

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
НАЦІОНАЛЬНИЙ АВІАЦІЙНИЙ УНІВЕРСИТЕТ

ДОПУСТИТИ ДО ЗАХИСТУ
Завідувач випускової кафедри
_____ Аліна САВЧЕНКО
«__» _____ 2024 р.

КВАЛІФІКАЦІЙНА РОБОТА
(ПОЯСНЮВАЛЬНА ЗАПИСКА)

ЗДОБУВАЧА ОСВІТНЬОГО СТУПЕНЯ “БАКАЛАВР”

Тема: «Вебхостинг зображень з колективним доступом»

Виконавець: Антон БИТЬКО

Керівник: к.т.н., доцент Олег ЗУДОВ

Нормоконтролер: к.т.н., доцент Вікторія СИДОРЕНКО

НАЦІОНАЛЬНИЙ АВІАЦІЙНИЙ УНІВЕРСИТЕТ

Факультет *комп'ютерних наук та технологій*

Кафедра *комп'ютерних інформаційних технологій*

Спеціальність *122 «Комп'ютерні науки»*

Освітньо-професійна програма *«Інформаційні технології проектування»*

ЗАТВЕРДЖУЮ:

завідувач кафедри КІТ

Аліна САВЧЕНКО

(підпис)

«__» _____ 2024 р.

ЗАВДАННЯ

на виконання кваліфікаційної роботи

Битька Антона Віталійовича

(прізвище, ім'я, по батькові здобувача вищої освіти в родовому відмінку)

1. Тема роботи: «Вебхостинг зображень з колективним доступом», затверджена наказом ректора від «05» квітня 2024р. № 517/ст.
2. Термін виконання роботи: з 06 травня 2024 року по 16 червня 2024 року.
3. Вихідні дані до роботи: вебхостинг зображень з колективним доступом на основі технологій PHP/JS/HTML5/CSS/SQL та фреймворку Laravel.
4. Зміст пояснювальної записки: 1. Огляд та аналіз предметної області, а також існуючих вебхостингів зображень. 2. Вибір і опис використаних мов програмування, бібліотек та інших технологій. 3. Розробка сайту вебхостинга зображень з колективним доступом.
5. Перелік обов'язкового ілюстративного матеріалу: приклади інтерфейсу існуючих вебхостингів зображень, знімки екрану інтегрованого середовища розробки PhpStorm та системи управління базами даних MySQL, а також демонстрація інтерфейсу робочої програми (вебхостингу).

6. Календарний план-графік

№ з/п	Завдання	Термін виконання	Підпис керівника
1.	Огляд та аналіз предметної області, а також існуючих фото вебхостингів. Написання першого розділу та представлення керівнику.	06.05.2024 - 13.05.2024	
2.	Вибір та опис використаних мов програмування, бібліотек та інших технологій. Написання другого розділу та представлення керівнику.	14.05.2024 - 21.05.2024	
3.	Розробка сайту вебхостинга зображень з колективним доступом. Написання третього розділу та представлення керівнику.	22.05.2024 - 29.05.2024	
4.	Загальне редагування та друк пояснювальної записки.	30.05.2024 - 02.06.2024	
5.	Проходження нормоконтролю та перепліт пояснювальної записки.	03.06.2024 - 05.06.2024	
6.	Розробка тексту доповіді. Оформлення графічного матеріалу для презентації.	06.06.2024 - 10.06.2024	

7. Дата видачі завдання «06» травня 2024 р.

Керівник кваліфікаційної роботи _____

(підпис керівника)

Олег ЗУДОВ

Завдання прийняв до виконання _____

(підпис здобувача вищої освіти)

Антон БИТЬКО

РЕФЕРАТ

Пояснювальна записка до кваліфікаційної роботи на тему: «Вебхостинг зображень з колективним доступом» містить: 49 сторінок, 29 рисунків, 12 інформаційних джерел.

Об'єктом дослідження кваліфікаційної роботи є фото вебхостинги, їх можливості, а також переваги та недоліки.

Предмет дослідження – вебхостинг зображень, розроблений з використанням сучасних бібліотек та фреймворків.

Мета кваліфікаційної роботи – розробка власного вебхостингу зображень з розподілом доступу для користувачів.

Методи дослідження – логічний аналіз, синтез, порівняльний аналіз, обробка літературних джерел, моделювання та реалізація програмного забезпечення вебхостингу.

Наукова новизна – створений вебхостинг містить корисні функції, які відсутні в більшості аналогів, наприклад: створення власного альбому користувача в спільному груповому альбомі.

Результат роботи: може бути використаний для зручного зберігання, організації та поширення зображень, а також для спільного користування вебхостингом через завантаження фото у спільні групові альбоми та ін.

Для розробки власного фото вебхостингу було знайдено та використано найбільш ефективні технології, особливо фреймворк Laravel для мови програмування PHP, завдяки якому розробка сайту була швидка та зручна для розуміння.

Ключові слова: PHP, ВЕБСАЙТ, WEB HOSTING SERVICE, LARAVEL, ХОСТИНГ ЗОБРАЖЕНЬ, PHPSTORM, SQL, JAVASCRIPT.

ЗМІСТ

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ, СКОРОЧЕНЬ, ТЕРМІНІВ	6
ВСТУП.....	7
РОЗДІЛ 1. ОГЛЯД ТА АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ, А ТАКОЖ ІСНУЮЧИХ ВЕБХОСТИНГІВ ЗОБРАЖЕНЬ	10
1.1. Поняття хостингу та вебхостингу	10
1.2. Види вебхостингів.....	12
1.3. Огляд існуючих фото вебхостингів, їх переваг та недоліків	14
ВИСНОВКИ ДО РОЗДІЛУ 1	21
РОЗДІЛ 2. ВИБІР І ОПИС ВИКОРИСТАНИХ МОВ ПРОГРАМУВАННЯ, БІБЛІОТЕК ТА ІНШИХ ТЕХНОЛОГІЙ	22
2.1. Опис використаних мов програмування, а також HTML і CSS	22
2.2. Фреймворк Laravel та основні бібліотеки.....	29
2.3. Docker та Docker Compose, Git та GitHub, а також PhpStorm.....	31
ВИСНОВКИ ДО РОЗДІЛУ 2.....	35
РОЗДІЛ 3. РОЗРОБКА САЙТУ ВЕБХОСТИНГА ЗОБРАЖЕНЬ З КОЛЕКТИВНИМ ДОСТУПОМ	36
3.1. Загальна структура вебхостингу	36
3.2. Архітектура бази даних	40
3.3. Інтерфейс робочого додатку.....	43
ВИСНОВКИ ДО РОЗДІЛУ 3	47
ВИСНОВКИ	48
СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ	49

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ, СКОРОЧЕНЬ, ТЕРМІНІВ

PHPSTORM	– Інтегроване середовище розробки для PHP та JS
JS	– JavaScript – мова програмування
PHP	– Hypertext Preprocessor – мова програмування
ФРЕЙМВОРК (FRAMEWORK)	– Інфраструктура програмних рішень, що полегшує розробку складних систем
LARAVEL	– PHP-фреймворк
СУБД	– Система управління базами даних
SQL	– Structured query language – мова програмування для взаємодії з базами даних
HTML	– HyperText Markup Language – стандартизована мова розмітки документів для браузера
CSS	– Cascading Style Sheets – каскадні таблиці стилів веб-сторінок
AJAX	– Asynchronous JavaScript and XML
URL	– Uniform Resource Locator – стандартизована адреса певного ресурсу в Інтернеті
ORM	– Object-relational mapping – технологія, що зв'язує бази даних з концепціями ООП
ООП	– Об'єктно-орієнтоване програмування

ВСТУП

Вебхостингові сервіси в сучасному світі відіграють ключову роль у забезпеченні доступу до веб-сайтів та зберіганні великої кількості різноманітного контенту. Їхні можливості та характеристики є важливими для розуміння принципів функціонування Інтернету та його інфраструктури. Існують різні види вебхостингу, такі як: спільний хостинг (англ. Shared Hosting), віртуальний виділений сервер (англ. VPS), власний сервер (англ. Dedicated Server) та хмарний вебхостинг (англ. Cloud Hosting або Cloud Storage), що відповідають різним потребам користувачів.

Окрім загального вебхостингу, особливим видом є вебхостинги зображень (англ. Photo Hosting), які спеціалізуються на зберіганні та організації фотографій та відео. Прикладом таких вебхостингів є Google Photos, що надає можливість зручного зберігання та спільного використання медіафайлів для користувачів.

Компанія, що займається наданням послуг розміщення обладнання, даних та веб-сайтів на своїх технічних майданчиках, називається Хостинг компанією (або хостингова компанія, хостер, хостинг-провайдер, вебхостер, англ. Hosting Service Provider). Однією з найпопулярніших таких компаній, наприклад, є Amazon Web Services (AWS).

Сервіси вебхостингу зображень стають все більш важливими для користувачів, оскільки вони дозволяють зберігати та організовувати великі обсяги фотографій та відео, надаючи зручний доступ до цього вмісту з різних пристроїв через Інтернет.

Актуальність теми кваліфікаційної роботи «Вебхостинг зображень з колективним доступом» обумовлена сучасними тенденціями в цифровій комунікації та потребами користувачів в зручному та безпечному зберіганні, організації та обміні візуальним контентом, зокрема – фотографіями. Сучасні користувачі потребують можливостей для спільного доступу до зображень, що сприяє колективній роботі та обміну контентом. Це особливо актуально для

груп користувачів, таких як студенти, співробітники компаній, члени сім'ї або просто спільноти за інтересами. Таким чином, дослідження і розробка вебхостингу зображень з колективним доступом є досить актуальним завданням, яке відповідає сучасним вимогам ринку та запитам користувачів.

Об'єктом дослідження кваліфікаційної роботи є фото вебхостинги, їх можливості, а також переваги та недоліки.

Предмет дослідження – вебхостинг зображень, розроблений з використанням сучасних бібліотек та фреймворків.

Мета кваліфікаційної роботи – створення власного вебхостингу зображень з колективним і особистим доступом та зручним інтерфейсом, який надає можливості зберігання, поширення та організації фото, а також створення групових і особистих альбомів.

Відповідно до поставленої мети роботи визначено основні **завдання дослідження**:

- Провести аналіз сучасних технологій вебхостингу: дослідити існуючі рішення для зберігання та управління зображеннями, вивчити їх переваги та недоліки, а також проаналізувати найбільш успішні та популярні платформи (наприклад, Google Photos, Flickr, Instagram).

- Вивчити вимоги користувачів: дослідити потреби і очікування, від вебхостингу зображень, потенційних користувачів. Включити аспекти зручності використання, безпеки, швидкості доступу та можливостей для спільної роботи.

- Розробити архітектуру вебхостингу: створити архітектурну модель вебхостингу зображень, що включатиме серверну частину, базу даних, механізми зберігання зображень та засоби для забезпечення колективного доступу.

- Розробити інтерфейс користувача: створити інтуїтивно зрозумілий та зручний користувацький інтерфейс, що дозволить легко завантажувати, переглядати, організовувати та ділитися зображеннями з іншими користувачами.

- Протестувати та оптимізувати створений продукт: провести тестування створеного вебхостингу для виявлення та усунення помилок, а також оптимізувати продуктивність системи.

Наукова новизна полягає в тому, що дана робота об'єднує в собі використання сучасних технологій програмування та потреби користувачів в зручному та ефективному керуванні своїми зображеннями. В результаті, можна буде створити фото вебхостинг, використовуючи найбільш ефективні методи реалізації сучасних вебсайтів, додавши у проект деякі корисні функції, наприклад – створення власного альбому користувача в спільному груповому альбомі.

Практичне значення отриманих результатів. Результати кваліфікаційної роботи рекомендується використовувати зручного зберігання, організації та поширення зображень, а також для спільного користування вебхостингом через завантаження фото у спільні групові альбоми або можливості поставити «лайк» на фото іншого користувача.

