

**МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ  
ВСП «ОДЕСЬКИЙ ТЕХНІЧНИЙ ФАХОВИЙ КОЛЕДЖ  
ОДЕСЬКОГО НАЦІОНАЛЬНОГО ТЕХНОЛОГІЧНОГО  
УНІВЕРСИТЕТУ»**

*Спеціальність: 121 «Інженерія програмного забезпечення»*

*Освітня програма: «Розробка програмного забезпечення»*

*Група: 4РП-05*

# **Дипломний проект**

**здобувача освіти денної форми навчання**

**РП.05.11.000.ДП**

***КОВАЛЕНКО  
ГЕОРГІЙ  
ОЛЕГОВИЧ***

**м. Одеса  
2022 р.**

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ  
ВСП «ОДЕСЬКИЙ ТЕХНІЧНИЙ ФАХОВИЙ КОЛЕДЖ ОДЕСЬКОГО  
НАЦІОНАЛЬНОГО ТЕХНОЛОГІЧНОГО УНІВЕРСИТЕТУ»

Спеціальність: 121 «Інженерія програмного забезпечення»

Освітня програма: «Розробка програмного забезпечення»

Група: 4РП-05

## ПОЯСНЮВАЛЬНА ЗАПИСКА

до дипломного проекту (роботи) на тему:

### Розробка веб-сховища для зберігання і виведення стічних фотографій за рахунок інструментів JavaScript

Проектний матеріал складається з пояснювальної записки на 48 сторінках та графічного (презентаційного) матеріалу на 10 аркушах (слайдах).

Дипломник \_\_\_\_\_ ( Коваленко Г.О. )

Керівник \_\_\_\_\_ ( Медведєв А.О. )

#### **Консультанти:**

з економічної частини \_\_\_\_\_ (Копайгородська Т.Г. )

з охорони праці \_\_\_\_\_ ( Чорновол Н.І. )

з дотримання вимог ЄСКД \_\_\_\_\_ ( Петрашова В.І.)

старший консультант \_\_\_\_\_ ( Скорнякова О.В. )

#### **До захисту допущений**

Голова циклової комісії \_\_\_\_\_ ( Скорнякова О.В. )

Завідувач відділення \_\_\_\_\_ (Суліма Ю.Ю.)

Захист «    » \_\_\_\_\_ 2022 р.                      Протокол ДКК № \_\_\_\_\_

Оцінка ДКК \_\_\_\_\_

Секретар ДКК \_\_\_\_\_

**МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ**  
**ВСП «ОДЕСЬКИЙ ТЕХНІЧНИЙ ФАХОВИЙ КОЛЕДЖ ОДЕСЬКОГО**  
**НАЦІОНАЛЬНОГО ТЕХНОЛОГІЧНОГО УНІВЕРСИТЕТУ»**

Відділення комп'ютерних систем Комісія КТ та ПІ  
Спеціальність 121 «Інженерія програмного забезпечення»  
Освітня програма «Розробка програмного забезпечення»

ЗАТВЕРДЖУЮ:

Заст. дир. з НВР \_\_\_\_\_

“ \_\_\_\_\_ ” \_\_\_\_\_ 2022 р.

## ЗАВДАННЯ

### на дипломний проект (роботу)

Здобувачеві (здобувачці) освіти Коваленко Георгій Олегович  
(прізвище, ім'я, по батькові)

1. Тема проекту (роботи) Розробка веб-сховища для зберігання і виведення стічних фотографій за рахунок інструментів JavaScript

затверджена наказом по коледжу від “ 30 ” грудня \_\_\_\_\_ 202 1 р. № 306-A2-ОД

2. Термін здачі закінченого проекту (роботи) \_\_\_\_\_

3. Вихідні данні до проекту (роботи) Microsoft Visual Studio, .Net, C#, MS SQL Server, HTML CSS, JavaScript, API, UI, UX, HTTP, Microsoft Edge, Postman

4. Зміст розрахунково-пояснювальної записки (перелік питань, які необхідно розробити)  
1. Технічний огляд проекту. 2. Економічний розрахунок. 3. Охорона праці.

5. Перелік графічного (презентаційного) матеріалу (з точним зазначенням обов'язкових креслень, кількості слайдів)  
Презентація (10 слайдів)

6. Консультанти по проекту (роботі), із зазначенням розділів проекту, що їх стосується

Розділ	Консультант	Підпис, дата	
		Завдання видав	Завдання прийняв
Основний	Медведев А.О.		
Економічний	Копайгородська Т.Г.		
Охорона праці	Чорновол Н.І.		
Нормоконтроль	Петрашова В.І.		
Старший консультант	Скорнякова О.В.		

7. Дата видачі завдання \_\_\_\_\_

Керівник \_\_\_\_\_  
(підпис)

Завдання прийняв до виконання \_\_\_\_\_  
(підпис)

#### КАЛЕНДАРНИЙ ПЛАН

№ з/р	Назва етапів дипломного проекту (роботи)	Термін виконання етапів дипломного проекту (роботи)	Відмітка про виконання
1	Розділ 1. Технічний огляд проекту		
2	Розділ 2. Економічний розрахунок		
3	Розділ 3. Охорона праці		
4	Розробка презентації до дипломної роботи		
5	Чистове оформлення пояснювальної записки		
6	Підготовка доповіді до захисту		
7	Отримання рецензії, відповіді на зауваження рецензента		
8	Захист роботи		

Дипломник \_\_\_\_\_  
(підпис)

Керівник \_\_\_\_\_  
(підпис)



## АНОТАЦІЯ

**Об'єкт дослідження:** об'єктом дослідження при написанні роботи послужив процес розробки веб-сховища для зберігання стічних фотографій.

**Предмет дослідження:** предметом дослідження роботи став проект, розроблений за допомогою інструментів JavaScript.

**Мета дипломної роботи** – розробка веб-сховища для зберігання стічних фотографій

**Досягнуті результати:** Проаналізовано інструменти для реалізації UX/UI, розроблено дизайн-макет проекту, реалізовано веб-проект на платформі ASP.Net та детально опрацьований дизайн додатку на різних пристроях.

У дипломній роботі розкривається актуальність дослідження за обраним напрямом, ставиться проблема, мета і завдання дослідження, визначаються об'єкт, його теоретична, практична значущість значущості.

**Ключові слова:** користувальницький інтерфейс, верстка сайту, UX/UI, стилі, галерея, компоненти, сховище, бази даних.

**Обсяг:** 48 стор., 24 Рисунок, 6 табл., 14 джерел.

## ЗМІСТ

ВСТУП .....	8
1 ТЕХНІЧНИЙ ОГЛЯД ПРОЕКТУ .....	10
1.1 Опис ядра на якому побудована веб система .....	10
1.2 Опис Entity Framework .....	11
1.3 Адмін панель .....	14
1.4 Сторінки додатку .....	16
1.5 Написання скриптів для інтерфейсу .....	22
1.6 Огляд проекту.....	24
1.6.1 Сторінки для користувачів .....	24
1.6.2 Аутентифікація і авторизація .....	30
1.6.3 Сторінка адміністратора .....	31
1.6.4 Крос-браузерність проекту .....	32
2 ЕКОНОМІЧНИЙ РОЗРАХУНОК.....	38
3 ОХОРОНА ПРАЦІ .....	43
ВИСНОВКИ .....	47
СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ.....	48

					РП 05.11.000 ДП ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		7

## ВСТУП

З появою Web-технології комп'ютер починають використовувати абсолютно нові верстви населення Землі. Можна виділити дві найбільш характерні групи, що знаходяться на різних соціальних полюсах, які були стрімко залучені в нову технологію, можливо, навіть всупереч їхнім власним бажанням. З одного боку, це були представники елітарних груп суспільства - керівники великих організацій, президенти банків, топ-менеджери, впливові державні чиновники і т.д. З іншого боку, це були представники найширших верств населення - домогосподарки, пенсіонери, діти.

Вплив глобальної комп'ютерної мережі Internet на сучасний світ не має історичних аналогів. Його сьогоднішній день - це початок епохи електронного проникнення в усі сфери людського життя, це щось більше, ніж просто маркетингова кампанія, це основа нової філософії і нової ділової стратегії. Цілком логічно припустити, що і з точки зору реклами продукції або послуги Інтернет - найбільш вагомий ресурс.

Більшість сучасних людей користуються Інтернетом, як найбільш доступним джерелом інформації. Web-технологія повністю перевернула уявлення про роботу з інформацією, та й з комп'ютером взагалі. Виявилось, що традиційні параметри розвитку обчислювальної техніки - продуктивність, пропускна здатність, ємність запам'ятовуючих пристроїв - не враховували головної частини системи - інтерфейсу з людиною. Застарілий механізм взаємодії людини з інформаційною системою стримував впровадження нових технологій і зменшував вигоду від їх застосування. І тільки коли інтерфейс між людиною і комп'ютером був спрощений до природності сприйняття звичайною людиною, послідував безпрецедентний вибух інтересу до можливостей обчислювальної техніки.

З розвитком технологій гіпертекстової розмітки в Інтернеті стало з'являтися все більше сайтів, тематика яких була абсолютно різною - від сайтів

					РП 05.11.000 ДП ПЗ	Арк.
						8
Змн.	Арк.	№ докум.	Підпис	Дата		

великих компаній, що оповідають про успіхи компанії і її провали, до сайтів маленьких фірм, що пропонують відвідати їх офіси в межах одного міста.

Інформація, доступна користувачам Internet, розташовується на комп'ютерах (Web-серверах), на яких встановлено спеціальне програмне забезпечення. Значна частина цієї інформації організована у вигляді Web-сайтів. Кожен з них має своє ім'я (адреса) в Internet.

Web-сайт – це інформація, представлена в певному виді, яка розташовується на Web-сервері і має своє ім'я (адреса). Для перегляду Web-сайтів на комп'ютері користувача використовуються спеціальні програми, які називаються браузером. Залежно від того, яке ім'я (адреса) сайту ми поставимо в рядку "Адреса", браузер буде завантажувати в своє вікно відповідну інформацію.

Отже, предметом дослідження роботи став проект, розроблений за допомогою інструментів JavaScript.

**Мета роботи** – розробка веб-сховища для зберігання стічних фотографій

Рішення даної роботи поділено на 3 частини:

Частина 1: Етап розробки. У ході якої будуть показані методи і підходи до реалізації проекту.

Частина 2: Етап реалізації та виведення результатів.

					РП 05.11.000 ДП ПЗ	Арк.
						9
Змн.	Арк.	№ докум.	Підпис	Дата		

# 1 ТЕХНІЧНИЙ ОГЛЯД ПРОЕКТУ

## 1.1 Опис ядра на якому побудована веб система

Основну ідею використання DI фреймворків для створення об'єктів можна описати так: "Мені потрібен об'єкт класу *A*, створи його, і мене не цікавить, що і як ти для цього будеш робити".

Незважаючи на те, що в конструктор контролера не передається ніякої конкретної реалізації, але все буде працювати. Розглянемо поетапно, як відбувається в даному випадку впровадження залежності:

- Фреймворк MVC отримує запит і звертається до контролера *PortfolioController*.
- Фреймворк MVC звертається до класу зіставлення залежностей (в даному випадку клас *NinjectDependencyResolver*), щоб той створив новий об'єкт *PortfolioController*, передаючи параметр *Type* в метод *GetService* (в клас *NinjectDependencyResolver*)
- Сопоставитель залежності викликає інфраструктуру *Ninject* для створення нового об'єкта *PortfolioController*, передаючи тип створюваного об'єкта в метод *TryGet*.
- *Ninject* дивиться на конструктор *PortfolioController* і бачить, що там використовується залежність від *CreateRepository*, для якого він встановлює зіставлення з конкретною реалізацією
- *Ninject* створює екземпляр класу *portfolioAdding* і потім використовує його для створення контролера *PortfolioController*
- *Ninject* передає створений об'єкт *PortfolioController* сопоставителю залежності, який, в свою чергу, передає його фреймворку MVC. І далі відбувається обробка запиту.

