ЗАТВЕРДЖЕНО

Вченою радою факультету
інформаційно-комп'ютерних
технологій
28 серпня 2024 р., протокол № 8
Голова Вченої ради
Тетяна НІКІТЧУК

РОБОЧА ПРОГРАМА НАВЧАЛЬНОЇ ДИСЦИПЛІНИ
ОК24 «Серверні технології та бекенд-розробка»

для здобувачів вищої освіти освітнього ступеня «бакалавр»
спеціальності 122 «Комп'ютерні науки»
освітньо-професійна програма «Комп'ютерна графіка та розробка ігор»
факультет інформаційно-комп'ютерних технологій
кафедра комп'ютерних наук

Схвалено на засіданні кафедри
комп'ютерних наук
26 серпня 2024 р., протокол № 1

Завідувач кафедри
Марина ГРАФ
Гарант освітньо-професійної
програми
Юрій БРОДСЬКИЙ

Розробники: PhD з комп'ютерних наук, завідувач кафедри комп'ютерних наук
Марина ГРАФ

Житомир
2025 – 2026 н.р.

Робоча програма навчальної дисципліни «Серверні технології та бекенд-розробка» для здобувачів вищої освіти освітнього ступеня «бакалавр» спеціальності 122 «Комп’ютерні науки» освітньо-професійна програма «Комп’ютерна графіка та розробка ігор» затверджена Вченою радою факультету інформаційно-комп’ютерних технологій від 28 серпня 2024 р., протокол № 8.

1. Опис навчальної дисципліни

<table>
<tr><td rowspan="2">Найменування показників</td><td rowspan="2">Галузь знань, спеціальність, освітній рівень</td><td colspan="2">Характеристика навчальної дисципліни</td></tr>
<tr><td>денна форма навчання</td><td>заочна форма навчання</td></tr>
<tr><td>Кількість кредитів – 4</td><td>Галузь знань
12 «Інформаційні технології»</td><td>Обов’язкова</td><td>-</td></tr>
<tr><td>Модулів – 1</td><td rowspan="4">Спеціальність
122 “Комп’ютерні науки”</td><td colspan="2">Рік підготовки</td></tr>
<tr><td>Змістових модулів – 3</td><td>2-й</td><td>-</td></tr>
<tr><td rowspan="3">Загальна кількість годин 120</td><td colspan="2">Семестр</td></tr>
<tr><td>2-й</td><td>-</td></tr>
<tr><td colspan="2">Лекції</td></tr>
<tr><td rowspan="11">Тижневих годин для денної форми навчання:
аудиторних – 5
самостійної роботи студента – 2,5</td><td rowspan="11">Освітній ступінь:
бакалавр</td><td>32 год.</td><td>- год.</td></tr>
<tr><td colspan="2">Практичні, семінарські</td></tr>
<tr><td>- год.</td><td>- год.</td></tr>
<tr><td colspan="2">Лабораторні</td></tr>
<tr><td>48 год.</td><td>- год.</td></tr>
<tr><td colspan="2">Самостійна робота</td></tr>
<tr><td>40 год.</td><td>- год.</td></tr>
<tr><td colspan="2"></td></tr>
<tr><td colspan="2">Вид контролю:</td></tr>
<tr><td>Екзамен, курсова робота</td><td>-</td></tr>
</table>

Частка аудиторних занять і частка самостійної та індивідуальної роботи у загальному обсязі годин з навчальної дисципліни становить:

для денної форми навчання – 67% аудиторних занять, 33% самостійної та індивідуальної роботи.

2. Мета та завдання навчальної дисципліни

Метою вивчення навчальної дисципліни є надання студентам необхідних знань з основ мови програмування PHP та знань про PHP-фреймворк Laravel для створення веб-додатків для розробки відповідно до шаблону model-view-controler.

Завдання навчальної дисципліни: ознайомити здобувачів вищої освіти з теоретичними знаннями з бекенду: серверного програмування, набути теоретичні та відпрацювати практичні навички з використання мови програмування PHP та одного з його фреймворків.

За результатами вивчення дисципліни студент повинен:

Знати:

- синтаСКис та алгоритмічні конструкції мови PHP;
- принципи та особливості реалізації об’єктно-орієнтованого програмування у мові програмування PHP;
- читати та розуміти чужий код;
- основи роботи з бекенд-фрейворком.

Вміти:

- писати код за допомогою мови програмування PHP;
- створювати власні, підтримувати та розвивати існуючі веб-додатки;
- застосовувати функціональне та об’єктно-орієнтоване програмування на клієнті за допомогою PHP та його фреймворку Laravel.

Зміст навчальної дисципліни направлений на формування наступних **компетентностей**, визначених стандартом вищої освіти зі спеціальності 122 «Комп’ютерні науки» та освітньо-професійною програмою «Комп’ютерна графіка та розробка ігор»:

ЗК1. Здатність до абстрактного мислення, аналізу та синтезу.

ЗК2. Здатність застосовувати знання у практичних ситуаціях.

ЗК3. Знання та розуміння предметної області та розуміння професійної діяльності.

ЗК6. Здатність вчитися й оволодівати сучасними знаннями.

ЗК9. Здатність працювати в команді.

ЗК10. Здатність бути критичним і самокритичним.

ЗК11. Здатність приймати обґрунтовані рішення.

ЗК12. Здатність оцінювати та забезпечувати якість виконуваних робіт.

ЗК13. Здатність діяти на основі етичних міркувань.

СК1. Здатність до математичного формулювання та досліджування неперервних та дискретних математичних моделей, обґрунтовування вибору методів і підходів для розв’язування теоретичних і прикладних задач у галузі комп’ютерних наук, аналізу та інтерпретування.

СК8. Здатність проєктувати та розробляти програмне забезпечення із застосуванням різних парадигм програмування: узагальненого, об’єктно-орієнтованого, функціонального, логічного, з відповідними моделями, методами й алгоритмами обчислень, структурами даних і механізмами управління.

СК13. Здатність до розробки мережевого програмного забезпечення, що функціонує на основі різних топологій структурованих кабельних систем, використовує комп’ютерні системи і мережі передачі даних та аналізує якість роботи комп’ютерних мереж.

