

ХМЕЛЬНИЦЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ  
Кафедра інженерії програмного забезпечення

ЛАБОРАТОРНА РОБОТА №2

з дисципліни «Вебтехнології»

на тему: «Додавання контролера, представлень та моделей до проєкту.  
Використання Create, Read, Update та Delete (CRUD)»

ЛРІПЗ.2201101.01.08.ЛР  
(шифр, назва)

**Виконав:**

Студент 3 курсу, група	ПЗ-22-1	_____	Владислав КВІТ
	(шифр)	(підпис)	(ініціали, прізвище)

**Перевірив:**

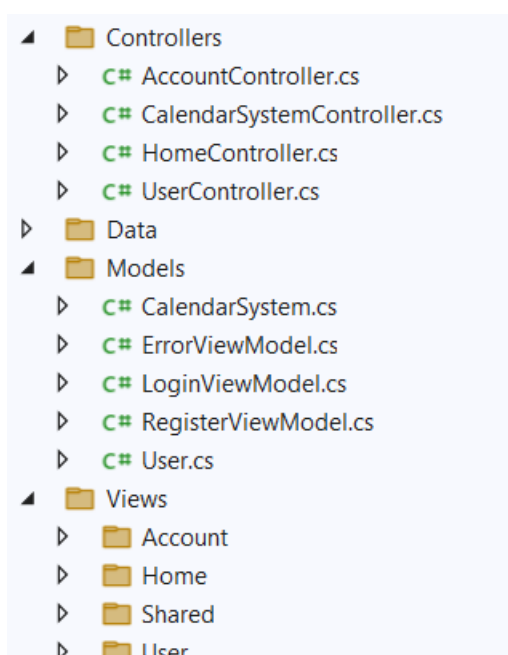
_____	Оксана ЯШИНА
(підпис)	(ініціали, прізвище)

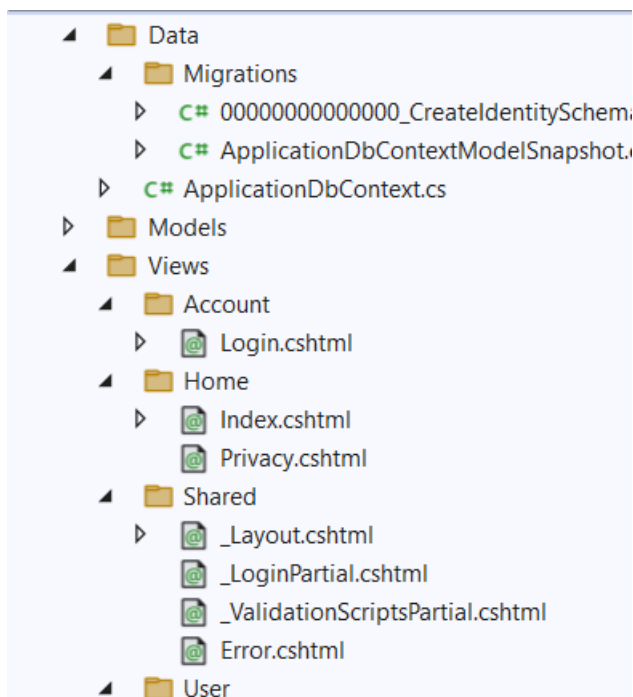
**Мета роботи:** здійснити додавання необхідних контролерів, представлень та моделей до програми MVC ASP.NET Core.

## ЗВІТ

Завдання: Згідно із тематикою курсового проекту додати всі необхідні контролери, представлення та моделі.

На рисунках нижче подано необхідні контролери, представлення та моделі та інше.





### Контрольні запитання

1. Яким є об'єкт Model?
  2. Що таке модель?
  3. Що таке представлення?
  4. Які способи додавання контролерів існують?
  5. Чого не повинні робити шаблони представлень?
  6. Як створюються шаблони представлення?
  7. Як змінити елементи у представленні?
  8. Що таке Словник ViewData?
  9. Які способи додавання моделей існують?
  10. Що таке CRUD?
1. У контексті веб-додатків, модель (Model) є компонентом патерну MVC (Model-View-Controller), який відповідає за дані додатку, їхню структуру та логіку. Об'єкт моделі представляє дані та правила, за якими ці дані можуть бути змінені або опрацьовані.
  2. Це частина додатку, яка відповідає за доступ до даних та бізнес-логіку. Вона зберігає стан даних і забезпечує їхню обробку, часто взаємодіючи з базою даних.

3. Це частина програми, яка відповідає за відображення даних користувачу. Представлення приймає інформацію від контролера та відображає її в зрозумілому для користувача вигляді (зазвичай HTML).

4. Способи додавання контролерів:

-Використання звичайних класів контролерів, що наслідують базовий клас `Controller`.

-Додавання контролерів через атрибути маршрутизації для контролерів (attribute routing).

-Використання IoC-контейнера для впровадження залежностей в контролери.

5. Шаблони представлень не повинні містити бізнес-логіку або обробку даних. Вони відповідають тільки за відображення даних, наданих контролером.

6. Шаблони створюються як файли з розширенням `.cshtml` у разі використання `Razor` (для `ASP.NET`) або інших відповідних шаблонізаторів. Вони можуть містити HTML-код, а також інструкції для відображення змінних і даних із моделі.

7. Можна змінити елементи представлення шляхом редагування HTML-коду у шаблоні або використання додаткового коду `Razor` для відображення різних елементів залежно від умов.

8. Це об'єкт, що дозволяє передавати дані від контролера до представлення у вигляді пари "ключ-значення". Це один із способів передавання даних, поряд з `ViewBag` та прямою передачею моделі.

9. Способи додавання моделей:

- Моделі можуть додаватися через стандартні класи, що представляють дані.

- Використання Entity Framework або іншої ORM для автоматичного створення моделей на основі бази даних.

- Впровадження моделей через DI-контейнер (dependency injection).

10.CRUD: Це акронім, що позначає чотири основні операції з даними:

- Create (створення),

- Read (читання),

- Update (оновлення),

- Delete (видалення).