РОЗДІЛ 1

ОГЛЯД ТА АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ, А ТАКОЖ ІСНУЮЧИХ ВЕБХОСТИНГІВ ЗОБРАЖЕНЬ

1.1. Поняття хостингу та вебхостингу

Хостинг (або вебхостинг, англ. *hosting*, від слова *host* – приймати гостей) – це послуга, що дозволяє організаціям та окремим особам розміщувати свої вебсайти чи вебсторінки в Інтернеті. Поняття хостингу включає в себе широкий спектр послуг із використанням різного апаратного та програмного забезпечення. Зазвичай під поняттям послуги хостингу мають на увазі, як мінімум, послугу розміщення файлів сайту сервері, на якому запущене програмне забезпечення, необхідне для обробки запитів до цих файлів (вебсервер).

Як правило, до послуг хостингу вже входить надання місця для поштової кореспонденції, баз даних, DNS (система доменних імен, англ. *domain name system*) файлового сховища тощо, а також підтримка функціонування відповідних сервісів, однак вони можуть надаватися і окремо. А саме, послуга може бути обмежена розміщенням поштової кореспонденції та відповідного програмного забезпечення (поштовий хостинг), клієнтських файлів (файловий хостинг), виключно відео файлів (відео хостинг) або інших файлів певного типу та за певними умовами.

Послуги хостингу можуть надаватися у пакеті з іншими інформаційними послугами, такими як реєстрація доменного імені, створення сайту, надання додаткового програмного забезпечення тощо.

Провайдерами хостингу можуть виступати як компанії, що спеціалізуються на цих послугах («хостери»), так і великі провайдери

Кафедра КІТ				НАУ 24 01 78 000 ПЗ			
	ШБ			ОГЛЯД ТА АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ	Літ.	Аркуш	Архівів
Виконав	Битько А.В.					10	49
Керівник	Зудов О. М.				ТП-416Б – 122		
Норм-контр.	Сидоренко В.М.						

інформаційних послуг, що спеціалізуються на інших послугах (такі як Google, Microsoft, Amazon та інші).

Розрізняють безкоштовний та платний хостинг. Безкоштовні «хостери» заробляють на тому, що розміщують рекламу на своїх сайтах або на наданні інших платних послуг (у пакеті з безкоштовними або опціонально).

Вебхостинг є одним із типів інтернет-хостингу, який дозволяє окремим особам та організаціям зробити свій вебсайт доступним через Інтернет. Сфера послуг вебхостингу часто змінюється. Однак основними є вебсторінки та хостинг дрібних файлів, де файли можуть бути завантажені через File Transfer Protocol (англ. *FTP*) або вебінтерфейси. Файли, як правило, доставляються в Інтернеті «як є» або з мінімальною обробкою.

Багато Інтернет-провайдерів (англ. *ISP*) пропонують дану послугу безкоштовно для абонентів. Приватні особи та організації можуть також отримати вебсторінку хостингу від альтернативних постачальників послуг. Особистий вебсайт хостинг, як правило, безкоштовний або недорогий.

Односторінковий хостинг використовується, як правило, у персональних вебсторінках. Комплексний сайт вимагає більш всеосяжний пакет, який забезпечує підтримку баз даних і розробки додатків платформи (наприклад, PHP, Java, ColdFusion, або ASP.NET). Адже саме дані послуги дозволяють клієнтам написати або встановити скрипти для додатків, таких як форуми та управління контентом. Крім того, Secure Sockets Layer (рівень захищених сокетів, англ. *SSL*) зазвичай використовується для електронної комерції.

Хостинг може також забезпечити інтерфейс або панелі керування для управління вебсервера та установки скриптів, а також інші модулі та обслуговування додатків, таких як електронна пошта. Вибір правильного вебхостингу є важливим кроком для успішного функціонування вашого вебсайту в Інтернеті.

1.2. Види вебхостингів

Основні типи вебхостингів включають:

- Спільний хостинг (англ. *Shared Hosting*): на одному сервері розміщуються декілька вебсайтів, які спільно використовують ресурси цього сервера (процесор, пам'ять, дисковий простір). Переваги: низька вартість, простота управління. Недоліки: обмежені ресурси, можливий вплив на продуктивність через інші сайти.

- Віртуальний приватний сервер (англ. *VPS Hosting*): сервер ділиться на віртуальні машини, кожна з яких працює як окремий сервер. Переваги: більший контроль та ресурси порівняно з спільним хостингом, висока гнучкість. Недоліки: дорожче ніж спільний хостинг, вимагає більше технічних знань.

- Виділений сервер (англ. *Dedicated Hosting*): весь сервер виділяється одному клієнту. Переваги: повний контроль над сервером, висока продуктивність. Недоліки: висока вартість, потреба в адмініструванні сервера.

- Хмарний хостинг (англ. *Cloud Hosting*): використовуються ресурси кількох серверів, об'єднаних в одну систему (хмару). Переваги: висока надійність, масштабованість, оплата за фактичне використання ресурсів. Недоліки: може бути складним в управлінні, вартість залежить від використання ресурсів.

- Реселерський хостинг (англ. *Reseller Hosting*): один клієнт купує хостинг послугу і перепродає її своїм клієнтам. Переваги: можливість створення власного бізнесу хостингу. Недоліки: відповідальність за обслуговування клієнтів.

Типи вебхостингів можуть бути класифіковані не лише за технічними аспектами, але й за призначенням або типом вмісту, який вони підтримують. Декілька основних типів вебхостингів за призначенням:

- Фото вебхостинг, використовується для зберігання, організації та спільного доступу до фотографій. Серед особливостей має: високий обсяг зберігання, швидке завантаження та відображення зображень, інструменти для

організації та пошуку фото, підтримка різних форматів зображень. Прикладами є: Flickr, Google Photos, SmugMug.

- Відеохостинг, використовується для зберігання та потокового відтворення відеофайлів. Серед особливостей має: висока пропускна здатність, підтримка різних форматів відео, можливість потокового відтворення, інструменти для редагування та організації відео. Прикладами є: YouTube, Vimeo, Dailymotion.

- Хостинг для файлообмінників, використовується для зберігання та обміну файлами. Серед особливостей має: великі обсяги зберігання, інструменти для завантаження та скачування файлів, підтримка різних типів файлів. Прикладами є: Dropbox, Google Drive, Mega.

- Блог хостинг, використовується для розміщення блогів та статей. Серед особливостей має: платформи для ведення блогів, інтегровані системи управління контентом (англ. *CMS*), соціальні інтеграції. Прикладами є: WordPress, Blogger, Medium.

- Хостинг електронної комерції (англ. *E-commerce*), використовується для розміщення онлайн-магазинів. Серед особливостей має: інструменти для управління продуктами, кошик покупок, платіжні шлюзи, захист даних. Прикладами є: Shopify, BigCommerce, WooCommerce (на WordPress).

- Музичний хостинг, використовується для зберігання та потокового відтворення музичних файлів. Серед особливостей має: інструменти для організації та прослуховування музики, плейлисти, потокове відтворення, підтримка різних аудіоформатів. Прикладами є: SoundCloud, Spotify, Mixcloud.

- Хостинг для форумів, використовується для розміщення онлайн-форумів та дискусійних платформ. Серед особливостей має: інструменти для створення та управління темами, модерація, управління користувачами. Прикладами є: phpBB, vBulletin, Discourse.

- Хостинг для навчання (англ. *E-learning*), використовується для розміщення онлайн-курсів та навчальних платформ. Серед особливостей має: інструменти для управління курсами, інтерактивні завдання, відео-уроки,

відстеження прогресу учнів. Прикладами є: Google Classroom, Teachable, Udeemy.

Кожен тип вебхостингу має свої специфічні вимоги та особливості, які забезпечують оптимальну роботу для конкретних типів вмісту. Вибір відповідного вебхостингу залежить від користувацьких потреб та цілей.

1.3. Огляд існуючих фото вебхостингів, їх переваг та недоліків Google Photos

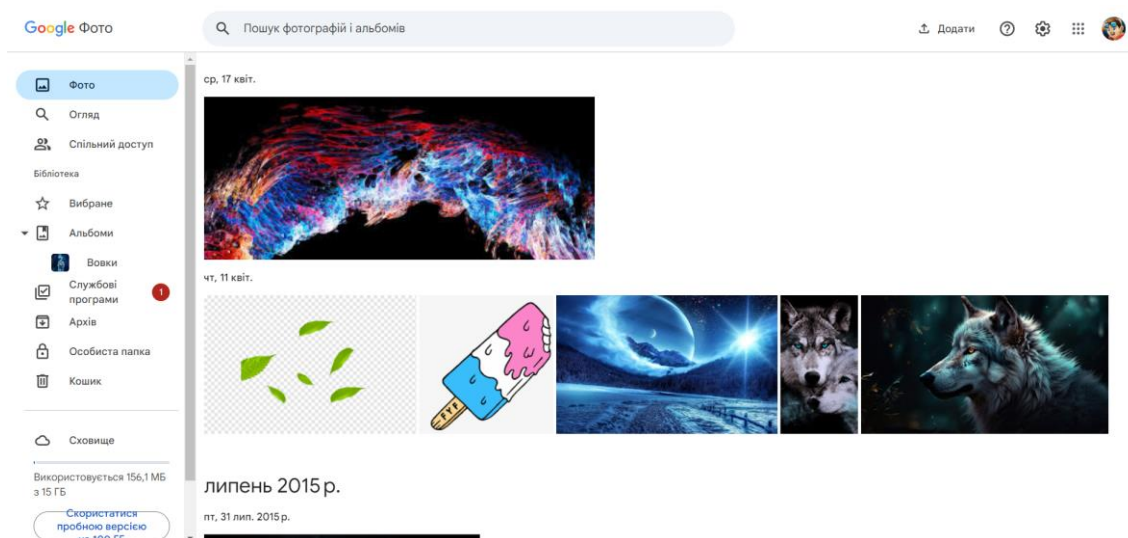


Рис. 1.1. Інтерфейс фото вебхостингу Google Photos

Google Photos - це вебхостинг зображень, який надає безкоштовне зберігання високоякісних фотографій та відео в хмарному середовищі. Сервіс автоматично розпізнає обличчя та об'єкти на зображеннях, має зручний інтерфейс для організації фотоальбомів та надає інструменти для редагування фотографій. Однак після певної кількості збережених фото у високому розширенні, безкоштовний обсяг зберігання може бути обмежений, що може вплинути на користувачів з великими обсягами медіафайлів.

Особливості Google Фото:

- Безкоштовне зберігання у хмарі (англ. *Cloud*): однією з ключових особливостей Google Фото є безкоштовне зберігання фотографій та відео у високому розширенні. Це робить його привабливим варіантом для

користувачів, які хочуть ефективно організувати свої зображення без зайвих витрат.

- Автоматична організація та класифікація: Google Фото використовує машинне навчання для автоматичної організації фотографій за категоріями, такими як обличчя, місця, об'єкти та інші. Це допомагає швидко знаходити та переглядати зображення за певними критеріями.

- Пошук за ключовими словами: Користувачі можуть легко знаходити фотографії за ключовими словами, які автоматично надаються на основі змісту зображення. Наприклад, ви можете знайти всі фотографії з пляжу, собаками або святковими подіями, просто ввівши відповідне ключове слово.

- Спільний доступ до фотографій: ви можете легко ділитися своїми фотографіями з іншими користувачами, надаючи їм доступ до певних альбомів або зображень. Це зручно для спільного перегляду фото або для спільної роботи над проектами.

- Створення колажів та альбомів: Google Фото пропонує автоматичне створення колажів, альбомів та відео із зображень, що допомагає створювати цікаві та естетичні презентації своїх фотоісторій.

- Інтеграція з Google Диском та іншими послугами: ви можете легко зберігати свої фотографії у Google Диску та інтегрувати їх з іншими сервісами Google, такими як Gmail, Google Maps та інші.

