

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
ВІДОКРЕМЛЕНИЙ СТРУКТУРНИЙ ПІДРОЗДІЛ
«ФАХОВИЙ КОЛЕДЖ ПРОМИСЛОВОЇ АВТОМАТИКИ
ТА ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ
ОДЕСЬКОГО НАЦІОНАЛЬНОГО ТЕХНОЛОГІЧНОГО УНІВЕРСИТЕТУ»

Циклова комісія комп'ютерних наук та інженерії програмного забезпечення

ЗАТВЕРДЖУЮ
директор ФКПАІТ ОНТУ
підписано _____ Ольга ЄПУР
30.08.2023 року

ПРОГРАМА НАВЧАЛЬНОЇ ДИСЦИПЛІНИ

WEB-ПРОГРАМУВАННЯ

обов'язкова

Освітньо-професійна програма Інженерія програмного забезпечення

Код та найменування спеціальності 121 «Інженерія програмного
забезпечення»

Шифр та найменування галузі знань 12 «Інформаційні технології»

Мова навчання українська

Розроблено та забезпечується: цикловою комісією Комп'ютерних наук та інженерії програмного забезпечення ВСП «Фаховий коледж промислової автоматики та інформаційних технологій Одеського національного технологічного університету»

Розробники:

- Тетяна КОСТИРЕНКО, викладач вищої кваліфікаційної категорії ФКПАІТ ОНТУ

Розглянуто та схвалено на засіданні циклової комісії Комп'ютерних наук та інженерії програмного забезпечення

Протокол №1 від 28.08.2023 р.

Голова циклової комісії

підписано
(підпис)

Тетяна КОСТИРЕНКО
(Власне ім'я, ПРИЗВИЩЕ)

Гарант освітньо-професійної програми

підписано
(підпис)

Тетяна КОСТИРЕНКО
(Власне ім'я, ПРИЗВИЩЕ)

Розглянуто та схвалено Методичною радою ФКПАІТ ОНТУ

Протокол №1 від 29.08.2023 р.

Голова Методичної ради ФКПАІТ ОНТУ підписано Вікторія ОКСАНІЧЕНКО
(підпис)

1. Пояснювальна записка

Вступ

Програма вивчення навчальної дисципліни «WEB-програмування» складена відповідно до освітньо-професійної програми підготовки молодших спеціалістів спеціальності 121 «Інженерія програмного забезпечення» та в стандарту фахової передвищої освіти із спеціальності 121 Інженерія програмного забезпечення.

Предмет вивчення

Предметом вивчення навчальної дисципліни є створення сучасних та ефективних веб-застосунків, вивчення основи програмування на мові PHP, СУБД MySQL.

Міждисциплінарні зв'язки

Попередні – Конструювання програмного забезпечення, Комп'ютерні мережі, послідовні – Розробка UX/UI дизайну.

Мета та завдання навчальної дисципліни

Метою дисципліни "WEB-програмування" є надання студентам необхідних теоретичних знань та практичних навичок для розробки динамічних веб-застосунків. Дисципліна спрямована на формування у студентів глибокого розуміння концепцій та технологій, що стосуються розробки веб-додатків, а також на розвиток їхніх умінь використовувати сучасні інструменти та фреймворки.

Основні завдання:

1. Оволодіння мовою програмування PHP:
 - Розуміння синтаксису та основних конструкцій мови PHP.
 - Вивчення можливостей та обмежень PHP для розробки веб-застосунків.
2. Вивчення принципів взаємодії PHP та HTML/CSS:
 - Розробка динамічних веб-сторінок, використовуючи PHP для вставки коду в HTML та забезпечення взаємодії з користувачем.
3. Створення та оптимізація баз даних з використанням PHPMyAdmin:
 - Навчання створенню та налагодженню баз даних засобами PHPMyAdmin.
 - Оптимізація та управління базами даних для підтримки веб-застосунків.
4. Розробка серверно-орієнтованих веб-застосунків:
 - Вивчення та практика розробки серверних скриптів на мові PHP для обробки запитів від клієнтів та взаємодії з базами даних.
5. Безпека веб-застосунків на основі PHP:
 - Врахування основних принципів безпеки при розробці веб-застосунків на PHP.
 - Вивчення методів захисту від потенційних загроз та атак.

6. Практична реалізація проєктів з використанням PHP та PHPMyAdmin:
 - Здобуття практичного досвіду розробки та впровадження веб-застосунків з використанням PHP та PHPMyAdmin.
 - Розвиток навичок роботи з інтегрованими середовищами розробки та інструментами управління базами даних.
7. Вивчення кращих практик та патернів програмування на PHP:
 - Ознайомлення з ефективними та модульними підходами до розробки веб-додатків на мові PHP.
 - Використання рекомендацій та патернів для створення зрозумілого та підтримуваного коду.

