

Міністерство освіти і науки України
Державний заклад
«Луганський національний університет імені Тараса Шевченка»

Навчально-науковий інститут математики та інформаційних технологій

Кафедра інформаційних технологій та систем

Кузнецов Віталій Сергійович

Створення плагіну інтегрування Google Meet в навчальну платформу LMS Moodle та дослідження його ефективності

кваліфікаційна робота

здобувача вищої освіти другого (магістерського) рівня освітньої програми

«Комп'ютерні мережі»

за спеціальністю 123 Комп'ютерна інженерія

Особистий підпис _____ Віталій КУЗНЕЦОВ

Науковий керівник _____ Микола СЕМЕНОВ,
кандидат педагогічних наук, доцент
кафедри інформаційних технологій
та систем

Завідувач кафедри _____ Микола СЕМЕНОВ,
кандидат педагогічних наук, доцент
кафедри інформаційних технологій
та систем

АНОТАЦІЯ

Кузнецов В.С.

Тема: Створення плагіну інтегрування Google Meet в навчальну платформу LMS Moodle та дослідження його ефективності.

Спеціальність: 123 «Комп'ютерна інженерія».

Установа: ЛНУ імені Тараса Шевченка, 2023р.

Магістерська робота містить: (59 с., 18 рис., 13 джерел).

Об'єкт дослідження – інтеграція Google Meet в навчальну платформу LMS Moodle та оцінка ефективності цього плагіну для поліпшення процесу навчання та спілкування студентів та викладачів.

Предмет дослідження – створення та оцінка ефективності плагіну для інтеграції Google Meet в навчальну платформу LMS Moodle.

Мета роботи - в розробці та інтеграції плагіну Google Meet в навчальну платформу LMS Moodle з метою поліпшення можливостей віддаленої освіти та сприяння активній комунікації між викладачами та студентами.

Результати роботи – створення функціонального плагіну для інтеграції Google Meet з Moodle на мові PHP за допомогою PHP Storm, Open Server та Google Cloud Console. Плагін повинен дозволяти користувачам легко створювати, приєднуватися та управляти відеоконференціями через Google Meet безпосередньо з Moodle завдяки успішному інтегруванню Google аккаунту в навчальне середовище.

Ключові слова: ПЛАГІН, LMS MOODLE, PHP, PHP STORM, OPEN SERVER, GOOGLE CLOUD CONSOLE, GOOGLE MEET.

ANNOTATION

Kuznetsov Vitalii

Theme: Creating a plugin for integrating Google Meet into the LMS Moodle training platform and researching its effectiveness.

Speciality: 123 "Computer Engineering".

Institution: Luhansk Taras Shevchenko National University (LTSNU), 2023 year.

Master's work of: 58 p., 18 im, 13 sources.

A research object of: - is the integration of Google Meet into the LMS Moodle learning platform and the assessment of the effectiveness of this plugin to improve the learning process and communication between students and teachers.

The article of research- is the creation and evaluation of the effectiveness of the plugin for integrating Google Meet into the LMS Moodle training platform.

An aim of research is - develop and integrate the Google Meet plugin into the LMS Moodle learning platform in order to improve remote education opportunities and promote active communication between teachers and students.

Job performanes.- the creation of a functional plugin for integrating Google Meet with Moodle in PHP using PHP Storm, Open Server and Google Cloud Console. The plugin should allow users to easily create, join and manage video conferences via Google Meet directly from Moodle thanks to the successful integration of the Google account into the learning environment.

Keywords: PLUGIN, LMS MOODLE, PHP, PHP STORM, OPEN SERVER, GOOGLE CLOUD CONSOLE, GOOGLE MEET.

ЗМІСТ

ВСТУП.....	5
РОЗДІЛ 1. ВВЕДЕННЯ В PHP, LMS MOODLE, OPENSERVR TA GOOGLE MEET	8
1.1 Загальні відомості про мову програмування PHP.....	8
1.1.2 Сфери застосування PHP	8
1.1.3 Ключові переваги	9
1.1.4 Базові теоретичні знання про LMS Moodle.....	10
1.2 Відомості про OpenServer.....	12
1.2.1 Переваги OpenServer	13
1.3 Загальні відомості про Google Meet	14
1.3.1 Які особливості google meet?	14
1.3.2. Історія	15
РОЗДІЛ 2. ПРОГРАМУВАННЯ ТА НАЛАШТУВАННЯ ПЛАГІНУ, НАЛАШТУВАННЯ OPENSERVR.....	16
2.1 Налаштування OpenServer	16
2.2 Налаштування PhpStorm	19
2.3 Програмування плагіну	20
2.3.1 Розбір основних елементів коду. Setting.php.....	20
2.3.2. Розбір основних елементів коду. Index.php.....	21
2.4 Висновки до розділу	24
РОЗДІЛ 3. ІНТЕГРУВАННЯ ПЛАГІНУ ТА ЙОГО НАЛАШТУВАННЯ НА ПЛАТФОРМІ LMS MOODLE.....	26
3.1 Завантаження плагіну на платформу LMS Moodle	26
3.2 Налаштування Google OAuth service	29
3.2.1 Що таке OAuth 2.....	29
3.2.2 Різниця між OpenID і OAuth	29
3.2.3 Як працює OAuth 2	29
3.2.4 Недоліки OAuth 2	32
3.4 Налаштування OAuth Service під нашу платформу.....	32
3.4.1 Get ID та secret key	34
3.4.2 Перевірка коректної роботи плагіну інтегрування Google Meet.....	36
3.5 Висновки до розділу	37
ВИСНОВКИ	38
СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ	39
ДОДАТОК А.....	40
ДОДАТОК Б.....	43
ДОДАТОК В.....	47

ВСТУП

Moodle (акронім від Modular Object-Oriented Dynamic Learning Environment) — це освітня платформа, призначена для об'єднання викладачів, адміністраторів і студентів в інтегровану, надійну та безпечну систему для створення персоналізованого освітнього середовища.

Moodle — це безкоштовна відкрита (з відкритим кодом) система управління навчанням. Він реалізує філософію «соціальної конструкціоністської педагогіки» та зосереджений насамперед на організації взаємодії між викладачами та студентами, але не на традиційній організації курсу дистанційного навчання чи підтримці очного навчання.

За допомогою LMS Moodle ви можете створювати курси, завантажувати навчальні матеріали (тексти, відео, аудіо тощо), розподіляти їх серед студентів, проводити тестування та оцінювання, спілкуватися зі студентами через форуми та приватні повідомлення, відстежувати прогрес студентів тощо.

LMS Moodle є популярною платформою в освітніх установах, вищих навчальних закладах та компаніях, які надають навчальні курси або тренінги. Вона надає зручні інструменти для ефективного навчання та спілкування у віртуальному середовищі.

Плагіни Мудл – це розширення для LMS під різні формати контенту: інтерактивні курси, мікромодулі і навіть таймлайни постів у соцмережах, які дозволяють організувати чіткий і зрозумілий показ певних процедур, процесів або взаємозв'язків між різними компонентами навчального матеріалу.

За допомогою плагінів можна розширити функціональні можливості платформи. Більшість плагінів безкоштовні. Але можливе й індивідуальне доопрацювання плагінів під конкретні завдання. Завдяки великій кількості готових плагінів та можливості розробки нових розширюється і потенціал системи управління навчанням.

Також встановлення ряду плагінів Moodle допомагає змінити стандартний інтерфейс платформи під корпоративний дизайн. Можна провести брендування головної сторінки платформи, особистого кабінету користувача, сторінок бібліотеки та каталогу курсів тощо. За умови співпраці з Міжнародним інжиніринговим центром SENSYS такі роботи доступні в рамках розширеного пакету в рамках послуги впровадження LMS Moodle.

Об’єкт дослідження – інтеграція Google Meet в навчальну платформу LMS Moodle та оцінка ефективності цього плагіну для поліпшення процесу навчання та спілкування студентів та викладачів.

Предмет дослідження – створення та оцінка ефективності плагіну для інтеграції Google Meet в навчальну платформу LMS Moodle.

Мета роботи - в розробці та інтеграції плагіну Google Meet в навчальну платформу LMS Moodle з метою поліпшення можливостей віддаленої освіти та сприяння активній комунікації між викладачами та студентами.

Для досягнення поставленої мети необхідно вирішити наступні завдання:

- 1) проаналізувати практичне застосування плагіну та коректність його роботи.
- 2) проаналізувати навчальну платформу LMS Moodle чи сприймає платформа даний модуль.
- 3) розробити плагін для того щоб покращити роботу викладачів та студентів, встановити та налаштувати всі додатки.

Новизна отриманих результатів:

При коректній роботі плагіну буде можливість авторизуватись за допомогою Google, створення посилання Google Meet одразу всередині платформі, не запобігаючи самого Google Meet, а також запис відео на Google Drive та створення автоматичного посилання на зустріч.

Методи дослідження. Поставлені завдання вирішені методами програмування, аналізу платформи та інтегрування.

У першому розділі розглянемо теорію, що таке PHP, платформа LMS Moodle, OpenServer та Google Meet.

У другому розділі розглянуто програмування та налаштування плагіну, а також налаштування Open Server.

У третьому розділі розглянуто інтегрування плагіну та його налаштування на платформі LMS Moodle.

РОЗДІЛ 1. ВВЕДЕННЯ В PHP, LMS MOODLE, OPENSERVR TA GOOGLE MEET

1.1 Загальні відомості про мову програмування PHP

PHP - це популярна мова програмування, особливо серед веб-розробників. Автором початкової версії є Расмус Лердорф, ідея якого полягала у розробці набору інструментів спрощення процесу створення динамічних веб-сторінок. Незважаючи на те, що сучасний PHP є мовою загального призначення, найчастіше його використовують як серверний інструмент для генерації HTML-коду, який потім інтерпретується веб-браузером.

Що таке PHP? Це мова програмування з відкритими вихідними кодами, над розвитком якого працюють програмісти-ентузіасти з усього світу. Він має простий синтаксис, частково схожий на Java та C++. Це проект, що постійно розвивається, на даний момент актуальною є 7-ма версія мови. За статистикою, кожен шостий програмний продукт створено на PHP.

1.1.2 Сфери застосування PHP

Сьогодні виділяють три основні області використання PHP:

- 1) Для написання скриптів та повноцінних веб-застосунків, що виконуються на серверній стороні. Це найпопулярніша сфера застосування, оскільки мова спочатку створювалася саме для веб-розробок. Для повноцінної роботи веб-програми, написаної на PHP, необхідні сервер, парсер (CGI-додаток) і клієнтське програмне забезпечення (веб-браузер), яке відображає результат виконання коду.
- 2) Для створення сценаріїв, які запускаються в командному рядку. Такі маленькі програми можуть працювати на будь-якому ПК. Для їх запуску потрібен лише парсер. Оскільки PHP містить потужні інструменти для роботи з рядками, ці сценарії часто створюють для обробки текстових даних.
- 3) Написати графічні інтерфейси. PHP має багато гілок, створених для виконання різних завдань. Однією з таких гілок є PHP-GTK. Його часто використовують програмісти, які знайомі з синтаксисом PHP.

