 МІНІСТЕРСТВО ОСВІТИ I НАУКИ УКРАЇНИ

НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ

«КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ   
ІМЕНІ ІГОРЯ СІКОРСЬКОГО»

ФАКУЛЬТЕТ БІОМЕДИЧНОЇ ІНЖЕНЕРІЇ

КАФЕДРА БІОМЕДИЧНОЇ КІБЕРНЕТИКИ

**Комп’ютерний практикум №9**

з дисципліни «Веб-технології та веб-дизайн»

**Виконав:**

студент гр. БС-81

Мумінов А.О.

**Перевірив:**

ас. каф. БМК Матвійчук О. В.

Зараховано від \_\_\_.\_\_\_.\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

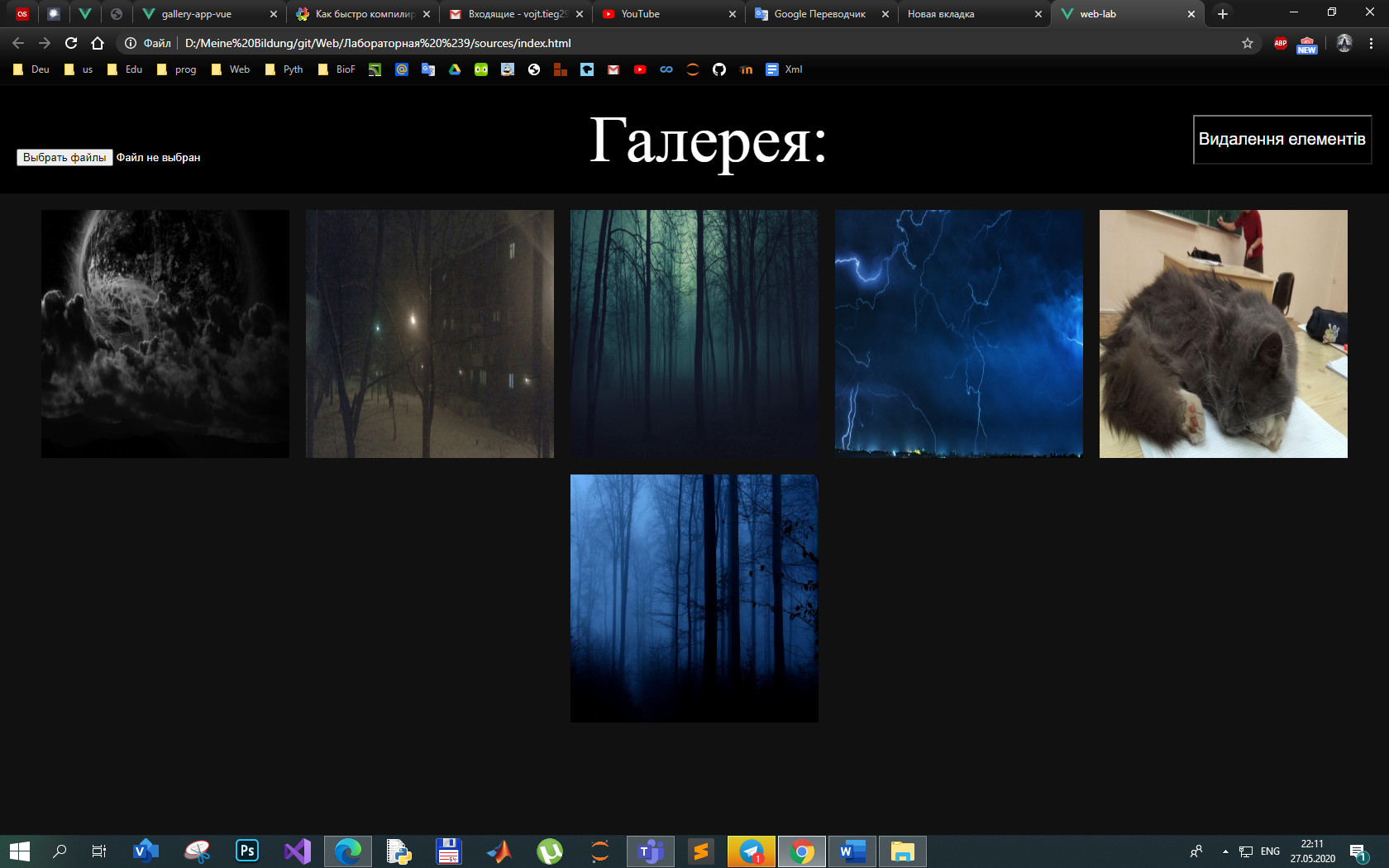
(підпис викладача)

