

**ДЕРЖАВНИЙ УНІВЕРСИТЕТ ТЕЛЕКОМУНІКАЦІЙ**  
Навчально-науковий інститут Інформаційних технологій  
Кафедра комп'ютерної інженерії

## **Пояснювальна записка**

до бакалаврської роботи

на ступінь вищої освіти бакалавр

на тему: **«РОЗРОБКА ПРОГРАМНИХ ЗАСОБІВ ДЛЯ ТЕСТОВОЇ ОЦІНКИ  
ЗНАНЬ МОВОЮ C#»**

Виконав: студент 4 курсу, групи ПД– 43

спеціальності

121 Інженерія програмного  
забезпечення

(шифр і назва спеціальності)

Борота І.Р.

(прізвище та ініціали)

Керівник Коба А.Б.

(прізвище та ініціали)

Рецензент Шуклін Г.В.

(прізвище та ініціали)

Київ – 2023

# ДЕРЖАВНИЙ УНІВЕРСИТЕТ ТЕЛЕКОМУНІКАЦІЙ

## Навчально-науковий інститут інформаційних технологій

Кафедра Інженерії програмного забезпечення

Ступінь вищої освіти - «Бакалавр»

Спеціальність - 121 Інженерія програмного забезпечення

ЗАТВЕРДЖУЮ

Завідувач кафедри  
Інженерії програмного  
забезпечення

\_\_\_\_\_ О.В. Негоденко

«\_\_\_\_» \_\_\_\_\_ 2023 року

### ЗАВДАННЯ

#### НА БАКАЛАВРСЬКУ РОБОТУ СТУДЕНТУ

Борота Ілля Романович

(прізвище, ім'я, по батькові)

1. Тема роботи: «Розробка програмних засбів для обліку матеріально-технічного забезпечення навчального процесу мовою с#»

Керівник роботи \_\_\_\_\_ Старший викладач кафедри Коба А.Б

(прізвище, ім'я, по батькові, науковий ступінь, вчене звання)

затверджені наказом вищого навчального закладу від «24» лютого 2023 року №26.

2. Строк подання студентом роботи «1» червня 2023 року

3. Вихідні дані до роботи:

3.1.Офіційна документація С#

3.2.Науково-технічна література

4. Зміст розрахунково-пояснювальної записки (перелік питань, які потрібно розробити):

4.1.Аналіз актуальності та огляд існуючих додатків

4.2.Аналіз інструментів реалізації та розробка структури додатку

4.3.Реалізація програмної частини додатку.

4.4.Висновки

5. Перелік графічного матеріалу
  - 5.1. Титульний слайд
  - 5.2. Мета, об'єкт та предмет дослідження
  - 5.3 Задачі дипломної роботи
  - 5.4 Аналіз аналогів
  - 5.5 Вимоги до програмного забезпечення
  - 5.6 Програмні засоби для розробки
  - 5.7 Екрані форми
  - 5.8 Апробація результатів дослідження
  - 5.9 Висновки
6. Дата видачі завдання «25» лютого 2023 року

### КАЛЕНДАРНИЙ ПЛАН

№ з/п	Назва етапів бакалаврської роботи	Строк виконання етапів роботи	Примітка
1	Утвердження теми бакалаврської роботи	24.02.2023	Виконано
2	Підбір науково-технічної літератури	26.02.2023 – 15.03.2023	Виконано
3	Аналіз актуальності та дослідження існуючих додатків	15.03.2023 – 20.03.2023	Виконано
4	Аналіз та вибір інструментів для розробки програмного забезпечення	20.03.2023 – 14.04.2023	Виконано
5	Розробка програмного забезпечення для обліку матеріально-технічного забезпечення навчального процесу мовою с#	14.04.2023 – 04.05.2023	Виконано
6	Висновки та оформлення роботи	04.05.2023 – 12.05.2023	Виконано
8	Попередній захист	1.06.2023	
9	Захист роботи		

Студент \_\_\_\_\_ Борота І.Р  
( підпис ) (прізвище та ініціали)

Керівник роботи \_\_\_\_\_ Коба А.Б  
( підпис ) (прізвище та ініціали)





## РЕФЕРАТ

Актуальність теми, обумовлена потребою створенні автоматизованої інформаційної системи для навчання та тестування знань учнів та відпрацювання навчального матеріалу з інструментарієм для аналізу результатів, щоб за допомогою цієї системи підвищити якість освітньої програми та автоматизувати процес перевірки пройдених студентами тестів.

Мета роботи: розробити необхідне математичне та програмне забезпечення для відпрацювання навчального матеріалу та аналізу результатів навчання.

Для виконання поставленої задачі необхідне вирішення наступних завдань:

- проаналізувати інформаційні системи аналогічного призначення;
- розглянути інструменти розробки та вибрати найбільш підходящі для розробки даної інформаційної системи;
- сформулювати технічне завдання;
- Розробити ІС;
- Запровадити ІС;
- здійснити оцінку ефективності взаємодії користувачів із ІС.

Об'єктом дослідження дипломної роботи є навчання і засвоєння матеріалу школярами.

Предметом дослідження є онлайн інструмент для тренувань та тестування студентів.

Наукова новизна полягає в тому, що створено інформаційну систему, що має в своєму розпорядженні інструменти збору та аналізу даних для досліджень, розроблено формулу для розрахунку показника рівня засвоєння матеріалу учнів для побудови кривої навчання.

Практична значущість полягає в тому, що дана інформаційна система буде застосовуватись в університеті для відпрацювання навчального матеріалу та аналізу результатів учнів.

Економічна ефективність проекту полягає в тому, що система, що розробляється, дозволяє багаторазово розширювати користувальницьку базу, звільняючи викладача від рутини, що забирає велику кількість часу. Розрахунки з наближеною до сьогоднішньої кількості користувачів показують, що навіть у такому разі автоматизована система тестування знижує витрати на бізнес-процес перевірки завдань більш ніж удвічі.

***Ключові слова:*** база даних, аналіз результатів, тестування

## ЗМІСТ

<b>ВСТУП.....</b>	<b>9</b>
<b>1.ТЕОРЕТИЧНІ ОСНОВИ НАВЧАЛЬНОГО ПРОЦЕСУ .....</b>	<b>12</b>
1.1 Огляд існуючих аналогів .....	12
1.2 Вибір інструментальних засобів розробки .....	20
1.3 Опис середовища та системи .....	23
<b>2. ІНФОРМАЦІЙНІ ТЕХНОЛОГІЇ НАВЧАЛЬНОГО ПРОЦЕСУ .....</b>	<b>26</b>
2.1 Основні поняття.....	26
2.2 Форми навчального процесу .....	30
2.3 Основні вимоги до запуску та роботи з програмним засобом .....	34
2.3.1. Функціональні вимоги .....	34
2.3.2. Вимоги до складу та параметрів технічних засобів .....	34
2.3.3. Вимоги до вхідних та вихідних даних .....	35
2.3.4. Вимоги до інтерфейсу.....	35
2.3.5. Вимоги до тестування програмного забезпечення .....	35
2.3.6 Вимоги до програмної документації .....	36
2.5 Проектування архітектури системи.....	36
2.6 Проектування інтерфейсу користувача.....	37
<b>РОЗДІЛ 3. ЕТАПИ РОЗРОБКИ ТА СТВОРЕННЯ ПРОГРАМНОГО</b>	
<b>ДОДАТКУ .....</b>	<b>42</b>
3.1 Структура програмного додатку.....	42
3.2 Проектування та розробка БД.....	44
3.3 Опис функціоналу .....	45
3.4 Тестування.....	49
<b>ВИСНОВКИ .....</b>	<b>12</b>
<b>СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ .....</b>	<b>13</b>
<b>ДОДАТКИ.....</b>	<b>14</b>
ДОДАТОК А.....	14
ДОДАТОК Б.....	15



## ВСТУП

В сучасному світі комп'ютеризація грає значну роль у розвитку суспільства. В часи пандемії досить поширеним стало дистанційне навчання, а отже, створення різних програмних засобів, які активізують навчальну активність студентів, залишається досить актуальним. Серед різних навчальних програмних засобів таких, як, програми-тести, електронні підручники та довідники, особливу увагу займають програми-тренажери.

Створення такої програми в рамках даної роботи дає можливість відпрацювання студентами навичок розв'язування задач з дистанційного навчального курсу.

Серед тренажерів даного дистанційного курсу подібних задач не було реалізовано, тому створення такого програмного продукту, який може бути використаний для всіх форм навчання, є важливим.

Мета роботи – побудувати алгоритм та створити програмну реалізацію навчального процесу.

Завдання, які необхідно виконати для досягнення мети:

- опрацювати методичні рекомендації до виконання дипломної роботи;
- сформулювати постановку задачі;
- виконати огляд літературних джерел, що використовується для вивчення даної теми, а також огляд існуючих програмних продуктів, які використовуються при вивченні різних дисциплін;
- вивчення теоретичного матеріалу з теми;
- розробка алгоритму програмного забезпечення;
- побудова блок–схеми алгоритму програми;
- вибір оптимальної платформи для програмної реалізації;
- створення програмної реалізації.

Об'єктом розробки є програмне забезпечення навчального процесу мовою

C#/

Предметом розробки є програмне забезпечення для навчального процесу мовою С#.

Методи розробки. Для створення програми було обрано мову програмування С# та середовище Microsoft Visual Studio.

## 1.ТЕОРЕТИЧНІ ОСНОВИ НАВЧАЛЬНОГО ПРОЦЕСУ

### 1.1 Огляд існуючих аналогів

Для створення тестів з предметних областей розроблені та розробляються спеціальні інструментальні оболонки, які дозволяють створювати комп'ютерні тести шляхом формування бази даних із набору тестових завдань.

Існуючі інструментальні програми, що дозволяють розробляти комп'ютерні навчальні тести, можна розділити на дві категорії: загального призначення та спеціального призначення. Загальна програма містить навчальну оболонку як невід'ємну частину. Серед них Adonis, Linkway (Microsoft), Fairy, Raduga тощо. Спеціальна навчальна оболонка використовується лише для створення тестів.

Слід детально розглянути деякі існуючі початкові тестові тренажери.

– Мультитестер

Тестовий комплекс з гнучкими системами та роботою по мережі. Готуйте та перевіряйте результати навчання через локальну мережу за допомогою MultiTester (рис. 1.1). Дивно, що система має можливість відстежувати хід навчання в реальному часі. Крім того, в кінці тесту проводиться автоматичне оцінювання за встановленими критеріями.

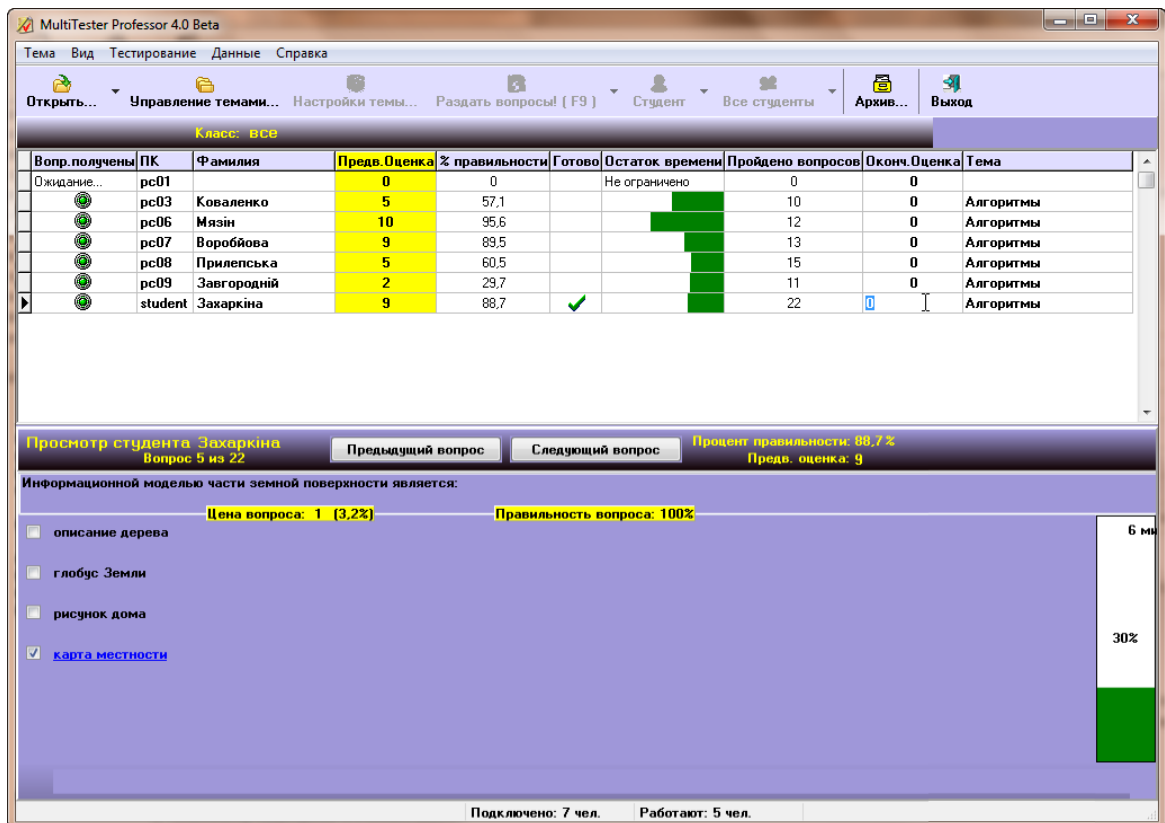


Рисунок 1.1 – зображення роботи MultiTester

До складу системи MultiTester входять:

- Програма для вчителів (професор)
- Редактор питань (QuEditor)
- Студентська програма (Студенти)

Зауважте, що «Політест» працює за моделлю клієнт-сервер, де MultiTester Student є клієнтом, а сервером є MultiTester Professor. На персональному комп'ютері вчителя є база даних, де зберігаються всі дані. При необхідності викладач надсилає матеріал клієнту (MultiTester Student) через мережу. Зв'язок між клієнтом і сервером налаштовується автоматично, але його також можна налаштувати вручну.

Така структура зберігання інформації дозволяє правильно налаштувати систему захисту даних, тобто виключити несанкціоновані дії з боку просунутих студентів, такі як дублювання відповідей, виправлення результатів тестування, виправлення питань.

