

Міністерство освіти і науки України
Національний технічний університет
«Дніпровська політехніка»

Інститут електроенергетики
(інститут)

Факультет інформаційних технологій
(факультет)

Кафедра Програмного забезпечення комп'ютерних систем
(повна назва)

ПОЯСНЮВАЛЬНА ЗАПИСКА
кваліфікаційної роботи ступеня
бакалавра

(назва освітньо-кваліфікаційного рівня)

студента *Пешкова Дар'я Євгенівна*
(ПІБ)

академічної групи *122-18-3*
(шифр)

спеціальності *122 Комп'ютерні науки*
(код і назва спеціальності)

освітньої програми *Комп'ютерні науки*
(назва освітньої програми)

на тему: *Розробка веб-додатку для*
кейтерингової компанії на мові PHP та JavaScript

Керівники	Прізвище, ініціали	Оцінка за шкалою		Підпис
		рейтинговою	інституційною	
кваліфікаційної роботи	<i>доц. Приходченко С.Д.</i>			
розділів:				
спеціальний	<i>доц. Приходченко С.Д.</i>			
економічний	<i>доц. Касьяненко Л.В.</i>			
Рецензент				
Нормоконтролер	<i>доц. Гуліна І.Г.</i>			

Дніпро
2022

Міністерство освіти і науки України
НТУ «Дніпровська політехніка»

ЗАТВЕРДЖЕНО:

завідувач кафедри
програмного забезпечення комп'ютерних систем

(повна назва)

І.М. Удовик

(підпис)

(прізвище, ініціали)

« » 2022 року

ЗАВДАННЯ

на кваліфікаційну роботу

бакалавра

(назва освітньо-кваліфікаційного рівня)

студента 122-18-3 Пешикової Д.Є.

(група)

(прізвище та ініціали)

тема кваліфікаційної роботи

Розробка веб-додатку для

кейтерингової компанії на мові PHP та JavaScript

затверджена наказом ректора НТУ «ДП» від «18» травня 2022р. №268-с

Розділ	Зміст виконання	Термін виконання
<i>Спеціальний</i>	<i>На основі матеріалів виробничої практики та інших науково-технічних джерел провести аналіз стану рішення проблеми та постановку задачі. Обґрунтувати вибір та здійснити реалізацію методів вирішення проблеми</i>	13.05.2022 р.
<i>Економічний</i>	<i>Провести розрахунок трудомісткості розробки програмного забезпечення, витрат на створення ПЗ й тривалості його розробки</i>	27.05.2022 р.

Завдання видав

(підпис)

доц. Приходченко С.Д.

(посада, прізвище, ініціали)

Завдання прийняв до виконання

(підпис)

Пешикова Д.Є.

(прізвище, ініціали)

Дата видачі завдання: 14.01.2022 р.

Термін подання кваліфікаційної роботи до ЕК: 16.06.2022 р.

РЕФЕРАТ

Пояснювальна записка: 78 с., 29 рис., 0 табл., 3 дод., 21 джерел.

Об'єкт розробки: сайт кейтирінгової компанії

Мета кваліфікаційної роботи: створення web-застосунка для кейтирінгової компанії, за допомогою якого можна забезпечити приплив нових клієнтів та отримання зворотного зв'язку для компанії.

У вступі розглядається аналіз та сучасний стан проблеми, конкретизується мета кваліфікаційної роботи та галузь її застосування, наведено обґрунтування актуальності теми та уточняється постановка завдання.

У першому розділі проаналізовано предметну галузь, визначено актуальність завдання та призначення розробки, сформульовано постановку завдання, зазначено вимоги до програмної реалізації, технологій та програмних засобів.

У другому розділі проаналізовані наявні рішення, обрано платформи для розробки, виконано проєктування і розробка програми, описана робота програми, алгоритм і структура її функціонування, а також виклик та завантаження програми, визначено вхідні і вихідні дані, охарактеризовано склад параметрів технічних засобів.

В економічному розділі визначено трудомісткість розробленої інформаційної системи, проведений підрахунок вартості роботи по створенню програми та розраховано час на його створення.

Список ключових слів: САЙТ, ВЕБ-РЕСУРС, РОЗРОБКА, ІНТЕРНЕТ.

ABSTRACT

Explanatory note: 78 p., 29 figs., 0 tabl, 3 appx., 21 sources.

Object of development: the site of the catering company

The purpose of the qualification work: to create a web-application for a catering company, which can provide an influx of new customers and receive feedback for the company. In the introduction it is considered the analysis and the current state of the problem, specifies the purpose of the qualification work and the scope of its application, provides a justification for the relevance of the topic and clarifies the problem.

In the first section the subject branch is analyzed, the urgency of the task and purpose of development are defined, the statement of the task is formulated, requirements to software realization, technologies and software are specified.

In the second section it is analyzed the existing solutions, selected platforms for development, designed and developed the program, describes the program, algorithm and structure of its operation, as well as calling and loading the program, determines the input and output data, describes the parameters of hardware.

In the economic section it is determined the complexity of the developed information system, calculates the cost of work to create a program and calculates the time for its creation.

List of keywords: SITE, WEB RESOURCE, DEVELOPMENT, INTERNET.

ЗМІСТ

РЕФЕРАТ.....	3
ABSTRACT.....	4
СПИСОК УМОВНИХ ПОЗНАЧЕНЬ.....	7
ВСТУП.....	8
РОЗДІЛ 1. АНАЛІЗ ПРЕДМЕТНОЇ ГАЛУЗІ ТА ПОСТАНОВКА ЗАДАЧІ.....	9
1.1 Загальні відомості з предметної галузі.....	9
1.2 Призначення розробки та область застосування.....	15
1.3 Підстава для розробки.....	15
1.4 Постанова завдання.....	16
1.5 Вимоги до програми або програмного виробу.....	17
1.5.1 Вимоги до функціональних характеристик	17
1.5.2 Вимоги до інформаційної безпеки.....	17
1.5.3 Вимоги до складу та параметрів технічних засобів.....	17
1.5.4 Вимоги до інформаційної та програмної сумісності.....	18
РОЗДІЛ 2. ПРОЕКТУВАННЯ ТА РОЗРОБКА ІНФОРМАЦІЙНОЇ СИСТЕМИ.....	19
2.1 Функціональне призначення системи.....	19
2.2 Опис застосованих математичних методів.....	19
2.3 Опис використаних технологій та мов програмування.....	19
2.4 Опис структури системи та алгоритмів її функціонування.....	23
2.5 Обґрунтування та організація вхідних та вихідних даних програми.....	31
2.6 Опис роботи розробленої системи.....	32
2.6.1 Використані технічні засоби.....	32
2.6.2 Використані програмні засоби.....	32

2.6.3	Виклик та завантаження програми.....	33
2.6.4	Опис інтерфейсу користувача.....	33
РОЗДІЛ 3. ЕКОНОМІЧНИЙ РОЗДІЛ.....		40
3.1	Розрахунок трудомісткості та вартості розробки програмного продукту	40
3.2	Розрахунок витрат на створення програми.....	44
ВИСНОВКИ.....		47
СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ.....		48
Додаток А. Код програми.....		50
Додаток Б. Відгук керівника економічного розділу.....		77
Додаток В. Перелік файлів на диску.....		78

СПИСОК УМОВНИХ ПОЗНАЧЕНЬ

PTC – Paid To Click

PTW – Paid To Work

ООП – Об'єктно-орієнтоване програмування

CSS – Cascading Style Sheets

HTML – HyperText Markup Language

PHP – Personal Home Page

AJAX – Asynchronous Javascript and XML

DOM – Document Object Model

ВСТУП

В даний час, у вік інформаційних технологій людство адаптує усі сфери життя в мережу Інтернет.

Щодня Інтернет розширює можливості людини в різних сферах таких як освіта, розваги, і, найголовніше, у сфері бізнесу. Більшість людей бажають заробляти гроші, не виходячи із дому. І такою можливістю може мати кожна людина завдяки мережі. Існує багато варіацій бізнесу в Інтернеті, такі як: інтернет-магазин, розробка та розкрутка сайту, інтернет-реклама, РТС та РТW. Головними перевагами організації бізнесу в Інтернеті можна назвати безперервну роботу сайту 24 години на добу та невисоку вартість його створення та просування, а також за допомогою сайту можна надавати клієнтам необхідну інформацію, відстежувати їх інтерес до товарів та послуг, оперативно приймати замовлення. Просування сайту в Інтернеті допомагає компаніям швидко реагувати на будь-які зміни ринку, знаходити нові методи продажу товарів та послуг, і навіть збільшувати обсяги продаж[1].

Суть кваліфікаційної роботи полягає в творенні інформаційної системи, а саме в розробці сайту кейтирінгової компанії, що безпосередньо пов'язано зі спеціальністю «Комп'ютерні науки» та відповідає узагальненій тематиці кваліфікаційних робіт і переліку зазначених виробничих функцій, типових задач діяльності, умінню та компетенціям, якими повинні володіти бакалаври спеціальності.

Компанія, своїм рішенням розробити web-сайт, забезпечує приплив нових клієнтів. Сайт завдає замовнику можливість швидко реагувати на побажання користувача, а також вносити відповідні зміни в роботу web-сайту.

Web-сайт, розроблений із застосуванням сучасних технологій, являє собою основний інформаційний ресурс, за допомогою якого можна здійснювати:

- передачу всієї необхідної інформації користувачеві;
- безпосередній контакт з користувачем, а також інформаційну підтримку;
- здійснювати рекламну компанію.

РОЗДІЛ 1

АНАЛІЗ ПРЕДМЕТНОЇ ГАЛУЗІ ТА ПОСТАНОВКА ЗАВДАННЯ

1.1 Загальні відомості с предметної галузі

Глобальна комп'ютеризація та поява Інтернету має величезний вплив на наше та життя нашого майбутнього покоління. Інтернет - це величезний важіль для просування існуючого бізнесу та створення нових. Однією з основ сучасного бізнесу є web-сайт.

Web-сайти прийнято поділяти на статичні та динамічні.

Статичні сайти складаються з незмінних сторінок. При запиті такого сайту в браузері сервер відразу надає готовий HTML-документ в початковому вигляді, в якому він і був створений. Крім HTML, в коді таких сторінок використовується CSS і JavaScript, що забезпечує їх легкість і швидке завантаження.

До переваг статичного сайту можна віднести:

- для розробки не потрібно мати спеціальні програми, код можна писати навіть у Блокноті;
- для такого сайту не має значення сервер хостингу, він буде працювати навіть на локальному
- легке редагування зовнішнього вигляду, можна змінювати будь-яку сторінку та додавати нову и це не вплине на інші частини сайту;
- швидке завантаження в браузері.

Незважаючи на свої переваги, також є свої недоліки:

- при збільшені веб-сторінок, підтримувати сайт стає складніше, через складнощі з внесенням змін, так-як не має ділення на «компоненти»;
- неможливо застосовувати такі елементи сайту, як голосування, чат, коментарі, відгуки і т.д., не використовуючи сторонніх інструментів;
- також важко, якщо взагалі можливо, розділення прав доступу до вмісту сайту.

Динамічні сайти, в свою чергу, мають змінювані сторінки, адаптуються під

конкретного користувача. Такі сторінки не розміщені на сервері в готовому вигляді, а збираються заново по кожному новому запиту. Спочатку сервер знаходить потрібний документ і відправляє його інтерпретатору, який компілює код з HTML-документа і звіряється з файлами і базою даних. Після цього документ повертається на сервер і потім відображається в браузері. Для інтерпретації сторінок на серверній стороні використовуються мови програмування Java, PHP, ASP і інші [2].

Яскравим представником динамічних сайтів можна назвати сайти створенні на основі систем управління контентом (CMS). За допомогою цих систем можна використовувати вже готові компоненти та програмні модулі, без необхідності створювати їх заново. На основі однієї CMS можна створити будь-яку кількість динамічних сайтів.

У динамічних сайтах реалізовано поділ змісту і оформлення web-сторінок, що, в свою чергу, дозволяє швидко змінювати інформацію на сайтах без необхідності змінювати програмні коди сторінок. Це є одним з найголовніших переваг динамічних сайтів.

Крім переваг, динамічні сайти мають і ряд недоліків. У порівнянні зі статичними сайтами, динамічні більш «важкі», дають велике навантаження на сервер - отже, вони більш вимогливі до хостингу, ресурсів сервера.

Для розробки інтерфейсу користувача використовуються наступні інструменти.

HTML (Hypertext Markup Language) — це текстовий підхід до опису структури вмісту, що міститься у файлі HTML. Ця розмітка повідомляє веб-браузеру, як відображати текст, зображення та інші форми мультимедіа на веб-сторінці.

Принцип роботи HTML – це текстовий файл, що містить певний синтаксис, файли та правила найменування, які показують комп'ютеру та веб-серверу, що він знаходиться в HTML і повинен читатися як такий. Веб-браузери отримують HTML-документи з веб-сервера або з локальної пам'яті і передають документи в мультимедійні веб-сторінки.

CSS (Cascading Style Sheets, Каскадні таблиці стилів) — це спеціальна мова

стилю сторінок, яку використовують для опису їхнього зовнішнього вигляду.

CSS є основною технологією всесвітньої павутини, поряд із HTML та JavaScript.

Найчастіше CSS використовують для візуальної презентації сторінок, написаних HTML та XHTML, але формат CSS може застосовуватися до інших видів XML-документів.

CSS наповнює веб-сайт кольором та стилем. Використовуючи CSS, ви можете керувати кольором тексту, стилем шрифтів, інтервалом між абзацами, розміром і розташуванням стовпців, які фонові зображення або кольори використовуються, дизайном макета, варіантами відображення для різних пристроїв і розмірів екрана. а також ряд інших ефектів.

Стандарт CSS визначає порядок та діапазон застосування стилів, тобто, в якій послідовності і для яких елементів застосовуються стилі. Таким чином, використовується принцип каскадності, коли для елементів вказується лише та інформація про стилі, що змінилася або не визначена загальнішими стилями.[3]

JavaScript – мультіпарадигменна мова програмування. Підтримує об'єктно-орієнтована, імперативний і функціональний стилі. Є реалізацією стандарту ECMAScript.

JavaScript - була створена для отримання доступу зі сценаріїв до всіх елементів HTML-документ. Іншими словами, вона надає засоби для динамічної взаємодії з користувачем Основні архітектурні риси: динамічна типізація, слабка типізація, автоматичне керування пам'яттю, прототипне програмування, функції як об'єкти першого класу.

Для динамічної зміни даних без перезавантаження сторінки використовують Ajax технологію.

Ajax (Asynchronous Javascript and XML - «асинхронний JavaScript і XML») - підхід до побудови призначених для користувача інтерфейсів веб-додатків, що полягає в «фоновому» обміні даними браузера з веб-сервером. В результаті при оновленні даних веб-сторінка не перезавантажується повністю, і веб-додатки стають швидше і зручніше[5].

AJAX - не самостійна технологія, а концепція використання декількох суміжних технологій. AJAX базується на двох основних принципах:

- використання технології динамічного звернення до сервера «на льоту», без перезавантаження всієї сторінки повністю, наприклад з використанням XMLHttpRequest (основний об'єкт);
- використання DHTML для динамічної зміни змісту сторінки.

Дії з інтерфейсом перетворюються в операції з елементами DOM (Document Object Model), за допомогою яких обробляються дані, доступні користувачеві, в результаті чого уявлення їх змінюється. Тут же проводиться обробка переміщень і клацань мишею, а також натискань клавіш. Каскадні таблиці стилів, або CSS, забезпечують узгоджений зовнішній вигляд елементів програми та спрощують звернення до DOM-об'єктів. Об'єкт XMLHttpRequest використовується для асинхронного взаємодії з сервером, обробки запитів користувача і завантаження в процесі роботи необхідних даних.

Основою для розробки продукту був аналіз вже існуючих сайтів кейтерингових компаній міста та країни.

Ось деякі з них:

1) <https://www.d-a.dp.ua>

«Драйв апетит» - кейтерингова компанія розташована у Дніпрі, так звані «старожили» цієї ніші у місті. Сайт функціональний та задовольняє усі потреби користувача.

На головній сторінці розташована інформація про послуги, які надає компанія та їх пріоритети. На сайті присутня можливість авторизуватись та залишити відгук.

Головна сторінка сайту зображена на рисунку 1.1.

