**ЗМІСТ**

Перелік умовних позначень 2

Вступ 3

1 ТЕОРЕТИЧНА ЧАСТИНА 5

1.1 JavaScript 5

1.2 HTML 6

1.3 CSS 7

1.4 РНР 7

1.5 Symfony 9

2 ПРАКТИЧНА ЧАСТИНА 11

2.1 Структура веб-ресурсу 11

2.2 Етапи розробки проекту 15

2.3 Огляд аналогів 16

Висновки 19

Список використаної літератури 21

Додаток 1 Лістинг програми

**Список умовних скорочень.**

JavaScript – назва реалізації стандарту мови програмування ECMAScript компанії Netscape, базується на принципах прототипного програмування.

jQuery – популярна JavaScript-бібліотека з відкритим сирцевим кодом.

HTML (HyperText Markup Language) – стандартна мова розмітки веб-сторінок в Інтернеті.

CSS (Cascading Style Sheets) – спеціальна мова, що використовується для відображення сторінок, написаних мовами розмітки даних.

PHP (Hypertext Preprocessor) – скриптова мова програмування, була створена для генерації HTML-сторінок на стороні веб-сервера.

**Вступ**

Слід розпочати з того, що сучасний користувач всесвітньої мережі Internet є надзвичайно вибагливий, а оскільки майже кожен розробник будь-якого веб-ресурсу бажає сповна догодити цьому користувачу, все більше і більше сайтів стають інтерактивним, краще графічно оформленими та зручнішими у використання. Для того, щоб на розроблюваному сайті все більше і більше елементів могли напряму взаємодіяти з гостем, часто застосовуються можливості CSS та JavaScript, що дозволяють більш «оживити» сторінку, додати анімацію на неї, зробити її динамічною.

За допомогою розробленого додатку можна завантажувати, редагувати, переглядати та сортувати зображення. Користувач може створювати власні бібліотеки та редагувати їх вміст. Великою перевагою розробленого веб-ресурсу є те, що він має легкий та зрозумілий інтерфейс, не потребує великих затрат часу на освоєння та може ви користуватися будь-ким незалежно від рівня володіння комп’ютером та знаннями в сфері роботи з подібними додатками.

Для вирішення проблеми забезпечення швидкого пошуку здійснюється обробка бази даних, яка включає в себе побудову індексного структури на базі аксіоми нерівності трикутника метричного простору, мета якої полягає в зменшенні витрат часу пошуку в порівнянні з повним перебором. Ефективність індексної структури оцінюється за такими параметрами, як кількість звернень до пам'яті, витрати на пошук по індексного структурі і кількість операцій обчислення відстані між поданням запиту і об'єктами бази даних. Очевидно, що в разі пошуку зображень має місце порівняння описів високої розмірності. Крім цього, витрати на порівняння об'єктів зростають в разі використання складних метрик, спрямованих на порівняння високорівневих концепцій. Як приклад слід привести отримали значне поширення метрик, що дозволяють порівнювати результати сегментації. Безсумнівно, прагнення до усунення семантичного конфлікту призводить до все більш і більш широкого використання методів пошуку відеоданих по областям, а значить, до більш складним метрик порівняння об'єктів

Розроблюваний проект являє собою повноцінне середовище, де знаходяться різного роду зображення, наповнене різноманітними функціями та підтримуючи при цьому можливість доповнення згадуваного функціоналу та додання нових рівнів та складових.

Проте, не беручи до уваги той факт, що використовуючи можливості PHP та фреймворку Symfony можна створити практично будь-яку базу із зображеннями, організувати пошук в ній, але задля кращого зовнішнього вигляду, покращенню структури та функціоналу застосовується ряд сторонніх інструментів, таких як HTML, CSS

**РОЗДІЛ 1. ТЕОРЕТИЧНА ЧАСТИНА**

**1.1 JavaScript**

JavaScript - це порівняно нова мова для написання сценаріїв, розроблений компанією Netscape. За допомогою мови JavaScript ми можемо створювати інтерактивні web-сторінки найбільш зручним і ефективним способом. У цьому практичному посібнику наведені приклади, які, на думку автора, є найбільш важливими. Вони зможуть продемонструвати можливості JavaSript і принципи організації мови. У цьому посібнику наводяться приклади того, що можна зробити, використовуючи JavaScript, а також, що не менш важливо, розповідається, як це зробити.

JavaScript - це не те ж саме, що Java. Багато людей вважають, що мова JavaScript - це те ж саме, що мова Java, недарма вони носять однакові імена. Однак це невірно. Не будемо розбиратися в існуючих відмінностях, важливо лише пам'ятати, що JavaScript та Java - це різні мови, хоча в них є багато спільного.

Найчастіше мова JScript використовується для обробки подій на веб-сторінці таких як натиснення на кнопку відправки форми, вибір нового елементу зі списку, позиціонування курсору мишки над якимось елементом, завершення завантаження сторінки тощо. Призначення такого обробника подій - попередити користувача про можливу помилку в його діях чи створити більш зручну обстановку для роботи зі сторінкою. Проте іноді застосування мови скриптів може бути й більш прозаїчним : в момент завантаження виводити на сторінку HTML-код , що сформує той чи інший елемент сторінки після її завантаження чи провести певні обрахунки.

jQuery - популярна JavaScript-бібліотека з відкритим сирцевим кодом. Вона була представлена у січні 2006 року у BarCamp NYC Джоном Ресіґом (John Resig). Згідно з дослідженнями організації W3Techs, JQuery використовується понад половиною від мільйона найвідвідуваніших сайтів. jQuery є найпопулярнішою бібліотекою JavaScript, яка посилено використовується на сьогоднішній день.

