1. Компоненты и общая архитектура

* Микросервис "Website": отвечает за управление пользователями и чат-комнатами. Здесь пользователи могут зарегистрироваться, войти в систему, создать чат-комнату, найти комнаты и отправить сообщения.
* Микросервис WebSocket "Chat": обрабатывает соединения WebSocket для чат-комнат, передавая сообщения между пользователями в одной и той же комнате.
* База данных (PostgreSQL): хранит данные о пользователях, чат-комнатах и сообщениях.
* Nginx: выступает в роли обратного прокси, перенаправляя HTTP-запросы к микросервису "Website" и WebSocket-соединения к микросервису "Chat".

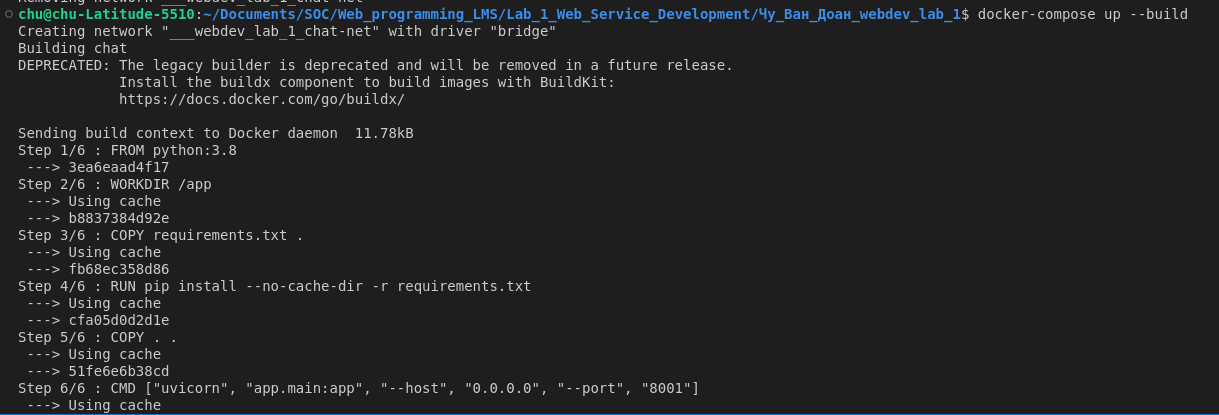
1. Принципы работы каждого компонента
   1. Микросервис "Website"

* Управление пользователями: пользователи могут создать новый аккаунт, указав email, имя пользователя и пароль. При регистрации система сохраняет эти данные в таблице users в зашифрованном виде.
* Авторизация: при входе пользователя система проверяет учетные данные и, если они верны, создает JWT-токен для дальнейшей аутентификации. Этот токен сохраняется в cookies браузера для использования в последующих запросах.
* Создание комнаты: авторизованный пользователь может создать новую комнату. Информация о комнате сохраняется в таблице chat\_rooms вместе с идентификатором создателя комнаты.
* Поиск комнат: пользователь может искать чат-комнаты по имени, чтобы присоединиться к уже существующим беседам.
* Удаление комнаты: только владелец комнаты может ее удалить. При удалении комнаты также удаляются все сообщения в ней.
  1. Микросервис WebSocket "Chat"
* Управление WebSocket-соединениями: когда пользователь входит в чат-комнату, устанавливается WebSocket-соединение с микросервисом по пути /ws/{room\_name}, где room\_name — это имя комнаты.
* Отправка и получение сообщений: после успешного подключения пользователь может отправлять сообщения другим участникам. Сообщение передается микросервису "Chat", который затем передает его всем пользователям, подключенным к этой комнате.
* Хранение сообщений: каждое новое сообщение сохраняется в таблице messages с содержимым, именем отправителя и временем отправки.
  1. База данных (PostgreSQL)
* Таблица users: хранит информацию о пользователях, включая id, email, username, и hashed\_password.
* Таблица chat\_rooms: хранит информацию о чат-комнатах, включая id, name (имя комнаты) и owner\_id (идентификатор создателя).
* Таблица messages: хранит сообщения, включая id, content (текст сообщения), room\_id (идентификатор комнаты), username (имя отправителя), и timestamp (время отправки).
  1. Nginx
* Перенаправление HTTP-запросов: HTTP-запросы пользователей перенаправляются на микросервис "Website", включая запросы на вход в систему, регистрацию, создание и поиск чат-комнат.
* Перенаправление WebSocket-соединений: WebSocket-соединения перенаправляются на микросервис "Chat", обеспечивая обмен сообщениями в реальном времени между пользователями в комнате.