СК14. Здатність застосовувати методи та засоби забезпечення інформаційної безпеки, розробляти й еСКплуатувати спеціальне програмне забезпечення захисту інформаційних ресурсів об’єктів критичної інформаційної інфраструктури.

Отримані знання з навчальної дисципліни «Серверні технології та бекенд-розробка» стануть складовими наступних програмних результатів навчання за спеціальністю 122 «Комп’ютерні науки»:

ПР10. Використовувати інструментальні засоби розробки клієнт-серверних застосувань, проєктувати концептуальні, логічні та фізичні моделі баз даних, розробляти та оптимізувати запити до них, створювати розподілені бази даних, сховища та вітрини даних, бази знань, у тому числі на хмарних сервісах, із застосуванням мов веб-програмування.

ПР15. Застосовувати знання методології та CASE-засобів проєктування складних систем, методів структурного аналізу систем, об'єктноорієнтованої методології проєктування при розробці і дослідженні функціональних моделей організаційно-економічних і виробничотехнічних систем.

ПР16. Розуміти концепцію інформаційної безпеки, принципи безпечного проєктування програмного забезпечення, забезпечувати безпеку комп’ютерних мереж в умовах неповноти та невизначеності вихідних даних.

ПР17. Виконувати паралельні та розподілені обчислення, застосовувати чисельні методи та алгоритми для паралельних структур, мови паралельного програмування при розробці та еСКплуатації паралельного та розподіленого програмного забезпечення.

Під час вивчення навчальної дисципліни здобувачі вищої освіти зможуть отримати наступні Soft skills:

– *комунікативні навички*: письмове, вербальне й невербальне спілкування; ведення дискусії та відстоювання своєї позиції; вміння шукати, аналізувати та використовувати інформацію;

– *уміння виступати привселюдно*: уміння публічно та професійно презентувати результати власних досліджень;

– *гнучкість і адаптивність:* уміння аналізувати ситуацію, орієнтування на вирішення проблеми;

– *особисті якості:* креативне й критичне мислення; етичність, доброчесність, повага до оточуючих, уміння працювати в команді.

3. Програма навчальної дисципліни

МОДУЛЬ 1

Змістовий модуль 1. **Основи PHP**

Тема 1. Вступ до PHP (ЗК1, ЗК2, ЗК3, ЗК6, ЗК9, ЗК10, ЗК11, ЗК 2, ЗК 3, СК , СК8, СК13, СК14, ПР16, ПР17).

- Історія, особливості, сфери застосування.
- Необхідне програмне забезпечення, його встановлення та налаштування.
- СинтаСКис, основні конструкції, змінні, типи даних, оператори.
- Умови та цикли та масиви
- Функції та методи

Тема 2. Функції та масиви в PHP (ЗК1, ЗК2, ЗК3, ЗК6, ЗК9, ЗК10, ЗК11, ЗК12, ЗК13, СК1, СК8, СК13, СК14, ПР16, ПР17).

- Створення та використання функцій.
- Робота з масивами: індеСКація, асоціативні масиви, вбудовані функції.

Тема 3. Об'єктно-орієнтоване програмування в PHP (ЗК1, ЗК2, ЗК3, ЗК6, ЗК9, ЗК10, ЗК11, ЗК12, ЗК13, СК1, СК8, СК13, СК14, ПР15, ПР16, ПР17).

- Класи та об'єкти.
- Властивості та методи.
- Принципи ООП в PHP.

Тема 4. Файли та введення-виведення (ЗК1, ЗК2, ЗК3, ЗК6, ЗК9, ЗК10, ЗК11, ЗК12, ЗК13, СК1, СК8, СК13, СК14, ПР15, ПР16, ПР17).

- Робота з файлами: читання, запис, переміщення.
- Сесії та куки.

Змістовий модуль 2. MVC та веб-розробка

Тема 5. Шаблон MVC (ЗК1, ЗК2, ЗК3, ЗК3, ЗК6, ЗК9, ЗК10, ЗК11, ЗК12, ЗК13 , СК1, СК8, СК13, СК14, ПР15, ПР16, ПР17).

- Концепція, перевари, компоненти.
- Переваги та недоліки MVC порівняно з іншими шаблонами.
- Приклади застосування MVC.
- Роль кожного компонента MVC: модель (Model): взаємодія з базою даних, валідація даних, бізнес-логіка; вид (View): відображення даних користувачеві,

використання шаблонізаторів; Контролер (Controller): обробка запитів користувача, взаємодія між моделлю та видом.

- Принципи розділення відповідальності в MVC: чітке розділення логіки додатку, переваги такого розділення для розробки та підтримки.

Тема 6. HTTP протокол (ЗК1, ЗК2, ЗК3, ЗК6, ЗК9, ЗК10, ЗК11, ЗК12, ЗК13, СК1, СК8, СК13, СК14, ПР15, ПР16, ПР17).

- Основи HTTP: роль HTTP в мережі Інтернет; методи HTTP (GET, POST, PUT, DELETE та інші); статусні коди HTTP (200, 404, 500 та інші); заголовки HTTP.
- Цикл обробки HTTP-запиту: як веб-сервер обробляє HTTP-запит і формує відповідь; роль веб-браузера в цьому процесі;
- Практичні аспекти: створення простих HTTP-запитів за допомогою PHP; аналіз HTTP-заголовків.

Тема 7. Робота з базами даних в PHP, форми (ЗК1, ЗК2, ЗК3, ЗК6, ЗК9, ЗК10, ЗК11, ЗК12, ЗК13, СК1, СК8, СК13, СК14, ПР10,ПР15, ПР16, ПР17).

- Підключення до бази даних за допомогою PDO.
- Мова SQL, нормалізація даних та виконання SQL-запитів.
- Обробка результатів запитів.
- Оптимізація запитів до бази даних: індеСКи, кешування, вибір оптимального типу бази даних для конкретного завдання.
- Обробка даних форм на сервері: отримання даних з форм за допомогою PHP; валідація даних; захист від XSS та інших атак.
- Файли: завантаження файлів на сервер; обробка завантажених файлів; безпека при роботі з файлами.
-

Тема 8. Створення веб-додатку з використанням MVC (ЗК1, ЗК2, ЗК3, ЗК6, ЗК9, ЗК10, ЗК11, ЗК12, ЗК13, СК1, СК8, СК13, СК14, ПР10,ПР15, ПР16, ПР17).

