

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

Булило Ірина Ігорівна

**Веб-ресурс обліку абітурієнтів кафедри
комп'ютерної інженерії / Web resource for the
entrants accounting of the computer
engineering department**

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

Виконав студент групи КСМ-42/1
Булило Ірина Ігорівна

Науковий керівник: к.т.н.,
Піцун О.Й

Тернопіль - 2018

РЕЗЮМЕ

Дипломний проект містить 76 сторінок пояснюючої записки, 15 рисунків, 17 таблиць, 2 додатки. Обсяг графічного матеріалу 3 аркуші формату А3.

Метою дипломного проекту є розробка веб – ресурсу обліку абітурієнтів кафедри комп'ютерної інженерії ТНЕУ.

В дипломному проекті на основі порівняльного аналізу існуючих веб – ресурсів обліку абітурієнтів та студентів виділено їх переваги та недоліки, що дозволило сформулювати перелік обов'язкового функціоналу.

Враховуючи ріст популярності застосування інформаційних технологій в освіті, розроблено систему реєстрації та обліку абітурієнтів кафедри комп'ютерної інженерії. Для забезпечення можливості зручного перегляду інформації та взаємодії з системою на мобільних пристроях з допомогою фреймворку Semantic UI розроблено адаптивний графічний інтерфейс для дисплеїв із різним розширенням.

Розроблений сайт складається з клієнтської та адміністративної частини. У клієнтській частині відображається інформація про умови вступу на спеціальність та форма анкети абітурієнта. У адміністративній частині можна переглядати інформацію про зареєстрованих користувачів за такими полями: ПІБ, адреса електронної пошти, номер телефону, інформація про те, звідки абітурієнт дізнався про кафедру.

Можливість формування рейтингу абітурієнтів та розсилання електронних повідомлень значно підвищить продуктивність роботи кафедри у профорієнтаційній області.

Ключові слова: ВЕБ – РЕСУРС, ОБЛІК АБІТУРІЄНТІВ, РЕЙТИНГ АБІТУРІЄНТІВ, АДАПТИВНИЙ ГРАФІЧНИЙ ІНТЕРФЕЙС.

RESUME

The degree project contains 76 pages of explanatory notes, 15 figures, 17 tables, and 2 appendixes. The volume of graphic material is 3 sheets of A3 format.

The purpose of the degree project is the development of a web resource for the entrants of the TNEU Computer Engineering Department.

In the degree project, on the basis of a comparative analysis of existing web resources for the account of university entrants and students, their advantages and disadvantages are highlighted, which made it possible to form a list of obligatory functional.

Taking into account the growing popularity of application of information technologies in education, the system of registration and registration of entrants of the department of computer engineering has been developed. To provide convenient information viewing and interaction with the system on mobile devices, the Semantic UI framework has developed an adaptive graphical interface for displays with different extensions.

The developed site consists of client and administrative part. In the client part information about the conditions of admission to a specialty and the form of the applicant's application form are displayed. In the administrative part you can view information about registered users in the following fields: Pub, e-mail address, telephone number, information about where the entrant learned about the department.

Possibility of forming the rating of entrants and sending e-mail will significantly increase the productivity of the department in the career-oriented area.

Key words: WEB - RESOURCE, ACCOUNTING OF ABITURIENTS, RATING OF ABITURIENTS, ADAPTIVE GRAPHIC INTERFACE.

ЗМІСТ

Вступ.....	9
1 Аналітичний огляд підходів до створення web-сайтів.....	11
1.1 Програмні засоби створення web-сайтів	11
1.2 Класифікація веб-ресурсів	18
1.3 Аналіз систем обліку користувачів.....	22
1.4 Висновки та постановка задачі.....	24
2 Алгоритми роботи модулів сайту.....	26
2.1 Алгоритм реєстрації абітурієнтів в інформаційній системі	26
2.2 Алгоритм розподілу доступу користувачів.....	31
2.3 Структура бази даних	35
3 Програмна реалізація веб-сайту обліку абітурієнтів.....	41
3.1 Структура програмного модуля	41
3.2 Модуль роботи з базою даних	45
3.3 Порівняльний аналіз розробленого сайту з аналогами.....	48
4 Техніко-економічне обґрунтування	51
4.1 Розрахунок витрат на виконання проектного рішення	51
4.2 Визначення експлуатаційних витрат	58
4.3 Розрахунок ціни споживання проектного рішення	61
4.4 Визначення економічної ефективності.....	63
Висновки	65
Списки використаних джерел.....	66
Додаток А. Лістинг файлу «AbiturientController.php».....	70
Додаток Б. Лістинг файлу «index.blade.php».....	73
Додаток Г. Довідка про використання	76

					ДП. КСМ. 07123/14. 00.00.000 ПЗ			
Змн.	Арк.	№ докум.	Підпис	Дата				
Розроб.	Булило І.І.				ВЕБ-РЕСУРС ОБЛІКУ АБІТУРІЄНТІВ КАФЕДРИ КОМП'ЮТЕРНОЇ ІНЖЕНЕРІЇ	Літ.	Арк.	Акрушів
Перевір.	Піцун О.Й.					н	8	76
Консульт.	Паздрій І.Р.					ТНЕУ.ФКІТ.КСМ-42/1		
Н. Контр.	Гураль І. В.							
Затверд.	Березький О.М							

ВСТУП

Розвиток Інтернет технологій відкриває перед навчальними закладами новий рівень можливостей виходу на відкритий інформаційний простір і створення свого іміджу [8]. Цьому якнайкраще сприяє веб-сайт ВНЗ. Веб-сайт навчального закладу можна трактувати як інформаційний ресурс навчального призначення, виконаний засобами веб-технологій. Своєчасне інформування, оперативне й ефективне задоволення потреб студентів, абітурієнтів та їхніх батьків, адміністрації, педагогічного колективу нині забезпечують веб-сайти, які дозволяють розширити й урізноманітнити спектр навчально-методичного й інформаційного сервісу, зробити інформаційно-освітнє середовище навчального закладу відкритим для кожного користувача.

Все гостріше перед вищими навчальними закладами постає проблема автоматизованого опрацювання інформації. Особливо автоматизації потребує облік абітурієнтів, адже кількість абітурієнтів з кожним роком зростає. Автоматизація є необхідною, оскільки не вимагає кропіткої роботи з різноманітними паперами. Все більше програм для зберігання даних використовують бази даних. Так як пошук в них виробляти значно простіше ніж, у звичайному файлі. Так само СУБД має вбудовану систему безпеки, що зменшує ризик крадіжки даних та їх псування, формує вихідні документи, проводить зміни в записах бази даних. Такі програми призначені для полегшення праці і уникнення помилок, а також вони є зручними для користувачів, для яких достатньо мати доступ до Інтернету щоб здійснити необхідні дії. Саме тому створюючи веб-сайт важливим є розробити систему обліку абітурієнтів.

Зараз майже кожен ВНЗ України має власний веб-сайт на просторах інтернету, всі вони звісно мають свої переваги та недоліки і більшість з них ведуть облік абітурієнтів. Однією з важливих функцій автоматизованого керування є організація зберігання інформації з метою швидкого пошуку необхідних даних. Для цього вся збережена в комп'ютері інформація повинна бути розсортована по ряду ознак. Будь-яка зміна інформації повинна миттєво

					ДП. КСМ. 07123/14.00.00.000 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		9

враховуватися. В цілому облік абітурієнтів полегшить роботу адміністрації ВНЗ чи факультету, а також абітурієнти, які зареєструвалися будуть отримувати необхідну інформацію через електронну розсилку повідомлень.

Основною метою даного дипломного проекту є розробка веб-ресурсу обліку абітурієнтів кафедри комп'ютерної інженерії.

Актуальність розробки даного веб-ресурсу всебічно сприяти оперативному забезпеченню абітурієнтів, максимально повною інформацією, створити більш комфортні умови як для майбутніх студентів, так і для працівників кафедри.

					ДП. КСМ. 07123/14.00.00.000 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		10

1 АНАЛІЗ ЗАСОБІВ РОЗРОБКИ WEB-САЙТІВ

1.1 Програмні засоби розробки web-сайтів

Всесвітня павутина - глобальний інформаційний простір, заснований на фізичній інфраструктурі Інтернету і протоколі передачі даних HTTP. Всесвітня павутина викликала справжню революцію в інформаційних технологіях і бум в розвитку Інтернету. Часто, кажучи про Інтернет, мають на увазі саме Всесвітню павутину.

Інтернет утворюють мільйони веб-серверів мережі Інтернет, розташованих по всьому світу. Веб-сервер є програмою, що запускається на підключеному до мережі комп'ютері і використовує протокол HTTP для передачі даних. В найпростішому вигляді така програма одержує по мережі HTTP-запит на певний ресурс, знаходить відповідний файл на локальному жорсткому диску і відправляє його по мережі комп'ютеру, що запитав. Складніші веб-сервери здатні динамічно сформувати ресурси у відповідь на HTTP-запит.

Для перегляду інформації, отриманої від веб-серверу, на клієнтському комп'ютері застосовується спеціальна програма – веб - браузер. Основна функція веб-браузера відображення гіпертексту. Всесвітня павутина невід'ємно зв'язана з поняттями гіпертексту і гіперпосилання. Велика частина інформації у Веб представляє з себе саме гіпертекст. Для полегшення створення, зберігання і відображення гіпертексту у Всесвітній павутині традиційно використовується мова HTML, мова розмітки гіпертексту. Робота по розмітці гіпертексту називається версткою. Після HTML-розмітки гіпертекст, що вийшов, поміщається у файл, такий HTML-файл є найпоширенішим ресурсом Всесвітньої павутини. Після того, як HTML-файл стає доступний веб-серверу, його починають називати «веб-сторінкою». Набір веб-сторінок утворює веб-сайт.

Веб-сайт - сукупність веб-сторінок, доступних у мережі Інтернеті, які об'єднані як за змістом, так і навігаційно [1]. Фізично сайт може розміщуватися як на одному, так і на кількох серверах. Вузол мережі Інтернет або комп'ютер з

					ДП. КСМ. 07123/14.00.00.000 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		11

унікальною IP-адресою чи будь-який об'єкт в мережі Інтернеті за яким закріплена адреса, що ідентифікує його в мережі також називається сайтом.

Веб-додаток - додаток, в якому клієнтом виступає браузер, а сервером - веб-сервер. Браузер є різновидом так званих тонких клієнтів. Браузер здатний відображати веб-сторінки і, як правило, входить до складу операційної системи, а функції його оновлення і супроводу лежать на постачальнику операційної системи. Вся логіка додатку зосереджується на сервері, а браузер лише відображає інформацію, завантажену по мережі з серверу. Однією з переваг такого підходу є той факт, що клієнти не залежать від конкретної операційної системи, і веб-додатки, таким чином, є міжплатформеними сервісами.

Для створення веб-сайтів на стороні сервера використовуються різноманітні технології та мови програмування, такі як:

- PHP;
- ASP.NET;
- Java;
- Python;
- Perl;
- Ruby та інші.

Ряд з них (PHP, Python) мають відкритий код, поширюються вільно і можуть бути використані практично на будь-яких веб-серверах, інші (ASP, ASP.NET, Java) - прив'язані до конкретних веб-серверів.

PHP (англ. PHP:Hypertext Preprocessor - PHP: гіпертекстовий препроцесор), попередня назва: Personal Home Page Tools - скриптова мова програмування, була створена для генерації HTML - сторінок на стороні веб-сервера. PHP є однією з найпоширеніших мов програмування, що використовуються у сфері веб-розробок (разом із Java, NET, Perl, Python, Ruby).PHP підтримується переважною більшістю хостинг провайдерів. PHP – проект відкритого програмного забезпечення. Порівняльна характеристика мов програмування наведена у таблиці 1.1.

