

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

Сич Тарас Андрійович

**Програмна система комунікації у навчальному
процесі/ Software system of communication in
the educational process**

Виконав: студент групи КІ-41
Сич Тарас Андрійович

Науковий Керівник
к.т.н., Савка Н. Я.

ТЕРНОПІЛЬ-2023

РЕЗЮМЕ

Кваліфікаційна робота на тему «Програмна система комунікації у навчальному процесі» зі спеціальності 123 «Комп'ютерна інженерія» освітнього ступеня «бакалавр» містить 55 сторінок пояснюючої записки, 50 рисунків, 3 додатки. Обсяг графічного матеріалу 2 аркуші формату А3.

Метою кваліфікаційної роботи є розроблення допоміжної системи навчання на основі навчальних програм для студентів та інших осіб.

Методи дослідження включають методи навчання у дистанційній формі та дослідження методів самонавчання, методи навчання у навчальних платформах із різним нахилом інструментарію та навантаженню.

Дослідження успішності навчання, що посідають проблемою під час дистанційного навчання. Розроблено алгоритм, аналізу дистанційного навчання за допомогою освітніх платформ в режимі онлайн. Проведено аналіз необхідних елементів, які завжди позитивно впливають на покращення навчання та покращує швидкість засвоєння інформації. Перевагою створеного алгоритму є незалежність від навчального процесу в структурі навчання та створення власного режиму.

Розроблене програмне забезпечення для дистанційної форми навчання у навчальному процесі; що реалізує наступні функції: безпека даних користувачів; зберігання всіх даних у потрібному вигляді; збереження необхідних інтернет джерел; швидкість пошуку даних, інструменти для побудови проєктів та створення плану побудови, публікацію власних робіт та елемент комунікації у вигляді месенджера.

Ключові слова: НАВЧАЛЬНА ПЛАТФОРМА, ДИСТАНЦІЙНЕ НАВЧАННЯ, ЗБЕРІГАННЯ ДАНИХ.

RESUME

Qualification thesis “Software for structured network cable system designing” in the specialty 123 "Computer Engineering" of bachelor education degree contains 55 pages of explanatory notes, 19 figures, 12 tables, 3 appendixes. The volume of graphic material is 2 sheets of A3 format.

The purpose of the qualification work is the development of an auxiliary system of training based on training programs for students and other persons.

Research methods include distance learning methods and research on self-learning methods, learning methods in educational platforms with different inclination of tools and workload.

Studying the success of learning, which is a problem during distance learning. An algorithm for the analysis of distance learning using online educational platforms has been developed. An analysis of the necessary elements, which always have a positive effect on improving learning and improving the speed of assimilation of information, has been carried out. The advantage of the created algorithm is its independence from the educational process in the structure of education and the creation of its own mode.

Developed software for distance learning in the educational process; which implements the following functions: security of these users; storage of all data in the required form; preservation of necessary Internet sources; the speed of data search, tools for building projects and creating a construction plan, publishing one's own works and an element of communication in the form of a messenger.

Keywords: EDUCATIONAL PLATFORM, DISTANCE LEARNING, DATA STORAGE.

ЗМІСТ

Вступ.....	9
1 Аналіз навчальних платформ комунікаційних систем.....	12
1.1 Особливості навчального процесу при використанні навчальних платформ	12
1.2 Аналіз існуючих програмних систем із навчальними елементами	14
1.3 Аналіз програмних засобів	20
1.4 Постановка задачі кваліфікаційної роботи	24
2. Алгоритм розробки системи комунікацій у навчальних платформах.....	26
2.1 Етапи створення навчальної системи із певними технологіями	26
2.2 Аналіз вимог до програмного забезпечення для розробки освітніх систем ..	30
2.3 Алгоритм розробки системи комунікацій у навчальному процесі.....	37
3. Програмна реалізація алгоритму роботи системи комунікації.....	45
3.1 Структура програмного забезпечення навчальної системи	45
3.2 Реалізація системи комунікації	62
3.3 Експериментальні дослідження системи комунікації.	67
4 Техніко-економічний розділ.....	75
4.1 Визначення витрат на оплату праці та відрахувань у соціальні фонди	75
4.2 Розрахунок ціни проекту	83
4.3 Визначення економічної ефективності роробки проекту	85
Висновки.....	87
Список використаних джерел.....	88
Додаток А Вихідний текст програмного коду goom.php.....	93
Додаток Б Довідка про використання	95
Додаток В Світлокопії виданих публікацій.....	96

					КР.КІ. 8351649.00.00.000 ПЗ			
Змн.	Лист	№ докум.	Підпис	Дата				
Розробив	Сич Т.А.				Програмна система комунікації у навчальному процесі	Літ.	Арк.	Акрушів
Перевір.	Мельник Г.М.					7		
Консульт.	Савка Н.Я.					ЗУНУ,ФКІТ, КІ-41		
Н. Контр.	Мельник Г.М.							
Затвердив								

ВСТУП

Навчання для кожного відбувається по-різному, але якщо розібратись, то воно необхідне кожному. Із певними навичками ставати кращим і приносити користь суспільству. В давні часи терміни наука і навчання поширювались далеко не для всіх. Для більшості це було розкішшю, якою не могли скористатись усі. Більше використовували такий метод заможні родини. Тому популяція навчання була не сильно розвинена та існувала дуже велика різниця між багатьма людьми.

З часом ситуація почала змінюватись, але не завжди у користь звичайних людей. В більшості випадків люди так і залишались без освіти чи елементарних знань. Зараз це не так виглядає, але в давнину дуже цінувались люди, які мали певні знання в багатьох областях. Хорошим прикладом слугує Стародавня Греція. Ще у ті часи зрозуміли користь навчання і те, що потрібно все своє життя вчитись, для того, щоб ставати кращим та розуміти себе. Як говорив Арістотель: «...навчання не є життєвою функцією, оскільки не має самоцілі; життєва функція – це вчитися». Згідно цієї цитати, ще в ті часи розуміли наскільки важливо весь час пізнавати нове і вчитись[1,3,5].

Навчання завжди є складною річчю, тому воно було настільки непоширеним. Якщо була можливість навчатись, то це не означало, що навчання могло легко вдаватись. В більшості випадків лише одиниці могли зрозуміти саму суть навчання і матеріалу, який потрібно засвоїти учневі. На жаль, в давнину не сильно приділяли увазі тому, чи всі учні зрозуміли матеріал. З часом почали використовувати різні методи навчання. Уже прийшло розуміння, що кожна людина є індивідум і не завжди один метод навчання може підійти усім. Тому з часом на підсвідомому рівні люди почали пристосовуватись і самостійно уже використовували різні методи навчання. Також ця тема не обійшла і різних вчених, які хотіли детально дослідити це питання. Таким чином, існують наступні методи навчання:

					КР.КІ. 8351649.00.00.000.ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		9

- словесні;
- наочні;
- практичні.

Кожен із цих підвидів можна більше розібрати, але якщо для загального розвитку, то слід пам'ятати, що багато із цих методів змішують для покращення і подвоєння ефекту. Як завжди, спочатку теоретичний матеріал потрібно прочитати і завчити. Після цього, як матеріал засвоєно, слід його використовувати на практиці. Згідно вищезгаданого, навички закріплюються, і таким чином ми навчаємось. В будь-якій сфері слід спочатку зрозуміти саму суть, а після цього пробувати самостійно повторити те, що викладено в теорії.

В сучасному світі середньостатистична людина повинна пройти певні етапи навчання, щоб могла функціонувати і співіснувати із суспільством. Певні правила етикету і порядку повинні бути зрозумілі для всіх, щоб був порядок і можна було жити в мирі [2, 6, 3].

Отже, метою кваліфікаційної роботи є розробка програмного засобу, за допомогою якого можна здобувати знання та практичні навички у певних областях з можливістю комунікації і обміном інформацією між учасниками навчального процесу.

Для досягнення поставленої мети потрібно виконати нижчезазначені задачі:

1. проаналізувати особливості дистанційного навчання і переваги навчальної платформи;
2. дослідити методи та алгоритми роботи навчальної платформи;
3. розробити алгоритм навчання та показник успішної роботи платформи;
4. розробити програмний модуль для реалізації алгоритму;
5. здійснити реалізацію розробленого програмного модуля навчальної платформи;

					КР.КІ. 8351649.00.00.000.ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		10

6. дослідити алгоритм роботи платформи та оцінити результати навчання, а саме: успішність, швидкість засвоєння інформації, практичні навички;

7. обґрунтувати техніко-економічні показники розробленого програмного забезпечення.

Об'єктом дослідження процесу комунікації при навчанні.

Предмет дослідження – алгоритм розробки системи комунікації на основі веб-інтерфейсу і бази даних.

Практичне значення отриманих результатів. Програмна система комунікації у навчальному процесі дозволяє корегувати режим роботи у навчальному процесі, створюючи зручний графік, що дозволяє не понижувати показник успішності та створити комфортні умови для навчання[5, 6, 7].

За результатами кваліфікаційної роботи опубліковано тези доповіді на VII науково-практичній конференції «Інтелектуальні комп'ютерні системи та мережі» [14]. Копії публікації наведено у додатку В.

					КР.КІ. 8351649.00.00.000.ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		11

1 АНАЛІЗ НАВЧАЛЬНИХ ПЛАТФОРМ КОМУНІКАЦІЙНИХ СИСТЕМ

1.1 Особливості навчального процесу при використанні навчальних платформ

Протягом останніх декількох років дуже поширений метод навчання полягає не в школах чи університетах. Один із різних видів інновацій в організації освіти в вищих закладах є застосування такого виду, як дистанційне навчання. Останні роки студентам потрібно вчитись, але враховуючи події які відбуваються в Україні – це важко реалізувати в реальному часі. Тому для того, щоб рівень навчання зберігав свою планку, вирішили використовувати різноманітні системи, які підтримують дистанційне навчання. Насамперед, дистанційне навчання орієнтується на навчальний процес, в який впроваджено нові моделі навчання, що передбачають проведення занять у дистанційному режимі. В такому режимі студенти виконують самостійно роботу, використовуючи різноманітні інформаційні ресурси та інші види комунікації, які підтримують інформаційно-телекомунікаційну технологію.

В час інформаційних технологій і технологічного прогресу, навчатись, використовуючи інформаційні технології стає дедалі простіше і доступніше. Головна перевага такого навчання полягає у її зручності. Студент сам вирішує, де йому навчатись, що дозволяє альтернативно працювати чи перебувати зовсім в іншій частині міста, чи навіть іншій країні[13,14,15]. Також матеріал і конспектування. Студенту не обов'язково записувати увесь матеріал на лекціях. Вся інформація, яку викладачі викладають на лекціях, розміщена на навчальних платформах у документах. Студент в будь-який зручний момент може самостійно опрацювати ці матеріали, і відповідно уже на лекціях запитувати лише те, що йому не зрозуміло, для додаткової інформації. Таким чином, можна виділити переваги зазначеної системи навчання:

– Для того, щоб навчатись і засвоювати матеріал кожному потрібна різна кількість часу. Якщо навчатись дистанційно у онлайн-платформі, то студент

					КР.КІ. 8351649.00.00.000.ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		12

сам може вирішувати, скільки часу приділяти для підготовки і навчання у семестрі.

– Одна умова дистанційного навчання полягає у наявності інтернету і гаджету, з якого можна навчатись. Із цього можна зрозуміти, що навчатись можна будь-де і в будь-який час. Також неможливість бути присутнім в аудиторіях полегшує навчання людям із обмеженим станом здоров'я, адже вже не потрібно бути присутнім на лекціях і практичних, щоб отримати необхідний матеріал.

– Також, якщо студент уже працює, то йому не потрібно брати відпустку чи відгул, щоб навчатись. Навчання тепер можна поєднувати із роботою та розподіляти час так, щоб дозволяти їм комфортно співіснувати.

– Доступність навчальних матеріалів є дуже великим плюсом у навчанні. При дистанційному навчанні усі матеріали чи посилання повинні бути розміщені в курсі чи семестрі. Відповідно для того, щоб вивчити матеріал, необхідно лише виділити час, а це може бути в будь-який момент доби.

– Також не потрібно забувати, що таке навчання є зручним не тільки студентам. Викладачі також з цього можуть винести багато плюсів. Усі, хто займаються викладацькою діяльністю, можуть навчати більше студентів і надавати консультації більшій кількості людей. Також, як і зазначалось, – це можна робити у будь-якій точці планети при наявності інтернету.

Для того, щоб комфортно навчатись, звичайно, можна використовувати і багато інших методів, але якщо глянути з іншої сторони, то можна виділити й певні недоліки. Щоб навчатись дистанційно, також необхідно долати певні труднощі, які будуть виникати при виконанні завдань чи у процесі навчання. До недоліків належать:

– для того, щоб постійно навчатись і приділяти багато часу навчальному процесу, необхідна велика мотивація. В сучасному світі є настільки багато цікавих речей, що дуже легко можна перевести увагу на щось більш захоплююче, ніж навчання;

					КР.КІ. 8351649.00.00.000.ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		13

– наступний мінус полягає в практичності. Є багато сфер навчання, такі як : медицина, менеджмент і т.д. В таких сферах необхідна практика або робота із людьми. Дистанційне навчання на даний момент не може цього надати, відповідно не зможе в повній мірі дати повну освіту;

– якщо сфера діяльності, в якій потрібно мати вміння спілкування з людьми, то дистанційне навчання тут також не зможе допомогти. Можна практикуватись через веб-ресурси, але вони кардинально відрізняються у режимі реального часу. Комунікабельність потрібно розвивати на практиці із реальними людьми;

– також викладачам набагато складніше прослідкувати за успішністю студента і те, що він може знати, а в чому він суттєво просідає. Дистанційне навчання обмежує співпрацю між студентами і викладачами, які є необхідні при навчальному процесі.

Зважаючи на вищепроведений аналіз, у наступному підрозділі розглянемо детально існуючі програмні системи, що уможливають здійснювати навчання.

1.2 Аналіз існуючих програмних систем із навчальними елементами

На сьогоднішній день існує багато навчальних платформ із різними напрямками. Основними діями, які необхідні для того, щоб почати вчитись – це знайти зручний для себе сайт і зареєструватись. Є багато платформ, де окрім платних є і частина безкоштовних.

Одним із найпопулярніших навчальних платформ можна обрати сайт prometheus.org.ua. На цій платформі є досить багато курсів і програм, за допомогою яких можна вивчити багато різних дисциплін і предметів. На цій платформі також приступні не тільки курси по ІТ-специфіці. Якщо перейти у вкладку курси, то можна побачити, що в переліку є багато різних видів і сфер, які можна вивчити. В такі види курсів входять:

					КР.КІ. 8351649.00.00.000.ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		14

- бізнес;
- психологія;
- особистий розвиток;
- ІТ;
- іноземні мови.

Вищенаведений список не є повний, але є одним із найпопулярніших. На даний момент на сайті згідно статистики, яку адміністратори сайту викладають на головній сторінці, то в загальному кількість слухачів налічує 2 300 000. В загальному – це достатньо багато і, відповідно, платформа користується дуже великою популярністю. Також одним із нюансів в навчанні є те, що багато курсів вимагає уже мати початкові знання в тій чи іншій сфері навчання. Відповідно, новачкам буде складно освоїтись, що може й відлякати на перший погляд. На рисунку 1.1 можна поспостерігати зовнішній дизайн і початковий функціонал платформи.

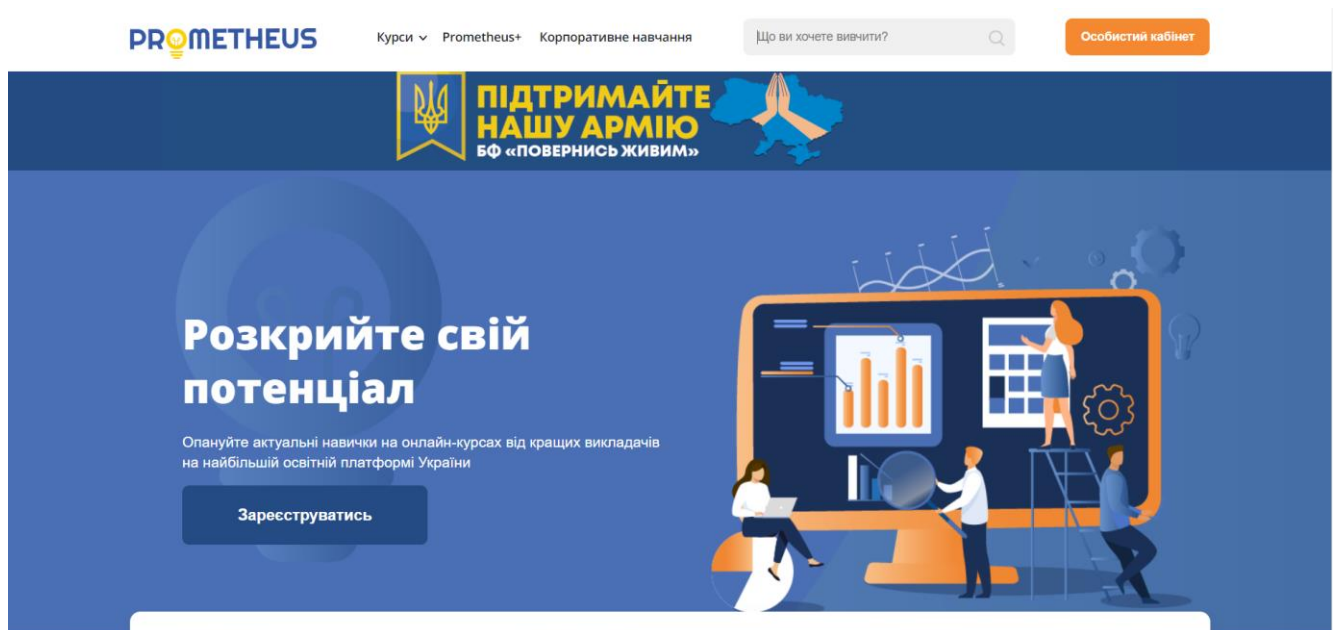


Рисунок 1.1 – Зовнішній вигляд платформи PROMETHEUS

Не заважаючи на те, що платформа є досить популярна через те, що там є велика кількість безкоштовних курсів, також є певні сторони, які можуть трохи

					КР.КІ. 8351649.00.00.000.ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		15

ускладнити навчання або подальші процеси. Нижче наведено ряд недоліків даної платформи:

– Успішне завершення курсів та отримання сертифікатів не гарантує наявність потрібних знань і навичок.

– Недостатня кількість спеціалізованих матеріалів вищого рівня складності (більшість матеріалів вступного рівня – для того , щоб охопити якомога більшу аудиторію).

– Необхідність постійного підключення до мережі.

Наступну, як приклад, можна розглянути платформу GOOGLE CLASROOM. Ця платформа також є популярною не тільки на теренах України, але й по всьому світі. Як не складно догадатися, ця платформа створена компанією Google. Для того, щоб скористатись такою платформою, потрібно бути користувачем, і відповідно за допомогою цього облікового запису, можна створювати чи проходити певні курси.

На рисунку 1.2 візуально можна спостерігати простий дизайн і невеликий функціонал платформи від GOOGLE.

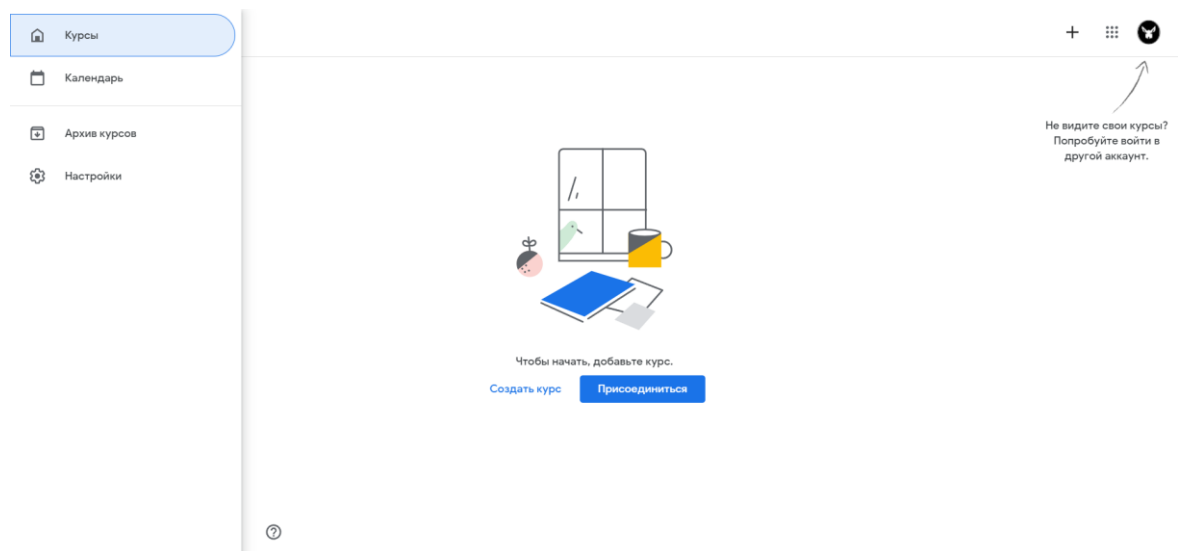


Рисунок 1.2 – Зовнішній вигляд GOOGLE CLASROOM

На зазначеній платформі, для того, щоб приєднатись до курсу потрібно, щоб викладач чи хтось інший надіслав вам ідентифікатор курсу – код, за допомогою якого можна буде проходити певний курс. По термінах курси

					КР.КІ. 8351649.00.00.000.ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		16

поділяють по-різному. Є курси, які можуть тривати декілька тижнів, а є курси, що можуть тривати рік й значно довше. Все залежить від програми і специфіки курсу. Навігаційна панель цього курсу є досить простою і сама платформа не видає на вибір курси. Єдиним кроком для проходження курсу є отримання коду доступу, який відкриває шлях до навчання.

Також до можливостей можна віднести те, що сама платформа легко інтегрується з іншими програмами G-Suite. Викладачі можуть зберігати свої документи на Диску Google, використовувати Таблиці Google для відстеження прогресу учнів, надавати відгуки та редагувати роботи студентів у режимі реального часу, а також проходити тести за допомогою онлайн-опитувань на основі Google Form.

В загальному, продукт від Google Classroom є досить хорошим для дистанційного навчання. В його функціоналі є переваги того, що здійснюється безпарольне подання завдань та постійне оновлення класів. Повторне використання завдань та рубрик дозволяє викладачам більше зосередитись на освітньому рівні учнів.

Google Classroom також має певні недоліки, які можуть бути критичними для початківця і можуть відлякати в користуванні.

– Обмежені можливості інтеграції. Google Classroom ще не інтегровано з Google Calendar або будь-яким іншим календарем, що може спричинити певні проблеми з організацією матеріалів і термінами виконання завдань.

– Немає автоматичних оновлень. Стрічка активності не оновлюється автоматично, тому учням потрібно регулярно оновлювати сторінку, щоб не пропустити важливі оголошення.

– Жодних автоматизованих тестів. Одна з головних причин того, що Google Classroom ще не може повністю замінити вашу систему керування навчанням, полягає в тому, що вона не забезпечує автоматизованих тестів і завдань для ваших учнів. Загалом Google Classroom більше підходить для змішаного навчання, ніж повністю онлайн-програма.

					КР.КІ. 8351649.00.00.000.ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		17

Також однією із найпопулярніших платформ у часи епідемії і у воєнні часи є MOODLE. Платформа широко використовується вищими навчальними закладами для навчання студентів. Сама платформа також є досить простою і не потребує багатьох знань, для того щоб нею користуватись. На даний момент платформа є досить зручною для навчання, але в більшості випадків вона складається лише з двох елементів. Читання документів, які викладачі викладають для ознайомлення студентів і завантаження уже готових робіт самих студентів. На рисунку 1.3 можна спостерігати зовнішній вигляд навчальної платформи і також список курсів, які можуть студенти, згідно номеру своєї дисципліни пройти.

