

**РІВНЕНСЬКИЙ ДЕРЖАВНИЙ ГУМАНІТАРНИЙ УНІВЕРСИТЕТ**

Факультет математики та інформатики

Кафедра інформатики та прикладної математики

«До захисту допущено»  
Завідувач кафедри  
\_\_\_\_\_ Батишкіна Ю.В.  
Протокол № \_\_\_\_\_  
від \_\_\_\_\_

**Дипломний проект (робота)**

ступеня **«Магістр»**

з спеціальності 113 Прикладна математика

на тему: Інформаційна система «Розклад занять». Проектування та реалізація

Виконав: студент 2 курсу, групи М-ПМ-21

Ляцевич Дмитро Володимирович \_\_\_\_\_

Керівник: м.т.н., ст. викладач Вороницька В.М. \_\_\_\_\_

Рецензент \_\_\_\_\_

Рівне, 2021 рік

## ЗМІСТ

СПИСОК УМОВНИХ ТЕРМІНІВ ТА ПОЗНАЧЕНЬ.....	4
ВСТУП.....	5
РОЗДІЛ 1. ОГЛЯД СУЧАСНИХ WEB-ЗАСТОСУНКІВ.....	7
1.1. Web-застосунок. Технічні особливості.....	7
1.2. Типи web-застосунків.....	8
1.2.1. Статичні web-застосунки.....	8
1.2.2. Динамічні web-застосунки.....	9
1.2.3. Інтернет-магазин або електронна комерція.....	9
1.2.4. Портал (web-застосунок).....	10
1.2.5. Анімовані web-застосунки.....	10
1.2.6. Web-застосунки з системою керування (CMS).....	10
1.3. Архітектура web-застосунків та застосування їх у різних сферах життя..	12
РОЗДІЛ 2. ОГЛЯД ІСНУЮЧИХ ПРОГРАМНИХ РІШЕНЬ ПОСТАВЛЕНОЇ ЗАДАЧІ.....	16
2.1. Розклад занять Житомирської політехніки.....	16
2.2. КПП Desk.....	18
2.3. НУВГП Desk.....	21
РОЗДІЛ 3. ВИБІР ПРОГРАМНИХ ЗАСОБІВ ТА ПРОЕКТУВАННЯ ІНФОРМАЦІЙНОЇ СИСТЕМИ.....	24
3.1. Front-end веб-ресурсу (зовнішній вигляд сайту).....	24
3.1.1. Особливості використання HTML та CSS.....	25
3.1.2. Сітка Bootstrap 4.....	28
3.1.3. JavaScript та його значення для веб-ресурсу.....	30
3.2. Back-end веб-ресурсу (внутрішні функції сайту).....	32
3.2.1. Використання PHP.....	33
3.2.2. База даних MySQL.....	37
3.2.3. Використання JQuery.....	40
3.3. Опис середовища розробки Sublime Text 3.....	43
3.4. Створення панелі адміністратора.....	44
3.5. Розробка клієнтської частини.....	51
3.6. Завантаження сайту на хостинг.....	52

ВИСНОВКИ.....	54
СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ.....	55

## СПИСОК УМОВНИХ ТЕРМІНІВ ТА ПОЗНАЧЕНЬ

Ajax – Asynchronous JavaScript and XML (Асинхронний JavaScript і XML)

API – Application Programming Interface (Інтерфейс прикладного програмування)

CMS – Content Management System (Система управління контентом)

CSS – Cascading Style Sheets (Каскадні таблиці стилів)

HTML – Hyper Text Markup Language (Мова розмітки гіпертексту)

HTTP – Hyper Text Transfer Protocol (Протокол передачі гіпертексту)

HTTPS – Hyper Text Transfer Protocol Secure (Захищений протокол передачі гіпертексту)

PHP – Hypertext Preprocessor (гіпертекстовий препроцесор – скриптова мова програмування)

SEO – Search Engine Optimization (Пошукова оптимізація)

SQL – Structured Query Language (Мова Структурованих Запитів)

XML – Extensible Markup Language (Розширювана мова розмітки)

## ВСТУП

Актуальний стан проблеми. Однією з основних складових навчального процесу є розклад занять, який регламентує освітній процес денної та заочної форм навчання всіх освітніх рівнів, регулює навчальні заняття за днями тижня, курсами, групами та місцем проведення, а також забезпечує логічну послідовність у засвоєнні знань та практичних навичок здобувачів вищої освіти. Тому, редагувати та отримувати доступ до розкладу потрібно якомога швидше та зручніше.

Існують різні способи подання розкладу для студентів та викладачів. Найбільш простий та вживаний друкований розклад, який можна переглянути на факультеті.

В сучасних реаліях, коли світ накрила пандемія COVID-19, і ЗВО переходять на змішану та дистанційну форми навчання досить актуальною є проблема організації навчання здобувачів вищої освіти. Важливим аспектом є легкий доступ викладачів і студентів до розкладу занять. Тому актуальність розробки web-застосунку, який зробить візуалізацію розкладу занять доступною через мережу Інтернет є незаперечною.

Об'єкт досліджень. Розклад навчальних занять.

Предмет досліджень. Інформаційна система для вирішення задачі формування розкладу занять в університеті та доступу до нього через мережу Інтернет.

Методи дослідження. Для вирішення проблеми в даній роботі використовуються методи аналізу і синтезу, порівняння, логічного узагальнення результатів.

Мета. Розробити інформаційну систему «Розклад занять» для формування, коригування даних для розкладу занять університету.

Структура роботи. Робота складається з списку умовних позначень, трьох розділів, в яких описуються сучасні web-застосунки, аналіз різних ІС

«Розклад занять» та вибір програмних засобів і проектування власної інформаційної системи. Робота має 56 сторінок, 16 джерел, 42 рисунка, 7 таблиць.

*Апробація роботи (Додаток А).* Результати дослідження доповідалися на звітній науковій конференції викладачів, співробітників і здобувачів вищої освіти Рівненського державного гуманітарного університету за 2020 рік та на XIV Всеукраїнській науково-практичній конференції «ІНФОРМАЦІЙНІ ТЕХНОЛОГІЇ В ПРОФЕСІЙНІЙ ДІЯЛЬНОСТІ», тези доповіді опубліковані у збірнику [16].

## РОЗДІЛ 1. ОГЛЯД СУЧАСНИХ WEB-ЗАСТОСУНКІВ

### 1.1. Web-застосунок. Технічні особливості

Web-застосунок — клієнт-серверний додаток, в якому клієнт взаємодіє з веб-сервером за допомогою браузера. Логіка роботи розподіляється між сервером та клієнтом, зберігання даних відбувається на сервері, обмін інформацією здійснюється по мережі. Перевагою є те, що клієнти ніяк не залежать від операційної системи пристрою, тому web-застосунок являє собою деяку кросплатформлену службу. Широко застосовуватись вони почали в кінці 1990-х – початок 2000-х років.

Велика перевага побудови web-застосунків для підтримки стандартних функцій браузера є те, що робота застосунка повинна виконуватися в будь-якому випадку на різних операційних системах. Замість того, щоб створювати різні версії для Microsoft Windows, Mac OS X, GNU/Linux та інших систем, застосунок створюється один раз для довільно вибраної платформи і на ній розгортається. Звичайно ж різна реалізація, CSS, DOM та інших специфікацій в браузерах може викликати проблеми під час розробки web-застосунків і майбутньої підтримки. Окрім цього, можливість користувача налаштовувати браузер «під себе» (наприклад: розмір шрифту, колір, вимкнення сценаріїв підтримки) може перешкоджати успішній роботі програми [1].

Наступний підхід полягає у використанні Adobe Flash, Silverlight або прикладних програм Java для повної або часткової реалізації інтерфейсу користувача. Оскільки більшість браузерів підтримує ці технології (за допомогою плагінів) Flash- або Java-додатки можуть виконуватися з легкістю. Так як вони надають програмісту великий контроль над інтерфейсом, вони здатні вирішувати велику кількість відмінностей в конфігураціях браузерів, але нусумістність між Java- або Flash-реалізаціями на стороні клієнта може послугувати великою кількістю ускладнень.

На 2015 рік технологію Adobe Flash не підтримують Chrome, Safari та інші не менш популярні браузері.

У зв'язку з архітектурним збігом з традиційним клієнт-серверним застосунком, існують спори відносно коректності віднесенням подібних систем до веб-додатків.

Веб-застосунок являє собою веб-сайт, на якому розміщені сторінки з частковим або повністю несформованим вмістом. Закінчений вміст формується тільки після того, як відвідувач сайта перейде на сторінку з веб-сервера. У зв'язку з тим що кінцевий контент сторінки залежить від запиту, створеного на основі дій відвідувача, така сторінка називається динамічною. Спектр використання веб-застосунків дуже великий [1].

Статичний веб-сайт містить в собі набір відповідних HTML-сторінок і файлів, розміщених на комп'ютері, на якому встановлений та налаштований веб-сервер.

Веб-сервер — це програмне забезпечення, яке надає веб-сторінки у відповідь на запити веб-браузера. Зазвичай запит сторінок створюється при натисканні посилання на веб-сторінці, виборі закладки в браузері або введенні URL-адреси в адресний рядок браузера.

## **1.2. Типи веб-застосунків**

Відштовхуючись від того як веб-застосунки показують контент, який вони приймають, їх можна поділити на 6 різних типів.

### **1.2.1. Статичні веб-додатки**

Якщо Ви вирішите створити статичний веб-застосунок, то перше, що потрібно знати, це те, що цей тип веб-застосунків показує дуже мало контенту і є не сильно гнучким.

Статичні веб-застосунок зазвичай розробляються на HTML та CSS, але це не єдині платформи для розробки статичного застосунку. Ви можете використовувати jQuery та Ajax на власний розсуд. Ви також можете легко



включати або показувати об'єкти з анімацією, такі як банери, GIF-файли, відео і т.д.

На жаль, змінити вміст статичних веб-застосунків не так просто. Для початку потрібно загрузити HTML код, потім змінити його і, на кінець, відправити назад на сервер. Ці зміни можуть бути здійснені тільки веб-програмістом або компанією-розробником, яка спланувала і розробила перший застосунок [2].

Приклади розробки статичних веб-застосунків включають в себе професійне портфоліо або цифрове резюме. Аналогічно, сторінка, яка представляє компанію, також може використовувати веб-застосунок цього типу для відображення їхньої контактної інформації.

### **1.2.2. Динамічні веб-додатки**

Динамічні веб-застосунки набагато складніші на технічному рівні. Вони використовують бази даних для загрузки даних, і їхній вміст оновлюється кожний раз, коли користувач звертається до них. Зазвичай вони мають панель адміністрування (названа CMS), де адміністратори можуть виправляти або змінювати вміст застосунка, будь то текст або зображення.

Різні мови програмування можуть використовуватися для розробки динамічних веб-застосунків. PHP і ASP є найбільш поширеними мовами, використовуваними для цього завдання, оскільки вони дозволяють структурувати вміст.

В даному типі застосунку оновлення вмісту дуже просто і не потрібний навіть доступ до сервера для внесення змін. Крім цього, він дозволяє реалізувати безліч функцій, таких як формули або бази даних. В застосунок до цього проект може бути змінений в залежності від уподобань адміністратора.

### **1.2.3. Інтернет-магазин або електронна комерція**

Якщо веб-застосунок являє собою інтернет-магазин, його розробка, скоріш за все, буде нагадувати розробку сайта електронної комерції. Цей тип процесу розробки застосунків є більш важчим, оскільки він повинен дозволяти

електронні платежі, які можуть бути зроблені за допомогою кредитних карт, PayPal або інших способів оплати. Розробник також повинен створити панель керування для адміністратора; він буде використовуватися для обрахунку нової продукції, їх оновлення, видалення записів та управлінням застосунком і платежами [3].

Ваш веб-застосунок встановлюється на мобільні пристрої так само, як мобільний додаток, що дозволяє взаємодіяти з ним.

#### **1.2.4. Портал (веб-застосунок)**

Через портал я маю на увазі тип застосунку, яке звертається до різних розділів або категорій через домашню сторінку. Ці застосунки можуть включати в себе багато речей: форуми, чати, електронна скринька, браузері тощо.

#### **1.2.5. Анімовані веб-застосунки**

Анімація неминуче пов'язана з технологією Flash. Цей програмний підхід дозволяє відображати контент з ефектами анімації. Даний тип застосунків дозволяє створювати більш креативний і сучасний дизайн та являє собою одну з ключових технологій, яку використовують дизайнери і креативні директори. Недолік, властивий розробці веб-застосунків, заключається в тому, що цей тип технології не підходить для позиціонування в Інтернеті та оптимізації SEO, оскільки пошукові системи не можуть правильно читати інформацію, яку вони містять.

#### **1.2.6. Веб-застосунки з системою керування (CMS)**

Контент сторінки повинен постійно оновлюватись, коли мова заходить про розробку веб-застосунків, тому установка системи керування контентом (CMS) має високий пріоритет для розгляду. Адміністратор може використовувати цю CMS тільки для внесення змін та оновлень.

Ці системи керування контентом зрозумілі навіть для початкових користувачів і прості у використанні. Деякі приклади систем керування контентом:

- WordPress: безсумнівно, найбільш поширений спосіб керування контентом. В Інтернеті є багато інформації, навчальних посібників та статей, які допоможуть Вам налаштувати його під свої задачі та цілі, і зрозуміти, як він працює. Крім цього, він є безкоштовним для використання, але має платні модулі для більш обширного використання даної системи.
- Joomla: ця CMS поступається тільки WordPress. Не так багато користувачі, порівняно з вище згаданою системою, але у неї сильне співтовариство, і вона дуже інтуїтивно зрозуміла.
- Drupal: це безкоштовне програмне забезпечення CMS – дуже адаптивне і особливо рекомендується для створення спільнот.

Даний тип веб-застосунків дуже розповсюджений серед контентних сторінок: особисті блоги, корпоративні блоги, професійні блоги, сторінки новин, статті, ЗМІ і т.д [4].

Кожний тип веб-застосунків має свої сильні та слабкі сторони, але давайте не будемо забувати, що, в кінцевому вигляді, він залишається веб-сайтом. Вони не являються рідними застосунками, незалежно від того, наскільки вони схожі на них (це буде залежати від проекту веб-застосунка, а не від його розробки). Вам необхідно буде дотримуватися правил використання файлів cookie і збільшити безпеку застосунків від можливих хакерських атак – аналогічно, як це робиться на інших веб-сайтах.

Також не забувайте привально звертатися до SEO, оскільки ваш веб-застосунок буде відображатися в результатах пошукових систем, таких як Google, Yahoo або подібним.

Одним із сучасних типів веб-застосунків є CRM система. Це система управління проектами, направлена на автоматизацію обробки повного спектру інформації про клієнта та продукції. Подібні рішення — це комплексний продукт, який об'єднує функції баз даних, почти, календаря, обліку фінансів та інше. В них можуть бути інтегровані, в залежності від потреб, різні модулі: звіти, бухгалтерія, облік кадрів і т.д.

CRM називають основою бізнеса телемаркетингових компаній і кол-центрів. Незамінні, коли потрібно налаштувати проектну роботу з чітким поділом по ролям і зонам відповідальності, взаємодія між відділами, робота з клієнтами. Це актуально для банків, агентств маркетингових комунікацій, компаній-розробників ІТ, онлайн-магазинів товарів та послуг.

Більш заточений під потреби конкретного бізнеса — це ERP. Це web-застосунки, розроблені для автоматизації процесів управління внутрішньогосподарської діяльності великих компаній з розвиненою філіальною мережею, різними напрямками діяльності. Включає в собі модулі виробничого, фінансового управління, закупки і т.д [5].

### **1.3. Архітектура веб-застосунків та застосування їх у різних сферах життя**

Архітектура веб-застосунків визначає взаємодію між застосунками, системами проміжного програмного забезпечення і базами даних, щоб забезпечити спільну роботу декількох додатків. Коли користувач вводить URL-адрес і натискає «Перейти», браузер виявляє комп'ютер, підключений до Інтернету, на якому працює веб-сайт, і робить запит на цю конкретну сторінку.