					ДП. КСМ. 07123/14.00.00.000 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		12

Таблиця 1.1 – Порівняльна характеристика мов програмування

Мова програмування	Переваги	Недоліки
1	2	3
PHP	Підтримка ООП, наявність анонімних функцій, наявність явної та неявної типізації, наявність розвинених фреймворків.	Ручне керування пам'яттю, відсутність розвинених бібліотек Ручне керування пам'яттю, відсутність розвинених бібліотек для роботи із зображеннями.
JSP	Підтримка ООП, функціональність, наявність анонімних функцій та інтерфейсів.	Ручне керування пам'яттю, відсутність явної типізації.
Java	Переносимість, надійність, наявність «збирача сміття», та стандартних бібліотек, підтримка ООП.	Ручне керування пам'яттю, відсутність анонімних функцій.
Python	Відкрита розробка, досить простий у вивченні, безліч корисних бібліотек і розширень мови, підтримка ООП, наявність явної типізації.	Відсутність комерційної підтримки засобів розробки початкова обмеженість коштів для роботи з базами даних, ручне керування пам'яттю.
Perl	Вбудовані засоби для роботи зі складними структурами; вільний синтаксис.	Ручне керування пам'яттю, наявність багатьох не потрібних бібліотек-модулів.

- безпека - PHP надає в розпорядження розробників і адміністраторів гнучкі і ефективні засоби безпеки, які умовно поділяються на дві категорії: засоби системного рівня і засоби рівня додатки;

- гнучкість - оскільки PHP є вбудовуваною (embedded) мовою, вона відрізняється винятковою гнучкістю по відношенню до потреб розробника. Хоч PHP в більшості випадків рекомендується використовувати в поєднанні з HTML, вона з таким же успіхом інтегрується і в JavaScript, WML, XML та інші мови. Крім того, добре структуровані додатки PHP легко розширюються в міру.

Для оформлення веб-сайтів і деяких інших видів документів використовується переважно мова CSS (каскадні таблиці стилів). CSS використовується творцями веб-сторінок для задання кольорів, шрифтів, розташування окремих блоків і інших аспектів представлення зовнішнього вигляду цих веб-сторінок.

Основною метою розробки CSS було розділення опису логічної структури веб-сторінки (яке проводиться за допомогою HTML або інших мов розмітки) від опису зовнішнього вигляду цієї веб-сторінки (яке тепер проводиться за допомогою мови CSS). Такий поділ може збільшити доступність документа, надати велику гнучкість і можливість управління його поданням, а також зменшити складність і повторюваність в структурному вмісті.

Apache HTTP-сервер - безкоштовний веб-сервер. Безпосередніми перевагами Apache є надійність та гнучкість конфігурації [16]. Apache дозволяє підключати зовнішні модулі, використовувати СКБД для аутентифікації користувачів, модифікувати повідомлення про помилки і т. д. Apache зіграв ключову роль у початковому зростанні всесвітньої павутини, і продовжує бути найпопулярнішим у світі веб-сервером, де-факто платформою, на яку орієнтуються інші веб-сервери.

Відповідно до статистики Netcraft за червень 2008 року, Apache є найпоширенішим серверним програмним забезпеченням в Мережі: на цей веб-сервер припадала частка близько 49% відповідного сегменту ринку (майже 85 мільйонів сайтів).

					ДП. КСМ. 07123/14.00.00.000 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		15

MySQL - вільна система керування реляційними базами даних. MySQL був розроблений для підвищення швидкодії обробки великих баз даних. Ця система керування базами даних (СКБД) з відкритим кодом була створена як альтернатива комерційним системам. MySQL з самого початку була дуже схожою на mSQL, проте з часом вона все розширювалася і зараз MySQL - одна з найпоширеніших систем керування базами даних. Вона використовується, в першу чергу, для створення динамічних веб-сторінок, оскільки має чудову підтримку з боку різноманітних мов програмування.

MySQL - компактний багатопотоковий сервер баз даних. Характеризується високою швидкістю, стійкістю і простотою використання.

Можливості сервера MySQL:

- простота у встановленні та використанні;
- підтримується необмежена кількість користувачів, які можуть одночасно працювати із БД;
- кількість рядків у таблицях може досягати 50 млн;
- висока швидкість виконання команд;
- наявність простої і ефективної системи безпеки.

PhpMyAdmin — веб-додаток з відкритим кодом на мові PHP із графічним веб-інтерфейсом для адміністрування бази даних MySQL. PhpMyAdmin дозволяє через браузер здійснювати адміністрування сервера MySQL, запускати запити SQL, переглядати та редагувати вміст таблиць баз даних. Ця програма користується великою популярністю у веб-розробників, оскільки дозволяє керувати базу даних MySQL без вводу SQL команд через дружній інтерфейс і з будь-якого комп'ютера під'єданого до інтернету без необхідності встановлення на ньому додаткового програмного забезпечення.

На сьогоднішній день phpMyAdmin широко застосовується на практиці. Останнє пов'язано з тим, що розробники інтенсивно розвивають свій продукт, з огляду на всі нововведення СКБД MySQL. Переважна більшість українських провайдерів використовують цей застосунок як панель керування для того, щоб надати своїм клієнтам можливість адміністрування виділених їм баз даних.

					ДП. КСМ. 07123/14.00.00.000 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		16

В основі веб-сайтів лежить архітектура "клієнт-сервер", в якій завдання або мережева навантаження розподілені між постачальниками послуг (сервісів), які називаються серверами, і замовниками цих послуг, званих клієнтами. Як середовище взаємодії клієнта з сервером використовується Інтернет. Графічне представлення архітектури "клієнт-сервер" зображено на рисунку 2.1.

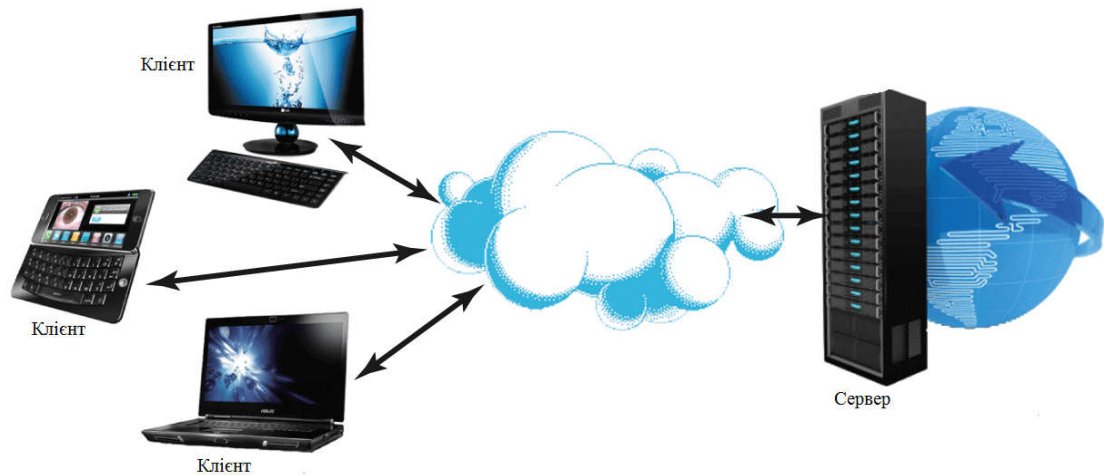


Рисунок 1.1 – Архітектура "клієнт-сервер"

Основними перевагами архітектури "клієнт-сервер" є:

- можливість, в більшості випадків, розподілити функції обчислювальної системи між декількома незалежними комп'ютерами в мережі. Це дозволяє полегшити обслуговування обчислювальної системи. Зокрема, заміна, ремонт, модернізація або переміщення сервера, не зачіпають клієнтів;
- всі дані зберігаються на сервері, який, як правило, значно краще захищений за більшість клієнтів. На сервері простіше забезпечити контроль повноважень, щоб вирішувати доступ до даних тільки клієнтам з відповідними правами доступу;
- дозволяє об'єднати різні клієнти. Використовувати ресурси одного сервера часто можуть клієнти з різними апаратними платформами, операційними системами і т.п.

Основні недоліки:

- у разі використання централізованої системи, непрацездатність

					ДП. КСМ. 07123/14.00.00.000 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		17

основного сервера може зробити непрацездатним весь додаток;

- адміністрування даної системи вимагає кваліфікованого професіонала;
- висока вартість обладнання.

Свою популярність дана архітектура отримала саме завдячуючи своїй універсальності, а також можливості організувати мережі з великою кількістю робочих станцій, що стало актуально через збільшення кількості Інтернет користувачів у світі [25]. До того ж, архітектура "клієнт-сервер" характеризується високою швидкістю обробки інформації.

Вибір засобів розробки веб-сайтів повинен забезпечувати умови відповідності розробленого програмного продукту необхідними показниками якості. На завершальному етапі повинні бути виявлені і усунені всі помилки.

Отже, у даному підрозділі на основі аналітичного підходу проведено аналіз сучасних архітектур та засобів розробки веб – сайтів, що дозволило обрати мову програмування PHP, СУБД – mysql та Apache – як веб-сервер.

1.2 Класифікація web-ресурсів

Завдяки зростанню впливу Інтернету, більшість компаній, організацій та установ вдається до створення власного веб-ресурсу. Це є потужним маркетинговим інструментом, тому що власний веб-сайт в інтернеті є однією з найефективніших видів впливу на користувача. Проаналізувавши досліджувану предметну область, можна відзначити, що велика кількість навчальних установ в більшості випадків володіють власними веб-ресурсами. Кожен намагається виділитися і максимально зацікавити потенційних користувачів та майбутніх абітурієнтів, надати їм максимум інформації про обрану спеціальність, деталі вступу, життя університету в цілому. Таким чином, сайт є обличчям своєї установи, і для того щоб вона була гідно представлена в мережі, необхідно розібратися в існуючих видах веб-ресурсів. Однак побудувати одну-єдину класифікацію сайтів навряд чи можливо, через те, що існує надто багато ознак, за

					ДП. КСМ. 07123/14.00.00.000 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		18

якими вони відрізняються. На рисунку 1.2 зображено основні категорії за якими поділяють веб-ресурси.

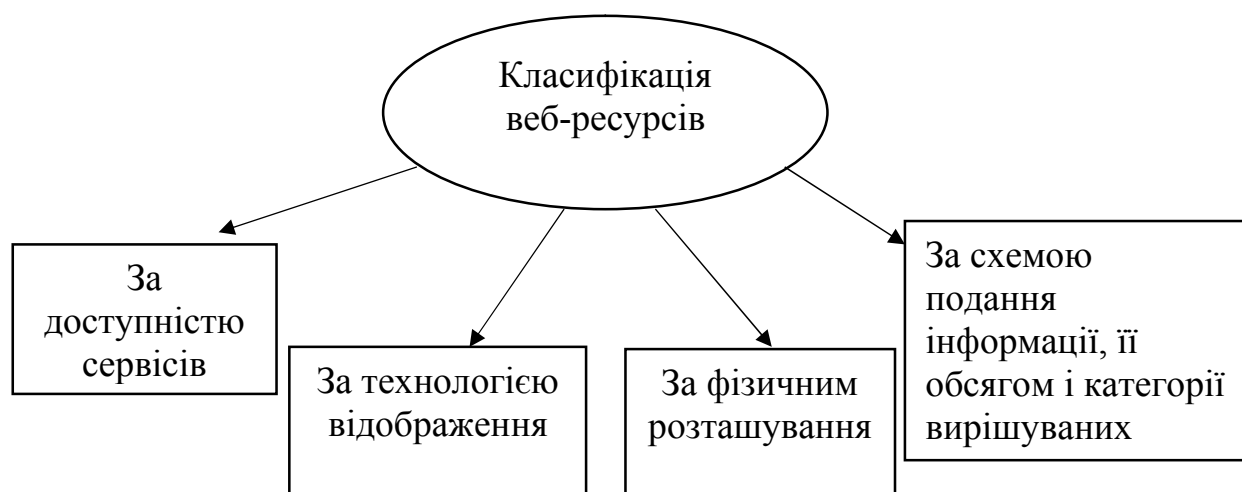


Рисунок 1.2 – Класифікація веб-ресурсів

На рисунку 1.3 зображено детальну класифікацію веб-ресурсів за категоріями.