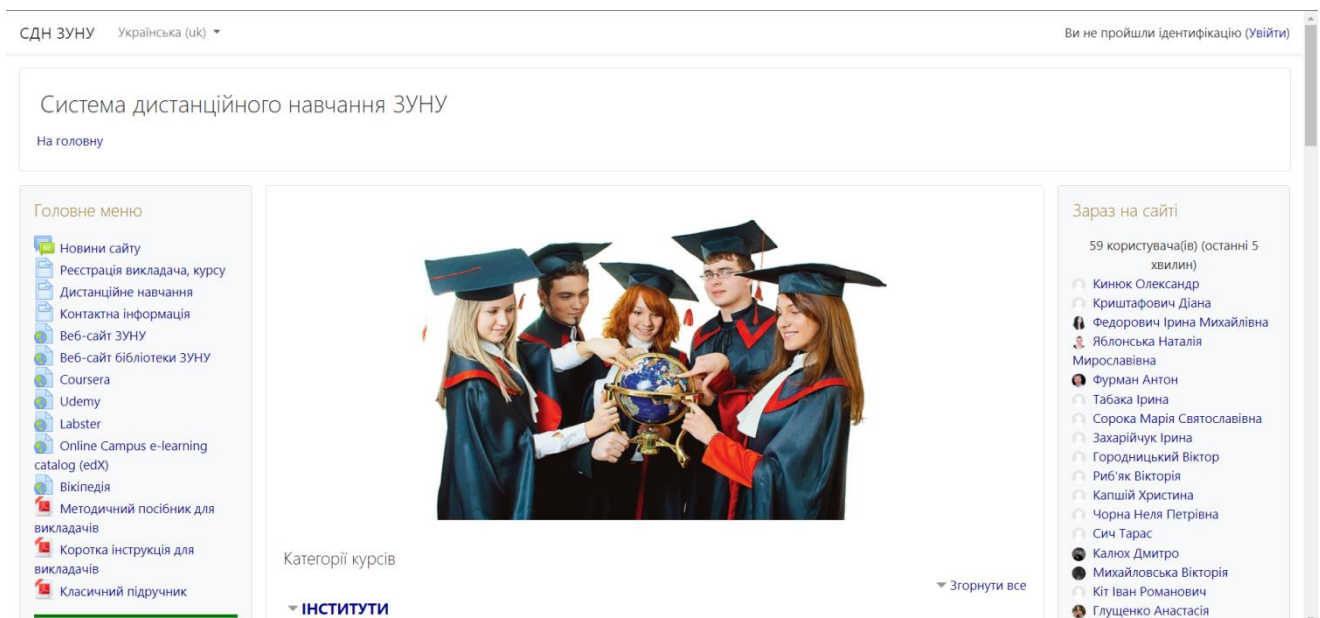


Рисунок 1.3 – Зовнішній вигляд стартової сторінки MOODLE

Платформа також передбачає месенджер, але він значно відстає від аналогів інших компаній, тому використовують цей елемент досить рідко, а то й не задіюють взагалі для здійснення комунікації. Цей елемент не є досить гнучким у використанні. Відповідно, для зручного користування потрібен більш гнучкий месенджер, оскільки аналоги, які існують містять в собі дуже багато функцій, які насправді зручні і також потрібні для навчання.

Тестування також є вагомим елементом Moodle. Для того, щоб теоретичні знання закріплювались, необхідно проходити контроль знань, що й відбувається

					КР.КІ. 8351649.00.00.000.ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		18

після кожної теми чи в кінці курсу. Уже залежить від дисципліни і викладача, який веде певну дисципліну. Ця платформа також побудована, як і більшість, але із взаємодією викладача. На відміну від інших платформ, де проходження курсу є більш автоматизоване, в цій усе відбувається безпосередньо із роботою викладача. Звичайно, усе налаштовується, але система не вміє оцінювати рівень знань. Єдиний метод оцінювання студентів, який можна автоматизувати, це тестування.

Для того, щоб курс був зданий потрібно пройти тестування, за допомогою якого можна і оцінити знання студента. В цій платформі нахил більше на комунікацію і співпрацю із викладачами, що є корисним і потрібним у навчанні та є дуже хорошим елементом у навчальній програмі. На даний момент платформа є безкоштовною, але вона доступна і співпрацює лише із вищими навчальними закладами. Також платформа має ряд недоліків:

- Відсутність можливості однозначної ідентифікації студента на відстані (немає гарантії, що тест пройдено саме тим студентом, чиє прізвище вказано в журналі оцінок);
- Ризик об'єктивної оцінки знань студента викладачем (основа дистанційного навчання – письмова, а деяким студентам може бути важко передати свої відповіді усно);
- Вимога до достатньо високого рівня підготовки студентів для успішної роботи в системі дистанційного навчання.

Для того, щоб користуватись цією платформою необхідно бути студентом. В більшості випадків навчатись стороннім людям не має можливості, оскільки реєстрація тут відбувається за допомогою закладів вищої освіти або інших навчальних закладів. Загалом на платформі наявна можливість лише прочитати матеріали за умови, що користувач є студентом. У випадку, коли користувач не є учасником академічної групи, якій надано доступ до певного курсу, то скористатися матеріалами не вдасться[9,12]. Структурна схема недоліків сучасних навчальних платформ наведена у додатку А.

					КР.КІ. 8351649.00.00.000.ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		19

1.3 Аналіз програмних засобів

Веб розробка у нашому світі зараз дуже сильно вкорінилась. Загальний опис веб-розробки включає ряд технічних та творчих процесів, пов'язаних з створенням та підтримкою веб-сайтів та веб-додатків. Це включає розробку веб-дизайну, програмування, веб-тестування, оптимізацію продуктивності, налаштування веб-серверів та розгортання веб-додатків на серверах. Веб-розробка вимагає знань різних програмних мов, таких як HTML, CSS, JavaScript, PHP, Python, Ruby та інших, а також розуміння принципів веб-дизайну, доступності, безпеки та SEO (пошукової оптимізації). Загальний опис веб-розробки може також включати розробку баз даних, інтеграцію зі сторонніми сервісами, розробку мобільних версій веб-додатків, роботу з веб-фреймворками та системами керування вмістом (CMS) та інші аспекти, пов'язані з створенням та розвитком веб-проектів.

Немає єдиного шаблону чи стандарту, щоб створювати сайти. В більшості випадків усе залежить від самого розробника і від його уяви. Усі можливості сьогодення дозволяють створювати не тільки зовні красиві веб сайти, але й досить функціональні, і з досить розумним штучним інтелектом. Для загального розуміння потрібно знати, що не завжди потрібно робити сайти із дуже великим функціоналом або, щоб це було інноваційним рішення у дизайнерській сфері. Потрібно розуміти в першу чергу призначення сайту, основний її функціонал, та для чого саме він створений. Також потрібно грамотно розподіляти частини і інформацію на сайті, що ресурси, які він використовує були задовільними і не виникати проблем із сторони клієнта чи сервера [35,36,37].

Веб-сайти - це важливий елемент сучасного цифрового світу, який дозволяє організаціям, бізнесам, особистостям та іншим користувачам розміщувати вміст і взаємодіяти зі своєю аудиторією. Однак, вони можуть відрізнитися за своєю призначеністю, функціональністю, метою та виглядом.

Отже, основні види веб-сайтів:

					КР.КІ. 8351649.00.00.000.ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		20

- інформаційні сайти – це найпоширеніший тип веб-сайтів, які містять інформацію про різні теми, такі як новини, розваги, наука, освіта, культура, спорт тощо. Вони можуть бути створені організаціями, редакціями, урядовими установами, науковими організаціями та іншими джерелами, які надають корисну інформацію відвідувачам;

- Блоги – це веб-сайти, де автори публікують свої особисті думки, дослідження, досвід, поради та інші матеріали в формі статей або записів. Блоги можуть мати різні тематики, такі як мода, кулінарія, подорожі, техніка, бізнес;

- електронна комерція – веб-сайти, які дозволяють користувачам купувати та продавати товари та послуги в Інтернеті. Вони можуть бути віртуальними магазинами, онлайн-аукціонами, ринками, платформами для електронної торгівлі та іншими.

Для початку потрібно з'ясувати, як саме буде працювати платформа і для чого вона буде використовуватись. Можна вибрати декілька видів засобів, які можуть підійти для розробки навчальної платформи, зокрема, веб-сайт, додаток.

Якщо використовувати веб-сайт, то для цього потрібно застосовувати середовища програмування і певну мову програмування при розробці проекту. В більшості випадків, сайт – це сукупність мов програмування, які співіснують між собою. За допомогою різних мов програмування можна використовувати різний функціонал, що допоможе розробити сайт більш гнучкішим і візуально приємним для користувачів. Процес створення складних сайтів може займати багато часу та сил. Проблематика більше заключається не в процесі написання коду, чи вибору середовища, на якому можна описати усі сценарії та синтаксис програми. Проблеми полягає ще у виборі технології створення сайту [36,38]. Щоб мови програмування співіснували між собою і могли передавати дані один одному, чи могли виконувати один сценарій подій. Ось список популярних технологій, які використовуються для створення веб-сайтів:

- HTML (Hypertext Markup Language) – це мова розмітки, використовувана для створення структури та відображення контенту на веб-сторінках;
- CSS (Cascading Style Sheets) – мова стилів, використовувана для оформлення веб-сайтів, задання зовнішнього вигляду та розміщення елементів на сторінці;
- JavaScript – скриптова мова програмування, яка використовується для створення динамічних та інтерактивних елементів на веб-сайтах, таких як анімація, форми, взаємодія з користувачем та інші функції;
- PHP (Hypertext Preprocessor) – серверна мова програмування, яка використовується для створення динамічних веб-сайтів та взаємодії з базами даних.
- CMS (Content Management System) – системи керування вмістом, такі як WordPress, Joomla, Drupal та інші, які дозволяють створювати, редагувати та керувати вмістом веб-сайту без необхідності програмування;
- Фреймворки – набори готових бібліотек та інструментів, які спрощують процес розробки веб-сайтів, такі як Bootstrap, Laravel, React, Angular, Vue та інші;
- Бази даних – системи для зберігання та управління даними веб-сайту, такі як MySQL, PostgreSQL, MongoDB та інші.
- Веб-сервери – програмне забезпечення, яке дозволяє розміщувати веб-сайти на сервері та забезпечує їх доступність в Інтернеті, такі як Apache, Nginx, IIS та інші.

Не обов'язково використовувати всі технології, які вищеперечислені. На розсуд розробника і в міру його знань, можна використовувати лише певний функціонал. Також на даний момент прогрес не стоїть на місці, і багато технологій містять більшість функціоналу, що притаманний іншим технологіям. В більшості випадків, такі платформи для навчання – це групові проекти, тому у кожній галузі повинен працювати фахівець.

					КР.КІ. 8351649.00.00.000.ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		22

Що стосується мобільних додатків, то ситуація досить подібна до розглянутого вище прикладу. В більшості випадків для створення додатку, який буде містити навчальні елементи і в принципі буде навчальною платформою використовують декілька технологій.

Також потрібно ще розібрати такий момент. На даний момент в світовій культурі технологій для смартфонів створюють два види додатків: Android і IOS. В більшості, користувачі будуть завантажувати перший варіант, але на даний момент у світову культуру досить активно вживаються айфони. Таких користувачів налічується дуже багато, відповідно для них також потрібно використовувати притаманні технології при розробці додатків. У кваліфікаційній роботі розробку платформи передбачено на основі першого методу.

Ось список популярних технологій, які використовуються для розробки додатків на платформах Android та iOS:

- Java/Kotlin (для Android) – мови програмування, що використовуються для розробки додатків на платформі Android. Java є основною мовою розробки на Android, а Kotlin – це мова, що стає все більш популярною, забезпечуючи більш сучасні можливості;

- Swift/Objective-C (для iOS) – мови програмування, що використовуються для розробки додатків на платформі iOS. Swift – це відносно нова мова, розроблена компанією Apple, яка стає все популярнішою, а Objective-C – це старіший варіант мови, що залишається популярним для деяких проектів;

- React Native – відкритий фреймворк, розроблений компанією Facebook, який дозволяє розробляти крос-платформні додатки для Android та iOS, використовуючи JavaScript;

- Xamarin – фреймворк, розроблений компанією Microsoft, який дозволяє розробляти крос-платформні додатки для Android та iOS, використовуючи C#;

					КР.КІ. 8351649.00.00.000.ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		23

– Flutter – відкритий фреймворк, розроблений компанією Google, який дозволяє розробляти крос-платформні додатки для Android та iOS, використовуючи мову програмування Dart;

– Firebase – хмарна платформа, розроблена компанією Google, яка надає інструменти для розробки функціональності додатків, таких як аутентифікація, бази даних, зберігання файлів та багато іншого.

Із вищеприведеного випливає, що для розробки навчальних додатків існують мови програмування, які використовують веб-технології. Проте, як було зазначено, на сьогоднішній день практично усі технології веб-напрямку можна використовувати при розробці мобільних додатків[28,29,31].

1.4 Постановка задачі кваліфікаційної роботи

Кваліфікаційна робота передбачає створення нової навчальної платформи, яка буде мати не тільки загальні навчальні елементи, але й допоможе підтримувати комунікацію студентів між собою і мати швидкий зворотній зв'язок з викладачами.

Така розробка полягає у покращенні здобуття освіти студентами у процесі навчання. Платформа спрямована на покращення навчання різноманітними методами, тобто будуть використовуватись допоміжні елементи, щоб краще змотивувати, полегшити засвоєння матеріалу та допомогою створення своїх проектів.

Задача створення навчальної платформи із комунікаційними елементами потребує застосування технічних знань, а саме в області програмування. Також для того, щоб на даній платформі навчання могло приносити результат, потрібен аналіз методик і варіантів навчання. Також варто враховувати усі потреби, які необхідні для навчання студента. Основне завдання платформи полягає у простоті використання та легкості засвоєння матеріалу.

					КР.КІ. 8351649.00.00.000.ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		24

Таким чином, мета дослідження полягає розробці програмної системи комунікації у навчальному процесі для забезпечення гнучкості, мобільності та результативності здобутті знань. Для досягнення мети роботи потрібно виконати такі завдання:

- розробити алгоритм навчання та показник успішної роботи платформи;
- розробити програмний модуль для реалізації алгоритму;
- здійснити реалізацію розробленого програмного модуля навчальної платформи;
- дослідити алгоритм роботи платформи та оцінити результати навчання, а саме: успішність, швидкість засвоєння інформації, практичні навички;
- обґрунтувати техніко-економічні показники розробленого програмного забезпечення.

					КР.КІ. 8351649.00.00.000.ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		25

2. АЛГОРИТМ РОЗРОБКИ СИСТЕМИ КОМУНІКАЦІЙ У НАВЧАЛЬНИХ ПЛАТФОРМАХ

2.1 Етапи створення навчальної системи із певними технологіями

На сьогоднішній день, ресурси і можливості дозволяють звичайному користувачеві створювати різні веб-ресурси. Для того, щоб користуватись усіма знаннями людства, достатньо буде мати доступ до інтернету і бажання витратити декілька годин для навчання. Також для розробки різноманітних веб-ресурсів необхідні не лише певні знання, але і практичні навички та досвід. Оскільки розробка стосується навчальної платформи, то крім зовнішнього вигляду, потрібно вміти вибирати дані та зберегти їх у відведеному місці.

Відповідно для того, щоб розробити будь-який проект необхідно побудувати план і приблизно уявити кінцевий вигляд того, що необхідно розробити. Також потрібно розуміти, що розробка є дуже тривалим процесом. Чим більші вимоги і функціонал, тим більше часу витрачається на те, щоб розробити необхідний функціонал.

Зазвичай, для того, щоб продукт розробити швидше, використовується команда людей, які розподіляють між собою обов'язки для роботи. В більшості випадків, розділяються на фронтенд і бекенд команди. Тобто, перша половина групи розробляє зовнішній вигляд і приблизний вигляд полів та функціоналу, який має бути для сайту. Для цього розробляють дизайн сайту для основної тематики сайту, щоб усе відповідало дійсності і було привабливим для звичайного користувача. Відповідно до замовлень, уже в процесі розробляється зовнішній вигляд та здійснюють виправлення для покращення чи додаткового функціоналу.

До другої групи припадає робота, яка не буде відчутна для користувачів. Це передача даних та її зберігання. Спеціалісти більше звертають увагу на швидкодії усіх функцій. Потрібно розуміти запити клієнтів і, попри всі навантаження, швидко виконувати усі вимоги. Щоб сайт працював завжди, використовують сервери, на яких розміщується сам сайт і інформація для

					КР.КІ. 8351649.00.00.000.ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		26

користувачів. В залежності від тематики сайту, розміщується і відповідна інформація. В навчальних платформах завжди зберігається інформація повчального характеру. Це можуть бути статті, наукові роботи чи повчальні матеріали. Відповідно, робота другої групи – це створення умов для того, щоб навчання чи створення наукового контенту було сприятливе [39,42].

Розробка будь-яких систем підпорядковується створеним планом розробки. Немає точного шаблону, за допомогою якого можна створити будь-який сайт. Відповідно, вже більше залежить від поставленої задачі та вимог, згідно яких і створюється план роботи. Нижченаведено основні етапи розробки будь-яких веб-ресурсів, які повинні існувати, щоб розробка була ефективною:

- визначення тематики сайту та його призначення;
- розробка технічного завдання;
- макетування сайту та створення його прототипу;
- програмування та верстка;
- наповнення сайту даними;
- тест та видача готового продукту.

Для того, щоб почати роботу над сайтом, потрібно визначити його мету та для чого він потрібний. Від цього залежить типологія ресурсу та зовнішній портрет цільової аудиторії та вимоги. Чітке розуміння цілої картини, мети та остаточного результату дасть змогу побудувати структуру проекту та сформулювати етапи, які потрібні для створення сайту, щоб досягти кінцевої мети.

На початковому етапі потрібно обговорювати усі необхідні особливості та деталі. Щоб вимоги співпадали із вимогами завдання, потрібно тісно обговорити всі деталі розробки. Потрібно розуміти, що перші варіанти сайтів можуть бути нереалізовані, тому краще обдумати усі варіанти та розробити декілька альтернатив. За допомогою порівнянь та обробки інформації, можна порівняти альтернативні продукти та створити на основі свій із унікальним функціоналом.

					КР.КІ. 8351649.00.00.000.ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		27

Також для виконання всього завдання, потрібно розуміти кінцеву суть платформи, та на що буде її нахил. Після прийняття рішень та узгодження всіх деталей розробки, можна переходити до наступного етапу [21,22].

Для подальшої роботи потрібно розробити план завдання. В ньому вказуються усі деталі та розписується структура сайту, її ще називають мапою сайту. До розгляду структури сайту, зазвичай, розглядають певні критерії:

- кількість сторінок у сайті;
- кількість розділів;
- кількість категорій;
- кількість блоків на кожній сторінці.

Також для зовнішнього вигляду ще розглядають вимоги стосовно дизайну та наповнення функціонального, візуального, текстового. Важливим аспектом у технічному завданні розглядається технічна можливість. Під час розробки плану обов'язкова потрібно розуміти певні деталі та критерії, які потрібно врахувати, щоб можна було створити освітній сайт:

- деталізація – прописується кожен аспект та кроки, які буде виконувати користувач;
- чітке об'єктивне формулювання завдання;
- зрозумілість – усі вимоги, що розписуються в документі чи в плані повинні бути зрозумілі для фахівця. Також повинна використовуватись відповідна термінологія.

План – це інструкція, яка буде орієнтувати усіх працівників під час розробки сайту. Після узгодження усіх деталей в цьому етапі можна буде перейти до виконання основних робіт.

На етапі макетування та створення прототипу сайту обговорюється питання реалізації ідеї у реальний об'єкт. На цьому етапі не буде розроблятися ще повний функціонал проекту. Розглядають зовнішній вигляд та основний орієнтир. Для навчальної платформи буде розглядатися розміщення вкладок та зручність читання інформації.

					КР.КІ. 8351649.00.00.000.ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		28

Щоб правильно розмістити всі кладки та навігаційні посилання, створюється декілька варіантів зовнішнього вигляду сайту. Особливо увагу тут приділяють навігаційному меню. Воно повинно бути легким у сприйнятті та містити в собі хороший навігаційний функціонал. Зазвичай, навігаційне меню знаходиться у

верхній частині сайту або із правої сторони. Уже згідно тематики сайту, необхідно орієнтуватись і пробувати усі варіанти створення зручного меню. Ці елементи повинні бути обов'язково враховані, оскільки вони впливають на стабільність та швидкодію сайту.

Після узгодження попередніх етапів розробки можна переходити до найважливішої частини – верстки та програмування. Даний процес передбачає спроби створити сприятливі умови для зовнішнього вигляду та функціоналу сайту. Потрібно усі елементи розробити так, щоб вони знаходились на своїх місцях, виконували свою функцію та не перешкоджали роботі іншим елементам. Для верстки, зазвичай, використовують такі елементи як HTML, та для задання стилів і яркого зовнішнього вигляду CSS [40].

Після створення зовнішнього вигляду необхідно “оживити” сайт та наповнити його необхідними функціоналом. Необхідно перевірити, чи усі запити від користувачів будуть прийматись. Також перевірити зв'язок із базою даних. Для того, щоб дані користувачів десь зберігались, використовують сервери. Також сервери використовують різноманітні мови, за допомогою яких і можна здійснювати передачу, обробку та зберігання даних.

Для того, щоб дані були передані із сайту до сервера використовуватись буде мова PHP. Ця мова є спеціалізованою для таких задач, що полегшує розробку певного елемента. Також, щоб дані зберігались і мали змогу назад повернутись для розгляду, використовують спеціалізовану мову запитів MYSQL. За допомогою цієї мови, усі дані можуть записуватись в базу даних, оброблятись, змінюватись так повертатись назад. Оскільки сайт буде із освітнім напрямом, відповідно, потрібно впевнитись, що всі дані, які будуть вносити користувачі, відображаються коректно на сайті та зберігаються у базі [36,38].

					КР.КІ. 8351649.00.00.000.ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		29

Після завершення створення верстки та зв'язку із базою даних, сайт може виконувати свій функціонал, але він пустий. Необхідно наповнити його різноманітним контентом. Тобто, сторінки розділів являються пустими і для загальної інформації, щоб користувачі могли інтуїтивно користуватись платформою. Оскільки, часто на багатьох платформах відбуваються ситуації, що користувачі не можуть ознайомитись із усім функціоналом сайту, через те що він є ускладненим.

Завершальний етап у розробці веб-ресурсу полягає у тестуванні. Для того, щоб усі користувачі не стикались із помилками та проблемами на кожному кроці проводяться тестування. Відштовхуючись від тематики сайту, проводять різноманітні тести. Для навчальних платформ, відповідно, проводиться тест на можливість ефективного навчання. Тобто, потрібно створити навчальний матеріал і використати його для навчання. Необхідно врахувати усі моменти і етапи у звичайному навчанні і максимально подібне перенести у платформу. Відповідно, після того, як платформа буде готова і матиме змогу навчати свої користувачів, вона буде готова для загального користування.

2.2 Аналіз вимог до програмного забезпечення для розробки освітніх систем

Аналіз вимог до програмного забезпечення для розробки освітніх систем є одним із найважливіших етапів у процесі розробки будь-якої системи. В сучасному світі освітні системи будуються на складних і багатофункціональних системах. Для їх розробки ретельно і досить тривалий час аналізується вимоги користувачів та установ, що пов'язані із освітою.

Основна мета аналізу вимог полягає в тому, щоб з'ясувати, які функціональні та нефункціональні вимоги повинна задовільняти система. При цьому, потрібно враховувати різноманітні специфіки і методи освітніх систем, їх мету та завдання, а також багато вимог на рахунок безпеки та даних

					КР.КІ. 8351649.00.00.000.ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		30

інтеграції з іншими системами. В більшості, потрібно зрозуміти, яким чином будуть захищені дані.

У процесі аналізу вимог проводяться різноманітні дослідження та опитування на рахунок позитивних та негативних відгуків про ті чи інші функціональні складові розробки. Також, такий метод дозволяє зібрати важливі дані на рахунок потреб та побажання користувачів, які можуть і повідомляти про негативні та позитивні сторони розробки та вказувати на проблематичні чи недоречні частини, які псують загальну картину розробки. Результатом такого аналізу можна виділити певні аспекти розробки програмного продукту та приділити увагу недолікам, які можуть бути критичними, або взагалі не припустимими для стабільної роботи.

Правильність проведених аналізів вимог посідає важливе місце та є ще одним кроком для формування структури та реалізації ідеї освітньої платформи. Розробка за таким планом дозволяє точно визначити, який функціонал мусить містити система та які вимоги потрібно враховувати при розробці [21,22].

Аналіз вимог допомагає розробникам зрозуміти та задіяти певні програмні засоби та програмні середовища, на основі яких система повинна працювати, які функції вона буде містити та виконувати, які ресурси вона повинна мати та які механізми безпеки повинні буди налагоджені.