Потім сервер відповідає, відправляє файли в браузер. Після цієї дії браузер виконує ці файли, щоб показати користувачу шукану сторінку. В результаті виконаних дій, користувач отримує можливість взаємодіяти з сайтом. Звичайно, всі ці дії виконуються протягом декількох секунд (Рис 1.1.).

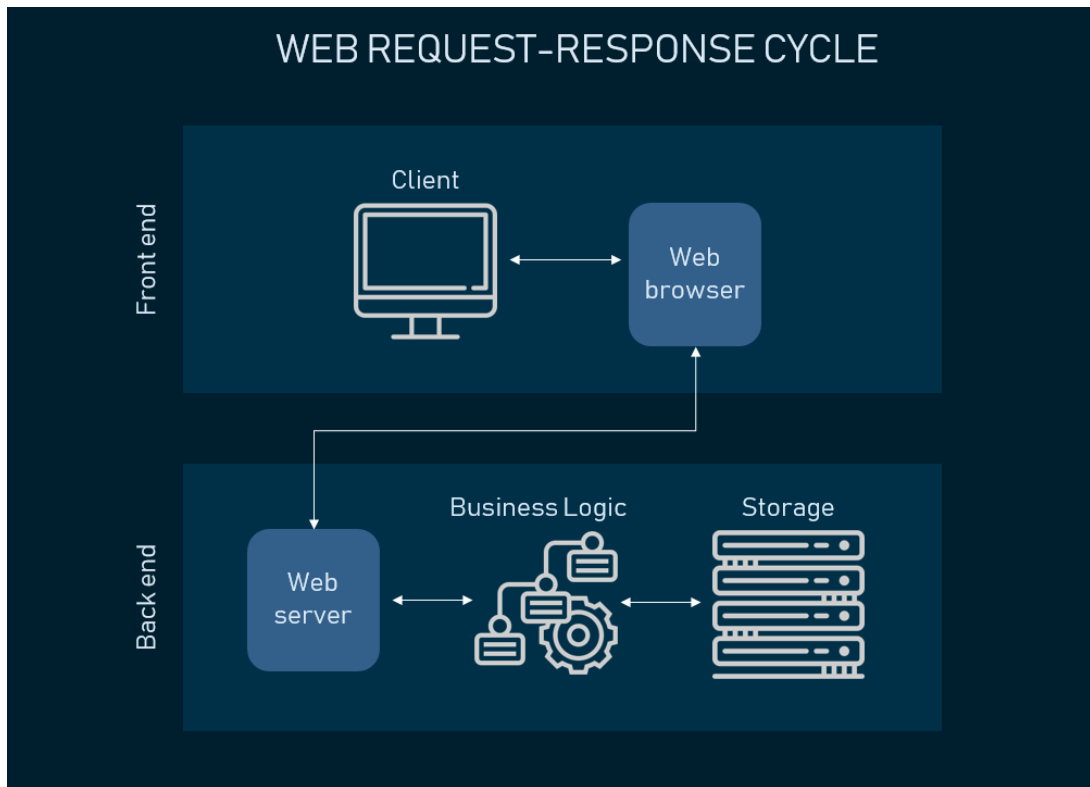


Рис 1.1. Схема роботи клієнта і сервера

Тут важливий код, який отримує і аналізує браузер. Цей самий код може мати або не мати конкретні інструкції по яким браузер розуміє як реагувати на широкий діапазон вхідних даних. В результаті архітектура веб-застосунка включає в себе всі підкомпоненти і взаємозамінність зовнішніх застосунків для загального програмного додатку (Рис 1.2.). По суті, одночасно працює дві програми:

- Код, який знаходиться в браузері і відповідає на введення користувача
- Код, який знаходиться на сервері і відповідає на HTTP-запити

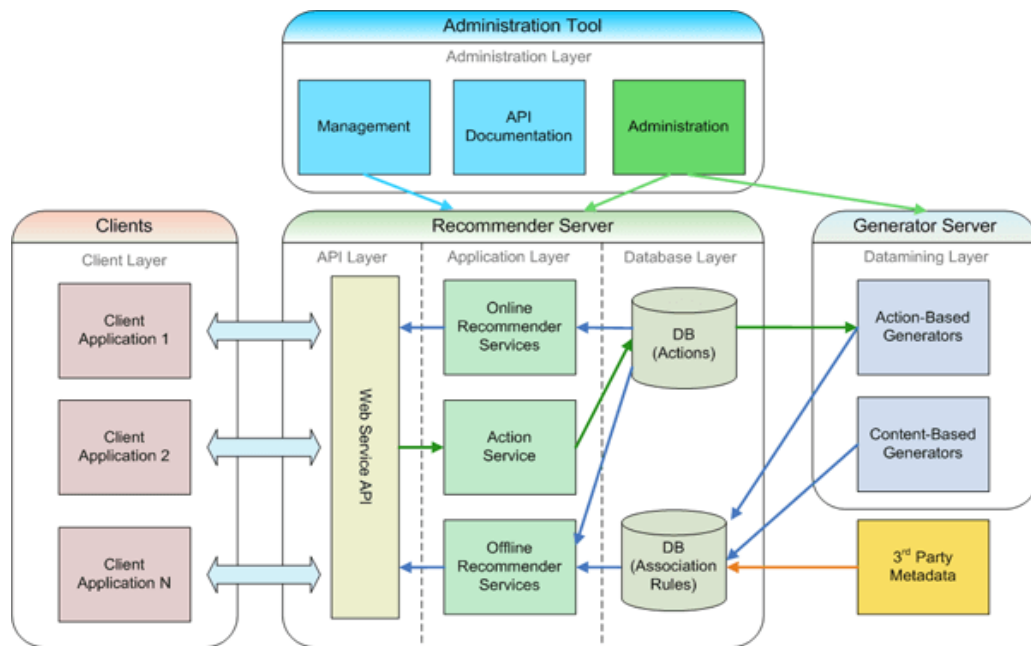


Рис 1.2. Схема архітектури роботи web-застосунка

Під час написання застосунка веб-розробник повинен знати, що буде робити код на сервері по відношенню до того, що повинен робити код в браузері. Мови програмування, які мають серверний код:

- Ruby on Rails
- PHP
- C#
- Java
- Python
- Javascript

По факту, будь-який код, який може відповідати на HTTP-запити, може працювати на сервері.

З клієнтським кодом використовуються наступні мови програмування:

- CSS
- Javascript
- HTML

Оскільки технології продовжують розвиток, архітектура веб-застосунків — також. Одним з таких напрямків є використання та створення сервіс-орієнтованої архітектури. Саме тут велика частина кода всього додатку існує у вигляді сервісів. Крім того, у кожного є свій власний HTTP API. Результатом цього — один аспект кода може відправити запит другій частині кода, яка може виконуватися на іншому сервері.

Ще одна тенденція — односторінковий застосунок. Це де веб-інтерфейс представлений через великомасштабний Javascript. Потім він залишається в браузері користувача через різні взаємодії. Що стосується запитів, він використовує AJAX або WebSockets для виконання асинхронних або синхронних запитів до веб-сервера без необхідності завантаження сторінки.

За своїм змістом багато веб-застосунків побудовані навколо об'єктів. Об'єкти зберігаються в таблицях через базу даних SQL. Кожний рядок в таблиці має певний запис. Ви можете викликати запис, просто вказавши рядок і стовпець для цільової точки даних.

З двома вищезгаданими тенденціями веб-застосунка тепер набагато краще підходять для перегляду на різних платформах і на різних пристроях. Навіть якщо велика частина кода для додатка залишається старою, їх всерівно можна легко і ясно переглядати на невеликому екрані [6].

У вас може бути робочий застосунок, але він так само повинен мати хорошу веб-архітектуру.

## РОЗДІЛ 2. ОГЛЯД ІСНУЮЧИХ ПРОГРАМНИХ РІШЕНЬ ЕЛЕКТРОННОГО РОЗКЛАДУ ЗАНЯТЬ

Існує безліч веб-ресурсів «Розклад занять» для різних університетів як України, так і інших країн, в яких кожен студент зможе переглянути свій розклад занять онлайн і буде завжди знати, які в нього пари мають бути в майбутньому. Дані веб-ресурси зменшили роботу викладачів та деканатів університетів. Тепер їм не потрібно друкувати таблиці розкладу занять на принтері і вішати їх на головну афішу.

Після вивчення ринку технологій і пошуку схожого програмного забезпечення для управління розкладом занять були винесені наступні веб-ресурси: Розклад занять Житомирської політехніки, НУВГП Desk, КПІ Desk.

### **2.1. Розклад занять Житомирської політехніки**

Першим з розглянутих веб-ресурсів був «Розклад занять Житомирської політехніки». На сьогоднішній день важко уявити собі хоч один університет без власної створеної системи розкладу занять. Оскільки ця система дуже потрібна кожному працівнику університету, а також студенту. Вона зменшує кількість рутинної роботи працівників.

На головній сторінці веб-ресурсу «Розклад занять Житомирської політехніки» відображені головні новини і далі після них список факультетів по курсам. В кожному курсі знаходяться групи. Якщо натиснути на назву групи, то потрапимо на сторінку розкладу занять даної групи (Рис. 2.1).

Розклад занять має вигляд двох таблиць. Перша таблиця відображає перший тиждень навчання, друга таблиця відображає другий тиждень навчання. Рядки таблиць – це нумерація пари і години відведені для неї. Стовпці – назва дня тижня. Кожна комірка містить у собі: назву групи, назву предмета, вид пари (лекція або практика), номер аудиторії та назву викладача (Рис. 2.2).



Розклад занять Житомирської політехніки

Аспіранти    Викладачі    Аудиторія

## Розклад занять академічних груп на I семестр 2021/22 н.р.

**Увага!** Розклад за вибірковими навчальними дисциплінами загальної підготовки для 3-го курсу

**Увага!** Розклад за вибірковими навчальними дисциплінами загальної підготовки для 4-го курсу

У випадку виявлення помилок або при виникненні запитань щодо розкладу просимо звертатися на електронну пошту [morozov@ztu.edu.ua](mailto:morozov@ztu.edu.ua)

Шановні студенти та викладачі! Враховуючи стан епідемічної небезпеки та загальнодержавні протиепідемічні заходи, відповідно до листа Міністерства освіти і науки України від 21 жовтня 2021 року № 1/9-558 «Про організацію освітнього процесу в закладах освіти під час епідемічної небезпеки» з 25 жовтня 2021 року по 15 листопада 2021 року у Державному університеті «Житомирська політехніка» запроваджується проведення усіх видів навчальних занять в академічних групах усіх курсів, освітніх ступенів і форм навчання з використанням технологій дистанційного навчання.

### Гірничо-екологічний факультет

1 курс	2 курс	3 курс	4 курс	Магістри 1 курс
ГТ-28к	ГТ-28	ГТ-27	ГТ-26	ГТ-29м
ГТ-28к	ГР-1мб	ГТ-27к	ЕО-38	ЕО-37м
ГТ-29	ЕО-1мб	ЕО-39	РР-47	РР-46м
ГТ-29к	ЕО-40	РР-48	ТЗНС-38	ТЗНС-37м
ГР-2мб	НЗ-1	РР-48к		
Додатково	РР-49	ТЗНС-39		
ЕО-2мб	РР-49к	ТЗНС-39к		

Рис. 2.1. Головна сторінка веб-ресурсу «Розклад занять Житомирської політехніки»

Розклад занять Житомирської політехніки

## Розклад групи ГР-1мб

### I тижень

	Понеділок	Вівторок	Середа	Четвер	П'ятниця
8:30-9:59	ГР-1мб Основи економіки гірничого виробництва Практика ауд. Дистанційно Крайцеев Ганна Вікторівна	Вікторія ГР-1мб Історія розвитку геології та гірничої справи Практика ауд. Дистанційно Півесюцький Віктор Тобіолович	Середа ГР-1мб, РР-48, РР-51к Фізичне виховання Практика ауд. Дистанційно Крулені Ольга Миколаївна, Петренко Іван Іванович		ГР-1мб Пошуки та розвідка родовищ корисних копалин Лещяк ауд. Дистанційно Півесюцький Віктор Тобіолович
10:00-11:20	ГР-1мб Маршайдерська справа Практика ауд. Дистанційно Ковалевч Лейбміла Анатолівна	ГР-1мб, РР-49, РР-51к Гіричне креслення Практика ауд. Дистанційно Башинський Сергій Іванович	ГТ-28, ГТ-29к, ГР-1мб, РР-49, РР-50, РР-51к Прикладна механіка Лещяк ауд. Дистанційно Шостакун Андрій Миколаєвич	ГР-1мб Основи економіки гірничого виробництва Лещяк ауд. Дистанційно Крайцеев Ганна Вікторівна	ГР-1мб, РР-49, РР-51к Гіричне креслення Практика ауд. Дистанційно Башинський Сергій Іванович
11:40-13:30		ГТ-28, ГТ-29к, ГР-1мб, РР-49, РР-50, РР-51к Відкриті гірничі роботи Лещяк ауд. Дистанційно Коробічкін Валентин Валерійович		ГР-1мб Історія розвитку геології та гірничої справи Лещяк ауд. Дистанційно Півесюцький Віктор Тобіолович	ГР-1мб Відкриті гірничі роботи Практика ауд. Дистанційно Кальчук Сергій Володимирович
13:50-14:50		ГР-1мб Маршайдерська справа Лещяк ауд. Дистанційно Ісакєв Сергій Степанович, Ковалевч Лейбміла Анатолівна		ГР-1мб Пошуки та розвідка родовищ корисних копалин Практика ауд. Дистанційно Півесюцький Віктор Тобіолович	
15:00-16:20					
16:30-17:50					
18:00-19:20					

### II тижень

	Понеділок	Вівторок	Середа	Четвер	П'ятниця
8:30-9:59	ГР-1мб Основи економіки гірничого виробництва Практика ауд. Дистанційно Крайцеев Ганна Вікторівна	Вікторія ГР-1мб Пошуки та розвідка родовищ корисних копалин Практика ауд. Дистанційно Півесюцький Віктор Тобіолович	Середа ГТ-28, ГТ-29к, ГР-1мб, РР-49, РР-50, РР-51к Прикладна механіка Лещяк ауд. Дистанційно Шостакун Андрій Миколаєвич	ГР-1мб Маршайдерська справа Лещяк ауд. Дистанційно Ісакєв Сергій Степанович, Ковалевч Лейбміла Анатолівна	ГР-1мб Пошуки та розвідка родовищ корисних копалин Лещяк ауд. Дистанційно Півесюцький Віктор Тобіолович
10:00-11:20	ГР-1мб Політологія Практика ауд. Дистанційно	ГР-1мб Політологія Практика ауд. Дистанційно	ГТ-28, ГТ-29к, ГР-1мб, РР-49, РР-50, РР-51к Відкриті гірничі роботи Лещяк ауд. Дистанційно	ГР-1мб, РР-49, РР-51к Фізичне виховання Практика ауд. Дистанційно	ГР-1мб Прикладна механіка Практика ауд. Дистанційно

Рис. 2.2. Сторінка розкладу занять

Підсумовуючи, можемо сказати, що даний веб-ресурс немає динамічного пошуку по викладачам та факультетами. Це погано впливає на зручність пошуку свого розкладу і робить дану систему не простою у використанні.

## 2.2. КПП Desk

Наступним аналогом системи управління розкладом занять розглянемо веб-ресурс «КПП Desk».

Відвідавши даний ресурс відразу було помітно швидкість роботи і мінімалістичний дизайн сайту. Головна сторінка містить три кнопки: «Розклад занять», «Розклад сесії» та «Розклад для викладачів». Також на даній сторінці ми відразу бачимо поле для вводу назви своєї групи (Рис. 2.3).

Після вводу назви групи і натиснення кнопки «Розклад занять», завантажуються сторінка з розкладом занять. Вона містить дві таблиці: «Перший тиждень» та «Другий тиждень». Перший стовбець таблиць містить номер та годину початку заняття. Перший рядок – назви днів тижня. Кожна комірка таблиць має назву дисципліни, П.І.Б. викладача та номер аудиторії (Рис. 2.4).

