

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ  
ФАХОВИЙ КОЛЕДЖ РАКЕТНО-КОСМІЧНОГО МАШИНОБУДУВАННЯ  
ДНІПРОВСЬКОГО НАЦІОНАЛЬНОГО УНІВЕРСИТЕТУ ім. О. ГОНЧАРА

Циклова комісія програмної інженерії

**КУРСОВИЙ ПРОЕКТ**  
з навчальної дисципліни  
«ОБ'ЄКТНО-ОРІЄНТОВАНЕ ПРОГРАМУВАННЯ»

на тему: «Програма ведення обліку задач НМТ»

(вказати тему курсового проекту)

Студента IV курсу ПЗ-21-1 групи  
галузь знань 12 «Інформаційні технології»  
спеціальності 121 «Інженерія програмного  
забезпечення»

Гуненко Я.М.

(прізвище та ініціали студента)

Керівник

викладач Гапоненко Н.В.

Національна шкала \_\_\_\_\_

Кількість балів: \_\_\_\_\_ Оцінка ECTS: \_\_\_\_\_

Члени комісії \_\_\_\_\_

(підпис)

Любохинець В.М.

(прізвище та ініціали)

Ланська С.С.

(підпис)

(прізвище та ініціали)

Гапоненко Н.В.

(підпис)

(прізвище та ініціали)

м. Дніпро - 2024 рік

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ  
ФАХОВИЙ КОЛЕДЖ РАКЕТНО-КОСМІЧНОГО МАШИНОБУДУВАННЯ  
ДНІПРОВСЬКОГО НАЦІОНАЛЬНОГО УНІВЕРСИТЕТУ ім. О. ГОНЧАРА

Циклова комісія програмної інженерії

**ЗАТВЕРДЖУЮ**

Голова комісії ПІ

С.С.Ланська

«\_\_\_» \_\_\_\_\_ 2024 р.

**ЗАВДАННЯ**  
**на виконання курсового проекту**

з дисципліни Об'єктно-орієнтоване програмування  
студенту Гуненко Ярославу Максимовичу  
(прізвище, ім'я та по батькові)  
Відділення Комп'ютерної та програмної інженерії  
Спеціальність 121 Інженерія програмного забезпечення  
Курс IV Група (шифр) ПЗ-21-1  
1 Тема проекту «Програма ведення обліку здачі НМТ»

2 Початкові дані Перелік даних про учня (ID учня, Номер паспорту, Тип ,  
Паспорту, ПІБ, Дата народження, Стать, Е-mai, Номер телефону, Номер  
свідоства про освіту, ПІН, Примітка), перелік навчальних закладів(місто,  
область, тип, Е-mail, ПІБ відповідального, Шифр навчального закладу)  
перелік предметів(Код предмету, Назва предмету, Опис, Зразок завдань)  
термін дії, статус) перелік сертифікатів(Номер сертифікату, ID учня, PIN,  
Дата створення, Статус) перелік результатів тесту(Шифр результату,  
ID умови, Отриманий бал, Статус здачі, Дата проходження, Шифр Статус  
навчального закладу) Умови проходження(ID умови, Код предмету,  
Максимальний бал, Мінімальний бал, Мінімальний бал для проходження,  
предмету, Дата ухвалення умов)

Розглянуто і ухвалено на засіданні циклової комісії програмної інженерії  
Протокол № 2 від 21.09.2021 р.

Керівник КП

Н.В.Гапоненко

(підпис)

(ініціали та прізвище)

Завдання до виконання  
одержав студент

(підпис)

(ініціали та прізвище)

Дата видачі

21 вересня 2021 р.

Термін виконання

22 листопада 2021 р.

## ЗМІСТ

ВСТУП.....	4
1 ПОСТАНОВКА ЗАДАЧІ.....	5
2 ХАРАКТЕРИСТИКА ПРОГРАМНИХ ЗАСОБІВ РЕАЛІЗАЦІЇ ПРОЕКТУ .....	7
2.1 ОПИС СЕРЕДОВИЩА ПРОГРАМУВАННЯ .....	7
2.2 ОПИС МОВИ ПРОГРАМУВАННЯ .....	8
2.3 ОПИС СКБД .....	9
2.4 ОПИС ОСНОВНИХ ПРИНЦИПІВ ООП.....	10
2.5 ОПИС ПОДІБНИХ ПРОГРАМНИХ ПРОДУКТІВ .....	13
3 ПРОЕКТУВАННЯ ТА РЕАЛІЗАЦІЯ ПРОГРАМНОГО ПРОДУКТУ .....	14
3.1 ОПИС БАЗИ ДАНИХ.....	14
3.2 ПРОЕКТУВАННЯ КОРИСТУВАЦЬКОГО ІНТЕРФЕЙСУ ПРОГРАМИ .....	16
3.3 КОНТРОЛЬ ВХІДНИХ ДАНИХ ПРОГРАМИ .....	30
4 ІНСТРУКЦІЯ З КОРИСТУВАННЯ ПРОГРАМНОЮ СИСТЕМОЮ .....	34
ВИСНОВКИ.....	50
СПИСОК ВИКОРИСТАНИХ МАТЕРІАЛІВ.....	51

					КП.ПЗ.211.06.ПЗ				
Змн.	Лист	№ докум.	Підпис	Дата	Програма ведення обліку здачі НМТ	Літ.	Арк.	Аркушів	
Розроб.	Гуненко Я.М.						3	57	
Перевір.	Гапоненко Н.В.								
Реценз.						ВСП" ФКРКМ ДНУ" ім. Олеся Гончара			
Н. контр.									
Затверд.									

## ВСТУП

У сучасному світі точне зберігання даних та автоматизоване управління документацією потрібні скрізь. Цифрова епоха поширюється і на сферу освіти. Щороку в Україні учні, які закінчують 11-й клас, складають іспит, що відображає рівень знань, які вони здобули за цей час. Наразі, у зв'язку з воєнним станом, Зовнішнє незалежне оцінювання, скорочено ЗНО, замінено на Національний мультимедійний тест, скорочено НМТ.

Для центрів оцінювання доступно багато програмного забезпечення, але ці програми складні у вивченні та незручні у використанні. Більшість цих програм базуються на залежних від мережі інтернет WEB технологіях, що не завжди зручно. Програми повинні бути простими у використанні, швидкими, легкими в освоєнні інструментами та універсальними для центрів оцінювання будь-якого розміру. Програмне забезпечення повинно бути доступним для використання навіть недосвідченими користувачами комп'ютерів. Воно може використовуватися як однією особою (наприклад, у невеликому центрі оцінювання), так і групою осіб (наприклад, у великому центрі оцінювання з кількома посадами).

Основна мета курсового проекту - створити корисний і простий додаток для великих і малих центрів оцінювання.

	Вик.	Гуненко Я.М.			КП.ПЗ.211.06.ПЗ	Арк.
	Пер.	Гапоненко Н.В.				4
Змн.	Арк.	№ докум.	Підпис	Дата		

## 1 ПОСТАНОВКА ЗАДАЧІ

Тема курсового передбачає створення програмного забезпечення для обліку та зберігання інформації про осіб, які складали НМТ, їх результатів та створення сертифікатів та включає в себе такий функціонал:

### 1. Ведення обліку інформації про осіб які складали НМТ:

- Додавання інформації про учасника НМТ, шляхом введення типу паспорта, номеру паспорта, ПІН, номера свідоцтва про освіту, статі, дати народження, електронної пошти, контактного номеру телефону та необов'язкової помітки для екстрених випадків.

- Редагування інформації про учасника НМТ.

### 2. Ведення обліку інформації про умови тестування затверджені МОН:

- Додавання інформації про умови проведення тестування, шляхом введення статусу обов'язковості, дати укладання умов, прохідного балу, максимального можливого балу, мінімального можливого балу.

- Редагування інформації про умови проведення тестування.

### 3. Ведення обліку інформації про навчальні заклади для проведення НМТ:

- Додавання інформації про умови проведення тестування, шляхом введення шифру навчального закладу, міста, області, типу, E-mail, ПІБ відповідального за проведення.

- Редагування інформації про навчальний заклад.

- Видалення інформації про навчальний заклад.

### 4. Ведення обліку інформації про результати тестування:

- Додавання інформації про результати тестування, шляхом введення номеру паспорту учасника, навчального закладу, дати укладання умов тестування, назви навчальної дисципліни, результат тестування у балах, дата складання тестування.

- Редагування інформації про умови проведення тестування.

### 5. Ведення обліку інформації про навчальні дисципліни:

	Вик.	Гуненко Я.М.			КП.ПЗ.211.06.ПЗ	Арк.
	Пер.	Гапоненко Н.В.				5
Змн.	Арк.	№ докум.	Підпис	Дата		

- Додавання інформації про навчальні дисципліни, шляхом введення назви предмету, короткого опису предмету, зразка тестування, піктограми навчальної дисципліни.

- Редагування інформації про умови навчальні дисципліни.

- Видалення інформації про умови навчальні дисципліни

#### 6. Ведення обліку інформації про сертифікати НМТ:

- Додавання інформації про сертифікати НМТ, шляхом введення номеру паспорту учня, PIN-коду сертифікату, терміну дії сертифікату, статусу дії сертифікату, дати створення сертифікату, дати укладання умов тестування.

- Редагування інформації про сертифікати НМТ.

- Генерація сертифікатів в розширенні “\*.html” з можливістю встановлення шаблону для назви при першому запуску програми.

#### 7. Вимоги до операційної системи:

- Операційна система: Windows 10 x64.

- Процесор: Intel Celeron або еквівалент.

- Графічна підсистема: DirectX 10 і вище.

Таблиця 3.1 – Виділення інформаційних об’єктів предметної області

Особа	ID учня, номер паспорту, тип паспорту, ПІБ, дата народження, стать, E-mail, контактний номер телефону, Номер свідоцтва про освіту, примітка, ІПН
Навчальний заклад	Шифр навчального закладу, місто, область, тип, E-mail, ПІБ відповідального за проведення
Предмет	Код предмету, назва предмету, опис, зразок завдань, шлях до зображення предмету
Умови проходження тестування з предмету	ID умови, код предмету, максимальний бал, мінімальний бал, мінімальний бал для проходження, статус, дата укладання вимог
Результат тестування	Шифр результату, ID умови, ID учня, отриманий бал, статус здачі, дата проходження, шифр навчального закладу
Сертифікат учасника НМТ	Номер сертифікату, ID учня, PIN-код, дата створення, термін дії, статус дійсності

Вик.	Гуненко Я.М.				КП.ПЗ.211.06.ПЗ	Арк.
Пер.	Гапоненко Н.В.					6
Змн.	Арк.	№ докум.	Підпис	Дата		

## 2 ХАРАКТЕРИСТИКА ПРОГРАМНИХ ЗАСОБІВ РЕАЛІЗАЦІЇ ПРОЕКТУ

### 2.1 Опис середовища програмування

C++ Builder - це програмний продукт, інструмент швидкої розробки додатків (RAD), інтегроване середовище розробки (IDE) та система, що використовується програмістами для розробки програмного забезпечення мовою програмування C++. Visual C++ Builder дозволяє створювати графічні інтерфейси користувача шляхом перетягування компонентів з палітри інструментів на форми. За допомогою Builder можна створювати додатки для Windows, які використовують велику бібліотеку візуальних компонентів (VCL). C++ Builder автоматично генерує більшу частину коду програми, як тільки ви починаєте працювати над проектом. Для завершення решти логіки програми використовується текстовий редактор коду, який надає такі функції, як рефакторинг, паралельне редагування, завершення коду, збережені макроси натискання клавіш і власні комбінації клавіш. C++ Builder інтегрований в MSBuild як середовище збірки з командами збірки та компіляції, які викликають MSBuild; середовище RAD Studio надає мови програмування Delphi та C++ для розробки. Для курсових проектів корисним є C++ builder, оскільки він повністю підтримує мову програмування C++; середовище C++ builder підтримує найновіші стандарти C++, що дозволяє розробляти сучасне, ефективне програмне забезпечення. Вбудований компілятор ефективно оптимізує код і збільшує швидкість виконання програми.

Ще однією важливою особливістю цього середовища є підтримка високопродуктивних бібліотек: Visual Component Library (VCL) - це набір компонентів для побудови багатих графічних інтерфейсів користувача, VCL забезпечує швидкість і надійність. FireDAC - сучасна бібліотека доступу до даних, що підтримує широкий спектр баз даних; RTL (Runtime Library) - рутинна бібліотека часу виконання, що містить безліч корисних функцій для роботи з рядками, файлами, математичними операціями тощо.

	Вик.	Гуненко Я.М.			КП.ПЗ.211.06.ПЗ	Арк.
	Пер.	Гапоненко Н.В.				7
Змн.	Арк.	№ докум.	Підпис	Дата		



C++ Builder дозволяє створювати додатки для Windows, macOS, Android та iOS з єдиної кодової бази; додатки, створені за допомогою C++ Builder, використовують нативні компоненти платформи, що забезпечує високу продуктивність та інтеграцію з операційною системою.

## 2.2 Опис мови програмування

Для виконання курсового проєкту було обрано мову програмування C++ з кількох вагомих причин. По-перше, C++ – це універсальна мова загального призначення, яка дозволяє створювати високопродуктивні, ефективні та надійні програмні системи. Її потужність полягає у можливості низькорівневого доступу до пам'яті та апаратних ресурсів, що робить її ідеальним вибором для розробки критичних до продуктивності додатків.

Однією з ключових особливостей C++ є підтримка об'єктно-орієнтованого програмування. Ця парадигма дозволяє моделювати реальний світ за допомогою класів та об'єктів. Класи визначають структуру даних та поведінку об'єктів, а об'єкти є екземплярами цих класів. Такий підхід сприяє модульності, повторюваності коду та легкості підтримки великих програмних систем. Як зазначає Б'ярн Страуструп, творець C++, "ООП дозволяє розробникам створювати більш складні та надійні системи, ніж це було б можливо за допомогою процедурного програмування".

Шаблони (templates) – це ще одна потужна особливість C++, яка дозволяє створювати універсальні алгоритми та структури даних. За допомогою шаблонів можна писати код, який працює з різними типами даних, не дублюючи його. Це значно зменшує обсяг коду та підвищує його надійність. Як зазначається в стандарті C++ (ISO/IEC 14882:2011), "шаблони є фундаментальною частиною мови C++ і дозволяють розробникам писати більш загальний та ефективний код".

STL – це багата колекція контейнерів, алгоритмів та ітераторів, яка є невід'ємною частиною сучасного C++. STL забезпечує ефективні та зручні засоби для роботи з даними, що значно спрощує розробку складних програмних систем.

	Вик.	Гуненко Я.М.			КП.ПЗ.211.06.ПЗ	Арк.
	Пер.	Гапоненко Н.В.				
Змн.	Арк.	№ докум.	Підпис	Дата		8

Як зазначає Александре Степанов, один із розробників STL, "STL – це інструмент для створення ефективних та елегантних програм".

