

ДОДАТОК А

Слайди презентації



Програмна система для онлайн-навчання кулінарному мистецтву



Нех Тарас Віталійович, ПЗПІ-21-1
Керівник: доц. Афанасьєва І.В.

16 червня 2025

Мета роботи

Метою розробки є створення зручної та функціональної веб-платформи для онлайн-навчання кулінарному мистецтву, яка враховує потреби як учнів, так і викладачів.

Актуальність роботи зумовлена зростанням інтересу до дистанційного навчання та браком якісного україномовного контенту у сфері кулінарії. Запропоноване рішення спрямоване на покращення доступності та ефективності освітнього процесу онлайн.



Аналіз проблеми (аналіз існуючих рішень)

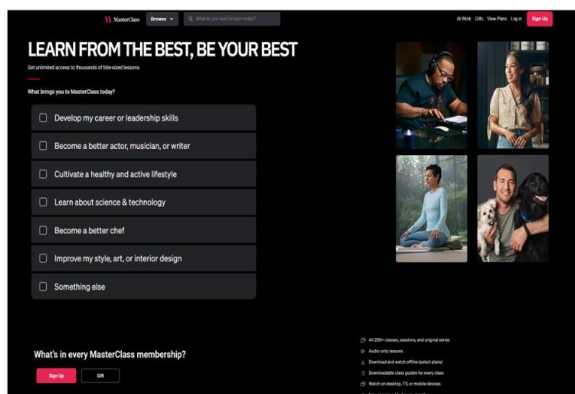
BBC Good Food



- Один із найвідоміших кулінарних сайтів.
- Пропонує тисячі рецептів, інструкцій, відео та порад.
- Орієнтований на самостійне приготування, без курсової структури.



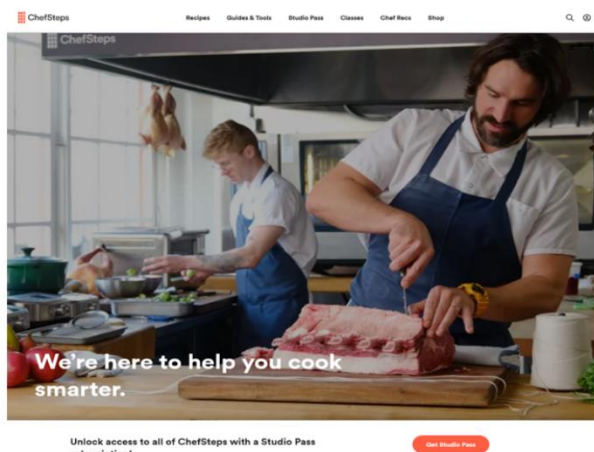
MasterClass



- Преміум-платформа з відеоуроками від зірок кулінарії.
- Висока якість контенту, модульна структура, наявність робочих зошитів.
- Підходить більше для натхнення, ніж для системного навчання.



ChefSteps



- Орієнтований на просунутих користувачів і сучасні техніки приготування.
- Велика увага до технологій та точності.
- Має вузьку спеціалізацію та прив'язку до власних продуктів (наприклад, Joule).



Визначені переваги

- Якісний контент: відео з професійною зйомкою, покрокові інструкції.
- Велика база рецептів: безліч варіантів приготування на будь-який смак.
- Інтуїтивна навігація: зручність користування навіть без технічної підготовки.
- Мотивація через відомих викладачів (у MasterClass): залучення користувачів до навчання.




Виявлені недоліки


- ✘ Відсутність структурованого навчання (BBC Good Food)
- ✘ Немає перевірки знань і прогресу (усі платформи)
- ✘ Відсутність інтерактивності й зворотного зв'язку з викладачем
- ✘ Недоступність українською мовою
- ✘ Обмежена або платна доступність контенту (MasterClass, ChefSteps)





Постановка задачі


 Формулювання проблеми:


Попри велику кількість кулінарних ресурсів, на ринку відсутні структуровані україномовні онлайн-курси, які б поєднували послідовне навчання, перевірку знань, інтерактивність та зворотний зв'язок із викладачами. Існуючі рішення переважно не мають освітньої логіки або доступні лише англійською мовою.


 Необхідно провести наступну роботу:

 **Аналіз предметної області.** Провести дослідження сфери онлайн-навчання кулінарії та вивчення наявних конкурентних рішень.

 **Моделювання системи.** Виконати концептуальне, інфологічне та логічне моделювання з використанням UML-діаграм, діаграм станів та ER-діаграми бази даних.

 **Реалізація програмної системи.** Розробити веб застосунок з використанням Razor Pages для клієнтської частини та ASP.NET Core (C#) для серверної логіки.

 **Створення бази даних.** Реалізувати реляційну базу даних у Microsoft SQL Server із застосуванням ORM Entity Framework Core.

 **Тестування системи.** Провести перевірку функціональних можливостей, зручності інтерфейсу, стабільності роботи та відповідності заданим вимогам.



Опис системи

У результаті очікується створення повноцінної системи онлайн-навчання кулінарному мистецтву. Основна мета полягає у забезпеченні користувачів зручним, інтерактивним та ефективним інструментом для здобуття знань та навичок у галузі кулінарії.