Синтаксис jQuery розроблений, щоб зробити орієнтування у навігації зручнішим завдяки вибору елементів DOM, створенню анімації, обробки подій, і розробки AJAX-застосунків. jQuery також надає можливості для розробників, для створення плагінів у верхній частині бібліотеки JavaScript. Використовуючи ці об'єкти, розробники можуть створювати абстракції для низькорівневої взаємодії та створювати анімацію для ефектів високого рівня. Це сприяє створенню потужних і динамічних веб-сторінок.

**1.2 HTML**

HTML (від англ. Hypertext Markup Language - мова розмітки гіпертексту) - це стандартна мова розмітки документів у Всесвітній павутині. Всі веб-сторінки створюються за допомогою мови HTML (або XHTML). Мова HTML інтерпретується браузером і відображається у вигляді документа, зручному для людини. HTML є додатком SGML (стандартної узагальненої мови розмітки) і відповідає міжнародному стандарту ISO 8879.

HTML-документ є текстовим файлом розмічений за допомогою спеціальних (природно, текстових) команд. Текстовий формат представлення веб-документів був вибраний виходячи з основних вимог до веб-документу: простота, можливість безпосередньої інтерпретації в будь-якій операційній системі, мінімальний розмір файлу, зручність редагування і інтерпретації.

Мова розмітки гіпертекстових документів HTML дозволяє визначити різні типи елементів (у оригіналі element), що забезпечують функціональність документа: текстові фрагменти із заданими параметрами форматування, списки, таблиці, зображення, гіперпосилання і т.д. Елементи HTML оголошуються за допомогою команд розмітки, званих тегами (від англійського tag - ярлик). Усs HTML-теги, що зустрічаються в тексті документа інтерпретуються браузером при відображенні документа.

**1.3 CSS**

Cascading Style Sheets (каскадні таблиці стилів) - технологія опису зовнішнього вигляду документа, написаного мовою розмітки. CSS використовується переважно для оформлення HTML- і XHTML-документів, але іноді і для інших XML-структурованих документів (наприклад, в браузері Mozilla для оформлення елементів графічного інтерфейсу, XUL).

CSS використовується розробниками веб-сторінок для завдання кольорів, шрифтів, розташування і інших аспектів представлення документа. Основною метою розробки CSS було розділення вмісту (написаного на HTML або іншій мові розмітки) від представлення стилю документа. Це розділення може збільшити доступність документа, надати велику гнучкість і можливість управління його виглядом, а також зменшити складність і повторюваність в структурному вмісті. Крім того, CSS дозволяє представляти один і той же документ в різних стилях.

Стандарт CSS визначає пріоритети, у порядку яких застосовуються правила стилів, якщо для якогось елементу підходять деякі правила одночасно. Це називається "каскадом", в якому для правил розраховуються пріоритети або "ваги", що робить результати передбаченими.

Таблиця стилів складається з набору правил. Кожне правило, у свою чергу, складається з одного або декількох селекторів, розділених комами і блоку визначень.

**1.4 PHP**

PHP (англ. PHP:Hypertext Preprocessor — PHP: гіпертекстовий препроцесор), попередньо: Personal Home Page Tools — скриптова мова програмування, була створена для генерації HTML-сторінок на стороні веб-серверу. PHP є однією з найпоширеніших мов, що використовуються у сфері веб-розробок (разом із Java, .NET, Perl, Python, Ruby). PHP підтримується переважною більшістю хостинг-провайдерів. Проект за яким був створений PHP — проект з відкритими програмними кодами.

Історія PHP починається з [1995](https://uk.wikipedia.org/wiki/1995) року, коли Расмус Лердорф ([англ.](https://uk.wikipedia.org/wiki/%D0%90%D0%BD%D0%B3%D0%BB%D1%96%D0%B9%D1%81%D1%8C%D0%BA%D0%B0_%D0%BC%D0%BE%D0%B2%D0%B0) Rasmus Lerdorf) створив простий [застосунок](https://uk.wikipedia.org/wiki/%D0%97%D0%B0%D1%81%D1%82%D0%BE%D1%81%D1%83%D0%BD%D0%BE%D0%BA) мовою Perl, що аналізував відвідування користувачами його резюме на веб-сайті. Потім, коли цим [застосунком](https://uk.wikipedia.org/wiki/%D0%97%D0%B0%D1%81%D1%82%D0%BE%D1%81%D1%83%D0%BD%D0%BE%D0%BA) вже користувалися кілька чоловік, а число охочих одержати його постійно збільшувалося, Лердорф назвав своє творіння Особисті інструменти домашньої сторінки [англ.](https://uk.wikipedia.org/wiki/%D0%90%D0%BD%D0%B3%D0%BB%D1%96%D0%B9%D1%81%D1%8C%D0%BA%D0%B0_%D0%BC%D0%BE%D0%B2%D0%B0) Personal Home Page Tools версія 1 і виставив для вільного завантаження. З цієї миті почався небувалий зліт популярності PHP.

