

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
Тернопільський національний економічний університет
Факультет комп'ютерних інформаційних технологій
Кафедра комп'ютерної інженерії

Новосад Христина Василівна

**Веб-ресурс наукової групи "Аналіз біомедичних
зображень" / Web resource of the scientific group
"Biomedical Image Analysis "**

напрямок підготовки: 6.050102 - Комп'ютерна інженерія
фахове спрямування - Комп'ютерні системи та мережі
Бакалаврська робота

Виконав студент групи КСМ 42/1
Христина Василівна Новосад

Науковий керівник:
Піцун О. Й.

Тернопіль –2018

РЕЗЮМЕ

Дипломний проект містить 73 сторінки пояснюючої записки, 20 рисунків, 17 таблиць, 3 додатки. Обсяг графічного матеріалу 3 аркуші формату А3.

Метою дипломного проекту є розробка веб – сайту наукової групи «Аналіз біомедичних зображень» кафедри комп'ютерної інженерії ТНЕУ.

В дипломному проекті на основі аналізу існуючих сайтів наукових груп обгрунтовано вибір обов'язкових модулів, розроблено алгоритм пошуку публікацій в базі даних з використанням сучасних технологій відображення інформації в режимі реального часу, розроблено алгоритм авторизації на сайті.

Враховуючи сучасні тенденції до розробки веб – сайтів та недоліки існуючих сайтів наукових груп, використано фреймворк Semantic UI для клієнтської частини та Laravel 5.3 для серверної, що дозволило забезпечити адаптивність сайту до дисплеїв із різними розширеннями, наприклад, мобільних телефонів, планшетів, широкоформатних моніторів. Розроблений сайт складається з таких модулів: наукові публікації учасників групи, статті з практичними результатами досліджень, інформація про учасників наукової групи, групи в соціальних мережах. Така інформація дозволить підвищити інформативність сайту, а наявність адаптивного графічного інтерфейсу забезпечить зручність отримання інформації на різних пристроях та покращить індексацію сайту пошуковими системами.

Ключові слова: ВЕБ – САЙТ, НАУКОВА ГРУПА, БІОМЕДИЧНІ ЗОБРАЖЕННЯ, АДАПТИВНИЙ ГРАФІЧНИЙ ІНТЕРФЕЙС.

RESUME

Diploma project contains 73 pages of main text, 20 figures, 17 tables, 3 additions. Volume of graphic material 3 leaves of format A3.

The purpose of the diploma project is to develop a web site of the scientific group «Biomedical Image Analysis» of the TNEU computer engineering department.

In the diploma project, based on the analysis of existing sites of scientific groups, the choice of obligatory modules was justified, the algorithm of publication of publications in the database with the use of modern technologies of information display in real time was developed, the authoring algorithm on the site was developed.

Given the current trends in the development of web - sites and disadvantages of existing sites research groups Semantic UI framework used for client side and Laravel 5.3 for server, allowing for adaptability to the site displays a variety of extensions naprylad, mobile phones, tablets, large-format displays. A site consists of the following modules: scientific publications group members, articles on practical research results, information on members of scientific groups, groups in social networks. This information will improve information content of the site and the availability of adaptive graphical user interface provide ease of obtaining information on different devices and improve site indexing by search engines.

Key words: WEB SITE, SCIENTIFIC GROUP, BIOMEDICAL IMAGES, ADAPTIVE GRAPHIC INTERFACE.

ЗМІСТ

Вступ.....	9
1 Аналітичний огляд підходів до створення веб-сайтів.....	11
1.1 Програмні засоби розробки веб-сайтів.....	11
1.2 Особливості веб-сайтів наукових груп.....	16
1.3 Аналіз веб-сайтів відомих наукових груп.....	18
1.4 Висновки та постановка задачі.....	23
2 Алгоритми роботи модулів сайту.....	24
2.1 Алгоритм пошуку наукових публікацій.....	24
2.2 Алгоритм авторизації в системі.....	26
2.3 Розробка структури бази даних.....	29
3 Програмна реалізація веб – сайту наукової групи.....	35
3.1 Структура програмного модуля.....	35
3.2 Unit – тестування.....	41
3.3 Порівняльний аналіз розробленого сайту з аналогами.....	44
4 Техніко – економічне обґрунтування.....	47
4.1 Розрахунок витрат на виконання проектного рішення.....	47
4.2 Визначення експлуатаційних витрат.....	54
4.3 Розрахунок ціни споживання проектного рішення.....	56
4.4 Визначення економічної ефективності.....	58
Висновки.....	60
Список використаних джерел.....	61
Додаток А Лістинг файлу «index.php».....	65
Додаток Б Лістинг файлу «index.blade.php».....	66
Додаток В Довідка про використання.....	73

					ДП. КСМ. 07139/14. 00.00.000 ПЗ			
Змн.	Арк.	№ докум.	Підпис	Дата				
Розроб.	Новосад Х.В.				ВЕБ – РЕСУРС НАУКОВОЇ ГРУПИ «АНАЛІЗ БІОМЕДИЧНИХ ЗОБРАЖЕНЬ»	Літ.	Арк.	Аркушів
Перевір.	Піщун О.Й.					н	8	73
Консуль.	Паздрій І.Р.					ТНЕУ.ФКІТ.КСМ-42/1		
Н. Контр.	Гураль І. В.							
Затверд.	Березький О.М							

ВСТУП

Робота працівників вищих учбових закладів полягає не лише у викладанні навчального матеріалу студентам, але й у проведенні наукової роботи, зокрема у написанні публікацій, участі у вітчизняних і міжнародних наукових конференціях тощо. Враховуючи те, що ми живемо в еру інформаційних технологій важливим завданням є розроблення інформаційних ресурсів для обміну даними, публікаціями, результатами досліджень наукових груп у всьому світі. Для цього завдання найкращим підходом є застосування веб – сайтів.

За допомогою веб – технологій науковці можуть заявити про себе та підвищити популярність власного навчального закладу. Основними елементами веб – сайтів наукових груп є модуль відображення наукових публікацій учасників групи (пошук публікацій, фільтр публікацій), модуль відображення інформації про учасників (Піб, електронна адреса, посилання на наукометричні бази), модуль виводу статей, модуль «Контакти». Форма зворотнього зв'язку дозволить налагодити комунікацію між відвідувачами сайту, адміністраторами та науковцями.

Наукова група «Аналіз біомедичних зображень» займається дослідженням та розробкою методів і програмних засобів для діагностування передракових та ракових станів на основі аналізу зображень. Публікації учасників наукової групи пов'язані із розробкою алгоритмів для низького, середнього та високого рівнів комп'ютерного зору. Включаючи засоби покращення якості зображень, фільтрації, адаптивного гістограмного вирівнювання, сегментації, виділенні кількісних характеристик мікрооб'єктів. Окрему увагу заслуговує розробка програмних засобів класифікації гістологічних та цитологічних зображень за допомогою згорткових нейронних мереж та сучасних класифікаторів. Також дана наукова група займається розробкою інтелектуальної системи автоматизованої мікроскопії для діагностування передракових станів. Дослідження учасників групи опубліковано у наукових фахових виданнях, виданнях, що індексуються наукометричною базою Scopus, презентовано на міжнародних конференціях.

					ДП. КСМ. 07139/14.00.00.000 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		9

Завдяки можливості обміну статтями та іншими публікаціями між науковцями зростає якість досліджень.

Метою дипломного проекту є розробка веб – сайту наукової групи «Аналіз біомедичних зображень» кафедри комп'ютерної інженерії Тернопільського національного економічного університету.

Актуальність роботи полягає у знаходженні нових шляхів популяризації наукових досліджень та наукових груп закладів вищої освіти, зокрема наукової групи «Аналіз біомедичних зображень».

					ДП. КСМ. 07139/14.00.00.000 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		10

1 АНАЛІТИЧНИЙ ОГЛЯД ДО СТВОРЕННЯ ВЕБ - САЙТІВ

1.1 Програмні засоби розробки веб-сайтів

Веб-сайт — це набір веб-сторінок, доступних у мережі Інтернеті, з об'єднаними навігаційно одним доменним ім'ям [1]. Доменне ім'я обслуговується групою серверів системи доменних імен. Такі сервери транлюють імена хостів в адреси. Веб – сайт розміщується на одному або декількох серверах. Доменне ім'я серверів складається із набору ASCII символів.

Для перегляду інформації через мережу використовують спеціальні програмні засоби – веб - браузері [2]. У більшості випадків браузер є складовою операційних систем, таких як Windows, Linux. Браузер лише відображає у зручному вигляді клієнту інформацію, отриману від сервера, а логіка роботи сайту, база даних, поштові сервери знаходяться на стороні сервера, тому користувач немає безпомереднього доступу до них.

Користувач може переглядати інформацію надіслану від сервера у текстовому, графічному, звуковому форматах [3]. Перевагою використання веб – сайтів для обміну інформацією у порівнянні із спеціалізованими додатками є універсальність, тобто не потрібно розробляти спеціалізовані програмні комплекси для різних платформ, достатньо використовувати веб - браузері [4]. Щоб не писати версії для Microsoft Windows, GNU/Linux, Mac OS X та інших операційних систем, додаток розробляється один раз для обраної платформи і на ній розгортається [5]. Ріст популярності розробки та використання веб – сайтів припадає на кінець 1990 - х — почато 2000 - х років [6]. Протягом останній 5-х років різко зросла популярність використання технологій для розробки додатків, що працюють в режимі реального часу. Зараз існує велика кількість технологій розробки веб – сайтів на клієнтській та серверній стороні [7]. На рисунку 1.1 наведено класифікацію мов програмування.

					ДП. КСМ. 07139/14.00.00.000 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		11

сайтів спеціальних фреймворків, розроблених на PHP.

PHP — це скриптова мова програмування, була розроблена для генерації HTML-сторінок на стороні веб-сервера [8]. Дана мова програмування останніми роками займає одне з перших місць за популярністю використання для розробки сайтів. PHP - мова, що поєднує переваги C, Perl та спеціально націлена на роботу в Інтернеті. Ключовою перевагою даної технології є те, що вона використовується більшістю хостинг провайдерами, тому ресурси, написані на ній легко імплементувати.

Основні переваги мови програмування PHP такі:

- підтримка ООП – завдяки використанню об'єктно – орієнтованого підходу забезпечується керованість та стійкість системи. Застосування базових парадигм об'єктно – орієнтованого програмування та використання спеціальних шаблонів розробки значно підвищує швидкість написання коду в команді та створює основні правила розробки програмного забезпечення;

- можливість компіляції – існують PHP компілятори, котрі коректно компілюють PHP скрипти, наприклад Roadsend PHP Compiler;

- наявність інтерфейсів - семантична і синтаксична конструкція в коді програми, використовується для специфікації послуг, які надаються класом. За допомогою інтерфейсів розширюються можливості використання ООП у програмі;

- наявність фреймворків – дозволяє спростити процес розробки базових модулів. Існуючі модулі реєстрації, авторизації користувачів, роботи з базою даних, захисту від взлому дозволяють пришвидшити та підвищити надійність розробленого засобу;

- узагальнене програмування - дозволяє записувати алгоритми, що приймають дані будь-якого типу, що у свою чергу спрощує написання програми.

Для налаштування та запуску веб – сайту потрібно встановити таке програмне забезпечення [13]:

- веб – сервер Apache;
- phpMyAdmin;
- mysql;

					ДП. КСМ. 07139/14.00.00.000 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		13

- інтерпретатор PHP;
емулятор sendmail і сервера SMTP.

Часто початківці використовують спеціальні програмні пакети для налаштування веб – серверів: Денвер, ХАМР.

Apache HTTP-сервер — це HTTP сервер. Головними перевагами є надійність, висока поширеність використання серед розробників та простота налаштування [14]. Apache дозволяє підключати зовнішні модулі, використовувати системи керування базами даних (СКБД) для аутентифікації користувачів, проксі - сервери. Однак, для використання високопродуктивних систем рекомендується використовувати сервер nginx, який є швидшим.

MySQL – безкоштовна СКБД [9]. СКБД - це програмний комплекс, котрий управляє базою даних, тобто бере на себе всі низькорівневі операції по роботі з файлами, завдяки чому програміст при роботі з базою даних може оперувати лише логічними конструкціями за допомогою мови програмування, не вдаючись до низькорівневих операцій.

PhpMyAdmin — це веб-додаток, призначений для адміністрування СКБД MySQL [11]. За допомогою браузера можна здійснювати адміністрування сервера MySQL, виконувати запити, здійснювати редагування вмісту таблиць баз даних. PhpMyAdmin користується великою популярністю у веб-розробників, адже реалізовує керування СКБД MySQL без написання SQL команд через зручний інтерфейс та із будь-якого комп'ютера під'єданого до мережі інтернет без необхідності інсталювання додаткового ПЗ [17].

Sendmail - це поштовий агент передачі для передачі повідомлень [18]. Даний програмний засіб поширюється безкоштовно разом з вихідними кодами. Sendmail випускає версії програми для всіх операційних систем та апаратних платформ. SMTP - це поштовий протокол, призначений для пересилання повідомлень до поштового сервера або між поштовими серверами. Для SMTP зареєстровано порт 25.

Робота практично будь-якого веб-сайту ґрунтується на архітектурі «клієнт-сервер» [10]. Розподіл завдань між клієнтом та сервером визначає модель клієнт-

					ДП. КСМ. 07139/14.00.00.000 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		14

серверної взаємодії. WWW є типовим прикладом «клієнт-серверної» взаємодії. Графічне представлення архітектури «клієнт – сервер» зображено на рисунку 1.2.