Переваги Google Photos:

- Безкоштовне зберігання;
- Редагування та обмін фото.

Недоліки Google Photos:

- Обмеження на безкоштовне зберігання: після досягнення певного обсягу збережених файлів у високому розширенні, Google Фото може вимагати оплату за додатковий простір, що може бути не вигідним для користувачів з великими колекціями медіафайлів.

Узагальнюючи, Google Фото є потужним інструментом для організації, редагування та спільного використання фотографій та відео, що дозволяє

користувачам зберігати та керувати великим обсягом зображень у хмарі безкоштовно.

Flickr

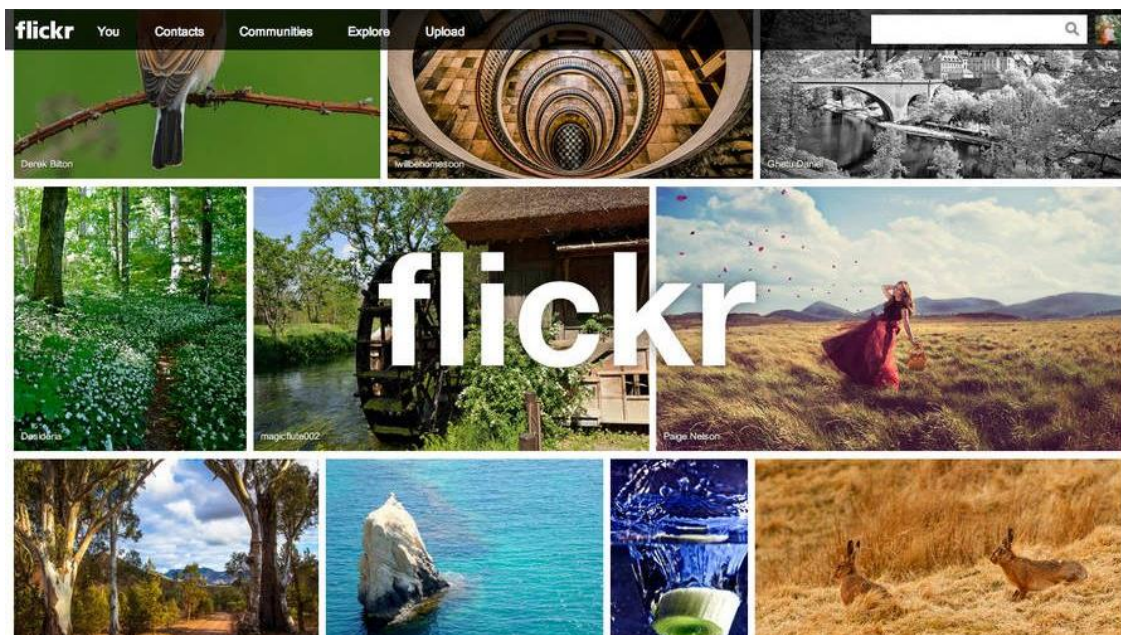


Рис. 1.2. Інтерфейс фото вебхостингу Flickr

Flickr - це одна з найвідоміших і найбільших платформ для зберігання та обміну фотографіями в Інтернеті. Створений у 2004 році компанією Ludicorp, Flickr здобув популярність завдяки своїм функціям спільноти, якості зображень та можливості взаємодії з іншими користувачами.

Особливості Flickr:

- Спільнота фотографів: Flickr має активну та різноманітну спільноту фотографів. Ви можете долучатися до груп за інтересами, обмінюватися досвідом та отримувати зворотний зв'язок щодо своїх фотографій.

- Великий обсяг зберігання: безкоштовний акаунт на Flickr надає до 1000 фотографій, а платні тарифи розширюють цей обсяг до необмеженого зберігання.

- Продаж фотографій: Flickr пропонує можливість продавати свої фотографії через платформу, отримуючи винагороду за використання вашого контенту.

- Відмінна якість зображень: платформа підтримує високу якість фотографій, включаючи RAW-фотографії (мінімально оброблені зображення), що робить її популярною серед професіоналів та фотоаматорів.

- Мобільний додаток: Flickr має зручний мобільний додаток для Android та iOS, що дозволяє завантажувати, редагувати та ділитися фотографіями прямо зі смартфона чи планшета.

Переваги Flickr:

- Безкоштовний обсяг зберігання;
- Велика та різноманітна спільнота фотографів;
- Продаж фотографій;

Недоліки Flickr:

- Реклама в безкоштовній версії: на безкоштовних акаунтах може відобразитися реклама, що може бути неприємною для деяких користувачів.
- Обмеження на безкоштовне завантаження: безкоштовні акаунти мають обмеження на кількість завантажень фотографій за місяць.

Загалом, Flickr є популярним серед фотографів вебхостингом зображень, що забезпечує широкий спектр функцій для зберігання, організації та публікації фотографій онлайн.

SmugMug

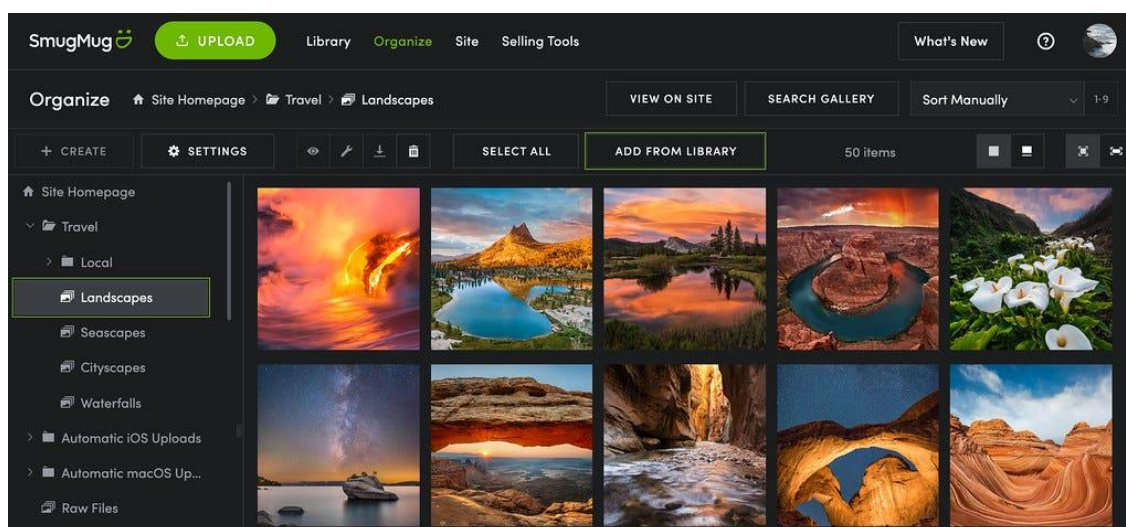


Рис. 1.3. Інтерфейс фото вебхостингу SmugMug

SmugMug - це платформа для хостингу та демонстрації фотографій та відео, яка спрямована на професіоналів та любителів фотографії.

Особливості SmugMug:

- Професійний вигляд: SmugMug пропонує різноманітні дизайни для відображення фотографій користувачів у вигляді портфоліо, галерей або блогів. Це дозволяє користувачам створювати вражаючі та професійні сторінки для демонстрації своїх робіт.

- Висока пропускна здатність: сервіс має велику пропускну здатність, що дозволяє завантажувати великі файли та демонструвати високоякісні зображення без втрати якості.

- Зберігання та резервне копіювання: SmugMug надає безпечне зберігання фотографій та відео на своїх серверах. Крім того, ви можете зробити резервну копію своїх файлів для додаткової безпеки.

- Приватність та керування доступом: SmugMug надає можливості для керування приватністю ваших фотографій. Ви можете налаштувати доступ до зображень для різних користувачів, створюючи приватні альбоми або надаючи обмежений доступ до певних фотографій.

- Інтеграція з соціальними мережами: SmugMug дозволяє легко ділитися своїми фотографіями на соціальних мережах, таких як Facebook, Twitter, Instagram тощо.

- Мобільний доступ: платформа має додатки для мобільних пристроїв, що дозволяє зручно керувати та переглядати ваші фотографії в будь-який час та з будь-якого пристрою.

Переваги SmugMug:

- Професійний Вигляд;
- Велика Пропускна Здатність;
- Зберігання та резервне копіювання;
- Продаж фотографій;
- Керування приватності;
- Мобільний Доступ;

Недоліки SmugMug:

- Вартість: SmugMug вважається досить дорогим сервісом порівняно з іншими аналогічними платформами, особливо для користувачів, які потребують більш простих функцій.

- Складність налаштувань: для новачків може бути складно розібратися у всіх можливостях та налаштуваннях SmugMug, оскільки платформа має багато функцій.

В цілому, SmugMug є потужним інструментом для фотографів, але перед використанням варто врахувати вартість та складність налаштувань як основні фактори, що можуть негативно вплинути на досвід нових користувачів.

500px

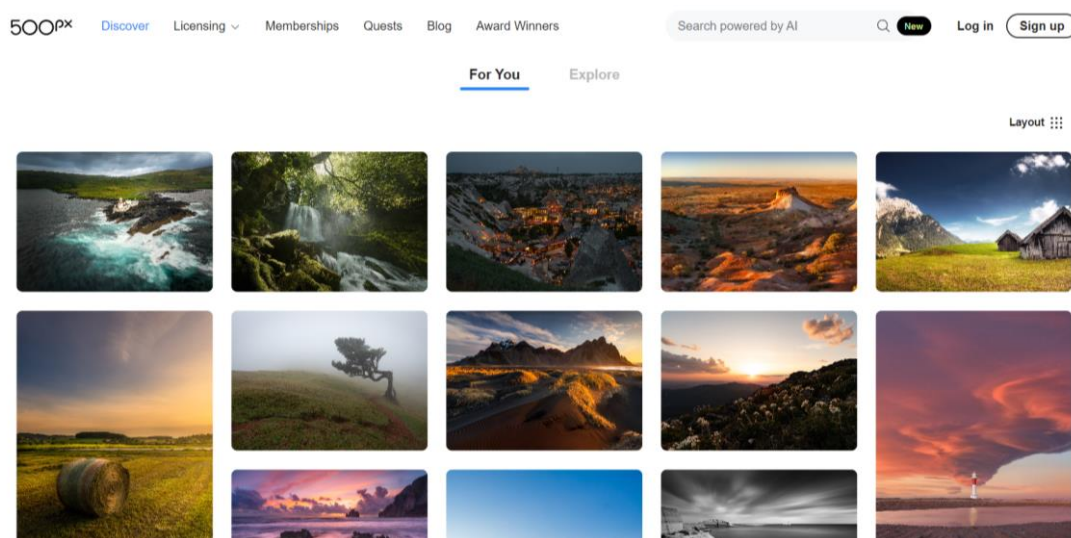


Рис. 1.4. Інтерфейс фото вебхостингу 500px

500px - це онлайн-платформа для фотографів, де вони можуть демонструвати свої творчість, спілкуватися з іншими фахівцями та отримувати оцінки своїх робіт від спільноти.

Особливості 500px:

- Профілі фотографів: кожен користувач може створити власний профіль, де вони можуть завантажувати свої фотографії, вказувати інформацію про себе, свої звіти, контактні дані та інше.

- Оцінка та коментарі: спільнота 500px має можливість оцінювати фотографії за допомогою "лайків" та коментарів. Це дозволяє отримувати зворотній зв'язок від інших фахівців та любителів фотографії.

- Групи та колекції: користувачі можуть приєднатися до різних груп і створювати колекції фотографій за тематикою, стилем чи іншими критеріями. Це дозволяє спілкуватися з однодумцями та ділитися своїми творчими знахідками.

- Комерційні можливості: 500px пропонує можливості для комерційної діяльності, включаючи продаж фотографій через їхню платформу. Користувачі можуть отримувати гонорари за використання своїх зображень у рекламних, друкованих чи онлайн-проектах.

