

Міністерство освіти і науки України
Національний технічний університет України
“Київський політехнічний інститут імені Ігоря Сікорського”
Факультет інформатики та обчислювальної техніки
Кафедра інформаційних систем і технологій

Лабораторна робота №1

з дисципліни «Основи WEB-технологій»

Тема: «Протокол WebSocket. Використання socket.io для розробки вебсайт додатків»

Виконав:

студент групи ІА-11

Воробей Антон Олегович

Перевірив:

Альбрехт Й. О.

Київ 2024

Завдання: Розробити додаток для обміну повідомленнями між учасниками в режимі реального часу (chat) за допомогою будь-яких технологій

Посилання на вихідний код проекту: <https://github.com/tonujet/WEB-basics/tree/main/Lab1>

Хід роботи

Переглянути увесь код додатку можна вище

Створимо ендпоінт для вебсокетів на мові програмування rust

```
#[tokio::main]
async fn main() {
    let chats = Chats::default();
    let chats = warp::any().map(move || chats.clone());

    let chat = warp::path("chat")
        .and(warp::ws())
        .and(chats)
        .and(warp::path::param())
        .and(warp::query::<User>())
        .and(warp::query::<Pagination>())
        .map(|ws: warp::ws::Ws, chats, chat_name, user: User, page| {
            ws.on_upgrade(move |socket| {
                chat_service::on_user_connection(socket, chats, chat_name,
user.username, page)
            })
        });

    warp::serve(chat).run(([127, 0, 0, 1], 3030)).await;
}
```

Створимо користувацький інтерфейс за допомогою VUE

```
<script setup lang="ts">
import LoginComponent from '@components/LoginComponent.vue'
import ChatComponent from '@components/ChatComponent.vue'
import { ref } from 'vue'
import {
    type ChatInfo,
    type ChatMessage,
    connectToChat,
    type ErrorMessage,
    type IncomingMessage
} from '@service/chat'

const isLoggedIn = ref(false)
const messages = ref<ChatMessage[]>([])
const chatInfo = ref<ChatInfo>({ chat: '', username: '' })
const errorMessage = ref('')
let error = ref<Error | null>(null)
let socket: null | WebSocket = null

const clear = () => {
    isLoggedIn.value = false
    chatInfo.value = {} as ChatInfo
    messages.value = []
}
```

```

const connect = async (info: ChatInfo) => {
  try {
    errorMessage.value = ''
    if (!info.username.trim() || !info.chat.trim()) {
      return
    }
    socket = await connectToChat(info)
    socket.onmessage = (event) => {
      let requestedMessage: ChatMessage[] | ErrorMessage = JSON.parse(event.data)
      if ('error_message' in requestedMessage) {
        errorMessage.value = requestedMessage['error_message']
        socket?.close()
        clear()
        return
      }
      console.log(requestedMessage)
      messages.value.unshift(...requestedMessage)
    }
    if (errorMessage.value) {
      return
    }
    isLoggedIn.value = true
    chatInfo.value = info
  } catch (e: unknown) {
    if (e instanceof Error) error.value = e
    clear()
  }
}

const send_message = (text: string) => {
  const message: IncomingMessage = { text }

  if (socket) {
    socket.send(JSON.stringify(message))
    console.log(socket)
    messages.value.unshift({ username: chatInfo.value.username, text })
  }
}

const disconnect = () => {
  if (socket) {
    socket.close()
    isLoggedIn.value = false
    messages.value = []
  }
}
</script>

<template>
  <div class="wrapper">
    <div class="login">
      <LoginComponent
        :is_logged="isLoggedIn"
        :error_message="errorMessage"
        @connect="connect"
        @disconnect="disconnect"
      />
    </div>
    <ChatComponent :is_logged="isLoggedIn" :messages="messages" @send-
message="send_message" />
  </div>
</template>

```

← → ↻ http://localhost:5173 ☆ 🗑️ 👤 ⋮

Username

Room name

Перевіримо роботу чату

Vite App x +

← → ↻ http://localhost:5173 ☆ 🗑️ 👤 ⋮

Room name

@e
123123123

@w
asdsad

@q
adfafasas

message

Vite App x +

← → ↻ http://localhost:5173 ☆ 🗑️ 👤 ⋮

Room name

@e
123123123

@w
asdsad

@q
adfafasas

message

Vite App x +

← → ↻ http://localhost:5173 ☆ 🗑️ 👤 ⋮

Room name

@e
123123123

@w
asdsad

@q
adfafasas

message

Висновок: На цій лабораторній роботі я вивичив протокол Вебсокетів та використовуючі здобувші знання розробив додаток для обміну повідомленнями в режимі реального часу з використанням мови програмування rust, та фреймворку для розробки фронтенд застосунків VUE