МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ

НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ "ЛЬВІВСЬКА ПОЛІТЕХНІКА"

Кафедра інформаційних систем

та мереж

**Звіт**  до лабораторної роботи №2

з дисципліни “Екстремальне програмування ”

**Основи роботи з Angular Framework, JS та Java**

Виконала: студентка гр. КН-311 Боштан А.О.

Прийняв:

Щербак С.С.

Львів - 2020

Теоретичні відомості

Раніше серед JS фреймворків AngularJS був улюбленцем, так як був представлений корпорацією Google в 2012. Він був побудований за концепцією Model-View-Controller. Автори ж фреймворка називали концепцію "Model-View- \*" або навіть "Model-View-Whatever".

Фреймворк на чистому JS повинен був роз'єднати логіку програми та маніпуляції з DOM. Його метою було поновлення динамічних сторінок. Але це було не так нав'язливо: під контролем AngularJS могла бути тільки частина сторінки. Цей фреймворк представив багато потужних функцій, що дозволяють розробникам легко створювати багаті, односторінкові додатки.

Зокрема, була представлена ​​цікава концепція прив'язки даних, по якій дані автоматично оновлюються при будь-якій зміні моделі і навпаки. Поверх цього представили ідею директив, які дозволяють створювати власні HTML теги і оживляти їх через JS.

Також була представлена ​​важлива концепція ін'єкції залежностей, яка дозволяє пов'язувати компоненти програми, полегшуючи повторне використання і тестування коду.

AngularJS став популярний дуже швидко і отримав велику підтримку. Проте, розробники вирішили піти ще далі і перейшли до створення нової версії Angular 2 (пізніше просто Angular без частини JS). Фреймворк отримав нове ім'я не просто так: він був повністю переписаний і перепроектований, а багато концепції були переглянуті.

Перший стабільний реліз Angular 2 опубліковано в 2016, після чого AngularJS став втрачати популярність на догоду новій версії. Одна з головних функцій Angular 2 - можливість розробляти під кілька платформ: веб, мобільні пристрої і нативний десктоп (AngularJS за замовчуванням не мав мобільного підтримки).

У листопаді 2017 вийшов Angular 5. Він сумісний з попередніми версіями Angular.

Переваги Angular. Він підтримується на різних платформах (веб, мобільні пристрої, нативний десктоп), він потужний, сучасний.

Angular представляє не тільки інструменти, але і шаблони дизайну для створення обслуговується проекту. При правильному створенні Angular додатки у вас не буде плутанини класів і методів, які складно переправити і ще складніше тестувати. Код зручно структурований, можна швидко зрозуміти, що до чого.

Це JS, але краще. Angular побудований на TypeScript, який, в свою чергу, покладається на ES6. Вам не потрібно вчити повністю нову мову, і ви отримуєте функції типу статичної типізації, інтерфейсів, класів, простору імен, декоратори і т.д.

У Angular вже є багато інструментів для створення програм. Завдяки директивам, HTML елементи можуть вести себе динамічно. Ви можете підсилити форми за допомогою FormControl і представити різні правила валідації. Можна легко посилати асинхронні HTTP запити різних типів.

Також Angular намагався прибрати жорстку зв'язок між різними компонентами програми. Ін'єкція проходить подібно NodeJS, що дозволяє легко замінювати компоненти.

Всі маніпуляції з DOM проходять там, де повинні. У Angular уявлення і логіка програми не пов'язані, що сильно очищає і спрощує розмітку.

Angular ретельно протестований і підтримує юніт ,Jasmine і Protractor.

Можна сказати, що Angular не просто фреймворк, а платформа, яка дозволяє розробникам будувати додатки для інтернету, мобільних пристроїв і робочого столу.

Виконання лабораторної роботи

Даний проект зроблений за допомогою мови програмування Java, JS, HTML, CSS.