1: Підготовка середовища розробки

- Встановлення необхідного програмного забезпечення:
  - Веб-сервер (Apache, Nginx).
  - База даних (MySQL, PostgreSQL).
  - Інтегроване середовище розробки (IDE) (PhpStorm, Visual Studio Code).
  - Систему контролю версій (Git).
- Створення нового проекту:
  - Ініціалізація проекту в IDE.
  - Структура папок: public, app, storage, resources.

- Підключення бази даних:
    - Створення бази даних та таблиці.
    - Налаштування конфігураційного файлу для підключення до бази.

2: Створення моделі

- Створення класу моделі:
    - Вибір таблиці з бази даних для представлення як моделі.
    - Визначення атрибутів моделі.
    - Реалізація методів для взаємодії з базою даних (CRUD операції).
- Використання ORM (Eloquent):
    - Створення міграцій для створення таблиць.
    - Використання Eloquent для взаємодії з моделями.

3: Створення виду

- Створення шаблонів:
    - Вибір шаблонізатора (Blade).
    - Структура шаблону: шаблон макета, шаблони сторінок.
    - Виведення даних з моделі у шаблон.
- Динамічні сторінки:
    - Передача даних з контролера у вид.
    - Умовні конструкції та цикли в шаблонах.

4: Створення контролера

- Створення контролера:
    - Опис функціональності контролера.
    - Обробка HTTP-запитів.
    - Взаємодія з моделлю та видом.
    - Маршрутизація.
- Приклад простого контролера:
    - Отримання даних з форми.
    - Збереження даних в базу.
    - Перенаправлення на іншу сторінку.

Змістовий модуль 3. PHP-фреймворк Laravel

Тема 9. Вступ до Laravel (ЗК1, ЗК2, ЗК3, ЗК6, ЗК9, ЗК10, ЗК11, ЗК12, ЗК13, СК1, СК8, СК13, СК14, ПР10,ПР15, ПР16, ПР17).

- Архітектура.
- Встановлення.
- Перший додаток.

Тема 10. Маршрутизація: Створення маршрутів, параметри маршрутів, групи маршрутів (ЗК1, ЗК2, ЗК3, ЗК6, ЗК9, ЗК10, ЗК11, ЗК12, ЗК13, СК1, СК8, СК13, СК14, ПР10,ПР15, ПР16, ПР17).

- Створення маршрутів.
- Параметри маршрутів.
- Групи маршрутів.

Тема 11. Blade: Шаблонізатор Laravel, директиви, секції, компонент (ЗК1, ЗК2, ЗК3, ЗК6, ЗК9, ЗК10, ЗК11, ЗК12, ЗК13, СК1, СК8, СК13, СК14, ПР10,ПР15, ПР16, ПР17).

- Шаблонізатор Laravel.
- Директиви.
- Секції.
- Компонент.

Тема 12. Eloquent ORM: Модель даних, міграції, сидератори, відносини між моделями (ЗК1, ЗК2, ЗК3, ЗК6, ЗК9, ЗК10, ЗК11, ЗК12, ЗК13, СК1, СК8, СК13, СК14, ПР10,ПР15, ПР16, ПР17).

- Модель даних.
- Міграції.
- Сидератори.
- Відносини між моделями.

Тема 13. Контролери: Створення контролерів, методи, залежності (ЗК1, ЗК2, ЗК3, ЗК6, ЗК9, ЗК10, ЗК11, ЗК12, ЗК13, СК1, СК8, СК13, СК14, ПР10,ПР15, ПР16, ПР17).

- Створення контролерів.
- Методи.
- Залежності.

Тема 14. Запити та відповіді: HTTP-запити, middleware, валідація даних (ЗК1, ЗК2, ЗК3, ЗК6, ЗК9, ЗК10, ЗК11, ЗК12, ЗК13, СК1, СК8, СК13, СК14, ПР10,ПР15, ПР16, ПР17).

- HTTP-запити.
- Missleware.
- Валідація даних.

Тема 15. Аутентифікація та авторизація: Вбудовані механізми, створення власних (ЗК1, ЗК2, ЗК3, ЗК6, ЗК9, ЗК10, ЗК11, ЗК12, ЗК13, СК1, СК8, СК13, СК14, ПР10,ПР15, ПР16, ПР17).

- Вбудовані механізми.
- Створення власних.

Тема 16. Керування помилками та виключеннями: Обробка виключень, логування (ЗК1, ЗК2, ЗК3, ЗК6, ЗК9, ЗК10, ЗК11, ЗК12, ЗК13, СК1, СК8, СК13, СК14, ПР10,ПР15, ПР16, ПР17).

- Обробка виключень.
- Логування.

4. Структура (тематичний план) навчальної дисципліни

Назви змістових модулів і тем	Кількість годин					
	Денна форма					
	усього	у тому числі				
		лекції	практичні	лабораторні	інд. робота	Сам ост. робота
МОДУЛЬ 1						
Змістовий модуль 1. Основи PHP						
Тема 1. Вступ до PHP	4	2	-	-	-	2
Тема 2. Функції та масиви в PHP	8	2	-	4	-	2
Тема 3. Об’єктно-орієнтоване програмування в PHP	10	2	-	4	-	4
Тема 4. Файли та введення-виведення	8	2	-	4	-	2
Разом за змістовий модуль 1	30	8	0	12	0	10
Змістовний модуль 2. MVC та веб-розробка						
Тема 5. Шаблон MVC	10	2	-	4	-	4
Тема 6. HTTP протокол	8	2	-	4	-	2
Тема 7. Робота з базами даних в PHP, форми	8	2	-	4	-	2
Тема 8. Створення веб-додатку з використанням MVC	10	2	-	4	-	4
Разом за змістовий модуль 2	36	8	0	16	0	12
Змістовний модуль 3. PHP-фреймворк Laravel						
Тема 9. Вступ до Laravel	6	2	-	2	-	2
Тема 10. Маршрутизація: Створення маршрутів, параметри маршрутів, групи маршрутів	6	2	-	2	-	2