У курсовому проєкті мова програмування C++ дозволила структурувати програму за допомогою класів. Класи були створені для представлення різних доменних сутностей, що полегшує розуміння та підтримку коду. Для взаємодії з базою даних MySQL було використано функціонал C++, що забезпечує швидкий та надійний доступ до даних. Для роботи з динамічною пам'яттю використовувалися вказівники мови C++, низькорівневий доступ до пам'яті якої дозволив додатку працювати швидше.

### 2.3 Опис СКБД

Для виконання курсового проєкту було використано СУБД MySQL. MySQL – це одна з найпопулярніших у світі систем управління реляційними базами даних з відкритим кодом. Її широко застосовують для створення веб-додатків, корпоративних систем та в багатьох інших сферах. MySQL здатна обробляти великі обсяги даних і одночасно обслуговувати тисячі запитів. Це досягається завдяки оптимізації запитів, ефективним алгоритмам сортування та індексування. MySQL забезпечує високий рівень доступності даних завдяки підтримці транзакцій, реплікації та механізмам відновлення після збоїв. Вона підтримує різні типи таблиць і механізми зберігання даних, що дозволяє адаптувати систему до конкретних потреб проєкту. СУБД сумісна зі стандартом SQL, що значно спрощує роботу з даними та дозволяє використовувати широкий спектр інструментів і бібліотек.

Вирішальними факторами для вибору СУБД MySQL для курсового проєкту стали її технології та можливості. InnoDB – основний рушій зберігання даних у MySQL, який забезпечує високу швидкість роботи, підтримку транзакцій, цілісність даних і масштабованість. MyISAM – ще один рушій зберігання даних, оптимізований для швидкого читання. Реплікація дозволяє створювати копії бази даних на інших серверах для підвищення доступності й резервування. Партіціювання – це можливість розбивати великі таблиці на менші частини для

	Вик.	Гуненко Я.М.			КП.ПЗ.211.06.ПЗ	Арк.
	Пер.	Гапоненко Н.В.				9
Змн.	Арк.	№ докум.	Підпис	Дата		

ефективнішої обробки даних і покращення швидкодії. Тригери й збережені процедури автоматизують виконання певних дій при зміні даних у таблицях.

Для реалізації курсового проєкту було використано такі можливості MySQL: збережені процедури для автоматизації часто виконуваних завдань, тригери для автоматичного виконання дій у відповідь на зміну даних, представлення (віртуальні таблиці) для об'єднання даних з кількох таблиць, індекси для прискорення пошуку, транзакції для забезпечення цілісності даних, реплікація для підвищення доступності та резервування, а також кластеризація для масштабування бази даних.

## 2.4 Опис основних принципів ООП

Об'єктно-орієнтоване програмування (ООП) – це парадигма програмування, яка фокусується на створенні програм за допомогою об'єктів, що взаємодіють між собою. Основні принципи ООП – інкапсуляція, наслідування, поліморфізм та абстракція.

Інкапсуляція – це об'єднання даних і методів, що працюють з ними, в єдиний об'єкт. Це приховує внутрішню структуру об'єкта від зовнішнього світу, забезпечуючи безпеку даних та спрощуючи підтримку коду. В C++ інкапсуляція досягається за допомогою модифікаторів доступу: `private`, `protected` та `public`. Приклад фрагменту коду що демонструє інкапсуляцію продемонстровано у лістингу 2.1.

### Лістинг 2.1 – Фрагмент коду

```
class Neuron {
private:
    double output;
    std::vector<double> weights;
public:
    double calculateOutput(const std::vector<double>& inputs) {
        // Розрахунок виходу нейрона
        // ...
        return output;
    }
};
```

	Вик.	Гуненко Я.М.			КП.ПЗ.211.06.ПЗ	Арк.
	Пер.	Гапоненко Н.В.				10
Змн.	Арк.	№ докум.	Підпис	Дата		

Наслідування дозволяє створювати нові класи (похідні) на основі існуючих (базових), успадковуючи їхні властивості та методи. Це забезпечує повторне використання коду та створення ієрархій класів. Типи наслідування в C++: `public`, `protected` та `private`. При публічному наслідуванні публічні члени базового класу стають публічними членами похідного класу, а захищені – захищеними. Це означає, що об'єкти похідного класу можуть безпосередньо звертатися до цих членів. Публічне наслідування зазвичай використовується для створення ієрархій класів, де похідні класи є більш специфічними видами базового класу. При захищеному наслідуванні як публічні, так і захищені члени базового класу стають захищеними членами похідного класу. Це означає, що до них можуть звертатися лише члени похідного класу та інших похідних класів. Захищене наслідування часто використовується для створення ієрархій класів, де похідні класи можуть розширювати базовий клас, але приховувати деякі деталі реалізації від зовнішнього світу. При приватному наслідуванні публічні і захищені члени базового класу стають приватними членами похідного класу. Це означає, що до них можуть звертатися лише члени самого похідного класу. Приватне наслідування рідко використовується, оскільки обмежує можливості повторного використання коду. Зазвичай використовується для повного приховування реалізації базового класу і створення нового, більш абстрактного класу. Приклад фрагменту коду що демонструє наслідування продемонстровано у лістингу 2.2.

### Лістинг 2.2 – Фрагмент коду

```
class NeuralNetwork {
public: virtual void train(const
std::vector<std::pair<std::vector<double>, double>>& data) = 0;
};
class ConvolutionalNeuralNetwork : public NeuralNetwork { public:
void train(const std::vector<std::pair<std::vector<double>,
double>>& data) override {
    // Тренування згорткової нейронної мережі // ...
}
};
```

	Вик.	Гуненко Я.М.			КП.ПЗ.211.06.ПЗ	Арк.
	Пер.	Гапоненко Н.В.				11
Змн.	Арк.	№ докум.	Підпис	Дата		

Поліморфізм – це здатність об'єктів різних типів відповідати на один і той самий запит по-різному. Це досягається за допомогою перевантаження функцій та віртуальних функцій. Поліморфізм забезпечує гнучкість та розширюваність програм. Приклад фрагменту коду що демонструє поліморфізм продемонстровано у лістингу 2.3.

#### Лістинг 2.3 – Фрагмент коду

```
class ActivationFunction {
public:
    virtual double activate(double z) = 0;
};

class Sigmoid : public ActivationFunction {
public:
    double activate(double z) override {
        // Функція сигмоїди
        // ...
    }
};

class ReLU : public ActivationFunction {
public:
    double activate(double z) override {
        // Функція ReLU
        // ...
    }
};
```

Абстракція – це процес фокусування на суттєвих характеристиках об'єкта, приховуючи деталі реалізації. Абстрактні класи та чисті віртуальні функції в C++ дозволяють створювати абстрактні інтерфейси, які не можуть бути інстанційовані безпосередньо. Приклад фрагменту коду що демонструє абстракцію продемонстровано у лістингу 2.4.

#### Лістинг 2.4 – Фрагмент коду

```
class Layer {
public:
    virtual void forward(const std::vector<double>& input) = 0;
    virtual void backward(const std::vector<double>& output_error) =
0;
};
```

	Вик.	Гуменко Я.М.			КП.ПЗ.211.06.ПЗ	Арк.
	Пер.	Гапоненко Н.В.				12
Змн.	Арк.	№ докум.	Підпис	Дата		

## 2.5 Опис подібних програмних продуктів

Схожою за функціоналом є система [my.testportal.gov.ua](http://my.testportal.gov.ua).

[my.testportal.gov.ua](http://my.testportal.gov.ua) — це веб-портал, призначений для організації та обліку результатів національних тестів. Система працює виключно онлайн, що означає, що будь-яка перерва в доступі до інтернету може спричинити затримку в роботі, оскільки програма не підтримує роботу в автономному режимі.

Платформа використовує власну базу даних для збереження інформації про учнів, навчальні заклади та результати тестувань. Проте, для більш гнучкої обробки даних немає можливості прямої інтеграції з СКБД на стороні користувача. Дані можна експортувати у форматі PDF, але це обмежує можливості подальшого аналізу та структуризації інформації.

Платформа також вимагає значного часу на налаштування, особливо для нових користувачів або навчальних закладів. Потрібен спеціаліст для розподілу ролей та доступів між адміністраторами, вчителями та учнями, оскільки самостійно це зробити складно без відповідного досвіду.

Висновок: [my.testportal.gov.ua](http://my.testportal.gov.ua) підходить для великих організацій, що мають відповідні ресурси для підтримки та адміністрування системи. Однак, для невеликих закладів або окремих користувачів платформа може виявитися занадто складною та вимагати значних зусиль для ефективного використання.

	Вик.	Гуненко Я.М.			КП.ПЗ.211.06.ПЗ	Арк.
	Пер.	Гапоненко Н.В.				13
Змн.	Арк.	№ докум.	Підпис	Дата		

## 3 ПРОЕКТУВАННЯ ТА РЕАЛІЗАЦІЯ ПРОГРАМНОГО ПРОДУКТУ

### 3.1 Опис бази даних

Для збереження великих об'ємів інформації потрібно розробити низку правил та відношень між цими даними. Відношення між таблицями показані на ER-діаграмі. Скрипт створення БД показаний у додатку Б. Зміст таблиць бази даних показаний у таблицях 3.2-3.8.

Таблиця 3.2 – Поля таблиці «Студент» (Student)

Назва поля	Опис	Тип	Розмір	Ключ
Student_id	Унікальний ідентифікатор студента	Числовий		Первинний
Passport_num	Номер паспорта	Символьний	До 20 символів	
Passport_type	Тип паспорта	Символьний	До 10 символів	
PIB	Прізвище, ім'я, по-батькові	Символьний	До 100 символів	
Birth_date	Дата народження	DATE	Стандарт ISO	
Gender	Стать	Символьний	1 символ	
E-mail	Електронна пошта	Символьний	До 100 символів	
Phone_num	Номер телефону	Символьний	До 15 символів	
EduCerf_num	Номер освітнього сертифіката	Символьний	До 30 символів	
PN	Ідентифікаційний код	Символьний	10 символів	
Additional	Додаткова інформація	Текстовий		

Таблиця 3.3 – Поля таблиці «Користувачі» (Users)

Назва поля	Опис	Тип	Розмір	Ключ
Login	Логін користувача	Символьний	До 50 символів	Первинний
Password	Пароль користувача	Символьний	До 50 символів	
Role	Роль користувача	Символьний	До 30 символів	

Таблиця 3.4 – Поля таблиці «Результат» (Result)

Назва поля	Опис	Тип	Розмір	Ключ
Res_id	Унікальний код результату	Числовий		Первинний
Subj_id	Код предмета	Числовий		Зовнішній
Condition_id	Код умов	Числовий		Зовнішній
Student_id	Код студента	Числовий		Зовнішній
Reached_score	Досягнутий бал	Числовий		
Status	Статус	Tinyint(1)	1 символ	
School_id	Код навчального закладу	Числовий		Зовнішній
Attemp_date	Дата спроби	DATE	Стандарт ISO	

Таблиця 3.5 – Поля таблиці «Сертифікат» (Certificate)

Назва поля	Опис	Тип	Розмір	Ключ
Cerf_num	Номер сертифіката	Числовий		Первинний
Student_id	ID студента	Числовий		Зовнішній
PIN	PIN-код	Символьний	До 15 символів	
Creation_date	Дата створення	DATE	Стандарт ISO	
Efect_time	Термін дії	DATE	Стандарт ISO	
Status	Статус сертифіката	Tinyint(1)		

Таблиця 3.6 – Поля таблиці «Умови тестування» (Condition)

Назва поля	Опис	Тип	Розмір	Ключ
Condition_id	Унікальний код умови	Числовий		Первинний
Subject_id	Код предмета	Числовий		Зовнішній
Max_point	Максимальний бал	Числовий		
Min_r_point	Мінімальний бал для проходження	Числовий		
Min_point	Мінімальний бал	Числовий		
Status	Обов'язкова умова	Tinyint(1)		
Date	Дата	DATE	Стандарт ISO	

Таблиця 3.7 – Поля таблиці «Навчальний заклад» ()

Назва поля	Опис	Тип	Розмір	Ключ
School_id	Унікальний код навчального закладу	Числовий		Первинний
City	Місто	Символьний	До 50 символів	
Region	Область	Символьний	До 50 символів	
Type	Тип навчального закладу	Символьний	До 30 символів	
E-mail	Електронна пошта	Символьний	До 100 символів	
Teacher_PIB	Прізвище, ім'я, по-батькові відповідального	Символьний	До 100 символів	

Таблиця 3.8– Поля таблиці «Предмети» (Subject)

Назва поля	Опис	Тип	Розмір	Ключ
Subject_id	Унікальний код предмета	Числовий		Первинний
Name	Назва предмета	Символьний	До 100 символів	
Description	Опис предмета	Текстовий		
Image_name	Шлях до піктограми завдань	Символьний	До 255 символів	
Test_sample	Шлях до прикладу завдань	Символьний	До 255 символів	

ER-діаграма зв'язку таблиць бази даних наведена нижче на рисунку 3.1.

	Вик.	Гуненко Я.М.			КП.ПЗ.211.06.ПЗ	Арк.
	Пер.	Гапоненко Н.В.				15
Змн.	Арк.	№ докум.	Підпис	Дата		



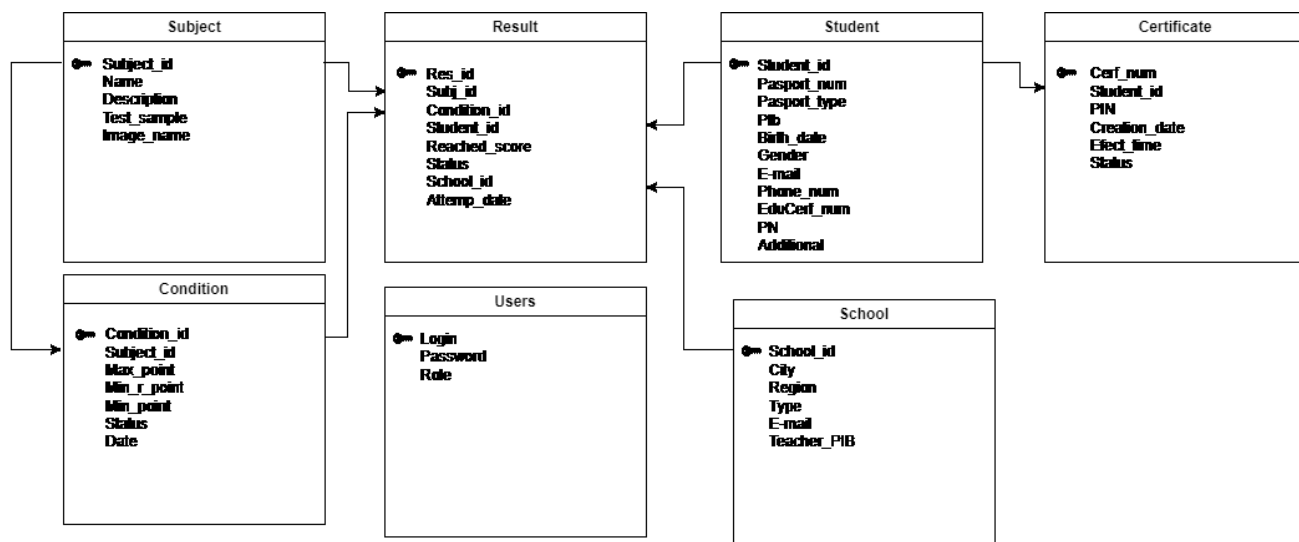


Рисунок 3.1 – ER діаграма бази даних

### 3.2 Проектування користувацького інтерфейсу програми

Для такого курсового проекту потрібно грамотно спроектувати взаємодію та ієрархію класів. Головною задачею проектування курсового проекту було створення логічних та програмних зв'язків між компонентами системи. Кожна підсистема виконує свою роль максимально абстраговано від інших та паралельно з ними.

