

НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ
«КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ імені Ігоря Сікорського»
ФАКУЛЬТЕТ ПРИКЛАДНОЇ МАТЕМАТИКИ
**Кафедра системного програмування та спеціалізованих
комп'ютерних систем**

Лабораторна робота №3
з дисципліни
«Web-дизайн»

Тема **«Розробка Web-додатка засобами
Javascript/VueJS»**

Виконав: студент III курсу

ФПМ групи KB-94

Іус І. О.

Перевірів(-ла): _____

Київ – 2022

Мета: ознайомитись із засобами фреймворка VueJS та навчитись створювати асинхронні запити до Web-сервера.

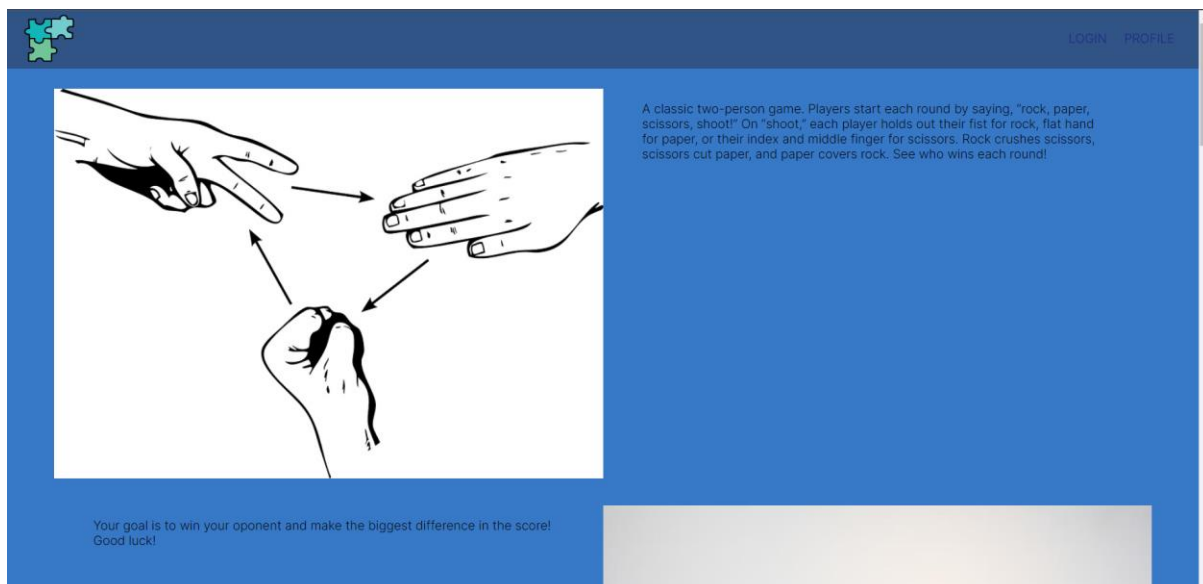
Загальне завдання: адаптувати програмний код ЛР№2 до вимог фреймворка VueJS та забезпечити завантаження необхідних даних з Web-сервера

Тема - Сайт гра (камінь ножиці папір)

Посилання на GitHub репозиторій:

<https://github.com/van2ivan?tab=repositories>

Копії екрана усіх сторінок статичного Web-додатка



Your goal is to win your opponent and make the biggest difference in the score!
Good luck!



0



1

Oponent Wins



rock

paper

scissors

Email address

Password

☒ Remember me

[Sign in](#)

First name

Last name

Email address

Password

☐ Men
☐ Women

[Sign up](#)