Тема 11. Blade: Шаблонізатор Laravel, директиви, секції, компонент	8	2	-	2	-	4
Тема 12. Eloquent ORM: Модель даних, міграції, сидератори, відносини між моделями	8	2	-	2	-	4
Тема 13. Контролери: Створення контролерів, методи, залежності	8	2	-	2	-	4
Тема 14. Запити та відповіді: HTTP-запити, middleware, валідація даних	8	2	-	2	-	4
Тема 15. Аутентифікація та авторизація: Вбудовані механізми, створення власних	10	2	-	4	-	4
Тема 16. Керування помилками та виключеннями: Обробка виключень, логування	9	2	-	3	-	4
Разом за змістовий модуль 3	63	16	-	19	-	28
МОДУЛЬНИЙ КОНТРОЛЬ	1		-	1	-	-
Усього годин	120	32	-	48	-	40

5. Теми лабораторних занять

№	Назва теми	Кількість годин
1.	Створення веб-сайту з використанням HTML-форм та PHP**.** Створення інженерного калькулятора: прийняття чисел та операцій, виконання обчислень, виведення результату, обробка помилок	4
2.	Створення системи авторизації користувачів: реєстрація, авторизація, сесій, захист від SQL-ін’єкцій	4
3.	Робота з базами даних MySQL за допомогою PHP: підключення до бази даних, виконання SQL-запитів, результати запитів, оптимізація запитів	4
4.	Створення простого веб-додатоку за шаблоном MVC	4
5.	Реалізація CRUD-операції над базою даних: створення, читання, оновлення, видалення даних в базі даних через веб-інтерфейс	4
6.	Створити форму з валідацією даних: перевірка введених користувачем даних на коректність, використання вбудованих механізмів валідації або сторонніх бібліотек	4
7.	Опрацювання завантажених файлів: обробка, перевірка типів файлів, збереження файлів на сервері.	4
8.	Встановлення та налаштування Laravel. Створення простого веб-додаток на Laravel	2
9.	Використання Blade для створення шаблонів, використання директив, секцій, компонентів	2
10.	Створити форму з валідацією даних.	2
11.	Реалізувати CRUD-операції над моделлю даних.	2
12.	Створення системи авторизації з ролями та дозволами: використання	2

	вбудованого механізму Laravel для авторизації, створення власних провайдерів аутентифікації	
13.	Створення API для мобільного додатку: Створення RESTful API за допомогою Laravel, використання ресурсів, серіалізація даних	2
14.	Використати кешування для покращення продуктивності додатку: використання різних типів кешу (файловий, Redis), оптимізація запитів до бази даних. Написання юніт-тестів для перевірки коректності роботи контролерів та моделей.	4
15.	Створення адмін-панелі: вибір готового рішення або створення власної адмін-панелі для керування додатком. Реалізація пошуку по даним за допомогою Elasticsearch або інших інструментів	3
	Модульний контроль	1
РАЗОМ		**48**

6. Завдання для самостійної роботи

№ з/п	Назва теми	Кількість годин
1.	Тема 1. ООП в PHP: Абстрактні класи, інтерфейси, трейти, магічні методи, патерни проектування	2
2.	Тема 2. Рекурсивний обхід директорій, робота з різними форматами файлів, завантаження та обробка файлів. Регулярні вирази: складні шаблони, валідація даних, парсинг теСКту	2
3.	Тема 3. Помилки та виключення: створення власних класів виключень, обробка різних типів помилок, логування. Сесії та куки: gорівняння сесій та куків, безпечне використання сесій, захист від CSRF	4
4.	Тема 4. Функціональне програмування в PHP: використання анонімних функцій, лямбда-виразів, вищих порядків функцій. Розширення PHP: створення власних розширень для PHP. Оптимізація коду: профілювання, кешування, мініфікація, об'єднання файлів	2
5.	Тема 5. Створення власного простого фреймворку MVC: Розуміння основних принципів MVC, реалізація базових компонентів (модель, вид, контролер) без використання готових фреймворків	4
6.	Тема 6. Правильне оброблення помилок та виключень, створення власних класів виключень	2
7.	Тема 7. Вивчення всіх можливостей PDO для роботи з базами даних, оптимізація запитів, використання підготовлених запитів	2
8.	Тема 8. Порівняння сесій та куків, використання для зберігання стану користувача, забезпечення безпеки	4
9.	Тема 9. Laravel. Вивчення всіх типів маршрутів, груп маршрутів, middleware, регулярні вирази в маршрутах. Створення складних форм, валідація даних, використання різних типів полів, завантаження файлів	2

10.	Тема 10. Laravel. Дослідження всіх можливостей Eloquent: відносини між моделями, глобальні scope, мутатори, аСКесори	2
11.	Тема 11. Laravel. Blade: розширені можливості: створення власних директив, компонентів, використання слотів. Middleware: створення власних middleware для автентифікації, авторизації, логування, кешування	4
12.	Тема 12. Тестування: написання різних типів тестів (юніт-тести, інтеграційні тести), використання PHPUnit. Кешування: різні типи кешу в Laravel, стратегії кешування, оптимізація продуктивності	4
13.	Тема 13. Laravel. API: створення RESTful API з використанням Laravel, версіонування API, документація	4
14.	Тема 14. Розгортання Laravel-додатків на різних платформах (shared hosting, VPS, контейнери), використання інструментів автоматизації (Ansible, Puppet) (деплой)	4
15.	Тема 15. Захист додатків від найпоширеніших вразливостей (SQL injection, XSS, CSRF), використання безпечних практик. Використання WebSockets (Pusher, Socket.IO) для створення чатів, систем обміну повідомленнями.	4
16.	Тема 16. Інтеграція з платіжними системами, управління корзиною, обробка замовлень. Використання готових рішень (Voyager, Nova) або створення власної адмін-панелі.	4
	Всього	40

7. Курсова робота

Метою курсової є формування у студентів практичних навичок в створенні клієнтської та серверної частин веб-сайтів та веб-додатків різного призначення відповідно до сучасних норм і стандартів, розвиток професійної компетентності в галузі інформаційних технологій, здатності застосовувати теоретичні знання для вирішення прикладних завдань у майбутній практичній діяльності.

