Техническое задание для проекта «Чат с использованием WebSocket»

Проект: Чат с использованием WebSocket

Цель: разработать простое приложение для чата с использованием WebSocket для обмена сообщениями в реальном времени.

Архитектура: есть центральный модуль (сервер), к которому обращаются клиенты для регистрации/авторизации/отправке сообщений другим пользователям.

Основные функциональные возможности:

* регистрация нового пользователя;
* авторизация зарегистрированного пользователя;
* выход из профиля и закрытие сессии;
* возможность отправить сообщение по Login собеседника в реальном времени с отображением сообщений обоих собеседников;
* возможность переключаться между имеющимися диалогами;
* логирование информации о регистрациях, авторизациях и выходах из системы в таблицу AuthLog;
* сохранение истории переписки в таблицу ChatLog;
* индикация непрочитанного сообщения.

# Модель данных

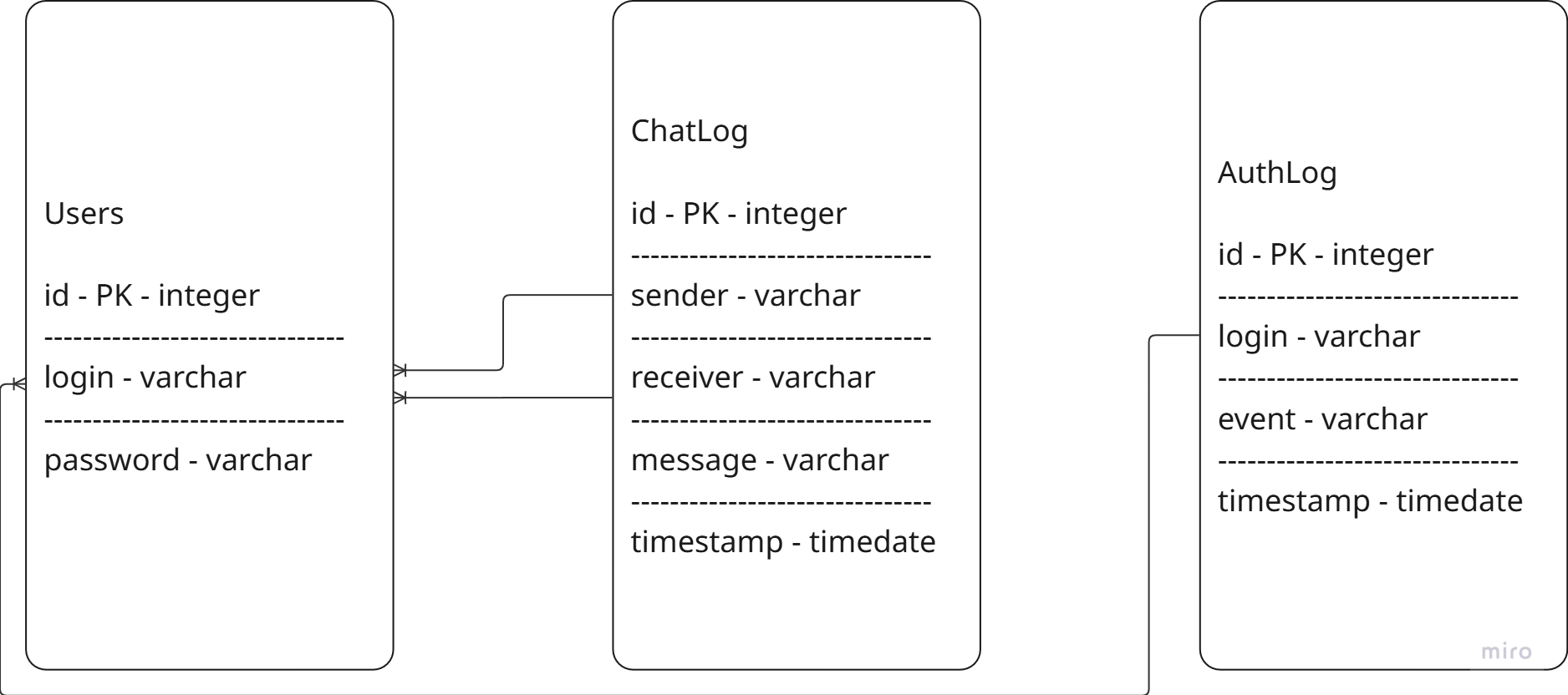


Таблица Users хранит данные о пользователях: логин и пароль.

Таблица AuthLog хранит данные о входах пользователей: ссылку на пользователя, тип (регистрация, авторизация, выход из системы), дату и время.

Таблица ChatLog хранит данные истории сообщений: отправителя сообщения, получателя, сообщение и дату со временем.

## Use cases

## Регистрация

|  |  |
| --- | --- |
| Предусловие | Сервер доступен |
| Основной сценарий | 1. Пользователь заходит в программу, вводит логин и пароль.  2. Пользователь нажимает на «Зарегистрироваться»  3. Идет проверка на отсутствие такого же логина.  4. Приходит ответ, что все данные корректны, пользователю отображается функционал чата (происходит авторизация с введенными данными), сохраняется две записи в таблицу AuthLog: о регистрации с типом «Регистрация» и авторизации с типом «Авторизация» |
| Альтернативный сценарий 1 | 4а. Приходит ответ, что данный логин существует  5. Отображается сообщение «Такой пользователь уже существует» |

## Авторизация

|  |  |
| --- | --- |
| Предусловие | Сервер доступен |
| Основной сценарий | 1. Пользователь заходит в программу, вводит логин и пароль.  2. Пользователь нажимает на «Войти»  3. Идет проверка на корректность введенных данных (существует ли введенный логин, корректно ли введен пароль для данного логина)  4. Приходит ответ, что все данные корректны, пользователю открывается функционал чата (происходит авторизация, данные об авторизации сохраняются в таблицу AuthLog с типом «Авторизация») |
| Альтернативный сценарий 1 | 4а. Приходит ответ, что данный логин не существует или введенный пароль некорректен  5. Отображается сообщение «Некорректные введенные данные» |

## Выход из профиля и закрытие сессии

|  |  |
| --- | --- |
| Предусловие | Сервер доступен, пользователь авторизован |
| Основной сценарий | 1. Пользователь нажимает на «Выйти»  2. Происходит выход из системы, пользователю отображается страница авторизации, запись о выходе из системы сохраняется в таблицу AuthLog |

## Отправка сообщения по Login собеседника в реальном времени с отображением сообщений обоих собеседников

|  |  |
| --- | --- |
| Предусловие | Сервер доступен, пользователь авторизован |
| Основной сценарий | 1. Пользователь вводит логин собеседника, вводит сообщение  2. Пользователь нажимает на «Отправить»  3. Происходит проверка существования пользователя с введенным никнеймом, проверка проходит успешно  4. Сообщение отправляется, отображается как отправителю, так и получателю  5. В таблицу ChatLog сохраняется запись о сообщении |
| Альтернативный сценарий 1 | 3а. Пользователя с введенным никнеймом не существует в системе  4. Отображается сообщение «Пользователя не существует» |