Переваги 500px:

- Велика спільнота фахівців;
- Висока Якість зображень;
- Продаж фотографій;
- Оцінка та рейтинг зображень;
- Співпраця та експозиції;

Недоліки 500px:

- Конкурентне середовище: багато фотографів на платформі означає високий рівень конкуренції, що може ускладнити виділення вашого портфоліо серед інших.

- Можливість крадіжки контенту: хоча 500px робить все можливе для захисту авторських прав, існує ризик крадіжки вашого контенту або його використання без вашої згоди.

- Залежність від алгоритмів рекомендацій фото: успіх на 500px часто залежить від алгоритмів, що визначають популярність вашого контенту та його відображення користувачам.

Загалом, 500px - це велика спільнота фотографів з широкими можливостями, але успіх на платформі залежить від ваших творчих здібностей та вкладених зусиль у розвиток вашого портфоліо та спільноти.

ВИСНОВКИ ДО РОЗДІЛУ 1

Вебхостинги зображень є критично важливим інструментом для сучасних фотографів, художників та аматорів, який надає їм можливість зберігати, організовувати та ділитися своїми роботами з іншими. Такі сервіси пропонують різноманітні функції та можливості, що задовольняють різні потреби користувачів.

Однією з основних переваг вебхостингів зображень є можливість безпечного зберігання великого обсягу фотографій у хмарі. Це забезпечує доступ до зображень з будь-якого пристрою, що має підключення до Інтернету, та зберігає фото від втрати через фізичні пошкодження пристроїв. Крім того, сервіси автоматично організовують та сортують зображення за різними критеріями (датою, місцем, обличчям тощо), що значно полегшує їх пошук та управління.

Деякі вебхостинги зображень надають можливість заробляти гроші на своїх фотографіях шляхом їх продажу або друку на замовлення. Це особливо актуально для професійних фотографів, які шукають додаткові джерела заробітку. Крім того, професійні функції, такі як підтримка RAW-зображень, захист зображень паролями та водяними знаками, а також можливість створення персоналізованих портфоліо, забезпечують додаткову цінність для користувачів.

Розуміння переваг та недоліків різних платформ дозволяє користувачам зробити усвідомлений вибір, що найкраще відповідає їхнім потребам та забезпечує ефективне використання цих сервісів для досягнення своїх цілей у фотографії та мистецтві.

РОЗДІЛ 2

ВИБІР І ОПИС ВИКОРИСТАНИХ МОВ ПРОГРАМУВАННЯ, БІБЛІОТЕК ТА ІНШИХ ТЕХНОЛОГІЙ

2.1. Опис використаних мов програмування, а також HTML і CSS

Для створення власного сайту вебхостингу зображень з різним рівнем доступу для користувачів були використані декілька популярних мов програмування, а саме: PHP, JavaScript та SQL. Також, для покращення інтерфейсу сервісу, були використані такі мови, як HTML та CSS. Усі вони мають свої особливості, переваги та недоліки, тому розглянемо детальніше кожен з них.

PHP

При виборі мови програмування, яка буде відповідати за обробку запитів, зберігання інформації, серверну архітектуру, зв'язок із «фронтом», було вирішено використати саме мову програмування високого рівня – PHP. Вона ідеально підходить для реалізації клієнт-серверних архітектур, таких як вебхостинг.



Рис. 2.1. Логотип мови програмування PHP.

Кафедра КІТ				НАУ 24 01 78 000 ПЗ			
	ПІБ			ВИБІР І ОПИС ВИКОРИСТАНИХ ТЕХНОЛОГІЙ	Літ.	Аркуш	Аркушів
Виконав	Битько А.В.					22	49
Керівник	Зудов О. М.				ТП-416Б – 122		
Норм-контр.	Сидоренко В.М.						

PHP (англ. *Hypertext Preprocessor*) є однією з найпопулярніших мов програмування для розробки веб-додатків. Нижче наведено кілька ключових аспектів мови PHP:

- Скриптова мова: PHP є мовою скриптів, що означає, що код виконується на сервері, а не в браузері користувача. Це дозволяє створювати динамічні веб-сайти, які можуть взаємодіяти з базами даних, обробляти форми, створювати файли «cookie», сесії та інше.

- Крос-платформенність: PHP підтримується на більшості операційних систем, включаючи Windows, Linux та macOS, що робить її універсальною для розробки на різних платформах.

- Широке застосування: PHP використовується для розробки різноманітних веб-додатків, від невеликих блогів до великих корпоративних порталів та онлайн-магазинів. Приблизно 80 відсотків усіх вебсайтів світу написані на PHP.

- Підтримка взаємодії з базами даних: PHP має вбудовану підтримку для багатьох систем управління базами даних (СУБД), таких як MySQL, PostgreSQL, SQLite та інші, що робить роботу з даними досить зручною.

- Легка для вивчення: синтаксис PHP простий і зрозумілий, особливо для тих, хто вже знайомий з іншими мовами програмування, такими як C, Java або JavaScript.

- Багата функціональність: PHP має широкий набір функцій та бібліотек для роботи з різними аспектами веб-розробки, такими як обробка форм, робота з файлами, робота з URL, робота з мережею, шифрування даних та багато іншого.

Незважаючи на свої переваги, важливо пам'ятати, що безпека є важливим аспектом розробки в PHP, і програмісти повинні дотримуватися найкращих практик безпеки, таких як уникання SQL-ін'єкцій, XSS-атак та інших потенційних загроз.

JavaScript

При виборі мови програмування, що буде відповідати за динамічні аспекти веб-розробки, обробку користувацьких дій та забезпечення чудового користувацького досвіду, було обрано JavaScript. Ця мова, завдяки своїй універсальності та широкому застосуванню у веб-технологіях, стала необхідним інструментом для реалізації інтерактивності, анімацій та асинхронної взаємодії у сучасних веб-додатках.



Рис. 2.2. Логотип мови програмування JavaScript.

JavaScript (аббр. *JS*) - це скриптова мова програмування, яка використовується для реалізації динамічної взаємодії на веб-сторінках. Ця мова має ряд особливостей, які роблять її надзвичайно потужним інструментом для веб-розробки, а саме:

- Легкість при вивченні: JavaScript має простий синтаксис, що робить її досить доступною для початківців. Багато ресурсів та документації доступні для швидкого навчання.

- Швидкість виконання: сучасні рушії JavaScript, такі як V8 в Google Chrome або SpiderMonkey в Firefox, дозволяють виконувати код швидко, що робить JavaScript ефективним для веб-додатків з великою кількістю динамічного контенту.

- Асинхронність: JavaScript підтримує асинхронні запити, що дозволяє реалізовувати складні веб-додатки з асинхронною взаємодією з сервером без блокування веб-сторінки.

- Багатофункціональність: JavaScript може виконувати широкий спектр завдань, включаючи маніпулювання DOM (англ. *Document Object Model*), валідацію форм, роботу з анімаціями, обробку подій і багато іншого.

- Розширюваність: за допомогою бібліотек і фреймворків, таких як jQuery, React, Angular, Vue.js, можна значно розширити можливості JavaScript і спростити розробку складних веб-додатків.

- Кросплатформенність: JavaScript підтримується на різних платформах, включаючи веб-браузери, серверні середовища (наприклад, Node.js) та мобільні додатки, за допомогою фреймворків, таких як React Native або Ionic).

- Співробітництво з іншими технологіями: JavaScript легко інтегрується з іншими технологіями, такими як HTML, CSS, AJAX, що дозволяє створювати повноцінні веб-додатки з багатим функціоналом та ефективною взаємодією з користувачем.

У підсумку, JavaScript є незамінним інструментом для реалізації динамічної та інтерактивної взаємодії користувачів з сервером на веб-сторінках. Завдяки розширеним можливостям та великому співробітництву з іншими технологіями, JavaScript дозволяє створювати складні веб-додатки з великою функціональністю, які забезпечують зручний та ефективний користувацький досвід.

SQL та MySQL

Для того, щоб мати можливість зберігати різну інформацію про користувачів, зображення, які вони завантажують та іншу допоміжну інформацію в базі даних, було вирішено використати **SQL** (англ. *Structured query language*) – мову програмування для взаємодії користувача з базами даних. Вона застосовується для формування запитів, оновлення і керування реляційними базами даних, створення схеми бази даних та її модифікації. SQL також включає CLI (англ. *Call Level Interface*) для доступу і керування базами даних дистанційно.



Рис. 2.3. Логотип мови SQL.

На відміну від дійсних мов програмування, SQL може формувати інтерактивні запити або, будучи вбудованою в прикладні програми, виступати як інструкції для керування даними. Окрім цього, стандарт SQL містить функції для визначення зміни, перевірки та захисту даних.

Сама по собі SQL не є ані системою керування базами даних, ані окремим програмним продуктом, тому в якості системи управління базами даних було обрано MySQL.



Рис. 2.4. Логотип СУБД MySQL.

MySQL - це відкрита система управління базами даних, яка використовується для зберігання та керування великими обсягами даних у веб-додатках та інших системах. Декілька важливих пунктів про MySQL:

- Тип бази даних: MySQL є реляційною СУБД і це означає, що дані зберігаються у вигляді таблиць зі зв'язками між ними. Цей підхід дозволяє ефективно керувати та оптимізувати роботу з даними.

- Підтримка функцій: MySQL має широкий набір функцій для роботи з даними, включаючи різні типи даних (числа, рядки, дати тощо), агрегатні функції (SUM, AVG, COUNT тощо), функції рядкових операцій (CONCAT, SUBSTRING, REPLACE тощо) та інші.

- Масштабованість: MySQL дозволяє масштабувати бази даних з невеликих проєктів до великих систем з великими обсягами даних та великою кількістю користувачів.

- Безпека: MySQL має вбудовані механізми безпеки, такі як рівні доступу (перегляд, редагування, видалення), автентифікація користувачів та захист від SQL-ін'єкцій та інших атак на базу даних.

- Підтримка інструментів: щоб спростити роботу з MySQL, існують різні інструменти, такі як phpMyAdmin (веб-інтерфейс для адміністрування бази даних), MySQL Workbench (графічний інструмент для розробки та адміністрування бази даних) та інші. У цій роботі було вирішено обрати MySQL Workbench.

MySQL є дуже популярною СУБД у веб-розробці через свою швидкодію, надійність та широкі можливості роботи з даними. Вона використовується у багатьох веб-проєктах, від невеликих блогів до великих корпоративних порталів та онлайн-магазинів і звичайно вона дуже добре підходить для створення вебхостингів.

HTML та CSS

HTML – це стандартизована мова розмітки, яка використовується для створення та відображення вебсторінок у браузерях. Браузери отримують HTML документ від сервера, а потім інтерпретують код в інтерфейс користувача.



Рис. 2.5. Логотип мови HTML

Деякі з основних особливостей HTML:

- Структурованість: HTML дозволяє розміщувати контент на сторінці у вигляді різних елементів, таких як заголовки, параграфи, списки, таблиці,

зображення тощо. Це дозволяє створювати структуровані та організовані веб-сторінки.

- Теги: HTML використовує теги для визначення типу контенту та його відображення. Наприклад, `<h1>` для заголовків першого рівня, `<p>` для параграфів, `` для вставки зображень тощо.

- Гіпертекстові посилання: HTML дозволяє створювати посилання за допомогою тегу `<a>` для переходу між різними сторінками та ресурсами в Інтернеті.

- Можливості форм: HTML має вбудовану підтримку для створення форм `<form>`, які дозволяють користувачам взаємодіяти з веб-сторінками, наприклад, відправляти дані на сервер або вводити текстові дані.

- Семантичність: HTML має семантичні теги, які дозволяють вказати значення та призначення кожного елемента на сторінці. Наприклад, `<header>`, `<footer>`, `<nav>`, `<article>` та інші, що полегшує розуміння структури сторінки для пошукових систем та розробників.