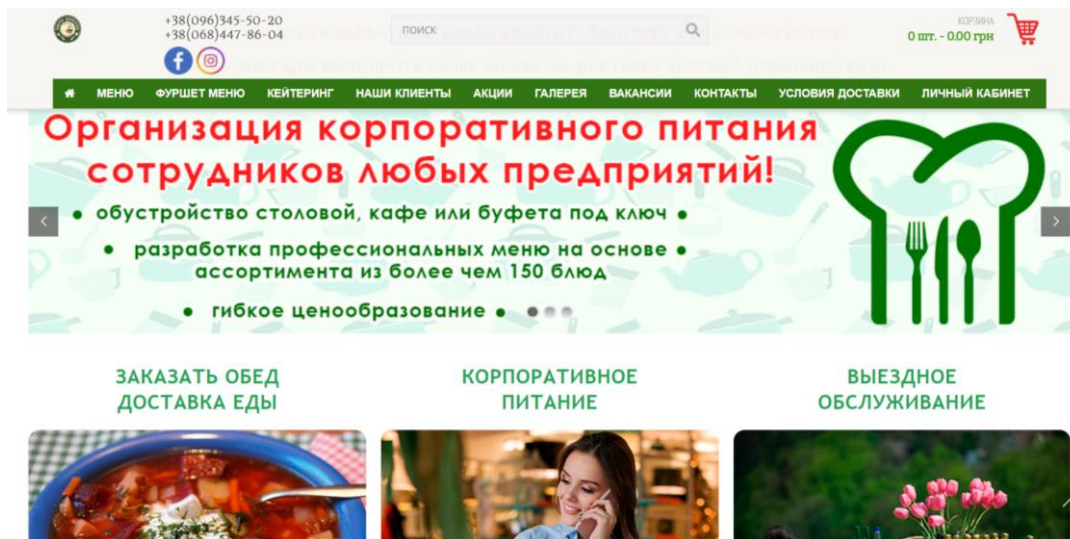


Рис. 1.1 – Драйв апетит головна сторінка

2) <http://dubki-cafe.com.ua>

«Дубки» - один з представників кейтерингу в м. Житомир. Цей заклад відрізняється від попереднього тим, що вони займаються не лише доставкою та організацією святкових заходів, а ще й є функціонуючим рестораном.

З недоліків сайту можна зазначити зовнішній вигляд, дивлячись на те як розвивається веб-дизайн, цей сайт вже застарілий.

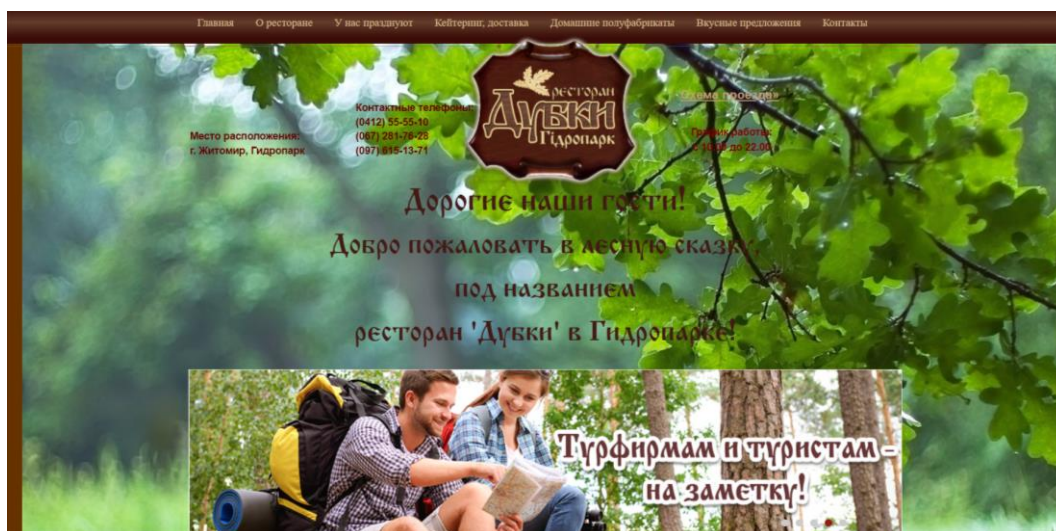


Рис. 1.2 – Дубки головна сторінка

Також можна зробити висновок, що на сайті не актуальна інформація, дивлячись на галерею фото, представлену на головній сторінці (Рис.1.3).



Рис. 1.3 – Дубки фотогалерея

3) bo.party

Данна кейтерингова компанія розташована у місці Кривий Ріг, вони займаються організацією святкових заходів, шляхом надання ресторанних послуг на виїзді. Це може бути банкет на весіллі, кейтеринг на корпоративі, або ж доставка десертів та закусок у боксах.

Сторінка даної кейтерингової компанії розташована в соціальній мережі Instagram (Рис. 1.4), в шапці сторінки розміщена головна інформація – це меню, відгуки, фото звіти і т.д.

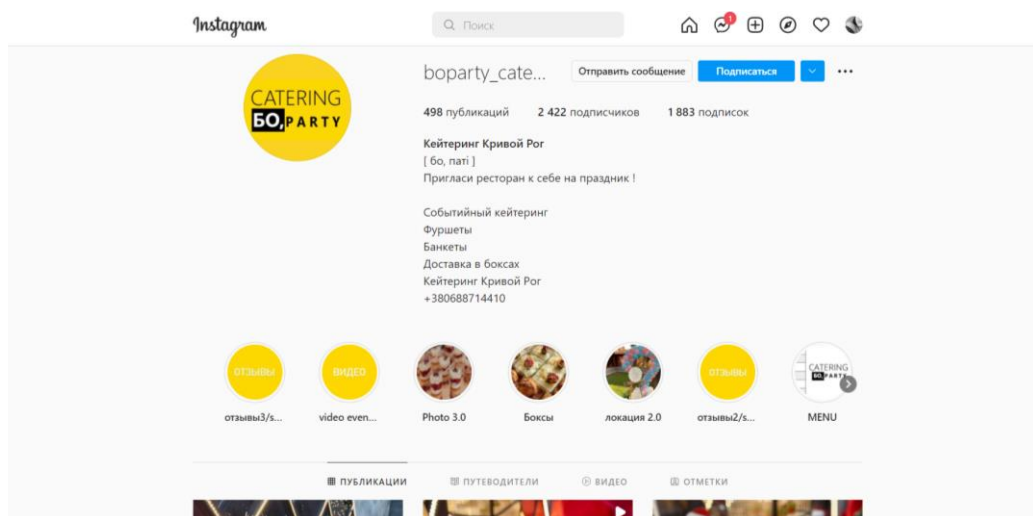


Рис. 1.4. - сторінка bo.party

Проаналізувавши подібні рішення, мною було вирішено створити веб- сайт із інтерфейсом подібним до сайтів великих предстаників цієї ніші. Щоб буди конкурентоспроможними

1.2 Призначення розробки та область застосування

Темою бакалаврської кваліфікаційної роботи виступає: «Розробка Web сайту для кейтерингової компанії». Головною метою роботи є створення веб-сайту, який представляє собою інструмент для бронювання числа на захід та ознайомлення з послугами компанії.

Розроблений проєкт повинен відповідати сучасним стандартам, та задовольняти вимоги продиктовані замовником, а також бути зручним для кінцевого користувача, мати інтуїтивно зрозумілий інтерфейс для користувача, працювати на різних пристроях різних конфігурацій, та під управлінням різних ОС.

1.3. Підстава для розробки

Відповідно до освітньої програми, згідно навчального плану та графіків навчального процесу, в кінці навчання студент виконує кваліфікаційну роботу.

Тема роботи узгоджується з керівником проєкту, випускаючою кафедрою, та затверджується з наказом ректора.

Таким чином підставами для розробки (виконанням кваліфікаційної роботи) є:

- освітня програма спеціальності 122 «Комп'ютерні науки»;
- навчальний план та графік навчального процесу;
- наказ ректора Національного технічного університету «Дніпровська політехніка» № 268-с від 18.05.2022 р;
- завдання на кваліфікаційну роботу на тему «Розробка веб-додатку для кейтерингової компанії на мові PHP та JavaScript».

1.4. Постановка завдання

Темою дипломної роботи є розробка і створення web-сайту. Тема є актуальною, так як, на даний момент комунікація із аудиторією здійснюється через так зване «сарафане радіо» та у соціальній мережі – Інстаграм, яких не достатньо для повноцінної комунікації з глядачем, також із створенням веб-сайту буде полегшено просування послуг, а також допоможе привести нових відвідувачів. Метою роботи є розробка web-сайту із застосуванням засобів HTML, CSS та мови програмування JavaScript та PHP.

Даний ресурс повинен бути простим у використанні та повинен мати приємний та інтуїтивно зрозумілий інтерфейс, за допомогою якого можна здійснювати:

- передачу всієї необхідної інформації про послуги які надає компанія;
- безпосередній контакт з користувачем, проведення опитувань а також інформаційну підтримку.

Для реалізації проекту необхідно виконати наступні завдання:

- вивчити теоретичні аспекти створення web-сайтів;
- провести аналіз існуючого програмного забезпечення для створення web-сайтів;
- розробити інтерфейс сайту;
- створити базу даних для сайту;
- провести аналіз працездатності створеного сайту.

Кінцевим результатом має бути web-сайт, що надає інформацію про послуги, які надає компанія, має функцію для бронювання дати події та зворотній зв'язок з користувачем.

1.5. Вимоги до програми або програмного вибору

1.5.1. Вимоги до функціональних характеристик

Метою дипломної роботи є розробка й створення на основі сучасних комп'ютерних технологій програмного продукту для перегляду інформації про послуги компанії, функції зворотного зв'язку.

Розроблюваний веб-додаток повинен забезпечувати виконання таких функцій:

- надати користувачеві інформацію про можливі послуги;
- показати доступні дати бронювання з можливістю вибору;
- швидко обмінюватися даними з базою даних;
- надати можливість забронювати дату події;
- прийняти інформацію від користувача (контактні дані, коментарі, побажання);

1.5.2. Вимоги до інформаційної безпеки

При роботі в веб сайтом досить встановлених в системі (ПК, ноутбука, мобільних пристроїв) вимог до інформаційної безпеки. Особливих вимог до інформаційної безпеки при роботі в сервісом не встановлюється. Веб-сервіс працює через браузер користувача і не потребує особливих прав для його використання.

1.5.3. Вимоги до складу та параметрів технічних засобів.

Програмний продукт не є вимогливий до складу та параметрів технічних засобів та може завантажуватись на ПК, ноутбуках, планшетах чи смартфонах різних типів та конфігурацій під управлінням різних ОС.

Для завантаження сторінки необхідно лише веб-браузер та доступ до мережі Інтернет.

Для нормального функціонування програми рекомендовано, наприклад, для ПК чи планшету, щоб обчислювальна машина, відповідала наступним вимогам:

- процесор класу Intel чи AMD з тактовою частотою не менш 2.0 ГГц;
- не менше 1 GB оперативної пам'яті;
- монітор з діагоналлю не менше 6";
- 2 Гб вільного місця на жорсткому диску;
- доступ до мережі Internet;
- клавіатура/маніпулятор «миша».

Наведені вище технічні характеристики є рекомендованими, тобто при наявності технічних засобів не нижче зазначених, розроблений програмний виріб буде функціонувати відповідно до вимог щодо надійності, швидкості обробки даних і безпеки, висунутими замовником.

1.5.4. Вимоги інформаційної та програмної сумісності.

Для коректного функціонування програми необхідно, щоб програмне забезпечення обчислювальної машини, на якій буде експлуатуватися веб-додаток, відповідало наступним вимогам:

- Веб браузер Firefox або Google Chrome.
- Операційна система Windows 7/10.

Веб-орієнтована підсистема має бути реалізована на мові програмування. Для візуалізації було використано CSS який описує зовнішній вигляд сторінки.

РОЗДІЛ 2

ПРОЄКТУВАННЯ ТА РОЗРОБКА ІНФОРМАЦІЙНОЇ СИСТЕМИ

2.1. Функціональне призначення системи

Призначенням сайту, що представлений в даній роботі, є запропонування послуг з кейтерінгу у місті Дніпро.

Розроблена інформаційна система призначена для:

- надання можливості обмінюватися даними та запитами між користувачем та адміністратором;
- систематизувати дані компанії.

На сайті використані різні інструменти для побудови довіри, наприклад: акційні пропозиції, можливість залишати відгуки і тд. Все це дозволяє розвивати бізнес.

За допомогою сайту замовник розширити зону пошуку клієнтів та демонструвати послуги компанії без менеджерів з продажів.

Даний сайт дозволяє замовнику швидко обробляти та відстежувати замовлення.

2.2. Опис застосованих математичних методів

Оскільки особливості предметної галузі розв'язуваної задачі не передбачають застосування математичних методів, при розробці системи відображення та управління математичні методи не використовувалися.

2.3. Опис використаних технологій та мов програмування

Для розробки front-end частини сайту по наданню послуг ретуші було використано наступні мови програмування та технології:

- HTML;

- CSS;
- JavaScript;
- AJAX;
- JQuery.

Для розробки back-end частини сайту застосовувалась мова:

- PHP.

Стиль та структура сторінок будь-якого сайту є головним пунктом при створенні системи, за це відповідають HTML і CSS. HTML відповідає за основну структуру сайту. Формування стилю та презентація додатка контролюється CSS, а от за поведінку елементів відповідає JavaScript.

HTML (HyperText Markup Language — «мова гіпертекстової розмітки») — самий базовий будівельний блок Інтернету. Він визначає зміст і структуру веб-контенту.

Під гіпертекстом («hypertext») розуміються посилання, які з'єднують веб-сторінки один з одним або в межах одного веб-сайту, або між веб-сайтами. Посилання є фундаментальним аспектом Інтернету [10].

Для створення уявлення Web-сторінок призначена технологія каскадних таблиць стилів (Cascading Style Sheets, CSS), або просто таблиць стилів. Таблиця стилів містить набір правил (стилів), що описують оформлення самої Web-сторінки та окремих її фрагментів. Ці правила визначають колір тексту та вирівнювання абзацу, відступи між графічним зображенням та обтікаючим його текстом, наявність[5]. З допомогою каскадних таблиць можна контролювати розміщення різних частин однієї сторінки. Це дуже зручно при адаптації застосунку на різних пристроях, таких як смартфон, планшет та ПК з різним розширенням.

Стилі складаються зі списку правил. Кожне правило, у свою чергу, складається з одного або більше селекторів, розділених комами, і блоку визначень. Блок визначення оточеного фігурними дужками, і складається з набору властивостей і їх значень. Схематично це можна показати так:

```
селектор, селектор {  
    властивість: значення;
```

властивість: значення;

властивість: значення; }

Якщо, при розробці, елементу призначено властивості декількох правил, то стандарт CSS визначає пріоритети. Це називається "каскадом", в якому для властивостей розраховуються пріоритети, що робить результати передбачуваними.

Селектор - це спосіб встановити стиль певному HTML-елементу. У CSS є три види основних селекторів:

1. Селектор тег. Це коли у CSS-властивості ви задаєте стилі не якомусь конкретному елементу, а всім елементам документа, які визначаються певним тегом. Наприклад: усім елементам "h1", "p", "ul", "td" та ін.
2. Селектор класу. І тут розробник задає один клас кільком елементам. А потім у CSS-властивості прописує стилі цього класу. У цьому випадку стилі будуть застосовуватися до всіх елементів, яким задано цей клас. Такий підхід дозволяє задавати властивості відразу декільком HTML-елементам. При цьому список елементів класу розробник задає самостійно.
3. Селектор ідентифікатора. Один ідентифікатор належить одному елементу. Не можна задавати той самий ідентифікатор декільком елементам. І тут потрібно застосувати клас елементів. Таким чином, ідентифікатор це спосіб задати CSS-властивості одному певному HTML-елементу.[7]

Пріоритет CSS-стилю між селекторами буде таким:

- найвищий пріоритет CSS у ідентифікатора;
- найнижчий CSS пріоритет у селектора тега.

Але після всього вище сказаного найвищий пріоритет має термін "!important". Наприклад, терміново потрібно встановити певний стиль якомусь HTML-елементу. Але знайти місце, де йому визначаються стилі, тому що до проекту підключено багато зовнішніх та внутрішніх таблиць стилів, не можете. Розбиратися в їхній ієрархії немає часу. У цьому випадку на допомогу приходять термін «!important».

Що робить цей термін? Він «ламає» каскадність та пріоритет усіх стилів CSS, тому що визначає той стиль, поряд з яким він прописаний. Селектор або рівень

таблиці йому неважливий. Термін "!important" "перебиває" навіть ті стилі, які прописані в атрибутах HTML-елемента, хоча вони за пріоритетом є найвищими.