Орієнтований перелік тем:

Веб-додатки різної спрямованості

1. Розробка CRM-системи для управління клієнтами на Laravel.
2. Створення платформи для онлайн-навчання з використанням PHP та Laravel.
3. Розробка соціальної мережі з функціоналом мікроблогінгу на базі Laravel.
4. Створення веб-додатку для управління проектами на PHP Framework Laravel.
5. Розробка системи бронювання квитків на заходи за допомогою Laravel.
6. Створення дошки оголошень з можливістю фільтрації та пошуку на Laravel.
7. Розробка веб-сайту для новинного агентства на PHP та Laravel.
8. Створення платформи для обміну файлами з використанням Laravel.

9. Розробка інтернет-магазину з інтеграцією платіжних систем на Laravel.
10. Створення веб-додатку для автоматизації роботи call-центру на PHP.
11. Розробка системи управління персоналом (HRM) на базі Laravel.
12. Створення веб-додатку для проведення опитувань та голосувань на Laravel.
13. Розробка платформи для онлайн-аукціонів з використанням PHP та Laravel.
14. Створення веб-додатку для управління подіями на Laravel.
15. Розробка системи для ведення блогу з можливістю коментування на Laravel.

API та інтеграції

16. Розробка RESTful API для мобільного додатку з використанням Laravel.
17. Створення API для інтеграції з сервісами Google Maps на PHP.
18. Розробка API для роботи з платіжною системою PayPal на Laravel.
19. Створення API для інтеграції з соціальними мережами (Facebook, Twitter) на PHP.
20. Розробка API для роботи з сервісами обробки зображень (Cloudinary) на Laravel.
21. Створення API для інтеграції з CRM-системами (Salesforce) на PHP.
22. Розробка API для роботи з базами даних NoSQL (MongoDB) на Laravel.
23. Створення API для інтеграції з сервісами аналітики (Google Analytics) на PHP.
24. Розробка API для роботи з сервісами email-розсилок (Mailchimp) на Laravel.
25. Створення API для інтеграції з системами SMS-сповіщень на PHP.

Безпека та оптимізація

26. Безпека веб-додатків на PHP: методи захисту від SQL-ін'єкцій та XSS.
27. Оптимізація продуктивності Laravel-додатків: кешування та черги завдань.
28. Аутентифікація та авторизація користувачів у Laravel: реалізація різних методів.
29. Захист від CSRF-атак у веб-додатках на PHP.
30. Використання SSL-сертифікатів для захисту веб-додатків на Laravel.
31. Моніторинг та логування помилок у PHP-додатках.
32. Тестування безпеки веб-додатків на Laravel: пентестинг та аудит коду.
33. Захист від DDoS-атак на веб-сервери з PHP-додатками.
34. Використання Captcha для захисту від ботів у Laravel-додатках.
35. Безпечне зберігання паролів користувачів у PHP-додатках.

Розширені теми

36. Розробка веб-додатку з використанням WebSocket для онлайн-чату на Laravel.
37. Створення системи управління контентом (CMS) на базі Laravel.

38. Розробка веб-додатку з використанням технологій машинного навчання на PHP.
39. Інтеграція з платіжними системами (LiqPay, Privat24) у Laravel-додатках.
40. Використання Redis для кешування даних у PHP-додатках.
41. Розробка веб-додатку з використанням Docker для розгортання на серверах.
42. Створення системи аналітики та звітів для веб-додатків на Laravel.
43. Розробка веб-додатку з використанням GraphQL на PHP.
44. Використання Composer для управління залежностями у Laravel-проектах.
45. Розробка веб-додатку з використанням мікросервісної архітектури на PHP.

Теми, що поєднують PHP та інші технології

46. Інтеграція Laravel-додатку з фронтенд-фреймворками (Vue.js, React).
47. Використання PHP та Laravel для роботи з базами даних PostgreSQL.
48. Розробка веб-додатку з використанням PHP та Elasticsearch для пошуку.
49. Інтеграція Laravel-додатку з сервісами AWS (Amazon S3, EC2).
50. Використання PHP та Laravel для створення Telegram-ботів.
51. Розробка веб-додатку з використанням PHP та RabbitMQ для обробки черг.
52. Інтеграція Laravel-додатку з системою управління версіями Git.
53. Використання PHP та Laravel для створення мобільних додатків за допомогою Cordova або React Native.

Розподіл балів за виконання курсової роботи

Пояснювальна записка	Програмний код	Захист роботи	Сума
до 25 балів	до 50 балів	до 25 балів	100

8. Індивідуальні самостійні завдання

Оскільки кожен студент має різний рівень підготовки, пропонується обрати індивідуальні завдання на основі матриці складності та тематики.

Матриця складності та тематики:

Рівень	Приклади завдань
Початковий	**Створити блог на Laravel:** Реалізувати функціонал додавання, редагування та видалення статей, категорій, коментарів. Використати Markdown для форматування теСКту
	Розробити систему авторизації з ролями: Реалізувати реєстрацію, авторизацію, відновлення пароля, різні ролі користувачів (адміністратор, модератор, користувач) з різними правами доступу
	Створити онлайн-магазин з базовим функціоналом: Додати в кошик, оформлення замовлення, обробка платежів (можна використовувати платіжні шлюзи-імітатори)
	Реалізувати систему обміну повідомленнями: Використовувати WebSocket для створення чату в реальному часі
	Створити систему управління завданнями: Додати функціонал створення, редагування, видалення завдань, призначення відповідальних, встановлення дедлайнів
Середній	**Розробити REST API для мобільного додатку:** Створити API для отримання та оновлення даних, забезпечити автентифікацію та авторизацію
	Інтегрувати систему пошуку (Elasticsearch): Реалізувати повнотеСКтовий пошук по даним в додатку
	Створити систему логування: Записувати інформацію про дії користувачів, помилки системи в лог-файли
	Реалізувати систему кешування: Використовувати кешування для прискорення роботи додатку, зменшення навантаження на базу даних
	Створити власний пакет для Laravel: Розробити пакет для розв’язання конкретної задачі (наприклад, валідація форм, генерація звітів)
Продвинутий	Розробити систему електронної комерції з підтримкою різних валют, податків, способів доставки

	Створити систему управління версіями документів
	Реалізувати систему обробки платежів з використанням різних платіжних шлюзів
	Розробити систему аналітики веб-сайту
	Створити додаток для управління проектами з використанням Agile-методологій
	Інтегрувати штучний інтелект в додаток (наприклад, чат-бот, рекомендаційна система)

9. Методи навчання

Під час викладання навчальної дисципліни використовуються наступні методи навчання.