🎯 Очікувані результати:

- Створення веб-застосунку для онлайн-навчання кулінарії з повноцінною структурою навчальних курсів
- Реалізація функцій реєстрації, проходження уроків, тестування, комунікації з викладачем
- Забезпечення зручного і привабливого інтерфейсу з анімаціями та адаптивністю
- Формування бази курсів, рецептів і відеоматеріалів, що підтримують освітню логіку
- Впровадження механізмів оцінювання та збереження прогресу користувача



Вибір технологій розробки

◆ Backend:

- **ASP.NET Core (C#)** – для створення RESTful API та реалізації логіки застосунку
- **Entity Framework Core** – для роботи з базою даних
- **Microsoft SQL Server** – як основна система управління базами даних

◆ Frontend:

- **Razor Pages** – для створення динамічних веб-сторінок
- **HTML/CSS, JavaScript** – для розмітки, стилізації та інтерактивних елементів
- **Bootstrap** – для швидкої розробки адаптивного інтерфейсу

◆ Анімації та UI:

- **CSS-анімації** – для реалізації ефектів появи, наведення та масштабування
- **(Планується) Framer Motion / інші бібліотеки** – для складніших анімацій

◆ Інше:

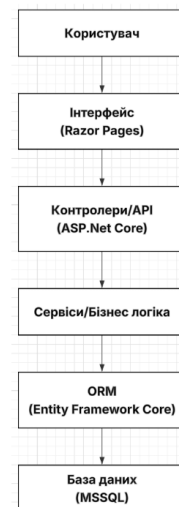
- **Git / GitHub** – для контролю версій
- **Visual Studio** – як основне середовище розробки



Архітектура створеного програмного забезпечення

◆ Ключові компоненти:

- **Інтерфейс користувача (Razor Pages):**
Відповідає за взаємодію з користувачем, відображення курсів, матеріалів, тестів, результатів.
- **Контролери ASP.NET Core:**
Обробляють HTTP-запити, координують дії між клієнтом, сервісами та базою даних.
- **Сервіси (Business Logic Layer):**
Реалізують основну логіку: запис на курс, обробку тестів, збереження результатів тощо.
- **Entity Framework Core:**
Забезпечує доступ до бази даних через ORM, спрощує роботу з SQL Server.
- **SQL Server:**
Містить таблиці для курсів, користувачів, матеріалів, тестів, результатів, коментарів і чату.

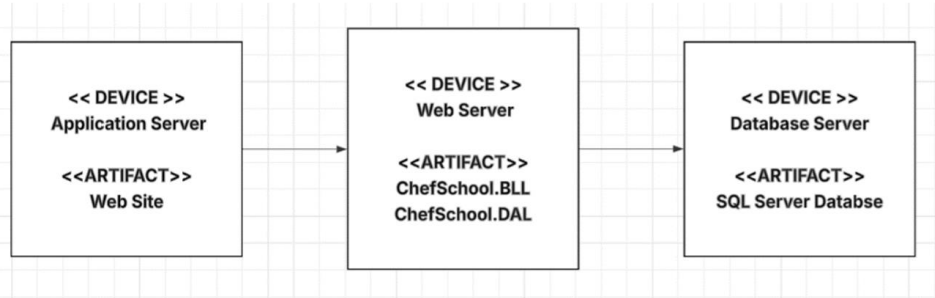


Опис програмного забезпечення, що було використано у дослідженні

◆ Процес розробки:

Процес розробки було розпочато з аналізу вимог користувача та визначення основних функцій програмної системи. Після цього було створено початковий макет інтерфейсу та продумано логіку взаємодії користувача з системою. Далі почалася реалізація серверної частини за допомогою ASP.NET Core, включаючи налаштування бази даних, створення API для обробки запитів та впровадження авторизації й реєстрації користувачів. Паралельно тривала розробка клієнтської частини з використанням Razor Pages, де оформлювались сторінки курсів, тестів, профілю користувача та чатів. Після поєднання усіх компонентів було проведено ручне тестування функціональності, перевірено стабільність роботи застосунку у браузері та адаптивність інтерфейсу.

У процесі реалізації використовувалась Visual Studio як основне середовище розробки, SourceTree для контролю версій з використанням Git, а також SQL Server Management Studio для тестування запитів та роботи з базою даних MSSQL.

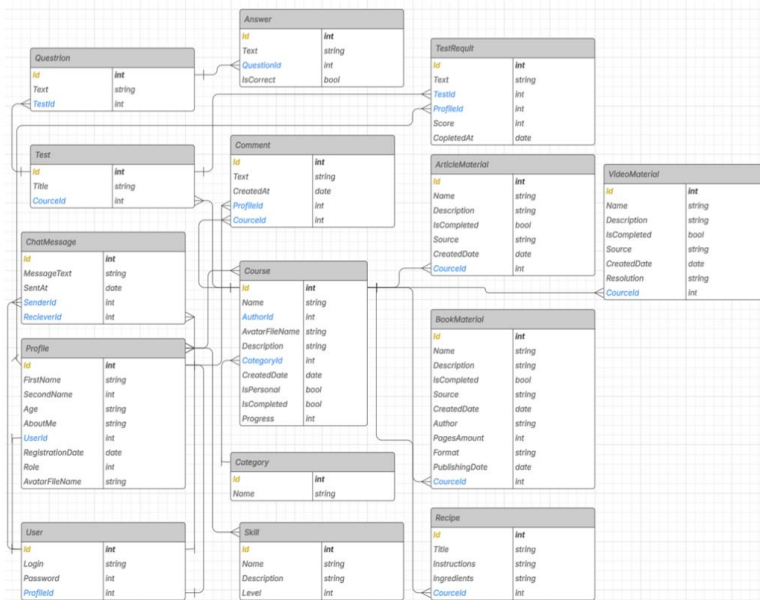


Діаграма розгортання ілюструє, як клієнтська частина взаємодіє з серверною частиною і базою даних у хмарному середовищі. Ця діаграма дозволяє побачити, як саме інформація буде передаватися між компонентами системи і як буде забезпечено доступ до даних. Діаграма розгортання є корисним інструментом для розуміння, як функціонує система в цілому і як її компоненти взаємодіють у фізичному середовищі.