Якщо перейти на вкладку «Розклад сесії» даного веб-ресурсу, можна побачити поле для вводу назви групи (Рис. 2.5).. Після вводу назви групи і натиснення кнопки «Розклад сесії» завантажуються сторінка з розкладом сесії по нашій введеній групі (Рис. 2.6). Дана сторінка містить таблицю, яка має два стовпця. Перший стовбець – дата екзамена чи заліка. Другий – назва дисципліни, аудиторія, П.І.Б. викладача і початок заходу.

При переході на останню вкладку «Розклад для викладачів», завантажуються сторінка, яка має поле для вводу П.І.Б. викладача і кнопку «Розклад занять» (Рис 2.7). Після введення П.І.Б. викладача і натиснення кнопки, завантажуються таблиці, ідентичні до розкладу занять студентів. Вони також поділяються на перший і другий тиждень. Містять ту саму інформацію, але по конкретному викладачу (Рис 2.8).

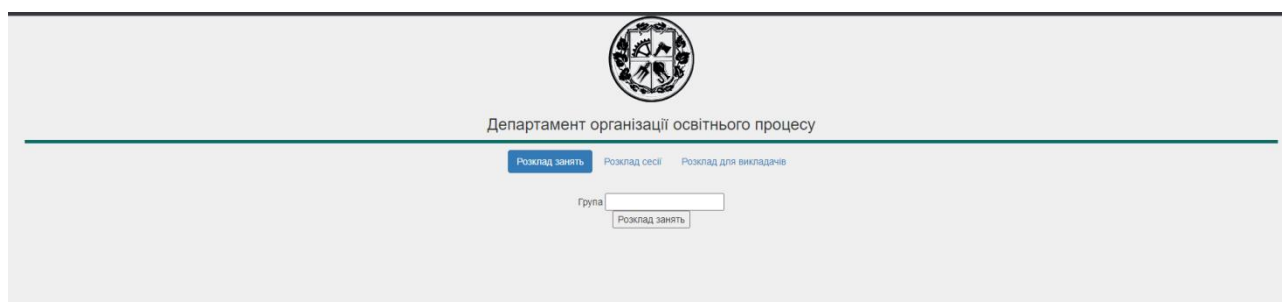



Рис. 2.3 – Головна сторінка веб-ресурсу «КПІ Desk»

Перший тиждень						
	Понеділок	Вівторок	Середа	Четвер	П'ятниця	Субота
1 08:30	ТАК 1 Теор зас доц. Писарець А. В. 317-01 Лек on-line	Комп модел проц та сист-1 ст.вик. Толочко Т. О. 168-01 Прак	Перетвор фіз величин проф. Безвесьільна О. М. 317-01 Лек on-line	Правознавство ст.вик. Костенко І. В. 1-201-01 Лек on-line	Перетвор фіз величин проф. Безвесьільна О. М. 170а-01 Прак	
2 10:25	Електр та осн мікропроц техн 2 доц. Чепюк Л. О. 170-01 Лек on-line	Електр та осн мікропроц техн 2 доц. Чепюк Л. О. 170-01 Прак	Вироб прилад ас. Котляр С. С. 170-01 Лек on-line		Сис CAD/CAE CATIA ас. Котляр С. С. 170-01 Прак	
3 12:20	Комп модел проц та сист-1 проф. Черепанська І. Ю. 317-01 Лек on-line	ТАК 1 Теор зас доц. Писарець А. В. 168-01 Прак	Вироб прилад ас. Котляр С. С. 170-01 Лек on-line		Сис CAD/CAE CATIA ас. Котляр С. С. 170-01 Прак	
4 14:15	Технол вир прил доц. Шевченко В. В. 08-01 Лек on-line	Технол вир прил доц. Шевченко В. В. 08-01 Прак				
5 16:10						
6 18:30						

Другий тиждень						
	Понеділок	Вівторок	Середа	Четвер	П'ятниця	Субота
1 08:30	ТАК 1 Теор зас доц. Писарець А. В. 317-01 Лек on-line	Комп модел проц та сист-1 ст.вик. Толочко Т. О. 168-01 Прак	Перетвор фіз величин проф. Безвесьільна О. М. 317-01 Лек on-line		Перетвор фіз величин проф. Безвесьільна О. М. 170а-01 Лек	
2 10:25	Електр та осн мікропроц техн 2 доц. Чепюк Л. О. 170-01 Лек on-line	Електр та осн мікропроц техн 2 доц. Повщенко О. А. 511-20 Лек	Вироб прилад ас. Котляр С. С. 170-01 Лек on-line	Правознавство ст.вик. Костенко І. В. 901-21 Прак	Сис CAD/CAE CATIA ас. Котляр С. С. 170-01 Прак	
3 12:20	Сис CAD/CAE CATIA ас. Котляр С. С.			Ін мов проф спрям - 1. Практ урс І вик. Амерідзе О. С.	Вироб прилад ас. Котляр С. С.	

Рис. 2.4 – Сторінка розкладу занять студента



Департамент організації освітнього процесу


---

Розклад занять [Розклад сесії](#) [Розклад для викладачів](#)

Група

[Розклад сесії](#)

Рис. 2.5 – Сторінка «Розклад сесії»



Департамент організації освітнього процесу

---

Розклад занять [Розклад сесії](#) [Розклад для викладачів](#)

Розклад сесії: ПМ-91

Перший семестр

01/01/2022	
01/02/2022	
01/03/2022	
01/04/2022	
01/05/2022	Перетворення фізичних величин проф. Безвесьільна О. М. 317-01 09:00
01/06/2022	
01/07/2022	
01/08/2022	
01/09/2022	
01/10/2022	Електроніка та основи мікропроцесорної техніки - 2 доц. Чепюк Л. О. 170-01 09:00
01/11/2022	
01/12/2022	
01/13/2022	
01/14/2022	Технологія виробництва приладів доц. Шевченко В. В. 08-01 09:00
01/15/2022	

Рис. 2.6 – Розклад сесії для конкретної групи

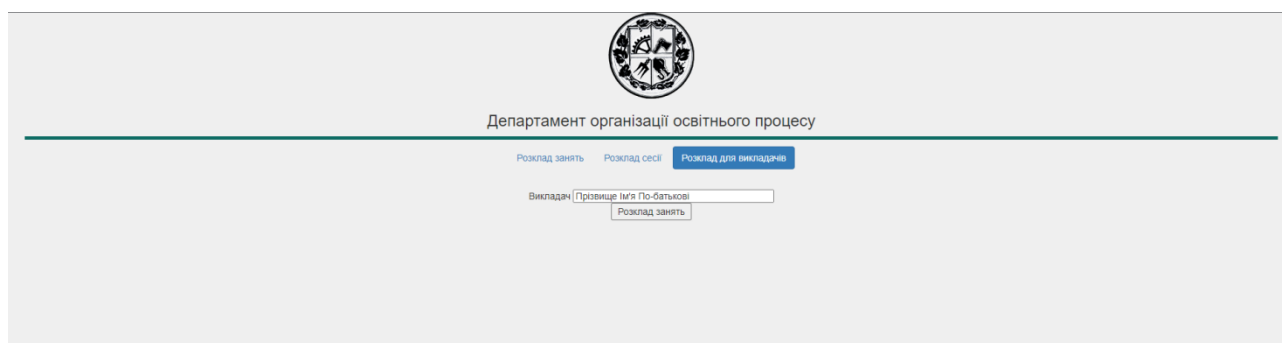


Рис. 2.7 – Сторінка «Розклад для викладачів»

Перший тиждень				
	Понеділок	Вівторок	Середа	Четвер
1 08:30			Загальна фізика-1 ас. Керіта О. О. 209-30 Прак Т1-12	Загальна фізика-1 ас. Керіта О. О. 209-30 Прак Т1-11
2 10:25			Загальна фізика-1 ас. Керіта О. О. 209-30 Прак Т1-12	Загальна фізика-1 ас. Керіта О. О. 209-30 Прак Т3-12
3 12:20			Загальна фізика-1 ас. Керіта О. О. 209-30 Прак Т3-11	Загальна фізика-1 ас. Керіта О. О. 220-07 Лаб Т1-11
4 14:15			Загальна фізика-1 ас. Керіта О. О. 209-30 Прак Т1-11	Загальна фізика-1 ас. Керіта О. О. 220-07 Лаб Т3-11
Другий тиждень				
	Понеділок	Вівторок	Середа	
1 08:30				
2 10:25				
3 12:20				
4				

Рис. 2.8 – Розклад занять для конкретного викладача

Отже, «КП Desk» має більший функціонал ніж попередній аналог. Система швидко працює і має простий та легкий для сприйняття дизайн. Але дана система має пошук тільки по назві групи, через це вона не є динамічною.

### 2.3. НУВГП Desk

Останнім аналогом системи управління розкладом занять розглянемо «НУВГП Desk».

Головна сторінка даного веб-ресурсу має найбільший функціонал з всіх вище описаних аналогів. Система має п'ять різновидів фільтрів, з яких два основних мають бути обов'язково заповненими (або назва групи, або П.І.Б. викладача): факультет, П.І.Б. викладача, назва групи, з дати (з якої дати завантажити розклад) та по дату (по яку дату завантажити розклад) (Рис. 2.9).

Після введення назви групи та натиснення кнопки «Показати розклад занять», завантажуються розклад занять по фільтрам, які я вибрав. Сторінка складається з декількох таблиць однієї і тієї ж структури. Кількість таблиць залежить від кількості днів вибраних у фільтрах. Тобто, одна таблиця – один день (Рис. 2.10). За замовчуванням завантажуються розклад занять з теперішнього дня на тиждень вперед. Над кожною таблицею є її назва. Вона складається з дати та назви дня тижня. Під назвою вже знаходиться сама таблиця з розкладом занять. Вона складається з трьох стовпців. Перший – номер заняття. Други – години заняття (початок - кінець). Третій – номер аудиторії, посада та П.І.Б. викладача, список назв груп або одна назва групи, назва дисципліни та в кінці в дужках тип заняття (лекція, практична чи лабораторна).

Якщо здійснити пошук по П.І.Б. викладача, то таблиці будуть ідентичні, але дані відображаються саме по шуканому викладачу (Рис. 2.11).

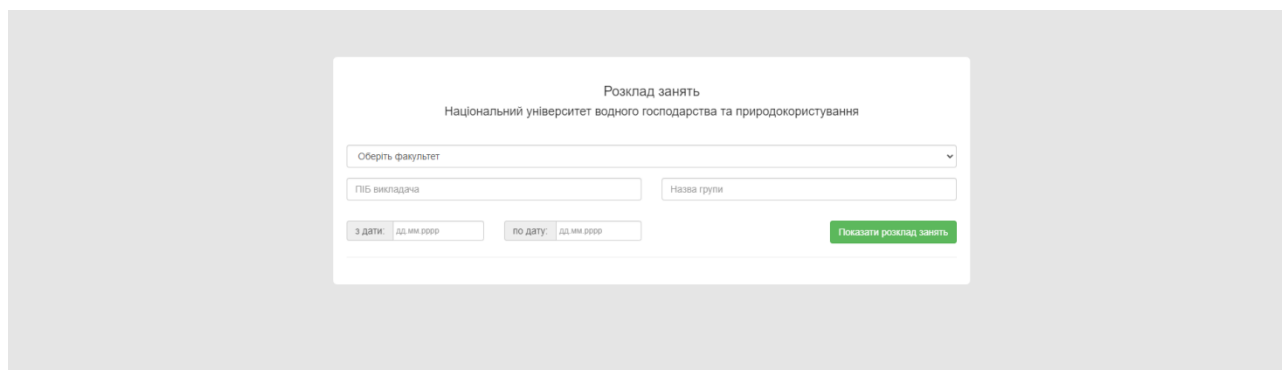


Рис. 2.9 – Головна сторінка веб-ресурсу «НУВГП Desk»

Розклад групи ПМ-41 з 30.10.2021 по 06.11.2021	
<b>01.11.2021 П'ятниця</b>	
1	08:00 - 09:20 Фізичне виховання
2	09:40 - 11:00
3	11:15 - 12:35 276 доцент Жувовська Наталя Анатоліївна Збірна група ПМ-21шт, ПМ-41, КН-11шт Теорія систем, системний аналіз та теорія графів/теорія решіток (Лаб)
4	13:00 - 14:20 276 доцент Жувовська Наталя Анатоліївна Збірна група ПМ4+2шт Математичне моделювання (Лаб)
5	14:35 - 15:55 доцент Поталчук Світлана Сергіївна Збірна група ЗПС-27 Комунікативний курс сучасної англійської мови (ГРС)
6	16:05 - 17:25
7	17:35 - 18:55
<b>02.11.2021 Вівторок</b>	
1	08:00 - 14:5(41) ст. викладач Герус Володимир Андрійович Збірна група ПМ4+2шт Архітектура обчислювальних систем та системне програмування (Лаб)
2	09:40 - 11:00
3	11:15 - 12:35
4	13:00 - 14:20
5	14:35 - 15:55
6	16:05 - 17:25
7	17:35 - 18:55
<b>03.11.2021 Середа</b>	
1	08:00 - 09:20
2	09:40 - 11:00
3	11:15 - 12:35 453 ст. викладач Герус Володимир Андрійович Потк КН-1шт, КН-31, ПМ-2шт, ПМ-41 Архітектура обчислювальних систем та системне програмування (Л)
4	13:00 - 14:1
<b>04.11.2021 Четвер</b>	
1	08:00 - 09:20
2	09:40 - 11:00
3	11:15 - 12:35 242 ст. викладач Зубик Ярослав Ярославович Збірна група ПМ4+2шт Бізнесові процеси та їх моделювання (Лаб)
4	13:00 - 14:20 223а ст. викладач Зубик Ярослав Ярославович Збірна

Рис. 2.10 – Сторінка розкладу занять для певної групи

Розклад викладача Жуковська Наталія Анатоліївна з 30.10.2021 по 06.11.2021

01.11.2021 Понеділок		02.11.2021 Вівторок	
1	08:00 09:20	1	08:00 09:20
2	09:40 276 Збірна група ІПЗЗ+21інт Теорія систем, системний аналіз та теорія прийняття рішень (Лаб)	2	09:40 414 11:00 КН-51м Проектування систем штучного інтелекту (Лаб)
3	11:15 276 12:35 Збірна група ПМ-21інт, ПМ-41, КН-11інт Теорія систем, системний аналіз та теорія прийняття рішень (Лаб)	3	11:15 12:35
4	13:00 276 14:20 Збірна група ПМ4+2інт Математичне моделювання (Лаб)	4	13:00 14:20
05.11.2021 П'ятниця			
1	08:00 658 09:20 Збірна група ПМ3+інт Рівняння математичної фізики (ПрС)		
2	09:40 658 11:00 Полк ПМ-11інт, ПМ-31 Рівняння математичної фізики (Л)		
3	11:15 658 12:35 Збірна група ПМ3+інт Рівняння математичної фізики (ПрС)		
4	13:00 14:20		

Рис. 2.11 – Розклад занять для певного викладача

Розглянута система є максимально наближеною до розроблюваної серед інших аналогів. Відмінність полягає в тому, що розроблювана система буде керуватися за допомогою admin панелі, в якій можна просто та швидко керувати даними розкладу занять.

Таким чином було проаналізовано три системи розкладу занять. Кожна з них має свої недоліки та переваги. На основі проведених аналізів було відібрано основні переваги систем та усвідомлено декілька недоліків, які впливають на швидкість роботи та простоту інтерфейсу. Тому доцільність розробки системи управління розкладом занять, стала ще більш пріоритетною.

## РОЗДІЛ 3. ВИБІР ПРОГРАМНИХ ЗАСОБІВ ТА РОЗРОБКА ІНФОРМАЦІЙНОЇ СИСТЕМИ

Одним із найважливіших завдань при розробці будь-якого веб-сервісу є вибір програмних засобів, які були б зручними для програміста і покращили б його роботу. На даний момент існує дуже багато різних мов програмування для створення сайтів (веб-ресурсів), а також великий вибір текстових редакторів для роботи з мовами програмування. Будь-який сайт складається з двох основних частин:

- Зовнішній вигляд сайту (front-end)
- Внутрішні функції сайту (back-end)

### **3.1. Front-end веб-ресурсу (зовнішній вигляд сайту)**

Під поняттям Front-end мається на увазі розробка видимого для користувача інтерфейса і всіх функцій, з якими він може взаємодіяти. По суті, коли ви переходите на будь-який сайт, то бачите там кнопки, текст, різні анімації і інші складники – це все реалізовано за допомогою фронтенда. Для створення цих елементів використовується три різних мови програмування – HTML, CSS та JavaScript [7].