Компетентності та результати навчання

У результаті вивчення навчальної дисципліни «Конструювання програмного забезпечення» здобувач освіти отримує наступні програмні компетентності та програмні результати навчання, які визначені в Стандарті фахової передвищої освіти із спеціальності 121 Інженерія програмного забезпечення (<https://mon.gov.ua/storage/app/media/vishcha-osvita/zatverdzeni%20standarty/12/21/121-inzhener.programn.zabezp.bakalavr-1.pdf>) та освітньо-професійній програмі «Інженерія програмного забезпечення» (<https://dev-kpa.fakel.com.ua/storage/uploads/4t0YvRV8MBZ1IXWV9i190ZBGF5H7rglXYy sWLzuH.pdf>) підготовки фахових молодших бакалаврів.

Загальні компетентності:

ЗК05. Знання та розуміння предметної області та розуміння професійної діяльності

ЗК06. Здатність до пошуку, оброблення та аналізу інформації з різних джерел

ЗК07. Здатність застосовувати знання у практичних ситуаціях

Спеціальні (фахові, предметні) компетентності:

СК02. Здатність накопичувати знання в галузі інформаційних технологій та усвідомлювати важливість навчання протягом усього життя.

СК03. Здатність застосовувати теоретичні та емпіричні знання для розроблення, тестування, впровадження та супроводу програмного забезпечення.

СК04. Здатність дотримуватися стандартів при розробці програмного забезпечення.

СК05. Здатність брати участь у визначенні та формулюванні вимог до програмного продукту.

СК06. Здатність брати участь у проектуванні програмного забезпечення.

СК07. Здатність розробляти модулі і компоненти програмного забезпечення за допомогою типових алгоритмів та інструментів.

СК09. Здатність вибирати та використовувати ефективні інструментальні засоби розробки програмного продукту.

СК10. Здатність реалізовувати всі етапи життєвого циклу програмного забезпечення.

СК12. Здатність проєктувати, розробляти та обслуговувати веб-застосунки з динамічним контентом, використовуючи веб-технології, технології комп'ютерної графіки та анімації.

Програмні результати навчання:

РН02. Систематизувати та узагальнювати інформацію про підходи, методи та засоби розробки супроводу програмного забезпечення.

РН03. Застосовувати спеціалізовані емпіричні та теоретичні знання у сфері інженерії програмного забезпечення.

РН05. Розробляти та супроводжувати програмне забезпечення.

РН06. Використовувати основні методології та підходи до організації життєвого циклу програмного забезпечення.

РН08. Аналізувати вимоги до програмного забезпечення.

РН10. Обирати та застосовувати ефективні методи оптимізації алгоритмів.

РН11. Обирати інструментальні засоби, ефективні методи та здійснювати тестування програмних систем.

РН14. Розуміти предметну область, застосовувати знання у професійній діяльності.

РН15. Аналізувати та узагальнювати необхідну інформацію з різних джерел та ресурсів для розв'язання професійних задач з урахуванням сучасних досягнень інформаційних технологій.

РН16. Розуміти загальні принципи та моделі побудови комп'ютерних мереж.

РН17. Розробляти застосунки, використовуючи сучасні веб-технології.

2. Інформаційний обсяг навчальної дисципліни

2.1 Тематичний план

Таблиця 2.1 – Тематичний план навчальної дисципліни

№	Назва змістових модулів і тем
Змістовий модуль 1. Технологій веб-програмування	
Тема 1	Принципи роботи Інтернету.
Тема 2	Динамічний зміст HTML-сторінки
Тема 3	Установлювання сервера
Тема 4	Створення веб-сторінки формату HTML 5
Тема 5	Форматування HTML-сторінки із використанням каскадних таблиць стилів CSS
Тема 6	Мова сценаріїв JavaScript
Змістовий модуль 2 PHP	
Тема 7	Можливості мови PHP
Тема 8	Робота з масивами в PHP
Тема 9	Використання функцій PHP
Тема 10	Використання AJAX (Asynchronous JavaScript and XML).
Тема 11	Використання PHP.
Тема 12	Обробка даних форм у PHP.
Тема 13	Сеанси і сесії в PHP. Файли Cookies.
Тема 14	Робота з файлами в PHP.
Змістовий модуль 3 Взаємодія PHP з базою даних	
Тема 15	Взаємодія PHP з базою даних
Тема 16	Вбудовані функції для роботи з базою даних MySQL.
Тема 17	Реляційні бази даних MySQL
Тема 18	Об'єкти PHP.
Тема 19	Методи оптимізації веб-проектів
Тема 20	Оптимізація статичних html -сторінок.
Тема 21	Показники SEO
Разом: 105 годин	

2.2 Зміст дисципліни

Змістовий модуль 1. Технологій веб-програмування

Тема 1. Принципи роботи Інтернету

Адресація у мережі. Протоколи передачі даних. Архітектура «клієнт-сервер». Протокол HTTP. Передача документів користувача. Огляд web-серверів і технологій програмування серверних додатків.

