

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ  
ХАРКІВСЬКИЙ НАЦІОНАЛЬНИЙ ЕКОНОМІЧНИЙ УНІВЕРСИТЕТ  
ІМЕНІ СЕМЕНА КУЗНЕЦЯ

**ЗАТВЕРДЖЕНО**  
на засіданні кафедри  
кібербезпеки та  
інформаційних технологій  
Протокол № 2 від 31.08.2023 р.

**ПОГОДЖЕНО**  
Проректор з навчально-методичної роботи  
Каріна НЕМАШКАЛО



**МЕТОДОЛОГІЯ РОЗРОБЛЕННЯ  
СУЧАСНИХ ВЕБ-РЕСУРСІВ ТА СЕРВІСІВ**

**робоча програма навчальної дисципліни (РПНД)**

Галузь знань **12 Інформаційні технології**  
Спеціальність **122 Комп'ютерні науки**  
Освітній рівень **третій (освітньо-науковий)**  
Освітня програма **Комп'ютерні науки**

Статус дисципліни  
Мова викладання, навчання та оцінювання

**вибіркова  
українська**

Розробник:  
д.т.н., проф.

підписано КЕП

Володимир АЛЕКСІЄВ

Завідувач кафедри  
кібербезпеки та  
інформаційних технологій

Ольга СТАРКОВА

Гарант програми

Ольга СТАРКОВА

Харків  
2023

## ВСТУП

Актуальність вивчення дисципліни "Методологія розроблення сучасних веб-ресурсів та сервісів" обумовлена неупинною еволюцією Інтернет-технологій, що вимагає від спеціалістів постійного оновлення знань та навичок. Ця дисципліна надає міцну основу, яка дозволяє фахівцям розробляти веб-ресурси, що відповідають сучасним технічним вимогам та потребам користувачів. З огляду на зростання числа інтернет-користувачів та їх залежності від онлайн-сервісів, розуміння принципів створення ефективних веб-сайтів та застосунків стає критично важливим. Глибоке вивчення методологій розроблення веб-ресурсів є ключовим для створення інтерфейсів, які були б інтуїтивно зрозумілі та зручні для користувачів. У цьому аспекті важливим є не лише технічне втілення задуму, але й урахування психологічних та поведінкових факторів взаємодії людини з системою.

Вивчення цієї дисципліни готує до вирішення складних задач, пов'язаних з безпекою веб-сервісів, забезпеченням їхньої масштабованості та високої доступності. Програма вивчення навчальної дисципліни "Методологія розроблення сучасних веб-ресурсів та сервісів" складена відповідно до освітньо-наукової програми підготовки докторів філософії зі спеціальності 122 "Комп'ютерні науки".

Метою викладання дисципліни "Методологія розроблення сучасних веб-ресурсів та сервісів" є формування та розвиток здатності до застосування сучасних методів та підходів щодо створення ефективних корпоративних інформаційних систем на базі застосування технологій веб-ресурсів та сервісів.

Предметом дисципліни є інструментальні засоби та основи їх застосування у галузі розроблення сучасних веб-ресурсів та сервісів.

Об'єктом – процеси неперервної інтеграції та доставки веб-ресурсів та сервісів користувачам.

Основними завданнями вивчення дисципліни "Методології розроблення сучасних веб-ресурсів та сервісів" є формування цілісного уявлення щодо застосування певних засобів та технологічних рішень щодо забезпечення життєвого циклу сучасних веб-ресурсів та сервісів; надання аспірантам знання щодо оптимального вибору програмних рішень та методик забезпечення виконання етапів проектування, розроблення й супроводу веб-ресурсів та сервісів із застосуванням сучасних інформаційно-комунікаційних технологій; ознайомлення з вимогами, що пред'являються до систем та програмних засобів управління проектом щодо впровадження веб-ресурсів та сервісів; передати аспірантам комплекс знань та вмінь, які допоможуть їм здійснювати діяльність та пропонувати рішення щодо створення корпоративних інформаційних систем та рішень у галузі проведення наукових досліджень на базі застосування технологій веб-ресурсів та сервісів.

Результати навчання та компетентності, які формує навчальна дисципліна визначено в табл. 1.

