КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ ім. І. Сікорського

Кафедра Автоматизованих Систем Обробки Інформації і Управління

(АСОІУ)

**КУРСОВА РОБОТА**

з дисципліни «Програмування-2. Алгоритми та структури даних»

на тему: «Сайт. Онлайн-новини»

студента 1 курсу групи ІС-02

спеціальності «Інформаційні системи та технології»

Плостака Іллі Михайловича

Керівник: Вітковська І. І.

Кількість балів:\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Національна оцінка:\_\_\_\_\_\_\_\_\_\_\_

Члени комісії:

\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

(підпис) (вчене звання, науковий ступінь, прізвище та ініціали)

\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

(підпис) (вчене звання, науковий ступінь, прізвище та ініціали)

Київ – 2021 рік

Національний технічний університет України “КПІ ім. Сікорського”

Кафедра автоматизованих систем обробки інформації і управління

Дисципліна: Програмування-2. Алгоритми та структури даних

**Спеціальність 126 «Інформаційні системи та технології»**

Курс I Група ІС-02 Семестр II

**ЗАВДАННЯ**

**на курсову роботу студента**

Плостака Іллі Михайловича

1. Тема проекту (роботи): Сайт. Онлайн-новини.
2. Строк здачі студентом закінченого проекту (роботи): 17.05.2021.
3. Вихідні дані до проекту (роботи): Тема згідно з варіантом 10, мова програмування C#, документації мови C#.
4. Зміст розрахунково-пояснювальної записки (перелік питань, які підлягають розробці): будування діаграми класів, розробка алгоритмів необхідних для коректної роботи програми.
5. Перелік графічного матеріалу (з точним зазначенням обов’язкових креслень): Діаграма класів.
6. Дата видачі завдання: 25.03.2021.

**Календарний план**

|  |  |  |  |
| --- | --- | --- | --- |
|  | Назва етапів курсової роботи | Термін виконання етапів роботи | Підписи керівника, студента |
| 1 | Отримання теми курсової роботи | 25.03 |  |
| 2 | Підготовка ТЗ | 25.03 |  |
| 3 | Аналіз предметної області |  |  |
| 4 | Проектування архітектури програмної системи |  |  |
| 5 | Розробка сценарію роботи програми |  |  |
| 6 | Узгодження з керівником інтерфейсу користувача |  |  |
| 7 | Розробка програмного забезпечення |  |  |
| 8 | Узгодження з керівником плану тестування |  |  |
| 9 | Тестування програми |  |  |
| 10 | Підготовка пояснювальної записки |  |  |
| 11 | Здача курсової на перевірку | 17.05 |  |
| 12 | Захист курсової роботи | 17.05 |  |

Студент\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Плостак Ілля Михайлович

Керівник\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Вітковська Ірина Іванівна

**Анотація**

Структура та обсяг:

Мета роботи: розробити структуру «Сайт. Онлайн-новини» та алгоритми його роботи.

Об’єкт дослідження: структура «Сайт. Онлайн-новини».

Проведено аналіз предметної області. Побудована діаграма класів. Описано функціонал програмного забезпечення. Реалізована структура «Сайт. Онлайн-новини» на мові C#. Проведено тестування програмного забезпечення.

**Зміст**

[ВСТУП 4](#_Toc72163585)

[РОЗДІЛ 1. Аналіз предметної області 5](#_Toc72163586)

[1.1. Розробка сценарію роботи програми 5](#_Toc72163587)

[1.2. UseCase Діаграма 6](#_Toc72163588)

[1.3. Опис можливостей програми 7](#_Toc72163589)

[РОЗДІЛ 2. Опис програмного забезпечення 8](#_Toc72163590)

[2.1. Логіка ПЗ 8](#_Toc72163591)

[2.1.1. Документація 8](#_Toc72163592)

[2.1.2. Діаграма класів 16](#_Toc72163593)

[2.2. Інтерфейс ПЗ 17](#_Toc72163594)

[2.2.1. Документація 17](#_Toc72163595)

[2.2.2. Діаграма класів 21](#_Toc72163596)

[РОЗДІЛ 3. Тестування ПЗ 22](#_Toc72163597)

[3.1. Методика тестування 22](#_Toc72163598)

[3.2. План тестування 22](#_Toc72163599)

[3.3. Результати тестування ПЗ 23](#_Toc72163600)

[ВИСНОВКИ 28](#_Toc72163601)

# ВСТУП

Мета курсової роботи (КР) – розробити структуру «Сайт. Онлайн-новини».