Одним із ключових аспектів аналізу вимог є розуміння, для чого буде використовуватись програмний продукт та розуміння контексту використання системи. Для прикладу, освітня система може бути розроблена для навчання в університеті, школах або для самостійного навчання. В більшості випадків, для створення освітніх платформ, схиляються до дистанційного навчання. Оскільки навчання уже по створеному алгоритму має свій план та структуру, за допомогою якої користувач зможе зрозуміти суть і отримати певні знання в певній області. Важливо також зрозуміти, що освітні платформи можуть бути різноманітними і, відповідно, вимагати різноманітний функціонал. В залежності від характеристики системи потрібно визначити, хто буде саме

					КР.КІ. 8351649.00.00.000.ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		31

користуватись системою, та як буде використовуватись її функціонал так можливості.

Насамперед, для аналізу вимог важливо враховувати законодавчі та регуляторні вимоги, які можуть стати основою для методики освітньої платформи та її подачу матеріалу. Також, для того, щоб освітня платформа була безпечною, вона повинна відповідати стандартам безпеки даних або правилам захисту персональної інформації. Відповідно, необхідно також певну частину розробки обдумувати систему безпеки та шифрування даних.

Для того, щоб визначити можливості програмного забезпечення, використовуються функціональні вимоги до освітньої системи комунікацій. Такий метод перевірки дозволить перевірити функціонал і межі, які дозволяє система своїм користувачам. Важливим аспектом вимог можна рахувати сумісність системи. В першу чергу, потрібно розуміти, щоб комфортно працювати над своїм проектом, потрібно задумуватись про сумісність із освітніми програмами, які є основою навчального процесу, та чи сумісна система із пристроєм, з яким працює користувач [24].

Кожен користувач, який навчається чи просто використовує навчальний продукт, повинен внести необхідні дані для системи, що можна було після цього ідентифікувати та зберігати інформацію по цьому користувачу. В більшості випадків зберігають основну інформацію, таку, як логін і пароль. Ці дані повинні бути захищені, оскільки вони є основним ключем, за допомогою якого можна отримати певні дані по тому чи іншому користувачеві.

Також, найбільш важлива роль безпеки платформи полягає в обміні даних. Користувачі, які взаємодіють між собою на цій платформі, повинні бути впевненими, що їхні дані не будуть втраченими чи передані зловмисниками. Щоб користувачі могли комфортно працювати і не боятись за безпеку своїх даних, система повинна мати необхідний рівень безпеки та точність. Освітня платформа в основному працює із даними, які вводять користувачі, відповідно, система також повинна грамотно відтворювати інформацію. В освітніх системах потрібно, щоб грамотна система надавала інформацію, та за потреби могла коректно оцінювати користувачів.

					КР.КІ. 8351649.00.00.000.ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		32

Одна із найосновніших вимог для розробки освітніх платформ полягає у відповідності і дійсності зазначеної тематики. Освітня платформа повинна містити лише навчальні матеріали та інструменти, які допомагають навчатись. Зазвичай, освітні платформи мають багато елементів від інших типів платформ та сайтів, але вони є другорядними і лиш доповнюють загальний функціонал.

Крім функціональних вимог, оцінюють програмні розробки по не функціональних критеріях. В цих критеріях перевіряється і вимагається, яка систем повинна бути. Потрібно розуміти також загальну картину і вигляд системи, щоб розуміти взаємодію між всіма вкладками та функціями. Перевіряється продуктивність системи, тобто, чи система може бути корисною і зможе надавати відповідний результат до вимог. Також, перевіряється система на навантаження, як буде поводити себе система, і чи буде виконуватись такий ж самий точний розрахунок, якщо користувачів буде декілька. Згідно цього, потрібно розуміти, що користувачів може бути багато і система повинна працювати стабільно при такій кількості роботи і запитів [21].

Потрібно звернути увагу на те, що система не може бути вічною і також має свої межі у роботі. В будь-який момент можуть статись непередбачувані ситуації, і доступ до платформи може зникнути. В таких випадках розробники одразу беруться виправляти проблеми та виявляють причину несправності.

Щоб усе було швидко і якісно, програмована система повинна бути ремонтно-придатною. Якщо під час ремонту системи відбуваються труднощі та різноманітні перешкоди, які потрібно виправити перш, ніж виправляти проблему, то систему, яку розробили для навчання є ремонтно-придатною із низьким показником. Такий показник в загальному вимірюють на рівні коду, використовуючи в цьому цикломатичну складність. Цей показник визначає складність виконання роботи. Тобто, якщо код є надто складним, то, відповідно, виявити і усунути проблему набагато складніше. На рисунку 2.1 наведено приклад коду без загальноприйнятої структури [18].

					КР.КІ. 8351649.00.00.000.ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		33

```
1 <php
2
3 function generateNumbers()
4 {
5     for ($i = 0; $i <= 5; $i++) { yield $i; }
6 }
7 foreach(generateNumbers() as $number) { echo $number; // 012345 }
8 /php>
9
```

Рисунок 2.1 – Приклад коду із неправильною структурою

На рисунку вище можна розглянути код на мові програмування PHP. Навіть початківець, який тільки побачив цей код візуально може зрозуміти, що елементи, які знаходяться в рядку, стоять не на своїх місцях. На даний момент великих труднощів не займе розібрати код в декілька стрічок, але якщо це багатофункціональна платформа, яка складається із багатьох файлів і має зв'язок із базою даних, то такий елемент у побудові програмного забезпечення є просто необхідним. На рисунку 2.2 можна розглянути правильну побудову коду на тому ж самому прикладі.

```
1 function generateNumbers()
2 {
3     for ($i = 0; $i <= 5; $i++) {
4         yield $i;
5     }
6 }
7
8 foreach(generateNumbers() as $number)
9 {
10     echo $number; // 012345
11 }
```

Рисунок 2.2 – Приклад коду із правильною побудовою

На рисунку вказано приклад коду мови програмування PHP, де дотримано синтаксис і побудову. За допомогою такого написання коду, можна комфортно оглядати усі складові коду і розуміти, яка функція відповідає за конкретний елемент у системі. Цілісність і правильна побудова коду являє

ключовий елемент при створенні програмного продукту і в подальшому її модернізації [42].

Однією із основних проблеми при написанні коду є неорганізована стратегія найменування сутностей. Система повинна працювати, як єдиний механізм, відповідно, назви функцій та змінних також повинні відповідати тематиці та ідеї проекту. Якщо стандарт назв та імен не впорядковані, то структура коду може потім заплутати розробника і, відповідно – це може призвести до ще більших пошкоджень. Якщо платформа складається із багатьох сторінок, то найменування вкрай важливе для орієнтації та розставлення грамотних зв'язуючих між передачею даних. Також, можна виділити декілька помилок та проблем, з якими стикаються розробники при налагодженні системи та її ремонту.

- Завеликі методи, занадто великі методи являються джерелом помилок та складні для розуміння. Є загальноприйняте правило, в якому функції повинні виконувати лише одне завдання та виконувати його добре;

- Завеликий клас, який робить дуже багато різних речей також є джерелом незрозумілостей і труднощів у виконанні. Клас, як і функція повинен мати лише одне завдання, тобто його робота повинна полягати лише в чомусь одному. Такий клас повинен бути розділений на кілька сутностей, кожна з яких буде виконувати конкретно своє завдання;

- Код, що дублюється. Ідентичний код, який розкиданий по всьому додатку збільшує складність його розпізнавання. Оскільки при модернізації та виправленні помилок можна обрати не цю частину коду, може привести до проблем у роботі всього сайту;

- Надлишок параметрів. Довгий список параметрів ускладнює читання, виклик та тестування функцій. Якщо зменшувати кількість параметрів коду, то час на вивчення та тестування буде значно менший;

- Недоречна складність. Надмірно складні шаблони, коли проектують там, де це зовсім не потрібно. Складові, де простого виконання функції чи простої архітектури було достатньо, краще не ускладнювати лишніми методами та додатковими параметрами. Використання складних патернів без великої

					КР.КІ. 8351649.00.00.000.ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		35

необхідності вказує на здатність побачити картину цілком та здатність уникнути зайвої роботи та складності.

Також, одним із важливих вимог для системи є здатність програмної системи пристосовуватись до змін у середовищі. Система повинна бути стабільною і не змінюватись у функціоналі чи в інших моментах.

До вищеперелічених вимог, ще можна додати декілька вимог, які також враховують під час розробки програмного продукту:

- збереження даних;
- конфіденційність;
- стабільність;
- цілісність даних;
- резервне копіювання;
- ефективність;
- управління відмовами.

Програмні розробки можна оцінювати по багатьох критеріях, в більшості випадків вони злежать від вимог самої системи. В залежності, який функціонал містить ця розробка, відповідно і по таких критеріях можна оцінювати. В освітніх сайтах найбільшу увагу приділяють інформації. Тобто інформація повинна бути доступна, зрозуміла та візуально читабельна. Також цілісність даних є важливим у цій сфері розробки [18].

На рисунку 2.3 можна візуально розглянути один із прикладів функціональних та не функціональних вимог, які необхідно використовувати для створення програмного продукту. Ця схема для кожного програмного продукту може бути різною, але вона дозволяє зрозуміти основну концепцію вимог, які необхідно виконати, щоб програмний продукт мав кінцевий і цілісний вигляд.

					КР.КІ. 8351649.00.00.000.ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		36

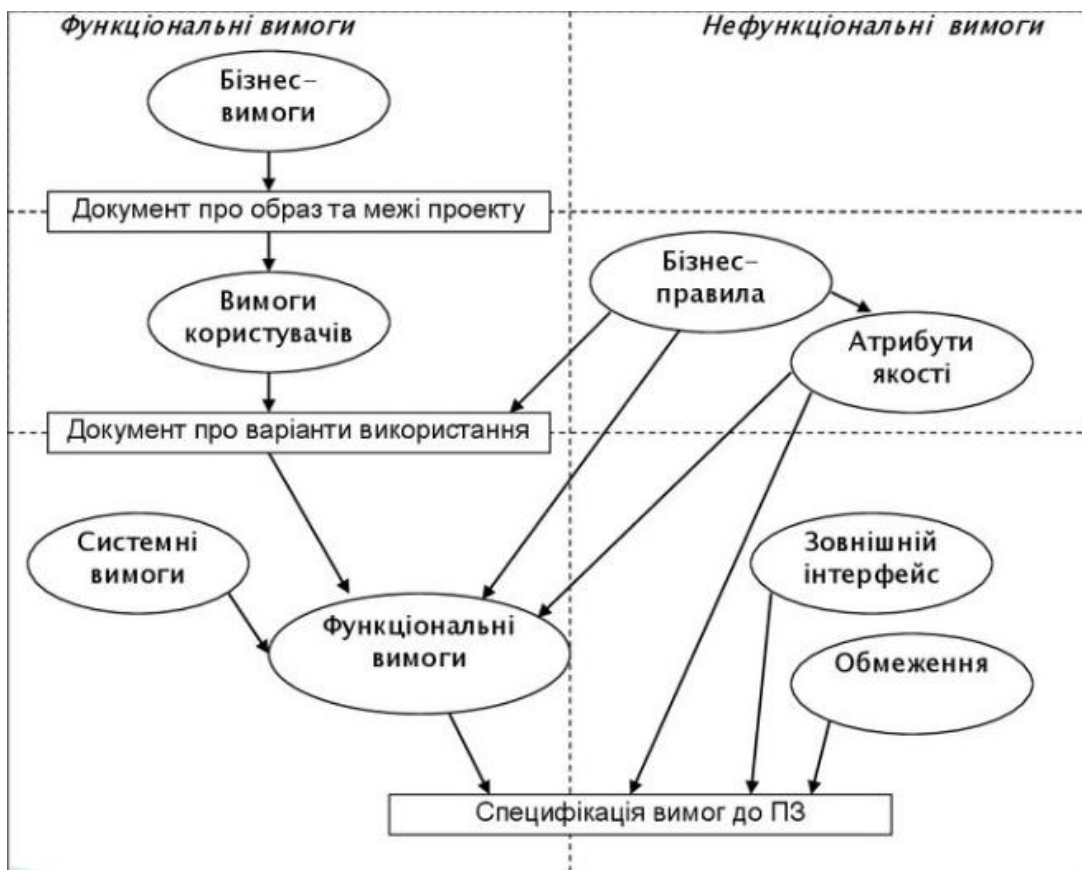


Рисунок 2.3 – Структура функціональних та не функціональних вимог до програмної розробки

Візуально можна побачити також, що вимоги для розробки програмного забезпечення тісно пов'язані між собою та переплітаються. Фактично, одні вимоги залежать від інших, і вони повинні гармонійно співіснувати, щоб система мала цілісну структуру та можливість стабільно функціонувати.

2.3 Алгоритм розробки системи комунікацій у навчальному процесі

Для дистанційного навчання, як зазначалось вище, найкраще мати середовище, де можна отримувати свої знання, зберігати дані, та користуватись інструментами, які допоможуть створити навчальну атмосферу. В таких випадках, розробляються навчальні платформи із можливостями комунікацій з іншими студентами та вчителями. Також, розробка освітніх платформ робить освітній процес цікавим завдяки інструментарію сайту та його наповнення.

Освітні сайти так, як і усі інші, мають певні елементи, які потрібно розробити, щоб сайт був прийнятий для навчання. Щоб сайт мав цікавий та корисний функціонал, він повинен містити елементи, які необхідні для освітніх закладів.

– Методи гейміфікації. Для того, щоб почати навчання в будь-якій сфері, потрібно мати мотивацію. Також додатковою мотивацією буде слугувати сам сайт. При навчанні для студентів буде система нагород. За успішне виконання тестів та задач можуть видаватись різноманітні бонуси, такі як: спеціальні статуси, іконки профілю, додаткові бали до рейтингу;

– Система виставлення оцінок. По завершенню навчання з певної дисципліни студенти повинні отримати підсумковий бал, щоб дізнатись рівень своїх знань та наскільки ефективно вони засвоїли матеріал поданий сайтом;

– Графік занять. Завдяки такому елементу студенти можуть самостійно організувати свій час так, як їм це зручно і, відповідно, навчатись, тоді коли їм це зручно;

Першим кроком у створенні онлайн платформи має стати дослідження інтересів користувачів та розгляд вмісту конкурентних сайтів. Це допоможе створити загальну картину та чітке бачення програмного продукту. Потрібно розуміти, що якщо основна цільова аудиторія буде складатись із студентів, відповідно, потрібно створити найбільш підходящу систему навчання.

В першу чергу, основною інформацією платформ використовується веб-частина “текст”. Ця частина дозволяє виконувати передачу основної маси інформації, яка потрібна для навчання. Для того, щоб текст гармонійно і коректно сприймався, потрібно встановити відповідний шрифт та розмір тексту, який будуть використовувати користувачі при навчанні[17].

У досягненні хороших результатів та хорошої відвідуваності студентів необхідно наповнити навчальні матеріали не тільки текстом, а і розміщувати різноманітні мультимедійні елементи. Веб-частина сайту повинна містити зображення, це можуть бути різноманітні графіки, діаграми або навіть використовуватись для впливу на текст. Зображення та різноманітні елементи мультимедійного характеру можуть із собою нести різноманітну і корисну

					КР.КІ. 8351649.00.00.000.ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		38

інформацію. Такими елементами можна замінити посилання, або різноманітну інформацію подати у вигляді картинки чи анімації.

Якщо цільовою аудиторією системи будуть студенти, відповідно, потрібно створити профіль, за допомогою якого можна буде розміщувати різноманітні досягнення та роботи користувачів. При розробці особистого кабінету студента платформи найкраще зробити акцент на максимальну простоту та комфортність у навчанні та відслідковування рейтингу та іншої інформації. Для успішності навчання важливо мати можливість для відстеження своїх робіт та проходження дисциплін, що будуть знаходитись на платформі. Таке реалізується за допомогою рейтингової таблиці або статистики, яка може повідомляти інформацію щодо наступних критерій :

- кількість пройденого матеріалу;
- теми, які можуть бути проблемними;
- показник успішності.

Платформа із навчальним матеріалом з часом повинна оновлюватись і поповнятись. Для цього створені особливі профілі для викладачів, які, власне, можуть створювати свої навчальні матеріали та розміщувати їх на сайті. Створення кабінету викладача створює зручні та комфортні умови для роботи всіх учасників платформи. Розробникам фронтенд і бекенд під час розробки потрібно враховувати, що викладач може мати більший інструментарій та доступ до своїх навчальних матеріалів. Такі можливості забезпечують для репетиторів і просто викладачів наступні можливості:

- перегляд успішності занять;
- створення і редагування всіх курсів викладача;
- відстеження відвідуваності та рейтинг дисципліни.

Для того, щоб система навчання працювала і не виникало проблем, потрібно перевіряти співіснування усіх функцій. Потрібно розробити алгоритм роботи, який дозволяє викладачам розміщувати інформацію на сайті, а студенти могли би її читати. Схему взаємодії викладача та навчальної системи можна розглянути на рисунку 2.4.

					КР.КІ. 8351649.00.00.000.ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		39

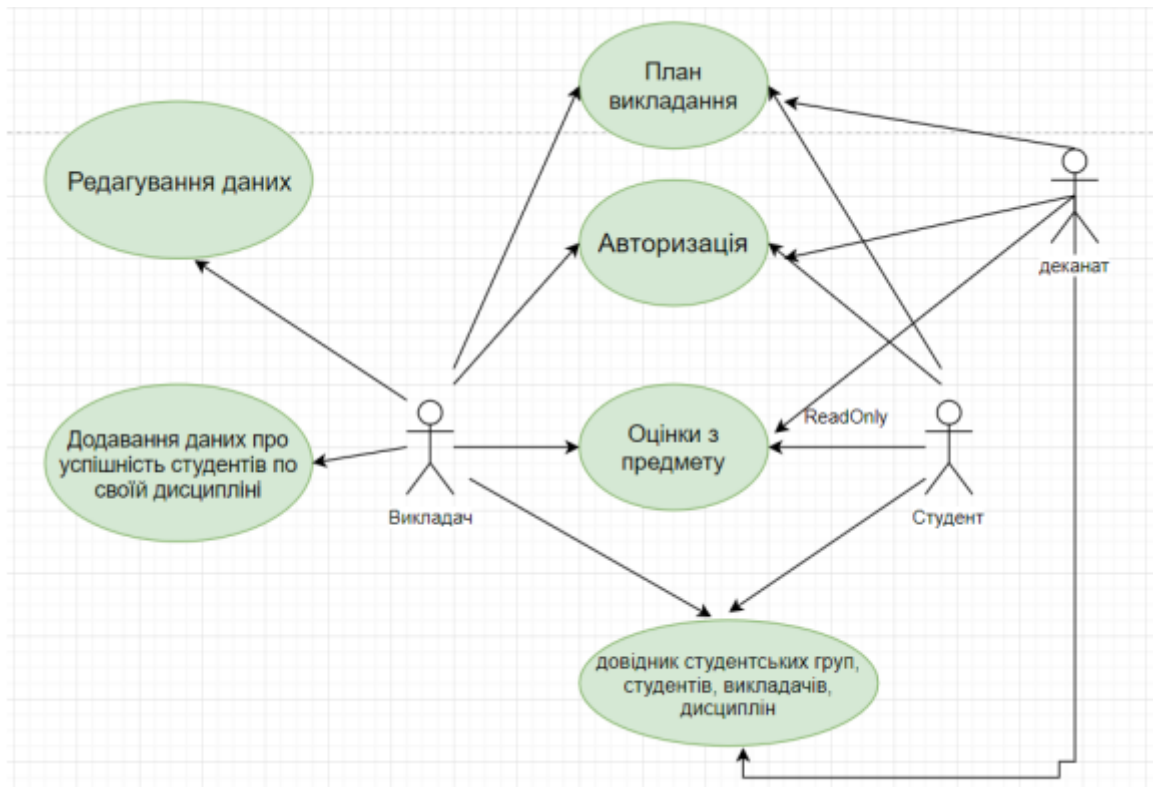


Рисунок 2.4 – Візуальна схема взаємодії викладачів та студентів у навчальній системі

Розуміння загальної картини та схеми взаємодії всіх елементів системи дозволяє побудувати грамотний механізм роботи. Грамотна робота системи дозволить без проблем пересуватись по сайту у пошуках необхідної інформації за допомогою навігаційного меню – основного інструменту, який потрібен для роботи і який уможливорює роботу на платформі. Також можна виділити декілька таких видів навігаційного меню, які можна використати для навчальної платформи:

- Горизонтальний текст. Найпростіший стиль побудови навігаційного меню, який можна знайти на просторах інтернету. Такий тип навігації має вигляд горизонтального списку, що містить розділи сайтів, які, загалом, називають одним або двома словами. Це меню може бути створене за допомогою графіки або верстки HTML-текст. Обидва мають невеликий функціонал взаємодії із користувачем. На рисунку 2.5 можна побачити, як саме виглядає такий тип навігаційного меню;



Рисунок 2.5 – Структура горизонтального меню із графічним елементом

- Підменю використовують у тому випадку, коли потрібно представити користувачу глибинну та детальну інформацію. Наявність підменю вимагає більше місця на сайті. Особливості такого підменю полягають в тому, що користувачі можуть отримати більш детальну інформацію. На рисунку 2.6 можна побачити приклад випадаючого підменю, за допомогою якого можна отримати більше посилань;

- Вертикальний текст на даний момент є досить поширеним елементом навігації по навчальному сайті. Такі види навігаційного меню використовують, де потрібен великий список посилань, на які є можливість перейти. Також назви посилань можуть бути довші і вертикальне меню може усунути проблему із розміщенням елементів, не спотворюючи сусідній вміст. Вертикальна навігація найбільше зустрічається у лівій стороні веб-сайтів. Вона може слугувати як додатковим елементом горизонтальної навігації, і як випадаюче меню для детальної інформації. На рисунку 2.7 розміщений приклад вертикального меню.

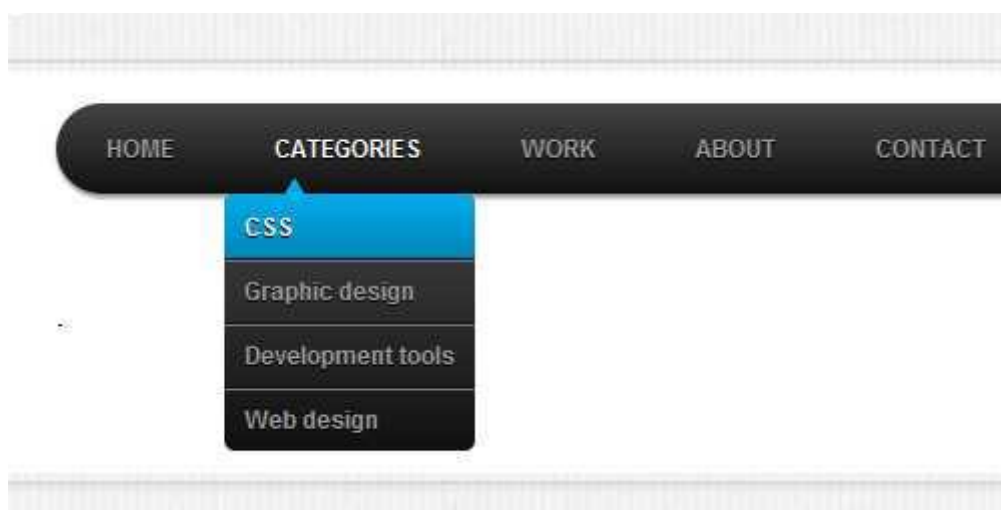


Рисунок 2.6 – Структура навігаційного меню та підменю при наведенні



Рисунок 2.7 – Приклад горизонтального навігаційного меню

Навігаційне меню можна спостерігати не тільки у текстовому вигляді. У теперішній час дуже популярне меню у вигляді графічних іконок. Інтеграція значків меню створює простий зрозумілий та інтуїтивний інтерфейс, за допомогою якого можна комфортно та швидко пересуватись по різноманітних розділах освітнього веб-ресурсу. Для такого виду навігаційного меню необхідно використовувати значки, які будуть по вигляду нагадувати суть і значення самого сайту. Також потрібно розуміти чи додавання піктограм буде доповняти дизайн та функціональність сайту. На рисунку 2.8 можна побачити приклад навігаційного меню з іконками замість тексту.