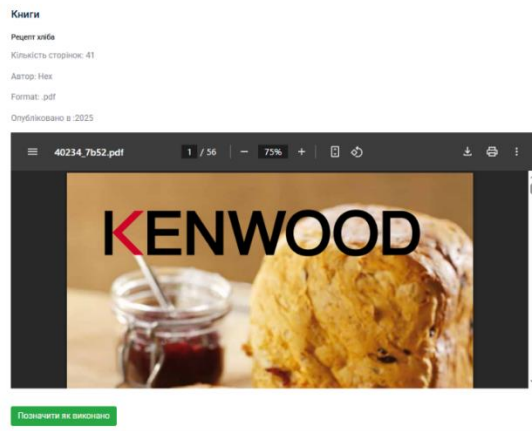
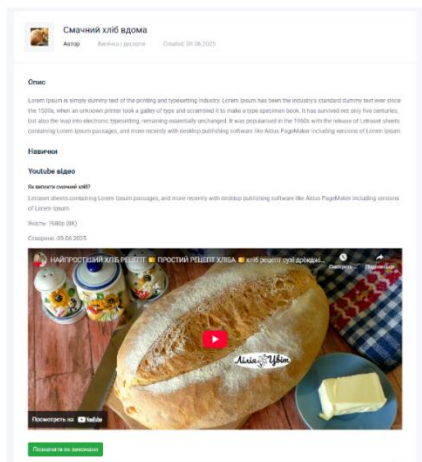
ER-діаграма

Розроблена модель бази даних дозволяє ефективно зберігати навчальний контент, результати проходження тестів та комунікацію між користувачами



Приклад реалізації

Реалізація моделі курсів із мультимедійним наповненням



Реалізація моделі курсів із мультимедійним наповненням

```
public abstract class BaseMaterial
{
    public int Id { get; set; }
    public string Title { get; set; }
    public DateTime CreatedAt { get; set; }
    public int CourseId { get; set; }
    public Course Course { get; set; }
}

public class VideoMaterial : BaseMaterial
{
    public string VideoUrl { get; set; }
    public TimeSpan Duration { get; set; }
}

public class BookMaterial : BaseMaterial
{
    public string FilePath { get; set; }
    public string Description { get; set; }
}

public class ArticleMaterial : BaseMaterial
{
    public string Content { get; set; }
}
```

```
foreach (var video in @Model.Videos)
{
    <h6>@video.Name</h6>
    <p>@video.Description</p>
    if (!string.IsNullOrEmpty(@video.Resolution))
    {
        <p>Resolution: @video.Resolution</p>
    }
    <p>Created :@video.CreatedDate.ToShortDateString()</p>
    if (Model.IsPersonal)
    {
        <iframe class="w-100" height="500"
src="https://www.youtube.com/embed/@video.Source" title="YouTube video
player" frameborder="0" allow="accelerometer; autoplay; clipboard-write;
encrypted-media; gyroscope; picture-in-picture" allowfullscreen></iframe>
    }
    if (Model.IsPersonal && !video.IsCompleted)
    {
        <a style="margin-top: 15px" asp-
action="CompleteMaterial" asp-route-courseId="@Model.Id" asp-route-
materialId="@video.Id" asp-route-materialTypeId="1" class="btn btn-success
w-30">Mark as Completed</a>
    }
    <hr />
}
```

Механізм проходження тестування з підрахунком результатів

```

public class Test
{
    public int Id { get; set; }
    public string Title { get; set; }
    public int CourseId { get; set; }
    public List<Question> Questions { get; set; }
}

public int CalculateScore(Test test, Dictionary<int, int>
userAnswers)
{
    int score = 0;
    foreach (var question in test.Questions)
    {
        if (userAnswers.TryGetValue(question.Id, out int answerId))
        {
            if (question.Answers.Any(a => a.Id == answerId &&
a.IsCorrect))
                score++;
        }
    }
    return score;
}

```



Answer the question

Питання 1: Який інгредієнт є основою класичного соусу бешамель?

а) Томатна паста

б) Боршно та масло

в) Оцет

г) Вино

19

Реалізація внутрішнього чату між учнями та викладачами

Ваші чати

Викладач Ірина
Останнє повідомлення: "Добре, спробуйте..."

Шеф Андрій
Останнє повідомлення: "Дякуємо за відео..."

Викладач Марія
Останнє повідомлення: "Коли будете готові..."

Чат із викладачем Іриною

Учень: Доброго дня! Я не зрозумів, як правильно приготувати соус. 10:02

Викладач: Вітаю! Спочатку розтопіть масло, додайте борошно і перемішайте до однорідності. 10:04

Учень: Дякую, спробую ще раз. 10:05

```

public class ChatMessage
{
    public int Id { get; set; }
    public int SenderId { get; set; }
    public int ReceiverId { get; set; }
    public string MessageText { get; set; }
    public DateTime SentAt { get; set; }
}