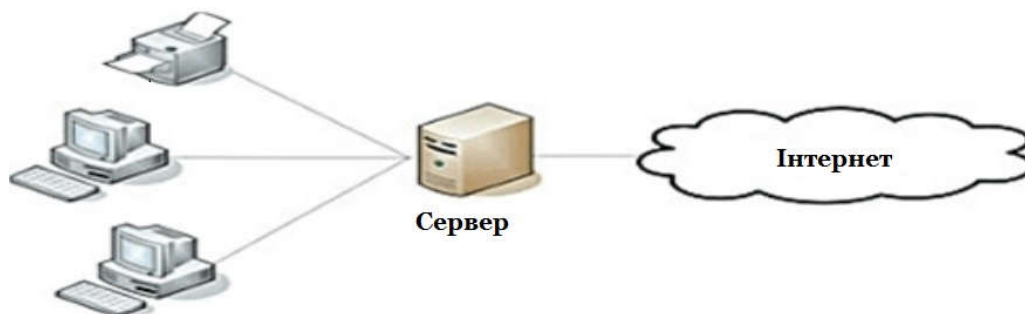


Рисунок 1.2 – Архітектура «клієнт – сервер»

Переваги та недоліки використання архітектури «клієнт - сервер» наведено на рисунку 1.3.

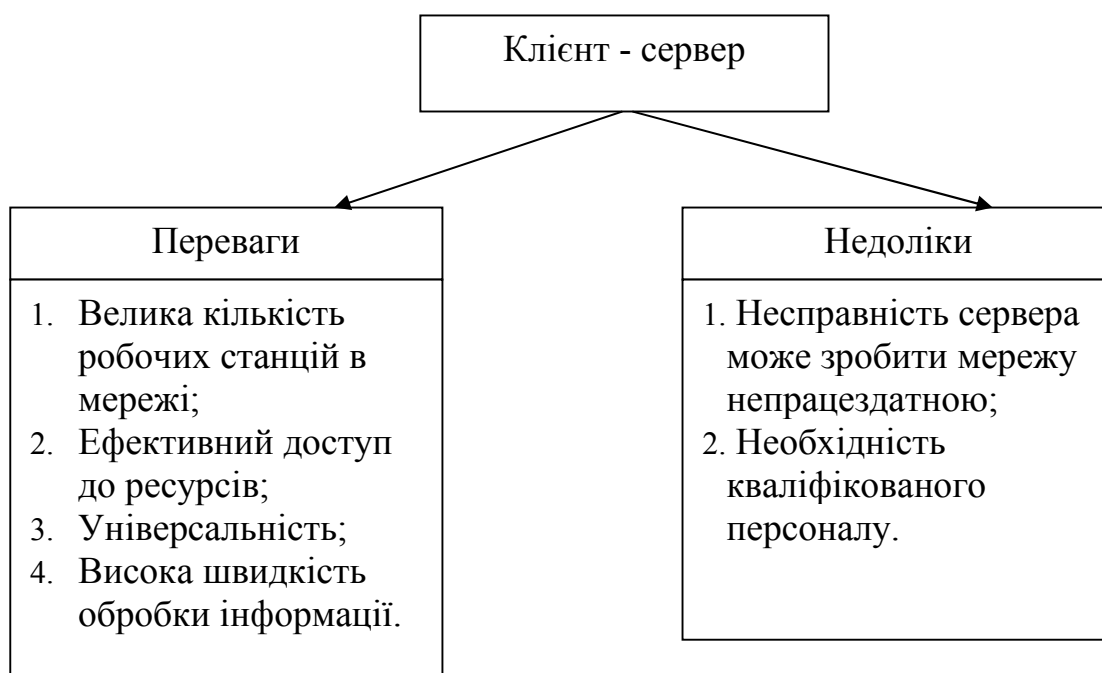


Рисунок 1.3 – Переваги та недоліки архітектури «клієнт - сервер»

Основною перевагою даної архітектури є універсальності, можливість організувати комп'ютерні мережі з великою кількістю робочих станцій, що стало актуально через збільшення кількості Інтернет користувачів у світі [12]. До того ж, архітектура «клієнт - сервер» характеризується високою швидкістю обробки інформації.

Отже, у даному підрозділі проведено огляд та аналіз сучасних програмних засобів розробки веб – сайтів, що дозволило виділити їх переваги та недоліки та встановити, що мова програмування PHP – найкраще підходить для реалізації поставленої задачі, сервер Apache2, mysql, інтерпретатор PHP – необхідне програмне забезпечення.

1.2 Особливості веб-сайтів наукових груп

Наука — сфера діяльності людини, спрямована на отримання (вироблення і систематизацію у вигляді теорій, гіпотез, законів природи або суспільства тощо) нових знань про навколишній світ. Основою науки є збирання, оновлення, систематизація, критичний аналіз фактів, синтез нових знань або узагальнень, що описують досліджувані природні або суспільні явища та (або) дозволяють будувати причинно-наслідкові зв'язки між явищами і прогнозувати їх перебіг.

У сучасному суспільному розвитку помічається посилений зв'язок між наукою, технікою і виробництвом. Наука дедалі глибше перетворюється на безпосередню виробничу силу суспільства. У такому процесі наука не стежить за розвитком техніки, а випереджає її, визнає сучасний прогрес матеріального виробництва, наукове знання пронизує усі сфери суспільного життя. Наука орієнтується, насамперед, на людину, розвиток її інтелекту, творчих здібностей, цілісний і всебічний розвиток. Посилюється взаємозв'язок між наукою, суспільством і культурою з багатьох напрямків, у тому числі у розрізі прикладних фундаментальних досліджень.

Останнім часом науковцями все більше застосовуються інформаційні системи для популяризації власних наукових досягнень та обміну досвідом. Ключовим у цьому аспекті є веб – ресурси наукових груп, що працюють в університетах.

					ДП. КСМ. 07139/14.00.00.000 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		16

Оскільки для більшості сторінок сайту рекомендується застосовувати єдиний стиль оформлення, то потрібно визначити схему розташування на сторінках основних блоків: як буде розташований основний матеріал, додаткові інформаційні та рекламні блоки, анонси, меню, лічильник відвідувачів тощо.

Як правило, на веб-сторінках передбачено розміщення [15]:

- заголовок;
- меню;
- інформаційний блок;
- підвал.

Наукова група – це діяльність викладачів та студентів яка спрямована на організаційне забезпечення та координацію наукової діяльності. На рисунку 1.4 зображено якими модулями володіють сайти наукових груп.

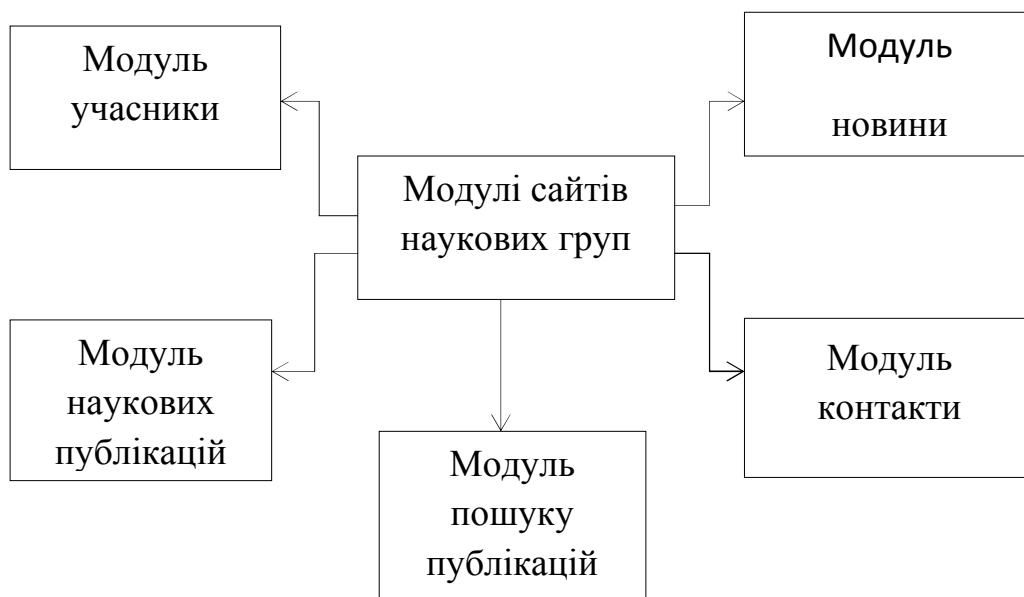


Рисунок 1.4 – Модулі сайтів наукових груп

Вміст модулів сайту наукової групи такий:

- меню (гіперпосилання на внутрішні та зовнішні ресурси складових веб - сайту);
- наукові публікації (список наукових публікацій учасників наукової групи, із вказанням приналежності до наукометричних баз);

- пошук та фільтрація публікацій (пошук інформації про публікації за усіма полями бази даних в режимі реального часу, фільтрація публікацій за авторами та роком видання);

- новини (наукові досягнення учасників групи, подані у зручній та зрозумілій формі, включаючи назву статті, короткий опис, автора, дату публікації, титульне зображення);

- учасники (відображає список учасників наукової групи із вказанням прізвища, імені, по-батькові, посадою, науковим ступенем, фото, посиланнями на перональні сторінки наукометричних баз).

- зворотній зв'язок (форма зворотнього зв'язку та блок соціальних мереж дозволяє здійснювати двосторонню комунікацію між учасниками групи та відвідувачами сайту).

Отже, у даному підрозділі на основі аналітичного підходу проведено аналіз базових елементів веб – сайту наукової групи, що дозволило виділити структурні елементи веб – сайту для вирішення поставленого завдання.

1.3 Аналіз веб-сайтів відомих наукових груп

Комп'ютеризація суспільства та необхідність поширення та обміну інформацією між науковцями стали причиною формування веб – сайтів та веб – сторінок наукових груп.

Типи веб-сайтів:

- рекламні веб-сайти;
- комерційні веб-сайти;
- веб-сайти альтруїсти;
- веб - сайти для підримки.

Даний підхід дозволяє ділитись результатами досліджень та розширювати коло наукових інтересів учасників навчального процесу. Найвідоміші сайти наукових груп наведено на рисунку 1.5.

					ДП. КСМ. 07139/14.00.00.000 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		18

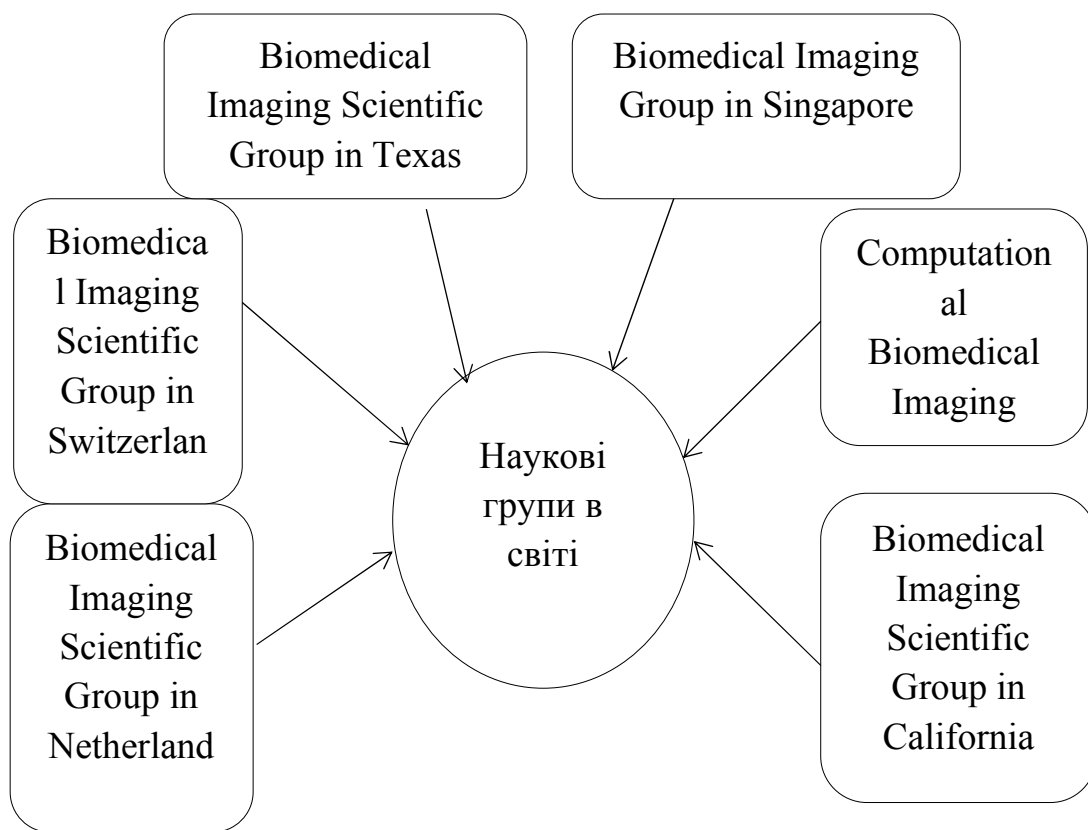


Рисунок 1.5 – Наукові групи

Веб-сторінка— інформаційний ресурс, доступний у мережі World Wide Web, який можна переглянути у веб-браузері [16]. Зазвичай ця інформація записана у форматі HTML або XHTML і може містити гіпертекст із навігаційними гіперпосиланнями на інші веб-сторінки. Веб-сайт— сукупність веб-сторінок, доступних в Інтернеті, які об'єднані як за змістом, так і навігаційно.