					РП 05.11.001 ДП ПЗ	Арк.
						10
Змн.	Арк.	№ докум.	Підпис	Дата		

## 1.2 Опис Entity Framework

*ORM* або *Object-relational mapping* (укр. Об'єктно-реляційне відображення) - це технологія програмування, яка дозволяє перетворювати несумісні типи моделей в ООП, зокрема, між сховищем даних і об'єктами програмування. ORM використовується для спрощення процесу збереження об'єктів в реляційну базу даних і їх вилучення, при цьому ORM сама піклується про перетворення даних між двома несумісними станами. Більшість ORM-інструментів значною мірою покладаються на метадані бази даних і об'єктів, так що об'єктам нічого не потрібно знати про структуру бази даних, а базі даних - нічого про те, як дані організовані в додатку. ORM забезпечує повне розділення завдань в добре спроектованих додатках, при якому і база даних, і додаток можуть працювати з даними кожен у своїй вихідній формі.



Рис. 1.1 – Принцип роботи ORM

**Принцип роботи ORM.** Ключовою особливістю ORM є відображення, яке використовується для прив'язки об'єкта до його даними в БД. ORM як би створює «віртуальну» схему бази даних в пам'яті і дозволяє маніпулювати даними вже на рівні об'єктів. Відображення показує як об'єкт і його властивості пов'язані з однією або декількома таблицями і їх полями в базі даних. ORM використовує інформацію цього відображення для управління процесом перетворення даних між базою і формами об'єктів, а також для

створення SQL-запитів для вставки, оновлення та видалення даних у відповідь на зміни, які додаток вносить в ці об'єкти.

*Entity Framework* - це платформа ORM з відкритим вихідним кодом для додатків .NET, підтримувана Microsoft. Це дозволяє розробникам працювати з даними, використовуючи об'єкти класів, специфічних для предметної області, що не зосереджуючись на базових таблицях бази даних і шпальтах, де зберігаються ці дані. З *Entity Framework* розробники можуть працювати на більш високому рівні абстракції, коли мають справу з даними, і можуть створювати і підтримувати орієнтовані на дані додатки з меншою кількістю коду в порівнянні з традиційними програмами.

Офіційне визначення: «*Entity Framework* - це об'єктно-реляційний картограф (ORM), який дозволяє розробникам .NET працювати з базою даних, використовуючи об'єкти .NET. Це усуває необхідність в більшій частині коду доступу до даних, який зазвичай припадає писати розробникам ».

#### *Особливості Entity Framework*

- Кроссплатформеність: EF Core - це кроссплатформенная інфраструктура, яка може працювати в Windows, Linux і Mac.
- Моделювання: EF (Entity Framework) створює EDM (Entity Data Model) на основі об'єктів POCO (Plain Old CLR Object) з властивостями отримання / установки різних типів даних. Ця модель використовується при запиті або збереженні даних об'єкта в базовій базі даних.
- Запити: EF дозволяє нам використовувати запити LINQ (C # / VB.NET) для отримання даних з базової бази даних. Постачальник бази даних переведе ці запити LINQ на специфічний для бази даних мову запитів (наприклад, SQL для реляційної бази даних). EF також дозволяє нам виконувати необроблені запити SQL безпосередньо до бази даних.
- Відстеження змін: EF відстежує зміни, що відбулися з екземплярами ваших сутностей (значеннями властивостей), які необхідно відправити в базу даних.

					РП 05.11.001 ДП ПЗ	Арк.
						12
Змн.	Арк.	№ докум.	Підпис	Дата		

- Збереження: EF виконує команди INSERT, UPDATE і DELETE для бази даних на основі змін, що відбулися з вашими сутностями при виклику SaveChanges () методу. EF також надає асинхронний SaveChangesAsync () метод.

- Паралелізм: EF за замовчуванням використовує Оптимістичний паралелізм для захисту перезаписувати змін, зроблених іншим користувачем з моменту отримання даних з бази даних.

- Транзакції: EF виконує автоматичне керування транзакціями при запиті або збереженні даних. Він також надає опції для настройки управління транзакціями.

- Кешування: EF включає перший рівень кешування з коробки. Таким чином, повторний запит поверне дані з кешу, а не потрапить в базу даних.

- Вбудовані угоди: EF слід угодами за шаблоном програмування конфігурації і включає в себе набір правил за замовчуванням, які автоматично конфігурують модель EF.

- Конфігурації: EF дозволяє нам конфігурувати модель EF, використовуючи атрибути анотації даних або Fluent API для перевизначення угод за замовчуванням.

- Міграції: EF надає набір команд міграції, які можна виконувати на консолі диспетчера пакетів NuGet або в інтерфейсі командного рядка для створення або управління базової схемою бази даних.

В цьому проекті було створено базу даних. Її підключення було реалізовано за допомогою *connectionString*, яка була добавлена до файлу *Web.config* наступним чином:

```
<connectionStrings>
  <add name="EFDbContext" connectionString="Data
Source=(localdb)\MSSQLLocalDB;Initial Catalog=Admins;Integrated
Security=True;Connect
```

					РП 05.11.001 ДП ПЗ	Арк.
						13
Змн.	Арк.	№ докум.	Підпис	Дата		

```
Timeout=30;Encrypt=False;TrustServerCertificate=False;ApplicationIntent=ReadWrite;MultiSubnetFailover=False"
providerName="System.Data.SqlClient"/>
</connectionStrings>
<appSettings>
```

Даний алгоритм звертається до EFDbContext, який використовує DbSet для створення, читання, оновлення та видалення елементів бази даних. А далі вже йде підключення до бази, якщо бази даних не існують, то при правильному вказанні маршрутизації автоматично створиться нульова база даних, яка має всі властивості, які ми вказали в моделі.

### 1.3 Адмін панель

Структура сайту полягає у наступному. Адміністратор сайту може змінювати інформацію не використовуючи кодингу. Це реалізовано за допомогою поділу на звичайних користувачів та адміністраторів. Для реалізації цього, було створено сторінку Login, де адміністратор може авторизуватися та розпочати роботу з базою даних.

Аудентифікація була реалізована в *Web.config*, наступним чином:

```
<authentication mode="Forms">
  <forms loginUrl="~/Login/Login" timeout="500">
    <credentials passwordFormat="Clear">
      <user name="qwerty" password="1234567890" />
    </credentials>
  </forms>
</authentication>
```

					РП 05.11.001 ДП ПЗ	Арк.
						14
Змн.	Арк.	№ докум.	Підпис	Дата		

Даний код лист говорить наступне: знаходження сторінки, де буде авторизація користувача `<forms loginUrl="~/Login/Login" timeout="500">`, а також пароль, який уже обирає сам творець проекту `<user name="qwerty" password="1234567890" />`.

Після вдалої авторизації, адміністратор починає працювати з адмініструванням даних. Спочатку він знаходиться в вікні `"Index.cshtml"`, де було реалізовано додавання, зміну та видалення даних.

Вивід даних було реалізовано таблицею, куди занесено інформацію про всі об'єкти бази даних. Алгоритм появи інформації полягає у тому, що `Index.cshtml` звертається до контролера `AdminController.cs`, який звертається до бібліотеки де зберігаються моделі та інтерфейси проекту, які детально говорить про всі об'єкти: кількість, зміст.

Додавання об'єктів було реалізовано окремою кнопкою, яка викидає на нову сторінку, де може додати інформацію.

Видалення проекту було реалізовано окремою кнопкою, при натиску на кнопку, сайт звертається до `AdminController.cs` та передає ідентифікатор об'єкту для того, щоб можливо було видалити інформацію за допомогою контролерів. Основну роль в цьому ході відіграє контролер, а не html – код. Сценарії кнопки `«Удалить»` у `AdminController.cs`.

Для зміни інформації в середині об'єкта було створено окрему сторінку `«EditProject.cshtml»`, яка звертається до `AdminController.cs` для того, щоб отримувати повний алгоритм виконання своєї ролі, а також було реалізовано підключення до `PortfolioList`, де знаходиться модель бази даних. Підключення до `PortfolioList.cs` було реалізовано наступним чином: `@model Project.Library.Models.PortfolioList`. Даний код говорить про детальне розташування моделі в проекті.

					РП 05.11.001 ДП ПЗ	Арк.
						15
Змн.	Арк.	№ докум.	Підпис	Дата		



- Сторінка для зв'язку

Сьогодні, завдяки існуванню CSS, є можливість відокремити дизайн сторінки від її вмісту, а також прискорити процес роботи і в разі скоротити розмір HTML-документа. Тим, хто вже дуже давно займається створенням сайтів, належить позбутися від старих звичок і навчитися сприймати HTML тільки як мову розмітки, призначений для структурування і впорядкування даних. Новачкам буде легше освоювати нові принципи створення веб-сторінок в зв'язку з відсутністю поглядів на HTML як на інструмент додання сторінок привабливого вигляду. За це тепер відповідає CSS.

У процесі верстки кодом html відбувається розбивка «скелета» сайту на частини. А за допомогою css (каскадних таблиць стилів) задаються розміри його «кісток», колір і розташування.

Блочна - в даний момент є основним способом верстки. На відміну від табличній блокова верстка має ряд переваг:

- Відділення стилю елементів від коду html;
- Можливість накладення одного шару на інший - така можливість багато в чому полегшує позиціонування елементів.
- Краща індексація пошуковими системами;
- Висока швидкість завантаження сторінки, що складається від взаємно незалежних елементів;
- Легкість створення візуальних ефектів (випадають меню, списків, спливаючих підказок).

Основним недоліком блочної верстки є якась «двозначність» розуміння її коду різними браузерами. Тому часто html сторінки доводиться «доводити» шляхом використання спеціальних хаков.

З появою блочної верстки народилося таке поняття, як «кроссбраузерність». Через відмінності відображення одного і того ж елемента в різних браузерах верстальникам доводиться вставляти в основний html цілі шматки коду.

