

# Розробка програми для тестування студентів

Розробив:  
Зайцев А.Д.

Керівник:  
Дужий В.І.

# Програма тестування студентів

Програма для тестування студентів має віконний інтерфейс який складається з декількох вікон, основні з котрих:

- Вікно авторизації;
- Вікно проходження тесту;
- Вікно результату;

Після авторизації студент потраплятиме у головне вікно програми де й буде проходити тести та відповідати на запитання. Реалізовані наступні типи тестових запитань:

- Тест з однією правильною відповіддю;
- Тест з декількома вірними відповідями;
- Тест у якому потрібно обрати одне зображення яке відповідає запитанню тесту;
- Тест де потрібно з клавіатури увести правильну відповідь.

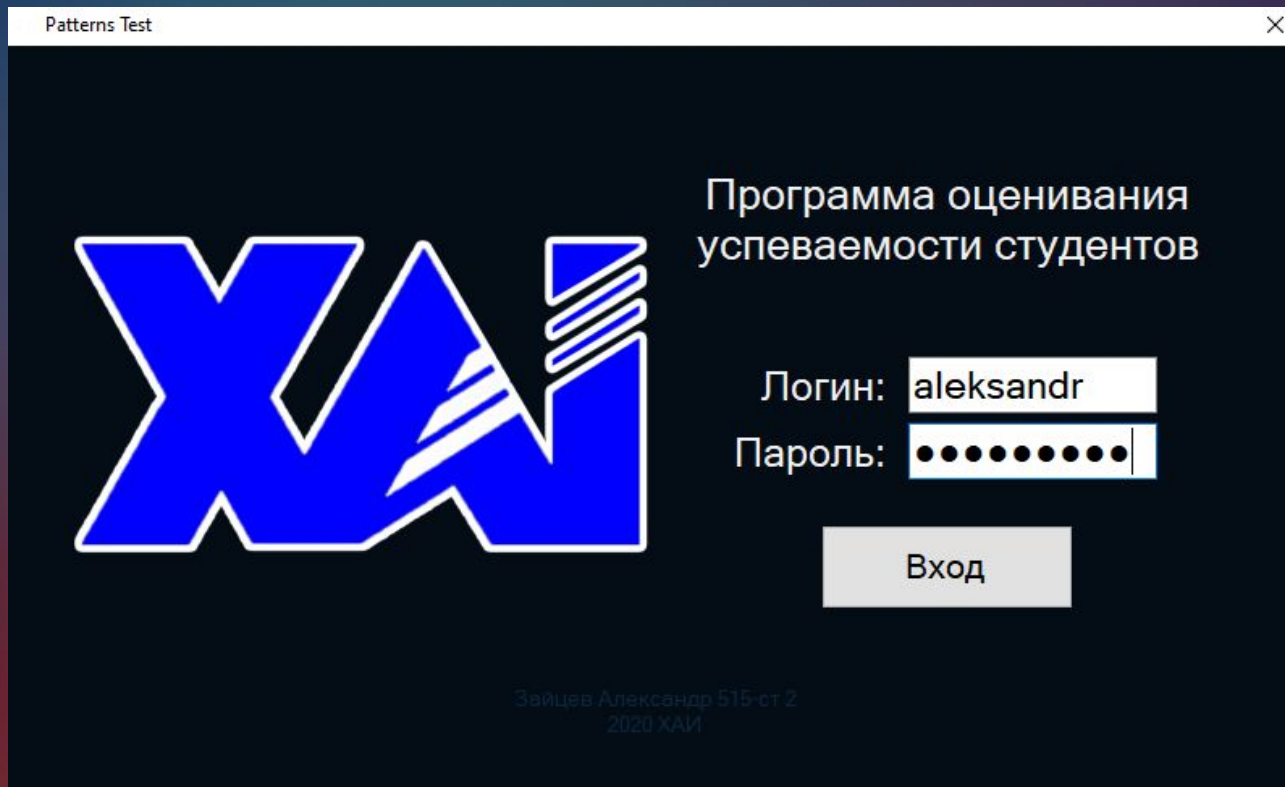
Після завершення тесту студентові буде повідомлено його результат у вікні результату.

# Вибір програмного забезпечення



- Мова: C#
- IDE: Visual Studio 2017
- API Windows Forms

# Вікно авторизації користувача



Для початку тестування студенту потрібно увести свій логін та пароль.