Тема 2. Динамічний зміст HTML-сторінки

Загальна схема роботи концепції Hyper Text Transfer Protocol, найбільш популярні технології створення динамічних сторінок

Тема 3. Установлювання сервера

Поняття серверу, Переваги сервера Apache.

Тема 4. Створення веб-сторінки формату HTML 5

отримати навички створення сучасних HTML-сторінок відповідно до стандарту HTML5. Теоретичні відомості.

Тема 5. Форматування HTML-сторінки із використанням каскадних таблиць стилів CSS

отримати навички створення каскадних таблиць стилів CSS для HTML-сторінок. Теоретичні відомості.

Тема 6. Мова сценаріїв JavaScript

отримати навички написання простих сценаріїв JavaScript та використання готових JS-бібліотек. Теоретичні відомості

Змістовий модуль 2 PHP

Тема 7. Можливості мови PHP

Можливості мови PHP. Основи синтаксису мови PHP. Основні операції.

Тема 8. Робота з масивами в PHP

Два типи масивів у PHP. Створення масиву.

Створення сценарію обробки масивів.

Тема 9. Використання функцій PHP

Поняття функції. Визначення власних функцій PHP.

Область визначення функції. Функції для роботи з рядками. Регулярні вирази в PHP. Функції для роботи з регулярними виразами

Розгляд прикладу з внутрішніми (вбудованими) функціями

Розробка програм з користувацькими функціями. Передача параметрів функції

Тема 10. Використання AJAX (Asynchronous JavaScript and XML)

отримання студентом інформації щодо основних особливостей методів AJAX, а також отримання навичок щодо створення вебдодатку.

Теоретичні відомості.

Створити програму, що обробляє дані від форми реєстрації

Тема 11. Використання PHP

отримання студентом інформації щодо основних особливостей мови програмування PHP, а також отримання навичок щодо створення простих програм.

Створити клас User.

Тема 12. Обробка даних форм у PHP

Використання HTML-форм для пересилання даних на сервер. Методи пересилання даних сервера.

Процедура розгляду запитів за допомогою PHP.

Розробка сторінки з формою та передача параметрів за допомогою форми.

Тема 13. Сеанси і сесії в PHP. Файли Cookies

Змінна `$_SESSION[]`. Функція `header("Location".)`

Параметри `setcookie`.

Тема 14. Робота з файлами в PHP

Маніпулювання файлами. Читання та запис даних з файла / у файл

Розробка сторінок з передачею даних у файл.

Змістовний модуль 3 *Взаємодія PHP з базою даних*

Тема 15. Взаємодія PHP з базою даних

СУБД MySQL. Взаємодія PHP та MySQL. Виконання запитів MySQL.. Вибірка наборів даних. Реалізація засобів контролю помилок.

Тема 16. Вбудовані функції для роботи з базою даних MySQL

Установка з'єднання з БД ;

Вибір робочої БД; отримання списку полів таблиці; відображення полів в html - форму.

Тема 17. Реляційні бази даних MYSQL

Вивчення основ роботи реляційних баз даних на прикладі СУБД MySQL, основних інструкцій для створення запитів до бази, проектування розподілених Web-застосунків

Розробити клієнтську та серверну частину, а також використовувати серверну СУБД

Розробити клієнтську та серверну частину, а також використовувати серверну СУБД.

Тема 18. Об'єкти PHP

Клас у PHP. Наслідування класів у PHP. Функції самодіагностики.

Тема 19. Методи оптимізації веб-проектів

Методи та способи оптимізації веб додатків.

Тема 20. Оптимізація статичних html -сторінок

Оптимізація текстів, оптимізація html-сторінок.

Тема 21. Показники SEO

Тематичний індекс, Page Rank Google, траст сайтатраст сайту.

2.3 Перелік практичних робіт з дисципліни

Таблиця 2.2 – Тематичний план практичних робіт

№	Тема практичної роботи
1.	Розробка моделі сайту.
2.	Розробка Web-сторінок з використанням мови HTML, CSS, JavaScript.
3.	Знайомство з PHP. Лінійна програма
4.	Умовні конструкції
5.	Програми з повторенням
6.	Робота з масивами в PHP.
7.	Використання функцій PHP
8.	Обробка даних форм у PHP
9.	Робота з файлами в PHP
10.	Робота із СУБД mySQL. SQL запити в PHPMyAdmin.
11.	Вивід даних з БД.
12.	Розробка корпоративного Web-сайту за допомогою мови PHP та бази даних MySQL
	Всього: 30 годин