					РП 05.11.001 ДП ПЗ	Арк.
						17
Змн.	Арк.	№ докум.	Підпис	Дата		

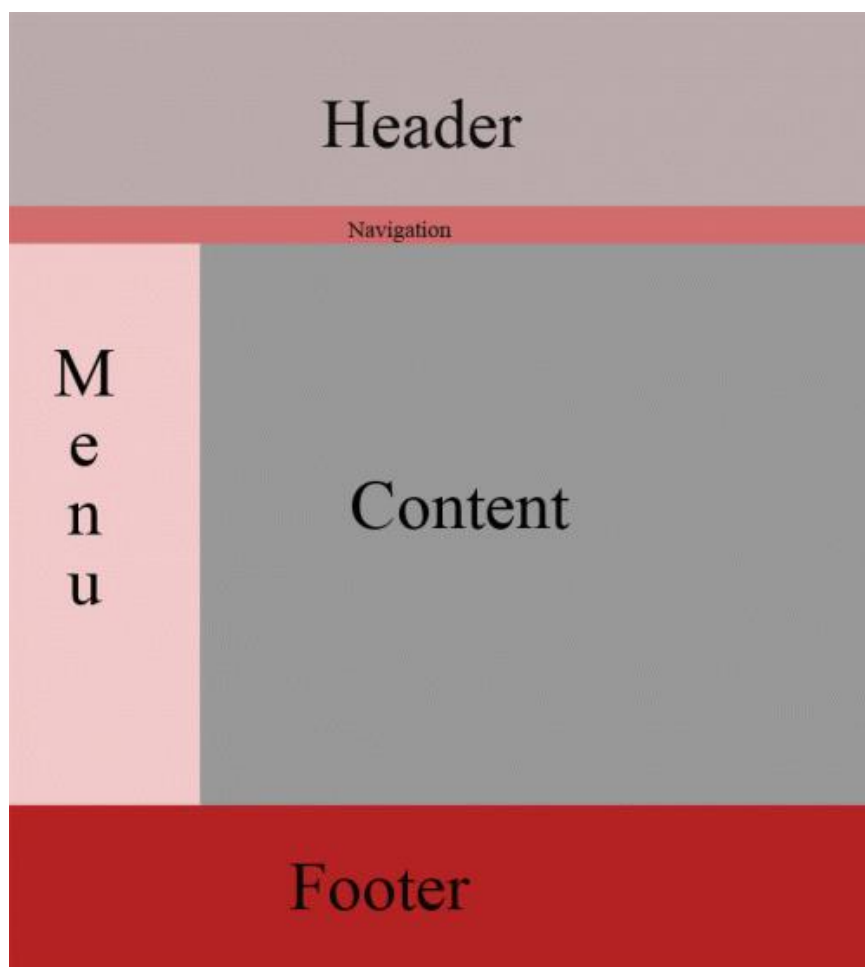
Дія хака є вузькоспеціалізованим і вирішує проблему некоректного відображення лише в одному браузері.

```
<div class="hero-item">
  <div class="hero-text">
    <div class="ht-cata">Nature</div>
    <h2>Photo Studio</h2>
    <p>Cassi - a creative lifestyle photo studio making emotional photography
    <a href="#" class="ht-btn">See More <i class="arrow_right"></i></a>
  </div>
  <div class="hi-bg set-bg" data-setbg="img/slider-1.jpg"></div>
</div>
```

Основним елементом, що застосовуються в блокової верстці, є тег <div>. Ділянка коду, відокремлений цим тегом, називається шаром. Всі стильові рішення винесені за межі коду html в каскадні таблиці стилів. Доступ до них здійснюється через ідентифікатори або класи css:

```
392 .hero-item {
393     position: relative;
394     display: flex;
395     align-items: center;
396 }
397 .hero-text {
398     position: relative;
399     padding-left: 80px;
400     max-width: 595px;
401     z-index: 5;
402 }
```

Перед початком верстки готовий psd макет сайту в графічному редакторі розрізають на блоки (шари). В окрему папку поміщають вирізані фонові картинки, які будуть прикріплюватися окремо до кожного шару.



**Рисунок 1.5 – Прототип макету проекту**

Для прикладу візьмемо ось такий макет сайту, створений в Photoshop (рис.1.5). Спочатку в текстовому редакторі за допомогою div задаємо структуру майбутнього ресурсу і присвоюємо кожному шару свій селектор id. Виходить така структура:

					РП 05.11.001 ДП ПЗ	Арк.
						19
Змн.	Арк.	№ докум.	Підпис	Дата		

```

<!-- Page Preloader -->
<div id="preloader">
  <div class="loader"></div>
</div>

<!-- Offcanvas Menu Section -->
<div class="offcanvas-menu-wrapper">...
</div>
<!-- Offcanvas Menu Section end -->

<!-- Header section -->
<header class="header-section">...
</header>
<!-- Header section end -->

<!-- Hero section -->
<section class="hero-section">...
</section>
<!-- Hero section end -->

<!-- Footer section -->
<footer class="footer-section">...
</footer>
<!-- Footer section end -->

```

Потім до готової структури сайту на html рядком `<link rel=«stylesheet» type=«text/css» href=«style.css» />` прикріплюємо файл css. Після чого додаємо в нього стильове опис кожного шару, позиціонування щодо інших елементів і його розміри.

```

h1, h2, h3, h4, h5, h6 {
  margin: 0;
  color: #333333;
  font-weight: 600;
}
h1 {
  font-size: 70px;
}
h2 {
  font-size: 36px;
}
h3 {
  font-size: 30px;
}
h4 {
  font-size: 24px;
}

```

Адаптивна верстка сайту дозволяє веб-сторінкам автоматично підлаштовуватися під екрани планшетів і смартфонів. Мобільний інтернет-трафік зростає з кожним роком і щоб ефективно обробляти цей трафік, потрібно пропонувати користувачам адаптивні сайти зі зручним інтерфейсом.

Пошукові системи використовують ряд критеріїв для оцінки адаптивності сайту при перегляді на мобільних пристроях. Google намагається спростити користування Інтернетом для власників смартфонів і планшетів,

					РП 05.11.001 ДП ПЗ	Арк.
						20
Змн.	Арк.	№ докум.	Підпис	Дата		

відзначаючи в мобільній видачу адаптовані під мобільні пристрої сайти спеціальною позначкою mobile-friendly.

Розглянемо, як можна використовувати CSS3-медіазапроси для створення адаптивного дизайну. min-width задає мінімальну ширину вікна браузера або екрану, до якої будуть застосовані певні стилі. Якщо якесь значення буде нижче min-width, то стилі будуть проігноровані. max-width робить протилежне.

```
@media (max-width: 576px), only screen and (min-width: 768px) and (max-width: 991px) {  
  .recent-blog.blog-item {  
    padding: 0;  
  }  
}  
.recent-blog.blog-item .blog-text {  
  position: relative;  
  bottom: 0;  
  max-width: 655px;  
}  
.comment-title {  
  color: #fff;  
  margin-bottom: 30px;  
  padding-top: 80px;  
}  
.comment-form {  
  margin-bottom: 60px;  
}
```

Цей код буде виконаний тільки для екранів браузерів шириною від 768 до 991 рх. В результаті це допомагає нам домогтися адаптивності сторінок під різні пристрої (рис. 2.6).

					РП 05.11.001 ДП ПЗ	Арк.
						21
Змн.	Арк.	№ докум.	Підпис	Дата		

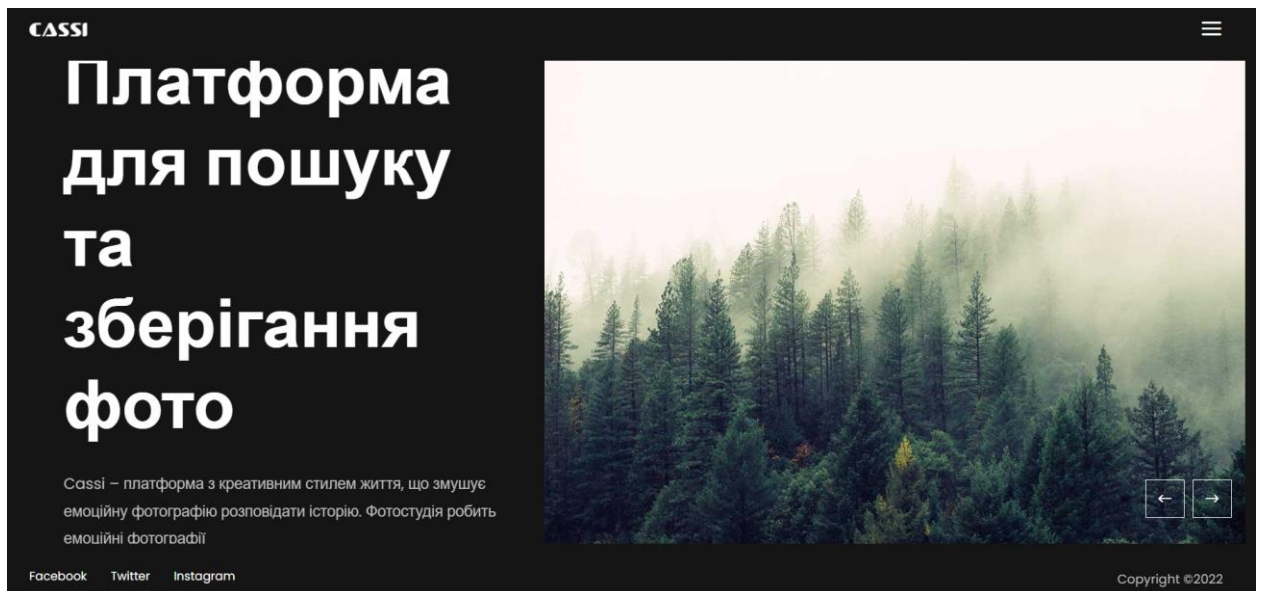


Рис. 1.6 – Реалізація макету головної сторінки

## 1.5 Написання скриптів для інтерфейсу

Як і у будь-якого мову програмування, основне завдання Javascript створювати послідовність дій, які будуть приводити до певного результату. Це можуть бути умови «якщо-то», цикли, які створюють певну послідовність дій, математичні обчислення і.т.д. Найголовніше в тому, що всі ці операції можна проводити на веб-сторінках, у вікні браузера. Причому, javascript може працювати і без підключення до мережі Інтернет.

JavaScript дозволяє додавати динамічність і зручність при використанні сайту. Наприклад, за допомогою JavaScript можна створити бічне меню, яке відкривається при натисканні на кнопки з трьома горизонтальними лініями.

```
$(function(){
    $('.menuToggle').on('click', function() {
        $('.menu').slideToggle(300, function(){
            if ($(this).css('display') === "none"){
                $(this).removeAttr('style');
            }
        });
    });
});
```

					РП 05.11.001 ДП ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		22

На початку ми вішаємо обробник подій .on при натисканні на саму кнопку «Меню». У середині цієї події ми додаємо на \$ («. Menu») виппадаюче меню функцію slideToggle() яка буде виробляти плавне відкриття і закриття меню, що випадає.

Є невелика проблема, яка виникне при переході версій сайту при натисканні на кнопку «Меню», а сама буде ховатися блок з самими пунктами, меню, тому за замовчуванням функція slideToggle() при закритті додає атрибут «display: none». Щоб виправити цей баг потрібно викликати колбек (функція яка буде визиватися після відпрацювання основної функції) для функції slideToggle(). В середині колбек ми пишемо умову. Якщо у нас атрибут в блоці меню дорівнює «display: none» тоді ми видаляємо атрибут «style».

Також, для додавання ефектів можна використовувати Веб-API Animations, який дозволяє нам створити анімацію і управляти її відтворенням за допомогою JavaScript. З Web Animations API, ми можемо управляти інтерактивною анімацією з стилями JavaScript, відокремлюючи подання від дії. Нам більше не потрібно покладатися на DOM-heavy методики написання CSS властивостей і оглядових класів для елементів керування відтворенням напрямки. І на відміну від чисто декларативного CSS, JavaScript також дозволяє встановлювати динамічні значення властивостей і їх тривалість. Щоб створити власні бібліотеки і інтерактивної анімації Web Animations API може стати прекрасним інструментом в роботі.

					РП 05.11.001 ДП ПЗ	Арк.
						23
Змн.	Арк.	№ докум.	Підпис	Дата		

```

@-webkit-keyframes loader {
  0% {
    -webkit-transform: rotate(0deg);
    border: 4px solid #f44336;
    border-left-color: transparent;
  }
  50% {
    -webkit-transform: rotate(180deg);
    border: 4px solid #673ab7;
    border-left-color: transparent;
  }
  100% {
    -webkit-transform: rotate(360deg);
    border: 4px solid #f44336;
    border-left-color: transparent;
  }
}

```

Тепер зберемо всі частини разом використовуючи animate () методи, такі як fadeout.

```

$(window).on('load', function() {
  /*-----
  |   Preloder
  |-----*/
  $(".loader").fadeOut();
  $("#preloder").delay(400).fadeOut("slow");

  /*-----
  |   Masonry
  |-----*/
  $('.gallery-gird').masonry();
});

```

Таким чином, JavaScript є хорошим інструментом для налаштування і реалізації грамотного UX / UI підходу.