Подивіться на структуру сторінки зараз – для цього потрібно на будь-якому сайті викликати консоль розробника, натиснувши кнопку F12. Ви побачите набір рядків кода, які формують зовнішній вигляд даної сторінки (Рис. 3.1).

Код сторінки використовується для опису всього видимого: шрифтів, зображень, розмірів елементів, анімацій, спливаючих вікон, звуків при натисненні і багато іншого, з чим може взаємодіяти користувач. За кожною із частин інтерфейсу відповідають різні мови програмування.

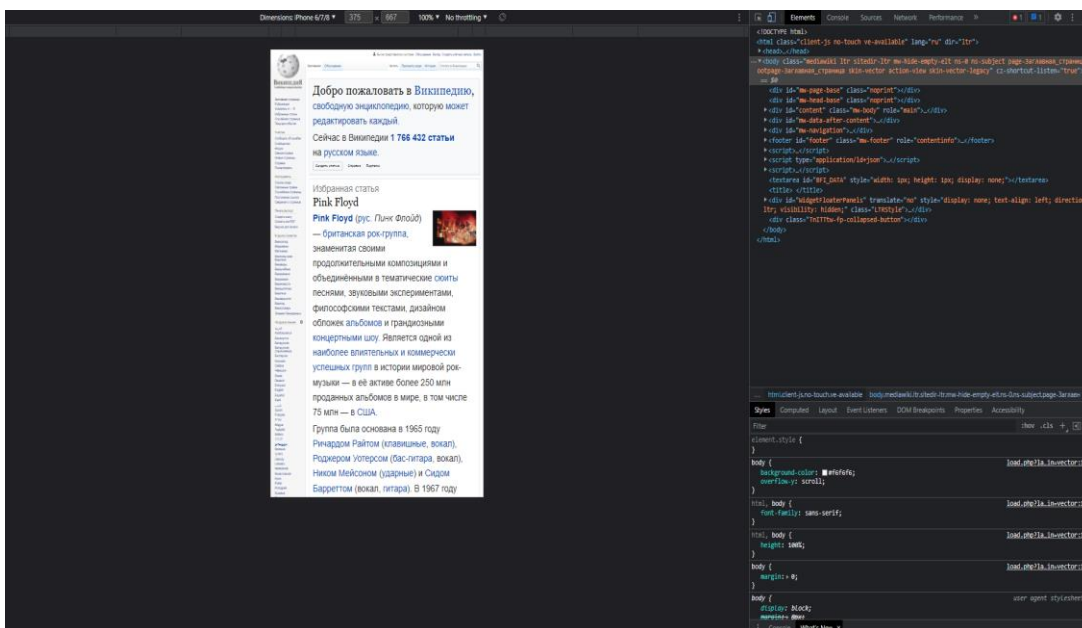


Рис. 3.1. Консоль розробника в браузері Google Chrome

### 3.1.1. Особливості використання HTML та CSS

Якщо ви думаєте про створення веб-сайта, ви напевно зустрічали аббревіатури HTML та CSS. Цих два основних інструмента складають основу будь-якої веб-сторінки. Хоча вони часто використовуються в поєднанні один з одним, вони служать зовсім різним цілям, коли справа доходить до веб-розробки [9].

Ключові відмінності знаходяться в їхній реалізації, простоті використання, їхніх різних функціях і в тому, як вони структуровані, і, в кінці кінців, і HTML, і CSS є ключами до того, як працює World Wide Web.

Давайте почнемо з того, що розберемося, що означає аббревіатура «HTML». HTML – це мова розмітки гіпертекста. Гіпертекст – це фактично текст в тексті, а мова розмітки – це мова, зрозуміла комп'ютерам, розроблена для опису веб-сторінок і підвищенням інтерактивності використовуваного тексту.

Найперше він був винайдений в 1989 році як мова публікацій в Інтернеті, і, простіше кажучи, це перший будівельний блок при створенні веб-сторінок.



Коли ви вперше познайомитесь з світом HTML, ви, мабуть, зіткнетесь з цілим рядом термінів і фраз, які теж трішки збивають з толку. Три основних – це елементи, теги та атрибути.

HTML5 – це остання «версія» HTML, і знання в даній області – зростаюча тенденція на ринку праці. По великому рахунку, він має ту саму функціональність, що і стандартний HTML, але набагато динамічніший і використовує набагато менше кода для створення чогось фантастичного.

По суті, HTML – використовується для створення основного вмісту веб-сторінки, надаючи йому певну структуру. Ви починаєте з написання слів, а потім застосовуєте до них теги або елементи. Після цього веб-браузер зчитує це і може зрозуміти заголовок сторінки, будь-які абзаци і місце початку і кінець сторінки, наповнюючи вашу веб-сторінку контентом.

HTML підтримується будь-яким браузером і встановлений майже на кожній існуючій веб-сторінці. Вам не потрібні ніякі ліцензії, вам не потрібно за це комусь платити, і його досить легко вивчити і написати свій код. Якщо ми можемо порівняти веб-сторінку з тілом людини, то HTML – це скелет тіла.

CSS – це аббревіатура від Cascade Styling Sheets. Коротше кажучи, це мова стилів таблиць, яка являє собою тип мови, який ви можете використовувати для опису представлення мови розмітки – в даному випадку для опису рухів HTML. Він ефективно визначає, як будівельні блоки, закладені в HTML, оформлюються і представляються користувачу [8].

CSS був вперше створений приблизно в 1996 році, щоб розібратись в HTML і створити веб-сторінку приголомшливою. Як і в випадку с будь-якою мовою, ви повинні писати CSS код, і знання того, як правильно писати CSS – це навик, який повинен мати будь-який розробник веб-сторінок. CSS простий в обслуговуванні і є другою частиною двокомпонентного набору інструментів для створення веб-сторінок.

Якщо HTML – це скелет тіла, то CSS – його оболонка. Він використовується для кольорів фону, стилів, макета, границь, затемнення – всіх основних елементів дизайну, які роблять веб-сторінку привабливою і розумною. CSS дозволяє розрізнити представлення і контент, змінюючи дизайн і відображення елементів HTML.

Презентація і простота використання – це пара основних речей, які CSS отримав для веб-дизайну, перекладаючи те, як контент виглядає на веб-сторінці, і що ще розміщується на ній, щоб доповнити цей контент. Хоч він частіше використовується разом з HTML, насправді він не залежить від нього і може використовуватись з будь-якою мовою розмітки на основі XML.

Багато людей вважають, що HTML і CSS – це одне і теж, але насправді це зовсім різні мови. Хоча вони часто використовуються разом в написанні коду і веб-дизайні, їх можна використовувати і окремо, і обидва мають багато різних додатків. Основні відмінності між HTML і CSS:

- HTML визначає зміст і структуру веб-сторінки, тоді як CSS переводить відображення і дизайн елементів HTML.
- В результаті HTML може містити в собі код CSS в своїх файлах, але CSS не може містити в собі HTML.
- HTML – це один встановлений «тип» або «метод», тоді як, в залежності від потреб, CSS може бути внутрішнім або зовнішнім.
- HTML не використовується для представлення або візуалізації.
- Коли справа доходить до написання кода, HTML має чіткий і простий для розуміння синтаксис, тоді як CSS може стати трішки складнішим і заплутаним.
- HTML використовує «тегі» по обидві сторони від вмісту, тоді як CSS використовує «селектори», за якими слідує блок оголошення.
- Код HTML, набагато більший, чим код CSS, що зменшує швидкість завантаження веб-сторінки CSS.

### 3.1.2. Сітка Bootstrap 4

Bootstrap – це найбільш популярний фреймворк HTML, CSS і JavaScript на планеті. Незавжди від того, чи ви є новачок або досвідченим веб-розробником, Bootstrap – потужний інструмент для будь-якого типу веб-додатків, які ви створюєте. З випуском версії 4 Bootstrap став більш актуальним, чим раніше, і пропонує повний набір компонентів, якими легко навчитися користуватися [10].

Є декілька причин використовувати Bootstrap. Окрім компонентів, які Bootstrap надає «з коробки», існує сотні інших сторонніх компонентів, з якими ви можете його інтегрувати, і велика спільнота інших розробників може вам допомогти.

Bootstrap також є потужним інструментом для створення прототипів в світі стартапів. Він дозволяє вам швидко створити прототип, щоб показати свою ідею, без великих затрат часу на створення інтерфейсу, яким ви, можливо в кінці кінців не будете використовувати. Велика частина конфігурації вже налаштована для вас, і ви можете швидше приступити до створення проекту.

З випуском Bootstrap 4 фреймворк перетерпів ряд ключових оновлень. Одна з самих великих змін – це перехід від Less, препроцесора CSS, до Sass. Коли Bootstrap тільки починав своє існування, в препроцесорах не було явного фаворита. За останні пару років Sass отримав невелику перевагу, через це цей перехід не повинен викликати здивувань. Якщо ви раніше не використовували Sass, не переживайте, він схожий на Less і дійсно легкий в освоєнні.

Ще одна велика нова функція в версії 4 – це покращена система сіток і включення flexbox. Для звичайної сітки був доданий ще один рівень сітки, що краще орієнтуватися на мобільних пристроях, а також перероблені медіа-запити. Flexbox – це сітка майбутнього, і дуже цікаво, що вона була включена. За замовчуванням звичайна сітка буде працювати не стандартно, але ви можете переключитися на сітку flexbox, переключивши просту змінну Sass, щоб використати переваги цього нового компонента макета [10].

Bootstrap 4 бачить застарілі компоненти, такі як колодязі, ескізи і панелі, а також введення нового компонента карти. Це добре по декільком причинам. Перш за все, він видаляє кілька схожих компонентів і замінює їх одним компонентом карти. Це робить структуру трішки легшою і легшою для вивчення для нового користувача. Компонент карти також став популярним останнім часом, тому є сенс включити його. Все, що потрібно зробити, - це подивитись на популярність Google Material Design, щоб зрозуміти, наскільки карти є відмінним компонентом для використання в веб-додатку.

Одна зміна, яку ви можете не помітити відразу, але тим паче – це покращення вбудованого скидання CSS. Bootstrap взяв файл `normalize.css` і розширив його новим модулем під назвою `Reboot`. `Reboot` покращує `Normalize` і посилює стиль браузера по замовчуванню, який необхідно скинути для всіх веб-проектів.

Підтримка Internet Explorer 8 припинено. Я дуже зрадів даній новині. Прийшов час залишити цей браузер в минулому.

Всі плагіни JavaScript, які входять разом з Bootstrap, були переписані в ES6, що дозволяє використовувати самі нові функції JavaScript. Компоненти спливаючої підказки і спливаючих вікон були розширені для використання бібліотеки `Tether`. Це тільки верхівка на поверхні, так як в фреймворк вбудовано багато інших незначних оновлень [11].

Щоб підключити Bootstrap 4, потрібно скачати з офіційного сайту архів і розархівувати в папку з проектом. Є швидший спосіб підключення. Скопіювати і вставити таблицю стилів `<link>` в свій `<head>` перед всіма іншими таблицями стилів, щоб завантажити наш CSS (Рис. 3.2).

```
<link rel="stylesheet" href="https://cdn.jsdelivr.net/npm/bootstrap@4.6.1/dist/css/bootstrap.min.css" integrity="sha384-zCbKRCUGaJDkqS1kPbPd7TveP5iyJE0EjAuZQTgFLD2yLzquKkdK1fG/eSrtxUkn" crossorigin="anonymous">
```

Рисю 3.2. Посилання для вставки в `<head>`

Багато з компонентів потребують використання JavaScript для роботи. Частково, для них потрібно jQuery, Popper і плагіни Bootstrap. Замість одного з

наступних `<script>` майже в кінці вашої сторінки, прям перед закриваючим тегом `</body>`, щоб ввімкнути їх. Спочатку повинен бути jQuery, потім Popper, а потім плагіни Bootstrap (Рис. 3.3).

```
<script src="https://cdn.jsdelivr.net/npm/jquery@3.5.1/dist/jquery.slim.min.js" integrity="sha384-DFXdz2htPH0lsSSs5nCTpuj/zy4CtOgrapoFVy38MVBnE+IbbVYUew+OXCxARkFj" crossorigin="anonymous"></script>
<script src="https://cdn.jsdelivr.net/npm/popper.js@1.16.1/dist/umd/popper.min.js" integrity="sha384-9/reFTGAW83EM2RDu2S0VKaIzap3H66lZ81PoYfFhBgU+6BZp6G/nlu735Sk71N" crossorigin="anonymous"></script>
<script src="https://cdn.jsdelivr.net/npm/bootstrap@4.6.1/dist/js/bootstrap.min.js" integrity="sha384-VhvPCCyXqtD5DqJeNx12dtTyhF78xXNXdkwX1CZeRusQFRKp+tA7hAshOK/B/FQ2" crossorigin="anonymous"></script>
```

Рис. 3.3. Посилання для вставки в `</body>`

### 3.1.3. JavaScript та його значення для веб-ресурсу

Програми написані на цій мові називаються скриптами. Вони можуть написані прямо в HTML розмітці та виконуватись автоматично після завантаження сторінки.

JavaScript – це текстова мова програмування, яка використовується як на стороні клієнта, так і на стороні сервера, що дозволяє зробити веб-сторінки інтерактивними. Якщо HTML і CSS є мовами, які надають структуру та стиль веб-сторінкам, JavaScript надає веб-сторінкам інтерактивні елементи, які залучають користувача. Поширені приклади JavaScript, які ви можете використовувати щодня, включають вікно пошуку Google, відео з підсумком новин або оновлення вашої сторінки Facebook [11].

Включення JavaScript покращує роботу веб-сторінки, перетворюючи її зі статичної сторінки в інтерактивну. Нагадаю, JavaScript додає поведінку веб-сторінкам.

JavaScript використовується в основному для веб-додатків і веб-браузерів. Але JavaScript також використовується за межами Інтернету в програмному забезпеченні, серверах та вбудованих апаратних елементах керування. Ось деякі основні речі для яких використовується JavaScript:

1. Додавання інтерактивної поведінки на веб-сторінки. JavaScript дозволяє користувачам взаємодіяти з веб-сторінками. Практично немає обмежень для того, що ви можете робити за допомогою JavaScript на веб-сторінці – це лише кілька прикладів: показати або

приховати інформацію одним натисканням кнопки; змінити колір кнопки, коли мишка наведе на неї курсор; прокрутити карусель із зображеннями; збільшити або зменшити масштаб зображення; відображення таймера або зворотнього відліку на веб-сайті; відтворення аудіо та відео на веб-сторінці; відображення анімацій.

2. Створення веб- та мобільних додатків. Розробники можуть використовувати різні фреймворки JavaScript для розробки та створення веб- та мобільних додатків. Фреймворки JavaScript – це колекції бібліотек коду JavaScript, які надають розробникам попередньо написаний код для використання рутинних функцій і завдань програмування – буквально каркас для створення веб-сайтів або веб-додатків. Популярні інтерфейсні фреймворки JavaScript: React, React Native, Angular і Vue. Багато компаній використовують Node.js, середовище виконання JavaScript, побудоване на движку JavaScript V8 Google Chrome. Кілька відомих прикладів включають PayPal, LinkedIn, Netflix та Uber.
3. Створення веб-серверів і розробка серверних додатків. Крім веб-сайтів і програм, розробники можуть використовувати JavaScript для створення простих веб-серверів і розкитку внутрішньої інфраструктури за допомогою Node.js.
4. Розробка гри. Звичайно, ви також можете використовувати JavaScript для створення браузерних ігор. Це чудовий спосіб для початківців розробників практикувати свої навички JavaScript.