Виходячи з цього можна спроектувати зручний інтерфейс. Його було створено саме під написані класи зв'язків з БД. Створено налаштування інтерфейсу для вдоволення потреб більшості користувачів (розмір інтерфейсу та кольорова палітра).

При проектуванні додатку першим вікном, що користувач бул обрано вікно створення шаблону назви сертифікату зі збереженням у ini файлі у папці запуску проекту. Для роботи цієї функції використано бібліотеку "IniFiles.cpp". Фрагмент коду для створення ini файлу продемонстровано у лівстингу 3.1.

#### Лістинг 3.1 – Фрагмент коду для створення ini файлу

```

void __fastcall TForm13::SaveTemplateToIni()
{
    TIniFile *IniFile = new TIniFile(ExtractFilePath(Application->ExeName) + "config.ini");
}
  
```

Вик.	Гуненко Я.М.				КП.ПЗ.211.06.ПЗ	Арк.
Пер.	Гапоненко Н.В.					16
Змн.	Арк.	№ докум.	Підпис	Дата		

```

try {
    IniFile->WriteString("Certificate", "TemplateName",
template_name);
}
__finally {
    delete IniFile;
}
}

```

Для введення шаблону користувачу запропоновано об'єкт класу LabelEdit та кнопку для запуску створення ini файлу. Для перевірки введення користувача використано обробник подій OnExit для LabelEdit. Зовнішній вигляд вікна продемонстровано на рисунку 3.1.

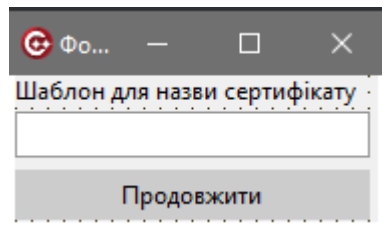


Рисунок 3.1 – Вікно введення шаблону для назви сертифікату

Після створення шаблону назви сертифікату користувачу програмного додатку буде відображено головне вікно програмного додатку. Інтерфейс головного вікна продемонстровано на рисунку 3.2.

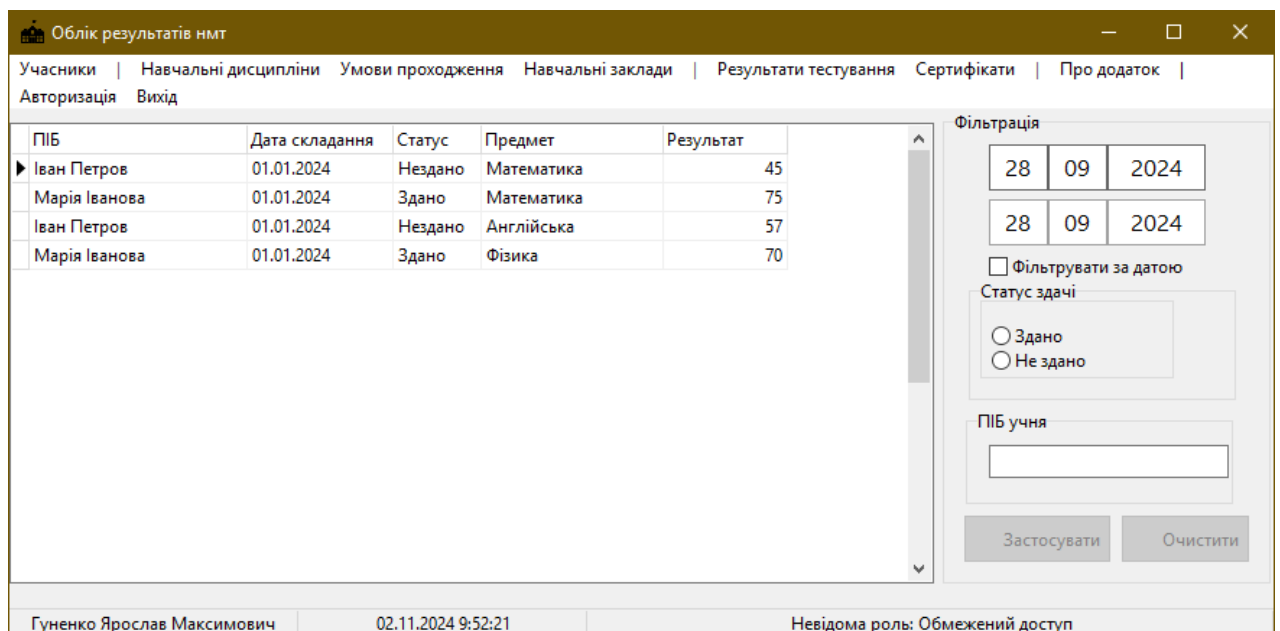


Рисунок 3.2 – Головне вікно програмного додатку

На головному вікні створено DBGrid, у якому відображено стислу інформацію про результати тестування. Для зручної роботи з даними у таблиці створено сортування даних за стовпцями за допомогою обробника OnTitleClick. Лістинг фрагменту коду для сортування продемонстровано у лістингу 3.2.

### Лістинг 3.2 – Фрагмент коду для сортування записів у таблиці

```
void __fastcall TForm3::DBGrid1TitleClick(TColumn *Column)
{
    String columnName = Column->FieldName;
    static bool sortAsc = true;
    String sortOrder = sortAsc ? "ASC" : "DESC";
    sortAsc = !sortAsc;

    String query =
        "SELECT s.PIB, "
        "r.Attemp_date, "
        "CASE WHEN r.Status = 1 THEN 'Здано' ELSE 'Нездано' END AS
Status, "
        "subj.Name, "
        "r.Reached_score "
        "FROM Result r "
        "JOIN Student s ON r.Student_id = s.Student_id "
        "JOIN Subject subj ON r.Subj_id = subj.Subject_id "
        "ORDER BY " + columnName + " " + sortOrder;

    try
    {
        DataModule1->MainQuery->Close();
        DataModule1->MainQuery->SQL->Clear();
        DataModule1->MainQuery->SQL->Add(query);
        DataModule1->MainQuery->Open();
        DBColumnSizes();
    }
    catch (Exception &e)
    {
        ShowMessage("Помилка сортування: " + e.Message);
    }
}
```

Також для роботи з великою кількістю записів створено функціонал для фільтрації записів, винесений в окремий GroupBox з можливістю комбінації фільтрів. За замовчуванням кнопки для запуску фільтрації та її відміни заблоковані для користувача. Після вибору будь-якого пункту у RadioGroup “Статус” або встановлення CheckBox “Фільтрувати за датою” чи введення піб

	Вик.	Гуненко Я.М.			КП.ПЗ.211.06.ПЗ	Арк.
	Пер.	Гапоненко Н.В.				
Змн.	Арк.	№ докум.	Підпис	Дата		18

учня кнопки активуються. Для фільтрації за датою використано об'єкти DatePicker.

Для зручної навігації використано MainMenu. Інтерфейс головного меню продемонстровано на рисунку 3.3. Лістинг фрагменту коду для запуску іншого вікна на обробник OnClick для пунктів меню продемонстровано у лістингу 3.3.

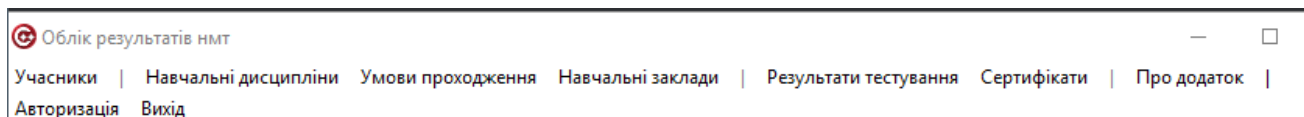


Рисунок 3.3 – Інтерфейс головного меню

Лістинг 3.3 – Запуск вікна по кліку на пункт головного меню

```
void __fastcall TForm3::N2Click(TObject *Sender)
{
    Form14->ShowModal();
}
```

Додатково на формі передбачено StatusBar, що відображає активну роль користувача з коротким описом задач до яких він має доступ та активну дату та час. Для розмежовування доступу користувачів створено вікно авторизації що містить введення логіну і паролю, що видається відповідальному за свою частину інформації. Інтерфейс вікна авторизації продемонстровано на рисунку 3.4.

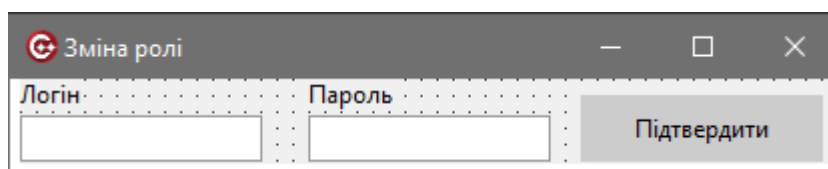


Рисунок 3.4 – Вікно авторизації

Після успішної авторизації значення у StatusBar змінюється на назву авторизованої ролі. У випадку неуспішної авторизації користувач отримає роль “Обмежений доступ”. Щоб змінити активну роль або просто вийти з активної ролі розроблено підпункт “Вийти” пункту меню “Авторизація”.

Для роботи з інформацією про учасників НМТ створено форму “Робота з інформацією про учасників НМТ”. Інтерфейс форми продемонстровано на рисунк 3.5.

	Вик.	Гуненко Я.М.			КП.ПЗ.211.06.ПЗ	Арк.
	Пер.	Гапоненко Н.В.				19
Змн.	Арк.	№ докум.	Підпис	Дата		

Робота з інформацією про учасників НМТ

Головне меню Про додаток

Student_id	Passport_num	Passport_type	PIB
1	BC6543219	ID-карта	Іван Петров
2	BC654321	Паперовий	Марія Іванова
3	BC654329	Паперовий	Петренко Пітрої
4	UA325419	ID-карта	Назаренко Інна
5	UA12345678	ID-карта	Зеленський Володимир
6	BC654322	Паперовий	AAAA
7	BC654325	Паперовий	Макаров Антон
8	BC654254	ID-карта	Микола

Фільтрація

Тип паспорту

Стать ☐ Чоловіча ☐ Жіноча

☐ Фільтрувати за датою народження

28 09 2024 E-mail

28 09 2024

☒ Застосувати ☐ Очистити

Рисунок 3.5 – Форма “Робота з інформацією про учасників НМТ”

Також для роботи з великою кількістю записів створено функціонал для фільтрації записів, винесений в окремий GroupBox з можливістю комбінації фільтрів. За замовчуванням кнопки для запуску фільтрації та її відміни заблоковані для користувача. Після вибору будь-якого пункту у RadioGroup “Стать” або встановлення CheckBox “Фільтрувати за датою народження” чи введення E-mail або вибору типу паспорту з ComboBox кнопки активуються. Для фільтрації за датою використано об’єкти DatePicker. Також для сортування використано обробник події OnTitleClick для таблиці. Для додавання або редагування даних створено окрему форму та PopUpMenu для вибору над записом, яке прив’язане до об’єкту DBGrid. Інтерфейс вікна роботи з даними продемонстровано на рисунку 3.6. Інтерфейс контекстного меню продемонстровано на рисунку 3.7.

Вик.	Гуменко Я.М.				КП.ПЗ.211.06.ПЗ	Арк.
Пер.	Гапоненко Н.В.					20
Змн.	Арк.	№ докум.	Підпис	Дата		

Рисунок 3.6 – Інтерфейс вікна для додавання та редагування записів

Номер паспорту	Тип паспорту	ПІБ
BC654321		
BC654321		
BC654329		

Рисунок 3.7 – Контекстне меню для DBGrid з роллю “Обмежений доступ”

Робота з інформацією про учасників НМТ		
Головне меню Про додаток		
Номер паспорту	Тип паспорту	ПІБ
BC654321		Іван Петров
BC65432		Марія Іванов
BC65432		Петренко Піт

Рисунок 3.8 – Контекстне меню для DBGrid з роллю “Менеджер учнів”

Для роботи з інформацією про навчальні дисципліни створено форму “Робота з інформацією про предмети”. Інтерфейс форми продемонстровано на рисунку 3.9. Для виведення зображення використано TImage.

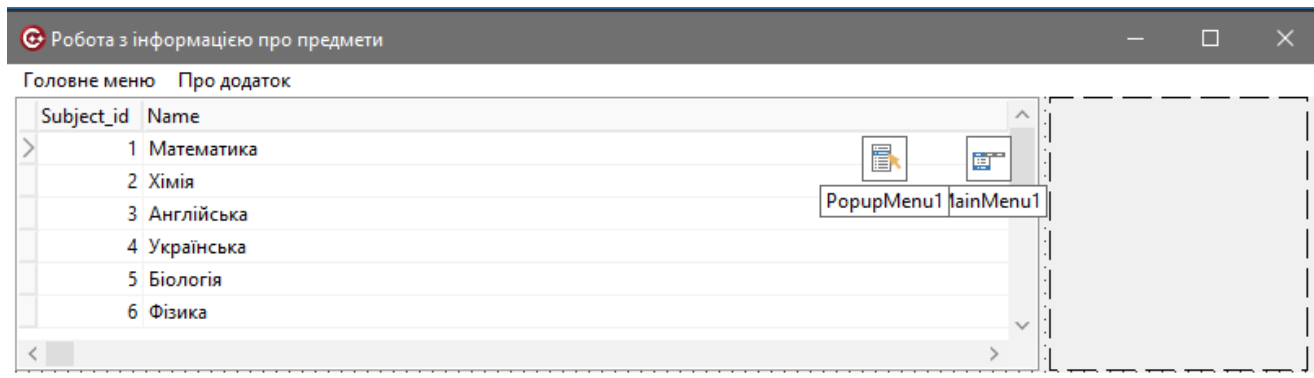


Рисунок 3.9 – Форма “Робота з інформацією про предмети”

Для додавання та редагування або видалення даних створено окрему форму та PopUpMenu для вибору над записом, яке прив'язане до об'єкту DBGrid. Інтерфейс вікна роботи з даними продемонстровано на рисунку 3.10. На формі використано OpenPictureDialog1 та OpenTextFileDialog1 для додання зразків завдань та піктограм предметів. Також за замовчування доступний пункт контекстного меню “Подивитися зразки”. Інтерфейс контекстного меню продемонстровано на рисунку 3.11.

