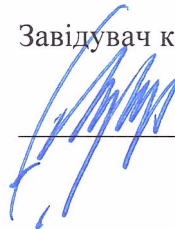


МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
Львівський національний університет імені Івана Франка
Факультет електроніки та комп'ютерних технологій
Кафедра системного проектування

Затверджено

На засіданні
кафедри системного проектування
факультету електроніки та комп'ютерних
технологій
Львівського національного університету
імені Івана Франка
(протокол № 1 від 30.08.2022 р.)

Завідувач кафедри:



Роман ШУВАР

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

Львів 2022 р.

Назва дисципліни	Веб-програмування
Адреса викладання дисципліни	м. Львів, вул. Драгоманова, 50
Факультет та кафедра, за якою закріплена дисципліна	Факультет електроніки та комп'ютерних технологій, кафедра системного проектування
Галузь знань, шифр та назва спеціальності	12 Інформаційні технології 121 Інженерія програмного забезпечення (ВПК)
Викладачі дисципліни	Демків Лідія Степанівна, доцент
Контактна інформація	lidiya.demkiv@lnu.edu.ua https://electronics.lnu.edu.ua/employee/demkiv-l-s/
Консультації з питань навчання по дисципліні відбуваються	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.
Сторінка дисципліни	https://moodle.elct.lnu.edu.ua/course/view.php?id=188
Інформація про дисципліну	Дисципліна “Веб-програмування” є нормативною дисципліною з спеціальності 121 Інженерія програмного забезпечення для освітньої програми «Високопродуктивний комп'ютинг», яка викладається в 4 семестрі в обсязі 5 кредитів (за Європейською Кредитно-Трансферною Системою ECTS).
Коротка анотація дисципліни	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 фрейворків Javascript, таких як: Node.js та React.
Мета та цілі дисципліни	Метою навчальної дисципліни є засвоєння необхідних знань з інформаційних технологій та основ веб-технологій, а також формування твердих практичних навичок щодо розробки сучасних веб-застосунків. Предметом вивчення дисципліни є інформаційні та веб-технології, а також методи їх використання при розробці веб-застосунків різноманітного призначення. Цілями 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.
Література для вивчення	Основна література:

дисципліни	<ol style="list-style-type: none"> 1. Jackson, Jeffrey C. Web technologies : a computer science perspective. - Pearson Education: New Jersey. - 2007. - 591 p. - Режим доступу: http://seu1.org/files/level6/IT230/Book/(web.tech%201st%20book)%20Web%20Technologies%20-%20A%20Computer%20Science%20Perspective.pdf 2. Advance Web Technology: Vardhman Mahaveer Open University, Kota, 2018. - 459 p. - Режим доступу: http://assets.v mou.ac.in/MCA304.pdf . 3. Alex Jouog. Node.js in Action. / Bradley Meck, Mike Canteron // Manning Publication Co. - 2017. - 420 p. - Режим доступу: https://github.com/AbdullahMoawad/Books/raw/master/Node.js%20in%20Action%2C%202nd%20Edition.pdf 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: https://edu.anarcho-copy.org/Programming%20Languages/web/fullstackreact-book-r40.pdf
Обсяг курсу	Кількість кредитів ЄКТС: 5 (150 год), з них: 64 годин аудиторних занять (лекції: 32 год, лабораторні: 32 год.) та 86 год. самостійної роботи.
Очікувані результати навчання	<p>Після вивчення даного курсу здобувачі набудуть таких Загальних(ЗК)/Фахових(ФК) компетентностей та Програмних результатів навчання (ПРН):</p> <p>ЗК01. Здатність до абстрактного мислення, аналізу та синтезу. ЗК02. Здатність застосовувати знання у практичних ситуаціях. ЗК04. Здатність спілкуватися іноземною мовою як усно, так і письмово. ЗК05. Здатність вчитися і оволодівати сучасними знаннями. ЗК06. Здатність до пошуку, оброблення та аналізу інформації з різних джерел. ФК17. Здатність дотримуватися специфікацій, стандартів, правил і рекомендацій в професійній галузі при реалізації процесів життєвого циклу. ФК18. Здатність аналізувати, вибирати і застосовувати методи і засоби для забезпечення інформаційної безпеки (в тому числі кібербезпеки). ФК22. Здатність накопичувати, обробляти та систематизувати професійні знання щодо створення і супроводження програмного забезпечення та визнання важливості навчання протягом всього життя. ФК25. Здатність обґрунтовано обирати та освоювати інструментарій з розробки та супроводження програмного забезпечення. ФК28. Володіння методами розроблення систем підвищеної продуктивності, серверними та розподіленими технологіями, інструментальними засобами проектування та розробки веб-застосунків і нових технологій. ПРН07. Знати і застосовувати на практиці фундаментальні концепції, парадигми і основні принципи функціонування мовних, інструментальних і обчислювальних засобів інженерії програмного забезпечення. ПРН08. Вміти розробляти людино-машинний інтерфейс. ПРН14. Застосовувати на практиці інструментальні програмні засоби доменного аналізу, проектування, тестування, візуалізації, вимірювань та документування програмного забезпечення.</p>