1.1.3 Ключові переваги

- 1) **Простий та інтуїтивно зрозумілий синтаксис.** PHP швидко освоюють навіть програмісти-новачки. Він увібрав усі найкращі особливості таких популярних мов, як C, Java та Perl. PHP-код легко читається незалежно від способу використання (для написання невеликих скриптів або створення потужних додатків з використанням об'єктно-орієнтованого підходу до реалізації програми).
- 2) **Кросплатформність та гнучкість.** PHP сумісний із усіма популярними платформами (Linux, Windows, MacOS). Написані на ньому програми успішно працюють на різному серверному ПЗ (IIS, Nginx, Apache та багатьох інших).
- 3) **Відмінна масштабованість.** PHP дозволяє досягти максимальної продуктивності додатків, написаних на ньому, зі зростанням апаратних ресурсів. Веб-програми, розподілені на кілька серверів, здатні справлятися із суттєвими навантаженнями (великим трафіком).
- 4) **Вбудовуваність у HTML-документи.** На просту HTML-сторінку можна легко додати контент, що динамічно змінюється шляхом вставки блоків коду PHP.
- 5) **Активний розвиток та вдосконалення.** Спільнота розробників постійно працює над впровадженням додаткового функціоналу, що розширює можливості мови, спрощенням синтаксису та покращенням захисту від можливих атак.
- 6) **Детальна документація.** На офіційному сайті проекту представлені повні відомості про кожну функціональну одиницю мови з прикладами використання.
- 7) **Простий пошук рішень проблем, що виникають.** В інтернеті існує безліч форумів, присвячених програмуванню на PHP.
- 8) **Широкі перспективи її подальшого розвитку.** Більшість CMS були створені на чистому PHP та фреймворках. Цим зумовлені популярність та затребуваність PHP програмістів.

1.1.4 Базові теоретичні знання про LMS Moodle

Moodle – це безкоштовна онлайн-платформа навчання, яка дозволяє створювати онлайн-курси та надавати студентам доступ до них через глобальну мережу Інтернет, тим самим організовуючи навчання за принципом «будь-коли та будь-де».

Крім того, система Moodle є найпопулярнішою платформою для організації онлайн-навчання.

Орієнтуючись переважно на організацію взаємодії між викладачами та студентами, він використовується для підтримки відповідних форм очного навчання та організації офіційного онлайн-навчання.

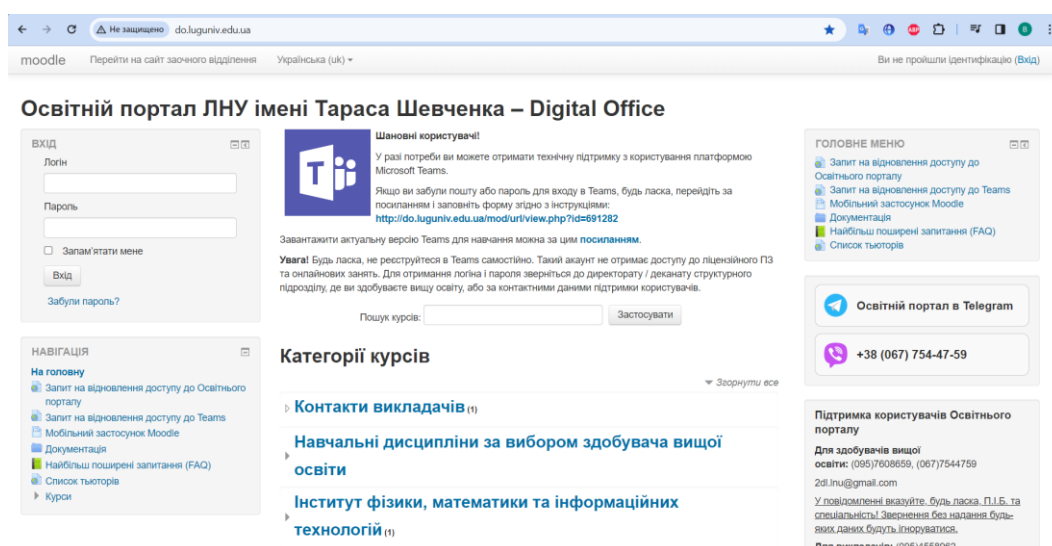


Рис 1.1 Освітній портал нашого університету

Використовуючи Moodle, викладачі можуть надавати доступ до навчальних матеріалів (методичних рекомендацій, конспектів лекцій, питань для самоконтролю, лабораторних та практичних робіт, презентацій та ін.), надсилати повідомлення студентам, розподіляти, збирати й перевіряти завдання, вести електронні журнали оцінок тощо.

Аби полегшити контакт між викладачем і студентом, існує низка так званих ресурсів: форуми, чати, опитування, тестування, блоги та ін. Робота студента з ними може бути оцінена викладачем вручну чи автоматично комп'ютером.

Тести в системі Moodle можна виконувати як під час пари, так і вдома у вільний від навчання час.

Навчальна інформація та названі інтерактивні ресурси в Moodle організовані у вигляді курсів, для яких, окрім автора, може бути призначено менеджера (редактора, який наповнює курс навчальною інформацією) та асистента (вони допоможуть організовувати процес контролю знань).

Поступово підготовку можна трансформувати і розвинути в унікальний інноваційний підручник, який постійно вдосконалюється і доповнюється працею учнів.

Робота в такій системі дуже інтуїтивно зрозуміла для кожного користувача - заповнення документів і доступ до них не вимагає спеціальних знань в області інформаційних технологій.

Moodle також вирізняється своєю великою спільнотою користувачів та розробників, яка активно співпрацює для вдосконалення системи. Це означає, що користувачі можуть розраховувати на широкий спектр рішень, плагінів та допомоги від експертів у випадку виникнення питань чи потреби у розширенні функціоналу.

Однією з ключових переваг є можливість інтеграції Moodle з іншими навчальними інструментами та системами. Це робить платформу гнучкою та пристосованою до конкретних потреб освітніх установ та організацій.

Крім того, Moodle активно розвивається і оновлюється, враховуючи сучасні тенденції у галузі освіти та технологій. Використання останніх версій платформи дозволяє користувачам отримувати доступ до новітніх функцій та забезпечує стабільну роботу системи.

Завдяки відкритому вихідному коду, Moodle дозволяє користувачам змінювати та пристосовувати код платформи під свої потреби. Це надає додаткову свободу у роботі з системою та створенні унікальних рішень для конкретного освітнього контексту.

1.2 Відомості про OpenServer

OpenServer — локальний сервер WAMP, який працює як у стаціонарному, так і в мобільному режимах.

Ця аббревіатура позначає перші літери програмного забезпечення, включеного в платформу.

Виглядає це так:

- Windows — сучасна операційна система з графічним інтерфейсом користувача.
- Apache є найпопулярнішим веб-сервером з відкритим кодом.
- MySQL — це система керування базами даних.
- Інтерпретатор мови PHP.

OpenServer забезпечує зручний інтерфейс для управління цими компонентами та спрощує розгортання та тестування веб-додатків на локальному рівні перед їхнім розміщенням на віддаленому сервері.

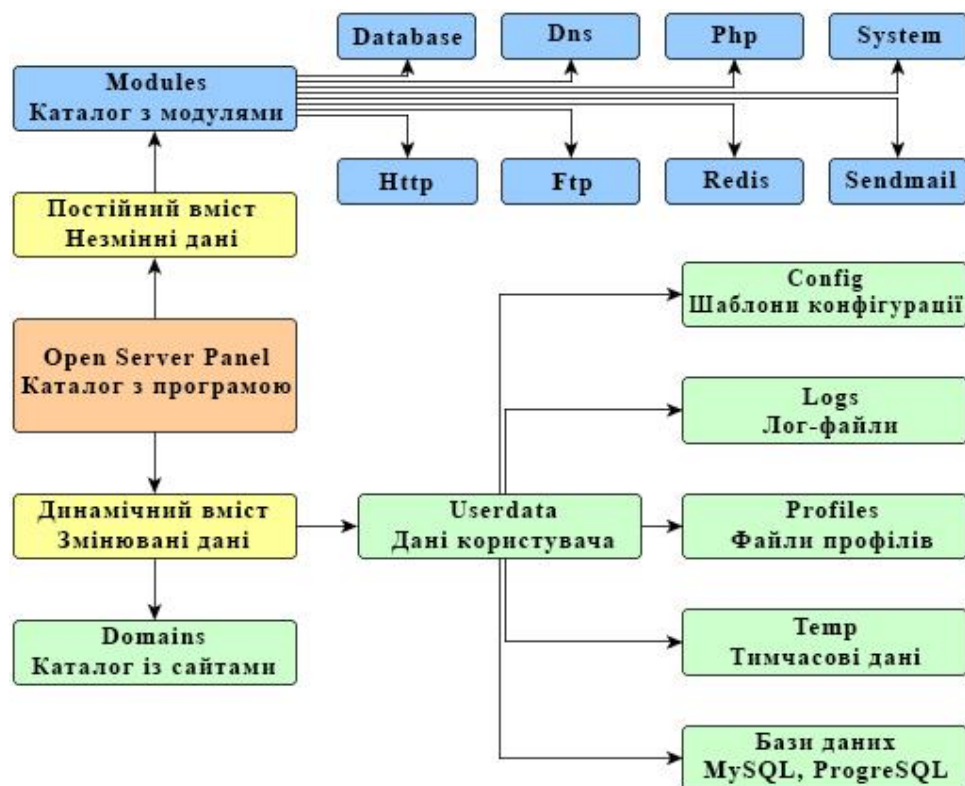


Рис 1.2 Архітектура OpenServer

1.2.1 Переваги OpenServer

Створивши сайт на OpenServer, ви отримаєте цінну можливість познайомитися з інструментом, повчитися на своїх помилках, яких ви навряд чи допустите, користуючись платним хостингом.

Таким чином, у вас є цінна можливість виявити сильні та слабкі сторони вашої CMS.

OpenServer є відмінним інструментом для розробників і веб-мастерів з кількома вагомими перевагами. Початковою перевагою є можливість створення власного сайту без необхідності витратити кошти на платний хостинг. Це надає унікальну можливість вивчити основи веб-розробки, експериментуючи з різними технологіями та практиками.

Створення сайту на OpenServer також дозволяє вам виявити сильні та слабкі сторони вашої системи управління контентом (CMS). Зазначивши помилки та вдосконалюючи функціонал, ви розвиваєте свої навички і підвищуєте рівень своєї ефективності у веб-розробці.

Крім того, встановлення розширень, таких як Joomla, надає додаткові можливості для розширення функціоналу вашого сайту. Це важливий етап у вдосконаленні та адаптації вашого проекту під конкретні потреби та вимоги користувачів.

З врахуванням цих переваг OpenServer стає не тільки ідеальним середовищем для початківців, але й потужним інструментом для досвідчених розробників, які шукають ефективні засоби для тестування, вдосконалення та оптимізації своїх веб-проектів.

Розділ "Введення в PHP, LMS Moodle та OpenServer" в магістерській роботі спрямований на введення читача у ключові технології, які лягли в основу створення та розгортання плагіну для інтеграції Google Meet в навчальну платформу LMS Moodle. Основною метою цього розділу було ознайомити читача із середовищем розробки, мовою програмування та технологією, які використовувалися для досягнення завдань дослідження.

Розглянуто основні принципи та можливості PHP як мови програмування для веб-розробки. Пояснено важливі аспекти використання PHP для створення динамічних веб-сайтів та плагінів.

Надана інформація про LMS Moodle як навчальну платформу, зокрема її структуру та основні можливості. Виділено ключові елементи, які були важливі при розробці плагіну для цієї платформи.

Пояснено використання OpenServer як локального сервера для розробки та тестування веб-додатків. Описані основні переваги використання OpenServer у контексті програмування та розгортання плагінів.

Важливі Аспекти Інтеграції Технологій: Виділено взаємодію між PHP, LMS Moodle та OpenServer у контексті розробки плагіну для інтеграції Google Meet.

Пояснено, як ці технології взаємодіють та спільно забезпечують ефективність та надійність плагіну. Цей розділ не лише вводить читача у технічні аспекти дослідження, але й створює фундамент для подальшого розуміння процесів програмування та інтеграції в рамках вибраних технологічних середовищ.

1.3 Загальні відомості про Google Meet

Google Meet – це служба відеочату, призначена в першу чергу для використання в бізнесі та офісі, яка дозволяє колегам спілкуватися в чаті за допомогою відео та тексту.

1.3.1 Які особливості google meet?

На відміну від багатьох інших продуктів Google, вам потрібно платити за обліковий запис Google Meet, вартість якого починається від 6 доларів США на місяць за користувача.