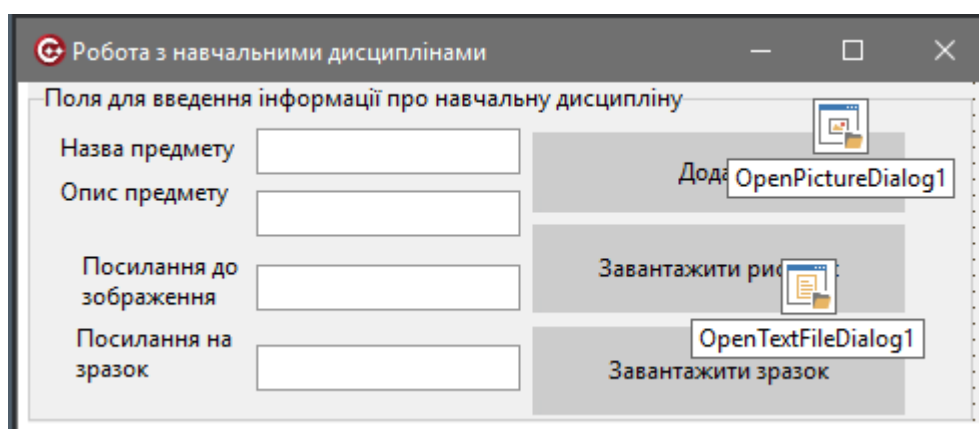


Рисунок 3.10 – Інтерфейс вікна для додавання та редагування записів

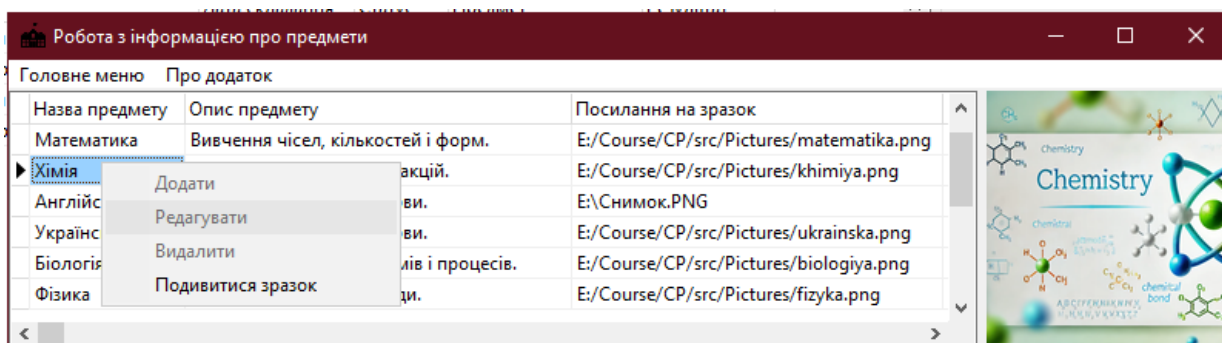


Рисунок 3.11 – Контекстне меню для DBGrid з роллю “Обмежений доступ”

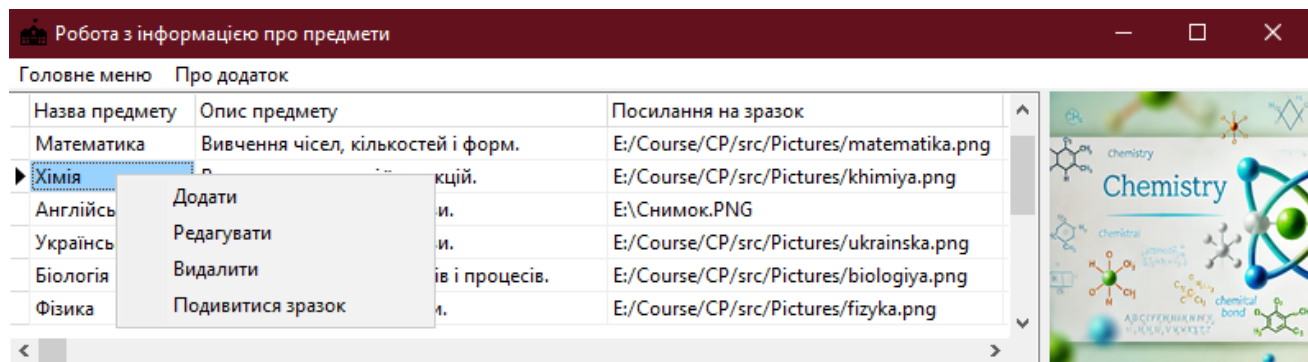


Рисунок 3.12 – Контекстне меню для DBGrid з роллю “Менеджер предметів”

Для роботи з інформацією про навчальні дисципліни створено форму “Робота з інформацією про умови проходження НМТ”. Інтерфейс форми продемонстровано на рисунку 3.13.

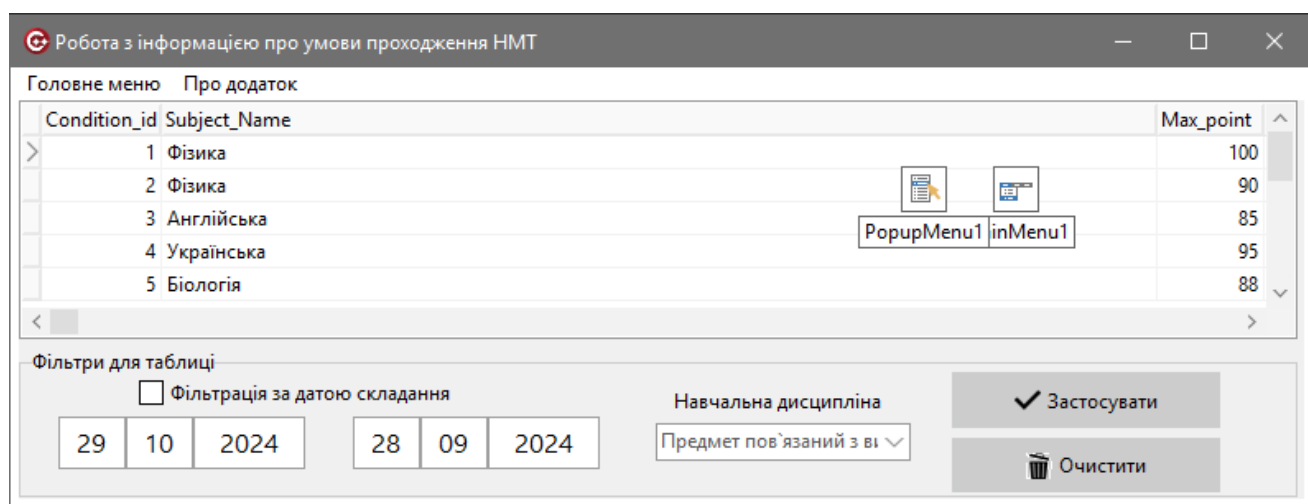


Рисунок 3.13 – Форма “Робота з інформацією про предмети”

Також для роботи з великою кількістю записів створено функціонал для фільтрації записів, винесений в окремий GroupBox з можливістю комбінації фільтрів. За замовчуванням кнопки для запуску фільтрації та її відміни заблоковані для користувача. Після вибору будь-якої дисципліни у ComboBox “Навчальні дисципліни” або встановлення CheckBox “Фільтрувати за датою складання” кнопки активуються. Для фільтрації за датою використано об’єкти DatePicker. Також для сортування використано обробник події OnTitleClick для таблиці.



Для додавання або редагування даних створено окрему форму та PopUpMenu для вибору над записом, яке прив'язане до об'єкту DBGrid. Інтерфейс вікна роботи з даними продемонстровано на рисунку 3.14. Інтерфейс контекстного меню продемонстровано на рисунку 3.15.

Рисунок 3.14 – Інтерфейс вікна для додавання та редагування записів

Рисунок 3.15 – Контекстне меню для DBGrid з роллю “Обмежений доступ”

Рисунок 3.16 – Контекстне меню для DBGrid з роллю “Менеджер предметів”

Для роботи з інформацією про навчальні дисципліни створено форму “Робота з навчальними закладами”. Інтерфейс форми продемонстровано на рисунку 3.17.

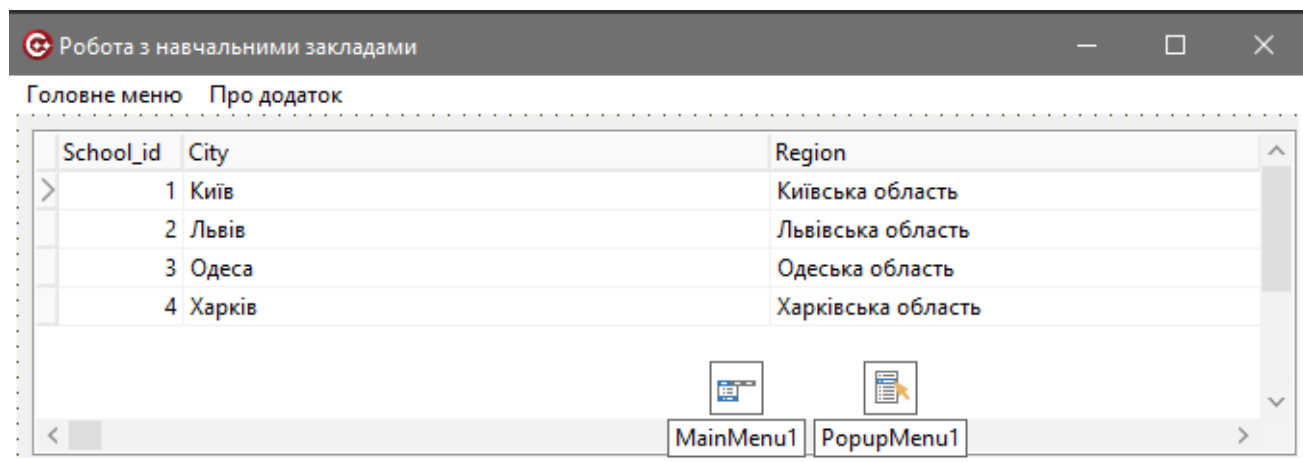


Рисунок 3.17 – Форма “Робота з навчальними закладами ”

Для сортування використано обробник події OnTitleClick для таблиці.

Для додавання або редагування даних, а також для видалення даних створено окрему форму та PopUpMenu для вибору над записом, яке прив’язане до об’єкту DBGrid. Інтерфейс вікна роботи з даними продемонстровано на рисунку 3.18. Інтерфейс контекстного меню продемонстровано на рисунку 3.19.

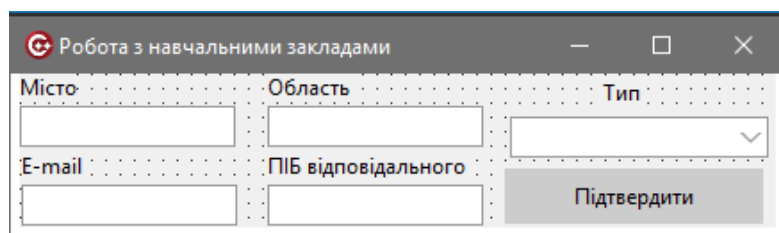


Рисунок 3.18 – Інтерфейс вікна для додавання та редагування записів

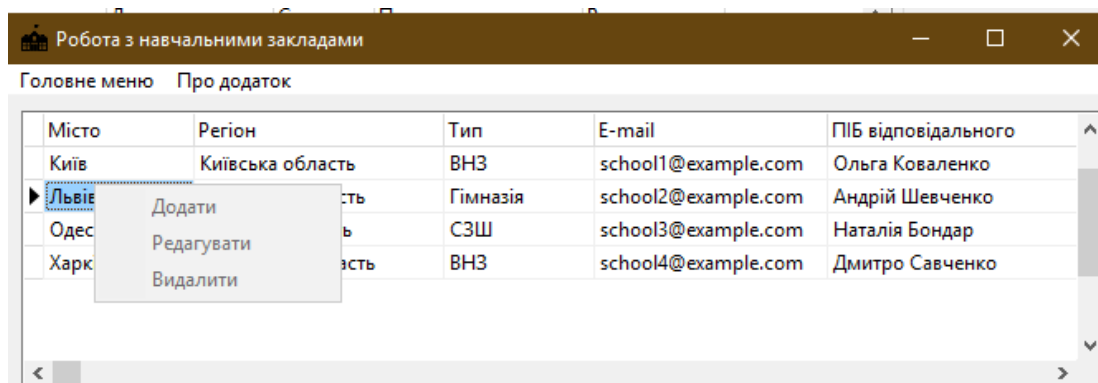


Рисунок 3.19 – Контекстне меню для DBGrid з роллю “Обмежений доступ”

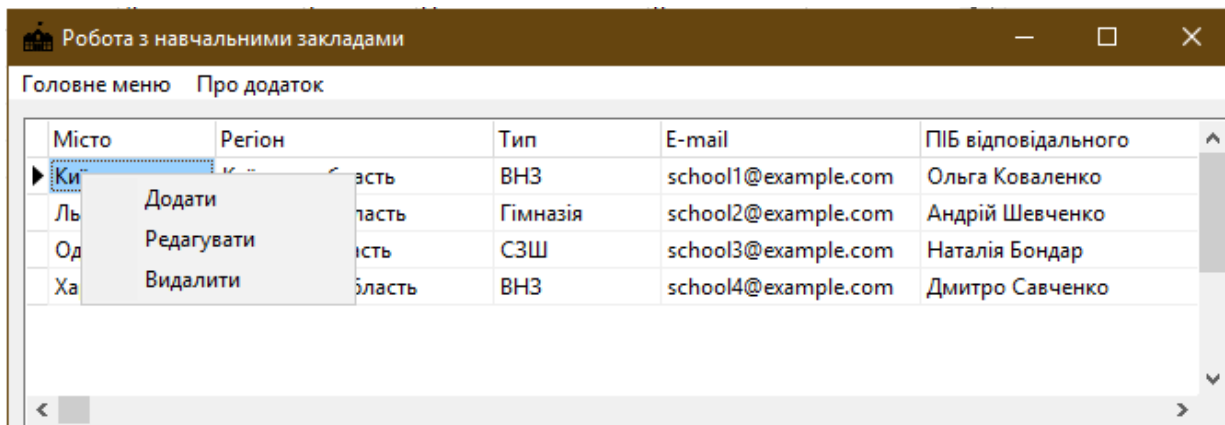


Рисунок 3.20 – Контекстне меню для DBGrid з роллю “Менеджер предметів”

Для роботи з інформацією про результати тестування створено форму “Робота з інформацією про результати тестування”. Інтерфейс форми продемонстровано на рисунку 3.21.

