

**МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ**  
**Львівський національний університет імені Івана Франка**  
**Факультет електроніки та комп'ютерних технологій**  
**Кафедра системного проектування**

**Затверджено**

На засіданні  
кафедри системного проектування  
факультету електроніки та комп'ютерних  
технологій  
Львівського національного університету  
імені Івана Франка  
(протокол № 1 від 28.08.2023 р.)

Завідувач кафедри:



Роман ШУВАР

**Силабус з навчальної дисципліни**  
**“Веб-технології та програмування (ч.2)”**,  
**що викладається в межах ОПП**  
**“ Високопродуктивний комп'ютинг ”**  
**першого (бакалаврського) рівня вищої освіти для здобувачів з**  
**спеціальності 121 – Інженерія програмного забезпечення**

Львів 2023 р.

<b>Назва дисципліни</b>	Веб-технології та програмування (ч.2)
<b>Адреса викладання дисципліни</b>	м. Львів, вул. Драгоманова, 50
<b>Факультет та кафедра, за якою закріплена дисципліна</b>	Факультет електроніки та комп'ютерних технологій, кафедра системного проектування
<b>Галузь знань, шифр та назва спеціальності</b>	12 Інформаційні технології 121 Інженерія програмного забезпечення (ВПК)
<b>Викладачі дисципліни</b>	Демків Лідія Степанівна, доцент Гера Орест Богданович, доцент
<b>Контактна інформація</b>	<a href="mailto:liדיya.demkiv@lnu.edu.ua">liדיya.demkiv@lnu.edu.ua</a> <a href="https://electronics.lnu.edu.ua/employee/demkiv-l-s/">https://electronics.lnu.edu.ua/employee/demkiv-l-s/</a> <a href="mailto:orest.hera@lnu.edu.ua">orest.hera@lnu.edu.ua</a> <a href="https://electronics.lnu.edu.ua/employee/hera-o-b">https://electronics.lnu.edu.ua/employee/hera-o-b</a>
<b>Консультації з питань навчання по дисципліні відбуваються</b>	Консультації в день проведення лекційних занять (за попередньою домовленістю). Також можливі он-лайн консультації через MS Teams або систему електронного навчання Moodle. Для погодження часу онлайн консультацій слід писати на електронну пошту викладача.
<b>Сторінка дисципліни</b>	<a href="https://moodle.elct.lnu.edu.ua/course/view.php?id=188">https://moodle.elct.lnu.edu.ua/course/view.php?id=188</a>
<b>Інформація про дисципліну</b>	Дисципліна “Веб-технології та програмування (ч.2)” є нормативною дисципліною з спеціальності 121 Програмна інженерія для освітньої програми «Високопродуктивний комп'ютинг», яка викладається в 4 семестрі в обсязі 4 кредитів (за Європейською Кредитно-Трансферною Системою ECTS).
<b>Коротка анотація дисципліни</b>	Основна увага в цьому курсі зосереджена на Всесвітній павутині як платформі для інтерактивних програм, публікації контенту та соціальних послуг. Розробка веб-додатків вимагає знань про основну технологію, формати та стандарти, на яких базується Інтернет. Дисципліна розкриває питання протоколів зв'язку HTTP, інтерактивну графіку та мультимедійний вміст в Інтернеті, основні поняття фреймворків Javascript, таких як: Node.js та React.
<b>Мета та цілі дисципліни</b>	<b>Метою</b> навчальної дисципліни є засвоєння необхідних знань з інформаційних технологій та основ веб-технологій, а також формування твердих практичних навичок щодо розробки сучасних веб-застосунків. <b>Предметом</b> вивчення дисципліни є інформаційні та веб-технології, а також методи їх використання при розробці веб-застосунків різноманітного призначення. <b>Цілями</b> вивчення навчальної дисципліни є засвоєння студентами

