

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
Національний університет «Запорізька політехніка»

Інститут інформатики та радіoeлектроніки,
Факультет комп'ютерних наук і технологій
(повне найменування інституту, назва факультету)

Кафедра програмних засобів
(повне найменування кафедри)

Пояснювальна записка

до дипломного проєкту (роботи)

магістр

(ступінь вищої освіти)

на тему ДОСЛІДЖЕННЯ ТА РОЗРОБКА ЗАСОБІВ ВЗАЄМОДІЇ
ІНСТРУМЕНТІВ ОНЛАЙН НАВЧАННЯ ДЛЯ РЕАЛІЗАЦІЇ
МІЖНАРОДНИХ НАВЧАЛЬНИХ ПРОЄКТІВ
RESEARCH AND DEVELOPMENT OF INTERACTION TOOLS
FOR ONLINE STUDY INSTRUMENTS TO IMPLEMENT
INTERNATIONAL EDUCATIONAL PROJECTS

Виконав: студент(ка) 2 курсу, групи КНТ-119м
Спеціальності 121 Інженерія програмного
забезпечення

(код і найменування спеціальності)

Освітня програма (спеціалізація)

Інженерія програмного забезпечення

Чадаєва А.К.

(прізвище та ініціали)

Керівник Шитікова О.В.

(прізвище та ініціали)

Рецензент Терещенко Е.В.

(прізвище та ініціали)

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
Національний університет «Запорізька політехніка»
 (повне найменування закладу вищої освіти)

Інститут, факультет ІРЕ, ФКНТ
 Кафедра програмних засобів
 Ступінь вищої освіти магістр
 Спеціальність 121 Інженерія програмного забезпечення
 (КОД І НАЙМЕНУВАННЯ)
 Освітня програма (спеціалізація) Інженерія програмного забезпечення
 (назва освітньої програми (спеціалізації))

ЗАТВЕРДЖУЮ

Завідувач кафедри ПЗ, д.т.н, проф.
С.О. Субботін
 “ ” 20 року

З А В Д А Н Н Я

НА ДИПЛОМНИЙ ПРОЄКТ (РОБОТУ) СТУДЕНТА(КИ)

Чадаєвої Анастасії Костянтинівни

(прізвище, ім'я, по батькові)

1. Тема проєкту (роботи) Дослідження та розробка засобів взаємодії інструментів онлайн навчання для реалізації міжнародних навчальних проєктів. Research and Development of Interaction Tools for Online Study Instruments to Implement International Educational Projects

керівник проєкту (роботи) Шитікова Олена Вікторівна, к.т.н.

(прізвище, ім'я, по батькові, науковий ступінь, вчене звання)

затверджені наказом закладу вищої освіти від “03” листопада 2020 року № 301

2. Строк подання студентом проєкту (роботи) 5 грудня 2020 року

3. Вихідні дані до проєкту (роботи) система дистанційного навчання Moodle, навчальні курси, користувачі системи: учні, студенти, працівники компаній, керівники курсів

4. Зміст розрахунково-пояснювальної записки (перелік питань, які потрібно розробити) 1. Аналіз проблеми та постановка завдань дослідження. 2. Огляд інструментів сумісності з навчальними середовищами. 3. Проєктування програмного застосунку. 4. Основні рішення щодо реалізації компонентів системи. 5. Експлуатація програмного застосунку. 6. Економіко-організаційна частина. 7. Охорона праці та безпека у надзвичайних ситуаціях. 8. Додатки.

5. Перелік графічного матеріалу (з точним зазначенням обов'язкових креслень) Слайди презентації

6. Консультанти розділів проєкту (роботи)

Розділ	Прізвище, ініціали та посада консультанта	Підпис, дата	
		завдання видав	прийняв виконане завдання
1-5 Основна частина	Шитікова О.В., доцент		
Економіко-організаційна частина	Пожуєва Т.О., професор		
Охорона праці та безпека у надзвичайних ситуаціях	Коробко О.В., ст. викладач		
Нормоконтролер	Андрєєв М.О., асистент		

7. Дата видачі завдання “ 4 ” вересня 2020 року.

КАЛЕНДАРНИЙ ПЛАН

№ з/п	Назва етапів дипломного проєкту (роботи)	Строк виконання етапів проєкту (роботи)	Примітка
1	Постановка завдання роботи.	1 тиждень	Завдання, ТЗ
2	Аналіз предметної області.	2–3 тижні	Розділ 1
3	Розробка та удосконалення методів, моделей й алгоритмів вирішення задачі.	4–5 тижні	Розділ 2
4	Розробка архітектури програми.	6 тиждень	Розділ 3
5	Розробка програми.	7–8 тижні	Розділ 4
6	Тестування та експериментальне дослідження програмного забезпечення.	9 тиждень	Розділ 5
7	Розробка економічно-організаційної частини	10 тиждень	Розділ 6
8	Розробка заходів з охорони праці та безпеки у надзвичайних ситуаціях	11 тиждень	Розділ 7
9	Оформлення пояснювальної записки та документів до неї.	12-13 тиждень	Додатки
10	Нормоконтроль та рецензування.	14–15 тижні	
11	Захист роботи.	16 тиждень	

Студент(ка)

(підпис)

Чадаєва А.К.

(прізвище та ініціали)

Керівник проєкту (роботи)

(підпис)

Шитікова О.В.

(прізвище та ініціали)

РЕФЕРАТ

Пояснювальна записка до дипломної кваліфікаційної роботи магістра:
113 с., 14 табл., 36 рис., 2 дод., 70 джерел.

СИСТЕМА ДИСТАНЦІЙНОГО НАВЧАННЯ, MOODLE, МОДУЛЬ
АКТИВНОСТІ, ПРИКЛАДНИЙ ПРОГРАМНИЙ ІНТЕРФЕЙС, ВІДГУК.

Об'єкт дослідження – засоби взаємодії інструментів онлайн навчання.

Предмет дослідження – інтегровані програмні засоби до систем дистанційного навчання для розширення функціональних можливостей.

Мета роботи – огляд та розробка засобів розширення функціональних можливостей систем дистанційного навчання на прикладі інтеграції плагіну зворотнього зв'язку до системи дистанційного навчання Moodle з метою надання студентами об'єктивного анонімного відгуку про курс.

Матеріали, методи та технічні засоби – мови програмування Javascript, Typescript, платформа Node.js та бібліотека Express.js.

Результати. Розроблено плагін анонімного зворотнього зв'язку впроваджений до системи дистанційного навчання Moodle.

Висновки. Створена система, що вирішує поставлені задачі.

Галузь використання – розширення функціоналу систем дистанційного навчання.

ABSTRACT

Explanatory note to the diploma qualifying work of the master: 113 pages, 14 tables, 36 figures, 2 appendixes, 70 sources.

LEARNING MANAGEMENT SYSTEM, MOODLE, ACTIVITY MODULE, APPLICATION PROGRAMMING INTERFACE, FEEDBACK.

Object of study – the means of interaction tools for online study instruments.

Subject of study – an integrated software for extending of learning management systems functionality.

The purpose of the work is research and development of means of extending learning management systems functionality by integrating a feedback module into Moodle LMS to provide students with objective anonymous feedback to the course.

Design tools – Javascript, Typescript programming languages, Node.js platform and Express.js library.

Results. An anonymous feedback module implemented in the Moodle learning management system was developed.

Conclusion. A software system has been created that solves the given set of problems.

Field of use – extending the functionality of learning management systems.

ЗМІСТ

	С.
Перелік скорочень та умовних позначок.....	9
Вступ.....	10
1 Аналіз проблеми та постановка завдань дослідження.....	12
1.1 Поняття системи дистанційного навчання, онлайн навчальних платформ.....	12
1.2 Основні критерії онлайн навчальних платформ.....	12
1.3 Огляд існуючих систем дистанційного навчання.....	12
1.3.1 Огляд системи ILIAS.....	13
1.3.2 Огляд системи Sakai.....	14
1.3.3 Опис системи Docebo.....	15
1.3.4 Опис системи Moodle.....	16
1.3.5 Порівняльна таблиця існуючих LMS.....	17
1.4 Огляд інструментів взаємодії онлайн навчання для систем дистанційного навчання.....	19
1.4.1 Дослідження типів плагінів.....	19
1.4.2 Дослідження типів сторонніх застосунків.....	20
1.4.3 Дослідження типів плагінів та сторонніх застосунків Moodle.....	21
1.5 Огляд засобів діджиталізації для систем дистанційного навчання.....	22
1.5.1 Засоби впровадження відеоконференцій.....	22
1.5.2 Засоби впровадження інструментів перевірки програм.....	25
1.6 Засоби впровадження навчальних платформ.....	26
1.7 Висновки з розділу 1.....	27
2 Огляд інструментів сумісності з навчальними середовищами.....	28
2.1 Огляд інструментів сумісності з навчальними середовищами.....	28
2.1.1 Поняття набору стандартів SCORM.....	29
2.1.2 Поняття стандарту LTI.....	30
2.2 Вибір системи дистанційного навчання.....	33
2.2.1 Поняття інструменту External tool.....	33

2.3 Висновки з розділу 2	34
3 Проєктування програмного застосунку	35
3.1 Розгортання та вимоги системи дистанційного навчання Moodle	35
3.2 Вибір мови програмування для реалізації програмного застосунку	38
3.3 Вибір технологій для реалізації програмного застосунку	39
3.3.1 NodeJS	39
3.3.2 Typescript.....	39
3.3.3 ExpressJS	40
3.4 Висновки з розділу 3	40
4 Основні рішення щодо реалізації компонентів системи.....	41
4.1 Алгоритм роботи застосунку	41
4.2 Головні модулі розробленого застосунку	42
4.3 Висновки з розділу 4	46
5 Експлуатація програмного застосунку	48
5.1 Призначення програми	48
5.2 Умови виконання програми	48
5.3 Додавання застосунку до системи	48
5.4 Використання застосунку	50
5.5 Функціональні можливості розробленого застосунку	51
5.6 Висновки з розділу 5	53
6 Економіко-організаційна частина	54
6.1 Планування розробки програмного виробу.....	55
6.2 Визначення витрат на розробку програми.....	57
6.2.1 Розрахунок основної і додаткової заробітної плати.....	58
6.2.2 Відрахування на єдиний соціальний внесок	59
6.2.3 Визначення витрат на матеріали	60
6.2.4 Витрати на спеціальне обладнання	61
6.2.5 Інші витрати	64
6.2.6 Накладні витрати	65
6.3 Калькуляція кошторисної вартості робіт	65

6.4 Оцінка економічної ефективності.....	66
7 Охорона праці та безпека у надзвичайних ситуаціях.....	68
7.1 Аналіз потенціальних небезпек	68
7.2 Заходи щодо забезпечення безпеки	70
7.3 Заходи щодо забезпечення виробничої санітарії та гігієни праці.....	73
7.4 Заходи щодо забезпечення пожежної безпеки	78
7.5 Заходи щодо забезпечення безпеки в надзвичайних ситуаціях	83
7.5.1 Порядок проведення аварійно-рятувальних та інших невідкладних робіт	83
7.6 Висновки з розділу 7	86
Висновки	88
Перелік джерел посилання.....	90
Додаток А Текст програми.....	97
Додаток Б Слайди презентації	103

ПЕРЕЛІК СКОРОЧЕНЬ ТА УМОВНИХ ПОЗНАК

- API – Application Programming Interface;
LMS – E-learning Management System;
LTI – Learning Tools Interoperability;
SCORM – Sharable Content Object Reference Model;
ПЗ – програмне забезпечення;
СКБД – система керування базами даних.

ВСТУП

Актуальність теми. Концепція дистанційного навчання просувалася ще з початку ХХ сторіччя. На сьогодні дистанційна освіта здійснюється з використанням ресурсів та технологій глобальної мережі Інтернет, за допомогою локальних мереж та відеозв'язку тощо. Поява і активне поширення дистанційних форм навчання є адекватним відгуком систем освіти багатьох країн на процеси інтеграції, рух до інформаційного суспільства, що в свою чергу призводить до проблеми вибору платформи для впровадження освітніх послуг.

Онлайн навчальні платформи є потужними програмними системами та містять широкий набір можливостей, що призначені для забезпечення доставки інформації в інтерактивному режимі за допомогою використання інформаційно-комунікаційних технологій. Тим самим процес навчання полегшується і стає доступним. Завдяки дистанційному навчанню, студент може прослухати той же курс в зручній і комфортній для нього обстановці в будь-який час. Тому основною метою таких систем є економія часу, зусиль, зберігання та розповсюдження усієї потрібної інформації, а також різноманітний функціонал, але незважаючи на це – виникає необхідність розширення функціональних можливостей платформ.

Мета і задачі дослідження. Опираючись на всі ці показники була поставлена задача дослідження засобів взаємодії інструментів онлайн навчання, тобто: проведення огляду існуючих систем з виявленням всіх переваг і недоліків, інтегрованих плагінів і сторонніх застосунків, а також розглядання принципів реалізації даних систем для вибору найоптимальнішого шляху рішень.

Для досягнення поставленої мети в роботі потрібно вирішити наступні задачі:

- виконати дослідження систем дистанційного навчання;

- виконати дослідження засобів взаємодії інструментів онлайн навчання;
- виконати дослідження засобів впровадження сторонніх модулів у систему дистанційного навчання Moodle;
- розробити модуль зворотнього зв'язку;
- виконати інтеграцію модулю зворотнього зв'язку у систему дистанційного навчання Moodle;
- виконати експериментальне дослідження розробленої системи.

Об'єкт дослідження – засоби взаємодії інструментів онлайн навчання.

Предмет дослідження – інтегровані програмні засоби до систем дистанційного навчання для розширення функціональних можливостей.

Методи дослідження. Розроблений модуль інтегрується до системи за LTI протоколом.

Наукова новизна роботи полягає в тому, що виконано дослідження засобів інтеграції у системи дистанційного навчання.

Практична цінність результатів роботи полягає в тому, що розроблене програмне забезпечення надає можливість користувачам системи Moodle залишити анонімний відгук про курс керівнику.

Апробація результатів. За результатами роботи було опубліковано тези, що були подані на X Міжнародну науково-практичну конференцію «Сучасні проблеми і досягнення в галузі радіотехніки, телекомунікацій та інформаційних технологій» на тему «Дослідження засобів діджиталізації для систем дистанційного навчання» [1].

Дослідження виконано в рамках міжнародного проекту Virtual Master Cooperation Data Science (ViMaCs) за підтримки DAAD.

1 АНАЛІЗ ПРОБЛЕМИ ТА ПОСТАНОВКА ЗАВДАНЬ ДОСЛІДЖЕННЯ

1.1 Поняття системи дистанційного навчання, онлайн навчальних платформ

Дистанційна освіта – сукупність сучасних технологій для забезпечення доступу до навчальних матеріалів в інтерактивному режимі за допомогою використання інформаційно-комунікаційних технологій до тих, хто навчається – студентів чи слухачів, від тих, хто навчає – викладачів [2].

Система дистанційного навчання або e-learning management system (LMS) – це система управління навчальною діяльністю, яка використовується для розробки, управління та поширення навчальних онлайн-матеріалів із забезпеченням спільного доступу. Створюються дані матеріали з завданням послідовності вивчення [3].

Онлайн навчальні платформи є потужними програмними системами, що містять широкий набір можливостей.

1.2 Основні критерії онлайн навчальних платформ

Система включає наступні основні критерії на основі змісту та комунікаційного компонента [3]:

- вибір навчального курсу;
- заняття зі студентами;
- індивідуальні завдання;
- проекти для роботи в малих групах;
- здача завдань, проміжних і фінальних онлайн іспитів та тестувань.

1.3 Огляд існуючих систем дистанційного навчання

Існує безліч платформ для дистанційного навчання або навчання з використанням елементів електронного навчання, але при виборі конкретної

платформи слід визначити конкретну стратегію і цілі, для чого саме буде впроваджуватися та чи інша платформа. Якщо мова про вищу освіту, то необхідно з'ясувати: для електронізації навчального процесу впроваджується або ж для відкриття саме дистанційного навчання. Якщо будь-яка установа хоче впровадити цю систему, то перед нею постає питання вибіру тієї чи іншої системи дистанційного навчання [10]-[12].

В ході дослідження було проведено огляд декількох ведучих систем дистанційного навчання. Серед усіх розглянутих навчальних середовищ було виділено наступні: ILIAS, Sakai, Docebo, Moodle.

1.3.1 Огляд системи ILIAS

ILIAS – це потужний і гнучкий інструмент для навчання та спільної роботи в Інтернеті [4]. Він служить платформою знань і співробітництва. ILIAS не обмежується будь-якої конкретної моделлю, а підтримує безліч застосувань і методів. Відповідає стандарту SCORM. Слід зазначити, що ILIAS – одна з небагатьох систем дистанційного навчання, яка може бути використана як повноцінна платформа для електронного навчання, завдяки здатності спілкування в групі, передачі та зберігання всіх документів. Система безкоштовна для всіх організацій, що займаються електронним навчанням, незалежно від кількості користувачів. Платформа постійно розвивається і включає в себе:

- особистий робочий стіл з портфоліо однокурсників онлайн, нові повідомлення електронної пошти, нові повідомлення на форумі;
- контекстна довідкова система для студентів і авторів;
- інтерфейс користувача та адміністратора;
- інтерфейс для імпорту вмісту і користувачів;
- багатомовність, підтримка української мови;
- нові версії виходять кожні кілька місяців.

Серед переваг можна виділити загальну статистику, що відображає середні показники, успішність виконання тестів, а також електронну пошту і нотатки. Керування успішністю навчання надає список користувачів, що заходили недавно на «курс». Список доступного для роботи матеріалу. Список активних тем на форумі. А також новини проєкту платформи.

Із недоліків даної системи можна виділити відносно застарілий дизайн і менш зрозумілий і інтуїтивний інтерфейс [4], [12].

1.3.2 Огляд системи Sakai

Sakai – це програмна оболонка с відкритим вихідним кодом на Java, яку розробляє і використовує міжнародне співтовариство найбільших вузів світу. Це проєкт, що постійно розвивається. Представляє собою систему з повністю відкритим вихідним кодом, яка підтримується спільнотою розробників. У систему інтегрована підтримка стандартів і специфікацій SCORM [5], [6].

LMS Sakai включає в себе безліч можливостей по розробці курсів і організації системи управління курсами, а також управління проходженням документів, форуми, чати, онлайн-тестування.

Sakai є єдиним інформаційним середовищем, де все що стосується освітнього курсу зібрано в одному місці: програма курсу, лекції, завдання, тести для самоконтролю, інформація про консультації та успішності, а також зворотній зв'язок. Студенту пропонується багато можливостей для спілкування з викладачами в системі Sakai, використовуючи такі інструменти як форуми, вебіари, чат і повідомлення. Контроль якості навчання здійснюється за допомогою інструменту статистики, що допомагає викладачеві відстежувати, як часто студент заходить на курс і яким ресурсом користується найчастіше. Всі результати роботи студента за курс можна подивитися в інструменті «залікова книжка», яку можна використовувати як контроль за самостійною роботою студента. В інструменті «оголошення»

викладач може розмістити всю інформацію про консультації і результати контрольних робіт.

Із недоліків даної системи можна виділити те, що LMS Sakai містить платні програми, термін демонстрації яких може закінчитися. Наприклад, редактор формул, а це для розробника курсу, у якого через слово формула, великий мінус, тобто всі символи, коріння, дробу і графіки доводиться вставляти як картинку [5], [6], [11].

1.3.3 Опис системи Docebo

Docebo – це сучасна платформа дистанційного навчання, призначена для підвищення продуктивності і залученості в процес навчання. Система відрізняється простотою використання і здатністю поєднувати менторство з соціальним і формальним навчанням. Система допомагає студентам вчити швидше і бути більш ефективними [7].

Це комплексне рішення для навчання студентів забезпечує повне навчання з екосистемою модулів і функцій, які можуть бути адаптовані до будь-яких вимог.

Платформа для різних видів навчання, як формального, так і експериментального. Входить до двадцятки кращих LMS. У платформі чотири модулі: Learn, Coach and Share, Extended Enterprise, Perform. Основним модулем є Learn, який входить в базову вартість, інші потрібно докуповувати. Learn є модулем формального навчання, де ви можете завантажувати навчальні курси, створювати навчальні програми, робити тести і звіти. Coach and Share – модуль неформального соціального навчання. У ньому студенти можуть задавати питання викладачам і ділитися власним досвідом.

Із переваг даної системи можна виділити сучасний, акуратний користувальницький інтерфейс. Сервіс можна налаштовувати і модифікувати зовнішній вигляд для різних ролей. Проста інтеграція з бізнес-системами. Можливість проводити онлайн-конференції з використанням відео. Можна

продавати онлайн-курси і змінювати контент для іншої аудиторії. Має одну з найгнучкіших модульних конструкцій, яка дозволяє змінювати функції відповідно до вимог.

Із недоліків – система не безкоштовна і не має української локалізації, але є русифікована версія. В інтерфейсі створення курсів та адміністратора системи все не відразу зрозуміло. Проте керівництво користувача написане вдало і доступно. Офіційна документація лише англійською та італійською мовами, але в Інтернеті достатньо інформації українською та російською [7], [12].

1.3.4 Опис системи Moodle

Moodle (Modular Object-Oriented Dynamic Learning Environment) – модульне об'єктно-орієнтоване середовище дистанційного навчання — безкоштовна LMS. На сьогоднішній день Moodle безсумнівно одна з найпопулярніших LMS з відкритим вихідним кодом. Система реалізує філософію «соціально-педагогічного конструктивізму» і зосереджується головним чином на організації взаємодії між викладачами та студентами, хоча вона підходить і для організації традиційних курсів дистанційного навчання. [8], [9].

Moodle пропонує користувачам можливість контролювати різні панелі інструментів, відстежувати успіхи учнів та підтримувати мультимедіа. Система дозволяє створювати індивідуальні курси для мобільних пристроїв і дуже зручна для інтеграції сторонніх доповнень від розробників.