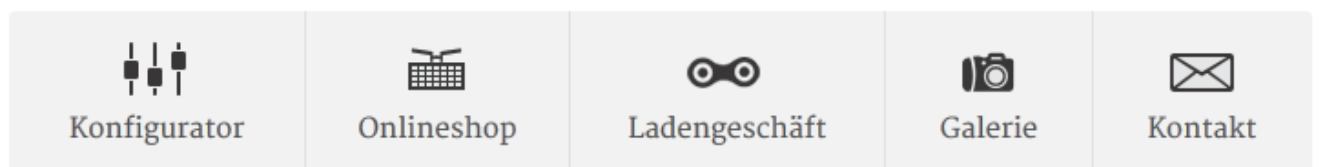


Рисунок 2.8 – Структура навігаційного меню у вигляді іконок, які передають загальне значення посилання

Комфортне навчання на платформах і освітніх сайтах полягає не лише у вивченні матеріалу та проходження тих чи інших дисциплін. Якщо весь інструментарій сайту налаштований коректно і дані відображаються вірно, то навчання можна здійснювати. Враховуючи, що основна маса користувачів

					КР.КІ. 8351649.00.00.000.ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		42

будуть студенти, відповідно, потрібно створити і засоби комунікації, щоб користувачі могли обмінюватись своїми знаннями. Також потрібно зазначити, що онлайн навчання відбувається без фізичного контакту з користувачами та викладачем, тому необхідно створити додатковий інструмент, який виправить частково цю ситуацію. Це може бути:

- чат-боти;
- форуми;
- онлайн чати.

Урізноманітнити навчання та користування освітньою платформою можна технологією чат-бот. Ця програма імітує справжню розмову із користувачем. Цей інструмент є досить корисним для виконання певних завдань. Чат-боти використовують машинне навчання. Через те, що вони постійно взаємодіють з людьми, то їхній алгоритм побудований таким чином, щоб із часом програма могла наслідувати справжню розмову із користувачем. В основному, такі інструменти створені по заданому розробником сценарію, але існують різноманітні варіації цієї технології, які можуть містити в собі різний функціонал [35,36,37].

Форуми корисні в онлайн навчанні тим, що під час навчання завжди виникають певні труднощі та запитання. На просторах сфери ІТ форуми є дуже популярними на рахунок вирішення проблем. За допомогою форм-чатів інші користувачі можуть розглянути проблему і допомогти у її реалізації. Також, для того щоб розробка і вибір варіації вкладок та методів кодування був можливий, потрібно розробити алгоритм, по якому буде будуватись система. В першу чергу, платформа повинна слугувати корисним інструментом для свого користувача, та мати тісний з'язок із користувачем.

Аналіз потреб. Потрібно почати з аналізу потреб різних учасників навчального процесу. Зрозуміти, які види кумнікації будуть необхідними, та які канали спілкування використовуються. Функції та мажливості, які потрібні.

					КР.КІ. 8351649.00.00.000.ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		43

Визначити функціональні вимоги. На основі аналізу потреб скласти список функціональних вимог до системи комунікацій. Нариклад, обмін повідомлення, форму для дискусій, доступ до спільних матеріалів тощо.

Вибір технологій. Обрати технологію створення платформи. Розглянути різні варіанти, такі як створення платформи на у веб просторі, використання декількох технологій разом.

Розробка базової структури. Створити базову структуру системи комунікацій, включаючи реєстрацію користувачів, автентифікацію, управління профілями і сновні функції комунікації, такі як повідомлення.

По завершенню розробки алгоритму, та побудови системи потрібно протестувати платформу.

					КР.КІ. 8351649.00.00.000.ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		44

3.ПРОГРАМНА РЕАЛІЗАЦІЯ АЛГОРИТМУ РОБОТИ СИСТЕМИ КОМУНІКАЦІЇ

3.1 Структура програмного забезпечення навчальної системи

Створення будь-якого програмного продукту займає багато часу та сил. В першу чергу, створення проекту полягає у його складності реалізації. Перенести ідею в реальність та знайти метод його застосування потребує багато часу на побудову взаємодії між всіма елементами усіх сайтів. Для створення сайту вище уже зазначались різноманітні технології та методи. Все, що потрібно для реалізації програмного продукту - це вміння грамотно будувати проект та володіти знаннями у технологіях, які будуть застосовуватись. Навчальна допоміжна платформа повинна взаємодіяти із користувачем. Усі дані, які будуть вводиться, система повинна зберігати обробляти, та видозмінювати. Для того, щоб це реалізувати побудова сайту буде розділена на дві частини:

- Frontend являється частиною веб-роробки, яка відповідає за розробку та побудову користувацького інтерфейсу веб-додатків. Фронтенд-розробники працюють з мовами програмування, такими як HTML, CSS та JavaScript, для створення веб-сторінок, що відображаються у веб-браузері. Вони займаються розробкою та оформленням сторінок, включаючи структуру, дизайн, анімацію та інтерактивні елементи. Фронтенд-розробники також працюють з бібліотеками та фреймворками, такими як React, Angular або Vue.js, що дозволяють їм ефективно розробляти складні веб-додатки з багатофункціональним інтерфейсом. Вони забезпечують, щоб веб-сторінки були зручними для взаємодії користувачів та мали сучасний та привабливий зовнішній вигляд;

- Backend являється складовою розробки програмного забезпечення, яка займається обробкою даних, логікою бізнес-процесів та взаємодіє з базами даних. Бекенд-розробники працюють з серверними мовами програмування, такими як Python, PHP, Java або Node.js, для створення серверної логіки, API

					КР.КІ. 8351649.00.00.000.ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		45

(інтерфейсів програмування додатків) та баз даних. Вони забезпечують обробку запитів від фронтенду, здійснюють розрахунки, зберігають та отримують дані з баз даних, виконують автентифікацію та авторизацію користувачів. Бекенд-розробники також відповідають за безпеку, виконують оптимізацію та масштабування серверної інфраструктури. Вони розробляють серверні додатки, які працюють у фоновому режимі і забезпечують функціональність веб-додатків, яку користувачі не бачать напряму.

Для створення програмного коду потрібно використовувати середовища, які допомагають у написанні коду та подальшого його читання. Використовувати можна будь-які редактори, щоб браузер могли зчитувати верстку та код. Для створення допоміжної навчальної платформи для студентів буде використовуватись Atom. На рисунку 3.1 зображено його вигляд із написаним кодом.

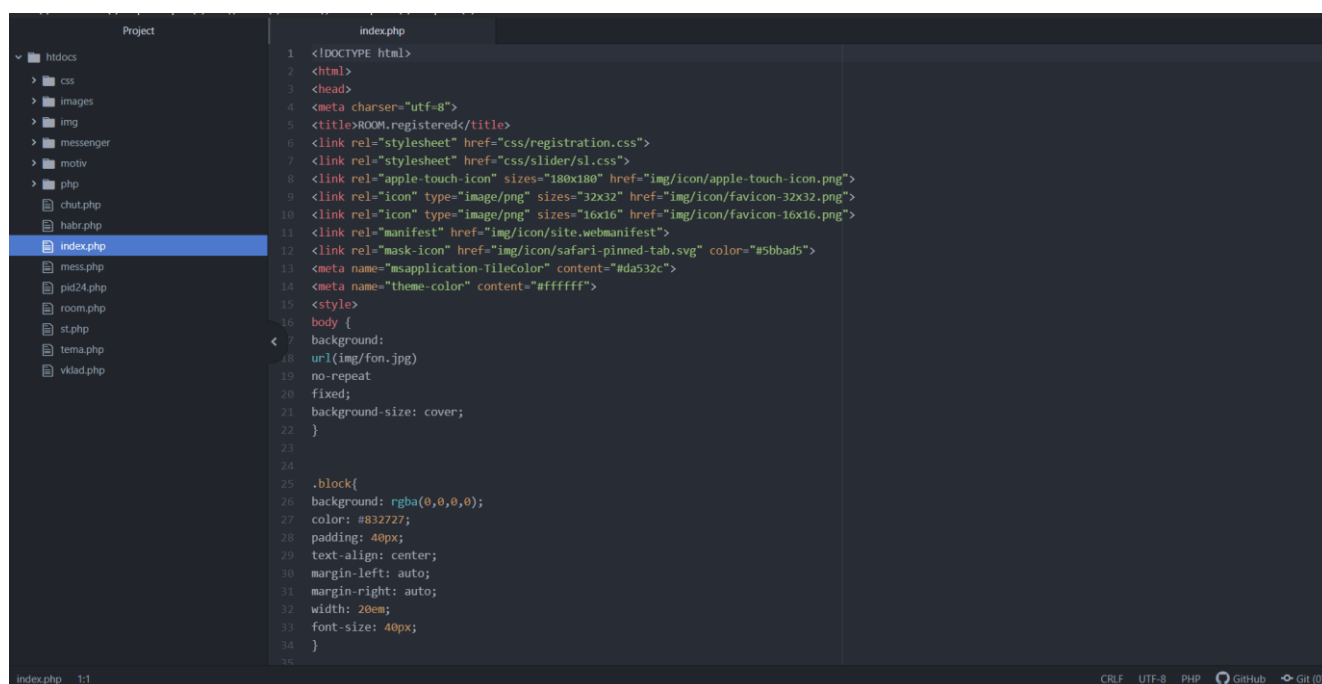
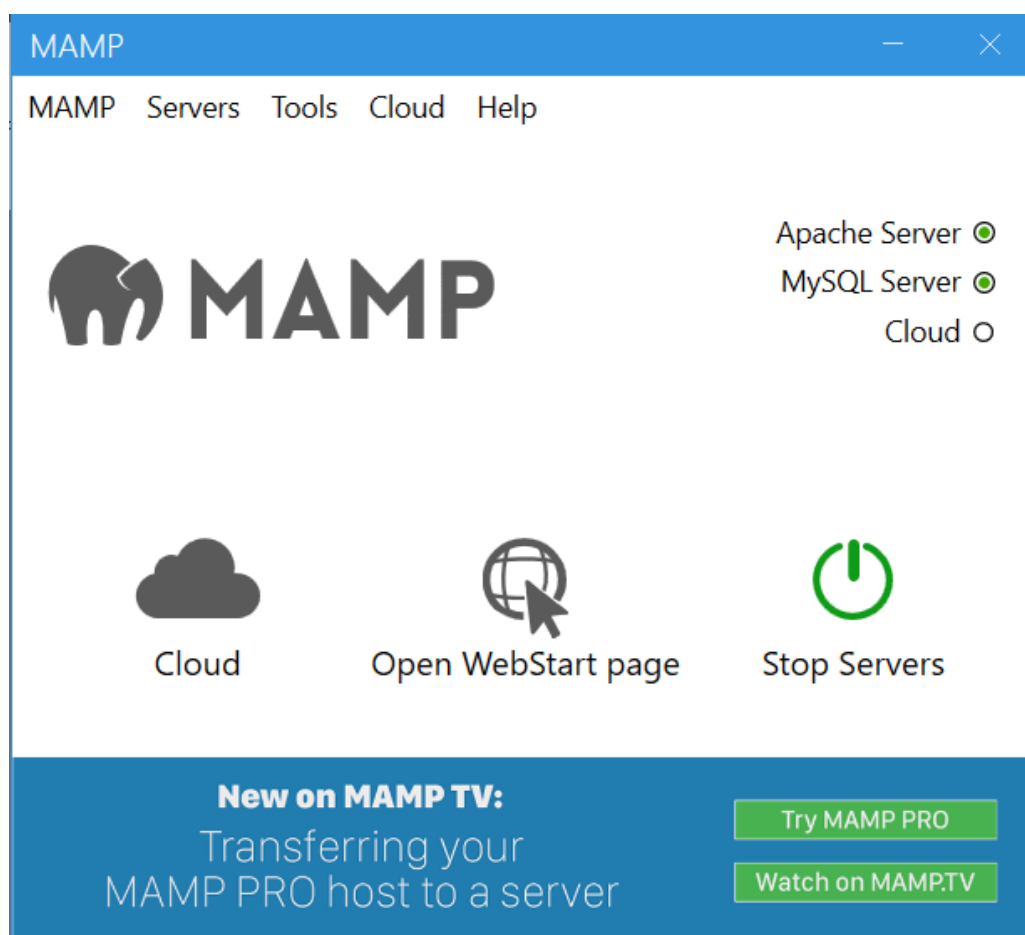


Рисунок 3.1 – зображення інтерфейсу редактора коду Atom

На рисунку також можна спостерігати, що цей редактор є досить простим в керуванні та пришвидшує написання платформи. За допомогою вбудованого розпізнавання тексту редактор автоматично підсвічує необхідні елементи чи ключові команди у файлі[38].

					КР.КІ. 8351649.00.00.000.ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		46

Усі файли, які будуть задіяні в проекті повинні мати розширення php. Такий формат необхідний, щоб код, який буде написаний на мові php виконувався, і сервер міг його розпізнавати. PHP мова є дуже зручною, якщо іде мова про збереження даних у базу даних та сервер. Для того, щоб php код виконувався, потрібно використовувати сервери, в яких і буде зберігатись вся інформація. Для того, щоб реалізувати платформу і проводити різного виду тестування, створили різноманітні віртуальні сервери. Один із найкращих і найпростіших прикладів хорошого віртуального серверу є MAMP. Для локальних перевірок, де не потрібно використовувати великий функціонал, можна завантажувати безкоштовну версію. На рисунку 3.2 зображено зовнішній вигляд програми MAMP та функціонал.



Рисунко 3.2 – MAMP –віртуальний сервер

Графічний інтерфейс користувача було вирішено виконати у червоних та чорних тонах. Переливання та анімації також виконані в цих кольорах. На

					КР.КІ. 8351649.00.00.000.ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		47

рисунку 3.3 візуально зображено, в яких тонах буде виглядати графічний інтерфейс.

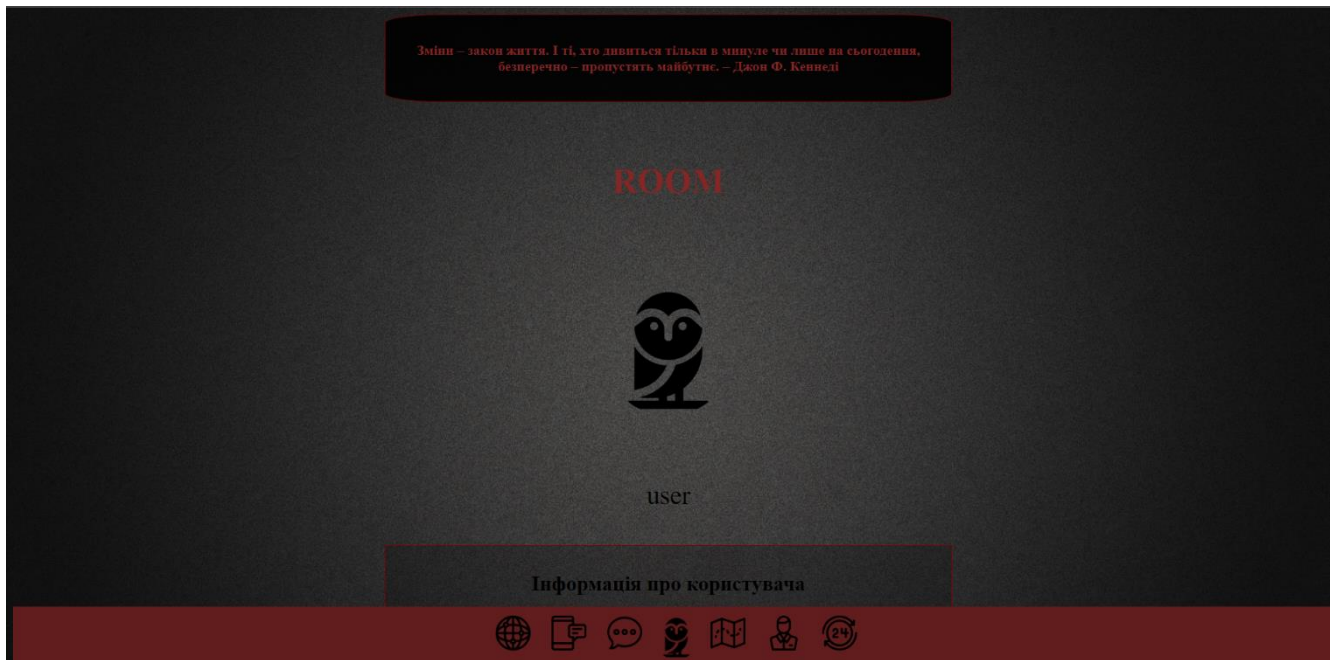


Рисунок 3.3 – зображення графічного інтерфейсу користувача

На центральній частині екрану зображений логотип і символ сайту. Сова – це символ мудрості, який відповідає тематиці платформи. У нижній частині розміщується навігаційне меню, в якому можна переходити на інші частини сайту та користуватись його функціоналом. Зараз на рисунку вищевказана головна сторінка користувача. У ній буде відображатись вся особиста інформація про користувача, яку сам користувач захоче змінити або ввести.

В першу чергу, для того, щоб отримати доступ до сайту, потрібно зареєструватись та авторизуватись на платформі. На рисунку 3.4 та рисунку 3.5 зображено зовнішній вид верифікації користувача. Для приємного зовнішнього вигляду та зручності, форму було створено у вигляді слайдеру. Такий вигляд авторизації економить місця і не робить візуально переповнений інтерфейс.

					КР.КІ. 8351649.00.00.000.ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		48

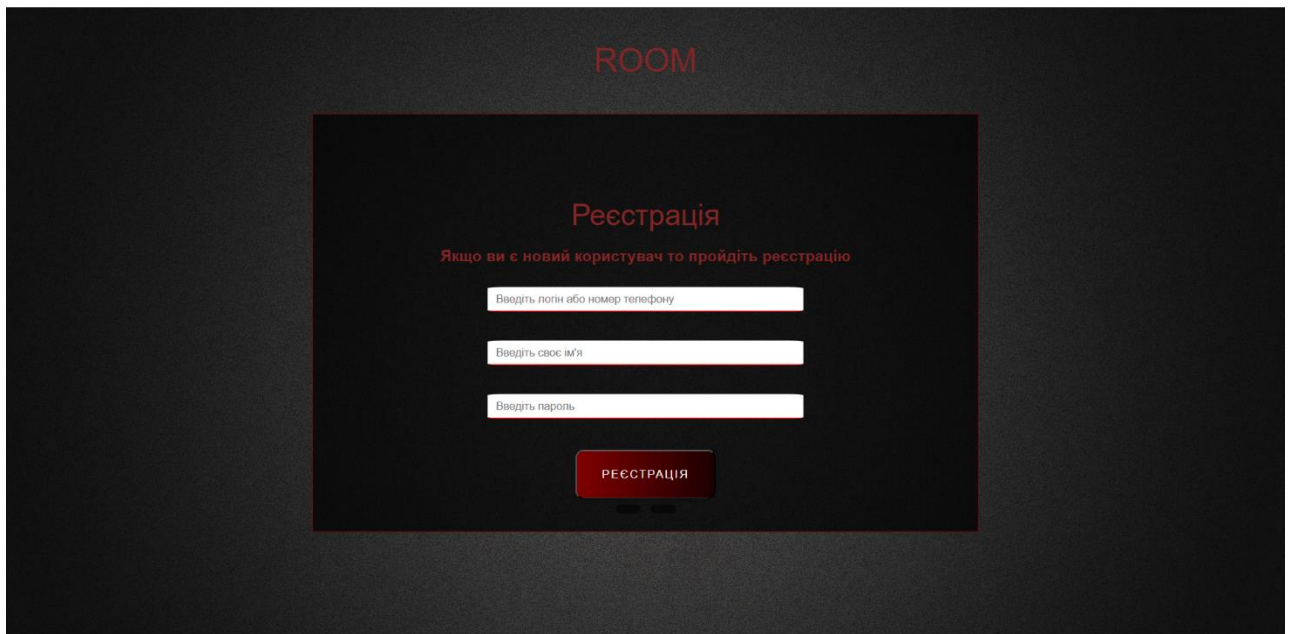


Рисунок 3.4 – реєстраційне віконце для створення нового облікового запису

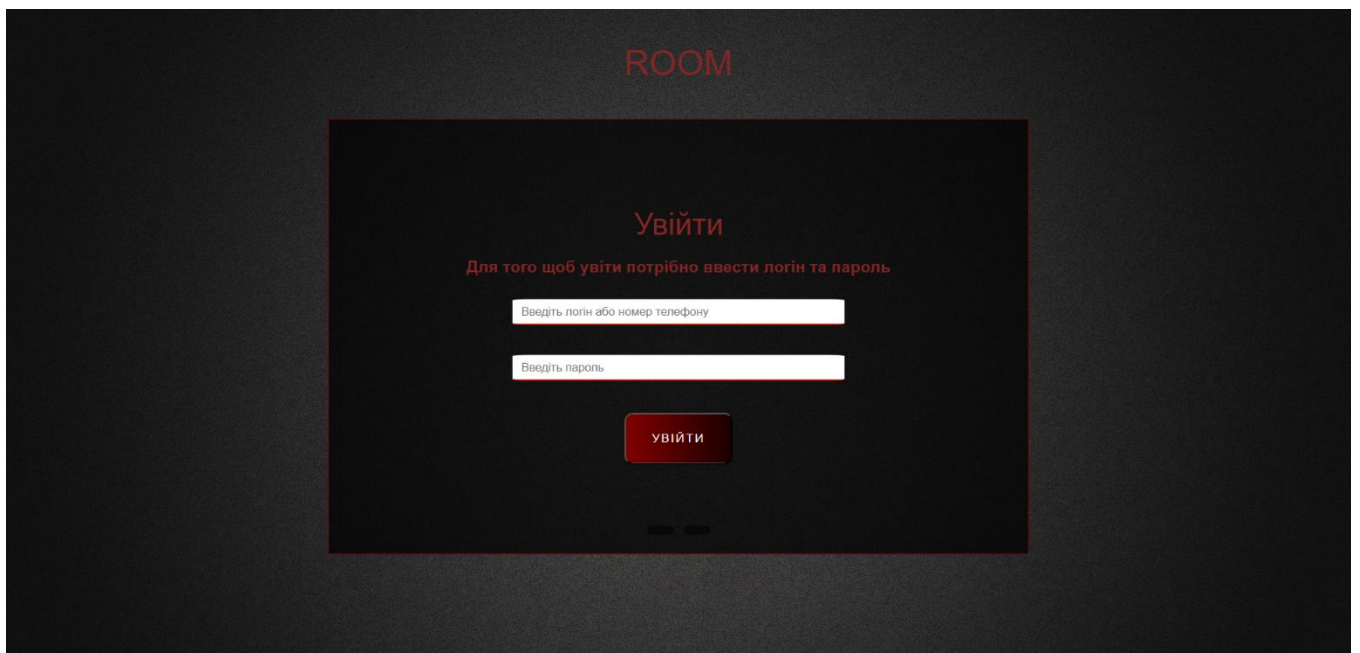


Рисунок 3.5 – вікно авторизації уже зареєстрованного користувача

Можна спостерігати під кнопкою реєстрація та увійти дві прямокутні клавіші. Якщо нажимати на них по черзі, то можна буде переміщатись між вікном реєстрації та авторизації. Із сторони користувача, все швидко відбувається, і система одразу переносить на головну сторінку. На рисунку 3.6 реалізовано на програмному рівні пересилання даних, які ввів користувач.

					КР.КІ. 8351649.00.00.000.ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		49

```

<form action="php/input.php" method="post" >
  <input class="input" type="text" name="login" id="login" placeholder="Введіть логін або номер телефону" ><br><br>
  <input class="input" type="text" name="name" id="login" placeholder="Введіть своє ім'я"><br><br>
  <input class="input" type="password" name="password" id="login" placeholder="Введіть пароль"><br><br>
  <button type="submit" class="gradient-button">Реєстрація</button>
</form>

```

Рисунок 3.6 - приклад реалізації функції вводу даних

За допомогою верстки html та стилів css реалізовано введення даних за допомогою тегу <form>. В цьому тезі також вказані параметри, а саме action, куди введенні дані повинні передаватись та method = "post". Другий параметр охарактеризовує, як саме будуть передаватись дані, точніше їхній метод. Для того, щоб дані зберігались та передавались коректно, уже використовується мова php. За допомогою зміни, в яку поміщають дані із форм масив \$_POST уже зберігає в собі дані. На рисунку 3.7 зображено, як передаються дані із форми і обробляються мовою програмування php[43].

```

$login = filter_var(trim($_POST['login']));
$password = filter_var(trim($_POST['password']));

```

Рисунок 3.7 – отримання даних з форми та поміщення їх у змінні.