Крім класифікації сайтів за характеристиками можна розрізняти веб-сторінки і за складністю. Всього існує два види: статичний або динамічний сайт. Статичним називають сайт з набором постійних, незмінних сторінок. Наприклад, сайт-візитка або сайт-вітрина. Інформація на таких сторінках жорстко закріплена, і зміни можуть здійснюватися тільки фахівцем у веб-технологіях. Такі сайти швидко завантажуються, можуть переглядатися в браузері, мають мінімальні вимоги до веб- сервера. Для компаній такі сайти - бюджетне рішення. Але для підтримки та актуалізації інформації на сторінці все одно будуть потрібні фахівці. Динамічний сайт має безліч переваг. Для функціонування подібних веб-сторінок використовується система CMS (система управління змістом). Це забезпечення, яке

дозволяє оновлювати інформацію, публікувати і змінювати її без залучення фахівців. Досить провести навчання персоналу лише в області порядку роботи з системою.

Фізично сайт може розміщуватися як на одному, так і на кількох серверах. Зовнішній вигляд кожного сайту є унікальним, проте в усіх сайтів можна знайти спільні за функціональністю частини. На будь-якому сайті першою відкривається головна сторінка. У верхній частині головної сторінки зазвичай розташована так звана шапка, яку дублюють на інших сторінках сайту. Дана частина відображається у вікні браузера першою і відвідувач насамперед звертає увагу на неї. Щоб забезпечити швидкий перехід до основних тематичних розділів сайту, створюють меню сайту — список внутрішніх чи зовнішніх посилань на його розділи. Горизонтальне меню зазвичай розташовують у шапці, іноді дублюючи його в нижній частині сторінки, а вертикальне — переважно в лівій частині сторінки, у місці, звідки відвідувач починає її переглядати. Меню є одним із найважливіших компонентів сайту, користувач постійно звертає на нього увагу, і тому вимоги до нього високі. Меню має бути зручним, помітним і зрозумілим, інакше користувач не знатиме, як перейти до потрібного розділу, і покине сайт. Пункти меню повинні бути чітко відділені один від одного. На сайтах із дуже великим обсягом інформації є сторінки третього рівня, а якщо необхідно — то й четвертого, п'ятого тощо. Виділяють три типи структур веб-сайтів — лінійну, деревоподібну та довільну [3]. Подорожуючи сайтом із лінійною структурою, з головної сторінки ви перейдете на другу сторінку, з неї — на третю. На сайті з деревоподібною структурою з головної сторінки можна потрапити на одну зі сторінок другого рівня, звідти — на одну зі сторінок третього рівня тощо. Сайт із довільною структурою видається зовсім неорганізованим, але саме в цьому й полягає принцип його створення. Подорожуючи таким сайтом, ви можете переходити з однієї його сторінки на інші в різні способи, і ваш шлях назад не обов'язково має бути таким самим. Вибір структури визначається особливостями завдань, що розв'язуються за допомогою веб-сайту. Наведемо додатково чотири приклади структур сайту. Кожна із цих структур має свої недоліки та

					ДП. КСМ. 07139/14.00.00.000 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		20

Таблиця 1.2 – Порівняльна таблиця наукових груп

Критерії	Техаський сайт	Швейцарський сайт	Австралійський сайт	Сінгапурський сайт
Змістовність	2	3	4	4
Дизайн	1	3	3	3
Структурованість інформації	+	+/-	+	+
Адаптація під мобільні платформи	+	-	-	-
Форма зворотнього зв'язку	+	-	-	-
Доступ до соціальних мереж	+	-	-	-
Пошук публікацій	-	-	-	-

Отже, до загальних недоліків даних наукових груп можна віднести: відсутність чіткої структури, нагромадженість сторонньої інформації та інформації, що дублюється, відсутність важливої інформації про учасників наукових груп та публікацій.

Також до недоліків можна віднести, що немає форми зворотнього зв'язку, а так як немає цієї форми то відвідувачі сайту не зможуть звернутись до адміністраторів.

1.4 Висновки та постановка задачі

В даному розділі проведено аналітичний огляд підходів до створення веб – сайту, що дозволило обрати найоптимальнішу мову програмування для написання веб – сайту, визначити основні структурні елементи, яким повинен відповідати веб – сайт наукової групи, провести порівняльний аналіз сучасних веб – сайтів наукових груп та виділити їх переваги та недоліки.

Метою даного дипломного проекту є розробка веб – сайту наукової групи аналізу біомедичних зображень кафедри комп’ютерної інженерії ТНЕУ.

Для досягнення поставленої мети потрібно виконати наступні задачі:

- провести огляд підходів до створення веб – сайтів;
- провести порівняльну характеристику сучасних наукових груп;
- порівняти результати роботи розробленого програмного модуля з іншими

відомими системами тестування знань.

В даному розділі було досліджено основні та найбільш популярні засоби для створення веб – сайтів. Зокрема, було проведено огляд сучасних мов програмування та програмних засобів для написання веб – сайту. В результаті проведення дослідження, визначено мову програмування та програмні засоби для написання веб – сайту.

Також було досліджено сучасні наукові групи. Проведено порівняльний аналіз відомих груп. Було проведено огляд сучасних груп за такими критеріями: стилістика, змістовність, доступність інформації.

2. АЛГОРИТМИ РОБОТИ МОДУЛІВ САЙТУ

2.1 Алгоритм пошуку наукових публікацій

Важливим елементом будь - якого сайту наукової групи є можливість пошуку та перегляду інформації про публікації у періодичних наукових виданнях учасників цієї групи. Наявність даного модуля значно підвищує інформативність сайту та забезпечує зручність взаємодії з користувачами. Важливим елементом також є наявність фільтрів за якими можна шукати та сортувати інформацію, наприклад, автори публікацій, рік видання.

Структуру програмних модулів для пошуку інформації по сайту можна представити у такому вигляді (рисунок 2.1).

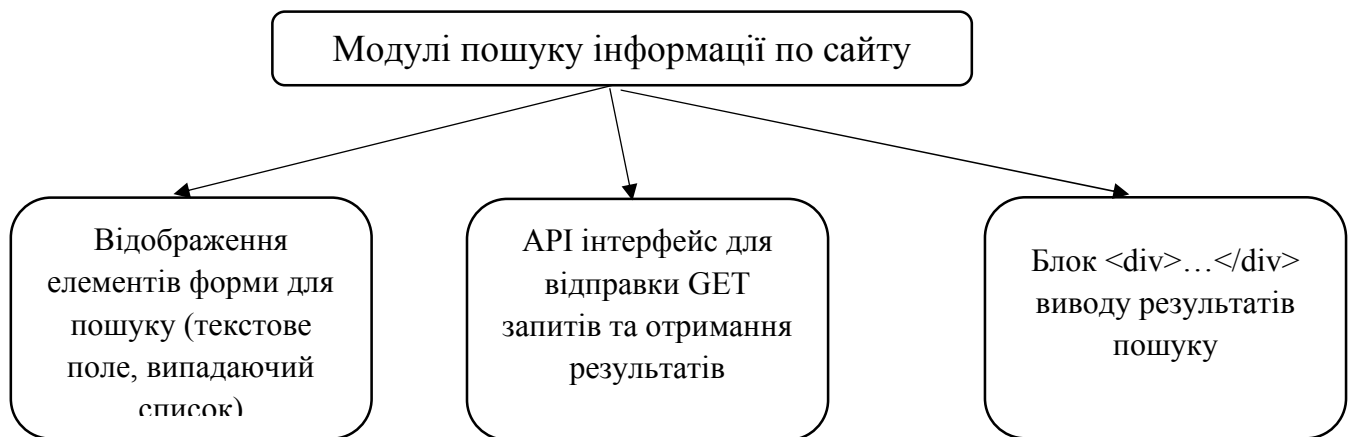


Рисунок 2.1 - Структуру програмних модулів для пошуку інформації по сайту

Алгоритм пошуку публікацій наведено у ДП.КСМ.070139/14.00.00.001 А1. На першому етапі відбування завантаження об'єктної моделі документа DOM сторінки, підключення каскадної таблиці стилів (CSS) та javascript бібліотек. Після цього відбувається завантаження головних скриптів фреймворку Laravel 5.3. Структуру фреймворку Laravel наведено на рисунку 2.2.

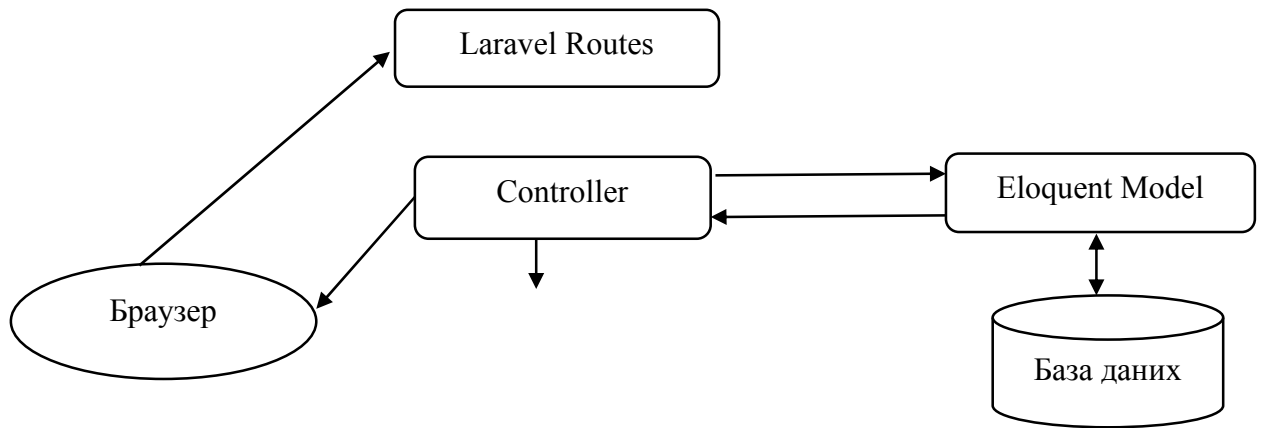


Рисунок 2.2 – Структура фреймворку Laravel 5.3

Обраний фреймворк працює за шаблоном MVC (модель – вигляд – контролер). Laravel Routes відповідає за права доступу до сторінок сайту для різних типів користувачів [19]. Якщо доступ дозволено, то відбувається завантаження контролера сторінки сайту, наприклад IndexController. За роботу із таблицями бази даних та іншими обчисленнями відповідає модель Eloquent Model сторінки. Даний клас має доступ до бази даних mysql та здійснює операції вивантаження, додавання, реагування чи видалення інформації.

Для забезпечення зручного пошуку та мінімізації зайвих дій було обрано бібліотеку Vue.js [20] для розробки real-time додатків. Кожен екземпляр Vue проходить за своїм створенням через послідовність логічних кроків, налаштовує спостереження за даними, компілює шаблон, монтує копію в DOM, а також оновлює DOM при зміні даних. Покроково викликає функції, які називаються хуками життєвого циклу, надаючи користувачам можливість додавати свій власний код на певні етапи.

Після того, як користувач ввів текст для пошуку або за допомогою випадального списку вибрав необхідні поля обробник подій автоматично звертається до функції search() та з допомогою методу GET передає текст для пошуку. У подальшому дана функція звертається до API [21]. API - це метод абстракції між низькорівневим та високорівневим програмним забезпеченням. На рисунку 2.3 наведено функції API.

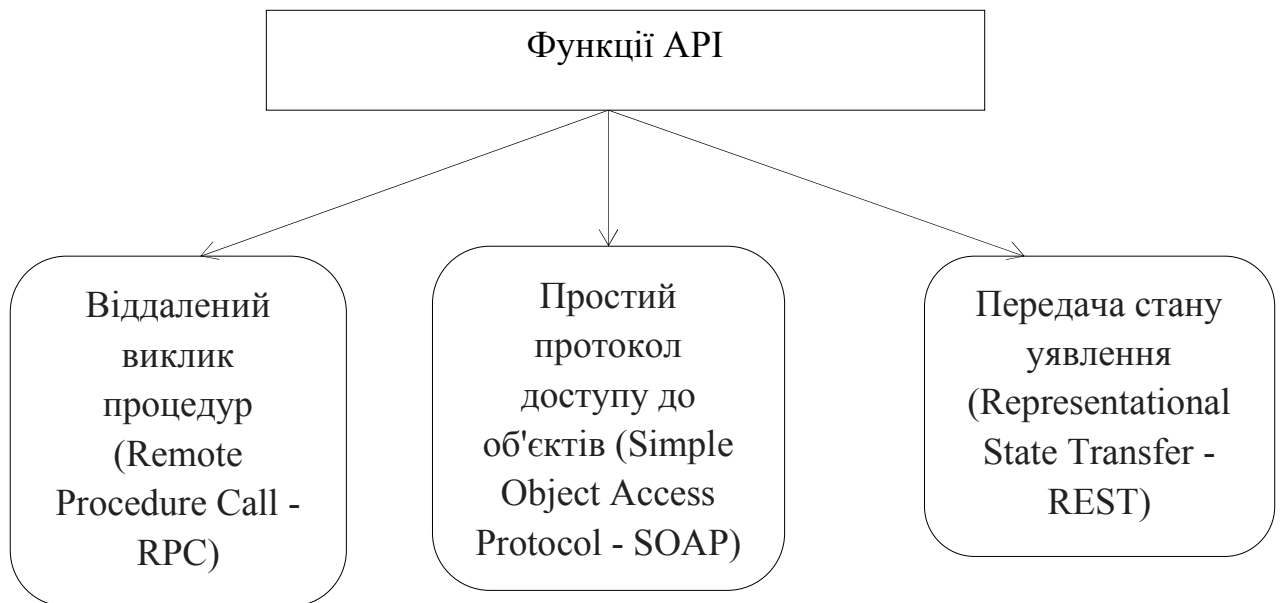


Рисунок 2.3 – Функції API

Одним з найпоширеніших призначень API є надання набору широко використовуваних функцій, наприклад для малювання вікна чи іконок на екрані. Програмісти використовують переваги API у функціональності, таким чином їм не доводиться розробляти все з нуля. API є абстрактним поняттям — програмне забезпечення, що пропонує деякий API, часто називають реалізацією даного API.