Крім того, система має низку готових шаблонів, за допомогою яких можна заощадити час і не створювати курс з нуля.

Moodle перекладена на десятки мов, у тому числі й на українську. Система використовується у 175 країнах світу [11].

Типова функціональність Moodle, окрім стандартних вимог до LMS, включає [9]-[12]:

- доступ до навчальних матеріалів (тексти лекцій, завдання до практичних/лабораторних та самостійних робіт; додаткові матеріали (книги, довідники, посібники, методичні розробки) та засобів для спілкування і тестування «24 на 7»;
- засоби для групової роботи (вікі, форум, чат, опитування, семінар, вебінар);
- завантаження файлів, електронних курсів, тестів, книг, текстових документів, відеолекцій;
- керування якістю навчання (через систему можна подивитись як часто студенти користуються площиною);
- можливість відображати результати проходження тестів, іспитів, дистанційного навчання студентом, тощо;
- календар подій;
- новини та анонси подій (для різних рівнів: сайт, курс, навчальна група), а також можливість використання нагадувань про події у курсі.

Перевагами даної системи є надання користувачеві максимальної кількості свобод. LMS націлена на активну роботу зі студентами. Система адаптована під мобільні пристрої, підтримує сторонні плагіни, наприклад, інтеграцію з Microsoft OneNote або Microsoft Office. Також серед всіх безкоштовних LMS, користувач, звернувшись до бази технічної підтримки онлайн, може миттєво отримати відповіді на більшість питань, що цікавлять.

Недоліками даної системи є складний, інтуїтивно незрозумілий інтерфейс Moodle, що бере час на вивчення. Користувачів не можна поділити по групам, створювання груп учнів можливо тільки всередині курсу [10]-[12].

1.3.5 Порівняльна таблиця існуючих LMS

В таблиці 1.1 наведено порівняння переваг та недоліків існуючих LMS, що вирішують аналогічні задачі.

Таблиця 1.1 – Порівняння існуючих аналогічних LMS

Критерій порівняння	Назва LMS			
	ILIAS	Sakai	Docebo	Moodle
Авторизація користувача	+	+	+	+
Безкоштовна	+	+	-	+
Обмеження безкоштовної версії	-	+	+	-
Багатомовний інтерфейс	+	+	+	+
Ліцензія	GNU GPL	GNU GPL	GNU GPL	GNU GPL
Підтримка української мови	+	+	-	+
Підтримка SCORM	+	+	+	+
Підтримка IMS	+	+	+	+
Підтримка LTI	+	+	+	+
Структура	Ядро+набір модулів	Ядро+набір модулів	Ядро+набір модулів	Ядро+набір модулів
Можливість інтеграції доповнень	+	+	+	+
Гнучкість платформи	+	+	+	+
Багатоплатформність	+	+	+	+
Середа розробки учбового матеріалу	+	+	+	+
Система перевірки знань	+	+	+	+
Система зворотного зв'язку	+	+	+	+

Продовження таблиці 1.1

Критерій порівняння	Назва LMS			
	ILIAS	Sakai	Docebo	Moodle
Керування якістю навчання	+	+	+	+
Обмеження на кількість слухачів	-	-	-	-

Проведений порівняльний огляд систем дистанційного навчання показав, що система Moodle є найбільш придатною системою для організації дистанційного навчання.

1.4 Огляд інструментів взаємодії онлайн навчання для систем дистанційного навчання

Можна виділити два принципово різні підходи до розширення функціональних можливостей систем дистанційного навчання:

- розробка внутрішніх модулів системи та інтегрованих у систему безпосередньо;
- розробка зовнішніх незалежних застосувань, які інтегруються з системою.

Для більш детального аналізу було проведено дослідження типів плагінів та вибір, які саме плагіни, корисні з точки зору студента, підходять до діджиталізації навчання.

1.4.1 Дослідження типів плагінів

Більшість систем дистанційного навчання, що підтримують можливість розширення власного основного функціоналу, використовують перший спосіб створення модулів – внутрішні модулі.

В ході розробки таких модулів використовується та сама мова програмування, за якої розроблена й сама система, та опирається на наданий набір класів розробниками системи. Виходячи з цього, можна зробити висновок, що для реалізації більшості додаткового функціоналу цього достатньо, але це надає обмеження.

Більшість систем дистанційного навчання підтримує такі типи модулів:

- типи діяльностей (включаючи можливі навчальні ігри);
- типи ресурсів;
- типи тестових питань;
- типи полів для бази даних (мається на увазі діяльність баз даних);
- теми для оформлення;
- різні методи автентифікації;
- різні методи зарахування на курс;
- фільтри для контенту [13].

1.4.2 Дослідження типів сторонніх застосунків

Для розширення систем дистанційного навчання у такий спосіб можна виділити другий підхід – організація взаємодії системи дистанційного навчання та стороннього застосунку. Таким чином, застосунок та система обмінюються даними, а обробка цих даних проходить незалежно, на стороні самого застосунку. Використання такого підходу дозволяє відкинути обмеження на мову програмування, платформу та умови використання застосунку, що розширює функціонал.

Було розглянуто цілу низку різноманітних сторонніх програм, серед них було обрано наступні [13], [14]:

- сервіси для проведення відеоконференцій та звукозапису;
- сервіси для перевірки програмного коду;
- віртуальні лабораторії.

1.4.3 Дослідження типів плагінів та сторонніх застосунків Moodle

Moodle активно підтримує всі зазначені типи модулів. Використовуючи цю можливість, розробниками було створено багато застосунків з додатковою функціональністю, як приклад, таких як [13], [14]:

- Database – база даних;
- Workshop – лабораторія;
- Quiz – інструмент для опитувань;
- Book – надає інформацію про книги;
- Forum – спілкування;
- Feedback – внутрішній модуль зворотного зв'язку для проведення досліджень і опитувань.

Існує багато корисних сторонніх застосунків, які необхідні для підтримки дистанційного навчання студентами, які можуть інтегруватися до існуючих LMS. Серед них основна увага приділяється стороннім сервісам для проведення відеоконференцій, інструментам перевірки програмного коду, а також віртуальним лабораторіям.

Серед сторонніх застосунків LMS Moodle підтримує, як приклад, наступні [13], [14]:

- BigBlueButton – сервіс для проведення відеоконференцій;
- Zoom – сервіс для проведення відеоконференцій;
- GoToWebinar – сервіс для проведення відеоконференцій;
- Virtual program Lab – інструмент перевірки програмного коду;
- VmWare Learning Platform – платформа, яка надає практичні віртуальні лабораторії.

1.5 Огляд засобів діджиталізації для систем дистанційного навчання

1.5.1 Засоби впровадження відеоконференцій

Виходячи з отриманих результатів огляду сторонніх застосунків для проведення відеоконференцій було обрано сторонній сервіс BigBlueButton [16], [17].

BigBlueButton (BBB) — програмне забезпечення з відкритими вихідними кодами для проведення веб-конференцій [15], розроблене в першу чергу для дистанційної освіти, що забезпечує обмін аудіо, відео, слайдами, дошкою, чатом та доступом до екрану у режимі реального часу. Це також дозволяє учасникам приєднуватися до конференцій за допомогою своїх веб-камер та запрошувати спікерів.

Можливості сервісу BigBlueButton [15], [16]:

- підтримує наявність декількох аудіодоріжок і обмін відео;
- можливість показу презентацій, документів Microsoft Office і OpenOffice, зображень, PDF документів у реальному часі;
- підтримуються розширені можливості багатокористувацької дошки — такі, як покажчик, масштабування і малювання;
- трансляція робочого столу для всіх користувачів;
- для зворотного зв'язку зі слухачами веб-конференції існують публічні та приватні чати;
- у віртуальній кімнаті BigBlueButton передбачено три ролі: глядач, ведучий і модератор. Як глядач, користувач може приєднатися до голосової конференції, скористатися веб-камерою, попросити слово, і спілкуватися з іншими людьми. В якості модератора користувач має можливість відключити або включити мікрофон будь-якого глядача, видалити будь-якого глядача з веб-конференції, а також передати слово будь-якому глядачеві для виступу, тобто зробити будь-якого користувача ведучим. Ведучий може завантажувати презентації, документи, використовувати дошку.

До переваг системи можна віднести безкоштовність, відкритість та низьке навантаження на канал. Серед недоліків – небажані звукові ефекти під час аудіоконференції, тому цю платформу краще використовувати для проведення веб-лекцій, на яких доповідає тільки лектор.

В платформі BigBlueButton значно менший функціонал і відповідно зручний зрозумілий інтерфейс. Водночас, завдяки невеликій кількості функціональних можливостей платформу BigBlueButton можуть використовувати недосвідчені користувачі [17]. На рисунку 1.1 наведений інтерфейс запущеного клієнту BigBlueButton запущеного в браузері Google Chrome.

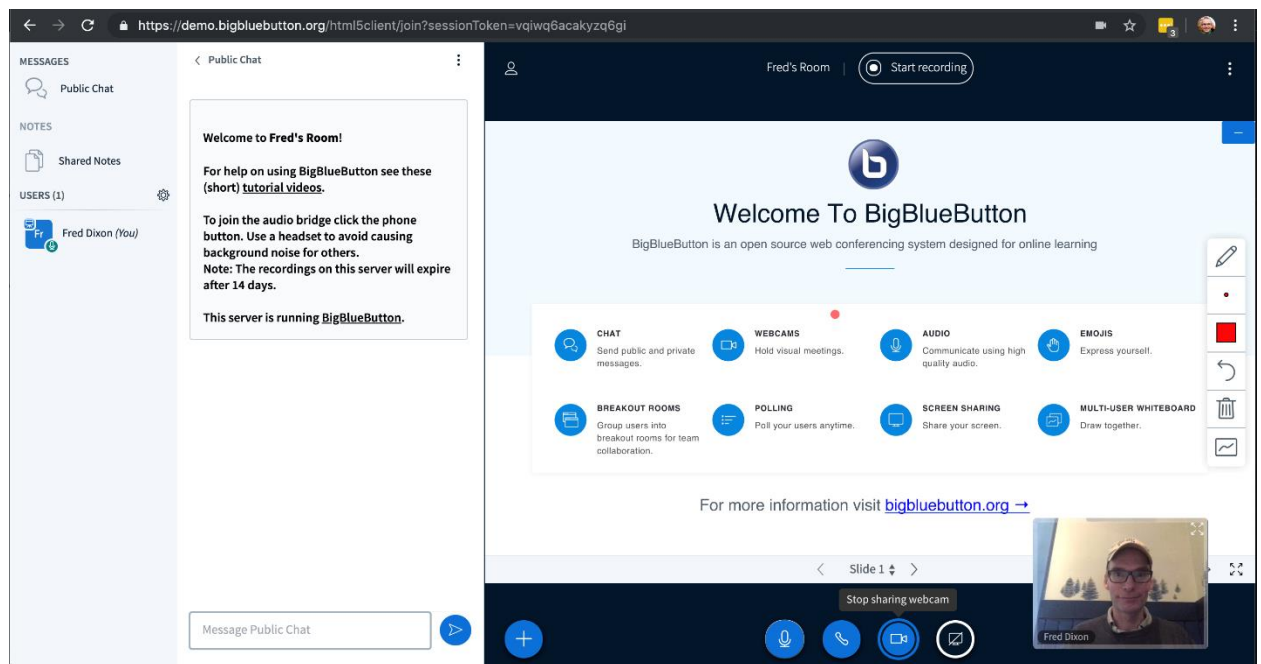


Рисунок 1.1 – Клієнт BigBlueButton запущений в браузері Google Chrome [17]

Було проведено огляд сервісу BigBlueButton. В таблиці 1.2 наведено огляд сервісу BigBlueButton.

Таблиця 1.2 – Огляд сервісу BigBlueButton.

Критерій порівняння	Назва сервісу
	BigBlueButton
Тип	веб
Відкритий вихідний код	+
Інтерфейс	веб
Зручність використання	+
Браузери, WebRTC	WebRTC – стандарт, що підтримується основними браузерами, включаючи Chrome, FireFox, Safari та Safari Mobile
Засоби використання	В браузері за допомогою додатку Greenlight, в LMS, як приклад, Moodle
Операційна система	Linux і UNIX-подібні операційні системи
Засоби установки сервера	Може бути установлений на операційній системі Ubuntu як з вихідного коду, так і з пакетів Ubuntu; працює як образ для віртуальної машини VMware Player для Windows або Unix, також VMWare Fusion на MacOS; сервер BBB може працювати в хмарній середі
Ліцензія	LGPL (Lesser General Public License)
Сертифікований LTI	+
Можливості	Публічні та приватні чати, відео, аудіо, багатокористувацькі дошки, трансляція робочого столу, опитування

Було проведено огляд існуючих LMS, що підтримують впровадження сервісу BigBlueButton [18]. В таблиці 1.3 наведено існуючі LMS, що підтримують впровадження сервісу BigBlueButton [19]-[21].

Таблиця 1.3 – Підтримка BigBlueButton існуючими LMS.

Критерій порівняння	Назва LMS			
	ILIAS	Sakai	Docebo	Moodle
BigBlueButton	-	+	+	+

1.5.2 Засоби впровадження інструментів перевірки програм

В ході виконання роботи основною платформою для перевірки програм було обрано Virtual program Lab (VPL) [22].

Virtual program Lab (VPL) – лабораторія віртуального програмування – це безкоштовний модуль діяльності с відкритим вихідним кодом, який керує завданнями програмування і основними функціями якого є [22]:

- редагування вихідного коду програм у браузері;
- запуск інтерактивних програм в браузері;
- запуск тестів для перегляду програм;
- пошук схожості між файлами;
- установка обмежень на редагування та уникнення зовнішнього вставлення тексту.

Особливості редагування, запуску та оцінки програм роблять процес навчання для студентів та завдання оцінювання для викладачів значно простішими.

В таблиці 1.4 наведено існуючі LMS, що підтримують впровадження сервісу Virtual program Lab.

Таблиця 1.4 – Підтримка Virtual program Lab існуючими LMS

Критерій порівняння	Назва LMS			
	ILIAS	Sakai	Docebo	Moodle
Virtual program Lab	-	-	-	+

1.6 Засоби впровадження навчальних платформ

На основі проведеного огляду сторонніх застосунків було обрано VmWare Learning Platform – платформу, яка надає практичні віртуальні IT-лабораторії в хмарному масштабі [23].

VmWare Learning Platform – це повне навчальне середовище, яке об'єднує навчальні матеріали, багаті лабораторії віртуальних машин (VM), відео, соціальні медіа тощо. VmWare Learning Platform заснована на хмарі, а її інструмент управління вмістом включає функції для запуску та відстеження навчальних лабораторій та курсів [23].

VmWare Learning Platform надає наступні можливості для користувачів [23]:

- запуск, зупинка і відновлення навчальних лабораторій, в яких ви взаємодієте із імітованими середовищами, такими як віртуалізовані робочі середовища або корпоративні програми;
- відстеження навчальної лабораторії та курсів, якими користується студент;
- перегляд стенограм, які показують завершені користувачем навчальні лабораторії та курси;
- перегляд каталогу усіх доступних навчальних лабораторій та курсів VmWare.

В таблиці 1.5 наведено існуючі LMS, що підтримують впровадження сервісу VmWare Learning Platform.

Таблиця 1.5 – Підтримка VmWare Learning Platform існуючими LMS

Критерій порівняння	Назва LMS			
	ILIAS	Sakai	Docebo	Moodle
VmWare Learning Platform	-	-	+	+

1.7 Висновки з розділу 1

В ході роботи було розглянуто цілу низку систем дистанційного навчання, плагінів і сторонніх засобів, які інтегруються в системи. Було проведено огляд ключових функціональних можливостей систем і загальних характеристик. Серед можливих функцій було проведено огляд необхідних ключових можливостей для будь-якої системи, так і додаткові можливості у вигляді плагінів і сторонніх застосунків, які полегшують роботу викладача чи студентів.

В ході огляду альтернативних систем дистанційного навчання було розглянуто та порівняно такі системи дистанційного навчання як ILIAS, Sakai, Docebo та Moodle, що надають схожі функції. Результати даного етапу роботи представлені у вигляді порівняльних таблиць.

2 ОГЛЯД ІНСТРУМЕНТІВ СУМІСНОСТІ З НАВЧАЛЬНИМИ СЕРЕДОВИЩАМИ

2.1 Огляд інструментів сумісності з навчальними середовищами

Багаторазове використання електронних засобів навчання вимагає адаптації їх до різних навчальних середовищ.

Стандарт – це формат, затверджений визнаним інститутом стандартизації або прийнятий підприємствами галузі в якості зразка. Наявність стандартів важливо для будь-якого користувача інформаційних технологій, так як саме завдяки стандартизації кожен користувач може комбінувати обладнання та програми різних виробників у відповідності зі своїми індивідуальними потребами. Стандартизації підлягають як обладнання, так і програмне забезпечення, зокрема, програми, що використовуються в електронному навчанні.

З розвитком інформаційно-комунікаційних технологій постала проблема багаторазового використання створеного коду для повторного використання у подальших розробках. Також це питання займає важливе місце в галузі дистанційного навчання, тобто можливість використання вже створених електронних навчальних ресурсів повторно у нових створюваних курсах, платформах [25]. Це забезпечує створення репозиторіїв повторно використовуваних навчальних ресурсів для перенесення на нові платформи кращі розробки минулих років не тільки на рівні окремих курсів, а й на корпоративному, національному і також на міжнародному рівнях.

Можливість повторного використання електронних навчальних ресурсів вимагає забезпечення їх сумісності з різними навчальними середовищами, що призвело до розробки міжнародних стандартів у галузі електронного навчання. Деякі з найбільш авторитетних міжнародних організацій включають Learning Technology Standards Committee (LTSC), IMS Global Learning Consortium, що об'єднує близько 50 учасників-контриб'юторів, Global Learning Objects Brokering Exchange, до якого

входять Ariadne Foundation та інші організації. Деякі з найвідоміших стандартів, розроблених ними можна назвати IEEE Learning Object Metadata [26], IMS Content Packaging [27], IMS Learning Design [28] та IMS Simple Sequencing [29].

В ході роботи було розглянуто набір стандартів Sharable Content Object Reference Model (SCORM), Learning Tools Interoperability (LTI), а також інструменти External tools.

2.1.1 Поняття набору стандартів SCORM

Нині найбільшого поширення набув набір специфікацій і стандартів під загальною назвою SCORM (Sharable Content Object Reference Model). SCORM – набір стандартів та специфікацій, розроблений для систем дистанційного навчання, перша версія якого була розроблена у 2001р. ініціативною групою ADL (Advanced Distributed Learning)[24][29]. Цей стандарт містить вимоги до організації навчального матеріалу та всієї системи дистанційного навчання. SCORM дозволяє забезпечити сумісність компонентів та можливість їх багаторазового використання: навчальний матеріал представлений окремими невеликими блоками, котрі можуть включатись у різні навчальні курси та використовуватись системою дистанційного навчання незалежно від того, ким, де та за допомогою яких засобів вони були створені [24], [25].

SCORM представляє навчальний матеріал окремими невеликими блоками, описаними мовою XML, які можуть включатися у різні навчальні курси і використовуватися в системах дистанційного навчання незалежно від засобів, якими вони були створені [35]. Було розроблено чимало інструментів для створення навчальних засобів навчальних об'єктів у стандарті SCORM, так званих SCORM-редакторів [30], і засобів, що дозволяють використовувати ці об'єкти – SCORM-плеєрів [31].

Недоліком використання пакетів SCORM є представлений у вигляді архівованого набору файлів, навчальний об'єкт, що експортується з одного середовища і потім імпортується до іншого. Однак, сучасні засоби дистанційного навчання надають перевагу навчальним об'єктам, представленим не як файли, а як послуги веб-сервера [35]. Такий підхід до розв'язання проблеми повторного використання навчальних ресурсів втілений, зокрема, в ідеї розподіленого навчального середовища [32], що передбачає спільне використання багатьох різномірних віртуальних середовищ, які взаємодіють з користувачем і між собою. Для забезпечення такої взаємодії IMS Global Learning Consortium був розроблений новий стандарт – Learning Tools Interoperability (LTI) [33], перша версія цього стандарту вийшла ще в 2010р. Він регламентує обмін даними між двома віртуальними середовищами.

2.1.2 Поняття стандарту LTI

Learning Tools Interoperability (LTI) – стандартний протокол, що дозволяє забезпечити сумісність засобів навчання, інтегрувати функціонал одних систем дистанційного навчання в інші та надає можливість реалізовувати і вбудовувати власні функціональні модулі в уже існуючі застосунки. Наприклад, LMS може використовувати LTI для розміщення вмісту курсу та інструментів, що надаються зовнішніми сторонніми системами на веб-сайті, не вимагаючи від учня входу окремо у зовнішні системи [33], [34].

Стандарт LTI регламентує обмін даними між двома навчальними середовищами, у процесі якого користувачі середовища А мають змогу одержувати доступ до ресурсів середовища Б, наприклад, виконувати певну навчальну діяльність. Водночас від середовища А до середовища Б передаються необхідні для виконання всієї діяльності дані користувачів

середовища А, а в зворотному напрямку від середовища Б до середовища А можуть передаватися результати виконання (рис. 2.1) [35].