Ivan Ius

Student

Articles: 7

Followers: 550

Rating: 9.3

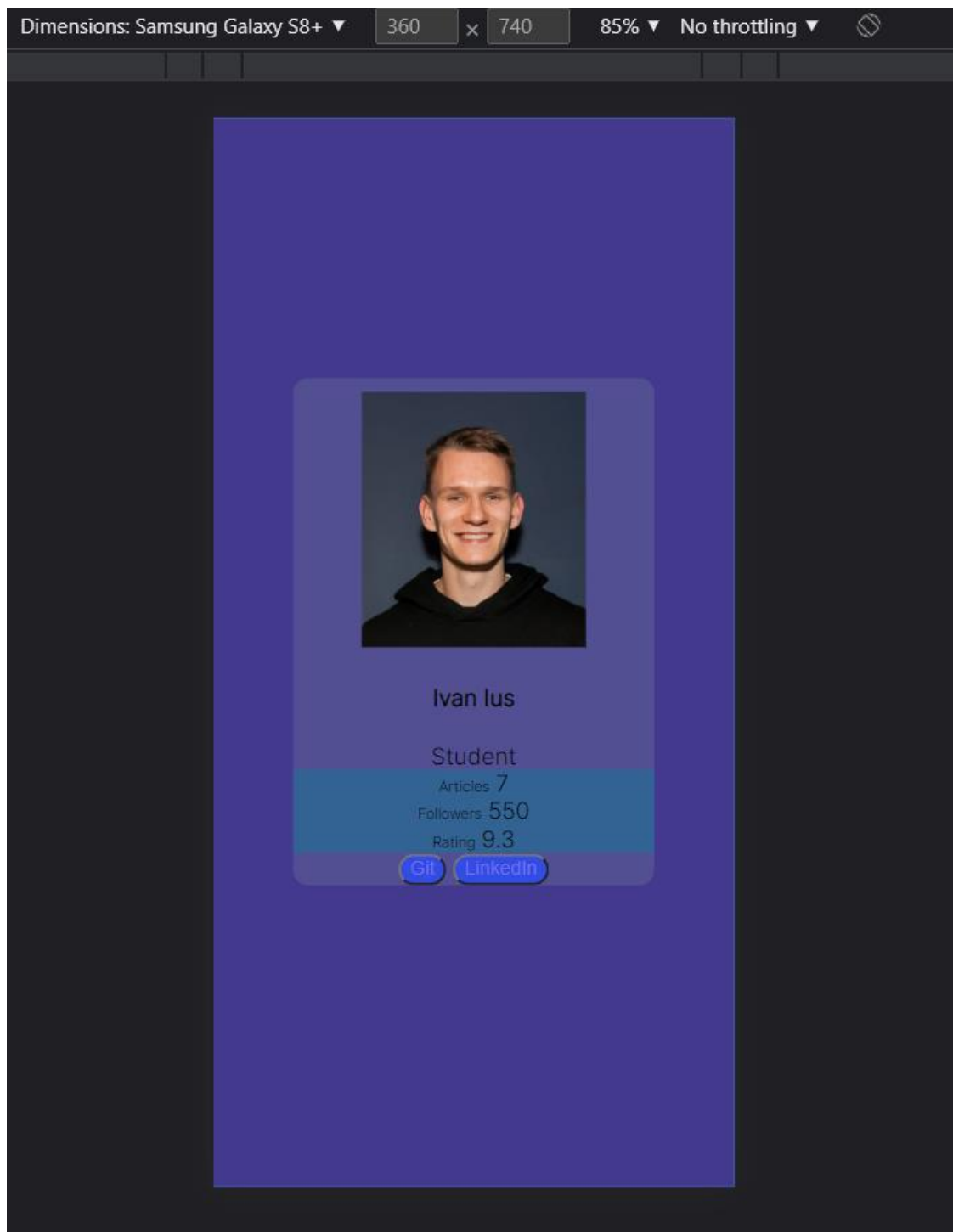
[View Profile](#)

[LOGIN](#)[PROFILE](#)

A classic two-person game. Players start each round by saying, "rock, paper, scissors, shoot!" On "shoot," each player holds out their fist for rock, flat hand for paper, or their index and middle finger for scissors. Rock crushes scissors, scissors cut paper, and paper covers rock. See who wins each round!

Your goal is to win your opponent and make the biggest difference in the score! Good luck!





Код програми

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta http-equiv="X-UA-Compatible" content="IE=edge">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Game</title>
```

```

    <link rel="icon" href="images/game-controller.svg">
    <link rel="preconnect" href="https://fonts.gstatic.com">
    <link
href="https://fonts.googleapis.com/css2?family=Inter:wght@300&display=swap"
rel="stylesheet">
    <link rel="stylesheet" href="style.css">
</head>
<body>
    <header class="header">
        <div class="header__wrapper">
            <div class="logo">
                
            </div>
            <nav class="menu">
                <ul class="menu__list">
                    <li class="menu_list-item"><a href="#login__id"
class="menu__list-link">LOGIN</a></li>
                    <li class="menu_list-item"><a href="#profile__id"
class="menu__list-link">PROFILE</a></li>
                </ul>
            </nav>
        </div>
    </header>
    <section class="about__raccoon">
        <div class="wrapper">
            <div class="about__top">
                