	сучасних web-технологій і суміжних галузей знань, вивчення та практичне засвоєння методів і засобів створення веб-сайтів та веб-застосунків за допомогою сучасних засобів JavaScript.
<b>Література для вивчення дисципліни</b>	<p>Основна література:</p> <ol style="list-style-type: none"> <li>1. Jackson, Jeffrey C. Web technologies : a computer science perspective. - Pearson Education: New Jersey. - 2007. - 591 p. - Режим доступу: <a href="http://seu1.org/files/level6/IT230/Book/(web.tech%201st%20book)%20Web%20Technologies%20-%20A%20Computer%20Science%20Perspective.pdf">http://seu1.org/files/level6/IT230/Book/(web.tech%201st%20book)%20Web%20Technologies%20-%20A%20Computer%20Science%20Perspective.pdf</a></li> <li>2. Advance Web Technology: Vardhman Mahaveer Open University, Kota, 2018. - 459 p. - Режим доступу: <a href="http://assets.vmou.ac.in/MCA304.pdf">http://assets.vmou.ac.in/MCA304.pdf</a> .</li> <li>3. Alex Jouog. Node.js in Action. / Bradley Meck, Mike Canteron // Manning Publication Co. - 2017. - 420 p. - Режим доступу: <a href="https://github.com/AbdullahMoawad/Books/raw/master/Node.js%20in%20Action%2C%202nd%20Edition.pdf">https://github.com/AbdullahMoawad/Books/raw/master/Node.js%20in%20Action%2C%202nd%20Edition.pdf</a></li> <li>4. Fullstack React The Complete Guide to ReactJS and Friends / Anthony Accomazzo, Nate Murray, Ari Lerner, Clay Allsopp, David Gutman, and Tyler McGinnis // Fullstack.io. - 2020. - 1028 p. - Режим доступу: <a href="https://edu.anarcho-copy.org/Programming%20Languages/web/fullstackreact-book-r40.pdf">https://edu.anarcho-copy.org/Programming%20Languages/web/fullstackreact-book-r40.pdf</a></li> </ol> <p>Інформаційні ресурси:</p> <ol style="list-style-type: none"> <li>5. Сучасний підручник з JavaScript, режим доступу: <a href="https://uk.javascript.info/">https://uk.javascript.info/</a></li> <li>6. <a href="https://pages.cs.wisc.edu/~remzi/OSTEP/threads-events.pdf">https://pages.cs.wisc.edu/~remzi/OSTEP/threads-events.pdf</a></li> </ol>
<b>Обсяг курсу</b>	Кількість кредитів ЄКТС: 4 (120 год), з них: 64 годин аудиторних занять (лекції: 32 год, лабораторні: 32 год.) та 56 год. самостійної роботи.
<b>Очікувані результати навчання</b>	<p>Після вивчення даного курсу здобувачі набудуть таких Загальних та Фахових компетентностей та Програмних результатів навчання:</p> <p>ЗК01. Здатність до абстрактного мислення, аналізу та синтезу.</p> <p>ЗК02. Здатність застосовувати знання у практичних ситуаціях.</p> <p>ЗК04. Здатність спілкуватися іноземною мовою як усно, так і письмово.</p> <p>ЗК05. Здатність вчитися і оволодівати сучасними знаннями.</p> <p>ЗК06. Здатність до пошуку, оброблення та аналізу інформації з різних джерел.</p> <p>ФК17. Здатність дотримуватися специфікацій, стандартів, правил і рекомендацій в професійній галузі при реалізації процесів життєвого циклу.</p> <p>ФК18. Здатність аналізувати, вибирати і застосовувати методи і засоби для забезпечення інформаційної безпеки (в тому числі кібербезпеки).</p> <p>ФК22. Здатність накопичувати, обробляти та систематизувати професійні знання щодо створення і супроводження програмного забезпечення та визнання важливості навчання протягом всього життя.</p> <p>ФК25. Здатність обґрунтовано обирати та освоювати інструментарій з розробки та супроводження програмного забезпечення.</p> <p>ФК28. Володіння методами розроблення і впровадження систем підвищеної продуктивності, серверних, мікросервісних, хмаркових, розподілених та інших новітніх технологій.</p> <p>ПРН07. Знати і застосовувати на практиці фундаментальні концепції, парадигми і основні принципи функціонування мовних, інструментальних і обчислювальних засобів інженерії програмного</p>