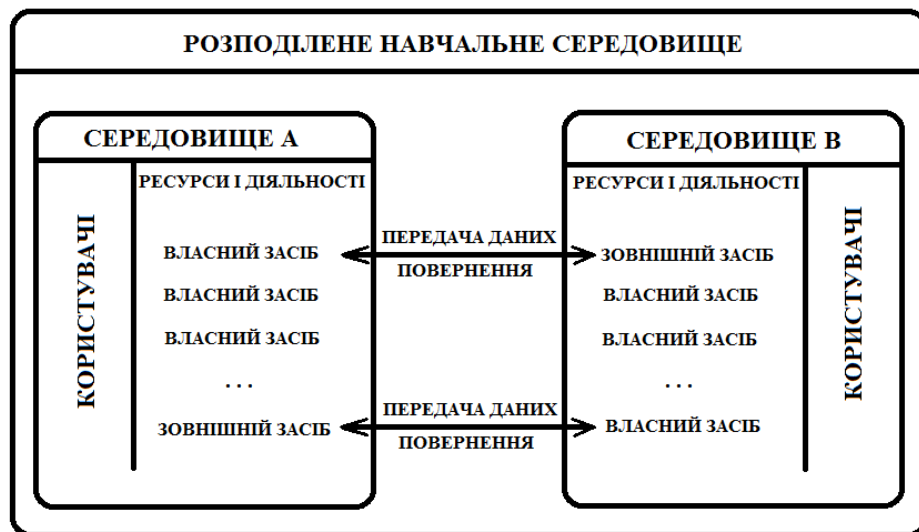


Рисунок 2.1 – Схема роботи протоколу LTI

До появи стандарту LTI задача інтеграції платформ дистанційного навчання з іншим програмним забезпеченням вирішувалося шляхом створення власних спеціальних додатків. Недоліком такого підходу було те, що подібні додатки повинні були бути розроблені для кожної платформи та кожного зовнішнього програмного засобу, з яким вони інтегрується, а потім повинні були оновлюватись з кожною наступною версією самої платформи так і зовнішнього засобу. Завдяки стандартизації немає необхідності в розробці, оновлюванні плагінів, якщо платформа і зовнішнє програмне забезпечення підтримують стандарт LTI [36]-[38].

Ключовими цілями даного стандарту є [37]:

- забезпечити просту і стислу модель взаємодії що включає в себе URL-адресу, ключ та секрет, які адміністратор системи або викладач курсу можуть ввести всистему дистанційного навчання;
- визначити протокол для запуску зовнішньої програми з системи управління вмістом таким чином, що буде підтримувати єдиний вхід що зберігає контекст навчання та ролі користувачів у цьому контексті;

– зробити портативні посилання на зовнішні програми, визначивши елементи даних, які можуть бути вбудовані в елементи внутрішньої системи.

Життєвий цикл LTI взаємодії починається коли керівник курсу, тобто викладач, чи адміністратор системи має необхідність отримати доступ до навчальних ресурсів, що знаходяться за межами LMS [39]. Адміністратор навчальних засобів повідомляє керівнику курсу LMS посилання, секрет і ключ від конкретного навчального засобу.

В першому випадку, для того, щоб додати сторонній ресурс до загальної структури курсу як посилання в інтерфейсі контрольної панелі системи дистанційного навчання, найчастіше використовується LTI засіб. Керівник додає посилання, ключ та секрет до елемента курсу, а учні в свою чергу при виборі засобу вже автоматично надають необхідні метадані.

У другому випадку додається «віртуальний засіб» до всієї системи, який потім перевикористовується учнями і керівниками. В даному випадку, при створенні або налаштуванні, зовнішній засіб визначається як елемент самої системи дистанційного навчання, і користувачі системи навіть можуть не знати, що вони користуються елементами стороннього засобу.

В обох випадках зовнішній засіб отримує запит на запуск, який включає в себе інформацію про курс, роль, особистість керівника, ключ та підпис. Інформація про запуск посилається засобами HTTP форми, що генерується в браузері користувача з прихованими елементами даних LTI протоколу. Приватні дані користувачів захищаються і підписуються OAuth стандартом захисту так, щоб інформація в період між генерацією і посиланням до зовнішнього засобу точно не була модифікованою або перехопленою. Після виконання запиту напуск користувач може бути перенаправленим в інше місце чи одразу почати роботу з засобом у випадку успіху операції [35].

2.2 Вибір системи дистанційного навчання

Платформа з якою планується інтеграція має підтримувати протокол LTI [39]. Підтримка даного стандарту передбачає те, що зовнішні елементи будуть інтегруватися до системи за допомогою технологій вбудовування і не очікують від системи якихось технічних особливостей, адже головну роль в даному випадку відіграє браузер користувача.

Також важливими критерієм при виборі системи є відкритість коду, можливість локального розгортання системи, невелике навантаження на апаратну частину.

За результатами попереднього розділу, виходячи з вимог та перебору можливих варіантів, особлива увага приділяється системі дистанційного навчання Moodle, що відповідає всім вище описаним критеріям. Розгортання системи не потребує особливих вимог від комп'ютерної системи, а також наявність документації полегшує процес роботи [40].

У системі дистанційного навчання Moodle цей стандарт підтримується починаючи з версії 2.2, опублікованої в 2011 році. Там серед основних елементів курсу з'явилась діяльність що називається External Tool.

2.2.1 Поняття інструменту External tool

External tool – інструменти для впровадження внутрішніх модулів і плагінів у LMS. Тобто це вправи або матеріали, до яких Moodle може підключитися [41].

External Tool, форма редагування, де зазначаються основні параметри: URL запуску, ключ клієнта і пароль. Багато застосунків пропонують матеріали та інтерактивні навчальні вправи, що відрізняються від власних ресурсів та діяльності Moodle та доповнюють їх. Зовнішній інструмент пропонує викладачам посилення на ці заходи на своїй сторінці курсу Moodle і там, де це

можливо, для повернення оцінок у Moodle. Студентам потрібно лише увійти в Moodle; їм не потрібно входити вдруге на приєднувальний сайт.

Застосунки, які дозволяють таким чином підключатися до Moodle, відомі як LTI-сумісні і називаються постачальниками інструментів [40].

2.3 Висновки з розділу 2

Аналіз функціональних можливостей інтеграції систем дистанційного навчання включає в себе характеристику стандартних протоколів SCORM, LTI, що є загальноприйнятими для сучасних застосунків дистанційного навчання.

Також було наведено обґрунтування чому була обрана система диставчання Moodle для інтеграції розробленого застосунку.

3 ПРОЄКТУВАННЯ ПРОГРАМНОГО ЗАСТОСУНКУ

На основі результатів виконаної роботи, наведених в попередніх розділах: огляду систем дистанційного навчання, їх ключових функціональних можливостей і загальних характеристик, плагінів і сторонніх засобів, які інтегруються в системи, було проведено аналіз щодо архітектури та концепції роботи розроблюваної системи.

3.1 Розгортання та вимоги системи дистанційного навчання Moodle

Розгортання системи дистанційного навчання Moodle потребує задіяння різних сервісів, таких як: база даних, веб сервер, інтерпретатор PHP та системні засоби операційної системи Linux, окрім власного програмного коду для готового програмного застосунку, що не є досить простою задачею. Зазвичай розгортання LMS заключається в запуску окремих модулів, завантаженні і встановленні визначених необхідних бібліотек і ресурсів, і виконується без засобів автоматизації.

База даних є однією з основних елементів системи. Moodle підтримує такі можливі варіанти СКБД для інтеграції до системи: MySQL, Oracle Database, MariaDB, PostgreSQL, Microsoft SQL Server (табл. 3.1).

Таблиця 3.1 – Вимоги до СКБД системи Moodle

Назва СКБД	Мінімальна версія	Рекомендована версія
MySQL	5.7	Остання
Oracle Database	11.2	Остання
MariaDB	10.2.29	Остання
PostgreSQL	9.6	Остання
Microsoft SQL Server	2012	Остання

Виходячи з цього, була обрана СКБД PostgreSQL, що запропонована стандартною документацією [42], [43].

Необхідно встановити інтерпретатор PHP мінімальної версії 7.2 або пізнішої, а також потребуються деякі бібліотеки і розширення для PHP окрім стандартних. Статичні файли та обробка динамічних файлів обслуговується вільним веб-сервером Apache.

Для економії часу розгортання системи Moodle досить зручним варіантом є обернення у віртуальні контейнери необхідних сервісів і подальший запуск їх у віртуальному середовищі Docker.

Docker – програмне забезпечення для автоматизації розгортання і управління додатками в середовищах з підтримкою контейнеризації, що дозволяє «упакувати» програмний додаток з усім його оточенням і залежностями в контейнер, який можна перенести на будь-яку Linux-систему [44], [45].

До появи Docker при розгортанні систем виникало безліч неприємних і складних питань, пов'язаних з конфліктом версій і незрозумілими залежностями. Docker дозволяє упакувати образ стану системи, розгорнути його у виробничому середовищі і там запустити. До переваг Docker можна віднести [45], [46]:

- стандартизований механізм створення середовища;
- прискорений процес розробки;
- немає необхідності встановлювати допоміжні інструменти, як зразок, PostgreSQL: їх можна запускати в контейнерах;
- зручна інкапсуляція додатків;
- захист від втрати інформації, надійність;
- зменшення часу відповіді;
- зрозумілий моніторинг і просте масштабування.

Таким чином завдяки описанню необхідних ресурсів в одному конфігураційному файлі (рис. 3.1) можна уникнути проблем з сумісністю

системних сервісів, а також встановити та запустити всю інфраструктуру за допомогою декількох консольних команд.

```
version: "2"
services:
  webservice:
    image: moodlehq/moodle-php-apache:${MOODLE_DOCKER_PHP_VERSION}
    depends_on:
      - db
    volumes:
      - "${MOODLE_DOCKER_WWWROOT}:/var/www/html"
      - "${ASSETDIR}/web/apache2_faildumps.conf:/etc/apache2/conf-enabled/apache2_faildumps.conf"
    environment:
      MOODLE_DOCKER_DBTYPE: pgsq
      MOODLE_DOCKER_DBNAME: moodle
      MOODLE_DOCKER_DBUSER: moodle
      MOODLE_DOCKER_DBPASS: "m@d131ng"
      MOODLE_DOCKER_BROWSER: firefox
      MOODLE_DOCKER_WEB_HOST: "${MOODLE_DOCKER_WEB_HOST}"
  db:
    image: postgres:11
    environment:
      POSTGRES_USER: moodle
      POSTGRES_PASSWORD: "m@d131ng"
      POSTGRES_DB: moodle
  exttests:
    image: moodlehq/moodle-exttests
  selenium:
    image: "selenium/standalone-firefox:${MOODLE_DOCKER_SELENIUM_SUFFIX}:2.53.1"
    volumes:
      - "${MOODLE_DOCKER_WWWROOT}:/var/www/html:ro"
```

Рисунок 3.1 – Конфігураційний файл інфраструктури Moodle

В результаті, за допомогою принципів роботи віртуального середовища Docker при запуску інфраструктури, вдається ізолювати системний простір від роботи Moodle, і в разі необхідності провести моніторинг кожного контейнеру та проаналізувати наявність проблем, або замінити його не впливаючі на інші елементи [46].

Таким чином, в процесі розробки програмного застосунку замість повного встановлення СКБД та її конфігурації, достатньо змінити деякі конкретні конфігураційні параметри і почати процес оновлення середовища контейнерів, що в свою чергу значно полегшує вибір бази даних та її розгортання.

3.2 Вибір мови програмування для реалізації програмного застосунку

Оскільки було вирішено, що розроблений застосунок буде працювати за LTI протоколом – не матиме обмежень, окрім підтримки стандартів і надання відкритого URL для обробки серверних запитів. Так як застосунок працює на серверній стороні і оброблює запити користувачів мережі, для відображення інформації та даних в браузері потрібно створити представлення для цих даних.

Було обрано мову програмування Javascript [47] в результаті порівняння альтернативних варіантів для того, щоб прискорити розробку застосунку через те, що вона може використовуватись на клієнтській і на серверній частинах застосунку. В порівняльній таблиці 3.2 наведені можливі варіанти вибору мови програмування.

Таблиця 3.2 – Порівняльна таблиця варіантів мов програмування

Мова	Парадигма	Клієнтська	Серверна
Javascript	Імперативна, функційна, прототипно-орієнтована	+	+
Python	Імперативна, об'єктно-орієнтована, функціональна	-	+
PHP	Імперативна, об'єктно-орієнтована	-	+
Java	Об'єктно-орієнтована, імперативна	-	+
Ruby	Об'єктно-орієнтована	-	+
Go	Функційна, імперативна, об'єктно-орієнтована	-	+
Scala	Функційна, імперативна	-	+
Rust	Імперативна, структурна, функційна	-	+

3.3 Вибір технологій для реалізації програмного застосунку

3.3.1 NodeJS

Так як для розробки застосунку була обрана мова програмування Javascript, був обраний конкретний набір технологій. Серверний Javascript має власну назву – NodeJS та цілу екосистему [48], [49].

Node.js – платформа з відкритим кодом для виконання високопродуктивних мережевих застосунків, написаних мовою Javascript [48].

Хоча раніше Javascript використовувався для обробки даних у браузері користувача, NodeJS пропонував можливість виконувати сценарії JavaScript на сервері та надсилати користувачеві результат їх виконання. Платформа Node.js перетворила JavaScript на мову загального використання з великою спільнотою розробників.

NodeJS має наступні властивості:

- асинхронна одно-нитева модель виконання запитів;
- неблокуючий ввід/вивід;
- система модулів CommonJS;
- рушій JavaScript Google V8.

Для керування модулями використовується пакетний менеджер npm – node package manager [50].

3.3.2 Typescript

Typescript – це мова програмування з відкритим вихідним кодом що ґрунтується на Javascript і є одним з найпопулярніших розширень [51]. Головною її відмінністю є додавання статичного аналізатору типів. Типи надають можливість описувати об'єкти для того щоб код був більш документованим і інтерпретатор міг валідувати програмний код ще до безпосереднього виконання. Написання типів є опціональним тому що вбудована система типового аналізу в більшості випадків коректно розпізнає

код і може доповнювати його та робити висновки в ході виконання програмного коду.

Перевагу пуло надано саме мові програмування Typescript через те, що вона набуває з часом все більшої популярності, розвивається і має широкую підтримку від спільноти розробників.

3.3.3 ExpressJS

Express.js, або просто Express – серверна платформа веб-розробки для NodeJS, реалізована як безкоштовне програмне забезпечення з відкритим кодом за ліцензією MIT. Він призначений для розробки веб-додатків та API. ліцензією MIT. Він спроектований для створення веб-застосунків і API [52].

По суті являє собою стандартний каркас для Node.js [52].

3.4 Висновки з розділу 3

В даному розділі було описано нюанси розгортання системи дистанційного навчання Moodle, а також було описано вибір та огляд особливостей набору технологій для розроблюваного програмного застосунку.

4 ОСНОВНІ РІШЕННЯ ЩОДО РЕАЛІЗАЦІЇ КОМПОНЕНТІВ СИСТЕМИ

4.1 Алгоритм роботи застосунку

Цільовою аудиторією є учні, студенти, працівники компаній, які мають різний рівень знань, які можуть об'єктивно оцінити корисність курсу і надати повний відгук. Розроблений модуль, заснований на Moodle, націлений на економію часу і надання відгуків, для поліпшення навчальних курсів.

Алгоритм роботи модулю є набагато простішим ніж процес роботи інтеграції за LTI протоколом. Даний модуль розпочинає роботу одразу після завантаження сторінки вебсайту системи Moodle і функціонує в залежності від типу користувача та етапу його роботи з модулем. Схематичне зображення алгоритму є достатньо простим та ясным, зображено на рисунку 4.1.

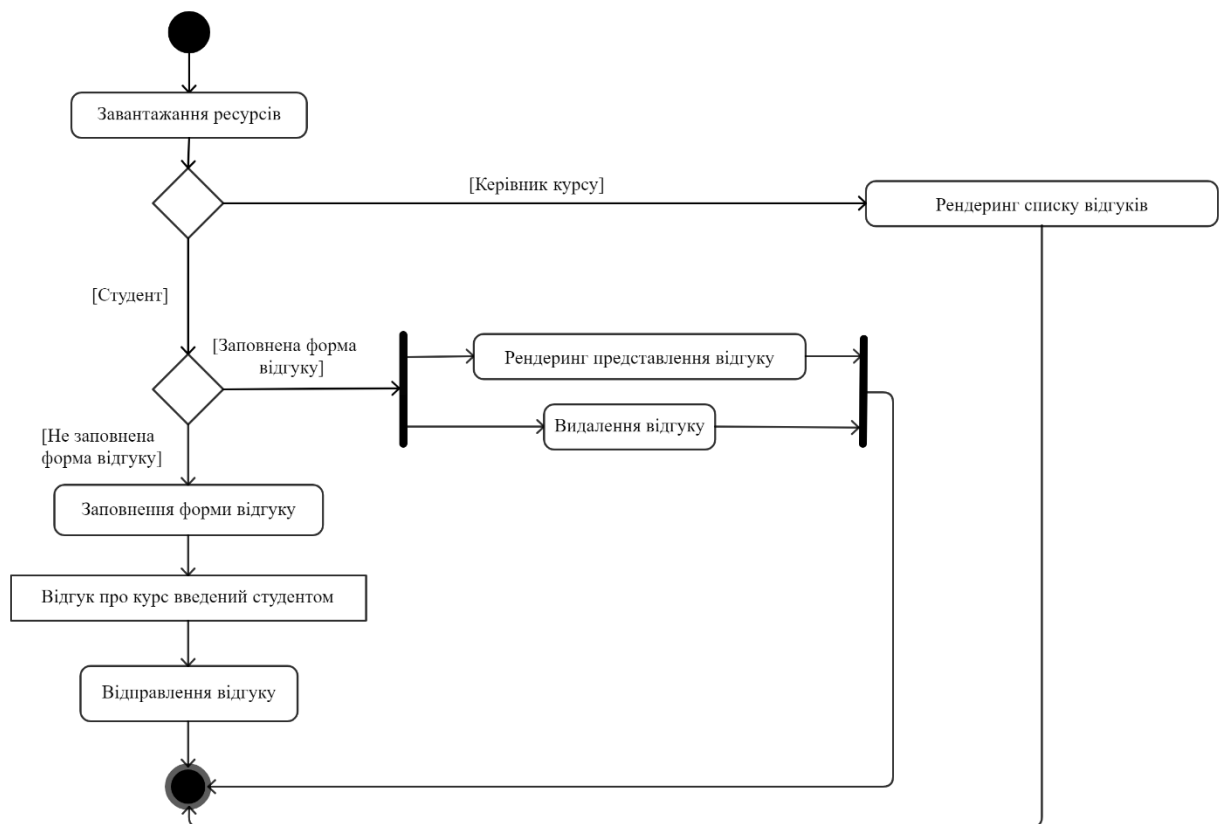


Рисунок 4.1 – Схема функціонування застосунку

Точкою входу до застосунку є відкриття користувачем посилання у курсі на модуль зворотнього зв'язку.

У випадку керівника курсу буде надано список анонімних відгуків студентів про курс.

Для студентів даний застосунок має вигляд звичайної форми відгуку. Після того як студент заповнить форму зворотнього зв'язку і підтвердить відправлення, форма повідомить про те, що студент вже відправив відгук. Робота завершується після закриття сторінки вебсайту.

4.2 Головні модулі розробленого застосунку

Згідно зі спроектованою архітектурою системи розділення логіки даних і інтерфейсу та виходячи з функціональних потреб даної системи і можливостей, розподіл модулів проєкту повністю співпадає з візуальним розділенням частин інтерфейсу.

Точка входу, яка відповідає за дії глобального рівня і є спільною для всіх модулів програми. Таким чином на цей клас покращено перший рендер застосунку, завантаження необхідних частин тексту програми, завантаження ресурсів та ініціалізація даних (рис. 4.2).

```

import { default as bodyParser } from "body-parser";
import { default as express } from "express";
import { default as exphbs } from "express-handlebars";
import { default as session } from "express-session";

import { applicationConfig } from "@/configs/ApplicationConfig";
import { router as feedbackRouter } from "@/routes/feedback";
import { router as ltiRouter } from "@/routes/lti";
import { router as participationRouter } from "@/routes/participation";
import { router as viewsRouter } from "@/routes/views";

const application = express();

/**
 * Configure express application.
 */
application.set("views", applicationConfig.VIEWS_DIR);
application.set("view engine", "handlebars");
application.engine("handlebars", exphbs({ layoutsDir: applicationConfig.LAYOUTS_DIR }));

application.use(session({ secret: process.env.SESSION_SECRET || "dev", resave: false, saveUninitialized: true }));
application.use(bodyParser.urlencoded({ options: { extended: false } }));

application.enable("trust proxy");

/**
 * Configure application routes matching.
 */
application.use("/", ltiRouter);
application.use("/", viewsRouter);
application.use("/", participationRouter);
application.use("/", feedbackRouter);

/**
 * Start listening after application configuration.
 */
application.listen(
  applicationConfig.PORT,
  () => console.log(`Example app listening on port ${applicationConfig.PORT}!`)
);

```

Рисунок 4.2 – Точка входу програми, main.ts

В даному проєкті використовуються характерні сервіси для подальшої обробки даних і відповідних дій над ними, а саме обробка запитів в залежності від користувача: керівника курсу чи студента, для яких модуль має різний вигляд і функціонал (рис. 4.3), а також обробка і видалення відгуків (рис. 4.4).

