

РАЗРАБОТКА ВЕБ-МОДУЛЯ ОБМЕНА СООБЩЕНИЯМИ С ИСПОЛЬЗОВАНИЕМ ТЕХНОЛОГИЙ РЕАЛЬНОГО ВРЕМЕНИ

В работе рассматривается разработка программного модуля обмена сообщениями, предназначенного для организации взаимодействия пользователей в режиме реального времени. Описаны основные подходы к построению архитектуры веб-мессенджеров, выбран стек технологий и реализован прототип системы с поддержкой текстовых сообщений, а также голосовой и видеосвязи.

Введение

Современные цифровые коммуникации играют ключевую роль в организации взаимодействия пользователей как в корпоративной среде, так и в повседневной жизни. Рост популярности веб-приложений и необходимость обеспечения мгновенного обмена данными привели к активному развитию мессенджеров и систем реального времени. Такие системы позволяют значительно повысить эффективность взаимодействия, сократить временные затраты на передачу информации и обеспечить удобство совместной работы.

Разработка эффективных программных решений в данной области требует использования современных технологий и подходов к построению масштабируемых и отказоустойчивых систем. Особую роль играет выбор архитектуры и технологий передачи данных, обеспечивающих минимальные задержки и стабильную работу при большом количестве пользователей. В связи с этим актуальной задачей является создание веб-модуля обмена сообщениями, способного обеспечить высокую производительность и гибкость в условиях динамично изменяющихся требований.

I. Анализ предметной области

Системы обмена сообщениями представляют собой программные решения, обеспечивающие передачу данных между пользователями в режиме реального времени. Они находят применение в различных сферах: от корпоративных систем взаимодействия до социальных платформ и образовательных сервисов.

К основным функциям таких систем относятся:

- обмен текстовыми сообщениями;
- организация голосовой и видеосвязи;
- управление каналами и пользовательскими группами;
- хранение и поиск сообщений.

Современные мессенджеры предоставляют широкий спектр возможностей, включая передачу файлов, интеграцию с внешними сервисами и поддержку различных типов уведомлений. Однако с увеличением функциональности возрастает сложность систем, что может затруднять их адаптацию под конкретные задачи и увеличивать требования к ресурсам.

Существующие решения, такие как Discord и Telegram, предоставляют развитый функционал и удобные интерфейсы взаимодействия. Тем не менее, они не всегда подходят для использования в специализированных системах, где требуется гибкость настройки, контроль над данными и возможность ин-

теграции с другими компонентами информационной инфраструктуры. Это обуславливает необходимость разработки собственных программных модулей обмена сообщениями.

II. Проектирование системы

Разрабатываемый программный модуль представляет собой веб-приложение, построенное на клиент-серверной архитектуре. Такой подход обеспечивает разделение ответственности между компонентами системы и позволяет эффективно масштабировать приложение.

Клиентская часть реализована с использованием современного фреймворка, обеспечивающего динамическое обновление интерфейса и удобство взаимодействия пользователя с системой. Серверная часть отвечает за обработку запросов, управление бизнес-логикой и взаимодействие с базой данных.

Для реализации обмена сообщениями в реальном времени используется технология WebSocket, обеспечивающая постоянное двустороннее соединение между клиентом и сервером. Это позволяет существенно снизить задержки при передаче данных по сравнению с традиционными HTTP-запросами.

Для организации голосовой и видеосвязи применяется технология WebRTC, позволяющая передавать мультимедийные данные напрямую между пользователями без необходимости использования промежуточного сервера. Это снижает нагрузку на серверную часть и повышает эффективность системы.

Система включает следующие основные компоненты:

1. модуль аутентификации и авторизации пользователей;
2. модуль обмена сообщениями;
3. модуль управления каналами и чатами;
4. модуль мультимедийного взаимодействия.

Дополнительно реализуется механизм хранения сообщений и пользовательских данных с использованием реляционной базы данных, что обеспечивает целостность информации и возможность её последующего анализа.

III. Результаты реализации

В ходе работы был разработан прототип веб-мессенджера, реализующий базовый функционал систем обмена сообщениями. При разработке использовались современные инструменты и подходы, позволяющие обеспечить высокую производительность и удобство дальнейшего расширения системы.

Приложение обеспечивает:

- регистрацию и авторизацию пользователей;
- обмен сообщениями в реальном времени;
- создание каналов и групп;
- поддержку голосовой и видеосвязи.

Реализованная система демонстрирует стабильную работу при одновременном подключении нескольких пользователей и обеспечивает низкие задержки при передаче сообщений. Архитектура приложения позволяет масштабировать систему за счёт добавления новых серверных узлов и расширения функциональности.

Заключение

В результате выполненной работы разработан программный модуль обмена сообщениями, соответствующий современным требованиям к системам реального времени. Использование технологий WebSocket и WebRTC позволило реализовать эффективный механизм передачи данных и обеспечить поддержку мультимедийного взаимодействия.

Полученные результаты подтверждают целесообразность использования выбранных технологий

и архитектурных решений. Разработанный модуль может быть использован в качестве основы для создания полнофункциональных коммуникационных платформ, а также интегрирован в существующие информационные системы. Перспективы дальнейшего развития включают расширение функциональности, повышение уровня безопасности и оптимизацию производительности системы.

Список литературы

1. WebSocket Protocol Specification (RFC 6455) [Электронный ресурс]. – Режим доступа: <https://datatracker.ietf.org/doc/html/rfc6455>.
2. WebRTC 1.0: Real-Time Communication Between Browsers [Электронный ресурс]. – Режим доступа: <https://www.w3.org/TR/webrtc/>.
3. NestJS Documentation [Электронный ресурс]. – Режим доступа: <https://docs.nestjs.com>.
4. Next.js Documentation [Электронный ресурс]. – Режим доступа: <https://nextjs.org/docs>.
5. PostgreSQL Documentation [Электронный ресурс]. – Режим доступа: <https://www.postgresql.org/docs/>.

Воробей Артем Святославович, студент ФИТиУ БГУИР, mramor91932765@gmail.com.

Научный руководитель: Шилин Леонид Юрьевич, доктор технических наук, профессор, leonidchilin@gmail.com.