Основна ціль КР – закріплення здобутих за семестр знань, подальше поглиблення у об’єктно орієнтованого програмування (ООП) на мові програмування C# та підготовка до виконання реальних завдань на прикладі індивідуального завдання КР.

Основна складність КР на відміну від лабораторних робіт – нечітка умова. Умова КР подається як побажання абстрактного клієнта, який не є спеціалістом у програмуванні. Задача програміста сформувати технічне завдання (ТЗ) на основі наданого завдання КР та реалізувати потрібним чином початкове завдання.

Суть завдання варіанту 10 – розробити сайт онлайн новин. Цей сайт повинен відповідати наступним критеріям:

1. Всі новини повинні розташовуватися за певними рубриками.
2. Для новин формується набір тегів, за якими зручно шукати новини зазначеної тематики.
3. Новини можуть вносити на сайт лише зареєстровані автори.
4. Можливо передивлятися новини: за рубрикою, тематикою, певного автора та за конкретний період.

Цей сайт повинен значно спростити перегляд новин порівняно зі звичайним їх списком та надати можливість реєстрації та створення новин для подальшого їх перегляду.

Для вирішення цієї задачі була розроблена діаграма класів завдяки якої надалі була створена повна структура програмного забезпечення (ПЗ). Таким чином завдання було формалізоване.

# РОЗДІЛ 1. Аналіз предметної області

## 1.1. Розробка сценарію роботи програми

На цьому етапі потрібно розробити сценарій роботи ПЗ (це і буде наше ТЗ).

Оскільки ПЗ представляє собою сайт новин, то сценарій повинен починатися з реєстрації користувача. Користувач повинен мати змогу зареєструватися, увійти в існуючий обліковій запис або при бажанні звичайного перегляду новин – увійти як Гість.

Наступний пункт сценарію – меню обслуговування Гостя та Автора. Оскільки Гість це обмежений у функціоналі Автор, то спочатку потрібно розробити меню для Автора.

Згідно з завданням КР Автор повинен вміти:

1. Переглядати новини:
   1. За рубрикою.
   2. За тематикою.
   3. Певного Автора.
   4. За конкретний період.
2. Шукати новини за тегами.
3. Додавати новини.

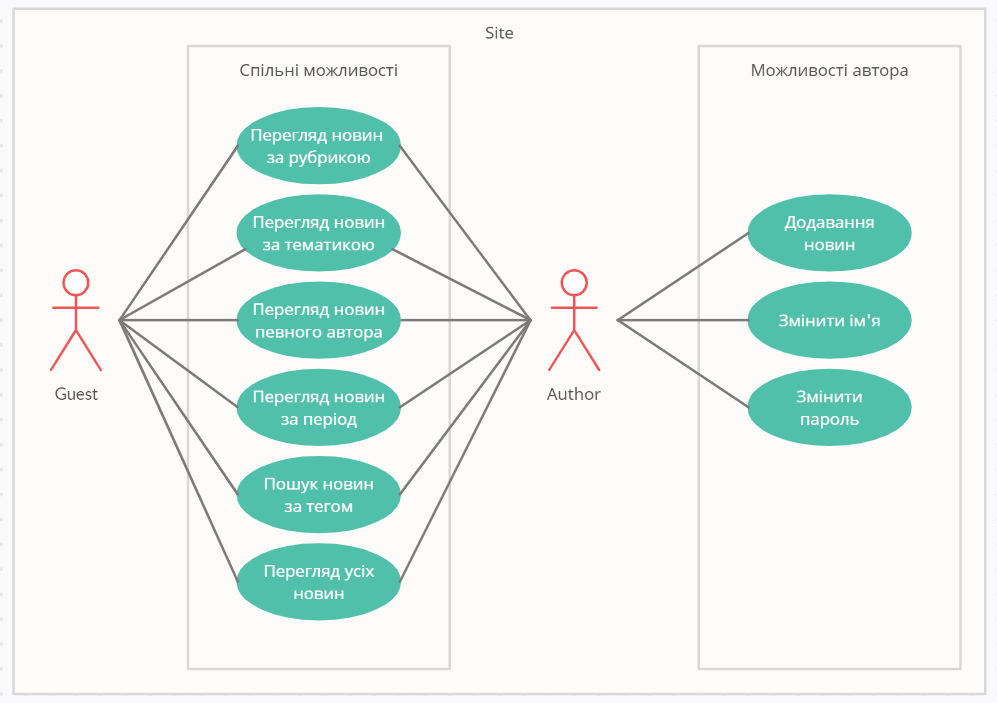
Додатково в процесі аналізу завдання додано ще 1 обов’язковий пункт: Вихід з облікового запису для можливості входу в інший.