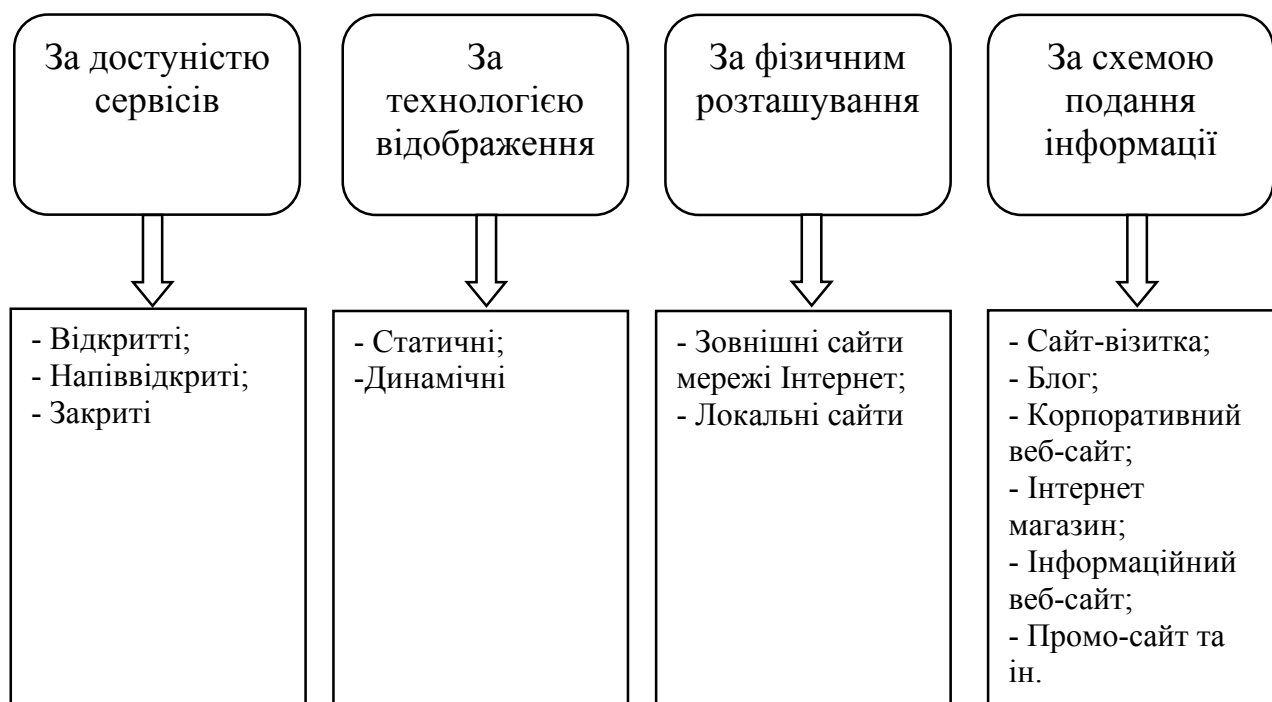


Рисунок 1.3 – Класифікація веб-ресурсів за категоріями

Відкриті веб-сайти - це сайти, де всі сервіси повністю доступні для будь-яких відвідувачів. Напіввідкриті - для доступу необхідно зареєструватися (зазвичай безкоштовно). Закриті - повністю закриті службові сайти організацій.

особисті сайти приватних осіб. Такі сайти доступні для вузького кола людей. Доступ новим людям зазвичай дається через так званий інвайт (запрошення).

Статичним прийнято називати сайт, що складається з незмінних, тобто статичних, HTML-сторінок. Статичні HTML-сторінки створюються вручну, після чого при кожному звертанні до сайту представляються користувачеві в незмінному вигляді. Щоб оновити інформацію на подібних сторінках, необхідно вручну внести зміни безпосередньо в програмний код сторінки. Динамічна веб-сторінка - веб-сторінка, вміст якої може змінюватись.

Переваги та недоліки статистичних та динамічних веб-сайтів наведено в таблиці 1.2.

Таблиця 1.2 – Веб-сайти за технологією відображення

Тип сайту	Переваги	Недоліки
Статичний	<ol style="list-style-type: none"> 1. Простота створення; 2. Невимогливість до апаратних ресурсів; 3. Дешевизна розробки; 4. Мінімальне навантаження на сервер; 5. Висока швидкість завантаження сторінок. 	<ol style="list-style-type: none"> 1. Відсутність інтерактивності; 2. Важко змінити інформацію, зовнішній вигляд і дизайн сторінок; 3. Не можливість повноцінної підтримки відвідувачів.
Динамічний	<ol style="list-style-type: none"> 1. Інтерактивність; 2. Миттєве відображення введених даних; 3. Мінімальні витрати на управління інформацією. 	<ol style="list-style-type: none"> 1. Підтримка додаткових технологій; 2. Підвищене навантаження на сервер; 3. Складно оптимізувати.

Із наведеної порівняльної характеристики можна зробити висновок, що найкращим та найоптимальнішим є динамічний веб сайт. Незважаючи на свої недоліки саме динамічний сайт може повноцінно задовільняти потреби

відвідувачів, надавати можливість перегляду зареєстрованих користувачів, авторизацію для адміна, а також бути більш зручними в управлінні.

Промо-сайт – це інтернет-ресурс, основна мета якого полягає в тому, щоб піднести певний товар, послугу, подію або бренд і заохотити покупця до його вибору. У цьому й полягає основна відмінність від сайту-візитки чи порталу, в яких подається загальна інформація про всі продукти без акценту на якому-небудь одному. Можна провести певну аналогію між промо-сайтом і буклетом або афішею. Як правило, промо-сайти використовуються для залучення уваги під час невеликих по тривалості рекламних кампаній або ж формування іміджевої складової діяльності фірми.

Результатом створення і просування промо-сайту для абітурієнтів кафедри комп'ютерної інженерії ТНЕУ має стати збільшення впізнаваності кафедри і отримання про неї більшої кількості інформації, залучення абітурієнтів до подальшого вступу. Відповідно, промо сайт для реєстрації абітурієнтів повинен виконувати наступні завдання:

- інформувати користувачів про останні новини, пов'язані із вступною кампанією;
- забезпечувати можливість швидкої та зручної реєстрації, як абітурієнта;
- надавати можливість адміністратору перегляду інформації про абітурієнтів;
- здійснювати власного оцінювання активності абітурієнтів;
- забезпечувати розсилання електронних повідомлень до абітурієнтів;
- мотивувати відвідувача на вчинення певної дії.

Отже, у даному підрозділі на основі аналітичного підходу проведено аналіз вимог до сучасних сайтів, що дозволило сформулювати критерії, яким повинен відповідати розроблений веб – сайт для реєстрації та обліку абітурієнтів.

					ДП. КСМ. 07123/14.00.00.000 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		21

1.3 Аналіз систем обліку користувачів

У теперішній час, коли розвиток комп'ютерних технологій та Інтернету все масштабніший, більшість речей ми можемо робити в онлайн-режимі. Покупка товарів, замовлення продукції, оплата певних послуг, реєстрація на конкретну подію, чи навіть заповнення анкети абітурієнта - все це можна зробити не виходячи з дому.

Оскільки абітурієнтів є чимала кількість, то все гострішою стає проблема автоматизованого опрацювання інформації. Адже все це не вимагає кропіткої роботи з різноманітними паперами. Все більше програм для зберігання даних використовують бази даних. Так як пошук в них виробляти значно простіше ніж, у звичайному файлі. На рисунку 1.4 зображено поточні сторінки веб-ресурсу обліку абітурієнтів.

а)

б)

в)

г)

Рисунок 1.4 – Веб-ресурси обліку абітурієнтів: а) сайт ФКІТу(ТНЕУ); б) сайт кафедри КІ (ТНЕУ); в) сайт ДонНАБА; г) сайт ЗДМУ

СУБД має вбудовану систему безпеки, що зменшує ризик крадіжки даних та їх псування, формує вихідні документи, проводить зміни в записах бази даних. Такі програми призначені для полегшення праці і уникнення помилок, а також вони є зручними для користувачів, для яких достатньо мати доступ до Інтернету щоб здійснити необхідні дії. Отож, більшість навчальних установ, створюючи власні веб-сайти, ведуть облік абітурієнтів. Всі вони мають свої переваги та недоліки. Перевагами сайту реєстрації ФКІТУ (див.рисунок 1.4 а) є:

- інформація зберігається в базі даних;
- зручний інтерфейс;
- можливість реєстрації кабінету користувача.

До недоліка можна віднести те, що інформація про спеціальності та форма реєстрації знаходяться на різних сторінках.

Головними недоліками сайту кафедри КІ (див.рисунок 1.4 б) є:

- використання гугл форми;
- дані зберігаються не в базу, а у файлі на гугл диску;
- неадаптивний інтерфейс.

Серед переваг можна відзначити те, що інформація про спеціальності та форма реєстрації знаходяться на одній сторінці.

Серед переваг сайту ДонНАБА (див. рисунок 1.4 в) та сайту ЗДМУ (див. рисунок 1.4 г) варто відзначити:

- адаптивний інтерфейс;
- вся інформація зберігається в базі даних.

Порівняльна характеристика даних веб-сайтів наведена у таблиці 1.3.

Таблиця 1.3 - Порівняльна характеристика веб-сайтів

Критерії оцінювання	ФКІТ	КІ	ДонНАБА	ЗДМУ
1	2	3	4	5
Адаптація під моб платформи + -	+	-	+	+

Продовження таблиці 1.3

1	2	3	4	5
Наявність полів для вказання ел – пошти і телефону + -	+	+	+	+ -
Електронна розсилка повідомлень	-	-	-	-
Формування звіту про джерела інформування абітурієнтів про кафедру. + -	-	+	+ -	+

Схема бази даних повинна бути коректна, тому що кожному об'єкту предметної області мають відповідати дані в пам'яті ЕОМ, а кожному процесу - адекватні процедури обробки даних, вона помітно зменшує рутинну роботу, підвищує швидкість пошуку і видачі необхідної інформації, підвищує достовірність результатів. БД забезпечується простотою і зручністю в експлуатації інформаційної системи.

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 тип створюваного сайту та проаналізувати аналоги.

Метою даного дипломного проекту є розробка веб-ресурсу обліку абітурієнтів кафедри комп'ютерної інженерії.

					ДП. КСМ. 07123/14.00.00.000 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		24

Для досягнення поставленої мети потрібно виконати наступні задачі:

- провести огляд підходів до створення веб – сайтів;
- провести порівняльну характеристику сайтів-аналогів;
- розробити алгоритм реєстрації абітурієнтів в системі;
- розробити структуру бази даних;
- порівняти результати роботи розробленого веб-ресурсу з іншими

відомими системами обліку абітурієнтів.

В даному розділі було досліджено основні та найбільш популярні засоби для створення веб – сайтів. Зокрема, проведено огляд сучасних мов програмування та програмних засобів для написання веб – сайту. В результаті проведення дослідження, обрано мову програмування та програмні засоби для написання веб – сайту. Також було досліджено класифікацію веб-ресурсів, проведено порівняльний аналіз відомих систем обліку абітурієнтів, проаналізовано основні переваги та недоліки деяких сайтів-аналогів.

Проведено огляд відомих веб-ресурсів обліку абітурієнтів за такими критеріями: можливість поповнення бази даних, хороший та зручний інтерфейс. У результаті проведеного аналізу можна дійти висновку, що недоліків у даних програмних засобів більше ніж переваг.

					ДП. КСМ. 07123/14.00.00.000 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		25

2 АЛГОРИТМИ РОБОТИ МОДУЛІВ САЙТУ

2.1 Алгоритм реєстрації абітурієнтів в інформаційній системі

Сучасне світ неможливий без ефективного управління. Однією з його найважливіших категорій є система обробки інформації. Від неї багато в чому залежить підвищення ефективності роботи та спрощення роботи відділень ВНЗ.

Алгоритм реєстрації абітурієнтів кафедри комп'ютерної інженерії наведено у ДП.КСМ.07123/14.00.00.000 А1. На першому етапі відбувається завантаження головної сторінки сайту, що складається з таких блоків:

- Новини;
- Етапи вступної кампанії;
- Форма реєстрації абітурієнтів;
- Блок «Питання – Відповідь».

Наступним кроком є під'єднання до бази даних mysql. У разі успішного під'єднання абітурієнт має можливість заповнювати форму реєстрації, що включає такі поля:

- Прізвище;
- Ім'я;
- По – батькові;
- Адреса електронної пошти;
- Телефонний номер;
- Адреса;
- Звідки Ви про нас дізнались.

Валідація елементів форми відбувається як на стороні клієнта, так і на стороні сервера. На клієнтській стороні використовуються бібліотека Vue.js, що володіє функціями перевірки введених даних в режимі реального часу. Такий підхід значно спрощує та полегшує процес вводу інформації. Переваги та недоліки Vue.js зображено в таблиці 2.1.