На даному веб-сайті можна побачити реалізацію галереї, яка зроблена у вигляді слайдера з фото. Дана компонента реалізована за допомогою JS. Ось невелика частинка її коду :

<**script type="text/javascript"**>  
 ***document***.querySelector(**'#gallery'**).appendChild(**JGallery**.create([{  
 **title**: **'Images'**,  
 **items**: [  
 {  
 **url**: **'images/large/photo1.jpg'**,  
 **thumbUrl**: **'images/thumbs/photo1.jpg'** },  
 {  
 **url**: **'images/large/photo2.jpg'**,  
 **thumbUrl**: **'images/thumbs/photo2.jpg'** },  
 {  
 **url**: **'images/large/photo3.jpg'**,  
 **thumbUrl**: **'images/thumbs/photo3.jpg'** },  
 {  
 **url**: **'images/large/photo4.jpg'**,  
 **thumbUrl**: **'images/thumbs/photo4.jpg'** },  
 {  
 **url**: **'images/large/photo5.jpg'**,  
 **thumbUrl**: **'images/thumbs/photo5.jpg'** },  
 {  
 **url**: **'images/large/photo6.jpg'**,  
 **thumbUrl**: **'images/thumbs/photo6.jpg'** },  
 {  
 **url**: **'images/large/photo7.jpg'**,  
 **thumbUrl**: **'images/thumbs/photo7.jpg'** },  
 {  
 **url**: **'images/large/photo8.jpg'**,  
 **thumbUrl**: **'images/thumbs/photo8.jpg'** },  
 {  
 **url**: **'images/large/photo9.jpg'**,  
 **thumbUrl**: **'images/thumbs/photo9.jpg'** },  
 {  
 **url**: **'images/large/photo10.jpg'**,  
 **thumbUrl**: **'images/thumbs/photo10.jpg'** }  
 ]  
 }, {  
 **title**: **'Video'**,  
 **items**: [  
 {  
 **element**: **JGallery**.createElement(**'<video src="TUZ.mp4" loop autoplay muted/>'**),  
 **thumbElement**: **JGallery**.createElement(**'<span>Video</span>'**)  
 }  
 ]  
 }]).getElement());  
</**script**>

Також необхідні файли з JS кодом підключаються в <head> :

<**head**>  
 <**meta charset="utf-8"**>  
 <**title**>Steak house</**title**>  
 <**link rel="stylesheet" href="master.css"**>  
 <**script type="text/javascript" src="js/jgallery.js"**></**script**>  
  
 <**script type="text/javascript" src="https://ajax.googleapis.com/ajax/libs/jquery/1.10.1/jquery.min.js"**></**script**>  
 <**script type="text/javascript" src="js/javas.js"**></**script**>  
 <**script type="text/javascript" src="js/javas2.js"**></**script**>  
 <**script type="text/javascript" src="js/javas3.js"**></**script**>  
  