Крім того, деякі учні можуть зловживати правилами самооцінювання. Щоб виключити протиправну діяльність, система MultiTester має такі засоби захисту, як одноразове відкриття та неможливість закриття програми. Тільки вчителі можуть закривати та перезапускати тест.

Навчання може проводитися процесом (тобто наступний студент сідає на місце студента, який виконав тест, отримує запитання та починає працювати незалежно від інших).

Мова інтерфейсу: російська.

Підтримувані операційні системи: Windows XP/7.

– Система UniTest

Додаток являє собою єдину систему для розробки комп'ютеризованих навчальних тестів, проведення контрольних тестів через мережу або локально, написання звітів та аналізу результатів через остаточну сертифікацію (рис. 1.2).

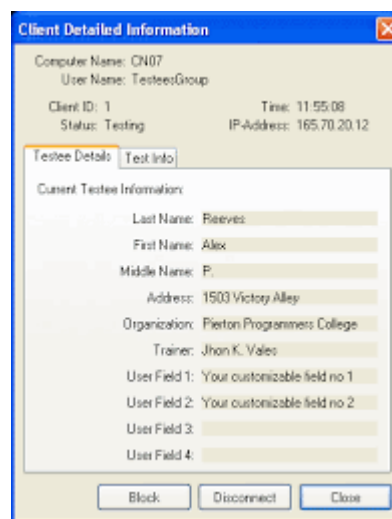


Рисунок 1.2 – зображення роботи UniTest

Система підтримує основні типи навчальних питань, а також додаткові типи питань, такі як питання з різним рівнем оцінювання, порядком сортування, побудови листування, питання, що потребують розгорнутої відповіді, тощо.

У додатку при розробці тестів можна використовувати графічну інформацію, таблиці, аудіо-, відеоматеріали, редагування тексту тощо.

UniTest System має широкі можливості для налаштування розділів навчання, створення самогенерованих запитань для тесту, вибору режимів навчання, керування загальним доступом та іншими налаштуваннями системи конфіденційності, запобігання дублюванню та передачі результатів тесту, а також гнучкого призначення балів між завданнями.

Простий і зручний інтерфейс дозволяє швидко і точно скласти підсумковий іспит з усіма запропонованими варіантами. Крім того, UniTest System може інтегруватися з пакетом програм Microsoft Office.

Програмний комплекс розроблений і постійно розвивається в тісній співпраці з викладачами вузів, співробітниками відділів кадрів і відділів підготовки кадрів підприємств різних країн Росії і Європи.

Деякі особливості цього пакета:

Тестові запитання можуть містити будь-який об'єкт OLE (документи, графіку...), включаючи інтерактивні об'єкти (анімацію, відео- та аудіокліпи...).

– У кожне окреме питання можна вводити різноманітні ефекти (наприклад, відео- та аудіоматеріали, анімацію). Система також дозволяє включати в складання презентацій та інші процеси для створення середовища інформації для максимальної інтерактивності.

– Вбудований текстовий редактор, спрощена версія програми Microsoft Word, дозволяє створювати візуальний текст, який відповідає вашим вимогам до інтерфейсу.

– Вважайте, що відповіді на певні питання можна вважати частково правильними, наприклад, використовуючи відсотки.

– Підтримуються всі основні та багато додаткових типів запитань: множинний вибір, запитання з різною вагою відповіді, запитання з вибором набору елементів, порядок сортування (наприклад, «виберіть процесори в порядку зменшення потужності» або «виберіть правильні твердження»), Установіть відповідний зв'язок («відповідність між королем і часом його правління»), чи є в тексті відповіді ключове слово, введіть його безпосередньо;

– Можливість «нарощувати» базу запитань і призначати розділи в навчанні. (Наприклад, у вас є 3 логічних розділи в навчанні: легкі, середні та складні завдання. База становить 200 запитань для легких, -150 для середніх і -170 для важких, і 15 питань вибираються випадковим чином з бази для кожного тесту. легкі питання, середні 10 запитань і важкі запитання - 5 питань);

– Вбудована система стиснення даних використовує 448-бітну технологію BlowFish, світовий стандарт захисту інформації. При використанні цієї технології всі дані багаторазово стискаються, додатково захищаючи їх від несанкціонованого читання.

– Контролювати кількість і складати список активних користувачів навчальної оболонки в реальному часі. Це дозволяє записувати поведінку учасника навчання, наприклад, щоб побачити поточний бал.

– Режим обмеженого доступу та режим блокування.

Навчальна оболонка RichTest. <http://soft.oszone.net/program/11544/RichTest/>

Тестові корпуси для навчальних закладів і корпоративне тестування (рис. 1.3). Складається з 3 модулів: Адміністратор, Клієнт, Звіт.

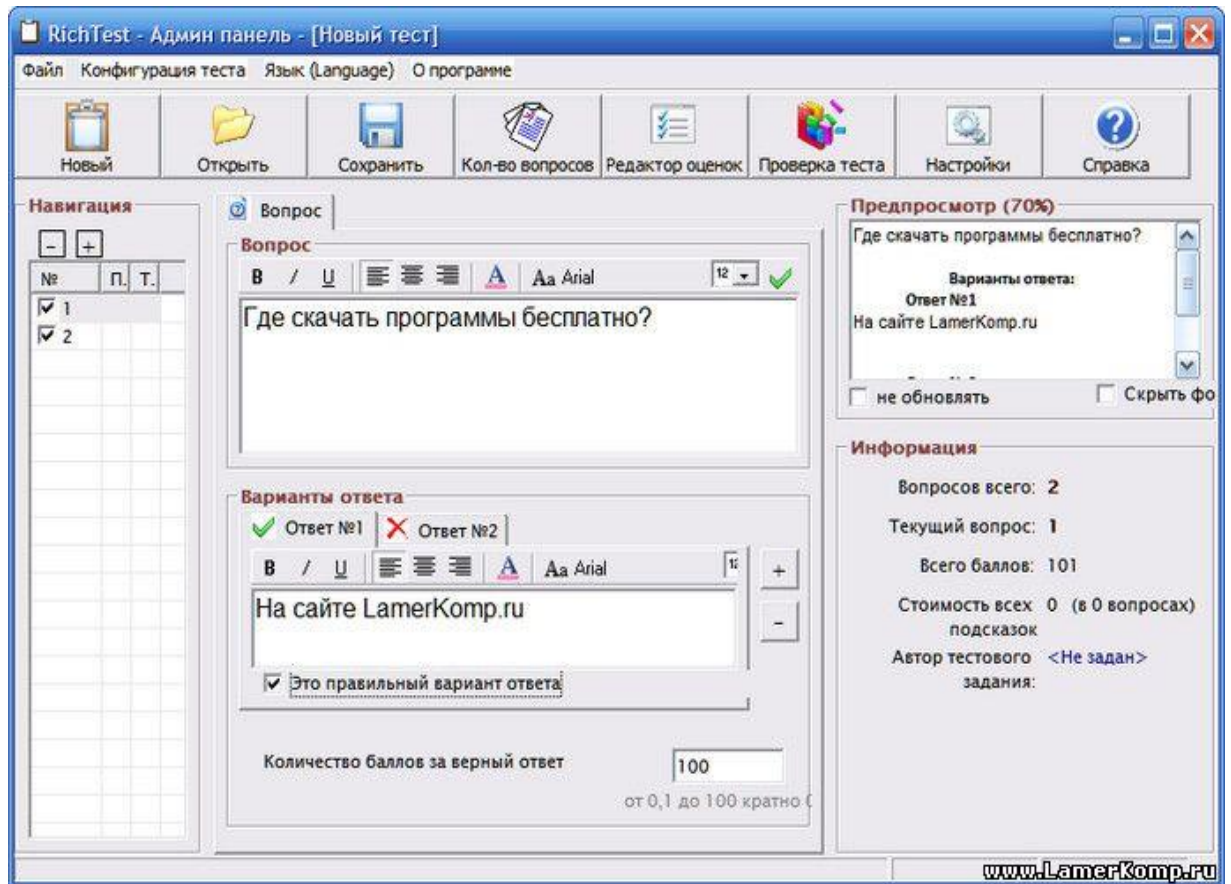


Рисунок 1.3 – зображення роботи RichTest

У RichTest є різноманітні вбудовані шаблони, які можна використовувати для зміни зовнішнього вигляду ваших тестів. Також можливе детальне налаштування системи оцінювання.

Система дозволяє вбудовувати в розробку тестів графічні зображення, таблиці, форматовані тести. Він має кілька різних варіантів запитань:

- Радіо
- Множинний вибір
- Введення текстової відповіді
- Картографування
- Упорядкувати відповіді

Операційна система: Windows 2000/XP/Vista/Server 2003/Server 2008/7  
 Мова інтерфейсу: російська.

Тип ліцензії: Безкоштовне програмне забезпечення



– «ІНДІГО» - програма для створення тестів, розташована за адресою <http://indigotech.ru/>

Навчальний тренажер INDIGO - це гнучкий інструмент для розробки комп'ютерних тестів і обробки результатів для наступних завдань (рис. 1.4):

- Підсумковий контроль для студентів коледжу.
- Претендентів на роботу відбирають потенційні роботодавці.
- Визначення кваліфікаційного рівня співробітників (переміщення, атестація, атестація, кадрові зміни).
- Пройдіть психологічні тести (наприклад, IQ тести).
- Використання в маркетингових опитуваннях, соціологічних дослідженнях
- Шкільні олімпіади, студентські першості тощо. Обґрунтованість програм для створення комп'ютерних тестів INDIGO

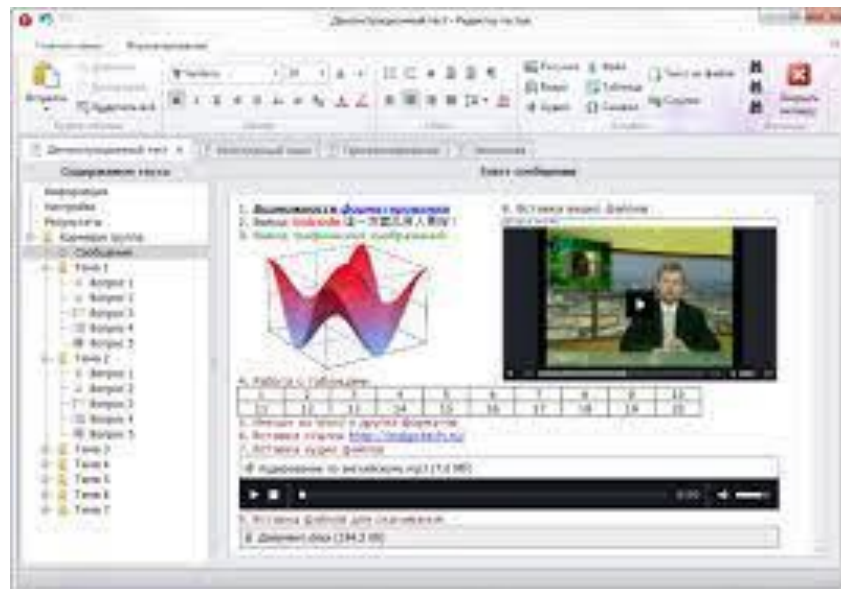


Рисунок 1.4 – зображення роботи INDIGO

Забезпечення впровадження та розвитку підвищеного рівня контролю в навчальних закладах за рахунок таких характеристик:

- Ефективно інтегрувати систему в робоче середовище навчальних закладів;
- Доступний та інтуїтивно зрозумілий інтерфейс, що дозволяє швидко та досконало освоїти програму;
- Наявність широкого спектру налаштувань обробки результатів, що значно скорочує час остаточної перевірки підсумкових атестаційних тестів.

INDIGO має такі переваги:

- Простота і послідовність установки програми;
- Доступний інтерфейс управління;
- Веб-інтерфейс користувача;
- Диференціація отримання інформації;
- Автоматична генерація варіантів тестів
- Кілька варіантів конструкторів тестів;
- Варіативність оцінки результатів тестування
- Доступність зберігання інформації;
- Правила структури тесту
- Роздруківка результатів тесту
- Експорт протоколу тесту.
- Ієрархічне групування питань.

Системні вимоги програмного забезпечення

Модулі керування сервером і клієнтом (настільні програми) потребують Windows XP/2003/Vista/2008/7/8.

Користувачам потрібне мережеве підключення до сервера та один із таких браузерів:

Google Chrome 6 і вище, Mozilla Firefox 3.6 і вище, Opera 10.6 і вище, Internet Explorer 9 і вище. Рекомендований браузер Google Chrome (завантажте офлайн-версію інсталятора).

Встановлений і запущений сервер INDIGO 2.0 займає до 100 МБ оперативної пам'яті (з Admin Client до 180 МБ оперативної пам'яті). Споживання пам'яті

збільшується в міру роботи користувача. Для 200 одночасних користувачів рекомендується комп'ютер із принаймні 2 ГБ оперативної пам'яті.

## 1.2 Вибір інструментальних засобів розробки

Для розробки програми була обрана мова програмування C#, яка є однією з найпопулярніших сучасних мов програмування. C# (вимовляється як "сі шарп") - це об'єктноорієнтована і типобезпечна мова програмування, розроблена компанією Microsoft. Він відповідає сучасним стандартам програмування та розроблявся спеціально для підтримки розвитку технології .NET Framework. Мова програмування була розроблена наприкінці 1990-х років групою інженерів на чолі зі Скоттом Вільтамотом та Андерсом Хейлсбергом у компанії Microsoft як основна мова розробки додатків для платформи Microsoft .NET (програмної платформи від компанії Microsoft, призначеної для створення звичайних програм та веб-додатків) [16], 17, 18].

Символ # (октоторп) у назві мови має кілька варіацій розшифровки: цей символ друкується на клавіатурі як Shift+3, що також може символізувати третю реалізацію C. Крім того, # можна інтерпретувати як дві пари плюсів ++ - що може означати новий крок у розвитку мови проти C++.

Інструментарій C# дозволяє вирішувати широке коло завдань, тому мова використовується для розробки:

- Веб-додатків;
- різних ігрових програм;
- мобільних додатків для Android та IOS;
- Програм для ОС Windows.