					ДП. КСМ. 07123/14.00.00.000 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		26

Таблиця 2.1 – Переваги та недоліки Vue.js

Переваги	Недоліки
<p>1. Простота і функціональність, для того щоб розібратись в бібліотеці потрібен мінімальний багаж знань.</p> <p>2. Відсутні вимоги до стеку, тому є можливість використовувати Vue.js на будь-якому проекті.</p> <p>3. Висока швидкість розробки.</p> <p>4. Можливість знайти і підключити до проекту практично будь-якого розробника, який хоч трохи знайомий з фронтенд розробкою.</p> <p>5. Мінімальне підключення нових ресурсів і як наслідок не висока собівартість.</p>	<p>1. Не зовсім підходить для створення складних інтерфейсів.</p> <p>2. Компонентний підхід у Vue.JS не такий гнучкий і очевидний.</p> <p>3. Не гнучка шаблонізація.</p>

Деякі поля введення самі надають стандартні способи валідації, без необхідності писати додатковий код. Наприклад, `<input type = "email" />` перевіряє поле, що введене значення відповідає шаблону правильної email адреси. Якщо в поле неправильно введено символ, форму не можна буде відправити, поки значення не буде виправлене. На рисунку 3.1 зображено не коректне заповнення поля e-mail адреси.

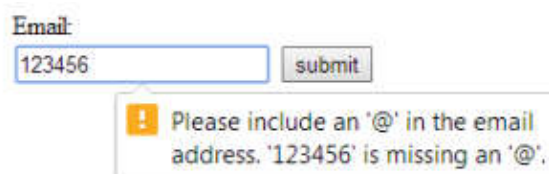


Рисунок 2.1 – Не коректне заповнення поля e-mail адреси

					ДП. КСМ. 07123/14.00.00.000 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		27

Існують і інші стандартні типи полів, для валідації чисел, і телефонних номерів. Також без заповнення обов'язкових полів користувач не зможе надіслати форму. На рисунку 2.2 наведено приклад обов'язкового поля для заповнення.

The image shows a web form with two input fields. The first field is labeled 'Name' and is empty, with a red border around it. Below it is a small grey box containing the text 'Please fill out this field.' The second field is labeled 'Address' and contains the text 'ww'. Below the 'Address' field is a 'send' button.

Рисунок 2.2 – Обов'язкове поле для заповнення

Після підтвердження користувачем правильності введених даних відбувається відправка POST – запиту до сервера. POST — один з багатьох методів запиту, що підтримуються протоколом передачі даних HTTP, який використовується у всесвітній мережі Інтернет. Метод POST призначений для запиту, при якому веб-сервер приймає дані збережені в тілі повідомлення, для зберігання. Метод часто використовується для завантаження файлу або передачі заповненої веб-форми.

В рамках методу POST-запиту довільну кількість даних будь-якого типу може бути відправлено на сервер в тексті листа запиту. Поля заголовка в POST-запиті зазвичай вказують на тип вмісту.

Для недопущення використання SQL та XSS – ін'єкцій на стороні сервера проводиться додаткова перевірка отриманого запиту.

SQL ін'єкція — один з поширених способів злому сайтів та програм, які працюють з базами даних, заснований на впровадженні в запит довільного SQL-коду. Впровадження SQL, залежно від типу СКБД та умов впровадження, може надати можливість атакуючому здійснити довільний запит до бази даних (наприклад, прочитати вміст будь-яких таблиць, видалити, змінити або додати дані), отримати можливість читання та/або запису локальних файлів та виконання довільних команд на сервері.

Атака типу впровадження SQL може бути можлива за некоректної обробки

					ДП. КСМ. 07123/14.00.00.000 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		28

Продовження таблиці 2.2

1	2
ПО-БАТЬКОВІ	ОБОВ'ЯЗКОВЕ. ПОВИННЕ СКЛАДАТИСЬ ЛИШЕ ІЗ ЛІТЕРА ПРОБІЛІВ.
Адреса електронної пошти	Обов'язкове. Повинне складатись із коректної електронної адреси (із @).
Телефонний номер	Обов'язкове. Повинне складатись із коректного номеру телефону (із 380).
Адреса	Обов'язкове. Повинне складатись із коректної адреси місця проживання (с./сmt./м., р-йн, обл, вул, буд/кв).
Звідки Ви про нас дізнались?	Необов'язкове. Багаторядкове поле введення даних.

Якщо валідація пройшла успішно, то інформація буде занесена в базу даних, а користувачу буде виведено повідомлення про успішну реєстрацію. Якщо ж валідація буде не пройдена, користувачу буде видано повідомлення про помилку і він повернеться до заповнення поля форми.

Отже, у даному підрозділі на основі результатів дослідження основних принципів додавання користувачів розроблено алгоритм в системі обліку абітурієнтів кафедри комп'ютерної інженерії. В результаті описано послідовність дій користувача під час заповнення форми.

2.2 Алгоритм розподілу доступу користувачів

					ДП. КСМ. 07123/14.00.00.000 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		30

Ключовою особливістю ролі моделі є те, що весь доступ до інформаційних систем і ресурсів надається тільки через ролі. Роль – це набір прав доступу. Користувачі отримують доступ тільки через присвоєні їм ролі.

Ідея надання доступу на основі ролей виникла оскільки було необхідне угруповання наборів прав в якісь окремі сутності, які давали б чітко зрозуміти, який доступ є у користувача, з яким було б зручніше і зрозуміліше працювати в разі змін як положення користувача в організації, так і набору прав доступу користувача.

Формування ролей повинне визначити чіткі і зрозумілі для користувачів комп'ютерної системи правила розмежування доступу. Рольове розмежування доступу дозволяє реалізувати гнучкі, що змінюються динамічно в процесі функціонування комп'ютерної системи правила розмежування доступу. Таке розмежування доступу є складовою багатьох сучасних комп'ютерних систем. Як правило, даний підхід застосовується в системах захисту СУБД, а окремі елементи реалізуються в мережевих операційних системах. Рольовий підхід часто використовується в системах, для користувачів яких чітко визначено коло їх посадових повноважень і обов'язків.

Так як привілеї не застосовуються до користувачів безпосередньо, а купуються ними тільки через свою роль (або ролі), то управління індивідуальними правами користувача по суті зводиться до призначення йому ролей. Це спрощує такі операції, як додавання користувача або зміна підрозділу користувачем.

Рольове управління доступом має низку необхідних позитивних властивостей, наприклад:

- можливість побудови ієрархії ролей з успадкуванням набору прав. Це дозволяє спростити рольову модель, особливо в організаціях з різномірною інфраструктурою, де використовується багато інформаційних систем. З використанням ієрархії немає необхідності повторно вказувати права в декількох

					ДП. КСМ. 07123/14.00.00.000 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		31

подібних ролях, досить помістити їх в одну велику роль, як дочірні, вказавши лише унікальні права для кожної ролі;

- просто і ефективно здійснюється надання однакових прав великій кількості користувачів - досить призначити користувачам одну роль;
- при необхідності зміни набору прав великій кількості користувачів досить змінити набір прав в ролі;
- можливість реалізації принципу поділу повноважень. Це значно знижує ризик надання користувачам надлишкових повноважень, наприклад, коли дві ролі не можуть бути в один момент часу призначені одному користувачеві.

Рівні доступу до системи обліку абітурієнтів наведено у ДП. КСМ.07123/14.00.00.000 С1.

Важливу роль у розподілі доступу відіграє ACL. ACL (список контролю доступу) – список прав доступу до об'єкта, який визначає, хто або що може отримувати доступ до нього, і які саме операції дозволено або заборонено цьому суб'єкту проводити над об'єктом. В системі з моделлю безпеки, заснованої на ACL, коли суб'єкт запитує виконання операції над об'єктом, система спочатку перевіряє список дозволених для цього суб'єкта операцій, і тільки після цього дає (або не дає) доступ до запитуваної дії.

Для того щоб одержати певний рівень доступу користувачу спочатку необхідно авторизуватися у системі. Для цього йому потрібно заповнити відповідні поля у формі авторизації. Авторизація - це надання певній особі прав на виконання деяких дій, а також процес перевірки даних прав при спробі виконання цих дій.

На першому етапі користувачу відображається форма для введення логіну і паролю. Наступним кроком є під'єднання до бази даних mysql. Логін, введений користувачем, порівнюється із уже зареєстрованими логінами в базі даних, якщо такий логін присутній, то відбувається пошук паролю. У разі не співпадіння користувачу виводиться повідомлення про помилку.

Логін — алфавітно цифровий набір символів, що ідентифікує користувача комп'ютерної мережі. Логін та пароль зберігаються в обліковому записі користувача.

					ДП. КСМ. 07123/14.00.00.000 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		32

Ще один крок - це шифрування введеного паролю. Шифрування - метод захисту інформації способом перетворення. Шифрування ускладнює перехоплення інформації іншими користувачами і може стати вагомим елементом в забезпеченні безпеки інформації. На рисунку 2.3 зображено алгоритми шифрування.

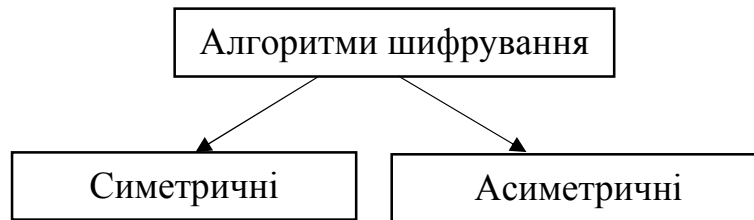


Рисунок 2.3 – Алгоритми шифрування

Суть симетричних алгоритмів шифрування (використання закритого ключа) заключається в тому, що відправник і отримувач інформації використовує один і той же ключ. Цей ключ повинен зберігатися таємно. Безпека системи зростає, якщо для кожного сеанса використовується унікальний ключ.

Суть асиметричних алгоритмів шифрування (використання відкритого ключа) в тому, що для зашифрування інформації використовують один ключ (відкритий), а для розшифрування інший (таємний). Відкриті і таємні ключі не співпадають.

Далі, згідно з алгоритмом авторизації, порівнюються паролі введений користувачем та наявний у базі даних. У разі співпадіння паролів користувачу відкривається головна сторінка і він може продовжувати працювати з системою. Якщо паролі не співпадають видається повідомлення про помилку введених даних та надається можливість повторного введення даних.

З таблицки БД "role_user" отримуємо значення ролі (режиму доступу) користувача. Режими доступу користувача наведені на рисунку 2.4.



Рисунок 2.4 – Режими доступу

Кожна роль має свої унікальні функції. Якщо це абітурієнт, то його функції наступні:

- перегляд інформації про умови вступу на спеціальність;
- заповнення анкети абітурієнта;
- отримання e-mail повідомлень.

Роль адміністратора є однією з найважливіших у роботі сайту, адже адміністратор сайту – це людина, яка контролює роботу сайту і підтримує його в робочому стані. У адміністратора є чимало обов’язків та функцій. Режим доступу адміністратора включає в себе такі функції:

- перегляд інформації про абітурієнтів;
- відправки email –повідомлень абітурієнтам;
- формування рейтингу абітурієнтів;
- видалення записів про абітурієнта;
- додавання інформації про умови вступу на спеціальність.

У адмін частині буде відображено список усіх зареєстрованих абітурієнтів, а адміністратор матиме можливість виставляти так звані очки (бали) абітурієнтам за участь в олімпіаді, зв’язок з адміністрацією кафедри і таким чином формуючи рейтинг. Також буде реалізовано сортування за оцінками.

Що стосується функцій викладача, то вони проявляються в наступному:

- перегляд інформації про абітурієнтів;
- перегляд інформації про умови вступу на спеціальність.

Отже, у даному підрозділі було описано алгоритм авторизації користувача в системі для визначення його ролі. Також було детально описано всі наявні рівні

доступу до системи обліку абітурієнтів та проаналізовано функції кожного режиму доступу.

2.3 Структура бази даних

Людина в процесі інформаційної діяльності збирає й накопичує відомості про навколишній світ. До появи обчислювальної техніки вся інформація зберігалася в письмовому або друкованому виді. Однак чим більше були обсяги інформації, з якими приходилось оперувати людині, тим гостріше вставало питання збереження інформації та її обробки.

Однією з важливих функцій інформатики є організація зберігання інформації з метою швидкого пошуку необхідних даних. Для цього вся збережена в комп'ютері інформація повинна бути розсортована по ряду ознак. Будь-яка зміна інформації повинна миттєво враховуватися.