Отже, використання API скорочує необхідність створювати самостійно складні програми. Завдяки цій функції можна використовувати готові частини існуючих ресурсів, у яких є доступ до потрібної інформації і даних.

2.2 Алгоритм авторизації в системі

Модуль авторизації є ключовим у більшості динамічних веб – сайтів, тому що дозволяє реалізувати різні режими доступу до ресурсу [22]. На рисунку 3.4 зображено основний принцип аутентифікації.

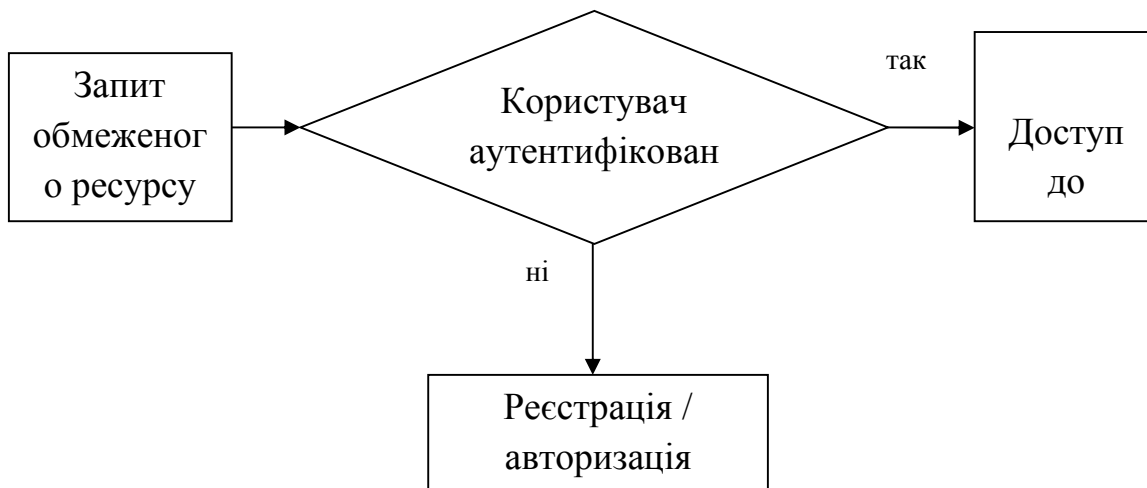


Рисунок 2.4 – Основний принцип аутентифікації

Авторизація – це процес надання особі прав на виконання певних ресурсів (наприклад, перегляд зареєстрованих користувачів) та процес перевірки даних прав при спробі доступу до ресурсу.

Аутентифікація - це процес перевірки легальності доступу користувача до певного ресурсу наприклад, перевірки правильності введеного пароля до для доступу до власного кабінету, перевірка цифрового підпису за ключем шифрування, перевірка контрольної суми файлу [23]. У більшості випадків для підвищення рівня безпеки процеси авторизації та аутентифікації поєднуються. Логін — це набір символів, що ідентифікує користувача веб-сайту [25]. Логін характеризується своєю унікальністю та відповідає певним вимогам щодо кількості та видів символів. Пароль — це секретна послідовність символів, призначена для підтвердження особи або прав на доступ до певного ресурсу. Унікальність символів у паролі обумовлює його складність та відповідно надійність [24]. Найпростішою атакою щодо паролів, відмінною від атаки грубою силою, є їх перебір за словником. Паролі (123, admin) є слабкими і вразливими.

Іноді не потрібно ідентифікувати користувача, а досить лише виконання процедури аутентифікації. Це відбувається у випадку, коли необхідно підтвердити зареєстрованого користувача при здійсненні дій, що вимагають додаткового захисту, наприклад запиту до бази даних [26].

Графічне представлення алгоритму авторизації в системі зображено ДП.КСМ.070139/14.00.00.001 А1.

Алгоритм авторизації користувача відбувається в системі покроково:

Крок 1. Ввід логіну і паролю користувачем на сторінці авторизації. Валідація даних відбувається як на стороні клієнта, так і на стороні сервера. У полі «Логін» користувач повинен ввести електронну адресу з відповідним розділювачем «@» та доменом поштового сервісу, наприклад «ukr.net»;

Крок 2. Якщо дані введені вірно відповідно до правил валідації відбувається пошук введеного поля у базі даних сайту. Якщо логін знайдено, переходимо до кроку 3, в іншому випадку на сторінці авторизації виводиться повідомлення про помилку;

Крок 3. Шифрування введеного користувачем паролю отримується в результаті POST – запиту до серверу. Отриманий пароль шифрується 128 – алгоритм хешування md5. Даний алгоритм призначений для створення «відбитків» повідомлень довільної довжини;

Крок 4. У базі даних паролі зберігаються у зашифрованому вигляді. Введений пароль порівнюється із тим, що зберігається у базі даних. У разі успіху користувач переходить на наступний етап, інакше – вивід повідомлення про помилку на сторінці авторизації;

Крок 5. Із бази даних отримується унікальний ідентифікатор користувача. За даним ідентифікатором користувачу надається відповідний рівень доступу (наприклад, адміністратор). Ідентифікатор, роль, прізвище, ім'я, по-батькові завантажуються у сесійні змінні на стороні сервера та файли – куки на стороні клієнта;

Крок 6. При оновленні сторінки дані на стороні сервера та клієнта порівнюються і відповідно до «таблиці доступу» надається або не надається доступ до ресурсу;

Крок 7. Якщо користувач виходить із свого акаунту, то файли-куки та сесійні змінні знищуються і відбувається перенаправлення на головну сторінку.

Отже, у даному підрозділі на основі результатів дослідження основних

					ДП. КСМ. 07139/14.00.00.000 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		28

принципів аутентифікації та авторизації розроблено алгоритм авторизації сайту наукової групи аналізу біомедичних зображень. В результаті описано послідовність дій користувача під час авторизації в системі та у додатку Б наведено алгоритм авторизації на сайті наукової групи.

2.3 Розробка структури бази даних

База даних — впорядкований набір взаємопов'язаних між собою даних, які використовуються спільно та застосовуються для задоволення потреб користувачів. Головне завдання БД — збереження великих обсягів інформації та надання доступу до неї прикладній програмі або користувачу [27].

Бази даних класифікують за таким ступенем [28]:

- централізована - це яка повністю підтримується на одному комп'ютері;
- розподілена база даних - складові частини якої розміщуються в різних вузлах комп'ютерної мережі;
- неоднорідна – фрагменти якої підтримуються засобами більше однієї СУБД;
- однорідна - фрагменти розподіленої БД в різних вузлах мережі підтримуються засобами однієї і тієї ж СУБД [29];
- фрагментована методом розподілу даних є фрагментованість, вертикальна або горизонтальна;
- тиражова розподіл даних відбувається способом тиражування.

У базі даних основними об'єктами є запити, таблиці, звіти, форми, макроси, модулі та сторінки доступу до даних. Розроблена система передбачає наявність декількох рівнів доступу до системи, тому у базі даних передбачено наявність таблиць для зберігання інформації про користувачів.

Також база даних має 9 типів для зберігання даних, а саме [30]:

- текстовий – використовується для зберігання звичайного тексту;
- числовий – призначений для зберігання дійсних чисел;

					ДП. КСМ. 07139/14.00.00.000 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		29

- дата/час – для зберігання дати та часу;
- грошовий – для зберігання грошових сум;
- логічний – для зберігання логічних даних (типу «так», «ні»);
- поле MEMO – спеціальний тип для зберігання великих обсягів тексту;
- гіперпосилання - спеціальне поле для зберігання адреси URL;
- поле об'єкту OLE – призначений для зберігання об'єктів OLE, наприклад мультимедійних;
- рахівник – використовується для запису чисел.

У ДП.КСМ. 07139/14.00.00.000 А1 наведено датологічну модель розробленої бази даних.

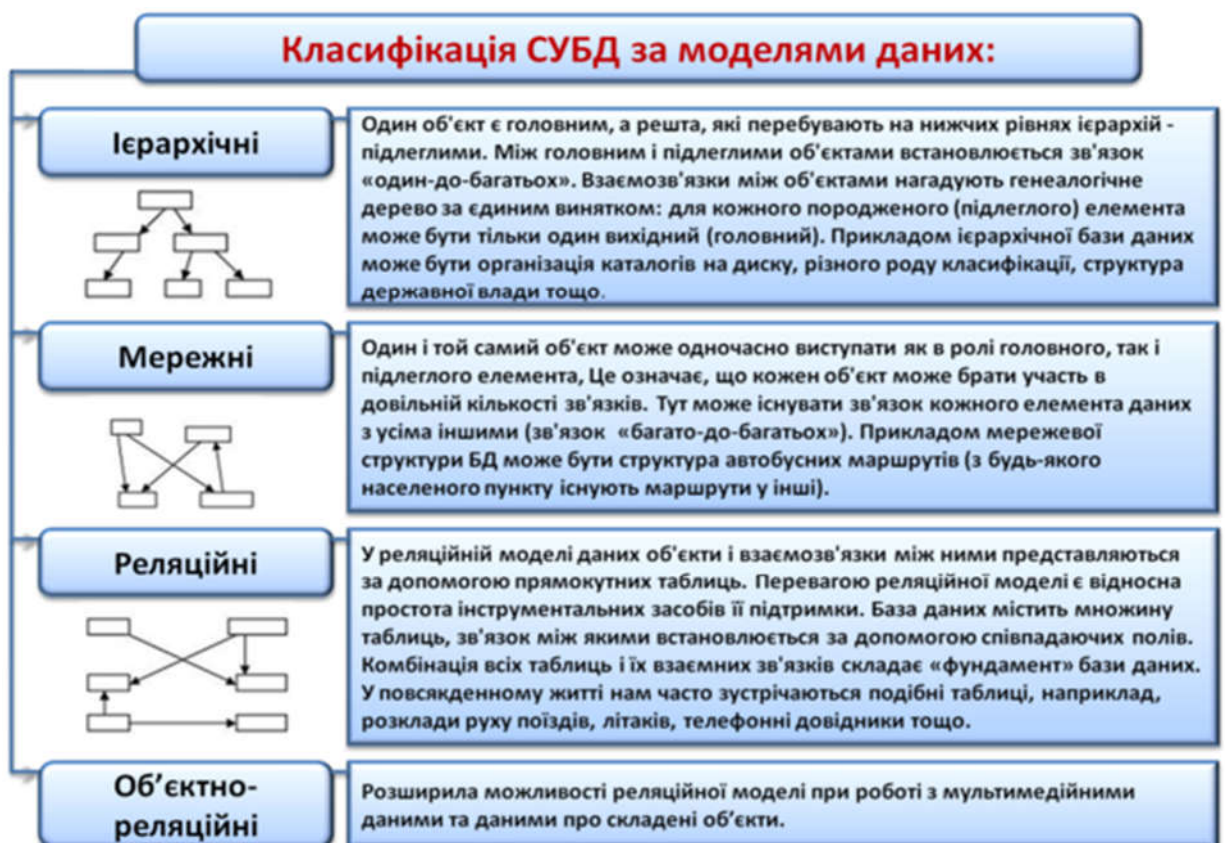


Рисунок 2.5 – Класифікація баз даних за моделлю

Класифікацію баз даних наведено на рисунку 2.6.

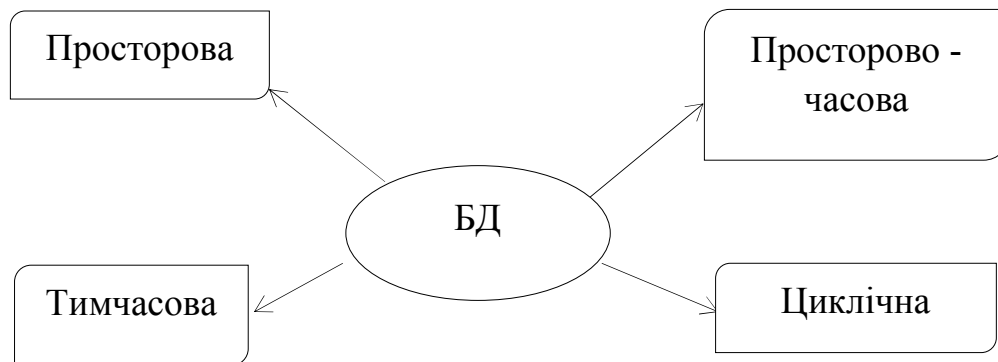


Рисунок 2.6 – Класифікація БД

Просторова база даних підтримує просторові властивості сутностей предметної області. Такі БД широко використовуються в геоінформаційних системах, тому що вона оптимізована для зберігання даних у формі точок, ліній, багатокутників [31]. Недоліком є те, що такі бази даних потребують додаткової функціональності для оброблення просторових типів даних. Тимчасова підтримує будь-який аспект часу, беручи до уваги час, який визначається користувачем. Тимчасові таблиці існують лише під час сеансу роботи користувача. У більшості випадків у тимчасових таблицях зберігаються дані, необхідні для подальших обчислень. Просторово - часова в якій одночасно підтримується одне або більше вимірів в аспектах як простору, так і часу. У циклічних баз даних обсяг збережених даних з часом не змінюється і залишається сталим. Записи користувачем використовуються циклічно. Дану базу даних використовують для зберігання тієї інформації, що перезаписується за певні проміжки часу.