	<p>забезпечення.</p> <p>ПРН08. Вміти розробляти людино-машинний інтерфейс.</p> <p>ПРН14. Застосовувати на практиці інструментальні програмні засоби доменного аналізу, проектування, тестування, візуалізації, вимірювань та документування програмного забезпечення.</p> <p>ПРН17. Вміти застосовувати методи компонентної розробки програмного забезпечення.</p>
<b>Ключові слова</b>	Веб, Інтернет, Веб-технології, Веб-програмування, Web, Search Engine, Semantic Web, HTML, HTTP, CSS, JavaScript, Node.
<b>Формат курсу</b>	Очний. Проведення лекцій, лабораторних робіт та консультації для кращого розуміння тем.
<b>Теми</b>	Див. СХЕМА КУРСУ
<b>Підсумковий контроль, форма</b>	Екзамен в кінці семестру
<b>Сертифікація</b>	<p>Сертифікація не є обов'язковим елементом дисципліни, а тільки дозволяє оцінити свої можливості для працевлаштування:</p> <ul style="list-style-type: none"> <li>- <a href="#">CSS Skills</a></li> <li>- <a href="#">JavaScript (Basic) Skills</a></li> <li>- <a href="#">JavaScript (Intermediate) Skills</a></li> <li>- <a href="#">Node (Basic) Skills</a></li> <li>- <a href="#">Node.js (Intermediate) Skills</a></li> <li>- <a href="#">React (Basic) Skills</a></li> <li>- <a href="#">Rest API (Intermediate) Skills</a></li> </ul>
<b>Пререквізити</b>	Для вивчення курсу студенти не потребують жодних додаткових знань крім тих, що отримали протягом навчання в середній школі.
<b>Навчальні методи та техніки, які будуть використовуватися під час викладання курсу</b>	Презентація, лекції, лабораторні роботи, обговорення, дискусія.
<b>Необхідне обладнання</b>	<p><b>Для проведення лекційних занять:</b></p> <p>Комп'ютер (мінімальні характеристики: процесор Intel Core i3, 8ГБ оперативної пам'яті, 50ГБ вільного місця на диску, відеокарта Nvidia GEFORCE GT1030 2048MB), доступ до мережі Internet, засоби мультимедіа (в т.ч. проектор).</p> <p><b>Для проведення лабораторних занять:</b></p> <p>Комп'ютер (мінімальні характеристики: процесор Intel Core i3, 8ГБ оперативної пам'яті, 50ГБ вільного місця на диску), необхідне програмне забезпечення включає в себе ОС Ubuntu 20.04 LTS, програмне забезпечення для побудови діаграм draw.io, PlantUML, пакет Docker.IO, докер контейнери з базами даних Microsoft SQL Server та MySQL, утиліти для взаємодії з СКБД sqlcmd, DBeaver Community, контейнери з Nginx та NodeJS.</p>
<b>Критерії оцінювання</b>	Оцінювання проводиться впродовж семестру за 100-бальною

**(окремо для кожного виду навчальної діяльності)**

шкалою, де враховано бали за два контрольні заміри по 25 балів за кожний модуль та 50 балів за складання іспиту.

Бали нараховуються за такими видами робіт з наступним співвідношенням:

- контрольні заміри (2 модулі): 50% семестрової оцінки; максимальна кількість балів: 50, а саме:
  - лабораторні роботи: 60% оцінки контрольного заміру; максимальна кількість балів: 32 (14 лабораторних робіт).
  - теоретичний матеріал: 40% оцінки контрольного заміру; максимальна кількість балів: 18 (2 тести по 9 балів кожний).
- екзамен: 50% семестрової оцінки; максимум 50 балів.

Оцінки за лабораторні заняття розподіляються наступним чином: викладач оцінює на скільки балів від максимальної оцінки заслуговує студент за виконання і відповідями на запитання.

**Академічна доброчесність:** Очікується, що лабораторні роботи та контрольні роботи студентів будуть їх оригінальними дослідженнями чи міркуваннями. Відсутність посилань на використані джерела, фабрикування джерел, списування, втручання в роботу інших студентів становлять, але не обмежують, приклади можливої академічної недоброчесності. Виявлення ознак академічної недоброчесності в роботі студента є підставою для її незарахування викладачем.

**Відвідування занять** є важливою складовою навчання. Очікується, що всі студенти відвідають усі лекції і лабораторні заняття курсу. Студенти мають інформувати викладача про неможливість відвідати заняття. Студенти зобов'язані дотримуватися усіх термінів визначених для виконання усіх видів робіт, передбачених курсом.

**Література.** Уся література, яку студенти не зможуть знайти самостійно, буде надана викладачем виключно в освітніх цілях без права її передачі третім особам. Студенти заохочуються до використання також й іншої літератури та джерел, яких немає серед рекомендованих.

**Політика виставлення балів.** Враховуються бали набрані на поточному тестуванні, самостійній роботі та бали підсумкового тестування. При цьому обов'язково враховуються присутність на заняттях та активність студента під час лабораторного заняття; недопустимість пропусків та запізнь на заняття; користування мобільним телефоном, планшетом чи іншими мобільними пристроями під час заняття в цілях не пов'язаних з навчанням; списування та плагіат; несвоєчасне виконання поставленого завдання і т. ін. Жодні форми порушення академічної доброчесності не толеруються.

#### **Критерії оцінювання результатів неформальної освіти:**

Нарахування балів відбувається за написання студентом тез доповідей на конференціях, наукових статей, участь у діяльності наукових гуртків, участь у наукових семінарах та круглих столах, конкурсах, участь у заходах неформальної освіти за отримання сертифікатів про проходження навчання на різних освітніх платформах (Coursera, Udey, Prometheus тощо), курсах на провідних ІТ компаніях за тематикою навчальної дисципліни. Кількість балів визначається відсотком покриття результатів

	відповідної активності до вимог результатів навчання з навчальної дисципліни.
<b>Питання до контрольних робіт</b>	Перелік питань та завдань для проведення підсумкової оцінки знань певних тем до контрольних робіт розміщені на веб-сторінці курсу (система електронного навчання Moodle)
<b>Опитування</b>	Анкету-оцінку з метою оцінювання якості курсу буде надано по завершенню курсу.