- Кросплатформенність: HTML підтримується всіма сучасними веб-браузерами та платформами, що дозволяє створювати веб-сторінки з однаковим відображенням на різних пристроях.

CSS - це мова стилів, яка використовується для візуального оформлення веб-сторінок, написаних мовою HTML.



Рис. 2.6. Логотип мови CSS

Деякі з основних особливостей CSS:

- Розділення змісту та представлення: CSS дозволяє відокремлювати стильове оформлення веб-сторінки від її змісту. Це робить код більш структурованим, зрозумілим та легким для редагування.

- Селектори: CSS використовує селектори для вибору елементів сторінки, до яких будуть застосовуватися стилі. Наприклад, можна застосувати стилі до всіх елементів певного класу або до конкретного ідентифікатора.

- Типи стилів: CSS підтримує різні типи стилів, такі як кольори, шрифти, розміри, відступи, рамки, фони та інші. Це дозволяє створювати різноманітні та привабливі дизайни веб-сторінок.

- Властивості та значення: CSS має велику кількість властивостей та значень, які можна використовувати для налаштування вигляду елементів сторінки. Наприклад, можна змінювати колір тексту, задній фон, величину шрифту, вирівнювання та інші параметри.

- Каскадування: CSS використовує каскадність, що дозволяє встановлювати пріоритети стилів в залежності від їхнього типу (наприклад, властивостей елемента, класу чи ідентифікатора), а також застосовувати спадковість стилів.

- Медіазапити: CSS підтримує медіазапити, що дозволяють налаштовувати вигляд веб-сторінки в залежності від параметрів пристрою, таких як розмір екрану, орієнтація, кольорова палітра тощо. Це дозволяє створювати адаптивний дизайн.

У підсумку, HTML та CSS є основними технологіями для створення веб-сторінок, які забезпечують їхню структурованість, взаємодію з користувачами, підтримку на різних платформах, візуальне оформлення та багато іншого.

2.2. Фреймворк Laravel та основні бібліотеки

На сьогоднішній день існує декілька популярних фреймворків для розробки веб-застосунків на мові PHP і їх використання має кілька переваг:

- Прискорення розробки: фреймворки надають готові інструменти та структуру для швидкого створення веб-додатків без необхідності писати все з нуля.

- Масштабованість: фреймворки дозволяють легко масштабувати додатки з ростом обсягів даних та користувацького трафіку.
- Стандартизація коду: використання фреймворків сприяє стандартизації коду, що робить його більш читабельним, зрозумілим та підтримуваним.
- Безпека: фреймворки мають вбудовані механізми безпеки, що дозволяють захистити додатки від різних видів атак та загроз.

При розробці вебхостингу, було вирішено обрати саме Laravel, який особливо популярний серед розробників через свою простоту використання, багатий функціонал та активну спільноту, яка надає багато ресурсів для підтримки та розвитку веб-додатків.



Рис. 2.7. Логотип фреймворку Laravel

Laravel - це відомий веб-фреймворк для розробки веб-додатків на мові програмування PHP. Детальніше про його особливості та переваги:

- Модульність та зручність: Laravel забезпечує модульну структуру, яка дозволяє розділити код на логічні блоки (модулі) та працювати з ними окремо. Це спрощує розробку, підтримку та масштабування веб-додатків.
- Маршрутизація: у Laravel використовується зручна система маршрутизації, що дозволяє легко визначати URL-шляхи та зв'язки з контролерами для обробки запитів.
- ORM та база даних: Laravel має вбудований ORM, який дозволяє працювати з базою даних у вигляді об'єктів та моделей, що робить роботу з даними більш зручною та ефективною.
- Шаблонізація: фреймворк надає потужні інструменти для роботи з шаблонами та виглядами (англ. *view*), що дозволяє швидко та зручно створювати веб-сторінки та динамічний контент.

- Безпека: Laravel має вбудовані механізми безпеки, такі як захист від XSS-атак, SQL-ін'єкцій, CSRF-атак та інших загроз безпеці.

- Спільнота та документація: фреймворк має велику та активну спільноту розробників, яка постійно вносить удосконалення та підтримує додаткові пакети та розширення. Крім того, Laravel має добре організовану та зрозумілу документацію, що дозволяє швидко засвоювати фреймворк і розробляти на ньому.

Spatie/Laravel-permission, Intervention Image та Axios

При створенні власного сервісу фото вебхостингу, було вирішено використати такі корисні бібліотеки, як: Spatie/Laravel-Permission, Intervention Image та Axios - це популярні бібліотеки, які додають додатковий функціонал та можливості для розробки веб-додатків.

Spatie/Laravel-Permission - ця бібліотека надає простий та зручний спосіб управління ролями та дозволами (англ. *role* та *permission*) користувачів у веб-додатках (наприклад Admin, User, Guest). Вона дозволяє легко визначати ролі для користувачів та обмежувати їх доступ до певних ресурсів за допомогою дозволів.

Intervention Image - ця бібліотека надає потужні методи для обробки та редагування зображень при розробці веб-додатків. Вона дозволяє легко змінювати розмір, формат, покращувати якість та виконувати інші операції з зображеннями безпосередньо у коді додатка.

Axios - це популярна бібліотека для виконання HTTP-запитів у JavaScript, яка працює як у веб-браузерах, так і у середовищах виконання JavaScript. Axios використовує обіцянки (англ. *promises*), що дозволяє зручно обробляти асинхронні запити та отримувати результати в зручному форматі.

2.3. Docker та Docker Compose, Git та GitHub, а також PhpStorm

Для того, щоб прискорити і спростити розробку вебхостингу зображень, було вирішено використати декілька допоміжних сервісів, а саме: Docker, Docker Compose, Git, GitHub, а також IDE PhpStorm.

Docker та Docker Compose

Docker - це інструмент для контейнеризації додатків, що забезпечує стандартизацію та ізоляцію середовищ розробки. Контейнеризація полягає в тому, що додаток та всі його залежності (бібліотеки, конфігураційні файли тощо) упаковуються в контейнер, який можна легко переносити між різними середовищами, такими як розробка, тестування та випуск у «production».

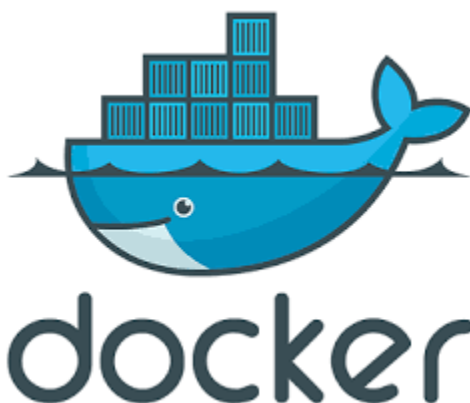


Рис. 2.8. Логотип сервісу Docker.

Основні переваги використання Docker:

- Стандартизація середовищ: Docker дозволяє стандартизувати середовище розробки та робочі середовища, що полегшує управління додатками та їх залежностями.
- Ізоляція: контейнеризація дозволяє ізолювати додатки та їх залежності, що забезпечує безпеку та надійність роботи системи.
- Легкість установки та використання: Docker має зручний інтерфейс та легкий процес установки, що дозволяє швидко почати працювати з ним.
- Масштабованість: Docker дозволяє легко масштабувати додатки за необхідності, додавши або видаливши контейнери.
- Портативність: контейнери Docker можна без перешкод переносити між різними середовищами, такими як різні операційні системи або хмарні середовища.

Docker Compose - це інструмент, який дозволяє описувати та керувати багатоконтейнерними додатками у Docker. Завдяки Docker Compose можна описати всю конфігурацію додатка у файлі YAML, включаючи контейнери,

мережі, змінні середовища тощо, і легко запуснути всі ці контейнери однією командою. Це дозволяє спростити розгортання та управління багатоконтейнерними додатками, зменшити кількість помилок та покращити продуктивність розробників.

Git та GitHub



Рис. 2.9. Логотип сервісу Git.

Git – це розподілена система керування версіями, яка використовується для відстеження змін у файлах та спільної роботи над проектами з командою. Вона дозволяє розробникам зберігати історію змін коду, відстежувати різні версії проекту, вносити зміни, відкатувати непотрібні зміни та об'єднувати зміни з різних гілок. Основні концепції Git:

- Репозиторій (англ. *repository*): це місце, де зберігається весь код та історія змін. Репозиторії можуть бути локальними (на вашому комп'ютері) або віддаленими (на веб-сервері, такому як GitHub).

- Коміт (англ. *commit*): коміт в Git - це фіксація змін, яку ви робите в своєму коді. Кожен коміт має повідомлення, яке пояснює, що ви змінили.

- Гілка (англ. *branch*): гілка - це версія коду, яка розділяється від головної гілки (зазвичай відомої як `main`). Ви можете створювати нові гілки для розробки нового функціоналу, виправлення помилок тощо, які не будуть впливати на основний код.

- Злиття (англ. *merge*): злиття - це процес об'єднання змін з однієї гілки в іншу. Це дозволяє вам вносити зміни у ваш основний код після завершення розробки функціоналу у відокремленій гілці.

GitHub - це веб-платформа для спільної роботи над проектами, яка базується на Git. Вона надає хмарне сховище для ваших репозиторіїв Git, спрощуючи спільну роботу з командою, відстеження змін, управління завданнями та багато іншого.

PhpStorm

Для розробки власного фото вебхостингу було вирішено використовувати IDE PhpStorm як основний інструмент розробки.



Рис. 2.10. Логотип програми PhpStorm

PhpStorm – це комерційне крос-платформове інтегроване середовище розробки для мови програмування PHP, яке надає широкий набір функцій та інструментів для зручного та ефективного програмування веб-додатків.

PhpStorm являє собою інтелектуальний редактор для PHP, HTML і JavaScript з можливостями аналізу коду, автоматизованими засобами рефакторинга, автодоповненням коду, має вбудовані інструменти для роботи з базами даних та підтримку Git. Всі ці фактори роблять PhpStorm ідеальним вибором для розробки вебхостингу зображень та інших веб-проектів.

ВИСНОВКИ ДО РОЗДІЛУ 2

Провівши дослідження всіх необхідних елементів для створення вебхостингу зображень, були обрані інструменти, які найкраще підходять для реалізації та досягнення поставленої задачі. В ході роботи були обрані мови програмування, бібліотеки та інші сервіси і технології, а саме:

- мова програмування PHP, фреймворк для зручної розробки на її основі – Laravel та інтегроване середовище розробки PhpStorm;
- мова програмування SQL та СУБД MySQL для реалізації баз даних;
- бібліотеки Spatie/Laravel-permission, Intervention Image та Axios для створення додаткового корисного функціоналу;
- технології Docker, Docker Compose, Git та GitHub для покращення і прискорення розробки вебхостингу.

Отже, реалізація власного фото вебхостингу, використовуючи та працюючи з найактуальнішими технологіями і сервісами, дозволить отримати корисний досвід, який може знадобитись в майбутньому для створення інших потужних проектів.

РОЗДІЛ 3

РОЗРОБКА САЙТУ ВЕБХОСТИНГА ЗОБРАЖЕНЬ З КОЛЕКТИВНИМ ДОСТУПОМ

3.1. Загальна структура вебхостингу

Для початку потрібно розглянути архітектуру папок і файлів веб-додатку (див. рис. 3.1. і рис. 3.2.). При ініціалізації нового проекту на Laravel, фреймворк автоматично, “з коробки”, створює велику частину папок і файлів, таким чином працювати з цим проектом можна відразу без необхідності налаштовувати додаткові параметри. Тому розберемо тільки ту частину структури, яка була створена в процесі розробки.