Окрім необмежених можливостей, для веб-розробників є багато причин використовувати JavaScript замість інших мов програмування: JavaScript є єдиною мовою програмування, яка є рідною для веб-браузера; JavaScript є найпопулярнішою мовою; для початку існує низький поріг; це цікава мова для вивчення.

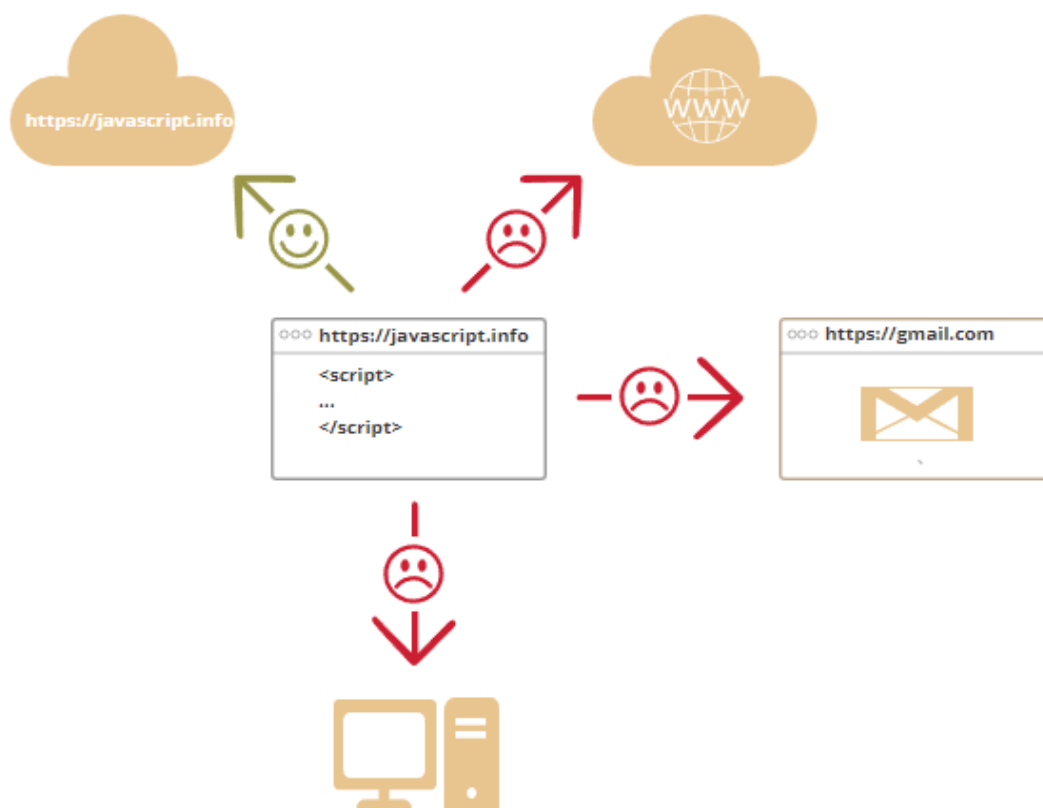


Рис. 3.4. Приклад роботи JavaScript

### 3.2. Back-end веб-ресурсу (внутрішні функції сайту)

Back-end – це серверна частина веб-сайту. Він зберігає та впорядковує дані, а також забезпечує належну роботу всього на стороні клієнта веб-сайту. Це частина веб-сайту, яку ви не можете побачити та взаємодіяти. Це частина програмного забезпечення, яке не контактує безпосередньо з користувачами. До частин і характеристик, розроблених бекенд-розробниками, користувачі опосередковано отримують доступ через інтерфейс. Діяльність, як от написання API, створення бібліотек і робота з системними компонентами без клієнтських інтерфейсів або навіть систем наукового програмування, також включені до back-end [9].

Мови програмування серверної частини – це внутрішня частина побудована з використанням деяких мов:

- PHP – це мова сценаріїв на стороні сервера, розроблена спеціально для веб-розробки. Оскільки PHP-код виконується на стороні сервера.

- C++ - це мова програмування загального призначення, яка сьогодні широко використовується для конкурентноспроможного програмування. Він також використовується як мова back-end.
- Java є однією з найпопулярніших і широко використовуваних мов і платформ програмування. Він дуже масштабований. Компоненти Java легко доступні.
- Python – це мова програмування, яка дозволяє швидко працювати та ефективніше інтегрувати системи.
- Node.js є відкритим вихідним кодом і міжплатформним середовищем для виконання коду. Node.js – це не фреймворк і не мова програмування. Більшість людей розгублені і розуміють, що це фреймворк або мова програмування. Він часто використовується для створення внутрішніх служб, таких як API. Його використовують великі компанії, такі як Paypal, Uber, Netflix, Walmart тощо.

Різниця між front-end і back-end полягає в тому, що вони дуже відрізняються один від одного, але все ж це два аспекти однієї ситуації. Front-end – це те, що бачать із чим взаємодіють користувачі, а back-end – це те, як працює front-end.

### **3.2.1. Використання PHP**

PHP – це мова сценаріїв на стороні сервера, який використовується для розробки статичних або динамічних веб-сайтів чи веб-додатків. PHP означає Hypertext Preprocessor, що раніше означало Personal Home Pages [6].

PHP-скрипти можна інтерпретувати лише на стороні сервера, на якому встановлено PHP.

Клієнтським комп'ютерам, які мають доступ до PHP-скриптів, потрібен лише веб-браузер. Файл містить теги PHP і закінчується розширенням «.php».

Сценарій – це набір програмних інструкцій, які інтерпретуються під час виконання.



Мова сценаріїв – це мова, яка інтерпретує сценарії під час виконання. Скрипти зазвичай вбудовуються в інші програмні середовища. Метою сценаріїв зазвичай є підвищення продуктивності або виконання рутинних завдань для програм. Скрипти на стороні сервера інтерпретуються на сервері, тоді як сценарії на стороні клієнта інтерпретуються клієнтською програмою.

PHP – це сценарій на стороні сервера, який інтерпретуються на сервері, тоді як JavaScript – це приклад сценарію на стороні клієнта, який інтерпретуються клієнтським браузером. І PHP, і JavaScript можна вбудувати в сторінки HTML.

PHP означає – Персональна домашня сторінка, але тепер це означає рекурсивний бекронім PHP; Hypertext Preprocessor.

PHP-код може бути вбудований в HTML-код, або його можна використовувати в поєднанні з різними системами веб-шаблонів, системою керування веб-контентом та веб-фреймворками.

```
<?php
    echo 'Hello World';
?>
```

Рис. 3.5. Синтаксис PHP

Файл PHP також може містити такі теги, як HTML, і клієнтські скрипти, такі як JavaScript.

- HTML є додатковою перевагою під час вивчення мови PHP. Ви навіть можете вивчати PHP, не знаючи HTML, але рекомендується знати принаймні основи HTML.
- Системи управління базами даних СУБД для програм, що працюють на базі даних.

- Для більш складних тем, таких як інтерактивні програми та веб-сервіси, вам знадобляться JavaScript та XML.

Наведені нижче блок-схема ілюструє основну архітектуру веб-програми PHP і те, як сервер обробляє запити (Рис. 3.6).

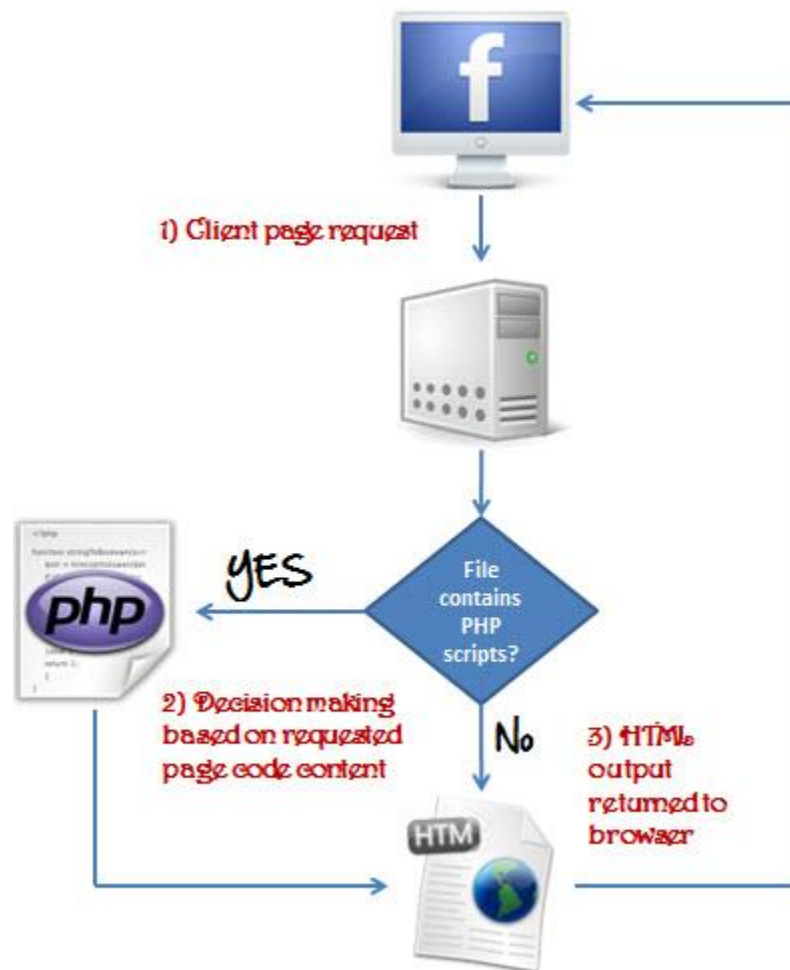


Рис. 3.6. Ілюстрація архітектури веб-програми

Навіщо використовувати PHP:

- PHP є відкритим вихідним кодом і безкоштовним.
- Коротка крива навчання в порівнянні з іншими мовами.
- Велика спільнота.
- Більшість серверів веб-хостингу за замовчуванням підтримують PHP, на відміну від інших мов, таких як ASP, які потребують IIS. Це робить PHP економічно ефектним вибором.

- PHP регулярно оновлюється, щоб бути в курсі останніх технологічних тенденцій.
- Потрібно лише встановити PHP на сервер, щоб почати використовувати.
- PHP має вбудовану підтримку для роботи рука бо руку з MySQL.
- PHP є кросплатформним – це означає, що ви можете розгорнути свою програму в кількох різних операційних системах.

З точки зору частки ринку, існує понад 20 мільйонів веб-сайтів і програм в Інтернеті, розроблених за допомогою мови сценаріїв PHP. Це може бути пов'язано з вищезгаданими моментами. На схемі нижче показано деякі з популярних сайтів, які використовують PHP (Рис. 3.7).

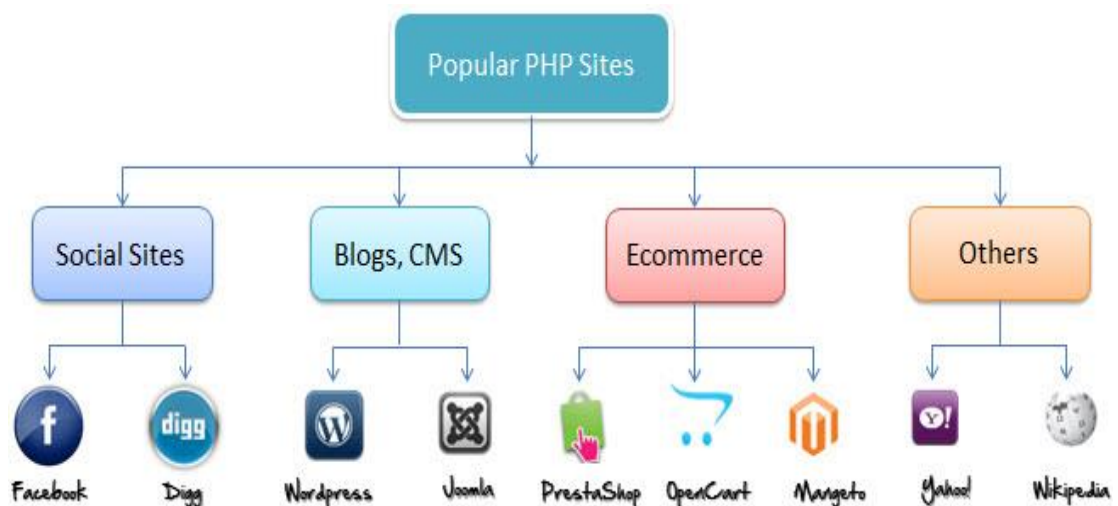


Рис. 3.7. Популярні сайти, які використовують PHP

PHP був розроблений для роботи з HTML, і тому його можна вбудувати в HTML-код. Ви можете створювати файли PHP без будь-яких тегів HTML, і це називається файлом Pure PHP. Сервер інтерпретує PHP-код і виводить результати у викладі HTML-коду для веб-браузерів. Для того, щоб сервер ідентифікував PHP-код із HTML-коду, ми повинні завжди вкладати PHP-код у теги PHP. Тег PHP починається з символу «менше», за яким слідує знак питання, а потім слова «php». PHP є мовою, чутливою для регістру, «VAR» не те саме, що «var». Самі теги PHP не чутливі до регістру, але рекомендую використовувати

малі літери. Операції PHP закінчуються крапкою з комою «;». Якщо у вас є лише одне твердження, ви можете не писати крапку з комою. Якщо у вас є більше одного оператора, ви повинні закінчувати кожен рядок крапкою з комою. Для послідовності рекомендується завжди закінчувати своє речення крапкою з комою. PHP-скрипти виконуються на сервері. Результат повертається у вигляді HTML [5].

В даному розділі я навів більш ніж достатньо аргументів, чому я використовую PHP для написання свого веб-додатка.

### **3.2.2. База даних MySQL**

Шведська компанія під назвою MySQL AB спочатку розробила MySQL у 1994 році. Американська технологічна компанія Sun Microsystems стала повною власністю, коли вони купили MySQL AB у 2008 році. Американський технологічний гігант Oracle у 2010 році придбав самі компанію Sun Microsystems, і з тих пір MySQL практично належить Oracle.

Що стосується загального визначення, то MySQL – це система управління реляційною базою даних з відкритим вихідним кодом (RDBMS) з моделлю клієнт-сервер. СУБД – це програмне забезпечення або сервіс, який використовується для створення та керування базами даних на основі реляційної моделі [12].

База даних – це просто набір структурованих даних. Подумайте про селфі: ви натискаєте кнопку і знімаєте себе. Ваше фото – це дані, а галерея вашого телефону – це база даних. База даних – це місце, в якому зберігаються та організовані дані. Слово «реляційний» означає, що дані, що зберігаються в наборі даних, організовані у вигляді таблиць. Кожна таблиця певним чином пов'язана. Якщо програмне забезпечення не підтримує реляційну модель даних, просто назвіть його СУБД.

Відкритий код означає, що ви можете вільно використовувати та змінювати його. Будь-хто може встановити програмне забезпечення. Ви також

можете вивчити та налаштувати вихідний код, щоб краще задовольнити свої потреби. Однак GPL (GNU Public License) визначає, що ви можете робити залежно від умов. Комерційно ліцензована версія доступна, якщо вам потрібна більш гнучкі права власності та розширена підтримка.

Комп'ютер, які встановлюють і запускають програмне забезпечення СУБД, називаються клієнтами. Щоразу, коли їм потрібно отримати доступ до даних, вони підключаються до сервера RDBMS. ЦЕ частина «клієнт-сервер».

MySQL є одним із багатьох варіантів програмного забезпечення СУБД. RDBMS і MySQL часто вважаються однаковими через популярність MySQL. Кілька великих веб-додатків, таких як Facebook, Twitter, YouTube, Google і Yahoo! всі використовують MySQL для цілей зберігання даних. Незважаючи на те, що спочатку він був створений для обмеженого використання, тепер він сумісний з багатьма важливими обчислювальними платформами, такими як Linux, macOS, Microsoft Windows і Ubuntu [12].