Київ-2020

**Завдання : Створити динамічну галерею зображень (index.html).**

* Вибрати фреймворк для реалізації галереї (Vue, React, Angular)
* Реалізувати галерею зображень як у лабораторній роботі #6 (включаючи режим preview)
* в Preview режимі відображати картинки сусіди (по 1 з кожної сторони, на кільцевому списку. В галереї в кожен момент часу має бути не менше 3х зображень).
* Додати функціонал з видалення зображень
* Додати функціонал з розширення галереї (форму для додавання нових зображень)
* Галерею реалізувати за паттерном Model-View-ViewModel

Виконання,



Лістинг:  
**App.vue**  
<template>

<div id="app">

<div class="previewNotActive" id="prev">

<Items v-bind:image = "left" v-on:clickimg="clickimg"/>

<Items v-bind:image = "center" v-on:clickimg="clickimg"/>

<Items v-bind:image = "right" v-on:clickimg="clickimg"/>

</div>

<div id="header">

<form

enctype="multipart/form-data">

<input type="file" @change ="AddImage" multiple accept="image/\*,image/jpeg">

</form>

<div >Галерея:</div>

<div class = "block-button">

<button v-bind:class="{on:status\_delete}" v-on:click="status\_delete = !status\_delete"

>Видалення елементів</button>

</div>

</div>

<List\_image v-bind:List="List" v-on:clickimg="clickimg"/>

</div>

</template>

<script>

import List\_image from '@/components/list\_img.vue'

import Items from '@/components/item'

export default {

name: 'App',

data() {

return {

List: [

{id:0,src:"image/1.jpg",preview:false,left:false,right:false},

{id:1,src:"image/2.jpg",preview:false,left:false,right:false},

{id:2,src:"image/3.jpg",preview:false,left:false,right:false},

{id:3,src:"image/4.jpg",preview:false,left:false,right:false},

{id:4,src:"image/5.jpg",preview:false,left:false,right:false},

{id:5,src:"image/6.jpg",preview:false,left:false,right:false}

],

status\_delete:false,

status\_preview:false,

last\_id:5,

center:{},

left:{},

right:{}

}

},

methods: {

clickimg(id)

{

if(this.status\_delete)

this.List = this.List.filter(t => t.id !== id);

else if(!this.status\_preview)

{

document.getElementById("prev").setAttribute("class","previewActive");

this.status\_preview = true;

for (let i = 0; i < this.List.length; i++)

{

if(this.List[i].id == id)

{

let l = this.List[((i)?(i-1):this.List.length-1)];

let r = this.List[((i == this.List.length-1)?0:(i+1))];

this.left = {id:l.id,src:l.src,preview:false,left:true,right:false};

this.center = {id:this.List[i].id, src:this.List[i].src,preview:true,left:false,right:false};

this.right = {id:r.id,src: r.src,preview:false,left:false,right:true};

}

}

}

else

{

if(id == this.center.id)

{

document.getElementById("prev").setAttribute("class","previewNotActive");

this.status\_preview = false;

}else{

this.status\_preview = false;

this.clickimg(id);

}

}

},

AddImage: function(e)

{

let file = e.target.files[0];

let reader = new FileReader();

reader.addEventListener('load',event=>

{

this.List.push({id:++(this.last\_id), style\_cut:"height:100%",src:event.target.result});

});

reader.readAsDataURL(file);

}

},

components: {

List\_image,

Items

}

}

</script>

<style>

html,body

{

margin:0;

padding: 0;

background: #111;