**Результати навчання та компетентності, які формує навчальна дисципліна**

<b>Результати навчання</b>	<b>Компетентності, якими повинен оволодіти здобувач вищої освіти</b>
РН 04	ЗК 01, СК 01, СК 02, СК 04
РН 06	ЗК 02, СК 01, СК 03, СК 06
РН 07	ЗК 02, СК 01, СК 03

де,

РН04. Розробляти та досліджувати концептуальні, математичні і комп'ютерні моделі процесів і систем, ефективно використовувати їх для отримання нових знань та/або створення інноваційних продуктів у комп'ютерних науках та дотичних міждисциплінарних напрямках.

РН06. Застосовувати сучасні інструменти і технології пошуку, оброблення та аналізу інформації, зокрема, статистичні методи аналізу даних великого обсягу та/або складної структури, спеціалізовані бази даних та інформаційні системи.

РН07. Розробляти та реалізовувати наукові та/або інноваційні інженерні проекти, які дають можливість переосмислити наявне та створити нове цілісне знання та/або професійну практику і розв'язувати значущі наукові та технологічні проблеми комп'ютерної науки з дотриманням норм академічної етики і врахуванням соціальних, економічних, екологічних та правових аспектів.

ЗК 01 Здатність до абстрактного мислення, аналізу та синтезу.

ЗК02. Здатність до пошуку, оброблення та аналізу інформації з різних джерел.

СК01. Здатність виконувати оригінальні дослідження, досягати наукових результатів, які створюють нові знання у комп'ютерних науках та дотичних до них міждисциплінарних напрямках і можуть бути опубліковані у провідних наукових виданнях з комп'ютерних наук та суміжних галузей.

СК02. Здатність застосовувати сучасні методології, методи та інструменти експериментальних і теоретичних досліджень у сфері комп'ютерних наук, сучасні цифрові технології, бази даних та інші електронні ресурси у науковій та освітній діяльності.

СК03. Здатність виявляти, ставити та вирішувати дослідницькі науковоприкладні задачі та/або проблеми в сфері комп'ютерних наук, оцінювати та забезпечувати якість виконуваних досліджень.

СК04. Здатність ініціювати, розробляти і реалізовувати комплексні інноваційні проекти у галузі комп'ютерних наук та дотичні до неї міждисциплінарних проектах, демонструвати лідерство під час їх реалізації.

СК06. Здатність аналізувати та оцінювати сучасний стан і тенденції розвитку комп'ютерних наук та інформаційних технологій.

# ПРОГРАМА НАВЧАЛЬНОЇ ДИСЦИПЛІНИ

## Зміст навчальної дисципліни

**Змістовий модуль 1. Теоретичні основи та програмні технології побудови веб-ресурсів та сервісів.**

**Тема 1. Основні терміни, визначення та класифікація сучасних веб-орієнтованих проєктів та систем**

### **1.1 Вступ до дисципліни**

Загальний огляд веб-технологій та їх розвиток з часом. Історія та еволюція інтернету як платформи для веб-орієнтованих проєктів та систем.

### **1.2 Ключові терміни**

Визначення ключових термінів, таких як веб-сайт, веб-додаток, веб-сервіс, API, CMS (системи управління контентом), SPA (односторінкові додатки).

### **1.3 Класифікація веб-проєктів**

Розподіл сучасних типів веб-проєктів залежно від їхньої складності, масштабу, призначення та аудиторії. Включення категорій, таких як корпоративні сайти, електронна комерція, освітні платформи, соціальні мережі, блоги тощо.

**Тема 2. Організація локальної мережі рівня підприємства.**

### **2.1 Основні концепції та топології локальних мереж.**

Фундаментальні принципи локальних мереж (LAN). Визначення LAN та їх класифікації. Топології мереж: зірка, кільце, шина та меш, їх переваги та недоліки в контексті використання на підприємстві.

### **2.2 Мережеве Обладнання та Протоколи**

Мережеве обладнання, яке використовується у локальних мережах підприємств, включаючи комутатори, маршрутизатори, містки, точки доступу Wi-Fi тощо. Протоколи, які забезпечують взаємодію мережевих компонентів і передачу даних, такі як Ethernet, TCP/IP, DHCP, DNS та інші.

### **2.3 Планування та розгортання мережі**

Опис процесу планування локальної мережі, вибір адекватної топології, розрахунок необхідного обладнання залежно від масштабу підприємства та вимог до функціональності. Критерії вибору мережевого обладнання та викладення основних кроків з розгортання мережевої інфраструктури.