Структуру таблиці «users» наведено у таблиці 2.1. Призначенням даної таблиці є зберігання інформації про користувачів системи.

Таблиця 2.1 - Структура таблиці «users»

Назва поля	Тип даних	Опис
1	2	3
Id	Int(11)	Первинний ключ

Продовження таблиці 2.1

1	2	3
Name	Varchar(255)	Ім'я
fathurname	Varchar(255)	По-батькові
Surname	Varchar(255)	Прізвище
Login	Varchar(100)	Логін
Password	Varchar(100)	Пароль
Email	Varchar(100)	Адреса електронної пошти
Auth_key	Varchar(100)	Ключ авторизації
Created_at	TIMESTAMP	Дата та час створення запису
Updated_at	TIMESTAMP	Дата та час редагування запису

Інколи виникає необхідність надання декількох прав доступу одному користувачу, наприклад адміністратор може бути одночасно адмініструвати систему та редагувати тексти статей сайту. Структуру таблиці «role_user» наведено у таблиці 2.2.

Таблиця 2.2 - Структура таблиці «role_user»

Назва поля	Тип даних	Опис
User_id	Int (11)	Унікальний ідентифікатор користувача
Role_id	Int (11)	Ідентифікатор ролі, наприклад адміністратора

Структуру таблиці «roles» наведено у таблиці 2.3. Дана таблиця, окрім стандартних полів вміщує інформацію про назву ролей (наприклад, адміністратор) та опис ролі.

Таблиця 2.3 - Структуру таблиці «roles»

Назва поля	Тип даних	Опис
1	2	3
Id	Int(11)	Первинний ключ

Таблиця 2.5 - Структура таблиці «publications»

Назва поля	Тип даних	Опис
Id	Int(11)	Первинний ключ
Title	Varchar(100)	Бібліографічний опис публікації
Href	Varchar(50)	Посилання на зовнішній ресурс, для отримання доступу до публікації повністю або частково
Year	Date	Дата опублікування наукової роботи
Article_id	Int(11)	Ідентифікатор статті у якій використовуються результати, наведені в публікації
Created_at	TIMESTAMP	Дата та час створення запису
Updated_at	TIMESTAMP	Дата та час редагування запису

Отже, у даному підрозділі на основі дослідження основних етапів розробки баз даних було розроблено структуру бази даних для функціонування сайту наукової групи та описано структуру п'яти таблиць, з яких складається база даних. В результаті у ДП.КСМ.07139/14.00.00.000 С1 наведено структуру розроблювальної бази даних.

3 ПРОГРАМНА РЕАЛІЗАЦІЯ ВЕБ – САЙТУ НАУКОВОЇ ГРУПИ

3.1 Структура програмного модуля

3.1.1 Серверна частина

Більшість інформації обробляється на серверній частині веб – сайту, зокрема вибірка інформації з бази даних, пошук по базі даних за ключовими словами, вибірка інформації із випадального списку, збереження записів користувачів із форми зворотного зв'язку тощо [32]. Структуру програмної частини розроблено сайту наведено на рисунку 3.1.

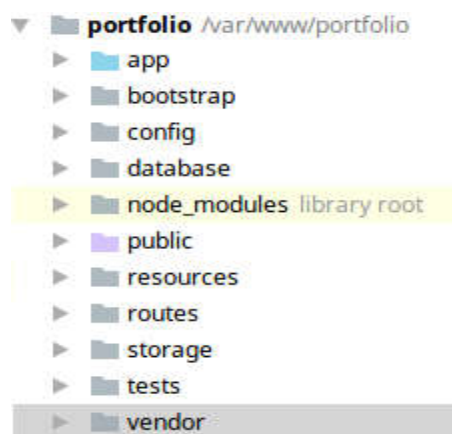


Рисунок 3.1 – Структура програмної частини розроблено сайту

У директорії «app» знаходяться контролери та моделі модулів сайту наукової групи. У директорії «config» знаходяться файли конфігурації: файл налаштування доступу до бази даних, сесії. У директорії «database» знаходяться файли дампу таблиць бази даних, наприклад «publications.sql». Директорії «public» є кореневою. У ній зберігається головний файл «index.php» та зображення у форматах «.png», «.jpeg», «.bmp». Директорія «resources» вміщує файли з html – версткою, css – файлами стилів, js – файлами для візуальних ефектів на клієнтській частині.

Універсальна мова моделювання (Unified Modelling Language або UML) — це мова позначень або побудови діаграм, призначена для визначення, візуалізації і

документування моделей зорієнтованих на об'єкти систем програмного забезпечення [33]. UML не є методом розробки, у конструкціях цієї мови не повідомляється про те, що робити першим, а що останнім, і не надається інструкцій щодо побудови вашої системи, але ця мова допомагає вам наочно переглядати компонування системи і полегшує співпрацю з іншими її розробниками. Конструкції UML створюються з багатьох модельних елементів, які позначають різні частини системи програмного забезпечення [34]. Елементи UML використовуються для побудови діаграм, які відповідають певній частині системи.

Існують такі типи діаграм:

- діаграма випадків використання;
- діаграми класів;
- діаграми послідовності;
- діаграми стану;
- діаграми компонентів та інші.

У дипломній роботі я використовую діаграми класів [35]. На діаграмах класів зображуються різноманітні класи, які утворюють систему і їх взаємозв'язки. Діаграми класів називають «статичними діаграмами», оскільки на них показано класи разом з методами і атрибутами, а також статичний взаємозв'язок між ними: те, яким класам відомо про існування яких класів, і те, які класи є частиною інших класів, — але не показано методи, які при цьому викликаються.

Є декілька способів побудови діаграм в залежності від того, яким чином вони будуть використовуватися.

1. Концептуальна. В даному випадку діаграма класів UML здійснює опис моделі певної предметної області, і в ній передбачаються тільки класи прикладних об'єктів.

2. Специфічна. Діаграма використовується в процесі проектування різних інформаційних систем.

3. Реалізаційна. Діаграма класів включає в себе класи, які використовуються у програмному коді.

UML діаграму класів – контролерів наведено на рисунку 3.2.

					ДП. КСМ. 07139/14.00.00.000 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		36

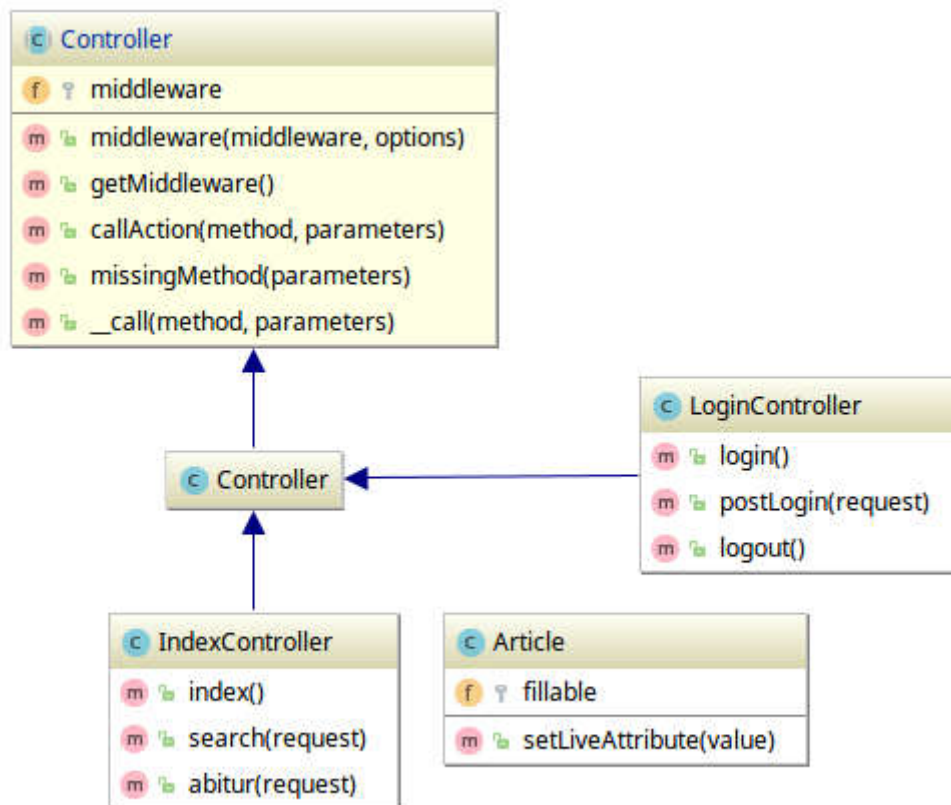


Рисунок 3.2 – UML діаграма класів – контролерів

Головний клас «IndexController» вміщує два основні методи для роботи. Метод «index()» здійснює вибірку із бази даних інформацію про публікації, учасників, новини наукової групи. Метод «search(request)» призначений для здійснення пошуку за ключовим словом. Параметр request приймає значення у стрічковій формі. Також здійснюється опис форматів як XML та JSON [36].

XML – призначений для збереження і передачі даних. Стандарт XML визначає набір базових лексичних та синтаксичних правил для побудови мови описання інформації шляхом застосування простих тегів. [37] XML вирішує задачу збереження і транспортує дані фокусуючись на тому, що таке ці самі дані. Формат JSON – це простий формат обміну даними, зручний для читання і написання [38]. Він заснований на підмножині мови програмування JavaScript. За допомогою JSON можна пов'язувати між собою масиви і об'єкти, створюючи складні структури даних. Найчастіше поширене використання JSON – пересилання від сервера до браузера.

Зазвичай дані JSON доставляються за допомогою AJAX, який дозволяє обмінюватися даними браузеру і серверу без необхідності перезавантажувати сторінку. На відміну від XML, пропозиції JSON легко складаються як людиною, так і комп'ютером. Дані в JSON форматі можна, привласнити змінній (цим ми створюємо об'єкт) потім ці дані можна прочитати за допомогою новоствореного об'єкта [39].

Клас «LoginController» вміщує методи, призначені для виводу форми авторизації в системі для адміністрації та обробки введених даних. Laravel - PHP фреймворк призначений для розробки веб – додатків. Особливість даного фреймворку є модульна система яка відповідає за права доступу до сторінок сайту для різних типів користувачів. Laravel Routes працює за шаблоном MVC (модель – вигляд - контролер). Такий підхід дозволяє розділити системи на такі три частини:

- модуль керування;
- модуль вигляду (верстка);
- модуль моделей даних.

У таблиці 3.1 наведено перелік системних вимог до сервера.

Таблиця 3.1 – Системні вимоги до сервера

Найменування	Параметри
ЦП	2xXeon 3.0 ГГц
ОЗП	2 Гб
Об'єм жорсткого диску	145 Гб
Операційна система	Windows XP/2000/2003 Linux/FreeBSD
Веб-сервер	Apache 1.3/2.0/2.2 із включеним mod_rewrite
Інтерпретатор PHP	PHP версії 5.6 і вище
Фреймворк	Laravel 5.3
База даних	mysql

Використання стабільної версії фреймворку Laravel 5.3 та реляційної бази даних mysql дозволяє забезпечити стабільність роботи розробленої системи.

3.1.2 Клієнтська частина

Основним призначенням інтерфейсу користувача є обробка та відображення інформації максимально зручно для клієнта. У графічних системах інтерфейсу користувача реалізовується багатовіконним режимом, змінами кольору, розміру, видимості вікон, їхнім розташуванням, сортуванням елементів вікон. Інтерфейс відіграє важливу роль у взаємодії користувача з системою, адже він може прискорювати час прийняття рішень та покращувати або погіршувати якість роботи.

Графічний інтерфейс головного вікна сайту наведено на рисунку 3.2. Титульне зображення складається з набору додаткових зображень, що асоціюються із предметом досліджень наукової групи аналізу біомедичних зображень.



Рисунок 3.2 – Інтерфейс головного вікна сайту наукової групи

У графічному інтерфейсі який зображено на рисунку 3.2 знаходяться посилання на блоки пошуку учасників та публікацій, соціальні мережі, статті, топ публікації також присутня форма зворотного зв'язку.

Інтерфейс головного вікна є простим та зрозумілим у користуванні. Робота з системою не викликає у користувача ускладнень у пошуках необхідних елементів

					ДП. КСМ. 07139/14.00.00.000 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		39

інтерфейсу. До того ж, передбачено введення користувачем тільки мінімальної інформації. У головному вікні є категорія підказок, користувач клікнувши на неї отримує інформацію з блоками. Натиснувши на потрібний блок його перекидає на бажану сторінку.