                <p class="about__text-top">
                    A classic two-person game. Players start each round by
saying,
                    “rock, paper, scissors, shoot!” On “shoot,” each player
holds
                    out their fist for rock, flat hand for paper, or their index
and middle finger for scissors. Rock crushes scissors,
scissors cut paper,
                    and paper covers rock. See who wins each round!
                </p>
            </div>
            <div class="about__bottom">
                <p class="about__text-bottom">
                    Your goal is to win your oponent and make the biggest
difference in the score!
                    Good luck!
                </p>
                
            </div>
        </div>
    </section>

```

```

    </div>
</section>
<section class="game">
  <div class="score">
    <div class="player-score">
      
      <p>0</p>
    </div>
    <div class="computer-score">
      
      <p>0</p>
    </div>
  </div>

  <div class="intro">
    <button>Play</button>
  </div>

  <div class="match fadeOut">
    <h2 class="winner">Choose an option</h2>
    <div class="hands">
      
      
    </div>
    <div class="options">
      <button class="rock">rock</button>
      <button class="paper">paper</button>
      <button class="scissors">scissors</button>
    </div>
  </div>
</section>
<section class="login" id="login__id">
  <div class="login__wrapper" id="log_wrap">
    <div class="sign__in">
      <form>
        <!-- Email input -->
        <div class="form-outline mb-4">
          <input type="email" id="form1Example1" class="form-control" />
          <label class="form-label" for="form1Example1">Email
address</label>
        </div>

        <!-- Password input -->
        <div class="form-outline mb-4">
          <input type="password" id="form1Example2" class="form-control"
/>
          <label class="form-label" for="form1Example2">Password</label>
        </div>

```



```

        <!-- 2 column grid layout for inline styling -->
        <div class="row mb-4">
            <div class="col d-flex justify-content-center">
                <!-- Checkbox -->
                <div class="form-check">
                    <input
                        class="form-check-input"
                        type="checkbox"
                        value=""
                        id="form1Example3"
                        checked
                    />
                    <label class="form-check-label" for="form1Example3">
Remember me </label>
                </div>
            </div>

            <div class="col">
                <!-- Simple link -->
                <a href="#"></a>
            </div>
        </div>

        <!-- Submit button -->
        <button type="submit" class="btn btn-primary btn-block">Sign
in</button>
    </form>
</div>
<div class="register" id="register">
    <form>
        <!-- 2 column grid layout with text inputs for the first and
last names -->
        <div class="row mb-4">
            <div class="col">
                <div class="form-outline">
                    <input type="text" id="form3Example1" class="form-control"
/>
                    <label class="form-label" for="form3Example1">First
name</label>
                </div>
            </div>
            <div class="col">
                <div class="form-outline">
                    <input type="text" id="form3Example2" class="form-control"
/>
                    <label class="form-label" for="form3Example2">Last
name</label>

```

```

        </div>
    </div>
</div>

<!-- Email input -->
<div class="form-outline mb-4">
    <input type="email" id="form3Example3" class="form-control" />
    <label class="form-label" for="form3Example3">Email
address</label>
</div>

<!-- Password input -->
<div class="form-outline mb-4">
    <input type="password" id="form3Example4" class="form-control"
/>

    <label class="form-label" for="form3Example4">Password</label>
</div>

<!-- Checkbox -->
<div class="form-check d-flex justify-content-center mb-4">
    <div class="form-check form-check-inline">
        <input
            class="form-check-input"
            type="radio"
            name="inlineRadioOptions"
            id="inlineRadio1"
            value="option1"
        />
        <label class="form-check-label"
for="inlineRadio1">Men</label>
    </div>

    <div class="form-check form-check-inline">
        <input
            class="form-check-input"
            type="radio"
            name="inlineRadioOptions"
            id="inlineRadio2"
            value="option2"
        />
        <label class="form-check-label"
for="inlineRadio2">Women</label>
    </div>
</div>

<!-- Submit button -->
<button type="submit" class="btn btn-primary btn-block mb-
4">Sign up</button>

```