}

#app {

-webkit-font-smoothing: antialiased;

-moz-osx-font-smoothing: grayscale;

}

#app .previewActive{

position: absolute;

background: black;

height: 100%;

width: 100%;

display: flex;

justify-content: space-between;

align-items: center;

}

#app .previewNotActive{

display: none;

}

#app #header

{

user-select: none;

font-size: 80px;

text-align: center;

background: black;

color: white;

padding: 20px;

margin-bottom: 10px;

display: flex;

flex-wrap: wrap;

justify-content: space-between;

}

#app #header div.block-header{

width: 300px;

height: 100px;

display: flex;

align-items: center;

margin:0;

padding: 0;

}

#app .block-button{

display: flex;

flex-wrap: wrap;

max-width: 300px;

align-items: center;

}

#app button{

display: block;

color: #fff;

height: 60px;

font-size: 20px;

padding-left: 5px;

background: black;

outline: none;

transition:0.2s;

}

#app .on

{

background: blue;

}

</style>

**Add\_image.vue**

<!-- <template>

<form @submit.prevent="Add()"

enctype="multipart/form-data">

<input type="file" @change ="loadImage" multiple accept="image/\*,image/jpeg">

<button type="submit">Добавить</button>

</form>

</template>

<script>

export default{

data(){

return{

NewImage:""

}

},

methods: {

Add()

{

// this.$emit('AddImage', 11);

alert(this.NewImage);

// this.$emit('AddImage',this.NewImage);

},

loadImage: function(e)

{

let file = e.target.files[0];

let reader = new FileReader();

this.AddIm(1000);

// reader.onload = function(e){

// this.NewImage = e.target.result;

this.AddIm(e.target.result);

// alert(this.NewImage);

// this.$emit('AddImage',e.target.result);

// };

// reader.readAsDataURL(file);

//document.getElementById("image").src = e.target.result;

}

}

}

</script>

<style>

form{

display: flex;

flex-wrap: wrap;

max-width: 300px;

}

</style>

-->

**item.vue**

<template>

<div id = "item"

v-bind:class ="{preview:image.preview,left:image.left, right:image.right}"

>

<img id="img" v-on:click="$emit('clickimg',image.id)":src=image.src>

</div>

</template>

<script>

export default

{

props:{

image:{

type: Object,

required: true

}

}

}

</script>

<style scoped>

div

{

margin:10px;

width: 300px;

height: 300px;

background: white;

overflow: hidden;

display: flex;

justify-content: center;

align-items: center;

padding: 0;

background: black;

}

div.preview{

position: fixed;

width: 100vw;

height: 100vh;

z-index: 1;

margin:0;

}

.preview > #img{

height: 100vh;

width: 100vw;

object-fit: contain;

animation: zoom 0.75s;

}

@keyframes zoom{

from{ height: 20vh; width: 20vw;}

to{height: 100vh; width: 100vw;}

}

div.left, div.right{

position: fixed;

z-index: 2;

margin:0;

padding: 0;

width: 20vh;

height: 20vh;

opacity: 0.75;

}

div.left{

animation: zoomL 0.75s;

}

div.right{

right: 0;

animation: zoomR 0.75s;

}

@keyframes zoomL{

from{ left: 50%;height: 0vh; width: 0vh;}

to{left: 0%;height: 20vh; width: 20vh;}

}

@keyframes zoomR{

from{ right: 50%; height: 0vh; width: 0vh;}

to{ right: 0;height: 20vh; width: 20vh;}

}

img

{

width: 120%;

height: 120%;

margin: 0.1vh;

}

</style>

**List\_img.vue**

<template>

<div id="list\_image">

<Items

v-for="image of List"

v-bind:image="image"

v-on:clickimg="clickimg"

/>

</div>

</template>

<script>

import Items from '@/components/item'

export default {

props: ['List'],

components: {

Items

},

methods: {

clickimg(id)

{

this.$emit('clickimg',id);

}

}

}

</script>

<style scoped>

#list\_image{

display: flex;

flex-wrap: wrap;

justify-content: center;

}

</style>