того, руководитель предоставлял обратную связь на каждом этапе работы, что позволило мне корректировать реализацию функций и улучшать её в соответствии с требованиями проекта. Его советы помогли мне не только разобраться с текущими задачами, но и глубже понять подходы к разработке серверных приложений. Такая поддержка не только обеспечила успех проекта, но и стала важным этапом моего профессионального роста.

2.8 Оценки работы руководителя

Руководитель проекта оказал значительную поддержку и сыграл ключевую роль в успешной реализации моего задания. Его помощь и подход к управлению проектом были весьма эффективными и способствовали высокому качеству работы команды. Руководитель всегда объяснял детали проекта и поставленные задачи достаточно подробно и понятно. Это позволило мне быстро понять, что от меня требуется, и эффективно двигаться вперёд в процессе разработки. Руководитель всегда проявлял интерес к прогрессу каждого участника проекта, что поддерживало командный дух и мотивировало работать на результат. Он умел дать конструктивную критику, не создавая чувства неуверенности, что способствовало открытости в общении и улучшению качества работы.

Работа руководителя была исключительно положительной. Он не только руководил проектом, но и обеспечивал хорошую коммуникацию и поддержку для каждого члена команды. Благодаря этому удалось эффективно решать задачи и избегать многих ошибок в процессе разработки.

3 ЗАКЛЮЧЕНИЕ

В ходе работы над проектом по созданию веб-приложения для обмена видеопотоками мне удалось успешно выполнить поставленные задачи и развить свои навыки в разработке серверной части.

Большую помощь в процессе разработки оказали мои коллеги. Более опытный товарищ помог мне разобраться в сложных аспектах реализации, а руководитель проекта дал чёткие указания и поддерживал на протяжении всей работы. В команде царила атмосфера сотрудничества, и все решения принимались на основе обсуждений, что значительно повысило качество выполнения проекта.

Этот опыт позволил мне не только глубже понять работу с Node.js и Express.js, но и научиться работать в команде, обсуждать и решать технические задачи, а также учиться на ошибках и замечаниях. Я уверен, что полученные знания и навыки будут полезны в будущем для решения более сложных задач.

Этот опыт послужит хорошей основой для будущих разработок и будет способствовать моему дальнейшему профессиональному росту.

* 1. Оценка выполнения всего проекта

Проект по созданию веб-приложения для обмена видеопотоками был успешно выполнен. Основные цели, поставленные перед командой, были достигнуты: реализован функционал авторизации пользователей, создан сигнальный сервер для передачи данных между клиентами, а также разработана система лобби для взаимодействия участников видеосвязи.

С технической точки зрения проект оказался сложным, так как потребовал освоения новых инструментов и технологий, таких как Prisma для работы с базой данных, WebSocket для обмена данными в реальном времени и WebRTC для обеспечения видеосвязи. Несмотря на это, команда смогла справиться с задачами благодаря совместной работе, поддержке друг друга и грамотному распределению обязанностей.

Сроки выполнения проекта были соблюдены, хотя настройка некоторых функций, например, сигнального сервера, потребовала больше времени, чем планировалось. Эти трудности были успешно преодолены за счёт гибкости и оперативного реагирования на возникающие проблемы.

Особенно стоит отметить слаженность команды. Все участники активно участвовали в обсуждениях, делились идеями и прислушивались к мнению друг друга. Это позволило не только улучшить качество технической реализации, но и учесть важные аспекты, связанные с пользовательским опытом. В результате проект оказался удобным, функциональным и стабильным.

Итоговая оценка выполнения проекта — высокая. Основные задачи были реализованы, и хотя остались аспекты, которые можно улучшить, например, производительность и тестирование, текущая версия приложения полностью соответствует поставленным целям. Проект стал ценным опытом, который поможет каждому участнику в дальнейшей профессиональной деятельности.

* 1. Мой вклад в выполнение проекта

Я реализовал основные функции серверной части, включая создание API, настройку сигнального сервера и разработку серверных эндпоинтов для взаимодействия с клиентской частью. API стало центральным элементом, который обеспечивает обработку запросов от фронтенда, передачу данных и управление сессиями пользователей. Сигнальный сервер был настроен для работы с протоколом WebRTC, что позволило осуществлять передачу сигнальных сообщений, необходимых для установления видеосвязи между клиентами. Серверные эндпоинты обеспечили функционал, связанный с авторизацией пользователей, созданием лобби и управлением сессиями.

Несмотря на определённые сложности, такие как необходимость настройки WebSocket-сервера для передачи данных в реальном времени, а также решение проблем с асинхронным присоединением пользователей к сессиям, все задачи были успешно выполнены. Эти этапы потребовали дополнительных усилий и времени, но результат оправдал ожидания.

СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

1. Express.js: <https://expressjs.com/>
2. Typescript: <https://typescriptlang.org>
3. Stackoverflow: <https://stackoverflow.com/>
4. WebRTC: <https://webrtc.org/getting-started/overview?hl=ru>
5. JavaScript: <https://developer.mozilla.org/en-US/docs/Web/JavaScript>