### **2.4 Безпека та адміністрування мережі**

Значення та методи забезпечення безпеки в локальних мережах, включаючи встановлення брандмауерів, використання VLAN, реалізація політик безпеки та доступу.

**Тема 3. Застосування ресурсів хмарних обчислень для ефективного виконання науково-дослідних та науково-технічних проєктів.**

### **3.1. Вступ до хмарних обчислень**

Огляд основних понять та визначень хмарних обчислень, історії їх розвитку та основних характеристик (IaaS, PaaS, SaaS). Переваги та можливості, які хмарні обчислення пропонують науковій спільноті.

### **3.2 Моделі розгортання хмарних ресурсів**

Огляд моделей хмарних ресурсів: приватна, публічна, гібридна та спільнотна хмара, та їх придатності для наукових проєктів. Приклади використання кожної моделі в академічних дослідженнях та розробці.

### **3.3 Платформи та Інструменти для Наукових Досліджень**

Аналіз популярних хмарних платформ: AWS, Google Cloud, Azure, які пропонують спеціалізовані сервіси для обробки даних, машинного навчання, високопродуктивних обчислень (HPC). Їх застосування в наукових проєктах.

**Тема 4. Архітектура програмних рішень сучасних веб-ресурсів та сервісів.**

#### **4.1 Основні засади веб-архітектури**

Базові компоненти та концепції, які формують архітектуру веб-ресурсів і сервісів. Многорівнева архітектура: клієнтський бік (front-end), серверний бік (back-end), бази даних та міжсерверне зв'язування.

#### **4.2 Клієнтський бік (Front-end).**

Огляд технологій, які використовуються на клієнтському боці: HTML, CSS, JavaScript. Сучасні фреймворки та бібліотеки: React, Angular, Vue.js. Дизайн UX/UI для створення інтуїтивно зрозумілих та ефективних інтерфейсів.

#### **4.3 Серверний бік (Back-end)**

Опис архітектури серверного боку: серверні мови програмування (Python, Java, Node.js), фреймворки (Django, Spring, Express.js). Вибір та робота з серверним ПЗ.

#### **4.4 Бази Даних та Зберігання Даних**

Системи управління базами даних (SQL та NoSQL), їхня ролі у веб-архітектурі, Стратегії забезпечення безпеки, реплікації, шардування та високої доступності.

#### **4.5 Мікросервіси та Контейнеризація**

Переваги та недоліки мікросервісної архітектури. Використання контейнеру Docker і оркестрації контейнерів Kubernetes для побудови модульних веб-сервісів.

**Тема 5. Приклад та теоретичні основи реалізації складної інформаційної системи на базі веб-ресурсів та сервісів**

#### **5.1 Розробка документації RESTful API**

Основні Концепції RESTful API. Визначення REST (Representational State Transfer). Основні принципи архітектури REST та її переваги для веб-розробки.

Огляд структури документації, включаючи опис ендпоінтів, HTTP-методів (GET, POST, PUT, DELETE), параметрів запитів, форматів відповідей, статус-кодів та прикладів використання.

**5.2 Розробка за специфікацією OpenAPI 3.0 документації на REST запити до серверу**

Призначення специфікації OpenAPI, її роль у стандартизації та документуванні RESTful API. Опис компонентів специфікації: об'єкти `paths`, `servers`, `components`, `security`, та інші. Розгляд прикладів визначення шляхів, параметрів, тіла запитів (request bodies), відповідей (responses) та прикладів

використання. Документування кінцевих точок API, включаючи GET, POST, PUT, DELETE запити. Методи автентифікації та авторизації, включаючи API ключі, OAuth, JWT та інші схеми безпеки, які підтримуються в OpenAPI 3.0.

**Змістовий модуль 2. Розгортання та супроводження рішень на основі веб-ресурсів та сервісів.**

**Тема 6. Забезпечення життєвого циклу інформаційної системи на базі сучасних веб-ресурсів та сервісів**

**6.1 Інструментальні засоби та методології управління виконанням проекту**

Огляд методологій управління проектами: Agile, Scrum, Kanban, Lean, Waterfall, PRINCE2. Сучасні інструменти та ПЗ, які використовуються для планування, відстеження та управління проектами. Методи моніторингу прогресу проекту, включаючи встановлення KPI (ключові показники ефективності), використання дашбордів для відстеження ходу робіт, та методи звітності про статус проекту.