# Структура головного вікна тестування

1. Кількість даних відповідей, а також загальна кількість запитань;
2. Вікно вибору тестового запитання. Питання на які користувач дав відповідь помічені білим кольором. Питання на які ще треба дати відповідь помічені червоним. Обрати бажане запитання також можливо за допомогою кнопок "Назад" та "Далее";
3. Повний текст тестового запитання;
4. Кнопки з варіантами відповідей на тестові питання. В залежності від типу обраного у даний момент тестового запитання замість них може бути текстове поле у яке потрібно з клавіатури ввести відповідь, або обрати декілька відповідей;
5. Таймер що вказує відведений час на проходження тесту. Таймер може бути вимкнений якщо це передбачене у файлі тестових запитань.
6. Логін поточного користувача, а також його попередній результат проходження тесту.

Patterns Test


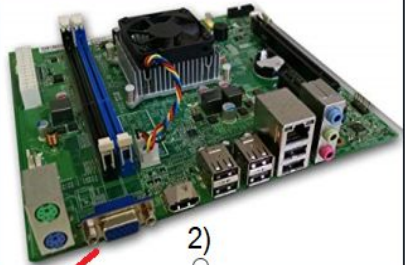
Программа Текущий пользователь: aleksandr Предыдущий результат: 0/0(0%)



Текущий вопрос:

1 Вопросы: 3/9

2

3 На какой картинке изображено оперативное запоминающее устройство?

1)  2) 

3)  4) 