## 1.6 Огляд проекту

### 1.6.1 Сторінки для користувачів

					РП 05.11.001 ДП ПЗ	Арк.
						24
Змн.	Арк.	№ докум.	Підпис	Дата		

*Landing page* (вхідні, цільова сторінка) - це сторінка сайту, на яку потрапляє користувач при кліці по рекламному оголошенню. Дана сторінка відрізняється від інших сторінок тим, що містить якийсь заклик до дії і насичена ключовими фразами. Головний принцип *Landing page* - це дати відвідувачеві саме ту інформацію, яку він прийшов на даний сайт, приховавши при цьому з його поля зору все інше, що може перешкоджати його діям.

*Landing page* (цільова сторінка, сторінка захоплення) - це лише однієї сторінки сайт в інтернеті, на який потрапляє ваш потенційний клієнт.

Завдання Landing page - залучити відвідувачів сайту, зробивши їх Вашими клієнтами з максимально можливою вірогідністю. Якщо сайт зроблений погано - ви втрачаєте клієнта, якщо грамотно і якісно - ви отримуєте прибуток від нового замовлення. Landing page ідеально підходить для:

- роздрібної та оптової торгівлі;
- сфери послуг;
- унікального і дорогого товару;

Переваги Landing page над звичайним сайтом:

- для побудови сторінки використовується тільки важлива і потрібна інформація;
- використовується індивідуальний дизайн;
- сайт буде коректною працювати як стаціонарному комп'ютері, так і на мобільних пристроях;
- розробка ведеться з урахуванням маркетингових досліджень, що дозволяє досягати високої конверсії.

					РП 05.11.001 ДП ПЗ	Арк.
						25
Змн.	Арк.	№ докум.	Підпис	Дата		

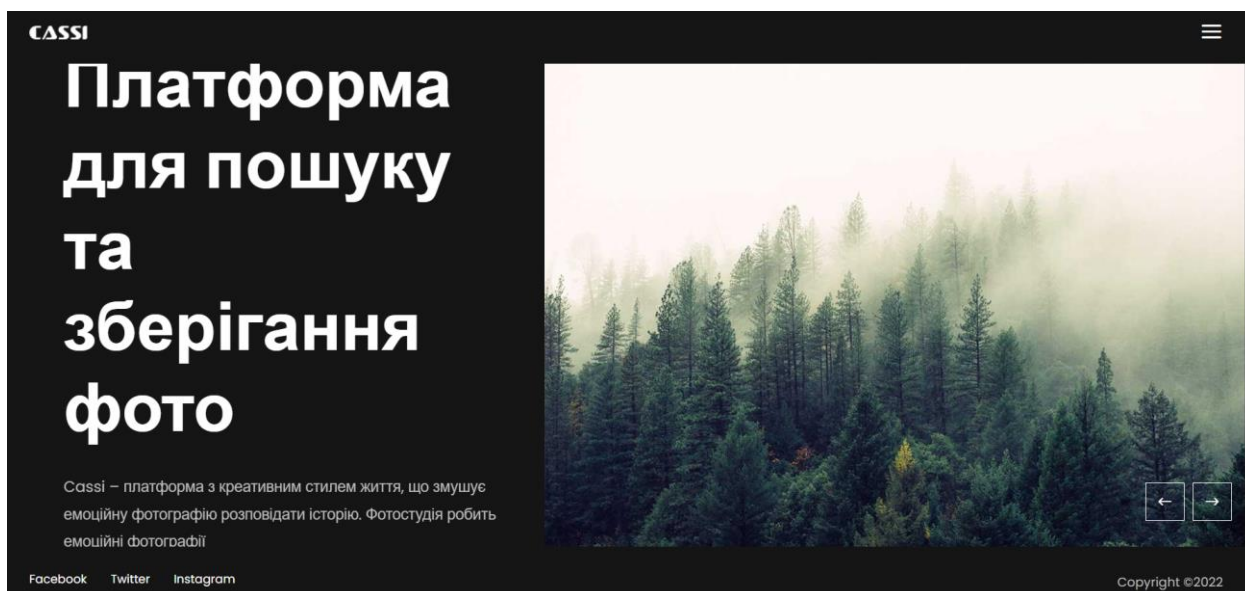


Рис. 1.7 – Лендінг сторінка проекту

На лендінг – сторінці було реалізовано інформацію про основні напрями роботи проекту, технології, які використовуються для досягнення мети, а також було представлено інформацію про переваги проекту.

Так як головна сторінка – це перший дотик, то на цій сторінці одразу ж для користувача знаходяться відповідь на запитання «Скільки?» та «Чому маю обрати саме цей проект?».

Так як у проекті реалізовано чотири сторінки для користувача, тому для зручного пересування було реалізовано панель.

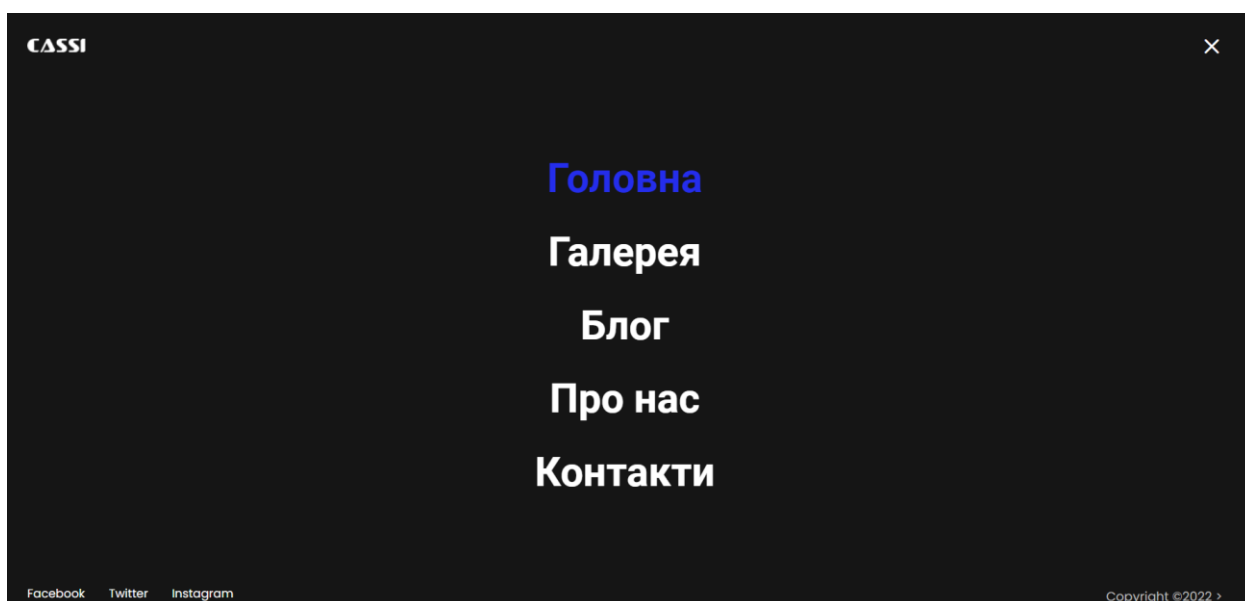


Рис. 1.8 – Панель для швидкого пересування між сторінками

					РП 05.11.001 ДП ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		26

Натиснувши на кнопку «Галерея», користувач переходить до іншої сторінки, знаходиться база даних. У цій сторінці знаходяться всі фотографії, які може продивитися користувач.

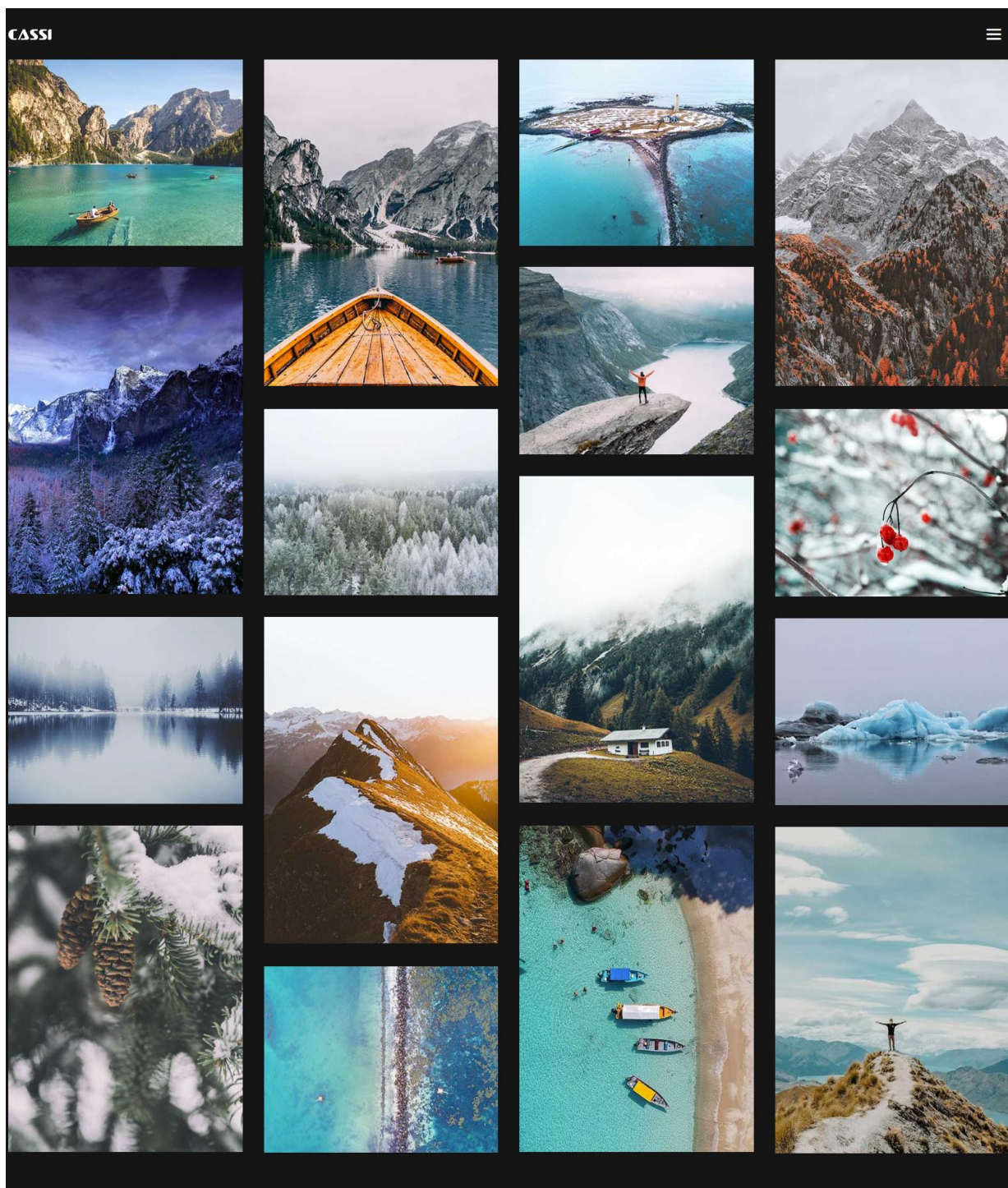


Рис. 1.9 – Сторінка «Галерея»

Змн.	Арк.	№ докум.	Підпис	Дата

Також для того, щоб побачити повну фотографію, користувачу треба обрати певне вікно і при кліку, вийде віконце, яке буде оповідати детально про зміст проекту.