Мова C# має багато корисних функцій, таких як:

- поліморфізм;
- інкапсуляція;
- успадкування;

- події;
- атрибути;
- властивості;
- ітератори;
- навантаження операторів;
- узагальнені типи та методи;
- статична типізація;
- винятки;
- коментарі у форматі XML;
- складання сміття.

C# будучи об'єктно-орієнтованою мовою програмування підтримує всі три його основні складові: інкапсуляцію, поліморфізм та успадкування.

Інкапсуляція – механізм, який зв'язує разом код та дані, якими він маніпулює. Даний механізм дозволяє захищати цей код та дані від неправильного використання та зовнішнього доступу. Тобто це контроль доступу до полів і методів об'єкта за допомогою модифікаторів доступу: `private`, `public` і `protected`.

Поліморфізм - властивість системи, що дозволяє кільком об'єктам із загальними характеристиками мати доступ до одного інтерфейсу. Узагальнено концепцію поліморфізму слід висловити так: «один інтерфейс – безліч методів» [18].

Спадкування – така властивість, за допомогою якого один об'єкт може набувати властивості іншого. Тобто, з'являється можливість наслідувати одними класами якості та поведінка інших класів.

Мова C# пов'язана з такими мовами програмування, як C, C++ та Java.

У C++ було визначено об'єктну модель, де тепер будується мову C#. Зв'язок з мовою C ґрунтується на успадкованому від нього синтаксисі, операторах та ключових словах. Між C# і Java прямого зв'язку немає, але обидві мови були розроблені, щоб створити спеціальний код, що переноситься.

Цей код базується на C та C++, використовуючи їх синтаксис та об'єктну модель. Володіння навичками тієї чи іншої мови полегшить перехід будь-якої з них.

Найважливіша риса мови програмування C# - здатність працювати у багатомовному оточенні. Під цим розуміється здатність коду, написаного різними мовами, працювати спільно, що є важливою здатністю при створенні великих програм.

У мови досить багато переваг:

- Підтримка більшості продуктів Microsoft;
- Використання об'єктно-орієнтованого підходу;
- наявність великої кількості бібліотек та шаблонів;
- типи даних мають фіксований розмір: 32-бітний int та 64-бітний long, що підвищує «мобільність» мови та спрощує програмування;
- схожість синтаксису мови з іншими мовами програмування, що полегшує перехід для програміста (наприклад, Java або C++);
- величезна кількість спеціальних готових конструкцій («синтаксичний цукор»), розроблених для розуміння та написання коду, які не мають значення при компіляції;
- автоматичне «складання сміття». Це полегшує роботу зі звільненням пам'яті, оскільки загальномовне середовище CLR сама викличе збирач сміття та очистить пам'ять;
- можливість створювати кросплатформні програми. За допомогою Xamarin на C# можна писати програми та програми для таких операційних систем, як iOS, Android, MacOS та Linux;
- Додано функціональне програмування (F#).

Однак є й недоліки:

- орієнтованість, переважно, лише з .NET (на Windows платформу);
- мова безкоштовна лише для невеликих компаній або груп учнів.

### 1.3 Опис середовища та системи

Оскільки C# одна із найпоширеніших мов програмування, йому існує велика кількість інтегрованих середовищ розробки з різним функціоналом. Оскільки висока продуктивність є однією з основних вимог до розробки програми.

Важливим критерієм вибору середовища розробки програми є наявність профільника. Як об'єкти дослідження були обрані інтегровані середовища розробки MS Visual Studio, JetBrains Clion, Qt Creator, NetBeans та Eclipse SDK , як найбільш функціональні та поширені.

#### – MS Visual Studio

Microsoft Visual Studio - це інтегроване середовище, призначене для розробки програмного забезпечення, що має низку додаткових інструментів (1.5).

Продукти MS Visual Studio дозволяють розробляти консольні програми, програми з графічним інтерфейсом, веб-сайти, веб-програми та веб-сервіси для платформ.



Рисунок 1.5 –Microsoft Visual Studio

MS Visual Studio містить вбудований інструмент для редагування вихідного коду програми та дозволяє проводити рефакторинг коду. Відладчик, вбудований у MS Visual Studio, має два рівні роботи: рівень налагодження вихідного коду та рівень налагодження машинного коду. Додаткові вбудовані інструменти включають форми GUI та зручний редактор для створення програм з інтерфейсами, містять веб-редактор та редактор класів. Це середовище розробки підтримує можливість створення та підключення плагінів (надбудов), що використовуються для розширення функціональності. MS Visual Studio підтримує системи контролю версій вихідного коду (наприклад, Visual SourceSafe або Subversion), містить безліч інструментів, що дозволяють редагувати та проектувати код предметно-орієнтованими мовами програмування.

Для роботи з серверною частиною було обрано локальний сервер MAMP.

MAMP означає Macintosh, Apache, MySQL і PHP (рисунок 1.6).



Рисунок 1.6 - MAMP

Після встановлення MAMP все, що вам потрібно, є: Apache, MySQL і PHP.

Існує дві версії програми: платна MAMP PRO і безкоштовна MAMP, які відрізняються функціональністю та доступом до контенту та керування сервером.

Стандартний дистрибутив включає дві версії. Якщо ви використовуєте безкоштовну версію, досить скопіювати її в папку з програмою і почати працювати.

Якщо ви купуєте ліцензію на платний продукт, щоб використовувати MAMP PRO, вам потрібно буде інстальовати обидві версії, оскільки MAMP і MAMP PRO складаються разом. В основному списку включених програм доступні всі ярлики для обох версій, але не рекомендується запускати їх одночасно.

Для роботи з таблицями даних використовується phpMyAdmin.

phpMyAdmin – це безкоштовний програмний інструмент, призначений для адміністрування СУБД MySQL через браузер (рисунок 1.7). phpMyAdmin підтримує широкий спектр операцій на MySQL. phpMyAdmin дозволяє управляти базами даних, таблицями, відносинами, індексами, користувачами, дозволами тощо.

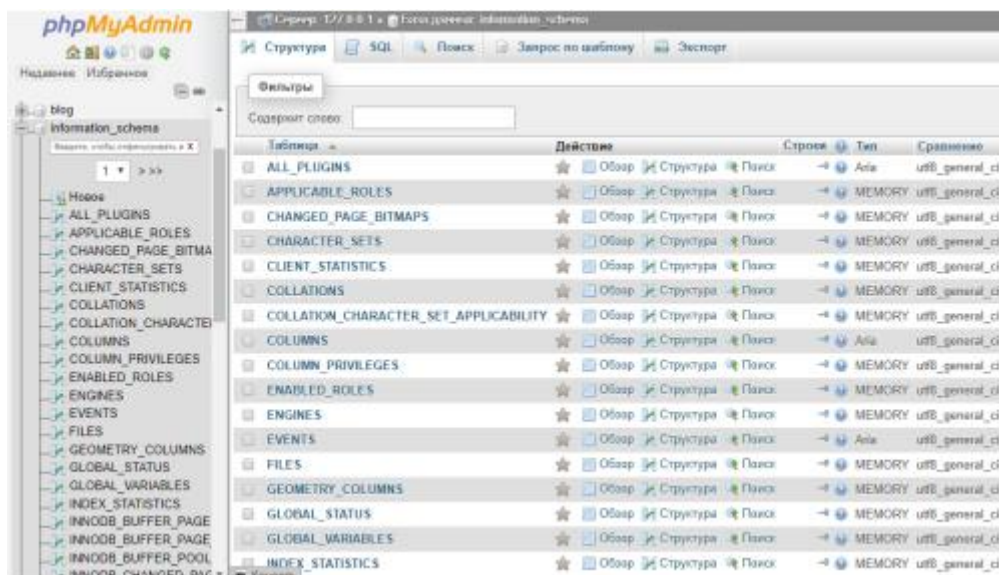


Рисунок 1.7 - phpMyAdmin



## 2. ІНФОРМАЦІЙНІ ТЕХНОЛОГІЇ НАВЧАЛЬНОГО ПРОЦЕСУ

### 2.1 Основні поняття

Протягом 20 століття спостерігалася стійка глобальна тенденція до розширення використання широкомасштабних методів навчання та тестування у всіх сферах людської діяльності. Для українських викладачів питання стає актуальним після появи в країні альтернативних закладів різних форм власності, і, як наслідок, державного механізму акредитації, акредитації та акредитації вищих навчальних закладів у його нинішньому вигляді, який включає елементи технології навчання та тестування учнів.

Найважливішою складовою системи навчання і частиною навчального процесу є контроль навчальної діяльності студентів, спрямований на визначення успішності навчання кожного студента, аналіз отриманих результатів і корекцію процесу безперервної освіти. Удосконалення системи оцінювання та модернізація діагностики навчання учнів є інструментами підвищення якості освіти при реформуванні змісту освіти.

Останнім часом, окрім уже згадуваного поняття «діагностика», замість традиційного поняття «контроль» все частіше вживається поняття «моніторинг». При проектуванні навчального процесу перед учителем постає завдання вибору методів і форматів контролю навчальних досягнень учнів, нормативів якості засвоєння навчального матеріалу, розробки процедур контролю, обґрунтування індивідуальних корекцій навчальної діяльності учнів.

Забезпечення не лише контролю та оцінювання результатів, а й процесу навчання сприяє пошуку дієвих і об'єктивних методів контролю знань. Системи оцінювання та контролю мають відповідати вимогам до управління пізнавальною діяльністю учнів і слугувати відповідним інструментом для їх реалізації [10-11].

Суб'єктивність оцінювання знань певною мірою пов'язана з недостатньою розробленістю методів контролю системи знань. Об'єктивний підхід полягає в тому, щоб завжди використовувати відповідні інструменти для виявлення

наявності знань. Сучасні технології використовують тести як інструмент навчання та вимірювання рівня знань, за допомогою якого можна не тільки визначити якість освіти, а й оптимізувати управління навчальним процесом.

Ефективність функціонування системи професійної освіти значною мірою залежить від діяльності, яка привносить у ВНЗ інноваційну діяльність. Вони змушують усіх учасників освітнього процесу виявляти й аналізувати свої знання, уміння та рівень навичок, спрямовуючи їхню діяльність на перспективний шлях трансформації. При цьому головна роль належить вчителю. Його бажання запроваджувати інновації та його здатність організувати та керувати цим процесом є невід'ємною частиною успішних інноваційних зусиль.

Однією з інноваційних форм комп'ютеризованих систем навчання є тестування.

Test — слово англійського походження, що означає іспит, спроба, перевірка. Тест — стандартизоване завдання, яке за результатами тестування робить висновок про знання, уміння, навички (здібності, професійну придатність, обдарованість тощо) особи, що перевіряється. У сучасній теорії та практиці тестового контролю розрізняють понад 20 видів тестів: залежно від мети, характеру та функції контролю, характеру та форми відповіді.

Іспити як система завдань певної форми та відповідного змісту є науковими інструментами оцінювання знань, умінь і навичок студентів, що сприяють індивідуальному контролю результатів навчання кожного, мобільним процесом управління освітою. Контрольні роботи, як правило, ефективніші за традиційні форми контролю знань (контрольні роботи, іспити, заліки, диференційовані оцінки). Адже недостатньо оцінити рівень знань, треба ще й передбачити, як учні ці знання використовуватимуть. При цьому зростає роль психологічного тестування.

Необхідно поєднати його з перевіркою рівня знань. Тоді результати будуть більш об'єктивними, враховуючи не лише рівень і кількість знань, а й риси характеру учня, його нахили та навчальні можливості, уміння аналізувати, узагальнювати, встановлювати причинно-наслідкові зв'язки. Досвід показує, що в

більшості випадків оцінювання та контроль успішності учнів здійснюється за кінцевими результатами. На жаль, його активність, рівень динамічного розвитку та вплив зовнішніх факторів на момент оцінки оцінити практично неможливо.

Практика показала, що навчання за допомогою тестування має такі переваги перед іншими формами контролю знань:

- Може перевірити якість знань і навичок визначеної кількості студентів протягом досить обмеженого періоду часу;
- Можливий контроль знань, умінь, навичок на необхідних, заздалегідь запланованих рівнях;
- Справжній самоконтроль;
- більш-менш об'єктивно оцінювати знання;
- Увага учнів фіксується не на формуванні відповіді, а на характері її розуміння;
- Створити умови для постійного зворотного зв'язку між учнями та вчителями.

Проте тестовий контроль знань має і суттєві недоліки:

- Імовірність випадкового вибору правильної відповіді;
- Використовуйте закрите тестування, щоб оцінити лише ймовірність кінцевого результату (вірно-хибно), а процес, який призвів до цього, не розкривається;
- Психологічний недолік — стандартизоване мислення, незалежно від рівня розвитку особистості;
- Багато часу, витраченого на складання необхідних тестових «бібліотек», їх варіантів, складність процесу;
- Тести не сприяють розвитку мови [5].

Систематичне навчання за допомогою тестування запроваджено з метою підвищення ефективності системи перевірки знань учнів. Поступово вводяться іспити на спеціальності «Офіс-менеджмент», що дозволяє психологічно підготувати студентів. Спочатку надавалися прості тести з необов'язковими

відповідями, і лише через деякий час з'явилися більш складні конструкції. Навчальні тести, складені з використанням критично-ситуаційного підходу, виявилися високопродуктивними.

Інформація в тестових питаннях може бути подана в довільній формі – текст, графічні зображення, звукова інформація, відео сюжети, формули тощо. Крім того, ми практикуємо відкритість і доступність бази екзаменаційних питань, тобто зміст екзаменаційних питань добре відомий студентам. Але при цьому тести постійно оновлюються з урахуванням змін курсів, програм і сьогоденних реалій. Контрольні роботи з діловодства гарантують об'єктивність в оцінюванні знань, умінь і навичок учнів, сприяють усуненню проявів суб'єктивізму, а отже формують позитивне ставлення до предмета та вчителя.

При проведенні екзаменаційного контролю дуже важливо дотримуватися організаційних моментів (пояснити мету іспиту, послідовність і схему його проведення, визначити терміни і проведення, а також надати кожному студенту бланк відповідей встановленого зразка, що значно економить час іспиту. Студенти та викладачі).

Тестові завдання значно скорочують час очікування студентами оцінок, що є важливим фактором – як психологічним, так і педагогічним.