```

```

public async Task SendMessageAsync(int senderId, int receiverId, string
message)
{
    var chatMessage = new ChatMessage
    {
        SenderId = senderId,
        ReceiverId = receiverId,
        MessageText = message,
        SentAt = DateTime.UtcNow
    };
    _context.ChatMessages.Add(chatMessage);
    await _context.SaveChangesAsync();
}

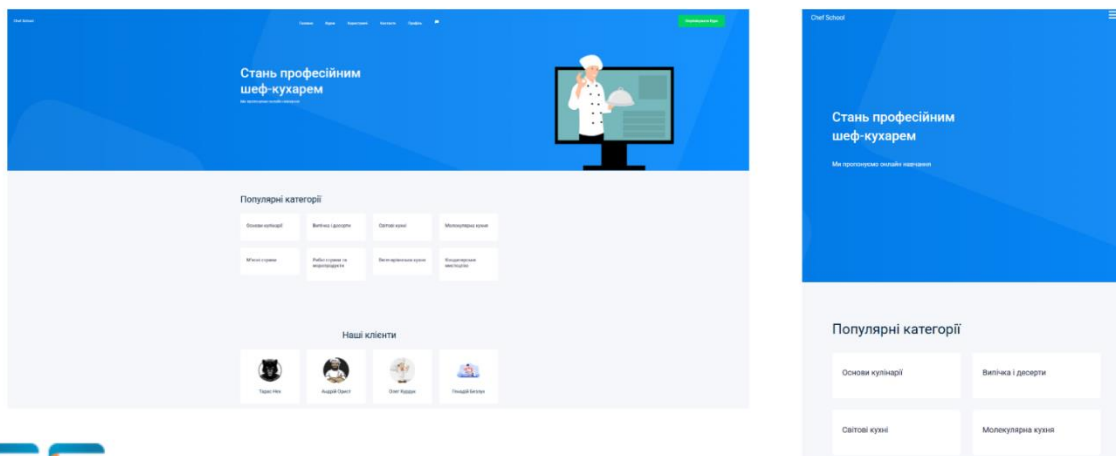
```



20

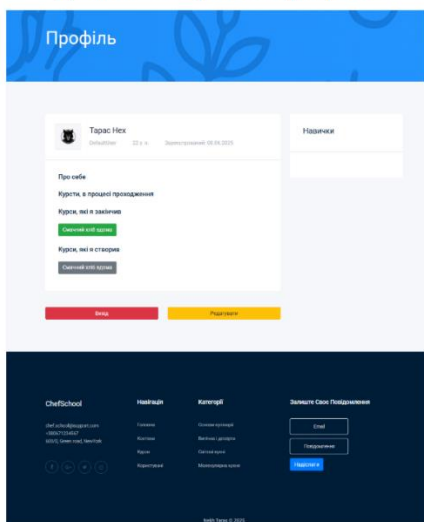
Інтерфейс користувача

Дизайн головної сторінки

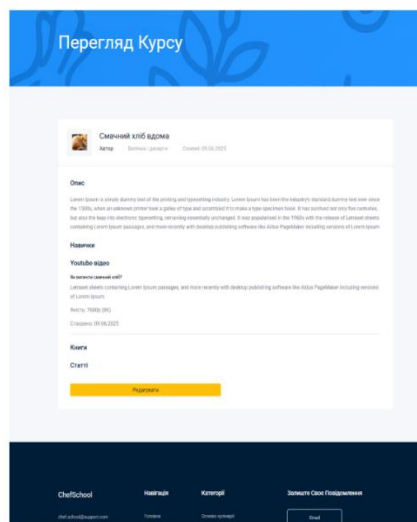


21

Дизайн сторінки профілю

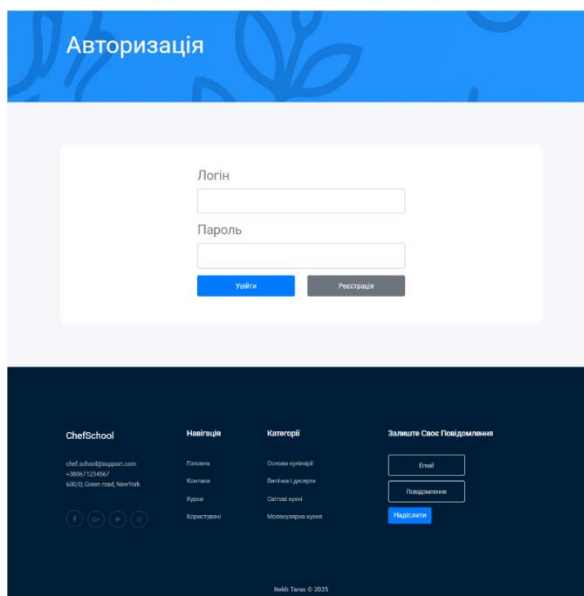


Дизайн сторінки курсу



22

Дизайн сторінки авторизації



Тестування

Тест-план

№	Назва функції	Тип тестування	Ціль	Статус
1	Реєстрація користувача	Функціональне	Перевірка створення нового облікового запису	Пройдено
2	Проходження тесту	Функціональне	Перевірка відповіді на запитання та результат	Пройдено
3	Чат з викладачем	Функціональне/UX	Перевірка надсилання та збереження повідомлень	Пройдено

Тест-кейс №1

Інформація про тест-кейс			
Ідентифікатор тесту:	Тест-кейс №1		
Опис функції:	Ресстрація нового користувача		
Власник тесту:	Нех Тарас Віталійович		
Дата створення:	25.05.2025		
Мета тесту:	Перевірити працездатності форми ресстрації		
Передумова			
№	Опис випадку	Очікуваний результат	Висновок
1	Відкрити веб-застосунок	Користувач має доступ до сайту, який відкритий	Пройдено
2	Користувач не має облікового запису та хоче зареєструватися	Перенаправлено на сторінку авторизації	Пройдено
Ресстрація			
№	Опис випадку	Очікуваний результат	Висновок
1	Відкрити сторінку ресстрації	З'являється форма для ресстрації	Пройдено
2	Ввести ім'я користувача	Ім'я введено без помилок	Пройдено
3	Ввести пароль і підтвердження	Обидва поля заповнені і паролі збігаються	Пройдено
4	Натиснути кнопку для ресстрації	Користувач успішно зареєструвався в системі	Пройдено
5	Користувач автоматично перенаправляється в профіль	Профіль завантажується, користувача аутентифіковано	Пройдено
Результати тестування			
Тестувальник: Нех Т. В.	Дата прогону тесту: 25.05.2025	Результат тесту (P/F/B): ПРОЙДЕНО (P)	