Інформація, збережена в комп'ютері й об'єднана у взаємозалежну сукупність за рядом ознак, називається базою даних. Щоб керувати даними, які складають базу, необхідна окрема програма. Програми, які управляють зберіганням, обробкою й пошуком інформації в БД, називаються системами керування базами даних (СКБД).

Бази даних (БД)– це систематизоване сховище інформації певної предметної області.

Система керування базами даних (СКБД)– це програма, призначена для організації зберігання, обробки й пошуку інформації в БД. На рисунку 2.5 зображено основні можливості СКБД.

					ДП. КСМ. 07123/14.00.00.000 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		35



Рисунок 2.5 – Можливості СКБД

У базі даних основними об'єктами є запити, таблиці, звіти, форми, макроси, модулі та сторінки доступу до даних. Дані у базі організують відповідно до моделі організації даних. Таким чином, сучасна база даних, крім даних, містить їх опис та може містити засоби для їх обробки. Бази даних класифікують за різними критеріями. На рисунку 2.6 зображено класифікацію бази даних за двома критеріями.

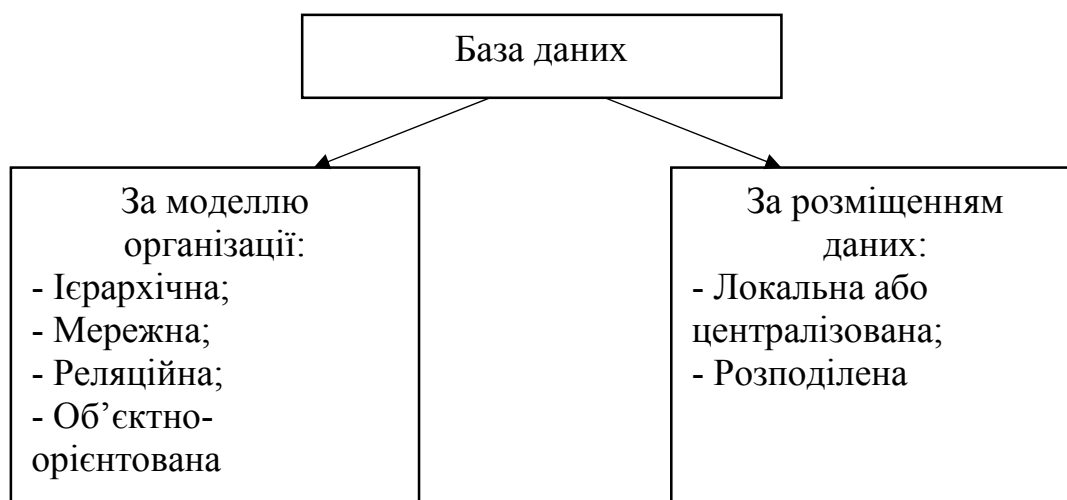


Рисунок 2.6 – Класифікація БД

Особливістю ієрархічної бази даних є те, що вона може бути представлена як дерево, що складається з об'єктів різних рівнів. Між об'єктами існують зв'язки типу «предок-нащадок». При цьому можлива ситуація, коли об'єкт не має нащадків або має їх декілька.

Мережна база даних подібна до ієрархічної, за винятком того, що кожен об'єкт може мати більше одного предка. В мережній моделі до вертикальних

					ДП. КСМ. 07123/14.00.00.000 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		36

ієрархічних зв'язків додаються горизонтальні. Тому мережна БД є більш гнучкою у здійсненні пошуку потрібних даних.

Реляційна модель даних являє собою набір двовимірних таблиць, які складаються зі стовпців (полів) і рядків (записів), а також мають ім'я, унікальне в межах даної БД. Реляційна модель даних, як правило, складається з декількох таблиць, які зв'язуються між собою ключами.

У Об'єктно-орієнтованій базі даних дані оформляють у вигляді моделей об'єктів. Також вона не потребує модифікації ядра при додаванні нового типу даних, так як це використовується у реляційних базах даних. У зовнішню структуру бази даних потрапляють нові класи та їх екземпляри.

Що стосується класифікації бази даних за розміщенням, то локальна, або централізована база даних підтримується на одному комп'ютері, а розподілена - це така база даних частини якої розміщують на різних комп'ютерах мережі.

Розробка бази даних відіграє велику роль у проектуванні веб-сайтів чи веб-модулів. Отож створення бази даних слід починати з її проектування (розробки). У результаті проектування має бути визначена структура бази, тобто склад таблиць, їхня структура та логічні зв'язки. Процес проектування можна здійснювати двома підходами. За першого підходу спочатку визначають основні задачі, для розв'язання яких створюється база, та потреби цих задач у даних. За другого підходу визначають предметну область (сферу), здійснюють аналіз її даних і встановлюють типові об'єкти предметної області. Найбільш раціональним підходом проектування бази даних є поєднання обох підходів.

Основними етапами розробки баз даних є:

- постановка завдання, яке включає в себе основні завдання щодо розробки БД, описує склад БД, призначення та цілі її створення;
- аналіз об'єктів - формується склад об'єктів БД, їхні параметри та типи цих параметрів;
- синтез моделі - обирається модель бази даних;
- програмна частина, яка містить в собі вибір СКБД;

					ДП. КСМ. 07123/14.00.00.000 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		37

- синтез комп'ютерної моделі об'єкта, що включає запуск системи управління базою даних, створення нового файлу бази даних, створення таблиць у БД.

У теперішніх мережевих інформаційних системах для роботи із загальною базою даних використовують архітектуру «клієнт-сервер». При цьому в мережі розміщують сервер баз даних. Ним виступає комп'ютер (або комп'ютери), який містить бази даних, СКБД та пов'язане з ними програмне забезпечення, і налаштований для надання користувачам інформаційної системи доступу до бази даних. Клієнти, які працюють із даними (вони можуть бути розташовані на різних комп'ютерах мережі), надсилають відповідні запити серверу. Сервер їх отримує, опрацьовує, та надсилає відповідь клієнту.

У ДП. КСМ.07123/14.00.00.001 С1 наведено структуру бази даних веб-ресурсу обліку абітурієнтів. Головне призначення даної бази даних – зберігання інформації про абітурієнтів, користувачів системи та даних рейтингу.

Таблиця «abiturients» є ключовою у даному проєкті та дозволяє зберігати інформацію про абітурієнтів, що заповнили відповідну анкету. Структуру таблиці «abiturients» наведено у таблиці 2.3.

Таблиця 2.3 - Структура таблиці «abiturients»

Назва поля	Тип даних	Опис
1	2	3
Id	Int(11)	Первинний ключ (Ідентифікатор абітурієнта)
Name	Varchar(255)	Ім'я
fathename	Varchar(255)	По-батькові
Surname	Varchar(255)	Прізвище
email	Varchar(100)	Адреса електронної пошти
telephone	Varchar(100)	Контактний телефон
address	Varchar(100)	Адреса

Продовження таблиці 2.3

					ДП. КСМ. 07123/14.00.00.000 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		38

1	2	3
Created_at	TIMESTAMP	Дата та час створення запису
Updated_at	TIMESTAMP	Дата та час редагування запису

Структуру таблиці «users» наведено у таблиці 2.4. Призначенням даної таблиці є зберігання інформації про користувачів системи.

Таблиця 2.4 - Структуру таблиці «users»

Назва поля	Тип даних	Опис
Id	Int(11)	Первинний ключ (Ідентифікатор користувача)
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 в форматі TIMESTAMP

Інколи виникає необхідність надання декількох прав доступу одному користувачу, наприклад адміністратор може одночасно адмініструвати систему та редагувати тексти статей сайту. Структуру таблиці «role_user» наведено у таблиці 2.5.

Таблиця 2.5 - Структура таблиці «role user»

					ДП. КСМ. 07123/14.00.00.000 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		39

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

Ролі бази даних можна використовувати для призначення дозволів бази даних в групі користувачів.

Отже, у даному підрозділі на основі дослідження основних етапів розробки баз даних було розроблено структуру бази даних для системи обліку абітурієнтів та описано структуру чотирьох таблиць, з яких складається база даних.

3 ПРОГРАМНА РЕАЛІЗАЦІЯ ВЕБ – САЙТУ ОБЛІКУ АБІТУРІЄНТІВ

					ДП. КСМ. 07123/14.00.00.000 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		40

3.1 Структура програмного модуля

3.1.1 Серверна частина

Клієнт-серверна архітектура набула своєї популярності завдяки динамічному розвитку мережі Інтернет та зосередження значної частини інформації в базах даних на серверах. Клієнт-серверну архітектуру можна означити, як концепцію інформаційної мережі в якій основна частина її ресурсів зосереджена в серверах, обслуговуючих своїх клієнтів. Саме тому робота більшості веб-сайтів включає в себе клієнтську та серверну частину. При розробці даного сайту було використано Apache HTTP-сервер. Apache HTTP-сервер - безкоштовний веб-сервер. Основними перевагами Apache є надійність та гнучкість конфігурації [16]. Apache дозволяє підключати зовнішні модулі, використовувати СКБД для аутентифікації користувачів, модифікувати повідомлення про помилки і т. д. [17].

Структуру веб – сайту реєстрації абітурієнтів наведено на рисунку 3.1.

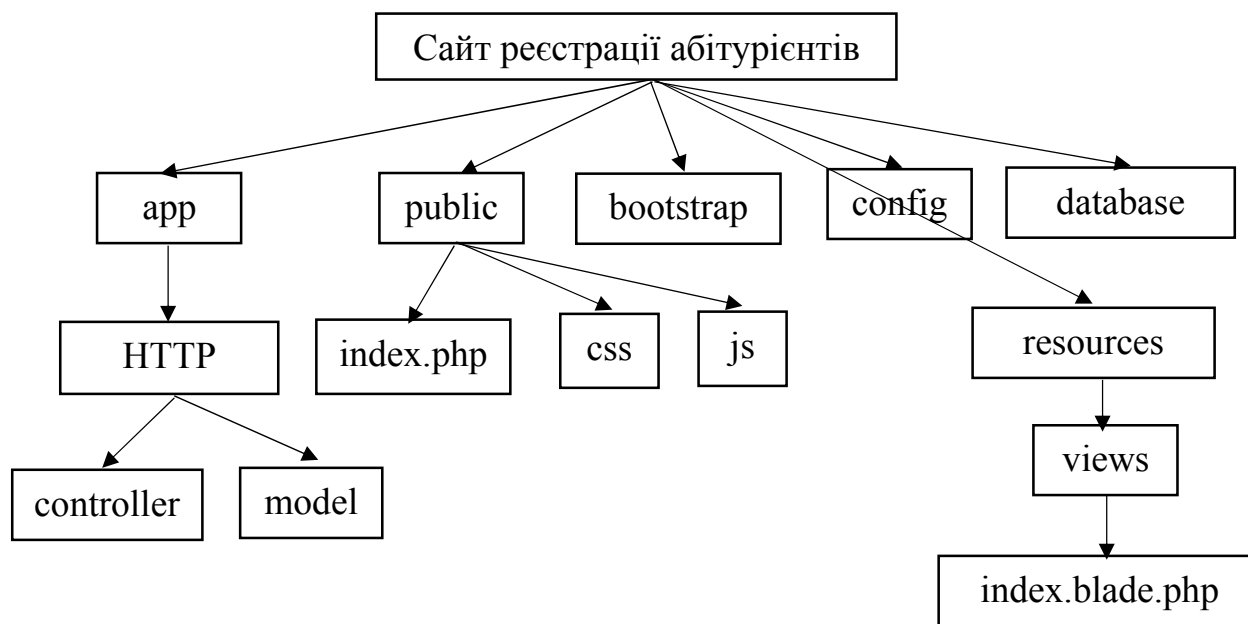


Рисунок 3.1 – Структура веб – сайту реєстрації абітурієнтів

Директорія app призначена для зберігання контролерів і моделей, що відповідають за логіку роботи сайту. Директорія model відповідає за зберігання

					ДП. КСМ. 07123/14.00.00.000 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		41

покрашувати або погіршувати якість роботи. На рисунку 3.2 зображено інтерфейс головної сторінки сайту.

