

УДК 004.4

DOI <https://doi.org/10.32851/tnv-tech.2022.1.4>

## ТЕОРЕТИЧНІ АСПЕКТИ РОЗРОБКИ СИСТЕМИ УПРАВЛІННЯ НАВЧАННЯМ

**Коротун О.В.** – кандидат педагогічних наук, доцент кафедри комп'ютерних наук  
Державного університету «Житомирська політехніка»  
ORCID ID: 0000-0003-2240-7891

**Вакالیук Т.А.** – доктор педагогічних наук, професор,  
професор кафедри інженерії програмного забезпечення  
Державного університету «Житомирська політехніка»  
ORCID ID: 0000-0001-6825-4697

**Зубрицький В.В.** – студент  
Державного університету «Житомирська політехніка»  
ORCID ID: 0000-0002-9216-8032

**Гордієнко І.В.** – кандидат педагогічних наук  
Дрогобицького державного педагогічного університету імені Івана Франка  
ORCID ID: 0000-0001-6182-4968

У статті описана розробка веб-додатку «система управління навчанням», що ґрунтується на використанні архітектурного патерну для побудови таких додатків – MVC. Також при створенні були використані наступні інструменти та технології, а саме: Microsoft SQL Server – система керування базами даних та мова, що використовується для запитів – Transact-SQL, MS SQL обрано завдяки її надійності та захищеності та вбудовану підтримку .NET Framework; ASP.NET Core як вільно-розповсюджуваний крос-платформний фреймворк для створення веб-додатків з відкритим вихідним кодом, що підтримують паралельне управління версіями, при якому різні програми, що працюють на одному комп'ютері, можуть ґрунтуватися на різних версіях ASP.NET Core; з метою побудови GraphQL API в даному веб-додатку буде використано бібліотеку з відкритим кодом GraphQL.Net; Entity Framework CORE як об'єктно-орієнтовану технологію доступу до даних; React – відкрита JavaScript бібліотека для створення інтерфейсів користувача; Redux – відкрита JS бібліотека призначена для управління станом програм JavaScript та Apollo Client як бібліотеку управління станом для JavaScript, яка дозволяє управляти локальними та віддаленими даними за допомогою GraphQL. У роботі описана структура системи управління навчанням, представлена схема база даних розробленої системи, у системі визначені три ролі (учитель, учень, користувач) та їх права, продемонстрована діаграма компонентів системи, до якої входять модулі модулі LearningService.EF відповідає за роботу з БД, бізнес логіка LearningService.Core та веб-частина сисме LearningService.Web, а також визначені скалярні функції для підрахунку середнього результату користувачів за проходження тесту, кількості проходжень тесту, рейтингу студента, прогресу проходження курсу студентом, кількості студентів, що підписались на окремий курс.

**Ключові слова:** дистанційне навчання, система управління навчанням, веб-додаток, MVC, база даних.

**Korotun O.V., Vakaliuk T.A., Zubrytskyi V.V., Hordiienko I.V. Theoretical aspects of learning management system development**

The article describes the development of a web application “learning management system”, which is based on the use of an architectural pattern for the construction of such applications – MVC. The following tools and technologies were also used in the creation, namely: Microsoft SQL Server – database management system and language used for queries – Transact-SQL, MS SQL was chosen due to its reliability and security and built-in support for .NET Framework; ASP.

*NET Core as a free-to-air cross-platform framework for creating open-source web applications that support parallel version control, in which different programs running on the same computer can target different versions of ASP.NET Core; to build the GraphQL API in this web application will use the open-source library GraphQL.Net; Entity Framework CORE as an object-oriented data access technology; React – open JavaScript library for creating user interfaces; Redux is an open-source JS library designed to manage the status of JavaScript and Apollo Client programs as a JavaScript management library that allows you to manage local and remote data using GraphQL. The structure of the learning management system is described, the database of the developed system is presented, the system defines three roles (teacher, student, user) and their rights, the diagram of system components is shown, which includes modules. business logic LearningService.Core and web part of the system LearningService.Web, as well as defined scalar functions for calculating the average user score for the test, the number of tests, student rating, the progress of the student, the number of students who signed up for a course.*

**Key words:** distance learning, learning management system, web application, MVC, database.

**Вступ.** В останні роки головною тенденцією розвитку вітчизняної та зарубіжної освіти є дистанційне навчання, особливо гостро це питання постає в умовах пов'язаних з поширенням COVID-19. Дистанційне навчання ґрунтується на широкому використанні різноманітних новітніх інформаційних технологій, мережі Інтернет, моделей навчання, які адаптовані під таке навчання, та спрямоване на підвищення його якості, полегшення доступу студентів та викладачів до освітніх ресурсів і послуг, а також обміну навчальною інформацією та спільною роботою на відстані. Крім віддаленого навчання, дистанційне навчання стає доповненням очної та заочної форм навчання у закладах вищої освіти.