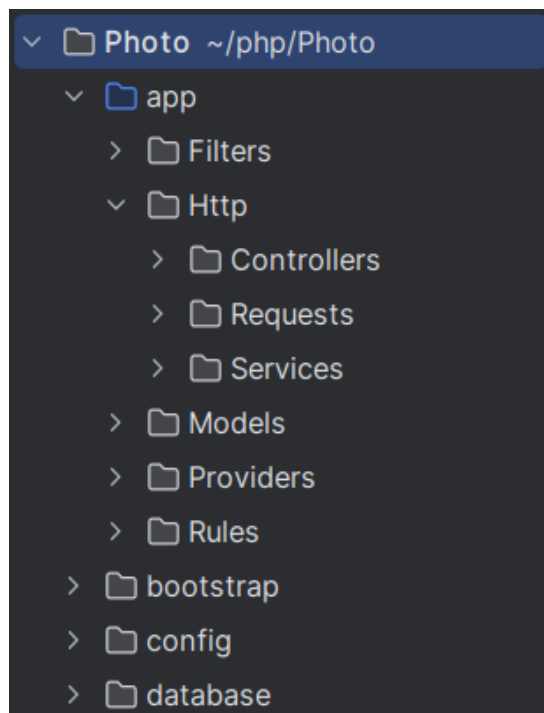


Рис 3.1. Перша частина файлової структури проекту

Кафедра КІТ				НАУ 24 01 78 000 ПЗ			
	ПІБ			РОЗРОБКА САЙТУ ВЕБХОСТИНГУ ЗОБРАЖЕНЬ	Літ.	Аркуш	Аркушів
Виконав	Битько А.В.					36	49
Керівник	Зудов О. М.				ТП-416Б – 122		
Норм-контр.	Сидоренко В.М.						

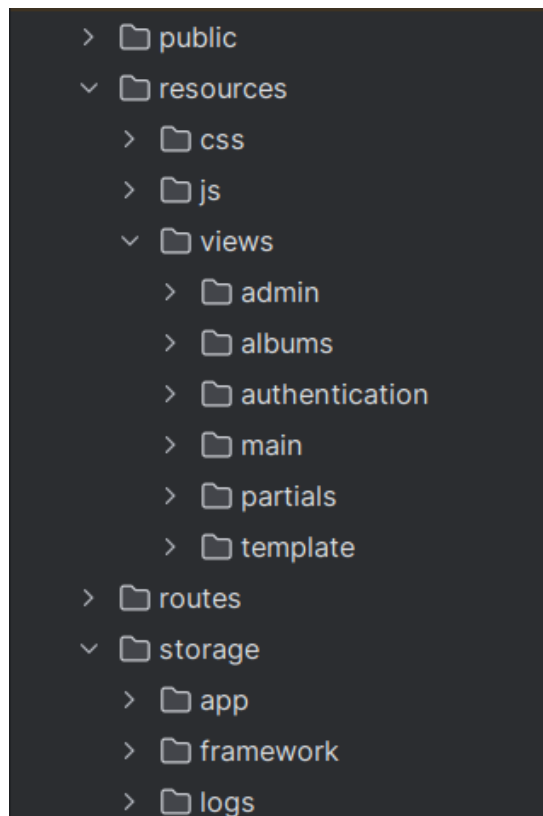


Рис. 3.2. Друга частина файлової структури проекту

Папка **“app”** у Laravel відповідає за більшу частину програми. Це основна директорія, де розміщуються головні компоненти додатку:

- Папка **“Filters”** відповідає за логіку, пов’язану з фільтрацією і сортуванням зображень, що завантажили користувачі вебхостингу. Приклад інтерфейсу панелі сортування зображено нижче (див. рис. 3.3.).

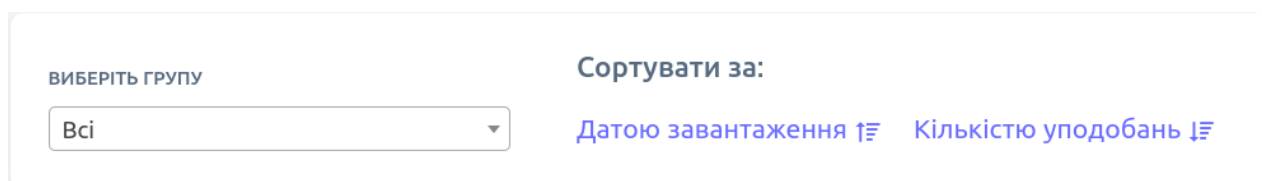


Рис. 3.3. Панель сортування і фільтрів

- Папка **“Http”** містить: контролери (папка **“Controllers”**), які обробляють HTTP-запити; запити (папка **“Requests”**), які виконують валідацію HTTP-запитів та сервіси (папка **“Services”**), що виконують різну бізнес-логіку додатку.

- Папка **“Models”** містить моделі Eloquent (ORM), які представляють таблиці бази даних і взаємодіють з ними (див. рис. 3.4.).

```

12 class Photo extends Model
13 {
14     use HasFactory, Filterable;
15
16     const PHOTO_PATH = 'public/images/';
17     const PHOTOS_PER_PAGE = 6;
18
19     protected $fillable = [
20         'user_id',
21         'album_id',
22         'name',
23         'size',
24         'original_name',
25         'is_public',
26     ];
27
28     protected $casts = [
29         'is_public' => 'boolean',
30     ];

```

Рис. 3.4. Приклад моделі “Photo”

- Папки **“Providers”** та **“Rules”** відповідають за провайдери служб і спеціальні правила валідації, відповідно.

Далі розташовані папки **“bootstrap”** і **“config”**. Папка **“bootstrap”** відповідає за ініціалізацію додатку та його компонентів на ранніх етапах завантаження, в той час як папка **“config”** відповідає за зберігання конфігураційних файлів.

Одна з найголовніших папок додатку — **“database”**. Вона містить міграції (англ. *migrations*), які використовуються для створення та зміни структури бази даних. Приклад одного з таких файлів наведено нижче (див. рис. 3.5.).

```

7   return new class extends Migration
8   {
9       /**
10      * Run the migrations.
11      */
12      public function up(): void
13      {
14          Schema::create(table: 'photos', function (Blueprint $table) {
15              $table->id();
16              $table->unsignedBigInteger(column: 'user_id')->nullable();
17              $table->foreign(columns: 'user_id')->references(columns: 'id')->on(table: 'users');
18              $table->string(column: 'name');
19              $table->string(column: 'original_name');
20              $table->unsignedBigInteger(column: 'size');
21              $table->boolean(column: 'is_public')->default(value: false);
22              $table->timestamps();
23          });
24      }

```

Рис. 3.5. Приклад міграції для створення таблиці бази даних “photos”

Папка **“public”** містить всі ресурси, які доступні користувачам через HTTP. Це можуть бути зображення, JavaScript файли, CSS стилі, медіа-файли та інші ресурси, які можуть бути викликані безпосередньо з браузера.

Папка **“resources”** містить ресурси, такі як шаблони “Blade” (папка “views”), JavaScript та CSS файли, які використовуються для розробки і відображення “фронтенд” частини додатку. Приклад одного з шаблонів наведено нижче (див. рис. 3.6.).

```

1 @foreach($photos as $photo)
2     <div class="col-md-6 col-lg-4 mb-3 image">
3         <div class="card h-100">
4             <div class="card-header pt-3 pb-1">
5                 <div class="d-flex justify-content-between">
6                     <h5 class="card-title">{{ $photo->name }}</h5>
7                     @if(auth()->id() === $photo->user_id)
8                         <form action="{{ route('photo.destroy', $photo->id) }}" method="post">
9                             @csrf
10                            @method('DELETE')
11                            <i class="bx bx-trash delete-icon font-size-25" title="Видалити"></i>
12                        </form>
13                    @endif
14                </div>
15            </div>
16
17            <a class="image-link" img-title="{{ $photo->name }}"
18                href="{{ \Illuminate\Support\Facades\Storage::url(\App\Models\Photo::PHOTO_PATH) . $photo->user_id .
19                <div class="img-wrapper">
20                    
23                </div>
24            </a>
25

```

Рис. 3.6. Частина коду шаблону для відображення фотографії

Папка **“routes”** відповідає за визначення маршрутів, які вказують фреймворку, як обробляти вхідні HTTP-запити.

Остання папка **“storage”** відповідає за зберігання різноманітних даних, які використовуються додатком. Основна частина цих даних — зображення, які можуть завантажити користувачі у фото вебхостинг.

3.2. Архітектура бази даних

Під час розробки вебхостингу було використано СУБД MySQL, яка чудово підходить для зберігання та керування даними про користувачів, групи, зображення, альбоми та інше. Були створені основні таблиці бази даних, а саме: **“users”**, **“roles”**, **“model_has_roles”**, **“photos”**, **“albums”**, **“groups”** і **“likes”**. Розглянемо детальніше кожен з них.

Таблиця **“users”** відповідає за збереження основної інформації про користувачів вебхостингу (див. рис. 3.7.). При створенні таблиці, були вказані такі поля, як:

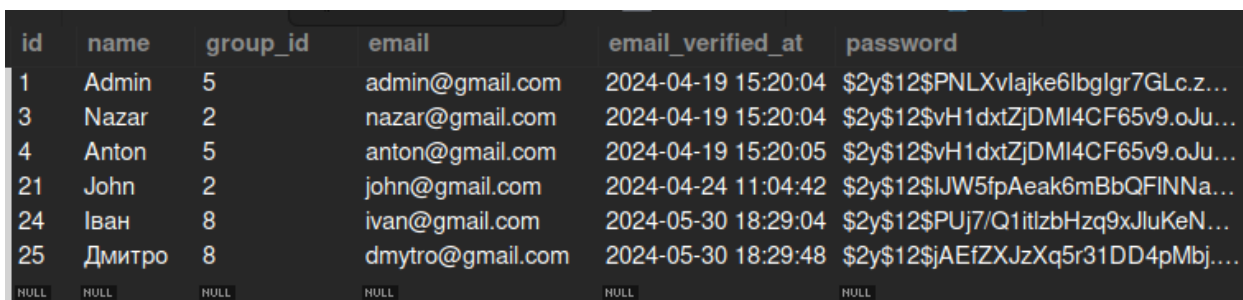
- **“id”**, що являє собою унікальний ідентифікатор кожного користувача;
- **“name”**, для збереження імен користувачів;
- **“email”**, для збереження електронної пошти кожного користувача.

Email повинен бути унікальним;

- **“password”**, для збереження паролю до аккаунта кожного користувача.

В базі даних паролі відображаються у вигляді хеш-значення;

- **“group_id”**, що являє собою поле, яке посилається на поле **“id”** в таблиці **“groups”**. Таким чином утворюється зв'язок між двома таблицями **“users”** та **“groups”**.



id	name	group_id	email	email_verified_at	password
1	Admin	5	admin@gmail.com	2024-04-19 15:20:04	\$2y\$12\$PNLXvIajke6lbgIgr7GLc.z...
3	Nazar	2	nazar@gmail.com	2024-04-19 15:20:04	\$2y\$12\$vH1dxtZJDMI4CF65v9.oJu...
4	Anton	5	anton@gmail.com	2024-04-19 15:20:05	\$2y\$12\$vH1dxtZJDMI4CF65v9.oJu...
21	John	2	john@gmail.com	2024-04-24 11:04:42	\$2y\$12\$IJW5fpAeak6mBbQFINNa...
24	Іван	8	ivan@gmail.com	2024-05-30 18:29:04	\$2y\$12\$PUj7/Q1itIzbHzq9xJluKeN...
25	Дмитро	8	dmytro@gmail.com	2024-05-30 18:29:48	\$2y\$12\$jAEfZXJzXq5r31DD4pMbj....
NULL	NULL	NULL	NULL	NULL	NULL

Рис. 3.7. Вигляд таблиці **“users”** в MySQL Workbench

Таблиці **“roles”** і **“model_has_roles”** відповідають за визначення прав доступу кожного користувача. Наразі існують три ролі: **“admin”**, **“user”** і **“guest”**.