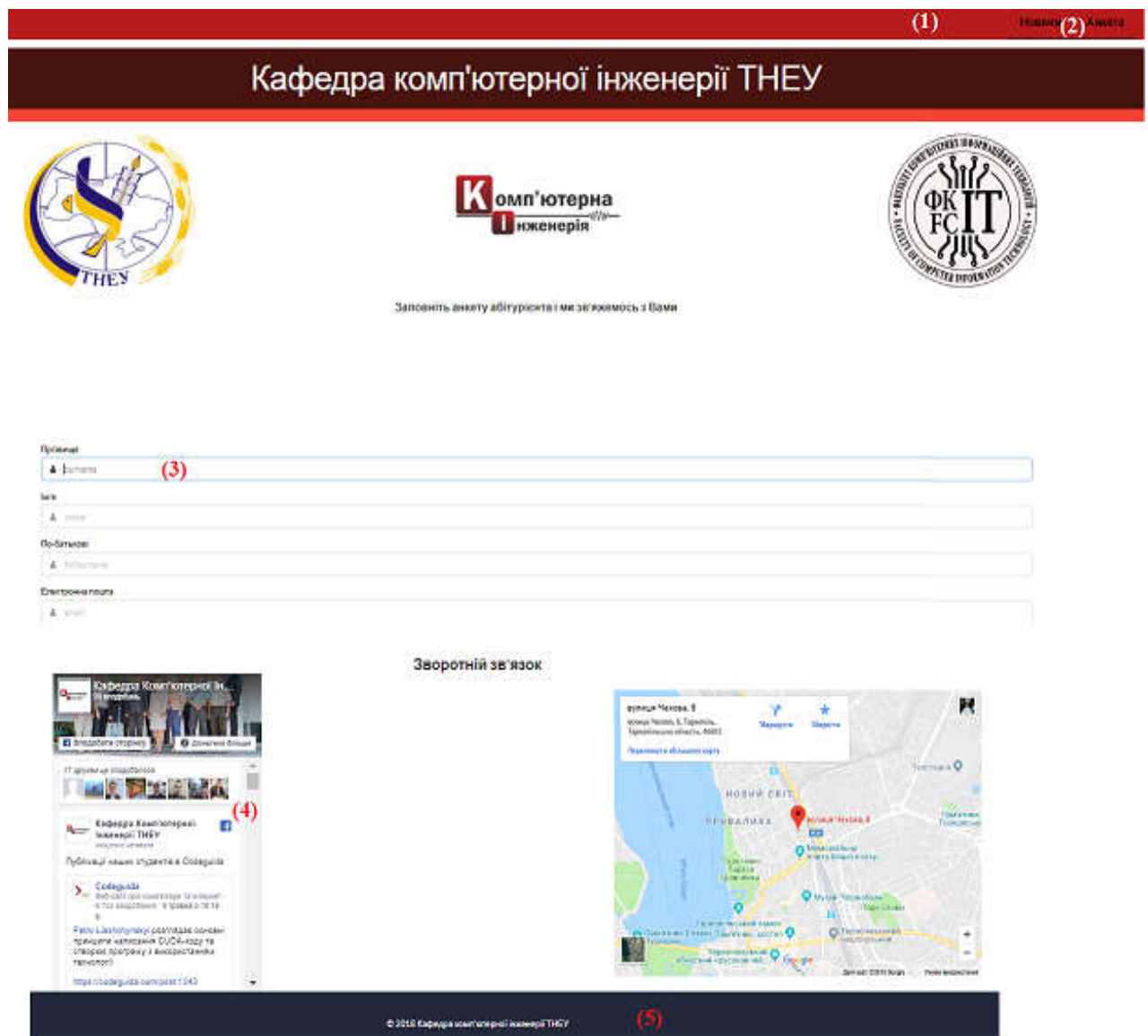


Рисунок 3.2 - Графічний інтерфейс головної сторінки сайту

Інтерфейс розроблюваного сайту є максимально простим та зрозумілим для будь-якого користувача. Також сайт не навантажений надмірною кількістю непотрібної інформації. У верхній частині головної сторінки розташована так звана (1) «шапка», яка дублюється на інших сторінках сайту. Це робиться спеціально, адже ця частина відображається у вікні браузера першою і відвідувач насамперед звертає увагу на неї. Щоб забезпечити швидкий перехід до основних тематичних розділів сайту, створене (2) меню сайту — список гіперпосилань на його розділи. Створене меню є зручним, помітним і зрозумілим, отож користувач

					ДП. КСМ. 07123/14.00.00.000 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		43

легко може перейти до потрібного йому розділу. Робота з системою не викликає у користувача ускладнень у пошуках необхідних елементів інтерфейсу. До того ж, при заповненні анкети (3) передбачено введення користувачем тільки мінімальної інформації. У нижній частині сайту знаходиться (4) посилання на сторінку кафедри комп'ютерної інженерії в соціальній мережі Facebook, звідки користувач може довідатись необхідну йому інформацію стосовно обраної спеціальності. «Футер» (6) є атрибутом будь-якого сайту та у даному випадку містить інформацію про рік розробки веб-ресурсу обліку абітурієнтів.

Існування сайту та його повноцінна робота неможливі без ряду регулярних заходів з його підтримки, а саме наповнення сайту новим контентом, редагування існуючих матеріалів та своєчасного оновлення необхідних компонентів і модулів, захисту від мережевих атак, резервного копіювання даних. Весь комплекс подібних робіт — це ніщо інше, як адміністрування сайту. Саме тому графічний інтерфейс сторінки адміністрування є не менш важливим. На рисунку 3.3 зображено графічний інтерфейс сторінки адміністрування.

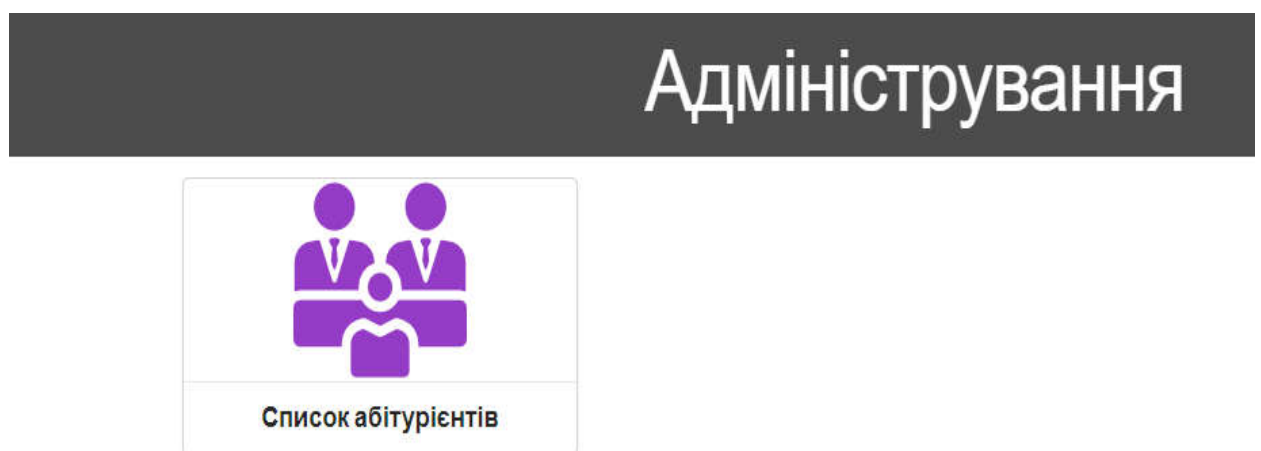


Рисунок 3.3 – Графічний інтерфейс сторінки адміністрування

На рисунку 3.3 зображено графічний інтерфейс сторінки відображення списку абітурієнтів.

					ДП. КСМ. 07123/14.00.00.000 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		44

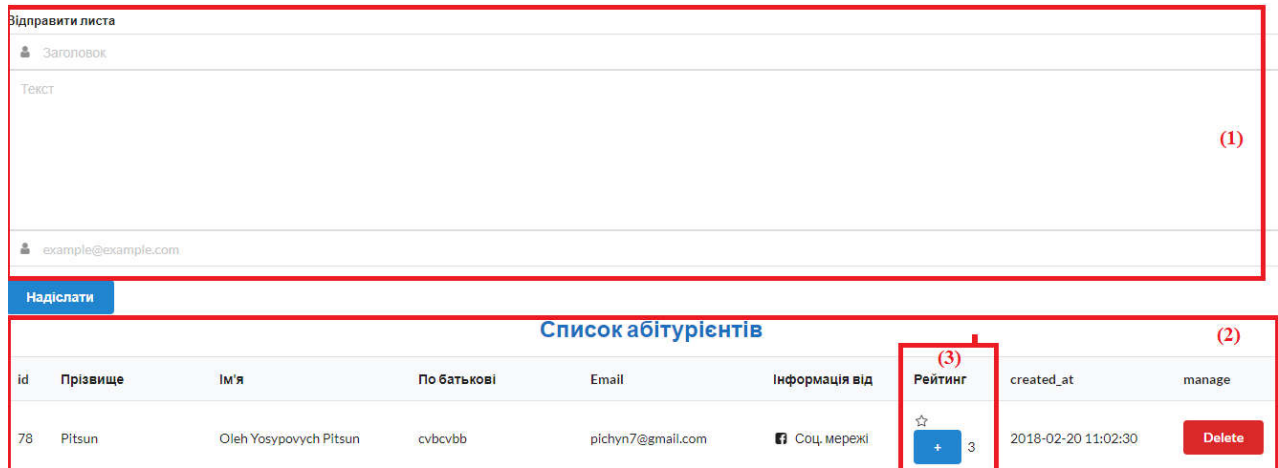


Рисунок 3.3 – Графічний інтерфейс сторінки відображення списку абітурієнтів

- (1) - модуль формування електронних повідомлень, що включає поля для вводу заголовку листа, тексту повідомлення та списку адрес для розсилки;
- (2) - модуль списку зареєстрованих абітурієнтів;
- (3) - блок формування рейтингу абітурієнта.

3.2 Модуль роботи з базою даних

База даних – це інструмент, використовуючи який, можна збирати й упорядковувати інформацію. Під час роботи з великими обсягами інформації значну роль приділяють організації самої бази даних. Наразі найбільш широко застосовується ієрархічна, мережева та реляційна моделі бази даних. Для розробки даного сайту було використано реляційну модель бази даних. Реляційна модель даних являє собою набір двовимірних таблиць, які складаються зі стовпців (полів) і рядків (записів), а також мають ім'я, унікальне в межах даної БД.

Для роботи з базою даних у даній дипломній роботі було використано систему керування базою даних MySQL. MySQL — вільна система керування реляційними базами даних. Характеризується високою швидкістю, стійкістю і простотою використання. Зараз MySQL — одна з найпоширеніших систем керування базами даних. Вона використовується, в першу чергу, для створення

динамічних веб-сторінок, оскільки має чудову підтримку з боку різноманітних мов програмування.

Для адміністрування сервера MySQL, перегляду та редагуванню вмісту таблиць баз даних було використано PhpMyAdmin. PhpMyAdmin дозволяє через браузер здійснювати адміністрування сервера MySQL, запускати запити SQL, переглядати та редагувати вміст таблиць баз даних. Ще однією вагомою перевагою використання PhpMyAdmin є можливість додавати, видаляти, редагувати вміст таблиць без використання SQL команд.

Приклад коду для налаштування доступу до БД:

```
'mysql' => [  
    'driver' => 'mysql',  
    'host' => env('DB_HOST', 'localhost'),  
    'port' => env('DB_PORT', '3306'),  
    'database' => env('DB_DATABASE', 'forge'),  
    'username' => env('DB_USERNAME', 'forge'),  
    'password' => env('DB_PASSWORD', ''),  
    'charset' => 'utf8',  
    'collation' => 'utf8_unicode_ci',  
    'prefix' => '',  
    'strict' => true,  
    'engine' => null,  
],
```

Приклад файл для формування запиту до БД такий:

```
class Abiturients extends Model  
{  
    protected $table = 'abiturients';  
    protected $fillable = [  
        'id', 'surname', 'name', 'fathername', 'email', 'phone', 'status', 'created_at', 'updated_at'  
    ]  
}
```

Клас «Abiturients» унаслідуює клас «Model». У масиві \$fillable знаходиться список полів таблиці бази даних 'abiturients', доступних для запиту та виводу на екран.