Результат навчання	**Методи навчання**
ПР10. Використовувати інструментальні засоби розробки клієнт-серверних застосувань, проєктувати концептуальні, логічні та фізичні моделі баз даних, розробляти та оптимізувати запити до них, створювати розподілені бази даних, сховища та вітрини даних, бази знань, у тому числі на хмарних сервісах, із застосуванням мов веб-програмування.	● Вербальні методи (лекція, пояснення) ● Наочні методи (презентація) ● Практичні методи (вирішення кейсів) ● Дискусійний метод ● Дослідницький метод ● Проблемний метод ● Методи самостійної роботи (анотування опрацьованого матеріалу, виконання завдань, проведення розрахунків)
ПР15. Застосовувати знання методології та CASE-засобів проєктування складних систем, методів структурного аналізу систем, об'єктноорієнтованої методології проєктування при розробці і дослідженні функціональних моделей організаційно-економічних і виробничотехнічних систем.	● Вербальні методи (лекція, пояснення) ● Наочні методи (презентація) ● Практичні методи (вирішення кейсів) ● Дискусійний метод ● Дослідницький метод ● Проблемний метод ● Методи самостійної роботи (анотування опрацьованого матеріалу, виконання завдань, проведення розрахунків)
ПР16. Розуміти концепцію інформаційної безпеки, принципи безпечного проєктування програмного забезпечення, забезпечувати безпеку комп’ютерних мереж в умовах неповноти та невизначеності вихідних даних.	● Вербальні методи (лекція, пояснення) ● Наочні методи (презентація) ● Практичні методи (вирішення кейсів) ● Дискусійний метод ● Дослідницький метод ● Проблемний метод ● Методи самостійної роботи (анотування опрацьованого матеріалу, виконання завдань, проведення розрахунків)
ПР17. Виконувати паралельні та	● Вербальні методи (лекція, пояснення)

розподілені обчислення, застосовувати чисельні методи та алгоритми для паралельних структур, мови паралельного програмування при розробці та еСКплуатації паралельного та розподіленого програмного забезпечення.	● Наочні методи (презентація) ● Практичні методи (вирішення кейсів) ● Дискусійний метод ● Дослідницький метод ● Проблемний метод ● Методи самостійної роботи (анотування опрацьованого матеріалу, виконання завдань, проведення розрахунків)

90. Методи контролю

Під час вивчення дисципліни використовуються наступні методи контролю: поточне та підсумкове тестування за теоретичним матеріалом, захист лабораторних робіт у формі співбесіди.

Результат навчання	**Методи контролю**
ПР10. Використовувати інструментальні засоби розробки клієнт-серверних застосувань, проєктувати концептуальні, логічні та фізичні моделі баз даних, розробляти та оптимізувати запити до них, створювати розподілені бази даних, сховища та вітрини даних, бази знань, у тому числі на хмарних сервісах, із застосуванням мов івеб-програмування.	● Усне опитування, участь у дискусії, відповіді на проблемні запитання ● Перевірка виконання та захист лабораторних робіт ● Перевірка виконання завдань модульного контролю ● Перевірка виконання і захист курсової роботи ● Екзамен
ПР15. Застосовувати знання методології та CASE-засобів проєктування складних систем, методів структурного аналізу систем, об'єктноорієнтованої методології проєктування при розробці і дослідженні функціональних моделей організаційно-економічних і виробничотехнічних систем.	● Усне опитування, участь у дискусії, відповіді на проблемні запитання ● Перевірка виконання та захист лабораторних робіт ● Перевірка виконання завдань модульного контролю ● Перевірка виконання і захист курсової роботи ● Екзамен
ПР16. Розуміти концепцію інформаційної безпеки, принципи безпечного проєктування програмного забезпечення, забезпечувати безпеку комп’ютерних мереж в умовах неповноти та	● Усне опитування, участь у дискусії, відповіді на проблемні запитання ● Перевірка виконання та захист лабораторних робіт ● Перевірка виконання завдань

невизначеності вихідних даних.	модульного контролю ● Перевірка виконання і захист курсової роботи ● Екзамен
ПР17. Виконувати паралельні та розподілені обчислення, застосовувати чисельні методи та алгоритми для паралельних структур, мови паралельного програмування при розробці та еСКплуатації паралельного та розподіленого програмного забезпечення.	● Усне опитування, участь у дискусії, відповіді на проблемні запитання ● Перевірка виконання та захист лабораторних робіт ● Перевірка виконання завдань модульного контролю ● Перевірка виконання і захист курсової роботи ● Екзамен

10. Оцінювання результатів навчання здобувачів вищої освіти

Оцінювання результатів навчання здобувачів вищої освіти з навчальної дисципліни здійснюється відповідно до Положення про оцінювання результатів навчання здобувачів вищої освіти у Державному університеті «Житомирська політехніка» та розподілу балів, що наведений нижче.

Система оцінювання результатів навчання здобувачів вищої освіти з навчальної дисципліни включає: поточний, модульний та підсумковий контроль.

Поточний контроль проводиться для оцінювання рівня засвоєння знань, формування умінь і навичок здобувачів вищої освіти впродовж вивчення ними матеріалу модуля (змістових модулів) навчальної дисципліни. Поточний контроль здійснюється під час проведення навчальних занять.