Для використання програми користувач повинен бути зареєстрований в конкретному екземплярі системи Moodle та доданий до навчального курсу, що використовує даний модуль. Під час роботи з програмою користувач може отримувати впливаючі повідомлення різного типу. Наприклад, якщо користувач не авторизований, повідомлення вкаже йому на неможливість продовження роботи (рис. 4.3).

```
function renderFeedbackApplication(req: Request, res: Response, next: NextFunction) {
  const session: { [index: string]: any } = req.session;

  if (session.userId) {
    const feedback = feedbacks.get(session.userId);

    const renderParams: { [index: string]: any } = {
      email: session.email,
      username: session.username,
      ltiConsumer: session.ltiConsumer,
      userId: session.userId,
      isTutor: session.isTutor,
      contextId: session.contextId,
      hasFeedback: Boolean(feedback),
      feedbacks: Array.from(feedbacks.values()),
      feedback: feedback
    };

    if (renderParams.isTutor) {
      return res.render( view: "tutor-feedback-report", renderParams);
    } else {
      return res.render( view: "student-feedback", renderParams);
    }
  } else {
    next(new Error("Session invalid. Please login via LTI to use this application."));
  }
}
```

Рисунок 4.3 – Обробка запитів в залежності від користувача, views.ts

```
import { default as express } from "express";
import { feedbacks } from "@store/feedback";

export const router = express.Router();

router.post( path: "/feedback", handlers: (req: Request<P, ResBody, ReqBody, ReqQuery>, res: Response<ResBody>, next: NextFunction) => {
  const session: { [index: string]: any } = req.session;
  const userId: string | undefined = session.userId;
  const feedback: string = req?.body["feedback"];

  if (!req.body) {
    return next(new Error("Expected a body")); // todo: Status for proper 403.
  }

  if (userId) {
    feedbacks.set(userId, { userId, date: new Date(), text: feedback }); // todo: Validation.

    // Redirect to actual handling of LTI after POST handling.
    res.redirect( url: "/application?type=feedback");
  } else {
    throw new Error("Cannot participate, user is not logged in");
  }
});

router.post( path: "/remove_feedback", handlers: (req: Request<P, ResBody, ReqBody, ReqQuery>, res: Response<ResBody>, next: NextFunction) => {
  const session: { [index: string]: any } = req.session;
  const userId: string | undefined = session.userId;

  if (userId) {
    feedbacks.delete(userId); // todo: Validation.

    // Redirect to actual handling of LTI after POST handling.
    res.redirect( url: "/application?type=feedback");
  } else {
    throw new Error("Cannot participate, user is not logged in");
  }
});
```

Рисунок 4.4 – Обробка і видалення відгуків, feedback.ts

Як і було описано раніше, для студента застосунок має вигляд звичайної форми відгуку і повідомить про те, що студент може залишити відгук анонімно. Після того як студент заповнить форму зворотнього зв'язку і підтвердить відправлення, форма повідомить про те, що студент вже відправив відгук, який побаче лише керівник курсу (рис. 4.5).

```
<main>
  {{#if hasFeedback }}
    <div class="alert alert-success">
      Thank you for your feedback! It is shared anonymously and visible only for tutor.
    </div>

    <h5> Your feedback: </h5>

    <div id="feedback-report" class="card">
      {{ feedback.text }}
    </div>

    <form action="/remove_feedback" method="POST">
      <button id="feedback-button" class="btn btn-danger" type="submit"> Dismiss </button>
    </form>

  {{else}}
    <form id="feedback-form" method="POST" action="/feedback">
      <div class="alert alert-secondary">
        You can leave feedback about course anonymously
      </div>

      <textarea class="form-control" name="feedback"></textarea>

      <button id="feedback-button" class="btn btn-primary" type="submit"> Submit </button>
    </form>

  {{/if}}
</main>
```

Рисунок 4.5 – Форма заповнення відгуку, student-feedback.handlebars

Також, як і було зазначено раніше, у випадку керівника курсу буде надано список анонімних відгуків студентів про курс (рис. 4.6).

```

<main>
  {{#if feedbacks.length }}
    <div class="alert alert-secondary">
      Anonymous reports for this course ({{feedbacks.length}}):
    </div>

    <ul class="list-group">
      {{#each feedbacks}}
        <li class="list-group-item">
          {{this.text}}
        </li>
      {{/each}}
    </ul>
  {{else}}
    <div class="alert alert-secondary">
      Currently there is no feedback provided for this course.
    </div>
  {{/if}}
</main>

```

Рисунок 4.6 – Список відгуків, tutor-feedback-report.handlebars

4.3 Висновки з розділу 4

В даному розділі було представлено алгоритм роботи застосунку, основою якого є вимоги, що були закладені своєрідною архітектурою, а також концепції роботи спроектованого веб-застосунку, охоплюючи загальний опис основних модулів сервісу.

Архітектурно розроблений застосунок є вебсайтом як і будь-який інший елемент що реалізує постачальника LTI сервісів для сторонніх систем. Головною метою такого архітектурного рішення застосунку є спрощення коду та чітке розподілення на сфери відповідальності, таким чином кожна частина коду має безпосередню належність тільки до певного модулю, а саме, відповідає тільки те, що зазначено. Виходячи з цього, запобігаючи недоліків, зменшується можливість потенційних ускладнень читання, розуміння та подальшої підтримки програмного коду.

Готовий програмний продукт підтверджує те що інтеграція зовнішніх засобів до систем дистанційного навчання є досить простою і відкриває двері

для широких можливостей розширення навчального процесу з перспективою подальшого перевикористання даних та сервісів.

5 ЕКСПЛУАТАЦІЯ ПРОГРАМНОГО ЗАСТОСУНКУ

5.1 Призначення програми

Цільовою аудиторією є учні, студенти, працівники компаній, які мають різний рівень знань, які можуть об'єктивно оцінити корисність курсу і надати повний відгук. Розроблений модуль, заснований на Moodle, націлений на економію часу і підтримку надання потрібних матеріалів, тобто відгуків, для поліпшення навчальних курсів. Створений програмний продукт має функцію анонімного відгуку про курс, що в свою чергу зменшить час визначення якості навчальних курсів, а також підвищить зацікавленість в покращенні навчання приблизно в 2-3 рази.

5.2 Умови виконання програми

Застосунок буде виконуватися за умови правильного функціонування комп'ютера та операційної системи. Повинен бути встановлений сучасний веб-браузер актуальної версії. Також комп'ютер повинен мати доступ до інтернет мережі.

5.3 Додавання застосунку до системи

Програма встановлюється у вигляді модуля активності у вже розгорнутий дистрибутив системи Moodle. Для того, щоб цей модуль був доступний користувачеві, керівник системи повинен додати екземпляр модуля у вигляді активності до будь-якого дійсного навчального курсу у системі.

Модуль при цьому буде доступний при переході на сторінку навчального курсу.

Додавання застосунку до системи дистанційного навчання Moodle як зовнішнього засобу наведено на рисунку 5.1.

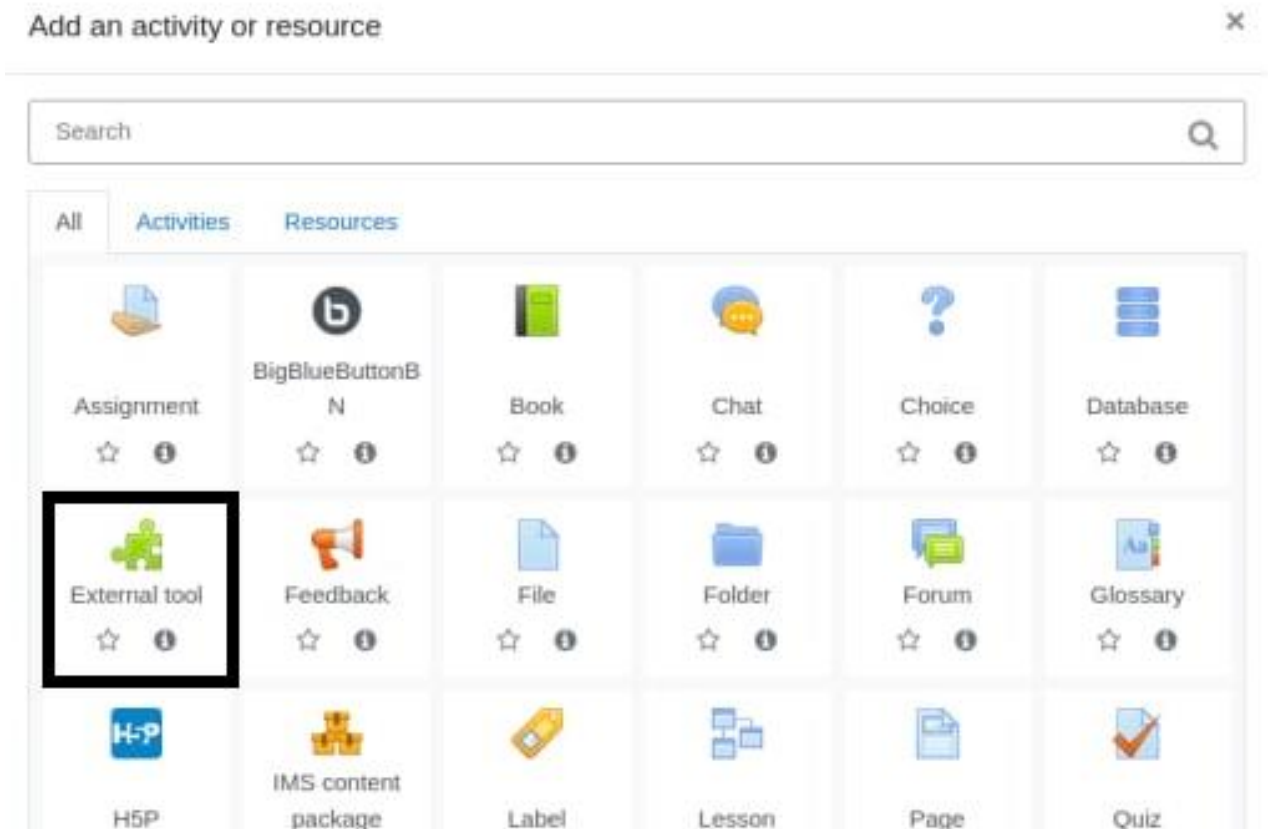


Рисунок 5.1 – Додавання зовнішнього засобу до системи Moodle

Подальший процес потребує від адміністратора курсу додаткових даних що є стандартними в процесі інтеграції зовнішніх засобів за протоколом LTI. В даному випадку в URL адресою слугує «localhost:3000/extension?type=feedback», оскільки локально застосунок запущено у вигляді веб сервера, що прослуховує порт 3000, і надає окремий шлях для запуску розширення і створення сесії користувача.

Окремо додаються поля ключа та секрету системи до якої додається розширення. Зазвичай ці дані генеруються в повністю реалізованих сервісах після процесу реєстрації, активації та підтвердження облікового засобу користувача, а в окремих випадках дані послуги можуть монетизуватися за допомогою продажів підписки чи то ключів. Оскільки застосунок є більше дослідницькою та демонстраційною програмою, коректні дані було вбудовано в програму без ускладнень процесу і додано як зовнішній засіб до системи (рис. 5.2).

The screenshot shows the 'Preconfigured tool' configuration form in Moodle. It includes fields for 'Tool URL' (http://localhost:3000/extension?type=feedback), 'Secure tool URL', 'Launch container' (Embed), 'Consumer key' (secretkey), and 'Shared secret' (zjkWxYS). A 'Select content' button is also visible. A green checkmark indicates 'Using custom tool configuration.' and a note at the bottom says 'Press enter to save changes'.

Рисунок 5.2 – Заповнення інформації про зовнішній засіб

Після додавання застосунку і надання коректних даних, зовнішній засіб можна побачити у списку конкретної теми курсу до якої його було додано (рис 5.3).

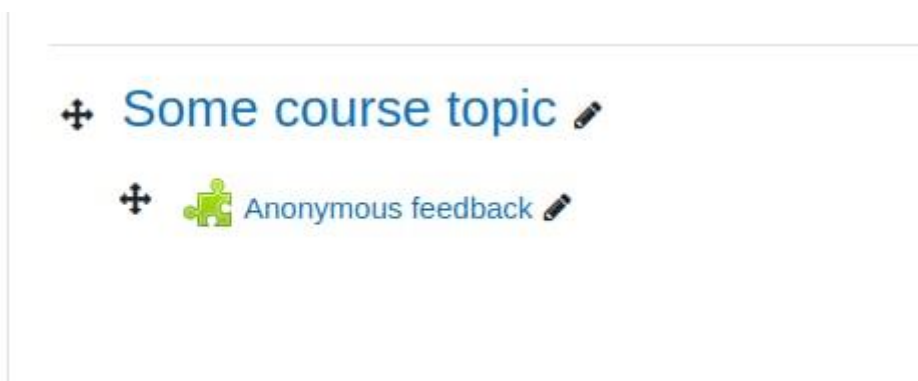


Рисунок 5.3 – Доданий зовнішній засіб до курсу

5.4 Використання застосунку

Для використання програми користувач повинен бути зареєстрований в конкретному екземплярі системи Moodle та доданий до навчального курсу, що використовує даний модуль. Під час роботи з програмою користувач може отримувати впливаючі повідомлення різного типу. Наприклад, якщо користувач не авторизований, повідомлення вкаже йому на неможливість продовження роботи (рис. 5.4).

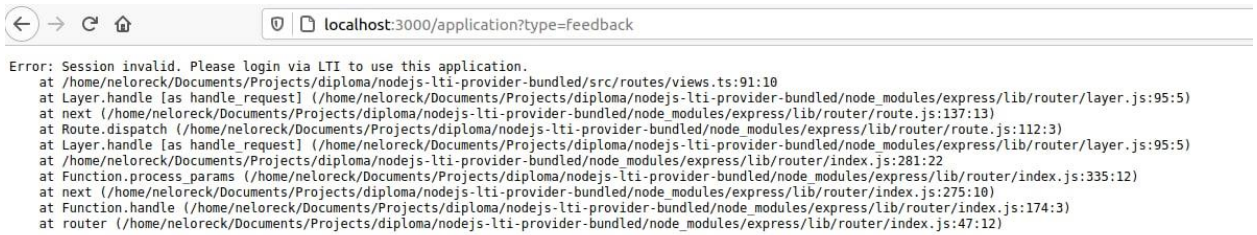


Рисунок 5.4 – Повідомлення про помилку

5.5 Функціональні можливості розробленого застосунку

Розроблений застосунок інтегрується до системи дистанційного навчання Moodle і являє собою просту форму анонімного зворотнього зв'язку студентами про курс керівнику.

Для студентів даний застосунок має вигляд звичайної форми відгуку (рис. 5.5).

> My courses > ExampleCourse > Some course topic > Anonymous feedback

Anonymous feedback

You can leave feedback about course anonymously

Submit

Рисунок 5.5 – Не заповнена форма відгуку

Після того як студент заповнить форму зворотнього зв'язку і підтвердить відправлення, форма повідомить про те, що студент вже відправив відгук (рис. 5.6).

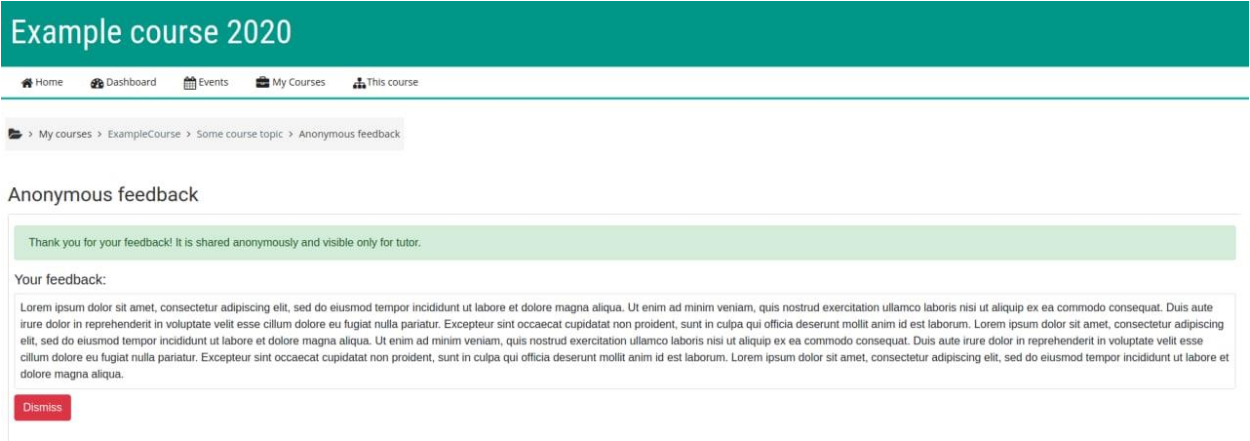


Рисунок 5.6 – Заповнена форма для учня курсу

Для керівників курсів в системі даний застосунок виглядає як список анонімних відгуків студентів про курс (рис. 5.7, рис. 5.8).

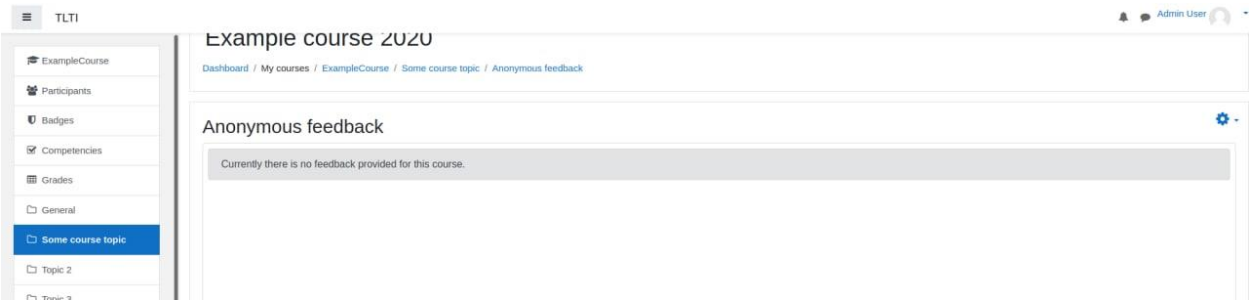


Рисунок 5.7 – Інтерфейс розширення без відгуків

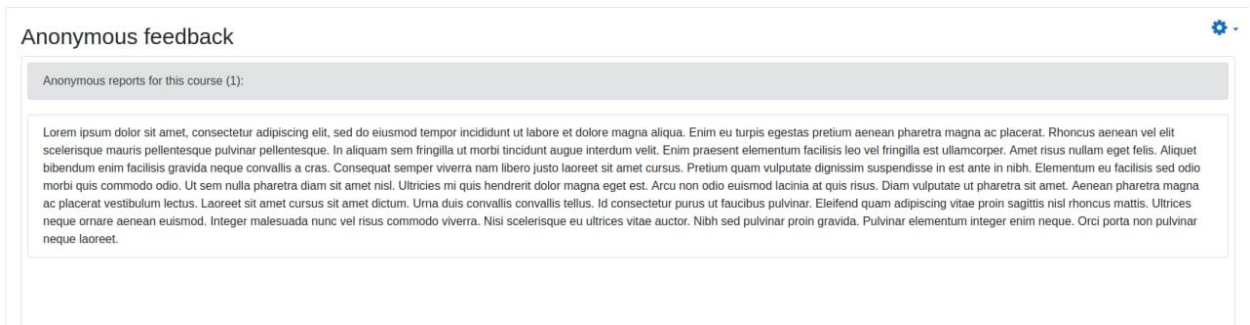


Рисунок 5.8 – Інтерфейс розширення з відгуком

При цьому для студентів даний застосунок не виявляє відгуків інших студентів, і на сторінці студент може побачити лише свій відгук.

5.6 Висновки з розділу 5

В даному розділі було описано призначення застосунку. А також додавання застосунку до системи дистанційного навчання Moodle як зовнішнього засобу, його використання і функціональні можливості.

6 ЕКОНОМІКО-ОРГАНІЗАЦІЙНА ЧАСТИНА

Метою дипломної роботи магістра є розробка зовнішнього модуля анонімного зворотнього зв'язку студентами про курс та дослідження засобів інтеграції розробленого модуля до системи дистанційного навчання Moodle.

В роботі розглянуто можливості розширення функціоналу на прикладі таких аналогічних систем як: ILIAS, Sakai, Docebo, Moodle. Всі вони є відкритими платформами і опубліковані під відкритими ліцензіями. Також усі ці системи надають можливість розширення власного функціоналу за допомогою модулів.

Для більш детального аналізу було проведено дослідження типів плагінів. Можна виділити два принципово різні підходи до розширення систем дистанційного навчання:

- розробка внутрішніх модулів системи та інтегрування у систему безпосередньо;
- розробка зовнішніх незалежних застосувань, які інтегруються з системою управління навчанням.

Системи дистанційного навчання, що підтримують розширення функціональних можливостей, використовують перший спосіб – створення і впровадження внутрішніх модулів. В ході розробки таких модулів використовується та ж сама мова програмування, за якої розроблена й сама система, та опирається на наданий набір класів розробниками системи.

На основі проведеного огляду аналогічних застосунків можна зробити висновок, що один з основних критеріїв керування якістю навчання є впровадження системи опитувань або анкетувань студентів.

Цільовою аудиторією є учні, студенти, працівники компаній, які мають різний рівень знань, які можуть об'єктивно оцінити корисність курсу і надати повний відгук. Розроблений модуль, заснований на Moodle, націлений на економію часу і підтримку надання потрібних матеріалів, тобто відгуків, для поліпшення навчальних курсів. Створений програмний продукт має функцію

анонімного відгуку про курс, що в свою чергу зменшить час визначення якості навчальних курсів, а також підвищить зацікавленість в покращенні навчання приблизно в 2-3 рази.

6.1 Планування розробки програмного виробу

Процес розробки програмного продукту можна розділити на декілька етапів. Для кожного етапу визначаються трудомісткість, кількість виконавців та тривалість робіт. В розробці даного модулю приймають участь програміст та аналітик. Розробка модулю починається 5 жовтня і повинно завершитися 27 листопада 2020 року.

Необхідні початкові дані [53]:

- час на розробку – до 2 місяців;
- робота програміста – до 2 місяців;
- робота аналітика – до 0,25 місяців.

Тривалість робіт розраховується за формулою (6.1).

$$T_{\text{ц}} = \frac{Q}{R}, \quad (6.1)$$

де $T_{\text{ц}}$ – довжина циклу, днів;

Q – трудомісткість, людино-днів;

R – кількість виконавців, люд.

Основні роботи, необхідні для розробки програмного продукту і їх трудомісткість представлені в таблиці 6.1.