Тест-кейс №2

Інформація про тест-кейс			
Ідентифікатор тесту:	Тест-кейс №2		
Опис функції:	Проходження навчального тесту		
Власник тесту:	Нех Тарас Віталійович		
Дата створення:	25.05.2025		
Мета тесту:	Перевірити, чи можна пройти тест і отримати результат		
Передумова			
№	Опис випадку	Очікуваний результат	Висновок
1	Відкрити веб-застосунок	Користувач має доступ до сайту, який відкритий	Пройдено
2	Користувач авторизований	Перенаправлено на головну сторінку	Пройдено
Проходження навчального тесту			
№	Опис випадку	Очікуваний результат	Висновок
1	Перейти до профілю	Відкривається профіль	Пройдено
2	Перейти до доступного курсу	Відкривається сторінка курсу	Пройдено
3	Натиснути на кнопку проходження тесту	Відкривається форма для проходження тесту	Пройдено
4	Обрати відповіді на всі питання	Всі запитання мають обрану відповідь	Пройдено
5	Натиснути «Завершити тест»	Тест завершується	Пройдено
6	Переглянути результат	Виведено результат тесту (наприклад: 80% правильних)	Пройдено
Результати тестування			
Тестувальник: Нех Т. В.	Дата прогону тесту: 25.05.2025	Результат тесту (P/F/B): ПРОЙДЕНО (P)	

25

Тест-кейс №3

Інформація про тест-кейс			
Ідентифікатор тесту:	Тест-кейс №3		
Опис функції:	Надсилання повідомлення в чаті		
Власник тесту:	Нех Тарас Віталійович		
Дата створення:	25.05.2025		
Мета тесту:	Перевірити працездатність чату між учнем та викладачем		
Передумова			
№	Опис випадку	Очікуваний результат	Висновок
1	Відкрити веб-застосунок	Користувач має доступ до сайту, який відкритий	Пройдено
2	Користувач авторизований	Перенаправлено на головну сторінку	Пройдено
3	Користувач рашіне спілкувався з викладачем	Викладач є у списку контактів учня	Пройдено
Користування чатом			
№	Опис випадку	Очікуваний результат	Висновок
1	Перейти до профілю	Відкривається профіль	Пройдено
2	Перейти до сторінки чату	Відкривається сторінка чату	Пройдено
3	Обрати у списку контактів викладача.	Відкривається форма чату з викладачем	Пройдено
4	Ввести текст повідомлення	Повідомлення з'являється у полі введення	Пройдено
5	Натиснути «Надіслати»	Повідомлення з'являється у списку повідомлень	Пройдено
6	Оновити сторінку	Історія повідомлень зберігається	Пройдено
7	Отримати відповідь від викладача	Відповідь з'являється у чаті	Пройдено
Результати тестування			
Тестувальник: Нех Т. В.	Дата прогону тесту: 25.05.2025	Результат тесту (P/F/B): ПРОЙДЕНО (P)	



26

Підсумки

В результаті роботи над кваліфікаційною роботою бакалавра було здійснено повноцінну розробку програмної системи для онлайн-навчання кулінарному мистецтву, що включала всі етапи – від аналізу вимог до реалізації та тестування.

Етапи, які було виконано під час роботи над проєктом:

- Досліджено сферу онлайн-навчання кулінарії та проаналізовано наявні конкурентні рішення.
- Розроблено концептуальну, інфологічну та логічну моделі із застосуванням UML-діаграм, діаграм станів і ER-діаграми бази даних.
- Спроектвано реляційну структуру бази даних у Microsoft SQL Server із використанням ORM Entity Framework Core.
- Розроблено веб застосунок з клієнтською частиною на Razor Pages та серверною логікою на ASP.NET Core (C#).
- Перевірено функціональні можливості, зручність інтерфейсу, стабільність роботи та відповідність програмного продукту визначеним вимогам.

Розроблене програмне забезпечення демонструє високу практичну цінність, адже охоплює основні потреби сучасної платформи онлайн-навчання кулінарії: від структурованих курсів до взаємодії між учнем і викладачем. Проведене тестування підтвердило стабільну роботу та зручність інтерфейсу для користувачів різного рівня підготовки.

Система може бути використана у навчальних закладах для дистанційного навчання кулінарії, приватними викладачами або онлайн-школами. Також її можна адаптувати під інші освітні напрями з мінімальними змінами в структурі контенту.

ДОДАТОК Б

Фрагменти коду програми

Метод реєстрації користувача в сервісі:

```
public async Task<BaseResponse<ClaimsIdentity>> Register(RegisterViewModel
model)
{
    var users = await _userService.GetAll();
    var user = users.Data.FirstOrDefault(u => u.Login == model.Login);

    if (user is not null)
    {
        return new BaseResponse<ClaimsIdentity>()
        {
            Info = "There is user with the same login",
            StatusCode = StatusCode.Error
        };
    }

    Profile profile = new Profile();

    user = new User()
    {
        Login = model.Login,
        Password = HashPasswordHelper.HashPassword(model.Password),
        Profile = profile
    };

    await _userService.Add(user);

    var result = AuthHelper.Authenticate(user);

    return new BaseResponse<ClaimsIdentity>()
    {
        Data = result,
        Info = "Registered",
        StatusCode = StatusCode.Ok
    };
}
```