Лістинг файлу для отримання інформації з БД в контролері такий:

```
public function index()
```

					ДП. КСМ. 07123/14.00.00.000 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		46

```

{
    $abiturients = Abiturients::paginate(10);

    return view('abiturients.index', compact('abiturients'));
}

```

Метод `paginate()` відповідає за вибірку кількості записів з БД. У даному випадку – 10. За допомогою цієї функції можна здійснити посторінковий вивід інформації для зручності. У метод `compact()` передається масив з БД. На стороні клієнта у циклі відбувається вивід інформації за певним ключем.

Електронні поштові повідомлення є основним способом комунікації. Тому розробка функціоналу є важливим завданням. Приклад коду для формування електронних повідомлень:

```

public function build()
{
    $address = 'ki@gmail.com';
    $name = 'Кафедра комп'ютерної інженерії';
    $subject = 'Інформація для абітурієнтів';
    return $this->view('emails.ki-found')
        ->from($address, $name)
        ->cc($address, $name)
        ->bcc($address, $name)
        ->replyTo($address, $name)
        ->subject($subject);
}

```

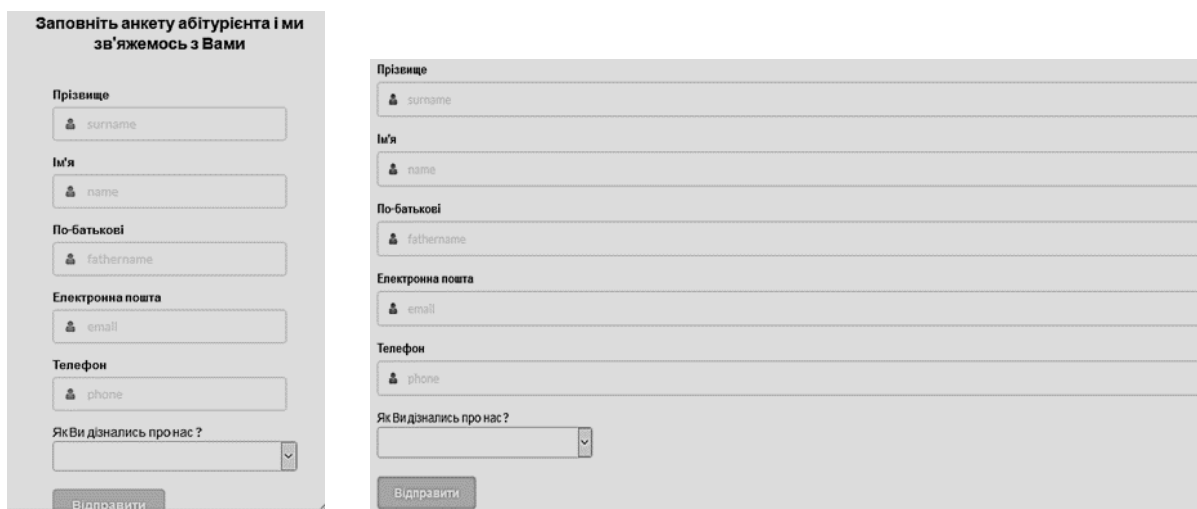
У полі `$address` вказується електронна адреса з якої відсилатимуться листи отримувачам. У полі `$name` вказується назва організації. Поле `$subject` відповідає за вміст листа. Вміст листа формується адміністратором у відповідному модулі.

3.3 Порівняльний аналіз розробленого сайту з аналогами

					ДП. КСМ. 07123/14.00.00.000 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		47

На сьогодні представлення організацій у мережі Інтернет є необхідним для покращення ефективності роботи організації, обміну інформацією між усіма учасниками ринкового процесу та способом заявити про свою діяльність великому загалу користувачів глобальної мережі. Отож не винятком є і навчальні установи. Оскільки абітурієнтів є чимала кількість, то все гострішою стає проблема автоматизованого опрацювання інформації. Саме тому зараз великого поширення набули системи обліку абітурієнтів.

Розроблений у даному дипломному проекті веб-ресурс обліку абітурієнтів має ряд переваг та недоліків. Серед переваг варто відзначити адаптивний інтерфейс. Адаптивний інтерфейс — це такий інтерфейс веб-сторінок, який забезпечує оптимальне відображення та взаємодію сайту з користувачем незалежно від роздільної здатності та формату пристрою, з якого здійснюється перегляд сторінки. Саме тому, щоб забезпечити максимальний комфорт було створено адаптивний інтерфейс сайту. Зовнішній інтерфейс сайту на моніторах із різними розширеннями наведено на рисунку 3.5.



а)

б)

Рисунок 3.5 - Зовнішній інтерфейс сайту на моніторах із різними розширеннями: а) 256 x 561; б) 1280 x 720

					ДП. КСМ. 07123/14.00.00.000 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		48

Як видно з рисунку 3.5 розроблений сайт є адаптивним під монітори з різними роширеннями.

Ще однією перевагою даного сайту є асинхронні запити до сервера. При асинхронному запиті користувач може продовжувати переглядати контент сайту, поки сервер все ще обробляє запит. Браузер не перезавантажує web-сторінку і дані посилаються на сервер без візуального підтвердження. В багатьох системах обліку абітурієнтів такої функції немає.

Не менш важливою перевагою даного сайту у порівнянні з аналогами є наявність рейтингу абітурієнтів. Адміністратор має можливість виставляти так звані очки (бали) абітурієнтам за участь в олімпіаді, зв'язок з адміністрацією кафедри і таким чином формувати рейтинг. Також реалізовано сортування за оцінками. Крім цього адміністратор може переглядати зареєстрованих абітурієнтів та здійснювати розсилку електронних повідомлень.

Порівняльний аналіз робробленого сайту із аналогами наведено у таблиці 3.2.

Таблиця 3.2 - Порівняльний аналіз робробленого сайту із аналогами

Критерій	Сайт ДонНАБА	Сайт ЗДМУ	Розроблений сайт
Адаптивний інтерфейс	-	+	+
Асинхронні запити до сервера	-	-	+
Перегляд зареєстрованих абітурієнтів	+	+	+
Формування рейтингу абітурієнтів	-	-	+
Розсилка електронних повідомлень	+	+	+

Як видно з таблиці 3.2 розроблений веб – ресурс не поступається аналогам, а у більшості випадків переважає їх

					ДП. КСМ. 07123/14.00.00.000 ПЗ		Арк.
Змн.	Арк.	№ докум.	Підпис	Дата			49

Отже, у даному розділі представлено структуру серверної та клієнтської частини системи обліку абітурієнтів, графічний інтерфейс сторінки адміністрування, та головної сторінки сайту, модуль роботи бази даних, а також визначено переваги та недоліки розроблювального у даному дипломному проєкті веб-ресурсу обліку абітурієнтів кафедри комп'ютерної інженерії. Для цього було здійснено порівняльний аналіз за кількома параметрами розроблювального сайту із уже існуючими.

					ДП. КСМ. 07123/14.00.00.000 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		50

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

№ 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. забезпечення	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 4.1:

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

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 4.2:

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

де C_{ij}^0 – 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.).

					ДП. КСМ. 07123/14.00.00.000 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		52

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	14 402
3	Студент	1290	0	1290

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

$$B_{оп} = 16 * \frac{3440}{168} + 2 * \frac{14402}{168} + 254 * \frac{1290}{168} = 2449$$

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:

$$B_{\phi} = \frac{20,5}{100} * 2449 = 502,04$$

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.

					ДП. КСМ. 07123/14.00.00.000 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		53

Загальна сума витрат на матеріальні ресурси (B_M) визначається за формулою 4.3:

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

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:

$$B_M = 344,30 \text{ грн.}$$

Проведені розрахунки занесемо у таблицю 4.3

Таблиця 4.3 - Зведені розрахунки покупних виробів

№ п/п	Найменування купованих виробів	Одиниця виміру	Ціна, грн	Кількість купованих виробів	Сума, грн	Транспортні витрати (10% від суми)	Загальна сума, грн
1	Папір (формат А4)	уп	85,0	2	170,00	17,0	187,0
2	Ручка кулькова	шт	10,0	2	20,00	2,0	22,0
3	Диски CD-R	шт	9,0	1	9,00	0,90	9,90
4	Зошит, 96 арк	шт	24	1	24	2,4	26,40
5	Тонер для принтера	уп	90	1	90	9,0	99,0
Разом							344,30

4.1.4 Розрахунок витрат на електроенергію

Для розробки КС використовується електрообладнання, тому необхідно розрахувати витрати на електроенергію.

Загальна сума витрат на електроенергію розраховується за формулою 4.4:

$$B_E = \sum_{i=1}^n P_i \cdot k_i \cdot T_i \cdot C, \quad (4.4)$$

де P_i - паспортна потужність i -го електрообладнання, кВт;

k_i - коефіцієнт використання потужності i -го електрообладнання;

T_i - час роботи i -го устаткування за весь період розробки, год;

C - ціна електроенергії, грн / кВт· год; 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 4.5:

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

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.

Для проектування даної системи використовувався один ноутбук 6300грн., та принтер вартістю 1150 грн, а також інші пристрої описані вище, та подані в таблиці 4.5 і обрахування витрат на амортизацію.

Тоді:

$$B_{AM} = \frac{6300 \cdot 10}{100} + \frac{1150 \cdot 20}{100} = 860$$

Таблиця 4.5 - Амортизація основних фондів

Найменування устаткування	Вартість устаткування, грн.	Річна норма амортизації, %	Сума, грн.
Ноутбук	6300,00	10	630,00
Принтер	1150,00	20	230,00
Разом			860,00

4.1.6 Визначення транспортних витрат

Транспортні витрати слід прогнозувати у розмірі 8–12 % від загальної суми матеріальних витрат. Транспортні витрати розраховуються за формулою 4.6:

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

де B_T – транспортні витрати.

$$B_T = 0,12 \cdot 344,30 = 41,31 \text{ грн.}$$

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 обчислюються за формулою 4.7:

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

де H_B – накладні витрати.

$$H_B = 1,5 \cdot 2\,449 = 3\,673,5 \text{ грн.}$$

4.1.8 Обчислення інших витрат

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

$$I = 2\,449 \cdot 0,1 = 244,9 \text{ грн}$$

4.1.9 Складання кошторису витрат та визначення собівартості

					ДП. КСМ. 07123/14.00.00.000 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		57

Загальні витрати (B_{KC}) розраховуємо за формулою 4.8

$$B_{KC} = B_{OP} + B_{\Phi} + B_M + B_E + B_{AM} + B_T + H_B \quad (4.8)$$

Тобто:

$$B_{KC} = 7959 \text{ грн.}$$

Результати проведених розрахунків зведемо у таблицю 4.6.

Таблиця 4.6 - Кошторис витрат

Зміст витрат	Сума, грн.
Витрати на оплату праці (осн. і дод. ЗП)	2 449
Відрахування на соціальні заходи	502,04
Матеріальні витрати	344,30
Витрати на електроенергію	88,85
Амортизаційні відрахування	860,00
Транспортні витрати	41,31
Накладні витрати	3 673,5
Інші витрати	244,9
Разом	8 203,9

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): Експлуатаційні витрати обчислюються за формулою 4.9:

					ДП. КСМ. 07123/14.00.00.000 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		58

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

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

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

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

Річні експлуатаційні витрати B_{En} визначаються за формулою 4.10

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

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

Вартість підготовки даних для роботи на комп'ютері визначається за формулою 4.11:

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

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 - середньогодинна 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.

					ДП. КСМ. 07123/14.00.00.000 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		59

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:

$$c_i = \frac{3200(1 + 0,57)}{22 * 8} = 28,54$$

Трудомісткість підготовки даних для проектного рішення складає 1 год., для аналога 1,5 год. Розрахунок витрат на підготовку даних та реалізацію проектного рішення на комп'ютері представлено в таблиці 4.1

Таблиця 4.1 - Розрахунок витрат на підготовку даних та реалізацію проектного рішення на комп'ютері

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

Витрати на експлуатацію комп'ютера визначається за формулою 4.13:

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

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

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

$$E_{2n} = 1 * 0,9 = 0,9 \text{ грн.}; E_{2a} = 1,5 * 0,9 = 1,35 \text{ грн.}$$

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

$$B_{en} = 13,2 * 252 = 3\,807,72 \text{ грн.}; B_{ea} = 19,75 * 252 = 5\,700,24 \text{ грн.}$$

					ДП. КСМ. 07123/14.00.00.000 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		60

4.3 Розрахунок ціни споживання проектного рішення

Ціна споживання - це витрати на придбання і експлуатацію проектного рішення за весь строк його служби. Ціна споживання визначається за формулою 4.14:

$$C_{C(n)} = C_n + B_{(E)NPV} \quad (4.14)$$

де C_n - ціна придбання проектного рішення, грн.