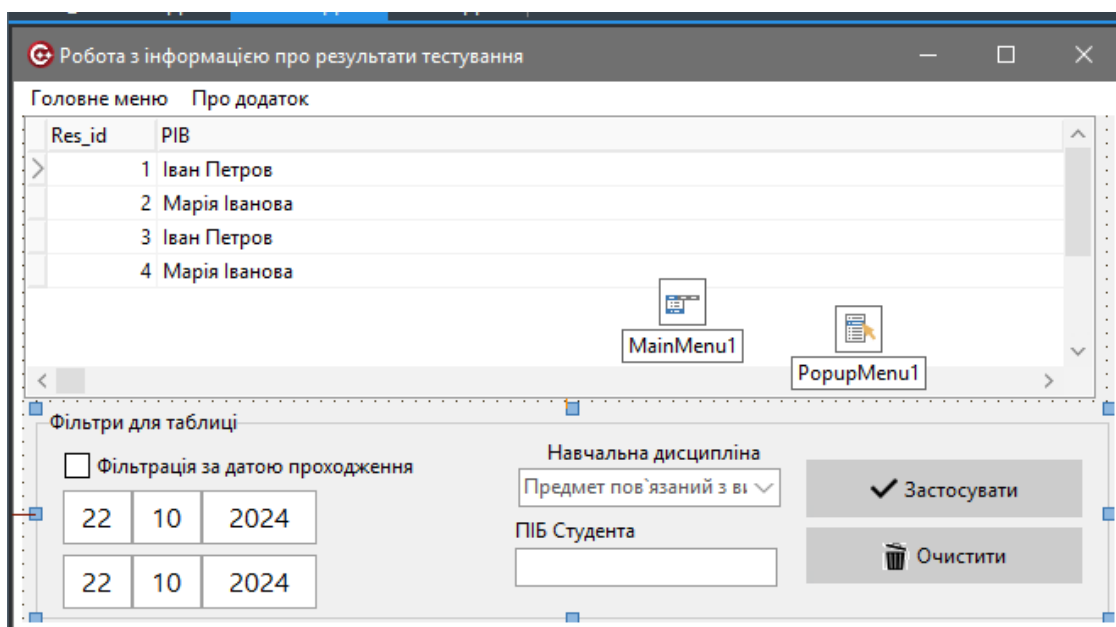


Рисунок 3.21 – Форма “Робота з інформацією про результати тестування”

Також для роботи з великою кількістю записів створено функціонал для фільтрації записів, винесений в окремий GroupBox з можливістю комбінації фільтрів. За замовчуванням кнопки для запуску фільтрації та її відміни заблоковані для користувача. Після вибору будь-якої дисципліни у ComboBox “Навчальні дисципліни” або встановлення CheckBox “Фільтрувати за датою складання” або введення ПІБ студента кнопки активуються. Для фільтрації за

датою використано об'єкти DatePicker. Також для сортування використано обробник події OnTitleClick для таблиці.

Для додавання або редагування даних створено окрему форму та PopUpMenu для вибору над записом, яке прив'язане до об'єкту DBGrid. Інтерфейс вікна роботи з даними продемонстровано на рисунку 3.22. Інтерфейс контекстного меню продемонстровано на рисунку 3.23.

Рисунок 3.22 – Інтерфейс вікна для додавання та редагування записів

Рисунок 3.23 – Контекстне меню для DBGrid з роллю “Обмежений доступ”

ПІБ учасника	Предмет	Результат	Максимально	Прохідний	Статус	Дата
Іван Петров	Математика	45	100	60	Не зараховано	01.0
Марія Іван		75	90	75	Зараховано	01.0
Іван Петро		57	85	65	Не зараховано	01.0
Марія Іван		70	90	75	Зараховано	01.0

Фільтри для таблиці

☐ Фільтрація за датою проходження

02 11 2024

22 10 2024

Навчальна дисципліна

Предмет пов'язаний з ві

ПІБ Студента

Застосувати

Очистити

Рисунок 3.24 – Контекстне меню для DBGrid з роллю “Менеджер результатів”

Для роботи з інформацією про результати тестування створено форму “Робота з інформацією про сертифікати НМТ”. Інтерфейс форми продемонстровано на рисунку 3.25.

Cerf_num	PIB	PIN
1	Іван Петров	12345
2	Марія Іванова	78901
3	Макаров Антон	12345

Робота з сертифікатами

☐ Фільтрація за датою створення

ПІН код сертифікату

Згенерувати

23 10 2024

23 10 2024

Застосувати

Очистити

Рисунок 3.25 – Форма “Робота з інформацією про сертифікати НМТ ”

Також для роботи з великою кількістю записів створено функціонал для фільтрації записів, винесений в окремий GroupBox з можливістю комбінації фільтрів. За замовчуванням кнопки для запуску фільтрації та її відміни

заблоковані для користувача. Після встановлення CheckBox “Фільтрувати за датою складання” кнопки активуються. Для фільтрації за датою використано об’єкти DatePicker. Для збереження сертифікату використано компонент SaveDialog. Також для сортування використано обробник події OnTitleClick для таблиці.

Для додавання або редагування даних створено окрему форму та PopUpMenu для вибору над записом, яке прив’язане до об’єкту DBGrid. Інтерфейс вікна роботи з даними продемонстровано на рисунку 3.26. Інтерфейс контекстного меню продемонстровано на рисунку 3.27.

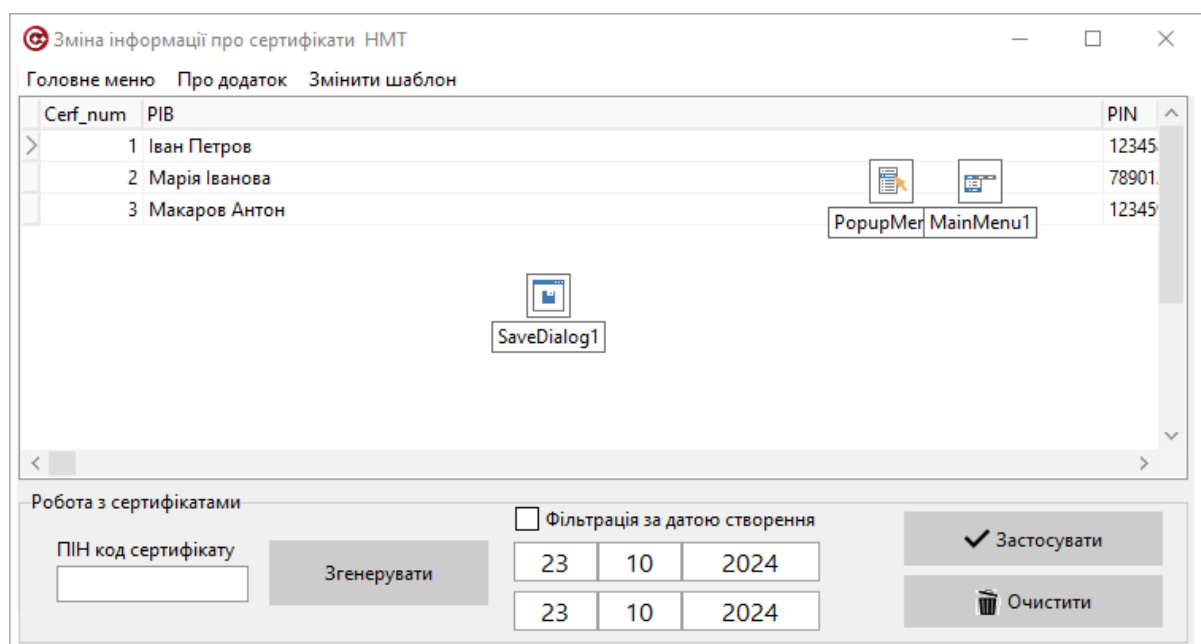


Рисунок 3.26 – Інтерфейс вікна для додавання та редагування записів

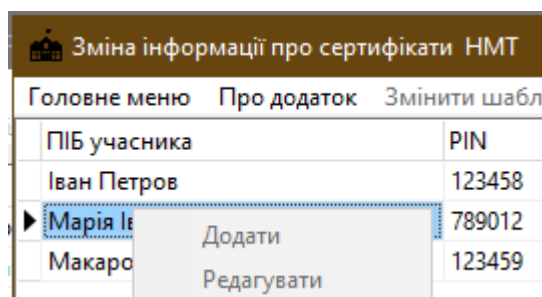


Рисунок 3.27 – Контекстне меню для DBGrid з роллю “Обмежений доступ”

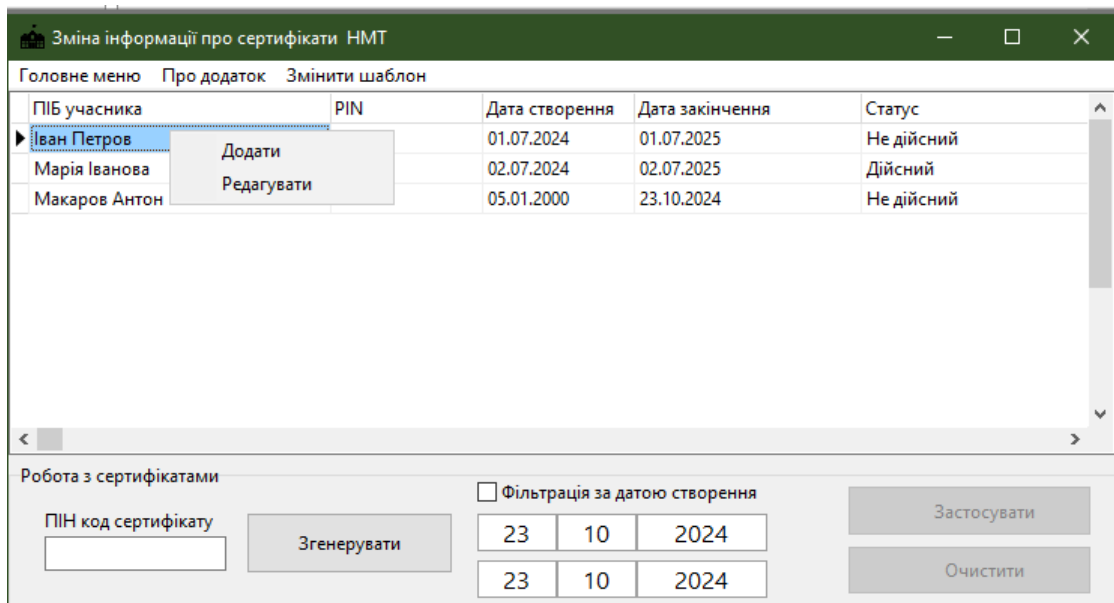


Рисунок 3.28 – Контекстне меню для DBGrid з роллю “Менеджер результатів”

Додатково реалізовано вікно “Про додаток”, що містить зображення TImage. Інтерфейс вікна продемонстровано на рисунку 3.29.

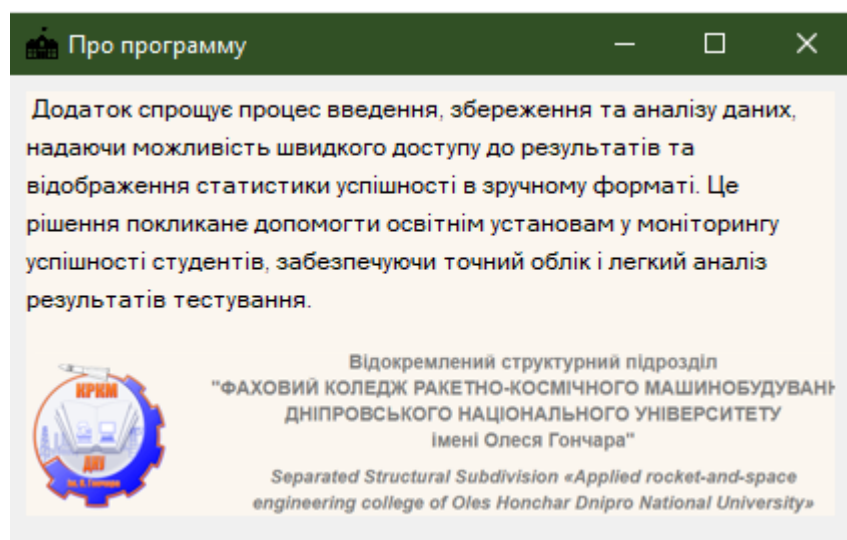


Рисунок 3.28 – Форма “Про додаток”

### 3.3 Контроль вхідних даних програми

У програмі для контролю вхідних даних було використано регулярні вираження класу `std::regex`, приклади регулярного виразів та створення валідаторів на введення даних продемонстровано у лістингах 3.1–3.2.

Вик.	Гуненко Я.М.				КП.ПЗ.211.06.ПЗ	Арк.
Пер.	Гапоненко Н.В.					30
Змн.	Арк.	№ докум.	Підпис	Дата		

### Лістинг 3.4 – Приклад регулярного виразу

```
void __fastcall TForm11::LabeledEdit1Exit(TObject *Sender)
{
    String pib = LabeledEdit1->Text;
    UnicodeString pattern = "^[А-ЯІІЄґа-яіієґ' ]+$";
    if (!TRegEx::IsMatch(pib, pattern)) {
        . ShowMessage("ПІБ повинен містити лише українські літери.");
        LabeledEdit1->SetFocus();
        return;
    }
}
```

### Лістинг 3.5 – Приклад регулярного виразу

```
void __fastcall TForm14::Edit5Exit(TObject *Sender)
{
    String e-mail = Edit5->Text;
    UnicodeString pattern = "^\\w+([-+.']\\w+)*@\\w+([-+.']\\w+)*\\.\\w+([-+.']\\w+)*$";
    if (!TRegEx::IsMatch(email, pattern)) {
        ShowMessage("Введіть дійсний e-mail.");
        Edit5->SetFocus();
    }
}
```

Для уникнення помилок введення дати та вибору полів для заповнення таблиць, використано об'єкти класів `TDatePicker` для зручності введення часу з перевіркою на логічні межі дати та `ComboBox` для вибору полів з інших таблиць, наприклад вибір навчальної дисципліни у введенні результату. Об'єкти продемонстровано на рисунках 3.29-3.30.

Рисунок 3.29 – Введення дати для фільтрації

Рисунок 3.30 – Вибір навчальної дисципліни

	Вик.	Гуненко Я.М.			КП.ПЗ.211.06.ПЗ	Арк.
	Пер.	Гапоненко Н.В.				
Змн.	Арк.	№ докум.	Підпис	Дата		31



Для коректного введення числових значень використано властивість NumbersOnly для LabeledEdit. Налаштування властивості продемонстровано на рисунку 3.31.



Рисунок 3.31 – Налаштування Edit

Для обмеження доступу до роботи з даними використано дезактивацію пунктів меню для роботи з даними для неавторизованих користувачів. Блокування продемонстровано на рисунку 3.32.

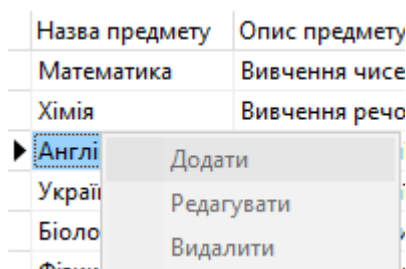


Рисунок 3.32 – Заблоковане меню

Для повідомлення про некоректне введення використано повідомлення для користувача. Приклад помилки продемонстровано на рисунку 3.33.

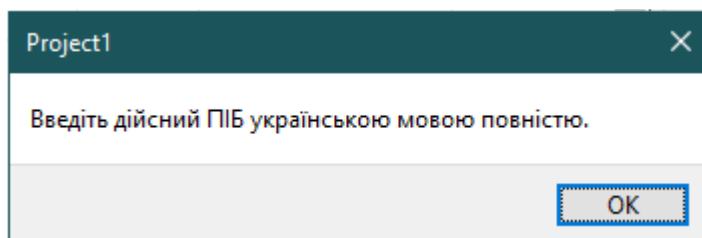


Рисунок 3.33 – Вікно помилки

Для підтвердження видалення використано діалогове вікно. Приклад вікна продемонстровано на рисунку 3.34.