Тобто меню для Автора буде мати мінімально 7 пунктів.

Меню для Гостя буде відрізнятися тільки відсутністю пункту додавання новини (згідно з завданням).

Кінець сценарію відбудеться коли користувач вийде з сайту (відповідний пункт початкового меню).

## 1.2. UseCase Діаграма



## 1.3. Опис можливостей програми

На основі ТЗ (сценарію роботи) формується перелік можливостей програми, основні вхідні та вихідні дані та приблизна структура інтерфейсу користувача.

Програма повинна мати такі функції:

1. Додавання нового облікового запису.
2. Вхід в існуючий обліковий запис.
3. Вхід як Гість.
4. Вивід списків:
   1. Рубрики.
   2. Тематики.
   3. Автори.
   4. Новини за конкретний період (при формуванні ТЗ було обрано за період 1 місяць).
   5. Усі новини.
5. Пошук новин за обраним тегом.

Вхідні дані для програми у більшості – рядки, які поступають від користувача. Наприклад логін та пароль облікового запису, назва новини тощо. Виключенням є обирання номерів пунктів меню – користувач вводить номер обраного пункту – ціле число.

Вихідні дані це завжди рядки. Наприклад список новин, Авторів, текст новини та її назва тощо.

Інтерфейс користувача – консоль – представляє собою декілька меню, які заміняють одне одного залежно від вибору пункту поточного меню.

Обрана найзручніша середа для розробки ПЗ на мові програмування C# – Visual Studio 2019 Community Edition.

# РОЗДІЛ 2. Опис програмного забезпечення

## 2.1. Логіка ПЗ

### 2.1.1. Документація

Список класів:

1. Site – головний клас логічної частини ПЗ, клас сайту новин:
   1. Поля:
      1. private \_name – Назва сайту.
      2. private \_news – Список новин сайту.
      3. private \_rubrics – Список рубрик сайту.
      4. private \_themes – Список тематик сайту.
      5. private \_user – Поточний користувач.
      6. private \_authors – Список усіх Авторів.
   2. Властивості:
      1. public AllNews – Отримання списку усіх новин.
      2. public AuthorsToPrint – Отримання списку усіх Авторів в отформатованому вигляді.
      3. public CurUserName – Отримання нікнейма поточного користувача.
      4. public Name – Отримання назви сайту.
      5. public Authors – Отримання списку усіх Авторів.
      6. public NewsToPrint – Отримання списку усіх новін в отформатованому вигляді.
      7. public Rubrics – Отримання списку усіх рубрик.
      8. public RubricsToPrint – Отримання списку усіх рубрик в отформотованному вигляді.
      9. public ThemesToPrint – Отримання списку усіх тематик в в отформатованому вигляді.
   3. Методи:
      1. public Site() – Конструктор[[1]](#footnote-1) с параметром (назвою сайту).
      2. public Register() – Метод реєстрації нового Автора. Метод перевіряє наявність переданих логіну та паролю для запобігання повторної реєстрації. Якщо не знаходить вже зареєстрованого Автора, створює нового користувача типу Author та додає його до списку Авторів. Інакше викликає виключення
      3. public NewsByTheme() – Метод пошуку новин за темою. Повертає список усіх знайдених новин за переданим номером теми у списку тем. Якщо не знаходить жодної новини – викликає виключення.
      4. public NewsByTag() – Метод пошуку новин за тегом. Повертає список усіх знайдених новин за переданим тегом. Якщо не знаходить жодної новини – викликає виключення.
      5. public NewsByRubric() – Метод пошуку новин за рубрикою. Повертає список усіх знайдених новин за переданим номером рубрики у списку рубрик. Якщо не знаходить жодної новини – викликає виключення.
      6. public NewsByMonth() – Метод пошуку новин за місяцем. Повертає список усіх знайдених новин за переданим номером місяця. Якщо не знаходить жодної новини – викликає виключення.
      7. public NewsByAuthor() – Метод пошуку новин за Автором. Повертає список усіх знайдених новин за переданим номером Автора у списку Авторів. Якщо не знаходить жодної новини – викликає виключення.
      8. private MatchTags() – Метод знаходження приналежності переданого тегу до переданого масиву тегів. Повертає True, якщо тег належить масиву. Інакше повертає False.
      9. public MatchPass() – Метод перевірки правильності переданого паролю відносно поточного користувача. Повертає True, якщо пароль правильний. Інакше повертає False.
      10. public Logout() – Метод виходу з поточного облікового запису. Анулює поточного користувача.
      11. public Login() – Метод входу в існуючий обліковий запис. Спочатку шукає за переданими логіном та паролем чи існує такий обліковий запис. Якщо існує – робить знайдений обліковий запис поточним. Інакше викликає виключення.
      12. private FindUser() – Метод знаходження облікового запису за переданними логіном та паролем або просто за логіном (метод перегружений). Якщо знаходить – повертає знайдений обліковий запис. Інакше повертає null.
      13. public ChangeUserPass() – Метод зміни пароля у поточному обліковому записі. Якщо передається null або пустий рядок – викликає виключення. Інакше викликає метод зміни пароля від поточного облікового запису.
      14. public ChangeUserName() – Метод зміни нікнейму у поточному обліковому записі. Якщо передається null або пустий рядок – викликає виключення. Інакше викликає метод зміни нікнейму від поточного облікового запису.
      15. public AddNews() – Метод додавання новини. Перевіряє чи є поточний обліковий запис Автором. Якщо так – створює новину на основі переданих параметрів (назва, номер рубріки, номер теми та текст). Інакше викликає виключення.
      16. public AddGuest() – Метод створення тимчасового облікового запису для перегляду новин. Визивається конструктор класу Guest.
2. User – абстрактний клас користувача, який містить основні дані користувача та є батьківським класом для класів Guest та Author:
   1. Поля:
      1. protected static \_guestsAmount – статичний лічильник кількості створених тимчасових облікових записів.
      2. protected \_hashPass – хеш переданого паролю.
      3. protected \_login – логін.
      4. protected \_nickname – ім’я облікового запису.
      5. protected \_type – тип облікового запису.
   2. Властивості:
      1. public Login – Отримання логіну облікового запису.
      2. public Nickname – Отримання імені облікового запису.
      3. public Type – Отримання типу облікового запису.
   3. Абстрактні методи:
      1. public ChangeName() – Шаблон методу зміни імені облікового запису.
      2. public ChangePass() – Шаблон методу зміни паролю облікового запису.
      3. public CheckPass() – Шаблон методу перевірки правильності пароля.
3. Guest – дочірній класу User клас Гість, тобто тимчасовий обліковий запис:
   1. Полів немає. Тільки ті що наслідуються.
   2. Властивостей немає. Тільки ті що наслідуються.
   3. Методи:
      1. public Guest() – Конструктор за замовчуванням. Ім’я облікового запису завжди: guest + \_guestsAmount. Логіну та хешу пароля присвоюється null.
      2. public overload ChangeName() – Викликає виключення.
      3. public overload ChangePass() – Викликає виключення.
      4. public overload CheckPass() – Викликає виключення.
4. Author – дочірній класу User клас Автор, тобто постійний обліковий запис:
   1. Полів немає. Тільки ті що наслідуються.
   2. Властивостей немає. Тільки ті що наслідуються.
   3. Методи:
      1. public Author() – Конструктор з параметрами (ім’я, логін та пароль). Замість паролю в клас записується його хеш[[2]](#footnote-2)).
      2. public overload ChangeName() – Замінює старе ім’я на передане.
      3. public overload ChangePass() – Замінює старий хеш на хеш переданого паролю.
      4. public overload CheckPass() – Порівнює поточний хеш облікового запису з хешем переданого паролю. Якщо хеши співпали – повертає True. Інакше повертає False.
5. News – клас який інкапсулює новину:
   1. Поля:
      1. private \_month – місяць, в якому була створена новина.
      2. private \_name – назва новини.
      3. private \_rubric – рубрика новини.
      4. private \_tags – масив тегів новини.
      5. private \_text – текст новини.
      6. private \_author – Автор новини.
      7. private \_newsTheme – тематика новини.
   2. Властивості:
      1. public Month – Отримання місяця, в якому була створена новина.
      2. public Name – Отримання назви новини.
      3. public Rubric – Отримання рубрики новини.
      4. public Tags – Отримання масиву тегів новини.
      5. public Text – Отримання тексту новини.
      6. public ItsAuthor – Отримання Автора новини.
      7. public NewsTheme – Отримання тематики новини.
   3. Методи:
      1. public News() – Конструктор за параметрами (назва, рубрика, тематика, текст, Автор, місяць та теги). Якщо теги не передані, то новина отримує теги тематики.
      2. public overload ToString() – Перевантаження автоматичного приведення до рядка (тут для вивода у консоль). Рядок формується таким чином: Назва новини, два перенесення рядка та текст новини.
6. Theme – клас який інкапсулює тематику:
   1. Поля:
      1. private \_name – назва тематики.
      2. private \_tags – масив тегів.
   2. Властивості:
      1. public Name – Отримання назви тематики.
      2. public Tags – Отримання масиву тегів.
   3. Методи:
      1. public Theme() – Конструктор з параметрами (назва, масив тегів). Якщо масив тегів не переданий, то він приймає значення null.
7. Tools – статичний клас допоміжних функцій:
   1. Полів немає.
   2. Властивостей немає.
   3. Статичні методи:
      1. public CreateMD5() – Функція хешування за алгоритмом MD5.