Таблиця 6.1 – Характеристика робіт по розробці програмного продукту

Найменування роботи	Трудомісткість		Виконавці	Тривалість
	люд.- дн.	% до підсумку	спеціальність	днів
1. Дослідження вимог до проектованого продукту	10	22,22	Програміст, аналітик	5 5
2. Розробка ТЗ	5	11,11	Програміст	5
3. Проектування структури програмного продукту	5	11,11	Програміст	5
4. Розробка схеми функціонування програми	5	11,11	Програміст	5
5. Створення програмного коду	14	31,11	Програміст	14
6. Тестування та налагодження	3	6,66	Програміст	3
7. Складання програмної документації	3	6,66	Програміст	3
Всього	45	100,00		40

Виходячи отриманих даних, приведених в таблиці 6.1, складається лінійний графік Ганта – зведений стрічковий графік планування розробки модулю, який представляє собою таблицю. В першому стовпці розміщені в порядку збільшення термінів початку виконання всі види роботи, а в другому – календарний період їх виконання. Результат наведений в таблиці 6.2.

Таблиця 6.2 – Лінійний графік Ганта планування процесу розробки

Найменування роботи	Календарний період, дні							
	05.10 – 09.10	12.10 – 16.10	19.10 – 23.10	26.10 – 30.10	02.11 – 06.11	09.11 – 13.11	16.10 – 20.11	23.10 – 27.11
1. Дослідження вимог до проектованого продукту	 							
2. Розробка ТЗ								
3. Проектування структури програмного продукту								
4. Розробка схеми функціонування програми								
5. Створення програмного коду								
6. Тестування та налагодження програмного продукту								
7. Складання програмної документації								

Позначення:  – програміст;  – аналітик.

6.2 Визначення витрат на розробку програми

Для визначення витрат на дослідження вимог та на розробку програми була складена калькуляція кошторисної вартості робіт, що включає в себе такі пункти:

- основна заробітна плата;
- додаткова заробітна плата;
- відрахування на єдиний соціальний внесок (ЄСВ);
- витрати на спеціальне обладнання;

- матеріали та комплектуючі вироби;
- накладні витрати;
- податки.

6.2.1 Розрахунок основної і додаткової заробітної плати

Витрати в цій статті складаються з планового фонду заробітної плати усіх категорій працівників, що найняті для розробки даного програмного продукту.

Розрахунок заробітної плати ведеться на основі даних про трудомісткість, представлених у таблиці 6.1.

Необхідні початкові дані:

- час на розробку – до 2 міс.;
- робота програміста – до 2 міс. (40 днів) із заробітною платою 14000 грн/міс;
- робота аналітика – 0,25 міс. (5 днів) із заробітною платою 7000 грн/міс.

Додаткову заробітну плату приймають рівною 10% від основної заробітної плати працівників і розраховують за формулою (6.2):

$$ЗП_{\text{дод}} = ЗП_{\text{осн}} \cdot 0,1, \quad (6.2)$$

де $ЗП_{\text{дод}}$ – додаткова заробітна плата, грн;

$ЗП_{\text{осн}}$ – основна заробітна плата, грн.

Отримані результати розрахунків основної заробітної плати представлено в таблиці 6.3.

Таблиця 6.3 – Розрахунок основної заробітної плати

Посада виконавця	Кількість, осіб	Кількість, міс. роботи	Місячний оклад, грн	Сума основної ЗП, грн.	Сума додаткової ЗП, грн.	Сумарна ЗП, грн.
Програміст	1	2	14000	28000	2800	30800
Аналітик	1	0,25	7000	1750	175	1925
Всього	2			29750	2975	32725

6.2.2 Відрахування на єдиний соціальний внесок

На даному етапі розраховуються відрахування на соціальні заходи – єдиний соціальний внесок, що включають у себе відрахування до Пенсійного фонду, фонду страхування на випадок тимчасової непрацездатності, фонду страхування по безробіттю та фонду страхування від нещасних випадків на виробництві. Ці відрахування визначають в процентному відношенні від суми основної і додаткової зарплат. Коефіцієнт відрахувань на соціальні заходи (єдиний соціальний внесок), згідно діючого законодавства становить 22%.

$$\text{ВІД} = (\text{ЗП}_{\text{осн}} + \text{ЗП}_{\text{дод}}) \cdot 0,22, \quad (6.3)$$

де $\text{ЗП}_{\text{осн}}$ – основна заробітна плата, грн;

$\text{ЗП}_{\text{дод}}$ – додаткова заробітна плата, грн.

$$\text{ВІД} = (29750 + 2975) \cdot 0,22 = 7199,5 \text{ грн.}$$

Виходячи з формули (6.3) отримуємо 7199,5 грн відрахування на ЄСВ.

6.2.3 Визначення витрат на матеріали

У цю статтю входять вартість основних і допоміжних матеріалів, комплектуючих виробів, що необхідні в ході розробки програмного продукту.

Матеріали, які знадобилися для розробки представлені в таблиці 6.4.

Таблиця 6.4 – Витратні матеріали

Матеріал	Кількість	Вартість, грн.
Папір формату А4	500 аркушів	85
CD - RW диск	1 шт.	5
Тонер для принтера	1 шт.	120

Витрати на матеріали розраховуються за формулою (6.4):

$$M = \sum_{i=1}^n (\text{Ц}_i \cdot N_i (1 + K_{\text{т.з.}}) - \text{Ц}_{io} \cdot N_{io}), \quad (6.4)$$

де M – витрати на матеріали, покупні напівфабрикати і комплектуючі вироби, грн.;

$K_{\text{т.з.}}$ – коефіцієнт, що враховує транспортно-заготівельні витрати (0,03 - 0,1);

Ц_i – ціна i -го найменування матеріалу, напівфабрикату, комплектуючого, грн.;

N_i – ціна i -го матеріалу, напівфабрикату, комплектуючого;

Ц_{io} – ціна поворотних відходів i -го найменування матеріалу, грн.;

N_{io} – кількість зворотних відходів i -го найменування;

n – кількість найменувань матеріалів, напівфабрикатів, комплектуючих.

Необхідні вхідні дані: $\text{Ц}_{io} = 0$; $N_{io} = 0$; $K_{\text{т.з.}} = 0,05$.

Розрахуємо витрати на матеріали:

$$M = (1 + 0,05) \cdot (85 + 5 + 120) = 220,5 \text{ грн.}$$

Разом витрати на матеріали склали 220,5 грн.

6.2.4 Витрати на спеціальне обладнання

В цю статтю входять витрати на придбання, транспортування, монтаж і налагодження обладнання, а також враховуються витрати на оплату машинного часу електронно-обчислювальної машини (ЕОМ).

Амортизаційні відрахування визначають по формулі (6.5):

$$A = \frac{\Phi_6 \cdot H_a}{100}, \quad (6.5)$$

де Φ_6 – балансова вартість обчислювальної техніки, грн.;

H_a – норма амортизаційних відрахувань на повне відновлення обчислювальної техніки.

Балансова вартість обчислювальної техніки становить 28000 грн., а норма амортизаційних відрахувань – 25%. Таким чином амортизаційні відрахування становлять:

$$A = 28000 \cdot 0,25 = 7000 \text{ грн.}$$

Одержуємо 7000 грн.

Балансова вартість машини визначається як вартість машини плюс 5% на транспортні витрати і монтаж.

Витрати на електроенергію (6.6):

$$C_e = N_n \cdot \Phi_{ef} \cdot K_{зв} \cdot K_{зм} \cdot C_{ел.е}, \quad (6.6)$$

де N_n – номінальна потужність ЕОМ, кВт;

Φ_{ef} – річний ефективний фонд часу роботи ЕОМ, ч.;

$K_{зв}$ – середній коефіцієнт завантаження за часом;

$K_{зм}$ – коефіцієнт завантаження по потужності;

$Ц_{ел.е}$ – ціна одного кВт·ч електроенергії, грн.

Номінальна потужність ЕВМ – 0,2 кВт. Річний ефективний фонд часу роботи ЕОМ становить 1800 годин. Середні коефіцієнти завантаження за часом і за потужністю рівні відповідно 0,9 і 0,6. Ціна однієї кіловат-години електроенергії становить 1,68 грн. Таким чином одержуємо:

$$C_e = 0,2 \cdot 1800 \cdot 0,9 \cdot 0,6 \cdot 1,68 = 326,592 \text{ грн.}$$

В результаті витрати на електроенергію становлять 326,592 грн.

Зарплата обслуговуючого персоналу розраховується по формулі (6.7):

$$ЗП_{обсл} = ФЗП_{г} \cdot (1 + K_{отч}) \cdot \frac{t_{обсл}}{\Phi_{эф.обсл}}, \quad (6.7)$$

де $ФЗП_{г}$ – річний фонд заробітної плати (основної та додаткової) обслуговуючих робітників, грн.;

$K_{відрах}$ – коефіцієнт, що враховує відрахування на соціальне страхування і в інші фонди;

$t_{обсл}$ – час протягом року, необхідний на технічне обслуговування ЕОМ, год./рік;

$\Phi_{эф.обсл}$ – річний ефективний фонд часу обслуговуючого персоналу, год./рік.

Заробітна плата обслуговуючого персоналу становить 2800 грн./міс.

Річний ефективний фонд робочого часу обслуговуючого ПК працівника дорівнює 1750 г/рік.

На обслуговування одного ПК по 0,5 години на місяць становить на рік 6 годин.

Отримаємо:

$$\Phi ЗП_r = 2800 \cdot 12 = 33600 \text{ грн.},$$

$$ЗП_{\text{обсл}} = 33600 \cdot (1 + 0,22) \cdot \frac{6}{1750} = 140,544 \text{ грн.}$$

Таким чином заробітна плата обслуговуючого персоналу становить 140,55 грн.

Сума витрат за розділом «Поточний ремонт устаткування» приймається рівною 3% від балансової вартості обладнання, що дорівнює 7000 грн., становить 210,00 грн. Інші витрати складають 5% від суми всіх попередніх витрат на утримання та експлуатацію обладнання за розділом «Інші витрати». Сума всіх попередніх розділів дорівнює 7677,15 грн., 5% від суми становлять 383,86 грн. Зведемо отримані дані в таблиці 6.5.

Таблиця 6.5 – Кошторис витрат на утримання і експлуатацію обладнання

Найменування статей витрат	Сума, грн.
Амортизація обладнання	7000,00
Експлуатація обладнання (крім витрат на поточний ремонт)	326,60
Основна і додаткова заробітна плата обслуговуючого персоналу з відрахуваннями на ЄСВ	140,55
Поточний ремонт обладнання	210,00
Інші витрати	383,86
Всього	8061,01

Витрати на оплату машинного часу ЕОМ для написання і налагодження програми визначають по формулі (6.8):

$$C_{\text{мо}} = P_{\text{екс}} \cdot t_{\text{мо}}, \quad (6.8)$$

де $C_{\text{мо}}$ – витрати на оплату машинного часу, грн.;

$P_{екс}$ – експлуатаційні витрати на одну годину машинного часу, грн./машино-год.;

$t_{мо}$ – машинний час ЕОМ для написання і відладки цього програмного продукту, машино-год.

Для розрахунку експлуатаційних витрат на одну годину машинного часу використовуваної ЕОМ розраховують діленням суми витрат за кошторисом витрат на утримання та експлуатацію обладнання (таблиця 6.5) на річний ефективний фонд часу роботи ПК. Річний ефективний фонд часу роботи ПК дорівнює 1800 годин.

В результаті експлуатаційні витрати на одну годину машинного періоду складають 4,48 грн./машино-год.

Розробка та відладка програмного продукту займає 40 днів в одну зміну по 8 робочих годин, що в сумі становить 320 годин.

Таким чином:

$$C_{мо} = 4,48 \cdot 320 = 1433,6 \text{ грн.}$$

Витрати на оплату машинного часу склали 1433,6 грн.

6.2.5 Інші витрати

В інші прями витрати входять витрати на комерційне програмне забезпечення, яке використовується при розробці, а саме загальне ПЗ, що використовується постійно при роботі з ПК (Windows 10 Professional 32/64bit) – 5780 грн, без ПДВ.

Розрахунок інших прямих витрат виконується за формулою (6.9):

$$S = \frac{Ц_{по} \cdot T_{ктс}}{\Phi_{еф.ктс} \cdot T_{с.по}}, \quad (6.9)$$

де C_{no} – ціна програмного забезпечення;

T_{kmc} – машинний час КТС, необхідне користувачу для розробки системи, машино-год/рік;

$\Phi_{ef.kmc}$ – річний ефективний фонд часу роботи КТС, машино-год/рік;

$T_{c.no}$ – строк служби програмного забезпечення, років.

$$S = \frac{5780 \cdot 320}{2 \cdot 1800} = 513,8 \text{ грн.}$$

Таким чином інші витрати становлять 513,8 грн.

6.2.6 Накладні витрати

До накладних витрат відносяться витрати на загальне управління і загальногосподарські потреби (заробітна плата апарату управління, канцелярські витрати і так далі), зміст і експлуатацію будинків. Накладні витрати включаються у вартість розробки програми непрямым шляхом – у відсотках до основної заробітної плати розробників. В даному випадку накладні витрати складають 40% до основної заробітної плати розробників, що складає 29750 грн, становить 11900 грн.

6.3 Калькуляція кошторисної вартості робіт

Визначення повної собівартості програмного продукту здійснюється шляхом додавання операційних витрат та витрат по збуту. Визначення сукупності операційних витрат здійснюється шляхом додавання витрат на оплату праці, відрахувань на ЄСВ, матеріальних витрат, витрат по спожитим послугам, амортизації та іншим.

Визначення ціни реалізації програмного продукту здійснюється виходячи з повної собівартості інноваційного продукту та відсотка

очікуваного прибутку, що становить 30%, і відсотка податку на додану вартість (ПДВ), що становить 20%.

Результати визначення витрат на розробку у вигляді калькуляції кошторисної вартості робіт наведені в таблиці 6.6.

Таблиця 6.6 – Калькуляція кошторисної вартості робіт з розробки програмного продукту та ціна продукту

Найменування статей витрат	Сума, грн.	Питома вага до підсумку, %
Основна заробітна плата	29750	55,1
Додаткова заробітна плата	2975	5,51
Відрахування на ЄСВ	7199,5	13,34
Витрати на матеріали	220,5	0,4
Витрати на оплату машинного часу	1433,6	2,66
Інші витрати	513,8	0,95
Накладні витрати	11900	22,04
Разом витрати	53992,4	100
Прибуток (30%)	16197,72	–
Всього	70190,12	–
ПДВ (20%)	14038	–
Всього	84228,12	–

6.4 Оцінка економічної ефективності

Цільовою аудиторією є учні, студенти, працівники компаній, які мають різний рівень знань, які можуть об'єктивно оцінити корисність курсу і надати повний відгук. Розроблений модуль, заснований на Moodle, націлений на економію часу і підтримку надання потрібних матеріалів, тобто відгуків, для поліпшення навчальних курсів за допомогою відповідних доповнень.

Створений програмний продукт має функцію анонімного відгуку про курс, що в свою чергу зменшить час визначення якості навчальних курсів, а також підвищить зацікавленість в покращенні навчання приблизно в 2-3 рази.

Планується продаж створеного ПЗ за ціною 1200 гривень в кількості 150 екземплярів. Таким чином очікується надходження від продажу у розмірі 180000 гривень. З урахуванням ризику 50% надходження від продажів складає 90000 гривень, що більше вартості програмного забезпечення, в якій вже була врахована рентабельність 30%.

Таким чином можна зробити висновок про економічну доцільність розробки даного програмного забезпечення.

7 ОХОРОНА ПРАЦІ ТА БЕЗПЕКА У НАДЗВИЧАЙНИХ СИТУАЦІЯХ

7.1 Аналіз потенціальних небезпек

В ході виконання дипломної роботи проводиться дослідження інструментів онлайн навчання та програмна реалізація модулю для системи дистанційного навчання. Під час проведення дослідження та розробки програмного продукту основні роботи виконано за допомогою ПК.

Стан здоров'я користувача комп'ютера значною мірою визначається типом роботи, який визначається основними потенційними небезпеками, що виникають при роботі з обладнанням:

- небезпека ураження електричним струмом, внаслідок недотримання правил електробезпеки або виходу з ладу електроприладів;
- негативний вплив електромагнітних полів, які створюються офісним обладнанням, на організм працюючого;
- негативний вплив на опорно-руховий апарат тривалим перебуванням працюючого в одній позі без активних рухів.
- незадовільній ергономічності робочого місця та неправильній організації робочого часу;
- негативний вплив недостатнього освітлення робочої зони на зір та продуктивність роботи працюючого, внаслідок несправності освітлювальних приладів або неправильного проектування освітлювальної системи;
- негативний вплив на зір через тривалу роботу з об'єктами малого розміру;
- негативний вплив незадовільних параметрів повітряного середовища робочої зони на здоров'я працюючого, внаслідок неправильного проектування системи вентиляції або несправності її несправності;
- негативний вплив підвищеного рівня шуму на психоемоційний стан працюючого, який пов'язаний з використанням застарілої периферійної техніки, кондиціонерів, копіювальної техніки, освітлювальних приладів;

- небезпека загоряння у зв'язку із несправністю електричного обладнання, недотримання, або порушення правил протипожежної безпеки обслуговуючим персоналом, що може призвести до пожежі;
- інтелектуальним навантаженням – необхідність швидкого прийняття рішень, творча діяльність, постійне сприйняття і оцінка нової інформації, високий ступінь складності завдання;
- емоційним навантаженням – ступінь відповідальності за завдання для виконання, дефіцит часу, значимість помилки;
- монотонністю трудового процесу – багаторазове повторення одноманітних дій, тривалість зосередженого спостереження;
- неправильні дії персоналу у надзвичайних ситуаціях.

При роботі з комп'ютерною технікою зокрема для користувачів можуть бути виявлені наступні небезпечні чинники:

- мерехтіння екрану або періодична зміна його яскравості;
- підвищена зовнішня освітленість екрану;
- нестабільність зображення на екрані; – аномальний рівень освітленості робочої зони;
- підвищена нерівномірність розподілу яскравості в полі зору користувача та підвищені пульсації світлового потоку джерел світла;
- підвищена яскравість фрагментів світлового зображення або світла, що потрапляє в поле зору користувача.

Статистичні дані показують, що основними та найпоширенішими комп'ютерними захворюваннями є запалення очей, глаукома, катаракта, різь в очах, головний біль, болі в області м'язів обличчя і ший, хвороби серцево-судинної системи, біль у хребті, біль при русі рук, свербіж шкіри, дерматити, лущення, нервово-психічні розлади.

Для створення сприятливих умов роботи з комп'ютером необхідна відповідність нормативно-правовим актам з охорони праці, які регламентують показники стану навколишнього середовища:

- достатню площу і об'єм приміщення, де встановлено комп'ютер;

- показники мікроклімату робочого приміщення (температура повітря, відносна вологість, та ін.);
- параметри світлового середовища робочого місця і робочої поверхні (наявність природного і штучного освітлення, освітленість екрана монітора, освітленість горизонтальної поверхні столу, відсутність прямого і відбитого сліпучого відблиску джерел світла, відсутність пульсації освітленості);
- рівень шуму і вібрації робочих місць в межах норм;
- відсутність в повітрі робочої зони шкідливих речовин.

Нижче розглядаються заходи з охорони праці, виробничої санітарії, гігієни праці та пожежної безпеки при експлуатації ЕОМ з візуальними дисплейними терміналами (ВДТ) в будівлях і приміщеннях.

7.2 Заходи щодо забезпечення безпеки

Забезпечення безпеки праці, організація оптимальних умов праці, а також захист навколишнього середовища в умовах праці за допомогою персонального комп'ютера (ПК) здійснюються із дотриманням стандартів та заходів, передбачених законодавством України. З метою зменшення негативного впливу ПК на організм людини враховані вимоги при організації роботи за НПАОП 0.00-7.15-18 «Вимоги щодо безпеки та захисту здоров'я працівників під час роботи з екранними пристроями» [54].

Приміщення з візуальними дисплейними терміналами (ВДТ) обладнано аптечкою першої медичної допомоги.

Персональний комп'ютер підключений до мережі з напругою 220В і частотою 50Гц. Згідно ПУЕ приміщення відноситься до 1-ї категорії «Приміщення без підвищеної небезпеки». », оскільки існує можливість торкання корпусів апаратури й батареї центрального опалення.

У приміщенні відсутні відкриті струмопровідні частини з напругою понад 12В. При виконанні правил ПУЕ «Правила улаштування

електроустановок» ураження електричним струмом можливо тільки в разі несправності апаратури і живлячих кабелів.

Апаратура, яка перебуває під небезпечною для життя напругою, передбачає забезпечення електробезпеки. Щоб уникнути ураження електричним струмом все електрообладнання на робочому місці заземлено за ГОСТ 12.1.030-81 ССБТ «Захисне заземлення, занулення» [55]. Захисне заземлення, опір якого складає 4 Ом, забезпечує захист людей від ураження електричним струмом при дотику до металевих неструмопровідних частин, які можуть опинитися під напругою в результаті пошкодження ізоляції. Захисне заземлення виконано навмисним електричним з'єднанням металевих частин електроустановок з «землею» або її еквівалентом.

Над кожною розеткою було розміщено попереджувальний напис «220В». Індикаторні лампи в блоках ЕОМ сигналізують про включення блоку в мережу.

Електрична ізоляція забезпечується конструкцією приладів і способами їх підключення. Під час роботи якість ізоляції періодично перевіряється шляхом вимірювання її опору. В електроустановках до 1000 В, згідно з ПУЕ, мінімальне значення опору ізоляції електричних і світлових проводів на відстані між сусідніми запобіжниками або кінцевими запобіжниками між проводом і землею не повинно бути менше 0,5 МОм.

Згідно з ПУЕ, захисне заземлення застосовується у всіх електроустановках зі змінним струмом під напругою 380 В і вище та постійним струмом 440 В і вище. До захисних заземлень застосовуються шафи-трансформатори, оболонки кабелів, металеві огорожі для електроустановок згідно з ГОСТ 12. 1. 030-81 «Електробезпека. Захисне заземлення, занулення» [55].