Дистанційне навчання орієнтоване на принцип «вчитися завжди і всюди», створює умови для реалізації принципів особистісно орієнтованої освіти. Важливою складовою побудови такого навчання та його впровадження є активне та постійне використання систем управління навчанням (СУН).

**Постановка проблеми.** На сьогодні освітньою спільнотою вже накопичений достатній досвід і навчально-методичний матеріал щодо впровадження та застосування різноманітних СУН та створення в них електронних навчальних курсів для підвищення ефективності процесу засвоєння знань та покращення якості підготовки студентів. У системі очної освіти такі курси можна використовувати як додаткові навчальні засоби, що дозволяють методично правильно організувати контрольовану викладачем самостійну роботу студентів. На сьогодні недостатньо вивчена технічна сторона розробки системи управління навчанням, тому висвітлено наступні завдання, що виникають при створенні такого додатку, а саме:

- проведення аналізу систем управління навчанням;
- визначення архітектури та узагальненої структури системи;
- обґрунтування та вибір засобів реалізації системи;
- реалізація веб-додатку «система управління навчанням».

**Мета роботи** – дослідження особливостей створення та розробки веб-додатку «система управління навчанням».

**Аналіз останніх досліджень і публікацій.** Вітчизняні та зарубіжні науковці зробили значний внесок щодо дослідження та висвітлення проблеми впровадження дистанційного навчання, а саме: Ю.М. Богачков, В.Ю. Биков, Т.А. Вакалюк, О.В. Коротун, В.М. Кухаренко, Н.Г. Сиротенко, О.М. Спірін, Ю.В. Триус та інші вчені. Н.М. Носовець [11] у результаті аналізу педагогічних досліджень описує основні функціональні компоненти педагогічної діяльності викладача дистанційної освіти, виділено такі види діяльності: методична; організаційно-управлінська; психолого-педагогічна; викладацька; дослідницька. Н.В. Бахмат [1] у своїй роботі наголошує на необхідності розвитку та впровадження

інформаційно-комунікаційних технологій і дистанційної форми навчання у вищій школі, визначає низку перспективних напрямів у розробленні та впровадженні інформаційноосвітнього середовища освітньо-наукової підготовки магістрів в умовах дистанційного навчання.

Н.С. Довбиш та М.В. Петченко [8] описують зародження дистанційного навчання в Україні, розглядають роль сучасного викладача в системі дистанційного навчання, а також роблять порівняльний аналіз впровадження дистанційного навчання у закладах вищої освіти України. Недоліки системи дистанційного навчання представлені у роботі К.М. Пересади та Л.М. Приблуди [12]. Науковець С.М. Бужинська [3] визначає роль дистанційного навчання в системі ЗВО та проблеми, що виникають на шляху адаптації студентів в реалізації даного освітнього середовища.

Проблемі організації та впровадження СУН в освітній процес присвячені праці таких науковців: А.Л. Бочкова [2], Т.А. Вакалюк [4], А. І. Гладир і Н. В. Зачепи [6], С.М. Горобця [7], О.В. Коротун [9], С.Г. Литвинова [10], О.М. Спіріна [5] та ін. Однак проведений аналіз наукових праць свідчить про недостатнє вивчення технічного аспекту проблеми створення та розробки веб-додатку «система управління навчанням».

**Виклад основного матеріалу.** Розглянемо завдання вибору архітектури розробки веб-додатку «система управління навчанням». Найпопулярнішим архітектурним патерном для побудови веб-додатку є MVC. Даний шаблон передбачає поділ системи на три взаємопов'язані частини: модель даних, вигляд (інтерфейс користувача) та модуль керування. Застосовується для відокремлення даних (моделі) від інтерфейсу користувача (вигляду) так, щоб зміни інтерфейсу користувача мінімально впливали на роботу з даними, а зміни в моделі даних могли здійснюватися без змін інтерфейсу користувача. Проте цей патерн має суттєвий недолік – кожна дія користувача супроводжується не лише запитом на сервер, а й повним презавантаження сторінки, що досить негативно впливає на довід користувача.

Тому для побудови користувацького інтерфейсу буде використано односторінковий застосунок. Односторінковий застосунок (single-page application, SPA), також відомий як односторінковий інтерфейс (single-page interface, SPI) – це веб-застосунок чи веб-сайт, який вміщується на одній сторінці з метою забезпечити користувачу досвід близький до користування настільною програмою.

