

ПІДСИСТЕМА ОРГАНІЗАЦІЇ ПЕРЕДАЧІ ІНФОРМАЦІЇ

Вінницький національний технічний університет

Анотація. У роботі було розроблено і представлено підсистему для обробки, зберігання та передачі інформації. Даний додаток є складовою комп'ютерної системи для організації планування і виконання навчальних обов'язків студента з мобільного пристрою. Представлена підсистема буде використовуватись іншими підсистемами для зв'язку та передачі інформації. Вона може надавати звіти, статистику та аналітичні дані, що дозволяє адаптувати процес навчання під кожного окремого студента та систему навчання в цілому.

Ключові слова: викладач, студент, база даних, статистика, аналіз, передача інформації.

Abstract. The work presents a subsystem developed for processing, storing, and transmitting information. This application is a component of a computer system for organizing the planning and execution of student learning tasks from a mobile device. The presented subsystem will be utilized by other subsystems for communication and information transfer. It can provide reports, statistics, and analytical data, allowing for the adaptation of the learning process to each individual student and the education system as a whole.

Keywords: teacher, student, database, statistics, analysis, information transmission.

Вступ

В сучасному цифровому світі важливо створити інструменти, які полегшують взаємодію між учасниками освітнього процесу, сприяючи навчанню та розвитку. Враховуючи обмеження, які були введені у зв'язку з пандемією COVID-19, а також війною в Україні ця тема є особливо актуальною. Дана система створена допомогти здолати ці труднощі, без шкоди для підготовки фахівців забезпечивши ефективну взаємодію між студентами та викладачами.

Представлена підсистема є серверним бекендом, який виступає в якості посередника, що забезпечує передачу та обробку даних між програмою для студентів та програмою для викладачів. Він також реалізує систему авторизації та аналізу існуючої інформації, забезпечуючи безпеку та ефективність обміну даними між учасниками навчального процесу. Ця підсистема дозволяє студентам отримувати завдання від своїх викладачів та зручно виконувати їх безпосередньо зі своїх мобільних пристроїв.

Результати дослідження

Постановка задачі

Для реалізації серверної частини використаємо фреймворк ASP.NET, який базується на мові програмування C#. За основу системи управління базами даних візьмемо MySQL.

ASP.NET є фреймворком для розробки веб-додатків, що був розроблений компанією Microsoft. Він базується на платформі .NET Framework і надає засоби для швидкої та ефективної розробки веб-додатків, веб-сервісів і веб-інтерфейсів користувача. ASP.NET може обробляти HTTP-запити від клієнтських додатків і відповідати на них згідно з логікою вашої програми. Також він дозволяє створити різноманітні веб-служби та API для взаємодії з клієнтськими додатками. MySQL - це система управління базами даних, яка використовується для зберігання та управління даними. Використання MySQL у системі допоможе забезпечити ефективне та надійне зберігання, управління та аналізування інформації, яка використовується системою. Вона є однією з найпопулярніших та надійних реляційних СУБД у світі, відомою своєю швидкістю, масштабованістю та простотою використання.

Основні функціональні вимоги додатку:

1. Зберігання користувацьких даних: Реалізація зберігання інформацію про студентів, викладачів та інші дані, необхідні для нормальної роботи системи.
2. Реєстрація та авторизація користувачів: Забезпечення можливості додавання нових користувачів з відповідними ролями та доступом до відповідної інформації, та авторизація уже існуючих.
3. Обробка запитів: Бекенд обробляє запити від клієнтських додатків (наприклад, запити на отримання завдань, створення нових користувачів тощо) та надсилає відповіді з відповідними даними.

4. Взаємодія з базою даних: Серверний бекенд взаємодіє з базою даних для збереження, оновлення та видалення інформації, необхідної для роботи системи.
5. Забезпечення безпеки даних: Підсистема повинна гарантувати конфіденційність інформації та забезпечувати відповідність стандартам безпеки даних.
6. Моніторинг та логування: Необхідно вести журнали дій користувачів та системи для подальшого аналізу та відстеження проблем, невиконаних задач.

Структура комп'ютерної програми

Перед розробкою програми, необхідно представити задачу у вигляді необхідних структурних блоків (див. рис. 1):



Рис. 1 Структура підсистеми організації передачі інформації

Опис основних функцій програмного засобу

При початку роботи з програмою користувач заповнює форму авторизації в додатку, надаючи свої особисті дані, такі як, електронна пошта та пароль. Якщо відповідних даних, ще немає у системі додаток відправляє відповідне повідомлення. Тоді користувачу необхідно пройти реєстрацію. Після заповнення форми реєстрації, що включає в себе введення, ім'я, групи, статусу, електронної пошти та паролю. Перед збереженням облікового запису, дані проходять через валідацію, яка включає перевірку на унікальність електронної пошти та пароля, а також наявність обов'язкових полів. Пароль користувача хешується для збереження в базі даних у захешованому вигляді. ASP.NET Identity використовує солі для безпеки хешування, що робить процес більш безпечним. Сіль - це випадковий рядок, який додається до вхідних даних перед їх хешуванням. В контексті безпеки паролів, сіль використовується для зміни хешування пароля, щоб запобігти атакам методом перебору за допомогою попередньо розрахованих хешів. Також у базу даних заносяться інші данні користувача. Якщо він вже був зареєстрований і введені пошта і пароль відповідають уже наявним у системі буде згенеровано токен доступу, який передається користувачу. Цей токен зазвичай містить інформацію про права доступу користувача та термін його дії.