Влаштований You-Tube відео плеєр:

```
<iframe class="w-100" height="500"
src="https://www.youtube.com/embed/@video.Source"
title="YouTube video player"
frameborder="0"
allow="accelerometer;
autoplay;
gyroscope;
picture-in-picture"
allowfullscreen
/>
```

Метод збереження файлів користувачів (фото, pdf-файли) з унікальною назвою:

```
private static string GetUniqueFileName(string fileName)
{
    fileName = Path.GetFileName(fileName);
    return Path.GetFileNameWithoutExtension(fileName)
        + "_"
        + Guid.NewGuid().ToString().Substring(0, 4)
        + Path.GetExtension(fileName);
}

private void ProccesTheContent(Course entity)
{
    if (entity.File is not null)
    {
        var uploads = Path.Combine(_hostingEnvironment.WebRootPath,
"uploads");
        entity.AvatarName = GetUniqueFileName(entity.File.FileName);
        var filePath = Path.Combine(uploads, entity.AvatarName);
        entity.File.CopyTo(new FileStream(filePath, FileMode.Create));
    }

    if (entity.Books.Count != 0)
    {
        foreach (var book in entity.Books)
        {
            if (book.BookFile is not null)
            {
                var uploads = Path.Combine(_hostingEnvironment.WebRootPath,
"uploads");

                book.Source = GetUniqueFileName(book.BookFile.FileName);
                book.Format = Path.GetExtension(book.Source);
                var filePath = Path.Combine(uploads, book.Source);
                book.BookFile.CopyTo(new FileStream(filePath,
FileMode.Create));
            }
        }
    }

    if (entity.Videos.Count != 0)
    {
        foreach(var video in entity.Videos)
        {
            if (video.Source.Contains("v="))
            {
                video.Source = video.Source.Split("v=")[1];
            }
        }
    }
}
```

Код статичного допоміжного класу HashPasswordHelper:

```
namespace Core.Static
{
    public static class HashPasswordHelper
    {
        public static string HashPassword(string password)
        {
            using (var sha256 = SHA256.Create())
            {
                var hashedBytes =
sha256.ComputeHash(Encoding.UTF8.GetBytes(password));
                var hash = BitConverter.ToString(hashedBytes).Replace("-",
"" ).ToLower();

                return hash;
            }
        }
    }
}
```

Методи ConfigureServices та Configure для початкової конфігурації проєкту:

```
public void ConfigureServices(IServiceCollection services)
{
    services.AddControllersWithViews();

    string connectionString =
Configuration.GetConnectionString("DefaultConnection");

    services.AddDbContext<AppDbContext>(x => x.UseSqlServer(
        connectionString
    ));

    services.AddAuthentication(CookieAuthenticationDefaults.AuthenticationScheme)
        .AddCookie(options => {
            options.LoginPath = new
Microsoft.AspNetCore.Http.PathString("/Account/Login");
            options.AccessDeniedPath = new
Microsoft.AspNetCore.Http.PathString("/Account/Login");
        });

    services.AddScoped<IBaseRepository<Course>, CourseRepository>();
    services.AddScoped<IBaseRepository<User>, UserRepository>();
    services.AddScoped<IBaseRepository<Chat>, ChatRepository>();
    services.AddScoped<IService<Course>, CourseService>();
    services.AddScoped<IService<User>, UserService>();
    services.AddScoped<IAccountService, AccountService>();
    services.AddScoped<IService<Chat>, ChatService>();
}
```

```
// This method gets called by the runtime. Use this method to configure the
// HTTP request pipeline.
public void Configure(IApplicationBuilder app, IWebHostEnvironment env)
{
    if (env.IsDevelopment())
    {
        app.UseDeveloperExceptionPage();
    }
    else
    {
        app.UseExceptionHandler("/Home/Error");
        // The default HSTS value is 30 days. You may want to change this
        // for production scenarios, see https://aka.ms/aspnetcore-hsts.
        app.UseHsts();
    }
    app.UseHttpsRedirection();
    app.UseStaticFiles();

    app.UseRouting();

    app.UseAuthorization();
    app.UseAuthentication();

    app.UseEndpoints(endpoints =>
    {
        endpoints.MapControllerRoute(
            name: "default",
            pattern: "{controller=Home}/{action=Index}/{id?}");
    });
}
```

Java-Script скрипт для автоматичного гортання історії чату в кінець при завантаженні сторінки:

```
@section Scripts {
    <script>
        window.addEventListener('DOMContentLoaded', (event) => {
            var container =
document.getElementById('messagesContainer');
            if (container) {
                container.scrollTop = container.scrollHeight;
            }
        });
    </script>
}
```

ДОДАТОК В

Тест-план та тест-кейси (тест-комплект)

Таблиця А.1 - Тест-план

№	Назва функції	Тип тестування	Ціль	Статус
1	Реєстрація користувача	Функціональне	Перевірка створення нового облікового запису	Пройдено
2	Проходження тесту	Функціональне	Перевірка відповіді на запитання та результат	Пройдено
3	Чат з викладачем	Функціональне/UX	Перевірка надсилання та збереження повідомлень	Пройдено