Але використовувати термін потрібно лише у надзвичайних ситуаціях, бо він що він ламає ієрархію стилів, а отже, може викликати їхній конфлікт. Тому його застосовувати у виняткових випадках, коли по-іншому не вдається встановити стилі елементу сторінки.

Синтаксис мови JavaScript багато в чому нагадує синтаксис Сі та Java, семантично ж мова набагато ближче до Self, Smalltalk або навіть Ліспу. У JavaScript:

- всі ідентифікатори реєстру, в назвах змінних можна використовувати літери, підкреслення, символ долара, арабські цифри, назви змінних не можуть починатися з цифри;
- для оформлення однорядкових коментарів використовуються `/*` `*/`, багаторядкові і однорядкові коментарі починаються з `/*` і закінчуються `*/`.

Структурно JavaScript можна представити у вигляді об'єднання трьох чітко помітних одна від одної частин:

- ядро (ECMAScript);
- об'єктна модель браузера (Browser Object Model або BOM);
- об'єктна модель документа (Document Object Model або DOM).

AJAX—це не мова програмування, а метод, який використовує інші мови. Він використовується для створення інтерактивних сторінок, на яких обробляються дані без перезавантаження всієї сторінки. Також використовується при завантаженні даних із зовнішнього API, даних про товари, доступні в інтернет-магазині або продовження активної сесії користувача у фоновому режимі.

Основні елементи, включені до техніки AJAX:

- XML - це мова розмітки, яка використовується для опису отриманих даних. Однак часто виявляється, що інформація надсилається в іншому форматі, а приймається у текстовому вигляді. Йдеться про готові фрагменти коду HTML або JavaScript.

- JavaScript — не є обов'язковою умовою, оскільки замість нього можна використовувати іншу мову сценаріїв, таку як VBScript або JScript.
- XMLHttpRequest — дозволяє надсилати дані асинхронно. Таким чином, дані можуть бути вилучені з різних місць одночасно і користувач може виконувати інші дії під час передачі даних.

jQuery та AJAX є одними з найчастіше використовуваних методів, корисних для створення зручного інтерактивного середовища на веб-сайтах. Вони часто комбінуються один з одним, щоб програма працювала швидше і, отже, була більш привабливою для користувача.

PHP – це широко використовувана мова програмування на стороні сервера, яка з роками стає все швидшою та потужнішою. PHP добре працює з HTML, що робить її чудовою мовою для створення динамічного веб-сайту

2.4. Опис структури системи та алгоритми її функціонування

Структура сайту представляє з себе зв'язані між собою файли, де кожен файл виконує свою відповідну функцію.

Перед початком роботи на сайтом потрібно визначити чітку структуру сайту. Бо правильна структура позитивно впливає на сприйняття користувачем нашої сторінки.

Можна привести асоціацію, що веб-сайт схожий на книгу. Подібно книзі, сайт опублікован для загального користування людьми, сайт може містити різну кількість сторінок. Але ж відмінністю є те, що структура вмісту сайту може бути сформована най різноманітним способом, використовуючи свободу вираження поглядів та дизайну.

В той же час не потрібно забувати про основи UX та звички користувачів, тому свобода повинна мати свої межі. Структура веб-сайту є ключовою для того, яку інформацію використовує цільова аудиторія.

Коли створюється структура веб-сайту, потрібно думати про те, як відвідувачі використовуватимуть веб-сайт і орієнтуються в його вмісті. Фокус зміщений з того, що публікувати, на те, як і чому.

UX – це не про стилістичні рішення, кольору, шрифти і т.п.. Більшою мірою це про архітектуру сайту, його структуру, логічну зв'язку в різних блоках. Саме це формує так званий «користувальницький досвід» (звідси і абревіатура UX, від «User eXperience»).

Зовнішня структура сайту представлена на рис. 2.1 та рис. 2.2

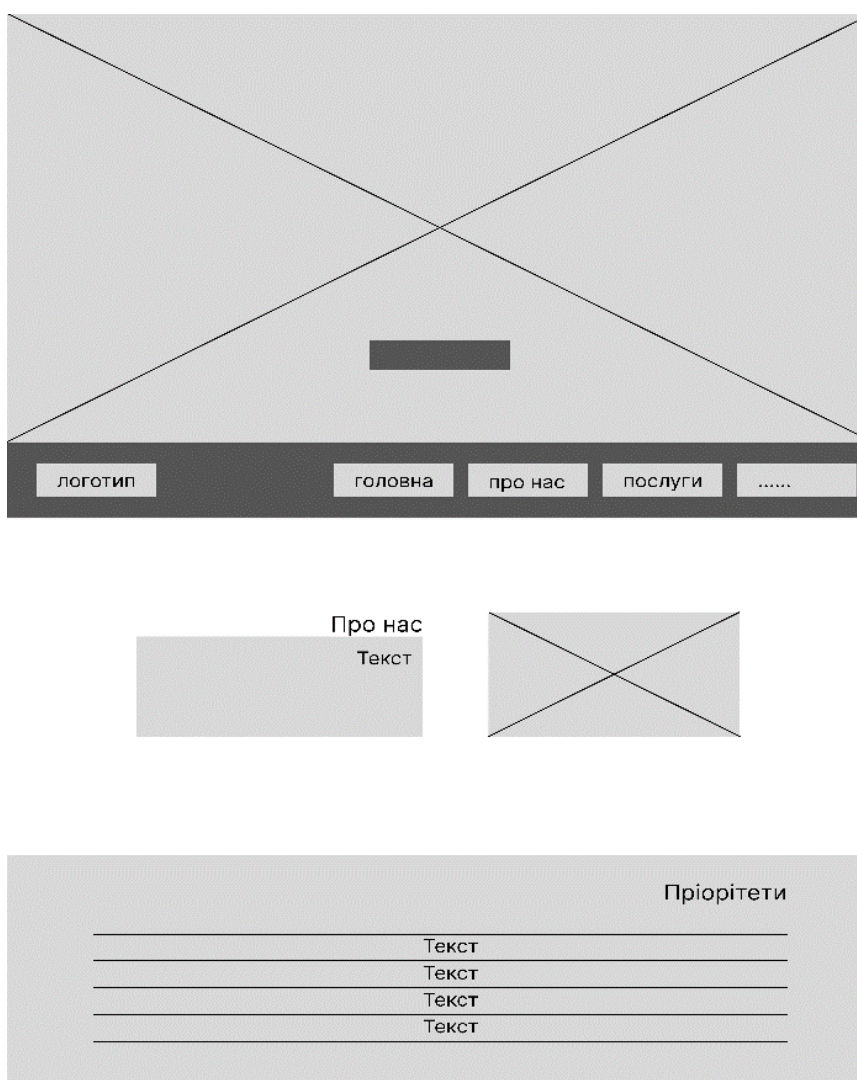


Рис. 2.1. Зовнішня структура веб-сайту

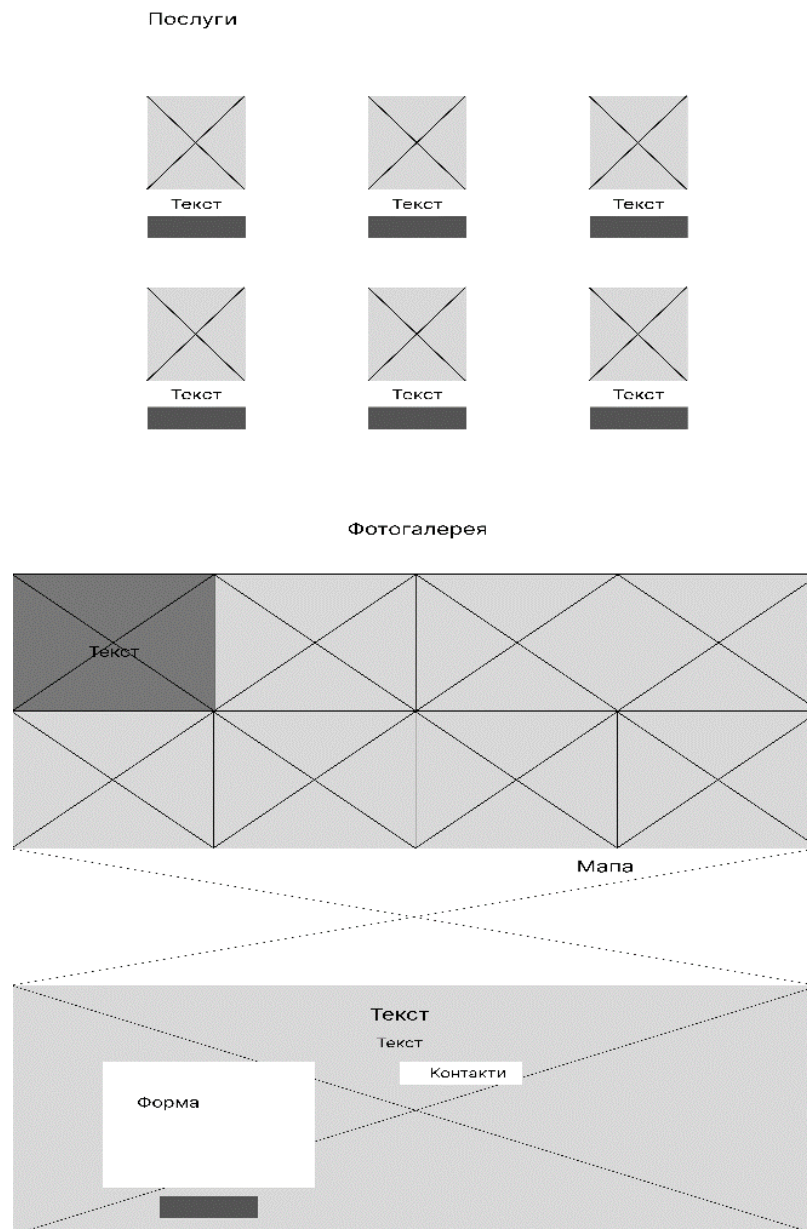


Рис. 2.2. Зовнішня структура веб-сайту

Перед початком розробки була створена мапа асоціацій, або як ще її називають Mind map. Вона має форму сітки, що складається з імен, ідей, слів — усього, що виходить з однієї спільної думки, це гарний інструмент для створення асоціативного ряду для полегшення розробки фірмового стилю компанії. Після її створення можна виділити кольорову палітру, якої далі буде притримуватися компанія, якщо її ще не було раніше, також можна виділити елементи на які можна спиратися при створенні логотипу і т.п.

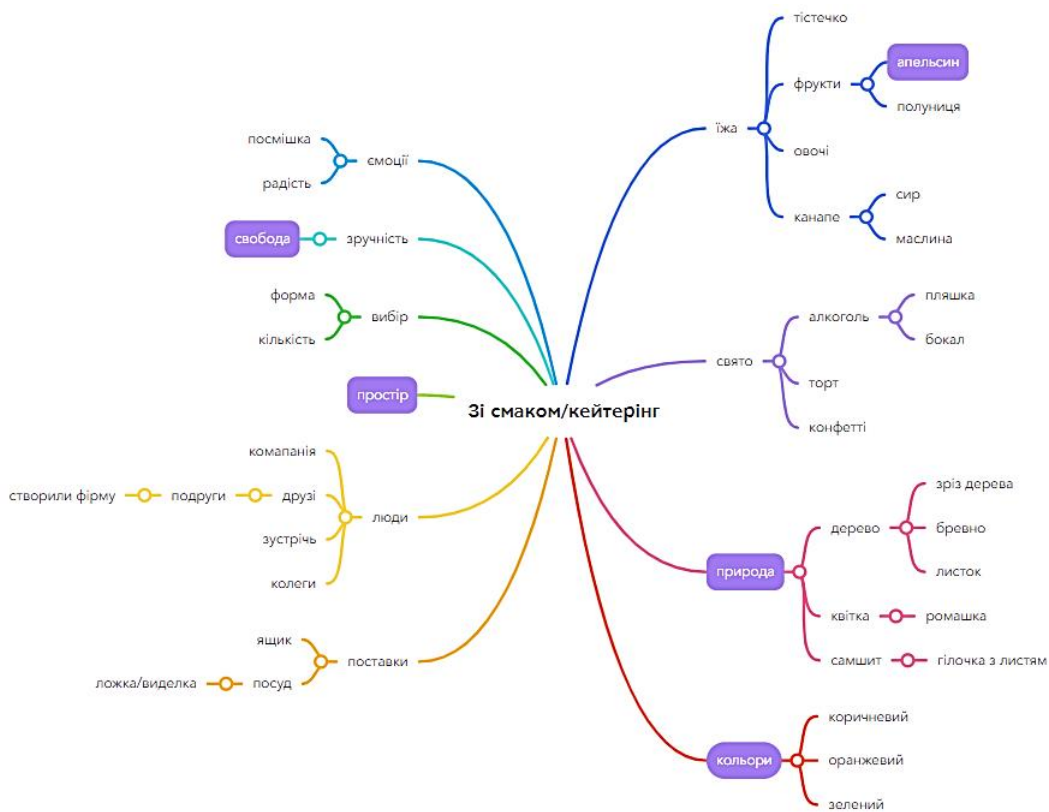


Рис. 2.3 Мапа асоціацій

Діаграма прецедентів відображає відносини між акторами і прецедентами системи, для того щоб описати систему на концептуальному рівні.

Діаграма прецедентів відображає відносини між акторами і прецедентами системи, для того щоб описати систему на концептуальному рівні.

Прецедент – це функціональна можливість системи, завдяки якій актор (користувач) може отримати конкретний, вимірний, потрібний для нього результат. Прецедент відповідає окремому сервісу системи, визначає один з можливих варіантів використання і описує спосіб взаємодії користувача з системою. Діаграма часто застосовуються для специфікації зовнішніх вимог до системи[8].

Нижче наведено дівграма прецедентів, на якій виділено такі дійові особи:

- "Користувач" - використовує систему для перегляду сайту з метою отримання інформації про послуги;
- «Адміністратор» - використовує систему для входу до адміністратора панель, маючи права на внесення будь-яких змін;
- «Менеджер» - використовує систему для обробки заявок.

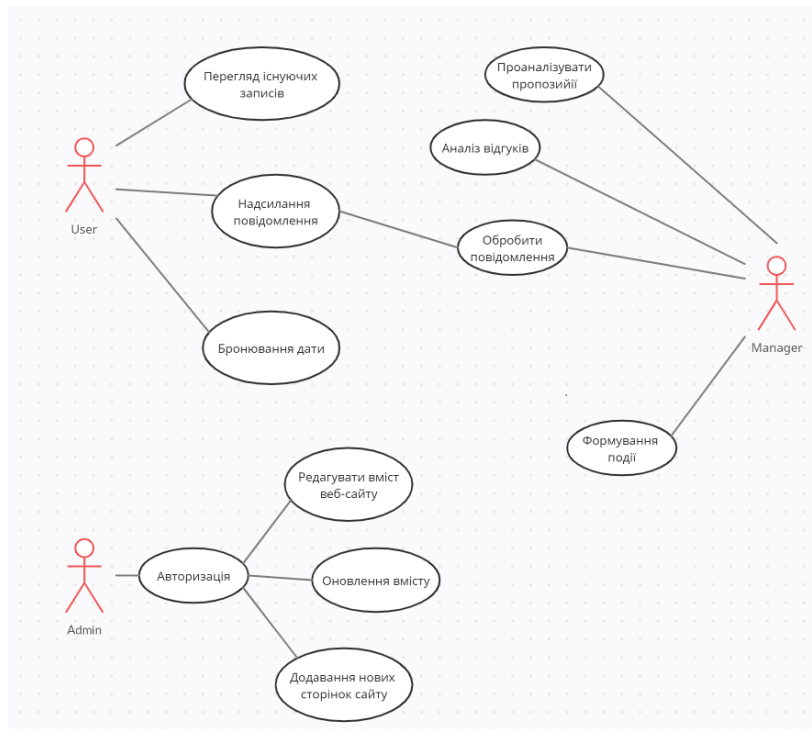


Рис. 2.4 Діаграма прецедентів

Діаграма станів описує поведінку окремого об'єкта у відповідь серію подій у системі. Призначенням даної діаграми є опис можливих послідовностей станів і переходів, які в загалом характеризують поведінку об'єкта протягом його життєвого циклу [9].

Діаграма станів, пов'язаних із діями користувача представлена на рисунку 2.5.