	ПРН17. Вміти застосовувати методи компонентної розробки програмного забезпечення.
Ключові слова	Веб, Інтернет, Веб-технології, Веб-програмування, Web, Search Engine, Ajax, React, HTTP, CSS, JavaScript, Node.
Формат курсу	Очний. Проведення лекцій, лабораторних робіт та консультації для кращого розуміння тем.
Теми	Див. СХЕМА КУРСУ
Підсумковий контроль, форма	Екзамен в кінці семестру
Сертифікація	Сертифікація не є обов'язковим елементом дисципліни, а тільки дозволяє оцінити свої можливості для працевлаштування: <ul style="list-style-type: none"> - CSS Skills - JavaScript (Basic) Skills - JavaScript (Intermediate) Skills - Node (Basic) Skills - Node.js (Intermediate) Skills - React (Basic) Skills - Rest API (Intermediate) Skills
Пререквізити	Для вивчення курсу студенти повинні отримати базові знання з дисциплін: “Бази даних”, “Операційні системи”, “Веб-технології”, “Об’єктно-орієнтоване програмування”.
Навчальні методи та техніки, які будуть використовуватися під час викладання курсу	Презентація, лекції, лабораторні роботи, обговорення, дискусія.
Необхідне обладнання	Мультимедіа, платформа Moodle
Критерії оцінювання (окремо для кожного виду навчальної діяльності)	Оцінювання проводиться упродовж семестру за 100-бальною 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: <ul style="list-style-type: none"> • контрольні заміри (2 модулі): 50% семестрової оцінки; максимальна кількість балів: 50, а саме: <ul style="list-style-type: none"> - лабораторні роботи: 60% оцінки контрольного заміру; максимальна кількість балів: 32 (14 лабораторних робіт, де: перші 12 лабораторних оцінюються по 2 бали, з яких 1 бал за виконання та 1 бал за опитування чи тестування; 13-а і 14-а роботи оцінюються по 4 бали, з яких 3 бали за виконання та 1 бал за опитування чи тестування). - теоретичний матеріал: 40% оцінки контрольного заміру; максимальна кількість балів: 18 (2 тести по 9 балів кожний, де 1 питання важить 0.5 бала за правильну відповідь). • екзамен: 50% семестрової оцінки; максимально 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.

Оцінювання екзаменаційних питань:

20-25 балів - розглянута тема відтворюється в повному обсязі, правильно, обґрунтовано, логічно, які містять аналіз і систематизацію, аргументовані висновки. Засвідчено глибоке володіння матеріалом. Наведені приклади коду повністю робочі та відповідають темі. Можуть бути присутні несуттєві помилки та невідповідності;

15-19 балів - відтворюється значна частина розглянутої теми. Виявлено знання і розуміння основних положень навчальної дисципліни, проте присутні неточності та/або невідповідності основній темі. Наведені приклади коду частково робочі, проте в загальному відповідають темі;

10-14 балів - відстежується загальне розуміння розглянутої теми. Виявлені множинні неточності та невідповідності, пояснення наведеного коду відсутні, код функціонує із значними неточностями (або відсутні приклади запуску коду на виконання взагалі);

5-9 балів – студент погано розуміє розглянуту тему. Виявлені суттєві неточності та невідповідності. Наведені приклади коду з суттєвими недоліками, або не відповідають темі;

0-4 бали – студент взагалі не розуміє розглянуту тему. Тему не розкрито, кількість викладеного матеріалу не відповідає загальним нормам обраного виду роботи. Наведений код не робочий, або відсутній як такий.

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.