2.4 Кількість годин разом за програмою дисципліни

Розподіл дисципліни у годинах									
Курс	I		II		III		IV		Всього
Семестр	1	2	3	4	5	6	7	8	
Повний обсяг часу								105	105
Аудиторні заняття, годин								52	52
із них (кількість годин):									
лекції								22	22
практичні								30	30
Самостійна робота								53	53

3. Інструменти, обладнання та програмне забезпечення, використання яких передбачає дисципліна

1. Вільний кроссплатформений HTTP-сервер Apache [сайт]/
<https://httpd.apache.org/download.cgi>
2. Редактор вихідного коду VSCode [сайт]/
<https://code.visualstudio.com/Download>
3. Онлайн-сервіс для розробки та прототипування Figma [сайт]/
<https://www.figma.com>

4. Інформаційні ресурси

Базові (основні):

1. Web-програмування. Лабораторний практикум [Електронний ресурс] : навч. посіб. для студ. спеціальності 125 «Кібербезпека» та 113 «Прикладна математика» / А. Ю. Шелестов, Н. М. Куцуль; КПІ ім. Ігоря Сікорського. – Електронні текстові дані (1 файл: 1047 Кбайт). – Київ : КПІ ім. Ігоря Сікорського, 2021. – 61 с.
2. С.В.Баран. Основи web-програмування: Навчальний посібник. – Кривий Ріг: Державний університет економіки і технологій, 2023. –316 с.
3. К.В. Двірничук, Д.О. Вацек Веб-програмування та веб-дизайн : навч. посіб. Чернівці : Чернівець. нац. ун-т ім. Ю. Федьковича, 2022. 472 с.
4. В. М. Бутенко, Є. П. Павленко, О. В. Головка Інженерія програмного забезпечення. WEB-програмування: навч. посіб. Харків: Український державний університет залізничного транспорту, 2019. 128 с.

Додаткові:

1. W3Schools.UA – електронний PHP підручник, URL: <https://w3schoolsua.github.io/php/index.html#gsc.tab=0>, дата останнього звернення 31.08.2023

2. apserver.org.ua – електронний PHP підручник, URL: <http://apserver.org.ua/peregl.php?d=view&tid=6>, дата останнього звернення 31.08.2023

3. Посібник з PHP – електронний PHP підручник, URL: <https://krypton.com.ua/tutorial/php/>, дата останнього звернення 31.08.2023

5. Форма підсумкового контролю

Екзамен (VIII семестр), поточний контроль.

6. Засоби діагностики результатів навчання

Перевірка та оцінювання знань здобувачів освіти може проводитись кількома методами:

1. Оцінювання знань здобувача освіти під час практичних занять.
2. Виконання індивідуальних навчально-дослідних завдань.
3. захист практичних робіт.
4. Тестування.
5. Проведення поточно-модульного контролю.
6. Проведення екзамену.

6.1 Питання для самоконтролю

1. HTML та CSS:
 - Які теги використовуються для створення заголовків та абзаців в HTML?
 - Як вирівняти текст по центру блока за допомогою CSS?
 - Як використовувати CSS-класи та ідентифікатори для стилізації елементів?
2. JavaScript та клієнтська взаємодія:
 - Як створити змінну в JavaScript?
 - Як викликати функцію при натисканні на кнопку за допомогою JavaScript?
 - Як взаємодіяти з DOM-елементами за допомогою JavaScript?
3. PHP та серверна розробка:
 - Як виглядає базовий синтаксис мови PHP?
 - Як передати дані з форми на сервер за допомогою PHP?
 - Як підключити файл PHP до іншого файлу?
4. Робота з базами даних:
 - Які SQL-запити використовуються для вибору та оновлення даних в таблиці?
 - Як здійснити з'єднання з базою даних та виконати запит на PHP?
 - Як забезпечити безпеку вводу даних в SQL-запитах?
5. Фреймворки та архітектура:

- Які основні переваги використання фреймворків для веб-розробки?
 - Що таке архітектурний патерн MVC, і як він застосовується в веб-розробці?
 - Які фреймворки для веб-розробки ви знаєте, і в чому їхня особливість?
6. Безпека веб-застосунків:
- Як запобігти атакам на основі введення користувача (SQL-ін'єкції, XSS)?
 - Як використовувати HTTPS для шифрування з'єднання з веб-сайтом?
 - Як забезпечити безпеку сесій користувачів у веб-додатках?
7. Адаптивний дизайн та розробка:
- Які основні принципи адаптивного дизайну?
 - Як використовувати медіа-запити для створення адаптивного веб-сайту?
 - Як перевірити адаптивність веб-сайту на різних пристроях?
8. Системи контролю версій та робота в команді:
- Як створити нову гілку в Git та злити її з основною гілкою?
 - Як додати зміни до коміту в Git?
 - Як вирішувати конфлікти при злитті гілок у Git?