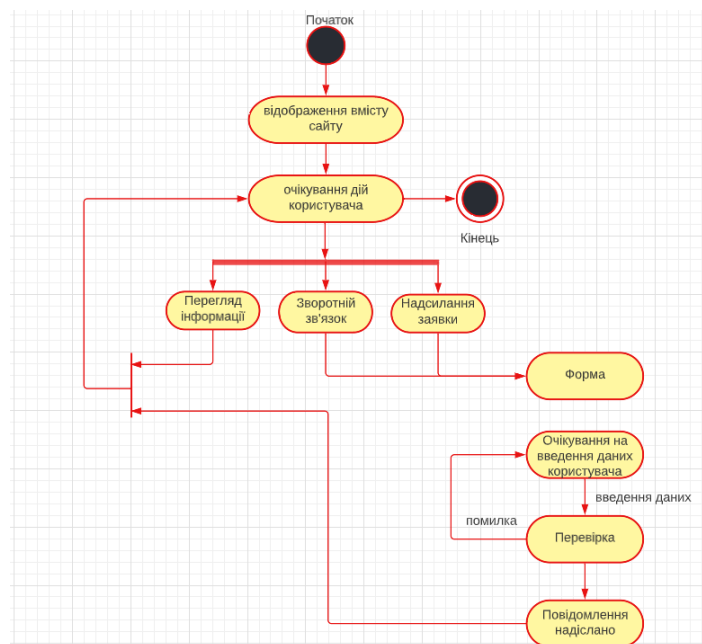


Рис. 2.5 Діаграма станів

Для зручності роботи с проектом, та подальшого його розширення, файли розташовані по відповідним їм папкам де кожна папка (рис. 2.3.) містить в собі файли з схожим за змістом файли, що відповідають за подібні задачі.

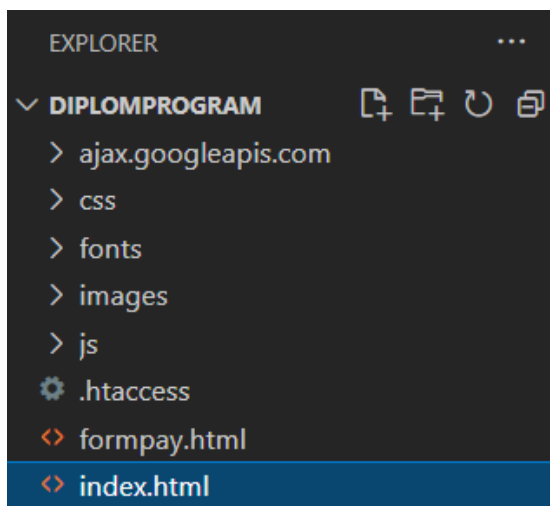


Рис. 2.6. Структура папок проекту

Папка `ajax.googleapis.com` (рис. 2.4.) містить в собі файл:

- `Jquery.min.js` - це бібліотека функцій, яка дозволяє дуже простим способом маніпулювати елементами DOM.

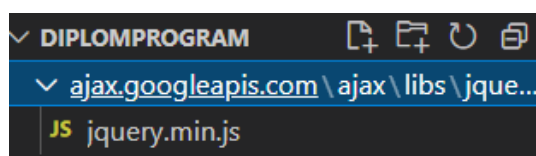


Рис. 2.7. Зміст папки `ajax.googleapis.com`

Папка `css` (рис. 2.5.) містить в собі всі файли зі стилями:

- `Animate.min.css` – файл, що відповідає за анімацію елементів на сайті;
- `Bootstrap.min.css` – файл, що відповідає за стилі фреймворка Bootstrap;
- `Daterangepicker.min.css` – файл, що відповідає за можливість, за допомогою якої можна додати input календар;

- Font-awesome.min.css – файл, що відповідає за колекцію маштабованих векторних іконок;
- Lightbox.css – файл, що відповідає за відображення зображення або будь-якого іншого контенту в спеціально оформленому вікні;
- Main.css – файл, що відповідає за основні стилі сайту;
- Responsive.css – файл, що відповідає за адаптивність сайту.

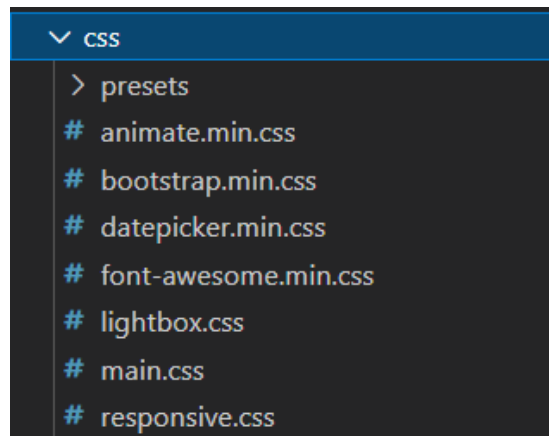


Рис.2.8. Зміст папки css

Папка fonts (рис. 2.6.) місти в собі шрифти:

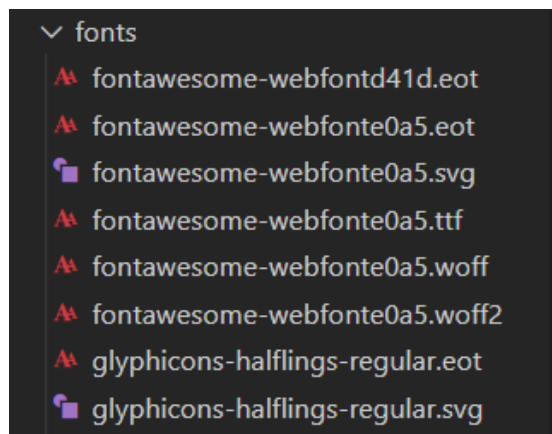


Рис. 2.9. Зміст папки fonts

Папка images (рис. 2.7.) містить в собі всі іконки та зображення, які використовувались на сайті:

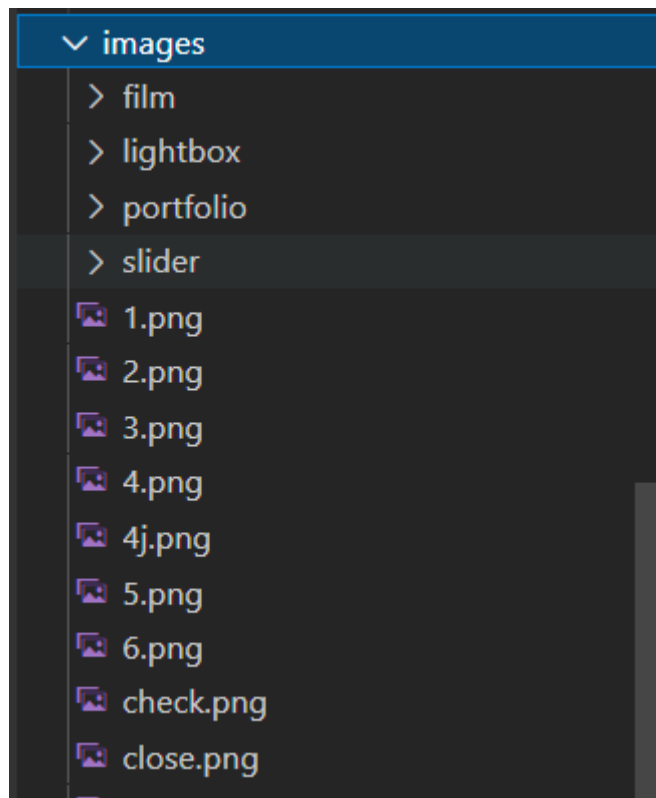


Рис. 2.10. Зміст папки images

Папка js (рис. 2.8) містить в собі файли, які відповідають за логіку на сайті:

- Bootstrap.min.js – файл, містить код javascript необхідні для виконання початкового завантаження;
- DatePicker.min.js – файл, що відповідає за створення календарю на сайті;
- JQuery.countTo.js – файл, що дає можливість запускати певний код, при прокручуванні до певних точок на екрані;
- JQuery.js – файл, що відповідає за JavaScript-бібліотеку, що фокусується на взаємодії HTML, CSS і JavaScript;
- Lightbox.min.js – файл, що містить плагін на jquery, який відкриває зображення на весь екран при натисканні;
- Main.js – файл, що відповідає за основну логіку на сайті.

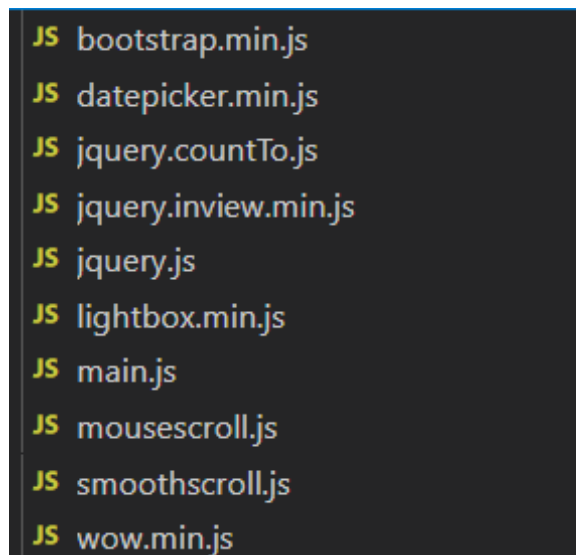


Рис. 2.11. Зміст папки js

Система починає працювати із головного файлу index.html (рис. 2.9.), що має наступний код.

```
index.html x # preset.css # main.css
index.html > html > body > div#popupFormBlock > div.formBlock-content > form
1 <!DOCTYPE html>
2 <html lang="en">
3 <meta charset="utf-8">
4 <meta name="viewport" content="width=device-width, initial-scale=1.0">
5 <meta name="description" content="">
6 <meta name="author" content="">
7 <title>зі смаком у Дніпрі</title>
8 <link href="css/bootstrap.min.css" rel="stylesheet">
9 <link href="css/animate.min.css" rel="stylesheet">
10 <link href="css/font-awesome.min.css" rel="stylesheet">
11 <link href="css/lightbox.css" rel="stylesheet">
12 <link href="css/main.css" rel="stylesheet">
13 <link id="css-preset" href="css/presets/preset1.css" rel="stylesheet">
14 <link href="css/responsive.css" rel="stylesheet">
15 <link href="css/datepicker.min.css" rel="stylesheet" type="text/css">
16 <!--[if lt IE 9]
17 <script src="js/html5shiv.js"></script>
18 <script src="js/respond.min.js"></script>
19 <![endif]-->
20 <link href="http://fonts.googleapis.com/css?family=Open+Sans:300,400,600,700" rel="stylesheet" type="text/css">
21 <link rel="shortcut icon" href="images/fav.png">
22
23 <body>
24 <div class="preloader"><i class="fa fa-circle-o-notch fa-spin"></i></div>
25
26 <header id="home">
27 <div id="home-slider" class="carousel slide carousel-fade" data-ride="carousel">
28 <div class="carousel-inner">
29 <div class="item active" style="background-image: url(images/slider/1.jpg)">
30 <div class="caption">
31 <h1 class="animated fadeInLeftBig">Кейтеринг <span>в Дніпрі</span></h1>
32 <p class="animated fadeInRightBig">Створюємо смак ваших подій</p>
33 <a data-scroll class="btn btn-start animated fadeInUpBig" href="#services">Розрахувати подію <br></a>
34 </div>
35 </div>
36 <div class="item" style="background-image: url(images/slider/3.jpg)">
37 <div class="caption">
38 <h1 class="animated fadeInLeftBig">Кейтеринг <span>в Дніпрі</span></h1>
```

Рис. 2.12. Зміст файлу index.html

2.5. Обґрунтування та організація вхідних та вихідних даних програми

В сайті використовується текстовий формат з кодуванням UTF-8 обміну даними.

В даний час існує значна кількість різних форматів, рекомендованих в літературі для використання в розподілених інформаційних системах. Найчастіше в співтоваристві розробників, перевагу віддають одному з трьох найбільш використовуваних форматів обміну даними: XML, JSON, YAML.

2.6. Опис розробленої системи

Розроблений програмний продукт являє собою веб додаток для роботи з яким достатньо мати ПК з мінімальним набором характеристик. Для підтримки додатку та впровадження нових функцій потрібно середовище програмування у вигляді довільного текстового редактору.

2.6.1. Використані технічні засоби

Під час розробки веб-додатку був використаний ноутбук фірми Асер, модель Aspire 5 A515-52. Даний комп'ютер має такі характеристики:

- процесор Intel Core i5-8265U;
- діагональ екрану 15.6 "(1920x1080);
- об'єм оперативної пам'яті 8 ГБ;
- операційна система Windows 10;
- обсяг накопичувача 512 ГБ;
- вага 1,80 кг;
- тип оперативної пам'яті DDR4 SDRAM.

2.6.2. Використані програмні засоби

Під час розробки кваліфікаційної роботи були використані наступні програмні засоби:

- IDE для розробки – Visual Studio Code;
- Веб-браузер – Google Chrome;

- Графічний редактор – Figma.

2.6.3. Виклик та завантаження програми

Для виклику та завантаження необхідно виконати наступні дії:

- необхідно орендувати або придбати серверне обладнання та доменне ім'я;
- встановити на серверне обладнання одне з серверного забезпечення (Apache або Nginx);
- налаштувати серверне програмне забезпечення;
- за допомогою FTP клієнта приєднатися до сервера та перемістити у папку
- src усі скомпільовані файли програми;
- підключити доменне ім'я та сертифікати до серверу;
- ввімкнути сервер.

2.6.4. Опис інтерфейсу користувача

При завантаженні сайту в браузері відкривається головна сторінка на якій розташовані елементи. Шапка сайту містить логотип та головне меню, а також анімований слайдер с фотографіями, які мають зацікавити користувача (рис 2.13.).

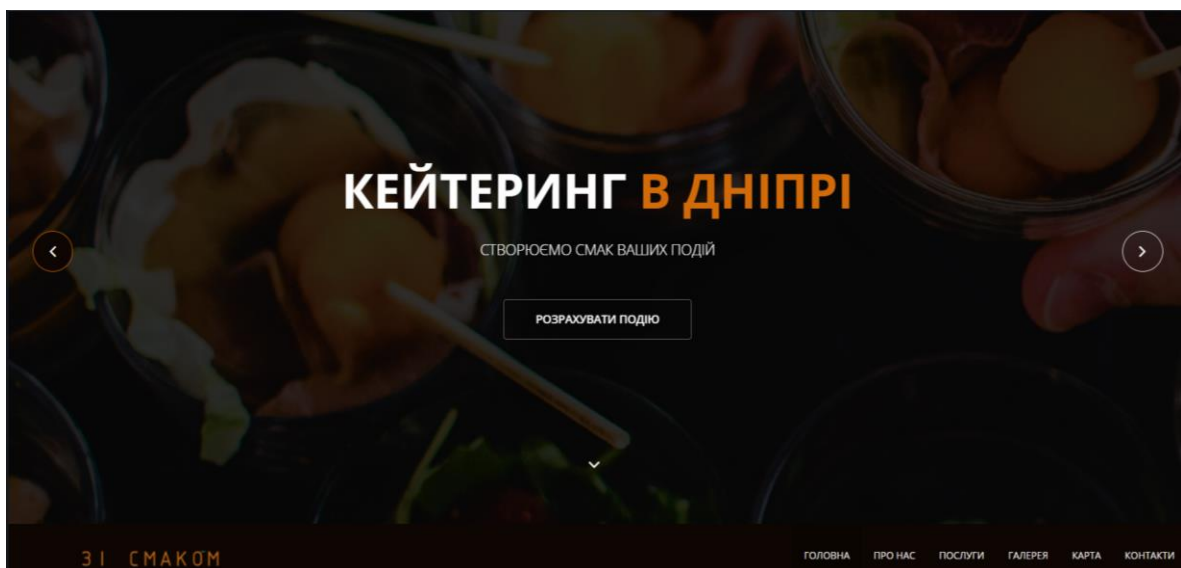


Рис. 2.13. Головна сторінка сайту

При натисканні на кнопку «Про Нас» ми перейдемо к блоку із інформацією про кінотеатр (Рис. 2.14.) та пріоритетами (Рис. 2.15.).