Після того, як дані зберігаються, вони обробляються та відправляються у базу даних. Також можна спостерігати, що дані, які передаються у масиві помістили в змінну. Це потрібно для того, щоб потім ці дані, що знаходяться в змінній мова запитів mysql змогла зчитати та передати уже на сервер у відповідну таблицю. Такий метод дозволяє передавати дані, які ввів користувач. Для покращення якості перед масивом вказані такі функції, як trim та filter_var. Вони потрібні для того, щоб пусті символи стирались та фільтрує дані.

Одним із головних елементів у платформі являється навігаційне меню. Воно зафіксоване у нижній частині сайту. В першу чергу, це створено для

					КР.КІ. 8351649.00.00.000.ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		50

зручності та в універсальності сайту. На даний момент небагато веб ресурсів містять в собі меню, яке знаходиться в нижній частині сайту. На рисунку 3.8 зображено зовнішній вигляд меню.



Рисунок 3.8 – зображення навігаційного меню платформи

Просте у використанні та інтуїтивно зрозуміле меню допоможе швидко переміщатись між вкладками та шукати різноманітну інформацію. За допомогою стилів css та посиланнями html була можливість створити інструмент для переміщення по сайту. На рисунку 3.9 та 3.10 можна спостерігати виконану верстку та стилі, які дають можливість посилатись на різноманітні файли у приємному вигляді.

```
<div class="navbar">
<a href="habr.php"><img style="padding:10px" src = "img/room.png" alt="you don't" title="Статті" width = "40" height = "40" class="scale2"></a>
<a href = "chut.php"><img style="padding:10px" src = "img/messenger.png" alt="you don't" title="Планнер" width = "40" height = "40" class="scale2"></a>
<a href = "mess.php"><img style="padding:10px" src = "img/mess.png" alt="you don't" title="Месенджер" width = "40" height = "40" class="scale2"></a>
<a href = "room.php"><img src = "img/user.png" alt = "It's your face" width = "50" title="Головна сторінка" height = "50" class="round"></a>
<a href="vklad.php"><img style="padding:10px" src = "img/map.png" alt="you don't" title="Навігація" width = "40" height = "40" class="scale2"></a>
<a href = "php/exit.php" ><img style="padding:10px" src = "img/main.png" alt="you don't" title="Вихід" width = "40" height = "40" class="scale2"></a>
<a href="pid24.php"><img style="padding:10px" src = "img/teh.png" alt="you don't" title="Налаштування" width = "40" height = "40" class="scale2"></a></div>
```

Рисунок 3.9 – опис навігаційного меню версткою html

Згідно зображення, можна спостерігати різноманітні параметри, які допомагають виглядати іконкам меню приємним виглядом. Також можна спостерігати class = "navbard", який і надає основний вигляд меню.

```
.navbar {
  overflow: hidden;
  background-color: #601D1D;
  color: #black;
  position: fixed;
  bottom: 0;
  width: 100%;
  padding: 1px;
  text-align: center;
}
```

Рисунко 3.10 – опис розміження навігаційного меню стилями CSS

Неозброєним оком можна зрозуміти, що тут описується вся загальна інформація про розміжування блоку та часткове розміщення об'єктів у ньому. Відповідно, для того, щоб меню відображалось завжди, його зафіксували налаштуванням `position: fixed;` та розтягнути на всю ширину екрану за допомогою `width: 100%;`

На головному екрані можна спостерігати інформацію про користувача та його ім'я, яке потрібно ввести при ресетрації користувача. Для того, щоб дані можна було ввести в навігаційному меню, відображається іконка. За порядком вона остання, якщо по ній перейти, то можна спостерігати поля, які заповнюються. На рисунку 3.11 зображено мінімальні налаштування профілю користувача та його особиста інформація. В цих полях потрібно вказувати інформацію згідно вимог, які вписані навпроти кожного поля. Таким чином, кожен бажаючий, який захоче перейти по конкретному користувачеві, зможе отримати інформацію, яку сам власник свого облікового запису зможе надати [36].

					КР.КІ. 8351649.00.00.000.ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		52

Для того щоб здійснити зміни введіть текст в полі та натисніть зберегти

Дані які будуть вказуватись в цій таблиці будуть відображатись на вашій основній сторінці

- вкажіть свій вік

- вкажіть свою ідею

- вкажіть свою адресу

- вкажіть щось про себе

- де ви навчається

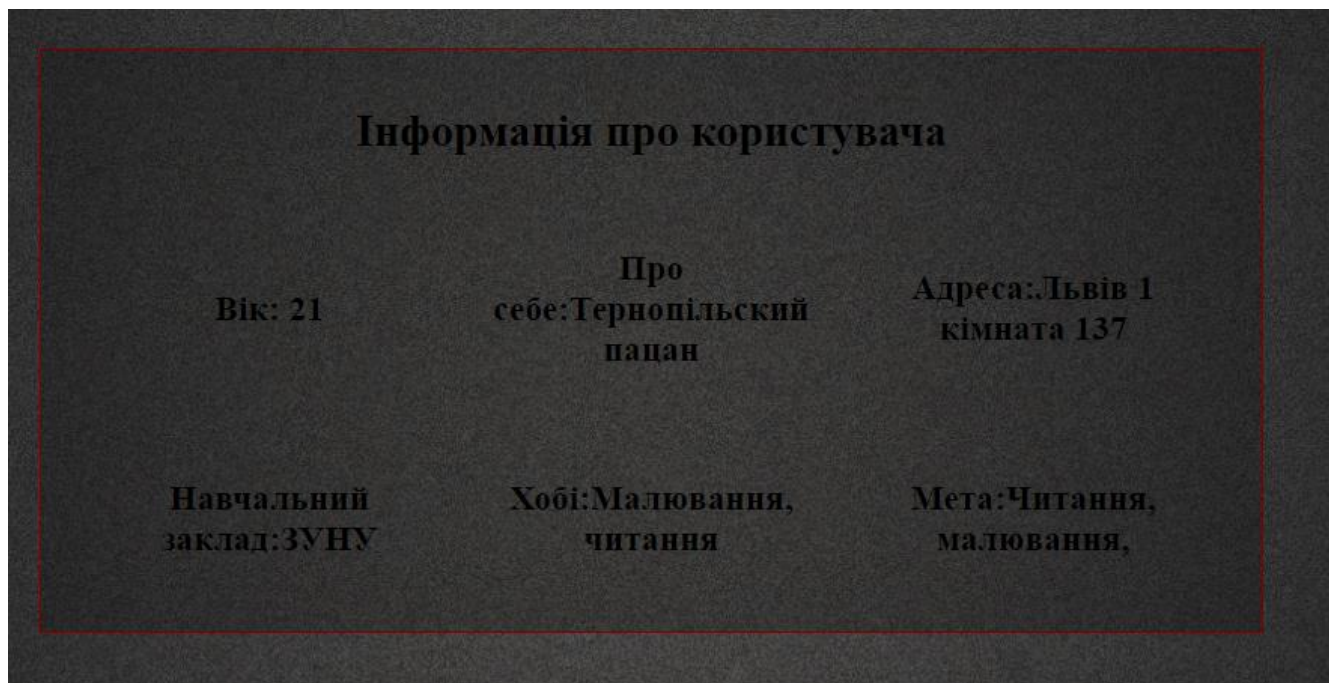
- яке у вас хобі

ЗБЕРЕГТИ

Рисунок 3.10 – відображення сторінки із мінімальними налаштуваннями профільної сторінки

За допомогою такого функціоналу, сторінка користувача стає більшою інформативною та цікавішою для осіб, які зацікавились користувачем. Також це не одноразова функція. Якщо інформація про користувача змінюється, чи її потрібно змінити, то власник облікового запису може в будь-який момент це зробити. Якщо виконати все правильно і натиснути зберегти, то інформація відобразиться на основній сторінці у вигляді таблиці, що зображено на рисунку 3.11.

					КР.КІ. 8351649.00.00.000.ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		53



Рисунко 3.11 – загальна інформація про користувача, яку можна змінювати з часом

Так як це допоміжна навчальна система для студентів, то відповідно платформа повинна містити в собі інструментарій для створення різноманітних своїх публікацій та робіт, які в майбутньому можуть допомогти іншим користувачам. Перша по списку іконка, у вигляді планети із мередіанами. Вона символізує зв'язок із всім світом, тобто всі користувачі, які забажають ознайомитись із інформацією тієї чи іншої теми, можуть отримати студенти в ході своїх досліджень. На рисунку 3.12 зображений перелік робіт та статей з якими можна ознайомитись. В таких роботах, важливо правильно формулювати тему та вказувати короткий опис, який може зацікавити свого користувача.

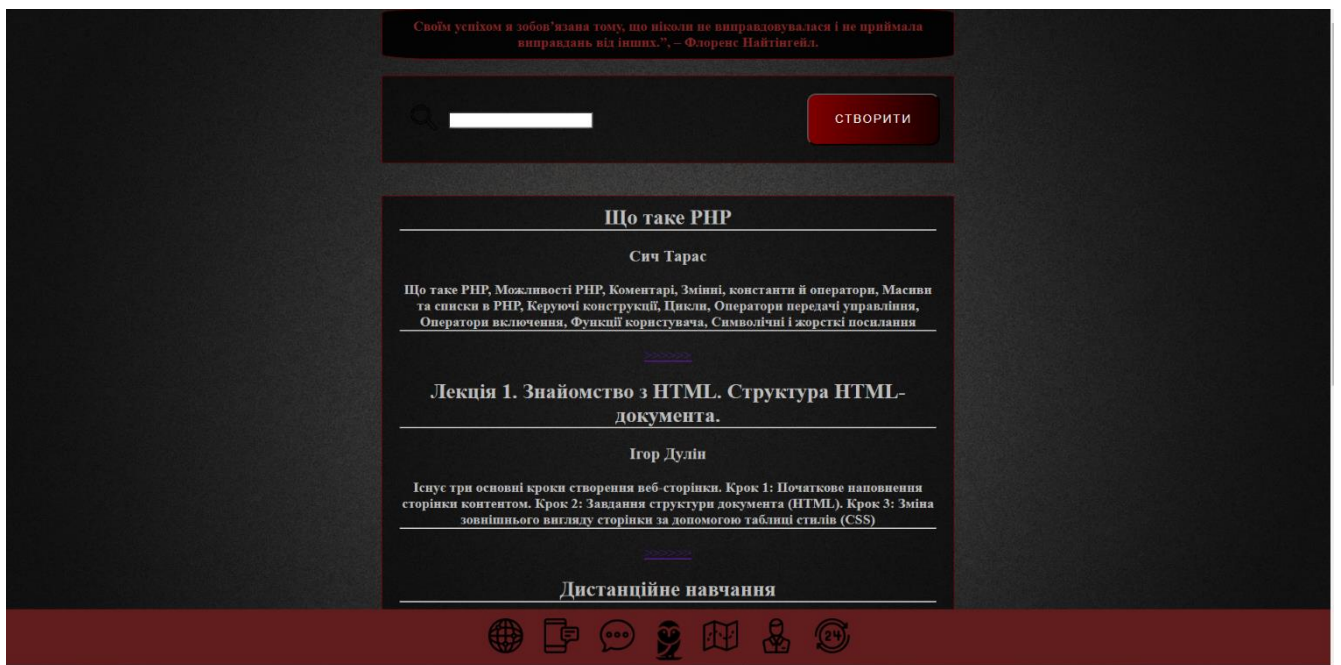


Рисунок 3.12 – відображення перелік статей, які можна прочитати нажавши на посилання під кожною статєю

У цей функціонал ще також входить можливість створювати свої статті та роботи, в яких можна ділитись досвідом та знаннями. Статті можуть бути на тематики будь-якого характеру, так як платформа є універсальною і створювалась для усіх видів навчання та наук. Також на верхній частині сайту можна спостерігати кнопку створити. Ця кнопка переміщує нас у конструктор, який можна спостерігати на рисунку 3.13.

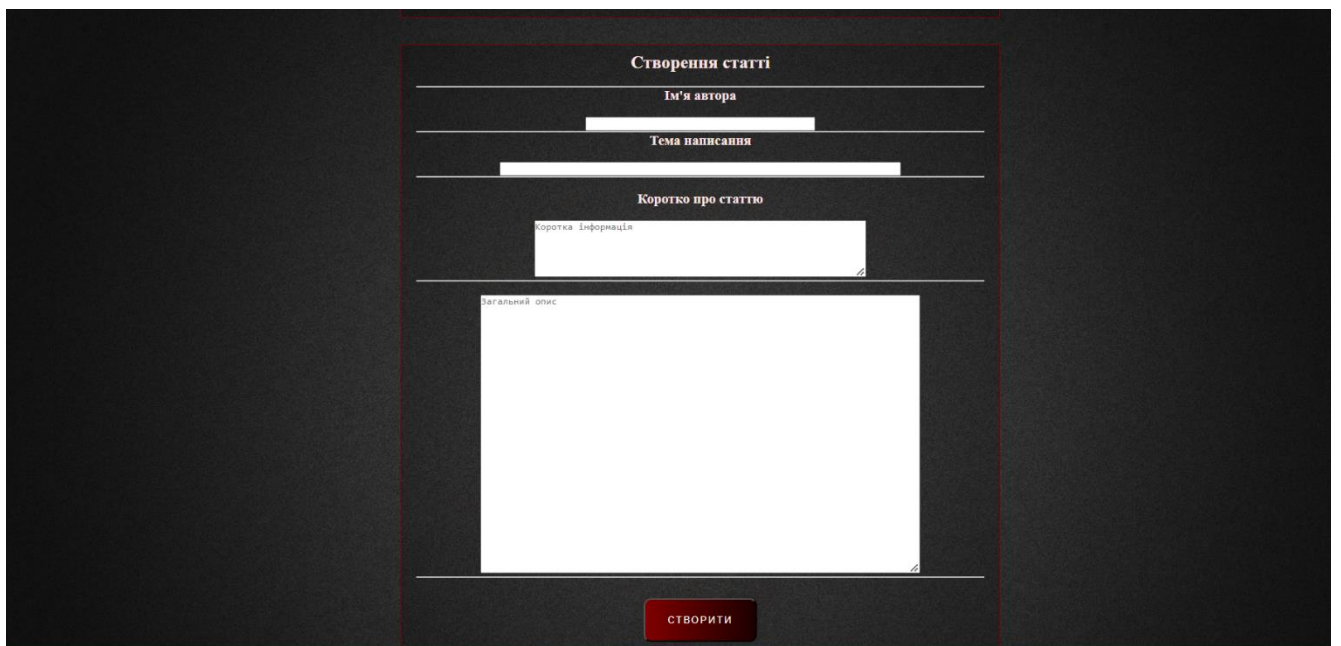


Рисунок 3.13 – конструктор статей

					КР.КІ. 8351649.00.00.000.ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		55

На рисунку можна спостерігати поля, які власне і містять в собі необхідні дані. Усі дані, які ввели в полях, передаються в базу даних та уже там обробляються, щоб вивестись в список статей, які ми спостерігали на рисунку 3.12. Для збереження інформації та передачі її в базу даних, реалізовується не тільки код php, а й mysql. Тобто, для того, щоб підключитись до бази даних та передавати дані, використовують одне із підключень, яке показано на рисунку 3.13.

```
<?php

$conn = mysqli_connect("localhost", "root", "root", "uniroom");
if (!$conn) {
    die("Ошибка: " . mysqli_connect_error());
}
$user = mysqli_real_escape_string($conn, $_POST["user"]);
$opys = mysqli_real_escape_string($conn, $_POST["opys"]);
$tex = mysqli_real_escape_string($conn, $_POST["tex"]);
$tema = mysqli_real_escape_string($conn, $_POST["tema"]);
$sql = "INSERT INTO statta (user, opys, tex,tema) VALUES ('$user','$opys','$tex','$tema')";
if(mysqli_query($conn, $sql)){
    echo "Данные успешно добавлены";
} else{
    echo "Ошибка: " . mysqli_error($conn);
}
mysqli_close($conn);
header('location:/habr.php');
```

Рисунок 3.13 – демонстрація підключення до бази та передача даних

Можна спостерігати, що для підключення і введення інформації вказують mysql команди в середині зміни, яку створили на початку, для підключення. Це необхідно для того, щоб дані швидко змогли розміститись в базі даних, у відповідній таблиці[48].

Ще одним важливим елементом у навчанні студентів є збереження даних. Це можуть бути замітки, список подій, які необхідні попорядку здійснити у програмному коді, чи звичайний план роботи. Для цього було теж

реалізована вкладка для таких цілей. На рисунку 3.14 графічно відображається інтерфейс записників та планерів.

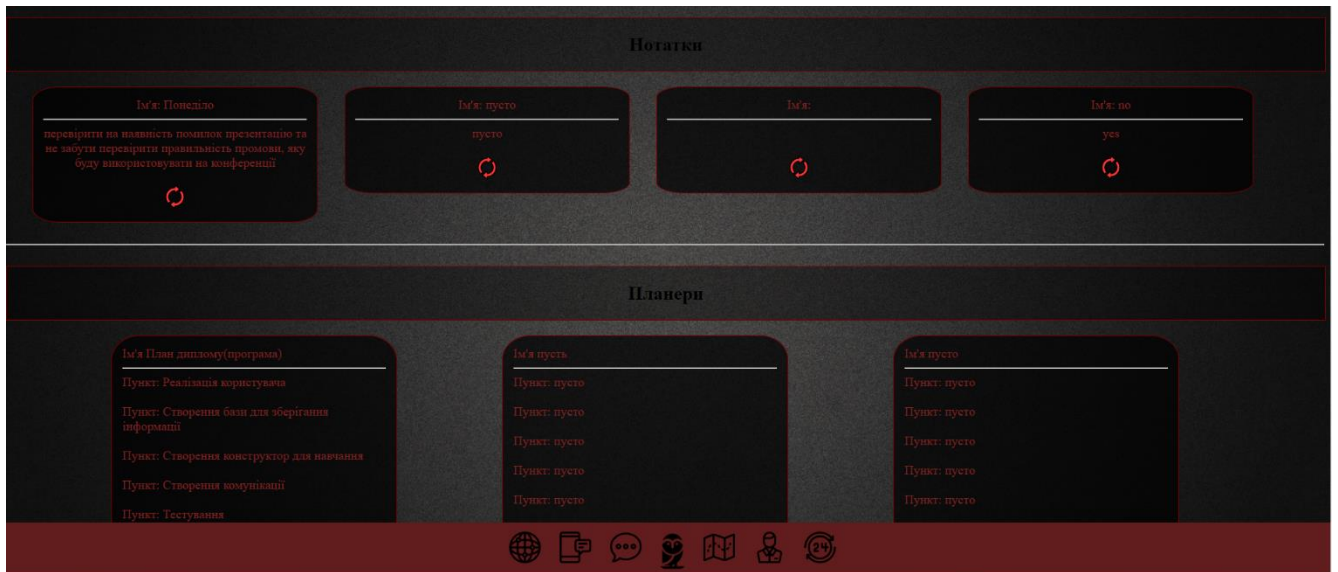


Рисунок 3.14 – графічне відображення інформації у записниках, та планерах

Інформація, яка тут відображається може бути корисною для всього. Тобто універсальність такого функціоналу велика, оскільки використання записників, чи побудови планів, завжди є ключовим елементом в усіх проектах. Уся інформація, яка тут вказана вона також зберігається у базі даних, відповідно, якщо ця платформа буде розміщена на сервері, то туди доступ буде із будь-якої точки світу, де є доступ до інтернету. Також для зміни інформації, достатньо просто натиснути на відповідний символ, який відображений на кожному блоці та переписати дані, які уже застаріли. На рисунку 3.15 демонструється зміна інформації у пунктах плану.

Рисунок 3.15 – відображення плану та оновлення даних

Можна спостерігати, що дані, які ввів користувач минулого разу, також відображаються. Тобто, для того, щоб щось змінити, не обов'язково переписувати весь план, а лише його ключові моменти. На рисунку 3.16 можна спостерігати в реалізації коду, як відбувається оновлення даних.

```

$userid = mysqli_real_escape_string($conn, $_POST["id"]);
$username = mysqli_real_escape_string($conn, $_POST["name"]);
$usertext = mysqli_real_escape_string($conn, $_POST["text"]);

$sql = "UPDATE note SET name = '$username', text = '$usertext' WHERE id = '$userid'";
if ($result = mysqli_query($conn, $sql)) {
    header("Location: ../../chut.php");
    exit();
} else {
    echo "Помилка: " . mysqli_error($conn);
}
else {
    echo "Некоректні дані";
}

```

Рисунок 3.16 – відображення даних які користувач ввів та оновлення їх за допомогою мови mysql

Також одним із основних елементів, який необхідний для навчання – це обговорення різноманітних тем. Тобто, для того, щоб поділитись своїми

заннями чи думкою, не потрібно завжди створювати статтю чи інші видання. Для того, щоб поділитись своєю думкою і не переживати, що вона буде некоректною чи не дуже правильною, можна скористатись форумом. Також платформа дає можливість комфортно користуватись, нажавши на іконку, що за допомогою підказки буде відображати слово форуми. На рисунку 3.17 можна спостерігати список форумів, за допомогою, яких можна і спілкуватись[49] .

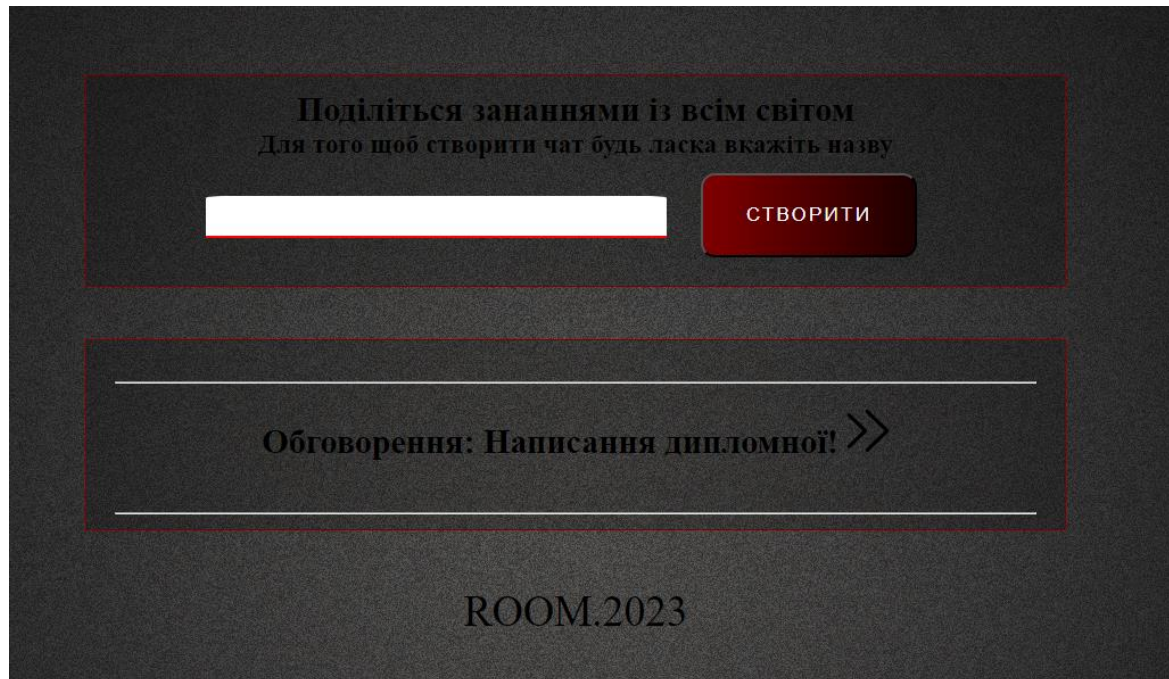


Рисунок 3.17 відображення створеного форум чата

Форуми завжди були популярні, і відповідно цей елемент, також грамотно зможе влитись у процес навчання та зможе принести свою користь. Для того, щоб створити свій форум, достатньо буде просто вказати назву форуму та натиснути на кнопку створити. Через секунду цей чат уже буде відображатись у списку доступних чатів, куди можна буде перейти та переглянути інформацію. На рисунку 3.18 можна спостерігати форум.