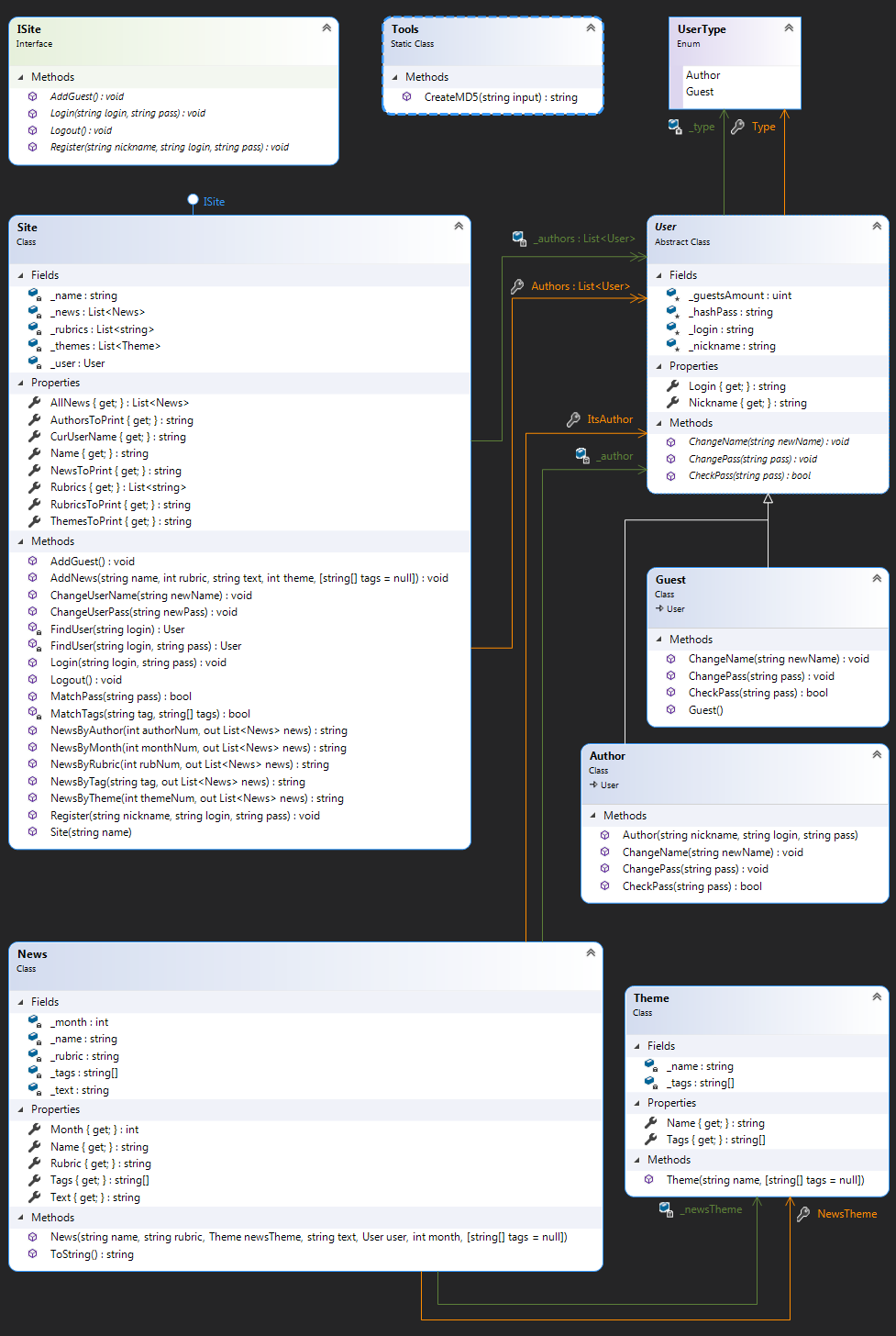
Інтерфейси:

1. ISite – інтерфейс для класу Site:
   1. Шаблонні методи:
      1. public AddGuest().
      2. public Login().
      3. public Logout().
      4. public Register().

Перерахування:

1. UserType – тип користувача:
   1. Guest.
   2. Author.

### 2.1.2. Діаграма класів



## 2.2. Інтерфейс ПЗ

### 2.2.1. Документація

Список класів:

Program – інтерфейс користувача (консольний додаток):

1. Статичні методи:
   1. Main – основана програма. Структура:

Замість баз даних або будь якого іншого способу збереження даних в цьому ПЗ використовується серіалізація[[3]](#footnote-3).

Спочатку перевіряється наявність файлу серіалізації для продовження попередньої сесії. У випадку коли цього файлу немає – створюємо новий екземпляр класу Site.

Далі розташовані 3 конструкції switch (головне меню, меню для Гостя та меню для Автора). Перехід між цими меню здійснюється за допомогою змінної state, яка може приймати три значення:

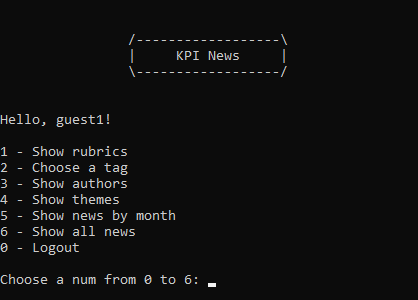
* StateType.NoOne – головне меню.
* StateType.Guest – меню Гостя.
* StateType.Author – меню Автора.

Головне меню:

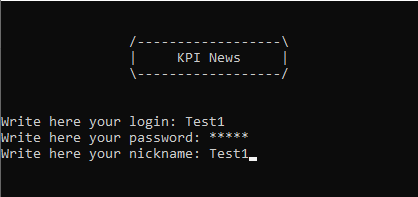


Вхід як гість одразу перекидає у меню Гостя.