Навчання на тренажері може охоплювати окрему тему, частину або весь матеріал курсу, а його результати повинні бути проаналізовані. Розбір необхідний для того, щоб учень міг перевірити, наскільки добре він оцінює свої знання, вірить у свої сили і коригує свою підготовку. Учитель не тільки фіксує неправильні факти і називає правильну відповідь, але й детально пояснює учням причини неправильної поведінки.

Учням, які допустили помилку, було запропоновано письмово або усно відповісти на запитання: Чому обрана відповідь неправильна? Надайте докази на підтримку правильного вибору. Вивчення неправильної відповіді, усвідомлення помилки та її причини, знаходження неправильної ланки в ланцюзі розумової діяльності значно зменшує можливість запам'ятовування неправильних знань і

«заміни» правильної відповіді. Таким чином виключається практика «вгадування» студентами правильного розв'язання поставлених на іспиті питань.

Наступним етапом впровадження та використання тестів при вивченні різних предметів є їх електронна форма. Їх робота забезпечується на локальному комп'ютері та локальній мережі класу. Такий формат часто викликає додатковий інтерес у студентів, крім того, він дозволяє їм самостійно перевірити та оцінити свій рівень знань з певної теми або набору тем курсу без участі викладача. Програми комп'ютерного тестування використовують анімацію, звуки, елементи гри та систему для запису прогресу та результатів тестування, особливо таких показників, як час вирішення кожного запитання, кількість помилок і правильних відповідей. Підсумковий рахунок [2-3].

Важливою умовою контрольної роботи як загально навчального засобу є її періодичність, яка залежить від навчального предмета, його ролі та місця в навчальній програмі, специфіки засвоєння знань. Тестування повинно стати звичною і зручною формою регулярного контролю знань студентів. Необхідно пам'ятати, що контрольна робота - це не самоціль, а ефективна форма повторення - узагальнення й організації вивченого. Контрольно-оцінювальна функція навчання є лише одним із елементів добре організованого, технічно продуманого навчального процесу. Якщо учні володіють міцними знаннями, їх оцінки, у будь-якій формі, не повинні викликати особливих труднощів.

## **2.2 Форми навчального процесу**

Сутнісними характеристиками виховного процесу є методи і засоби виховання. Вони можуть забезпечити досягнення необхідних освітніх цілей, а якщо для цього потрібна матеріально-технічна база, вчителі матимуть право вибирати організаційну сторону навчання, форму (або тип) занять.

У педагогічній практиці склалися відомі формати навчання, такі як лекції, семінари, лабораторні заняття, контрольні, курсові роботи, вікторини, іспити, консультації, самостійна робота тощо, а також безконтактне навчання. Дистанційне

навчання за допомогою комп'ютерного спілкування має курси в наступних форматах.

Лекції є однією з найважливіших форм навчання і складають основу теоретичної підготовки студентів. Мета — системне обґрунтування дисциплінарних наукових знань, визначення проблем, стану та перспектив розвитку окремих науково-технічних напрямів, зосередження уваги на найбільш складних і гострих проблемах. Систематичний курс лекцій, у якому навчальний матеріал викладається послідовно, традиційно складається зі вступу, вступу, огляду, огляду та заключної лекції.

Семінарські заняття є активною формою навчання, яка широко використовується при викладанні широкого кола предметів. Майстер-класи часто базуються на жвавих творчих дискусіях і є дружніми дискусіями на розглянуту тему. Семінари можна проводити в дитячих садках з використанням комп'ютерної відео- та телеконференції. За змістом навчання відеOVERсія нічим не відрізняється від традиційної, оскільки учасники процесу можуть бачити один одного на екрані монітора комп'ютера. Семінар, в якому використовується телеконференція (тобто використовується письмове, невербальне спілкування), можна назвати віртуальним семінаром, оскільки учасники не бачать один одного, а лише обмінюються текстовими повідомленнями. Семіни розширені та асинхронні, і викладачі можуть оцінити прогрес кожного студента. Якщо засідання не імітується (контролюється), кожен учасник може бачити на екрані всі тексти запитань і відповідей усіх інших учасників семінару.

Консультування – це форма наставництва студентів у їх роботі та допомоги їм у самостійному вивченні матеріалу. Консультування може бути індивідуальним або груповим. Під час консультації виявляються особистісні якості слухача як особистості (його інтелектуальні та моральні якості, психологічні та свідомі особливості студента: увага, пам'ять, уява та мислення). SDO в основному використовує такі методи консультацій, як телефон, електронна пошта, відео та конференц-дзвінки. Варіанти фінансування нових інформаційних технологій

залежатимуть від апаратного та програмного забезпечення, доступного студентам і викладачам. Найпоширенішими є телефонні дзвінки та електронні листи.

Лабораторна робота. Таке навчання в першу чергу важливе для технічних професій. Вони можуть бути реалізовані, коли необхідний віддалений доступ до лабораторних приміщень через комп'ютерні мережі або імітаційні експерименти на центральному комп'ютері, або коли є потреба забезпечити портативну лабораторну практику «вдома».

Контроль - Контроль у навчальному процесі включає перевірку процесу і результатів теоретичного і практичного засвоєння студентами навчального матеріалу. Оцінка знань, умінь і навичок, набутих у дистанційному навчанні, набуває все більшої актуальності через відсутність прямого контакту між викладачем та студентом. Зростає роль і значення об'єктивних і багатокритеріальних форм контролю якості знань. Особливістю контролю в LMS є необхідність введення додаткових засобів ідентифікації студентів з метою виключення можливості освітньої фальсифікації.

Веб-портали навчальних закладів належать до групи освітніх сайтів. Під освітнім веб-порталом розуміється набір веб-сторінок з повторюваним дизайном, що здійснюють цілеспрямований процес навчання та виховання в інтересах особистості, суспільства та держави, об'єднаних за змістом, навігаційно та фізично розташованих на одному сервері.

Створення веб-порталу – це подія, яка підвищує імідж навчального закладу. Зацікавившись будь-яким закладом чи подією в цьому закладі, сучасний користувач перш за все захоче зайти на офіційну сторінку та отримати більш достовірну інформацію. Грамотно продуманий веб-портал зможе в кілька разів збільшити увагу аудиторії та підвищити рейтинг навчального закладу. Створення веб-порталу – це чудова можливість продемонструвати всім свої досягнення, розмістити актуальні новини, наукові та публіцистичні статті, публікації. Це гарна можливість продемонструвати особливості свого закладу, розповісти про досягнення та надати всім користувачам інформацію, яка їх зацікавить [5].

Веб-портали навчального закладу залежно від змісту можуть бути різних типів. Такі веб-портали називають багатофункціональними. В результаті свого зростання на різних етапах свого існування сайт може змінювати свої завдання, зміст і функціональність.

Розрізняють такі види сайтів навчальних закладів:

- сайти навчальних закладів; - дослідницькі ділянки;
- довідкові сайти;
- сайти конкурсних та інформаційних Інтернет-проектів; - сайти дистанційної освіти;
- сайти для поширення культурно-освітньої інформації;
- сайти типу віртуальних методичних об'єднань;
- консультаційні сайти, освітні портали [8].

Для інформаційної підтримки та комунікації створено сайт навчального закладу. Він слугує мережевим інструментом для всіх учасників освітнього процесу та сприяє поширенню інформації про різноманітні заходи. На таких сайтах учні та студенти можуть мати особисті акаунти.

Сайти навчальних закладів можна розділити на:

- сайти навчальних закладів (початкові школи, гімназії, ліцеї);
- сайти закладів професійної освіти (училищ, технікумів);
- сайти вищих навчальних закладів (інститутів, університетів).

Основними завданнями таких сайтів є:

- надання загальної інформації про заклад;
- надання інформації про педагогічний та учнівський колективи;
- поповнення банку навчально-методичних матеріалів для викладача;
- висвітлення новин;
- розміщення повідомлень, корисної інформації, консультацій соціального педагога та практичного психолога для батьків;
- висвітлення виховної роботи;
- інформування батьків та учнів про корисні навчальні ресурси;



– забезпечення можливостей онлайн-спілкування з батьками, учнями, представниками державних служб, організацій та іншими зацікавленими особами [7].

## **2.3 Основні вимоги до запуску та роботи з програмним засобом**

### **2.3.1. Функціональні вимоги**

До програмного забезпечення висуваються наступні вимоги:

- можливість пройти тестування;
- реалізація алгоритму завантаження тестів;
- реалізація методу підключення до бази даних, пошук інформації;
- візуалізація результатів роботи програми у вигляді додатка.

### **2.3.2. Вимоги до складу та параметрів технічних засобів**

Мінімальні вимоги до апаратного забезпечення комп'ютера або мобільного пристрою для коректної роботи розробленого програмного продукту:

- Процесор - 2 x IntelXeon3 ГГц;
- обсяг оперативної пам'яті – 16 ГБ;
- дискова підсистема – 4 x 146 ГБ;
- Дискковод компакт-дисків (DVD-ROM);
- Мережевий адаптер - 100 Мбіт/с.

для ПК користувача:

- процесор - Intel Pentium 1,5 ГГц;
- обсяг оперативної пам'яті – 256 МБ;
- дискова пам'ять – 40 ГБ;
- мережевий адаптер - 100 Мбіт/с.

Для коректного відображення застосунку необхідна наявність веббраузера із підтримкою стандартів HTML5, наприклад Google Chrome, Mozilla Firefox чи інші аналоги.

Розроблене програмне забезпечення може виконуватись на операційних системах Windows та Linux за умови, що встановлене програмне забезпечення, необхідне для роботи продукту.

### **2.3.3. Вимоги до вхідних та вихідних даних**

Вхідними даними для розробленого програмного продукту є база даних з інформацією про тести, відповіді та користувачів.

Вихідними даними є додаток з можливістю реєстрації та проходження тестів.

### **2.3.4. Вимоги до інтерфейсу**

Інтерфейс програмного продукту повинен бути декларативним. Користувач повинен мати можливість реєструватися в системі, переглядати свій профіль, проходити навчання та тести.

Взаємодія із застосунком здійснюється через веб-інтерфейс.

Користувач має можливість переглядати завдання та свій профіль.

Окрім цього, у користувача повинна бути можливість розширювати функціональність розробленого програмного продукту, маючи можливість здійснювати сортування питань.

### **2.3.5. Вимоги до тестування програмного забезпечення**

Для тестування програмного забезпечення необхідно виконати наступні дії:

- Зареєструватися в системі;
- Зробити вхід в систему;
- Пройти тести.

При виконанні вище перерахованих дії для тестування роботи програмного забезпечення, на користувач буде зареєстрований у системі, його данні з'являться у базі даних, та матиме можливість проходити навчання.

### **2.3.6 Вимоги до програмної документації**

Програмне забезпечення постачається разом із супроводжувальною документацією, до складу якої входить:

- Технічне завдання.
- Опис та обґрунтування обраної архітектури.
- Функціональна специфікація.
- Технічна специфікація.
- Опис програми.
- Програма та методика випробувань.

## **2.4 Проектування архітектури системи**

Архітектура програмного забезпечення дає пояснення того, як ваші системи поведуться на структурному рівні. Системи, які ви використовуєте, мають набір компонентів, розроблених для виконання певного завдання або набору завдань. Архітектура програмного забезпечення забезпечує фундамент, на якому все програмне забезпечення, яке є у компанії, можна змінити, створити або вилучити з експлуатації.

Архітектура програмного забезпечення впливає на якість, продуктивність, обслуговування та успіх системи на основі дизайну. Не розглядаючи архітектуру програмного забезпечення на регулярній основі, компанія відкриває себе для довгострокових наслідків, і проблеми, які можуть поставити їх системи під загрозу поломки, злому або низької продуктивності.

У сучасних системах існують загальні шаблони в архітектурі програмного забезпечення, які називаються архітектурними системами для програмного забезпечення. У більшості випадків для створення цілісної системи використовується кілька різних архітектурних систем, особливо для систем, які створювалися роками або працювали, або тих, які були побудовані різними розробниками.

## 2.5 Проектування інтерфейсу користувача

Логічні моделі для розробки програмного забезпечення

За допомогою діаграм прецедентів, діаграм варіантів використання основні користувачі системи та завдання, які має вирішувати система. За допомогою діаграм діяльності описується послідовність дій для кожного прецеденту, яка необхідна для досягнення поставленої мети.

Діаграма зв'язку — це особливий вид діаграми взаємодії, яка зосереджена на обміні даними між різними учасниками взаємодії.

На діаграмі зв'язку не потрібно показувати кожного учасника як рятувальний круг, а також не потрібно показувати послідовність повідомлень по вертикалі як діаграму послідовності. Натомість учасників можна розміщувати за потреби, дозволяючи комунікаціям показувати стосунки між учасниками та використовувати числа для представлення послідовності повідомлень. Результат збірки показано на рисунку 2.1.