Модульний контроль проводиться з метою оцінювання результатів навчання здобувачів вищої освіти за модуль (змістові модулі) навчальної дисципліни. Модульний контроль проводиться під час навчального заняття після завершення вивчення матеріалу модуля (змістових модулів) навчальної дисципліни. Модульний контроль здійснюється у формі у формі підсумкового тестування.

Підсумковий контроль проводиться для підсумкового оцінювання результатів навчання здобувачів вищої освіти з навчальної дисципліни. Підсумковий контроль здійснюється після завершення вивчення навчальної дисципліни. Підсумковий контроль проводиться у формі екзамену. Процедура складання екзамену визначена у Положенні про організацію освітнього процесу у Державному університеті «Житомирська політехніка».

Розподіл балів з навчальної дисципліни

Види робіт здобувача вищої освіти	Кількість балів за семестр
Для здобувача денної форми навчання	
Виконання завдань поточного контролю	60
Виконання завдань модульного або підсумкового контролю	40
Підсумкова семестрова оцінка	**100**

Розподіл балів за виконання завдань поточного контролю

Види робіт здобувача вищої освіти	Кількість балів за семестр	
	денна форма	заочна форма
Виконання завдань під час навчальних занять	60	
Виконання науково-дослідної роботи та інших видів робіт (**додаткові – заохочувальні бали**)[3]: 1. Участь у студентських предметних олімпіадах, Всеукраїнському конкурсі студентських наукових робіт, грантах, науково-дослідних проектах 2. Підготовка наукових статей, тез доповідей наукових конференцій 3. Інші види робіт (проходження тематичних курсів на освітніх платфомах: Coursera, UDEMY тощо)	до 15	-
Разом за виконання завдань поточного контролю	**60**	**-**

Розподіл балів за виконання завдань під час навчальних занять

Види робіт здобувача вищої освіти[1]	Кількість балів за семестр	
	денна форма	заочна форма
Відповіді (виступи) на заняттях	12	-
Виконання та захист лабораторних робіт	48	-
Разом за виконання завдань під час навчальних занять	**60**	**-**

З метою застосування цілих чисел для оцінювання результатів роботи здобувачів під час навчальних занять може використовуватися 100-бальна шкала оцінювання щодо кожного окремо виду робіт. Розрахунок загальної кількості балів, які здобувач

може набрати за результатами роботи під час навчальних занять протягом семестру, проводиться за формулою:

$$Р_{НЗ} = \sum(Р_i \times ВК_i) \times К_{НЗ}, \quad (1)$$

де $Р_{НЗ}$ – загальна кількість балів, набраних здобувачем за виконання завдань під час навчальних занять за семестр;

$Р_i$ – кількість набраних здобувачем балів за семестр за виконання і-го виду робіт під час навчальних занять (за 100-бальною шкалою);

$ВК_i$ – ваговий коефіцієнт за виконання і-го виду робіт під час навчальних занять. Значення вагових коефіцієнтів розраховуються шляхом ділення кількості балів, яка передбачена за виконання окремого виду робіт під час навчальних занять, на сумарну кількість балів за виконання усіх видів робіт під час навчальних занять за семестр;

$К_{НЗ}$ – коригувальний коефіцієнт, який визначається шляхом ділення кількості балів, що передбачена за виконання завдань під час навчальних занять за семестр, на 100 балів.

Розподіл балів за виконання завдань модульного контролю

Види робіт здобувача вищої освіти денної форми навчання	Кількість балів за семестр
Виконання завдань модульного контролю	40
Разом за виконання завдань модульного контролю	**40**

Якщо здобувач вищої освіти денної форми навчання виконав завдання модульного контролю і з урахуванням отриманих балів за поточний контроль набрав у сумі 60 балів або більше, він може погодити дану оцінку в електронному кабінеті і вона стане семестровою оцінкою за вивчення навчальної дисципліни.

Якщо здобувач вищої освіти денної форми навчання під час вивчення навчальної дисципліни набрав 60 балів або більше і бажає покращити свій результат успішності, він проходить процедуру підсумкового контролю у формі екзамену. Набрані бали за виконання завдань підсумкового контролю, а також бали за поточний контроль сумуються і формується семестрова оцінка з навчальної дисципліни. Бали, які здобувач

вищої освіти набрав за виконання завдань модульного контролю, при цьому не враховуються під час розрахунку семестрової оцінки з навчальної дисципліни.

Здобувач вищої освіти допускається до процедури підсумкового контролю у формі екзамену, якщо за виконання завдань поточного контролю набрав 20 балів або більше.

Якщо здобувач вищої освіти за результатами поточного контролю набрав 15−19 балів, він отримує право за власною заявою опанувати окремі теми (змістові модулі) навчальної дисципліни понад обсяги, встановлені навчальним планом освітньої програми. Вивчення окремих складових навчальної дисципліни понад обсяги, встановлені навчальним планом освітньої програми, здійснюється у вільний від занять здобувача вищої освіти час.

Якщо здобувач вищої освіти за результатами поточного контролю набрав від 0 до 14 балів (включно), він вважається таким, що не виконав вимоги робочої програми навчальної дисципліни та має академічну заборгованість. Здобувач вищої освіти отримує право за власною заявою опанувати навчальну дисципліну у наступному семестрі понад обсяги, встановлені навчальним планом освітньої програми.

Процедура надання додаткових освітніх послуг здобувачу вищої освіти з метою вивчення навчального матеріалу дисципліни понад обсяги, встановлені навчальним планом освітньої програми, визначена у Положенні про надання додаткових освітніх послуг здобувачам вищої освіти в Державному університеті «Житомирська політехніка».

Визнання результатів навчання, набутих у неформальній та/або інформальній освіті

Визнання результатів навчання, набутих у неформальній та/або інформальній освіті в рамках окремих тем навчальної дисципліни, здійснюється викладачем за зверненням здобувача вищої освіти та представленням документів, які підтверджують результати навчання (сертифікати, свідоцтва, скріншоти тощо). Рішення про визнання та оцінка за відповідну частину освітнього компонента приймається викладачем за результатами співбесіди зі здобувачем вищої освіти.