					КР.КІ. 8351649.00.00.000.ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		59

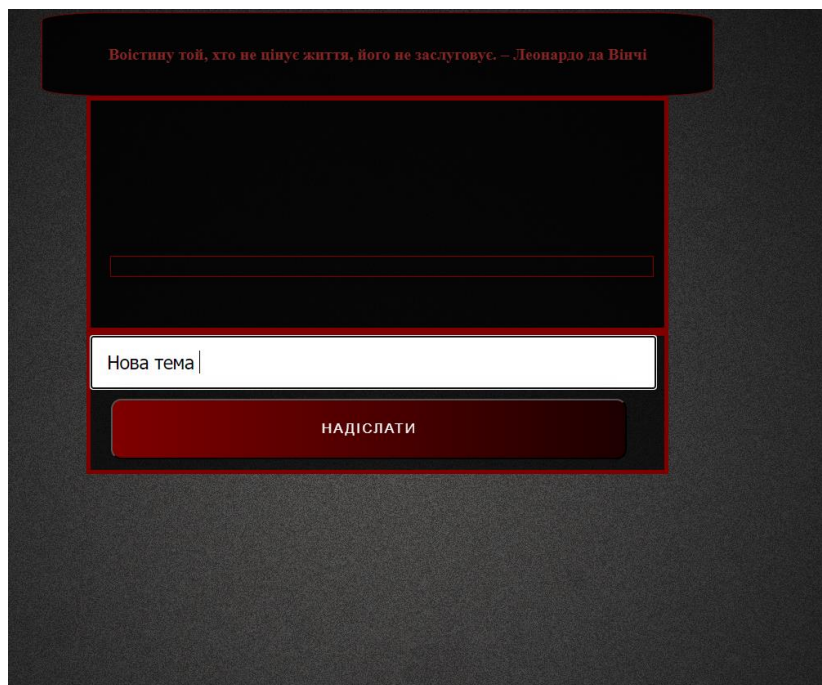


Рисунок 3.18 – відображення чату форуму та зовнішній вигляд відправки повідомлення

Для того, щоб відобразити коректно інформацію та надсилати повідомлення, було створено спеціальний скрипт, який виводить інформацію із форми у базу даних, щоб повідомлення було доступне для всіх, та відправляє назад у блок вище. На рисунку 3.18 і 3.19 відображено алгоритм роботи форум чату.

```
<div class = "block_mes">
  <div class = "ms">
    <?php
    foreach($result as $row){
      echo "<h3>".$row['mess']. "</h3>";
    }

    mysqli_free_result($result);
  } else{
    echo "Ошибка: " . mysqli_error($mysql);
  }
  mysqli_close($mysql); ?>
</div>
```

Рисунок 3.18 – php скрипт, який виводить інформацію із бази даних та виводить на екран

					КР.КІ. 8351649.00.00.000.ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		60

Даний PHP-код використовує цикл `foreach` для перебору елементів масиву `$result`. Кожен елемент масиву представляє рядок з результатами запиту до бази даних. Для кожного рядка, код виводить заголовок рівня "h3", який містить значення поля "mess" з кожного рядка. Наприклад, якщо поле "mess" містить значення "Hello World", то виводиться заголовок `<h3>Hello World</h3>`. Після виводу всіх елементів, викликається функція `mysqli_free_result()`, яка звільняє пам'ять, відведену для результатів запиту до бази даних[42].

```
<?php
if (isset($_POST["message"])) {

    $mysql = mysqli_connect("localhost", "root", "root", "users");
    if (!$mysql) {
        die("помилка: " . mysqli_connect_error());
    }
    $mess = mysqli_real_escape_string($mysql, $_POST["message"]);

    $sql = "INSERT INTO messages (mess) VALUES ('$mess')";
    if(mysqli_query($mysql, $sql)){
        echo "Дані успішно добавлені";
    } else{
        echo "помилка: " . mysqli_error($mysql);
    }
    mysqli_close($mysql);
}

header('Location:messenger.php');
?>
```

Рисунок 3.19- опис алгоритму передачі даних в базу даних

Даний PHP-код перевіряє, чи була надіслана форма з повідомленням (за допомогою методу POST та полем "message"). Якщо так, то код встановлює з'єднання з базою даних MySQL, використовуючи дані для підключення (хост, користувач, пароль, назва бази даних). Якщо з'єднання успішно встановлено, код отримує значення повідомлення з полів форми та екранує його за допомогою функції `mysqli_real_escape_string()`, щоб уникнути SQL-ін'єкцій. Далі код формує SQL-запит для вставки даних в таблицю

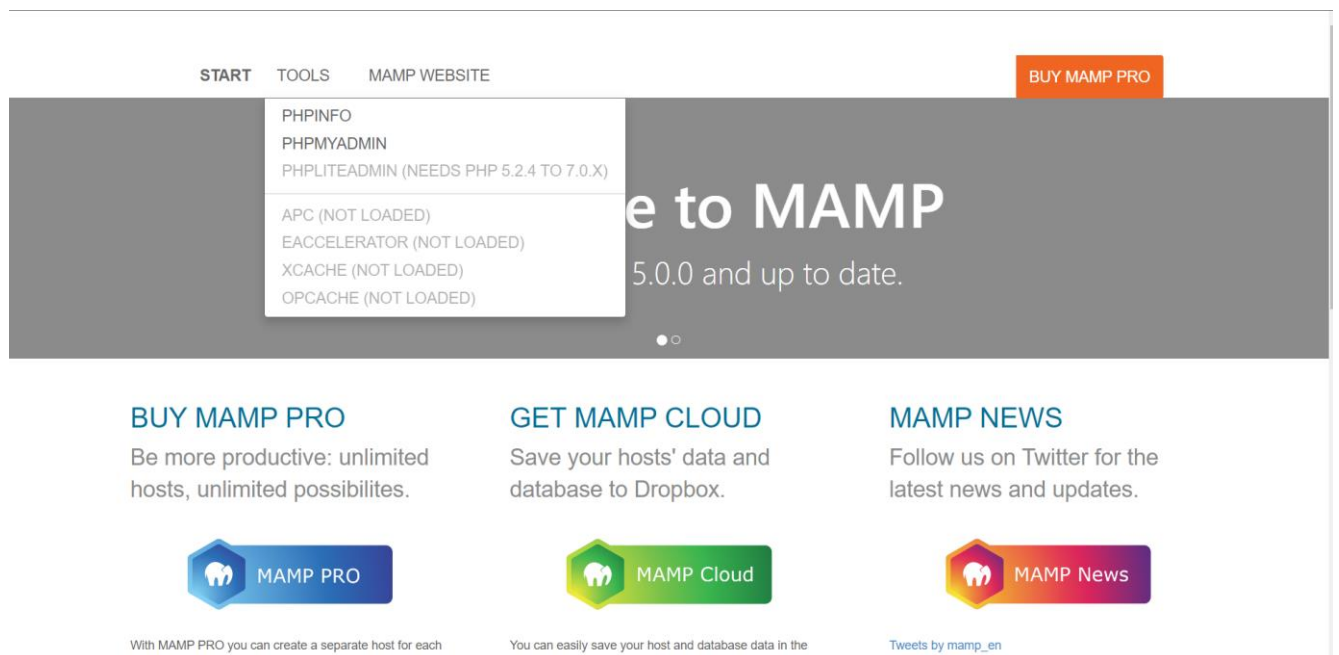
"messages" з використанням отриманого повідомлення. Запит виконується за допомогою функції `mysqli_query()`. Якщо запит виконується успішно, виводиться повідомлення "Дані успішно додані". У протилежному випадку, виводиться повідомлення про помилку, яке містить деталі про помилку MySQL. На завершення, з'єднання з базою даних закривається за допомогою функції `mysqli_close()`.

3.2 Реалізація системи комунікації

Бази даних є невід'ємною частиною багатьох сучасних додатків та систем. Вони виконують важливу роль у зберіганні та організації великого обсягу даних, які використовуються для різних цілей. Основна мета баз даних полягає у забезпеченні ефективного, безпечного та структурованого зберігання і доступу до інформації. Одна з ключових важливостей баз даних полягає в тому, що вони дозволяють ефективно управляти даними. База даних забезпечує структуровану організацію даних у вигляді таблиць, де кожен рядок представляє запис, а кожний стовпчик - поле даних. Це дозволяє швидко здійснювати пошук, сортування, фільтрацію та аналіз даних. Крім того, бази даних забезпечують цілісність даних. Вони використовують механізми перевірки правильності та валідації даних, що дозволяє запобігти введенню некоректної або невалідної інформації. Бази даних також забезпечують можливість контролю доступу до даних, що дозволяє встановлювати права доступу для користувачів і забезпечувати конфіденційність та безпеку інформації. Бази даних також забезпечують зв'язок між різними сутностями та об'єктами. Зв'язки можуть бути встановлені між таблицями шляхом використання ключів, таких як первинний ключ і зовнішній ключ. Це дозволяє створювати складні структури даних, які відображають взаємозв'язки між різними сутностями, такими як користувачі, замовлення, товари тощо.

					КР.КІ. 8351649.00.00.000.ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		62

База даних, яка буде використовуватись для створення серверної частини платформи уже встановлена в віртуальний сервер МАРМ. Для того, щоб перейти в конструктор бази даних, потрібно натиснути на веб сайт, і користувача автоматично повинно перенести на локальний сайт. На рисунку 3.20 можна візуально побачити дані про сервер та його веб сайт.



рисунко 3.20 – графічне відображення сервера МАРМ та його елементи навігації

Для того, щоб перейти у базу даних, необхідно натиснути на випадяючому меню `phpmyadmin`. Також, на цьому сайті можна спостерігати інформацію на рахунок підключення до бази даних. Тобто, який пароль, логін та ім'я бази даних, куди будуть надходити дані. Для кращого розуміння, база даних виконана у мінімальному графічному дизайні, який відображений на рисунку 3.21.

					КР.КІ. 8351649.00.00.000.ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		63

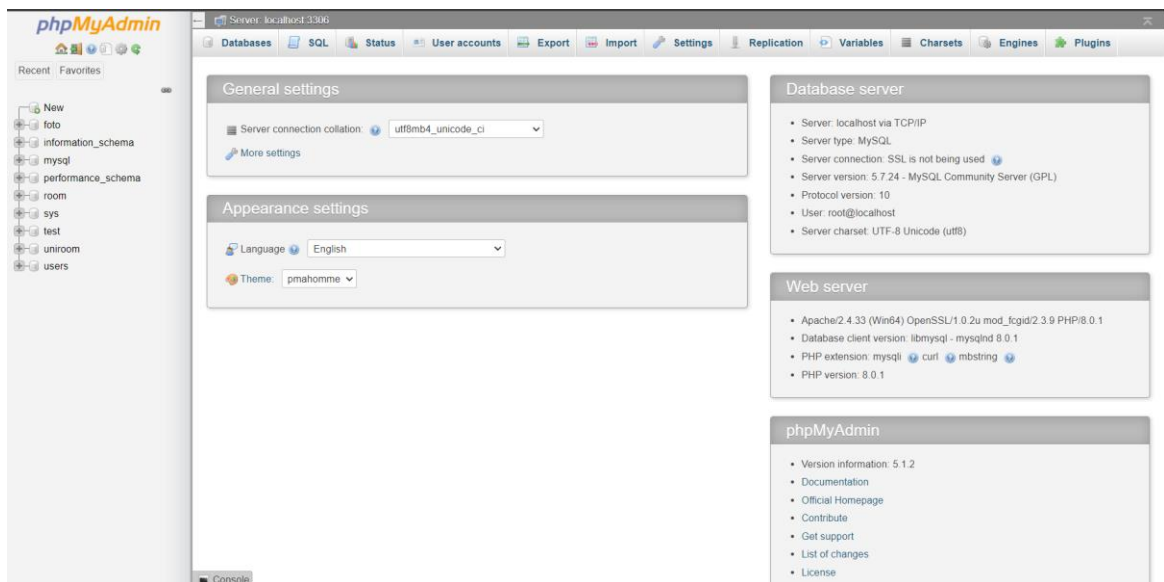


Рисунок 3.21 – відображення бази даних із всіма функціями та елементами

На рисунку спостерігається уже перелік баз даних, які були створені розробниками. Саме перелік цих баз і дають можливість використовувати цю базу даних, як і консольний інструмент, так і графічно оформлений. Щоб почати роботу із серверною частиною, в першу чергу потрібно створити базу, де буде зберігатись вся інформація про сайт та користувача. Щоб створити базу даних, достатньо у лівому меню нажати посилання new та вказати назву бази даних. Усі дані, які надсилаються у базу, зберігаються таблицях. Таблицю можна створити, як графічно, так і консольним шляхом[43,44]. На рисунку 3.22 можна побачити відображення графічного інструментарію для створення таблиць.

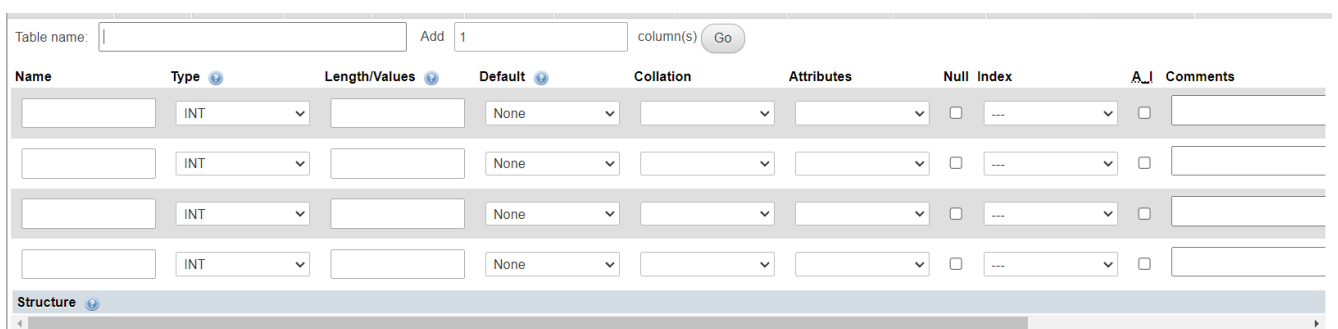


Рисунок 3.22 – графічне відображення побудови таблиці для зберігання даних

Таким чином, для того, щоб дані зберігались потрібно створювати окремо таблицю для різних цілей. Якщо в одну таблицю вказувати всі дані, то в подальшому це зможе погіршити розуміння загальної картини, створить незручності для знаходження помилки та буде перешкоджати написання скрипту, який буде виводити потім дані із таблиць. На рисунку 3.23 зображено перелік баз даних, які зберігають в собі дані усіх користувачів та на прикладі уже вказано таблицю, де дані уже записані.

Showing rows 0 - 2 (3 total, Query took 0.0005 seconds.)

SELECT * FROM `plan`

Number of rows: 25 Filter rows: Search this table Sort by key: None

id	name	plan1	plan2	plan3	plan4	plan5
1	План диплому(програма)	Реалізація користувача	Створення бази для зберігання інформації	Створення конструктор для навчання	Створення комунікації	Тестування
2	пусто	пусто	пусто	пусто	пусто	пусто
3	пусто	пусто	пусто	пусто	пусто	пусто

Рисунок 3.23 – відображення таблиці у графічному інтерфейсі та приклад заповнення рядків даними

Щоб мати можливість надіслати дані в базу і не отримати ніяких проблем із зв'язком, створюють з'єднання ще на початку роботи алгоритму. На рисунку 3.24 можна побачити, як здійснюється підключення до бази даних[41].

```
// якщо запит GET існує параметр id
if ($_SERVER["REQUEST_METHOD"] === "GET" && isset($_GET["id"])) {
    $userid = mysqli_real_escape_string($conn, $_GET["id"]);
    $sql = "SELECT * FROM plan WHERE id = '$userid'";
    if ($result = mysqli_query($conn, $sql)) {
        if (mysqli_num_rows($result) > 0) {
            $row = mysqli_fetch_assoc($result);
            $username = $row["name"];
            $userplan1 = $row["plan1"];
            $userplan2 = $row["plan2"];
            $userplan3 = $row["plan3"];
            $userplan4 = $row["plan4"];
            $userplan5 = $row["plan5"];
        }
    }
}
```

Рисунок 3.24 – підключення до бази даних та перевірка інформації в таблиці

Даний код перевіряє можливість проблем та помилок у ході роботи. За допомогою оператора if (), перевіряється чи знаходяться дані, які вказані в id змінній, що передалась GET методом. У випадку, якщо інформація не знаходиться згідно алгоритму, систему створює альтернативне повідомлення, в якому повідомляється, що інформація не знайдена або пуста поле у випадку цього коду. Загальне графічне відображення схеми таблиць баз даних наведено в додатку А.

Також на рахунок безпеки використовуються бази даних. Одним із найбільш важливим моментом у платформах, являється безпека користувачів, збереження паролей та особистих даних. При реєстрації та авторизації, дані користувача, точніше пароль, за допомогою допоміжної функції шифрується, та не дає змоги шахраям отримати доступ до облікового запису. На рисунку 3.25 і 3.26 відображено код, який шифрує дані користувача та відображення цих даних у таблиці[47,48].

```
<?php
mysql = new mysqli('localhost','root','root','uniroom');
login = filter_var(trim($_POST['login']));
name = filter_var(trim($_POST['name']));
password = filter_var(trim($_POST['password']));

password = md5("apsfj".$password."sdfsdffasfasdterry");
mysql ->query("INSERT INTO `users`(`login`, `name`, `password`) VALUES('$login','$name','$password')");
mysql ->close();

header('location:');
?>
```

Рисунок 3.25- створення підключення до бази даних та шифрування даних користувача

На даному коді можна спостерігати дані, які розміщені в змінній \$password проходять через функцію md5(); Дана функція створена для того, щоб кодувати дані, які користувач надає без втрати та ускладнення розшифрування в подальшому. Після того, як дані були зашифровані, вони в такому вигляді передаються у базу даних. На рисунку 3.26 відображаються дані, які уже закодовані у різні символи.

Options					id	login	password	name
<input type="checkbox"/>					1	380680572774	pppppppppp	Taras
<input type="checkbox"/>					2	380680572774	pppppppppp	Taras
<input type="checkbox"/>					3	Taras	taras	Taras
<input type="checkbox"/>					5	Taras	4cb5911b55e4993fcfe79a2e42d70cd9	taras
<input type="checkbox"/>					6	1234	f8eb955e7cff86c373eacbfff327a695	1234
<input type="checkbox"/>					7	380680572774	f8eb955e7cff86c373eacbfff327a695	Taras_Sych
<input type="checkbox"/>					8	таня	a93b0f302634950dd5fb3014f8bce776	таня
<input type="checkbox"/>					11	user	cc54f0426ea4d81bb746b6a1fbf4d7ed	user
<input type="checkbox"/>					12		a3a4e9f21f9caf23d7b6bf35447fa128	
<input type="checkbox"/>					13			
<input type="checkbox"/>					14	тарас	9c568a2e3b2ec790584f20173fd3dc41	Тарас Сич

Рисунок 3.26 – відображення закодованих даних у таблиці users

На даному рисунку можна спостерігати, що перші стовбці таблиці не закодовані. Такі дані не проходять кодування через функцію md5 і у випадку, якщо хтось зможе отримати доступ до бази даних, відповідно, усі дані всіх користувачів будуть в небезпеці. Наступні рядки уже можна побачити, що замість звичного тексту розміщені випадковий порядок символів. За допомогою цієї функції база зберігає дані лише у тому вигляді, в якому передає код.

3.3 Експериментальні дослідження системи комунікації.

Останнім важливим пунктом у розробці програмного продукту є його тестування. Після того, як усі важливі елементи, необхідні функції та графічні інтерфейси об'єднані в одну картину, потрібно на власному досвіді пересвідчитись чи все працює належним чином. Так як це допоміжна навчальна

платформа, то потрібно на прикладі невеличкого навчання випробувати функціонал. В першу чергу, потрібно створити обліковий запис. На рисунку 3.27 відображається інформація, яку потрібно буде внести в базу даних для створення користувача.

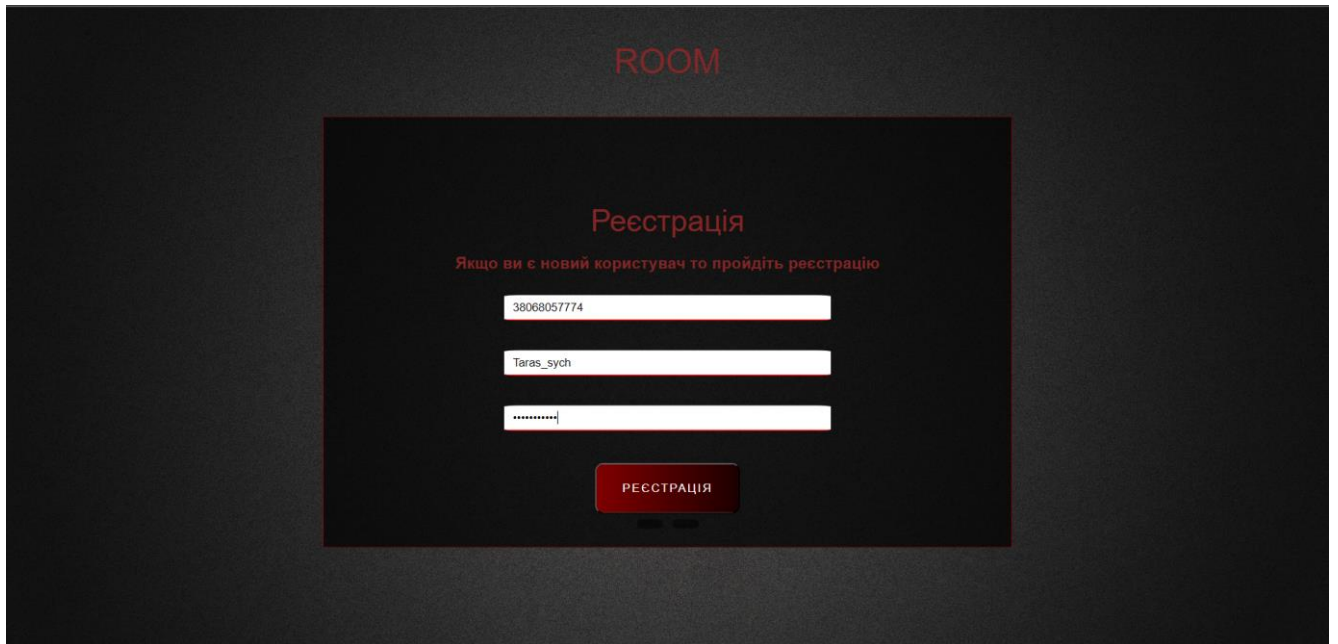


Рисунок 3.27 – створення облікового запису

Обліковий запис створений, то уже можливий для входу на особисту сторінку. Для того, щоб увійти, потрібно перейти на вкладку авторизації та ввести логін і пароль. На присунку 3.28 відображаються дані для входу в обліковий запис.

					КР.КІ. 8351649.00.00.000.ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		68

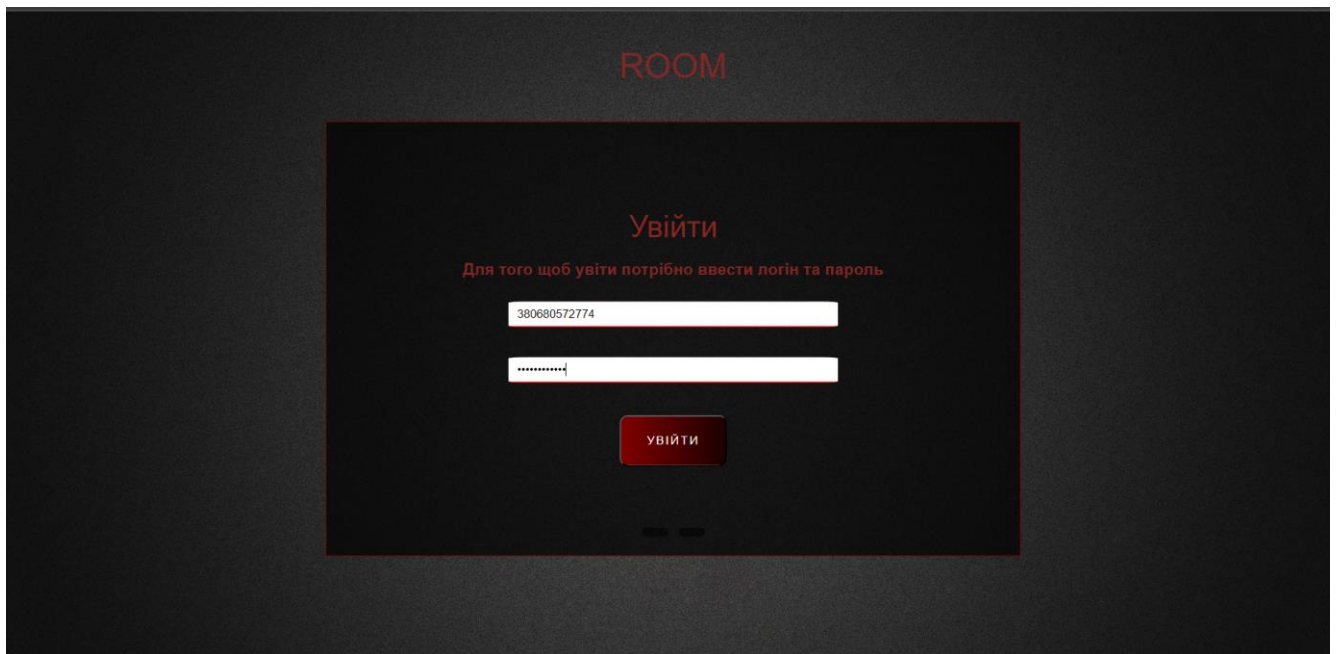


Рисунок 3.28 – дані для входу в обліковий запис

Якщо усе правильно вказано, то повинна відкритись сторінка для продовження створення сторінки та її оформлення. На рисунку 3.29 відображається сторінка із даними, які будуть відображатись на сторінці.