В односторінковому застосунку весь необхідний код – HTML, JavaScript, та CSS – завантажується разом зі сторінкою, або динамічно довантажується за потребою, зазвичай у відповідь на дії користувача. Сторінка не оновлюється і не перенаправляє користувача до іншої сторінки у процесі роботи з нею. Взаємодія з односторінковим застосунком часто включає в себе динамічний зв'язок з веб-сервером [4].

Для серверної частини додатку було обрано GraphQL на противагу REST (Representational State Transfer – архітектурний стиль для розподілених гіпертекстових систем). Усі запити на сервер будуть виконуватися за допомогою графової мови запитів, що працює наступним чином: клієнт в запиті вказує ті дані, які необхідно отримати. GraphQL – це мова запитів даних та мова маніпуляцій з відкритим кодом для API, і середовище виконання для виконання запитів із наявними даними. GraphQL було розроблено на противагу побудови сервісам на архітектурі Rest API [2].

Для розробки веб-додатку та його інтерфейсу існує безліч засобів, але в даній кваліфікаційній роботі використані наступні:

1) Microsoft SQL Server – комерційна система керування базами даних, що розповсюджується корпорацією Microsoft. Мова, що використовується для запитів – Transact-SQL, створена спільно Microsoft та Sybase. Transact-SQL є реалізацією стандарту ANSI / ISO щодо структурованої мови запитів SQL із розширеннями. Використовується як для невеликих і середніх за розміром баз даних, так і для великих баз даних масштабу підприємства. Багато років вдало конкурує з іншими системами керування базами даних. MS SQL використовує реляційну модель СУБД для контролю цілісності даних [8].

MS SQL було обрано через надійність та захищеність та вбудовану підтримку .NET Framework. Завдяки цьому, процедури бази даних, що зберігаються, можуть бути написані на будь-якій мові платформи .NET з використанням повного набору бібліотек, доступних для .NET Framework.

2) ASP.NET Core – вільно-розповсюджуваний крос-платформний фреймворк для створення веб-додатків з відкритим вихідним кодом. Дана платформа розробляється компанією Майкрософт спільно з спільнотою і має велику продуктивність в порівнянні з ASP.NET. Має модульну структуру і сумісна з такими операційними системами як Windows, Linux і macOS.

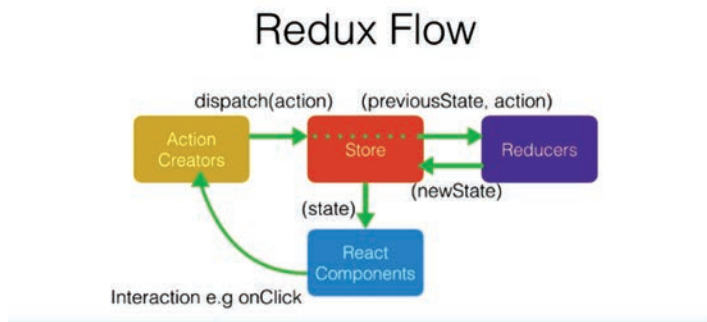
Незважаючи на те, що це новий фреймворк, побудований на новому веб-стеку, він має високий ступінь сумісності концепцій з ASP.NET. Додатки ASP.NET Core підтримують паралельне управління версіями, при якому різні програми, що працюють на одному комп'ютері, можуть орієнтуватися на різні версії ASP.NET Core. Це було неможливо в попередніх версіях ASP.NET [3].

Додаток ASP.NET Core виконується разом з внутрішньо процесною реалізацією HTTP-сервера. Реалізація сервера прослуховує HTTP-запити і передає їх в додаток як набір функцій запитів, об'єднаних в HttpContext. Як веб-сервер буде використовуватися Kestrel. Це багатоплатформний веб-сервер для ASP.NET Core. Kestrel за замовчуванням включається в шаблони проектів ASP.NET Core. Для побудови GraphQL API в даному веб-додатку буде використано бібліотеку з відкритим кодом GraphQL.Net [1].

3) Entity Framework CORE – об'єктно-орієнтована технологія доступу до даних, є object-relational mapping (ORM) рішенням для .NET CORE Framework від Microsoft. Надає можливість взаємодії з об'єктами як за допомогою LINQ у вигляді LINQ to Entities, так і з використанням Entity SQL. Вона була частиною .NET Framework, але з версії 6 Entity Framework стала відокремленою від .NET Framework [7].