1. Процесс работы системы

* Пользователь заходит на главную страницу: ему предлагается зарегистрироваться или войти в систему.
* Создание комнаты: после входа в систему пользователь может создать новую комнату или найти существующие комнаты для присоединения.
* Вход в комнату: при входе в комнату устанавливается WebSocket-соединение, позволяя пользователю отправлять и получать сообщения в режиме реального времени.
* Отправка и получение сообщений: сообщения передаются всем пользователям в комнате через WebSocket, а также сохраняются в базе данных для возможности последующего просмотра.
* Завершение сеанса: при отключении пользователя от комнаты соединение WebSocket закрывается.

1. Тестирование функций



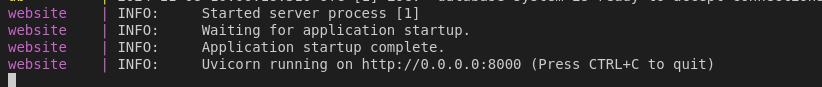


Рисунок 1: построение и запуск сервисов

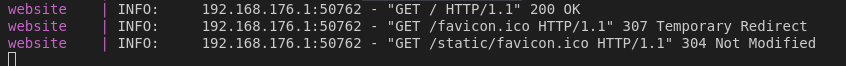
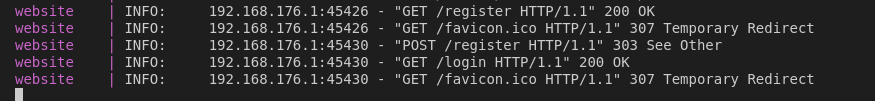




Рисунок 2: интерфейс при открытии веб-страницы



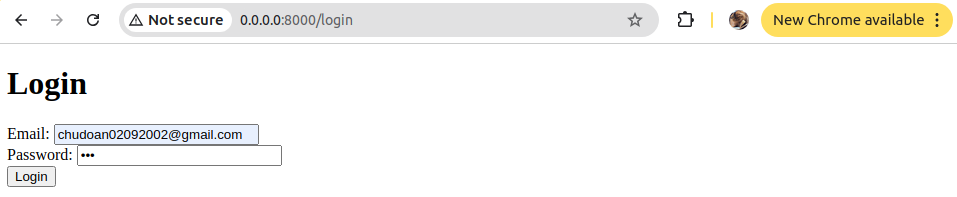


Рисунок 3: Регистрация пользователя прошла успешно, переходите к входу.