Ціна придбання проектного рішення розраховується за формулою 4.15:

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

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 4.16:

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

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

p - середній рівень рентабельності, % (приймаємо 30% за погодженням з керівником).

$$C_d = 8\,203,9 \cdot (1+0,3) = 10\,158,14 \text{ грн.}$$

					ДП. КСМ. 07123/14.00.00.000 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		61

Вартість витрат на експлуатацію проектного рішення (за весь час його експлуатації), розраховується за формулою 4.17:

$$B_{\text{епрв}} = \sum_{t=0}^T \frac{B_{\text{ен}}}{(1+R)^t}, \quad (4.17)$$

де $B_{\text{ен}}$ - річні експлуатаційні витрати, грн.;

T - строк служби проектного рішення, років;

R - річна ставка проценту банку.

$$B_{\text{епрв}} = \sum_{t=1}^5 \frac{3\,807,72}{(1+0,08)^t} = 17\,628,3 \text{ грн.}$$

$$B_{\text{епрв}} = \sum_{t=1}^5 \frac{5\,700,24}{(1+0,08)^t} = 28\,275,2 \text{ грн.}$$

Тоді ціна споживання проектного рішення дорівнюватиме:

$$Ц_{\text{сн}} = 10\,158,14 + 17\,628,3 = 27\,786,44 \text{ грн.}$$

Аналогічно визначається ціна споживання для аналогу:

$$Ц_{\text{са}} = 9\,500,0 + 28\,275,2 = 37\,775,20 \text{ грн.}$$

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

					ДП. КСМ. 07123/14.00.00.000 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		62

виробництва узагальнюючий показник економічної ефективності — норма прибутку.

Економічний ефект в сфері проектування рішення обчислюється за формулою 4.18:

$$E_{PP} = C_{II} - C_A \quad (4.18)$$

Тоді:

$$E_{PP} = 10\,158,14 - 9\,500,0 = 658,14 \text{ грн.}$$

Річний економічний ефект в сфері експлуатації розраховується за формулою 4.19:

$$E_{KC} = B_{EA} - B_{EP} \quad (4.19)$$

Тоді:

$$E_{KC} = 5\,700,24 - 3\,807,72 = 1\,892,52 \text{ грн.}$$

Додатковий економічний ефект у сфері експлуатації обчислюється за формулою 4.20:

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

Тоді:

$$\Delta E_{ekc} = \sum_{t=1}^5 1892,52(1 + 0,08)^{5-t} = 10219,61 \text{ грн.}$$

					ДП. КСМ. 07123/14.00.00.000 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		63

Сумарний ефект визначається за формулою 4.21:

$$E = E_{np} + \Delta E_{eks} \quad (4.21)$$

Тоді:

$$E = 10\,158,14 + 10\,219,61 = 20\,377,75 \text{ грн.}$$

В даному розділі проведено розрахунок витрат на розробку проектного рішення. Здійснено порівняння з існуючим аналогом, і цим показано, що дане проектне рішення має переваги в порівнянні з аналогами. Згідно проведеного економічного обґрунтування дане проектне рішення є конкурентоздатним.

					ДП. КСМ. 07123/14.00.00.000 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		64

ВИСНОВКИ

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

1. Проаналізовано програмні засоби проектування та розробки веб-сайтів, що дозволило обрати програмний засіб для реалізації веб-ресурсу обліку абітурієнтів з використанням сучасних інформаційних технологій;

2. Проведено аналіз та класифікацію веб-ресурсів мережі Інтернет, що дозволило виділити основні функції, які повинні бути присутні на сучасних веб-ресурсах;

3. Досліджено електронні системи обліку користувачів, що дозволило виділити основні задачі, які вирішуються за допомогою програмних засобів даного типу;

4. Розроблено алгоритм реєстрації абітурієнтів та проведено ранжування прав доступу користувачів до електронної системи обліку абітурієнтів, що дозволило провести розподіл функціональних обов'язків різних груп користувачів;

5. Розроблено структуру реляційної бази даних веб-ресурсу обліку абітурієнтів та проведено її нормалізацію, що дозволило реалізувати модуль обробки даних в електронній системі обліку абітурієнтів;

6. Реалізовано веб-ресурс обліку абітурієнтів кафедри комп'ютерної інженерії, проведено тестування та порівняно результати роботи з аналогами.

Основною перевагою розробленого у даному дипломному проекті веб-ресурсу обліку абітурієнтів є адаптивний інтерфейс та формування рейтингу абітурієнтів.

					ДП. КСМ. 07123/14.00.00.000 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		65

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. PHP: Мова програмування [Електронний ресурс] Режим доступу: <http://www.php.net>.
4. Лавріщева К.М. Програмна інженерія / - К.: Видавництво «Академперіодика», 2008.- 319 с.
5. Ларман К. Применение UML и шаблонов проектирования. / - М.: Издат. дом «Вильямс», 2002. - 617 с.
6. Вендров А.М. CASE-технологии. Современные методы и средства проектирования информационных систем. - М.: Финансы и статистика, 1998. – 412 с.
7. Липаева В. В. 'Программная инженерия. Методологические основы' Соммервилл, Иан. Инженерия программного обеспечения, 6-е издание. : Пер. с англ. - М. : Издательский дом «Вильямс», 2002. - 624 с.
8. Петюшкин А. В. HTML в Web-дизайне. СПб.: БХВ-Петербург, 2004.- 400 с.
9. Лекае В. А. Некоторые аспекты разработки, создания и эксплуатации web-сайтов и порталов / В.А. Лекае, М. А. Григорьева // Межотраслевая информационная служба. – 2007. - №2. – С. 44-47.
10. Алексеев А.П. Введение в Web-дизайн / А.П. Алексеев - М.: СОЛОН-ПРЕСС. – 2008. – 192 с.
11. Гаевский А.Ю. 100% самоучитель. Создание Web-страниц и Web-сайтов. HTML и JavaScript / А. Ю. Гаевский, В.А. Романовский. - М.: Триумф, 2008. - 464 с.
12. Салий В.В. Роль веб-сайтов фирм в информационном обслуживании специалистов / Культурная жизнь Юга России. – 2010. - №1. – С.101-102.

					ДП. КСМ. 07123/14.00.00.000 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		66

13. Браилов А.Ю. Подход к созданию web-сайта. / Труды Одесского политехнического университета. – 2005. - №1. – С. 88-92 с.
14. Крекин М. Е. Разработка веб-страниц: современный подход к обучению. / Информатика и образование. – 2007. - №2. – С. 109-119.
15. Мальчук Е. В. HTML и CSS. Самоучитель / - Вильямс. – 2006. – 416 с.
16. Ульман, Л. Основы программирования на PHP: [пер. с англ.] / - М.: ДМК, 2001. – 288 с.
17. Електронна енциклопедія Вікіпедія: Веб-сайт [Електронний ресурс] Режим доступу: <http://uk.wikipedia.org/wiki/Веб-сайт>.
18. Електронна енциклопедія Вікіпедія: Веб-додаток [Електронний ресурс] Режим доступу: <http://uk.wikipedia.org/wiki/Веб-додаток>.
19. PHP: Мова програмування [Електронний ресурс] Режим доступу: <http://www.php.net>.
20. Хокінс С. Администрирование веб-сервера Apache и руководство по электронной коммерции. / М.: Вильямс, 2001. - 336 с.
21. Єрмолаєв В. А. Програмне забезпечення ЕОМ. Структури даних та алгоритми / В.А. Єрмолаєв, О. Р. Онищук - Запоріжжя: ЗНУ, 2005. – 110 с.
22. Фролов А. В. Практика применения PERL, PHP, APACHE и MySQL для активных Web-сайтов / А. В. Фролов, Г. В.Фролов.- М.: Русская редакция, 2002. – 534 с.
23. Шафран Е.М Створення Web-сторінок. /- СПб.: Питер, 2001. –320 с.
24. Кузнецов М.В. PHP 5 на примерах / М.В Кузнецов, И.В. Симдянов, С.В. Гольшев. – СПб.: БХВ-Петербург, 2005. – 577 с.
25. Шлоснейгл Д.Л. Профессиональное программирование на PHP / - М.: Русская редакция, 2006 – 624 с.
26. Грубер М.О. Понимание SQL / – Донецк: ООО "Юго-Восток", 2002. – 234 с.
27. Федорчук А.І. Як створюються Web-сайти. / – СПб.: Пітер, 2000. – 224 с.
28. Аткинсон Л.П. Mysql. Библиотека профессионала / – М.: Энергоатомиздат, 2002. – 496 с.

					ДП. КСМ. 07123/14.00.00.000 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		67

29. Дейт К.Н. Руководство по реляционной СУБД DB2. /– М.: Финансы и статистика, 1988. – 320 с.
30. Мейер М.К. Теория реляционных баз данных /– М.: Мир, 1987. – 608 с.
31. Електронна енциклопедія Вікіпедія: Реляційні БД [Електронний ресурс] Режим доступу:[http://ru.wikipedia.org/wiki/ Реляційна БД](http://ru.wikipedia.org/wiki/Реляційна_БД) .
32. Дейт К. Дж. Введение в системы баз данных /– СПб: Изд-во "Пітер", 2005. – 1315 с.
33. Коннолли Т.А. Базы данных. Проектирование, реализация и сопровождение. Теория и практика / Т.А. Коннолли, К.М Бегг. – 3-е изд. – М. : Вильямс, – 2003. – 1436 с.
34. Коннолли Т.А. Базы данных: проектирование, реализация и сопровождение / Т.А Коннолли, К.М Бегг, А.О Страчан. – 2-е изд. – М. : Вильямс, 2000. – 1120 с.
35. Камер Д. Компьютерные сети и Internet / – М. : Вильямс, 2002. – 640 с.
36. Андон Ф.И. Информационные системы / Ф.И. Андон, В.П. Резниченко, У.У. Яшунин – К., 2001. – 396 с.
37. Дакетт Дж. HTML и CSS. Разработка и дизайн веб-сайтов / - К.: Эксмо - Москва, 2013. - 480 с.
38. Дронов В. PHP 5/6, MySQL 5/6 и Dreamweaver CS4. Разработка интерактивных Web-сайтов /- БХВ-Петербург - Москва, 2009. - 544 с.
39. Гаевский А.Ю. 100% самоучитель. Создание Web-страниц и Web-сайтов. HTML и JavaScript / А.Ю. Гаевский, В.А. Романовский. - М.: СПб. [и др.] : Питер, 2008. - 464 с.
40. Гарнаев А.М. WEB-программирование на Java и JavaScript / А.М Гарнаев , С.М Гарнаев. /- М.: БХВ-Петербург, 2005. - 822 с.
41. Петюшкин А.В. HTML экспресс-курс / - М.: СПб: БХВ-Петербург, 2004. - 250 с.
42. Поломошнов О.Б. Быстро и легко создаем, программируем и раскручиваем Web-сайт / - М.: Эксмо, 2011. - 352 с.

					ДП. КСМ. 07123/14.00.00.000 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		68

43. Круш П.В. Організація виробництва / П.В. Круш, В.І.Подвігіна, В.О.Гулевич. - К.: ЦУЛ, 2008.- 348 с.
44. Герасимчук В.Г. Економіка і організація виробництва / В.Г. Герасимчук, А.Е.Розенплентер, В.І. Кривда. /– К.: Політехніка, 2007р.
45. Подвігіна В.І. Організація виробничого процесу в часі та просторі. Потокове виробництво./ – К.: ЦУЛ, 2007. – 136 с.
46. Курочкин А.С. Організація производства, Учебн.пособие для студентов вузов. /– К.: МАУП, 2001. – 216 с.
47. Багрова І.В. Організація виробництва /- Київ, ЦНЛ, 2005. – 248 с.
48. 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.- 36 с.
49. 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» / О.М. Березький, Л.О. Дубчак, Р.Б. Трембач, Г.М. Мельник, Ю.М. Батько, С.В. Івасьєв / Під ред. О.М. Березького. – Тернопіль: ТНЕУ, 2016. – 60 с.

					ДП. КСМ. 07123/14.00.00.000 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		69