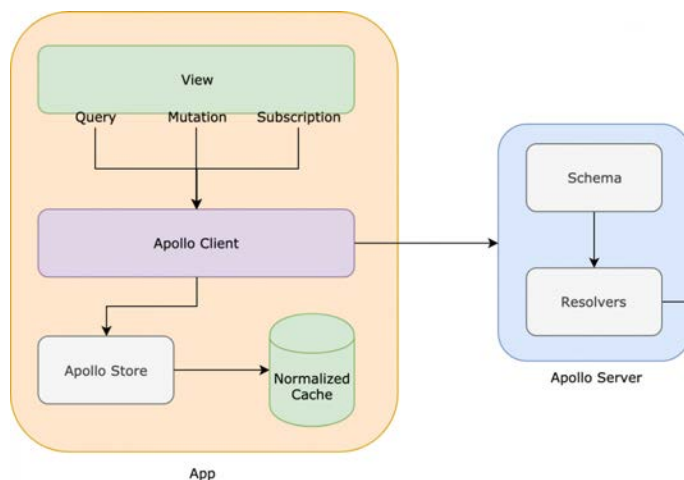
4) React – відкрита JavaScript бібліотека для створення інтерфейсів користувача, яка покликана вирішувати проблеми часткового оновлення вмісту веб-сторінки, з якими стикаються в розробці односторінкових застосунків. React дозволяє розробникам створювати великі веб-застосунки, які використовують дані, котрі змінюються з часом, без перезавантаження сторінки. Його мета полягає в тому, щоб інтерфейс системи був швидким, простим та масштабованим.

5) Redux – відкрита JS бібліотека призначена для управління станом програм JavaScript. Найчастіше використовується разом з React або Angular для побудови інтерфейсів користувача. Redux використовує наступний потік подій, що зображено на рис. 1. Головна особливість Redux – це підтримка великого єдиного незмінного стейту. Для кожної дії він бере старий стейт, робить копію, застосовує нові зміни та робить його новим стейтом.



*Рис. 1. Redux-Flow*

6) Apollo Client – це всебічна бібліотека управління станом для JavaScript, яка дозволяє управляти як локальними, так і віддаленими даними за допомогою GraphQL. Використовується його для отримання, кешування та модифікації даних програми, при цьому автоматично оновлюючи інтерфейс (рис. 2).



*Рис. 2. Схема використання Apollo Client в додатку*

Наступним завданням є визначення варіантів використання та структури системи управління навчанням. Аналіз вимог до програмного забезпечення дозволив побудувати діаграму варіантів використання СУН. Діаграми варіантів використання описують взаємозв'язки і залежності між групою варіантів використання і акторами, які беруть участь у процесі. За діаграмами варіантів використання можна сказати, що система має робити, але не те, як вона досягає потрібних результатів, для останнього ці діаграми просто не придатні.

Діаграми варіантів використання служать для позначення специфічної ролі актора в окремому варіанті застосування. Іншими словами, асоціація специфікує семантичні особливості взаємодії акторів і варіантів використання в графічній моделі системи. Таким чином, це відношення встановлює, яку конкретну роль грає актор при взаємодії з примірником варіанту використання

Актор – це зовнішній чинник (поза межами системи), який взаємодіє з системою шляхом участі у випадку використання. Акторами, на практиці, можуть бути звичайні люди (наприклад, користувачі системи), інші комп'ютерні системи або зовнішні події.

Ця діаграма дає змогу зрозуміти, які актори взаємодітимуть з веб-додатком. Це зокрема неавторизований та авторизований користувач, а також адміністративні актори такі як студент, викладач та адміністратор.

Для неавторизованого користувача є можливість перегляду списку доступних курсів і проходження реєстрації. В свою чергу, множина дій авторизованого користувача розширюється: 1) після знаходження відповідного курсу та запису на нього отримувати весь контент курсу (навчальний матеріал) та відповідні завдання для перевірки знань; 2) зможе відстежувати поточний хід навчання та перевіряти всі оцінки відповідно до системи оцінювання; 3) викладач повністю керує курсом (його наповненням);

Побудовану діаграму варіантів використання зображено на рис. 3.