Графічний інтерфейс блоку пошуку публікацій наведено на рисунку 3.3. Фільтрація пошуку відбувається на такими полями «Автори», «Рік публікації».

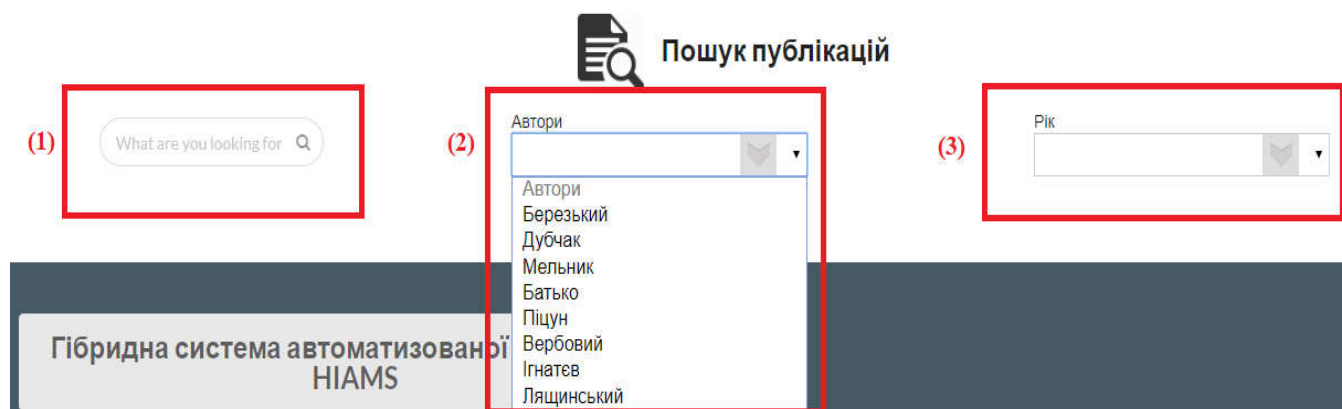


Рисунок 3.3 – Графічний інтерфейс блоку пошуку публікацій

У (1) знаходиться текстове поле для вводу текстової інформації. Пошук відбувається по усіх полях бази даних. У робочій зоні (2) знаходиться випадаючий список з прізвищами учасників наукової групи. Після вибору прізвища учасника з бази даних відбувається вивід інформації у нижньому блоці. Блок (3) здійснює фільтрацію запиту про роках випуску публікацій.

Графічний інтерфейс модулю «Статті» зображено на рисунку 3.4. Модуль статей поділено на три рівні блоки, що дозволяють забезпечити зрозумілість текстової і графічної інформації.

У (1) наведено блок статті із такими складовими: титульне зображення, дата публікації статті (2), назва статті (3).



Рисунок 3.4 - Графічний інтерфейс блоку «Статті»

Графічний інтерфейс блоку учасники групи наведено на рисунку 3.5.

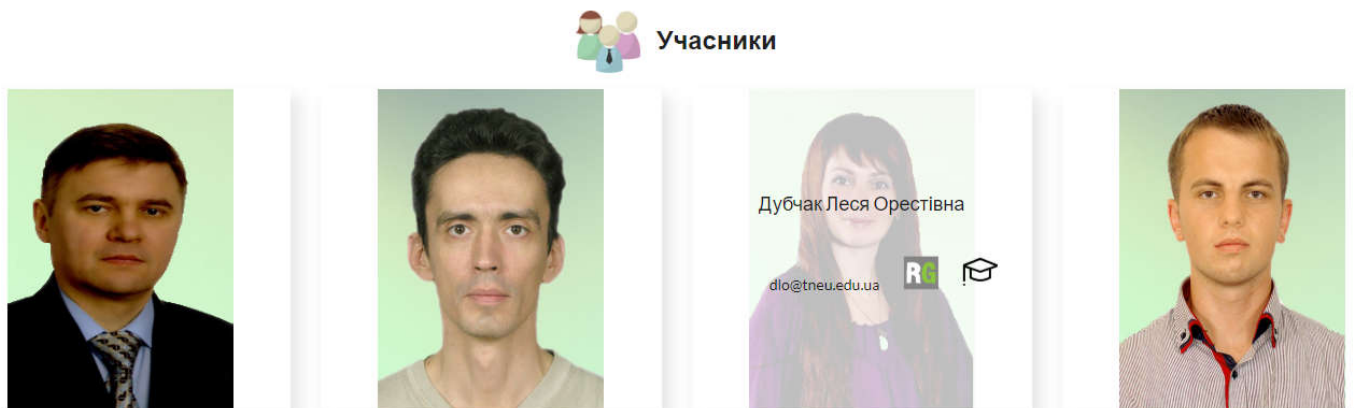


Рисунок 3.5 - Графічний інтерфейс блоку учасників наукової групи

При наведенні вказівником миші на зображення учасника відображається інформація про прізвище, ім'я та по-батькові, електронну пошту та посилання на профіль у researchgate, google scholar, github.

3.2 Unit – тестування

Модульне тестування - метод тестування програмного забезпечення, який полягає в окремому тестуванні кожного модуля коду програми [40]. Модулем називають найменшу частину програми, яка може бути протестованою. Зазвичай

unit-тести застосовують для того, щоб упевнитися, що код відповідає вимогам архітектури та має очікувану поведінку. Модульне тестування проводиться викликаючи код, який необхідно перевірити за підтримки середовищ розробки [41]. Всі знайдені дефекти, як правило виправляються в коді без формального їх опису в системі багів (Bug Tracking System). Один з найбільш ефективних підходів до модульного тестування – це підготовка автоматизованих тестів до початку основного кодування програмного забезпечення. Це називається розробка від тестування (test-driven development) або підхід тестування спочатку (test first approach). При цьому підході створюються і інтегруються невеликі шматки коду, навпроти яких запускаються тести, написані до початку кодування. Розробка ведеться до тих пір поки всі тести не будуть успішними.

Код методу для тестування статусу відклику від сервера такий:

```
public function testResponseStatus () {  
    $this->call('GET', 'http://ai-tern.in.ua/');  
    $this->assertResponseStatus(200);  
}
```

Код стану HTTP — частина першого рядка відповіді сервера при запитах по протоколу HTTP. Він являє собою ціле число з трьох арабських цифр. Перша цифра вказує на клас стану. За кодом відповіді зазвичай йде відділена пробілом пояснювальна фраза на англійській мові, яка пояснює людині причину саме такої відповіді. По коду відповіді клієнт дізнається про результати його запиту і визначає, що йому робити далі [41].

Результат роботи unit-тесту в середовищі PHPStorm наведено на рисунку 3.6.

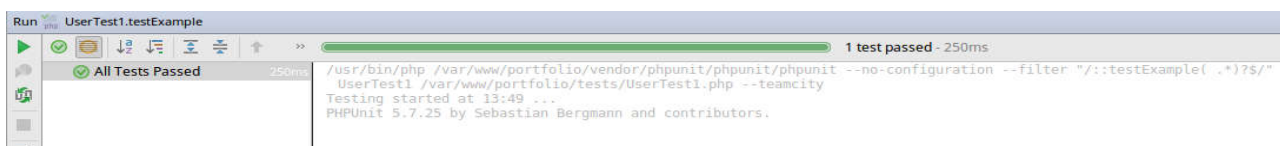


Рисунок 3.6 – Результат тестування методу testResponseStatus()

Код тесту для перевірки коректності авторизації на сайті такий:

```
$this->visit('/login');  
    $this->type(test@gmail.com, 'email');  
    $this->press('Увійти');  
    $this->seePageIs('/profile/test1');
```

`$this->visit()` – URL – адреса сторінки авторизації. `$this->press()` – назва кнопки для підтвердження введених даних, `$this->seePageIs()` - URL – адреса очікуваної сторінки у випадку успішної авторизації.

Код методу тесту для перевірки переадресації такий:

```
$this->assertRedirectedTo('login');  
$this->assertRedirectedToRoute('route.name');  
$this->assertRedirectedToAction('LoginController@login');
```

Код методу тесту для перевірки коректності переходу за внутрішніми посиланнями такий:

```
$this->visit('/')  
    ->click('Детальніше')  
    ->seePageIs('/HIAMS.html');
```

В результаті активації кнопки «Детальніше» користувач переходить на нову сторінку «/HIAMS.html». Нова сторінка підвантажує існуючі файли стилів та JS файли, що дозволяє зробити інтерфейс сторінки одноманітним та впорядкованим. На сторінці «/HIAMS.html» знаходиться детальна інформація про програмні розробки наукової групи.

Розроблені тести дозволяють пересвідчитись у працездатності сайту в автоматичному режимі.

					ДП. КСМ. 07139/14.00.00.000 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		43

3.3 Порівняльний аналіз розробленого сайту з аналогами

Сучасні серверні технології, розподілені бази даних, засоби боротьби із DDOS атаками на хостингах забезпечують високу продуктивність роботи веб – ресурсів, особливо односторінкових сайтів. Враховуючи значну популярність використання мобільних пристроїв для виходу в мережу інтернет, ключовим критерієм порівняння є адаптивність графічного інтерфейсу до дисплеїв з різними розширеннями. Порівняльний аналіз веб – сайтів [42] наукових груп наведено у таблиці 3.2 (+ наявність елементу, - відсутність елементу, 1-5 діапазон оцінювання).

Таблиця 3.2 – Порівняльний аналіз сайтів наукових груп

Критерії	Сайт наукової групи Техаського університету (США)	Biomedical Imaging Group університету École Polytechnique Fédérale De Lausanne (Швейцарія)	сайт наукової групи Австралійського університету	Розроблений веб - сайт
1	2	3	4	5
Змістовність (кількість інформації)	2	3	3	5
Дизайн	1	3	3	4
Структурованість інформації	+	-	-	+
Адаптація під мобільні платформи	+	-	-	+
Форма зворотнього зв'язку	+	-	-	+
Доступ до соціальних мереж	-	-	-	+
Пошук публікацій	-	-	-	+

Продовження таблиці 3.2

1	2	3	4	5
Пошук по базі даних	-	-	-	+
Пошук з випадального списку	-	-	-	+

Як видно з таблиці 3.2 розроблений веб – сайт наукової групи «Аналіз біомедичних зображень» ТНЕУ володіє розширеним функціоналом для роботи з відвідувачами у порівнянні з існуючими аналогами.

Зовнішній вигляд розробленого сайту на моніторах з різними розширеннями наведено на рисунку 3.7.



а) 320 x 480



б) 375 x 665



в) 1280 x 720

Рисунок 3.7 – Зовнішній вигляд розробленого сайту на моніторах з різними розширеннями

Адаптація під мобільні платформи значно покращує індексацію сторінок пошуковими системами, що дозволяє виводити їх у топ результатів запити [43].

В результаті аналізу можна зробити висновок, що розроблений сайт є адаптивним до різних дисплеїв з різним розширенням. Можливість пошуку інформації та застосування спеціальних фільтрів дозволяє покращити взаємодію з користувачем сайту та відповідно збільшити аудиторію активних користувачів: студентів, науковців.

Отже, у даному розділі представлено структуру клієнтської та серверної частини веб – сайту наукової групи «Аналіз біомедичних зображень». Здійснено розробку unit – тестів для тестування критичних ділянок коду в автоматичному режимі. Здійснено порівняльний аналіз розробленого сайту із існуючими аналогами, що дозволило виділити переваги та недоліки кожного сайту. До переваг розробленого сайту можна віднести: адаптивність до дисплеїв із різним розширенням, наявність модулів пошуку та фільтрації інформації про публікації в режимі реального часу.

					ДП. КСМ. 07139/14.00.00.000 ПЗ	Арк.
						46
Змн.	Арк.	№ докум.	Підпис	Дата		

4 ТЕХНІКО – ЕКОНОМІЧНЕ ОБГРУНТУВАННЯ

4.1 Розрахунок витрат на виконання проектного рішення

4.1.1 Стадії технологічного процесу

У розробці проектного рішення задіяні наступні спеціалісти - розробники, а саме: керівник проекту (К); студент-дипломник (С); консультант техніко-економічного розділу (КТЕО).

Форму поділу робіт по всіх основних етапах і видах робіт, які повинні бути виконані показано в таблиці 4.1.

Таблиця 4.1 - Середній час виконання проекту та стадії технологічного процесу

№ п/п	Назва операції (стадії)	Виконавець, посада	Середній час виконання операції, год.
1.	Підготовка	Студент	7
2	Розробка проекту системи	Керівник ДП	16
		Консультант ТЕО, доцент	2
		Студент	216
3	Проектування технічної частини системи	Студент	21
4	Розробка програмного продукту системи	Студент	6
5	Встановлення та налаштування прогр. зас.	Студент	8
6	Тестування системи	Студент	3
Разом			279

4.1.2 Визначення витрат на оплату праці та відрахувань на соціальні заходи.

Витрати на оплату праці включають заробітну плату (ЗП) всіх категорій працівників, безпосередньо зайнятих на всіх етапах проектування.

Витрати на оплату праці розробників проекту визначаються за формулою:

$$B_{ОП} = \sum_{i=1}^N \sum_{j=1}^M n_{ij} \cdot t_{ij} \cdot C_{ij}, \quad (4.1)$$

де n_{ij} – чисельність розробників i -ої спеціальності j -го тарифного розряду, осіб;

t_{ij} – затрачений час на розробку проекту співробітником i -ої спеціальності j -го тарифного розряду, год;

C_{ij} – годинна ставка працівника i -ої спеціальності j -го тарифного розряду, грн.