Звичайний **“юзер”** може переглядати публічні фото та ставити їм **“лайк”**, завантажувати і видаляти власні, робити свої фото публічними або приватними, створювати і видаляти альбоми та завантажувати в них фото. До того ж, якщо **“юзер”** знаходиться в групі, він може створювати і видаляти групові альбоми або особисті альбоми в групових, а також завантажувати і видаляти власні фото, завантажені в ці альбоми.

Користувач з правами **“адмін”**, може робити все те саме, що і **“юзер”**, а також створювати і видаляти інших користувачів та групи.

Користувачі, які не авторизувались в додатку, мають статус **“гість”**. Вони можуть лише переглядати публічні зображення.

Таблиці **“photos”** (див. рис. 3.8.) і **“albums”** відповідають за збереження різної інформації про фото та альбоми користувачів. Вони мають такі поля, як:

- **“id”**;
- **“user_id”**, для зв'язку з таблицею **“users”**;
- **“album_id”**, для зв'язку фото з альбомами та **“group_id”**, для зв'язку альбомів з групами;
- **“name”**;
- **“original_name”** — це поле таблиці **“photos”**, яке використовується для зберігання **“рандомно”** згенерованих назв зображень. Це потрібно тому, що зображення, завантажуванні користувачами, можуть мати однакові назви, а для коректної роботи з фото, потрібні унікальні назви;
- **“parent_id”** — це поле таблиці **“albums”**, яке описує самодостатній зв'язок (англ. *self-referencing*). Таким чином таблиця може посилатись сама на себе, що означає, що кожен запис цієї таблиці може мати зв'язок з іншим записом у тій же таблиці. Поле використовується для створення альбомів в альбомі;
- **“size”**, для збереження розміру фото у байтах;

- “is_public”, для визначення “публічності” зображень.

id	user_id	album_id	name	original_name	size	is_public
85	1	NULL	ultrawide.jpeg	hdcM5t2iB5.jpeg	21411	0
87	3	NULL	Karadokya'...	c4aY1jSllk.jpg	2585574	1
88	3	NULL	Karadokya'...	cgQin8Xcv7.jpg	2585574	1
91	3	7	Karadokya'...	3yAYelsdrT.jpg	2585574	1
93	3	7	Karadokya'...	eqyG1cDbEj.jpg	2585574	1
94	3	7	Karadokya'...	OMnfRmJgEc.jpg	2585574	0
96	1	NULL	cat_1.jpg	Ou5kGQn9J.jpg	335045	1
97	1	5	wolves_2.jpeg	1h6k5R1EhP.jpeg	3091291	0
98	1	16	ds3.jpg	6dfr5ePrFE.jpg	1954322	1
99	1	16	ds3.jpg	o45Nqr5lpb.jpg	1954322	0
100	1	16	cat_1.jpg	ZbAKzkZodc.jpg	335045	1

Рис. 3.8. Вигляд таблиці “photos”

Таблиці “groups” і “likes”. Таблиця “groups” використовується для зберігання назв та “id” кожної групи. Таблиця “likes” (див. рис. 3.9.) використовується як проміжна таблиця між користувачами і зображеннями для зберігання “лайків”. Вона представляє собою таблицю з трьома полями: “id”, “user_id” і “photo_id”. Кожен запис в таблиці “users” може мати багато відповідних записів в таблиці “photos”, і навпаки. Такий тип зв’язку називається “Many-to-many” (багато до багатьох).

id	user_id	photo_id
14	4	98
18	1	98
20	4	68
21	4	69
41	4	88
42	1	102
43	1	69
58	1	104
NULL	NULL	NULL

Рис. 3.9. Вигляд таблиці “likes”

Таким чином, правильне визначення зв’язків між таблицями дозволяє ефективно організувати та керувати даними, встановлювати правила цілісності та забезпечувати належну структуру бази даних.

3.3. Інтерфейс робочого додатку

Перед тим, як мати змогу користуватись вебхостингом повноцінно, потрібно в ньому авторизуватись. Приклад сторінки авторизації зображено нижче (див. рис. 3.10.).

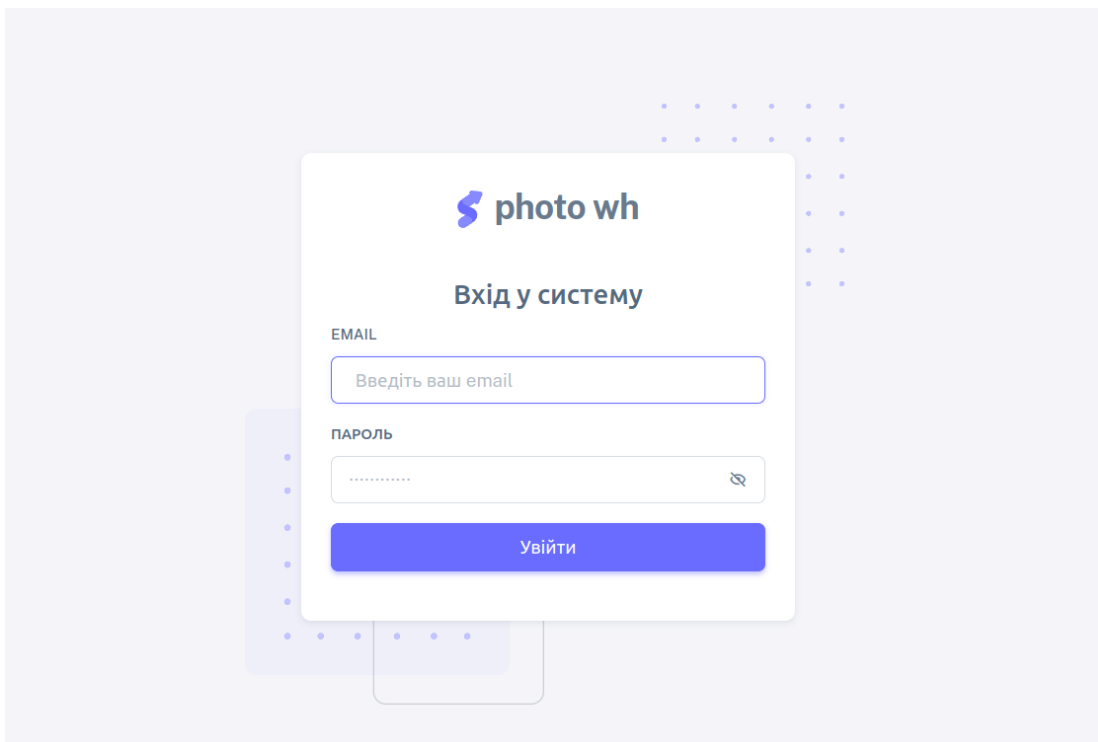


Рис. 3.10. Головне меню авторизації

Після того, як користувач увійшов до системи (як “user”), додаток відкриє головну сторінку сайту (див. рис. 3.11.).

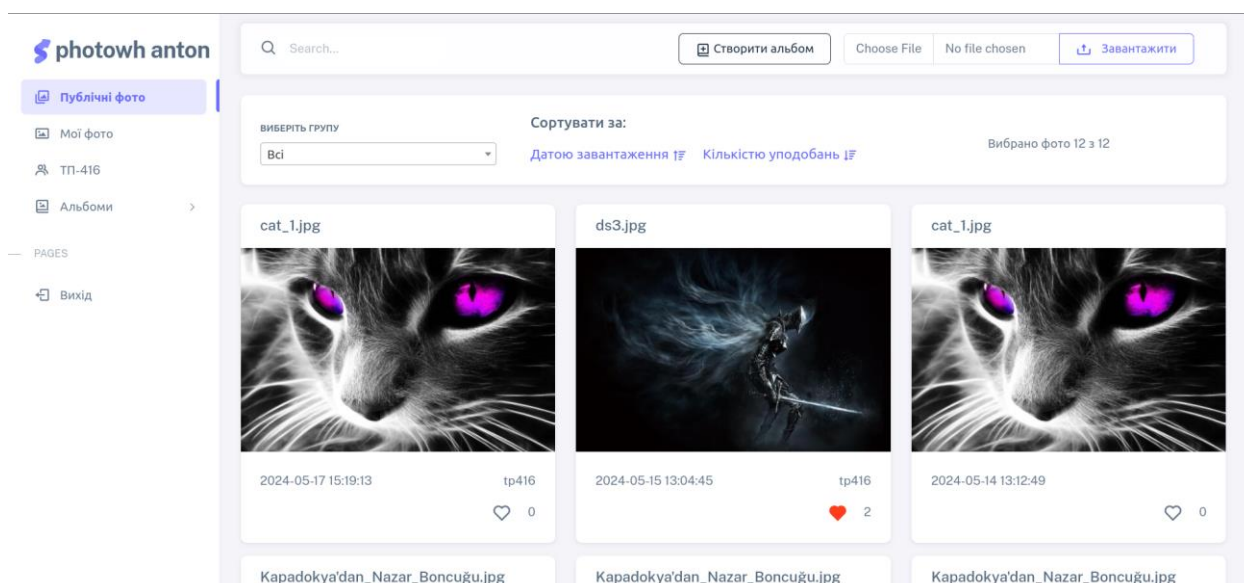


Рис. 3.11. Основна сторінка фото вебхостингу

На головній сторінці за замовчуванням відкривається пункт меню **“Публічні фото”**, в якому відображаються всі зображення, які користувачі вирішили зробити публічними для інших користувачів.

Зображення зроблені у вигляді карток, всередині яких відображається інформація про кожне зображення (назва, дата завантаження, кількість «лайків» і альбом), а також мініатюра самого фото. Для кожного зображення реалізована можливість перегляду його повно розмірної версії (при натисканні на мініатюру) та можливість поставити «лайк».

Також, на головній сторінці знаходиться панель фільтрів (див. рис. 3.3.) і декілька кнопок, а саме **“Створити альбом”** (особистий) та завантажити зображення.

На боковій панелі зліва знаходяться декілька пунктів меню: **“Мої фото”**, група користувача (якщо вона є), **“Альбоми”** та вихід із системи.

На вкладці **“Мої фото”** відображаються усі зображення, завантажені користувачем. В меню групи, **“ТП-416”**, відображаються усі групові альбоми, а також додаткова кнопка **“Створити груповий альбом”** (див. рис. 3.12.).

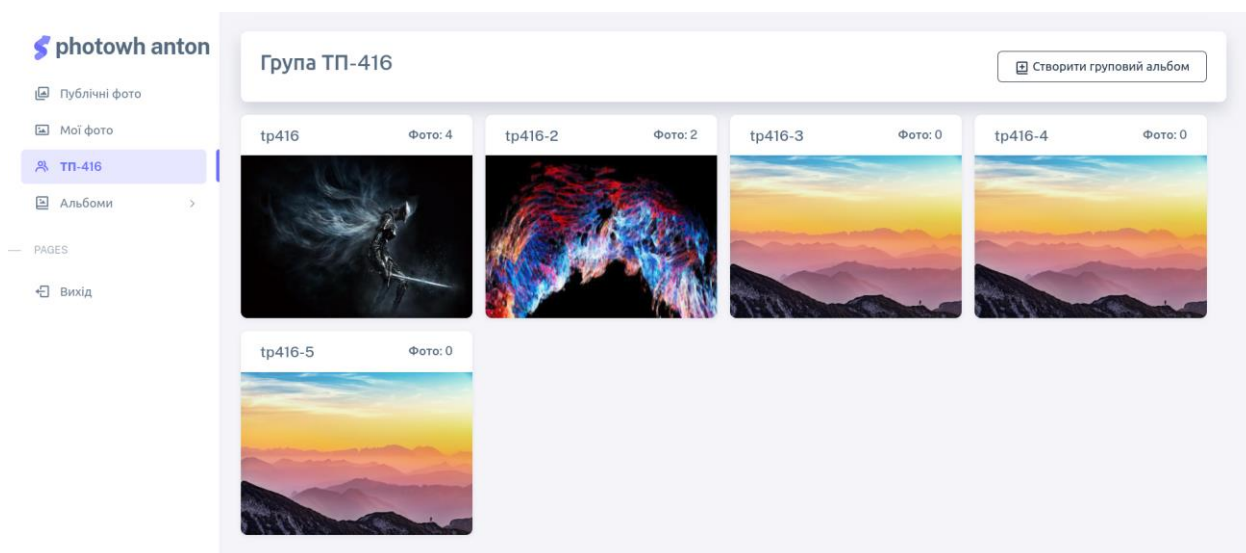


Рис. 3.12. Пункт меню з альбомами групи

Якщо натиснути на будь-який груповий альбом, то система відкриє сторінку цього альбому (див. рис. 3.13.).

