СОДЕРЖАНИЕ

[1. Введение 3](#_Toc148626161)

[1.1. Наименование программы 3](#_Toc148626162)

[1.2. Краткая характеристика области применения 3](#_Toc148626163)

[2. Основания для разработки 5](#_Toc148626164)

[3. Назначение разработки 6](#_Toc148626165)

[3.1. Функциональное назначение 6](#_Toc148626166)

[3.2. Эксплуатационное назначение 6](#_Toc148626167)

[4. Требования к программе 8](#_Toc148626168)

[4.1. Требования к функциональным характеристикам 8](#_Toc148626169)

[4.1.1. Требования к серверной части 8](#_Toc148626170)

[4.1.2. Требование к взаимодействию клиентской и серверной частей 9](#_Toc148626171)

[4.1.3. Требование к клиентской части 10](#_Toc148626172)

[4.2. Требования к надежности 11](#_Toc148626173)

[4.3. Требования к информационной и программной совместимости 12](#_Toc148626174)

[5. Требования к программной документации 13](#_Toc148626175)

[5.1. Состав программной документации 13](#_Toc148626176)

[6. Технико-экономические показатели 14](#_Toc148626177)

[7. Стадии и этапы разработки 15](#_Toc148626178)

[8. Порядок контроля и приемки 16](#_Toc148626179)

[9. Приложение (опционально) 17](#_Toc148626180)

# Глоссарий

* Виртуальные встречи представляют собой события, во время которых участники могут взаимодействовать между собой через интернет, не находясь физически в одном месте. Это может включать видеоконференции, аудиоконференции, чаты и другие способы виртуального общения.
* Веб-конференции — это события, в ходе которых участники собираются в интернете для проведения онлайн-встреч, обсуждения темы или выполнения определенных задач. Они часто включают в себя видео- и аудиокоммуникацию и обмен информацией через веб-браузеры или специализированные приложения.
* Платформа в контексте веб-приложения — это программное обеспечение или аппаратное обеспечение, которое обеспечивает основную инфраструктуру и функциональность веб-приложения. Она предоставляет средства для взаимодействия пользователя с приложением и обработки данных.
* Виртуальные классы — это онлайн-среда, предназначенная для образовательных целей, где учащиеся и преподаватели могут взаимодействовать и проводить уроки, лекции или обучение удаленно. Эти классы могут включать видеоконференции, чаты, обмен файлами и другие функции, подобные тем, которые предоставляют приложения, наподобие Microsoft Teams.
* Чат представляет собой средство коммуникации, которое позволяет пользователям обмениваться текстовыми сообщениями в реальном времени. Он может использоваться для общения один на один, в группах или командных чатах. Чаты могут также включать функциональность передачи файлов, изображений и других мультимедийных данных.

# Введение

## Наименование программы

SQUAD. Web-приложение для командного взаимодействия и совместной работы.

## Краткая характеристика области применения

SQUAD — это веб-приложение для командного взаимодействия и совместной работы. Оно предоставляет возможности для коммуникации и сотрудничества в рабочей и учебной среде, объединяя пользователей в области организации проектов, командных задач, обмена информацией и виртуальных встреч. SQUAD призвана упростить и улучшить совместную работу в офисе, удаленно или в учебных учреждениях.

Область применения SQUAD включает в себя:

* Командное взаимодействие: SQUAD предоставляет возможности для обмена сообщениями, аудио и видеозвонками, обмена файлами и документами, что делает его подходящим для команд и организаций, которые стремятся поддерживать связь и сотрудничество в режиме реального времени.
* Управление проектами и задачами: Пользователи могут создавать задачи, устанавливать сроки и отслеживать их выполнение, что делает SQUAD полезным инструментом для управления проектами и задачами.

Виртуальные встречи и веб-конференции: Платформа поддерживает организацию виртуальных встреч и веб-конференций с множеством участников, что идеально подходит для команд, которые работают удаленно или проводят обучающие мероприятия.

* Обучение и учебные заведения: SQUAD может использоваться в образовательных учреждениях для дистанционного обучения, а также для управления учебными материалами и взаимодействия между преподавателями и учениками.

Организация и обмен информацией: Платформа позволяет пользователям создавать и управлять каналами, обменяться информацией, документами и ресурсами, и, таким образом, способствует организации данных и коллаборации.

SQUAD предназначена для совместной работы, обучения, обмена информацией и коммуникации в различных контекстах и сферах деятельности, предоставляя средства для эффективного взаимодействия и достижения общих целей.

# Основания для разработки

Документ:

Издан:

Дата:

Номер:

Полное наименование: "Улучшение Процесса Образования и Взаимодействия в Образовательных Учреждениях через Разработку Программы SQUAD".