**6.2 Побудова RESTfull-сервісів**

Основні Принципи RESTful Архітектури. Концепція REST (Representational State Transfer), опис основних принципів, таких як безстановість, однорідність інтерфейсу, і системна роздільність. Технічні аспекти реалізації API, включаючи вибір технологій та фреймворків (Node.js з Express, Spring Boot для Java, Flask для Python). Документація та версіонування API. Методи моніторингу API для відстеження працездатності, швидкодії та виявлення можливих проблем.

**Тема 7. Визначення архітектурних рішень програмної реалізації сучасних веб-ресурсів та сервісів**

**7.1 Розробка безсерверного веб- застосунку**

Концепція безсерверної архітектури (Serverless). Безсерверна архітектура: визначення та основні переваги. Огляд сучасних платформ безсерверної розробки, таких як AWS Lambda, Azure Functions, Google Cloud Functions. Розробка та розгортання функцій, включаючи локальну розробку, тестування, розгортання та моніторинг.

**7.2 Інфраструктура як код**

Визначення та основні принципи IaC. Переваги використання IaC. Огляд інструментів для реалізації IaC, таких як Terraform, AWS CloudFormation, Azure Resource Manager, Google Cloud Deployment Manager, Ansible, Chef, Puppet. Безпека та дотримання норм у IaC.

**Тема 8. Особливості проектування та розроблення складних веб-рішень, що масштабуються.**

**8.1. Розробка у хмарному середовищі AWS веб-рішення**

Аналіз Amazon Web Services (AWS), огляд ключових компонентів і сервісів, які використовуються для розробки веб-рішень. Переваги використання AWS.

**8.2 Архітектура та моделювання веб-рішень на AWS**

Архітектурні підходи створення веб-рішень у хмарі AWS. Використання мікросервісів, безсерверної архітектури, контейнеризації (Docker, Kubernetes на AWS). Інтеграції з іншими сервісами та API. Безпека під час розробки застосунків у хмарному середовищі AWS. Автоматизація та CI/CD у AWS. Використання інструментів та сервісів AWS для автоматизації процесів розробки, тестування та розгортання.

## **Тема 9. Етапи розроблення, компоненти та технології неперервної інтеграції та розгортання складних веб-рішень**

### **9.1 Технології Serverless**

Основні принципи та переваги безсерверних (Serverless) технологій. Структура та основні компоненти безсерверних архітектур: безсерверні функції (AWS Lambda, Azure Functions), сервіси управління даними (Amazon DynamoDB, Google Cloud Firestore) та інтеграційні сервіси (наприклад, AWS API Gateway). Огляд процесів проектування та розробки безсерверних додатків. Управління ресурсами та моніторинг у Serverless архітектурі

### **9.2 Технології DevOps, моніторинг, аналіз журналу звітів про роботу сервісів**

Огляд ключових інструментів та практик, які використовуються в DevOps для автоматизації процесів розробки, тестування, розгортання та моніторингу. Інструмент неперервної інтеграції та розгортання Jenkins. Інструменти автоматизації налаштувань Ansible, Puppet та Chef. Методи аналізу даних з журналів для швидкого виявлення та вирішення інцидентів.

## **Тема 10. Перспективи розвитку**

### **10.1 Перспективи розвитку сучасних інформаційних систем серверних рішень на базі веб-ресурсів та сервісів**

Вплив поширення хмарних обчислень на архітектуру та розгортання сучасних інформаційних систем. Перспективи розвитку безсерверних архітектур та їхній вплив на розробку та експлуатацію веб-орієнтованих інформаційних систем. Мікросервісна архітектура. Аналіз потенціалу інтеграції інформаційних систем з технологіями штучного інтелекту, машинного навчання та великих даних (Big Data).

### **10.2 Застосування технології децентралізованих систем цифрової ідентифікації**

Основні технології та протоколи, які лежать в основі децентралізованих систем ідентифікації, наприклад, блокчейн, Distributed Ledger Technology (DLT), Smart Contracts, та Decentralized Identifiers (DIDs). Сценарії застосування децентралізованих систем ідентифікації у різних галузях.