Обліковий запис Google Meet Basic дозволить вам проводити дзвінки до 25 осіб одночасно, а вищі рівні членства підвищать ліміт до 100 одночасно.

Google Meet, також відомий як Google Hangouts Meet, створений для того, щоб десятки людей могли приєднатися до однієї віртуальної зустрічі.

Він призначений для використання підприємствами та іншими організаціями, і це чудовий спосіб для спілкування колег, які не працюють в одній будівлі. Організатор Google Meet може ділитися тим, що є на його екрані, з усіма, хто бере участь у дзвінку, і будь-який учасник може будь-коли вимкнути власний аудіо- та/або відеоканал, беручи участь, як захоче.

1.3.2. Історія

У березні 2020 року через пандемію COVID-19 було оголошено, що доступ до програми буде безкоштовним, що призвело до 30-кратного збільшення кількості користувачів у квітні 2020 року порівняно з січнем 2020 року, досягнувши 100 мільйонів користувачів на рік.

До 31 березня 2021 року тривалість безкоштовної конференції для облікових записів Gmail обмежена 24 годинами, з 1 квітня 2021 року тривалість буде обмежена 60 хвилинами для безкоштовних облікових записів (раніше було оголошено, що зміна умов використання буде реалізовано 1 жовтня 2020 р. У листопаді 2023 року Google покращив сервіс відеоконференцій Google Meet, додавши новий функціональний компонент: розпізнавання жестів рук.

Тепер учасники відеодзвінків можуть підняти руки, щоб показати, що вони хочуть говорити. Щоб ця функція працювала належним чином, камера користувача має бути ввімкнена, а руку чітко видно в кадрі. Через короткий проміжок часу система автоматично розпізнає жест і сповістить інших учасників конференції.

РОЗДІЛ 2. ПРОГРАМУВАННЯ ТА НАЛАШТУВАННЯ ПЛАГІНУ, НАЛАШТУВАННЯ OPENSERVR

2.1 Налаштування OpenServer

Для початку потрібно завантажити OpenServer з офіційного ресурсу. Після завантаження налаштовуємо OpenServer для майбутньої коректної роботи плагіну. Для цього потрібно натиснути правої кнопкою миші на прапорець, а потім налаштування.

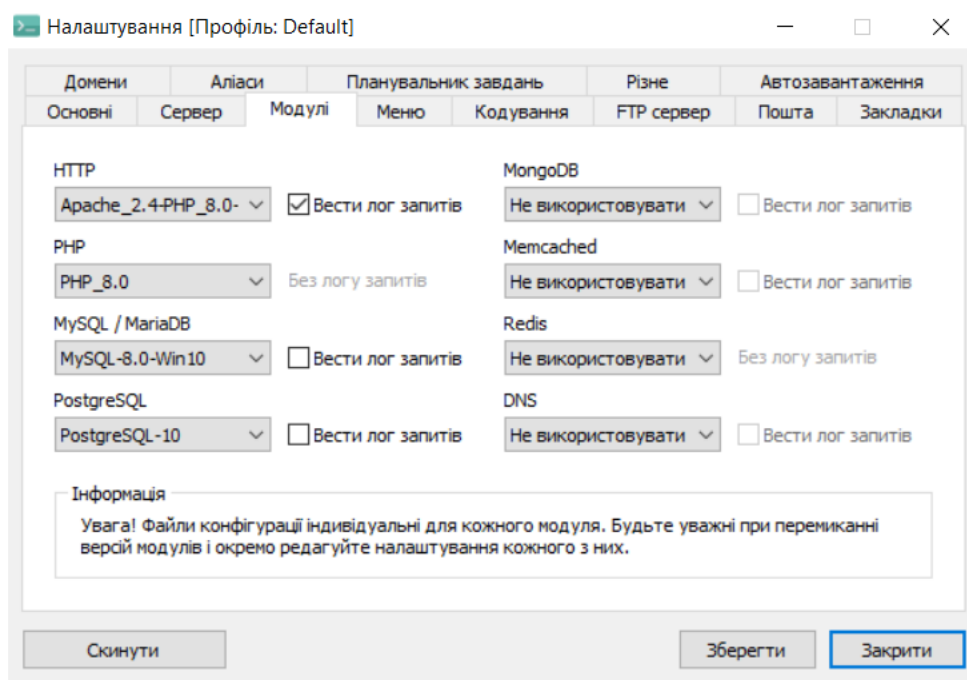


Рис 2.1 Персоналізоване налаштування OpenServer

Також, створюємо у каталозі OSPanel (куди завантажували додаток Open Server Panel). Створюємо папку потрібного домену, беремо наприклад edu2dl.net. Після чого завантажуюмо з офіційного сайту платформи Moodle. Потрібно авторизуватися та завантажити Standart Moodle у вигляді zip-файлу для роботи з сервером та обов'язково ознайомитись з документацією.

Також необхідно створити базу даних, для цього у панелі OpenServer запускаємо сервер, після чого обираємо Додатково — MyPhpAdmin. Переходимо у вікно «Бази даних». Після чого називаємо нашу базу даних та створюємо її. Після створення бази даних архів з Moodle`ом, котрий ми завантажили, потрібно розархівувати у папку з назвою домену, яку ми створили.

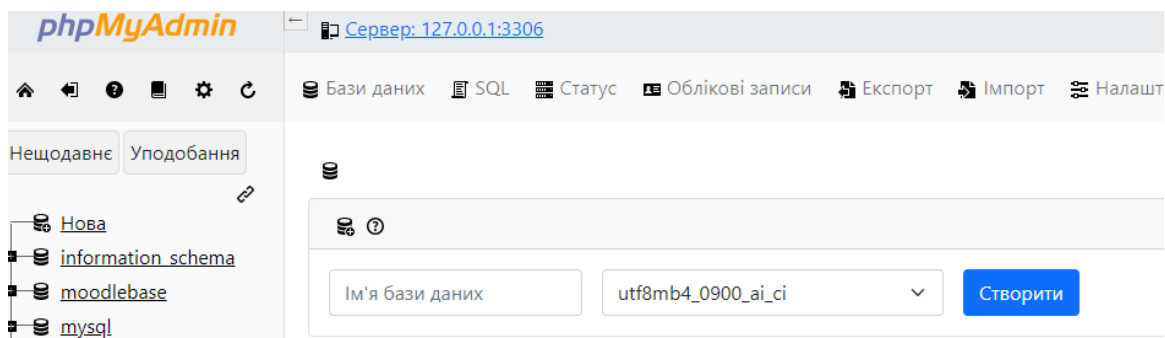


Рис 2.2 Панель створення бази даних

Якщо все зробили правильно, то почнеться встановлення та завантаження платформи LMS Moodle на домен користувача. У першому вікні налаштування потрібно обрати потрібну нам мову, обираємо українську.

Наступним кроком є підтвердження шляхів. Тобто, веб-адреса повна веб-адреса, за якою ваш сайт moodle буде доступним. Moodle може мати тільки одну адресу доступу. Якщо ваш сайт має кілька публічних адрес, то встановіть в DNS перенаправлення всіх інших адрес на цю. Якщо до вашого сайту мають одночасний доступ і з Інтернету, і з Інтранету (локальної мережі), то забезпечте доступ з локального середовища за публічною адресою. Moodle каталог де, повний шлях до каталогу встановлення moodle та каталог даних, де Moodle буде зберігати всі файли, які завантажують користувачі.

Далі йде драйвер бази даних, обираємо тип: Поліпшений MySQL. Після чого налаштування баз даних, де потрібно заповнити певні поля — Сервер бази даних, Ім'я бази даних (де вказуємо ім'я бази даних, яку створювали у PhpMyAdmin), Користувач бази даних (це логін авторизації у PhpMyAdmin, по стандарту «root»), Пароль бази даних (це пароль авторизації користувача «root» у PhpMyAdmin, але по останній документації Open Server Panel пароля немає), Префікс таблиць, Порт до бази даних. Після чого приймаємо «Угоду користувача». Після чого йде перевірка налаштувань серверу, де ми бачимо що у нас все добре та всі серверні залежності відповідають мінімальним вимогам.

Перевірка серверу

Назва	Інформація	Звіт	Модуль	Статус
php_setting	opcache.enable	<div> <div>Налаштування PHP повинні бути змінені.</div> <div>Кешування PHP з opcache caching підвищує продуктивність і знижує вимоги до пам'яті, розширення OPcache рекомендується і повністю підтримується.</div> </div>		Перевірка
unicode		вимагається для встановлення та включення		ОК
database	mysql (8.0.30)	потрібна версія 8.0, ви запускаєте 8.0.30		ОК
php		потрібна версія 8.0.0, ви запускаєте 8.0.22		ОК
pcreunicode		рекомендується встановити та включити		ОК
php_extension	iconv	вимагається для встановлення та включення		ОК
php_extension	mbstring	вимагається для встановлення та включення		ОК
php_extension	curl	вимагається для встановлення та включення		ОК
php_extension	openssl	вимагається для встановлення та включення		ОК
php_extension	tokenizer	рекомендується встановити та включити		ОК
php_extension	soap	рекомендується встановити та включити		ОК
php_extension	ctype	вимагається для встановлення та включення		ОК
php_extension	zip	вимагається для встановлення та включення		ОК
php_extension	zlib	вимагається для встановлення та включення		ОК
php_extension	gd	вимагається для встановлення та включення		ОК
php_extension	simplexml	вимагається для встановлення та включення		ОК
php_extension	spl	вимагається для встановлення та включення		ОК
php_extension	pcre	вимагається для встановлення та включення		ОК
php_extension	dom	вимагається для встановлення та включення		ОК
php_extension	xml	вимагається для встановлення та включення		ОК
php_extension	xmlreader	вимагається для встановлення та включення		ОК
php_extension	intl	вимагається для встановлення та включення		ОК
php_extension	json	вимагається для встановлення та включення		ОК

Рис 2.3 Результат перевірки серверу

Після успішної інсталяції, користувач прямує до початкового налаштування, де потрібно заповнити важливі дані для створення персоналізованого профілю. Цей етап не тільки забезпечує безперебійну роботу програми, а й створює комфортний простір для користувача.

На першому етапі необхідно вказати базову інформацію, таку як ім'я та прізвище. Це дає змогу застосунку звертатися до користувача на ім'я, роблячи взаємодію більш індивідуалізованою. Потім слід вказати контактні дані, такі як електронна пошта яку можна використовувати для зв'язку і відновлення облікового запису.

Далі користувачеві надається можливість додати додаткові відомості, як-от номеру телефону, місця розташування або вподобання. Ці дані не тільки допомагають поліпшити персональний досвід користувача.

Важливою частиною процесу є створення безпечного пароля, який забезпечує захист облікового запису користувача. Система може запропонувати рекомендації щодо створення надійного пароля, щоб гарантувати безпеку особистих даних.

Після завершення етапу заповнення даних, користувач отримує можливість переглянути та підтвердити надану інформацію перед збереженням. Це дає змогу уникнути помилок і переконатися в коректності даних перед початком активного використання програми. Після чого вже переходимо до нашої платформи Moodle. Можна вважати, що налаштування нашого Open Server`у та налаштування самої платформи завершено.

▼ **Основне**

Ім'я входу ?

Оберіть спосіб ідентифікації: ? Ручна реєстрація

Ваш пароль повинен мати принаймні 8 символів, принаймні 1 цифр(а), принаймні 1 букв(а) нижнього регістру, принаймні 1 букв(а) верхнього регістру, принаймні 1 спеціальних символів, таких як *, - або #

Новий пароль ! ? 👁

☐ Примусити змінити пароль ?

Прізвище !

Ім'я !

Електронна пошта !

Видимість електронної пошти ?

Місто

Рис 2.4 Початкове налаштування користувача

2.2 Налаштування PhpStorm

Відкрийте ваш веб-браузер і перейдіть на офіційний сайт PhpStorm. На головній сторінці сайту знайдіть розділ "Download" або "Скачати" і клацніть за відповідним посиланням. Виберіть операційну систему, під якою ви хочете встановити PhpStorm (Windows, macOS, Linux). Натисніть на кнопку "Download" або "Завантажити", щоб почати завантаження інсталяційного файлу.