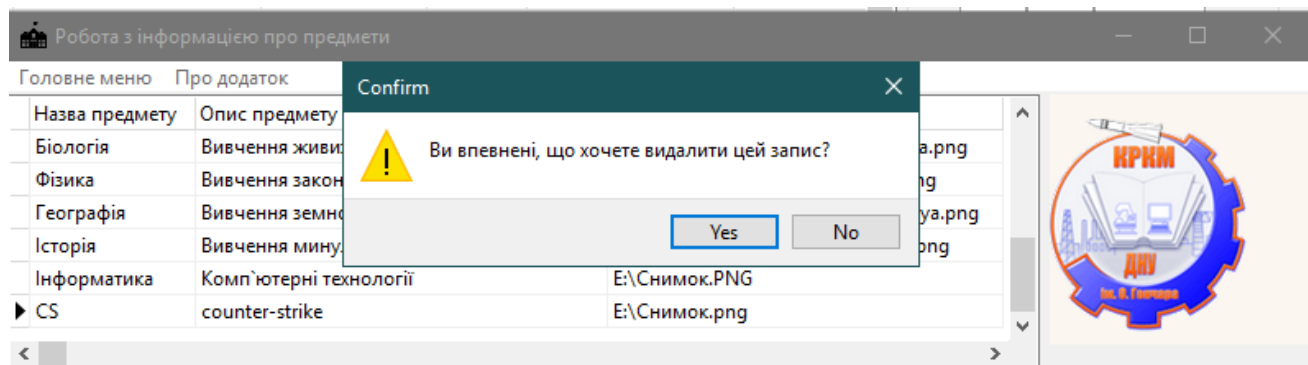


Рисунок 3.34 – Вікно помилки

#### 4 ІНСТРУКЦІЯ З КОРИСТУВАННЯ ПРОГРАМНОЮ СИСТЕМОЮ

Для початку роботи з програмою необхідно ознайомитися з її основним інтерфейсом та функціоналом, щоб зрозуміти, як виконувати облік і керування даними. У програмі реалізовано декілька ролей з різними рівнями доступу до даних, що дозволяє забезпечити безпечне і ефективне управління інформацією. Кожна роль має свої права, що визначають доступ до певних функцій, таких як перегляд, редагування, додавання або видалення записів.

Програма створена для полегшення обліку результатів НМТ, керування даними про навчальні заклади, дисципліни та сертифікати. Користувач може легко отримувати доступ до необхідної інформації, використовувати фільтрацію, а також генерувати сертифікати у форматі HTML для подальшого використання. Інструкція покликана допомогти зрозуміти основні принципи роботи програми, а також полегшити використання доступних інструментів і функцій.

Головне правило введення даних у поля полягає в тому, що символи, які не мають сенсу для конкретного поля, вводити заборонено. Для всіх інших полів передбачені вбудовані перевірки та варіанти вибору.

При завантаженні додатка відображається головне вікно, в якому представлено стислий зміст інформації, що зберігається у базі даних. У вікні містяться ПІБ особи, яка складала НМТ, дата складання, навчальна дисципліна та статус здачі або не здачі.

	Вик.	Гуненко Я.М.			КП.ПЗ.211.06.ПЗ	Арк.
	Пер.	Гапоненко Н.В.				34
Змн.	Арк.	№ докум.	Підпис	Дата		

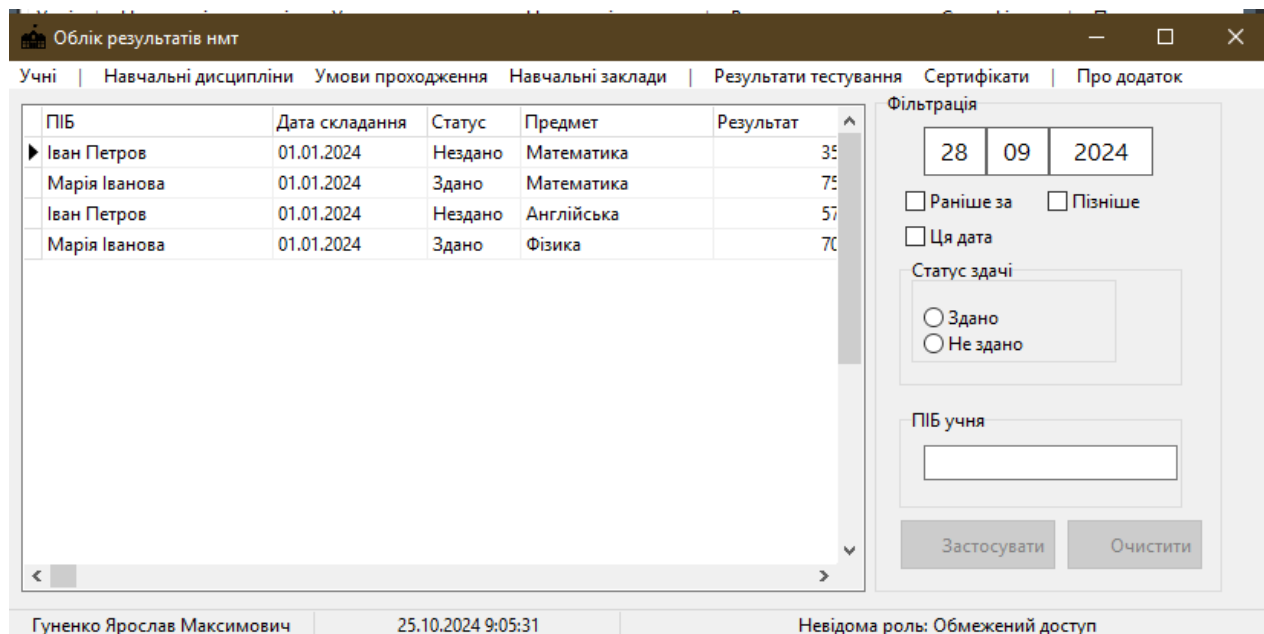


Рисунок 4.1 – Головне вікно

Щоб відсортувати записи у таблиці, достатньо натиснути на заголовок стовпця, щоб упорядкувати записи за цим стовпцем.

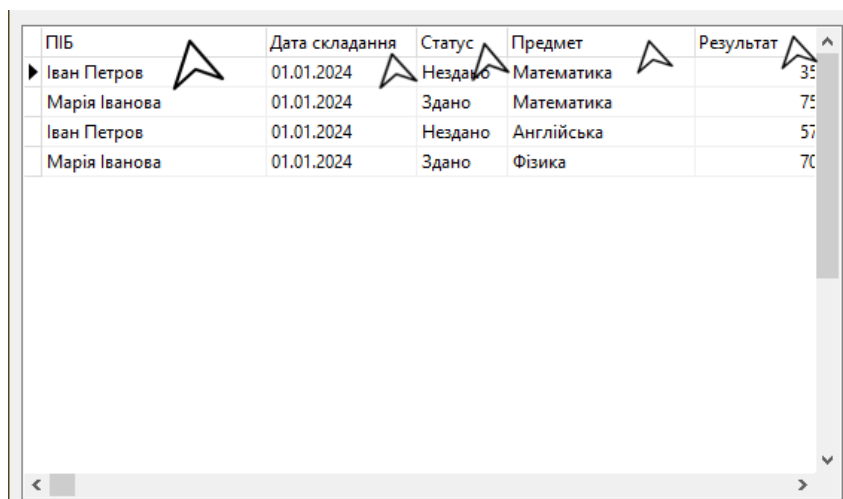


Рисунок 4.2 – Спосіб сортування

Також для зручного перегляду даних можна скористатися опцією фільтрації. За умови відсутності заповнених фільтрів кнопки запуску фільтрації та очищення фільтрів неактивні. Для їх активації необхідно заповнити хоча б один із запропонованих фільтрів, як показано на рисунку 4.3. Для фільтрації за датою потрібно вибрати дату у випадаючому списку та один із доступних варіантів у CheckBox. Для фільтрації за статусом здачі необхідно обрати потрібний варіант із двох кружечків. Для фільтрації за ПІБ учня потрібно ввести ПІБ у відповідне поле,

при цьому дозволені лише українські літери, пробіли та апостроф. Фільтри можна комбінувати, обираючи декілька варіантів одночасно.

Рисунок 4.3 – Список можливих фільтрів

Для переходу до роботи з іншими таблицями передбачено головне меню, яке відкриває відповідне вікно залежно від обраного пункту. Пункти головного меню продемонстровані на рисунку 4.4. За замовчуванням доступ обмежується лише переглядом даних у цих формах. Для отримання розширеного доступу необхідно пройти авторизацію, вибравши пункт головного меню "Авторизація" → "Увійти" або "Вийти". Після цього відкриється форма авторизації з полями для введення логіна та пароля користувача (рисунок 4.6). У разі успішної авторизації за однією з чотирьох можливих ролей (відповідальний за навчальні дисципліни та заклади, відповідальний за результати та сертифікати, відповідальний за учасників НМТ) статусбар оновлюється, відображаючи поточну роль користувача. У разі невдалої авторизації роль залишиться як "Невідома роль". Усі можливі ролі продемонстровані на рисунках 4.7-4.9.

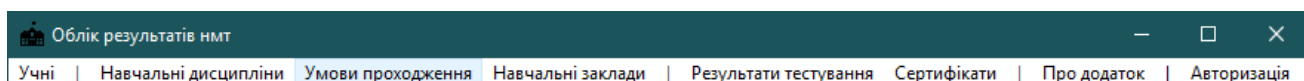


Рисунок 4.4 – Головне меню

	Вик.	Гуненко Я.М.			КП.ПЗ.211.06.ПЗ	Арк.
	Пер.	Гапоненко Н.В.				36
Змн.	Арк.	№ докум.	Підпис	Дата		

Гуменко Ярослав Максимович	25.10.2024 9:40:49	Невідома роль: Обмежений доступ
----------------------------	--------------------	---------------------------------

Рисунок 4.5 – Статусбар

Зміна ролі

Логін

Пароль

Підтвердити

Гуменко Ярослав Максимович	25.10.2024 9:47:58	Невідома роль: Обмежений доступ
----------------------------	--------------------	---------------------------------

Рисунок 4.6 – Форма авторизації

Гуменко Ярослав Максимович	25.10.2024 9:49:13	Менеджер предметів: Доступ до навчальних дисциплін
----------------------------	--------------------	--

Рисунок 4.7 – Доступ до навчальних дисциплін, закладів та умов проведення

Гуменко Ярослав Максимович	25.10.2024 9:50:00	Менеджер результатів: Доступ до результатів і сертифікатів
----------------------------	--------------------	--

Рисунок 4.8 – Доступ до результатів тестування та сертифікатів

Гуменко Ярослав Максимович	25.10.2024 9:51:05	Менеджер учнів: Повний доступ до даних учнів
----------------------------	--------------------	--

Рисунок 4.9 – Доступ до учнів

Після вибору пункту меню "Учні" відкривається вікно, яке демонструє таблицю з інформацією про учнів. У таблиці відображаються такі поля: номер паспорта, тип паспорта, ПІБ, дата народження, стать, E-mail, номер телефону, номер посвідчення та ІПН. Ця інформація забезпечує зручний перегляд та швидкий доступ до основних даних учнів.

Робота з інформацією про учнів

Головне меню

Про додаток

Номер паспорту	Тип паспорта	ПІБ	Birth_date	Стать	E-mail	Номер телефо	Номер посвідч	ІПН	Додаткова пс
BC654322	ID-карта	Іван Петров	20.04.2020	M	ivan@gmail.com	1111111111	EC1245	9876543219	(WIDEMEMO)
BC654321	Паперовий	Марія Іванова	22.11.2003	F	maria.ivanova@gmail.com	0987654321	EC5678	1234567899	(WIDEMEMO)
BC654329	Паперовий	Петренко Пітрої	28.09.2024	M	gunenkoym@gmail.com	09653403129	EC25478	9875421367	(WIDEMEMO)
UA325419	ID-карта	Назаренко Інна	22.10.1995	F	hh@gmail.com	0967825711	EC3458	8564723142	(WIDEMEMO)
UA12345678	ID-карта	Зеленський Воло	22.10.1993	M	hh@gmail.com	0967854712	EC5478	1234567832	(WIDEMEMO)
BC654322	Паперовий	AAAA	22.10.2006	M	hh@gmail.ru	0967825711	EC2547	9875463219	(WIDEMEMO)
BC654325	Паперовий	Макаров Антон	22.10.2007	F	hh_l@gmail.com	0987654322	EC1241	9876543218	(WIDEMEMO)

Фільтрація

Тип паспорта

Стать

Дата народження

E-mail

Застосувати

Очистити

☐ Раніше за
☐ Пізніше
☐ Ця дата

Рисунок 4.10 – Вікно з інформацією про учасників

Вик.	Гуменко Я.М.				КП.ПЗ.211.06.ПЗ	Арк.
Пер.	Гапоненко Н.В.					37
Змн.	Арк.	№ докум.	Підпис	Дата		

Для сортування записів у таблиці потрібно натиснути на заголовок відповідного стовпця, що дозволяє впорядкувати записи за обраним критерієм.

Додаток також підтримує функцію фільтрації даних. Якщо фільтри не заповнені, кнопки для запуску фільтрації та очищення фільтрів залишаються неактивними. Для активації необхідно заповнити хоча б один із запропонованих фільтрів у нижній частині форми. Можливі фільтри за типом паспорта, статтю, датою народження та полем E-mail. Для фільтрації за датою народження можна вибрати конкретну дату та встановити один із варіантів у чекбоксах: "Раніше за," "Пізніше" або "Ця дата." Фільтри можна комбінувати для звуження пошуку. Тип паспорта обирається зі списку, стать — із двох варіантів, а для введення E-mail необхідно використовувати шаблон "@.\*".

Рисунок 4.11 – Список можливих фільтрів

Окрім цього, користувачам з роллю "відповідальний за учнів" доступна можливість редагування або додавання записів через контекстне меню (РорирMenu) для таблиці, яке відкривається правою кнопкою миші. За допомогою цього меню можна обрати опції "Редагувати" або "Додати" для керування даними учнів.

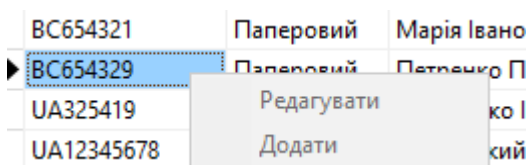


Рисунок 4.12 – РорирMenu для редагування/додавання записів без доступу

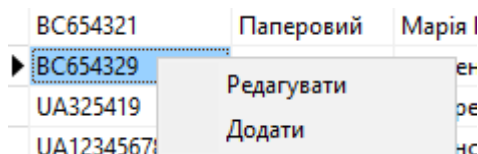


Рисунок 4.12 – РорирMenu для редагування/додавання записів з доступом

Після вибору пункту меню "Редагування" відкривається вікно з полями таблиці "Учні," заповненими відповідно до даних у базі. Кожне поле перевіряється на коректність введення, унікальність та логічний зміст. При виборі пункту меню "Додавання" відкривається вікно з порожніми полями для введення нових даних у таблицю "Учні." Кожне поле також перевіряється на коректність, унікальність та логічний зміст. У разі успішного додавання або зміни запису з'являється повідомлення про успішне виконання операції. У разі виникнення помилки з'являється повідомлення з її описом.