Як це завжди буває, терміново було потрібне доопрацювання і нові доповнення. Для їхньої реалізації Расмус створює нову версію пакету, тепер уже написану на [С](https://uk.wikipedia.org/wiki/%D0%A1). Отриманий таким чином інструмент набуває робочої назви [PHP/FI](https://uk.wikipedia.org/w/index.php?title=PHP/FI&action=edit&redlink=1) [Персональна Домашня сторінка / Інтерпретатор Форм](https://uk.wikipedia.org/w/index.php?title=%D0%9F%D0%B5%D1%80%D1%81%D0%BE%D0%BD%D0%B0%D0%BB%D1%8C%D0%BD%D0%B0_%D0%94%D0%BE%D0%BC%D0%B0%D1%88%D0%BD%D1%8F_%D1%81%D1%82%D0%BE%D1%80%D1%96%D0%BD%D0%BA%D0%B0_/_%D0%86%D0%BD%D1%82%D0%B5%D1%80%D0%BF%D1%80%D0%B5%D1%82%D0%B0%D1%82%D0%BE%D1%80_%D0%A4%D0%BE%D1%80%D0%BC&action=edit&redlink=1) надалі він також буде відомий під назвою [PHP 2](https://uk.wikipedia.org/w/index.php?title=PHP_2&action=edit&redlink=1). Ця версія вже більшою мірою схожа на сьогоднішній PHP. Вона мала синтаксис і спосіб іменування змінних в стилі мови Perl, можливість вбудовування PHP операторів в [html](https://uk.wikipedia.org/wiki/Html)-код сторінки, автоматичну інтерпретацію форм, [інтеграцію](https://uk.wikipedia.org/wiki/%D0%86%D0%BD%D1%82%D0%B5%D0%B3%D1%80%D0%B0%D1%86%D1%96%D1%8F) з базами даних

Закінчивши роботу Зеєв і Енді домовились з Расмусом про співпрацю в галузі розвитку та вдосконалення мови. З цієї миті з'являється PHP Group — група однодумців, що працюють над розвитком технології PHP. Одержаний продукт з'явився на світ у 1998 році під назвою PHP 3.

За час з [2000](https://uk.wikipedia.org/wiki/2000) по 2004 рік продовжувалися активні роботи з покращення 4 версії, але майже відразу [PHP Group](https://uk.wikipedia.org/w/index.php?title=PHP_Group&action=edit&redlink=1) приступила до продумування можливостей нової версії. В першу чергу було вирішено підсилити об'єктні можливості мови, що дозволяло використовувати його для реалізації масштабних проектів. Роботи із створення версії 5 велися тривалий час, в них брало участь рекордна кількість фахівців, зокрема Стерлінг Хьюз ([англ.](https://uk.wikipedia.org/wiki/%D0%90%D0%BD%D0%B3%D0%BB%D1%96%D0%B9%D1%81%D1%8C%D0%BA%D0%B0_%D0%BC%D0%BE%D0%B2%D0%B0) Sterling Hughes) і Маркус Бергера ([англ.](https://uk.wikipedia.org/wiki/%D0%90%D0%BD%D0%B3%D0%BB%D1%96%D0%B9%D1%81%D1%8C%D0%BA%D0%B0_%D0%BC%D0%BE%D0%B2%D0%B0) Marcus Boerger).

У липні [2004](https://uk.wikipedia.org/wiki/2004) року виходить офіційний [реліз](https://uk.wikipedia.org/wiki/%D0%A0%D0%B5%D0%BB%D1%96%D0%B7_%28%D0%BF%D1%80%D0%BE%D0%B3%D1%80%D0%B0%D0%BC%D0%BD%D0%B5_%D0%B7%D0%B0%D0%B1%D0%B5%D0%B7%D0%BF%D0%B5%D1%87%D0%B5%D0%BD%D0%BD%D1%8F%29) [PHP 5](https://uk.wikipedia.org/wiki/PHP_5). В першу чергу, як і планувалося, було перероблено весь механізм роботи з об'єктами. І якщо в попередніх версіях об'єктно-орієнтоване програмування на PHP було можливе в мінімальному ступені, а тому і використовувалося на практиці не часто, то PHP 5 володіє прекрасним потенціалом реалізації об'єктного програмування. Окрім цього, PHP збагатився рядом цінних розширень для роботи з [XML](https://uk.wikipedia.org/wiki/XML), різними джерелами даних, генерації [графіки](https://uk.wikipedia.org/wiki/%D0%93%D1%80%D0%B0%D1%84%D1%96%D0%BA%D0%B0) і інше.

З точки зору [системи типізації](https://uk.wikipedia.org/wiki/%D0%A1%D0%B8%D1%81%D1%82%D0%B5%D0%BC%D0%B0_%D1%82%D0%B8%D0%BF%D1%96%D0%B7%D0%B0%D1%86%D1%96%D1%97), PHP є мовою програмування з [динамічною типізацією](https://uk.wikipedia.org/wiki/%D0%94%D0%B8%D0%BD%D0%B0%D0%BC%D1%96%D1%87%D0%BD%D0%B0_%D1%82%D0%B8%D0%BF%D1%96%D0%B7%D0%B0%D1%86%D1%96%D1%8F). Немає необхідності явного визначення типу змінних, хоча така можливість існує. В разі звернення до змінної, [інтерпретатор](https://uk.wikipedia.org/wiki/%D0%86%D0%BD%D1%82%D0%B5%D1%80%D0%BF%D1%80%D0%B5%D1%82%D0%B0%D1%82%D0%BE%D1%80) PHP трактує її тип відповідно до контексту. За необхідності можливе приведення змінної до певного типу за допомогою відповідних конструкцій мови. Це може знадобитись, якщо зважити, що значення змінної можуть трактуватись по-різному в залежності від її типу. Також можливе визначення типу відповідної змінної на певному етапі виконання сценарію. Імена змінних чутливі до регістру символів.