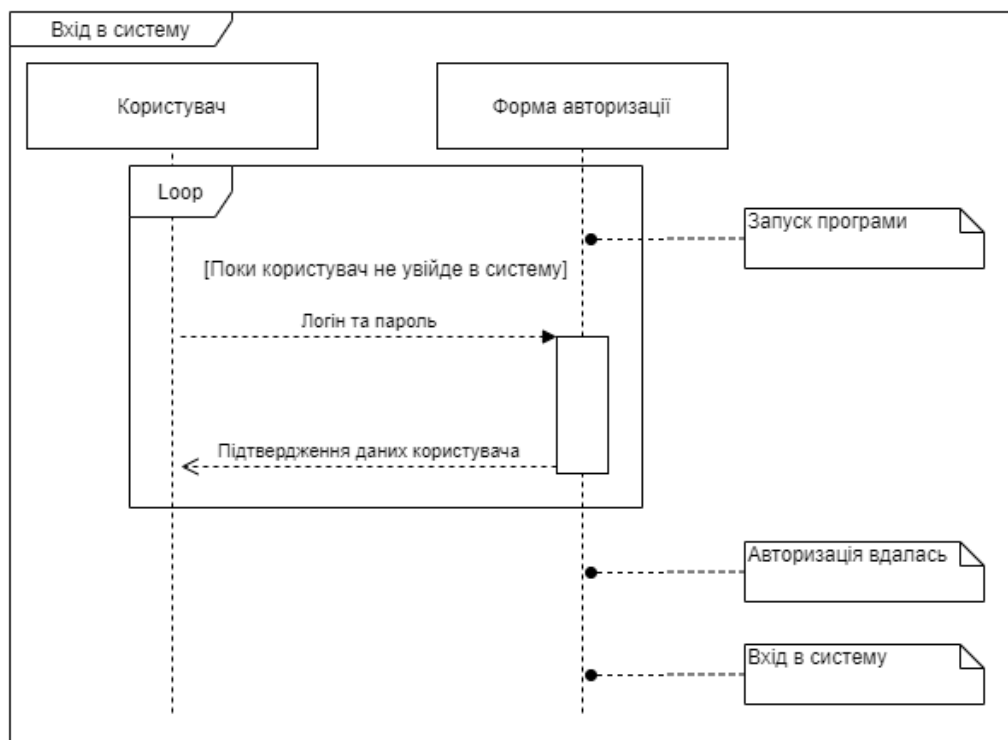


Рисунок 2.1 – Діаграма послідовності запису

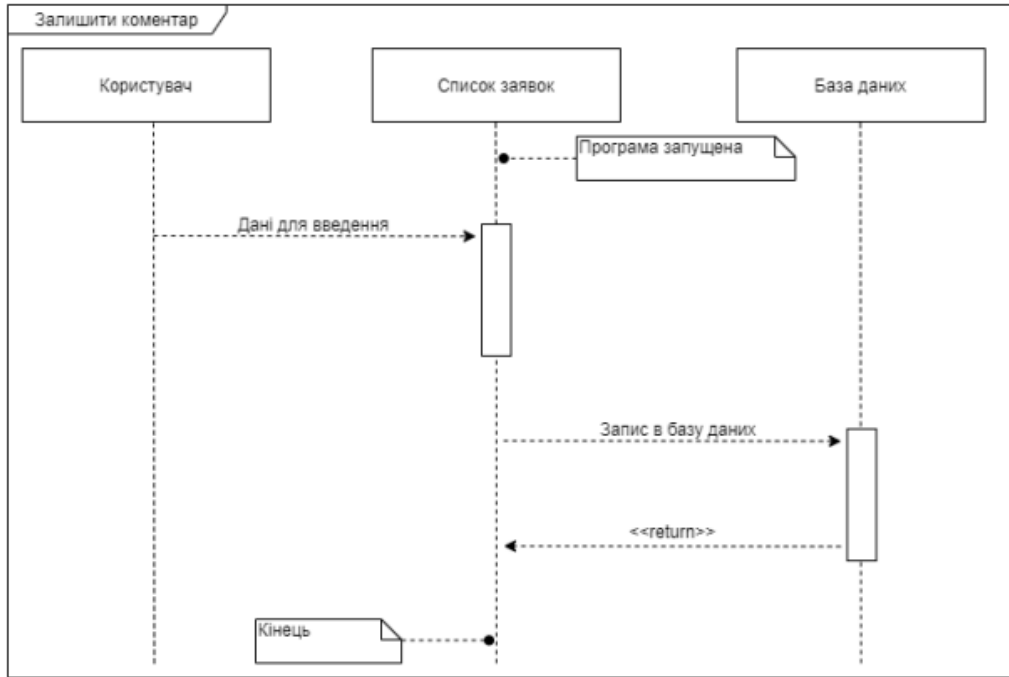


Рисунок 2.2 – Діаграма послідовності анотацій

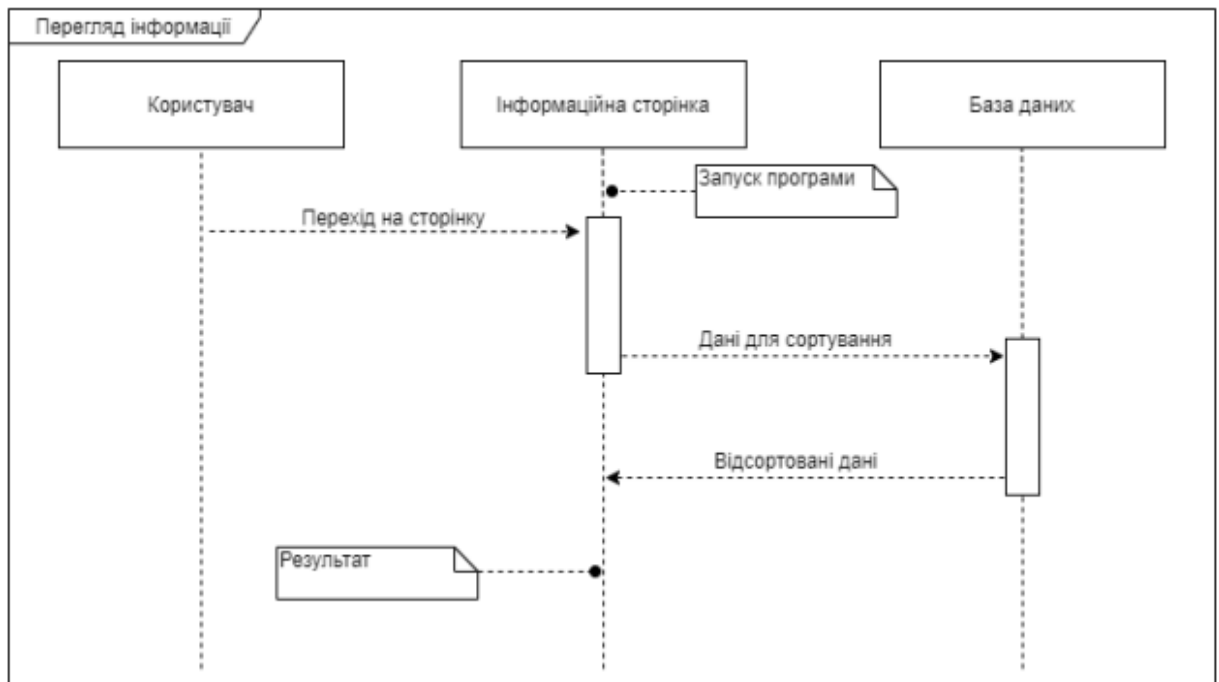


Рисунок 2.3 – Діаграма послідовності для перегляду інформації

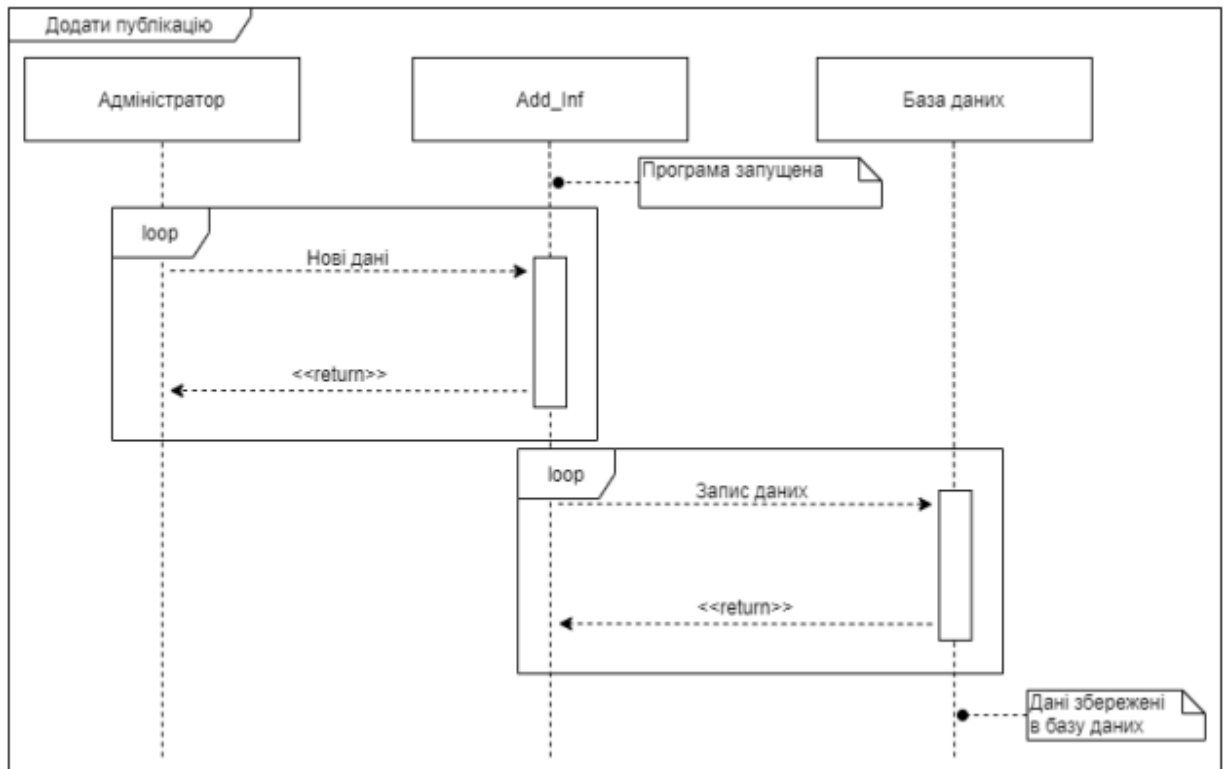


Рисунок 2.4 – Діаграма послідовності додавання інформації

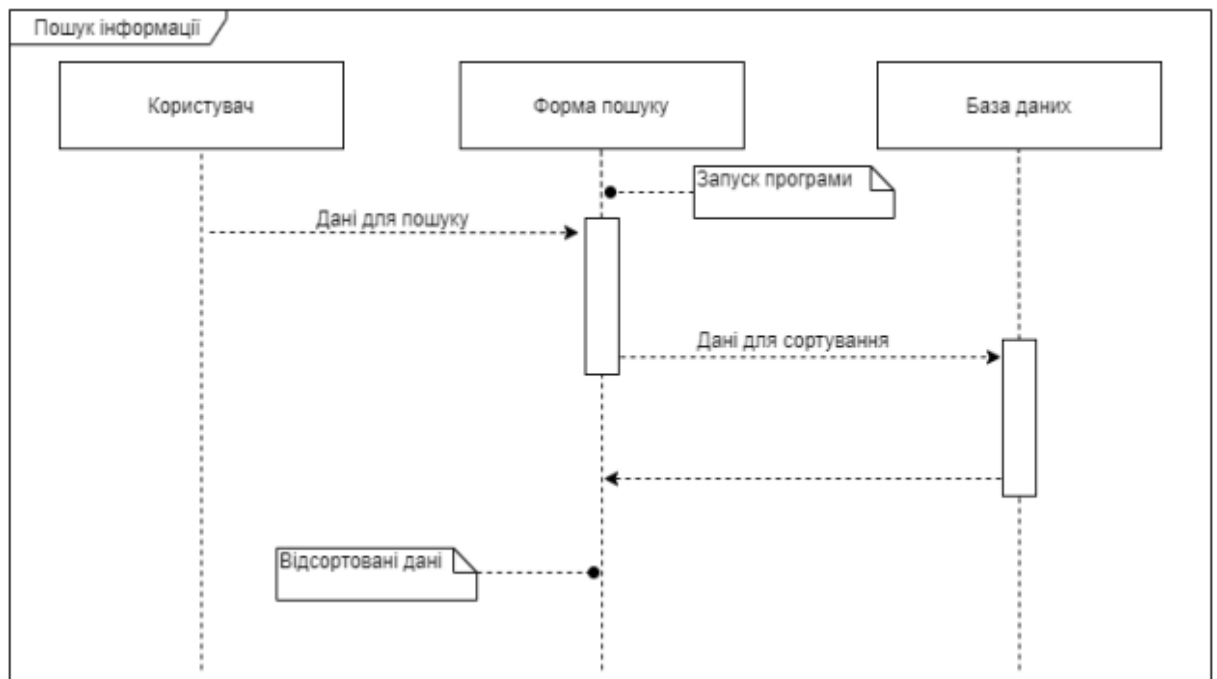


Рисунок 2.5 – Діаграма послідовності пошуку інформації

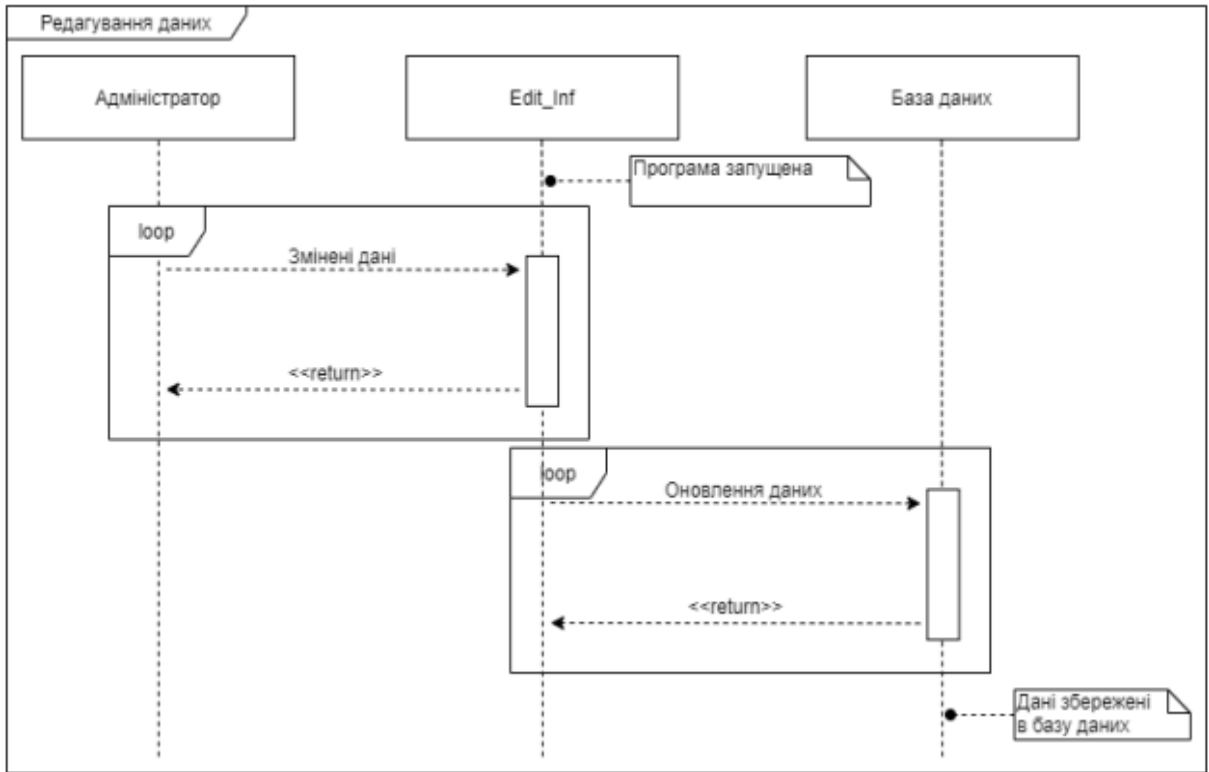


Рисунок 2.6 – Діаграма послідовності редагування інформації

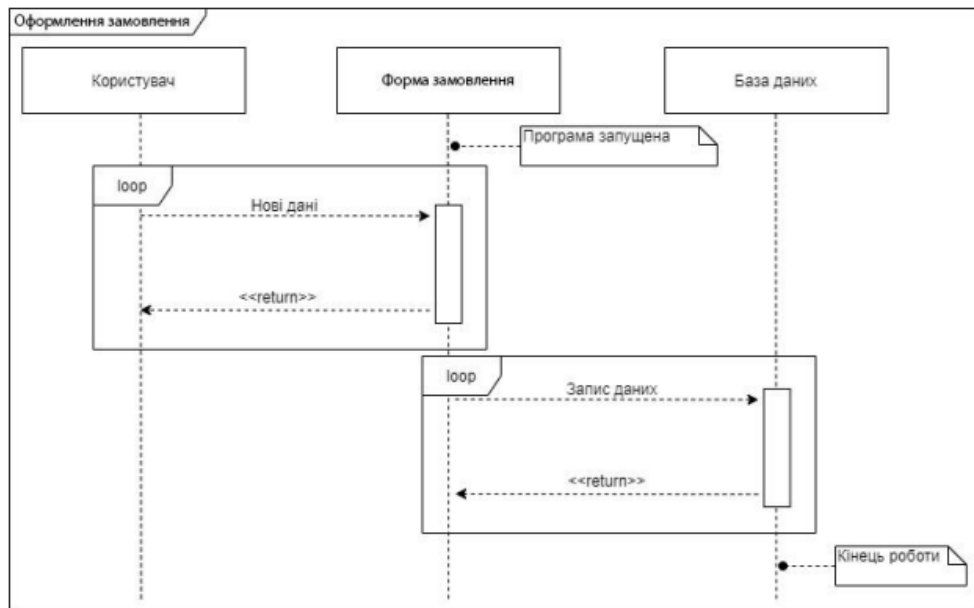


Рисунок 2.7 – Принципова схема процесу замовлення

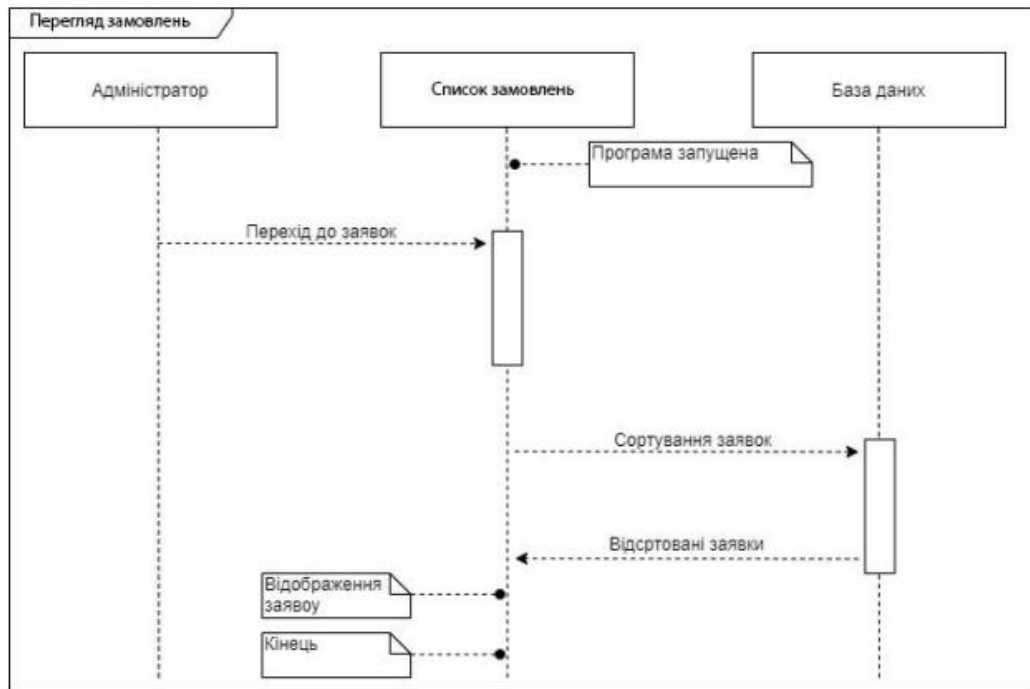


Рисунок 2.8 – Діаграма послідовності перегляду замовлення



## РОЗДІЛ 3. ЕТАПИ РОЗРОБКИ ТА СТВОРЕННЯ ПРОГРАМНОГО ДОДАТКУ

### 3.1 Структура програмного додатку

Розроблена програма в рамках даної дипломної роботи навчальний тренажер для тестування студентів є багаторівневою прикладною програмою.

Обробка даних у таких системах здійснюється за допомогою елементів, розбитих на кілька логічних секцій. Розділення компонентів програми на різні рівні може підвищити робочу надійність і масштабованість програми. Це досягається завдяки можливості застосування нових технологій до будь-якого рівня без зміни всього рішення. Архітектура типової багаторівневої програми включає рівень презентації, середній рівень і рівень даних. Середній рівень зазвичай включає рівень доступу до даних, рівень бізнес-логіки та загальні компоненти, такі як автентифікація та інші засоби перевірки. Рівень даних включає реляційні бази даних. Багаторівневі програми зазвичай зберігають конфіденційну інформацію на рівні доступу до даних середнього рівня, щоб підтримувати ізоляцію від кінцевих користувачів, які отримують доступ до рівня презентації.

Вид дизайну.

Навчальний тренажер реалізує такі функції:

- аутентифікація та керування користувачами;
- тестування користувача;
- збір статистичних даних;
- отримати сертифікат;
- опублікувати результати здачі тестів.

Тестовою системою є клієнт (тонкий клієнт)/сервер з трирівневою архітектурою.

Таким чином, розроблена система має рівень сервера, рівень клієнта та рівень бази даних.

Клієнтська частина (рівень 1) відповідає за представлення даних користувачеві. Користувачеві достатньо мати програму-браузер для взаємодії з тестовою системою. За допомогою програми-браузера користувач взаємодіє з сервером бази даних і прикладною програмою через мережу.

Бізнес-логіка (другий рівень) і рівень обробки даних відповідають за основну логіку додатків і логіку обробки даних. На цьому рівні знаходиться додаток тестової системи.

Сервер (рівень 3) відповідає за надання даних клієнтам.

Представляє інтерфейс користувача та основну бізнес-логіку програми керування послугами та забезпечує доступ до даних незалежно від їх розташування. Додаток встановлюється на сервер разом з базою даних і взаємодіє з користувачем через браузер.

Незалежність від платформи.

Оскільки операційна система Windows встановлена практично на всіх існуючих обчислювальних платформах, навчальна система гарантує взаємодію з користувачем, незалежно від платформи.