Рисунок 4.13 – Вікно редагування

Рисунок 4.14 – Вікно додавання

Вик.	Гуненко Я.М.				КП.ПЗ.211.06.ПЗ	Арк.
Пер.	Гапоненко Н.В.					
Змн.	Арк.	№ докум.	Підпис	Дата		39



Після натискання на пункт меню "Навчальні дисципліни" відкривається вікно, яке відображає таблицю з інформацією про навчальні предмети. У таблиці містяться такі поля: назва предмета, опис предмета, посилання на зображення та зразок завдань. Це забезпечує швидкий перегляд доступних дисциплін та надає доступ до візуального зразка для кожного предмета.

Для додавання або зміни інформації про предмет у вікні доступні функції додавання, редагування та видалення записів через контекстне меню, яке відкривається правою кнопкою миші. Ці дії доступні лише для користувачів із роллю "Відповідальний за навчальні дисципліни та заклади." Вони можуть керувати записами, редагуючи назви, опис, а також додаючи зображення для предметів.

Користувач може завантажити зображення для предмета двома способами: через вибір файлу з діалогу за допомогою кнопки "Обрати зображення" або вручну ввівши шлях до файлу зображення в рядок. Якщо файл не знайдено або він має неправильний формат, система повідомляє про помилку. Підтримуються файли з розширенням .jpg або .png.

Користувач також може завантажити зразок завдань двома способами: через вибір файлу з діалогу за допомогою кнопки "Обрати зразок" або вручну ввівши шлях до файлу у рядок. Якщо файл не знайдено або він має неправильний формат, система повідомляє про помилку. Підтримуються файли з розширенням .docx або .pdf.

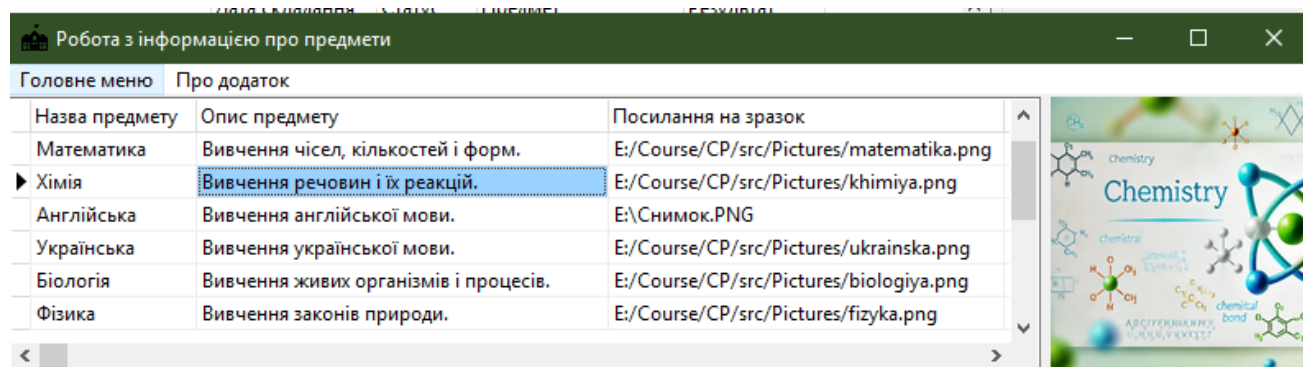


Рисунок 4.15 – Вікно з інформацією про навчальні дисципліни

Коли користувач натискає кнопку "Додати," відкривається вікно для введення нових даних: назви предмета, його опису та шляху до зображення. Усі поля мають бути заповнені коректно. Під час спроби збереження система перевіряє правильність введення даних і повідомляє про можливі помилки. Наприклад, назва предмета не повинна перевищувати 100 символів, а шлях до зображення та зразка має бути дійсним.

Після успішного додавання або редагування запису система оновлює дані в таблиці та виводить повідомлення про успішне збереження інформації.

Рисунок 4.16 – Вікно додавання нового предмету

Після вибору пункту "Редагувати" система відкриває вікно з попередньо заповненими полями для редагування даних про предмет. За потреби користувач може оновити назву, опис або шлях до зображення, попередньо перевіривши його доступність перед збереженням. Усі зміни після збереження негайно відображаються в таблиці.

Рисунок 4.17 – Вікно редагування предмету

	Вик.	Гуненко Я.М.			КП.ПЗ.211.06.ПЗ	Арк.
	Пер.	Гапоненко Н.В.				41
Змн.	Арк.	№ докум.	Підпис	Дата		

У разі видалення запису користувач отримає попередження, якщо предмет використовується в інших таблицях, і запис не може бути видалений.

Після натискання на пункт меню "Умови проходження" відкривається вікно з таблицею, що містить інформацію про умови проходження НМТ. У таблиці відображаються такі поля, як назва предмета, максимальний бал, мінімальний прохідний бал, мінімальний бал, статус та дата складання. Це дозволяє зручно переглядати та аналізувати умови складання для різних навчальних дисциплін.

Додаток підтримує фільтрацію даних. Користувач може обрати конкретну дату складання за допомогою селектора дати та вказати умову: "раніше за," "пізніше" або "ця дата," що дозволяє зосередитися на умовах, які діяли до або після певної дати, або точно на обрану дату. Також є можливість фільтрації за навчальною дисципліною, що дає змогу переглядати умови складання лише для конкретного предмета. Записи у таблиці автоматично оновлюються відповідно до вибраних критеріїв. Якщо жоден фільтр не обрано, кнопки для запуску фільтрації та очищення фільтрів залишаються неактивними.

Користувачі з роллю "Відповідальний за навчальні дисципліни та заклади" можуть додавати та редагувати записи через контекстне меню, яке відкривається правою кнопкою миші. При додаванні нового запису відкривається вікно для введення даних: максимального балу, мінімального прохідного балу, мінімального балу, статусу (обов'язковий чи необов'язковий), дати складання, а також вибору навчальної дисципліни з випадającego списку. Після введення даних користувач натискає кнопку "Додати," і система перевіряє коректність введених значень, зокрема перевіряється, щоб максимальний бал був більшим за мінімальний, а мінімальний прохідний бал не перевищував максимальний.

Якщо під час редагування запису користувач обирає запис з таблиці та натискає пункт "Редагувати," система відкриває вікно з попередньо заповненими даними, які можна змінити. Після успішного редагування запису система оновлює таблицю.

	Вик.	Гуненко Я.М.			КП.ПЗ.211.06.ПЗ	Арк.
	Пер.	Гапоненко Н.В.				42
Змн.	Арк.	№ докум.	Підпис	Дата		

Якщо користувач бажає повернутися до перегляду всіх записів без застосованих фільтрів, він може скористатися кнопкою "Очистити," яка скидає всі фільтри і повертає повний список умов складання у таблиці.

Робота з інформацією про умови проходження НМТ

Головне меню Про додаток

Назва предмету	Максимальний бал	Мінімальний прохідний бал	Мінімальний бал	Статус	Дата складання
► Інформатика	100	60	30	обов'язково	01.01.2024
Фізика	90	70	50	обов'язково	01.01.2024
Англійська	85	65	55	обов'язково	01.01.2024
Українська	95	80	70	не обов'язково	01.02.2024
Біологія	88	75	60	обов'язково	01.03.2024

Фільтри для таблиці

Дата створення ☐ Раніше за ☐ Ця дата ☐ Пізніше

Навчальна дисципліна

Рисунок 4.18 – Вікно з умовами проходження НМТ

Додавання даних

Статус ☐ Обов'язково ☐ Не обов'язково

Дата створення

Максимальний бал

Мінімальний бал

Прохідний бал

Навчальна дисципліна

Рисунок 4.19 – Вікно додавання нових умов

Редагування даних

Статус ☐ Обов'язково ☒ Не обов'язково

Дата створення

Максимальний бал

Мінімальний бал

Прохідний бал

Навчальна дисципліна

Рисунок 4.20 – Вікно редагування умов

Після натискання на пункт меню "Навчальні заклади" відкривається вікно з таблицею, що містить інформацію про навчальні заклади. Таблиця включає такі

поля: місто, регіон, тип, E-mail та ПІБ відповідальної особи. Це дозволяє користувачам швидко переглядати список навчальних закладів з основною інформацією про кожен заклад.

Для додавання, редагування або видалення записів у таблиці доступне контекстне меню, яке відкривається правою кнопкою миші. Ці дії можуть виконувати лише користувачі з роллю "Відповідальний за навчальні дисципліни та заклади." Вони можуть додавати нові записи або редагувати наявні, зокрема оновлювати назву міста, регіон, тип, E-mail та ПІБ відповідальної особи.

Записи для кожного навчального закладу можна додати за допомогою кнопки "Додати," яка відкриває вікно введення даних: міста, регіону, типу закладу, E-mail та ПІБ відповідальної особи. Під час додавання або редагування запису система перевіряє коректність введених даних, наприклад, щоб E-mail мав дійсний формат, а введені символи відповідали українській мові.

У разі видалення запису користувач отримає попередження, якщо даний заклад використовується в інших таблицях, зокрема у таблиці "result." У такому випадку запис не може бути видалений.

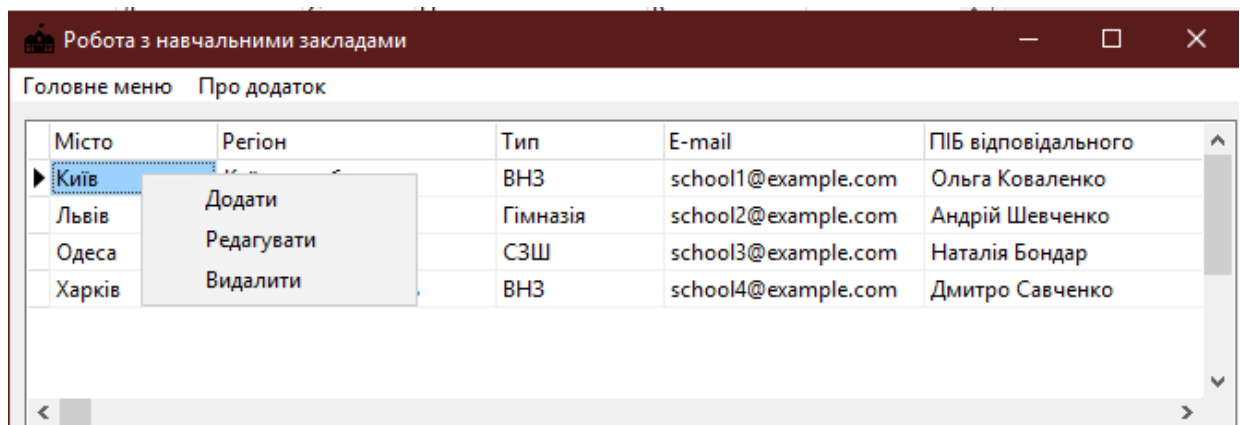


Рисунок 4.21 – Вікно роботи з навчальними закладами

Місто	Область	Тип
<input type="text"/>	<input type="text"/>	<input type="text"/>
E-mail	ПІБ відповідального	Додати
<input type="text"/>	<input type="text"/>	<input type="button" value="Додати"/>

Рисунок 4.22 – Вікно додавання даних

Рисунок 4.23 – Вікно редагування даних

Після натискання на пункт меню "Результати тестування" користувач з роллю "Відповідальний за результати та сертифікати" відкриває вікно з таблицею, що містить дані про результати тестування. У таблиці представлені такі поля: ПІБ студента, назва предмета, результат (бал), максимальний бал, прохідний бал, статус (зараховано/не зараховано), дата складання, дата укладання умов та E-mail навчального закладу. Це дозволяє користувачеві швидко переглядати результати тестів і фільтрувати їх за необхідними критеріями.

Для додавання, редагування та видалення записів доступне контекстне меню, яке відкривається правою кнопкою миші. Користувачі з роллю "Відповідальний за результати та сертифікати" можуть керувати записами, зокрема додавати нові результати тестування з вказанням ПІБ студента, предмета, балу, статусу та дати укладання умов, а також редагувати наявні записи з можливістю змінювати інформацію про студента, предмет або результат тесту. При видаленні запису система перевіряє наявність залежностей у таблиці результатів або сертифікатів і попереджає користувача, якщо запис не може бути видалений.

Функціонал фільтрації дозволяє обирати результати за різними критеріями, такими як ПІБ студента, назва предмета та дата складання (раніше, пізніше або обрана дата). При натисканні на кнопку "Застосувати" активується обраний користувачем фільтр, а при натисканні на кнопку "Очистити" всі параметри фільтрації скидаються, і таблиця відображає повний список записів.

Після додавання, редагування або видалення запису система автоматично оновлює дані в таблиці, щоб відобразити актуальну інформацію.

Додавання інформації про результати

Головне меню Про додаток

ПІБ студента	Предмет	Результат	Максимально	Прохідний	Статус	Дата
Іван Петров	Математика	35	100	60	Не зараховано	01.0
Марія Іванова	Математика	75	90	70	Зараховано	01.0
Іван Г	Математика	57	85	65	Не зараховано	01.0
Марія	Математика	70	90	70	Зараховано	01.0

Фільтри для таблиці

Дата складання

25 10 2024

☐ Раніше за  
☐ Ця дата  
☐ Пізніше

Навчальна дисципліна

Предмет пов'язаний з ві

ПІБ Студента

Застосувати

Очистити

Рисунок 4.24 – Вікно роботи з результатами тестування

Додавання даних

Поля для введення інформації про результати

ПІБ учня

Навчальний заклад

Навчальна дисципліна

Дата укладання умов

01 01 1990

Результат у балах

Статус

☐ Здано ☐ Не здано

Максимальний: -

Прохідний: -

Мінімальний: -

Дата складання

28 09 2024

Додати

Рисунок 4.25 – Вікно додавання даних

Рисунок 4.26 – Вікно редагування даних

Користувач із роллю "Відповідальний за результати та сертифікати" має доступ до пункту меню "Сертифікати," який відкриває вікно для управління інформацією про сертифікати студентів. У таблиці відображаються такі дані: ПІБ студента, PIN-код сертифіката, дати створення та завершення дії сертифіката, а також його статус ("Дійсний" або "Не дійсний").

Користувач може фільтрувати сертифікати за датою створення, обираючи конкретну дату або інтервал (раніше чи пізніше обраної дати), а також сортувати дані в таблиці, натискаючи на заголовки стовпців.

Додатково користувач має можливість генерувати HTML-файл сертифіката. Для цього потрібно ввести PIN-код студента, натиснути кнопку "Згенерувати," і система створить HTML-документ, який містить особисті дані студента, статус сертифіката та результати тестів. Результати оцінюються за 200-бальною шкалою, відображаючи набрані студентом бали.