Відвідання 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.
<p>Питання до контрольних робіт</p>	<p>Перелік питань та завдань для проведення підсумкової оцінки знань певних тем до контрольних робіт:</p> <ol style="list-style-type: none"> 1. Етапи розвитку інформаційних технологій (<i>Види сучасних інформаційних технологій. Структура і принципи Веб. Використання інформаційних технологій.</i>) 2. Поняття про веб-технології (<i>Структура і принципи Веб. Використання інформаційних технологій. Структура і принципи сучасних WEB-технологій.</i>) 3. Технології веб-програмування (<i>Клієнтські технології веб-програмування. Серверні технології веб-програмування. Клієнт-серверна WEB- архітектура. </i>) 4. Поняття про модель веб-додатку (<i>Сучасна модель веб-додатку. Засоби для розроблення сайтів. Поняття хостінгу. Поняття контент менеджмент систем (КМС)</i>) 5. Поняття HTML (<i>Прості вебсторінки. Теги HTML. Правила написання, атрибути. Форматування веб-сторінок за допомогою тегів. Візуальні Webредактори для розробки вебсторінок</i>) 6. Поєднання HTML + CSS (<i>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</i>) 7. Статичні веб-сторінки (<i>Створення статичних веб-сторінок з допомогою HTML. Document Type Definition (DTD). Об'єктна модель документа DOM. Спеціальні теги заголовка веб-документа. XHTML</i>) 8. Дизайн та стилізація сайту (<i>Типографіка сайту. Розробка дизайну. Підготовка графіки для сайту. Графічні формати. Верстка HTML-документу за макетом. Типи макетів.</i>) 9. Поняття CSS (<i>Визначення стиля CSS. Використання CSS в HTML документах. CSS синтаксис. Каскадування, наслідування і пріоритет стилів. CSS властивості.</i>) 10. Технології CSS (<i>DOM. Стили 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.</i>) 11. Стилiзація сайту за допомогою CSS (<i>Media-запити. Ідентифікатори. Контекстні селектори. Сусідні селектори. Дочірні селектори. Селектори атрибутів. Універсальний селектор. Псевдокласи. Псевдоелементи. Оформлення веб-сторінок допомогою CSS3</i>) 12. Динамічні компоненти веб-сторінок (<i>Блокова модель. Позиціонування. Потік виведення для HTML документа. Модульна сітка. Плаваючі (float) блоки. Wrap-блоки. z-index позиціонування. Flex-контейнер та flex-елементи, їх властивості</i>)

13. Основи розроблення сайту (*Послідовність дій і специфіка створення сайту. Основні засоби HTML та CSS для створення сайту.*)
14. Поняття JavaScript (*Історія появи JS. Стандарти JS. Змінні. Коментарі. Літерали. Зарезервовані слова. Типи даних. Wrapper objects. Числа. Чисельні оператори. Конвертація типів даних. Рядки. Методи для роботи з рядками. Булеві змінні та логічні оператори. Умови та цикли.*)
15. Поняття ECMAScript (*Поняття ECMAScript. Формування блоків. Оголошення на рівні блоку. Змінні блоку. Опис констант. Робиття структури даних на менші частини (Destructuring).*)
16. Робота із складними структурами даних в ECMAScript (*Розбиття масивів. Розбиття об'єктів. Вкладені розбиття. Функції. Параметри і аргументи функцій по замовчуванню. Оператор розповсюдження (Spread). Arrow functions. Рядки. Методи для рядків. Об'єкти і прототипи. Класи. Модулі. Колекції.*)
17. Об'єкти в JavaScript (*Способи створення об'єктів. Методи об'єктів. Цикли for-in, for-of. Класи. Прототипне наслідування. Огляд об'єкта Window. Адреса документа і навігація по ньому. Історія відвідувань. Об'єкт Navigator. Об'єкт Screen.*)
18. Об'єктна модель документа (*Огляд моделі DOM. CSSOM конструкції. Вибір елементів документа. Вибір елементів за значенням атрибута id. Вибір елементів за значенням атрибута name. Вибір елементів за типом. Вибір елементів за класом CSS. Вибір елементів з використанням селекторів CSS. Структура документа і навігація по документу. Взаємодія JavaScript і CSS.*)
19. Обробка подій на JavaScript (*Типи подій. Порядок опрацювання подій. Потік подій. Bubbling. Події браузера. Передача подій. Події руку. Події клавіатури. Події миші.*)
20. Вступ до AJAX (*WWW компоненти. Структурні та семантичні компоненти. WWW інфраструктура. URL. Робота HTTP протоколу. AJAX - Asynchronous JavaScript and XML. Синхронні та асинхронні моделі. XMLHttpRequest об'єкт. JQuery AJAX. JQuery AJAX методи. Створення динамічного наповнення вебсторінки з допомогою AJAX*)
21. Використання AJAX (*Основні поняття: Promise, fetch, async/await. Властивості об'єкта promise. setTimeout. XMLHttpRequest. Promise API. Promise.allSettled. Blob. Заголовки запитів та відповідей. Заборонені HTTP заголовки. Запити POST.*)
22. Прогресивні Web-технології (*Розвиток технологій Web. Технологія Web3. Технології: 5G, NFT, блокчейн та штучний інтелект. Технології метавсесвіту. Веб-дизайн та штучний інтелект. Вплив штучного інтелекту на веб-технології*)
23. Основи 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.*)