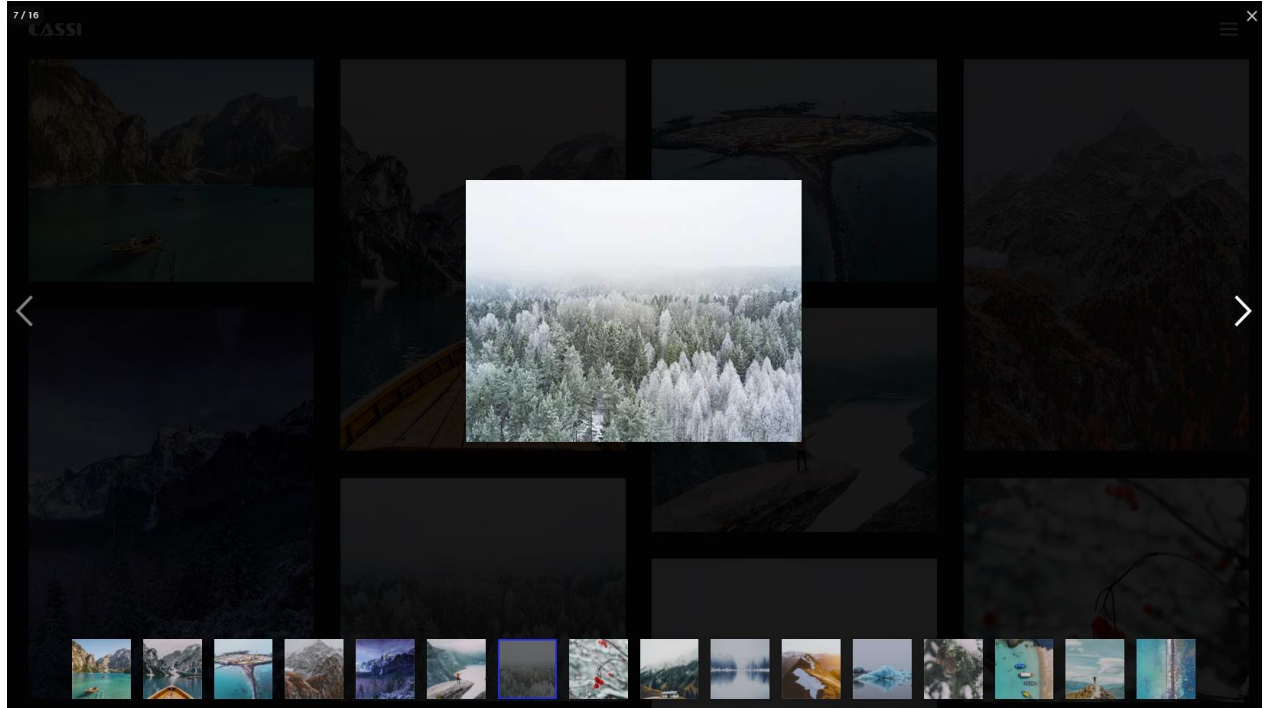


Рис. 1.10 – Фотографія у повному розмірі

На сторінці «О нас» користувач може більш детально ознайомитися з напрямками роботи компаніями. Можна дізнатися інформацію про реалізацію всіх проектів, яким чином проходить їх реалізація. А також може отримати інформацію про команду, її погляди та інтереси, та інформацію про кожну особу проекту окремо.

					РП 05.11.001 ДП ПЗ	Арк.
						28
Змн.	Арк.	№ докум.	Підпис	Дата		

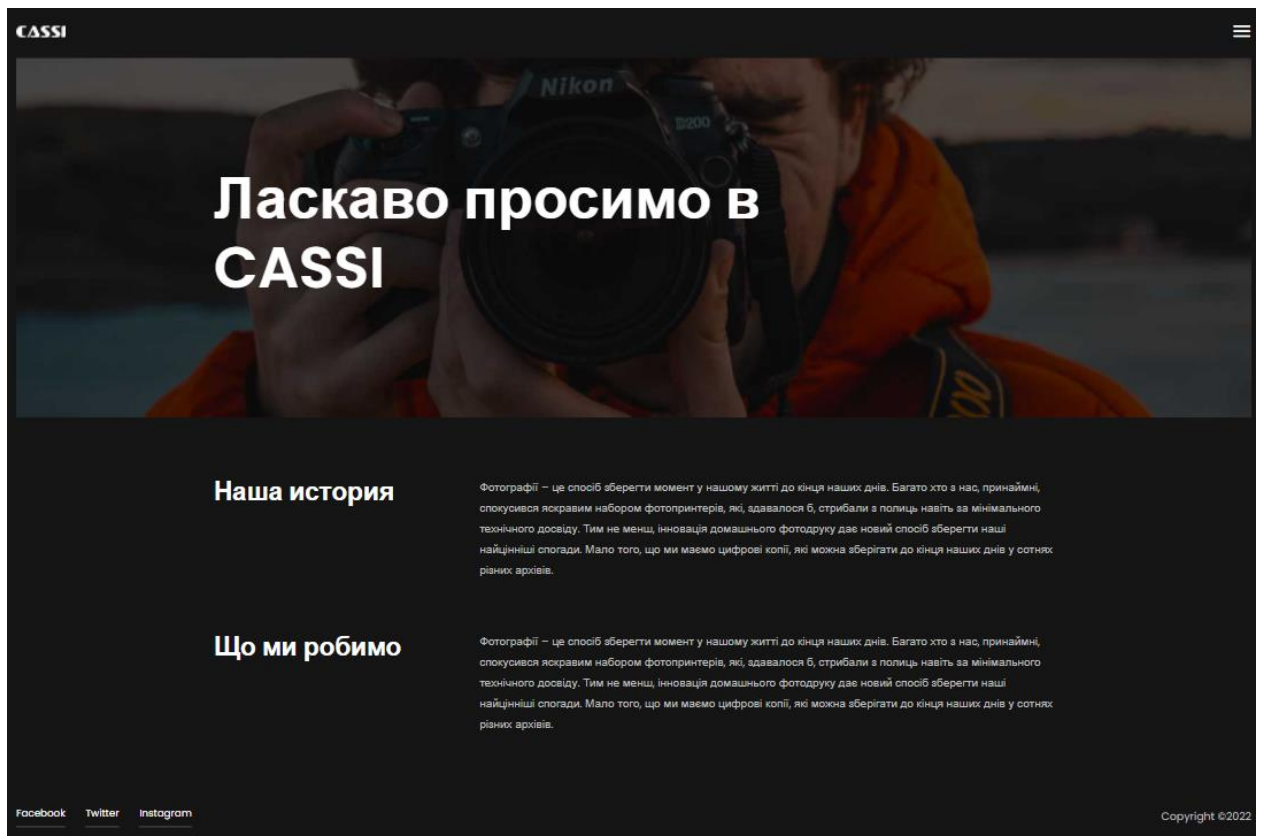


Рис. 1.11 – Сторінка «О нас»

Остання сторінка, яка доступна для звичайного користувача ресурсу називається «Контакти», де описана інформація, як можна зв'язатися з командою, для того, щоб зробити замовлення. Окрім цього було додано вікно, куди кожен може залишити свій відгук, який ми можемо отримати та на основі цього організувати робочий процес у кращому напрямі.

					РП 05.11.001 ДП ПЗ	Арк.
						29
Змн.	Арк.	№ докум.	Підпис	Дата		

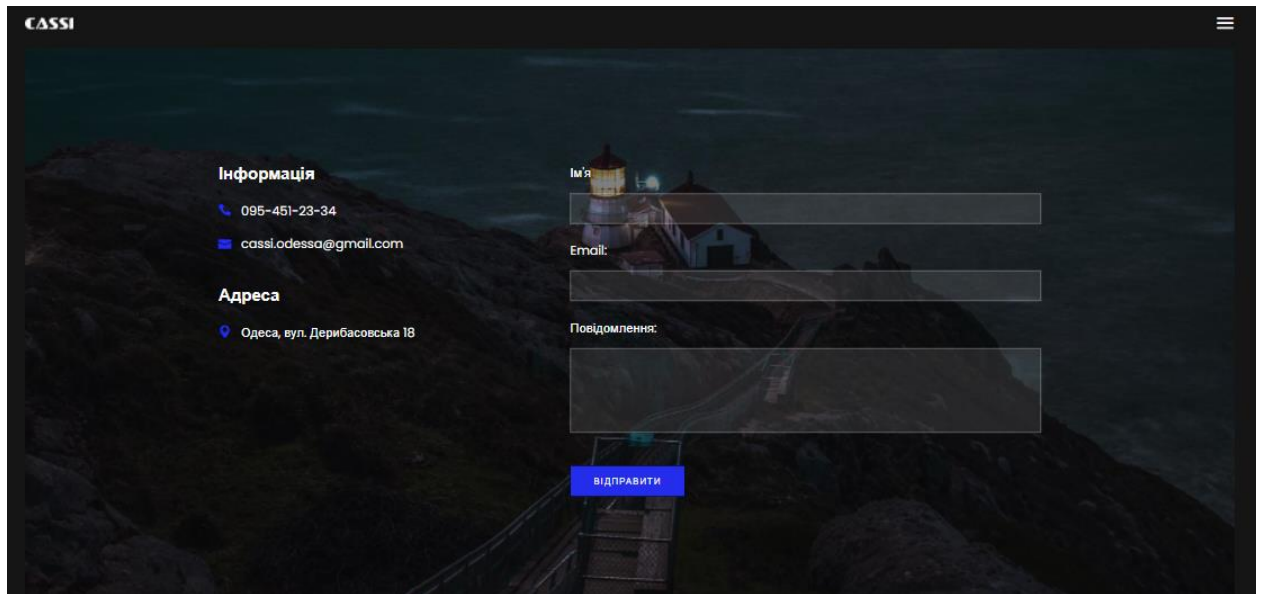


Рис. 1.12 – Сторінка «КОНТАКТИ»

## 1.6.2 Аутентифікація і авторизація

Для редагування інформації, було створено сторінку для адміністратора, який перед початком роботи з базою даних повинен авторизуватися. Для того, щоб це зробити, користувач повинен перейти за посиланням: <http://localhost:2390/Login/Login>.

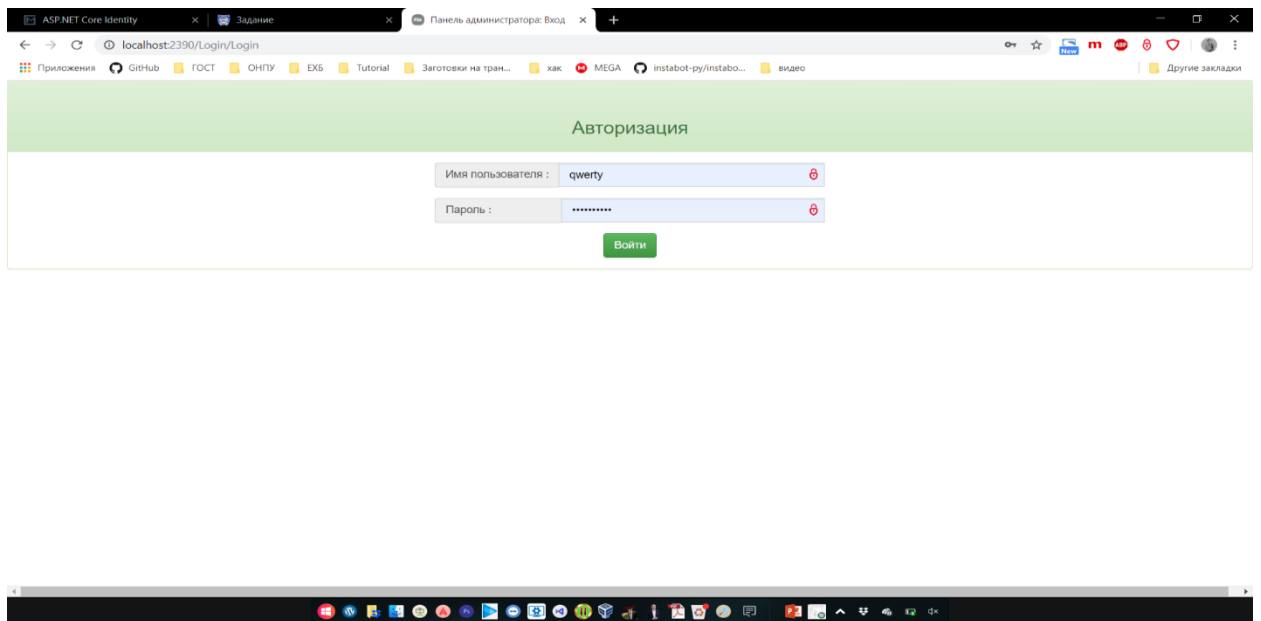


Рис. 1.13 – Сторінка авторизації

					РП 05.11.001 ДП ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		30

При правильному вводі даних користувач має повний доступ до сторінки редагування ресурсу. При неправильно вводі даних, ресурс не буде пускати користувача до сторінки редагування бази даних.