Майте на увазі, що MySQL є одним із найпопулярніших програмних засобів СУБД, яке реалізує модель клієнт-сервер. Отже, як клієнт і сервер взаємодіють? Вони використовують специфічну для домену мову – мову структурованих запитів (SQL). Якщо ви коли-небудь зіткнетеся з іншими іменами, які містять SQL, наприклад PostgreSQL і Microsoft SQL Server, швидше за все, це бренди, які також використовують синтаксис мови структурованих запитів. Програмне забезпечення RDBMS часто пишеться іншими мовами програмування, але завжди використовують SQL як свою основну мову для взаємодії з базою даних. Сам MySQL написаний на C і C++.

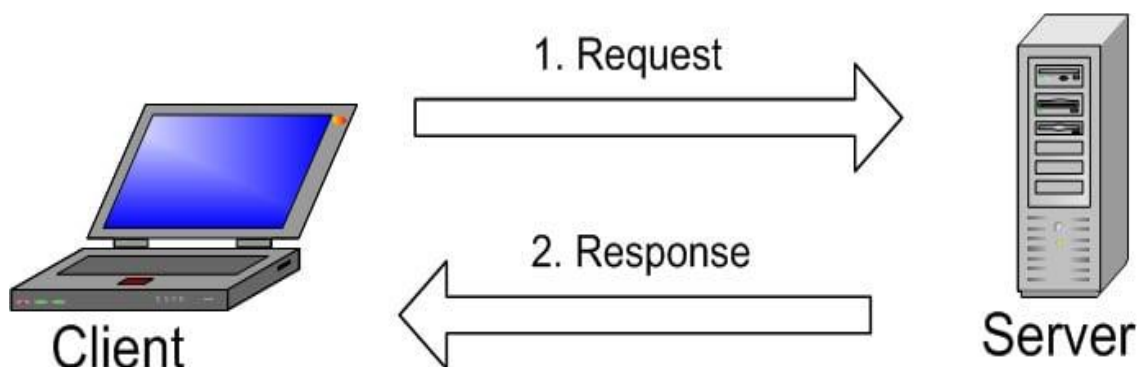


Рис. 3.8. Структура клієнт-серверної частини

Зображення пояснює основну структуру клієнт-серверної структури. Один або кілька пристроїв (клієнтів) підключаються до сервера через певну мережу (Рис. 3.8). Кожен клієнт може зробити запит із графічного інтерфейсу користувача (GUI) на своїх екранах, і сервер видаватиме потрібний результат, якщо обидві сторони розуміють інструкцію. Не будучи занадто технічним, основні процеси, що відбуваються в середовищі MySQL, однакові, а саме:

1. MySQL створює базу даних для зберігання та маніпулювання даними, визначаючи взаємозв'язок кожної таблиці.
2. Клієнти можуть робити запити, вводячи певні оператори SQL на MySQL.
3. Серверна програма відповідає запитаною інформацією, і вона з'явиться на стороні клієнтів.

MySQL дійсно не єдина СУБД на ринку, але вона є однією з найпопулярніших і лише поступається Oracle Database, коли оцінюється за такими критичними параметрами, як кількість згадок у результатах пошуку, професійні профілі в LinkedIn та частота технічних дискусій на інтернет-форумах. Той факт, що багато великих технологічних гігантів покладаються на нього, ще більше зміцнює залужені позиції (Рис. 3.9).

Rank			DBMS	Database Model	Score				
Dec	Nov	Dec			Dec	Nov	Dec	% May	%cumulati
2020	2020	2019			2020	2020	2019	2020	ve total
1	1	1	Oracle	Relational, Multi-model	1325.6	-19.4	-20.8	18.58	18.58
2	2	2	MySQL	Relational, Multi-model	1255.45	13.81	-20.21	17.60	36.18
3	3	3	MS SQL Server	Relational, Multi-model	1038.09	0.45	-58.11	14.55	50.72
4	4	4	PostgreSQL	Relational, Multi-model	547.57	-7.49	44.2	7.67	58.40
5	5	5	MongoDB	Document, Multi-model	457.73	3.9	36.61	6.42	64.81
6	6	6	IBM Db2	Relational, Multi-model	160.43	-1.19	-10.91	2.25	67.06
7	7	8	Redis	Key-value, Multi-model	153.63	-1.79	7.39	2.15	69.22
8	8	7	Elasticsearch	Search engine, Multi-model	152.49	0.94	2.24	2.14	71.35
9	9	11	SQLite	Relational	121.68	-1.63	1.32	1.71	73.06
10	10	10	Cassandra	Wide column	118.84	0.09	-1.87	1.67	74.72
11	11	9	Microsoft Access	Relational	116.74	-0.5	-12.73	1.64	76.36
12	12	13	MariaDB	Relational, Multi-model	93.61	1.31	6.82	1.31	77.67
13	13	12	Splunk	Search engine	87	-2.71	-3.53	1.22	78.89
14	14	15	Teradata	Relational, Multi-model	73.83	-1.77	-4.66	1.03	79.93
15	15	14	Hive	Relational	70.27	0.01	-15.78	0.98	80.91

Рис. 3.9. Статистика баз даних 2020 року

Широкий набір кластерних серверів підтримує MySQL. Нехалежно від того, зберігаєте ви величезні обсяги великих даних електронної комерції або займаєтеся серйозною бизнес-аналітикою, MySQL може допомогти вам безперешкодно з оптимальною швидкістю.

Індустрії роками використовують MySQL, а це означає, що для кваліфікованих розробників є багато ресурсів. Користувачі MySQL можуть очікувати швидкого розвитку програмного забезпечення та позаштатних експертів, які бажують працювати за меншу зарплату, якщо вони коли-небудь знадобляться.

Ваші дані повинні бути вашою основною турботою при виборі правильного програмного забезпечення СУБД. Завдяки своїй системі привілеїв доступу та управлінню обліковими записами користувачів MySQL встановлює високий рівень безпеки. Доступна перевірка на основі хоста та шифрування пароля.

### 3.2.3. Використання JQuery

Простіше кажучи, jQuery – це набір попередньо написаного коду JavaScript (відомого як бібліотека JavaScript), який ви можете додати до своїх власних проєктів.

Для певних рутинних завдань – немає абсолютно ніякого сенсу винаходити велосипед заново, тому потрібно використовувати jQuery. jQuery дозволяє веб-розробникам підключати рутинні функції JavaScript до веб-сторінки, щоб вони могли витратити більше часу на складні функції, які є унікальними для іншого сайту [13].

Окрім розуміння, що таке jQuery, корисно розуміти, чому розробники його використовують. Нижче ми глибше розглянемо деякі з цих пунктів, але ось список причин:

- jQuery робить програмування на JavaScript швидшим і ефективнішим.
- jQuery є відкритим вихідним кодом (це означає, що будь-хто може долучитися до нього або змінити його) і має велику спільноту користувачів, тобто він постійно підтримується та доповнюється.
- jQuery має велику документацію (включаючи приклади вбудованого коду).
- jQuery добре працює з будь-якими іншими бібліотеками JavaScript, які ви можете використовувати.
- jQuery має масу плагінів, які дозволяють розширювати функціональність jQuery за потреби.

JavaScript і jQuery згадуються настільки взаємозамінно, що може бути незрозумілим їхній розділ. Просто пам'ятайте: JavaScript – це окрема мова програмування, а jQuery – це набір коду JavaScript, а не власна мова.

Щоб швидко побачити різницю, розглянемо наступне. Скажімо, ви хочете, щоб користувачі отримували повідомлення з підтвердженням «дякую за реєстрацію», коли вони додадуть себе до списку електронної пошти вашого веб-



сайту. Ручне кодування цієї функції за допомогою JavaScript виглядало б приблизно так (Рис. 3.10):

```

window.onload = initAll;
function initAll() {
document.getElementById("submit").onclick = submitMessage;
}
function submitMessage() {
var greeting = document.getElementById("name").getAttribute("value");
document.getElementById("headline").innerHTML = "Дякуємо, що приєдналися
до нашого списку електронних листів" + привітання;
повернути false;
}

```

Рис. 3.10. Функція JavaScript

Це дуже багато коду для такої базової функції. Однак, використовуючи фрагменти коду jQuery, ви в кінцевому підсумку спростите його в щось подібно до цього прикладу jQuery (Рис. 3.11):

```

$("#submit").click(function () {
var greeting = $("#name").val();
$("#headline").html("Дякуємо, що приєдналися до нашого списку розсилки, " +
привітання);
повернути false;
});

```

Рис. 3.11. Функція jQuery

Цей набагато більш керований скрипт jQuery надсилає запит до бібліотеки jQuery, яку ви можете встановити на своєму веб-сайті або використовувати через Google, включивши посилання jQuery на розміщенні бібліотеки Google у вашому коді (Рис. 3.12):

```
<head>  
<script src="https://ajax.googleapis.com/ajax/libs/jquery/3.3.1/jquery.min.js">  
</script>  
</head>
```

Рис. 3.12. Посилання для підключення jQuery

Потім jQuery відповідає на ваш запит і виконує потрібну функцію для вашого користувача. Будь-який підхід (JavaScript або jQuery) видасть повідомлення «дякую», але сценарій jQuery доставить вас до цього набагато швидше та ефективніше. Крім того, ви зможете повторно використовувати цю функцію jQuery, коли виникне така ж потреба в цьому або будь-якому іншому проекті веб-розробки, над яким ви працюєте [13].

### 3.3. Опис середовища розробки Sublime Text 3

Sublime Text 3 – це популярний кросплатформений текстовий редактор та IDE з підсвіткою синтаксису та підтримкою плагінів. Свою неабияку популярність редактор здобув саме через свою гнучку архітектуру, підтримкою додатків на мові Python. Однак за замовчуванням Sublime Text 3 не містить локалізацій, відмінних від англійської. І якщо російський переклад легко шукається та встановлюється, то з українським не все так просто (Рис. 3.13).

Sublime Text 3 може використовувати ваш графічний процесор на Linux, Mac і Windows під час візуалізації інтерфейсу. Це призводить до плавного інтерфейсу користувача з роздільною здатністю до 8К, при цьому споживається менше енергії, ніж раніше [14].

Вкладки файлів покращені, щоб розділення переглядів було легким, з підтримкою в інтерфейсі та вбудованими командами. Бічна панель, панель вкладок, Goto Anything, Goto Definition, автозавершення налаштовані, щоб зробити навігацію по коду простішою та інтуїтивнішою, ніж будь-де.

Існує механізм автоматичного завершення, щоб забезпечити розумні завершення на основі існуючого коду в проекті. Пропозиції також доповнюються інформацією про їх тип і містять посилання на визначення.

Теми за замовчуванням та адаптивні оновлені новими стилями вкладок і неактивним затемненням панелі. Теми та кольорові схеми підтримують автоматичне перемикання темного режиму. Адаптивна тема для Windows і Linux має спеціальні рядки заголовка [14].

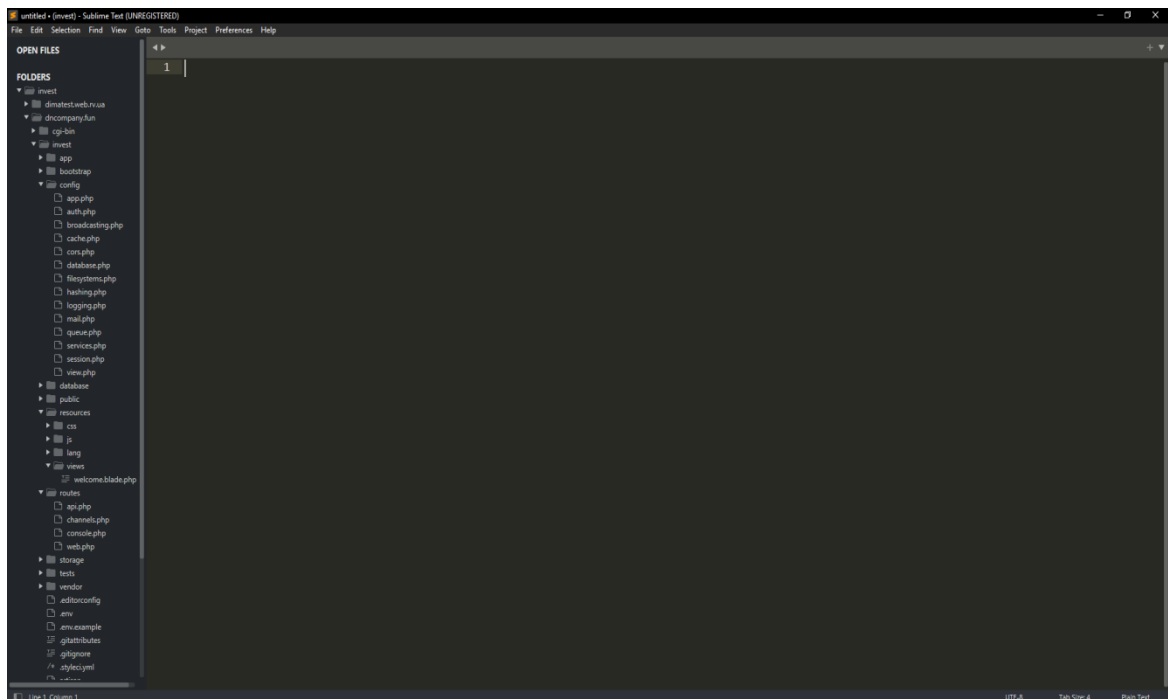


Рис. 3.13. Інтерфейс Sublime Text 3

### 3.4. Створення панелі адміністратора

Створення панелі адміністратора розпочинається з налаштування архітектури бази даних MySQL. База даних складається з дев'яти таблиць. Вісім з цих таблиць є основними і одна – це прив'язка однієї таблиці до іншої.

Панель адміністратора має налаштовувати всі дані, які можуть бути внесені в розклад занять. Тому я створив окремі сторінки в адміністративній панелі для налаштування даних. Для того, щоб почати корегувати дані в адміністративній панелі потрібно пройти авторизацію. Сторінка авторизації не

чим не відрізняється від авторизацій на будь-яких інших сайтах (Рис. 3.14). Потрібно мати e-mail та пароль. Але ці дані, буде мати лише обмежене коло людей, щоб ніхто не міг коригувати розклад немаючи на це підстав.

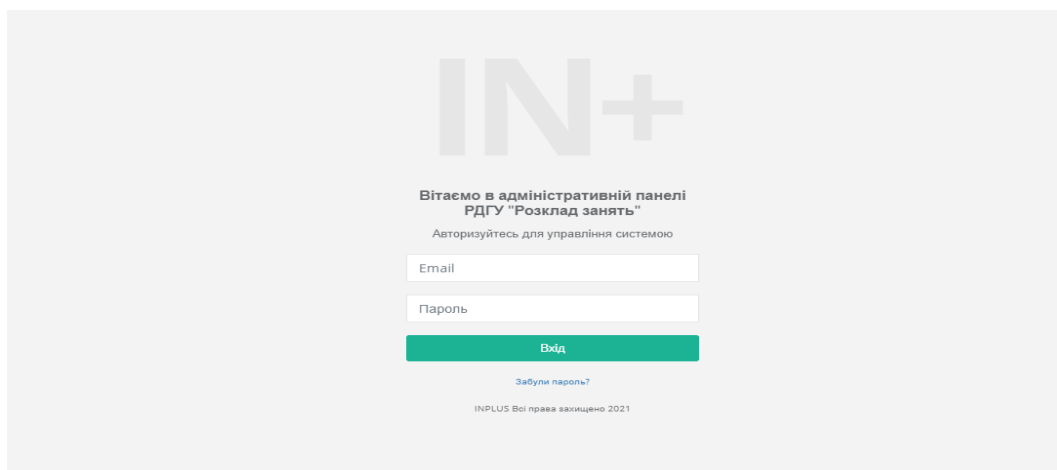


Рис. 3.14. Сторінка входу в адміністративну панель

Сторінка «Факультети». Містить у собі дані про факультети університету (Рис. 3.15), та дозволяє введення та коригування даних (Рис. 3.16). Також до факультету прив'язується виклад, який проводить заняття. Вся інформація про факультети зберігається в базі даних в двох окремих таблицях: таблиця з інформацією про факультети (Таблиця 3.1); таблиця з прив'язкою викладача до факультету (Таблиця 3.2).