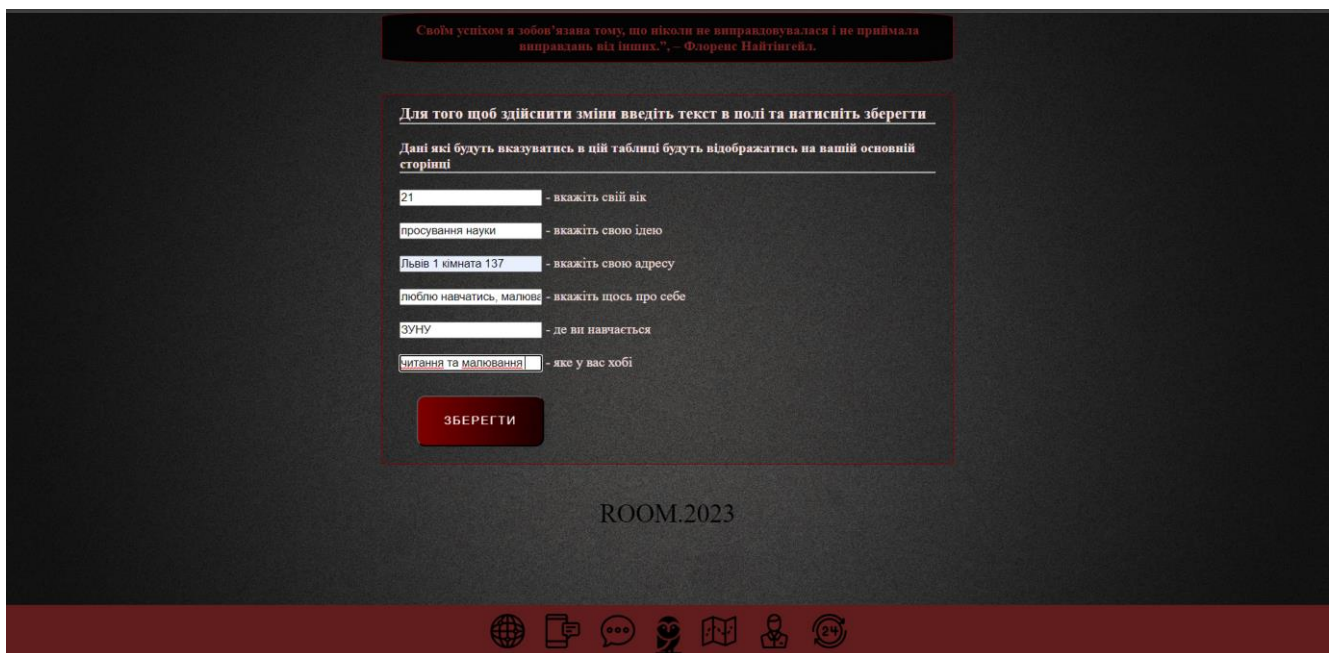
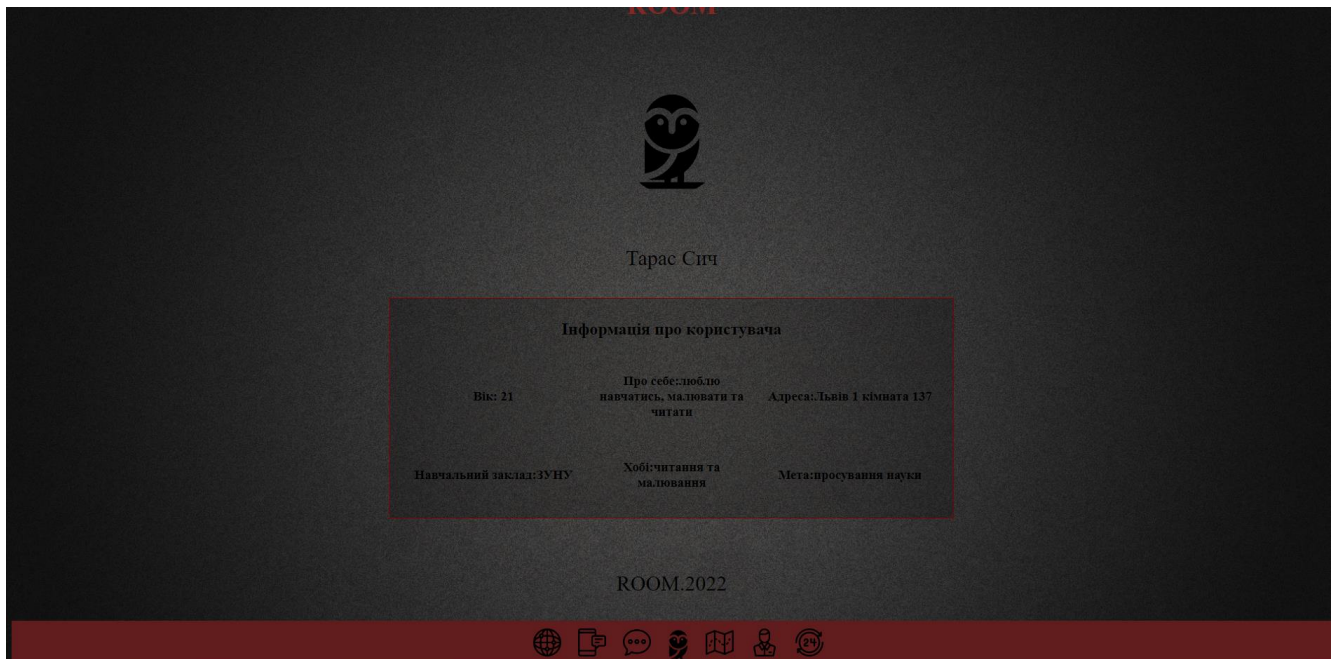


Рисунок 3.29 – налаштування профілю для головної сторінки

Налаштування профілю - це відповідальний момент, отже, від користувача залежить, наскільки він може відкрито повідомляти про себе.

					КР.КІ. 8351649.00.00.000.ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		69

Якщо усе правильно, алгоритм сайту повинен зберегти усі дані та перенести на головну сторінку користувача. На рисунку 3.30 відображається інформація, яку вказали при реєстрації профілю.



Рисунко 3.30 – відображення інформації про користувача

Профіль створений та інформація про користувача відображається коректно. Для того, щоб почати вчитись, потрібно спочатку обрати тему, за допомогою якої можна і спробувати навчатись. Відштовхуючись від даної теми, можна використати тему «Дистанційне навчання» для опрацювання. На рисунку 3.31 відображається пошукова стрічка платформи, за допомогою якої і можна швидко шукати інформацію для сайту[40].

					КР.КІ. 8351649.00.00.000.ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		70

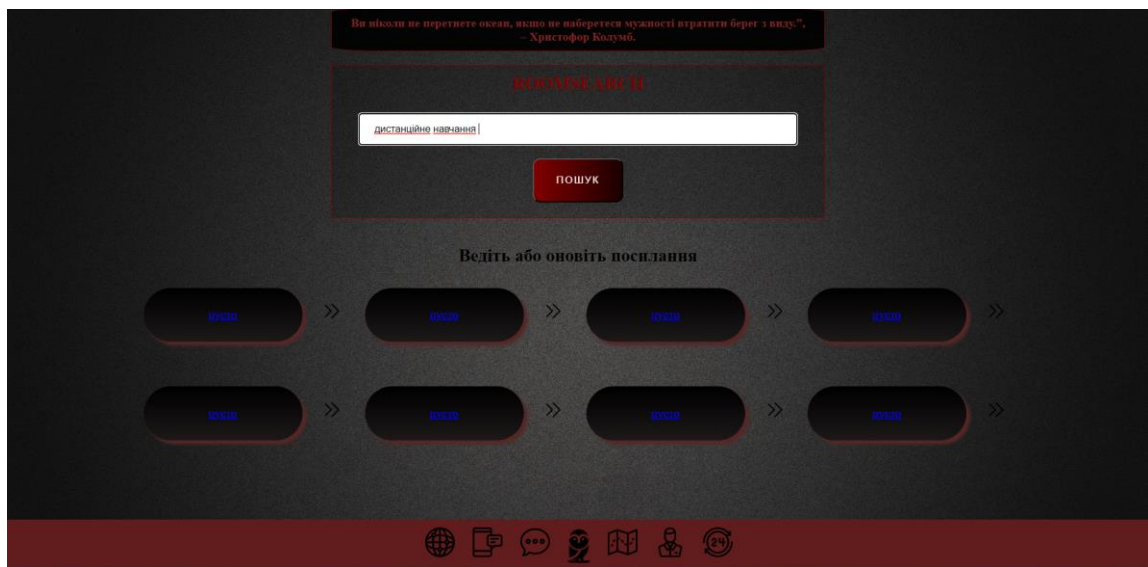


Рисунок 3.31 - пошук інформації у мережі інтернет

За допомогою такої системи сайт самостійно передає дані в гугл, який уже здійснює пошук даних. Як тільки необхідна інформація отримана, її можна швидко занотувати. На рисунку 3.32 і 3.33 відображається збереження інформації та її відображення.

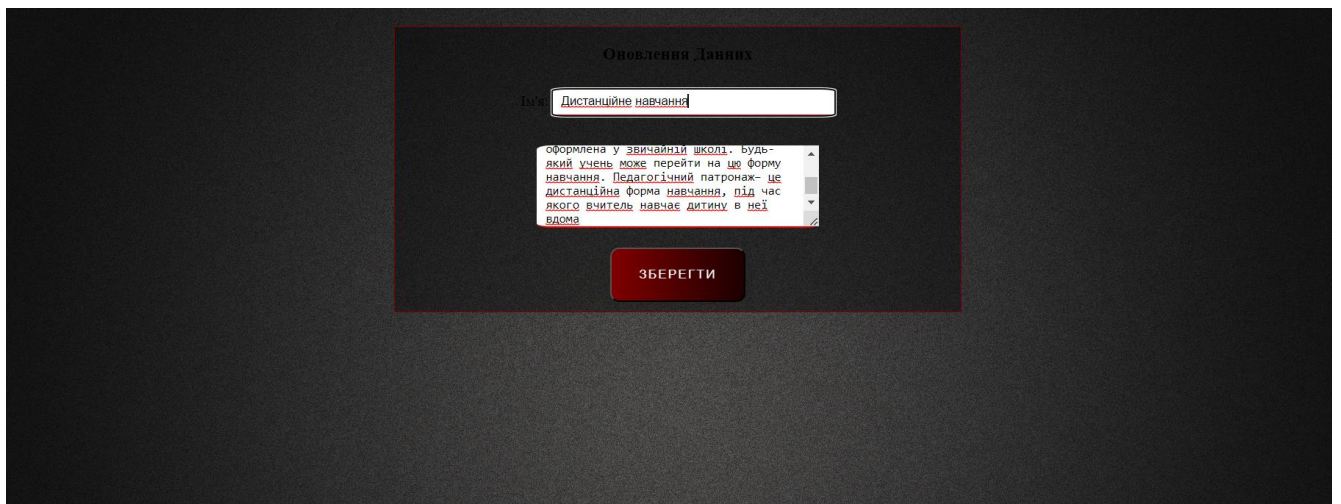


Рисунок 3.32 – збереження заміток про тему, яку необхідно дослідити

					КР.КІ. 8351649.00.00.000.ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		71

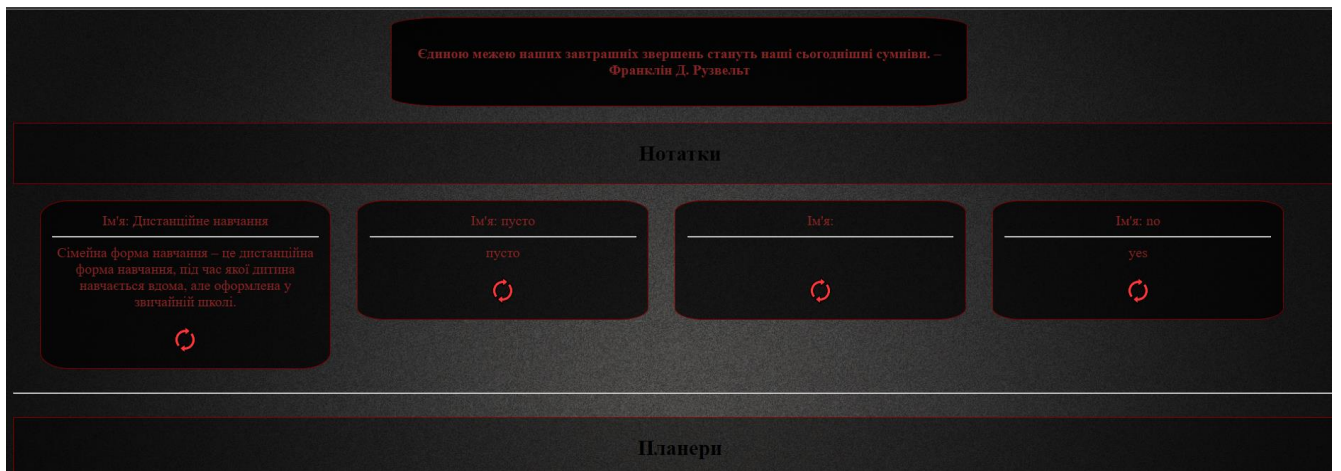


Рисунок 3.33 – відображення збережених даних у повні форми

За допомогою таких заміток, можна отримати певну інформацію та зберігати її, щоб із цього потім була можливість створити мету своєї роботи. Після того, як дані збережені, є можливість створити звіт свого дослідження, яке і допоможе іншим користувачам вивчити щось нове. На рисунку 3.34 відображається конструктор статі та наповнення її.

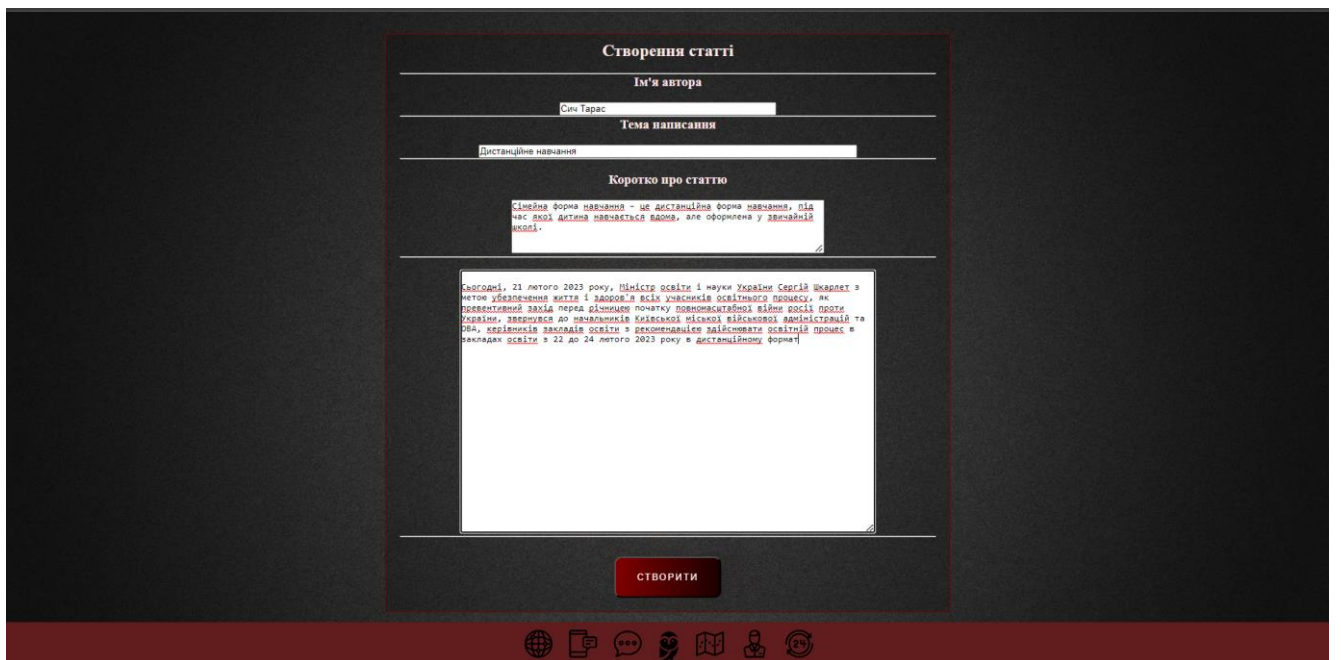


Рисунок 3.34 – створення нової статті та її наповнення новою інформацією

За допомогою такого конструктора можна своїми руками створити власні роботи, які можуть допомогти іншим користувачам у їхніх дослідженнях.

					КР.КІ. 8351649.00.00.000.ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		72

Також кожен свою роботу варто підписувати, що і вимагає поле, яке є обов'язковим для заповнення. Коли користувач уже завершує свою статтю, то необхідно натиснути на створити, щоб дані збереглись в базу та уже відобразились у переліку інших статей. Якщо перейти у перелік статей, то на рисунку 3.35 можна спостерігати статтю, яка уже є в переліку усіх статей.

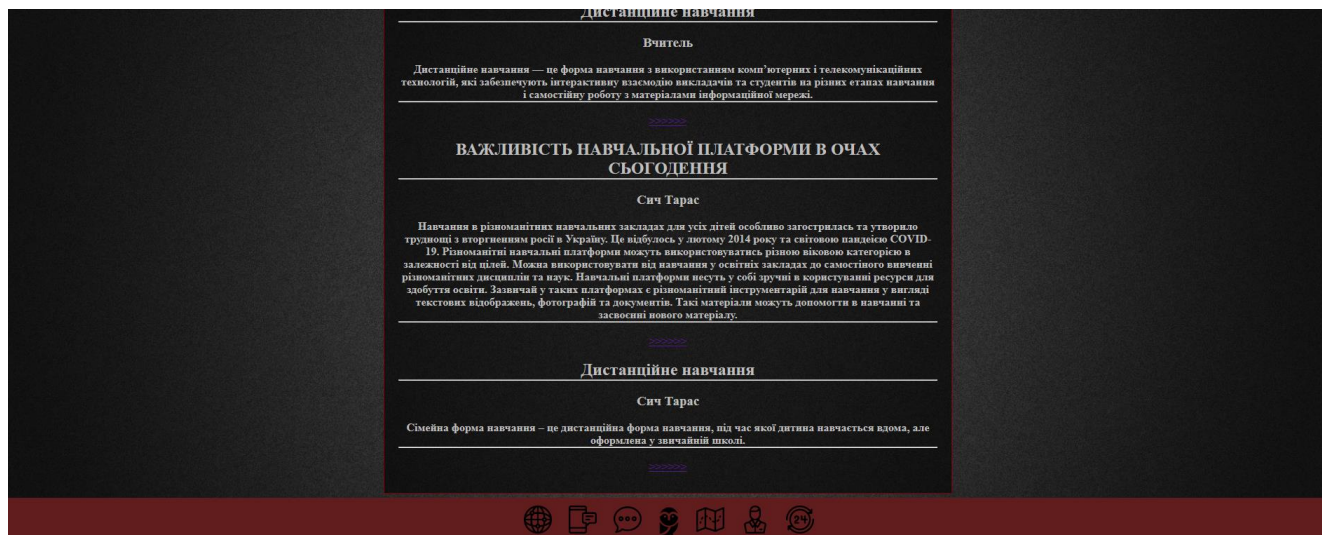


Рисунок 3.35 – відображення статей та їх користувачів

Стаття складається із декількох частин. Перша для ознайомлення та зацікавлення користувачів. Для того, щоб публікації різноманітного виду роботи читались, вони повинні зацікавити користувачів. Друга частина уже більш розгорнута. Щоб побачити всю інформацію та контент статті, достатньо натиснути на посилання під статтею, і система автоматично перенесе на уже весь контент.

					КР.КІ. 8351649.00.00.000.ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		73

Дистанційне навчання

Автор: Сич Тарас

Сьогодні, 21 лютого 2023 року, Міністр освіти і науки України Сергій Шкарлет з метою забезпечення життя і здоров'я всіх учасників освітнього процесу, як превентивний захід перед річницею початку повномасштабної війни росії проти України, звернувся до начальників Київської міської військової адміністрації та ОВА, керівників закладів освіти з рекомендацією здійснювати освітній процес в закладах освіти з 22 до 24 лютого 2023 року в дистанційному форматі



Рисунок 3.36 – відображення всього наповнення статі

Інформація написана доступно та приємно візуально для очей користувача. За допомогою такої допоміжної платформи навчання здійснюється набагато швидше, оскільки студент весь інструментарій має уже під рукою. Лише декілька кліків і вже можна отримати необхідну інформацію та зберегти її в зручному місці.

Також дослідження у цій області не завершуються і завжди буде щось, що можна дослідити та покращити. Перспективним є розвиток таких платформ, зокрема покращення збереження інформації. Можна покращити систему збереження та відображення даних, планери можна ускладнити додатковим функціоналом. Також покращити систему комунікацій додатковим функціоналом.

					КР.КІ. 8351649.00.00.000.ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		74

4 ТЕХНІКО-ЕКОНОМІЧНИЙ РОЗДІЛ

4.1 Визначення витрат на оплату праці та відрахувань у соціальні фонди

Розроблене програмне забезпечення призначене для покращення навчального процесу у всіх освітніх закладах. Алгоритм роботи системи не вимагає значних обчислювальних процедур та характеризується високою швидкістю реалізації, що уможливорює одержати модельоване значення результативних показників діяльності студента із мінімальними затратами часу.

Трудомісткість розробки ПЗ розраховують за формулою:

$$t = t_o + t_u + t_a + t_n + t_{отл} + t_d, \quad (4.1)$$

$$t = 50 + 42 + 89,6 + 89,6 + 560 + 196 = 1027,2 \text{ (люд/год)}.$$

де t_o - витрати праці на підготовку й опис поставленої задачі (приймається 50);

t_u - витрати праці на дослідження алгоритму рішення задачі;

t_a - витрати праці на розробку блок-схеми алгоритму;

t_n - витрати праці на програмування по готовій блок-схемі;

$t_{отл}$ - витрати праці на налагодження програми на ПК;

t_d - витрати праці на підготовку документації.

Складові витрати праці визначаються через умовне число операторів у ПЗ, яке розробляється. Умовне число операторів (підпрограм) розраховують за формулою:

$$Q = q \cdot C \cdot (1 + p), \quad (4.2)$$

					КР.КІ. 8351649.00.00.000.ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		75

де q - передбачуване число операторів;

Таблиця 4.1 –розрахунок задач від змін коефіцієнту.