### 1.6.3 Сторінка адміністратора

При правильному вводі паролю та логіну, користувач може перейти до сторінки редагування бази даних, де має повні права на адміністрування цією базою.

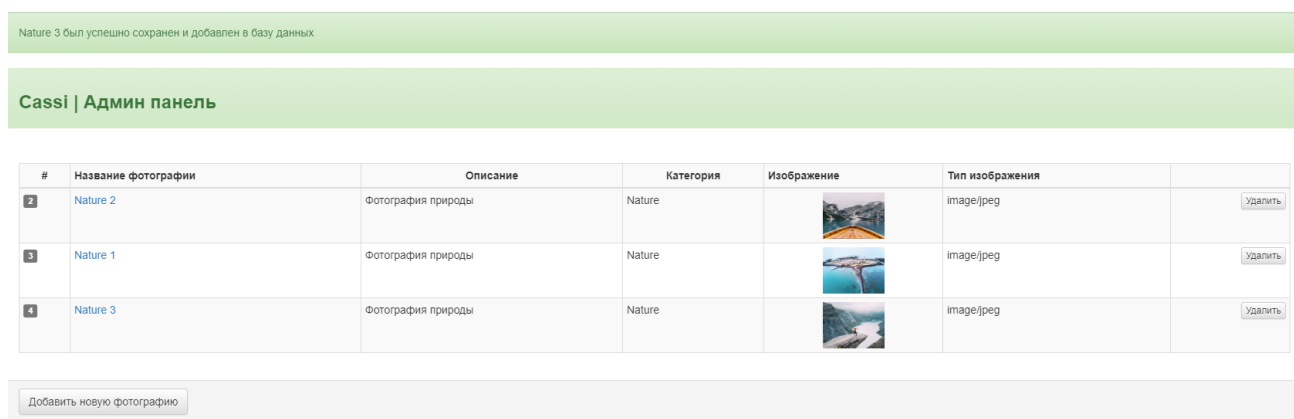


Рис. 1.14 – Сторінка «Адмін панель»

На цій панелі можливо редагування інформації, для того, щоб перейти до редагування користувач повинен натиснути на назву проекту, який він хоче змінити, натиснувши він переходить до нової сторінки, де проходить редагування інформації:

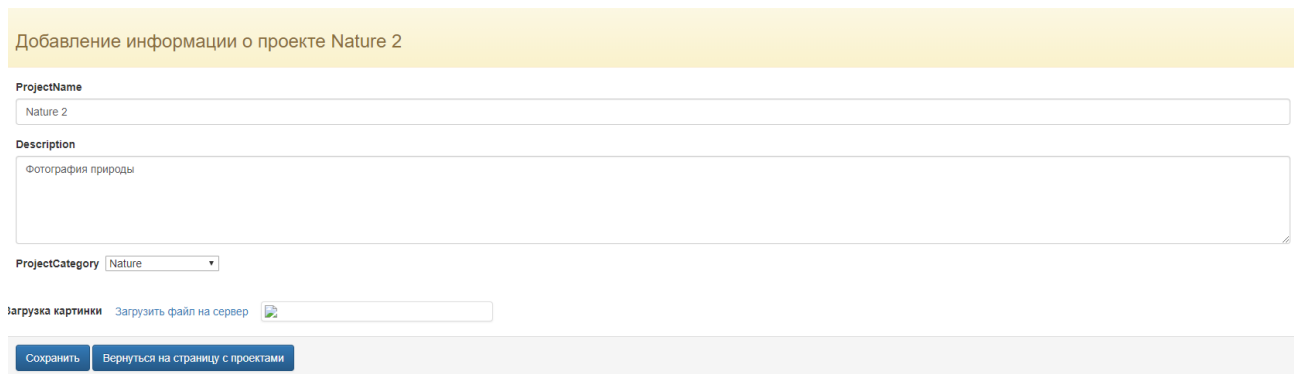


Рис. 1.15 – Панель редагування та додавання

Після редагування користувач повинен натиснути кнопку «Сохранить», після чого дані будуть змінені та з'явиться повідомлення про вдалу зміну інформації.

За допомогою кнопки «Добавить новый проект» користувач може додати новий об'єкт до бази даних. Натиснувши кнопку користувач переходить до панелі редагування та додавання після додавання інформації, буде повідомлення про вдале додавання інформації (Рис. 1.16).

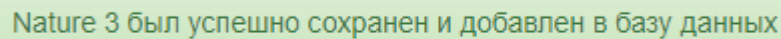
A light green rectangular notification box containing the text: "Nature 3 был успешно сохранен и добавлен в базу данных".

Рис. 1.16 – Повідомлення про зміну та додавання інформації

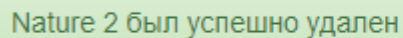
A light green rectangular notification box containing the text: "Nature 2 был успешно удален".

Рис. 1.17 – Повідомлення про вдале видалення інформації

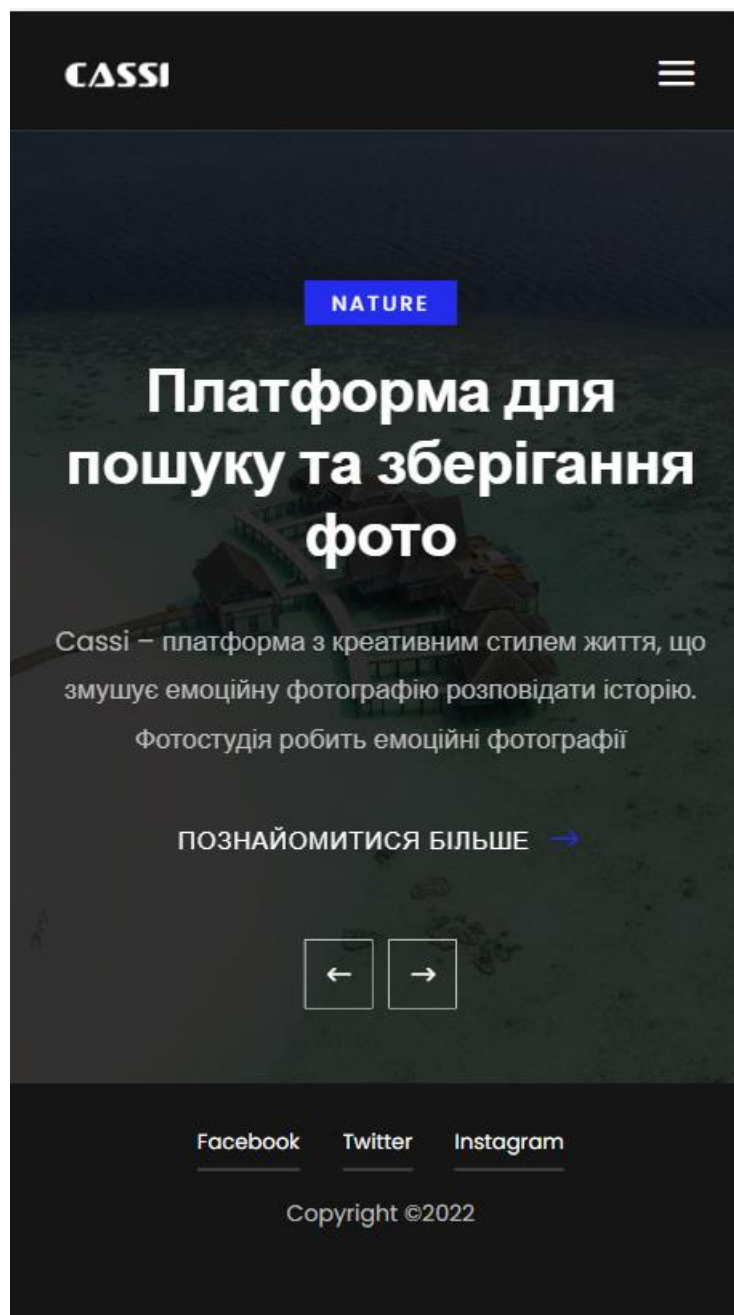
Для видалення інформації використовується кнопка «Удалить», яка знаходиться праворуч біля проекту. Після видалення інформації буде повідомлення про вдале видалення інформації (рис. 1.17).

#### 1.6.4 Крос-браузерність проекту

Даний проект був розроблений для підтримки на смартфонах, планшетах та інших пристроях з різними розмірами екрану. Для цього були використані медіазапити, щоб сторінки могли адаптуватися.

					РП 05.11.001 ДП ПЗ	Арк.
						32
Змн.	Арк.	№ докум.	Підпис	Дата		

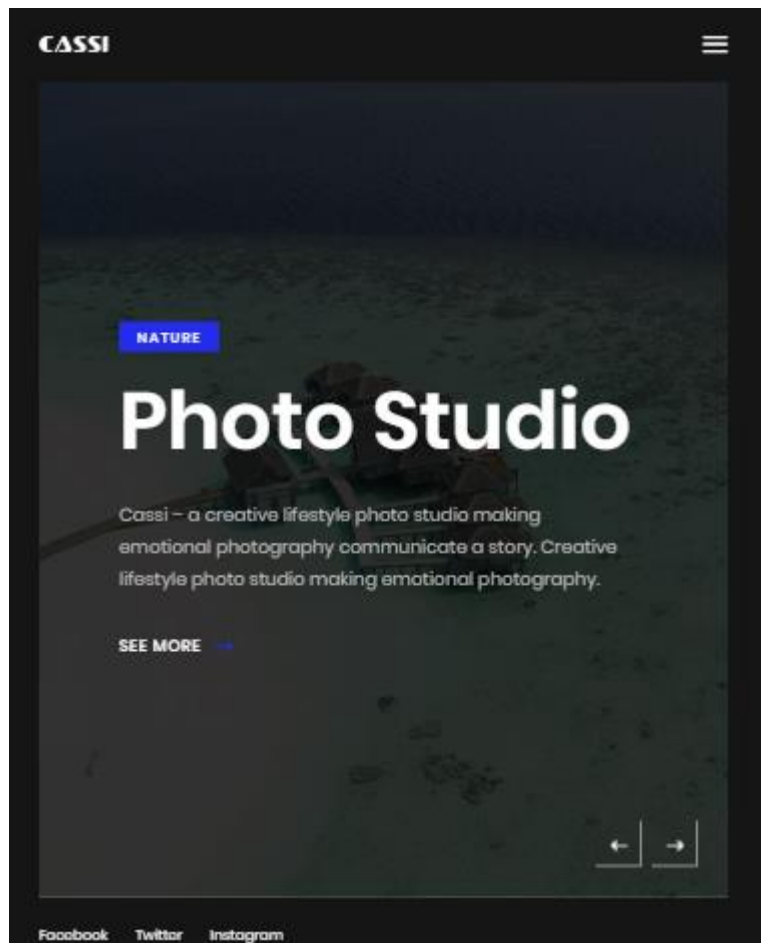
Дизайн головної сторінки для мобільних пристроїв орієнтований на те, щоб блоки на сторінки мали ширину 100% від розмірів екрану та розташувалися в потоці вертикально. Це дає зручність для користувача побачити всю необхідну інформацію(рис. 1.18)



**Рис. 1.18 – Головна сторінка на мобільному пристрої**

Для планшетного варіанту був обраний майже такий дизайн, як і для великого монітору, але планшет повинен розташовуватися вертикально (рис. 1.19).