#	Факультет	
2	Психолого-педагогічний факультет	+
3	Підготовчий факультет	+
4	Математико-інформаційний факультет	+
5	Тест	+

Рис. 3.15. Сторінка «Факультети»

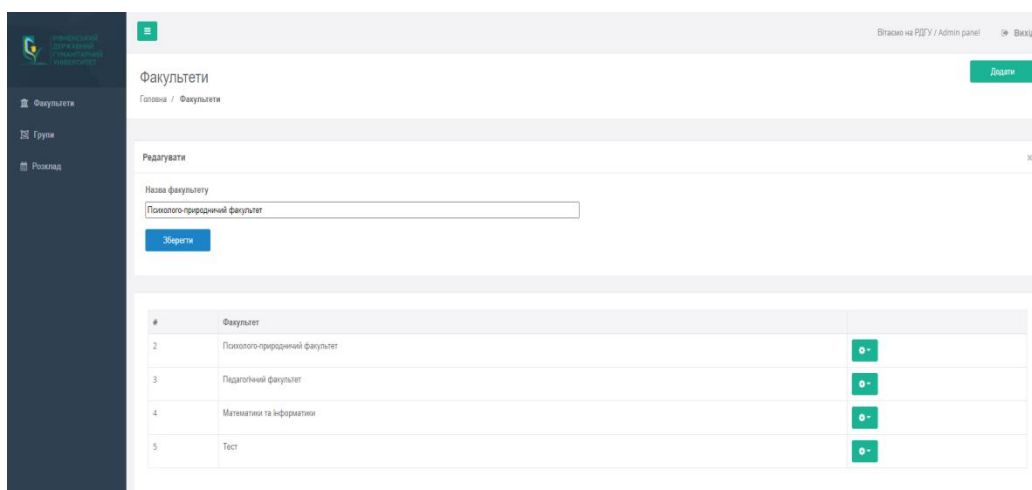


Рис. 3.16. Сторінка «Факультети», коригування даних

Таблиця 3.1. Структура таблиці «Факультети»

Ім'я поля	Тип і розмір поля	Опис поля
id	Int(11)	Первинний ключ
name	Varchar(255)	Назва факультету
upd_date	Datetime(0)	Дата оновлення даних

Таблиця 3.2. Структура таблиці «Факультети і викладачі»

Ім'я поля	Тип і розмір поля	Опис поля
id	Int(11)	Первинний ключ
id_faculty	Int(11)	Первинний ключ факультету
id_teacher	Int(11)	Первинний ключ викладача

Сторінка «Викладачі». Містить у собі дані про викладачів університету (Рис. 3.17), та дозволяє введення та коригування даних (Рис. 3.18). Вся інформація про викладачів зберігається в базі даних в окремій таблиці (Таблиця 3.3).



Рис. 3.17. Сторінка «Викладачі»

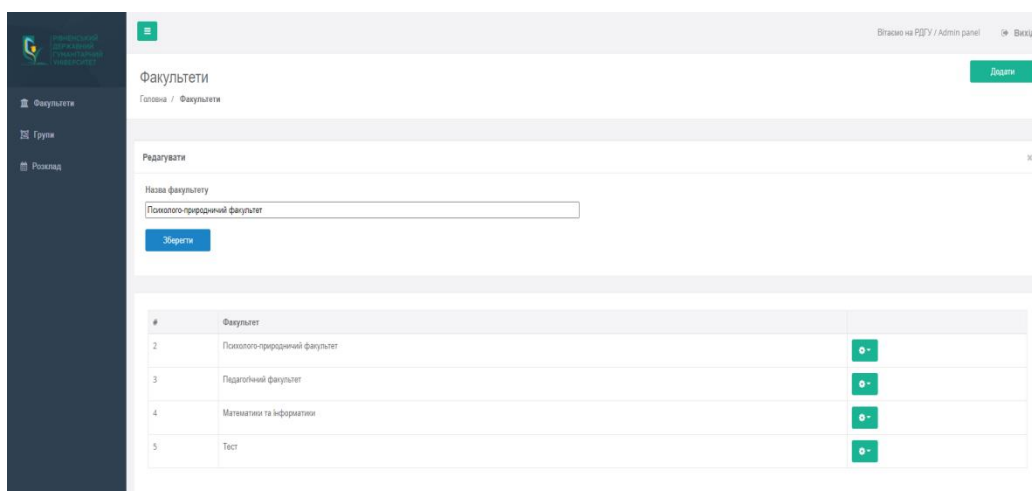


Рис. 3.18. Сторінка «Викладачі», коригування даних

Таблиця 3.3. Структура таблиці «Викладачі»

Ім'я поля	Тип і розмір поля	Опис поля
id	Int(11)	Первинний ключ
name	Varchar(255)	П.І.Б. викладача
upd_date	Datetime(0)	Дата оновлення даних

Сторінка «Групи». Містить у собі дані про групи університету (Рис. 3.19), та дозволяє введення та коригування даних (Рис. 3.20). Вся інформація про групи зберігається в базі даних в окремій таблиці (Таблиця 3.4).

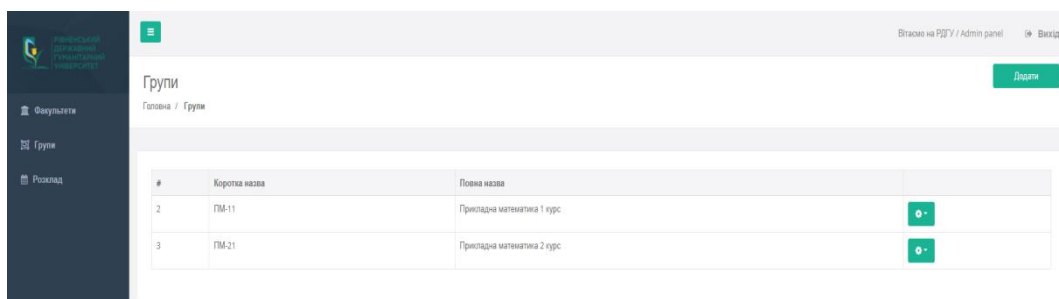


Рис. 3.19. Сторінка «Групи»

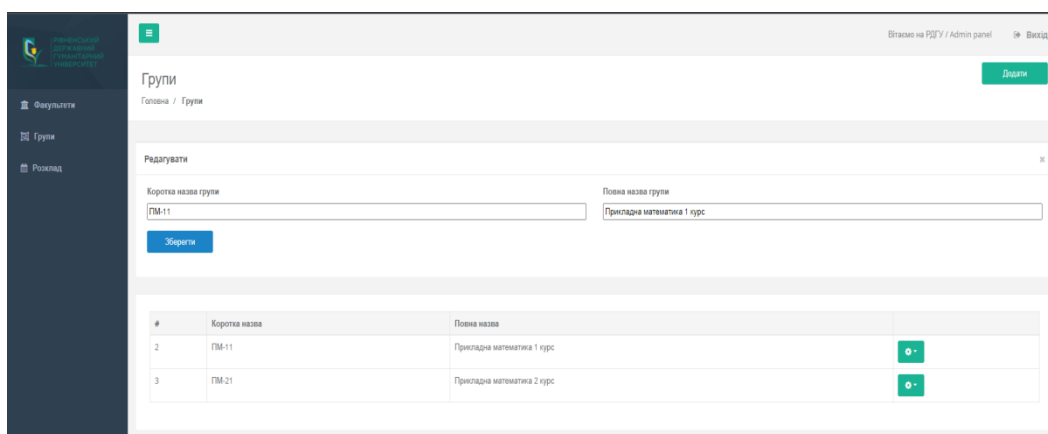


Рис. 3.20. Сторінка «Групи», коригування даних

Таблиця 3.4. Структура таблиці «Групи»

Ім'я поля	Тип і розмір поля	Опис поля
id	Int(11)	Первинний ключ
short_name	Varchar(255)	Коротка назва групи
name	Varchar(255)	Повна назва групи
upd_date	Datetime(0)	Дата оновлення даних

Сторінка «Корпуси». Містить у собі дані про корпуси університету (Рис. 3.21), та дозволяє введення та коригування даних (Рис. 3.22). Вся інформація про корпуси зберігається в базі даних в окремій таблиці (Таблиця 3.5).

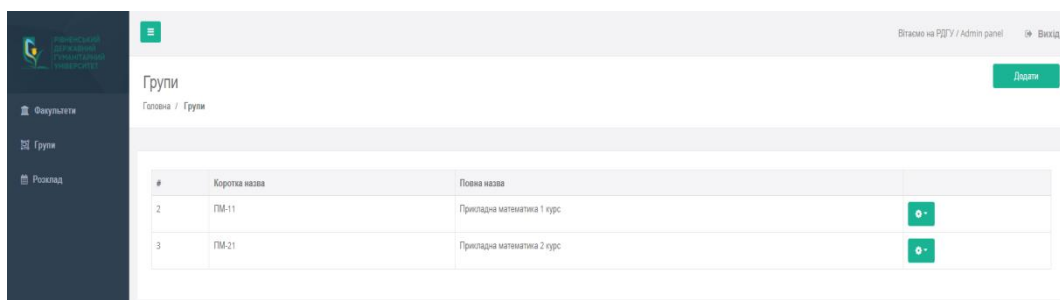


Рис. 3.21. Сторінка «Корпуси»

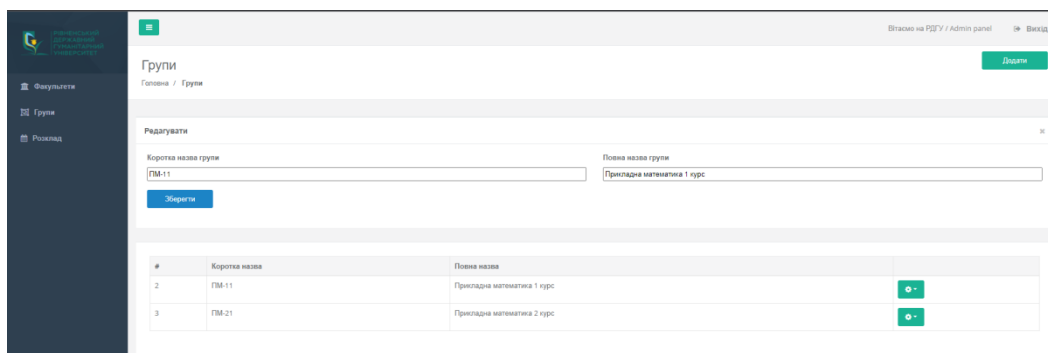


Рис. 3.22. Сторінка «Корпуси», коригування даних

Таблиця 3.5. Структура таблиці «Корпуси»

Ім'я поля	Тип і розмір поля	Опис поля
id	Int(11)	Первинний ключ
name	Varchar(255)	Назва і адреса корпусу
upd_date	Datetime(0)	Дата оновлення даних

Додатково ще створена таблиця з годинами проведення занять (Таблиця 3.6). Для неї немає інтерфейсу, так як ці дані є статичними.

Таблиця 3.6. Структура таблиці «Години проведення занять»

Ім'я поля	Тип і розмір поля	Опис поля
id	Int(11)	Первинний ключ
number	Varchar(255)	Загальний проміжок часу
start	Time(0)	Час початку заняття
end	Time(0)	Час кінця заняття

Сторінка «Розклад». Це є основна сторінка моєї інформаційної системи, адже в даному інтерфейсі формується і коригується розклад занять для викладачів та студентів. Головна сторінка містить фільтри для відображення інформації про розклад занять (Рис. 3.23). Після вибору одного чи декілька фільтрів і натисненні кнопки «Показати розклад», в даний інтерфейс загрузяться дані розкладу занять з двома кнопками: «Редагувати» та «Видалити» (Рис. 3.24). Якщо натиснути кнопку «Редагувати», то відкриється вікно коригування розкладу (Рис. 3.25). Якщо натиснути кнопку «Видалити», відкриється вікно з підтвердженням видалення розкладу і якщо натиснути кнопку «Так», відповідно інформація про розклад видалиться повністю. Інтерфейс коригування даних, такий самий як і інтерфейс створення даних (Рис. 3.26). Вся інформація про розклад занять зберігається в базі даних в окремій таблиці (Таблиця 3.7).



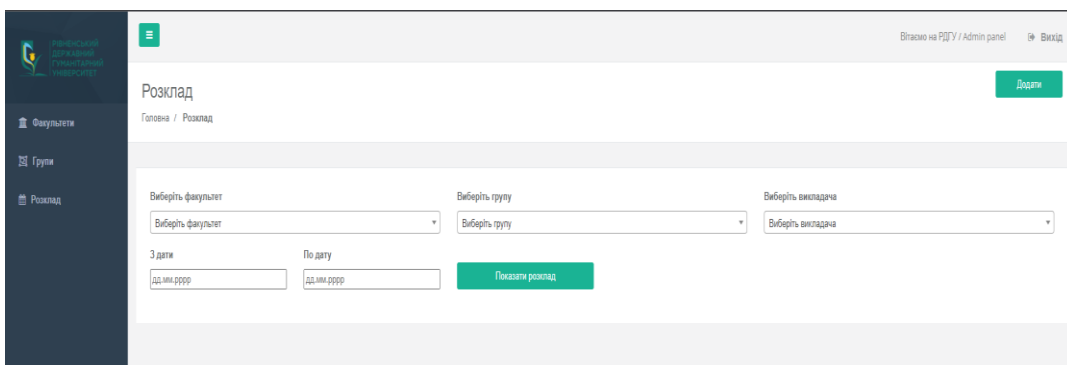


Рис. 3.23. Сторінка «Розклад»

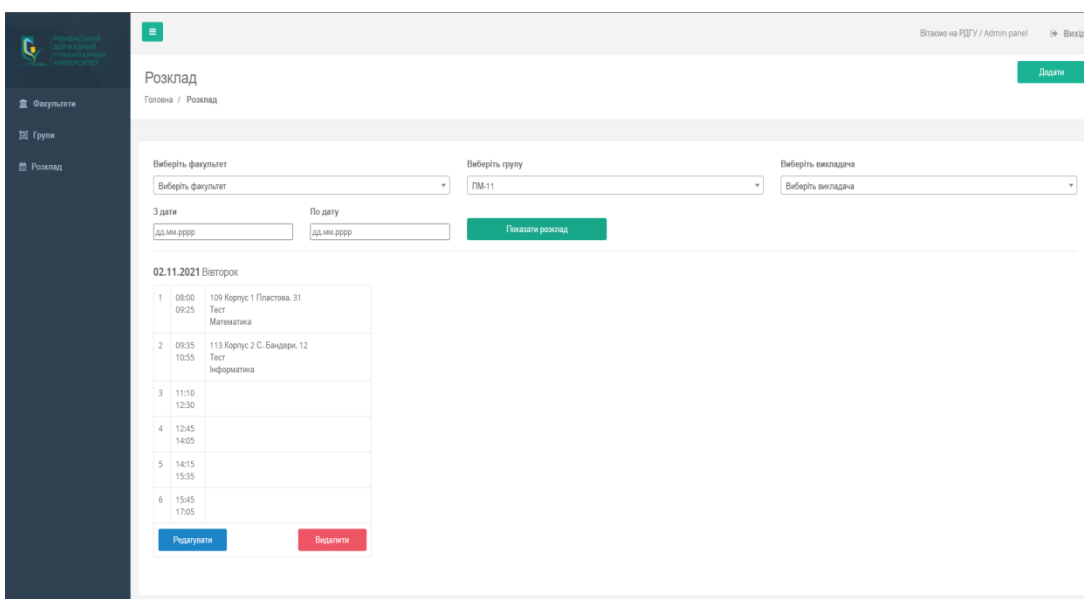


Рис. 3.24. Сторінка «Розклад» після фільтрації

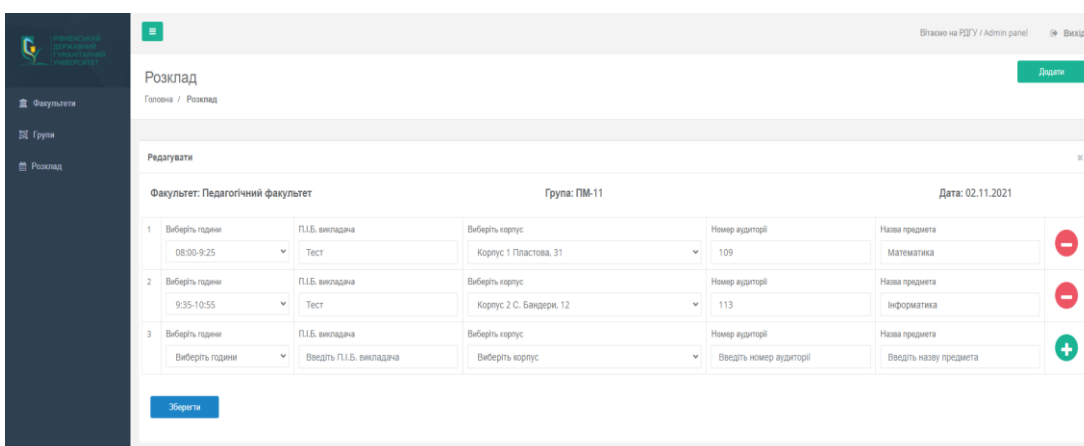


Рис. 3.25. Сторінка «Розклад», коригування даних

Рис. 3.26. Сторінка «Розклад», створення даних

Таблиця 3.7. Структура таблиці «Розклад»

Ім'я поля	Тип і розмір поля	Опис поля
id	Int(11)	Первинний ключ
date	Date(0)	Дата заняття
time	Varchar(255)	Проміжок часу заняття
faculty	Int(11)	Первинний ключ факультету
teacher	Int(11)	Первинний ключ викладача
group	Int(11)	Первинний ключ групи
subject	Int(11)	Первинний ключ предмета
korpus	Int(11)	Первинний ключ корпусу
cabinet	Varchar(255)	Назва аудиторії
upd_date	Datetime(0)	Дата оновлення даних

### 3.5. Розробка клієнтської частини

Клієнтська частина ІС «Розклад занять» складається з одної сторінки і динамічного завантаження даних. Сторінка містить логотип університету та його назву, фільтри для пошуку і кнопку «Показати розклад». Після вибору фільтрів, сторінка динамічно завантажує таблиці з розкладом занять по фільтрах, які вибрав користувач (Рис. 3.27). Тобто, будь-який студент або викладач за допомогою даного інтерфейсу зможе переглянути свій розклад занять.

