

## УДОСКОНАЛЕННЯ МЕТОДІВ ТА ЗАСОБІВ РЕАЛІЗАЦІЇ КОМУНІКАЦІЙ В ЕЛЕКТРОННОМУ ОСВІТНЬОМУ СЕРЕДОВИЩІ

### *Анотація*

*В даному проєкті розглянуто основні аспекти розроблення клієнт-серверної системи для реалізації комунікацій в електронному освітньому середовищі та технології, які були застосовані. Проаналізовано архітектуру серверної та клієнтської частин, розглянуто вимоги до інтерфейсу користувача. В результаті роботи було представлено серверну та клієнтську частини системи.*

**Ключові слова:** клієнт-серверна архітектура, комунікація, студент, викладач, електронне освітнє середовище.

### *Abstract*

*In this project, we have examined the main aspects of developing a client-server system for implementing communications in an electronic educational environment and the technologies that have been utilized. We have analyzed the architecture of the server and client components and discussed user interface requirements. As a result of the work, both the server and client components of the system have been presented.*

**Keywords:** client-server architecture, communication, student, teacher, electronic educational environment.

### **Вступ**

Актуальність і важливість комунікацій в електронному освітньому середовищі не можуть бути переоцінені, оскільки вони впливають на якість навчання, розвиток учнів та ефективність викладання [1]. Електронне освітнє середовище дозволяє студентам з різних частин світу отримувати якісну освіту, не виходячи з дому. Ефективна комунікація допомагає зменшити географічні обмеження та забезпечує доступ до ресурсів та експертів з будь-якої точки світу.

Комунікація в електронному освітньому середовищі сприяє активному навчанню, де студенти можуть взаємодіяти з матеріалами, викладачами та одне з одним. Це сприяє поглибленню розуміння та запам'ятовуванню матеріалу. Електронні платформи надають можливість індивідуалізувати навчання, а ефективна комунікація дозволяє викладачам і студентам адаптувати підхід до конкретних потреб та можливостей кожного учасника навчального процесу.

Ефективна комунікація сприяє залученості студентів до дискусій, обговорень та колективних проєктів. Вона створює зручний механізм для вираження власних думок, задавання питань та розуміння матеріалу. Комунікація у електронному освітньому середовищі дозволяє вдосконалити процес оцінювання завдяки зручній системі відстеження успішності студентів, після чого можна надати своєчасний та обґрунтований фідбек. В електронному освітньому середовищі студенти навчаються ефективно спілкуватися онлайн, що стає надзвичайно важливим в цифровому світі та на ринку праці.

Метою роботи є удосконалення методів та засобів реалізації комунікації в електронному освітньому середовищі для викладачів, студентів та інших учасників навчального процесу, за рахунок зручного та інтуїтивно зрозумілого інтерфейсу, та автоматизації процесів.

### **Аналіз стану питання**

Сучасний світ прямує до розвитку технологій, які допомагають ефективніше керувати виробничими процесами. Процес розвитку не оминув і сферу освіти. Найновіші інформаційні технології дозволяють підвищити ефективність роботи навчального закладу. Для забезпечення оперативного

автоматизованого обліку інформації розробляються та впроваджуються системи, що сприяють успішній діяльності установ.

Реалізація комунікації між викладачем та студентом в електронному освітньому середовищі може відбуватись за допомогою різних методів та засобів [2]. До прикладу комунікація через оцінювання діяльності студента в режимі онлайн може слугувати джерелом мотивації. Чітке, своєчасне та справедливе оцінювання може стимулювати студентів до більш високих досягнень і більш активного вивчення матеріалу.

Існує безліч платформ для онлайн навчання [3]. В них можна обмінюватись інформацією, навчальними матеріалами між викладачами та студентами, вести оцінювання роботи студентів.

Одною із найпопулярніших платформ для онлайн навчання є Google Classroom. Вона інтегрується з іншими Google-сервісами, такими як Google Drive та Google Docs. Студенти та викладачі можуть обмінюватись інформацією через коментарі до завдань, а також за допомогою електронної пошти. Забезпечує можливість проведення відеоконференцій через Google Meet. Є можливість створення завдань, тестів та опитувань. Оцінки автоматично вносяться до зошита Google. Google Classroom має недолік, що пріоритет віддається обміну інформацією та навчальними матеріалами, але оцінювання автоматизоване лише для заздалегідь підготовлених тестів. Таке оцінювання може не зовсім підходити до навчального процесу, так як він складається не лише із тестів, але і робіт, які перевіряються викладачем на правильність виконання та дотримання всіх норм.