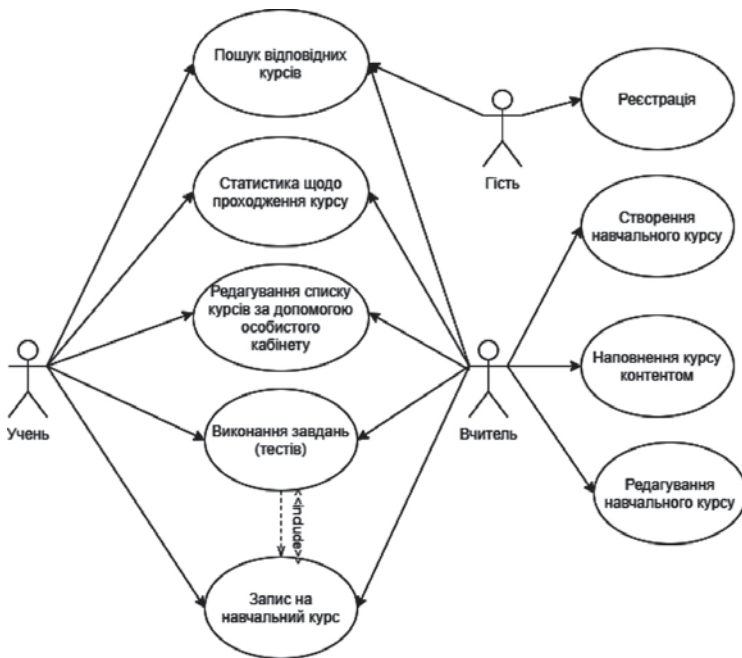


Рис. 3. Діаграма варіантів використання системи управління навчанням

Наступним етапом є проектування структури системи управління навчанням. Для цього буде доцільно побудувати діаграму компонентів. Діаграма компонентів – діаграма, на якій відображаються компоненти, залежності та зв'язки між ними.

Діаграми компонентів відображають залежності між програмними компонентами, що виникають на етапі компіляції або в процесі виконання програми. На діаграмах компонентів зображується входження класів і об'єктів в програмні компоненти системи (модулі, бібліотеки тощо). Побудовану діаграму компонентів зображено на рис. 4.



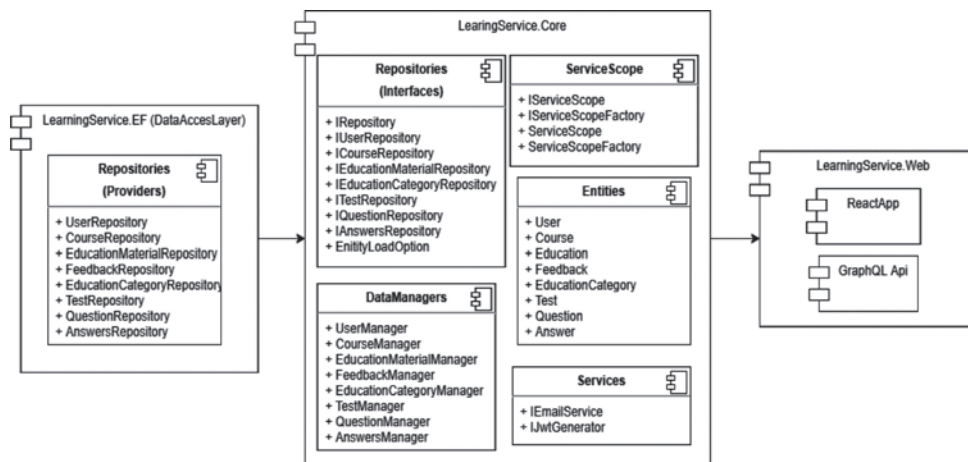


Рис. 4. Діаграма компонентів

У діаграмі система ділиться на 3 модулі:

1. LearningService.EF (DataAccessLayer) – відповідає за роботу з БД, реалізований на основі ORM (Object-relational mapping) – Microsoft.EntityFrameworkCore;
  - Repositories – класи, які реалізують інтерфейси (LearningService.Core.Repositories) та відповідають за роботу з базою даних, а саме створення, редагування, видалення та вилучення даних;
2. LearningService.Core – бізнес логіка:
  - Repositories – інтерфейси, що описують поведінку роботи з базою даних;
  - Services – інтерфейси, що описують поведінку додаткових операцій, таких як відправка листів, генерація токенів, тощо;
  - DataManagers – класи, які інкапсують в собі логіку роботи з даними;
  - Entities – класи, що виступають моделями даних для БД;
3. LearningService.Web – веб-частина системи:
  - GraphQL Api – ключова точка системи побудована для отримання та видачі даних, на основі графової мови запитів GraphQL;
  - client-app – front-end частина, з якою взаємодіє користувач.

Розглянемо схему бази даних системи управління навчанням. При проектуванні структури бази даних СУН включені наступні таблиці (рис. 5): Courses; Users; User\_Test; User\_Role; Roles; Tests; Questions; Answers; EducationCategories; EducationMaterial; Course\_User.

Розглянемо кожену таблицю БД, що має свою ключову роль та призначення. Таблиця «Courses» призначена для збереження інформації про курс, яка може бути цікавою та корисною для студенту.