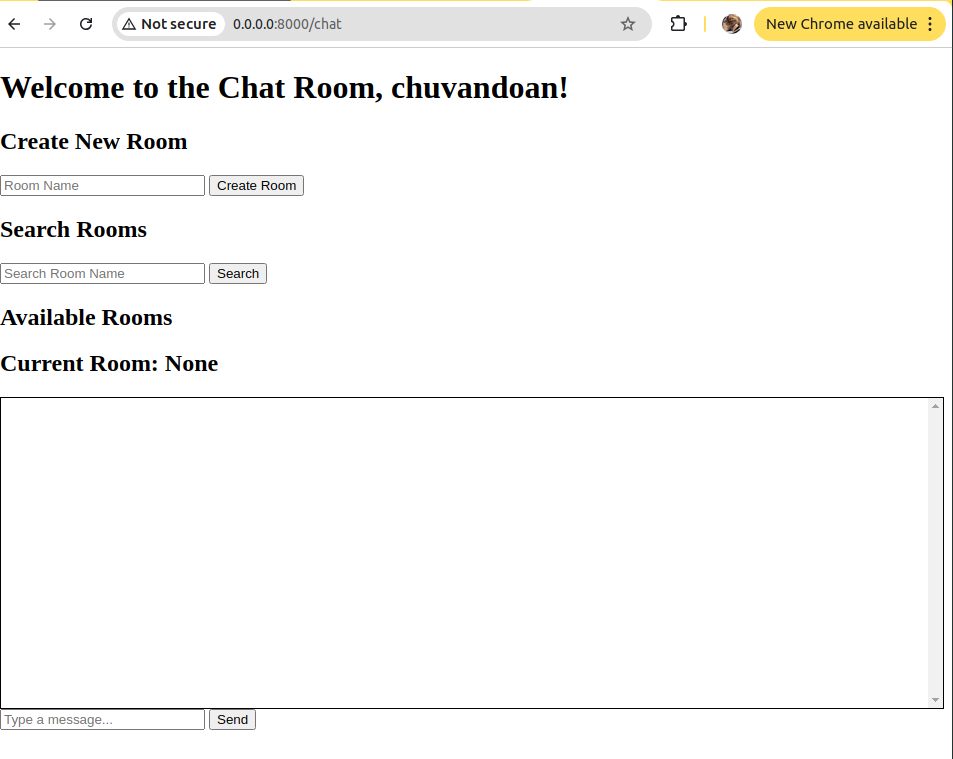


Рисунок 4: Вход выполнен успешно, пользователь перенаправлен на интерфейс использования веб-сайта