Подальший розвиток навчального тренажеру не вимагатиме зміни іншої операційної системи користувача або іншого інтерфейсу користувача.

Розширюваність.

Ця техніка економить час, витрачений на розгортання програми, розміщуючи всю функціональність програми на окремому сервері та видаляючи її з клієнтських програм. У той же час це спрощує оновлення системи та управління при роботі з різними обчислювальними платформами.

Додаток використовує внутрішній контейнер і зовнішнє сховище для зберігання результатів. Внутрішній контейнер призначений для зберігання таких речей:

- запитання та альтернативні відповіді
- графічні об'єкти та малюнки;

Цей сценарій демонструє роботу навчального тренажеру, її взаємодію з програмним продуктом, встановленим на сервері.

В програмі також було реалізовано перемикач мови, на вибір було: українська чи англійська.

### 3.2 Проектування та розробка БД

У системі керування контентом застосовується СУБД MySQL. База даних складається з таблиці quiz.

Усі дані зберігаються на локальному сервері в окремих базах даних

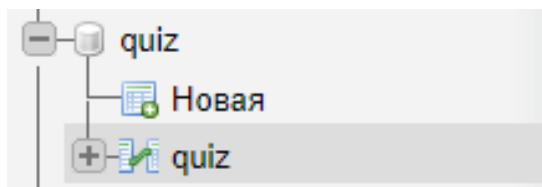


Рисунок 3.1 - Структура БД(Загальна)

Нижче описано структуру БД, яка була створена для додатку

+ Options		id	fotourl	surname	name	test1	test2	test3	test4	test5	test6	test7	test8	test9	test10	test11	login	pass
<input type="checkbox"/>	Edit Copy Delete	1	1.jpg	Іванов	Іван	1	1	1	1	1	1	0	0	0	0	0	1	1
<input type="checkbox"/>	Edit Copy Delete	2	2.jpg	Петров	Петр	0	0	1	0	0	0	0	0	0	0	0	2	2
<input type="checkbox"/>	Edit Copy Delete	3	3.jpg	Демідова	Тетяна	0	1	0	0	0	0	0	0	0	0	0	3	3
<input type="checkbox"/>	Edit Copy Delete	4	4.jpg	Неандерталець	Неандерталець	0	1	1	0	0	0	0	0	0	0	0	4	4
<input type="checkbox"/>	Edit Copy Delete	5	5.jpg	Сидоренко	Максим	1	1	1	0	0	0	0	0	0	0	0	5	5
<input type="checkbox"/>	Edit Copy Delete	6	10.jpg	Іванов	Іван	1	1	0	0	0	0	0	0	0	0	0	6	6
<input type="checkbox"/>	Edit Copy Delete	7	9.jpg	Іванов	Іван	1	0	1	0	0	0	0	0	0	0	0	7	7
<input type="checkbox"/>	Edit Copy Delete	8	10.jpg	Іванов	Іван	0	0	0	0	0	0	0	0	0	0	0	10	10

Рис. 3.2 - Структура БД(quiz)

Підключення до бази даних здійснюється за допомогою функції

```
class DB
```

```
{
```

```
        MySqlConnection connection = new
MySqlConnection("server=localhost; port = 3306; username=root;
password=root; database=quiz");

        public void openConnection()
        {
            if (connection.State ==
System.Data.ConnectionState.Closed)
                connection.Open();
        }

        public void closeConnection()
        {
            if (connection.State ==
System.Data.ConnectionState.Open)
                connection.Close();
        }

        public MySqlConnection getConnection()
        {
            return connection;
        }
    }
}
```

### 3.3 Опис функціоналу

Для запуску програми необхідно перейти наступним шляхом і запустити програму quiz (рис. 3.3)

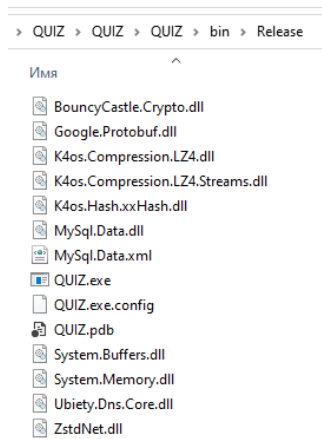


Рисунок 3.3 - Шлях до файлу

Запустивши програму, перед вами з'явиться наступне вікно (рис. 3.4)

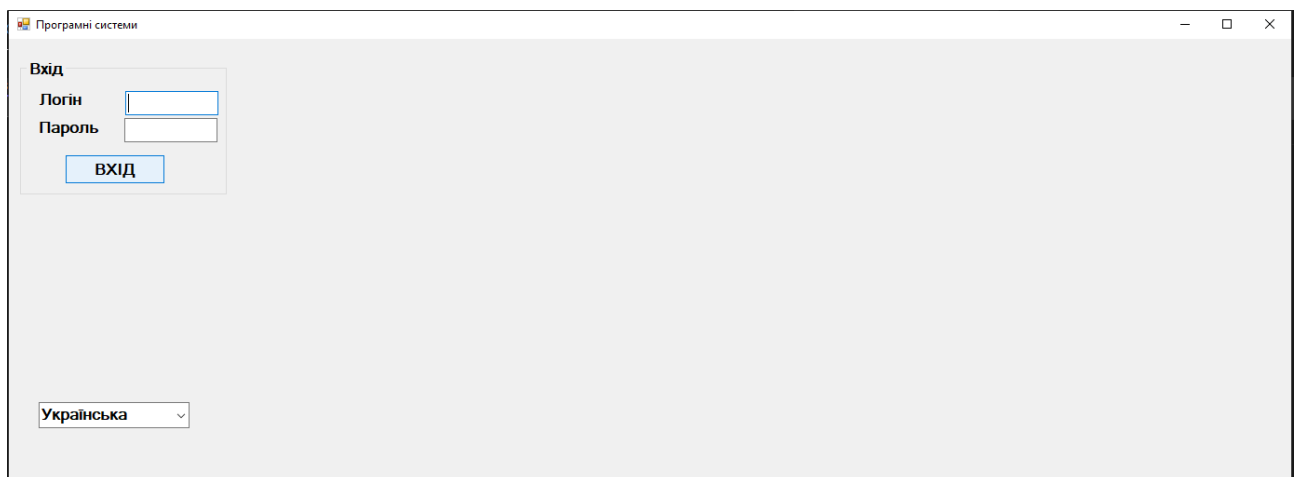


Рис. 3.4 – Стартове вікно програми

Ввівши логін та пароль, ви можете увійти як звичайний користувач (рис. 3.5) так і як адміністратор (рис. .6).