Наступна платформа Classtime, вона фокусується на інтерактивних заняттях та онлайн-тестуванні. Онлайн платформа забезпечує обмін повідомленнями між викладачами та студентами. Пропонує можливість взаємодії під час тестування. Спеціалізується на створенні та проведенні тестів, а також надає інструменти для їх оцінювання. Недолік даної платформи у тому, що є можливість оцінювати лише тести. Роботи для перевірки викладачем на правильність виконання не оцінюються, а вони є невід'ємною частиною навчального процесу.

Навчальна платформа N-Code, онлайн платформа для навчання ІТ технологіям. Комунікація між студентами та викладачами відбувається на великому календарі, де відображаються уроки, та періоди де ще можна забронювати уроки із викладачем. Можна залишати коментарі до кожного уроку. Також взаємодіяти із платформою можна за допомогою Telegram бота, де відслідковуються всі процеси в системі. Недоліком є повна відсутність будь якого оцінювання роботи студента в функціоналі платформи N-Code. Ефективне та своєчасну оцінювання є важливою складовою навчального процесу, та є мотивуючим фактором для студентів.

## Результати

Для реалізації подібних систем підходить клієнт-серверна архітектура (рис. 1). Вона є одним із архітектурних шаблонів програмного забезпечення та є домінуючою концепцією у створенні розподілених мережних застосунків і передбачає взаємодію та обмін даними між ними [4].

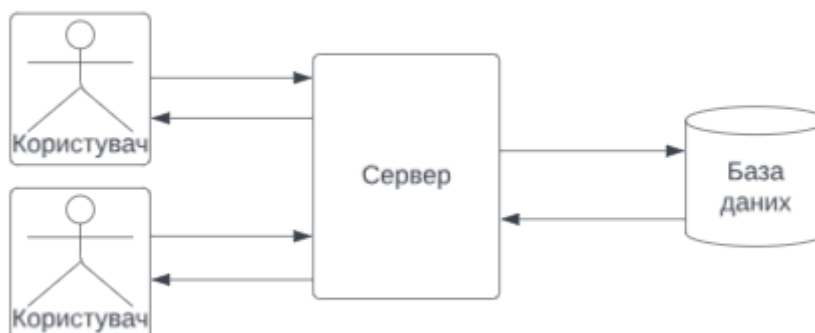


Рис. 1. Схема клієнт-серверної архітектури

Клієнт і сервер виконують різні функції, що дозволяє їм бути розділеними та незалежними один від одного. Це полегшує розробку, тестування і підтримку коду. Компоненти можуть бути масштабовані незалежно. Якщо потрібно збільшити витрати ресурсів для серверу, це може бути зроблено без змін у клієнті, і навпаки. Сервер може контролювати доступ до даних та логіки бізнес-процесів, забезпечуючи більший рівень безпеки. Використання контексту та хуків дозволяє легко управляти станом компонентів та спрощує код.

Для реалізації клієнтської частини системи обрано JavaScript бібліотеку React. Її основною перевагою є використання віртуального DOM, що призводить до ефективнішої роботи та високої продуктивності. React легко інтегрується з роутерами, дозволяючи зручно створювати SPA, що дає можливість працювати із сторінками без їх перезавантаження. Розробка на React базується на компонентах, що полегшує управління станом та створення коду, який можна використовувати декілька разів.

На відміну від React, Vue має меншу кількість готових бібліотек і компонентів, та менш активну спільноту. Це може вплинути на зручність пошуку рішень та обговорення проблем. Angular вимагає більше ресурсів для роботи порівняно з React або Vue, що може впливати на продуктивність додатку, особливо на стороні клієнта.