## СХЕМА КУРСУ

Тиж.	Тема, план, короткі тези	Форма діяльності	Література. Ресурси в Інтернеті.	Завдання (лаб. робота), год	Термін виконання, тиж.
1	Введення в HTML + CSS. Історія розвитку. Теги та їх атрибути. Структура HTML сторінки. Блокові та рядкові елементи. Елементи форми. Семантична розмітка. <style> та <script>. Налаштування середовища розробки. Корисні ресурси та валідація HTML.	лекція, лаб. робота	1-5	Створення базової структури веб-сторінки, 2 год.	2
2	CSS. DOM. Стилі DOM елементів та їх перевірка в браузері. Розміри та відступи. Кольори. Використання кольорів. Розміри та відступи. Позиціонування. Псевдоелементи та псевдокласи. Адаптивний та респонсивний дизайн. CSS анімація.	лекція, лаб. робота	1-5	Стилізація веб-сторінки з допомогою CSS, 2 год.	3
3	Введення в JavaScript. Web Development. Що таке JavaScript. Типи даних. Змінні (ідентифікатори). Перевірка типів даних. Коментарі в коді та виведення результату. Оператори присвоєння та порівняння. Логічні та оператори типів. Побітові оператори.	лекція, лаб. робота	1-5	Робота із різними типами даних в JavaScript, 2 год.	4
4	Умовні конструкції та об'єкти. Умовні конструкції. switch ... case. Об'єкти в JavaScript.	лекція, лаб. робота	1-5	Робота з об'єктами в JavaScript, 2 год.	

	Створення та модифікація. Методи для роботи з об'єктами. Використання об'єктів.				
5	Функції. Функції в JavaScript. Види функцій та як їх створювати. Класи в JavaScript. Використання функцій.	лекція, лаб. робота	1-5	Робота з функціями в JavaScript, 2 год.	
6	Масиви. Масиви в JavaScript. Створення масивів. Модифікація масивів. Використання. Деструктуризація масива. Дефолтне значення параметру функції + spread operator.	лекція, лаб. робота	1-5	Робота з масивами в JavaScript, 2 год.	
7	Цикли. Що таке цикл? Цикл for. Цикл while. Цикл do...while. Використання циклів.	лекція, лаб. робота		Робота з циклами в JavaScript, 2 год.	
8	Методи для роботи з масивами. Як працювати із методами масива? Метод .forEach(). Метод .map(). Метод .filter(). Метод .reduce(). Інші популярні методи.	лекція, лаб. робота	1-5	Робота з методами масивів в JavaScript, 2 год.	5
9	Методи для роботи з рядками. Конкатенація рядків. Template strings. Методи для роботи з рядками.	лекція, лаб. робота	1-5	Робота з рядками в JavaScript, 2 год.	
10	ООП в JavaScript. Принципи ООП. Метод, Конструктор, Наслідування, Інкапсуляція, Абстракція, Поліморфізм. Класи vs функції конструктори. Класи vs функції конструктори. Прототипи. this, execution context та області видимості. .bind(), .call(), .apply().	лекція, лаб. робота	1-5	ООП в JavaScript, 2 год.	6
11	Асинхронність в JavaScript. Що таке асинхронність? Promise. Робота із запитамі. Fetch API. Відловлювання помилок із try...catch.	лекція, лаб. робота	1-6	Робота з асинхронними операціями в JavaScript, 2 год.	9
12	Робота із DOM в JavaScript. Що таке HTML DOM? Методи DOM. Події (events) DOM. DOM EventListener. Зміна CSS властивостей в DOM. cookie, localStorage та sessionStorage.	лекція, лаб. робота	1-5	Робота з DOM в JavaScript, 2 год.	11
13	Введення в React. Початок роботи з React. Елементи та компоненти React. Знайомство з	лекція, лаб. робота	1-5	Створення React застосунку з використанням	12



	JSX. Компоненти функції та класу. Робота з компонентами та атрибутами.			даних із GitHub API, 4 год.	
14	Вступ до стану та життєвого циклу React. Hooks. State. Lifecycle. Рендеринг з умовами. Опрацювання подій. Робота з формами. Робота з HTTP запитами.	лекція, лаб. робота	1-5		13
15	Вступ до NodeJS. Що таке Node.js. Як працює Node.js. Коли використовувати Node.js. Встановлення Node.js. NVM Менеджер пакетів: NPM / Yarn. Глобальні об'єкти. Модулі Node.js. Основні модулі.	лекція, лаб. робота	1-5	Створення серверного застосунку з використанням середовища Node.js та фреймворку Express, 4 год.	15-16
16	NodeJS. Маршрутизація. Проміжне програмне забезпечення. Обробляти запити та відповіді. Обробка помилок. Статична подача. Шаблони візуалізації. Налаштування.	лекція, лаб. робота	1-5		16