```

        </form>
      </div>
    </div>
  </section>
  <section class="profile" id="profile__id" >
    <div class="container mt-5 d-flex justify-content-center">
      <div class="card p-3">
        <div class="d-flex align-items-center">
          <div class="image">  </div>
          <div class="ml-3 w-100">
            <h4 class="mb-0 mt-0">Ivan Ius</h4> <span>Student</span>
            <div class="p-2 mt-2 bg-primary d-flex justify-content-
between rounded text-white stats">
              <div class="d-flex flex-column"> <span
class="articles">Articles</span> <span class="number1">7</span> </div>
              <div class="d-flex flex-column"> <span
class="followers">Followers</span> <span class="number2">550</span> </div>
              <div class="d-flex flex-column"> <span
class="rating">Rating</span> <span class="number3">9.3</span> </div>
            </div>
            <div class="button mt-2 d-flex flex-row align-items-
center"> <button class="btn btn-sm btn-outline-primary w-100">Git</button>
<button class="btn btn-sm btn-primary w-100 ml-2">LinkedIn</button> </div>
          </div>
        </div>
      </div>
    </div>
  </section>
  <script src="script.js"></script>
  <script src="MVC/base.js"></script>
  <script src="MVC/controller.js"></script>
  <script src="MVC/model.js"></script>
  <script src="MVC/src.js"></script>
  <script src="MVC/view.js"></script>
</body>
</html>

```

Base

```

'use strict'

const persons = [
  {

```

```

        firstName : 'Ivan',
        lastName : 'Ius',
        email : 'ius6130@gmail.com',
        password : '12345',
        gender : 'Men',
    },
    {
        firstName : 'Anton',
        lastName : 'Potapenko',
        email : 'potap_enko@gmail.com',
        password : '54321',
        gender : 'Man',
    },
    {
        firstName : 'Michael',
        lastName : 'Mykolenko',
        email : 'mchl_mklmk@gmail.com',
        password : '67890',
        gender : 'Man',
    }
]

if(!localStorage.getItem('allPersons')){
    localStorage.setItem('allPersons',JSON.stringify(persons));
}

if(!localStorage.getItem('currentUserPosition')){
    localStorage.setItem('currentUserPosition',-1);
}

```

Controller

```

class Controller {
    constructor() {
        this.view = new View();
        this.model = new Model();
    }

    profileRegisterHandler = () => {
        const RegBtn = document.querySelector('.reg-button');
        RegBtn.addEventListener('click', (e) => {
            e.preventDefault();
            console.log('here');
            this.model.registrProfile();
        });
    };
};

```

```

profileLoginHandler = () => {
  const LogBtn = document.querySelector('.log-button');
  LogBtn.addEventListener('click', (e) => {
    e.preventDefault();
    this.model.loginProfile();
  });
};

}

const myApp = new Controller();

```

Model

```

function validate_name(fname,lname){
  if(fname.length <=0 || lname <=0 ){
    alert("Input name...");
    return false;
  }
  return true;
}

function validate_password(password){
  if(password.length<=3 || password.length > 9){
    alert("Incorect password input");
    return false;
  }
  return true;
}

function validate_email(email){
  if(email.indexOf("@")== -1){
    alert("Incorect email input");
    return false;
  }
  return true;
}

class Model{
  constructor(){
    this.currentPersonIndex = localStorage.getItem("currentPersonIndex");
    this.allPersons =JSON.parse(localStorage.getItem("allPersons"));
  }
}

```

```

loginProfile = () => {
  let currentAcc = -1;
  let currentAccIndex = -1;
  const LoginEmail = document.getElementById("logEmail");
  const LoginPassword = document.getElementById("logPassword");
  currentAcc = this.allPersons.find((acc)=>{
    acc.email === LoginEmail.value;
  })
  if (typeof currentAcc === "undefined") {
    alert("Acc with this email dosen't exist ");
    return;
  }else {
    currentAccIndex = this.allPersons.findIndex((acc) => acc.email ===
LoginEmail.value);
    if (currentAcc.password === LoginPassword.value) {
      localStorage.setItem("currentUserIndex", currentAccIndex);
      document.location.href = "./index.html";
    } else {
      alert("Wrong password");
      LoginPassword.value = "";
      return;
    }
  }
}