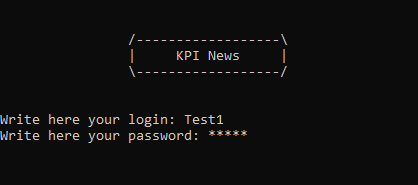
Меню Гостя:



Реєстрація:

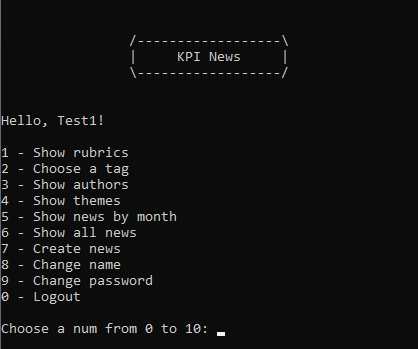


Логін:



Після реєстрації або логіну одразу перекидає в меню Автора.

Меню Автора:



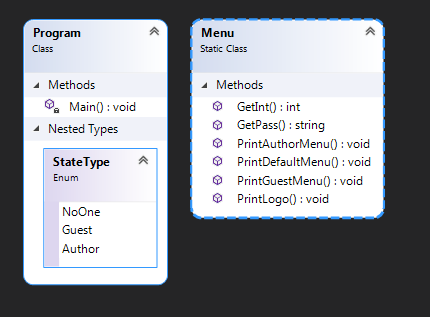
Усі пункти меню будуть розглянуті в розділі «Тестування ПЗ».

1. Статичний клас Menu – набор допоміжних функцій для інтерфейсу користувача:
   1. public GetInt() – Функція запитання у користувача цілого числа. Якщо введений рядок не є цілим числом, то повертає -1. Функція написана спеціально для коректного функціонування конструкцій switch – інтерфейсу користувача.
   2. public GetPass() – Функція запитання рядка, яка перехоплює символи, які вводить користувач, та замінює їх на \* у консолі.
   3. public PrintAuthorMenu() – Функція виводу в консоль меню для Автора.
   4. public PrintDefaultMenu() – Функція виводу в консоль головного меню.
   5. public PrintGuestMenu() – Функція виводу в консоль меню Гостя.
   6. public PrintLogo() – Функція виводу логотипу сайту в консоль.

Перерахування:

1. StateType:
   1. NoOne.
   2. Guest.
   3. Author.

### 2.2.2. Діаграма класів



# РОЗДІЛ 3. Тестування ПЗ

## 3.1. Методика тестування

Методика тестування розробленого ПЗ досить проста. Потрібно пройтись по всіх пунктах усіх меню і для кожного пункту протестувати 3 різних варіанта введення: очікуваний, гарантовано неправильний та залишити рядок порожнім.

## 3.2. План тестування

Використовуючи методику з пункту 3.1 виконуємо наступний план дій:

1. Зареєструватися.
2. Вийти у головне меню та увійти в уже існуючий обліковий запис.
3. Закрити програму, відкрити її знову та спробувати зареєструватися із вже існуючим логіном – повинне з’явитися повідомлення що «Ви вже зареєстровані».
4. Увійти в уже існуючий обліковий запис.
5. Пройти по перших 6 пунктах меню та переконатися що новин ще немає.
6. Змінити собі ім’я та пароль (пункти меню 8 та 9) на пустий рядок, на випадковий набір символів та наприклад на своє ім’я.
7. Створити новини для 1, 2 та 3 рубрики 6 тематики.
8. Вийти в головне меню та створити новий обліковий запис.
9. Створити новини для 1-5 тематики 4 рубрики.
10. Переглянути новини:
    1. За кожною рубрикою.
    2. По обраному існуючому тегу та по неіснуючому тегу.
    3. За автором.
    4. За місяцем.
    5. Всі новини.
11. Вийти в головне меню.
12. Увійти як гість.
13. Ще раз виконати дію номер 10 цього плану.
14. Вийти з програми.
15. Знову зайти в програму.
16. Повторити дії з 10 по 14.