**1.5 Symfony**

Symfony — відкритий [PHP](https://uk.wikipedia.org/wiki/PHP)-[фреймворк](https://uk.wikipedia.org/wiki/%D0%A4%D1%80%D0%B5%D0%B9%D0%BC%D0%B2%D0%BE%D1%80%D0%BA" \o "Фреймворк), що реалізує концепцію [модель-вид-контролер](https://uk.wikipedia.org/wiki/%D0%9C%D0%BE%D0%B4%D0%B5%D0%BB%D1%8C-%D0%B2%D0%B8%D0%B4-%D0%BA%D0%BE%D0%BD%D1%82%D1%80%D0%BE%D0%BB%D0%B5%D1%80) ([MVC](https://uk.wikipedia.org/wiki/MVC)) та автоматизовує найзагальніші веб-задачі, являє собою широконалаштовну систему пов'язаних [класів](https://uk.wikipedia.org/wiki/%D0%9A%D0%BB%D0%B0%D1%81_%28%D0%BF%D1%80%D0%BE%D0%B3%D1%80%D0%B0%D0%BC%D1%83%D0%B2%D0%B0%D0%BD%D0%BD%D1%8F%29) і призначений для розробки та керування [веб-застосунками](https://uk.wikipedia.org/wiki/%D0%92%D0%B5%D0%B1-%D0%B7%D0%B0%D1%81%D1%82%D0%BE%D1%81%D1%83%D0%BD%D0%BE%D0%BA). Випускається під [МІТ](https://uk.wikipedia.org/wiki/%D0%9B%D1%96%D1%86%D0%B5%D0%BD%D0%B7%D1%96%D1%8F_MIT) ліцензією. Symfony є [вільним програмним забезпеченням](https://uk.wikipedia.org/wiki/%D0%92%D1%96%D0%BB%D1%8C%D0%BD%D0%B5_%D0%BF%D1%80%D0%BE%D0%B3%D1%80%D0%B0%D0%BC%D0%BD%D0%B5_%D0%B7%D0%B0%D0%B1%D0%B5%D0%B7%D0%BF%D0%B5%D1%87%D0%B5%D0%BD%D0%BD%D1%8F). [Веб-сайт](https://uk.wikipedia.org/wiki/%D0%92%D0%B5%D0%B1-%D1%81%D0%B0%D0%B9%D1%82" \o "Веб-сайт) першоі версії symfony-project.com був запущений [18](https://uk.wikipedia.org/wiki/18) [жовтня](https://uk.wikipedia.org/wiki/%D0%96%D0%BE%D0%B2%D1%82%D0%B5%D0%BD%D1%8C) [2005](https://uk.wikipedia.org/wiki/2005) року. Symfony не варто плутати із [Symphony CMS](https://uk.wikipedia.org/w/index.php?title=Symphony_CMS&action=edit&redlink=1) відкритої [XML](https://uk.wikipedia.org/wiki/XML)/[XSLT](https://uk.wikipedia.org/wiki/XSL_Transformations) [CMS](https://uk.wikipedia.org/wiki/%D0%A1%D0%B8%D1%81%D1%82%D0%B5%D0%BC%D0%B0_%D0%BA%D0%B5%D1%80%D1%83%D0%B2%D0%B0%D0%BD%D0%BD%D1%8F_%D0%B2%D0%BC%D1%96%D1%81%D1%82%D0%BE%D0%BC)

Symfony спрямований на прискорення створення та підтримки веб-застосунків, а також для уникнення витрат часу для розв'язування тривіальних задач у розробці (наприклад, написання валідаторів форм).

Для встановлення Symfony необхідний [Unix](https://uk.wikipedia.org/wiki/Unix), [Linux](https://uk.wikipedia.org/wiki/Linux), [Mac OS](https://uk.wikipedia.org/wiki/Mac_OS) чи [Windows](https://uk.wikipedia.org/wiki/Windows) із [веб-сервером](https://uk.wikipedia.org/wiki/%D0%92%D0%B5%D0%B1-%D1%81%D0%B5%D1%80%D0%B2%D0%B5%D1%80) та встановленим [PHP 5](https://uk.wikipedia.org/wiki/PHP). Symfony сумісний із такими [об'єктно-реляційними відображеннями](https://uk.wikipedia.org/wiki/%D0%9E%D0%B1%27%D1%94%D0%BA%D1%82%D0%BD%D0%BE-%D1%80%D0%B5%D0%BB%D1%8F%D1%86%D1%96%D0%B9%D0%BD%D0%B5_%D0%B2%D1%96%D0%B4%D0%BE%D0%B1%D1%80%D0%B0%D0%B6%D0%B5%D0%BD%D0%BD%D1%8F), як [Doctrine](https://uk.wikipedia.org/wiki/Doctrine_%28PHP%29) та [Propel](https://uk.wikipedia.org/wiki/Propel).