	<p>24. Умовні конструкції та об'єкти в JavaScript (Умовні конструкції. <i>switch ... case</i>. Об'єкти в JavaScript. Створення та модифікація. Методи для роботи з об'єктами. Використання об'єктів)</p> <p>25. Функції в JavaScript (Поняття функції в JavaScript. Види функцій та як їх створювати. Класи в JavaScript. Використання функцій.)</p> <p>26. Обробка масивів в JavaScript (Масиви в JavaScript. Створення масивів. Модифікація масивів. Використання. Деструктуризація масива. Дефолтне значення параметру функції + <i>spread operator</i>.)</p> <p>27. Робота з циклами в JavaScript (Що таке цикл? Цикл <i>for</i>. Цикл <i>while</i>. Цикл <i>do...while</i>. Використання циклів.)</p> <p>28. Методи для роботи з масивами в JavaScript (Як працювати із методами масива? Метод <i>forEach()</i>. Метод <i>map()</i>. Метод <i>filter()</i>. Метод <i>reduce()</i>. Інші популярні методи.)</p> <p>29. Методи для роботи з рядками в JavaScript (Конкатенація рядків. <i>Template strings</i>. Методи для роботи з рядками. Приклади.)</p> <p>30. ООП в JavaScript (Принципи ООП. Метод, Конструктор, Наслідування, Інкапсуляція, Абстракція, Поліморфізм. Класи vs функції конструктори. Класи vs функції конструктори. Прототипи. <i>this</i>, <i>execution context</i> та області видимості. <i>.bind()</i>, <i>.call()</i>, <i>.apply()</i>)</p> <p>31. Асинхронність в JavaScript (Що таке асинхронність? <i>Promise</i>. Робота із запитами. <i>Fetch API</i>. Відловлювання помилок із <i>try...catch</i>.)</p> <p>32. Робота із DOM в JavaScript (Що таке HTML DOM? Методи DOM. Події (events) DOM. <i>DOM EventListener</i>. Зміна CSS властивостей в DOM. <i>cookie</i>, <i>localStorage</i> та <i>sessionStorage</i>.)</p> <p>33. Введення в React (Початок роботи з React. Елементи та компоненти React. Знайомство з <i>JSX</i>. Компоненти функції та класу. Робота з компонентами та атрибутами.)</p> <p>34. Вступ до стану та життєвого циклу React (<i>Hooks</i>. <i>State</i>. <i>Lifecycle</i>. Рендеринг з умовами. Опрацювання подій. Робота з формами. Робота з <i>HTTP</i> запитами.)</p> <p>35. Вступ до NodeJS (Що таке <i>Node.js</i>. Як працює <i>Node.js</i>. Коли використовувати <i>Node.js</i>. Встановлення <i>Node.js</i>. <i>NVM</i> Менеджер пакетів: <i>NPM</i> / <i>Yarn</i>. Глобальні об'єкти. Модулі <i>Node.js</i>. Основні модулі.)</p> <p>36. Робота з NodeJS (Маршрутизація. Проміжне програмне забезпечення. Обробляти запити та відповіді. Обробка помилок. Статична подача. Шаблони візуалізації. Налаштування.)</p>
<p>Опитування</p>	<p>Анкету-оцінку з метою оцінювання якості курсу буде надано по завершенню курсу.</p>

СХЕМА КУРСУ

Тиж.	Тема, план, короткі тези	Форма діяльності	Література. Ресурси в Інтернеті.	Завдання, 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	1-2	2	кінець поточного тижня
	Створення базової структури веб-сторінки	лаб. робота	1-2	2	кінець поточного тижня
	Синтаксис JavaScript	сам. робота	1-2	3,5	кінець поточного тижня
2	CSS. DOM. Стили 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	1-2	2	кінець поточного тижня
	Стилізація веб-сторінки з допомогою CSS	лаб. робота	1-2	2	кінець поточного тижня
	Функції на JavaScript	сам. робота	1-2	3,5	кінець поточного тижня
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	1-3	2	кінець поточного тижня
	Робота із різними типами даних в JavaScript	лаб. робота	1-3	2	кінець поточного тижня
	Робота з об'єктами на JavaScript	сам. робота	1-3	3,5	кінець поточного тижня
4	Умовні конструкції та об'єкти. Умовні конструкції. switch ... case. Об'єкти в JavaScript. Створення та модифікація.	лекція	1-3	2	кінець поточного тижня