Тема разработки: "Разработка и Внедрение Веб-приложения для Командного Взаимодействия Преподавателей и Студентов в Образовательных Учреждениях (SQUAD)".

# Назначение разработки

## Функциональное назначение

Web-приложение SQUAD предоставляет следующие функциональные возможности:

* Виртуальные классы и уроки: Позволяет преподавателям организовывать виртуальные классы и уроки, проводить онлайн-лекции и вебинары.
* Обмен учебными материалами: Студенты и преподаватели могут обмениваться учебными материалами, документами и заданиями.
* Чат и общение в реальном времени: Предоставляет возможность студентам задавать вопросы и общаться с преподавателями в режиме реального времени через чат и видеоконференции.
* Управление расписанием и календарем: Помогает в создании и управлении расписанием занятий и календарем активностей.
* Оценивание и обратная связь: Предоставляет инструменты для оценивания учебных достижений студентов и предоставления им обратной связи.

## Эксплуатационное назначение

Программа SQUAD является центральным инструментом для образовательных учреждений и обеспечивает следующие аспекты ее эксплуатационного назначения:

* Оптимизация образовательного процесса: Помогает учреждениям облегчить процесс обучения и сотрудничества между студентами и преподавателями.
* Поддержка удаленного обучения: Обеспечивает возможность удаленного обучения, что особенно актуально в современных условиях.
* Управление учебными ресурсами: Позволяет хранить и управлять учебными материалами и ресурсами, что снижает нагрузку на бумажную документацию.
* Улучшение взаимодействия: Содействует более эффективному взаимодействию между студентами и преподавателями, а также укрепляет связь между ними.
* Повышение качества образования: Позволяет более эффективно учиться и преподавать, что способствует улучшению качества образования в целом.

# Требования к программе

## Требования к функциональным характеристикам

Программа состоит из двух основных компонент: клиентской и серверной частей, между которыми должно быть налажено взаимодействие.

### Требования к серверной части

* Авторизация и аутентификация:

Сервер должен обеспечивать возможность регистрации и входа пользователей. Должен поддерживать аутентификацию с использованием логина/пароля и/или одноразовых кодов (например, двухфакторную аутентификацию). Должен обеспечивать сессионное управление и управление доступом.

* Работа с файлами:

Поддержка загрузки и скачивания учебных материалов, документов и файлов. Возможность создания, редактирования и удаления файлов и папок. Хранение файлов с учетом безопасности и резервного копирования.

* Чат и мгновенные сообщения:

Реализация чата для обмена мгновенными сообщениями между пользователями. Поддержка групповых чатов и приватных сообщений. Отправка медиафайлов и документов в чате.

* Аудио и видео звонки:

Возможность проведения аудио и видеоконференций между пользователями. Опции для управления звонками, включая возможность подключения и отключения аудио и видео.

* Работа с базой данных:

Сервер должен взаимодействовать с базой данных для хранения информации о пользователях, уроках, материалах и других данных. Эффективное выполнение запросов к базе данных для обеспечения быстрой загрузки и отображения данных. Использование нескольких типов баз данных для долгого и малого хранения.

* Валидация и обработка ошибок:

Проверка входных данных на корректность и безопасность. Обработка ошибок и отображение информативных сообщений об ошибках для пользователей. Журналирование событий и ошибок для отслеживания и диагностики.

### Требование к взаимодействию клиентской и серверной частей

* Протоколы коммуникации: Клиентская и серверная части должны использовать безопасные и эффективные протоколы для обмена данными, такие как HTTPS для веб-приложений. Все запросы и ответы должны быть зашифрованы.
* API (Интерфейс программирования приложения): Серверная часть должна предоставить API для клиентской части, определяющее доступные функции и методы взаимодействия. API должно быть документировано и обеспечивать стабильность интерфейса.
* Аутентификация и авторизация: Сервер должен проверить подлинность каждого запроса от клиента и убедиться в правильности авторизации пользователя. Должны применяться стандарты безопасности, такие как токены авторизации.
* Обработка ошибок: Клиентская и серверная части должны обеспечивать обработку ошибок, включая корректную передачу информации об ошибках между ними, чтобы обеспечить информативные сообщения для пользователя.
* Обмен данными: Данные, передаваемые между клиентской и серверной частями, должны соответствовать установленным стандартам и форматам. Это важно для обеспечения корректной передачи и интерпретации данных.
* Сессии и состояние: Если требуется, сервер и клиент должны поддерживать сессии и состояние (например, сохранение информации о пользователе, его действиях и настройках).
* Скорость и производительность: Взаимодействие должно быть оптимизировано для достижения высокой скорости и производительности приложения.
* Безопасность: Должны применяться меры безопасности для защиты от взлома, включая защиту от SQL-инъекций, CSRF (межсайтовая подделка запроса), XSS (межсайтовый скриптинг) и других атак.
* Тестирование и отладка: Должны быть предусмотрены средства для тестирования взаимодействия между клиентской и серверной частями, а также для отладки и мониторинга.
* Документация: Взаимодействие между клиентской и серверной частями, включая API, должно быть документировано, чтобы облегчить совместную работу разработчиков и поддержку приложения.