</**head**>

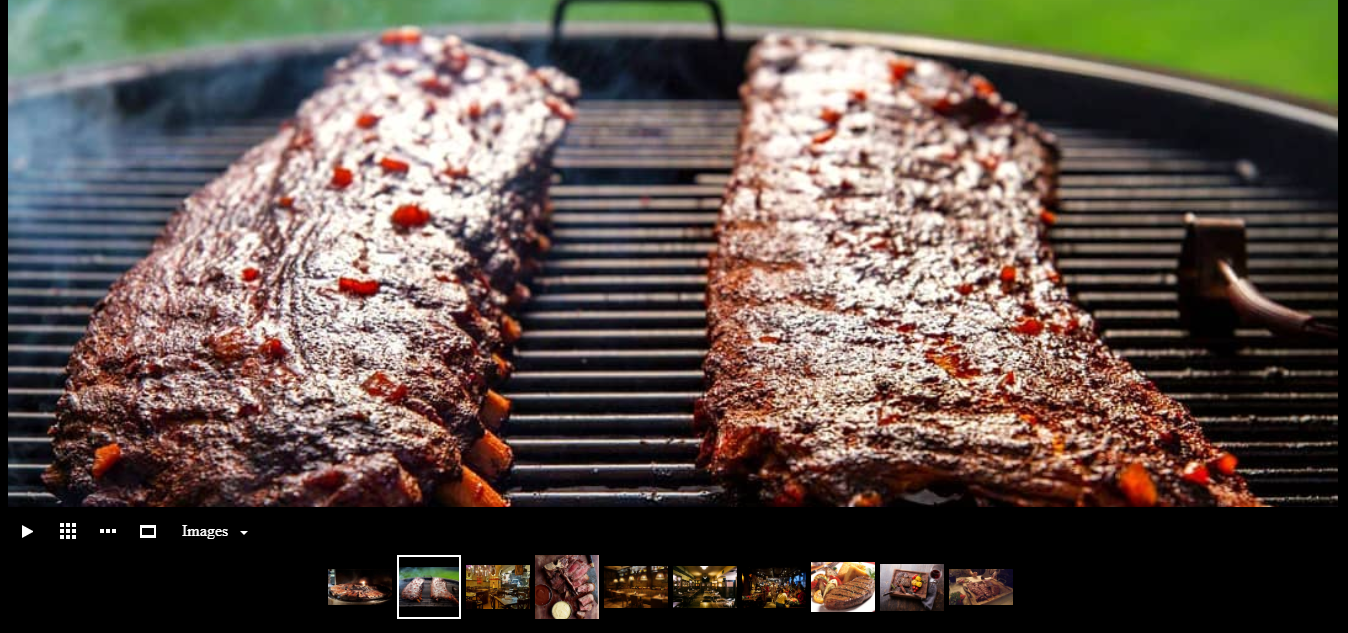


Рис.1 Вигляд галереї на сайті у вигляді слайдера

Якщо ж говорити про Angular, то для початку потрібно встановити Angular CLI. Для цього потрібно в терміналі виконати команду

npm install -g @angular/cli@latest

Це може зайняти певний час. Під час вам потрібно вказати параметри конфігурації вашої системи, а саме її версію, назву, автора, ключові слова і т.д. Все це також робиться в терміналі під час встановлення Angular CLI.

Після успішної інсталяції з’являється файл package.json з json файлом з наших налаштувань. Далі можна створювати власний проект. Для цього потрібно в терміналі виконати команду ng new StakeHouseApp.

Після цього ви можете вже запустити ваших додаток, який буде виглядати як стандартний шаблон.   
Для цього потрібно виконати команду ng serve -o (компіляція та запуск).

Після перевірки успішного запуску проекту ми можемо почистити .html файл та писати нашу програму. Можна видалити з нього все крім <router-outlet>.

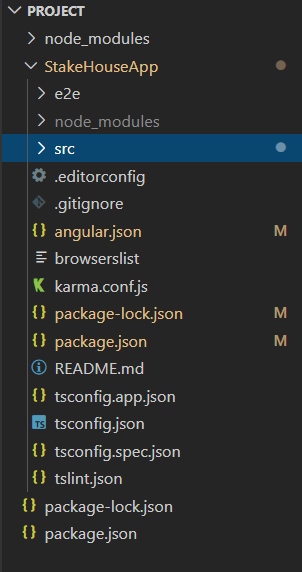


Рис.2 Загальна структура проекту

У папці src є також папка app, яка генерується автоматично. Її вміст такий:

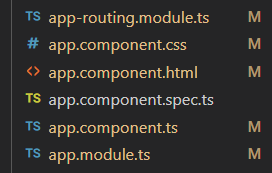


Рис.3 Структура app

app.component.ts – відповідає за створення та вміст компонент, data.service.ts – створення на налаштування сервісів, app.module.ts – створення та конфігурація модулів, app-routing.ts – створення роутингу по додатку.

Код app.module.ts:

import { BrowserModule } from '@angular/platform-browser';

import { NgModule } from '@angular/core';

import { AppRoutingModule } from './app-routing.module';

import { AppComponent } from './app.component';

@NgModule({

  declarations: [

    AppComponent

  ],

  imports: [

    BrowserModule,

     AppRoutingModule

  ],

  providers: [],

  bootstrap: [AppComponent]

})

export class AppModule { }

Код app.component.ts:

import { Component } from '@angular/core';

import { DataService } from './services/data.service';

@Component({

  selector: 'app-root',

  templateUrl: './app.component.html',

  styleUrls: ['./app.component.css']

})

export class AppComponent {

}}

У файлах app.component.html, app.component.css розміщується html та css код для компоненти, який був наведений в попередній лабораторній роботі.