};

registrProfile = () => {
  const RegisterEmail = document.getElementById("reg-email");
  const RegisterPassword = document.getElementById("u-password");
  const FirstName = document.getElementById("fName");
  const LastName = document.getElementById("sName");
  const RegisterMale = document.getElementById("inlineRadio1");
  const RegisterFemale = document.getElementById("inlineRadio2");
  const newProfile = {};

  const checked = (RegisterMale.checked || RegisterFemale.checked) ? true :
false ;

  if(validate_email(RegisterEmail)&&validate_password(RegisterPassword)&&val
idate_name(FirstName,LastName)&&checked){
    newProfile.email = RegisterEmail.value;
    newProfile.fname = FirstName.value;
    newProfile.lname = LastName.value;
    newProfile.password = RegisterPassword.value;
    newProfile.gender = RegisterMale.checked ? "male" : "female";
    this.allUsers.push(newProfile);
    localStorage.setItem("allPersons", JSON.stringify(this.allUsers));
    document.location.href = "./index.html";
  }
}

```

```
};
```

Src

```
'use strict';
myApp.view.showProfile(myApp.model.currentUserIndex, myApp.model.allUsers);
myApp.profileRegisterHandler();
localStorage.setItem('currentUserIndex', -1);
myApp.profileLoginHandler();
```

View

```
class View{
  constructor(){
    this.firstName = document.getElementById('fName');
    this.lastName = document.getElementById('lName');
    this.email = document.getElementById('address_email');
    this.password = document.getElementById('u_password');
    this.gender = document.getElementById('inlineRadio1');
  }

  showProfile = (currentUserPosition, allPersonsView) => {
    if (+currentUserIndexView === -1) {
      return;
    } else {
      this.profileName.forEach((element) => element.innerHTML =
allUsersView[currentUserPosition].firstName + " " +
allUsersView[currentUserPosition].lastName)
      this.profileEmail.innerHTML =
allUsersView[currentUserPosition].email;
      this.profileGender.innerHTML =
allUsersView[currentUserPosition].gender;
      this.profileId.innerHTML = allUsersView[currentUserPosition].userId;
    }
  };
}
```

Login.vue

```
<template>
```

```

<div class="login__wrapper" id="log_wrap">

  <div class = "log_width">

    <div class="sign__in">

      <form>

        <!-- Email input -->

        <div class="form-outline mb-4">

          <input type="email" id="form1Example1" class="form-control" />

          <label class="form-label" for="form1Example1">Email_Adress</label>

        </div>

        <!-- Password input -->

        <div class="form-outline mb-4">

          <input type="password" id="form1Example2" class="form-control" />

          <label class="form-label" for="form1Example2">Password</label>

        </div>

        <!-- 2 column grid layout for inline styling -->

        <div class="row mb-4">

          <div class="col d-flex justify-content-center">

            <!-- Checkbox -->

            <div class="form-check">

              <input

                class="form-check-input"

                type="checkbox"

                value=""

                id="form1Example3"

                checked

```



```

        />
        <label class="form-check-label" for="form1Example3"> Remember me
</label>

    </div>

</div>

</div>

    <!-- Submit button -->

    <button type="submit" class="btn btn-primary btn-block">Sign in</button>

</form>

</div>

</template>

<script>

import { defineComponent } from '@vue/composition-api'

export default defineComponent({
  setup() {

  },

})

</script>

<style>

</style>

```

Profile.vue

```

<template>

<div class="container mt-5 d-flex justify-content-center">

  <div class="card p-3">

    <div class="d-flex align-items-center">

      <div class="image">  </div>

      <div class="ml-3 w-100">

        <h4 class="mb-0 mt-0">Ivan Ius </h4> <span>Student </span>

        <div class="p-2 mt-2 bg-primary d-flex justify-content-between rounded text-
white stats">

          <div class="d-flex flex-column"> <span class="articles">Articles</span>
<span class="number1">7</span> </div>

          <div class="d-flex flex-column"> <span
class="followers">Followers</span> <span class="number2">550</span> </div>

          <div class="d-flex flex-column"> <span class="rating">Rating</span>
<span class="number3">9.3</span> </div>

        </div>

        <div class="button mt-2 d-flex flex-row align-items-center"> <button
class="btn btn-sm btn-outline-primary w-100">Git</button> <button class="btn btn-sm btn-
primary w-100 ml-2">LinkedIn</button> </div>

      </div>

    </div>

  </div>

</template>

```

```

<script>

export default {

  beforeRouteEnter(to, from, next) {

    if (from.meta.auth === true){

```

```
        next();
    }
    else {
        next('/index');
    }

    to.meta.auth = from.meta.auth;
},
}
</script>
```