Пристрої заземлення, що використовуються для заземлення електроустановок, відповідають всім вимогам до заземлення для цих електроустановок.

Матеріали, конструкція та розміри заземлювачів стійкі до механічних, хімічних та термічних впливів протягом терміну їх служби.

Для вирівнювання потенціалів металоконструкції та виробничі потужності підключаються до мережі заземлення.

Поділ електричної мережі включає поділ мережі на окремі електрично не пов'язані ділянки за допомогою ізолюючих трансформаторів. Підвищення електричної безпеки є наслідком зменшення пропускної здатності мережі та збільшення опору ізоляції фаз мережі відносно землі.

Зменшення напруги електроустановки істотно підвищує умови електробезпеки. Так при напрузі 10-12 В і електричному опорі тіла людини 5-6 кОм сила струму, що протікає через людину, не перевищить 2 мА, згідно ГОСТ 12.1.030-81 «Електробезпека. Захисне заземлення, занулення» [55].

Відповідно до НПАОП 0.00-1.28-10 «Правила захисту праці при роботі електронно-обчислювальних машин» [56] розміщення приміщень, призначених для роботи з ВДТ, у підвалах та підвалах є неприпустимим. Відповідно до НАПБ Б.03.002-2007 «Стандарти визначення категорій приміщень, будівель та зовнішніх установок на вибухонебезпеку та пожежну безпеку» [57] також забороняється розміщувати вибухонебезпечні простори категорій А та В та промислових мокрих процесів поблизу приміщення, де знаходяться ЕОМ, розташовані навіть вище або нижче таких просторів. Крім того, приміщення для роботи з ВДТ не повинні межувати з приміщеннями, де рівень шуму та вібрації перевищує допустимі значення, відповідно до ДСН 3.3.6.037-99 «Санітарні норми промислового шуму, ультразвуку та інфразвуку» [58].

Європейська Директива (ЕС) 90/270 ЕЕС, відповідно до вимог розділу «Мінімальні вимоги до безпеки та охорони здоров'я», відповідає основним вимогам до моніторів, які суворо регламентують безпечні умови праці та охорону здоров'я людей, що працюють з комп'ютерами:

- символи на екрані чіткі і добре розрізняються;
- зображення позбавлене блимання;

- яскравість та / або контрастність легко регулюються;
- екрани вільні від відблисків і відбиття;
- випромінювання знижені до надзвичайно малих рівнів.

7.3 Заходи щодо забезпечення виробничої санітарії та гігієни праці

Для зниження шкідливого впливу комп'ютера на людину повинні дотримуються певні вимоги до умов роботи, робочого місця і повинен дотримуватися режим роботи за комп'ютером.

У розділі 4 ДСанПіН 3.3.2.007-98 «Гігієнічні вимоги до організації роботи з візуальними дисплейними терміналами електронно-обчислювальних машин» [59] передбачені гігієнічні вимоги до об'ємно-планувальних рішень будівель та приміщень для роботи з ВДТ ЕОМ. Розміщення робочих місць з ВДТ ЕОМ в підвальних приміщеннях, на цокольних поверхах заборонено.

В обчислювальному центрі дотримуються норми ДБН В.2.5-28:2018 «Природне та штучне освітлення» [60]. Відносно штучного освітлення: освітленість робочих місць в горизонтальній площині на рівні 0,8 м від підлоги не менше 400лк, вертикальна освітленість в площині екранів дисплея – не менше 300лк, коефіцієнт пульсації освітленості – не більше 10%, забезпечуються розташованими на стелі світильники УСП 35 з люмінесцентними лампами типу ЛБ-40.

Звукоізоляція огорожувальних конструкцій приміщення з ВДТ забезпечує параметри шуму, що відповідають вимогам ДСН 3.3.6.037-99 «Санітарні норми виробничого шуму, ультразвуку та інфразвуку» [61].

Нормовані параметри мікроклімату, іонного складу повітря, вмісту шкідливих речовин відповідають вимогам ДСН 3.3.6.042-99 «Санітарні норми мікроклімату виробничих приміщень» [62].

Віконні отвори приміщень для роботи з ВДТ обладнані регульованими пристроями. Для внутрішнього оздоблення приміщень з ВДТ використовують

дифузно-відбивні матеріали з коефіцієнтами відбиття для стелі 0,7-0,8, для стін 0,5-0,6.

Покриття підлоги матове з коефіцієнтом відбиття 0,3-0,5. Поверхня підлоги рівна, неслизька, з антистатичними властивостями. В обробці інтер'єру приміщень ВДТ не застосовуються полімерні матеріали, що виділяють у повітря шкідливі хімічні речовини. Полімерні матеріали для внутрішнього оздоблення приміщень з ВДТ використовуються з наявністю дозволу органів і установ державної санітарно-епідеміологічної служби.

Площа на одного працівника регламентується ДСанПіН 3.3.2-007-98. «Гігієнічні вимоги до організації роботи з візуальними дисплейними терміналами електронно-обчислювальних машини» [59] становить не менше 6 м². Отже в приміщенні довжиною $a = 4$ м, шириною $b = 3$ м, висотою $h = 3$ м загальною площею $S = 12$ м² можна розмістити не більше 2-х робочих місць. Крім площі приміщення береться до уваги і об'єм приміщення, який становить не менше 20 м³ на одне робоче місце, що передбачено ДСанПіН 3.3.2.007-98. Приміщення має об'єм в 36 м³, а це означає, що в приміщенні можна розмістити тільки одне робоче місце з ВДТ ЕОМ.

У приміщенні забезпечений приплив свіжого повітря, кількість якого визначається техніко-економічним розрахунком і вибором схеми системи вентиляції.

При невеликому забрудненні зовнішнього повітря кондиціонування приміщень здійснюється із перемінними витратами зовнішнього повітря й рециркуляційного.

Для вибору типу природної вентиляції необхідно визначити коефіцієнт кратності повітрообміну за формулою (7.1):

$$k = \frac{L}{V}, \quad (7.1)$$

де L – необхідний повітрообмін, м³/год;

V – об'єм приміщення, вільний від обладнання ($V=33\text{м}^3$).

В приміщенні одне робоче місце, отже $L_1=30\text{м}^3/\text{год}$. Розрахунок L_2 виконується за формулою (7.2):

$$L_2 = \frac{Q \cdot 3600}{C \cdot \rho \cdot (t_e - t_n)}, \quad (7.2)$$

де Q – надлишок теплоти, кВт;

C – масова теплоємність повітря, кДж/(кгК);

ρ – густина повітря, кг/м³;

t_e – температура видаляемого повітря, градусів за Цельсієм;

t_n – температура зовнішнього повітря, градусів за Цельсієм.

Температура зовнішнього повітря для зимового періоду дорівнює -8 градусів, а для літнього – +25 градусів за Цельсієм. Надлишок теплоти в приміщенні дорівнює 0,35 кВт. L_2 для зимового періоду складає 33,8 м³/год, а для літнього – 525 м³/год, де $k = \frac{525}{33} = 15,9$. Оскільки $k > 10$, в приміщенні необхідно встановити штучну механічну вентиляцію.

Приміщення розташоване на другому поверсі виробничого корпусу, що допустимо і відповідає перерахованим вище правилам і нормам [57].

В ДСанПіН 3.3.2.007-98 передбачені гігієнічні вимоги до організації обладнання робочих місць ВДТ ЕОМ. Конструкція робочого столу повинна відповідати сучасним вимогам ергономіки і забезпечувати оптимальне розміщення на робочій поверхні використовуваного обладнання (дисплея, клавіатури, принтера) і документів. Висота робочої поверхні робочого столу з ВДТ повинні регулюватися в межах 680 ... 800 мм, а ширина і глибина – забезпечувати можливість виконання операцій в зоні досяжності моторного поля (рекомендовані розміри: 600 ... 1400 мм, глибина – 800..1000 мм). Приймаємо робочу поверхню з розмірами 1400 × 800. Щоб уникнути відблисків на екрані вікон доцільно розмістити робоче місце торцем до вікна

не ближче ніж на 1 м. Площа дозволяє розмістити додаткові меблі для зберігання документів.

Велике значення надається ергономічним та естетичним вимогам. Цього домоглися за допомогою колірного рішення приміщення й устаткування з урахуванням вимог ГОСТ 17.2.032-78 «Рабочее место при выполнении работ сидя» [63] – у приміщеннях стіни пофарбовані фарбами пастельних тонів.

Для підвищення продуктивності праці, а також зниження рівня стомлюваності при роботі з ЕОМ передбачені десятихвилинні перерви через кожну годину.

Велике значення має задоволення обстановки ергономічним і естетичним вимогам. Цього домоглися за допомогою колірного рішення приміщення і обладнання обчислювального центру з урахуванням вимог ГОСТ 17.2.032-78 «Рабочее место при выполнении работ сидя» – в приміщеннях стіни пофарбовані фарбами пастельних тонів [63].

Під час організації робочого простору враховано індивідуальні антропометричні параметри користувача з відповідними допусками на можливі зміни робочих поз та потребу у переміщеннях.

Проведено заходи мінімізації негативного впливу випромінювання від відео дисплейних терміналів на органи зору.

При організації роботи для збереження здоров'я, запобігання професійних захворювань і підтримки працездатності передбачені перерви для відпочинку. Тривалість обідньої перерви визначається чинним законодавством про працю і Правилами внутрішнього трудового розпорядку підприємства.

Передбачено такі внутрішньо-змінні режими праці та відпочинку при роботі з ПК при 8-годинній денній робочій зміні в залежності від характеру праці. За характером трудової діяльності виділено три професійні групи згідно з чинним класифікатором професій:

- розробники програм (програмісти) – виконують роботу з документацією, при необхідності, і з інтенсивним обміном інформацією з ЕОМ і значним ступенем прийняття рішень. Робота характеризується інтенсивною розумовою творчою працею з підвищеним напруженням зору, концентрацією уваги на фоні нервово-емоційного напруження, вимушеною робочою позою, загальною гіподинамією, періодичної навантаженням на кисті верхніх кінцівок;

- оператори електронно-обчислювальних машин – виконують роботу, пов'язану з урахуванням інформації, отриманої з ВДТ за попереднім запитом або надходить з нього, супроводжується перервами різної тривалості, пов'язана з виконанням іншої роботи і характеризується як робота з напруженням зору, невеликим фізичним зусиллям, нервовим напруженням середнього ступеня і виконується у вільному темпі;

- оператор комп'ютерного набору – виконує одноманітні за характером роботи з документацією та клавіатурою з нечастими нетривалими перемиканнями погляду на екран дисплея, з введенням даних з високою швидкістю. Робота характеризується як фізична праця з підвищеним навантаженням на кисті верхніх кінцівок на фоні загальної гіподинамії з напруженням зору, нервово-емоційним напруженням.

Таким чином передбачено такі режими праці і відпочинку при роботі з ЕОМ:

- для розробників програм із застосуванням ЕОМ призначені регламентовані перерви для відпочинку по 15 хвилин через кожну робочу годину на ВДТ;

- для операторів ЕОМ призначені регламентовані перерви тривалістю 15 хвилин через кожні дві робочі години;

- для операторів комп'ютерного набору призначені регламентовані перерви для відпочинку тривалістю 10 хвилин після кожної години роботи на ВДТ.

У всіх випадках, коли виробничі умови не дозволяють використовувати регламентовані перерви, час безперервної роботи з ВДТ не повинен перевищувати 4 години.

Для зменшення наслідків негативного впливу монотонної роботи рекомендується використовувати чергування операцій з усвідомленим текстом і числовими даними (зміна змісту роботи).

Для зменшення нервових та емоційних навантажень, стомлення зорового аналізатора, поліпшення мозкового кровообігу, подолання негативних наслідків гіподинамії, запобігання втоми рекомендуються використовувати деякі перерви для виконання комплексу вправ.

Активний відпочинок має полягати у виконанні комплексу гімнастичних вправ, спрямованих на зняття нервового напруження, м'язове розслаблення, відновлення функцій фізіологічних систем, що порушуються протягом трудового процесу, зняття втоми очей, поліпшення мозкового кровообігу і працездатності.

Всі перелічені вище заходи проведені з метою поліпшення умов праці програміста, і, отже, сприяє підвищенню ефективності праці.

7.4 Заходи щодо забезпечення пожежної безпеки

В приміщенні обчислювального центру джерелом загоряння можуть бути електронні схеми ПК, прилади для технічного обслуговування, електроустаткування, кондиціонери повітря, де за рахунок різних порушень відбувається перегрів тих чи інших елементів, електричні іскри, дуга, що можуть визвати загоряння горючих матеріалів, дерев'яні та пластикові стільці та столи, папір, перевантаження мережі живлення, людські чинники тощо.

Заходи щодо протипожежного захисту розділяються на організаційні, експлуатаційні, технічні й режимні (спеціальні).

Організаційні заходи: навчання робітників та службовців правилам пожежної безпеки, організація пожежної охорони, проведення бесід, лекцій, видання необхідних інструкцій, плакатів і т.п.

Технічні заходи передбачають дотримання протипожежних правил і норм при устрої систем опалення, вентиляції, кондиціонуванні повітря, блисковкозахисті при спорудженні будинків, установці технологічного устаткування й ін.

Експлуатаційні заходи передбачають правильну експлуатацію систем опалення, вентиляції й кондиціонування повітря, блисковкозахисту технологічних машин і устаткування, правильне утримання будинків і територій і т.п.

Режимні заходи передбачають заборону або обмеження застосування відкритого вогню в вогнебезпечних місцях, паління в невстановлених місцях, обов'язкове дотримання норм і правил при роботі з вогнебезпечними й вибухонебезпечними речовинами.

Одним з найбільш важливих завдань пожежної профілактики є класифікація пожежі для приміщень обчислювального центру, що встановлюється згідно ДСТУ Б В.1.1-36:2016 «Визначення категорій приміщень, будинків та зовнішніх установок за вибухопожежною та пожежною безпекою» категорія приміщення по вибухопожежобезпечності відноситься до категорії Д [64].

Захист конструкцій будівель від руйнування та підтримка їх достатньої стійкості в умовах впливу високої температури при пожежі. Залежно від меж вогнестійкості будівельних конструкцій ДБН В.1.1-7:2016 «Пожежна безпека об'єктів будівництва. Загальні вимоги» установлені вісім ступенів вогнестійкості будинків: I, II, III, IIIa, IIIб, IV, IVa, V [65].

Враховуючи високу вартість електронного обладнання, а також категорію їх пожежної безпеки, будівлі повинні мати ступінь вогнестійкості не нижче III. Тому для виготовлення будівельних конструкцій використовують, як правило, цегла, залізобетон, скло, метал та інші негорючі

матеріали. Застосування дерева – обмежена, а в разі його використання необхідно просочувати вогнезахисним составом – антипіреном або покривати вапняно-цементною, азбестоцементною, гіпсовою штукатуркою. При виготовленні будівельних конструкцій слід застосовувати спеціальні фарби, які в умовах звичайної експлуатації захищають метал від корозії, а при пожежі спучуються, що призводить до збільшення їх термічного опору і до збільшення межі вогнестійкості забарвлених конструкцій. Влаштуйте протипожежні перешкоди у вигляді протипожежних стін та перегородок, щоб запобігти поширенню вогню з однієї частини будівлі на іншу.

Пожежна профілактика – це комплекс організаційних і технічних заходів, спрямованих на забезпечення безпеки людей, запобігання пожежі, обмеження його розповсюдження, а також на створення умов для успішного гасіння пожежі.

Протидимний захист будівель, спрямований на незадимлюваність евакуаційних шляхів, окремих приміщень і видалення продуктів горіння в певному напрямку, є першорядним завданням пожежної профілактики.

Згідно ДСТУ 2273 – 2006 Протипожежна техніка. Терміни та визначення основних понять [66] для виявлення початкової стадії загоряння, швидкого й точного оповіщення служби пожежної охорони про час і місце виникнення пожежі встановлені системи автоматичної пожежної сигналізації (АПС). Вони можуть самостійно пускати в хід установки пожежогасіння, коли пожежа не досягла великих розмірів. Системи АПС складаються з пожежних сповіщувачів, ліній зв'язку й прийомних пультів (станцій).

Для подачі сигналу про пожежу встановлені пожежні сповіщувачі. Залежно від контролюваного фактору, вони діляться на теплові, димові, світлові, комбіновані. Приміщення обчислювального центру обладнано автоматичними пожежними сповіщувачами, що реагують на підвищення температури, дим, полум'я, наприклад, сповіщувачі моделей ДТЛ, ІТМ.

До первинних засобів гасіння пожеж, призначених для локалізації невеликих загорянь, належать пожежні цівки, внутрішні пожежні

водопроводи, вогнегасники, сухий пісок, азбестові ковдри й т.п. Згідно НАПБ Б.03.001-2004 «Типові норми належності вогнегасників» [67] при захисті від пожежі приміщення з ПК використовуються вуглекислотні вогнегасники ВВК-1,4, ВВК-2 або аерозольні водопінні вогнегасники ВВПА-400, можливо також використання порошкових вогнегасників. Згідно з НАПБ Б.03.001-2004 [67] приміщення з ПК необхідно оснащувати переносними вуглекислотними вогнегасниками з розрахунку один вогнегасник ВВК-1,4 або ВВК-2 або ВВПА-400 на три ПК, але не менше ніж один вогнегасник, перелічених типів на приміщення.

На випадок пожежі передбачені шляхи евакуації працюючих – проходи, проїзди, евакуаційні виходи згідно до ДБН В.1.1.7-2002 «Пожежна безпека об'єктів будівництва» [68].

Евакуаційні виходи розташовуються розосереджено, а їх кількість – не менше двох на споруду. При кількості працюючих у зміні до 50 чоловік передбачений другий (запасний вихід) на зовнішні сходи.

Коридори й проходи, призначені для евакуації, мають найменшу довжину й мінімальну кількість поворотів. На всьому протязі проходу немає порогів або проміжних щаблів.

Важливу роль у забезпеченні безпечного виходу людей грає протидимовий захист евакуаційних ходів. У будинках висотою до 9 поверхів незадимлюваність сходових кліток на час евакуації досягається їхньою ізоляцією від підвалів, горищ і поверхів. Для цього влаштовані відособлені входи в підвали, вхід на сходову клітку з поверхів здійснюють через тамбур – шлюз із підпором повітря, відокремлюють горища від сходових кліток перекриттями з негорючих матеріалів.

У таблиці 7.1 наведені відстані від найбільш віддаленого робочого місця до найближчого евакуаційного виходу.

Таблиця 7.1 – Відстані від найбільш віддаленого робочого місця до найближчого евакуаційного виходу

Об'єм приміщення, тис. м ³	Ступінь вогнестійкості будівлі	Відстань, м, при щільності людського потоку в загальному проході, люд/м ²		
		до 1	від 1 до 3	від 3 до 5
Незалежно від об'єму	I, II, III	Не обмежується		
	IV	160	95	65
	V	120	70	50

Для відводу надлишкового тепла від персонального комп'ютера задіяні системи вентиляції й кондиціонування повітря, але вони становлять додаткову пожежну небезпеку, тому що, з одного боку, вони забезпечують подачу кисню, тобто окислювача, в усі приміщення, а з іншого боку – при виникненні пожежі поширюють вогонь і продукти горіння по всіх приміщеннях і установкам, з якими зв'язані повітроводи. Однак система вентиляції і кондиціонування повітря була обладнана пристроєм, який забезпечує автоматичне відключення вентиляції, в разі виникнення пожежі.

Для тепло- і звукобірної ізоляції систем вентиляції застосовують такі негорючі матеріали, як мати з мінеральної вати, скловолокна, стиліту.

Напруга до електроустановок подається по кабельних лініях, які становлять особливу пожежну небезпеку з рівнем вогнестійкості не менше 0,5 год.

Одним з найбільш важливих завдань пожежної профілактики є захист будівельних конструкцій від руйнування й забезпечення їхньої достатньої міцності в умовах впливу високих температур при пожежі.

Підпільні простори під знімними підлогами розділяють неспаленими перегородками з межею вогнестійкості не менш 0.75 г на відсіки площею не більше 250 м², комунікації через які прокладають у спеціальних обоймах із застосуванням негорючих ущільнюючих матеріалів.

7.5 Заходи щодо забезпечення безпеки в надзвичайних ситуаціях

7.5.1 Порядок проведення аварійно-рятувальних та інших невідкладних робіт

Аварійно-рятувальні та інші невідкладні роботи (АРІНР) – роботи, спрямовані на пошук, рятування і захист населення, уникнення руйнувань і матеріальних збитків, локалізацію зони впливу небезпечних чинників, ліквідацію чинників, що унеможливають проведення таких робіт або загрожують життю рятувальників [69]. Рятувальні та інші невідкладні роботи включають також створення умов для наступного проведення відновлювальних робіт. РІНР поділяють на рятувальні роботи і невідкладні роботи. Проведення аварійно-рятувальних та інших невідкладних робіт під час ліквідації наслідків надзвичайних ситуацій здійснюється безоплатно [69].