5 Оставшееся время: 04:08

Назад Подтвердить Далее

6



# Ієрархія класів

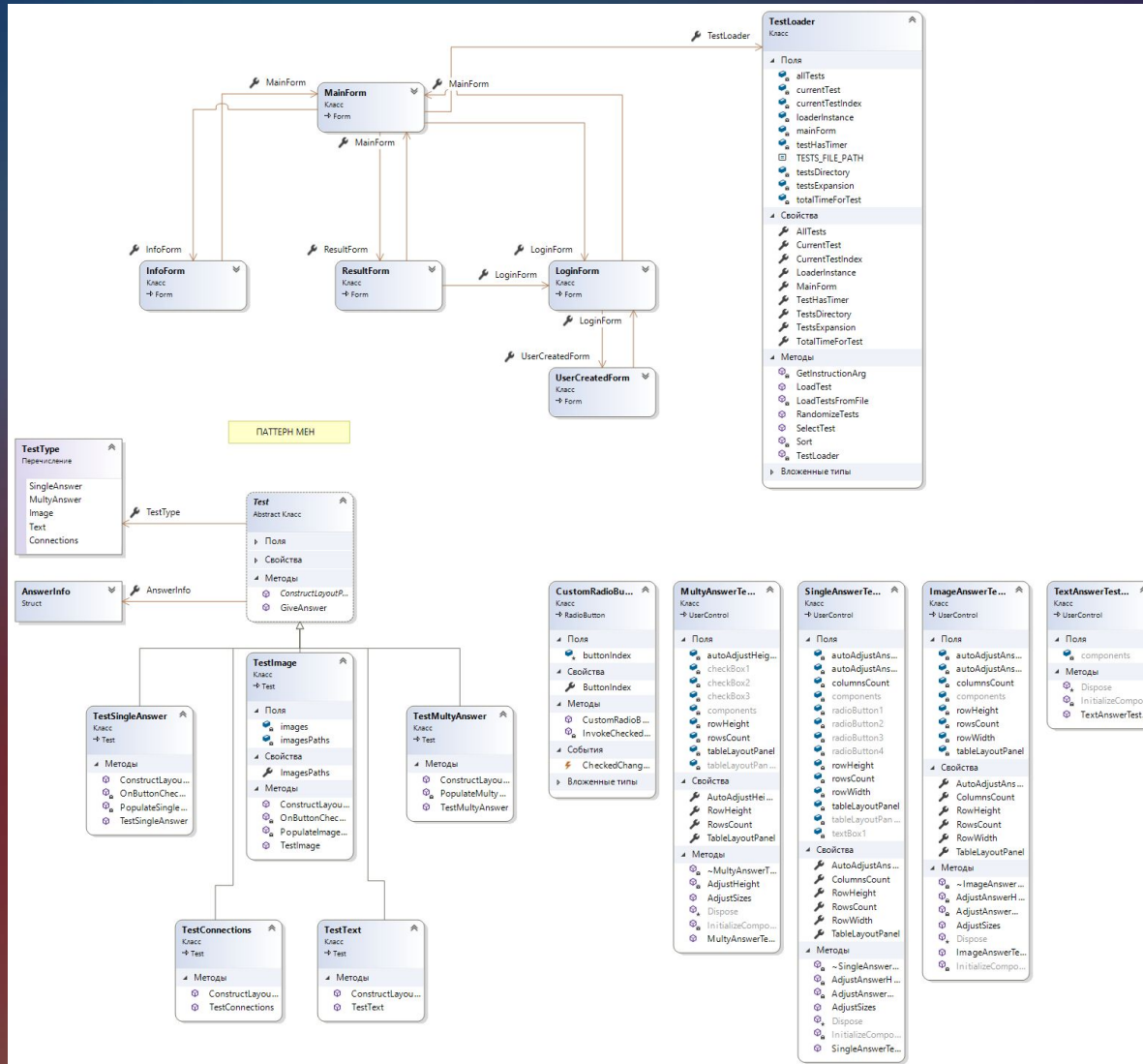


Схема успадковування класів

# Використання патернів програмування

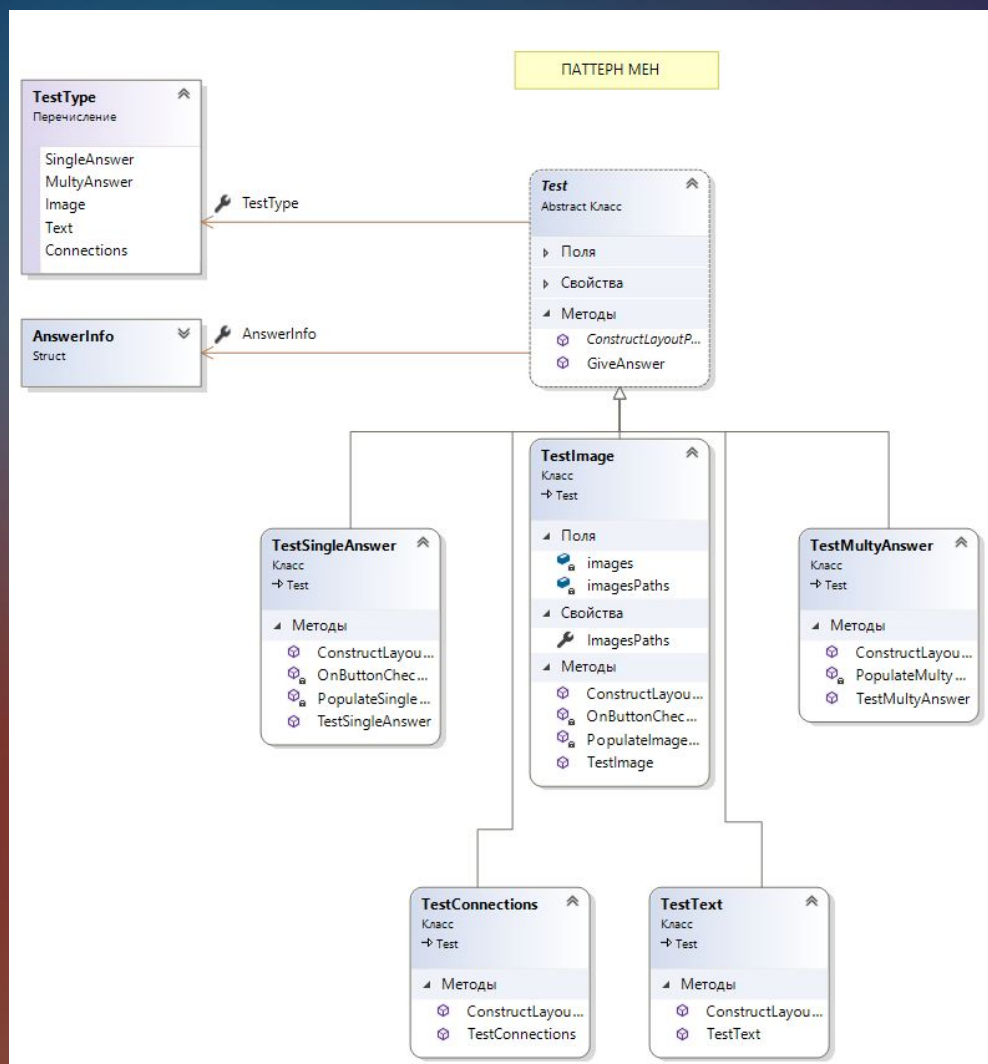
При розробці програми тестування студентів використовувався вбудований у Visual Studio конструктор віконних додатків, а також вбудований редактор UML-діаграм.

За допомогою цього редактору були створені ієрархії класів та зв'язків для віконного додатку.

При розробці ієрархії класів було використано два патерну програмування: "Singleton", або одиночка та "Template", або шаблонний метод.