Тип задачі	Границі змін коефіцієнту
Задачі обліку	від 1400 до 1500
Задачі оперативного управління	від 1500 до 1700
Задачі планування	від 3000 до 3500
Багатоваріантні задачі	від 4500 до 5000
Комплексні задачі	від 5000 до 5500

C - коефіцієнт складності програми;

Таблиця 4.2- розрахунок коефіцієнта складності програми

Мова програмування	Група складності	Ступінь новизни			
		А	Б	В	Г
Високогорівня	1	1,38	1,26	1,15	0,69
	2	1,3	1,19	1,08	0,65
	3	1,2	1,1	1,0	0,6
Низькогорівня	1	1,58	1,45	1,32	0,79
	2	1,49	1,37	1,24	0,74
	3	1,38	1,26	1,15	0,69

p - коефіцієнт кореляції програми в ході її розробки.

$$Q = 1400 \times 1 \times (1 + 1) = 2800.$$

Витрати праці на вивчення опису задачі t визначається з урахуванням уточнення опису і кваліфікації програміста:

$$t_u = \frac{Q \cdot B}{(75..85) \cdot k}, \quad (4.3)$$

де В - коефіцієнт збільшення витрат праці внаслідок недостатнього опису задачів (в залежності від складності задачі і числа змін обирається в інтервалі від 1,2 до 1,5);

k - коефіцієнт кваліфікації програміста, обумовлений від стажу роботи з даної спеціальності.

Таблиця 4.3 - розрахунок коефіцієнта кваліфікації програміста

Стаж програміста	Значення коефіцієнту К
до 2-х років	0,8
від 2 до 3 років	1,0
від 3 до 5 років	1,1 – 1,2
від 5 до 10 років	1,2 – 1,3
більше 10 років	1,3 – 1,5

$$t_u = 2800 \times 1,2 / 80 \times 1 = 42.$$

Витрати праці на розробку алгоритму рішення задачі розраховують за формулою:

$$t_a = \frac{Q}{(20...25) \cdot k} , \quad (4.4)$$

$$t_a = 2800 / 25 \times 0,8 = 89,6.$$

Витрати на складання програми по готовій блок-схемі розраховують за формулою:

$$t_n = \frac{Q}{(20...25) \cdot k} , \quad (4.5)$$

$$t_n = 2800 / 25 \times 0,8 = 89,6.$$

Витрати праці на налагодження програми на ПК розраховують: - за умови автономного налагодження одного завдання:

$$t_{oml} = \frac{Q}{(4..5) \cdot k}, \quad (4.6)$$

$$t_{oml} = 2800 / 4 \times 0,8 = 560.$$

Витрати праці на підготовку документації розраховують за формулою:

$$t_{\partial} = t_{\partial p} + t_{\partial o}, \quad (4.7)$$

$$t_{\partial} = 112 + 84 = 196.$$

де $t_{\partial p}$ - трудомісткість підготовки матеріалів і рукопису, що розраховують за формулою:

$$t_{\partial p} = \frac{Q}{(15..20) \cdot k}; \quad (4.7)$$

$$t_{\partial p} = 2800 / 20 \times 0,8 = 112.$$

$t_{\partial o}$ - трудомісткість редагування, друкування й оформлення документації:

$$t_{\partial o} = 0,75 \cdot t_{\partial p}, \quad (4.8)$$

$$t_{\partial o} = 0,75 \times 112 = 84.$$

Розрахунок витрат на оплату праці розробників Розрахуємо середньогодинну оплату програміста. Для цього необхідно спочатку визначити

					КР.КІ. 8351649.00.00.000.ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		78

його річний фонд грошового забезпечення. Це можна зробити, знаючи місячне грошове забезпечення програміста. Воно складає приблизно 15000,00 гривень. Таким чином, річний фонд грошового забезпечення 180000 гривень. Кількість робочих годин у році розраховуємо за формулою:

$$N_p = (N - N_n - N_e) * 8, \quad (4.9)$$

де N – загальна кількість днів у році (365),

N_n – кількість святкових днів у році (11),

N_e – кількість вихідних днів у році (107).

$$N_p = (365 - 11 - 107) \times 8 = 1976.$$

Середньогодинна оплата праці програміста визначається за формулою:

$$C_n = \frac{\Phi_p}{N_p}, \quad (4.10)$$

де Φ_p – річний фонд грошового забезпечення.

$$C_n = 180000 / 1976 = 91,1.$$

Витрати на оплату праці розробників програми складають:

$$B_{on} = C_n * T_z, \quad (4.11)$$

де T_z - загальна кількість годин роботи програміста над проектом.

$$B_{on} = 91,1 \times 1027,2 = 93\,577,92.$$

Витрати, пов'язані з розробкою програми на ПК розраховуються:

					КР.КІ. 8351649.00.00.000.ПЗ	Арк.
						79
Змн.	Арк.	№ докум.	Підпис	Дата		

$$B_{ПК} = T_{ПК} * t_c, \quad (4.12)$$

$$B_{ПК} = 1027,2 \times 0,09 = 92,448.$$

де $T_{ПК}$ - час використання ПК для розробки програми,

$C_{ПК}$ - собівартість машинного часу обчислювальної техніки (розраховує бухгалтерія підприємства).

Собівартість однієї години роботи ПК дорівнює:

$$C_{ПК} = \frac{B_e}{\Phi_{ПК}}, \quad (4.13)$$

$$C_{ПК} = 1450 / 1976 = 0,73.$$

де B_e - річні поточні витрати на експлуатацію ПК,

$\Phi_{ПК}$ річний фонд часу корисної роботи ПК.

Розрахуємо річний фонд часу роботи ПК. Визначивши дійсний річний фонд часу ПК у годинах, отримаємо можливість оцінити собівартість годин машинного часу.

Дійсний річний фонд часу ПК дорівнює:

$$\Phi_{\partial} = N_p - (\Phi_m + \Phi_{річ}), \quad (4.14)$$

$$\Phi_{\partial} = 1976 - (5 + 144) = 1797 \text{ (год на рік)}$$

де Φ_m - місячний фонд часу на профілактику і ремонт ПК (час профілактики щомісячно – 5 годин),

$\Phi_{річ}$ - річний фонд часу на профілактику і ремонт ПК (час профілактики щорічно – 6 діб (144год)).

					КР.КІ. 8351649.00.00.000.ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		80

Річні поточні витрати на експлуатацію програмного забезпечення визначаються за формулою:

$$B_{ПК_p} = B_{E_p} + B_{A_p} + B_{РЕМ_p} + B_{ДК_p} + B_{I_p}, \quad (4.14)$$

де B_{A_p} – річні відрахування на амортизацію,

B_{E_p} – річні витрати на електроенергію для ПК,

$B_{РЕМ_p}$ – річні витрати на ремонт ПК,

$B_{ДК_p}$ – річні витрати на додаткові комплектуючі ПК,

B_{I_p} – інші витрати.

$$B_{ПК_p} = 172,6 + 2610 + 1075,2 + 358,4 + 896 = 5112,2.$$

Суму річних амортизаційних відрахувань визначаємо за такою формулою:

$$B_{A_p} = Ц_{ПК} * H_A, \quad (4.15)$$

$$B_{A_p} = 17400 \times 0,15 = 2610.$$

де $Ц_{ПК}$ – балансова вартість ПК,

H_A – норма амортизаційних відрахувань (дорівнює 15% у квартал)

Балансову вартість ПК розраховуємо за формулою:

$$Ц_{ПК} = Ц_p * (1 + K_{УН}), \quad (4.16)$$

$$Ц_{ПК} = 16000 \times (1 + 0,12) = 17920.$$

де $Ц_p$ – ринкова вартість ПК,

					КР.КІ. 8351649.00.00.000.ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		81

K_{UH} – коефіцієнт, що враховує витрати на установку й налагодження ПК(приймається рівним 12%).

Витрати на електроенергію, що споживає ПК, визначаємо за формулою:

$$B_{Ep} = P_{ПК} * \Phi_{ПК} * C_E * K_{IB}, \quad (4.17)$$

$$B_{Ep} = 0,065 \text{кВт/год} \times 1976 \times 1,68 \times 0,8 = 172,6.$$

де $P_{ПК}$ – паспортна потужність ПК,

$\Phi_{ПК}$ – річний фонд корисного часу роботи ПК,

C_E – вартість 1 кВт/год електроенергії,

K_{IB} – коефіцієнт інтенсивного використання ПК (0,7 - 1).

Таким чином, розрахункове значення витрат на електроенергію, що споживає ПК, складає:

- витрати на поточний і профілактичний ремонт (приймаються рівними 6% від вартості ПК):

$$B_{РЕМр} = C_{ПК} * 0,06, \quad (4.18)$$

$$B_{РЕМр} = 17920 \times 0,06 = 1075,2.$$

- витрати на додаткові комплектуючі – витрати необхідні для забезпечення експлуатації ПК (приймаються рівними 2% від вартості ПК):

$$B_{ДКр} = C_{ПК} * 0,02, \quad (4.20)$$

$$B_{ДКр} = 17920 \times 0,02 = 358,4.$$

- інші витрати, тобто непрямі витрати пов'язані з експлуатацією ПК (приймаються рівними 5-10% від вартості ПК):

					КР.КІ. 8351649.00.00.000.ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		82

$$B_{I_p} = C_{ПК} * 0,05 \quad , \quad (4.21)$$

$$B_{I_p} = 17920 \times 0,05 = 896.$$

4.2 Розрахунок ціни проекту

У ході розробки програмного комплексу ПК використовується на таких етапах програмування:

- написання програми за готовою схемою алгоритму;
- налагодження програми на ПК;
- підготовки документації по задачі.

Таким чином, витрати машинного часу склали ($t_{маш}$):

$$t_{маш} = t_n + t_{опл}^k + t_{д} \quad , \quad (4.22)$$

$$t = 560 + 89,6 + 196 = 845,6$$

Витрати на оплату машинного часу розраховуємо за формулою:

$$B_{маш} = t_{маш} * C_{ПК} \quad , \quad (4.23)$$

$$B_{маш} = 845,6 \times 0,73 = 617,3.$$

Загальні витрати на розробку програмного комплексу складають:

$$B_{заг} = B_{оп} + B_{маш} \quad , \quad (4.24)$$

					КР.КІ. 8351649.00.00.000.ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		83

$$V_{\text{заг}} = 93\,577,92 + 617,3 = 94\,195,22.$$

Показник внутрішньої дохідності характеризує величину чистого прибутку (чистого валового доходу), що припадає на одиницю інвестиційних вкладень у кожному часовому інтервалі життєвого циклу проекту. Розрахунок цього показника виконується за такою формулою:

$$\sum_{i=0}^T \frac{Di}{(1+q)^i} - \sum_{i=0}^T \frac{Ki}{(1+q)^i} = 0, \quad (4.25)$$

$$(13\,187,38 / (1 + 0,15)) - (12\,245,9 / (1 + 0,15)) = 11360,8053.$$

де Di - дохід (прибуток) у i -му періоді;

Ki - інвестиційні вкладення в i -му періоді з урахуванням інфляційних процесів;

i - періоди виконання і впровадження проекту;

T - загальний період (тривалість) життєвого циклу проекту;

q - показник внутрішньої норми дохідності (15%).

Показник інвестиційних вкладень з урахуванням інфляційних процесів обчислюємо за формулою:

$$K_i = \varphi_i * R_i, \quad (4.26)$$

$$K_i = 1,3 \times 94\,195,22 = 122\,453,9.$$

де φ_i - коефіцієнт інфляції на поточний період;

R_i - інвестиційні платежі в i -му періоді (капітальні вкладення).

Дохід від розробки ПЗ у i -му періоді розраховуємо за формулою:

					КР.КІ. 8351649.00.00.000.ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		84

$$D_i = J_i(B_i - C_i), \quad (4.27)$$

$$D_i = 1 (107382,6 - 94 195,22) = 13 187 ,38.$$

де B_i - ціна продажу програмного продукту в i -му періоді;

C_i - собівартість програмного продукту (фактично дорівнює сумі витрат на розробку ПЗ);

J_i - кількість ПЗ.

Вартість продажу розробленого продукту розраховують за формулою:

$$B_i = B_{заг} \cdot \left(1 + \frac{p}{100} \right), \quad (4.28)$$

$$B_i = 94 195,22 \times (1 + 0,14) = 107382,6.$$

де p - середній рівень рентабельності на поточний період.

4.3 Визначення економічної ефективності розробки проекту

Показник рентабельності інвестицій. У практиці середнього бізнесу для визначення ефективності проектних рішень широко використовується показник рентабельності інвестицій. Економічний зміст – характеризує частку чистого приведенного доходу, що припадає на одиницю дисконтованих в період життєвого циклу проекту інвестиційних вкладень.

$$p = \frac{\sum_{i=0}^T \frac{D_i}{(1+q_n)^i}}{\sum_{i=0}^T \frac{K_i}{(1+q_n)^i}} - 1 > 0. \quad (4.29)$$

					КР.КІ. 8351649.00.00.000.ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		85

$$(12\,245,9 / (1 + 0,15)) / 13\,187,38 / (1 + 0,15) - 1 = 0,70216.$$

Економічна ефективність полягає у відношенні результату від розробленого програмного продукту до затрачених ресурсів:

$$E = \frac{D_i}{B_{заг}} \quad , \quad (4.30)$$

$$E = 12\,245,9 / 94\,195,22 = 0,53.$$

Тоді термін окупності можна розрахувати за такою формулою:

$$T = \frac{1}{E} \quad , \quad (4.31)$$

$$T = 1 / 0,53 = 1,8 .$$

Підсумовуючи усі результати із вищезазначених показників можна визначити що: Час роботи, який був затрачений для надання даного програмного забезпечення склав 3295 (люд.год) оскільки програмне забезпечення «Допоміжна навчальна платформа для студентів» важко реалізувати, а виконавець програми має досвід роботи до двох років, процес впровадження зайняв досить багато часу та зарплата розраховується за годину, як і загальна оплата його роботи за завершення цього завдання, яке буде виконано за 128,3 (1027,2 люд /год) трудових діб складе 93 577, 92 гривень.

Що стосується ефективності виробництва даного програмного забезпечення, то можна сказати, що термін його окупності склав 1,8 року, що свідчить про рентабельність і доцільність впровадження даного продукту.

					КР.КІ. 8351649.00.00.000.ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		86

ВИСНОВКИ

У кваліфікаційній роботі розв'язано практичну задачу розробки системи комунікації у навчальному процесі. При цьому сформовано висновки та отримано практичні результати.

1. Проведено аналіз комунікативних навчальних платформ й зазначено їх переваги в умовах дистанційної чи змішаної систем освіти, а також для самоосвіти.

2. На основі існуючих програмних засобів проаналізовано оптимальні програмні інструменти розробки системи комунікації, враховуючи простоту та зручність використання для звичайного користувача.

3. Сформульовано вимоги до навчальних платформ такого типу. Розроблено алгоритм проектування навчальної платформи із елементами комунікації.

4. Побудовано структуру програмного забезпечення навчальної системи та розроблено алгоритм її роботи.

5. На основі аналізу особливостей навчальних платформ, розроблено структуру бази даних навчальних матеріалів та реалізовано її як елемент освітньої системи комунікації.

6. Обґрунтовано техніко-економічні показники ефективності розробки програмної системи, що підтверджують доцільність розробки проєкту.

					КР.КІ. 8351649.00.00.000.ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		87

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Мала І. Дистанційне навчання як дієвий інструмент управлінської освіти. Вчені записки Університету «КРОК».2022. 2(66). С. 112–131.
2. Морган В.К., Бахметьева А.М. Навчальні платформи як складові комунікаційного середовища у школах в умовах карантину. Дніпро: ДНУ, 2020. С. 96-101.
3. Москаленко С. А., Кирилова О. В. Дистанційна медіаосвіта: досвід реалізації через мобільний додаток. Дніпро: ДНУ, 2020. С.101-107.
4. Калініна Л.М. Інформатизація освіти і зародження нового освітнього середовища як основи нової Української школи // Комп'ютер у школі та сім'ї, №4, 2017. – С. 13-17.
5. Опанасюк Ю.В. Дистанційне навчання як наслідок еволюції традиційної системи освіти // Вища освіта України, №11, 2016. – С. 49-53.
6. Методичні рекомендації до виконання кваліфікаційної роботи з освітнього ступеня “Бакалавр” спеціальності 123 «Комп'ютерна інженерія» галузі знань.
7. Гнатюк О.В., Дистанційне навчання: проблеми, пошуки, виклики. м. Київ : 2020 р
8. Концепція розвитку дистанційної освіти в Україні року: постанова Міністерства освіти і науки України від 20 грудня 2000 р. URL: <http://www.osvita.org.ua/distance/pravo/00.html>.
9. Толочко В.М. Проблемні аспекти дистанційної форми освіти та можливості її використання в Україні: URL: http://www.provisor.com.ua/archive/2009/N11/padfo_119.php.
10. Організація освітнього процесу із застосуванням технологій дистанційного навчання у 2020/2021 навчальному році: методичні рекомендації / за заг. ред. В.І. Шуляра. Миколаїв: ОПППО, 2020. 108 с.
11. Гороховський О. І. Методичні аспекти створення навчальної літератури

					КР.КІ. 8351649.00.00.000.ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		88

для дистанційного навчання / О. І. Гороховський – К., 2007. – 543 с

12. Концепція розвитку дистанційної освіти в Україні. – К. : КПІ, 2000. – 12 с.

13. Schlosser, L.A.; Simonson, M.R.; Hudgins, T.L. Distance education: definitions and glossary of terms, Third edition. Charlotte, N.C.: IAP -Information Age Pub., ©2010, p.1.

14. Сич Т.А., Заяць А.М. В. НАВЧАЛЬНІ ПЛАТФОРМИ ЯК ОСВІТНІЙ ІНСТРУМЕНТ СЬОГОДЕННЯ: матеріали наук.-практ.конф., м.Тернопіль, 23 травня 2023р.

15. Назарко І. Використання засобів дистанційної освіти для підвищення ефективності навчального процесу у ВНЗ. URL: http://elartu.tntu.edu.ua/bitstream/123456789/17337/1/konferencija_.pdf (дата звернення: 06.05.2023).

16. Сайт навчального порталу “eLearners.com”. [Електронний ресурс] – Режим доступу: <http://www.elearners.com> – Назва з екрану.

17. How to Build an Online Learning Platform: веб-сайт. URL <https://www.aimprosoft.com/blog/how-to-start-an-online-learning-platform/> (дата звернення 15.05.2023).

18. Створення сайту онлайн школи : веб-сайт. URL: <https://webcase.com.ua/uk/sozдание-onlajn-shkoly/> (дата звернення 15.04.2023).

19. 15 ознак того, що розробник має поганий код і поради щодо цього. URL: <https://pro-wp.in.ua/15-oznak-togo-shho-rozrobnik-ma%D1%94-poganiy-kod-i-poradi-shhodo-napisannya-chistogo-kodu/> (дата звернення 15.04.2023).

20. Структура сайту: основні види та правила їх розробки. URL: <https://webtune.com.ua/statti/web-rozrobka/struktura-sajtu/> (дата звернення 15.04.2023).

21. Поняття, структура та різновиди веб-сайтів. Автоматизоване розроблення веб-сайтів. URL: <http://www.ndu.edu.ua/liceum/web.pdf> (дата звернення 08.04.2023).

22. Особливості функціональних вимог та не функціональних вимог. URL:

					КР.КІ. 8351649.00.00.000.ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		89

<https://uk.myservername.com/features-functional-requirements/> (дата звернення 15.04.2023).

23. Етапи створення веб-сайтів. URL: <http://nikolay.in.ua/dstantsijne-navchannya/9-klas/845-etapi-stvorennja-veb-sajtiv> (дата звернення 10.04.2023).

24. Відновлення MySQL баз даних ручними і «механічними» способами. URL: <http://yoip.com.ua/vidnovlennja-mysql-baz-danih-ruchnimi-i-mehanichnimi-sposobami/> (дата звернення 10.04.2023).

25. Howto store articles in mysql database. URL: <https://stackoverflow.com/questions/15702043/how-to-store-articles-in-mysql-database/> (дата звернення 10.04.2023).

26. MySQL Tutorial .URL: <https://downloads.mysql.com/docs/mysql-tutorial-excerpt-5.7-en.pdf> (дата звернення 01.04.2023).

27. Introduction to MySQL. URL: <http://www.di.uniba.it/~cdamato/corsi/BasiDiDati-Materiale/Intro%20a%20MySQL.pdf> (дата звернення 01.04.2023).

28. I Creating your MySQL Database. URL: http://ooart.ru/uploads/book/creating_your_mysql_database_practical_design_tips_and_techniques_marc_delisle.pdf (дата звернення 01.04.2023).

29. Alcamn R L, Caracana C D. 2012. Introduction to Management Information Systems. Primera ediciu

30. Sangraa A, Sanmamed M G. 2010. The role of information and communication technologies in improving teaching and learning processes in primary and secondary schools. ALT-J, Research in Learning Technology Vol. 18, No. 3 pp: 207-220

31. Khattak W A, Nasir M, Sultan K. 2012. The Role of Information Technology in Media Industry . International Journal of Business and Social Science Vol. 3 No.6

32. Molina H G, Ullman J D, Widom J. 2009. DATABASE SYSTEMS The Complete Book 2nd Edition. Pearson Education Inc.

					КР.КІ. 8351649.00.00.000.ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		90

33. MySQL.URL: <https://upload.wikimedia.org/wikipedia/commons/e/e2/MySQL.pdf> (датазвернення 01.04.2023).
34. MySQL Database Management System (DBMS) On FTP Site LAPAN Bandung.URL: <https://media.neliti.com/media/publications/407145-mysql-database-management-system-dbms-on-9953a0f6.pdf> (дата звернення01.04.2023).
35. PHP Підручник.URL: <https://w3schoolsua.github.io/php/index.html#gsc.tab=0> (дата звернення 13.04.2023).
36. HTML Підручник. URL: <https://w3schoolsua.github.io/html/index.html#gsc.tab=0> (дата звернення 15.04.2023).
37. Підручник HTML-Підручники HTML і CSS. URL: https://htmlbook.at.ua/news/tutorial_html/1-0-1 (дата звернення 16.04.2023).
38. Середовища розробки. Основи HTML.URL: https://mmi.stu.cn.ua/wp-content/uploads/2016/09/Lektsiya-2_HTML.pdf (дата звернення 16.04.2023).
39. Php. URL: <https://www.php.net/> (дата звернення 18.04.2023).
40. HTML Styles – CSS. URL: https://www.w3schools.com/html/html_css.asp (дата звернення 18.04.2023).
41. CSS Підручник. URL: <https://w3schoolsua.github.io/css/index.html#gsc.tab=0> (дата звернення 18.04.2023).
42. Болье А. Learning SQL [Електронний ресурс]. 2005. URL: <http://shop.oreilly.com/product/9780596007270.do>.
43. Qiu M., Davis S. Database security mechanisms and implementation. IACIS, Issues in Inform. Syst. 2002 vol. 03 pp. 529-534.
44. MAMP.URL: <https://www.mamp.info/en/windows/>(дата звернення 18.04.2023).
45. MySql – Вікіпедія. URL: <https://uk.wikipedia.org/wiki/MySQL> (дата звернення 14.05.2023)

					КР.КІ. 8351649.00.00.000.ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		91

46. Балик Н.Р., Мандзюк В.І/ Бази даних MySQL: Навчальний посібник. — Тернопіль: Навчальна книга – Богдан, 2010

47. MySQL. Opty`mizaciya produkty`vnosti / В. Shvarcz, Р. Zajcev, V. Tkachenko, D. D. Zoodnaj, D. Dzh. Balling, A. Lencz. – М. : Vy`d. Sy`mvolPlyus, 2010. – 832 s.

48. Effective MySQL Optimizing SQL Statements / R. Bradford. – NY. : Vy`d. McGraw-Hill Education, 2011. – 184 s.

49. Relational Database Design and Implementation, Fourth Edition / J. L. Harrington. – В. : Vy`d. Morgan Kaufmann, 2016. – 712 s.

50. Bringing MySQL to the web.URL: <https://www.phpmyadmin.net/> (дата звернення 18.04.2023).

51. PostgreSQL – Вікіпедія. URL: <https://uk.wikipedia.org/wiki/PostgreSQL> (дата звернення 14.05.2023).

52. Методичні вказівки до випускних кваліфікаційних робіт освітнього рівня “Бакалавр” спеціальності “Комп’ютерна інженерія”/ О.М. Березький, Г.М. Мельник, Л.О.Дубчак, Ю.М. Батько / Під ред. О.М. Березького. Тернопіль: ЗУНУ, 2021. – 52 с.

53. Сич Т.А., Заяць А.-М. В. Навчальні платформи як освітній інструмент сьогодення. VII Науково-практична конференція молодих вчених і студентів «Інтелектуальні комп’ютерні системи та мережі». 23 травня 2023 р. Тернопіль. Україна. с. 26

					КР.КІ. 8351649.00.00.000.ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		92