Для реалізації серверної частини системи обрано Node.js та фреймворк Express. Node.js використовує асинхронний підхід, що дозволяє обробляти багато запитів одночасно, не блокуючи інші операції. Це особливо важливо для високопродуктивних застосунків з великою кількістю одночасних підключень. Node.js використовує двигун V8 від Google Chrome, що забезпечує високу швидкість виконання коду. Використання JavaScript на клієнті та сервері може спростити розробку та обмін кодом між клієнтом і сервером.

На відміну від JavaScript, написання коду на Java зазвичай займає більше часу, також запуск нового потоку на кожен запит може призводити до виділення великої кількості ресурсів, що обмежує масштабованість. Rails може бути менш продуктивним в обробці багатьох одночасних підключень порівняно з Node.js, а також вартість його використання вища, особливо для великих проєктів.

На основі дослідження існуючих аналогів можна зробити висновок, що пріоритет зазвичай віддається подачі навчальних матеріалів та обміну інформації, а не ефективному та своєчасному оцінюванню роботи студентів. Відслідковування своїх результатів у режимі онлайн, є важливим фактором мотивації для студентів, також це дає можливість своєчасно дізнатись про свої недоліки у роботі, та виправити їх. Онлайн платформа, де ефективно ведеться оцінювання, дає можливість адміністрації системи, або закладу, який використовує систему, відслідковувати статистику груп, студентів, викладачів, своєчасно нагадувати та мотивувати студентів до покращення своїх результатів.

## Висновки

Під час виконання поставленої задачі було проаналізовано аналоги навчальних платформ, описані переваги та недоліки. Основний недолік, неефективна система оцінювання студентів. Були описані клієнт-серверна архітектура, що підходить для реалізації подібних систем, та технічні рішення для клієнтської на серверної частини системи. Основною метою роботи стало удосконалення методів та засобів реалізації комунікації викладачів, студентів та інших учасників навчального процесу, за рахунок зручного та інтуїтивно зрозумілого інтерфейсу, та автоматизації процесів.

## СПИСОК ВИКОРИСТАНОЇ ЛІТЕРАТУРИ

1. МарковаТ.В. Інформаційно - комунікативне середовище як засіб підвищення майстерності педагога: веб-сайт. URL: <https://core.ac.uk/reader/42973840>
2. Способи комунікації з учасниками освітнього процесу: веб-сайт. URL: [https://rogizno-school.at.ua/index/sposobi\\_komunikaciji\\_z\\_uchasnikami\\_osvitnogo\\_procesu/0-189](https://rogizno-school.at.ua/index/sposobi_komunikaciji_z_uchasnikami_osvitnogo_procesu/0-189)
3. Найпопулярніші освітні платформи для організації дистанційного навчання: URL: <https://op.ua/news/osvita-v-ukraini/naupopulyarnishi-osvitni-platформи-dlya-organizaciyi-distancijnogo-navchannya>

4. Клієнт-серверна архітектура : веб-сайт. URL: <https://training.qatestlab.com/blog/technical-articles/client-server-architecture/>.

***Сай Олександр Олегович*** – студент групи ІСТ-22м, кафедра автоматизації та інтелектуальних інформаційних технологій, факультет інтелектуальних інформаційних технологій та автоматизації, Вінницький національний технічний університет, м.Вінниця, e-mail: [sashasai2311@gmail.com](mailto:sashasai2311@gmail.com)

***Паламарчук Євген Анатолійович*** – д.т.н., доцент кафедри автоматизації та інтелектуальних інформаційних технологій, факультет інтелектуальних інформаційних технологій та автоматизації, Вінницький національний технічний університет, м.Вінниця, e-mail: [p@vntu.edu.ua](mailto:p@vntu.edu.ua)

***Sai Oleksandr Olegovich*** – student of IIST-22M group, Department of Automatization and Intellectual Informational Technologies, Faculty of Intelligent Information Technology and Automation, Vinnytsia National Technical University, Vinnytsia, e-mail: [sashasai2311@gmail.com](mailto:sashasai2311@gmail.com)

***Palamarchuk Eugene Anatolyevich*** – Associate Professor of Automation and Intelligent Information Technologies Department, Faculty of Intelligent Information Technology and Automation, Vinnytsia National Technical University, Vinnytsia, e-mail: [p@vntu.edu.ua](mailto:p@vntu.edu.ua)