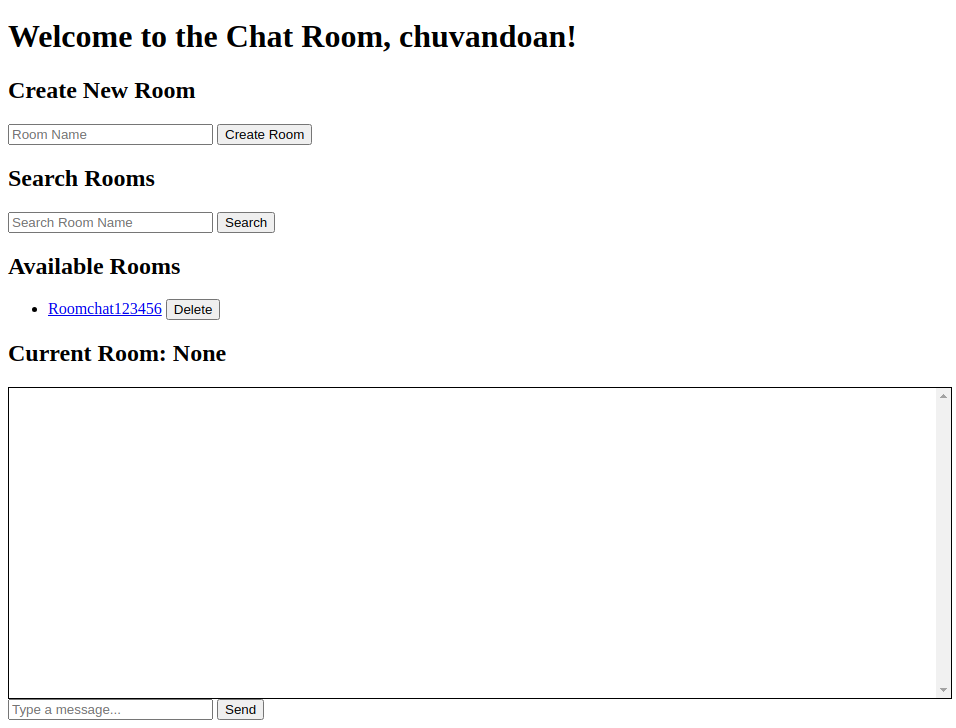


Рисунок 5: создание чата



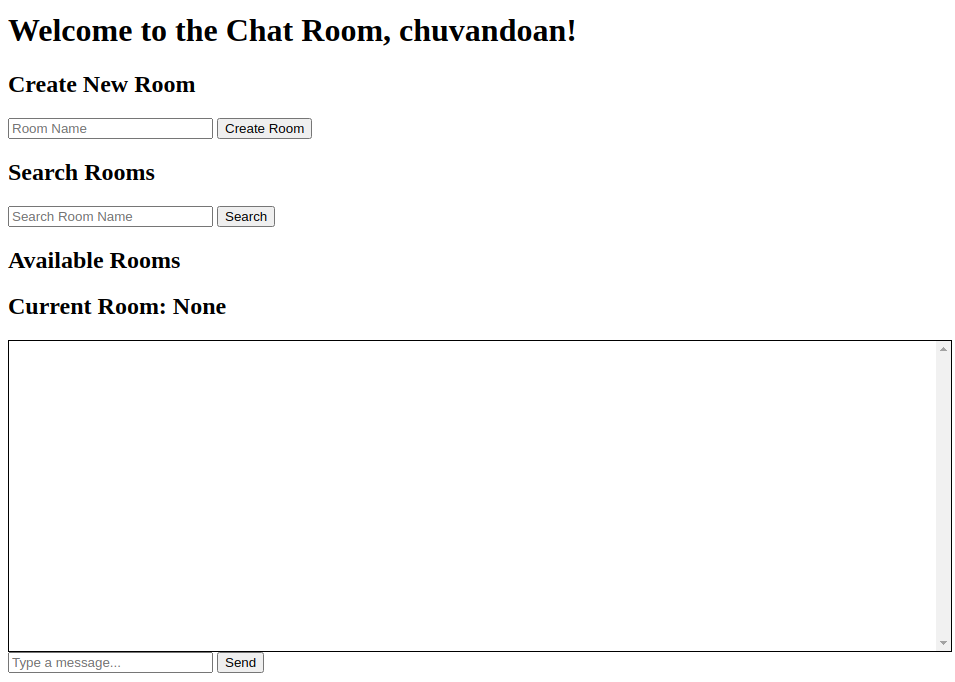
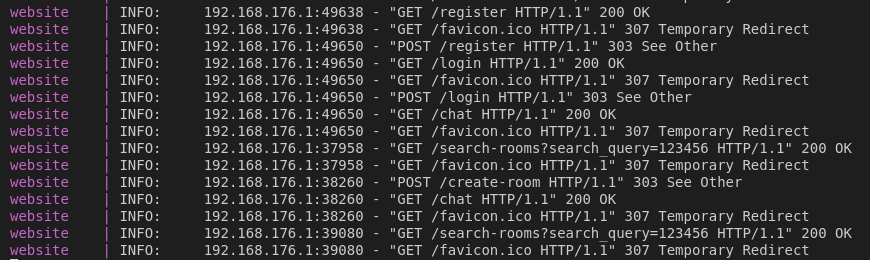


Рисунок 6: удаление чата



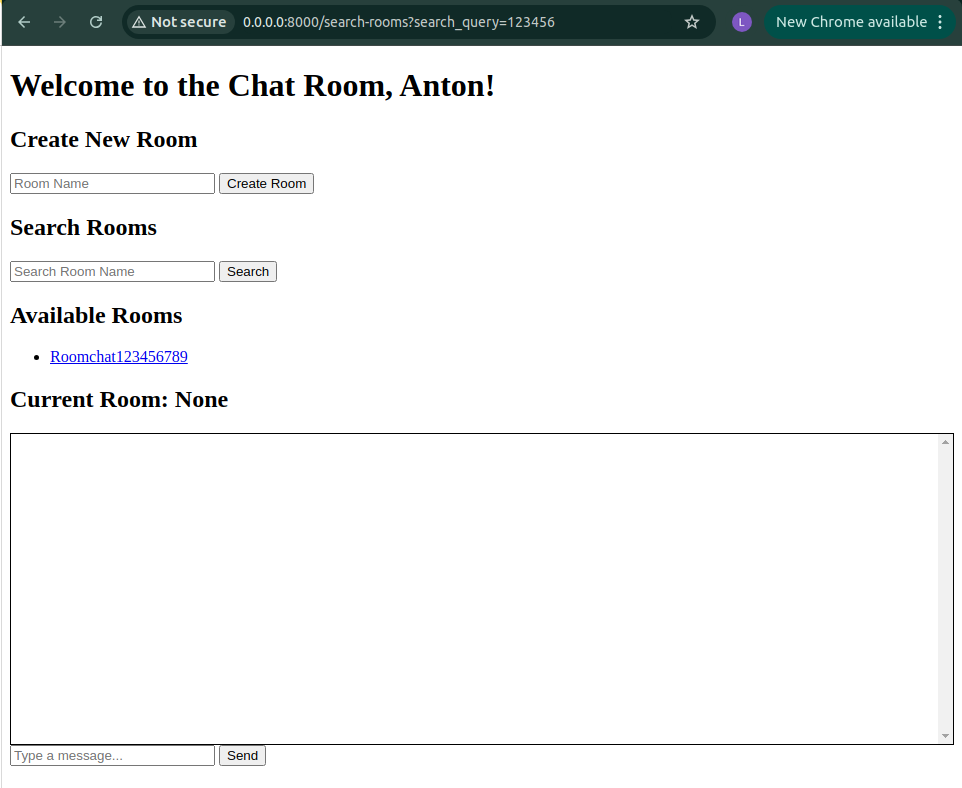
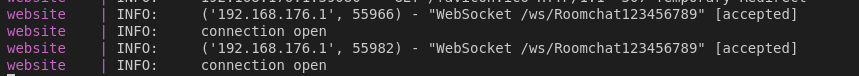