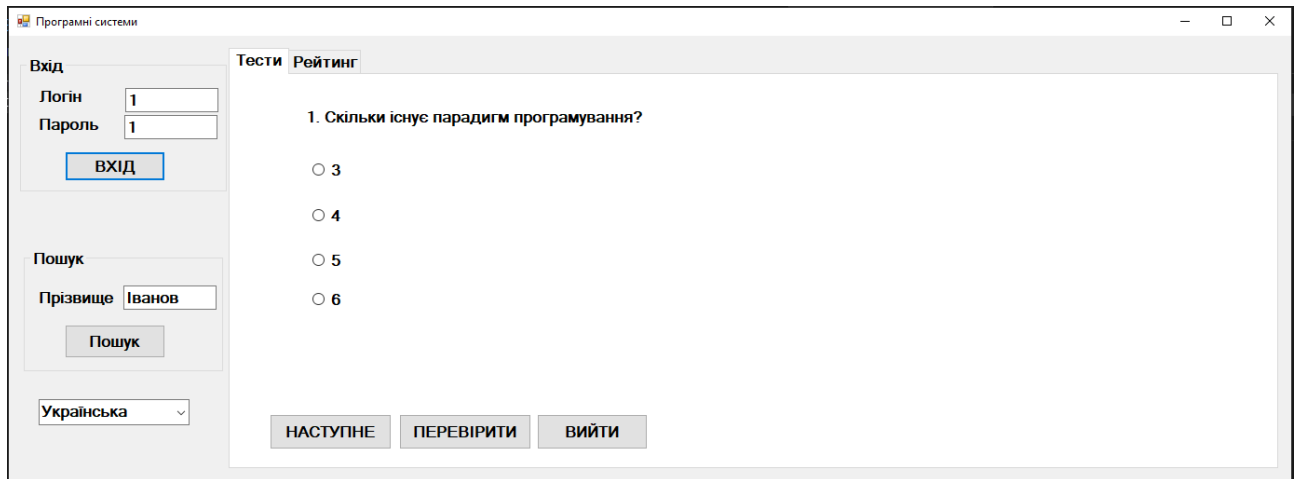


Рисунок 3.5.1 – Вид програми звичайного користувача

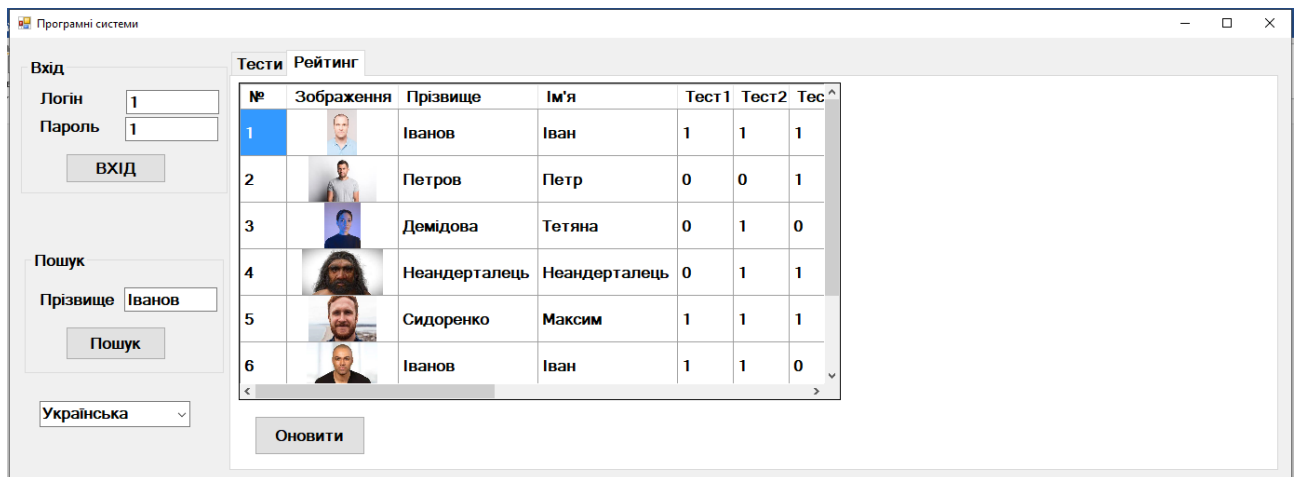


Рисунок 3.5.2 – Вид програми звичайного користувача

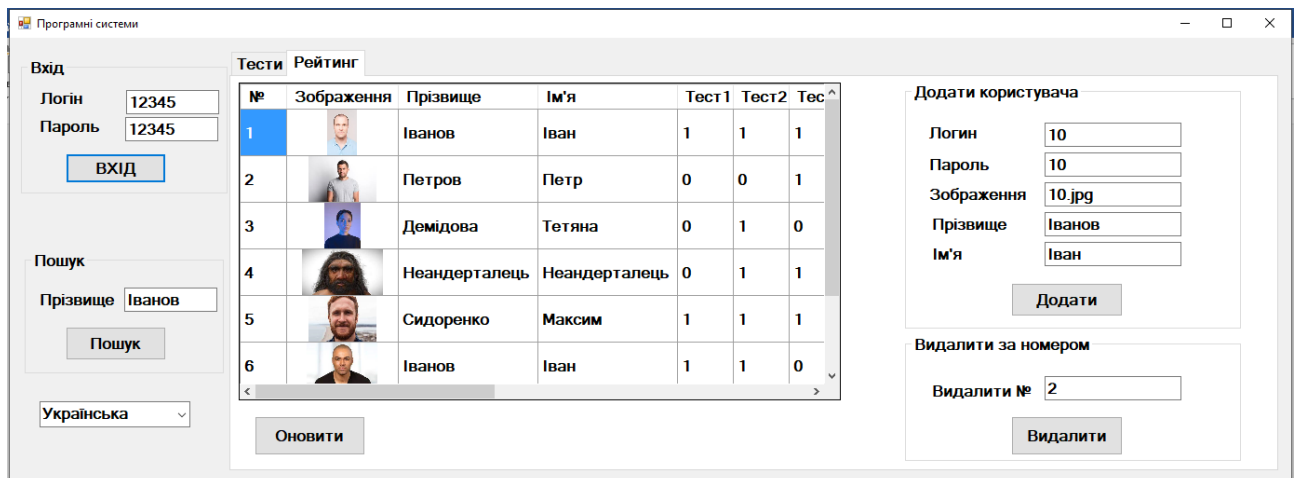


Рисунок 3.6 – Вид програми адміністратора

Для пошуку користувача необхідно ввести прізвище, і пошукова система знайде відповідних клієнтів (рис. 3.7)

The screenshot shows a web application window titled "Програмні системи". It features a search section on the left with fields for "Логін" (12345) and "Пароль" (12345), a "ВХІД" button, and a "Пошук" section with a "Прізвище" field containing "Іванов" and a "Пошук" button. Below the search is a language dropdown set to "Українська". The main area contains a table with columns "№", "Зображення", "Прізвище", "Ім'я", "Тест1", "Тест2", and "Тест3". The table lists four users with the surname "Іванов". The first row is highlighted in blue. To the right of the table is a "Додати користувача" section with fields for "Логін" (10), "Пароль" (10), "Зображення" (10.jpg), "Прізвище" (Іванов), and "Ім'я" (Іван), along with a "Додати" button. Below that is a "Видалити за номером" section with a "Видалити №" field containing "2" and a "Видалити" button. An "Оновити" button is located at the bottom of the table area.

№	Зображення	Прізвище	Ім'я	Тест1	Тест2	Тест3
1		Іванов	Іван	0	1	0
6		Іванов	Іван	1	1	0
7		Іванов	Іван	1	0	1
8		Іванов	Іван	0	0	0

Рисунок 3.7 – пошук користувача

Для видалення клієнта з бази даних необхідно ввести номер користувача, який хочете видалити у відповідне поле (рис. 4.8)

The screenshot shows a form titled "Видалити за номером". It contains a label "Видалити №" followed by a text input field containing the number "2". Below the input field is a "Видалити" button.

Рисунок 3.8 – Видалення користувача

Для додавання користувача необхідно заповнити поля та натиснути на кнопку «Додати» (рис. 3.9)

**Додати користувача**

Логин	<input type="text" value="10"/>
Пароль	<input type="text" value="10"/>
Зображення	<input type="text" value="10.jpg"/>
Прізвище	<input type="text" value="Іванов"/>
Ім'я	<input type="text" value="Іван"/>

Рисунок 3.9 – Додавання клієнта

### 3.4 Тестування

Під час тестування, всі помилки, що виникали, були виправлені. Нижче наведено приклади тестування програми.

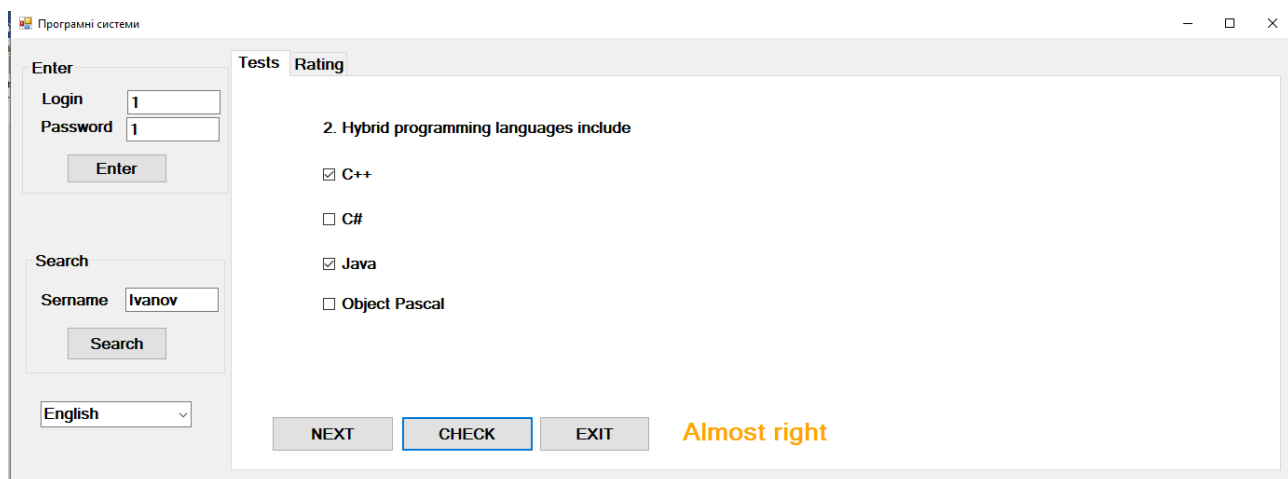


Рисунок 3.10 – тестування програми



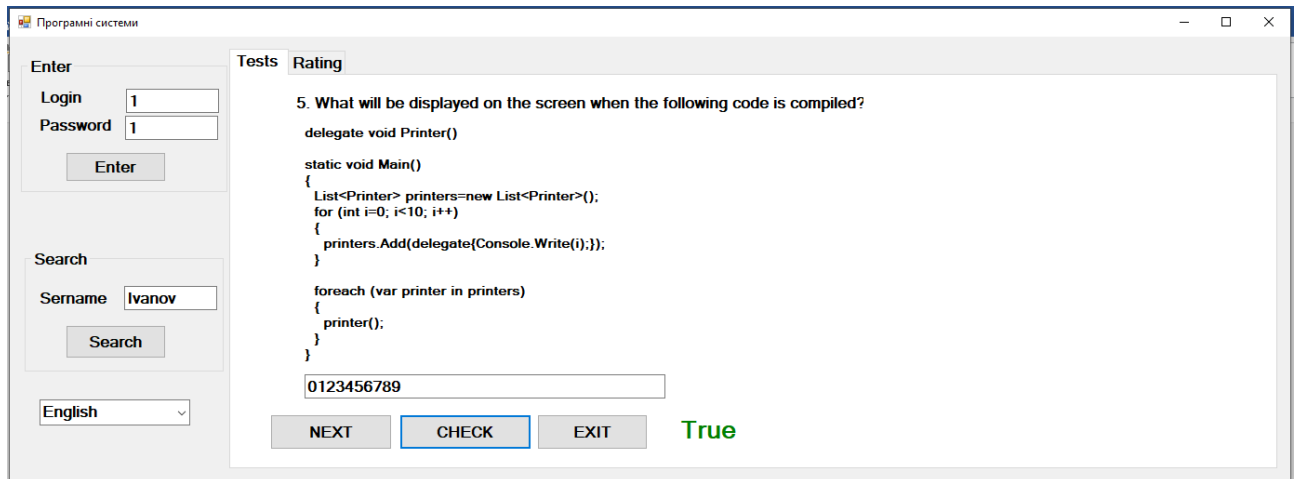


Рисунок 3.11 – тестування програми

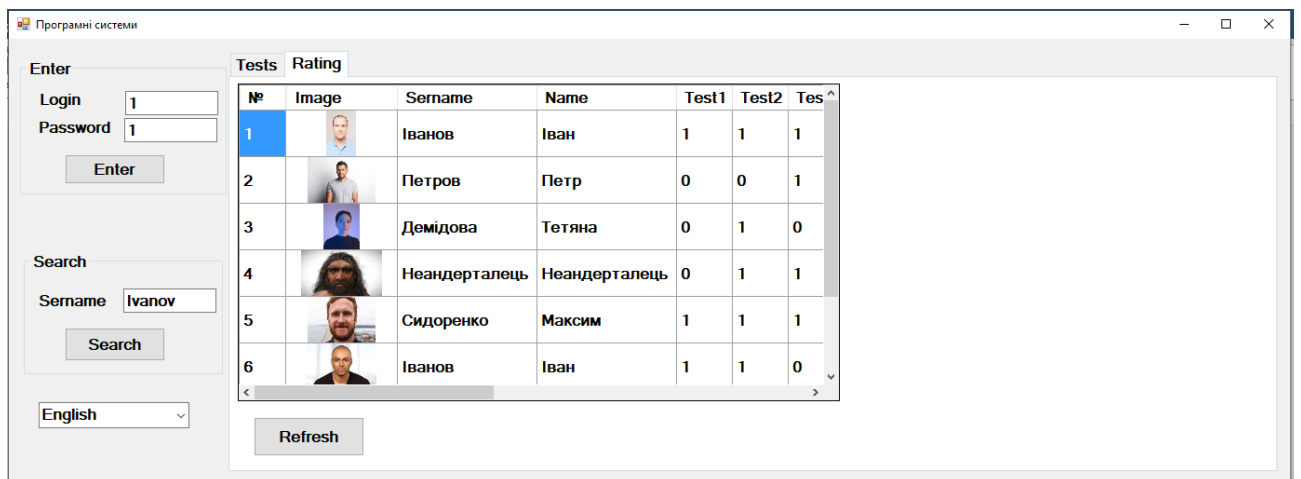


Рисунок 3.12 – тестування програми

## ВИСНОВКИ

В рамках виконання дипломної роботи побудовано алгоритм та створено програмну реалізацію навчального тренажеру на мові С#.