- "Singleton" - породжучий патерн, який гарантує, що для певного класу буде створений тільки один об'єкт, а також надасть до цього об'єкта точку доступу.
- Шаблонний метод (Template Method) визначає загальний алгоритм поведінки підкласів, дозволяючи їм перевизначити окремі кроки цього алгоритму без зміни його структури.

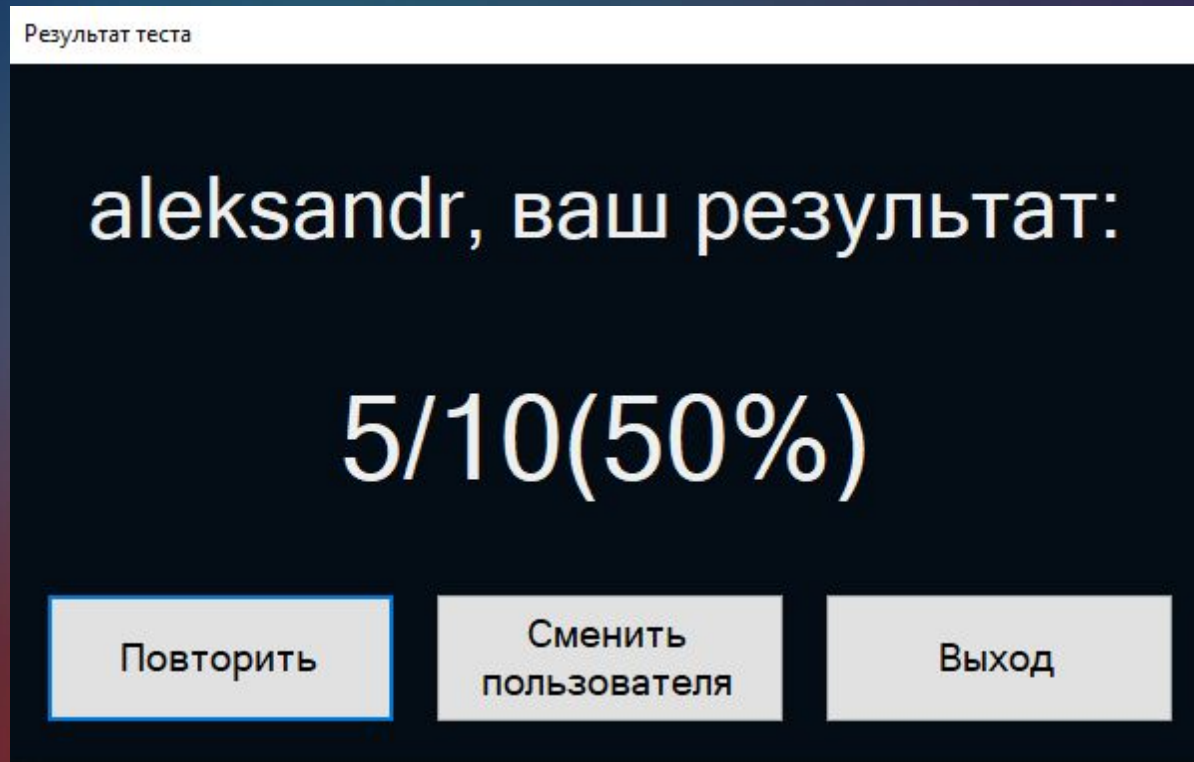
# Патерн програмування "Template"



Ієрархія класів що репрезентують тестові завдання використовує патерн "Template" (шаблонний метод)



# Вікно результату тесту



Після завершення тесту відчиняється вікно результату тесту. У ньому є наступні кнопки:

- “Повторить” – для повторного проходження тесту;
- “Сменить пользователя” – для повторної авторизації;
- “Выход” – для завершення роботи з програмою.

# Висновок

Метою курсового проекту була розробка програми для тестування студентів. Ця програма була розроблена за допомогою використання API Windows Forms для створення віконного інтерфейсу користувача. Внутрішня структура програми буду розроблена з використанням UML-діаграм, а також деяких загальних патернів програмування.

Використання візуального програмування за допомогою мови UML дозволило поліпшити процес розробки програмного забезпечення, а також робити його внутрішню структуру більш зрозумілою для людини. У ході розробки були розглянуті основні концепції об'єктно – орієнтованого програмування та його переваги, такі як: прискорений процес розробки, модульність, безпека, легкість модифікацій.

Розроблений віконний додаток відповідає усім поставленим критеріям у технічному завданні.

**Дякую за увагу!**