Рисунок 7: Создайте нового пользователя и попробуйте найти название чата.



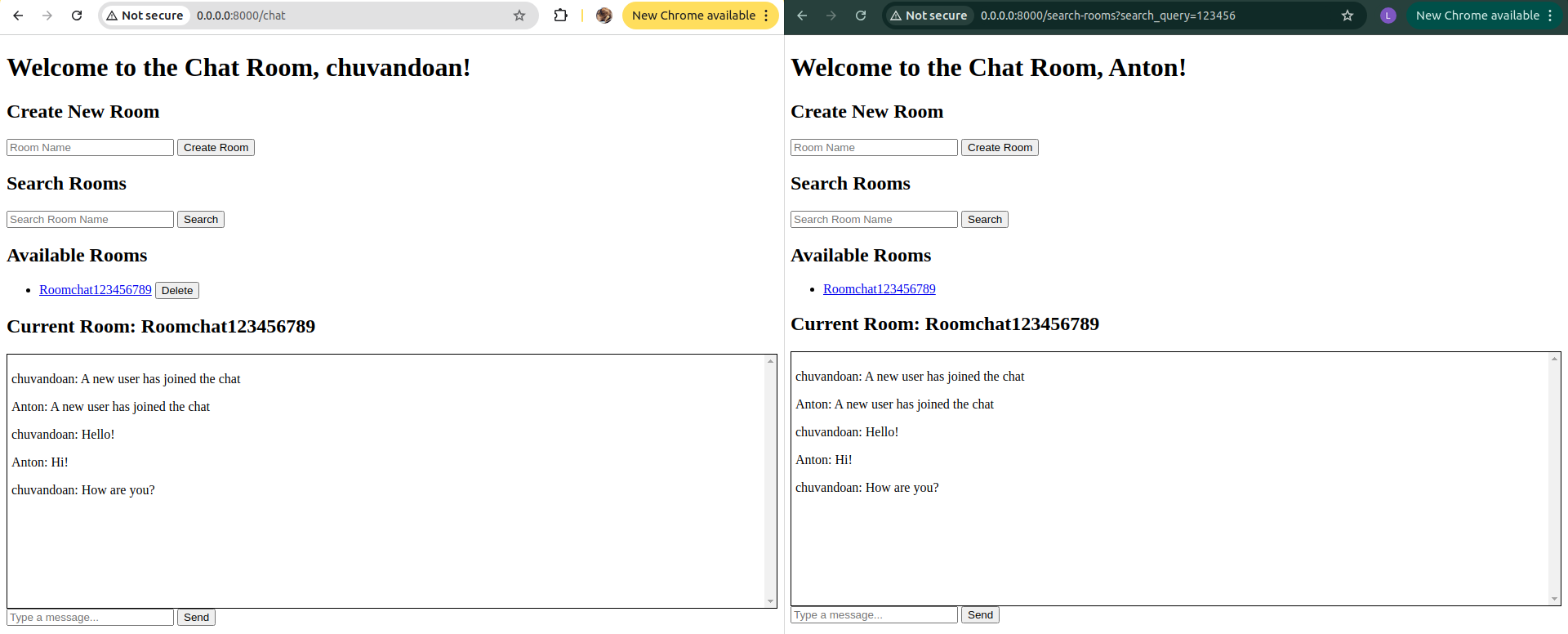


Рисунок 8: два пользователя присоединяются к чату и обмениваются сообщениями