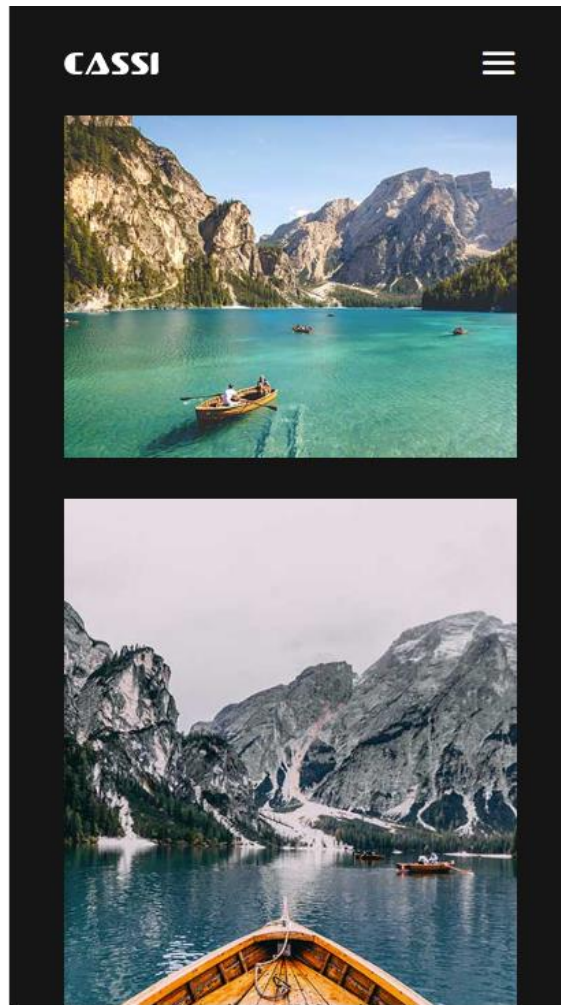
					РП 05.11.001 ДП ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		33



**Рис. 1.19 – Головна сторінка на планшетаі**

Сторінка галереї на мобільних пристроях також розташовує всі елементи вертикально(в даному випадку фотографії), таким чином користувач зможе продивитися фотографії у повному розмірі (рис. 1.20).

					РП 05.11.001 ДП ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		34



**Рис. 1.20 – Вигляд галереї на мобільному пристрої**

Для планшетного варіанту вигляд також орієнтований на вертикальне розташування планшета, щоб зблизити дизайн до десктопного варіанту (рис. 1.21).

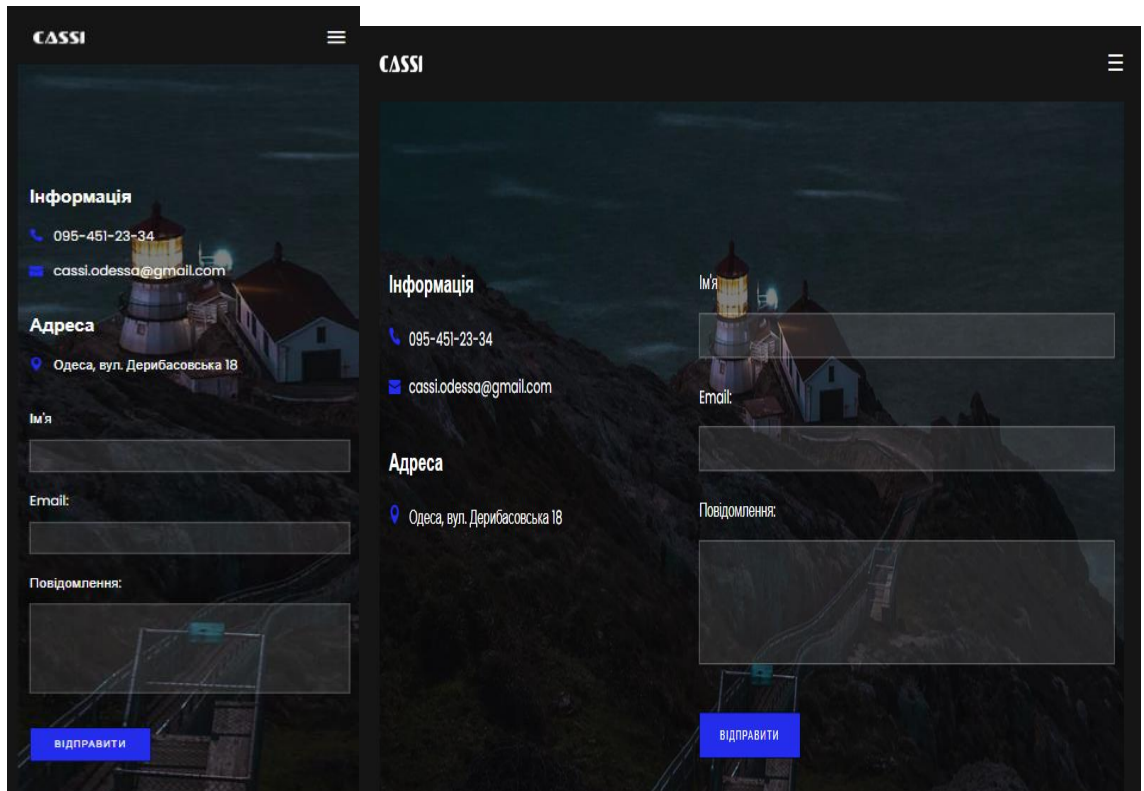
					РП 05.11.001 ДП ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		35



**Рисунок 1.21 – Зовнішній вигляд галереї на планшеті**

Форма для зв'язку також адаптується для користувача під різні пристрої (рис. 1.22).

					РП 05.11.001 ДП ПЗ	Арк.
						36
Змн.	Арк.	№ докум.	Підпис	Дата		



**Рис. 1.22 – Сторінка «Контакти» на мобільному та планшетному пристрої**

Таким чином, всі сторінки є адаптивними, що додає переваги для проекту, розробка адаптиву для сучасного користувача інтернету є найпріоритетнішим, тому що багато користувачів користуються інтернетом через смартфони або планшети.

					РП 05.11.001 ДП ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		37

## 2 ЕКОНОМІЧНИЙ РОЗРАХУНОК

Темою даного дипломного проекту є «Розробка веб-сховища для зберігання і виведення стічних фотографій за рахунок інструментів JavaScript».

Метою даних розрахунків є обчислення вартості виконання науково-дослідної розробки.

Спочатку проведемо розрахунки визначення трудомісткості виконання даної науково-дослідницької розробки. У технологічній структурі науково-дослідних робіт можна виділити декілька самостійних етапів, а саме: розробка технічного завдання, вибір напрямку дослідження, теоретичні і експериментальні дослідження, узагальнення і оцінка результатів.

Розподіл робіт по етапах і видах виконавців вироблений формою, наведено в таблиці 3.1.

Розподіл робіт по етапах і видах виконавців.

Таблиця 3.1.

Етап проведення НДР	Вигляд робіт	Посада виконавця
Розробка технічного завдання (ТЗ)	1.Складання і затвердження ТЗ для НДР «Розробка веб-сховища для зберігання і виведення стічних фотографій за рахунок інструментів JavaScript»	Дипломник, керівник
Вибір напрямку дослідження	1. Збір і вивчення науково-технічної літератури, технічної документації і інших матеріалів, на основі яких будуватиметься робота. 2. Вибір напрямку проведення досліджень для подальшої розробки.	Дипломник керівник

	3. Розробка плану проведення досліджень для подальшої розробки. .	
Теоретичні і експериментальні дослідження	1. Технічний огляд проекту 2. Довідкова інформація для користувача	Дипломник керівник консультанти
Узагальнення і оцінка результатів досліджень	11. Узагальнення результатів попередніх етапів роботи. 2Складання і оформлення звіту. Розгляд результатів проведеною НДР і прийняття результатів в цілому.	Дипломник керівник консультанти

*Оцінка тривалості виконання робіт.* В умовах відсутності нормативної бази тривалість виконання окремих робіт розраховується на основі вірогідних оцінок робіт, що задаються виконавцями.

Таблиця 3.2.

Очікувана трудомісткість робіт.

Вигляд роботи	Очікуваний час виконання (дні)
1. Складання і затвердження ТЗ для НДР по розробці «Розробка цифрового вимірювального приладу для визначення індуктивності, ємності та частоти»	2
2. Збір і вивчення науково – технічної літератури, технічної документації і інших матеріалів.	4
3. Вибір напрямку проведення досліджень і способів вирішення поставлених завдань. Розробка плану проведення досліджень для подальшої розробки.	2
4.Технічний огляд проекту	7
5.Довідкова інформація для користувача	6

7. Економічна частина	2
18. Охорона праці	2
Всього:	25

Результатом виконання НДР є науково-технічна продукція, що є закінчені науково – дослідницькі роботи, виконані відповідно до вимог, передбачених договором, і прийнятими замовником. Виходячи з особливостей створення науково – технічної продукції і її залежності від інтелектуальної праці, розрахунок собівартості і ціни виконання НДР включає наступні статті витрат: витрати на матеріали, основна і додаткова заробітна плата, відрахування до єдиного соціального фонду страхування, витрати на роботи, що виконуються сторонніми організаціями, і деякі інші.

1). Витрати на матеріали складаються з вартості придбаного папіру формату А4 і становитимуть 242 грн.

2) До витрат «Основна заробітна плата» відносяться оплата праці виконавців, безпосередньо притягнених до її виконання. Розмір основної зарплати встановлюється виходячи з чисельності різних категорій виконавців, трудомісткості, що витрачається ними на виконання різних видів робіт, а також їх середньої заробітної плати (ставки) за один робочий день. Відповідно до статті 8 «Закону про Державний бюджет України на 2022» встановлено мінімальну заробітну плату у місячному розмірі з 1 січня 2022 року - 6500 гривень; мінімальну погодинну тарифну ставку – 39,26 грн.

Середня зарплата за один робочий день для кожного виконавця визначена по формулі:

$$Зден = п.т.с. * 8;$$

де п.т.с – погодинна тарифна ставка, грн.;

8 – тривалість робочого дня, год.

$$Зден дипломника = 39,26 * 8 = 314,08 \text{ грн.}$$

$$Зден керівника = 50,50 * 8 = 404 \text{ грн.}$$

$$Зден консультантів = 50,50 * 8 = 404 \text{ грн.}$$

					РП 05.11.002 ДП ПЗ	Арк.
						40
Змн.	Арк.	№ докум.	Підпис	Дата		

Витрати на основну заробітну плату, НДР, що включаються в собівартість, приведені в таблиці 3.3.

Таблиця 3.3.

Витрати на основну заробітну плату.