Також є просто index.html, styles.css – у них прописується загальний код для всієї сторінки.

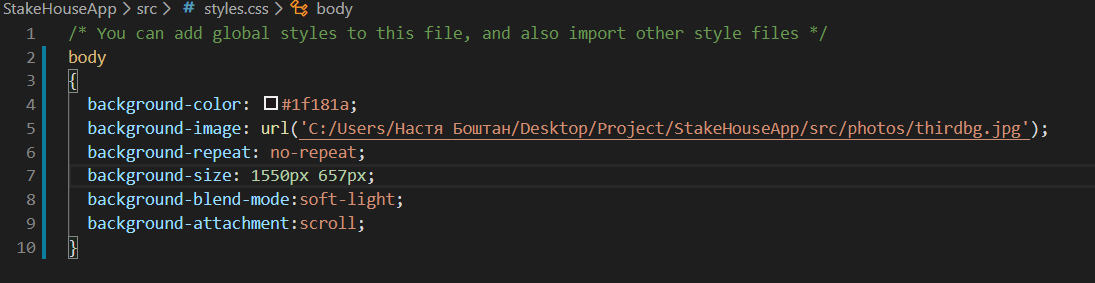


Рис.4 Вміст styles.css

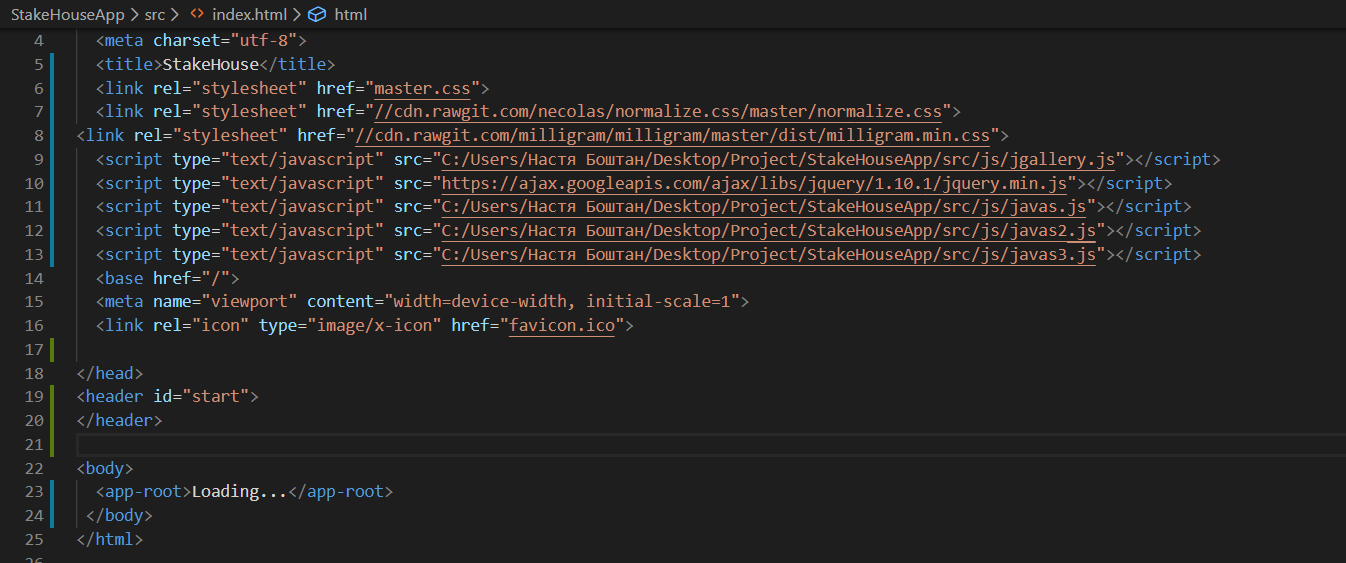


Рис.5 Вміст index.html

Висновок: у даній лабораторній роботі було ознайомлено з Angular framework.