Після завершення завантаження відкрийте інсталяційний файл PhpStorm. Дотримуйтесь інструкцій інсталятора, вибираючи необхідні параметри, як-от розташування інсталяції та створення ярликів. Після завершення процесу встановлення запустіть PhpStorm.

Під час першого запуску вам може знадобитися ввести ліцензійний ключ або вибрати опцію оціночного періоду, якщо у вас немає ліцензії.

2.3 Програмування плагіну

2.3.1 Розбір основних елементів коду. Setting.php

Код, представлений у цьому тексті, розв'язує задачу налаштування параметрів модуля Google Meet у Moodle, надаючи адміністраторам інтерфейс для управління різними аспектами інтеграції Google Meet із платформою навчання Moodle. Увесь процес виконання програми розбито на етапи, кожен з яких виконує певну функцію.

defined('MOODLE_INTERNAL') || die(); Тут перевіряється, чи визначена константа **MOODLE_INTERNAL**. Якщо ні, то виконання скрипта завершується. Це призначено для забезпечення безпеки та запобігання прямому доступу ззовні Moodle.

if (\$ADMIN->fulltree) {: Цей блок виконується тільки в разі, якщо змінна **\$ADMIN->fulltree** істинна. Зазвичай це означає, що код виконується в контексті адміністратора сайту, що гарантує, що ці налаштування будуть доступні тільки адміністраторам.

Створення масиву **\$options** з одним порожнім елементом. Ймовірно, це використовується для ініціалізації випадаючого списку.

Отримання списку всіх "випусків" в OAuth 2.0 з використанням методу **core\oauth2\api::get_all_issuers()** і збереження результату в змінній **\$issuers**.

Створення списку, що випадає (**\$options**), на основі даних про "випускників". Це робиться для подальшого використання в адміністративній формі.

Додавання різних адміністративних налаштувань з використанням різних класів Moodle (**admin_setting_configselect**, **admin_setting_configcheckbox**, **admin_setting_configtmleditor** та інші). Ці налаштування містять у собі списки, що випадають, чекбокси, редактори HTML-коду та інші елементи для конфігурації параметрів модуля Google Meet.

Завершення блоку коду, який виконується тільки за умови, що **\$ADMIN->fulltree** істинно. Це гарантує, що ці налаштування доступні тільки адміністраторам і не будуть відкриті для звичайних користувачів.

2.3.2. Розбір основних елементів коду. Index.php

Заголовок **GNU General Public License (GPL)**: На початку файлу вказаний заголовок, що описує ліцензію GNU General Public License, згідно з якою поширюється Moodle.

Підключення конфігураційного файлу Moodle: **require(__DIR__.'../../config.php');**

Цей рядок підключає конфігураційний файл Moodle, який містить основні налаштування та підключення.

Підключення необхідних бібліотек та класів: **require_once(__DIR__.'/lib.php');** Цей рядок підключає бібліотеки та класи, необхідні для роботи з модулем, представленим у цьому коді.

Отримання параметра 'id' з запиту: **\$id = required_param('id', PARAM_INT);**

Тут витягується параметр 'id' з запиту, який, ймовірно, представляє ідентифікатор курсу.

Отримання інформації про курс з бази даних Moodle: **\$course = \$DB->get_record('course', array('id' => \$id), '*', MUST_EXIST);**

Даний код використовує об'єкт бази даних Moodle (**\$DB**) для отримання інформації про курс з ідентифікатором, переданим у параметрі 'id'.

Перевірка авторизації користувача для доступу до курсу: **require_course_login(\$course);**

Цей код перевіряє, чи авторизований поточний користувач для доступу до вказаного курсу.

Створення та виклик події (event):

```
$event =  
\mod_googlemeet\event\course_module_instance_list_viewed::create(array(  
    'context' => context_course::instance($course->id)  
));  
$event->add_record_snapshot('course', $course);  
$event->trigger();
```

Цей відрізок коду відповідає за створення та виклик події (event) при перегляді списку модулів курсу. Події використовуються для відстеження та реєстрації різних подій в Moodle.

Встановлення параметрів сторінки (PAGE):

```
$PAGE->set_url('/mod/googlemeet/index.php', array('id' => $id));
```

```
$PAGE->set_title(format_string($course->fullname));
```

```
$PAGE->set_heading(format_string($course->fullname));
```

```
$PAGE->set_context($coursecontext);
```

Тут встановлюються різні параметри сторінки, такі як URL, заголовок, контекст та інші.

Вивід заголовку та початок вмісту сторінки: **echo \$OUTPUT->header();**

Цей код виводить заголовок сторінки та починає вміст.

Вивід інформації про модуль: **echo \$OUTPUT->heading(\$modulenameplural);**

Виводиться заголовок, який, ймовірно, представляє назву модуля.

Отримання списку модулів курсу та їх відображення: **\$googlemeets = get_all_instances_in_course('googlemeet', \$course);** Отримується список усіх модулів типу 'googlemeet' у вказаному курсі, після чого вони відображаються у вигляді таблиці.

Вивід таблиці з модулями: **echo html_writer::table(\$table);** Сформована таблиця з модулями виводиться на сторінку.

Вивід нижнього колонтитулу сторінки: **echo \$OUTPUT->footer();** Цей код виводить нижній колонтитул сторінки.

Загальна мета коду - відобразити список модулів типу 'googlemeet' для певного курсу в середовищі Moodle і зареєструвати подію перегляду цих модулів.

2.3.3. Розбір основних елементів коду. **Index.php**

Код починається з коментарів, що містять інформацію про проект Moodle, його ліцензію (**GNU GPL**) і посилання на Загальну публічну ліцензію GNU. Він перевіряє, чи визначена константа **MOODLE_INTERNAL**, і завершує скрипт, якщо ні.

Оголошення класу: Клас **mod_googlemeet_mod_form** розширює **moodleform_mod**, вказуючи, що це форма, яка використовується для створення або оновлення екземплярів модуля **mod_googlemeet** в Moodle.

Використання операторів та необхідних файлів:

Включає необхідні файли, такі як **moodleform_mod.php** та **locallib.php**.

Оператор **use mod_googlemeet\client;** вказує на використання класу клієнта з простору імен **mod_googlemeet**.

Властивості класу: Статичний масив **\$datefieldoptions** визначено для полів **date_time_selector** у формі.

Методи класу: визначення методу **Method**: Визначає елементи форми та їх розташування.

Використовує функції API форми Moodle для додавання елементів, таких як текстові поля, прапорці, заголовки тощо.

Обробляє логіку входу/виходу з облікового запису Google і відображає відповідні повідомлення.

Генерує елементи форми для налаштування кімнати Google Meet, включаючи назву кімнати, дату, час, повторюваність, URL-адресу, сповіщення тощо.

Метод попередньої обробки даних: Декодує формат JSON з бази даних для певних полів форми.

Метод валідації: Застосовує правила валідації для надісланих даних форми. Перевіряє достовірність часу початку та закінчення події, дати повторення, дні тижня, URL-адреси та адреси електронної пошти.

Обробляє спеціальну логіку перевірки на основі статусу входу в обліковий запис Google.

checkweekdays Метод: Перевіряє, чи вибрані дні тижня є дійсними для вказаного діапазону дат.

Метод `validate_url`: Перевіряє надану URL-адресу для Google Meet.

Коментарі: По всьому коду є коментарі, що пояснюють призначення певних блоків коду та надають контекстну інформацію.

Глобальні змінні: Використовує глобальні змінні, такі як **\$CFG** та **\$OUTPUT** для конфігурації та виведення даних Moodle.

Вихід у форматі HTML: Використовує функції виведення HTML Moodle (**html_writer::div**, **html_writer::notification** і т.д.) для створення елементів HTML з відповідним стилем і повідомленнями.

Конфігурації та умови: Читає конфігурації з налаштувань Moodle. Перевіряє умови, наприклад, чи користувач увійшов до системи, чи форма створена для нового екземпляра, та інші умови, необхідні для відображення певних повідомлень або елементів.

Цей клас по суті представляє форму, яка використовується для створення або оновлення екземплярів модуля Google Meet в Moodle, включаючи різні налаштування і перевірки, пов'язані з інтеграцією Google Meet.

2.4 Висновки до розділу

У цьому розділі було проведено розгортання та дослідження плагіну для інтеграції Google Meet в навчальну платформу LMS Moodle. Основною метою було забезпечити зручний та ефективний інструмент для ведення відеоконференцій, спрямований на поліпшення процесу дистанційного навчання. Нижче представлені основні висновки та результати цього дослідження.

Розроблений плагін успішно інтегрує Google Meet в середовище LMS Moodle, забезпечуючи користувачам простий доступ до відеоконференцій безпосередньо з навчального порталу.

Використані сучасні технології та стандарти програмування для забезпечення надійності та безпеки плагіну.

Плагін включає широкий спектр функцій, що полегшують проведення відеоконференцій в рамках навчального процесу.

Користувачі відзначили зручність та інтуїтивність інтерфейсу, що сприяє покращенню їхнього досвіду використання.

Проведено дослідження ефективності використання плагіну під час дистанційного навчання, що дозволило отримати важливі дані щодо його впливу на навчальний процес.

Результати показали позитивний вплив використання відеоконференцій на активність та взаємодію студентів.

Ідентифіковано можливості для подальшого розвитку плагіну, зокрема додавання нових функцій та підтримки додаткових платформ.

Рекомендується подальше дослідження впливу відеоконференцій на якість навчання та результативність студентів.

Загалом, програмування та впровадження плагіну інтеграції Google Meet в LMS Moodle представляє собою важливий крок у розвитку сучасних засобів дистанційного навчання, забезпечуючи зручність та ефективність віддаленого взаємодії студентів та викладачів.

Для встановлення плагіну потрібно обрати (рис) вкладинку «Модулі». Після обрати «Встановлення модулю». Після чого завантажуюмо наш ZIP-архів з написаним плагіном.

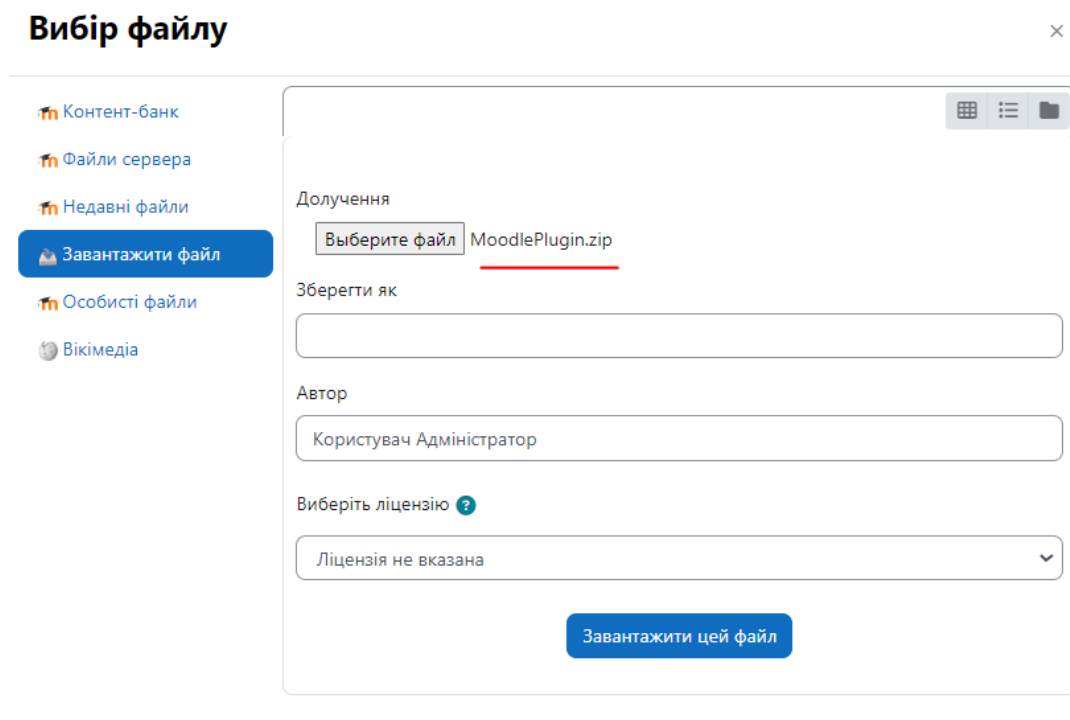


Рис 3.2 Вікно завантаження плагіну

Після чого в нас йде перевірка серверу як було при інсталяції платформи LMS Moodle.