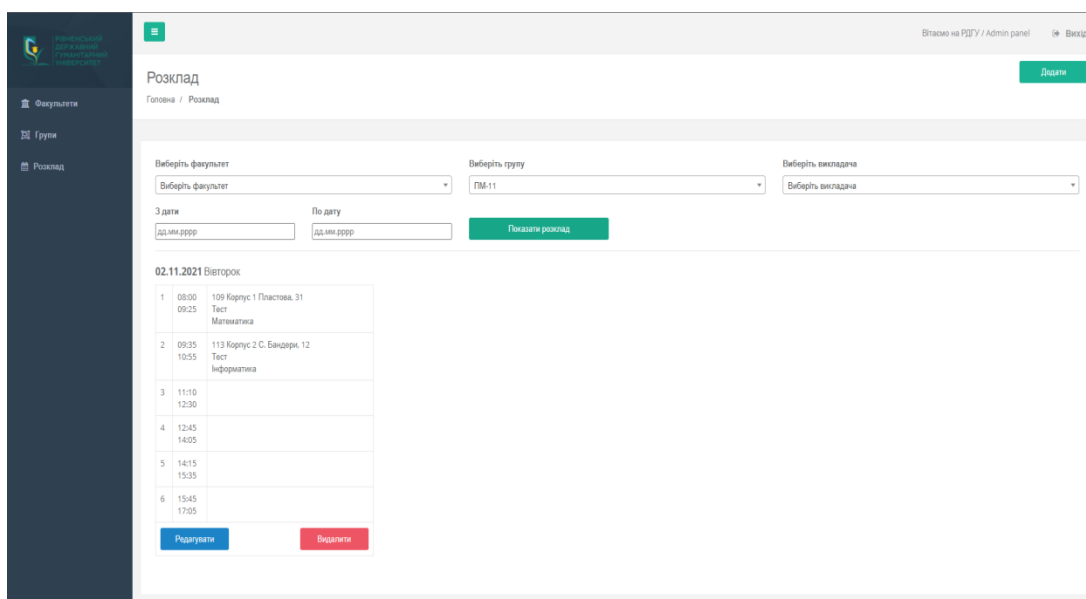


Рис. 3.27. Клієнтська частина

### 3.6. Завантаження сайту на хостинг

Хостинг – це онлайн-сервіс, який дає змогу опублікувати свій веб-сайт або веб-програму в Інтернеті. Коли ви реєструєтесь на послугу веб-хостингу, ви в основному орендуєте місце на фізичному сервері, де ви можете зберігати всі файли та дані, необхідні для належної роботи вашого веб-сайту.

Сервер – це фізичний комп’ютер, який працює без будь-яких перебоїв, тому ваш веб-сайт завжди доступний для всіх, хто хоче його переглянути. Ваш веб-хост відповідає за підтримку та роботу цього сервера, захист його від зловмисних атак і передачу вашого вмісту – такого як текст, зображення, файли тощо – із сервера до браузерів ваших відвідувачів.

Для тестового запуску свого веб-сайту я обрав хостинг Ukraine.com.ua. Цій платформі вже понад десять років і я співпрацюю з ними два роки без проблем. Я вибрав тестовий план на 30 днів, щоб показати роботу своєї ІС «Розклад занять». Для того щоб розмістити свій сайт на даному хостингу, потрібно підключитися до сервера, дані якого надав мені хостинг. Підключення до сервера відбувається через програму FileZilla (Рис. 3.28). Після підключення до серверу, я переніс файли свого проекту на нього і зберіг дані. Потім відкрив браузер і

пошукову строку написав посилання, яке видав мені хостинг, щоб перевірити чи все працює так само, як і на локальному сервері.

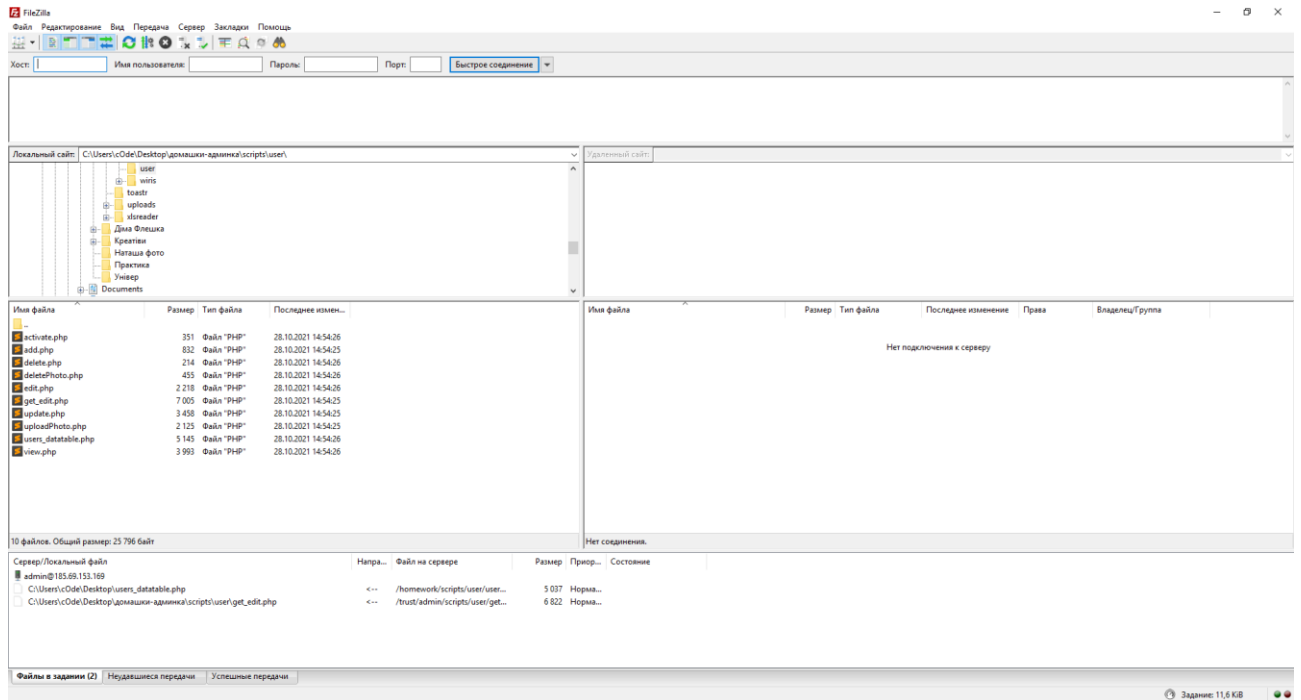


Рис. 3.28. Интерфейс программы FileZilla

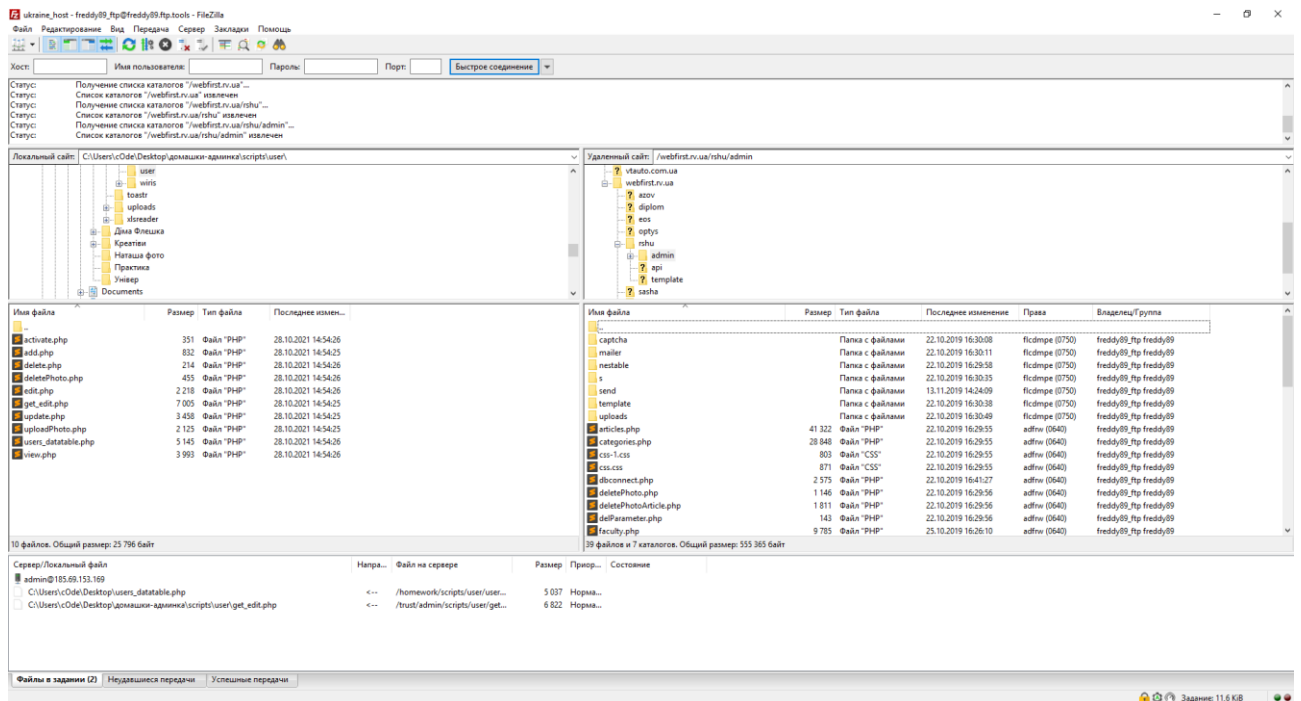


Рис. 3.29. Файлы ІС «Розклад занять» після завантаження

## ВИСНОВКИ

Вході виконання даної роботи було досліджено предметну область складання розкладів та розроблено алгоритм формування розкладу навчального процесу. Розглянуто проблематику, актуальність даної теми і її використання в наш час.

Отже, представлена ІС «Розклад занять» містить такі відомості, як спеціальність, курс, група, назва дисциплін, форма проведення занять (лекції; лабораторні, практичні, семінарські заняття; консультації), прізвище та ініціали викладачів, місце проведення занять.

Впровадження ІС в університеті, оптимізує роботу працівників деканатів, підвищить контроль за внесення змін в розклад та збільшить ефективність раціонального використання матеріально-технічної бази.

Користувачі системи повинні мати доступ до інтернету. Для розвитку даної системи в подальшому необхідно створити застосунок для смартфона з офлайн системою «Розклад занять».

Крім цього система зручна не тільки для перегляду, а й для редагування розкладу адміністрацією навчального підрозділу вищого навчального закладу. Усі зміни відразу заносяться до бази даних та автоматично синхронізуються з створеним web-застосунком.

## СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Suehring S. PHP, MySQL, JavaScript & HTML5 All-in-One For Dummies / Steve Suehring. – New York, United States: John Wiley & Sons Inc, 2018. – 800 с.
2. Девис М. Изучаем PHP и MySql / М. Девис, Д. Филлипс., 2015. – 448 с.
3. Nixon R. Learning PHP, MySQL & JavaScript : A Step-by-Step Guide to Creating Dynamic Websites / R. Nixon, Д. Филлипс. – Sebastopol, United States: O'Reilly Media, Inc, USA, 2021. – 825 с.
4. Ulman L. PHP Advanced and Object-Oriented Programming: Visual QuickPro Guide / L. Ulman, Д. Филлипс., 2012. – 504 с.
5. Занстра М. Книга PHP. Объекты, шаблоны и методики программирования / М. Занстра, Д. Филлипс., 2019. – 736 с.
6. Веб-ресурс php.net [Электронный ресурс] – Режим доступа до ресурсу: <https://www.php.net/>
7. HTML & CSS [Электронный ресурс] – Режим доступа до ресурсу: <https://www.w3.org/standards/webdesign/htmlcss>
8. What are HTML and CSS used for? The basics of coding for the web [Электронный ресурс] – Режим доступа до ресурсу: <https://www.futurelearn.com/info/blog/what-are-html-css-basics-of-coding>
9. Frontend vs Backend [Электронный ресурс] – Режим доступа до ресурсу: <https://www.geeksforgeeks.org/frontend-vs-backend/>
10. Bootstrap [Электронный ресурс] – Режим доступа до ресурсу: <https://getbootstrap.com/docs/>
11. JavaScript Info [Электронный ресурс] – Режим доступа до ресурсу: <https://javascript.info/intro>
12. MySql [Электронный ресурс] – Режим доступа до ресурсу: <https://dev.mysql.com/doc>
13. jQuery [Электронный ресурс] – Режим доступа до ресурсу: <https://jquery.com/>
14. Sublime Text [Электронный ресурс] – Режим доступа до ресурсу: <https://www.sublimetext.com/>

15. Хостинг Україна [Електронний ресурс] – Режим доступу до ресурсу:  
<https://www.ukraine.com.ua/>
16. Ляцевич Д. В., Вороницька В. М. Проектування та розробка ІС «Розклад занять» МАТЕРІАЛИ XIV Всеукраїнської науково-практичної конференції «ІНФОРМАЦІЙНІ ТЕХНОЛОГІЇ В ПРОФЕСІЙНІЙ ДІЯЛЬНОСТІ» / Рівне : РВВ РДГУ. 2021. 220 с. С.129-131

## ДОДАТОК А

Міністерство освіти і науки України  
Національний педагогічний університет ім. М.П. Драгоманова  
Інститут інформаційних технологій і засобів навчання  
Національної академії педагогічних наук  
Рівненський ІТ кластер  
Рівненський державний гуманітарний університет

### СЕРТИФІКАТ

учасника

XIV Всеукраїнської науково-практичної конференції

### “ІНФОРМАЦІЙНІ ТЕХНОЛОГІЇ В ПРОФЕСІЙНІЙ ДІЯЛЬНОСТІ”

1 листопада 2021 року, м. Рівне

**Ляцевич Дмитро Володимирович**

Декан факультету  
математики та інформатики РДГУ



доц. Шахрайчук М.І.