

## **РОЗРОБКА ІНФОРМАЦІЙНОЇ СИСТЕМИ ДЛЯ ПОШУКУ РЕПЕТИТОРІВ**

Морочковський О. М.

Науковий керівник – к.т.н., доц. Імангулова З. А.

Харківський національний університет радіоелектроніки, каф. СТ  
м. Харків, Україна

email: [oleksii.morochkovskiy@nure.ua](mailto:oleksii.morochkovskiy@nure.ua)

The work is focused on developing a platform for finding a tutor and studying. It includes both server and client components, allowing users to create personalized accounts and manage their profile information. On this platform, an ordinary user can search for tutors and sign up for classes with them, take part in group or personal classes. Tutors, in turn, can recruit students, create and edit a flexible schedule, and conduct online lessons. Also, each user has the opportunity to evaluate the work of the tutor and leave feedback in accordance with the quality of the tutor's services.

В сучасних умовах, через пандемію коронавірусу і дію воєнного стану, дуже актуальними стали різноманітні онлайн-сервіси, такі як доставка їжі, застосунки онлайн-спілкування, онлайн-магазини, освітні платформи, та багато інших, що дозволяють людям зручно і не виходячи з дому задовольняти свої потреби, працювати та навчатися, і найголовніше – не наражати себе та інших на небезпеку.

Онлайн-навчання через коронавірус сильно вплинуло на якість освіти, а під час повітряних тривог уроки скасовуються до відбою загрози, щоб учні та вчителі пройшли в безпечне місце до укриттів. За різними оцінками, якість навчання впала до 60-65% від тієї що була до 24 лютого 2022 року.

Інформаційна система пошуку репетиторів створена для того, щоб максимально покрити прогалини у знаннях учнів. Даний застосунок допоможе швидко обрати собі репетитора, враховуючи такі критерії, як предмет, вартість уроку, досвід викладання, відгуки інших учнів. З іншої сторони, інформаційна система допоможе вчителям які втратили роботу через війну. Щоб стати репетитором у застосунку достатньо заповнити форму з відомостями про себе, досвід роботи, вказати бажану ціну за урок, і чекати розгляду заявки адміністратором системи.

В рамках цього застосунку було реалізовано три основні ролі користувачів: «Учень», «Репетитор», «Адміністратор». А також присутня роль «Незарєєстрований користувач».

Для «Незарєєстрованого користувача» доступна лише реєстрація та авторизація, а також перегляд загальної інформації про інформаційну систему. Користувач з роллю «Учень» може переглядати наявних репетиторів, сортувати та обирати їх за різними критеріями, переглядати інформацію про конкретного репетитора та його розклад, записуватись на

урок онлайн. Також деякі репетитори можуть запропонувати учням вибір групових або індивідуальних занять. Користувач з роллю «Репетитор» має змогу редагувати інформацію особистого профілю, створювати свій розклад, редагувати його. Також він може контактувати з учнями з приводу уроків, переглядати їх профілі. Користувач з роллю «Адміністратор» виконує роль модератора, він слідкує щоб не порушувались правила інформаційної системи, розглядає скарги, а також заявки на набуття статусу репетитора.

Застосунок орієнтований на масштабованість, щоб легко адаптуватися до зростаючого числа користувачів. Використання технологій .NET дозволяє ефективно масштабувати систему, забезпечуючи стабільну роботу навіть при великих обсягах інформації та одночасних запитів.

У клієнтській частині застосунку використовуються різні мови та технології, такі як JavaScript, HTML та CSS. Інтерфейс користувача оптимізовано для простоти використання та візуальної зручності.

Для структури проекту була обрана Clean Architecture бо вона надає засоби для розбиття складних програмних систем на модулі, що працюють разом узгоджено та ефективно [1]. Також Clean Architecture вагомо спрощує тестування, подальше розширення застосунку та його підтримку в майбутньому. Серверна частина застосунку реалізована за допомогою ASP.NET та мови програмування C#. Для зберігання даних використовується MSSQL Server, що гарантує високий рівень доступності, швидкодію при навантаженнях [2]. Для роботи з базою даних MSSql Server обрано Entity Framework.

При розробці застосунку, було використано декілька архітектурних патернів. Перший – Repository pattern. Це патерн, який використовується для розділення бізнес-логіки взаємодії з даними та деталей роботи з базою даних. Він надає інтерфейс для створення, читання, оновлення та видалення об'єктів даних, забезпечуючи рівень абстракції, що полегшує тестування і зміну джерела даних без впливу на бізнес-логіку. Наступним є патерн Dependency Injection. Він дозволяє створювати залежності між об'єктами під час створення. Головна ідея патерну полягає в тому, щоб розділити створення об'єктів та їх залежностей, тим самим роблячи систему більш гнучкою, тестованою та легкою для розширення.

Таким чином, розроблена інформаційна система для пошуку репетиторів та навчання є відповіддю на сучасні виклики. Вона надає можливість учням знаходити репетиторів, записуватися на уроки та оцінювати роботу викладачів, підтримуючи якість навчання в складних умовах.

Список використаних джерел:

1. Петренко О.О. Порівняння типів архітектури систем сервісів // Системні дослідження і інформаційні технології. № 4. 2015. С. 48-62.
2. Пасічник В.В., Резніченко В.А. Організація баз даних та знань. Київ: Видавнича група ВНУ, 2006. 384 с.