Перевірка компонентів

На цій сторінці відображаються плагіни, які можуть вимагати вашої уваги під час оновлення, наприклад, нові плагіни, які потрібно встановити, плагіни, які потрібно оновити, відсутні плагіни тощо. Додаткові плагіни відображаються, якщо є доступне оновлення для них. Рекомендується перевірити, чи доступні новіші версії плагінів, і оновити їх вихідний код, перш ніж продовжити оновлення Moodle.

[Перевірити наявність оновлень](#)

Додатки, які потребують уваги

[Скасувати нові установки \(1\)](#)

[Додатки, які потребують уваги 1](#)

[Всі плагіни 440](#)

Назва модуля / Каталог	Поточна версія	Нова версія	Вимагає	Джерело / Статус
Модулі діяльності				
Google Meet™ for Moodle /mod/googlemeet		2023050101	• Moodle 2019052000	Розширення Буде встановлено Скасувати цю установку

[Оновити](#)

[Оновити](#)

Рис 3.3 Перевірка компонентів

Після перевірки компонентів та оновлення йде «Нові установки». Обираємо те, що нам потрібно та й надалі будемо працювати з інтеграцією Google та OAuth service.

Нові установки - Google Meet™ for Moodle

OAuth service
googlemeet | issuerid

Типово: Порожнє

[How to set up an OAuth Service](#)

Додаткові параметри повторення зустрічі
googlemeet | multieventdateexpanded

☐ Типово: Ні

Відображати розгорнутим "Налаштування повторення зустрічі" при створенні нової.

Розширене посилання (url) зустрічі
googlemeet | roomurlexpanded

☒ Типово: Так

Відображати налаштування "Посилання на зустріч" для нової зустрічі розгорнутими.

Розгорнуте нагадування
googlemeet | notificationexpanded

☐ Типово: Ні

Відображати розгорнутим налаштування "Нагадування" за замовченням для нової зустрічі.

Надсилати нагадування студентам
googlemeet | notify

☒ Типово: Так

Якщо відмічено, студенту буде надіслано нагадування щодо початку зустрічі.

За скільки хвилин до зустрічі в Google Meet
googlemeet | minutesbefore

Типово: 10

Кількість хвилин до початку зустрічі, коли має бути надіслано нагадування.

Зміст листа
googlemeet | emailcontent

Редагувати Вигляд Вставити Формат Інструменти Таблиця Довідка

tinymce

Типово:
<p>Добридень %userfirstname%,</p> <p>Це нагадування для Вас, що відбудеться зустріч у Google meet у купці %coursename%</p> <p>%googlemeetname%</p> <p>Коли: %eventdate% %duration% %timezone%</p> <p>Посилання для зустрічі: %url%</p>

Коли буде надсилатися нагадування студенту, зміст листа заповнити з наступних полів. Можуть бути використані наступні підстановки:

- %userfirstname%
- %userlastname%
- %coursename%
- %googlemeetname%
- %eventdate%

Рис 3.4 Вікно налаштування «Нові Установки»

28

3.2 Налаштування Google OAuth service

3.2.1 Що таке OAuth 2.

OAuth 2.0 (Open Authorization) - це протокол авторизації, який став широко використовуваним в інтернет-технологіях для забезпечення безпечного та обмеженого доступу до ресурсів користувача. Основною метою OAuth 2.0 є дозвіл зовнішнім службам (застосункам) використовувати обмежений доступ до користувацьких ресурсів, таких як фотографії, контакти або інші дані, без передачі користувачем свого логіну та паролю.

3.2.2 Різниця між OpenID і OAuth

Хоча на цю тему було багато пояснень, вона все одно викликає непорозуміння. OpenID використовується для аутентифікації, що означає розуміння того, що цей конкретний користувач є тим, ким він себе називає.

Наприклад, за допомогою OpenID якийсь сервіс Shop може зрозуміти, що користувач, який зайшов туди, - «Ім'я Прізвище» з Shop.com.

Під час наступної аутентифікації Shop знову зможе дізнатися і зрозуміти, що це «Ім'я», як і минулого разу.

OAuth - це протокол авторизації, тобто він дозволяє авторизувати дії, які «Ім'я» може здійснювати в Shop.com від імені «Ім'я».

У цьому разі після авторизації «Ім'я» взагалі не зможе брати участь у процесі проведення дій, наприклад, Ололо зможе самостійно завантажувати фотографії до акаунта «Ім'я».

2.3.3 Як працює OAuth 2

Як і перша версія, OAuth 2 заснована на використанні базових веб-технологій: HTTP-запитів, переспрямувань тощо.

Таким чином, використання OAuth можливе на будь-якій платформі з доступом до Інтернету і браузером: на вебсайтах, у мобільних і настільних додатках, плагінах браузера.

Основна відмінність від OAuth 1.0 - простота. У новій версії виключено громіздкі схеми підпису і зменшено кількість необхідних запитів на авторизацію. Загальна схема роботи застосунку, що використовує OAuth, така:

- запит авторизації
- доступ до захищених ресурсів

Результатом авторизації є access token - якийсь ключ (зазвичай просто набір символів), пред'явлення якого є перепусткою до захищених ресурсів. Звернення до них у найпростішому випадку відбувається за HTTPS із зазначенням у заголовках або як одного з параметрів отриманого access token'a.

У протоколі описано кілька варіантів авторизації, придатних для різних ситуацій:

- авторизація для застосунків, що мають серверну частину (найчастіше це сайти і веб-додатки)
- авторизація для повністю клієнтських додатків (мобільні та desktop-додатки)
- авторизація за логіном і паролем
- відновлення попередньої авторизації

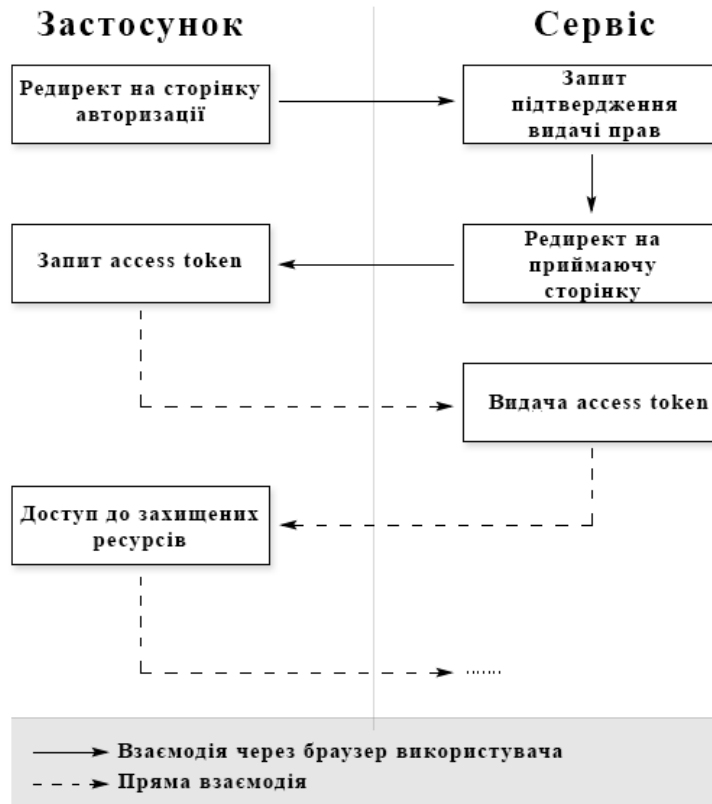


Рис 3.5 Авторизація для додатків, що мають серверну частину

Це найскладніший варіант авторизації, але дозволяє службі встановлювати лише ту програму, яка вимагає авторизації (це відбувається під час зв'язку між серверами на останньому кроці).

У всіх інших варіантах авторизація здійснюється повністю на клієнті, і, зі зрозумілих причин, можна замаскувати одну програму під іншу. Це потрібно враховувати під час реалізації автентифікації OAuth в API служби.

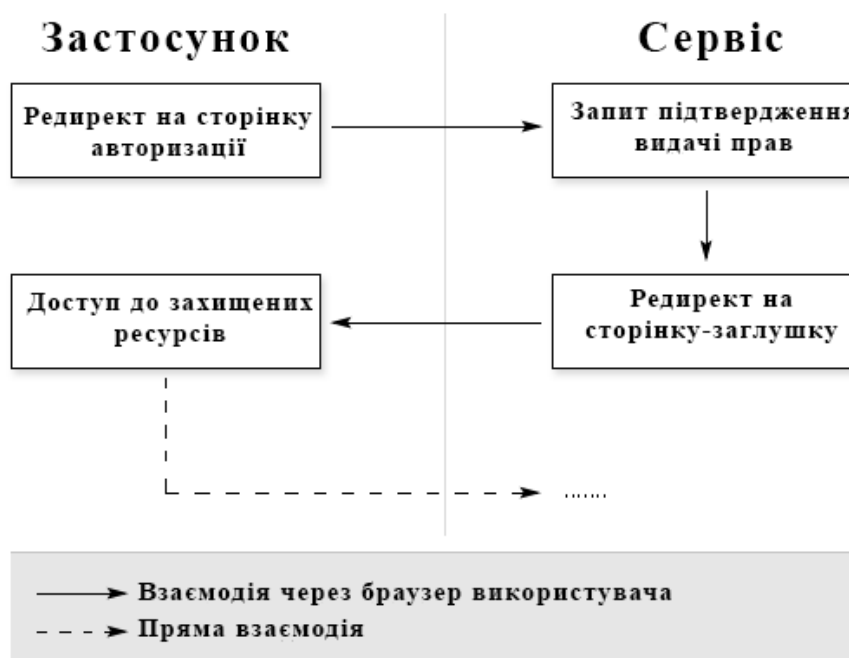


Рис 3.6 Авторизація повністю клієнтських додатків

- 1) Відкриття вбудованого браузера зі сторінкою авторизації
- 2) У користувача запитується підтвердження видачі прав
- 3) У разі згоди користувача, браузер редиректується на сторінку-заглушку у фрагменті (після #) URL якої додається access token
- 4) Додаток перехоплює редирект і отримує access token з адреси сторінки

Цей варіант вимагає підняття в застосунку вікна браузера, але не вимагає серверної частини та додаткового виклику сервер-сервер для обміну authorization code на access token.

2.3.4 Недоліки OAuth 2

OAuth 2.- це стандарт, що розвивається. Це означає, що технічні характеристики ще не фіксовані і постійно розвиваються, іноді суттєво. Тож якщо ви вирішите підтримувати стандарт зараз, будьте готові до того, що його підтримка повинна буде змінюватися в міру зміни специфікації.

З іншого боку, це також означає, що ви можете брати участь у стандартному процесі написання і висловлювати свою думку. Безпека OAuth 2. багато в чому залежить від SSL. Це істотно спрощує життя розробникам, але вимагає додаткових адміністративних та ІТ-ресурсів.

Це може стати серйозною проблемою в проєктах із високими навантаженнями.

OAuth - це простий стандарт авторизації, заснований на засадах Інтернету, що робить можливим авторизацію практично на будь-якій платформі. Цей стандарт підтримують найбільші платформи і зрозуміло, що його популярність буде тільки зростати.