Для збереження даних про особисту інформацію користувача створена таблиця «Users». Також містить дані необхідні для автентифікації. Для збереження інформації про роль користувача була створена допоміжна таблиця «User\_Role». Для збереження інформації про доступні ролі на сервері та в додатку була створена таблиця «Roles». Таблиця «EducationCategories» містить всю детальну інформацію про теми та навчальні категорії, які слугують орієнтиром для користувача в різноманітні контенту курсу. Таблиця «EducationMaterial» містить всю корисну та детальну інформацію для користувача, а саме вивчення начального матеріалу навчальних курсів. Таблиця «Tests» містить детальну інформацію про тест для користувача та деяку

статистичну інформацію для викладачів. Для збереження результатів проходження тестів студентами та водночас для формування статистики щодо проходження ним курсу створена таблиця «User\_Tests». Для збереження інформації щодо питань, які використовуються в тесті, створено таблицю «Questions». Для збереження інформації про відповіді до запитань тестів створено таблицю «Answers».

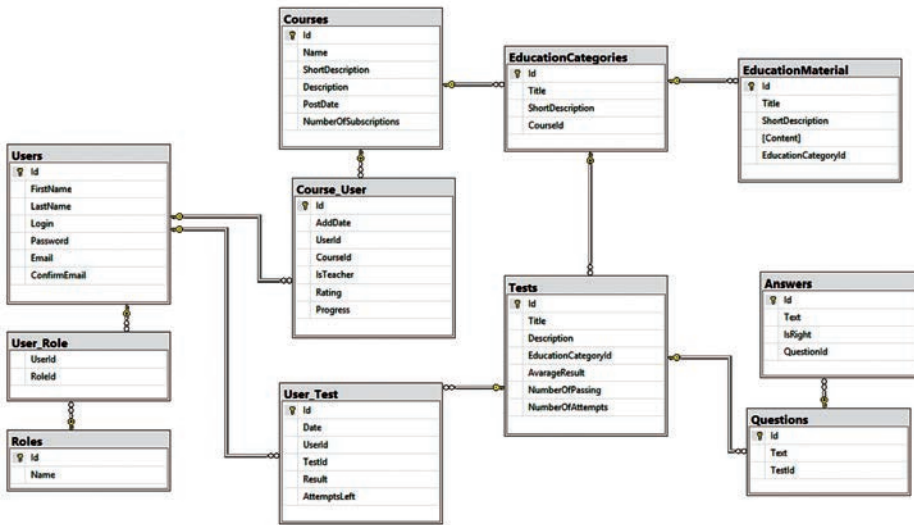


Рис. 5. Діаграма бази даних

Однією з основних можливостей СУН є відслідковування прогресу проходження курсу як викладачами, так і студентами та отримання статистики щодо активності студентів викладачами. Саме для цього було створено наступні функції в базі даних:

Таблиця 1

### Опис скалярних функцій

Назва	Вхідні параметри	Опис
CalculateAverageResult	TestId (Uniqueidentifier)	Підраховує середній результат користувачів за проходження тесту
CalculateNumberOfPassing	TestId (Uniqueidentifier)	Підраховує кількість проходжень тесту
CalculateUserRating	CourseId (Uniqueidentifier), UserId (Uniqueidentifier)	Підраховує рейтинг студента за результатами проходження тестів
CalculateUserProgress	CourseId (Uniqueidentifier), UserId (Uniqueidentifier)	Підраховує прогрес проходження курсу студентом
CalculateNumberOfSubscriptions	CourseId (Uniqueidentifier)	Підраховує кількість студентів, що підписались на курс



**Висновки і пропозиції.** Проведений аналіз предметної області дозволив визначити основні аспекти СУН. Для цього обрано оптимальну архітектуру з точки зору користувачького досвіду та на її основі підібрано відповідний набір технологій, що буде використовуватися для побудови серверної та клієнтської частини додатку. Визначившись із оптимальною архітектурою, обрано достатній набір програмних продуктів для реалізації задач: ASP.NET Core та GraphQL.Net для реалізації GraphAPI, EntityFramework Core для роботи з СУБД, MS SQL як СУБД, React, Redux, Apollo Client для клієнтської частини, Microsoft Visual Studio 2019 та Visual Studio Code для розробки серверної та клієнтської частини відповідно. Проаналізовано всі варіанти використання СУН та визначено основні вимоги для роботи системи. Визначені основні об'єкти в системі та зв'язки між ними. Спроековано базу даних додатку та діаграму класів.