	Методи для роботи з об'єктами. Використання об'єктів.				
	Робота з об'єктами в JavaScript	лаб. робота	1-3	2	кінець поточного тижня
	Робота з об'єктними моделями документів на JavaScript	сам. робота	1-3	3,5	кінець поточного тижня
5	Функції. Функції в JavaScript. Види функцій та як їх створювати. Класи в JavaScript. Використання функцій.	лекція	1-3	2	кінець поточного тижня
	Робота з функціями в JavaScript	лаб. робота	1-3	2	кінець поточного тижня
	Основні питання роботи з Angular	сам. робота	1-3	3,5	кінець поточного тижня
6	Масиви. Масиви в JavaScript. Створення масивів. Модифікація масивів. Використання. Деструктуризація масива. Дефолтне значення параметру функції + spread operator.	лекція	1-3	2	кінець поточного тижня
	Робота з масивами в JavaScript	лаб. робота	1-3	2	кінець поточного тижня
	Поняття програмування на Angular	сам. робота	1-3	3,5	кінець поточного тижня
7	Цикли. Що таке цикл? Цикл for. Цикл while. Цикл do...while. Використання циклів.	лекція	1-3	2	кінець поточного тижня
	Робота з циклами в JavaScript	лаб. робота	1-3	2	кінець поточного тижня
	Програмування на Angular	сам. робота	1-3	3,5	кінець поточного тижня
8	Методи для роботи з масивами. Як працювати із методами масива? Метод .forEach(). Метод .map(). Метод .filter(). Метод .reduce(). Інші популярні методи.	лекція	1-3	2	кінець поточного тижня
	Робота з методами масивів в JavaScript	лаб. робота	2-3	2	кінець поточного тижня

	Основні питання роботи з Ajax	сам. робота	2-3	3,5	кінець поточного тижня
9	Методи для роботи з рядками. Конкатенація рядків. Template strings. Методи для роботи з рядками.	лекція	2-3	2	кінець поточного тижня
	Робота з рядками в JavaScript	лаб. робота	2-3	2	кінець поточного тижня
	Взаємодія з клієнтом на Ajax	сам. робота	2-3	3,5	кінець поточного тижня
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	1-3	2	кінець поточного тижня
	ООП в JavaScript	лаб. робота	1-3	2	кінець поточного тижня
	Робота з React	сам. робота	4	3,5	кінець поточного тижня
11	Асинхронність в JavaScript. Що таке асинхронність? Promise. Робота із запитами. Fetch API. Відловлювання помилок із try...catch.	лекція	1-4	2	кінець поточного тижня
	Робота з асинхронними операціями в JavaScript	лаб. робота	1-4	2	кінець поточного тижня
	Програмування front-end на React	сам. робота	4	3,5	кінець поточного тижня
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	4	2	кінець поточного тижня
	Робота з DOM в JavaScript	лаб. робота	1-4	2	кінець поточного тижня
	Програмування back-end на React	сам. робота	4	3,5	кінець поточного тижня

13	Введення в React. Початок роботи з React. Елементи та компоненти React. Знайомство з JSX. Компоненти функції та класу. Робота з компонентами та атрибутами.	лекція	4	2	кінець поточного тижня
	Створення React застосунку з використанням даних із GitHub API	лаб. робота	4	4	кінець поточного тижня
	Програмування веб-сервісів на React	сам. робота	4	3,5	кінець поточного тижня
14	Вступ до стану та життєвого циклу React. Hooks. State. Lifecycle. Рендеринг з умовами. Опрацювання подій. Робота з формами. Робота з HTTP запитамі.	лекція	4	2	кінець поточного тижня
	Основні питання роботи з Node.js	сам. робота	2-3	3,5	кінець поточного тижня
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	3	2	кінець поточного тижня
	Створення серверного застосунку з використанням середовища Node.js та фреймворку Express	лаб. робота	2-3	4	кінець поточного тижня
	Програмування back-end на Node.js	сам. робота	2-3	3,5	кінець поточного тижня
16	NodeJS. Маршрутизація. Проміжне програмне забезпечення. Обробляти запити та відповіді. Обробка помилок. Статична подача. Шаблони візуалізації. Налаштування.	лекція	2-3	2	кінець поточного тижня
	Програмування веб-сервісів на Node.js	сам. робота	2-3	3,5	кінець поточного тижня