Таблиця А.2 - Тест-кейс 1. Реєстрація нового користувача

Інформація про тест-кейс			
Ідентифікатор тесту:	Тест-кейс №1		
Опис функції:	Реєстрація нового користувача		
Власник тесту:	Нех Тарас Віталійович		
Дата створення:	25.05.2025		
Мета тесту:	Перевірити працездатності форми реєстрації		
Передумова			
№	Опис випадку	Очікуваний результат	Висновок
1	Відкрити веб-застосунок	Користувач має доступ до сайту, який відкритий	Пройдено
2	Користувач не має облікового запису та хоче зареєструватися	Перенаправлено на сторінку авторизації	Пройдено
Реєстрація			
№	Опис випадку	Очікуваний результат	Висновок
1	Відкрити сторінку реєстрації	З'являється форма для реєстрації	Пройдено
2	Ввести ім'я користувача	Ім'я введено без помилок	Пройдено
3	Ввести пароль і підтвердження	Обидва поля заповнені і паролі збігаються	Пройдено
4	Натиснути кнопку для реєстрації	Користувач успішно зареєструвався в системі	Пройдено
5	Користувач автоматично перенаправляється в профіль	Профіль завантажується, користувача аутентифіковано	Пройдено
Результати тестування			
Тестувальник: Нех Т. В.	Дата прогону тесту: 25.05.2025	Результат тесту (P/F/V): ПРОЙДЕНО (P)	

Таблиця А.3 - Тест-кейс 2. Проходження навчального тесту

Інформація про тест-кейс			
Ідентифікатор тесту:	Тест-кейс №2		
Опис функції:	Проходження навчального тесту		
Власник тесту:	Нех Тарас Віталійович		
Дата створення:	25.05.2025		
Мета тесту:	Перевірити, чи можна пройти тест і отримати результат		
Передумова			
№	Опис випадку	Очікуваний результат	Висновок
1	Відкрити веб-застосунок	Користувач має доступ до сайту, який відкритий	Пройдено
2	Користувач авторизований	Перенаправлено на головну сторінку	Пройдено
Проходження навчального тесту			
№	Опис випадку	Очікуваний результат	Висновок
1	Перейти до профілю	Відкривається профіль	Пройдено
2	Перейти до доступного курсу	Відкривається сторінка курсу	Пройдено
3	Натиснути на кнопку проходження тесту	Відкривається форма для проходження тесту	Пройдено
4	Обрати відповіді на всі питання	Всі запитання мають обрану відповідь	Пройдено
5	Натиснути «Завершити тест»	Тест завершається	Пройдено
6	Переглянути результат	Виведено результат тесту (наприклад: 80% правильних)	Пройдено
Результати тестування			
Тестувальник: Нех Т. В.	Дата прогону тесту: 25.05.2025	Результат тесту (P/F/V): ПРОЙДЕНО (P)	

Таблиця А.4 - Тест-кейс 3. Надсилання повідомлення в чаті



Інформація про тест-кейс			
Ідентифікатор тесту:	Тест-кейс №3		
Опис функції:	Надсилання повідомлення в чаті		
Власник тесту:	Нех Тарас Віталійович		
Дата створення:	25.05.2025		
Мета тесту:	Перевірити працездатність чату між учнем та викладачем		
Передумова			
№	Опис випадку	Очікуваний результат	Висновок
1	Відкрити веб-застосунок	Користувач має доступ до сайту, який відкритий	Пройдено
2	Користувач авторизований	Перенаправлено на головну сторінку	Пройдено

Кінець таблиці А.4

Передумова			
3	Користувач рашіне спілкувався з викладачем	Викладач є у списку контактів учня	Пройдено
Користування чатом			
№	Опис випадку	Очікуваний результат	Висновок
1	Перейти до профілю	Відкривається профіль	Пройдено
2	Перейти до сторінки чату	Відкривається сторінка чату	Пройдено
3	Обрати у списку контактів викладача.	Відкривається форма чату з викладачем	Пройдено
4	Ввести текст повідомлення	Повідомлення з'являється у полі введення	Пройдено
5	Натиснути «Надіслати»	Повідомлення з'являється у списку повідомлень	Пройдено
6	Оновити сторінку	Історія повідомлень зберігається	Пройдено
7	Отримати відповідь від викладача	Відповідь з'являється у чаті	Пройдено
Результати тестування			
Тестувальник: Нех Т. В.		Дата прогону тесту: 25.05.2025	Результат тесту (P/F/V): ПРОЙДЕНО (P)

ДОДАТОК Г

Звіт результатів перевірки на унікальність тексту в базі ХНУРЕ

StrikePlagiarism.com  Дата звіту 6/6/2025
Дата редагування ---  Звіт не був оцінений

Звіт подібності

метадані

Назва організації
Kharkiv National University of Radio Electronics
Заголовок
2025_Б_ПІ_ПЗПІ-21-1_Нех_Т_В_скорочений
Автор Науковий керівник / Експерт
Нех Тарас Віталійович Євген Кардаш
підрозділ
каф. ПІ

Обсяг знайдених подібностей

Коефіцієнт подібності визначає, який відсоток тексту по відношенню до загального обсягу тексту було знайдено в різних джерелах. Зверніть увагу, що високі значення коефіцієнта не автоматично означають плагіат. Звіт має аналізувати компетентна / уповноважена особа.



Тривога

У цьому розділі ви знайдете інформацію щодо текстових спотворень. Ці спотворення в тексті можуть говорити про **МОЖЛИВІ** маніпуляції в тексті. Спотворення в тексті можуть мати навмисний характер, але частіше характер технічних помилок при конвертації документа та його збереженні, тому ми рекомендуємо вам підходити до аналізу цього модуля відповідально. У разі виникнення запитань, просимо звертатися до нашої служби підтримки.

Заміна букв		1
Інтервали		0
Мікропробіли		0
Білі знаки		0
Парафрази (SmartMarks)		19

Подібності за списком джерел

Нижче наведений список джерел. В цьому списку є джерела із різних баз даних. Колір тексту означає в якому джерелі він був знайдений. Ці джерела і значення Коефіцієнту Подібності не відображають прямого плагіату. Необхідно відкрити кожне джерело і проаналізувати зміст і правильність оформлення джерела.