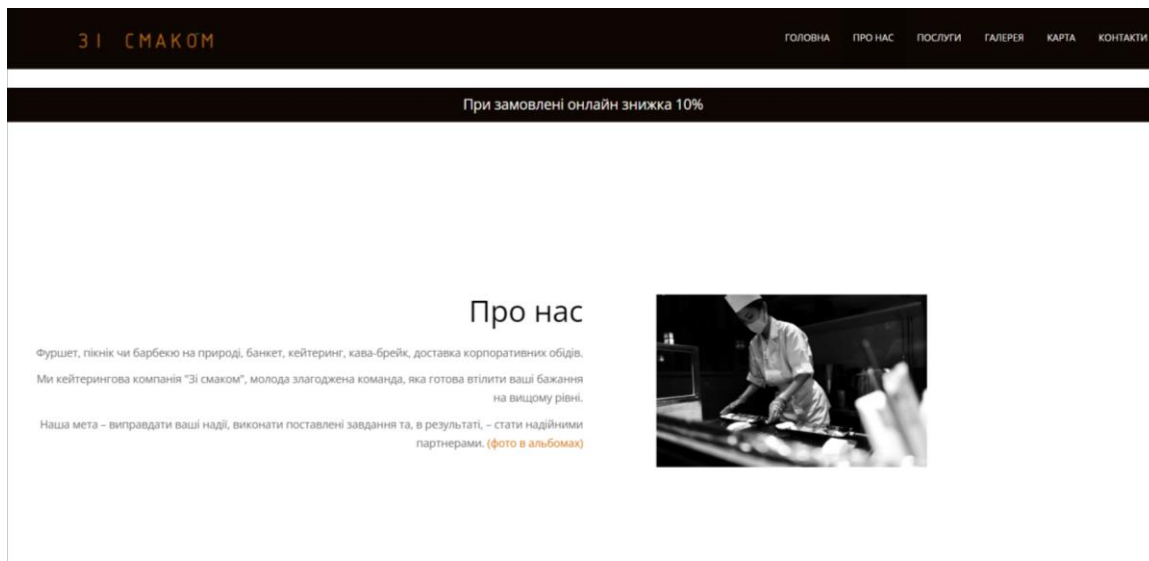


Рис. 2.14. Блок «Про нас»

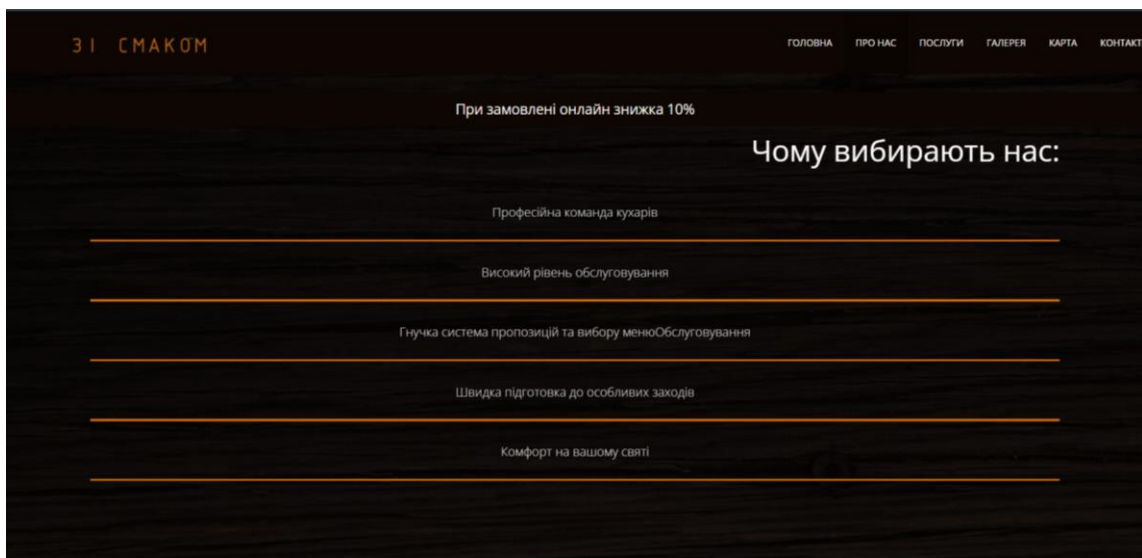


Рис. 2.15. Блок «Про нас»

Просунувшись далі по сайту або натиснувши на «Послуги» пререйдемо до блоку де представлено послуги компанії з активною кнопкою «Замовити онлайн» (Рис. 2.16.). Також до цього блоку можна перейти при натисканні на поле з текстом «При замовленні онлайн знижка 10%», це повідомлення супроводжує потенційного

клієнта увесь час при пересуванню по сайту. Воно слугує для здійснення цільової дії на сайті, а саме зробити замовлення.

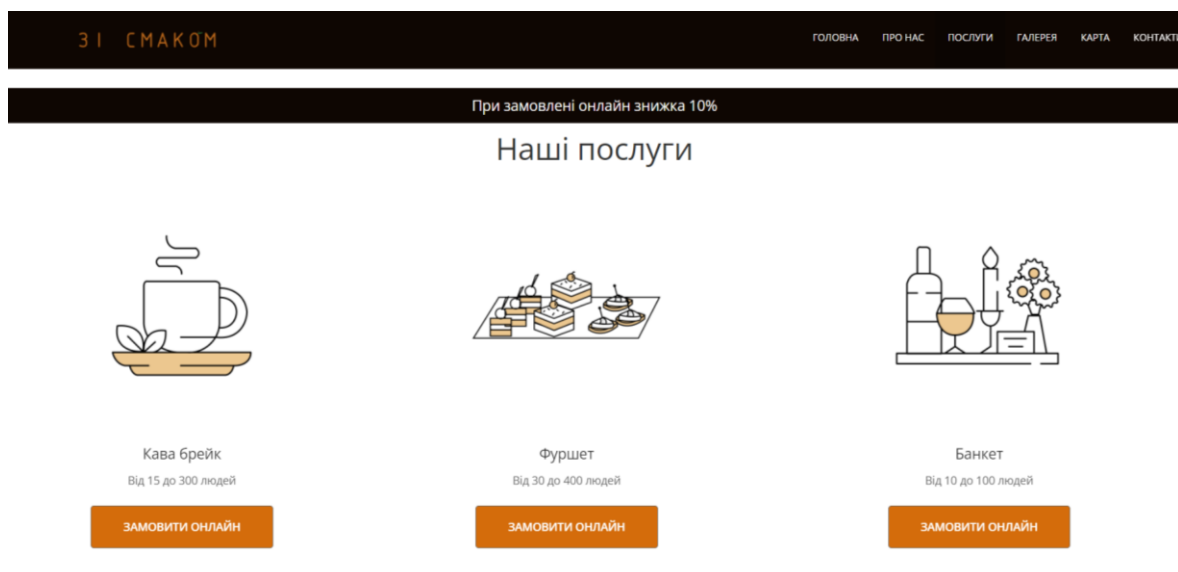


Рис. 2.16. Блок «Послуги»

Щоб забронювати дату та надіслати свої контактні дані треба натиснути на кнопку «Замовити онлайн». У нас з'являється форма зворотного зв'язку та календар з можливістю вибору дати (Рис. 2.17.).

3 | СМАКОМ

ГОЛОВНА ПРО НАС ПОСЛУГИ ГАЛЕРЕЯ КАРТА КОНТАКТИ

Оберіть дату: Телефон +38

< **Июнь, 2022** >

ПН	ВТ	СР	ЧТ	ПТ	СБ	ВС
30	31	1	2	3	4	5
6	7	8	9	10	11	12
13	14	15	16	17	18	19
20	21	22	23	24	25	26
27	28	29	30	1	2	3

Имя

Email-адреса

Ваше повідомлення

Натискаючи на кнопку, я приймаю Пользовательского соглашения и даю свое согласие на обработку моих персональных данных

Відправити

Рис. 2.17. Форма замовлення

У формі можна вибрати зручну для користувача дату, натиснувши на календар. Ввести свої персональні дані, такі як ім'я, номер телефону та пошту, для

можливості подальшого формування замовлення. Також блок для введення повідомлення. При натисканні на поле з'являються підказки, для зручності користувача (Рис. 2.18.)

The screenshot shows a web form titled "ЗІ СМАКОМ" with a navigation bar containing "ГОЛОВНА", "ПРО НАС", "ПОСЛУГИ", "ГАЛЕРЕЯ", "КАРТА", and "КОНТАКТ". The form includes a date picker set to "Июнь, 2022" with the 16th highlighted. A phone number field contains "+38". Below is a text input field for a name, with "Дар'я" entered and a dropdown menu showing "Дар'я". A message field contains "Ваше повідомлення (не менше 20 символів)". A checkbox is checked, with the text "Натискаючи на кнопку, я приймаю Пользовательского соглашения и даю своё согласие на обработку моих персональных данных". An orange "Відправити" button is at the bottom.

Рис. 2.18. Відображення підказок

Після заповнення форми (Рис. 2.19.) треба натиснути на кнопку «Відправити» при успішному надісланні з'явиться повідомлення (Рис. 2.20.),

The screenshot shows the same web form as in Fig. 2.18, but now filled out. The date picker is still set to "Июнь, 2022" with the 16th highlighted. The phone number field contains "+38 (098) 989-8989". The name field contains "Дар'я". The email field contains "pieshkova.d.y@nmu.one". The message field contains "Доброго дня! Маю...". The checkbox is checked, with the text "Натискаючи на кнопку, я приймаю Пользовательского соглашения и даю своё согласие на обработку моих персональных данных". An orange "Відправити" button is at the bottom.

Рис. 2.19. Заповнена форма

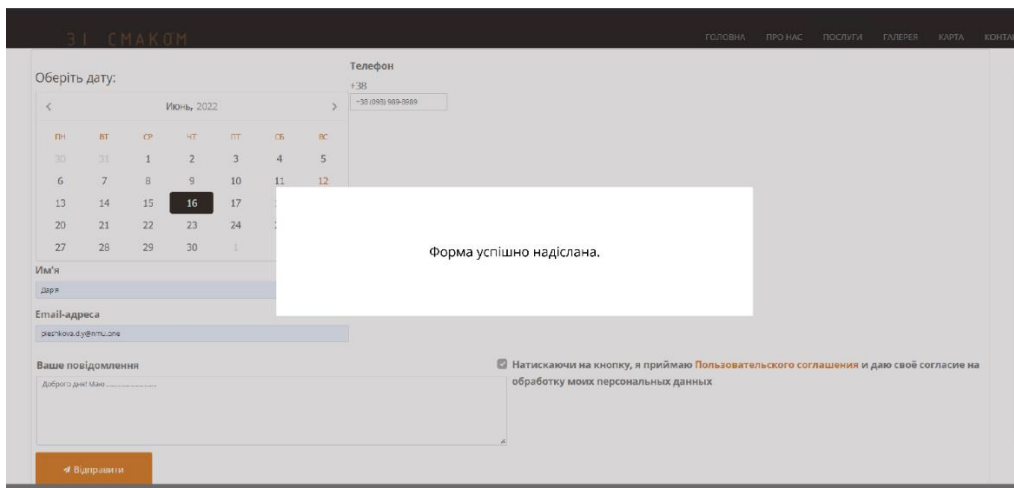


Рис. 2.20 Вікно повідомлення про успішно надіслану форму

Також на сайті присутній блок з фотографіями деяких подій. Натискаємо на кнопку «Галерея» (Рис. 2.21.).

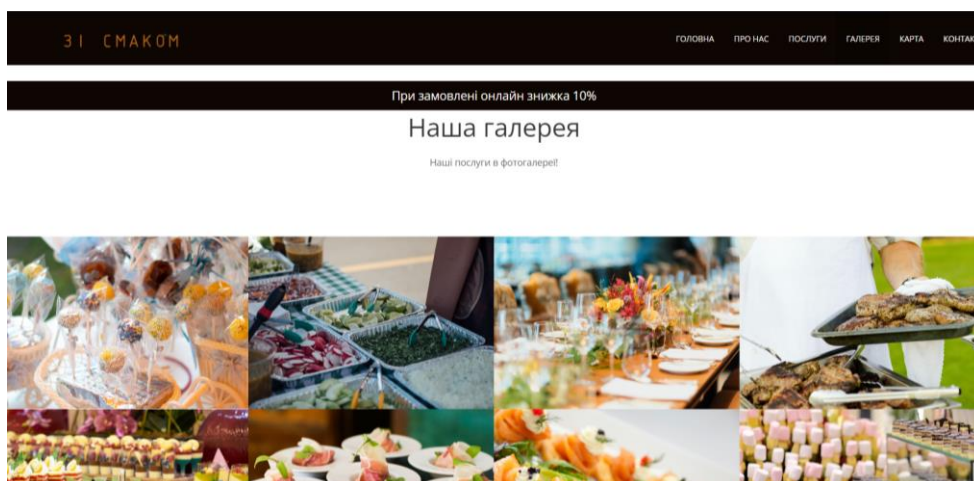


Рис. 2.21 Блок «Галерея»



Рис. 2.22. Блок «Галерея»

При натисканні на зображення можна переглянути фоткартку у збільшеному розмірі (Рис. 2.23.)

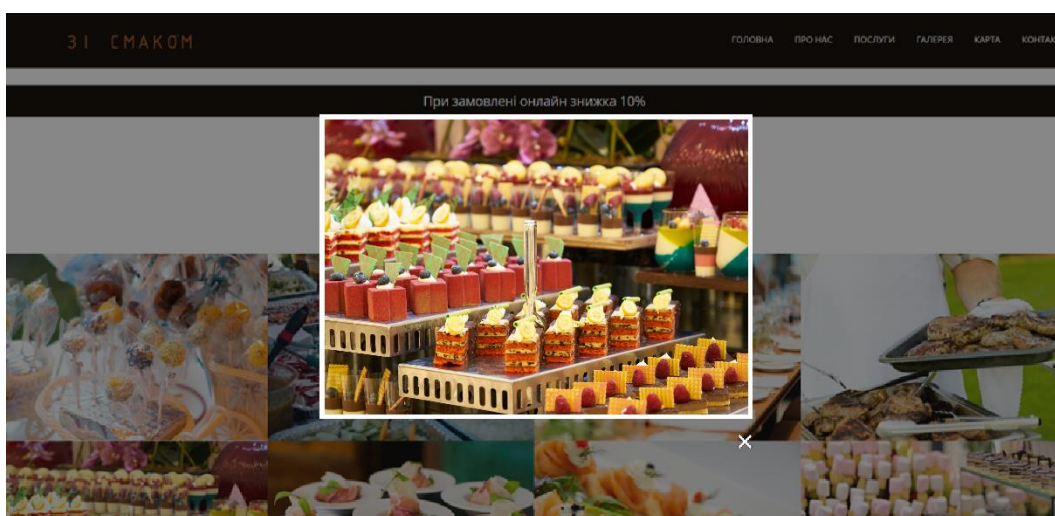


Рис. 2.23. Збільшене зображення

При натисканні на кнопку «Карта» ми перейдемо до блоку з інтерактивною картою з позначеним розташуванням компанії. (Рис. 2.24.).

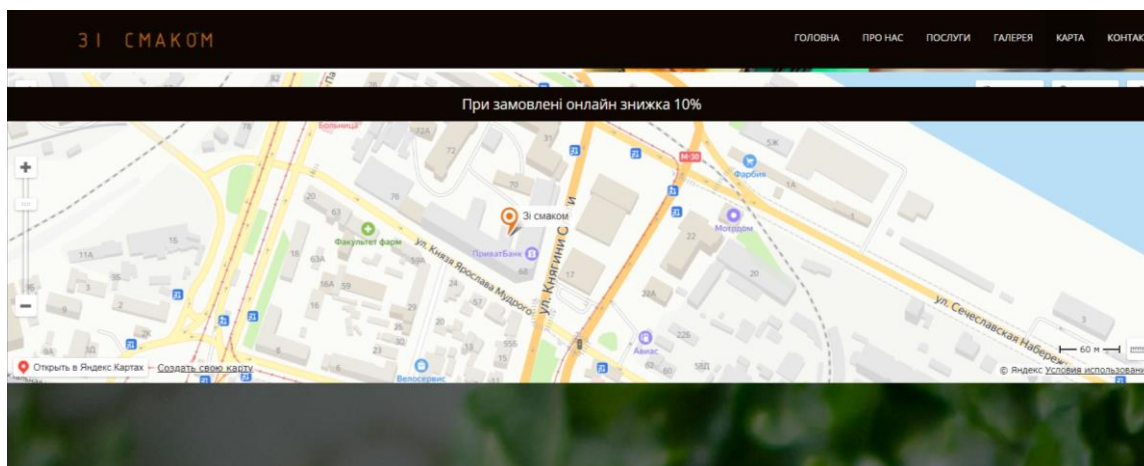


Рис. 2.24. Карта

Пункт меню «Контакти» головного меню переводить до блоку з контактними даними (Рис. 2.25.).