Проведення аварійно-рятувальних та інших невідкладних робіт з ліквідації наслідків надзвичайних ситуацій у мирний час та в особливий період включає [69], [70]:

- організацію та управління аварійно-рятувальними та іншими невідкладними роботами;
- розвідку районів, зон, ділянок, об'єктів проведення робіт з ліквідації наслідків надзвичайної ситуації;
- визначення та локалізацію зони надзвичайної ситуації;
- виявлення та позначення районів, які зазнали радіоактивного, хімічного забруднення чи біологічного зараження (крім районів бойових дій);
- прогнозування зони можливого поширення надзвичайної ситуації та масштабів можливих наслідків;
- ліквідацію або мінімізацію впливу небезпечних чинників, які виникли внаслідок надзвичайної ситуації;
- пошук та рятування постраждалих, надання їм екстреної медичної допомоги і транспортування до закладів охорони здоров'я;
- евакуацію або відселення постраждалих;

- виявлення та знешкодження вибухонебезпечних предметів;
- санітарну обробку населення та спеціальну обробку одягу, техніки, обладнання, засобів захисту, будівель, споруд і територій, які зазнали радіоактивного, хімічного забруднення чи біологічного зараження;
- надання медичної допомоги постраждалим, здійснення санітарно-протиепідемічних заходів, забезпечення санітарного та епідемічного благополуччя населення в районі виникнення надзвичайної ситуації та місцях тимчасового розміщення постраждалих;
- запровадження обмежувальних заходів, обсервації та карантину;
- надання психологічної та матеріальної допомоги постраждалим, проведення їх медико-психологічної реабілітації;
- забезпечення громадського порядку в зоні надзвичайної ситуації;
- проведення першочергового ремонту та відновлення роботи пошкоджених об'єктів життєзабезпечення населення, транспорту і зв'язку;
- здійснення заходів соціального захисту постраждалих внаслідок надзвичайних ситуацій;
- проведення інших робіт та заходів залежно від характеру та виду надзвичайної ситуації.

Управління під час ліквідації наслідків надзвичайних ситуацій полягає у керівництві силами цивільного захисту, а саме: оперативно-рятувальна служба цивільного захисту, аварійно-рятувальні служби, формування цивільного захисту, спеціалізовані служби, пожежно-рятувальні підрозділи або частини, добровільні формування цивільного захисту, при проведенні аварійно-рятувальних та інших невідкладних робіт.

Головна мета управління під час виконання АРІНР – забезпечити своєчасне та ефективне виконання завдань у зоні надзвичайної ситуації (НС) у найкоротші терміни та з мінімальними людськими й матеріальними втратами від наслідків НС наявними ресурсами, тобто силами та засобами різноманітного призначення для виконання АРІНР [69].

Управління роботами починається з моменту виникнення НС і завершується після ліквідації її наслідків. Аварійно-рятувальні та інші невідкладні роботи здійснюються у три етапи.

На першому етапі робіт здійснюється вирішення завдань [69]:

- екстрений захист населення;
- запобігання розвитку чи зменшення впливу наслідків;
- підготовка до виконання рятувальних та інших невідкладних робіт.

Щодо екстреного захисту населення основними заходами є [69]:

- оповіщення про небезпеку;
- використання засобів захисту,,
- додержання режимів поведінки;
- евакуація з небезпечних у безпечні райони;
- здійснення санітарно-гігієнічної, протиепідемічної профілактики і надання медичної допомоги;
- локалізація аварій;
- зупинка чи зміна технологічного процесу виробництва;
- попередження (запобігання) і гасіння пожеж.

На другому етапі робіт проводяться такі заходи [69]:

- пошук потерпілих;
- витягання потерпілих з-під завалів, з палаючих будинків, пошкоджених транспортних засобів;
- евакуація людей із-зони лиха, аварії, осередку ураження;
- надання медичної допомоги;
- санітарна обробка людей;
- знезараження одягу, майна, техніки, території;
- проведення інших невідкладних робіт, що сприяють і забезпечують здійснення рятувальних робіт.

На третьому етапі робіт здійснюється вирішення завдань щодо забезпечення життєдіяльності населення у районах, які потерпіли від наслідків НС [69]:

- відновлення чи будівництво житла;
- відновлення енерго-, тепло-, водо-, газопостачання, ліній зв'язку;
- організація медичного обслуговування;
- забезпечення продовольством і предметами першої необхідності;
- знезараження харчів, води, фуражу, техніки, майна, території;
- соціально-психологічна реабілітація;
- відшкодування збитків;
- знезараження майна, території, техніки.

Управління аварійно-рятувальними формуваннями має бути стійким, безперервним, оперативним, при цьому доцільно забезпечувати поєднання необхідного ступеня централізації з наданням підлеглим ініціативи у визначенні способів виконання завдань [69].

7.6 Висновки з розділу 7

Під час виконання розділу «Охорона праці та безпека в надзвичайних ситуаціях» був проведений аналіз потенційних небезпек на робочому місці, в робочій зоні і у виробничому середовищі відповідно до виду робіт, що розглядаються в проєкті.

На основі аналізу потенційних небезпек були обрані найбільш ефективні заходи, спрямовані на виключення або мінімізацію негативного впливу певних потенційно небезпечних і шкідливих факторів виробничого процесу.

Були розроблені заходи щодо забезпечення виробничої санітарії та гігієни праці, виробничих факторів санітарно-гігієнічного характеру, розроблені поліпшуючі заходи, проведені необхідні розрахунки.

Здійснено розробку заходів щодо пожежної безпеки спрямованих на збереження життя персоналу, обладнання, матеріальних ресурсів в умовах виникнення загорянь, організації ліквідації вогнищ загорянь, евакуації персоналу.

Також було описано порядок проведення аварійно-рятувальних та інших невідкладних робіт.

ВИСНОВКИ

В результаті виконання дипломної роботи було розглянуто цілу низку систем дистанційного навчання, плагінів і сторонніх засобів, які інтегруються в системи. Було проведено огляд ключових функціональних можливостей систем і загальних характеристик. Серед можливих функцій було проведено огляд необхідних ключових можливостей для будь-якої системи, так і додаткових – плагінів і сторонніх застосунків, які полегшують роботу викладача чи студентів. А також наведений огляд засобів сумісності інструментів онлайн навчання з навчальними системами. В ході роботи було розглянуто набір стандартів Sharable Content Object Reference Model (SCORM), Learning Tools Interoperability (LTI), а також інструменти External tools.

В ході огляду подібних систем дистанційного навчання було розглянуто та порівняно такі системи дистанційного навчання як ILIAS, Sakai, Docebo та Moodle, що надають схожі функції та засоби організації навчальних процесів. Результати даного етапу роботи представлені у вигляді порівняльних таблиць. За результатами аналізу було обрано систему дистанційного навчання Moodle, що є найбільш придатним програмним засобом для організації дистанційного навчання.

Дослідження показало, що можна виділити два принципово різні підходи до розширення систем дистанційного навчання: розробка внутрішніх модулів системи та зовнішніх незалежних застосувань, які інтегруються до систем.

Виконано огляд архітектури системи дистанційного навчання Moodle.

До системи дистанційного навчання Moodle було інтегровано такі сторонні застосунки: BigBlueButton – сервіс для проведення відеоконференцій та Virtual program Lab – інструмент перевірки програмного коду, як зовнішні засоби.

Проведений аналіз підходів щодо розширення функціональних можливостей показав, що підхід інтеграції застосунку як зовнішнього засобу, є найбільш придатним способом для вбудовування програми відкуків в систему Moodle.

Також було виконано дослідження інструментальних засобів розробки модулів для системи Moodle та технологій побудови веб-застосунків. Розроблений застосунок інтегрується до системи дистанційного навчання Moodle за LTI протоколом і являє собою просту форму анонімного зворотнього зв'язку студентами про курс керівнику. Даний модуль розпочинає роботу одразу після завантаження сторінки вебсайту системи Moodle і функціонує в залежності від типу користувача та етапу його роботи з модулем. Для студентів даний застосунок має вигляд звичайної форми відгуку.

Готовий програмний продукт вирішує актуальну проблему та містить в своїй роботі певну новизну та креативність зі сторони спроектованої архітектури з використанням досить нових веб технологій. Розроблена система повністю відповідає вимогам, що були сформульовані у технічному завданні. Завдання, які було поставлено в роботі були виконані в повному обсязі.

ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ

1. Миронова, Н.О. Дослідження засобів діджиталізації для систем дистанційного навчання [Електронний ресурс] / Н.О. Миронова, А.К. Чадаєва // Сучасні проблеми і досягнення в галузі радіотехніки, телекомунікацій та інформаційних технологій: X Міжн. наук.-практ. конф., 07–09 жовтня 2020 р.: тези доп. / Редкол.: Д. М. Піза. Електрон. дані. – Запоріжжя: НУ «Запорізька політехніка», 2020. – С.254-255. – 1 електрон. опт. диск (DVD-ROM). – назва з тит. екрана.
2. Дистанційне навчання [Електронний ресурс]. – Режим доступу: https://uk.wikipedia.org/wiki/Дистанційне_навчання.
3. Система управління навчанням [Електронний ресурс]. – Режим доступу: https://uk.wikipedia.org/wiki/Система_управління_навчанням.
4. ILIAS [Електронний ресурс]. – Режим доступу: <https://uk.wikipedia.org/wiki/ILIAS>.
5. Sakai [Electronic resource]. – Access mode: <https://www.sakailms.org/>.
6. Достоинства и недостатки дистанционного образования при использовании электронной системы Sakai [Электронный ресурс]. – Режим доступа: https://interactive-plus.ru/ru/article/118188/discussion_platform.
7. Docebo [Electronic resource]. – Access mode: <https://en.wikipedia.org/wiki/DoceboLMS>.
8. Moodle [Електронний ресурс]. – Режим доступу: <https://uk.wikipedia.org/wiki/Moodle>.
9. Moodle [Электронный ресурс]. – Режим доступа: <https://8d9.ru/program/moodle>.
10. 8 систем дистанционного обучения: какую выбрать школе, репетитору, тренеру? [Электронный ресурс]. – Режим доступа: <https://www.eduneo.ru/3-besplatnye-sistemy-distancionnogo-obucheniya-obzor/>.

11. Аналитическая записка « Выбор системы дистанционного обучения» [Электронный ресурс]. – Режим доступа: <https://ra-kurs.spb.ru/info/articles/?id=13>.
12. Відкриті системи дистанційного навчання [Електронний ресурс]. – Режим доступу: <http://ena.lp.edu.ua:8080/bitstream/ntb/11016/1/10.pdf>.
13. Plugin Types – MoodleDocs [Electronic resource]. – Access mode: https://docs.moodle.org/dev/Plugin_types.
14. Moodle Activities [Electronic resource]. – Access mode: <https://docs.moodle.org/310/en/Activities>.
15. BigBlueButton [Электронный ресурс]. – Режим доступа: <https://ru.wikipedia.org/wiki/BigBlueButton>.
16. BigBlueButton Features [Electronic resource]. – Access mode: <https://bigbluebutton.org/teachers/>.
17. BigBlueButton [Electronic resource]. – Access mode: <https://docs.bigbluebutton.org/>.
18. BigBlueButton [Електронний ресурс]. – Режим доступу: <https://bigbluebutton.org/integrations/>.
19. BigBlueButton for Sakai [Electronic resource]. – Access mode: <https://confluence.sakaiproject.org/display/BBB/Home>.
20. BigBlueButton for Docebo [Electronic resource]. – Access mode: https://www.docebo.com/knowledge-base/elearning-docebo-for-big-bluebutton/?utm_referrer=https://www.google.com/.
21. BigBlueButton for Moodle [Electronic resource]. – Access mode: https://moodle.org/plugins/mod_bigbluebuttonbn.
22. Virtual program Lab [Electronic resource]. – Access mode: <https://vpl.dis.ulpgc.es/>.
23. VmWare Learning Platform [Electronic resource]. – Access mode: <https://kb.vmware.com/s/article/79113>.
24. SCORM [Електронний ресурс]. – Режим доступу: <https://uk.wikipedia.org/wiki/SCORM>.

25. Стандарт SCORM и его применение [Электронный ресурс]. – Режим доступа: <http://cccp.ifmo.ru/scorm/index.html>.
26. IEEE 1484.12.1–2002 Learning Object Metadata Standard [Electronic resource]. – Access mode: <http://www.ieee.org/>.
27. Smythe C., Jackl A. IMS Content Packaging Information Model v1.1.4 // IMS Global Learning Consortium, Inc, [Electronic resource]. – Access mode: http://www.imsglobal.org/content/packaging/cpv1p1p4/imscp_bestv1p1p4.html.
28. Koper R. IMS Learning Design Best Practice and Implementation Guide. Final Specification / R. Koper, B. Olivier, T. Anderson. – 2003 [Electronic resource]. – Access mode: http://www.imsglobal.org/learningdesign/ldv1p0/imslld_bestv1p0.html.
29. IMS Simple Sequencing Behavior and Information Model v1.0 Final Specification [Electronic resource]. – Access mode: <http://www.imspjproject.org/>.
30. How do I create SCORM content [Electronic resource]. – Access mode: http://doc.claroline.net/en/index.php/How_do_I_create_SCORM_content%3F.
31. Icodeon SCORM Player Version 2 (Java Edition) [Electronic resource]. – Access mode: http://www.icodeon.com/release/download/j2ee/release_2p0p4/docs/.
32. Distributed Learning Environments. JISC CETIS Centre for educational technology & interoperability standards [Electronic resource]. – Access mode: http://publications.cetis.ac.uk/wp-content/uploads/2011/02/Distributed_Learning.pdf.
33. IMS Global Learning Consortium. Learning Tools Interoperability [Electronic resource]. – Access mode: <http://www.imsglobal.org/toolsinteroperability2.cfm>.
34. Learning Tools Interoperability [Electronic resource]. – Access mode: https://en.wikipedia.org/wiki/Learning_Tools_Interoperability.

35. Learning Tools Interoperability – Новий стандарт інтеграції для платформ дистанційного навчання [Електронний ресурс]. – Режим доступу: https://www.researchgate.net/publication/331467546_LEARNING_TOOLS_INTEROPERABILITY_NOVIJ_STANDART_INTEGRACII_DLA_PLATFORM_DISTANCIJNOGO_NAVCANNA.

36. Why Learning Tool Interoperability Should Be Part Of Your eLearning Application [Electronic resource]. – Access mode: <https://elearningindustry.com/learning-tool-interoperability-part-elearning-application>.

37. LTI [Electronic resource]. – Access mode: https://help.blackboard.com/Learn/Administrator/SaaS/Integrations/Learning_Tools_Interoperability.

38. IMS LTI® 1.3 and LTI Advantage [Electronic resource]. – Access mode: <http://www.imsglobal.org/activity/learning-tools-interoperability>.

39. Интеграция по протоколу LTI [Электронный ресурс]. – Режим доступа: <https://support.stepik.org/hc/ru/articles/360010307320-%D0%98%D0%BD%D1%82%D0%B5%D0%B3%D1%80%D0%B0%D1%86%D0%B8%D1%8F-%D0%BF%D0%BE-%D0%BF%D1%80%D0%BE%D1%82%D0%BE%D0%BA%D0%BE%D0%BB%D1%83>.

40. LTI and Moodle [Electronic resource]. – Access mode: https://docs.moodle.org/39/en/LTI_and_Moodle.

41. External tool [Electronic resource]. – Access mode: https://docs.moodle.org/39/en/External_tool.

42. PostgreSQL Documentation About [Electronic resource]. – Access mode: <https://www.postgresql.org/about/>.

43. PostgreSQL [Electronic resource]. – Access mode: <https://docs.moodle.org/310/en/PostgreSQL>.

44. Docker [Электронный ресурс]. – Режим доступа: <https://ru.wikipedia.org/wiki/Docker>.

45. Docker Documentation. docker docs [Electronic resource]. – Access mode: <https://docs.docker.com>.

46. Container Technologies Overview [Electronic resource]. – Access mode: <https://dzone.com/articles/container-technologies-overview>.

47. JavaScript [Електронний ресурс]. – Режим доступу: <https://ru.wikipedia.org/wiki/JavaScript>.

48. Node.js [Electronic resource]. – Access mode: <https://en.wikipedia.org/wiki/Node.js>.

49. Implement Learning Tools Interoperability modules for Moodle and related systems on Node.js [Electronic resource]. – Access mode: <https://developer.ibm.com/depmoels/cloud/tutorials/implement-learning-tool-interoperability/>.

50. Npm [Електронний ресурс]. – Режим доступу: <https://uk.wikipedia.org/wiki/Npm>.

51. TypeScript [Electronic resource]. – Access mode: <https://www.typescriptlang.org>.

52. Express.js [Electronic resource]. – Access mode: <https://en.wikipedia.org/wiki/Express.js>.

53. Методичні вказівки по економічному обґрунтуванню дипломних проектів для студентів спеціальностей 8.05010301 «Програмне забезпечення автоматизованих систем» та 8.05010201 «Інформаційні технології проектування» для усіх форм навчання /Укл.: Остапенко В.В., Сердюк Є.М.– Запоріжжя: ЗНТУ, 2015.– 22 с..

54. Вимоги щодо безпеки та захисту здоров'я працівників під час роботи з екранними пристроями. НПАОП 0.00-7.15-18. Державні санітарні правила і норми [Текст]. – [Чинний від 2018-02-14]. – Харків : Форт, 2018. – 48 с.

55. Електробезпека. Захисне заземлення, занулення: ГОСТ 12.1.030-81. – [Чинний від 1982-07-01]. – М.: Изд-во стандартов, 1982. – 10 с.

56. Правила охорони праці під час експлуатації електронно-обчислювальних машин : НПАОП 0.00-1.28-10. – [Чинний від 2010-03-26]. – К.: Державний комітет України з промислової безпеки, охорони праці та гірничого нагляду, 2010. – 8 с. – (Національний стандарт України).

57. Норми визначення категорій приміщень, будинків та зовнішніх установок по вибухопожежної та пожежної безпеки : НАПБ Б.03.002-2007. – [Чинний від 2007-12-03]. – К.: МНС України, 2007. – 9 с. – (Національний стандарт України).

58. ДСН 3.3.6.037-99 Санітарні норми виробничого шуму, ультразвуку та інфразвуку [Текст]. – [Чинний від 1999-12-01]. – К. : МОЗ України, 1999. – 24 с.

59. Гігієнічні вимоги до організації роботи з візуальними дисплейними терміналами електронно-обчислювальних машин: ДСанПіН 3.3.2.007-98. – К.: Держстандарт України, 1998. – 28 с. – (Державні санітарні правила і норми України).

60. ДБН В.2.5-28:2018. Природне та штучне освітлення [Текст]. – [Чинний від 2019-03-01]. – К. : Мінрегіон України, 2019. – 137 с.

61. ДСН 3.3.6.037-99 Санітарні норми виробничого шуму, ультразвуку та інфразвуку [Текст]. – [Чинний від 1999-12-01]. – К. : МОЗ України, 1999. – 24 с.

62. ДСН 3.3.6.042-99 Санітарні норми мікроклімату виробничих приміщень [Текст]. – [Чинний від 1999-12-01]. – Харків : Форт, 2000. – 24 с.

63. ГОСТ 17.2.032-78 Рабочее место при выполнении работ сидя. Система стандартов безопасности труда [Текст]. – [Введен с 1979-01-01]. – М. : ИПК Издательство стандартов, 2001. – 7 с.

64. Визначення категорій приміщень, будинків та зовнішніх установок за вибухопожежною та пожежною небезпекою : ДСТУ Б В.1.1-36:2016. – [Чинний від 2017-01-01]. – К.: Міненергобуд України, 2017. – 86 с. – (Національний стандарт України).

65. ДБН В.1.1-7:2016 Пожежна безпека об'єктів будівництва. Загальні вимоги [Текст]. – [Чинний від 2017-01-31]. – Київ : Міністерство регіонального розвитку, будівництва та житлово-комунального господарства України, 2017. – 39 с.

66. Протипожежна техніка. Терміни та визначення основних понять: ДСТУ 2273 – 2006. – К.: Держстандарт України, 2006. – 44 с. – (Національний стандарт України).

67. Типові норми належності вогнегасників: НАПБ Б.03.001-2004. – 2004. – К.: Держстандарт України, 2004. – 15 с. – (Національний стандарт України).

68. Захист від пожежі. Пожежна безпека об'єктів будівництва: ДБН В.1.1.7-2002. – К.: Держстандарт України, 2002. – 39 с. – (Державні будівельні норми).

69. Основи організації й проведення аварійно-рятувальних та інших невідкладних робіт під час ліквідації надзвичайної ситуації [Електронний ресурс]. – Режим доступу: <https://ns-plus.com.ua/2019/06/10/osnovy-organizatsiyi-j-provedennya-avarijno-yatuvalnyh-ta-inshyh-nevidkladnyh-robit-pid-chas-likvidatsiyi-nadzvychnoyi-sytuatsiyi/>.

70. Стаття 79. Проведення аварійно-рятувальних та інших невідкладних робіт [Електронний ресурс]. – Режим доступу: https://protocol.ua/ua/kodeks_tsivilnogo_zahistu_uk_vid_02_10_2012_5403_vi_statty_79/.