Виконавець	Погодинна тарифна ставка, грн	Денна ставка, грн	Трудомісткість робочих днів	Сума основної зарплати, грн
Дипломник	39,26	314,08	25	7222
Керівник	50,50	404	1	404
Консультант по економічній частині	50,50	404	0,2	101
Консультант по охороні праці	50,50	404	0,2	101
Нормоконтроль	50,50	404	0,2	101
Всього (Зо)				7929

3) Витрати на додаткову заробітну плату визначаються у відсотках від основної. У наукових закладах додаткова заробітна плата складає 10-20% від основної заробітної плати.

$$Зд = 15 \% * Зо : 100\% ;$$

$$Зд = 7929,00 * 0,15 = 1189,35 \text{ грн}$$

4) До складу собівартості НДР включаються податки, збори і інші обов'язкові платежі, встановлені системою оподаткування що діє. Сума до єдиного соціального внеску складає:

$$Зесв = 0,22 * (Зо + Зд) = 0,22 * (7929,00 + 1189,35) = 9118,35 * 0,22 = 2006,04 \text{ грн.}$$

5) До накладних витрат відносять витрати на управління і господарське обслуговування, що відноситься до всіх виконуваних НДР. По цій статті враховується заробітна плата апарату управління і загальногосподарських служб, витрати на потоковий ремонт будов, устаткування і інструментів, амортизаційні відрахування на їх повне відновлення і капітальний ремонт, витрати по охороні праці, витрати на винаходи і раціоналізацію, витрати на науково – технічну інформацію і рекламу, і так далі. Розмір накладних витрат

					РП 05.11.002 ДП ПЗ	Арк.
						41
Змн.	Арк.	№ докум.	Підпис	Дата		

на конкретну НДР визначається у відсотках до її виконання. У наукових закладах накладні витрати складають 40-120% від основної і додаткової заробітної плати.

$$P_{\text{накл}} = (30 + 3д) * 0,5 = (7929,00 + 1189,35) * 0,5 = 4559,18 \text{ грн.}$$

На підставі отриманих даних по окремих статтях витрат складена калькуляція планової собівартості в цілому НДР за формою, приведеною в таблиці 3.4.

Таблиця 3.4.

**Калькуляція планової собівартості**

Статті витрат	Сума, грн
1. Матеріали	242,00
2. Основна заробітна плата	7929,00
3. Додаткова заробітна плата	1189,35
4. Відрахування до єдиного соціального внеску	2006,04
5. Накладні витрати	4559,18
<b>Планова собівартість (Спл)</b>	<b>15925,57</b>

У наукових організаціях разом з плановою собівартістю визначають величину планового прибутку і договірну ціну НДР.

Плановий прибуток визначений по формулі:

$$Ппл = 0,1 * Спл = 0,1 * 15925,57 = 1592,56 \text{ грн.}$$

Де 0,1 – норматив, який враховує граничний рівень рентабельності, встановлений чинним законодавством для науково-технічної продукції.

Договірна ціна визначається по формулі:

$$Ц_{\text{нір}} = Спл + Ппл = 15925,57 + 1592,56 = 17518,13 \text{ грн.}$$

Ціну реалізації встановлюємо з урахуванням ПДВ

$$Цр = Ц_{\text{нір}} + ПДВ = 17518,13 + (0,2 * 17518,13) = 21021,76 \text{ грн.}$$

## 3 ОХОРОНА ПРАЦІ

### Вступ

В даному розділі дипломного проекту розглядається питання охорони праці програміста . Оператори і програмісти зіштовхуються із впливом таких фізично небезпечних і шкідливих виробничих факторів, як підвищений рівень шуму, підвищена температура зовнішнього середовища, відсутність або недостатня освітленість робочої зони, електричний струм, статична електрика тощо.

На робочому місці програміста повинні бути створені умови для безпечної та високопродуктивної праці.

### 3.1 Розробка заходів з охорони праці

#### 3.1.1. Вимоги до приміщень

Об'ємно-планувальні рішення будівель та приміщень для роботи з ВДТ мають відповідати вимогам ДСанПІН 3.3.2.007-98. Розміщення робочих місць з ВДТ ЕОМ і ПЕОМ у підвальних приміщеннях, на цокольних поверхах заборонено. Площа на одне робоче місце становить не менше 6,0 м<sup>2</sup>, а об'єм – не менше ніж 20,0 м<sup>3</sup>. У приміщеннях слід щоденно робити вологе прибирання. Вони повинні бути оснащені аптечками першої медичної допомоги. При приміщеннях мають бути обладнанні побутові приміщення для відпочинку.

#### 3.1.2 Освітлення.

Приміщення для роботи з ВДТ повинні мати природне та штучне освітлення, відповідно до ДБН В.2.5-28-2006. У приміщеннях, призначених для роботи з відео терміналами, доцільно, щоб вікна були орієнтовані на північ або північний захід. На вікнах повинні бути штора або жалюзі, що регулюють рівень освітленості і захищають від прямого влучення сонячних променів на робоче місце. Приміщення для роботи з ВДТ повинні мати природне та штучне освітлення, відповідно до ДБН В.2.5-28-2006. Для штучного освітлення у

					РП 05.11.003 ДП ПЗ	Арк.
						43
Змн.	Арк.	№ докум.	Підпис	Дата		

приміщенні використовуються люмінесцентні лампи типу ЛБ, які в порівнянні з лампами розжарювання мають ряд істотних переваг: за спектральним складом світла вони близькі до природного світла, мають підвищену світлову віддачу ( у 2-5 разів вищу, ніж у ламп розжарювання); мають триваліший термін служби – до 10 тис годин.. Допускається застосування ламп розжарювання у світильниках місцевого

При кольоровому оформленні виробничих і допоміжних приміщень необхідно враховувати орієнтацію їхніх вікон стосовно частин світу і використовувати гармонійне сполучення кольорів. Для стін і робочих поверхонь використовують мало насичені ( основні) кольори, для невеликих помешкань або ділянок, що рідко потрапляють у поле зору працюючих, а також для створення контрастності – кольори середньої насиченості ( допоміжні), для маленьких по площі поверхонь – насичені ( акценти) – як функціональне фарбування. Стелі у всіх приміщеннях повинні бути білими. Поверхні устаткування в приміщеннях повинні бути матовими або напівматовими, для виключення випадку відблисків світла в очі працюючого, а стіни бути пофарбованими фарбами пастельних тонів.

### 3.1.3 Гігієнічні нормування параметрів повітря робочої зони.

На робочих місцях мають забезпечуватись оптимальні значення параметрів мікроклімату: температури, відносної вологості й рухливості повітря – ГОСТ 12.1.005-88, СН 4088-86.

Параметри мікроклімату	значення параметри	
	Взимку	влітку
Температура, С <sup>0</sup>	22-24	23-25
Відносна вологість, %	40-60	40-60
Швидкість руху повітря, м/с	0,1	0,1-0,2

Для підтримки в приміщеннях нормального, що відповідає гігієнічним вимогам складу повітря, видалення з нього шкідливих газів, пилу використовують вентиляцію, кондиціонери.

### 3.1.4 Розміщення робочих місць у приміщенні.

Робочі місця повинні бути розташовані так, щоб у поле зору працюючого не попадали поверхні, що мають властивість віддзеркалювання, вікна освітлювальні прилади. Відеотермінали повинні встановлюватися під кутом 90-100 градусів від вікон, так, щоб світло падало з боку. Робочі місця з ВДТ доцільно розміщати в глибині приміщення. Розташування відео терміналу, при якому працюючий звернений обличчям або спиною до вікон, неприпустимо при будь-якому способі реалізації загального висвітлення, як прямим, так і відбитим світлом.

Робочий стіл повинен регулюватися по висоті в границях 680-800 мм, а ширина – забезпечувати можливість виконання операцій в зоні досяжності моторного поля. Рекомендовані розміри столу: висота 725 мм, ширина 600-1400 мм, глибина 800-1000 мм. Робочий стілець повинен бути оснащений підйомно-поворотним пристроєм для регулювання висоти сидіння і спинки, а також кута її нахилу. Регулювання кожного параметра повинне вироблятися легко, бути незалежним і надійно фіксуватися.

Розташування екрана ВДТ має забезпечувати зручність зорового спостереження у вертикальній площині під кутом  $+30^{\circ}$  до нормальної лінії погляду працюючого.



Клавіатуру слід розташовувати на поверхні столу на відстані 100...300 мм від краю, звернутого до працюючого.

					РП 05.11.003 ДП ПЗ	Арк.
						45
Змн.	Арк.	№ докум.	Підпис	Дата		

### 3.2 Пожежна безпека.

Під пожежною безпекою розуміють систему державних і суспільних заходів, спрямованих на охорону від вогню людей і власності. Пожежна безпека приміщень, що мають електричні мережі, регламентується ГОСТ 12.1.033-81, ГОСТ 12.1.004-85. Робота оператора ЕОМ повинна вестися в приміщенні, що відповідає категорії Д пожежної безпеки ( негорючі речовини й матеріали в холодному стані).

Пожежна безпека об'єкта забезпечується:

- Системою запобігання пожежі;
- Системою протипожежного захисту;
- Організаційно-технічними заходами.

Всі приміщення повинні бути забезпечені первинними засобами пожежогасіння: пожежним водопостачанням ( пожежні крани ПК), пожежні щити з набором пожежного інструменту, вуглекислотними або порошковими вогнегасниками. У випадку виникнення пожежі необхідно відключити електроживлення, викликати по телефону 101 пожежну команду, евакуювати людей із приміщення відповідно до плану евакуації і приступити до ліквідації пожежі.



					РП 05.11.003 ДП ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		46

## ВИСНОВКИ

В ході дипломного проектування була поставлена і досягнута наступна мета. Безпосередньо в процесі розробки сайту на початковому етапі його проектування були вивчені особливості сайтом, такого роду, викладені основні функції, які він повинен надавати користувачам. Потім було наведено обґрунтування вибору основних технології розробки проекту, таких як:

1. мову гіпертекстової розмітки HTML;
2. каскадні таблиці стилів CSS;
3. система управління базами даних MySQL;
4. мову програмування JavaScript.

В результаті роботи був розроблений web-проект наступній функціональності:

1. надання користувачам такої інформації: загальний опис проекту;
2. фотогалерея;
3. можливість організації зворотного зв'язку за допомогою сторінки контактів;
4. присутність адміністративної частини, за допомогою якої ведеться управління інформацією фотогалереї.

Тестування проекту проводилося за допомогою програмного пакета Microsoft Visual Studio.

					РП 05.11.000 ДП ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		47

## СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Пауэлл Т.А. Полное руководство по HTML.- Мн.: Попурри, – 2001. – 912с.
2. Jaworsky J. Mastering JavaScript and Jscript. – New York: Изд-во Sybex, 2000. – 940с.
3. Николенко Д.В. Практические занятия по Java Script.- М.: Наука и техника, 2000. – 128с.
4. Вагнер Р., Вайк А. Java Script.- К.: ДиаСофт, – 2001. – 464с.
5. Брандебау Дж. Java Script – Сборник рецептов.- СПб.: Питер, – 2000. – 416с.
6. Вуд Л. Web-графика. – СПб: «Диалектика», 2001. –488с.
7. Мюррей Д., Ван Райпер У. Энциклопедия графических форматов. – К.: ВНУ, 1997. – 672с.
8. Кирсанов Д. Web-дизайн.- СПб.: Символ-Плюс, – 2000. – 376с.
9. Буковецкая О.А. Дизайн текста: шрифт, эффекты, цвет. – М.: ДМК, 2000. – 304с.
10. Вин Дж. Искусство Web-Дизайна. - СПб: Изд-во «Питер», 2002. – 360с.
12. Шапошников И.В. Интернет программирование.- СПб.: ВНУ, – 2000. – 224с.
13. Федорчук А. Как создаются Web-сайты.- СПб.: Питер, – 2000. – 224с.
14. Леонтьев Б.К. Web-дизайн: тонкости, хитрости, секреты. – М.: Майор, 2001. – 176с.

					РП 05.11.000 ДП ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		48