## 3.3. Результати тестування ПЗ

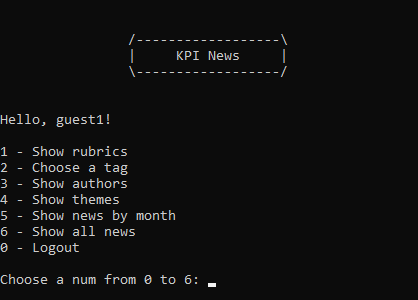
Скріншоти роботи програми (основні меню):

Головне меню:

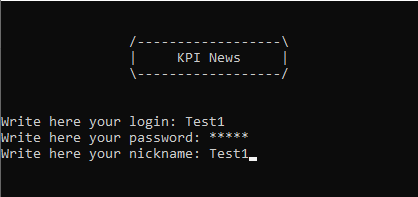


Вхід як гість одразу перекидає у меню Гостя.

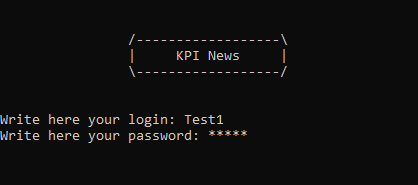
Меню Гостя:



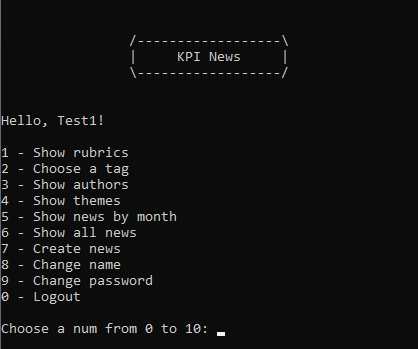
Реєстрація:



Логін:



Після реєстрації або логіну одразу перекидає в меню Автора.  
Меню Автора:



1. Рубрики:



2. Пошук по тегам:





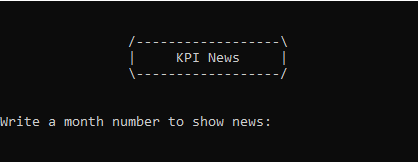
3. Автори:



4. Тематики:



5. За період:





6. Всі новини:



Приклад новини:



Відео результатів тестування у переліку посилань.

# ВИСНОВКИ

В процесі виконання індивідуального завдання КР були узагальнені знання з дисципліни «Програмування-2. Алгоритми та структури даних», а також були вивчені 2 нові теми поза програмою – Серіалізація та Хешування. Серіалізація застосована в ПЗ для збереження та подальшого відновлення попередньої сесії роботи програми. Хешування застосоване для більш безпечного зберігання даних користувачів, тому що замість паролів в ПЗ зберігаються хеші від паролів.

Серіалізований екземпляр класу Site на диск після тесту зайняв 4 кілобайти. Загрузка оперативної пам’яті під час роботи програми досягає не більше ніж 10 мегабайт.

Тестування ПЗ проводилося за власною методикою та за власним планом дій. Результати тестування підтверджують те, що ПЗ було реалізовано згідно з індивідуальним завданням КР.

Діаграми класів реалізовані в середовищі Visual Studio 2019 Community Edition за допомогою додатка до середовища – Class Designer.

ПЗ реалізоване в середовищі Visual Studio 2019 Community Edition на мові C#.

**Перелік посилань**

1. Програмний код [Електронний ресурс] – [github.com](https://github.com/feedblackg44/cource_prog).
2. Результати тестування ПЗ [Електронний ресурс] – [youtube.com](https://youtu.be/dZOECAbssFw).
3. [Електронний ресурс] – <metanit.com>.
4. Microsoft C# documentation [Електронний ресурс] – [docs.microsoft.com](https://docs.microsoft.com/en-us/dotnet/csharp/).
5. .NET Core 3.1 [Електронний ресурс] – [wikipedia.com](https://ru.wikipedia.org/wiki/.NET_Core).
6. С # 5.0 Кишеньковий довідник Д.Албахарі.
7. Schildt C# 4.0 The Complete Reference.

1. 1 Зауваження! – Рубрики, тематики та теги для тематик генеруються у конструкторі сайту. [↑](#footnote-ref-1)
2. Хеш отримується за допомогою однієї з найпопулярніших хеш-функцій – MD5. [↑](#footnote-ref-2)
3. Серіалізація - процес перетворення будь-якого об'єкта в потік байтів. [↑](#footnote-ref-3)