### СПИСОК ВИКОРИСТАНОЇ ЛІТЕРАТУРИ:

1. Бахмат Н.В. Інформаційно-освітнє середовище закладу вищої освіти в системі дистанційного навчання. *Наукові записки Національного педагогічного університету імені МП Драгоманова, Серія: «Педагогічні науки»*. 2018. С. 28–38.
2. Бочков А.Л. Использование LMS-систем для дистанционного обучения. *Современное машиностроение. Наука и образование*. 2014. С. 42–48.
3. Бужинська С., Стрюкова С. Адаптація студентів ЗВО до дистанційного навчання: психологічний аспект. *Матеріали конференції МЦНД*. 2020. С. 58–60.
4. Вакалюк Т.А. Переваги використання хмарної LMS NEO перед іншими аналогами при проектуванні хмаро орієнтованого середовища навчання для підготовки бакалаврів інформатики [Електронний ресурс]. *Сборник материалов XII Международной конференции «Стратегия качества в промышленности и образовании»*. *Международный научный журнал Acta Universitatis Pontica Euxinus*. Спец. вып. Днепрпетровск, Варна. Рр. 505–510.
5. Використання електронних відкритих систем для інформаційно-аналітичної підтримки педагогічних досліджень: короткий термінологічний словник / упоряд.: О.М. Спірін, С.М. Іванова, А.В. Яцишин, А.В. Кільченко та ін. Київ : ІТЗН НАПН України, 2017. 67 с.
6. Гладир А.Г., Зачепа Н.В., Мотруніч О.О. Системи дистанційного навчання: огляд програмних платформ. *Електромеханічні та енергетичні системи, методи моделювання та оптимізації* : матеріали міжнар. наук.-практ. конф. Кременчук, 2012. С. 43–44.
7. Горобець С.М., Горобець О.В. Перспективи використання інформаційно-комунікаційних технологій у дистанційному та змішаному навчанні студентів гуманітарних спеціальностей. *Нові технології навчання*: наук.-метод. зб. Київ : Ін-т інноваційних технологій і змісту освіти МОН України, 2016. Вип. 89. Ч. 2. С. 85–90.
8. Довбиш Н.Є. Сучасна роль викладача в системі дистанційного навчання. *Наукові тренди постіндустріального суспільства: матеріали міжнар. наук. конф. (м. Рівне, 28 лют. 2020 р.)* Рівне, 2020. Т. 1. С. 120–123.
9. Коротун О.В. Аналіз вітчизняного та зарубіжного досвіду використання систем управління навчанням. *Автоматизація та комп'ютерно-інтегровані технології у виробництві та освіті: стан, досягнення, перспективи розвитку*: матеріали Всеукр. наук.-практ. Internet-конф. Черкаси, 2017. С. 169–173.
10. Литвинова С.Г. Теоретико-методичні основи проектування хмаро орієнтованого навчального середовища загальноосвітнього навчального закладу: автореферат дис. ... д-ра пед. наук : 13.00.10 – інформаційно-комунікаційні технології в освіті / Литвинова Світлана Григорівна ; Академія пед. наук України, Інститут інформаційних технологій та засобів навчання. Київ, 2016. 40 с.

11. Носовець Н.М. Роль і функції викладача в системі дистанційного навчання. *Вісник Чернігівського національного педагогічного університету імені Т.Г. Шевченка*. Чернівці : ЧНПУ, 2017, Вип. 144. С. 90–94.

12. Пересада К.М. Недоліки системи дистанційного навчання. *Організація освітнього процесу в умовах дистанційного навчання у вищій школі: методологія, методика, практика : тези доповідей Всеукраїнської науково-методичної конференції*. Київ : НУХТ, 2021. С. 102–103.

#### REFERENCES:

1. Bakhmat N.V. (2018) Informacijno-osvitnje sredovyshhe zakladu vyshhoji osvity v systemi dystancijnogho navchannja [Information and educational environment of higher education institution in the distance learning system]. *Naukovi zapysky Nacionaljnogho pedagoghichnogho universytetu imeni MP Draghomanova, Serija: Pedagoghichni nauky*, pp. 28–38. Retrieved from: <http://enpuir.npu.edu.ua/bitstream/handle/123456789/24181/Bakhmat.pdf?sequence=1> (accessed January 9, 2022)

2. Bochkov A.L. (2014) Yspolzovanye LMS-system dlia dystantsyonnoho obuchenya [Using LMS systems for distance learning]. *Modern engineering. Science and education*, pp. 42–48.

3. Buzhynska S., Striukova S. (2020) Adaptatsiia studentiv ZVO do dystantsiinoho navchannia: psykholohichni aspekt [Adaptation of students of higher education to distance learning: psychological aspect]. *Scientific support of technological progress of the XXI century: materials of the International scientific conference (Ukraine, Chernivtsi, May 1, 2020)*, Chernivtsi pp. 58–60.