За допомогою [акселератора PHP](https://uk.wikipedia.org/wiki/%D0%90%D0%BA%D1%81%D0%B5%D0%BB%D0%B5%D1%80%D0%B0%D1%82%D0%BE%D1%80_PHP) Symfony збільшує продуктивність та зменшує навантаження на сервер.

Symfony ставить за мету дати розробникам повний контроль над конфігурацією: майже все можливо налаштувати, від структури каталогів до сторонніх [бібліотек](https://uk.wikipedia.org/wiki/%D0%9F%D1%80%D0%BE%D0%B3%D1%80%D0%B0%D0%BC%D0%BD%D0%B0_%D0%B1%D1%96%D0%B1%D0%BB%D1%96%D0%BE%D1%82%D0%B5%D0%BA%D0%B0).

Symfony використовує шаблон проектування [модель-вид-контролер](https://uk.wikipedia.org/wiki/%D0%9C%D0%BE%D0%B4%D0%B5%D0%BB%D1%8C-%D0%B2%D0%B8%D0%B4-%D0%BA%D0%BE%D0%BD%D1%82%D1%80%D0%BE%D0%BB%D0%B5%D1%80). Розробників Symfony надихнули такі фреймворки, як [Ruby on Rails](https://uk.wikipedia.org/wiki/Ruby_on_Rails), [Django](https://uk.wikipedia.org/wiki/Django) та [Spring Framework](https://uk.wikipedia.org/wiki/Spring_Framework).

Використовуючи систему [плагінів](https://uk.wikipedia.org/wiki/%D0%9F%D0%BB%D0%B0%D0%B3%D1%96%D0%BD), Symfony може підтримувати [JavaScript](https://uk.wikipedia.org/wiki/JavaScript) фреймворків та багато інших [PHP](https://uk.wikipedia.org/wiki/PHP) проектів:

Ще однією перевагою Symfony є наявність генераторів, за допомогою яких значно пришвидшується розробка.

Серед можливостей: інструменти для локалізації та інтернаціоналізації, unit-тестування, БД-абстракції, smart-URL, Debug Toolbar, development та production режими, form framework.

Відрізняється доволі якісною документацією та великою спільнотою.

**РОЗДІЛ 2. ПРАКТИЧНА ЧАСТИНА**

**2.1 Структура веб-ресурсу**

Після того, як користувач заходить в середовище веб-ресурсу перед ним зявляється головне вікно програми (Рис. 2.1)

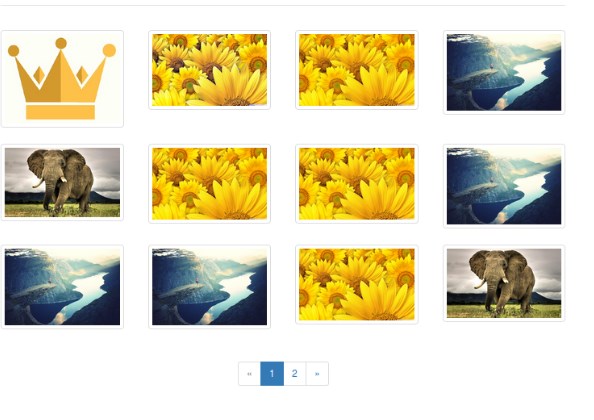


Рис. 2.1 – головне вікно програми

Як видно на зображення користувач може перейти на наступну сторінку або повернутися на попередню за допомогою нумерації сторінок внизу програми, а також за допомогою відповідних стрілок-вказівників.

Також користувач має можливість завантажити нове зображення. Для цього необхідно вибрати пункт меню «Upload», який знаходиться вверху екрану на головній сторінці (Рис. 2.2)

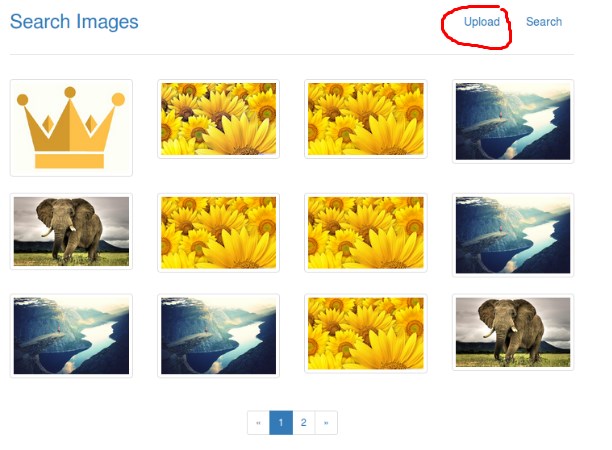


Рис. 2.2 - завантаження файлу

Після того як користувач вибере даний пункт меню перед ним з’явиться сторінка завантаження нового файлу (Рис. 2.3)

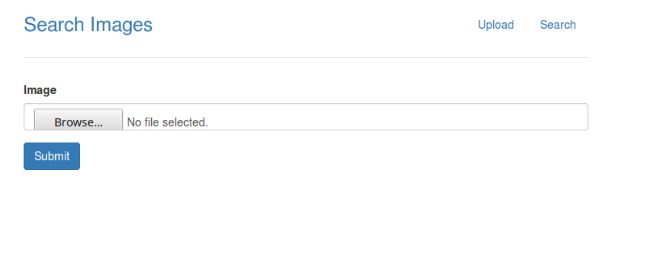


Рис. 2.3 – сторінка завантаження файлу

Після того користувач може обрати необхідний йому файл з свого комп’ютера. Для цього перед користувачем з’являється вікно завантаження файлу (Рис. 2.4)