Перелік практичних занять за навчальною дисципліною наведено в табл. 2

**Перелік практичних занять**

Назва теми та / або завдання	Зміст
Тема 1,2. Завдання 1.	Дослідження особливостей застосування технологій віртуалізації для формування ефективного простору розробки веб-орієнтованого програмного продукту.
Тема 3,4. Завдання 2.	Особливості та методологія застосування сучасних ресурсів хмарних обчислень для побудови ефективних рішень щодо створення веб-ресурсів та сервісів.
Тема 5,6. Завдання 3.	Визначення елементів методології та дослідження особливостей розроблення сучасних веб-ресурсів та сервісів на базі відкритих систем.
Тема 7 - 10. Завдання 4.	Створення технічного завдання на розробку веб-рішення.

Перелік самостійної роботи за навчальною дисципліною наведено в табл. 3.

**Перелік самостійної роботи**

Назва теми та / або завдання	Зміст
Тема 1. Завдання 1	Застосування інформаційно-комунікаційних систем. Застосування хмарних технологій у дослідженнях
Тема 2. Завдання 2.	Мови програмування. Розгортання веб-сервер на основі Linux та системи WordPress. Написання скрипта простої веб-сторінки, яка отримує від користувача електронну адресу та зберігає її до бази даних
Тема 3. Завдання 3.	Вивчення технології гнучкого управління проектами Agile
Тема 4. Завдання 4.	Вивчення структури сучасних ІТ-професій
Тема 5. Завдання 5.	Поглиблене вивчення методології розробки ПЗ клієнтської та серверної частин
Тема 6. Завдання 6.	Вивчення методології розробки ПЗ фул-стек, моноліт, мікросервіс, безсерверні технології
Тема 7. Завдання 7.	Розробка документації RESTfull API
Тема 8. Завдання 8.	Поглиблене вивчення технології Serverless
Тема 9,10. Завдання 9,10.	Вивчення технології DevOps, моніторинг, аналіз журналу звітів про роботу сервісів та ін.



Кількість годин лекційних та практичних занять та годин самостійної роботи наведено в робочому плані (технологічній карті) з навчальної дисципліни.

## МЕТОДИ НАВЧАННЯ

У процесі викладання навчальної дисципліни для набуття визначених результатів навчання, активізації освітнього процесу передбачено застосування таких методів навчання, як:

Словесні (лекції 1-10), проблемна лекція (Тема 1).

Наочні (демонстрація (Тема 1-10)).

Практичні (практичні роботи (Теми 1-10)).

## ФОРМИ ТА МЕТОДИ ОЦІНЮВАННЯ

Університет використовує 100 бальну накопичувальну систему оцінювання результатів навчання здобувачів вищої освіти.

**Поточний контроль** здійснюється під час проведення лекційних, практичних занять і має на меті перевірку рівня підготовленості здобувача вищої освіти до виконання конкретної роботи і оцінюється сумою набраних балів: для дисциплін з формою семестрового контролю залік: максимальна сума – 100 балів; мінімальна сума – 60 балів.

**Підсумковий контроль** включає семестровий контроль та атестацію здобувача вищої освіти.

**Семестровий контроль** проводиться у формах диференційованого заліку або заліку.

**Підсумкова оцінка за навчальною дисципліною** визначається для дисциплін з формою семестрового контролю залік – сумуванням всіх балів, отриманих під час поточного контролю.

Під час викладання навчальної дисципліни використовуються наступні контрольні заходи:

Поточний контроль: індивідуальні завдання під час захисту практичних робіт 60 балів (4 роботи по 15 балів кожна), письмові контрольні роботи 40 балів (2 роботи по 20 балів кожна).

Семестровий контроль: Залік.

Більш детальну інформацію щодо системи оцінювання наведено в робочому плані (технологічній карті) з навчальної дисципліни.