4. Vakaliuk T.A. (2016) Perevahy vykorystannia khmaroi LMS NEO pered inshymy analogamy pry proektuvanni khmaro oriietovanoho sredovyshcha navchannia dlia pidhotovky bakalavriv informatyky [Advantages of using cloud LMS NEO over other analogues in the design of cloud-based learning environment for the preparation of bachelors of computer science]. *Sbornyk materyalov XII Mezhdunarodnoi konferentsyy «Stratehiya kachestva v promyshlennosti y obrazovanyu»*. *Mezhdunarodnyi nauchnyi zhurnal Acta Universitatis Pontica Euxinus*. Spets. vyp. Dnepropetrovsk, Varna. pp. 505–510. Retrieved from: <http://eprints.zu.edu.ua/21653/> (accessed January 9, 2022)

5. Vykorystannia elektronnykh vidkrytykh system dlia informatsiino-analitychnoi pidtrymky pedahohichnykh doslidzen: korotkyi terminolohichni slovnyk (2016) uporiad.: O.M. Spirin, S.M. Ivanova, A.V. Yatsyshyn, A.V. Kilchenko ta in. Kyiv: IITZN NAPN Ukrainy. [in Ukrainian]

6. Hladyr A.I., Zachepa N.V., Motrunich O.O. (2012) Systemy dystantsiinoho navchannia: ohliad prohramnykh platform [Distance learning systems: an overview of software platforms]. *Elektromekhanichni ta enerhetychni systemy, metody modeliuвання ta optymizatsii : materialy mizhnar. nauk.-prakt. konf.* Kremenichuk, pp. 43–44.

7. Horobets S.M., Horobets O.V. (2016) Perspektyvy vykorystannia informatsiino-komunikatsiinykh tekhnolohii u dystantsiinomu ta zmishanomu navchanni studentiv humanitarnykh spetsialnostoni [Prospects for the use of information and communication technologies in distance and blended learning of humanities students.]. *Novi tekhnolohii navchannia: nauk.-metod. zb. Kyiv: In-t innovatsiinykh tekhnolohii i zmistu osvity MON Ukrainy*, vol. 89. no. 2. pp. 85–90. Retrieved from: <http://ir.znau.edu.ua/handle/123456789/8156> (accessed January 9, 2022)

8. Dovbysh N.Ye. (2020) Suchasna rol vykladacha v systemi dystantsiinoho navchannia [The modern role of the teacher in the distance learning system]. *Naukovi trendy postindustrialnogo suspilstva: materialy mizhnar. nauk. konf.* (Rivne, 28 liut. 2020 r.) Rivne, T. 1. pp. 120–123.

9. Korotun O.V. (2017) Analiz vitchyznianoho ta zarubizhnogo dosvidu vykorystannia system upravlinnia navchanniam [Analysis of domestic and foreign experience in the use of learning management systems]. *Avtomatyzatsiia ta kompiuterno-intehrovani*

*tehnolohii u vyrobnytstvi ta osviti: stan, dosiahnennia, perspektyvy rozvytku: materialy Vseukr. nauk.-prakt. Internet-konf. Cherkasy, pp. 169–173.*

10. Lytvynova S.H. (2016) *Teoretyko-metodychni osnovy proektuvannia khmaro oriientovanoho navchalnoho seredovyshcha zahalnoosvitnoho navchalnoho zakladu* [Theoretical and methodological bases of designing a cloud-based educational environment of a secondary school]: avtoreferat dys. ... d-ra ped. nauk : 13.00.10 – informatsiino-komunikatsiini tehnolohii v osviti. Kyiv: Akademiia ped. nauk Ukrainy, Instytut informatsiinykh tehnolohii ta zasobiv navchannia.

11. Nosovets N.M. (2017) Rol i funktsii vykladacha v systemi dystantsiinoho navchannia [The role and functions of the teacher in the distance learning system]. *Visnyk Chernihivskoho natsionalnoho pedahohichnoho universytetu imeni T.H. Shevchenka*. Chernihiv : ChNPU, 2017, Vol. 144. pp. 90–94. Retrieved from: <http://erpub.chnpu.edu.ua:8080/jspui/handle/123456789/661> (accessed January 9, 2022)

12. Peresada K.M. (2021) Nedoliky systemy dystantsiinoho navchannia [Disadvantages of the distance learning system]. *Orhanizatsiia osvitnoho protsesu v umovakh dystantsiinoho navchannia u vyshchii shkoli: metodolohiia, metodyka, praktyka : tezy dopovidei Vseukrainskoi naukovo-metodychnoi konferentsii*. Kyiv : NUKhT, pp. 102–103.