

**МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ "ЛЬВІВСЬКА ПОЛІТЕХНІКА"**

**Інститут КНІТ
Кафедра ПЗ**

ЗВІТ

До лабораторної роботи № 5

З дисципліни: *“Програмування в Інтернет”*

На тему: *“Node.js як засіб написання сценаріїв”*

Лектор:

викл. каф. ПЗ
Степанов Д. С.

Виконав:

ст. гр. ПЗ-22
Чаус Олег

Прийняла:

ст. викл. каф. ПЗ
Грицай О. Д.

« ____ » _____ 2023 р.

Σ= _____.

Львів – 2023

Тема роботи: Node.js як засіб написання сценаріїв.

Мета роботи: Засвоїти елементи програмування серверної частини застосунків мовою Node.js.

ЗАВДАННЯ

Розробити 2 складові частини проекту :

- Використати бібліотеку socket.io
- Сайт з формою, що надсилає на сервер дані чату
- Серверна частина перевіряє чи повідомлення конкретному студенту чи декільком/всім студентам і повертає відповідь (response) всім кому це повідомлення адресовано
- Сайт реагує на відповіді. І в потрібних чатах показує повідомлення

ХІД ВИКОНАННЯ

1. Використав бібліотеку socket.io для створення двохканального з'єднання.

```
//server.js
const io = socketIO(server, {
  cors: {
    credentials: true,
    origin: 'http://localhost:5173'
  }
});

io.engine.use(sessionMiddleware);
const connectedUsers = {};

io.on('connection', (socket) => {

  if (!socket.request.session.username) {
    console.log('⊗: Unauthorized connection!');
    socket.disconnect();
    return;
  }

  const username = socket.request.session.username;
  console.log(`⚡: ${username} just connected!`);
  connectedUsers[username] = socket;

  socket.on('message', (data) => {
```

```

const { receiver, message, global } = data;
const receiverSocket = connectedUsers[receiver];
if (global) {
  console.log(`✉️: ${username} just sent a global message!`);
  console.log(data);

  socket.broadcast.emit('message', {
    sender: username,
    message: message,
    global: true,
    date: Date.now()
  });
  return;
}

console.log(`✉️: ${username} just sent a message to ${receiver}!`);
console.log(data);

if (receiverSocket) {
  receiverSocket.emit('message', {
    sender: username,
    message: message,
    date: Date.now()
  });
}

});

socket.on('disconnect', () => {
  console.log(`💡: ${username} just disconnected!`);
});
});

//app.js
let socket = null;
app.config.globalProperties.initialiseSocketConnection = () => {
  app.config.globalProperties.socket = io('http://localhost:3000', {
    withCredentials: true,
  });
  console.log(socket);
};
app.config.globalProperties.socket = socket;

```

2. Сайт надсилає дані на сервер.

```
<script>
export default {
  data() {
    return {
      formInput: '',
    },
  },
  methods: {
    submit() {
      this.$emit('formSubmitted', this.formInput);
      this.$refs.input.value = '';
    }
  },
  name: "InputBox"
}
</script>

<template>
  <div class="chat-input">
    <form class="row gx-0">
      <input class="col-11" type="text" ref="input" placeholder="Type a message" v-model="formInput"/>
      <button class="col" type="submit"
@click.prevent="submit">Send</button>
    </form>
  </div>
</template>
```

3. Серверна частина приймає повідомлення та надсилає іншому студенту.

4. Чат реагує на відповіді та показує потрібні повідомлення.

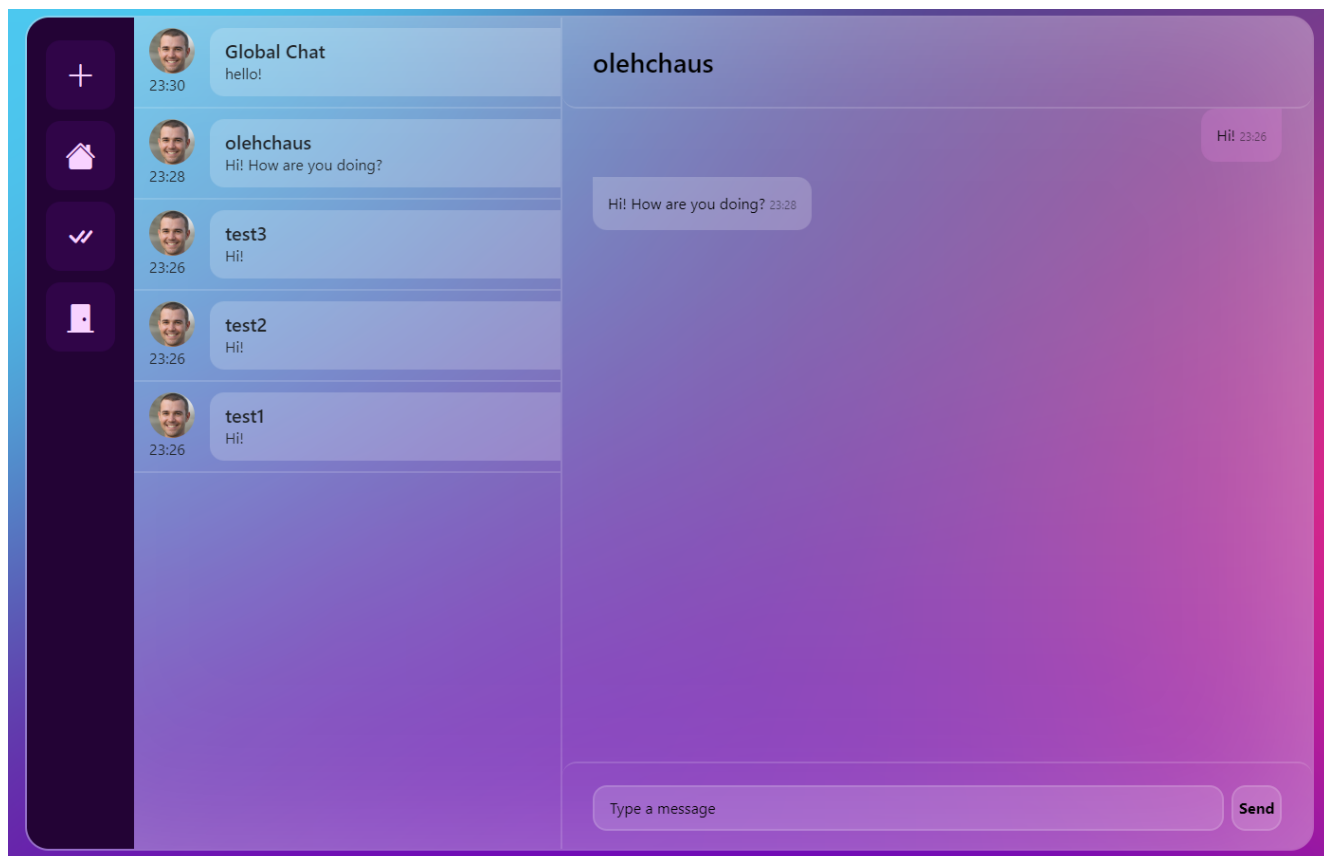


Рис. 1. Чат.

[Посилання на репозиторій в GitLab](#)

ВИСНОВКИ

Під час виконання лабораторної роботи засвоїв елементи програмування серверної частини застосунків мовою Node.js.