Визнання результатів навчання, набутих у неформальній та/або інформальній освіті в рамках цілого освітнього компонента, здійснюється за процедурою, яка визначена у Положенні про організацію освітнього процесу у Державному університеті «Житомирська політехніка».

Шкала оцінювання

Шкала ЄКТС	Національна шкала	100-бальна шкала

A	Відмінно	90-100
B	Добре	82-89
C		74-81
D	Задовільно	64-73
E		60-63
FX	Незадовільно	35-59
F		0-34

11. Глосарій

№ з/п	Термін державною мовою	Відповідник англійською мовою
1.	Серверні технології	Server-side technologies
2.	Бекенд-розробка	Back-end development
3.	Мова програмування PHP	PHP programming language
4.	Модель-Вид-Контролер (MVC)	Model-View-Controller (MVC)
5.	Фреймворк Laravel	Laravel framework
6.	Веб-додаток	Web application
7.	Веб-сервер	Web server
8.	База даних	Database
9.	Мова запитів (SQL)	Structured Query Language
10.	Запит до бази даних	Database query
11.	Таблиця (бази даних)	Table (database)
12.	Запис (бази даних)	Record (database)
13.	Поле (бази даних)	Field (database)
14.	Індекс (бази даних)	Index (database)
15.	Нормалізація даних	Data normalization
16.	Об'єктно-орієнтоване програмування (ООП)	Object-oriented programming (OOP)
17.	Клас	Class
18.	Об'єкт	Object
19.	Властивість	Property
20.	Метод	Method
21.	Наслідування	Inheritance

22.	Поліморфізм	Polymorphism
23.	Абстракція	Abstraction
24.	Інкапсуляція	Encapsulation
25.	Функція	Function
26.	Змінна	Variable
27.	Масив	Array
28.	Цикл	Loop
29.	Умова	Condition
30.	Оператор	Operator
31.	Маршрутизація	Routing
32.	Контролер	Controller
33.	Модель	Model
34.	Вид	View
35.	Шаблон (шаблонізатор)	Template (templating engine)
36.	Проміжне програмне забезпечення	Middleware
37.	Аутентифікація	Authentication
38.	Авторизація	Authorization
39.	Сесія	Session
40.	Куки	Cookie
41.	Об’єктно-реляційне відображення (ORM)	Object-Relational Mapping
42.	Міграція	Migration
43.	Сидератор	Seeder
44.	Інтерфейс прикладного програмування	Application Programming Interface
45.	Підхід до архітектури мережевих протоколів «передача репрезентативного стану» (REST)	Representational State Transfer
46.	Текстовий формат обміну даними між комп’ютерами (JSON)	JavaScript Object Notation
47.	Розширювана мова розмітки (XML)	Extensible Markup Language
48.	Версіювання коду	Version control
49.	Система контролю версій	Git
50.	Деплой	Deployment

12. Рекомендована література

Основна література

1. Васильєв О. Характеристики Програмування мовою PHP : навч. посіб. Ліра-К, 2022. 368 с. ISBN 978- 617- 520- 322- 4.
2. Vasilev A. PHP by Example: A Practical Guide to Creating Web Applications with PHP First Edition. New York : Apress, 2024. 442 с.
3. MacIntyre P., Tatroe K. Programming PHP: Creating Dynamic Web Pages 4th Edition. O'Reilly, 2020. 540 с.
4. Фримен Е., Робсон Е. Head first. Програмування на Javascript. Харків: Фабула, 2022. С. 672.
5. Thies C. Arntzen, Stig Bakken, Shane Caraveo, Andi Gutmans, Rasmus Lerdorf, Sam Ruby, Sascha Schumann, Zeev Suraski, Jim Winstead, Andrei Zmievski. *PHP*. *PHP: Hypertext Preprocessor*. URL: https://www.php.net/ (дата звернення: 28.08.2024).
6. Laravel Holdings Inc. *Laravel - The PHP Framework For Web Artisans*. *The PHP Framework for Web Artisans*. URL: https://laravel.com/ (дата звернення: 28.08.2024).

Допоміжна література

1. Петренко, О. І. Застосування фреймворку Laravel для розробки веб-додатків // Вісник Хмельницького національного університету. — 2020. — № 3. — С. 123-129.
2. Nixon R. Learning PHP, MySQL & Jav * aScript: A Step-by-Step Guide to Creating Dynamic Websites. O'Reilly, 2024. 650 с. ISBN 9781098152352.
3. Lockhart J. Modern PHP: New Features and Good Practices 1st Edition. O'Reilly, 2015. 270 с.
4. Otwell T. Laravel: From Apprentice To Artisan. Advanced Architecture With Laravel 4. 2013. 67 с. URL: https://online.fliphtml5.com/wuxi/iemd/#p=1.
5. Ілля Кантор. Сучасний підручник з JavaScript [Електронний ресурс] / Ілля Кантор. – 2025. – Режим доступу до ресурсу: https://uk.javascript.info/.
6. Matrunchyk S. Vue.js [Електронний ресурс] / Matrunchyk S. – 2025. – Режим доступу до ресурсу: https://ua.vuejs.org/guide/introduction.html.
7. Wang S. Learn to code — for free. Build projects. Earn certifications. [Електронний ресурс] / Wang S., Chima S., Bostian E. – Режим доступу до ресурсу: https://www.freecodecamp.org/.
8. Коцюк Ю.А. Бази даних.- Острог, 2019.
   Ситник Н.В. Проектування баз і сховищ даних: Навчальний посібник.- К.: КНЕУ, 2019.

13. Інформаційні ресурси в Інтернеті

1. Офіційний сайт PHP: https://www.php.net/
2. Документація Laravel**:** https://laravel.com/docs
3. Coursera**:** https://www.coursera.org/
4. Udemy: https://www.udemy.com/
5. edX: https://www.edx.org/
6. Prometheus**:** https://prometheus.org.ua/
7. Stack Overflow: https://stackoverflow.com/
8. W3School: https://www.w3schools.com/php/default.asp
9. Ознайомлення з паттерном MVC: https://javarush.com/ua/groups/posts/uk.2536.chastina-7-oznayomlennja-z-paternom-mvc-model-view-controller