10 найдовших фраз		Колір тексту
ПОРЯДКОВИЙ НОМЕР	НАЗВА ТА АДРЕСА ДЖЕРЕЛА URL (НАЗВА БАЗИ)	КІЛЬКІСТЬ ІДЕНТИЧНИХ СЛІВ (ФРАГМЕНТІВ)
1	https://uricsc.github.io/courses/_sources/assets/lecture/heaps.md	22 0.25 %
2	ВКР_Крупка 2/5/2025 State University of Trade and Economics (Кафедра комп'ютерних наук та інформаційних систем)	21 0.24 %
3	https://dut.edu.ua/repozitonii/pz/2021/%D0%9F%D0%94-42/%D0%94%D0%B0%D1%86%D0%B5%D0%BD%D0%BA%D0%BE.%D0%9C.%D0%90.pdf	18 0.21 %

4	https://dut.edu.ua/repozytorii/ipz/2021/%D0%9F%D0%94-42/%D0%94%D0%B0%D1%86%D0%B5%D0%BD%D0%BA%D0%BE.%D0%9C.%D0%90.pdf	16 0.18 %
5	ВЕБЗАСТОСУНОК ДЛЯ ВЗАЄМОДІЇ САДІВНИКІВ ТА ДОМОВЛАСНИКІВ. СЕРВЕРНА ЧАСТИНА 6/5/2024 Taras Shevchenko National University of Kyiv (Факультет комп'ютерних наук та кібернетики)	15 0.17 %
6	https://metod.vntu.edu.ua/getfile.php/10190.pdf	14 0.16 %
7	https://openarchive.nure.ua/bitstreams/7c3f3dfe-e12e-465e-94fb-558e4cf87424/download	12 0.14 %
8	БР_Прут_ІПЗ_4.02 6/3/2025 State University of Intellectual Technologies and Communications (Кафедра інженерії програмного забезпечення)	12 0.14 %
9	Konchatnyi_bac_rob.docx 6/11/2021 Sumy State University (Кафедра інформаційних технологій)	12 0.14 %
10	https://ela.kpi.ua/bitstream/123456789/60238/1/Smyrnov_bakalavr.pdf	11 0.13 %
з бази даних RefBooks (0.00 %)		
ПОРЯДКОВИЙ НОМЕР	ЗАГОЛОВОК	КІЛЬКІСТЬ ІДЕНТИЧНИХ СЛІВ (ФРАГМЕНТІВ)
з домашньої бази даних (0.00 %)		
ПОРЯДКОВИЙ НОМЕР	ЗАГОЛОВОК	КІЛЬКІСТЬ ІДЕНТИЧНИХ СЛІВ (ФРАГМЕНТІВ)
з програми обміну базами даних (1.10 %)		
ПОРЯДКОВИЙ НОМЕР	ЗАГОЛОВОК	КІЛЬКІСТЬ ІДЕНТИЧНИХ СЛІВ (ФРАГМЕНТІВ)
1	ВКР_Крупка 2/5/2025 State University of Trade and Economics (Кафедра комп'ютерних наук та інформаційних систем)	21 (1) 0.24 %
2	ВЕБЗАСТОСУНОК ДЛЯ ВЗАЄМОДІЇ САДІВНИКІВ ТА ДОМОВЛАСНИКІВ. СЕРВЕРНА ЧАСТИНА 6/5/2024 Taras Shevchenko National University of Kyiv (Факультет комп'ютерних наук та кібернетики)	15 (1) 0.17 %
3	Кваліфікаційна робота_Кабанов_К 12/14/2024 Zaporizhzhia National University (Кафедра програмної інженерії)	15 (2) 0.17 %
4	Konchatnyi_bac_rob.docx 6/11/2021 Sumy State University (Кафедра інформаційних технологій)	12 (1) 0.14 %
5	БР_Прут_ІПЗ_4.02 6/3/2025 State University of Intellectual Technologies and Communications (Кафедра інженерії програмного забезпечення)	12 (1) 0.14 %
6	Яворський_Тичковський 1/10/2025 Vasyl Stefanyk Precarpathian National University course papers (Факультет математики та інформатики)	11 (2) 0.13 %
7	Тема курсової 1/10/2025 Vasyl Stefanyk Precarpathian National University course papers (Факультет математики та інформатики)	10 (1) 0.11 %

з Інтернету (1.42 %)

ПОРЯДКОВИЙ НОМЕР	ДЖЕРЕЛО URL	КІЛЬКІСТЬ ІДЕНТИЧНИХ СЛІВ (ФРАГМЕНТІВ)
1	https://dut.edu.ua/repositorii/jpz/2021/%D0%9F%D0%94-42/%D0%94%D0%B0%D1%86%D0%B5%D0%BD%D0%BA%D0%BE.%D0%9C.%D0%90.pdf	55 (4) 0.63 %
2	https://uricsc.github.io/courses/_sources/assets/lecture/heap.md	22 (1) 0.25 %
3	https://metod.vntu.edu.ua/getfile.php/10190.pdf	14 (1) 0.16 %
4	https://openarchive.nure.ua/bitstreams/7c3f3dfe-e12e-465e-94fb-558e4cf87424/download	12 (1) 0.14 %
5	https://ela.kpi.ua/bitstream/123456789/60238/1/Smyrnov_bakalavr.pdf	11 (1) 0.13 %
6	https://openarchive.nure.ua/bitstreams/ece77560-01ff-451d-b29e-e016f4fc93db/download	10 (1) 0.11 %

Список прийнятих фрагментів (немає прийнятих фрагментів)

ПОРЯДКОВИЙ НОМЕР	ЗМІСТ	КІЛЬКІСТЬ ОДНАКОВИХ СЛІВ (ФРАГМЕНТІВ)
------------------	-------	---------------------------------------