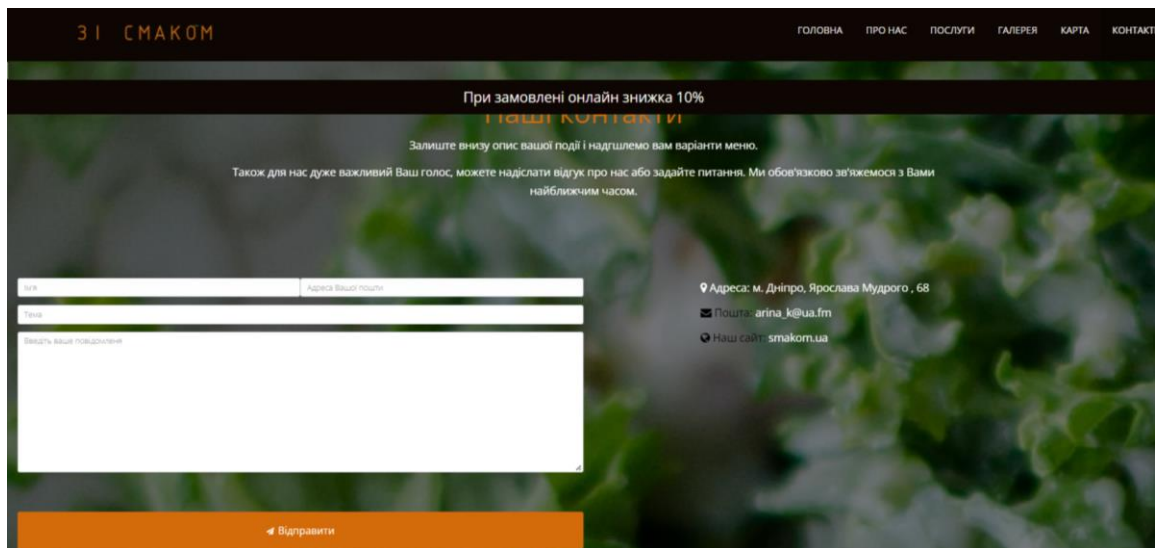


Рисунок 2.25. Контактні дані

Отже, розроблений сайт є інформативним, зручним в користуванні з яскравим та зрозумілим інтерфейсом.

РОЗДІЛ 3 ЕКОНОМІЧНИЙ РОЗДІЛ

3.1 Розрахунок трудомісткості та вартості розробки програмного продукту.

Вихідні дані:

1. передбачуване число операторів програми – 2755;
2. коефіцієнт складності програми – 1,3;
3. коефіцієнт корекції програми в ході її розробки – 0,05;
4. годинна заробітна плата програміста – 85 грн/год;

Досвід роботи з технологіями, що використовувались у кваліфікаційному проекті, оцінюється на ринку праці України приблизно у 15 000 грн помісячно, що дорівнює 85 грн/год [6].

5. коефіцієнт збільшення витрат праці в наслідок недостатнього опису задачі – 1,2;
6. коефіцієнт кваліфікації програміста, обумовлений від стажу роботи з даної спеціальності – 1,0;
7. вартість машино-годин ЕОМ – 20 грн/год.

Нормування праці в процесі створення ПЗ істотно ускладнено в силу творчого характеру праці програміста. Тому трудомісткість розробки ПЗ може бути розрахована на основі системи моделей з різною точністю оцінки. Трудомісткість розробки програмного забезпечення розраховується за наведеною формулою:

$$t = t_o + t_u + t_a + t_n + t_{отл} + t_d, \text{ людино/годин, (3.1)}$$

t_o – витрати праці на підготовку й опис поставленої задачі (60
людино/годин);

t_u – витрати праці на дослідження алгоритму рішення задачі;

t_a – витрати праці на розробку блок-схеми алгоритму;

t_n – витрати праці на програмування по готовій блок-схемі;

$t_{отл}$ – витрати праці на налагодження ЕОМ;

t_d – витрати праці на підготовку документації.

Умовне число операторів(підпрограм):

$$Q = q \cdot C \cdot (1 + p)$$

q – передбачуване число операторів (1947);

C – коефіцієнт складності програми (1,5);

p – коефіцієнт корекції програми в ході її розробки (0,07).

Розрахунок умовного числа операторів:

$$Q = 1,3 * 2755 * (1 + 0,05) = 3\,760,575 \text{ людино/години,}$$

Витрати праці на дослідження алгоритму рішення задачі:

$$t_u = \frac{Q \cdot B}{(75..85) \cdot k}, \text{ людино-годин,}$$

B – коефіцієнт збільшення витрат праці внаслідок недостатнього опису задачі

k – коефіцієнт кваліфікації програміста, обумовлений від стажу роботи зданої спеціальності. З досвідом роботи від 2 до 3 років він дорівнює 1,0, а при стажі роботи від 5 до 8 років він складає 1,4. З урахуванням коефіцієнта кваліфікації $k = 1,0$, отримуємо витрати праці на вивчення опису завдання:

$$t_u = \frac{3\,760,575 \cdot 1,2}{80 \cdot 1,0} = 55,712 \text{ людино/годин.}$$

Витрати праці на розробку алгоритму рішення задачі визначаються за формулою:

$$t_a = \frac{Q}{(20 \dots 25) \cdot k}, \text{ людино-годин, (3.2)}$$

де Q – умовне число операторів програми;

k – коефіцієнт кваліфікації програміста.

Підставивши відповідні значення в формулу (3.2), отримаємо:

$$t_a = 3760,575 / (22 \cdot 1,0) = 170,935 \text{ людино-годин.}$$

Витрати на програмування по завершеній блок-схемі:

$$t_n = \frac{Q}{(20 \dots 25) \cdot k}, \text{ людино/годин.}$$

$$t_n = \frac{3\,760,575}{23 \cdot 1,0} = 163,503 \text{ людино/годин.}$$

Витрати праці на налагодження програми на ЕОМ:

- за умови автономного налагодження одного завдання:

$$t_{oml} = \frac{Q}{(4..5) \cdot k}, \text{ людино-годин.}$$

$$t_{oml} = 3760,575 / (4,1 \cdot 1,0) = 917,2 \text{ людино-годин.}$$

Витрати праці на підготовку документації можна визначити за формулою:

$$t_{\partial} = t_{\partial p} + t_{\partial o} , \text{ людино-годин,}$$

$t_{\partial p}$ – трудомісткість підготовки матеріалів і рукопису:

$$t_{\partial p} = \frac{Q}{(15..20) \cdot k} , \text{ людино-годин,}$$

$t_{\partial o}$ – трудомісткість редагування, печатки й оформлення документації:

$$t_{\partial o} = 0,75 \cdot t_{\partial p} , \text{ людино-годин.}$$

Розраховуючи за наведеними формулами, отримуємо:

$$t_{\partial p} = 3760,575 / (16 \cdot 1,0) = 235,096 \text{ людино-годин.}$$

$$t_{\partial o} = 0,75 \cdot 235,096 = 176,276 \text{ людино-годин.}$$

$$t_{\partial} = 235,096 + 176,276 = 411,312 \text{ людино-годин.}$$

Повертаючись до формули (3.1), отримаємо повну оцінку трудомісткості розробки програмного забезпечення:

$$t = 60 + 55,712 + 170,935 + 163,503 + 917,2 + 411,312 = 1\,778,662 \text{ людино/годин.}$$

3.2. Розрахунок витрат на створення програми

Витрати на створення ПЗ КПО включають витрати на заробітну плату виконавця програми ЗЗП і витрат машинного часу, необхідного на налагодження програми на ЕОМ:

$$K_{\text{ПО}} = Z_{\text{ЗП}} + Z_{\text{МВ}}, \text{ грн.}$$

Заробітна плата виконавців визначається за формулою:

$$Z_{\text{ЗП}} = t * C_{\text{пр}}, \text{ грн.}$$

де t – загальна трудомісткість, людино/годин;

$C_{\text{пр}}$ – середня годинна заробітна плата програміста, грн/година.

З урахуванням того, що середня годинна зарплата програміста становить 85 грн / год, отримуємо:

$$Z_{\text{ЗП}} = 1\,778,662 * 85 = 151\,186,27 \text{ грн.}$$

Вартість машинного часу для налагодження програми на електронно-обчислювальній машині:

$$Z_{\text{МВ}} = t_{\text{отл}} * C_{\text{мч}}, \text{ грн. (3.3)}$$

$C_{\text{мч}}$ – вартість машино-годин ЕОМ, грн/год;

$t_{\text{отл}}$ – трудомісткість налагодження програми на ЕОМ за умови комплексного налагодження завдання:

$$t^k_{\text{отл}} = 1,5 * t \quad , \text{ людино/годин.}$$

$$t^k_{\text{отл}} = 1,5 * 917,2 = 1375,8 \quad , \text{ людино/годин}$$

Підставивши в формулу (3.3) відповідні значення, визначимо вартість необхідного для налагодження машинного часу:

$$З_{\text{МВ}} = 1375,8 * 20 = 27\,516 \text{ грн.}$$

Загальні витрати на створення програмного продукту будуть мати значення:

$$K_{\text{ПО}} = 151\,186,27 + 27\,516 = 178\,702,27 \text{ грн.}$$

Очікуваний період створення програмного забезпечення розраховується за формулою:

$$T = \frac{t}{B_k * F_p}, \text{ міс.}$$

де B_k – число виконавців (дорівнює 1);

F_p – місячний фонд робочого часу (при 40 годинному робочому тижні).

Звідси витрати на створення програмного продукту:

$$T = \frac{1\,269}{1 * 176} = 7,2 \text{ місяці.}$$

Висновок: Інформаційна система була розроблена для забезпечення доступу користувачів до основних пропозицій підприємства, яке надає послуги кейтерінгу, ефективної взаємодії між потенційними споживачами та компанією, створення позитивного іміджу та підвищення продажу товарів.

В процесі обчислення економічної частини кваліфікаційного проекту було отримано, що вартість даного програмного забезпечення близько 178 702 тис. грн і не вимагає додаткових витрат як при розробці програми. Очікуваний час розробки становить 1269 годин, тобто 7,2 місяці. Цей термін пов'язаний зі значним числом операторів, і включає час на дослідження і розробку алгоритму вирішення поставленого завдання, програмування по готовому алгоритму, налагодження програми і підготовку документації.

ВИСНОВКИ

В ході виконання кваліфікаційної роботи був реалізований повнофункціональний web-сайт для кейтерингової компанії, повністю готовий до застосування. Даний сайт орієнтований на клієнтів, що шукають виконавців для надання послуг з кейтерингу. З його допомогою користувачі зможуть отримувати необхідну інформацію, задавати питання в формах і впливати як саме буде виглядати їх подія..

В процесі виконання роботи були вирішені наступні задачі:

- розглянуті і застосовані сучасні технології створення Web-сайтів;
- вивчено сучасне програмне забезпечення, що застосовується для створення Web-сайтів;
- розглянуті різні методи і способи розміщення на web-сторінках різноманітної інформації;
- розглянуті і застосовані основні правила і рекомендації порозробці та створенню Web-сайтів;
- визначена структура і вміст Web-сторінок сайту;
- проведено аналіз працездатності Web-сайту.

Розроблений сайт задовольняє всі вимоги, поставлені на етапі постановки завдання. При розробці web-сайту були використані готові модулі, форм. Дані модулі були доопрацьовані з урахуванням специфіки web-сайту і успішно впроваджені в його структуру.

У розробленому web-сайті кейтерингової компанії вдало поєднується приємний інтерфейс з його функціональністю і простотою використання.

Виходячи з матеріалів кваліфікаційної роботи можна впевнено сказати, що поставлені завдання повністю вирішені. Мета кваліфікаційної роботи досягнута.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Бізнес в інтернеті [Електроній ресурс]/ – Режим доступу до журн.: <https://uk.wikipedia.org/wiki/>
2. Динамічні сайти. [Електроній ресурс]/ – [https://uk.wikipedia.org/wiki/Динамічні сайти](https://uk.wikipedia.org/wiki/Динамічні_сайти)
3. CSS [Електроній ресурс]/ – Режим доступу до журн.: https://www.tutorialspoint.com/css/what_is_css.htm
4. Створюємо динамічні веб-сайти за допомогою PHP, MySQL, JavaScript, CSS та HTML5. 4-те вид. - СПб.: Пітер, 2016. - 768 с.:
5. Дронов В. А.. HTML 5, CSS 3 та Web 2.0. Розробка сучасних веб-сайтів. - СПб.:БХВ-Петербург, 2011. - 416 с.:
6. Ваканція розробника[Електроній ресурс]/ – Режим доступу до журн.: <https://www.work.ua/jobs/4223324/>
7. Каскадність CSS: визначаємо пріоритет стилю, коли потрібен тег [Електронійм ресурс]/ –<https://codernet.ru/articles/drugoe/>
8. Проектування програмного забезпечення [Електроній ресурс]/ – <https://habr.com/ru/post/74330/>
9. UML Діаграми. URL [Електроній ресурс]/ – <https://infopedia.su/10x641c.html>
10. Пасічник Н.Р. Формалізм в постановці задачі створення якісного сайту / Н.Р. Пасічник // Наукові праці Донецького національного технічного університету. Інформатика, кібернетика та обчислювальна техніка. – Донецьк. – 2017. – Вип. 14 (188). – С. 325-329.
11. 10 MDN web docs [Електроній ресурс] JavaScript / - Режим доступу до журн. : <https://developer.mozilla.org/docs/Web/JavaScript>.
12. Пауерс Девід. Adobe Dreamweaver, CSS, Аjax та PHP / Девід Пауерс. - Угород: Карпати, 2019. - 829 с.
13. Прохоренко Н. А. HTML, JavaScript, PHP та MySQL. Джентльменський набір Web-майстра/Н.А. Прохоренко, В.А. Дронов. – Харків: Фоліо, 2017. – 768 с.

14. Пфаффенбергер HTML, XHTML та CSS. Біблія користувача / Пфаффенбергер та ін. - Тернопіль: Підручники і посібники; Видання 3-тє, 2019. - 752 с.
15. Створи свій веб-сайт за допомогою HTML та CSS. - Львів: Світ, 2020. - 569 с.
16. Ташков Петро. Веб-мастеринг HTML, CSS, JavaScript, PHP, CMS, AJAX, розкрутка / Петро Ташков. – Житомир: Книга на вимогу, 2018. – 512 с.
17. Тіттел Ед HTML, XHTML та CSS для чайників / Ед Тіттел, Джефф Ноубл. – Вінниця.: Діалектика, 2020. – 400 с.
18. Фрімен Елізабет Вивчаємо HTML, XHTML та CSS / Елізабет Фрімен, Ерік Фрімен. – Дніпро: Прес, 2016. – 720 с.
19. Хольцшлаг Моллі Е.. Використання HTML 4. Спеціальне видання (+ CD - ROM) / Хольцшлаг Моллі Е.. - Київ: Україна, 2017. - 174 с.
20. Чебикін Ростислав. Самовчитель HTML та CSS. Сучасні технології / Ростислав Чебикін. - Харків: Вогні, 2018. - 624 с.
21. Шафер Стівен HTML, XHTML та CSS. Біблія користувача / Стівен Шафер. – Київ: СИНТЕГ, 2019. – 656 с.