## РЕКОМЕНДОВАНА ЛІТЕРАТУРА

### Основна

1. Кібербезпека : сучасні технології захисту. Навчальний посібник для студентів вищих навчальних закладів. / С. Е. Остапов, С. П. Євсєєв, О.Г. Король. – Львів: «Новий Світ- 2000», 2020 . – 678 с.
2. Mikael Krief. Learning DevOps. – Packt Publishing, 2019. – 489 p
3. David Clinton, Christopher Negus. Ubuntu Linux Bible/John Wiley & Sons, 2020– 752 p.
4. Maarten van SteenAndrew S. Tanenbaum. Distributed Systems. Third edition., Maarten van Steen, 2018. – p. [Electronic resource]. – Access mode: <https://www.distributed-systems.net/index.php/contact/>
5. WordPress Security Fundamentals [Electronic resource] / Wordfence. Defiant. – Access mode : <https://www.wordfence.com/learn/>.
6. Jamon Camisso. Sysadmin eBook: Making Servers Work – DigitalOcean, 2020. – 281 p.
7. Lisa Tagliaferri. DigitalOcean eBook: How To Code in Python – DigitalOcean, 2018. – 459 p.
8. Молчанов В. П. Технології розробки WEB-ресурсів [Електронний ресурс] : навч. посіб. / В. П. Молчанов, О. К. Пандорін ; Харківський національний економічний університет ім. С. Кузнеця. - Електрон. текстові дан. (7,94 МБ). - Харків : ХНЕУ ім. С. Кузнеця, 2019. - 129 с. : іл. - Загол. з титул. екрану. - Бібліогр.: с. 126. <http://repository.hneu.edu.ua/handle/123456789/22466>
9. Milov O. Self-organizing organizational structures of cybersecurity systems / O. Milov, V Aleksiyev. // Modern Problems Of Computer Science And IT-Education : collective monograph / [editorial board К. Melnyk, O. Shmatko].– Vienna : Premier Publishing s.r.o., 2020. – P. 65–78. <http://repository.hneu.edu.ua/handle/123456789/24816>

### Додаткова

10. Погребняк Б. І. Операційні системи : навч. посібник / Б. І. Погребняк, М. В. Булаєнко ; Харків. нац. ун-т міськ. госп-ва ім. О. М. Бекетова. – Харків : ХНУМГ ім. О. М. Бекетова, 2018. – 104 с.
11. Жураковський, Б. Ю. Комп'ютерні мережі. Частина 1. Навчальний посібник [Електронний ресурс]: навч. посіб. / Б. Ю. Жураковський, І.О. Зенів; КПІ ім. Ігоря Сікорського. – Електронні текстові дані (1 файл: 8,6 Мбайт). – Київ : КПІ ім. Ігоря Сікорського, 2020. – 336 с. Режим доступу : <https://ela.kpi.ua/handle/123456789/36615>
12. Жураковський, Б. Ю. Комп'ютерні мережі. Частина 2. Навчальний посібник [Електронний ресурс] : навчальний посібник / Б. Ю. Жураковський, І. О. Зенів ; КПІ ім. Ігоря Сікорського. – Електронні текстові дані (1 файл: 4,73 Мбайт). – Київ : КПІ ім. Ігоря Сікорського, 2020. – 372 с. – Режим доступу : <https://ela.kpi.ua/handle/123456789/36641>

13. Proxmox VE Admin Guide for 8.x, 2023. – 462 p. [Electronic resource]. – Access mode: <https://www.proxmox.com/de/downloads/proxmox-virtual-environment/documentation/proxmox-ve-admin-guide-for-8-x>

14. Y. Hrabovskyi Automation of forming complex advertising products / Y. Hrabovskyi // Advanced Trends in ICT for Innovative Business Management. Monograph / Edited By Katarzyna Szymczyk, Ibrahiem M. M. El Emary. - London, Taylor & Francis, 2021. - P. 69-90.

<http://repository.hneu.edu.ua/handle/123456789/29194>

15. Shmatko O. Information support for distributed teamwork knowledge management / O. Shmatko, M. Bilova. // Modern Problems Of Computer Science And IT-Education : collective monograph / [editorial board K. Melnyk, O. Shmatko].– Vienna: Premier Publishing s.r.o., 2020.– P. 169–192.

<http://repository.hneu.edu.ua/handle/123456789/24818>

### **Інформаційні ресурси.**

16. How To Install Linux, Apache, MySQL, PHP (LAMP) stack on Ubuntu 20.04 [Electronic resource] / Erika Heidi. DigitalOcean, 2021. – Access mode : <https://www.digitalocean.com/community/tutorials/how-to-install-linux-apache-mysql-php-lamp-stack-on-ubuntu-20-04>.

17. Initial Server Setup with Ubuntu 20.04 [Electronic resource] / Brian Boucheron. DigitalOcean, 2021. – Access mode : <https://www.digitalocean.com/community/tutorials/initial-server-setup-with-ubuntu-20-04>.

18. Сайт персональних навчальних систем ХНЕУ ім. С. Кузнеця за дисципліною "Методологія розроблення сучасних веб-ресурсів та сервісів" <https://pns.hneu.edu.ua/course/view.php?id=5262>.