3.4 Налаштування OAuth Service під нашу платформу

У зв'язку з розширенням використанням платформи Moodle як освітнього середовища, налаштування OAuth Service стає важливим етапом для покращення безпеки та легкості взаємодії з іншими системами. Цей розділ надасть детальні інструкції та крок за кроком вказівки щодо налаштування OAuth Service для Moodle-сайту.

Мета цього процесу - забезпечити безпечний та автоматизований обмін даними між Moodle та іншими додатками, використовуючи стандартні механізми авторизації. Налаштування OAuth дозволяє ефективно і безпечно інтегрувати Moodle з іншими платформами, забезпечуючи зручність в управлінні доступом і використанні ресурсів.

Під час налаштування OAuth Service для Moodle важливо слідкувати за кращими практиками з безпеки та стандартами протоколу OAuth, щоб забезпечити захист конфіденційності та цілісності даних. У цьому розділі ви знайдете необхідні вказівки та рекомендації, які допоможуть вам успішно

налаштувати OAuth Service для вашого Moodle-сайту та використовувати його на повну міру.

Для початку переходимо на сайт [Google Developers Console](#) та авторизуємося під своєю електроною поштою. Після чого створюємо проєкт та даємо йому назву, наприклад «Moodle».

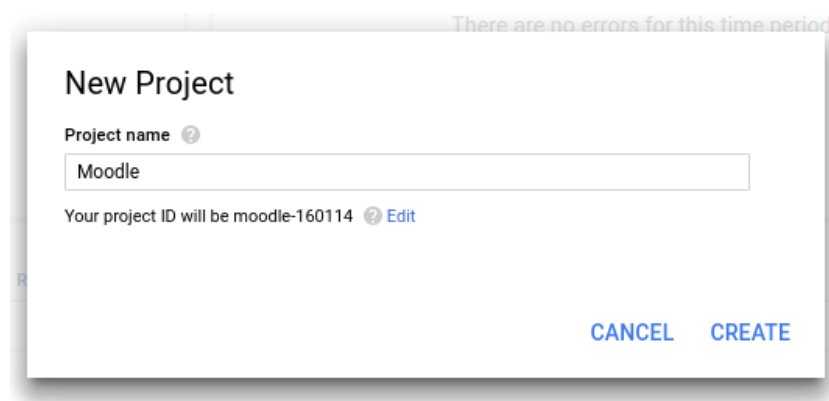


Рис 3.7 Створення проєкту «Moodle»

Після чого, потрібно активувати Google Calendar API та Google Drive API для коректної роботи плагіну, тому що є інтегрування календаря та сховища. Календар потрібен для того, щоб було видно коли та у скільки заходити, а у сховище при надання доступу буде зберігатись запис зустрічі.

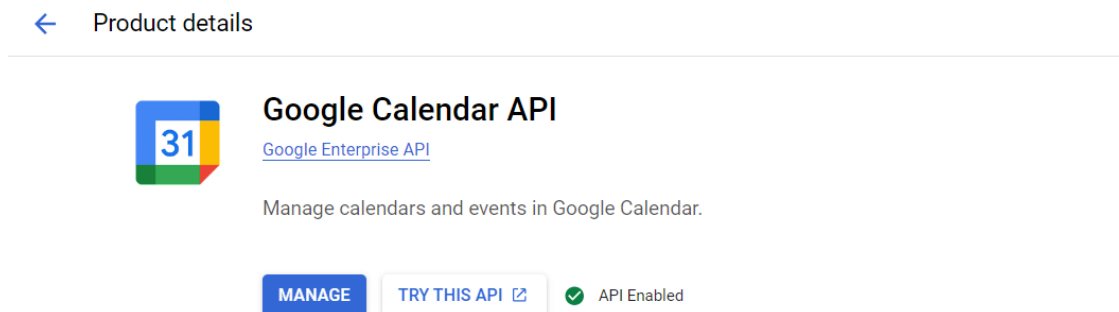


Рис 3.8 Приклад увімкненого Google Calendar API

3.4.1 Get ID та secret key

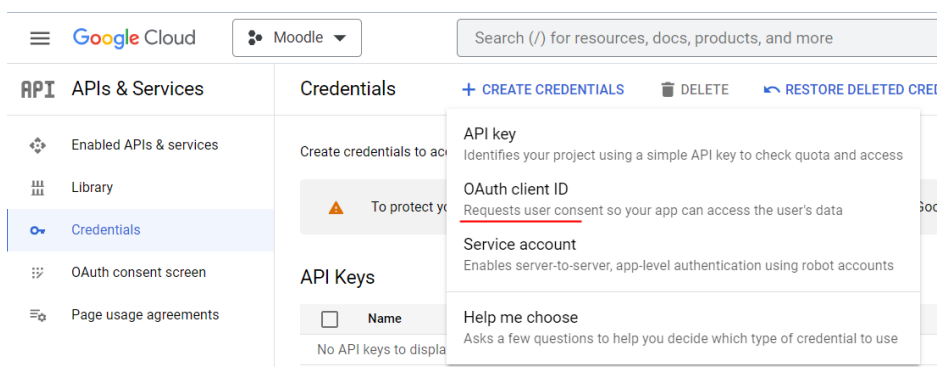


Рис 3.9 Створення Client ID

Відкриваємо сторінку Credentialspage та створюємо «Client ID». Після чого йде вікно створення, де ми пишемо ім'я застосунку, наприклад «Moodle». Далі додаємо «Add or remove scopes». Туди додаємо посилання на календар та сховище:

- <https://www.googleapis.com/auth/calendar.events>
- <https://www.googleapis.com/auth/drive>

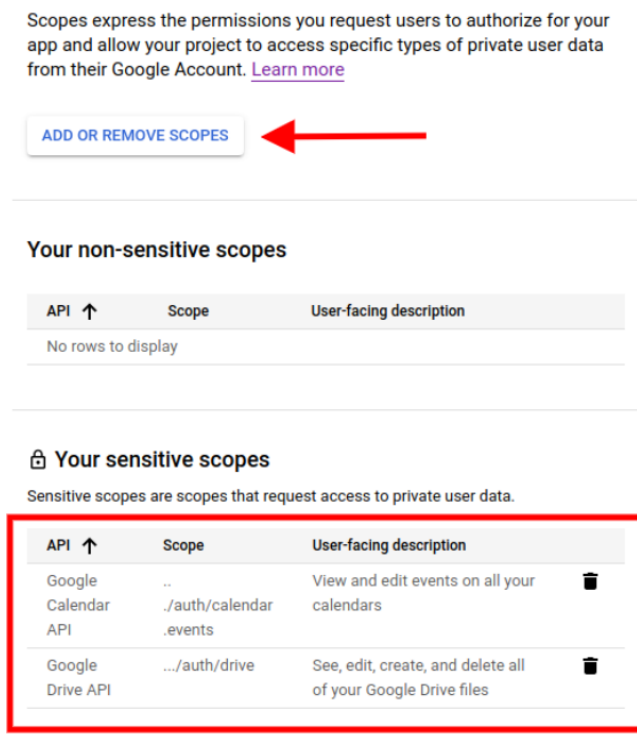

















Рис 3.10 Додавання URL-посилання в Sensitive scopes

Після чого додаємо наші URL-посилання у **Authorized JavaScript origins** та **Authorized redirect URIs**. Отримуємо Your Client Id та Your Client Secret. Далі переходимо до нашого «Moodle». Вмикаємо режим редагування та керування сайтом. У розділі керування сайтом обираємо «Сервер» та «Послуги OAuth2». Після чого обираємо «Google» та налаштовуємо під себе. У ідентифікатор клієнта та Клієнт секрет додаємо те, які дані отримали при створенні OAuth2.

Редагувати ідентифікатора: Google

[Детальні інструкції щодо налаштування загальних служб OAuth2](#)

Ім'я	 	<input type="text" value="Google"/>
Ідентифікатор клієнта	 	<input type="text" value="357240520654-m7uopilek"/>
Клієнт секрет	 	<input type="text" value="GOCSPX-bLWDNI338bIG5"/>
<input type="checkbox"/> Аутентифікація запитів маркера через заголовки H1		
URL базової служби		<input type="text" value="https://accounts.google.cc"/>
URL логотипу		<input type="text" value="https://accounts.google.cc"/>
Ця послуга буде використана		<input type="text" value="Сторінка входу та внутрішні послуги"/>
Ім'я відображається на сторінці входу		<input type="text"/>
Області, включені до запиту на вхід.		<input type="text" value="openid profile email"/>
Області, включені до запиту на вхід для автономного доступу.		<input type="text" value="openid profile email"/>
Додаткові параметри, включені до запиту на вхід.		<input type="text"/>
Додаткові параметри, включені до запиту на вхід для автономного доступу.		<input type="text" value="access_type=offline&prom"/>
Домени для входу		<input type="text"/>


☒ Потрібна перевірка електронної пошти 

Рис 3.11 Інтерфейс редагування ідентифікатора Google

3.4.2 Перевірка коректної роботи плагіну інтегрування Google Meet

Тепер на нашій платформі додаємо діяльність або ресурс, після чого обираємо Google Meet for Moodle. Після чого бачимо, що можна авторизуватись під своєю електронною поштою. Обираємо свою пошту, погоджуємося зі всіма пунктами після чого бачимо, що можна налаштувати зустріч у Google Meet, у якій URL-посилання автоматично створюється. На рисунку нижче ми бачимо, що Google акаунт під'єднаний, є авторизація з Google Drive та Google Calendar (якщо переглядати деталі). Після під'єднання до зустрічі, запис буде почато автоматично.

Назва зустрічі

ПриєднатисяEvent details

Записи

Запис	Назва	Дата	Тривалість	Видимий
Немає записів для відображення.				

Остання синхронізація: **Ніколи**
Записи з назвою: "gwz-hwod-ehp" або "Назва зустрічі (6305)"
Ініціатор зустрічі: **vitaliikuznetsov.mm@gmail.com**

Синхронізувати з Google Drive

Wait at least 10 minutes for the recording file to be generated and saved in "My Drive > Meet Recordings" of the organizer.

To remove a recording first delete the recording file from Google Drive and after click the sync button above.

To record a meeting, make sure:

- You haven't met your personal Drive quota.
- Your organization hasn't met its Drive quota.

If you have space in your Drive, but your organization doesn't have space, you can't record the meeting.

For more information, look this Help Center article:
<https://support.google.com/meet/answer/9308681>

V

Connected Google account
Vitalii Kuznetsov
vitaliikuznetsov.mm@gmail.com

Рис 3.12 Відображення коректного інтегрування плагіну

3.5 Висновки до розділу

Спочатку було детально розглянуто принципи роботи OAuth 2 і його використання для забезпечення безпеки взаємодії між платформами. Висвітлено, що OAuth 2 є ефективним механізмом авторизації, але, в той же час, виявлені його недоліки, зокрема стосовно складнощів розробки та ризиків для безпеки.

Проведено налаштування OAuth Service для його інтеграції з навчальною платформою LMS Moodle. Описані етапи та особливості налаштування, що забезпечують коректне функціонування плагіну.

Перевірка коректної роботи плагіну інтеграції Google Meet з Moodle. Здійснено аналіз різних сценаріїв використання та перевірено взаємодію між платформами. Результати дослідження вказують на успішну інтеграцію, що дозволяє зручно та ефективно використовувати Google Meet в навчальному процесі.

ВИСНОВКИ

Магістерська робота "Створення плагіну інтегрування Google Meet в навчальну платформу LMS Moodle та дослідження його ефективності" включає три основні розділи, які описують введення та обґрунтування, програмування та налаштування плагіну, а також інтеграцію його в платформу LMS Moodle.