КОД ПРОГРАМИ

Index.html:

```

<!DOCTYPE html>
<html lang="en">
<meta charset="utf-8">
<meta name="viewport" content="width=device-width, initial-scale=1.0">
<meta name="description" content="">
<meta name="author" content="">
<title>Зі смаком у Дніпрі</title>
<link href="css/bootstrap.min.css" rel="stylesheet">
<link href="css/animate.min.css" rel="stylesheet">
<link href="css/font-awesome.min.css" rel="stylesheet">
<link href="css/lightbox.css" rel="stylesheet">
<link href="css/main.css" rel="stylesheet">
<link id="css-preset" href="css/presets/preset1.css" rel="stylesheet">
<link href="css/responsive.css" rel="stylesheet">
<link href="css/datepicker.min.css" rel="stylesheet" type="text/css">
<link href="css/form-processing.css" rel="stylesheet">
<!--[if lt IE 9]>
  <script src="js/html5shiv.js"></script>
  <script src="js/respond.min.js"></script>
<![endif]-->
<link href='http://fonts.googleapis.com/css?family=Open+Sans:300,400,600,700' rel='stylesheet' type='text/css'>
<link rel="shortcut icon" href="images/fav.png">

<body>
  <div class="preloader"><i class="fa fa-circle-o-notch fa-spin"></i></div>

  <header id="home">
    <div id="home-slider" class="carousel slide carousel-fade" data-ride="carousel">
      <div class="carousel-inner">
        <div class="item active" style="background-image: url(images/slider/1.jpg)">
          <div class="caption">
            <h1 class="animated fadeInLeftBig">Кейтеринг <span>в Дніпрі</span></h1>
            <p class="animated fadeInRightBig">Створюємо смак ваших подій</p>
            <a data-scroll class="btn btn-start animated fadeInUpBig" href="#services">Розрахувати подію
          <br></a>
        </div>
      </div>
    </div>
    <div class="item" style="background-image: url(images/slider/3.jpg)">
      <div class="caption">
        <h1 class="animated fadeInLeftBig">Кейтеринг <span>в Дніпрі</span></h1>
        <p class="animated fadeInRightBig">Створюємо смак ваших подій</p>
        <a data-scroll class="btn btn-start animated fadeInUpBig" href="#services">Розрахувати подію
      <br></a>
    </div>
  </header>

```

```

        </div>
    </div>
</div>
<a class="left-control" href="#home-slider" data-slide="prev"><i class="fa fa-angle-left"></i></a>
<a class="right-control" href="#home-slider" data-slide="next"><i class="fa fa-angle-right"></i></a>

<a id="tohash" href="#about-us"><i class="fa fa-angle-down"></i></a>

</div>
<!-- Логотип в хетері -->
<div class="main-nav">
    <div class="navbar-header">
        <button type="button" class="navbar-toggle" data-toggle="collapse" data-target=".navbar-collapse">
            <span class="sr-only">Toggle navigation</span>
            <span class="icon-bar"></span>
            <span class="icon-bar"></span>
            <span class="icon-bar"></span>
        </button>
        <a class="navbar-brand" href="index.html">

            
        </a>
    </div>
<!-- Меню-->
    <div class="collapse navbar-collapse">
        <ul class="nav navbar-nav navbar-right">
            <li class="scroll active"><a href="#home">Головна</a></li>
            <li class="scroll"><a href="#about-us">Про нас</a></li>
            <li class="scroll"><a href="#services">Послуги</a></li>
            <li class="scroll"><a href="#portfolio">Галерея</a></li>
            <li class="scroll"><a href="#yandex-map">Карта</a></li>
            <li class="scroll"><a href="#contact">Контакти</a></li>
        </ul>
    </div>
    <div id="menuu">
        <a id="blockajax" class="advert"></a>
        <a onclick="openbox('Wrapp');return false;"></a>
        <a class="btn btn-primary btn-lg btn-block" href="#services">При замовлені онлайн знижка
10%
        </a>
    </div>
</div>
</div>
</div>
</header>

</div>
<!-- Блок Про нас -->
<section id="about-us" class="container">
    <div class="container">

```

```

<div class="row">
  <div class="col-sm-6">
    <div class="about-info wow fadeInUp" data-wow-duration="1000ms" data-wow-delay="320ms">
      <h2>Про нас</h2>

      <p>
        Фуршет, пікнік чи барбекю на природі, банкет, кейтеринг, кава-брейк, доставка
        корпоративних обідів.
      </p>
      <p>
        Ми кейтерингова компанія "Зі смаком", молода злагоджена команда, яка готова втілити
        ваші бажання на вищому рівні.
      </p>
      <p>
        Наша мета – виправдати ваші надії, виконати поставлені завдання та, в результаті, –
        стати надійними партнерами.
      </p>
      <a class="scroll" href="#portfolio">
        (фото в альбомах)
      </a>
      
    </p>
  </div>
</div>
</div>
</div>
</section>
<section id="benefits" class="container">
  <div class="col-sm-10">
    <div class="our-skills wow fadeInDown" data-wow-duration="1000ms" data-wow-delay="300ms">
      <h2 align="right" style="color:#fff" >Чому вибирають нас: </h2>
      <div class="single-skill wow fadeInDown" data-wow-duration="1000ms" data-wow-delay="300ms">
        <p class="lead">Професійна команда кухарів</p>
        <div class="progress">
          <div class="progress-bar progress-bar-primary six-sec-ease-in-out" role="progressbar"
            aria-valuetransitiongoal="100">
        </div>
      </div>
    </div>
    <div class="single-skill wow fadeInDown" data-wow-duration="1000ms" data-wow-delay="400ms">
      <p class="lead">Високий рівень обслуговування</p>
      <div class="progress">
        <div class="progress-bar progress-bar-primary six-sec-ease-in-out" role="progressbar"
          aria-valuetransitiongoal="100">
      </div>
    </div>
  </div>
</div>
</div>

```



```

        <button class="zakaz" data-zal="1" href="formpay.html" style="margin-top:20px">Замовити
            ОНЛАЙН</button>
    </div>
</div>
<div class="col-sm-4 wow fadeInDown">
    <div class=""><a href="formpay.html"></a>
        <i class=""></i>
    </div>
    <div class="">
        <h3>
            Фуршет
        </h3>
        <span>Від 30 до 400 людей</span><br>
        <button class="zakaz" data-zal="1" href="formpay.html" style="margin-top:20px">Замовити
            ОНЛАЙН</button>
    </div>
</div>
<div class="col-sm-4 wow fadeInDown">
    <div class=""><a href="formpay.html"></a>
        <i class=""></i>
    </div>
    <div class="">
        <h3>
            Банкет
        </h3>
        <span>Від 10 до 100 людей</span><br>
        <button class="zakaz" data-zal="1" href="formpay.html" style="margin-top:20px">Замовити
            ОНЛАЙН</button>
    </div>
</div>
<div class="col-sm-4 wow fadeInDown">
    <div class=""><a href="formpay.html"></a>
        <i class=""></i>
    </div>
    <div class="">
        <h3>
            Пікнік на природі
        </h3>
        <span>Від 15 до 200 людей</span><br>
        <button class="zakaz" data-zal="1" href="formpay.html" style="margin-top:20px">Замовити
            ОНЛАЙН</button>
    </div>
</div>
<div class="col-sm-4 wow fadeInDown">
    <div class=""><a href="formpay.html"></a>

```



```

<div class="overlay">
  <div class="overlay-content">
    <div class="overlay-text">
      <div class="folio-info">
        <h3>3i смаком</h3>
        <p>Смак твоїх подій</p>
      </div>
      <div class="folio-overview">
        <span class="folio-expand"><a href="images/portfolio/1/kN1rn83CIKM.jpg"
          data-lightbox="portfolio"><i class="fa fa-search-plus"></i></a></span>
      </div>
    </div>
  </div>
</div>
</div>
</div>
</div>
</div>
<div id="portfolio-single-wrap">
  <div id="portfolio-single">
  </div>
</div>
</section>
<div id="yandex-map">
  <script type="text/javascript" charset="utf-8"
    async
    src="https://api-
maps.yandex.ru/services/constructor/1.0/js/?um=constructor%3A485f3fe89daeb342eb86b95d335764684619c03
a5bc4340007ac9acfc8ebdbd8&width=100%25&height=400&lang=ru_RU&scroll=true">
  </script>
</div>
<section id="contact">

  <div id="contact-us" class="parallax">
    <div class="container" width=100% style="background-image: url(images/contact-bg.jpg)" >
      <div class="row">
        <div class="heading text-center col-sm-8 col-sm-offset-2 wow fadeInUp" data-wow-
duration="1000ms"
          data-wow-delay="300ms">
          <h1 style="color:#D46B0A">Наші контакти</h1>
          <p style="color:#fff"> Залиште внизу опис вашої події і надгшлемо вам варіанти меню. </p>
          <p style="color:#fff">Також для нас дуже важливий Ваш голос, можете надіслати відгук про
нас або задайте питання. Ми обов'язково
          зв'яжемося з Вами найближчим часом.</p>
        </div>
      </div>
    </div>
    <div class="contact-form wow fadeIn" data-wow-duration="1000ms" data-wow-delay="600ms">
      <div class="row">
        <div class="col-sm-6">
          <form id="main-contact-form" name="contact-form" method="post" action="#">
            <div class="row wow fadeInUp" data-wow-duration="1000ms" data-wow-delay="300ms">
              <div class="col-sm-6">

```



```

<div class="formBlock-content">
  <div class="form-container form__wrapper">
    <!-- Форма обратной связи -->
    <form id="feedback-form" action="form-processing.php" enctype="multipart/form-data" novalidate>
      <div class="form-row">
        <!-- Календарь -->
        <div class="input-box " id="randomChecked">
          <h3>Оберіть дату:</h3>
          <div class="datepicker-here"></div>
        <!-- Имя пользователя -->
        <div class="form-group">
          <label for="name" class="control-label">Имя</label>
          <input id="name" type="text" name="name" class="form-control" value="" placeholder="Имя"
minlength="2"
          maxlength="30" required="required">
          <div class="invalid-feedback"></div>
        </div>
        <!-- Email пользователя -->
        <div class="form-group">
          <label for="email" class="control-label">Email-адреса</label>
          <input id="email" type="email" name="email" required="required" class="form-control" value=""
placeholder="Email-адреса">
          <div class="invalid-feedback"></div>
        </div>
        </div>
        <!-- Телефон -->
        <div class="form-group">
          <label for="phone" class="control-label">Телефон</label>
          <div class="input-group">
            <div class="input-group-prepend">
              <div class="input-group-text"></div>
            </div>
            <input id="phone" type="tel" name="phone" class="form-control" value="(____)____-____-____"
pattern="^\((?[0-9]{3}\)?(\s+)?[0-9]{3}-?[0-9]{2}-?[0-9]{2}$">
            <div class="invalid-feedback"></div>
          </div>
        </div>
        <!-- Сообщение пользователя -->
        <div class="form-group">
          <label for="message" class="control-label">Ваше повідомлення </label>
          <textarea id="message" name="message" class="form-control" rows="3"
placeholder="Ваше повідомлення (не менше 20 символів)" minlength="20" maxlength="500"
required="required"></textarea>
          <div class="invalid-feedback"></div>
        </div>
        <!-- Пользовательское солашение -->

```

```

<div class="form-group form-agree form-check">
  <input class="form-check-input" type="checkbox" name="agree" id="agree" required value="true">
  <label class="form-check-label" for="agree">Натискаючи на кнопку, я приймаю <a
href="#">Пользовательского
  соглашения</a> и даю своё согласие на обработку моих персональных данных</label>
  <div class="invalid-feedback"></div>
</div>

<!-- Сообщение об ошибке -->
<div class="form-error form-error_hide">Исправьте данные и отправьте форму ещё раз.</div>

<!-- Кнопка для отправки формы на сервер -->
<div class="form-submit">
  <button type="submit">Відправити</button>
</div>

</form>

<!-- Сообщение об успешной отправке формы -->
<div class="form-success form-success_hide">
  <div class="form-success__message">Форма успішно надіслана.</div>
</div>

</div>
</div>

<script type="text/javascript" src="js/jquery.js"></script>
<script type="text/javascript" src="js/bootstrap.min.js"></script>
<!-- <script type="text/javascript" src="http://maps.google.com/maps/api/js?sensor=true"></script> -->
<script type="text/javascript" src="js/jquery.inview.min.js"></script>
<script type="text/javascript" src="js/wow.min.js"></script>
<script type="text/javascript" src="js/mousetscroll.js"></script>
<script type="text/javascript" src="js/smoothscroll.js"></script>
<script type="text/javascript" src="js/jquery.countTo.js"></script>
<script type="text/javascript" src="js/lightbox.min.js"></script>
<script type="text/javascript" src="js/main.js"></script>
<script src="../ajax.googleapis.com/ajax/libs/jquery/1.7.1/jquery.min.js"></script>
<script src="js/datepicker.min.js"></script>
<script src="js/form-processing.js"></script>
<script type="text/javascript">

function openbox(id, tt) {
  var div = document.getElementById(id);
  var tt_div = document.getElementById(tt);
  if (div.style.display == 'block') {
    div.style.display = 'none';

```

```

        } else {
            div.style.display = 'block';
        }
    }
</script>
</body>

</html>

```

Main.css

```

/*****
*****Typography*****
*****/
body {
    font-family: 'Open Sans', sans-serif;
    font-size: 14px;
    line-height: 24px;
    color: #666;
    background-color: #fff;
}
h1, h2, h3, h4, h5, h6 {
    color: #333;
}
h2 {
    font-size: 38px;
    margin-bottom: 20px;
    margin-top: 100px;
}
h3 {
    font-size: 18px;
}
.parallax {
    background-size: cover;
    background-repeat: no-repeat;
    background-position: center;
    background-attachment: fixed;
}
.parallax,
.parallax h1,
.parallax h2,
.parallax h3,
.parallax h4,
.parallax h5,
.parallax h6 {
    color: #000;
}
.parallax input[type="text"],
.parallax input[type="text"]:hover,
.parallax input[type="email"],

```

```

.parallax input[type="email"]:hover,
.parallax input[type="url"],
.parallax input[type="url"]:hover,
.parallax input[type="password"],
.parallax input[type="password"]:hover,
.parallax textarea,
.parallax textarea:hover {
  font-weight: 300;
  color: rgb(0, 0, 0);
}
.btn {
  border: 0;
  border-radius: 0;
}
.btn.btn-primary:hover {
  background-color: #D46B0A;
}
.navbar-nav li a:hover,
.navbar-nav li a:focus {
  outline:none;
  outline-offset: 0;
  text-decoration:none;
  background: transparent;
}
a {
  text-decoration: none;
  -webkit-transition: 300ms;
  -moz-transition: 300ms;
  -o-transition: 300ms;
  transition: 300ms;
}
a:focus,
a:hover {
  text-decoration: none;
  outline: none
}
section {
  padding: 80px 0px;
  padding-bottom: 0px;
}
.heading {
  padding-bottom:90px;
}
.preloader {
  position: fixed;
  top: 0;
  left: 0;
  right: 0;
  bottom: 0;
}

```

```

z-index: 999999;
background: #fff;
}
.preloader > i {
position: absolute;
font-size: 36px;
line-height: 36px;
top: 50%;
left: 50%;
height: 36px;
width: 36px;
margin-top: -15px;
margin-left: -15px;
display: inline-block;
}
/*****Home CSS*****/
*****/
#toTop {
width:100px;
text-align:center;
padding:5px;
position:fixed;
bottom:250px;
right:30px;
cursor:pointer;
display:none;
font-family:'FontAwesome' no-serif;
border-radius: 2px;
color: #fff;
font-size: 12px;
border: none;
background-color: #D46B0A;
text-transform: uppercase;
}
#toTop:hover {
transition: 0.7s;
text-transform: uppercase;
background-color: #D46B0A;
}
#home-slider {
overflow: hidden;
position: relative;
}
#home-slider .caption {
position: absolute;
top: 50%;
margin-top: -104px;
left: 0;

```

```

right: 0;
text-align: center;
text-transform: uppercase;
z-index: 15;
font-size: 18px;
font-weight: 300;
color: #fff;
}
#home-slider .caption h1 {
  color: #fff;
  font-size: 60px;
  font-weight: 700;
  margin-bottom: 30px;
}
.caption .btn-start {
  color: #fff;
  font-size: 14px;
  font-weight: 600;
  padding: 14px 40px;
  border: 1px solid #6e6d6c;
  border-radius: 4px;
  margin-top: 40px;
}
.caption .btn-start:hover {
  color: #fff
}
.carousel-fade .carousel-inner .item {
  opacity: 0;
  -webkit-transition-property: opacity;
  transition-property: opacity;
  background-repeat: no-repeat;
  background-size: cover;
  height: 2037px;
}
.carousel-fade .carousel-inner .item:after {
  content: " ";
  position: absolute;
  top: 0;
  bottom: 0;
  left: 0;
  right: 0;
  background: rgba(0,0,0,.7);
}
.carousel-fade .carousel-inner .active {
  opacity: 1;
}
.carousel-fade .carousel-inner .active.left,
.carousel-fade .carousel-inner .active.right {
  left: 0;