Для досягнення мети були реалізовані наступні завдання:

- опрацьовано методичні рекомендації до виконання дипломної роботи;
- сформульовано постановку задачі;
- виконано огляд літературних джерел, що використовується для вивчення даної теми, а також здійснено огляд існуючих програмних продуктів, які використовуються при вивченні різних дисциплін;
- опрацьовано теоретичний матеріал з теми;
- розроблено алгоритм навчального тренажеру;
- побудовано блок–схеми алгоритму тренажеру;
- здійснено вибір оптимальної платформи для програмної реалізації тренажеру;
- реалізована програма-тренажер;
- здійснено опис програми.

Для створення програми було обрано мову програмування С# у середовищі Visual Studio 2022. Даний тренажер впроваджено для навчання.

## СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Бабій М. С. Теорія програмування. Суми: Вид-во СумДУ, 2009. 181 с.
2. Вінник В. Ю. Алгоритмічні мови та основи програмування: мова С. Житомир: ЖДТУ, 2007. 328 с.
3. Ковалюк Т. В. Алгоритмізація та програмування: підручник з грифом МОН України. Львів: Магнолія-2016, 2013. 400 с.
4. Мартынов Н. Н. Программирование для Windows на C/C++ М.: Бином-Пресс, 2014. 528 с.
5. Нікітченко М. С. Теоретичні основи програмування. Київ: КНУ ім. Т. Г. Шевченка, 2019. 200 с.
6. Трофименко О. Г. С++. Теорія та практика. 587 с.
7. Документація Visual Studio [електронний ресурс]. Режим доступу: <https://docs.microsoft.com/ru-ru/visualstudio/ide/?view=vs-2017>
8. Введення мови програмування С і С++ [електронний ресурс]/режим доступу: <http://www.intuit.ru/studies/courses/1039/231/info>
9. Верифікація програмного забезпечення [електронний ресурс]/режим доступу: <http://www.intuit.ru/studies/courses/1040/209/info>
10. Інструменти, алгоритми та структури даних [електронний ресурс] / режим доступу: <http://www.intuit.ru/studies/courses/683/539/info>
11. MySQL. [Електронний ресурс] – Режим доступу: <https://www.mysql.com/> (дата звернення 12.05.23р)
12. Г. Шілдіт «Самовчитель С#» - Київ.: Освіта. 2017. – 670с.
13. А. Мешков, Ю. Тихомиров "Visual C++ і MFC" - Харків.: Місто. 2018 – 1017с.
14. Кульгін Н. «С# в задачах та прикладах» - Одеса.: МКТ, 2012. - 288 с.
15. Петров Б., Алексєєв Т. "С#" - Запоріжжя, 2021. - 370 с.: іл.
16. Брагін І. «Побудова мереж» – Київ.:Київ, 2020. – 480 с.: іл.

## ДОДАТКИ

### ДОДАТОК А

```
using MySql.Data.MySqlClient;
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Threading.Tasks;

namespace library
{
    class DB
    {
        MySqlConnection connection = new
        MySqlConnection("server=localhost; port = 3306; username=root;
        password=root; database=quiz");

        public void openConnection()
        {
            if (connection.State ==
            System.Data.ConnectionState.Closed)
                connection.Open();
        }

        public void closeConnection()
        {
            if (connection.State ==
            System.Data.ConnectionState.Open)
                connection.Close();
        }

        public MySqlConnection getConnection()
```

```
    {  
        return connection;  
    }  
}  
}
```

## ДОДАТОК Б



**ДЕРЖАВНИЙ УНІВЕРСИТЕТ ТЕЛЕКОМУНІКАЦІЙ**  
**НАВЧАЛЬНО-НАУКОВИЙ ІНСТИТУТ ІНФОРМАЦІЙНИХ**  
**ТЕХНОЛОГІЙ**  
**КАФЕДРА ІНЖЕНЕРІЇ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ**



## **РОЗРОБКА ПРОГРАМНИХ ЗАСОБІВ ДЛЯ ОЦІНКИ ЗНАТЬ** **МОВОЮ C#**

Виконав студент 4 курсу  
Групи ПД-43  
Борота Ілля Романович

Керівник роботи  
Старший викладач кафедри Державного університету телекомунікацій Коба Андрій Борисович

Київ – 2023

## МЕТА, ОБ'ЄКТ ТА ПРЕДМЕТ ДОСЛІДЖЕННЯ

- **Мета роботи** – спрощення процесу навчання та оцінювання студентів.
- **Об'єкт дослідження** – процес розробки програми-тренажеру на мові С#.
- **Предмет дослідження** – система обліку, допомоги у навчальному процесі.

2

## ЗАДАЧІ ДИПЛОМНОЇ РОБОТИ

1. Проаналізувати аналоги.
2. Розглянути інструменти розробки та вибрати найбільш оптимальні для розробки даної інформаційної системи.
3. Розробити та запровадити програмне забезпечення.
4. Провести тестування додатку.

3

## АНАЛІЗ АНАЛОГІВ

Функції	Multitester	Unitest	Richtest	INDIGO	QUIZ
Тестування	+	+	+	+	+
Перевірка	+	+	+	+	+
Діагностика	+	-	+	+	+
Автоматизація	-	+	-	+	+
Звітність	-	+	+	+	+
Сумісність	-	+	+	+	+

4

## ВИМОГИ ДО ДОДАТКУ

1. Реалізація алгоритму завантаження тестів.
2. Реалізація методу підключення до бази даних, пошук інформації.
3. Візуалізація результатів роботи програми у вигляді додатку.
4. Візуалізація рейтингу студентів.

5

## ПРОГРАМНІ ЗАСОБИ РЕАЛІЗАЦІЇ



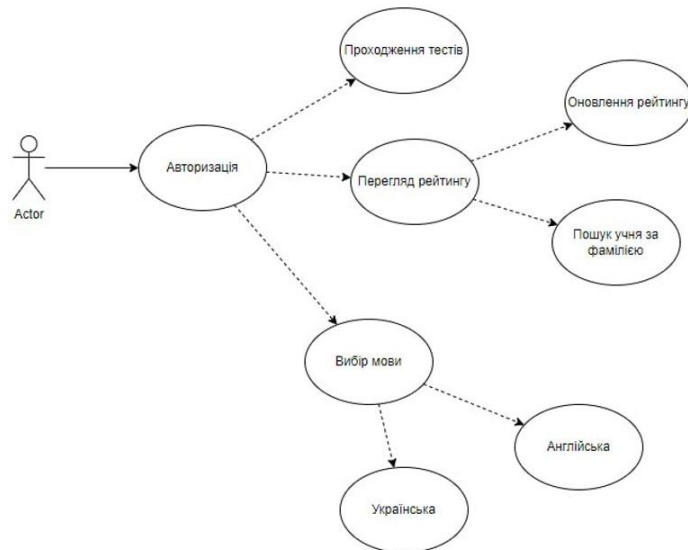
Microsoft Visual Studio



Мова програмування C#

6

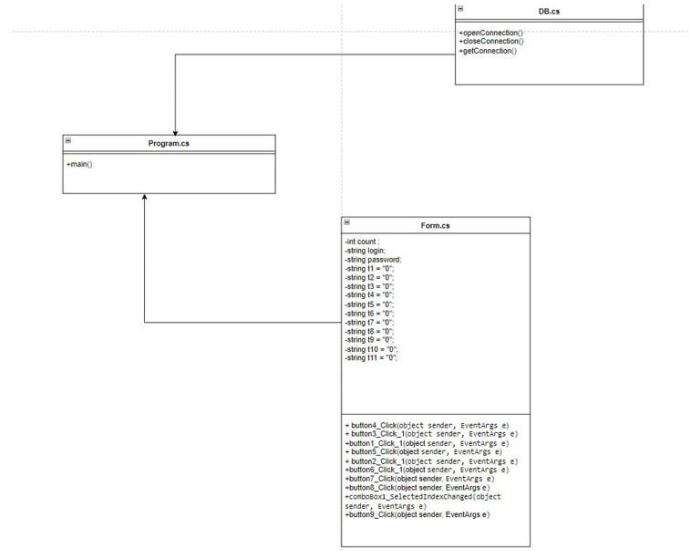
## ДІАГРАМИ ВАРІАНТІВ ВИКОРИСТАННЯ



7

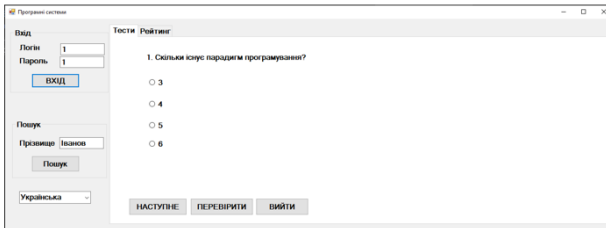


# ДІАГРАМА КЛАСІВ

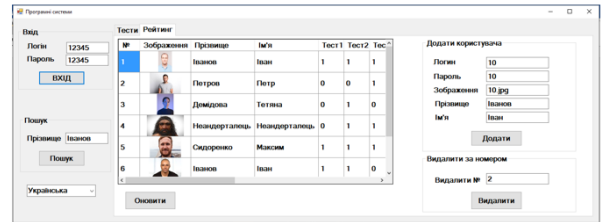


8

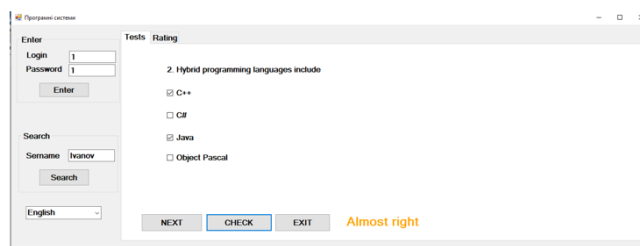
# ЕКРАННІ ФОРМИ



Вид програми звичайного користувача

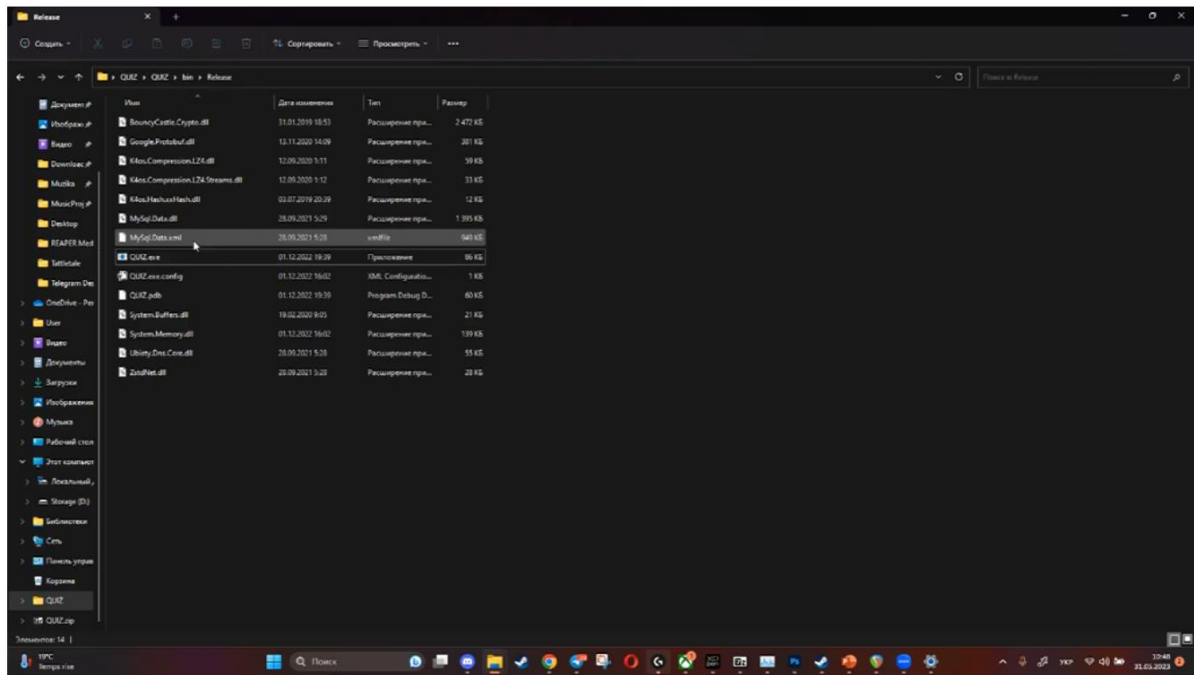


Вид програми адміністратора



Інтерфейс при проходженні тесту

9



10

## АПРОБАЦІЯ РЕЗУЛЬТАТІВ ДОСЛІДЖЕННЯ

1. Борота І.Р. РОЗРОБКА ПРОГРАМНИХ ЗАСОБІВ ДЛЯ ОБЛІКУ МАТЕРІАЛЬНО-ТЕХНІЧНОГО ЗАБЕЗПЕЧЕННЯ НАВЧАЛЬНОГО ПРОЦЕСУ МОВОЮ С# / Сучасні аспекти діджиталізації та інформатизації в програмній та комп'ютерній інженерії, 01- 03 червня 2023р., ДУТ, м. Київ - К: ДУТ, 2023. Подано до друку.

## ВИСНОВКИ

1. Опрацьовано методичні рекомендації до виконання дипломної роботи, проведено аналіз аналогів в наслідок якого виявилось, що програма “Quiz” має певні переваги серед конкурентів.
2. Виконано огляд літературних джерел, що використовується для вивчення даної теми, а також здійснено огляд існуючих програмних продуктів, які використовуються при вивченні різних дисциплін;
3. Розроблено програмне забезпечення для оцінки знань під назвою “Quiz”. Реалізовано підключення до бази даних та функціональність проходження тестів та зберігання рейтингу.
4. Проведене функціональне тестування продемонструвало відсутність помилок.