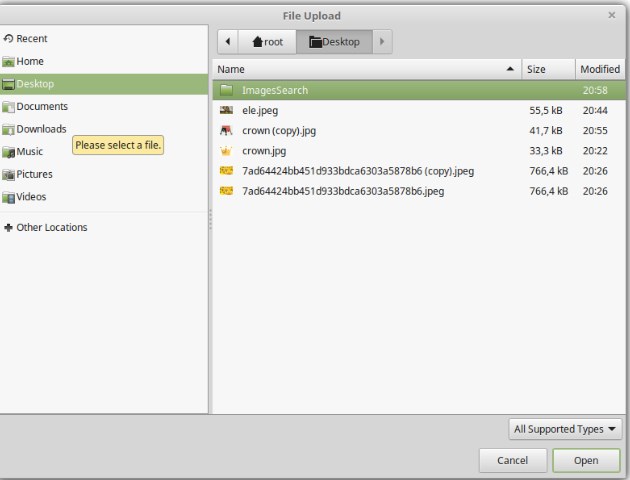


Рис. 2.4 – вікно завантаження файлу

Після цього, якщо файл зображення успішно завантажено, то з’являється сторінка успішного завантаження картинки (Рис. 2.5)

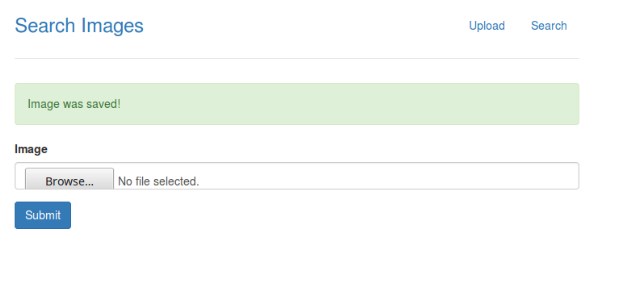


Рис. 2.5 – сторінка успішного завантаження

Крім цього, користувач для здійснення пошуку може завантажувати картинки користуючись таким самим принципом як описано вище.

Завантаживши вибрану картинку для пошуку її в базі для користувача виникають два можливих варіанти. Перший – це пошук не дав результатів, такого зображення нема (Рис. 2.6)

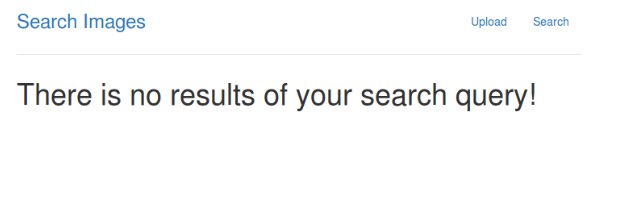


Рис. 2.6 – пошук не дав результатів

Другим варіантом є вдалий пошук, якщо така картинка дійсно знаходиться в базі (Рис. 2.7)

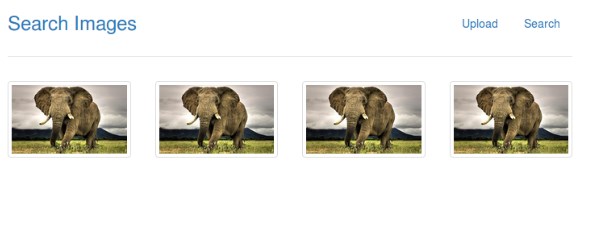


Рис. 2.7 – результати пошуку

Після цього користувач може вибрати знайдену картинку і перед ним з’явиться вікно перегляду (Рис. 2.8)



Рис. 2.8 – вікно перегляду зображення

**2.2 Етапи розробки проекту**

Послідовна розробка проекту дозволяє чітко відокремити кожен, з етапів реалізації тієї чи іншої функції. Така розмежованість надає змогу без затрат часу на вивчення коду вносити зміни тільки в ту частину проекту, котру дійсно потрібно, а не змінювати весь веб-додаток.

Першим етапом розробки було створення головної сторінки, де показані всі зображення, які є в базі розробленого ресурсу. Тут користувач отримує можливість переходити зі сторінки на сторінку та повертатись на попередню. Саме

Після закінчення реалізації оформлення головної сторінки та функції переходу до наступних елементів ресурсу, розпочалась розробка рівнів. Першочергово було розроблено функції пошуку та завантаження зображень для пошуку. Далі розроблено можливість вибирати одну із запропонованих картинок для окремого детального перегляду. Оскільки основна ідея роботи ресурсу не потребує якихось додаткових елементів на зразок коментарів, то ці деталі було упущено для того, щоб зберегти лаконічний інтерфейс

Наприкінці розпочався етап візуального оформлення, до складу якого входить вступна частина та візуальні блоки.

Реалізація цих етапів цілком і повністю покладена на каскадну модель програмування, котра дозволяє покроково виконувати весь функціонал проекту та розпочинати наступний етап тільки після того, як попередній був закінчений. Такий підхід до організації роботи дозволяє наперед визначити інструментарій та ресурсі, які будуть використовуватись при розробці веб – ресурсу.