ДОДАТОК А

Текст програми

```
// main.ts

import { default as bodyParser } from "body-parser";
import { default as express } from "express";
import { default as exphbs } from "express-handlebars";
import { default as session } from "express-session";

import { applicationConfig } from "@/configs/ApplicationConfig";
import { router as feedbackRouter } from "@/routes/feedback";
import { router as ltiRouter } from "@/routes/lti";
import { router as participationRouter } from "@/routes/participation";
import { router as viewsRouter } from "@/routes/views";

const application = express();

/**
 * Configure express application.
 */
application.set("views", applicationConfig.VIEWS_DIR);
application.set("view engine", "handlebars");
application.engine("handlebars", exphbs({ layoutsDir: applicationConfig.LAYOUTS_DIR }));

application.use(session({ secret: process.env.SESSION_SECRET || "dev", resave: false,
saveUninitialized: true }));
application.use(bodyParser.urlencoded({ extended: false }));

application.enable("trust proxy");

/**
 * Configure application routes matching.
 */
application.use("/", ltiRouter);
application.use("/", viewsRouter);
application.use("/", participationRouter);
application.use("/", feedbackRouter);

/**
 * Start listening after application configuration.
 */
application.listen(
  applicationConfig.PORT,
  () => console.log(`Example app listening on port ${applicationConfig.PORT}!`)
)
```

```

);

// main.handlebars

<html lang="en">
  <head>
    <title> Diploma 2020 LTI example </title>

    <link rel="stylesheet"
href="https://maxcdn.bootstrapcdn.com/bootstrap/4.3.1/css/bootstrap.min.css"/>
  </head>

  <body>
    {{{body}}}
  </body>
</html>

// student-feedback.handlebars

<style>
  header {
    padding: 8px;
    width: 100%;
    border-bottom: 1px solid lightgray;
  }

  main {
    padding: 8px;
  }

  #feedback-form {
    padding: 4px;
    display: flex;
    flex-direction: column;
    align-items: flex-start;
  }

  #feedback-button {
    margin-top: 8px;
  }

  #feedback-report {
    padding: 8px;
  }

  .alert {
    margin-top: 8px;
  }
</style>

<main>
  {{#if hasFeedback }}
    <div class="alert alert-success">
      Thank you for your feedback! It is shared anonymously and
visible only for tutor.
    </div>

    <h5> Your feedback: </h5>

    <div id="feedback-report" class="card">
      {{{ feedback.text }}}
    </div>

    <form action="/remove_feedback" method="POST">
      <button id="feedback-button" class="btn btn-danger"
type="submit"> Dismiss </button>
    </form>

  {{else}}
    <form id="feedback-form" method="POST" action="/feedback">

```

```

        <div class="alert alert-secondary">
            You can leave feedback about course anonymously
        </div>

        <textarea class="form-control" name="feedback"></textarea>

        <button id="feedback-button" class="btn btn-primary"
type="submit"> Submit </button>
        </form>

        {{/if}}
    </main>

    // tutor-feedback-report.handlebars

<style>
    main {
        padding: 8px;
    }

    table {
        padding: 8px;
        border: 1px solid lightgray;
    }

    td {
        border: 1px solid lightgray;
        padding: 4px;
    }

    li {
        margin: 4px 0;
    }
</style>

<main>
    {{#if feedbacks.length }}
        <div class="alert alert-secondary">
            Anonymous reports for this course ({{feedbacks.length}}):
        </div>

        <ul class="list-group">
            {{#each feedbacks}}
                <li class="list-group-item">
                    {{this.text}}
                </li>
            {{/each}}
        </ul>
    {{else}}
        <div class="alert alert-secondary">
            Currently there is no feedback provided for this course.
        </div>
    {{/if}}
</main>

// store: feedback.ts

// In-memory storage instead of database.

export interface ICourseFeedback {
    text: string;
    userId: string;
    date: Date;
}

export const feedbacks: Map<string, ICourseFeedback> = new Map();

// routes: feedback.ts

import { default as express } from "express";

```

```

import { feedbacks } from "@store/feedback";

export const router = express.Router();

router.post("/feedback", (req, res, next) => {
  const session: { [index: string]: any } = req.session;
  const userId: string | undefined = session.userId;
  const feedback: string = req?.body["feedback"];

  if (!req.body) {
    return next(new Error("Expected a body")); // todo: Status for proper 403.
  }

  if (userId) {
    feedbacks.set(userId, { userId, date: new Date(), text: feedback }); // todo:
    Validation.

    // Redirect to actual handling of LTI after POST handling.
    res.redirect("/application?type=feedback");
  } else {
    throw new Error("Cannot participate, user is not logged in");
  }
});

router.post("/remove_feedback", (req, res, next) => {
  const session: { [index: string]: any } = req.session;
  const userId: string | undefined = session.userId;

  if (userId) {
    feedbacks.delete(userId); // todo: Validation.

    // Redirect to actual handling of LTI after POST handling.
    res.redirect("/application?type=feedback");
  } else {
    throw new Error("Cannot participate, user is not logged in");
  }
});

// lti.ts

import { default as express } from "express";

import { ltiConfig } from "@configs/LtiConfig";

// eslint-disable-next-line @typescript-eslint/no-var-requires
const lti = require("ims-lti");

const nonceStore = new lti.Stores.MemoryStore();

export const router = express.Router();

function getConsumerSecret(consumerKey: string): string {
  if (ltiConfig.SECRETS[consumerKey]) {
    return ltiConfig.SECRETS[consumerKey];
  } else {
    throw new Error("Unknown consumer key provided");
  }
}

/**
 * Handle LTI launch from external consumers.
 */
router.post("/launch_lti", (req, res, next) => {
  const consumerKey: string = req?.body["oauth_consumer_key"];
  const type: string = req.query.type as string;

  if (!req.body) {
    return next(new Error("Expected a body")); // todo: Status for proper 403.
  } else if (!consumerKey) {
    return next(new Error("Expected a consumer"));
  }
});

```

```

    }

    if (!type) {
      return next(new Error("Expected application type from query"));
    }

    try {
      const consumerSecret: string = getConsumerSecret(consumerKey);
      const provider = new lti.Provider(consumerKey, consumerSecret, nonceStore,
        lti.HMAC_SHA1);

      provider.valid_request(req, (error: Error, isValid: boolean) => {
        if (error) {
          return next(error);
        }

        if (isValid) {
          req.session.regenerate((err) => {
            if (err) {
              next(err);
            }

            const session: { [index: string]: any } = req.session;

            session.email = provider.body.lis_person_contact_email_primary;
            session.contextId = provider.context_id;
            session.userId = provider.userId;
            session.username = provider.username;
            session.ltiConsumer = provider.body.tool_consumer_instance_guid;
            session.isTutor = provider.instructor === true;

            return res.redirect(301, `/application?type=${type}`);
          });
        } else {
          // todo: Exception? When fires?
        }
      });
    } catch (error) {
      next(error);
    }
  });
}

// views.ts

import { default as express, NextFunction, Request, Response } from "express";

import { feedbacks, ICourseFeedback } from "@store/feedback";
import { participants } from "@store/participation";

export const router = express.Router();

function renderFeedbackApplication(req: Request, res: Response, next: NextFunction) {
  const session: { [index: string]: any } = req.session;

  if (session.userId) {
    const feedback = feedbacks.get(session.userId);

    const renderParams: { [index: string]: any } = {
      email: session.email,
      username: session.username,
      ltiConsumer: session.ltiConsumer,
      userId: session.userId,
      isTutor: session.isTutor,
      contextId: session.contextId,
      hasFeedback: Boolean(feedback),
      feedbacks: Array.from(feedbacks.values()),
      feedback: feedback
    };

    if (renderParams.isTutor) {
      return res.render("tutor-feedback-report", renderParams);
    }
  }
}

```

```

    } else {
      return res.render("student-feedback", renderParams);
    }
  } else {
    next(new Error("Session invalid. Please login via LTI to use this application."));
  }
}

function renderParticipationApplication(req: Request, res: Response, next:
NextFunction) {
  const session: { [index: string]: any } = req.session;

  if (session.userId) {
    const renderParams: { [index: string]: any } = {
      email: session.email,
      username: session.username,
      ltiConsumer: session.ltiConsumer,
      userId: session.userId,
      isTutor: session.isTutor,
      contextId: session.contextId,
      hasParticipated: participants.some((it: { [index: string]: any }) => it.userId
=== session.userId),
      participants: participants
    };

    if (renderParams.isTutor) {
      return res.render("tutor-participation-report", renderParams);
    } else {
      return res.render("student-participation", renderParams);
    }
  } else {
    next(new Error("Session invalid. Please login via LTI to use this application."));
  }
}

/**
 * Generic route for service status checking.
 */
router.get("/", (req, res, next) => {
  return res.send({ status: "Up" });
});

/**
 * Generic route for service status checking.
 */
router.get("/application", (req, res, next) => {
  const type: string = req.query.type as string;

  if (!type) {
    return next(new Error("Expected application type to be provided"));
  }

  if ((req?.session as any)?.userId) {
    switch (type) {
      case "participation":
        return renderParticipationApplication(req, res, next);

      case "feedback":
        return renderFeedbackApplication(req, res, next);

      default:
        return next(new Error("Expected type to be provided"));
    }
  } else {
    next(new Error("Session invalid. Please login via LTI to use this application."));
  }
});

```

ДОДАТОК Б
Слайди презентації

Національний Університет «Запорізька Політехніка»
Кафедра програмних засобів

Магістерська робота на тему

**«Дослідження та розробка засобів
взаємодії інструментів онлайн
навчання для реалізації міжнародних
навчальних проектів»**



Виконала
студентка групи КНТ-119м

А.К. Чадаєва

Керівник
к.т.н., доцент

О.В. Шитікова

Рисунок Б.1 – Слайд №1



Мета роботи та поставлені задачі

Мета роботи – огляд та розробка засобів розширення функціональних можливостей систем дистанційного навчання на прикладі інтеграції плагіну зворотнього зв'язку до системи дистанційного навчання Moodle з метою надання студентами об'єктивного анонімного відгуку про курс.

Об'єкт дослідження – засоби взаємодії інструментів онлайн навчання.

Предмет дослідження – інтегровані програмні засоби до систем дистанційного навчання для розширення функціональних можливостей.

Рисунок Б.2 – Слайд №2



Задачі роботи

- виконати дослідження систем дистанційного навчання та дослідження засобів взаємодії інструментів онлайн навчання;
- виконати дослідження засобів впровадження сторонніх модулів у систему дистанційного навчання Moodle;
- розробити модуль зворотнього зв'язку та виконати інтеграцію у систему дистанційного навчання Moodle;
- виконати експериментальне дослідження розробленої системи.

3

Рисунок Б.3 – Слайд №3



Поняття системи дистанційного навчання

Система дистанційного навчання – система управління навчальною діяльністю, яка використовується для розробки, управління та поширення навчальних онлайн-матеріалів із забезпеченням спільного доступу.

До складу системи входять такі основні критерії:

- вибір навчального курсу;
- заняття зі студентами;
- індивідуальні завдання;
- проекти для роботи в малих групах;
- здача завдань, проміжних і фінальних онлайн іспитів та тестувань.

СДН – Система дистанційного навчання.

4

Рисунок Б.4 – Слайд №4



Дослідження існуючих СДН

Критерій порівняння	Назва СДН			
	ILIAS	Sakai	Docebo	Moodle
Авторизація користувача	+	+	+	+
Безкоштовна	+	+	-	+
Обмеження безкоштовної версії	-	+	+	-
Багатомовний інтерфейс	+	+	+	+
Ліцензія	GNU GPL	GNU GPL	GNU GPL	GNU GPL
Підтримка української мови	+	+	+	+
Підтримка SCORM	+	+	+	+
Підтримка IMS	+	+	+	+
Підтримка LTI	+	+	+	+

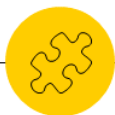
5

Рисунок Б.5 – Слайд №5

Критерій порівняння	Назва СДН			
	ILIAS	Sakai	Docebo	Moodle
Структура	Ядро+набір модулів	Ядро+набір модулів	Ядро+набір модулів	Ядро+набір модулів
Можливість інтеграції доповнень	+	+	+	+
Гнучкість платформи	+	+	+	+
Багатоплатформність	+	+	+	+
Середа розробки учбового матеріалу	+	+	+	+
Система перевірки знань	+	+	+	+
Система зворотного зв'язку	+	+	+	+
Керування якістю навчання	+	+	+	+
Обмеження на кількість слухачів	-	-	-	-

6

Рисунок Б.6 – Слайд №6



Підходи щодо розширення функціональних можливостей

- розробка внутрішніх модулів системи та інтегрованих у систему безпосередньо;
- розробка зовнішніх незалежних застосунків, які інтегруються з системою.

Внутрішні плагіни

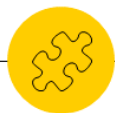
- типи діяльностей (включаючи можливі навчальні ігри);
- типи ресурсів;
- типи тестових питань;
- типи полів для бази даних;
- теми для оформлення;
- різні методи автентифікації;
- різні методи зарахування на курс;
- фільтри для контенту.

Сторонні застосунки

- сервіси для проведення відеоконференцій та звукозапису;
- сервіси для перевірки програмного коду;
- віртуальні лабораторії.

7

Рисунок Б.7 – Слайд №7



Типи плагінів та сторонніх застосунків Moodle

Внутрішні плагіни

- **Database** – база даних;
- **Workshop** – лабораторія;
- **Quiz** – інструмент для опитувань;
- **Book** – надає інформацію про книги;
- **Forum** – спілкування;
- **Feedback** – модуль зворотного зв'язку для проведення досліджень і опитувань.

Сторонні застосунки

- **BigBlueButton** – сервіс для проведення відеоконференцій;
- **Zoom** – сервіс для проведення відеоконференцій;
- **GoToWebinar** – сервіс для проведення відеоконференцій;
- **Virtual program Lab** – інструмент перевірки програмного коду;
- **VmWare Learning Platform** – платформа, яка надає практичні віртуальні лабораторії.

8

Рисунок Б.8 – Слайд №8



Огляд інструментів сумісності з навчальними середовищами

- Sharable Content Object Reference Model (SCORM)
- Learning Tools Interoperability (LTI)
- External tools

9

Рисунок Б.9 – Слайд №9



Інтегровані застосунки та призначення створеної програми

До СДН Moodle було інтегровано внутрішній плагін та сторонні застосунки:

- BigBlueButton
- Virtual program Lab

Розроблений модуль може бути інтегрований керівником навчального курсу у будь-який розділ.

Створений модуль має функцію анонімного відгуку про курс.

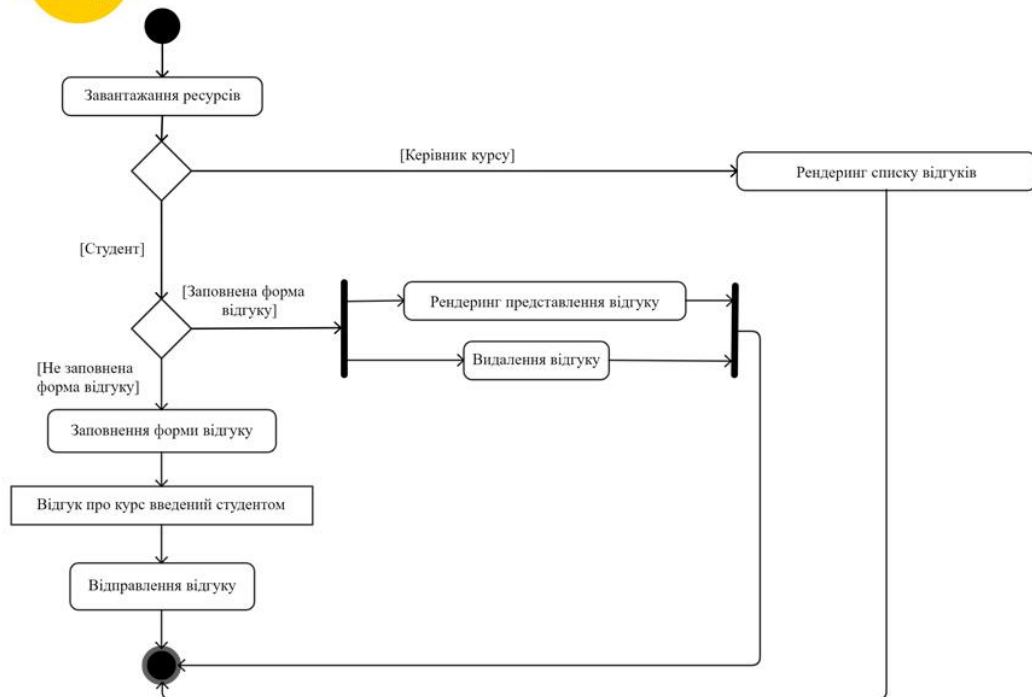
Цільовою аудиторією є учні, студенти, працівники компаній, які мають різний рівень знань, які можуть об'єктивно оцінити корисність курсу і надати повний відгук.

10

Рисунок Б.10 – Слайд №10



Схема функціонування застосунку

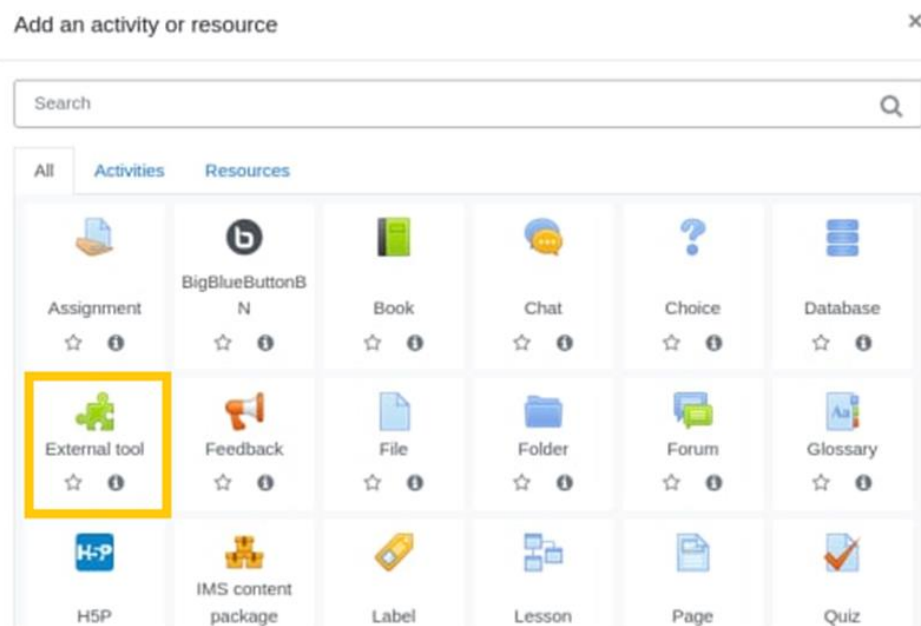


11

Рисунок Б.11 – Слайд №11



Додавання модулю до системи Moodle



12

Рисунок Б.12 – Слайд №12



Додавання модулю до системи Moodle

Preconfigured tool

Automatic, based on tool URL + ×

Select content

Tool URL ✓ Using custom tool configuration.

Secure tool URL

Launch container

Consumer key

Shared secret

Press enter to save changes

+ Some course topic

+ Anonymous feedback

13

Рисунок Б.13 – Слайд №13



Використання модулю

> My courses > ExampleCourse > Some course topic > Anonymous feedback

Anonymous feedback

You can leave feedback about course anonymously

Submit

14

Рисунок Б.14 – Слайд №14



Використання модулю

Example course 2020

Home Dashboard Events My Courses This course

My courses > ExampleCourse > Some course topic > Anonymous feedback

Anonymous feedback

Thank you for your feedback! It is shared anonymously and visible only for tutor.

Your feedback:

Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat. Duis aute irure dolor in reprehenderit in voluptate velit esse cillum dolore eu fugiat nulla pariatur. Excepteur sint occaecat cupidatat non proident, sunt in culpa qui officia deserunt mollit anim id est laborum. Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat. Duis aute irure dolor in reprehenderit in voluptate velit esse cillum dolore eu fugiat nulla pariatur. Excepteur sint occaecat cupidatat non proident, sunt in culpa qui officia deserunt mollit anim id est laborum. Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua.

Dismiss

15

Рисунок Б.15 – Слайд №15



Інтерфейс модулю

TLTI

Example course 2020

Dashboard / My courses / ExampleCourse / Some course topic / Anonymous feedback

Anonymous feedback

Currently there is no feedback provided for this course.

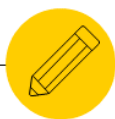
Anonymous feedback

Anonymous reports for this course (1):

Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Enim eu turpis egestas pretium aenean pharetra magna ac placerat. Rhoncus aenean vel elit scelerisque mauris pellentesque pulvinar pellentesque. In aliquam sem fringilla ut morbi tincidunt augue interdum velit. Erim praesent elementum facilisis leo vel fringilla est ullamcorper. Amet risus nullam eget felis. Aliquet bibendum enim facilisis gravida neque convallis a cras. Consequat semper viverra nam libero justo laoreet sit amet cursus. Pretium quam vulputate dignissim suspendisse in est ante in nibh. Elementum eu facilisis sed odio morbi quis commodo odio. Ut sem nulla pharetra diam sit amet nisl. Ultricies mi quis hendrerit dolor magna eget est. Arcu non odio euismod lacinia at quis risus. Diam vulputate ut pharetra sit amet. Aenean pharetra magna ac placerat vestibulum lectus. Laoreet sit amet cursus sit amet dictum. Utna dui convallis convallis tellus. Id consectetur purus ut faucibus pulvinar. Eleifend quam adipiscing vitae proin sagittis nisl rhoncus mattis. Ultrices neque ornare aenean euismod. Integer malesuada nunc vel risus commodo viverra. Nisi scelerisque eu ultrices vitae auctor. Nibh sed pulvinar proin gravida. Pulvinar elementum integer enim neque. Orci porta non pulvinar neque laoreet.

16

Рисунок Б.16 – Слайд №16

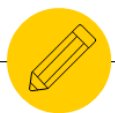


Висновки

- Було розглянуто та порівняно системи дистанційного навчання ILIAS, Sakai, Docebo та Moodle, що надають схожі функції та засоби організації навчальних процесів.
- Було проведено огляд необхідних ключових можливостей для будь-якої системи, так і додаткових – плагінів і сторонніх застосунків, що інтегруються до систем.
- Було наведено огляд засобів сумісності інструментів онлайн навчання з навчальними системами: набір стандартів Sharable Content Object Reference Model (SCORM), Learning Tools Interoperability (LTI), а також інструменти External tools.

17

Рисунок Б.17 – Слайд №17



Висновки

- Інтегровано до СДН Moodle такі сторонні застосунки: BigBlueButtonBN і Virtual Program Lab.
- Розроблено застосунок, що інтегрується до СДН Moodle за LTI протоколом і являє собою просту форму анонімного зворотнього зв'язку.
- За результатами роботи було опубліковано тези, що були подані на X Міжнародну науково-практичну конференцію «Сучасні проблеми і досягнення в галузі радіотехніки, телекомунікацій та інформаційних технологій» на тему «Дослідження засобів діджиталізації для систем дистанційного навчання».
- Дослідження виконано в рамках міжнародного проекту Virtual Master Cooperation Data Science (ViMaCs) за підтримки DAAD.

18

Рисунок Б.18 – Слайд №18



**Дякую
за увагу!**