У меню "Сертифікати" є контекстне меню, яке дозволяє додавати нові записи або редагувати наявні. При виборі команди "Додати" відкривається вікно для введення даних нового сертифіката. Користувач вводить ПІБ студента, PIN-код, дати створення та завершення дії сертифіката та обирає статус. Якщо PIN-код



вже існує, система виводить повідомлення про помилку. Після успішного введення даних, натискання кнопки "Додати" зберігає новий сертифікат у базі.

Для редагування сертифіката обирається команда "Редагувати," що відкриває вікно з уже заповненими полями. Користувач може змінити будь-які дані, після чого натискає кнопку "Зберегти," щоб зберегти зміни в базі.

**Зміна інформації про сертифікати НМТ**

Головне меню Про додаток

ПІБ студента	PIN	Дата створення	Дата закінчення	Статус
▶ Іван Петров	123458	01.07.2024	01.07.2025	Дійсний
Марія Іванова	789012	02.07.2024	02.07.2025	Дійсний
Макаров Антон	123459	05.01.2000	23.10.2024	Не дійсний

Робота з сертифікатами

ПІН код сертифікату

Дата створення

☐ Раніше за ☐ Ця дата ☐ Пізніше

23 10 2024

Рисунок 4.27 – Вікно роботи з сертифікатами

**Додавання даних**

Поля для введення інформації про результати

ПІБ учня

Навчальний заклад

Навчальна дисципліна

Дата укладання умов  01  01  1990

Результат у балах

Статус ☐ Здано ☐ Не здано

Максимальний: -

Прохідний: -

Мінімальний: -

Дата складання  28  09  2024

Рисунок 4.28 – Вікно додавання даних

Редагування даних

Поля для введення інформації про сертифікати

ПІН-код

789012

ПІБ учня

Марія Іванова

Дата створення

02072024

Статус

☒ Дійсний
☐ Не дійсний

Дата ануляції

02072025

Підтвердити

Рисунок 4.29 – Вікно редагування даних

Сертифікат НМТ

Національний Мультипредметний Тест

Дата створення: 01.07.2024

Номер сертифіката: 1

Прізвище, Ім'я, По батькові: Іван Петров

PIN-код: 123458

Термін дії сертифіката: 01.07.2025

Результати за рік

Предмет: Математика

Набрані бали (200-бальна шкала): 100,00

Предмет: Англійська

Набрані бали (200-бальна шкала): 100,00

Рисунок 4.30 – Створений сертифікат

	Вик.	Гуненко Я.М.			КП.ПЗ.211.06.ПЗ	Арк.
	Пер.	Гапоненко Н.В.				49
Змн.	Арк.	№ докум.	Підпис	Дата		

## ВИСНОВКИ

У процесі розробки курсового проєкту «Програма ведення обліку здачі НМТ» було спроектовано та реалізовано програмний додаток, який може використовуватися як великими центрами оцінювання, так і малими освітніми установами.

Розроблене програмне забезпечення є зручним для користувачів із різними рівнями навичок роботи з комп'ютером. Додаток має інтуїтивний інтерфейс, підтримує фільтрацію та сортування даних, а також можливість генерувати сертифікати у форматі HTML.

У результаті реалізації функцій додатка забезпечено:

- облік даних про студентів, їх результати тестування та навчальні дисципліни;
- редагування та видалення даних з перевіркою коректності введення та збереження цілісності бази даних;
- пошук та фільтрацію за різними критеріями, що дозволяє швидко знаходити потрібну інформацію;
- управління доступом до даних, що забезпечує захист інформації.

Створений додаток допомагає організувати облік результатів НМТ та аналізувати отримані дані, що сприяє покращенню управління процесом тестування.

	Вик.	Гуменко Я.М.			КП.ПЗ.211.06.ПЗ	Арк.
	Пер.	Гапоненко Н.В.				50
Змн.	Арк.	№ докум.	Підпис	Дата		

## СПИСОК ВИКОРИСТАНИХ МАТЕРІАЛІВ

1. Стандарт C++ (ISO/IEC 14882:2011): Офіційна специфікація мови C++.
2. Б'ярн Страуструп. "The C++ Programming Language" та інші роботи автора.
3. Документація до стандартної бібліотеки шаблонів (STL): Детальний опис контейнерів, алгоритмів та ітераторів.
4. X3 Secretariat: Standard — The C Language. X3J11/90-013. Computer and Business Equipment Manufactures Association, 311 First Street, NW, Suite 500, Washington, DC 20001, USA.
5. <https://ru.bmstu.wiki> [Електронний ресурс].
6. Офіційний сайт MySQL / Documentation [Електронний ресурс] – Режим доступу: <https://dev.mysql.com/doc/refman/8.0/en/what-is-mysql.html>.
7. <http://www.mysql.ru> [Електронний ресурс].
8. <https://ru.wikipedia.org/wiki/MySQL> [Електронний ресурс].

	Вик.	Гуменко Я.М.			КП.ПЗ.211.06.ПЗ	Арк.
	Пер.	Гапоненко Н.В.				51
Змн.	Арк.	№ докум.	Підпис	Дата		



```

CREATE DATABASE NMT_results;
USE NMT_results;

-- Створення таблиці "Student"
CREATE TABLE Student (
    Student_id INT PRIMARY KEY AUTO_INCREMENT,
    Passport_num VARCHAR(20),
    Passport_type VARCHAR(15) CHECK (Passport_type IN ('ID-карта', 'Паперовий')),
    PIB VARCHAR(100),
    Birth_date DATE,
    Gender CHAR(1) CHECK (Gender IN ('M', 'F')),
    Email VARCHAR(100),
    Phone_num VARCHAR(15) CHECK (Phone_num REGEXP '^[0-9]+$'),
    EduCerf_num VARCHAR(30),
    PN VARCHAR(10),
    Additional TEXT
);

-- Створення таблиці "Users" з новими ролями
CREATE TABLE Users (
    Login VARCHAR(50) PRIMARY KEY,
    Password VARCHAR(50),
    Role VARCHAR(30) CHECK (Role IN ('користувач', 'менеджер предметів', 'менеджер результатів', 'менеджер учнів'))
);

-- Створення таблиці "School" з новими типами
CREATE TABLE School (
    School_id INT PRIMARY KEY AUTO_INCREMENT,
    City VARCHAR(50),
    Region VARCHAR(50),
    Type VARCHAR(20) CHECK (Type IN ('ВНЗ', 'СЗШ', 'Гімназія', 'Ліцей')),
    Email VARCHAR(100),
    Teacher_PIB VARCHAR(100)
);

-- Створення таблиці "Subject" з додатковим полем "Image_name"
CREATE TABLE Subject (
    Subject_id INT PRIMARY KEY AUTO_INCREMENT,
    Name VARCHAR(100),
    Description TEXT,
    Image_name VARCHAR(255)
);

CREATE TABLE Conditions (
    Condition_id INT PRIMARY KEY AUTO_INCREMENT,
    Subject_id INT,
    Max_point INT,
    Min_r_point INT,

```

```

        Min_point INT,
        Status BOOLEAN,
        Date DATE,
        FOREIGN KEY (Subject_id) REFERENCES Subject(Subject_id),
        CHECK (Max_point > Min_point AND Min_r_point < Max_point AND
Min_r_point > Min_point)
);

-- Створення таблиці "Result"
CREATE TABLE Result (
    Res_id INT PRIMARY KEY AUTO_INCREMENT,
    Subj_id INT,
    Condition_id INT,
    Student_id INT,
    Reached_score INT,
    Status BOOLEAN,
    School_id INT,
    Attemp_date DATE,
    FOREIGN KEY (Subj_id) REFERENCES Subject(Subject_id),
    FOREIGN          KEY          (Condition_id)          REFERENCES
Conditions(Condition_id),
    FOREIGN KEY (Student_id) REFERENCES Student(Student_id),
    FOREIGN KEY (School_id) REFERENCES School(School_id)
);

-- Створення таблиці "Certificate"
CREATE TABLE Certificate (
    Cerf_num INT PRIMARY KEY AUTO_INCREMENT,
    Student_id INT,
    PIN VARCHAR(15) CHECK (PIN REGEXP '^[0-9]+$'),
    Creation_date DATE,
    Effect_time DATE,
    Status BOOLEAN,
    FOREIGN KEY (Student_id) REFERENCES Student(Student_id)
);

-- Заповнення таблиці "Student"
INSERT INTO Student (Passport_num, Passport_type, PIB,
Birth_date, Gender, Email, Phone_num, EduCerf_num, PN,
Additional) VALUES
('AB123456', 'ID-карта', 'Іван Петров', '2002-04-15', 'М',
'ivan.petrov@example.com', '1234567890', 'EC1234', '9876543210',
'Без додаткової інформації'),
('BC654321', 'Паперовий', 'Марія Іванова', '2003-11-22', 'F',
'maria.ivanova@example.com', '0987654321', 'EC5678',
'1234567891', 'Без додаткової інформації');

INSERT INTO Users (Login, Password, Role) VALUES
('koristuvach1', 'user1pass', 'менеджер предметів'),
('koristuvach2', 'user2pass', 'менеджер результатів'),
('koristuvach3', 'user3pass', 'менеджер учнів');

-- Заповнення таблиці "School"

```

```

INSERT INTO School (City, Region, Type, Email, Teacher_PIB)
VALUES
('Київ', 'Київська область', 'Ліцей', 'school1@example.com',
'Ольга Коваль'),
('Львів', 'Львівська область', 'СЗШ', 'school2@example.com',
'Андрій Шевченко'),
('Одеса', 'Одеська область', 'Гімназія', 'school3@example.com',
'Наталія Бондар'),
('Харків', 'Харківська область', 'ВНЗ', 'school4@example.com',
'Дмитро Савченко');

-- Заповнення таблиці "Subject" з новими предметами
INSERT INTO Subject (Name, Description, Image_name) VALUES
('Математика', 'Вивчення чисел, кількостей і форм.',
'matematika.jpg'),
('Хімія', 'Вивчення речовин і їх реакцій.', 'khimiya.jpg'),
('Англійська', 'Вивчення англійської мови.', 'angliyska.jpg'),
('Українська', 'Вивчення української мови.', 'ukrainska.jpg'),
('Біологія', 'Вивчення живих організмів і процесів життя.',
'biologiya.jpg'),
('Фізика', 'Вивчення законів природи.', 'fizyka.jpg'),
('Географія', 'Вивчення земної поверхні і клімату.',
'geografiya.jpg'),
('Історія', 'Вивчення минулих подій і епох.', 'istoriya.jpg');

INSERT INTO Conditions (Subject_id, Max_point, Min_r_point,
Min_point, Status, Date) VALUES
(1, 100, 60, 30, TRUE, '2024-01-01'),
(2, 90, 70, 50, TRUE, '2024-01-01'),
(3, 85, 65, 55, TRUE, '2024-01-01'),
(4, 95, 80, 70, FALSE, '2024-02-01'),
(5, 88, 75, 60, TRUE, '2024-03-01');

-- Заповнення таблиці "Result"
INSERT INTO Result (Subj_id, Condition_id, Student_id,
Reached_score, Status, School_id, Attemp_date) VALUES
(1, 1, 1, 85, TRUE, 1, '2024-06-15'),
(2, 2, 2, 75, TRUE, 2, '2024-06-16'),
(3, 3, 1, 60, FALSE, 3, '2024-06-17');

-- Заповнення таблиці "Certificate" з числовими значеннями в PIN
INSERT INTO Certificate (Student_id, PIN, Creation_date,
Effect_time, Status) VALUES
(1, '123456', '2024-07-01', '2025-07-01', TRUE),
(2, '789012', '2024-07-02', '2025-07-02', TRUE);

-- Зміна типу поля Description на VARCHAR(40)
ALTER TABLE Subject MODIFY COLUMN Description VARCHAR(40);

-- Оновлення таблиці Subject з укороченим описом
UPDATE Subject SET Description = 'Вивчення чисел, кількостей і
форм.' WHERE Name = 'Математика';
UPDATE Subject SET Description = 'Вивчення речовин і їх реакцій.'
WHERE Name = 'Хімія';

```



```

UPDATE Subject SET Description = 'Вивчення англійської мови.'
WHERE Name = 'Англійська';
UPDATE Subject SET Description = 'Вивчення української мови.'
WHERE Name = 'Українська';
UPDATE Subject SET Description = 'Вивчення живих організмів і
процесів.' WHERE Name = 'Біологія';
UPDATE Subject SET Description = 'Вивчення законів природи.'
WHERE Name = 'Фізика';
UPDATE Subject SET Description = 'Вивчення земної поверхні і
клімату.' WHERE Name = 'Географія';
UPDATE Subject SET Description = 'Вивчення минулих подій і епох.'
WHERE Name = 'Історія';

```

Запити:

```

SELECT s.PIB,
       r.Attemp_date,
       CASE WHEN r.Status = 1 THEN 'Здано' ELSE 'Не здано' END,
       subj.Name,
       r.Reached_score
FROM Result r
JOIN Student s ON r.Student_id = s.Student_id
JOIN Subject subj ON r.Subj_id = subj.Subject_id

```

```

UPDATE subject
SET Name = 'Математика', Description = 'Вивчення чисел,
кількостей і форм.'
WHERE Subject_id = 1;

```

```

UPDATE subject
SET Name = 'Хімія', Description = 'Вивчення речовин і їх реакцій.'
WHERE Subject_id = 2;

```

```

UPDATE subject
SET Name = 'Англійська', Description = 'Вивчення англійської
мови.'
WHERE Subject_id = 3;

```

```

UPDATE subject
SET Name = 'Українська', Description = 'Вивчення української
мови.'
WHERE Subject_id = 4;

```

```

UPDATE subject
SET Name = 'Біологія', Description = 'Вивчення живих організмів
і процесів.'
WHERE Subject_id = 5;

```

```

UPDATE subject
SET Name = 'Фізика', Description = 'Вивчення законів природи.'
WHERE Subject_id = 6;

```

```

UPDATE subject

```

```

SET Name = 'Географія', Description = 'Вивчення земної поверхні
і клімату.'
WHERE Subject_id = 7;

UPDATE subject
SET Name = 'Історія', Description = 'Вивчення минулих подій і
епох.'
WHERE Subject_id = 8;

UPDATE student
SET PIB = 'Іван Петров'
WHERE Student_id = 1;

UPDATE student
SET PIB = 'Марія Іванова'
WHERE Student_id = 2;

UPDATE student
SET PIB = 'Petrenko Petro'
WHERE Student_id = 3;

UPDATE student
SET PIB = 'Петренко Петро'
WHERE Student_id = 3;

ALTER TABLE school DROP CHECK school_chk_1;

INSERT INTO school (City, Region, Type, Email, Teacher_PIB)
VALUES
('Київ', 'Київська область', 'Ліцей', 'school1@example.com',
'Ольга Коваль'),
('Львів', 'Львівська область', 'СЗШ', 'school2@example.com',
'Андрій Шевченко'),
('Одеса', 'Одеська область', 'Гімназія', 'school3@example.com',
'Наталія Бондар'),
('Харків', 'Харківська область', 'ВНЗ', 'school4@example.com',
'Дмитро Савченко');

```