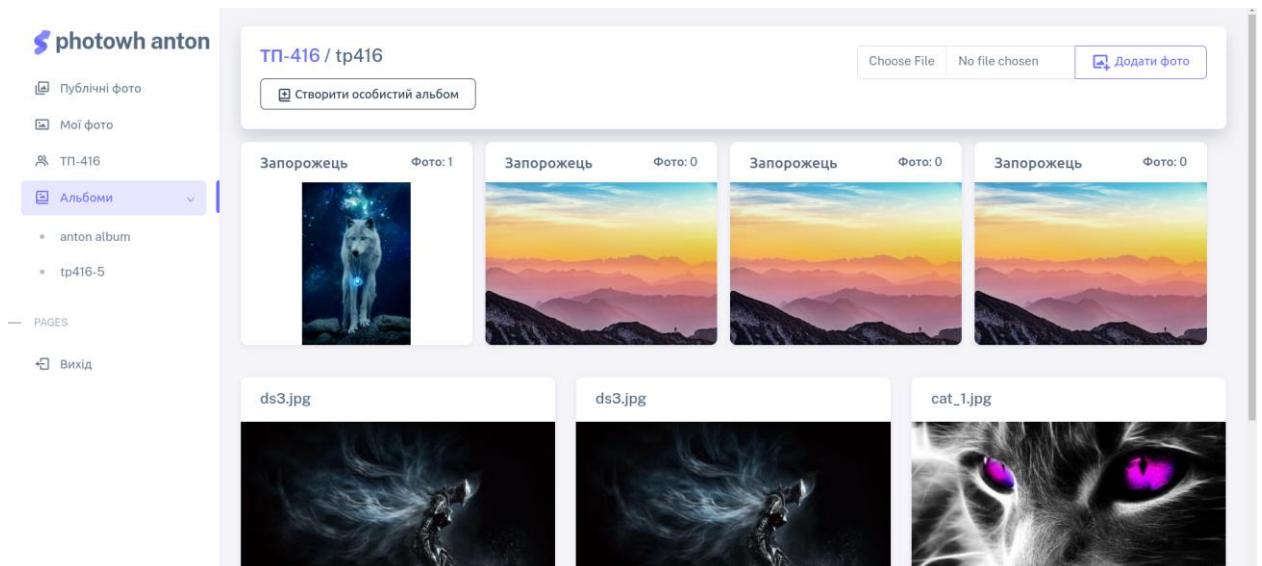


Рис. 3.13. Вигляд групового альбому

Як видно з рисунка 3.13, у груповому альбомі знаходяться ще чотири альбоми користувача (з назвою “Запорожець”) і деякі фото цього альбому. Також у верхньому меню з’явилась кнопка “Створити особистий альбом” (всередині альбому групи), поле завантаження фотографій і візуальна підказка, що показує місце розташування користувача у вигляді “Група / альбом групи / альбом користувача (опціонально)”.

В пункті бокового меню “Альбоми” відображаються усі альбоми, створені авторизованим в даний момент користувачем.

Якщо увійти до системи в якості «адміна», то на боковому меню з’являться два додаткових пункти “Список користувачів” (див. рис. 3.14.) і “Список груп” (див. рис. 3.15.).

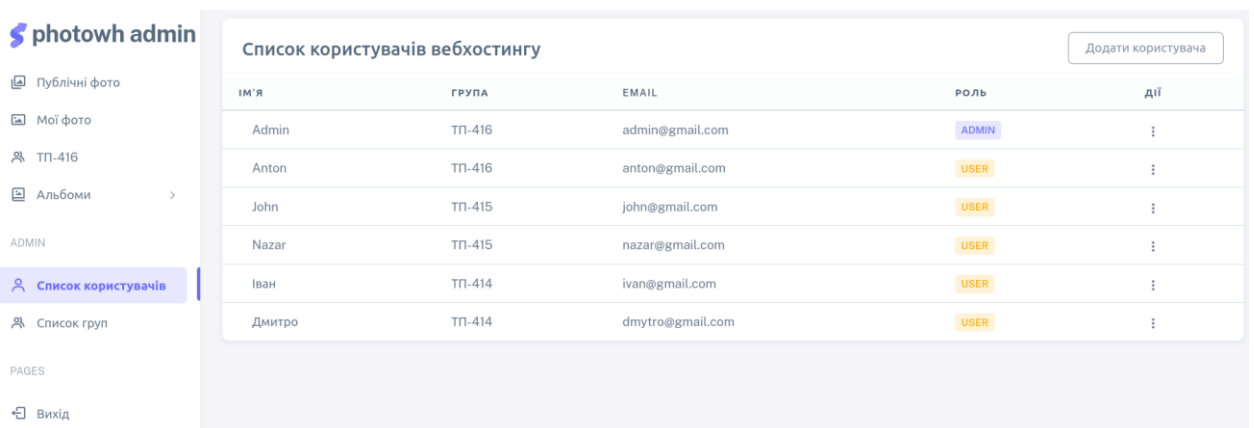


Рис. 3.14. Пункт меню з списком користувачів додатку

В пункті з списком користувачів відображається уся інформація про кожного авторизованого користувача (крім паролів): ім'я, група, до якої він належить, електронна пошта та права доступу у вигляді ролей. Також в цьому меню можна додавати нових, редагувати і видаляти існуючих користувачів.

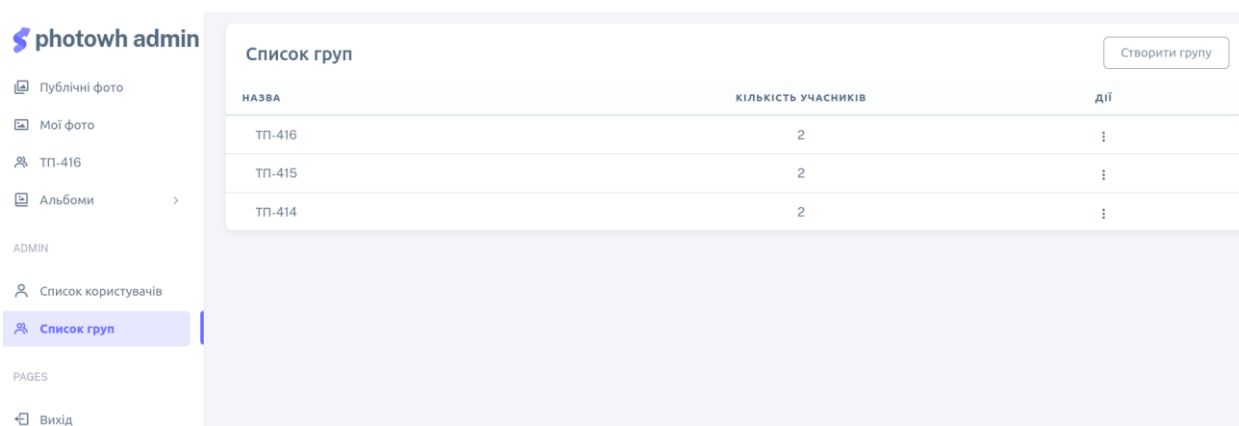


Рис. 3.15. Пункт меню з список груп додатку

В пункті з списком груп відображається назва кожної групи та кількість учасників. З групами можна проводити всі ті ж операції, що і з користувачами, за виключенням видалення групи, якщо в ній знаходиться хоча б один учасник. При спробі видалення не пустої групи, система виведе на екран повідомлення про помилку і сама група видалена не буде.

ВИСНОВКИ ДО РОЗДІЛУ 3

В цьому розділі було детально розглянуто ключові аспекти створення та функціонування веб-додатку для зберігання та обміну зображеннями з можливістю колективного доступу.

Було розроблено загальну структуру додатку, яка включає різні папки для контролерів, моделей, шаблонів, роутів, міграцій, зображень тощо. Ця структура дозволяє ефективно організувати код і підтримувати його в майбутньому.

Було описано архітектуру бази даних з використанням реляційних таблиць для зберігання даних про користувачів, зображення, альбоми, відносини між ними тощо. Використано міграції для автоматизації процесу створення таблиць і змін в базі даних, що спрощує розгортання додатку і управління базою даних.

Було представлено приклади скріншотів робочого інтерфейсу додатку, де показані різні сторінки для перегляду зображень, альбомів, управління користувачами та групами, налаштування доступу до зображень тощо.

Отже, у третьому розділі відображено загальну структуру веб-додатку та усіх його складових частин.

ВИСНОВКИ

У даній кваліфікаційній роботі було детально розглянуто та досліджено усі важливі аспекти, що стосуються теми «Вебхостинг зображень з колективним доступом», на основі яких було створено основні частини цієї теми:

- В першому розділі описано поняття хостингу, вебхостингу та існуючі види вебхостингів за технічними можливостями, а також за типом вмісту, який вони зберігають. До того ж, були розглянуті актуальні вебхостинги зображень, їх особливості, а також переваги та недоліки.

- В другому розділі було розглянуто та обґрунтовано вибір усіх інструментів і технологій, які були використані при розробці додатку.

- В третьому розділі представлена найважливіша структура веб-додатку, була описана архітектура бази даних з найголовнішими таблицями. Крім цього, були наведені приклади робочого інтерфейсу вебхостингу.

Усі ці аспекти разом створюють комплексний та функціональний веб-додаток для зберігання та обміну зображеннями з можливістю керування доступом, що відповідає поставленим цілям та вимогам кваліфікаційної роботи.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Локхарт Дж. Modern PHP: New Features and Good Practices. - Львів: Видавництво "Львівський національний університет", 2019. – С. 125 - 134.
2. Кейс А., Виджек М. PHP 7 Programming Cookbook. - Київ: Видавництво "Кодекс", 2019. – С. 69 - 82.
3. "Git". [Електронний ресурс] URL: <https://uk.wikipedia.org/wiki/Git>. (Дата звернення: 31.05.2024).
4. Сміт, Дж. Laravel: Up & Running. - Київ: Видавництво "Кодекс", 2019. – С. 127 - 136.
5. "PhpStorm". [Електронний ресурс] URL: <https://uk.wikipedia.org/wiki/PhpStorm>. (Дата звернення: 31.05.2024).
6. "Laravel Documentation". [Електронний ресурс] URL: <https://laravel.com/docs/11.x/readme>. (Дата звернення: 31.05.2024).
7. "Починаючи | Axios Docs". [Електронний ресурс] URL: <https://axios-http.com/uk/docs/intro>. (Дата звернення: 31.05.2024).
8. "Intervention Image | Intervention Image v3 | intervention.io". [Електронний ресурс] URL: <https://image.intervention.io/v3>. (Дата звернення: 31.05.2024).
9. "Introduction | laravel-permission". [Електронний ресурс] URL: <https://spatie.be/docs/laravel-permission/v6/introduction>. (Дата звернення: 31.05.2024).
10. "HTML – MDN Web Docs - Mozilla". [Електронний ресурс] URL: <https://developer.mozilla.org/ru/docs/Web/HTML>. (Дата звернення: 31.05.2024).
11. Williams, A., & Johnson, M. The Role of User Experience Design in Website Conversion Optimization. International Journal of Digital Marketing, 2020 - С. 29 - 42.
12. Положення про кваліфікаційні роботи (проекти) здобувачів вищої освіти Національного авіаційного університету: затверджено Вченою радою Національного авіаційного університету.