### Требование к клиентской части

* Пользовательский интерфейс (UI): Интуитивно понятный и привлекательный интерфейс для преподавателей и студентов. Легкость в навигации и доступ к основным функциям.
* Работа в различных браузерах и платформах: Поддержка различных веб-браузеров, включая Chrome, Firefox, Safari и Edge. Доступность на различных операционных системах, включая Windows, macOS и мобильных устройствах.
* Аутентификация и авторизация: Возможность регистрации и входа для пользователей. Обеспечение безопасной аутентификации с использованием паролей и, при необходимости, двухфакторной аутентификации.
* Взаимодействие с сервером: Отправка запросов к серверу для получения данных и выполнения операций. Обработка и отображение данных, полученных от сервера.
* Чат и мгновенные сообщения: Организация чата для обмена мгновенными сообщениями между пользователями. Возможность управления чатами и отправки медиафайлов.
* Аудио и видео звонки: Возможность начать и присоединиться к аудио и видеоконференциям. Управление звонками и мультимедийными параметрами.
* Работа с файлами: Загрузка, скачивание и управление учебными материалами, документами и файлами. Создание, редактирование и удаление файлов и папок.
* Локализация и поддержка языков: Поддержка различных языков и возможность переключения языков интерфейса.
* Скорость и производительность: Оптимизация производительности приложения для минимизации задержек и ускорения загрузки страниц.
* Безопасность: Защита от уязвимостей, таких как XSS, CSRF и других атак. Шифрование данных, передаваемых между клиентом и сервером.
* Сессии и сохранение состояния: Возможность сохранения сессий и состояния пользователя для удобства входа и навигации.
* Документация и поддержка: Обеспечение документации для пользователей и, при необходимости, технической поддержки.
* Настройка темы приложения: Поддержка различных цветовых тем приложения и возможность переключения между ними

## Требования к надежности

Программа должна обеспечивать стабильную и непрерывную работу приложения, минимизируя сбои и выход из строя. Если происходят непредвиденные ошибки, на сервере должны записаться логи, а на клиенте появиться заглушка. Проводить валидацию и фильтрацию входных данных для предотвращения атак, таких как SQL-инъекции и межсайтовый скриптинг (XSS). Обеспечивать корректное форматирование и безопасное отображение выходных данных. Обеспечивать регулярные резервные копии данных и возможность их восстановления. Вести журнал событий для регистрации ошибок и активности пользователей для последующего анализа и диагностики. Обеспечивать сохранность и целостность данных пользователя с помощью мер безопасности, таких как шифрование данных в покое и в передаче. Определить целевое время восстановления приложения после сбоя и разработать план восстановления данных и функциональности (сутки). Разработать процессы обслуживания и обновления программы с минимальными простоями и рисками для пользователей. Провести тестирование на отказ для определения поведения системы при сбоях и проверки работоспособности механизмов восстановления

## Требования к информационной и программной совместимости

* Язык программирования: Серверная часть будет разработана с использованием JavaScript и TypeScript на платформе Node.js с использованием фреймворка Express.
* Среда разработки: Для разработки серверной части вы можете использовать популярные интегрированные среды разработки (IDE) такие как Visual Studio Code или WebStorm.

Для разработки клиентской части, React и Next.js, также рекомендуется использовать Visual Studio Code или любую другую IDE, с которой вы комфортно работаете.

* Операционная система: Node.js, Express, и большинство баз данных, такие как Redis, MongoDB и PostgreSQL, поддерживаются на различных операционных системах, включая Windows, macOS и Linux. Вы можете выбрать ОС, которая наиболее удобна для вас.
* Библиотеки и технологии: Серверная часть: Node.js, TypeScript, Express.js, Redis, MongoDB, PostgreSQL, Git, Docker. Клиентская часть: React, Next.js, TypeScript, Redux, Storybook.

# Требования к программной документации

## Состав программной документации

* Техническое задание (ГОСТ 19.201-78);
* Пояснительная записка (ГОСТ 19.404- 79);
* Руководство оператора (ГОСТ 19.505- 79);
* Программа и методика испытаний (ГОСТ 19.301-79);
* Текст программы. (ГОСТ 19.401-78);

# Технико-экономические показатели

# Стадии и этапы разработки

# Порядок контроля и приемки

# Приложение (опционально)