Після отримання доступу користувач може здійснювати деякі дії, почнемо з викладача. Для створення завдання клієнт (викладач) надсилає запит на створення нового завдання до сервера через інтерфейс мобільного додатку. Сервер отримує цей запит та обробляє його, додаючи дані про нове завдання до бази даних MySQL. Це включає збереження назви завдання, опису, терміну виконання та інших параметрів. Звичайно завдання можна редагувати. Під час редагування завдання клієнт надсилає на сервер запит на оновлення параметрів і вмісту завдання. Сервер отримує цей запит, знаходить відповідне завдання у базі даних та оновлює його згідно з отриманими даними. Також передбачена можливість видалення завдання. Клієнт надсилає запит на видалення конкретного завдання до сервера. Сервер виконує операцію видалення, знаходячи та видаляючи відповідне завдання з бази даних MySQL.

Тепер розглянемо як процес отримання завдань студентом. Клієнт (студент) надсилає запит на сервер для отримання списку доступних завдань. Сервер обробляє запит, витягує інформацію про завдання з бази даних MySQL та надсилає список завдань назад клієнту. Завдання містять всю інформацію яку надав викладач включаючи картинки, текст, дату кінцевого терміну здачі, також завдання має статус виконаності та у випадку виконаного завдання може містити також інформацію про оцінку. Студент обирає конкретне завдання зі списку, яке він бажає виконати. Додаток знову надсилає запит але уже з конкретним завданням на сервер. Він же в свою чергу передає всю необхідну інформацію по цьому завданню студентів. Після виконання завдання студент відправляє результати (наприклад, відповіді, завдання, виконані файли) на сервер через мобільний додаток. Сервер отримує відповідний запит та зберігає у базу даних інформацію яку потім буде отримувати викладач. Викладач може переглянути ці та додати до нього оцінку.

Система здійснює аналіз даних, отриманих від студентів, для генерації звітів та статистики. Звіти можуть містити інформацію про успішність студентів, активність користувачів та інші показники, які допомагають у керуванні навчальним процесом. У ASP.NET ми можемо використовувати LINQ для зручного формування запитів. LINQ - це інструмент мови програмування, який дозволяє виконувати структуровані запити до джерел даних безпосередньо з використанням синтаксису C# або іншої мови .NET. Основною метою LINQ є спрощення та узагальнення процесу взаємодії з даними у мовах програмування .NET, забезпечуючи єдиний інтерфейс для роботи з різноманітними джерелами даних, такими як бази даних. У нашій програмі після того як викладач вибирає відповідну функцію у додатку сервер відсилає відповідний запит базі даних наприклад отримання середньої оцінки студентів кожної групи. Після цього ми витягаємо необхідну проаналізовану інформацію з бази даних та генеруємо звіт. Ці звіти можуть бути представлені у вигляді таблиць, графіків, діаграм або інших візуальних елементів, що надають викладачу зрозумілу інформацію. Загалом, наша система здійснює аналіз даних, отриманих від користувачів, шляхом збору, обробки цих даних за допомогою ASP.NET та MySQL. Це дозволяє нам надавати користувачам корисну інформацію та інсайти для прийняття рішень.

Висновки

Наша система, розроблена з використанням ASP.NET та MySQL, є потужним інструментом для ефективно організації взаємодії між студентами та викладачами. Вона дозволяє студентам отримувати завдання, виконувати їх та взаємодіяти з викладачами, забезпечуючи простий та зручний інтерфейс. Система також забезпечує аналіз та генерацію звітів для підвищення якості навчання та допомагає викладачам адаптувати процес навчання під потреби кожного студента.

Завдяки оптимізованій архітектурі та ефективному використанню сучасних технологій, система забезпечує швидкий доступ до інформації, миттєвну відповідь на запити користувачів та стабільну роботу додатку навіть при великому обсязі даних та великій кількості одночасних користувачів.

Таким чином, наша система гарантує не лише ефективну роботу та зручний інтерфейс користувача, але й високу якість, надійність та швидкодію, що робить її незамінним інструментом для організації дистанційного навчального процесу.

СПИСОК ВИКОРИСТАНОЇ ЛІТЕРАТУРИ

1. ASP.NET documentation [Електронний ресурс] – Режим доступу до ресурсу: <https://learn.microsoft.com/en-us/aspnet/core/?view=aspnetcore-8.0>
2. Database basics [Електронний ресурс] – Режим доступу до ресурсу: <https://support.microsoft.com/en-au/office/database-basics-a849ac16-07c7-4a31-9948-3c8c94a7c204>
3. “CLIENT/SERVER ARCHITECTURE” by Alex Berson. Hardcover, 1992.
4. MySQL Manual [Електронний ресурс] – Режим доступу до ресурсу: https://docs.oracle.com/cd/E17952_01/mysql-8.3-en/index.html

Снігур Анатолій Васильович – к.т.н., доцент кафедри обчислювальної техніки, Вінницький національний технічний університет, Вінниця.

Сурога Олексій Костянтинівич – студент групи 1СП-20б, факультет інформаційних технологій та комп’ютерної інженерії, Вінницький національний технічний університет, Вінниця, e-mail: a.surota2003@gmail.com

Snigur Anatoliy Vasyliovych – Ph.D., Associate Professor of the Computer Engineering Department, Vinnytsia National Technical University, Vinnytsia.

Sirota Oleksiy Kostiantynovych – student of group 1SP-20b, Faculty of Information Technologies and Computer Engineering, Vinnytsia National Technical University, Vinnytsia, e-mail: a.surota2003@gmail.com