Середньо годинна ставка працівника може бути розрахована за формулою:

$$C_{ij} = \frac{C_{ij}^0(1+h)}{PЧ_i}, \quad (4.2)$$

де C_{ij} – основна місячна заробітна плата розробника i -ої спеціальності j -го тарифного розряду, грн.;

h – коефіцієнт, що визначає розмір додаткової заробітної плати (при умові наявності доплат);

$PЧ_i$ - місячний фонд робочого часу працівника i -ої спеціальності j -го тарифного розряду, год. (приймаємо 168 год.).

Таблиця 4.2 - Вихідні дані для розрахунку витрат на оплату праці

№ п/п	Посада виконавців	Місячний оклад (стипендія), грн.	Коефіцієнт Додаткової з/п	Підсумок
1	Керівник ДП, викладач	2458	0,4	3440
2	Консультант техніко-економічного розділу, доцент	6026	1,39	14402
3	Студент	1680	0	1680

Звідси, загальні витрати на оплату праці ($B_{оп}$) дорівнюють:

$$B_{оп} = 16 * \frac{3440}{168} + 2 * \frac{14402}{168} + 254 * \frac{1680}{168} = 3048 (\text{грн})$$

Крім того, слід визначити відрахування на соціальні заходи. Величну відрахувань у спеціальні державні фонди визначають у відсотковому співвідношенні від суми основної та додаткової заробітних плат. Згідно діючого нормативного законодавства сума відрахувань у спеціальні державні фонди складає 20,5 % від суми заробітної плати:

$$B_{\phi} = \frac{20.5}{100} * 3048 = 624,84 (\text{грн.})$$

4.1.3 Розрахунок матеріальних витрат

Матеріальні витрати — це вартість витрачених матеріалів, малоцінних та швидкозношуваних предметів на виробництво продукції, робіт або послуг, а також матеріалів і МШП, витрачених на адміністративні, збутові та інші потреби підприємства.

Загальна сума витрат на матеріальні ресурси (B_M) визначається за формулою:

$$B_M = \sum_{i=1}^n K_i \cdot C_i, \quad (4.3)$$

де K_i - витрата i -го типу матеріалу, натуральні одиниці вимірювання;

C_i - ціна за одиницю i -го типу матеріалу, грн.;

i - тип матеріального ресурсу;

n - кількість типів матеріальних ресурсів.

Звідси, витрати на матеріальні ресурси дорівнюватимуть:

$$B_M = 401,6 \text{ грн.}$$

					ДП. КСМ. 07139/14.00.00.000 ПЗ	Арк.
						49
Змн.	Арк.	№ докум.	Підпис	Дата		

i - тип електрообладнання;

n - кількість електрообладнання.

Для розробки проекту даної системи використовується один ноутбук потужністю $P = 0,5$ кВт, який за весь період розробки працює 200 годин та друкуючий пристрій потужністю $P = 0,37$ кВт, який працює 2 години.

Проміжні розрахунки на витрату електроенергії подані в таблиці 4.4.

Таблиця 4.4 - Витрати на електроенергію

Найменування устаткування	Паспортна потужність, кВт	Коефіцієнт використання потужності	Час роботи обладнання для розробки, год	Ціна електроенергії,	Сума, грн.
Ноутбук	0,5	0,98	200	0,90	88,2
Принтер	0,37	0,98	2	0,90	0,65
Разом					88,85

4.1.5 Розрахунок суми амортизаційних відрахувань

Амортизація – це процес перенесення вартості основних фондів на вартість новоствореної продукції з метою їх повного відновлення.

Для визначення амортизаційних відрахувань застосуємо метод прямолінійного списання. Загальна сума амортизаційних відрахувань (B_{AM}) визначається за формулою:

$$B_{AM} = \sum_{i=1}^n \frac{B_i \cdot H_i}{100}, \quad (4.5)$$

де B_i - вартість i -го устаткування на початок звітного періоду, грн.;

H_i - річна норма амортизації i -го устаткування, %;

i - тип обладнання;

		n - кількість устаткування.							Арк.
									51
Змн.	Арк.	№ докум.	Підпис	Дата	ДП. КСМ. 07139/14.00.00.000 ПЗ				

Для проектування даної системи використовувався один ноутбук 5000грн., та принтер вартістю 1000 грн, а також інші пристрої описані вище, та подані таблиці 1.5 і обрахування витрат на амортизацію.

Тоді:

$$B_{AM} = \frac{500 \cdot 10}{100} + \frac{1000 \cdot 20}{100} = 700(\text{грн})$$

Таблиця 4.5 - Амортизація основних фондів

Найменування устаткування	Вартість устаткування, грн.	Річна норма амортизації, %	Сума, грн.
Ноутбук	5000,00	10	500,00
Принтер	1000,00	20	200,00
Разом			700,00

4.1.6 Визначення транспортних витрат

Транспортні витрати слід прогнозувати у розмірі 8–12 % від загальної суми матеріальних витрат.

$$B_T = 0.12 \cdot B_M, \quad (4.6)$$

де B_T – транспортні витрати.

$$B_T = 0,12 \cdot 401,6 = 48,192 \text{ грн.}$$

4.1.7 Обчислення накладних витрат

Накладні витрати пов'язані з обслуговуванням виробництва, утриманням апарату управління підприємства (фірми) та створення необхідних умов праці можуть становити до 150 % від суми основної та додаткової заробітної плати працівників. Накладні витрати для даного проекту подані далі.

					ДП. КСМ. 07139/14.00.00.000 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		52

$$H_B = 1,5 \cdot B_{ОП}, \quad (4.7)$$

де H_B – накладні витрати.

$$H_B = 1,5 * 3048 = 4572 \text{ грн.}$$

4.1.8 Обчислення інших витрат

Інші витрати є витратами, які не враховані в попередніх статтях. Вони становлять 10% від заробітної плати:

$$I = 3048 * 0.1 = 304,8 \text{ грн}$$

4.1.9 Складання кошторису витрат та визначення собівартості

Загальні витрати ($B_{КС}$) розрахуємо за формулою:

$$B_{КС} = B_{ОП} + B_{\phi} + B_M + B_E + B_{AM} + B_T + H_B \quad (4.8)$$

Тобто:

$$B_{КС} = 9788,28 \text{ грн.}$$

Результати проведених розрахунків зведемо у таблицю 4.6.

Таблиця 4.6 - Кошторис витрат

Зміст витрат	Сума, грн.
1	2
Витрати на оплату праці (осн. і дод. ЗП)	3048

Продовження таблиці 4.6

1	2
Відрахування на соціальні заходи	624,84
Матеріальні витрати	401,6
Витрати на електроенергію	88,85
Амортизаційні відрахування	700
Транспортні витрати	48,192
Накладні витрати	4572
Інші витрати	304,8
Разом	9788,28

4.2 Визначення експлуатаційних витрат

Для оцінки економічної ефективності розроблюваного програмного продукту слід порівняти його з аналогом, тобто існуючим програмним забезпеченням ідентичного функціонального призначення.

Експлуатаційні одноразові витрати по програмному забезпеченню і аналогу включають вартість підготовки даних і вартість роботи комп'ютера (за час дії програми):

$$E_n = E_{1n} + E_{2n}, \quad (4.9)$$

де E_n - одноразові експлуатаційні витрати на ПЗ (аналог), грн.;

E_{1n} - вартість підготовки даних для експлуатації ПЗ (аналог), грн.;

E_{2n} - вартість роботи комп'ютера для виконання проектного рішення (аналог).

Річні експлуатаційні витрати B_{en} визначаються за формулою:

$$B_{en} = E_n * N_n, \quad (4.10)$$

де N_n - періодичність експлуатації ПЗ (аналогу), раз/рік.

Вартість підготовки даних для роботи на комп'ютері визначається за формулою:

$$E_{ln} = \sum_{i=1}^n n_i t_i c_i, \quad (4.11)$$

де i - категорії працівників, які приймають участь у підготовці даних ($i=1,2,\dots,n$);

n_i - кількість працівників i -ої категорії, осіб;

t_i - трудомісткість роботи співробітників i -ої категорії по підготовці даних, год.;

c_i - середньо годинна ставка працівника i -ої категорії з врахуванням додаткової заробітної плати, що знаходиться із співвідношення:

$$c_i = \frac{c_i^0(1+b)}{m}, \quad (4.12)$$

де c_i^0 - основна місячна заробітна плата працівника i -ої категорії, грн.;

b - коефіцієнт, який враховує додаткову заробітну плату (прийmemo 0,57);

m - кількість робочих годин у місяці, год.

Для роботи з даними як для проектного рішення так і аналогу потрібен один працівник, основна місячна заробітна плата якого складає: $c^0 = 3200$ грн.

Тоді:

$$c_1 = \frac{3200(1+0,57)}{22*8} = 28,54(\text{грн/год})$$

Трудомісткість підготовки даних для проектного рішення складає 1 год., для аналога 1,5 год.

					ДП. КСМ. 07139/14.00.00.000 ПЗ	Арк.
						55
Змн.	Арк.	№ докум.	Підпис	Дата		

Таблиця 4.7 - Розрахунок витрат на підготовку даних та реалізацію проектного рішення на комп'ютері

№	Час роботи співробітників, год.	Середньогодинна заробітна плата, грн./год.	Витрати , грн.
Проектне рішення			
1	1	28,54	28,54
Аналог			
1	1,5	28,5	42,81

Витрати на експлуатацію комп'ютера визначається за формулою:

$$E_{2n} = t * S_{MG}, \quad (4.13)$$

де t - витрати машинного часу для реалізації проектного рішення (аналогу), год.;

S_{MG} - вартість однієї години роботи комп'ютера, грн./год.

$$E_{2n} = 1 * 0,9 = 0,9 \text{ грн.}; E_{2a} = 1,5 * 0,9 = 13,5 \text{ грн.}$$

$$E_n = 28,54 + 0,9 = 29,44 \text{ грн.}; E_a = 42,81 + 0,9 = 43,71 \text{ грн.}$$

$$B_{en} = 29,44 * 252 = 7418,88 \text{ грн.}; B_{ea} = 43,71 * 252 = 11014,92 \text{ грн.}$$

4.3 Розрахунок ціни споживання проектного рішення

Ціна споживання - це витрати на придбання і експлуатацію проектного рішення за весь строк його служби:

$$Ц_{C(I)} = Ц_{П} + B_{(E)NPV}, \quad (4.14)$$

					ДП. КСМ. 07139/14.00.00.000 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		56

де C_n - ціна придбання проектного рішення, грн.:

$$C_n = K \left(1 + \frac{P_p}{100}\right) + K_0 + K_k, \quad (4.15)$$

де K - кошторисна вартість;

P_p - рентабельність;

K_0 - витрати на прив'язку та освоєння проектного рішення на конкретному об'єкті, грн.;

K_k - витрати на доукомплектування технічних засобів на об'єкті, грн.

Договірна ціна (C_d) для проектних рішень розраховується за формулою:

$$C_d = B_{KC} * \left(1 + \frac{p}{100}\right), \quad (4.16)$$

де B_{KC} – кошторисна вартість, грн.;

p - середній рівень рентабельності, (приймаємо 0,3% за погодженням з керівником).

$$C_d = 9788,27 \cdot (1 + 0,3) = 12724,751 \text{ грн.}$$

Вартість витрат на експлуатацію проектного рішення (за весь час його експлуатації), грн.:

$$B_{enpv} = \sum_{t=1}^T \frac{B_{en}}{(1 + R)^t}, \quad (4.17)$$

де B_{en} - річні експлуатаційні витрати, грн.;

T - строк служби проектного рішення, років;

R - річна ставка проценту банку.

$$B_{enpv} = \sum_{\tau=1}^5 \frac{7418,88}{(1 + 0,08)^\tau} = 35628,3 \text{ (грн.)}$$

					ДП. КСМ. 07139/14.00.00.000 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		57

$$B_{еплр} = \sum_{t=1}^5 \frac{11014,92}{(1 + 0,08)^t} = 48275,2 \text{ (грн.)}$$

Тоді ціна споживання проектного рішення дорівнюватиме:

$$Ц_{сн} = 12\,724,751 + 35628,3 = 48353,051 \text{ (грн.)}$$

Аналогічно визначається ціна споживання для аналогу:

$$Ц_{са} = 9\,500,0 + 48275,2 = 57775,2 \text{ (грн.)}$$

4.4 Визначення економічної ефективності

Економічна ефективність — досягнення найбільших результатів за найменших затрат живої та уречевленої праці. Економічна ефективність є конкретною формою дії закону економії часу. За капіталістичного способу виробництва узагальнюючий показник економічної ефективності — норма прибутку.