**2.3. Огляд аналогів**

Зараз існує чимало програм для пошуку зображень. Для їх функціонування достатньо всього-лиш мати доступ до Інтернету. Найбільш відомою та популярною є методика пошуку зображення за допомогую функції пошуковика Google, яка і має відповідну назву «Google зображення ». Працює воно дуже просто. Достатньо в пошуковику ввести «Google зображення» і перед користувачем виникає наступне вікно (Рис. 2.9)

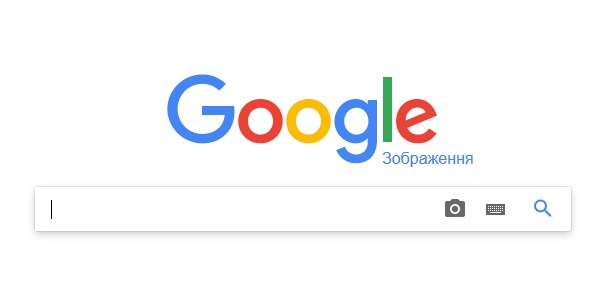


Рис. 2.9 – сторінка пошуку зображення

Потім необхідно вибрати пошук і один з методів завантаження зображення: посилання на нього або безпосередньо з комп’ютера (Рис. 2.10)

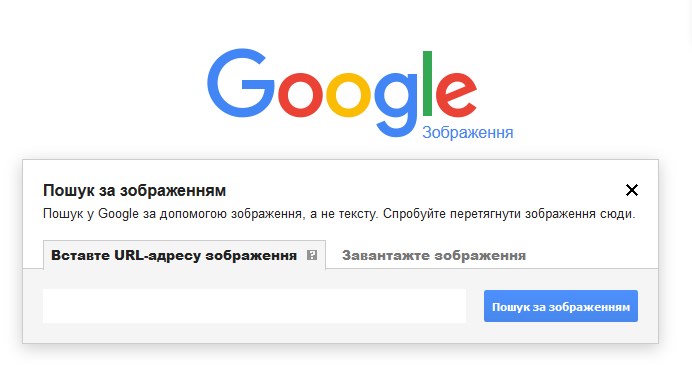


Рис. 2.10 – завантаження зображення

**Висновки**

Актуальність даного завдання обумовлена зростанням розмірів колекцій мультимедіа даних в різних сферах людської діяльності, що призводить до численних питань забезпечення ефективного зберігання, стиснення, інтерпретації і, в тому числі, забезпечення швидкого пошуку. Метою є дослідження алгоритмів пошуку зображень за зразком в базах даних і синтез методів індексації в метричному просторі для забезпечення шви-якого пошуку зображень. Об'єктом дослідження є система управління базами даних зображень. Предметом дослідження є методи метричної індексації вмісту баз відеоданих на базі метричного порівняння результатів сегментації зображень. Проблема пошуку набору елементів, які близькі до даного, запитуваного, елементу за деякими критеріями подібності є на сьогоднішній день надзвичайно актуальною. Для зберігання інформації в усіх сучасних аналізаторах зображень передбачено використання бази даних. База даних візуальної інформації перед-ставлять собою програмне забезпечення, призначене для накопичення, зберігання, архівування інформації, її систематизації і подальшого швидкого пошуку зображень подібних заданому. Це дозволяє вирішувати різні прикладні завдання, пов'язані з обробкою і розпізнаванням зображень.

В умовах постійно зростаючих обсягів реєстрованих відеоданих необхдно синтез нових, які відповідають висунутим вимогам по валідності і швидкодії методів і моделей пошуку зображень в базах даних. Аналіз поточного стану досліджень в області пошуку і розроблених програмних комплексів показав, що перспективним і вже зараз реалізовуються на практиці підходом є реалізація запитів за зразком з порівнянням зображень у вигляді результатів їх сегментації.

В рамках даної роботи були досліджені методи і моделі пошуку зображень в базах даних за зразком. Для порівняння зображень було вибрано метричний спосіб порівняння результатів сегментації зображень. На цій основі синтезовані і досліджені методи індексації, названі метричними, які дозволяють скорочувати кількість операцій порівняння на етапі пошуку, а саме: метод на повній матриці відстаней, на «розрідженій» матриці відстаней, метод, який створює дерева пошуку, метод, заснований на розбитті ( кластеризації) даних.

Для дослідження цільової моделі пошуку зображень і синтезованих методів метричної індексації був створений дослідний програмний комплекс. Результати імітаційного моделювання, результати експериментів по скороченню кількості операцій порівняння на етапі виконання запиту в системі пошуку зображень за допомогою методів метричної індексації, порівняльний аналіз з існуючими системами-аналогами підтвердили ефективність запропонованого підходу.

**Список використаних джерел**

1. Chan K., Liu F., Purnomo R., Xiong X. Content-based image retrieval using regional representation // International Workshop on Theoretical Foundations of Computer Vision: Multi-Image Analysis / Klette R. et al. (Eds.). – Lecture Notes In Computer Science. – London, UK: Springer-Verlag. – Vol. 2032. – 2000. – P. 238−250.

2. Bozkaya T., Ozsoyoglu M. Indexing large metric spaces for similarity search queries // ACM Trans. on Database Systems (TODS). – Vol. 24, No. 3. – 1999. – P. 361−404.

3. Гонсалес Р., Вудс Р. Цифровая обработка изображения: Пер. с англ. – М.: Техносфера. – 2005. – 1070 с.