Розділ 1: Введення в PHP, LMS Moodle, OpenServer та Google Meet

У цьому розділі ви визначили основні концепції та інструменти, які використовувалися під час створення плагіну. Ви розглянули особливості мови програмування PHP, важливі аспекти платформи LMS Moodle, серверну обстановку за допомогою OpenServer та інтеграцію з сервісом Google Meet. Зазначте важливі технічні відомості та причини вибору конкретних технологій.

Розділ 2: Програмування та налаштування плагіну, налаштування OpenServer. В цьому розділі ви детально описали процес створення та програмування плагіну для інтеграції Google Meet в LMS Moodle. Вказали на ключові етапи розробки. Також, розглянули основні кроки налаштування OpenServer, які були важливі для розгортання плагіну в тестовому середовищі.

Розділ 3: Інтеграція плагіну та його налаштування на платформі LMS Moodle. У цьому розділі ви докладно розглянули процес інтеграції розробленого плагіну в навчальну платформу LMS Moodle. Пояснили важливі етапи, використані API та інші інструменти для забезпечення сумісності.

У результаті виконаної роботи було розроблено та успішно інтегровано плагін, який дозволяє здійснювати зручне підключення до Google Meet прямо з платформи LMS Moodle. Отримані результати свідчать про ефективність плагіну в контексті навчального процесу, покращення комунікації та взаємодії між учнями та викладачами. Важливою частиною дослідження було також виявлення та вирішення можливих проблем та вдосконалення плагіну на основі отриманих відгуків.

Отже, розробка та інтеграція цього плагіну представляє значний внесок у розвиток освітніх технологій, забезпечуючи більш широкі можливості для взаємодії та навчання у віртуальному середовищі.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

- 1) Учасники проєктів Вікімедіа. Moodle – Вікіпедія. Вікіпедія. URL: [https://uk.wikipedia.org/wiki/Moodle#:~:text=Moodle%20%20це%20безкоштовна,%20відкрита%20\(,a%20також%20підтримки%20очного%20навчання](https://uk.wikipedia.org/wiki/Moodle#:~:text=Moodle%20%20це%20безкоштовна,%20відкрита%20(,a%20також%20підтримки%20очного%20навчання)
- 2) Moodle plugins directory | Moodle.org. Moodle challenge. URL: <https://moodle.org/plugins/?q=sort-by:downloads>
- 3) Учасники проєктів Вікімедіа. PHP – Вікіпедія. Вікіпедія. URL: <https://uk.wikipedia.org/wiki/PHP>.
- 4) PHP: What is PHP? - Manual. *PHP: Hypertext Preprocessor*. URL: <https://www.php.net/manual/en/intro-what-is.php>
- 5) Що таке PHP. URL: <https://freehost.com.ua/ukr/faq/wiki/chto-takoe-php/>
- 6) Open Server – професійний інструмент веб-розробника під Windows. Встановлення та налаштування OpenServer. Як правильно встановлювати та працювати з Open Server Як відкрити файл php в openserver. URL: <https://minomin.ru/uk/adobe-photoshop/open-server-professionalnyi-instrument-veb-razrabotchika-pod-windows.html>.
- 7) Open Server Panel / Локальный веб-сервер для Windows / Скачать WAMP сервер. Open Server Panel / Локальный веб-сервер для Windows / Скачать WAMP сервер. URL: <https://ospanel.io/>.
- 8) Moodle Organization. URL : <https://moodle.org/>
- 9) JetBrains. PhpStorm: PHP IDE and Code Editor from JetBrains. JetBrains. URL: <https://www.jetbrains.com/phpstorm/>.
- 10) Бітман Д. OAuth 2.0. *Хабр*. URL: <https://habr.com/ru/companies/vk/articles/115163/>
- 11) Що таке Google Meet: чим особливий цей відеочат?. FutureNow. URL: <https://futurenow.com.ua/shho-take-google-meet-chym-osoblyvyj-tsej-videochat/>.
- 12) Учасники проєктів Вікімедіа. Google Meet – Вікіпедія. *Вікіпедія*. URL: https://uk.wikipedia.org/wiki/Google_Meet
- 13) Васілев О.М. Програмування мовою PHP : Навч. посіб. Ліра-К, 2022. 368 с.

ДОДАТОК А

```
<?php
```

```
defined('MOODLE_INTERNAL') || die();
```

```
if ($ADMIN->fulltree) {
```

```
    $options = [];
```

```
    $issuers = \core\oauth2\api::get_all_issuers();
```

```
    foreach ($issuers as $issuer) {
```

```
        $options[$issuer->get('id')] = s($issuer->get('name'));
    }
```

```
    $settings->add(new admin_setting_configselect(
```

```
        'googlemeet/issuerid',
```

```
        get_string('issuerid', 'googlemeet'),
```

```
        get_string('issuerid_desc', 'googlemeet'),
```

```
        0,
```

```
        $options
```

```
    ));
```

```
    $settings->add(new admin_setting_configcheckbox(
```

```
        'googlemeet/multieventdateexpanded',
```

```
        get_string('multieventdateexpanded', 'googlemeet'),
```

```
        get_string('multieventdateexpanded_desc', 'googlemeet'),
```

```
        0
```

```
    ));
```

```
    $settings->add(new admin_setting_configcheckbox(
```



```

        'googlemeet/roomurlexpanded',
        get_string('roomurlexpanded', 'googlemeet'),
        get_string('roomurlexpanded_desc', 'googlemeet'),
        1
    ));

$settings->add(new admin_setting_configcheckbox(
    'googlemeet/notificationexpanded',
    get_string('notificationexpanded', 'googlemeet'),
    get_string('notificationexpanded_desc', 'googlemeet'),
    0
));

$settings->add(new admin_setting_configcheckbox(
    'googlemeet/notify',
    get_string('notify', 'googlemeet'),
    get_string('notify_help', 'googlemeet'),
    1
));

$minutes = array();
for ($i = 0; $i <= 120; $i = $i + 5) {
    $minutes[$i] = $i;
}

$settings->add(new admin_setting_configselect(
    'googlemeet/minutesbefore',
    get_string('minutesbefore', 'googlemeet'),
    get_string('minutesbefore_help', 'googlemeet'),
    10,

```

```
$minutes  
));  
  
$settings->add(new admin_setting_configtmleditor(  
    'googlemeet/emailcontent',  
    get_string('emailcontent', 'googlemeet'),  
    get_string('emailcontent_help', 'googlemeet'),  
    get_string('emailcontent_default', 'googlemeet'),  
    PARAM_RAW  
));  
}
```

ДОДАТОК Б

```
<?php
```

```
use mod_googlemeet\client;
```

```
require(__DIR__ . '/../../config.php');
```

```
require_once(__DIR__ . '/lib.php');
```

```
require_once(__DIR__ . '/locallib.php');
```

```
$config = get_config('googlemeet');
```

```
$id = optional_param('id', 0, PARAM_INT);
```

```
if ($id) {
```

```
    $cm = get_coursemodule_from_id('googlemeet', $id, 0, false, MUST_EXIST);
```

```
    $course = $DB->get_record('course', array('id' => $cm->course), '*',  
MUST_EXIST);
```

```
    $googlemeet = $DB->get_record('googlemeet', array('id' => $cm->instance), '*',  
MUST_EXIST);
```

```
    } else if ($g) {
```

```
        $googlemeet = $DB->get_record('googlemeet', array('id' => $n), '*',  
MUST_EXIST);
```

```
        $course = $DB->get_record('course', array('id' => $googlemeet->course), '*',  
MUST_EXIST);
```

```
        $cm = get_coursemodule_from_instance('googlemeet', $googlemeet->id, $course->  
id, false, MUST_EXIST);
```

```
    } else {
```

```
        throw new moodle_exception('missingidandcmid', 'mod_googlemeet');
```

```
    }
```

```

require_login($course, true, $cm);
$context = context_module::instance($cm->id);
require_capability('mod/googlemeet:view', $context);

$PAGE->set_url('/mod/googlemeet/view.php', array('id' => $cm->id));
$PAGE->set_context($context);

if (has_capability('mod/googlemeet:editrecording', $context)) {
    $client = new client();
    $logout = optional_param('logout', 0, PARAM_BOOL);
    if ($logout) {
        $client->logout();
    }
    $sync = optional_param('sync', 0, PARAM_BOOL);
    if ($sync) {
        $client->syncrecordings($googlemeet);
    }
}

// Make sure URL exists before generating output - some older sites may contain
empty urls
// Do not use PARAM_URL here, it is too strict and does not support general URIs!
$url = trim($googlemeet->url);
$pattern = "/^https:\\\\meet.google.com\\[-a-zA-Z0-9@:%._\\+~#=]{3}\\[-a-zA-Z0-9@:%._\\+~#=]{4}\\[-a-zA-Z0-9@:%._\\+~#=]{3}\\$/";
if (!preg_match($pattern, $url)) {
    googlemeet_print_header($googlemeet, $cm, $course);
    googlemeet_print_heading($googlemeet, $cm, $course);
    googlemeet_print_intro($googlemeet, $cm, $course);
}

```

```

        notice(get_string('invalidstoredurl', 'googlemeet'), new
moodle_url('/course/view.php', array('id' => $cm->course)));
        die;
    }
    unset($url);

    // Completion and trigger events.
    googlemeet_view($googlemeet, $course, $cm, $context);

    googlemeet_print_header($googlemeet, $cm, $course);
    googlemeet_print_heading($googlemeet, $cm, $course, true);
    googlemeet_print_intro($googlemeet, $cm, $course, true);

    echo html_writer::link($googlemeet->url,
        get_string('entertheroom', 'googlemeet'),
        ['class' => 'btn btn-primary', 'target' => '_blank', 'title' => get_string('entertheroom',
'googlemeet')]);

    if (has_capability('mod/googlemeet:editrecording', $context)) {
        if ($googlemeet->eventid != null) {
            echo
html_writer::link('https://calendar.google.com/calendar/u/0/r/eventedit/'.$googlemeet-
>eventid,
                get_string('eventdetails', 'googlemeet'),
                ['class' => 'btn btn-outline-primary ml-2', 'target' => '_blank', 'title' =>
get_string('eventdetails', 'googlemeet')]);
        }
    }

    echo $OUTPUT->render_from_template('mod_googlemeet/upcomingevents',

```

```
googlemeet_get_upcoming_events($googlemeet->id));
```

```
googlemeet_print_recordings($googlemeet, $cm, $context);
```

```
echo $OUTPUT->footer();
```

ДОДАТОК В

```
<?php
```

```
defined('MOODLE_INTERNAL') || die();
```

```
use mod_googlemeet\client;
```

```
require_once($CFG->dirroot . '/course/moodleform_mod.php');
```

```
require_once($CFG->dirroot . '/mod/googlemeet/lib.php');
```

```
class mod_googlemeet_mod_form extends moodleform_mod {
```

```
    /** @var array options to be used with date_time_selector fields in the quiz. */
```

```
    public static $datefieldoptions = array('optional' => true);
```

```
    /**
```

```
     * Defines forms elements
```

```
    */
```

```
    public function definition() {
```

```
        global $CFG, $OUTPUT;
```

```
        $config = get_config('googlemeet');
```

```
        $mform = $this->_form;
```

```
        $client = new client();
```

```
        $logout = optional_param('logout', 0, PARAM_BOOL);
```

```
        if ($logout) {
```

```
            $client->logout();
```

```
        }
```

```
        if (empty($this->current->instance)) {
```

```

$clientislogged = optional_param('client_islogged', false, PARAM_BOOL);

// Was logged in before submitting the form and the google session expired
after submitting the form.
if ($clientislogged && !$client->check_login()) {
    $mform->addElement('html',    html_writer::div(get_string('sessionexpired',
'googlemeet') .
        $client->print_login_popup(),    'mdl-align    alert    alert-danger
googlemeet_loginbutton'
    ));

    // Whether the customer is enabled and if not logged in to the Google
account.
} else if ($client->enabled && !$client->check_login()) {
    $mform->addElement('html',
html_writer::div(get_string('logintoyourgoogleaccount', 'googlemeet') .
        $client->print_login_popup(),    'mdl-align    alert    alert-info
googlemeet_loginbutton'
    ));
}

// If is logged in, shows Google account information.
if ($client->check_login()) {
    $mform->addElement('html', $client->print_user_info('calendar'));
    $mform->addElement('hidden', 'client_islogged', true);
}

} else {
    $mform->addElement('hidden', 'client_islogged', false);
}