```

```

    opacity: 0;
    z-index: 1;
}
.carousel-fade .carousel-inner .next.left,
.carousel-fade .carousel-inner .prev.right {
    opacity: 1;
}
.carousel-fade .carousel-control {
    z-index: 2;
}
.left-control, .right-control {
    position: absolute;
    top: 50%;
    height: 51px;
    width: 51px;
    line-height: 48px;
    border-radius: 50%;
    border: 1px solid #fff;
    z-index: 20;
    font-size: 24px;
    color: #fff;
    text-align: center;
    -webkit-transition: all 0.5s ease;
    -moz-transition: all 0.5s ease;
    -ms-transition: all 0.5s ease;
    -o-transition: all 0.5s ease;
    transition: all 0.5s ease;
}
.left-control {
    left: -51px;
}
.right-control {
    right: -51px;
}
.left-control:hover,
.right-control:hover {
    color: #fff;
}
#home-slider:hover .left-control {
    left: 30px;
}
#home-slider:hover .right-control {
    right: 30px;
}

#home-slider .fa-angle-down {
    position: absolute;
    left: 50%;
    bottom: 50px;
}

```



```

color: #fff;
display: inline-block;
width: 24px;
margin-left: -12px;
font-size: 24px;
line-height: 24px;
z-index: 999;
-webkit-animation: bounce 3000ms infinite;
animation: bounce 3000ms infinite;
}
.navbar-right li a {
color: #fff;
text-transform: uppercase;
font-size: 12px;
padding-top: 30px;
padding-bottom: 30px;
}
.navbar-right li.active a {
background-color: rgba(0,0,0,.2);
}
.navbar-brand h1 {
font-size: 24px;
color: #fff;
margin-top: 12px;
}
/*****
*****Service CSS*****
*****/
.service-icon {
border-radius: 10px;
color: #fff;
display: inline-block;
font-size: 36px;
height: 90px;
line-height: 90px;
width: 90px;
-webkit-transition: background-color 0.2s ease;
transition: background-color 0.2s ease;
}
.our-services .col-sm-4:hover .service-icon {
background-color: #333;
}

.our-services .col-sm-4 {
border-right: 70px solid #ffffff;
border-bottom: 1px solid #f2f2f2;
padding-bottom: 50px;
}
.our-services .col-sm-4:nth-child(4),

```

```

.our-services .col-sm-4:nth-child(5),
.our-services .col-sm-4:nth-child(6) {
  border-bottom:0;
  padding-top: 60px;
}
.our-services .col-sm-4:nth-child(3),
.our-services .col-sm-4:nth-child(6) {
  border-right:0;
}
.service-info h3 {
  margin-top: 35px;
}
.zakaz {
  color: #fff;
  font-size: 14px;
  font-weight: 600;
  padding: 14px 40px;
  border: 1px solid #6e6d6c;
  border-radius: 4px;
  margin-top: 40px;
  background-color: #D46B0A;
  text-transform: uppercase;
}
.zakaz:hover {
  transition: 0.7s;
  text-transform: uppercase;
  background-color: #0e0602;
}

```

Form-processing.php // форма

```
<?php
```

```
/*
```

```
* Форма зворотного зв'язку
```

```
*/
```

```
header('Content-Type: application/json');
```

```
// обробка тільки аjax запитів (за інших запитів завершуємо виконання скрипта)
```

```
if (empty($_SERVER['HTTP_X_REQUESTED_WITH'])) //
```

```
$_SERVER['HTTP_X_REQUESTED_WITH'] != 'XMLHttpRequest') {
```

```
  exit();
```

```
}
```

```
// обробка даних, надісланих лише методом POST (при інших методах завершуємо
виконання скрипта)
```

```

if ($_SERVER['REQUEST_METHOD'] !== 'POST') {
    exit();
}

// Ім'я файлу для зберігання логів
define('LOG_FILE', 'logs/' . date('Y-m-d') . '.log');
// писати попередження та помилки в лог
define('HAS_WRITE_LOG', true);
// Чи перевіряти капчу
define('HAS_CHECK_CAPTCHA', false);
// чи обов'язково наявність файлів, прикріплених до форми
define('HAS_ATTACH_REQUIRED', false);
// дозволені mime типи файлів
define('ALLOWED_MIME_TYPES', ['image/jpeg', 'image/gif', 'image/png']);
// Максимально допустимий розмір файлу
define('MAX_FILE_SIZE', 512*1024);
// директорія для зберігання файлів
define('UPLOAD_PATH', dirname(__FILE__) . '/uploads/');

// чи додати прикріплені файли в тіло листа у вигляді посилань
define('EMAIL_SETTINGS', [
    'addresses' => ['pieshkova.d.y@nmu.one'], // кому необхідно надіслати листа
    'from' => ['name@gmail.com', 'Ім'я'], // від якого email та імені необхідно надіслати
    листа
    'subject' => 'Повідомлення з форми зворотнього зв'язку', // тема листа
    'host' => 'ssl://smtp.gmail.com', // SMTP-хост
    'username' => 'name@gmail.com', // // SMTP-користувач
    'password' => '*****', // SMTP-пароль
    'port' => '465' // SMTP-порт
]);

define('HAS_SEND_NOTIFICATION', false);
define('BASE_URL', 'https://domain.com');
define('SUBJECT_FOR_CLIENT', 'Ваше повідомлення доставлено');
//
define('HAS_WRITE_TXT', true);

function itc_log($message)

```

```

{
  if (HAS_WRITE_LOG) {
    error_log('Date: ' . date('d.m.Y h:i:s') . ' | ' . $message . PHP_EOL, 3, LOG_FILE);
  }
}

```

```

$data = [
  'errors' => [],
  'form' => [],
  'logs' => [],
  'result' => 'success'
];

```

```

$attachs = [];

```

```

/* 4 ЕТАП - ВАЛІДАЦІЯ ДАНИХ (ЗНАЧЕНЬ ПОЛІВ ФОРМИ) */

```

```

// валідація name

```

```

if (!empty($_POST['name'])) {
  $data['form']['name'] = htmlspecialchars($_POST['name']);
} else {
  $data['result'] = 'error';
  $data['errors']['name'] = 'Заповніть це поле.';
  itc_log('Не заповнено поле name.');
```

```

// валідація email

```

```

if (!empty($_POST['email'])) {
  $data['form']['email'] = $_POST['email'];
  if (!filter_var($_POST['email'], FILTER_VALIDATE_EMAIL)) {
    $data['result'] = 'error';
    $data['errors']['email'] = 'Email неправильний.';
    itc_log('Email не коректний.');
```

```

} else {
  $data['result'] = 'error';
  $data['errors']['email'] = 'Заповніть це поле.';
  itc_log('Не заповнено поле email.');
```

```

// валідація phone
if (!empty($_POST['phone'])) {
    $data['form']['phone'] = preg_replace('/\D/', '', $_POST['phone']);
    if (!preg_match('/^(8|7)(\d{10})$/', $phone)) {
        $data['result'] = 'error';
        $data['errors']['phone'] = 'Поле містить неправильний номер.';
        itc_log('Phone не коректний.');
    }
}

// валідація message
if (!empty($_POST['message'])) {
    $data['form']['message'] = htmlspecialchars($_POST['message']);
    if (mb_strlen($data['form']['message'], 'UTF-8') < 20) {
        $data['result'] = 'error';
        $data['errors']['message'] = 'Це поле має бути не менше 20 символів.';
        itc_log('Поле message повинно бути не менше 20 символів.');
    }
} else {
    $data['result'] = 'error';
    $data['errors']['message'] = 'Заповніть це поле.';
    itc_log('Не заповнено поле message.');
```

```

}

// валідація agree
if ($_POST['agree'] == 'true') {
    $data['form']['agree'] = true;
} else {
    $data['result'] = 'error';
    $data['errors']['agree'] = 'Необхідно встановити цей прапорець.';
    itc_log('Не встановлено прапорець для поля agree.');
```

```

}

// валідація прикріплених файлів
if (empty($_FILES['attach'])) {
    if (HAS_ATTACH_REQUIRED) {
        $data['result'] = 'error';
```



```

        el[i].required = true;
    }
}
}

```

```

jQuery(function ($) {

    //Preloader
    var preloader = $('.preloader');
    $(window).load(function () {
        preloader.remove();
    });

    //main-slider
    var slideHeight = $(window).height();
    $('#home-slider .item').css('height', slideHeight);

    $(window).resize(function () {
        'use strict',
        $('#home-slider .item').css('height', slideHeight);
    });

    //Scroll Menu
    $(window).on('scroll', function () {
        if ($(window).scrollTop() > slideHeight) {
            $('.main-nav').addClass('navbar-fixed-top');
        } else {
            $('.main-nav').removeClass('navbar-fixed-top');
        }
    });

    // Navigation Scroll
    $(window).scroll(function (event) {
        Scroll();
    });

    $('.navbar-collapse ul li a').on('click', function () {
        $('html, body').animate({
            scrollTop: $(this.hash).offset().top - 5
        }, 1000);
        return false;
    });

    // User define function
    function Scroll() {
        var contentTop = [];
        var contentBottom = [];
        var winTop = $(window).scrollTop();
        var rangeTop = 200;
    }
}

```

```

var rangeBottom = 500;
$('.navbar-collapse').find('.scroll a').each(function () {
    contentTop.push($(this).attr('href')).offset().top);
    contentBottom.push($(this).attr('href')).offset().top + $(this).attr('href').height());
})
$.each(contentTop, function (i) {
    if (winTop > contentTop[i] - rangeTop) {
        $('.navbar-collapse li.scroll')
            .removeClass('active')
            .eq(i).addClass('active');
    }
})
};

```

```

$('#tohash').on('click', function () {
    $('html, body').animate({
        scrollTop: $(this.hash).offset().top - 5
    }, 1000);
    return false;
});

```

```

//Initiat WOW JS
new WOW().init();
//smoothScroll
smoothScroll.init();

```

```

// Progress Bar
$('#about-us').bind('inview', function (event, visible, visiblePartX, visiblePartY) {
    if (visible) {
        $.each($('.progress-bar'), function () {
            $(this).css('width', $(this).attr('aria-valuetransitiongoal') + '%');
        });
        $(this).unbind('inview');
    }
});

```

```

//Countdown
$('#features').bind('inview', function (event, visible, visiblePartX, visiblePartY) {
    if (visible) {
        $(this).find('.timer').each(function () {
            var $this = $(this);
            $({
                Counter: 0
            }).animate({
                Counter: $this.text()
            }, {
                duration: 2000,
                easing: 'swing',
                step: function () {

```



```

        $this.text(Math.ceil(this.Counter));
    }
    });
    $(this).unbind('inview');
}
});

// Portfolio Single View
$('#portfolio').on('click', '.folio-read-more', function (event) {
    event.preventDefault();
    var link = $(this).data('single_url');
    var full_url = '#portfolio-single-wrap',
        parts = full_url.split("#"),
        trgt = parts[1],
        target_top = $("#" + trgt).offset().top;

    $('html, body').animate({
        scrollTop: target_top
    }, 600);
    $('#portfolio-single').slideUp(500, function () {
        $(this).load(link, function () {
            $(this).slideDown(500);
        });
    });
});

// Close Portfolio Single View
$('#portfolio-single-wrap').on('click', '.close-folio-item', function (event) {
    event.preventDefault();
    var full_url = '#portfolio',
        parts = full_url.split("#"),
        trgt = parts[1],
        target_offset = $("#" + trgt).offset(),
        target_top = target_offset.top;
    $('html, body').animate({
        scrollTop: target_top
    }, 600);
    $("#portfolio-single").slideUp(500);
});

// Contact form
var form = $('#main-contact-form');
form.submit(function (event) {
    event.preventDefault();
    var form_status = $('<div class="form_status"></div>');
    $.ajax({
        url: $(this).attr('action'),
        beforeSend: function () {

```

```

        form.prepend(form_status.html('<p><i class="fa fa-spinner fa-spin"></i> Email is sending...</p>').fadeIn());
    }
    }).done(function (data) {
        form_status.html('<p class="text-success">Thank you for contact us. As early as possible we will contact you</p>').delay(3000).fadeOut();
    });
});

//Phone mask
$("#phone").mask("+38 (999) 999-9999");

//popup
$("#popup").hide();
$('.zakaz').on('click', function (e) {
    e.preventDefault();
    var currentZal = $(this).attr('data-zal');
    var currentFilm = $(this).siblings('h3');
    var currentFilmImg = $(this).parent().parent('.wow').find('img').attr('src');
    $('.chosen_film .filmTitle').text(currentFilm.text());
    $('#film').val(currentFilm.text());
    $('#zal').val(currentZal);
    $('input[name=label]').val("Зал №: " + currentZal);
    $('.chosen_film .filmImg').attr('src', currentFilmImg);

    $('input[name=targets]').val("Аренда зала на фильм: " + currentFilm.text());

    $('#popup').fadeIn();
    randomChecked();
});

function randomChecked() {

    var r = getRandomInt(1, 3),
        c = $('#randomChecked').find('input[type="checkbox"]').length;

    $('#randomChecked').find('input[type="checkbox"]').each(function () {
        $(this).removeAttr('disabled');
    });

    for (var i = 0; i < r; i++) {
        $('#randomChecked').find('input[type="checkbox"]').eq(getRandomInt(1, c) - 1).attr('disabled', 'disabled');
    }
};

function getRandomInt(min, max) {
    return Math.floor(Math.random() * (max - min + 1)) + min;
}
}

```

```

/*$('body').on('click','.datepicker--cell',function(){
    randomChecked();
})*

$('.our-services a').on('click', function (e) {
    e.preventDefault();
    $(this).parent().parent('.wow').find('.zakaz').click();
});

var fullComment = $('input[name=comment],input[name=short-dest]');

$('.formBlock #fio').on('change', function (e) {
    fullComment.val("Пользователь: " + $('#fio').val() + "; " + "Email: " + $('#email').val() + "; " + "Телефон
: " + $('#phone').val() + "; " + " Сеансы: " + allVals + "; ");
});

$('.formBlock #email').on('change', function (e) {
    fullComment.val("Пользователь: " + $('#fio').val() + "; " + "Email: " + $('#email').val() + "; " + "Телефон
: " + $('#phone').val() + "; " + " Сеансы: " + allVals + "; ");
});

$('.formBlock #phone').on('change', function (e) {
    fullComment.val("Пользователь: " + $('#fio').val() + "; " + "Email: " + $('#email').val() + "; " + "Телефон
: " + $('#phone').val() + "; " + " Сеансы: " + allVals + "; ");
});

var allVals = [];

$('.formBlock input[type=checkbox]').on('change', function (e) {
    var checks = $('.formBlock input[type=checkbox]:checked').length;
    var priceBase = 1799;
    var newPrice = 1799;
    $('.formBlock .input-submit .price .total').text(newPrice);
    $('#input[name=sum]').val(newPrice);
    var yandexSource = 'https://money.yandex.ru/quickpay/shop-
widget?account=410015332827769&quickpay=shop&payment-type-
choice=on&writer=seller&targets=%D0%B0%D1%80%D0%B5%D0%BD%D0%B4%D0%B0+%D0%B7%D
0%B0%D0%BB%D0%B0&targets-hint=&default-sum=' + newPrice + '&button-
text=01&fio=on&mail=on&phone=on&successURL=lofcinema.ru';
    $('#yandexPayment iframe').attr('src', yandexSource);
    allVals = [];
    $('.formBlock input[type=checkbox]:checked').each(function () {
        allVals.push($(this).val());
    });

    fullComment.val("Пользователь: " + $('#fio').val() + "; " + "Email: " + $('#email').val() + "; " + "Телефон
: " + $('#phone').val() + "; " + " Сеанс: " + allVals + "; ");

```

```
});  
  
$('#popup .overlay, #popup .close').on('click', function (e) {  
    e.preventDefault();  
    $('#popup').fadeOut();  
});  
$(document).keyup(function (e) {  
    if (e.keyCode === 27) $('#popup .close').click(); // esc  
});  
  
});
```

ВІДГУК

керівника економічного розділу
на кваліфікаційну роботу бакалавра
на тему:

**«Розробка веб-додатку для кейтерингової компанії на мові PHP та
JavaScript»**

студентки групи 122-18-3 Пешкової Дар'ї Євгенівни

ПЕРЕЛІК ФАЙЛІВ НА ДИСКУ

Ім'я файла	Опис
Пояснювальні документи	
Пешкова.doc	Пояснювальна записка до кваліфікаційної роботи. Документ Word.
Пешкова.pdf	Пояснювальна записка до кваліфікаційної роботи. в форматі PDF
Програма	
Diplom.zip	Архів. Містить коди програми і откомпільовану програму
Презентація	
Пешкова.ppt	Презентація кваліфікаційної роботи.