4. Chang N.S., and Fu K.S. Query by pictorial example // IEEE Trans. on Software Engineering. − Vol. 6, No. 6. – 1980. – P. 519−524.

5. Smeulders A.W., Worring M., Santini S., Gupta A., Jain R. Content-based image retrieval at the end of the early years // IEEE Trans. on Pattern Analysis and Machine Intelligence. − Vol. 22, No. 12. − 2000. – P. 1349–1380.

6. Форсайт Д., Понс Ж. Компьютерное зрение: современный подход: Пер. с англ.. − М.: Издательский дом «Вильямс». – 2004. – 928 с.

7. Шапиро Л., Стокман Дж. Компьютерное зрение: Пер. с англ. – М.: Бином. – 2006. – 752 с.

8. Rui Y., Huang T., Chang S.-F. Image retrieval: Current techniques, promising directions and open issues // Journal of Visual Communication and Image Representation. – Vol. 10, No. 4. − 1999. – P. 39–62.

9. Veltkamp R.C., Hagendoorn M. State-of-the-art in shape matching // Principles of visual information retrieval. – 2000. – P. 87−119.

10. Zhang D., Lu G. Review of shape representation and description techniques // Pattern Recognition. – Vol. 37. – 2003. − P. 1−19.

11. Mikolajczyk K., Schmid C. A performance evaluation of local descriptors // IEEE Trans. on Pattern Analysis and Machine Intelligence. – Vol. 27, No. 10. – 2005. − P. 1615−1630.

12. Liu Y., Zhanga D., Lua G., Ma W.-Y. A survey of content-based image retrieval with high-level semantics // Pattern Recognition. – Vol. 40, No. 1. – 2007. – P. 262–282.

13. Дуда Р., Харт П. Распознавание образов и анализ сцен: Пер. с англ. − М.: Мир. – 1976. – 512 с.

14. Прэтт У. Цифровая обработка изображений. − М.: Мир. – 1982. – 792 с

15. Путятин Е.П., Аверин С.И. Обработка изображений в робототехнике. − М.: Машиностроение. – 1990. – 320 с.

16. Ту Дж., Гонсалес Р. Принципы распознавания образов: Пер. с англ. – М.: Мир. – 1978. – 416 с.

17. Deng Y., Manjunath B. Unsupervised segmentation of color-texture regions in images and video // IEEE Trans. on Pattern Analisys and Machine Intelligence. – Vol. 23, No 8. – 2001. – P. 800−810.

18. Carson C., Belongie S., Greenspan H., Malik J. Blobworld: image segmentation using expectation-maximization and its application to image querying // IEEE Trans. on Pattern Analysis and Machine Intelligence. Vol. 24, No. 8. – 2002. – P. 1026–1038.

19. Shi J., Malik J. Normalized cuts and image segmentation // IEEE Trans. on Pattern Analysis and Machine Intelligence. – Vol. 22, No 8. – 2000. – P. 888−905.

20. Rubner Y., Tomasi C., Guibas L. The Earth Mover’s Distance as a metric for image retrieval // International Journal of Computer Vision / Springer Netherlands. – Vol. 40, No 2. – 2000. – P. 99−121.

21. Long F., Zhang H., Feng D. Fundamentals of content-based image retrieval // Multimedia Information Retrieval and Management − Techonological Fundamentals and Applications / D. Feng, W. Sie. (Eds.). − Springer. − 2003. – P. 1−32.

22. Ko B., Byun H. Integrated region-based image retrieval using region’s spatial relationships // Proc. of the 16th International Conference on Pattern Recognition. – Vol. 1. – 2002. – P. 196−199.

23. Jiang X., Marti C., Irniger C., Bunke H. Image segmentation evaluation by techniques of comparing clusterings // Image Analysis and Processing – ICIAP 2005 / Fabio R., Sergio V. (Eds.). – Lecture Notes in Computer Science. – Vol. 3617. – 2005. – P. 344−351.

24. Meila M. Comparing clusterings by the variation of information // Proc. of the 6th Annual Conference on Learning Theory. – 2003. – P. 173–187.

25. Kinoshenko D., Mashtalir V., Shlyakhov V. A partition metric for clustering features analysis // International Journal Information Theories and Applications. –Vol. 14, No. 3. – 2007. – Р. 230−236.

26. Bohm C., Berchtold S., Keim D.A. Searching in high-dimensional spaces: Index structures for improving the performance of multimedia databases // ACM Computing Surveys (CSUR). – Vol. 33, No. 3. – 2001. – P. 322−373.

27. Vidal E. An algorithm for finding nearest neighbours in (approximately) constant average time // Pattern Recognition Letters. – Vol. 4, No. 3. – 1986. – P. 145−157.

28. Mico M.L., Oncina J., Vidal E. A new version of the nearest-neighbour approximating and eliminating search algorithm (AESA) with linear preprocessing time and memory requirements // Pattern Recognition Letters. – Vol. 15, No. 1. – 1994. – P. 9−17.

29. Uhlmann J. Satisfying general proximity/similarity queries with metric trees // Information Processing Letters. – Vol. 40. – 1991. – P. 175−179.

30. Brin S. Near neighbor search in large metric spaces // Proc. of the 21st International Conference on Very Large Data Bases / Dayal U., Gray P., Nishio S. (Eds.). − Very Large Data Bases. – 1995. − P. 574−584.