```



```

$mform->setType('client_islogged', PARAM_BOOL);

// Adding the "general" fieldset, where all the common settings are shown.
$mform->addElement('header', 'general', get_string('general', 'form'));

// Adding the standard "name" field.
$mform->addElement('text', 'name', get_string('roomname', 'googlemeet'),
array('size' => '50'));

if (!empty($CFG->formatstringstriptags)) {
    $mform->setType('name', PARAM_TEXT);
} else {
    $mform->setType('name', PARAM_CLEANHTML);
}

$mform->addRule('name', null, 'required', null, 'client');
$mform->addRule('name', get_string('maximumchars', '', 255), 'maxlength', 255,
'client');

$this->standard_intro_elements();
$element = $mform->getElement('introeditor');
$attributes = $element->getAttributes();
$attributes['rows'] = 5;
$element->setAttributes($attributes);

$hours = [];
$minutes = [];
for ($i = 0; $i <= 23; $i++) {
    $hours[$i] = sprintf("%02d", $i);
}

```

```

for ($i = 0; $i < 60; $i++) {
    $minutes[$i] = sprintf("%02d", $i);
}

$eventtime = [
    $mform->createElement('date_selector', 'eventdate', ''),
    $mform->createElement('html', '<div style="width: 100%;"></div>'),
    $mform->createElement('html', '<div class="items-center">' . get_string('from',
'googlemeet') . '</div>'),
    $mform->createElement('select', 'starthour', get_string('hour', 'form'), $hours,
false, true),
    $mform->createElement('select', 'startminute', get_string('minute', 'form'),
$minutes, false, true),
    $mform->createElement('html', '<div class="items-center">' . get_string('to',
'googlemeet') . '</div>'),
    $mform->createElement('select', 'endhour', get_string('hour', 'form'), $hours,
false, true),
    $mform->createElement('select', 'endminute', get_string('minute', 'form'),
$minutes, false, true),
    $mform->createElement('html',
        '<div id="id_googlemeet_eventtime_error" class="form-control-feedback
invalid-feedback"></div>'
    ),
];

$mform->addGroup($eventtime, 'eventtime', get_string('eventdate', 'googlemeet'),
["", false]);

// For multiple dates.
$mform->addElement('header', 'headeraddmultipleeventdates',
get_string('recurrenceeventdate', 'googlemeet'));

```

```

        if (!empty($config->multieventdateexpanded) || !empty($this->current-
>addmultiply)) {
            $mform->setExpanded('headeraddmultipleeventdates');
        }

        $mform->addElement('checkbox', 'addmultiply', "", get_string('repeatasfollows',
'googlemeet'));
        $mform->addHelpButton('addmultiply', 'recurrenceeventdate', 'googlemeet');

        $days = [
            $mform->createElement('checkbox', 'days[Mon]', "", get_string('monday',
'calendar')),
            $mform->createElement('checkbox', 'days[Tue]', "", get_string('tuesday',
'calendar')),
            $mform->createElement('checkbox', 'days[Wed]', "", get_string('wednesday',
'calendar')),
            $mform->createElement('checkbox', 'days[Thu]', "", get_string('thursday',
'calendar')),
            $mform->createElement('checkbox', 'days[Fri]', "", get_string('friday',
'calendar')),
            $mform->createElement('checkbox', 'days[Sat]', "", get_string('saturday',
'calendar')),
        ];

        if ($CFG->calendar_startwday === '0') { // Week start from sunday.
            array_unshift($days, $mform->createElement('checkbox', 'days[Sun]', "",
get_string('sunday', 'calendar')));
        } else {
            array_push($days, $mform->createElement('checkbox', 'days[Sun]', "",
get_string('sunday', 'calendar')));
        }
    }
}

```

```

    }

    array_push($days,
        $mform->createElement('html',
            '<div id="id_googlemeet_days_error" class="form-control-feedback invalid-
feedback"></div>'
        )
    );

    $mform->addGroup($days, 'days', get_string('repeaton', 'googlemeet'),
        ['&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;'], false);
    $mform->disabledIf('days', 'addmultiply', 'notchecked');

    $period = array(
        1 => 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20,
        21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 31, 32, 33, 34, 35, 36
    );
    $periodgroup = [
        $mform->createElement('select', 'period', "", $period, false, true),
        $mform->createElement('html', '<div class="items-center">' .
get_string('week', 'googlemeet') . '</div>'),
        $mform->createElement('html',
            '<div id="id_googlemeet_periodgroup_error" class="form-control-feedback
invalid-feedback"></div>'
        ),
    ];
    $mform->addGroup($periodgroup, 'periodgroup', get_string('repeatevery',
'googlemeet'), [], false);
    $mform->disabledIf('periodgroup', 'addmultiply', 'notchecked');

```

```

$eventenddategroup = [
    $mform->createElement('date_selector', 'eventenddate', ""),
    $mform->createElement('html',
        '<div id="id_googlemeet_eventenddategroup_error" class="form-control-
feedback invalid-feedback"></div>'
    ),
];
$mform->addGroup($eventenddategroup, 'eventenddategroup',
    get_string('repeatuntil', 'googlemeet'), ["", false);
$mform->disabledIf('eventenddategroup', 'addmultiply', 'notchecked');

$mform->addElement('header', 'headerroomurl', get_string('roomurl',
'googlemeet'));
if (!empty($config->roomurlexpanded)) {
    $mform->setExpanded('headerroomurl');
}

if (!empty($this->current->instance) && $client->enabled) {
    $mform->addElement('static', 'url_caution', "",
        $OUTPUT->notification(get_string('roomurl_caution', 'googlemeet'),
'warning')
    );
}

if ($client->check_login() && empty($this->current->instance)) {
    $mform->addElement('static', 'url_desc', "", $OUTPUT-
>notification(get_string('roomurl_desc', 'googlemeet'), 'info'));
    $mform->addElement('text', 'url', get_string('roomurl', 'googlemeet'), ['size' =>
'50', 'readonly' => true]);
    $mform->setType('url', PARAM_RAW);
}

```

```

        $mform->addElement('text',      'creatoremail',      get_string('creatoremail',
'googlemeet'),
        ['size' => '50', 'readonly' => true]
    );
    $mform->setType('creatoremail', PARAM_RAW);
} else {
    $mform->addElement('text', 'url', get_string('roomurl', 'googlemeet'), ['size' =>
'50']);
    $mform->setType('url', PARAM_URL);
    $mform->addHelpButton('url', 'url', 'googlemeet');

    $mform->addElement('text',      'creatoremail',      get_string('creatoremail',
'googlemeet'), ['size' => '50']);
    $mform->setType('creatoremail', PARAM_RAW);
    $mform->addHelpButton('creatoremail', 'creatoremail', 'googlemeet');
}

$mform->addElement('header',    'headernotification',    get_string('notification',
'googlemeet'));
if (!empty($config->notificationexpanded)) {
    $mform->setExpanded('headernotification');
}

$mform->addElement('checkbox', 'notify', "", get_string('notify', 'googlemeet'));
$mform->setDefault('notify', $config->notify);
$mform->addHelpButton('notify', 'notify', 'googlemeet');

$minutes = [];
for ($i = 0; $i <= 120; $i = $i + 5) {

```

```

        $minutes[$i] = $i;
    }
    $minutesbefore = $mform->addElement('select',
        'minutesbefore', get_string('minutesbefore', 'googlemeet'), $minutes, false, true
    );
    $minutesbefore->setSelected($config->minutesbefore);
    $mform->addHelpButton('minutesbefore', 'minutesbefore', 'googlemeet');

    // Add standard elements.
    $this->standard_coursemodule_elements();

    // Add standard buttons.
    $this->add_action_buttons();

}

public function data_preprocessing(&$defaultvalues) {
    if ($this->current->instance) {
        $defaultvalues['days'] = json_decode($defaultvalues['days'], true);
    }
}

public function validation($data, $files) {
    global $COURSE;

    $errors = parent::validation($data, $files);

    $starttime = $data['starthour'] * HOURSECS + $data['startminute'] * MINSECS;

```

```

$endtime = $data['endhour'] * HOURSECS + $data['endminute'] * MINSECS;

if ($endtime < $starttime) {
    $errors['eventtime'] = get_string('invalideventendtime', 'googlemeet');
}

if (!empty($data['addmultiply']) &&
    $data['eventdate'] !== 0 &&
    $data['eventenddate'] !== 0 &&
    $data['eventenddate'] < $data['eventdate']
) {
    $errors['eventenddategroup'] = get_string('invalideventenddate', 'googlemeet');
}

$addmulti = isset($data['addmultiply']) ? (int)$data['addmultiply'] : 0;
$days = isset($data['days']);

if ($addmulti && !$days) {
    $errors['days'] = get_string('checkweekdays', 'googlemeet');
} else if ($addmulti && !$this->checkweekdays($data['eventdate'],
$data['eventenddate'], $data['days'])) {
    $errors['days'] = get_string('checkweekdays', 'googlemeet');
}

if ($addmulti && ceil(($data['eventenddate'] - $data['eventdate']) / YEARSECS)
> 1) {
    $errors['eventenddate'] = get_string('timeahead', 'googlemeet');
}

$startdate = $data['eventdate'] + $starttime;

```



```

if ($startdate < $COURSE->startdate) {
    $errors['eventtime'] = get_string('earlierto', 'googlemeet',
        userdate($COURSE->startdate, get_string('strftimedmyhm', 'googlemeet'))
    );
}

$client = new client();
$clientislogged = optional_param('client_islogged', false, PARAM_BOOL);

if (empty($this->current->instance)) {
    // Validates the url field only if not logged into Google account.
    if (!$client->check_login() && !$clientislogged) {
        $errors = $this->validate_url($data['url'], $errors);
        if (!validate_email($data['creatoremail'])) {
            $errors['creatoremail'] = get_string('creatoremail_error', 'googlemeet');
        }
    }

    // Forces an error if the Google session expired after submitting the form.
    if (!$client->check_login() && $clientislogged) {
        $errors['client_islogged'] = "";
    }
} else {
    // Validates url field if updating instance.
    $errors = $this->validate_url($data['url'], $errors);
    if (!validate_email($data['creatoremail'])) {
        $errors['creatoremail'] = get_string('creatoremail_error', 'googlemeet');
    }
}

```

```

    return $errors;
}

/**
 * Check weekdays function.
 * @param int $eventdate
 * @param int $eventenddate
 * @param array $days
 * @return bool
 */
private function checkweekdays($eventdate, $eventenddate, $days) {
    $found = false;

    if (!$days) {
        return false;
    }

    $daysofweek = [
        0 => "Sun",
        1 => "Mon",
        2 => "Tue",
        3 => "Wed",
        4 => "Thu",
        5 => "Fri",
        6 => "Sat"
    ];

    $start = new DateTime(date("Y-m-d", $eventdate));
    $interval = new DateInterval('P1D');
    $end = new DateTime(date("Y-m-d", $eventenddate));

```

```

$end->add(new DateInterval('P1D'));

$period = new DatePeriod($start, $interval, $end);
foreach ($period as $date) {
    if (!$found) {
        foreach ($days as $day => $value) {
            $key = array_search($day, $daysofweek);
            if ($date->format("w") == $key) {
                $found = true;
                break;
            }
        }
    }
}

return $found;
}

private function validate_url(string $url, array $errors) {
    if (googlemeet_clear_url($url) == null) {
        $errors['generateurlgroup'] = get_string('url_failed', 'googlemeet');
        $errors['url'] = get_string('url_failed', 'googlemeet');
    }
    return $errors;
}
}

```