Економічний ефект в сфері проектування рішення:

$$E_{np} = Ц_n - Ц_a, \quad (4.18)$$

Тоді:

$$E_{np} = 12\,724,751 - 9\,500,0 = 3224,751 \text{ (грн.)}$$

					ДП. КСМ. 07139/14.00.00.000 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		58

Річний економічний ефект в сфері експлуатації:

$$E_{kc} = B_{ea} * B_{en}, \quad (4.19)$$

Тоді:

$$E_{kc} = 11014,92 - 7418,88 = 3596,04 \text{ (грн.)}$$

Додатковий економічний ефект у сфері експлуатації:

$$\Delta E_{екс} = \sum_{t=1}^T E_{екс} (1+R)^{-t}, \quad (4.20)$$

Тоді:

$$\Delta E_{екс} = \sum_{t=1}^5 3596,04 (1+0,08)^{-t} = 20219,61 \text{ (грн.)}$$

Сумарний ефект складає:

$$E = E_{np} + E_{екс}, \quad (4.21)$$

Тоді:

$$12\,724,751 + 20\,219,61 = 32\,944,361 \text{ (грн.)}$$

Отже, в даному розділі проведено розрахунок витрат на розробку проектного рішення. Здійснено порівняння з існуючим аналогом, проведено аналіз обраних даних показує економічну доцільність розробки веб – ресурсу наукової групи «Аналіз біомедичних зображень». Згідно проведеного економічного обґрунтування дане проектно рішення є конкурентноздатним.

					ДП. КСМ. 07139/14.00.00.000 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		59

ВИСНОВКИ

Отже, в результаті розробки дипломного проекту можна зробити наступні висновки:

1. Проведено аналіз програмних засобів розробки веб – сайтів. На основі аналітичного підходу проведено порівняльний аналіз існуючих веб – ресурсів наукових груп, виділено їх переваги та недоліки, що дозволить врахувати їх під час розробки власного веб – ресурсу.

2. Розроблено алгоритм пошуку публікацій на веб – сайті в режимі реального часу з використанням технології vue.js, laravel та бази даних mysql. Розроблено структуру таблиць бази даних для зберігання інформації про користувачів системи, новини, публікації, учасників наукової групи. Запропонована структура таблиць дозволяє здійснювати швидкий пошук по публікаціях, що підвищує якість веб – ресурсу.

3. Представлено структуру клієнтської та серверної частини веб – сайту наукової групи «Аналіз біомедичних зображень». Здійснено розробку unit – тестів для тестування критичних ділянок коду в автоматичному режимі. Здійснено порівняльний аналіз розробленого сайту із існуючими аналогами, що дозволило виділити переваги та недоліки кожного сайту. До переваг розробленого сайту можна віднести: адаптивність до дисплеїв із різним розширенням, наявність модулів пошуку та фільтрації інформації про публікації в режимі реального часу

					ДП. КСМ. 07139/14.00.00.000 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		60

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Електронна енциклопедія Вікіпедія: Веб-сайт [Електронний ресурс]
Режим доступу: <http://uk.wikipedia.org/wiki/Веб-сайт>.
2. Електронна енциклопедія Вікіпедія: Веб-додаток [Електронний ресурс]
Режим доступу: <http://uk.wikipedia.org/wiki/Веб-додаток>.
3. Кузнецов М. В. PHP. Практика создания Web-сайтов. / М. В. Кузнецов, И. В. Симдянов. — СПб.: БХВ-Петербург, 2009 — 1264 с.
4. Хокінс С. Администрирование веб-сервера Apache и руководство по электронной коммерции. — М.: Вильямс, 2001 — 336 с.
5. Дари К. AJAX и PHP разработка динамических приложений: Підручник/К.Дари, Б.Бринзаре, Ф. Черчер-Тоза / За ред. К.Дари —М.:Символ-Плюс, 2006 — 334с.
6. Гутман Е. PHP5 Профессинальное программирование/Е. Гутманс, С.Баккен, Д.Ретанс / За ред. Е.Гутман — СПб.:Символ-Плюс, 2004 — 432с.
7. Гаевский А. Ю. Самоучитель по созданию Web-страниц: HTML, JavaScript, Dynamic HTML / А. Ю. Гаевский, В. А.Романовский — К.: А.С.К., 2002 - 472с.
8. PHP: Мова програмування [Електронний ресурс] Режим доступу: <http://www.php.net/Веб сайт>
9. Фролов А. В. Практика применения PERL, PHP, APACHE и MySQL для активных Web-сайтов / А. В. Фролов, Г. В.Фролов - М.: Русская редакция, 2002 — 534с.

					ДП. КСМ. 07139/14.00.00.000 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		61

39. Електронна енциклопедія Вікіпедія: JSON [Електронний ресурс]
Режим доступу: <https://uk.wikipedia.org/wiki/JSON>
40. Компонентне або модульне тестування: [Електронний ресурс] Режим доступу: <http://www.protesting.ru/testing/levels/component.html>
41. Електронна енциклопедія Вікіпедія: Модульне тестування [Електронний ресурс] Режим доступу: [https://uk.wikipedia.org/wiki/Модульне тестування](https://uk.wikipedia.org/wiki/Модульне_тестування)
42. Електронна енциклопедія Вікіпедія:Список кодів стану HTTP [Електронний ресурс] Режим доступу:[https://uk.wikipedia.org/wiki/список HTTP](https://uk.wikipedia.org/wiki/список_HTTP)
43. Шапошников І. Web-сайт своїми руками./ І. Шапошников – СПб: Изд-во "Пітер", 2002 – 390с.
44. Гаевский А. Ю. Самоучитель по созданию Web-страниц: HTML, JavaScript, Dynamic HTML / А. Ю. Гаевский, В. А.Романовский. — К.: А.С.К., 2002 — 472с.
45. Методичні вказівки до написання техніко – економічного розділу для дипломних проектів на здобуття освітньо – кваліфікаційного рівня «Бакалавр» напряму підготовки 6.050102 «Комп’ютерна інженерія» / І.Р. Паздрій. – Тернопіль: ТНЕУ, 2015. – 36с.
46. Методичні рекомендації до виконання дипломного проекту з освітньо – кваліфікаційного рівня «Бакалавр» напряму підготовки 6.050102 «Комп’ютерна інженерія» фахового спрямування «Комп’ютерні системи та мережі» / Л.О. Дубчак О.М. Березький, Р.Б Трембач, Г.М. Мельник, Ю.М. Батько, С.В. Івасьєв / Під ред. О.М. Березького. – Тернопіль: ТНЕУ, 2016.- 60с.

					ДП. КСМ. 07139/14.00.00.000 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		64

ДОДАТОК А
Лістинг файлу «index.php»

```
<?php

/**
 * Laravel - A PHP Framework For Web Artisans
 *
 * @package Laravel
 */
require __DIR__.'../bootstrap/autoload.php';

$app = require_once __DIR__.'../bootstrap/app.php';

$kernel = $app->make(Illuminate\Contracts\Http\Kernel::class);

$response = $kernel->handle(
    $request = Illuminate\Http\Request::capture()
);
```

					ДП. КСМ. 07139/14.00.00.000 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		65

```
$response->send();
```

```
$kernel->terminate($request, $response);
```

ДОДАТОК Б

Лістинг файлу «index.blade.php»

```
{{Session::get('message')}}
@extends('layouts.master')
@section('content')

<link rel="stylesheet" type="text/css" href="/css/menu.css ">
<div class="container" style="height: 500px;background-color: #596778">
  <div class="menu1" style="padding-top: 2%">
    <h2>TNEU Computer Vision & Artificial Intelligence Scientific Group</h2>

    <nav class="menu">
      <input type="checkbox" href="#" class="menu-open" name="menu-open" id="menu-open"
/>

      <label class="menu-open-button" for="menu-open">
        <span class="lines line-1"></span>
        <span class="lines line-2"></span>
        <span class="lines line-3"></span>
      </label>

      <a href="#news" class="menu-item blue">  </a>
      <a href="#search" class="menu-item green">  </a>
      <a href="#HIAMS" class="menu-item red">  </a>
```

Змн.	Арк.	№ докум.	Підпис	Дата

ДП. КСМ. 07139/14.00.00.000 ПЗ

Арк.

66


```

</div>

</div>
<div class="column">
  <div class="default text">Pik</div>
  <div class="styled-select slate">
    <select v-on:click="search()" v-model="query">
      <option disabled value="Pik">Pik</option>
      <option>2016</option>
      <option>2017</option>
      <option>2018</option>
    </select>
  </div>

</div>
</div>

<div class="ui relaxed divided list" style="margin: 5%">
  <div class="item" v-for="article in articles">
    <i class=""><img src="" height="22px"></i>
    <div class="content">
      <a class="header" href="" >@{{article.text}}</a>

<div class="description">@{{article.created_at}}</div>

</div>
  </div>
</div>
</div>
</section>
<section id="HIAMS" style="background-color: #5f9ea0; padding: 2%" >
  <div class="ui two column middle aligned very relaxed stackable grid">
    <div class="column" style="text-align: center ">
      <div class="ui massive label" style="margin-bottom: 2%; margin-top: 2%">
        Гібридна система автоматизованої мікроскопії <b>HIAMS</b>
      </div>
      
    </div>
    <div class="ui vertical divider">
      Or
    </div>
    <div class="center aligned column">
      <h2 style="-webkit-text-fill-color: white">Система володіє окремими інтерфейсами для
різних типів користувачів</h2>
      <div class="ui animated button" tabindex="0">
        <a href="/HIAMS.html">

        <div class="visible content">Детальніше</div>
        <div class="hidden content">
          <i class="right arrow icon"></i>
        </div>
      </div>
    </div>
  </div>
</a>

```



```

        </div>
    </div>
</div>
</section>
<section id="news">
<div class="container" style="text-align: center; background-color: #d8e1cf; padding-top: 1% ">
    <h2 class="ui header">
        
        <div class="content">
            Статті
        </div>
    </h2>
</div>
<div class="ui stackable grid" style=" background-color: #d8e1cf">

    @forelse($news as $n)

        <div class="example-2 card eight wide column stackable grid"
            style="background-color: #d8e1cf; border-color: #d8e1cf; text-align: center; ">
            <div class="wrapper" style="background: url(images/articles/{{ $n->img }}) center/cover
no-repeat; width: 100%">
                <div class="header">
                    <div class="date">

<span class="day">{{ $n->created_at }}</span>

                </div>
</div>
<ul class="menu-content">

        </ul>
    </div>
    <div class="data">
        <div class="content">
            <span class="author">Jane Doe</span>
            <h2 class="title"><a href="/{{ $n->href }}">{{ $n->title }}</a></h2>
            <p class="text">{{ $n->text }}</p>
            <a href="/{{ $n->href }}" class="button">Read more</a>
        </div>
    </div>
</div>
</div>
</div>
    @empty
        No news
    @endforelse
</div>

</section>

<section id="top-publication">
    <div class="container" style="text-align: center; padding: 2%">
        <h2 class="ui header">
            

```

```
<div class="content">
  Top публікації
</div>
</h2>
</div>
```

```
<div class="ui relaxed divided list" style="margin: 5%">
```

```
  @forelse($articles as $article)
```

```
    <div class="item">
      <i class=""></i>
      <div class="content">
        <a class="header" href="{{ $article->href }}">{{ $article->text }}</a>
        <div class="description">{{ $article->created_at }}</div>
      </div>
    </div>
```

```
  @empty
    No news
  @endforelse
```

```
</div>
</section>
```

```
<section id="contact">
```

```
  <div class="container" style="text-align: center; background-color: #fff; padding-top: 1% ">
```

```
<h2 class="ui header">
```

```
  <div class="content">
```

```
    Зворотній зв'язок
```

```
  </div>
```

```
</h2>
```

```
</div>
```

```
<div class="ui two column middle aligned very relaxed stackable grid">
```

```
  <div class="center aligned column" >
```

```
    <div class="ui form" style="padding-left: 5%">
```

```
      <form action="#" @submit.prevent="AddNewFeedback" method="POST">
```

```
        <input type="hidden" value="{{ url('/') }}" v-model="baseUrl" />
```

		{{ csrf_field() }}			ДП. КСМ. 07139/14.00.00.000 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		70

```

    <div class="field">
      <label>Email</label>
      <div class="ui left icon input">
        <input v-model="feedbackUser.email" type="text" name="email" class="form-
control" placeholder="email"
          required>
        <i class="user icon"></i>
      </div>
    </div>

    <div class="field">
      <label>ім'я (Name)</label>

      <input v-model="feedbackUser.name" type="text" name="name" class="form-control"
placeholder="name"
        required>

    </div>

    <div class="field">
      <label>Текст (text)</label>

      <textarea v-model="feedbackUser.text" type="text" name="text" class="form-control"
placeholder="text"
        required></textarea>
    </div>

    <div class="form-group">
      <button :disabled="!isValid" class="ui primary button" type="submit" v-
if="!edit">Відправити</button>
    </div>
  </form>

```

					ДП. КСМ. 07139/14.00.00.000 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		71

```

</div>

<div class="ui green labels" transition="success" v-if="success">
  <a class="ui label">Дані успішно відправлено. Дякуємо ! </a></div>
</div>

<div class="ui vertical divider">
  Or
</div>

<div class="center aligned column">
  <div id="fb-root"></div>
  <script>(function(d, s, id) {
    var js, fjs = d.getElementsByTagName(s)[0];
    if (d.getElementById(id)) return;

    js = d.createElement(s); js.id = id;

    js.src
    =
'https://connect.facebook.net/uk_UA/sdk.js#xfbml=1&version=v2.12&appId=279243079229003&auto
LogAppEvents=1';

    fjs.parentNode.insertBefore(js, fjs);

    })(document, 'script', 'facebook-jssdk');</script>
  </div>
</div>
</div>
</section>
@endsection
@push('scripts')
  <script src="/js/myapp.js"></script>
@endpush

```

					ДП. КСМ. 07139/14.00.00.000 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		72

ДОДАТОК В

					ДП. КСМ. 07139/14.00.00.000 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		73