

**МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ**  
**Львівський національний університет імені Івана Франка**  
**Факультет прикладної математики та інформатики**  
**Кафедра програмування**

**Затверджено**

На засіданні кафедри програмування  
факультету прикладної математики  
Львівського національного університету  
імені Івана Франка  
(протокол № 1 від 29 серпня 2025 р.)



Зав. кафедри к. ф.-м. н., доц. Ярошко С. А.

**Силабус з навчальної дисципліни**  
**«Прикладне програмування на NodeJS»,**  
**що викладається в межах ОПП “Середня освіта (Інформатика)”**  
**першого (бакалаврського) рівня вищої освіти**  
**для здобувачів зі спеціальності А4.09 Середня освіта (Інформатика)**

Львів 2025 р.

<b>Назва дисципліни</b>	Прикладне програмування на NodeJS
<b>Адреса викладання дисципліни</b>	Львівський національний університет імені Івана Франка, вул. Університетська 1, м. Львів, Україна, 79000
<b>Факультет та кафедра, за якою закріплена дисципліна</b>	Факультет прикладної математики та інформатики, кафедра програмування
<b>Галузь знань, шифр та назва спеціальності</b>	Галузь знань: А Освіта / Педагогіка Спеціальність: А4.09 Середня освіта (Інформатика)
<b>Викладачі дисципліни</b>	Нобіс Віталій Валерійович, асистент кафедри програмування
<b>Контактна інформація викладачів</b>	Електронна пошта: vitalii.nobis@lnu.edu.ua, веб-сторінка: <a href="https://ami.lnu.edu.ua/employee/nobis">https://ami.lnu.edu.ua/employee/nobis</a>
<b>Консультації з питань навчання по дисципліні відбуваються</b>	Консультації проводять раз на тиждень згідно з оприлюдненим розкладом консультацій викладача. Можливі он-лайн консультації через Microsoft Teams. Для погодження часу он-лайн консультацій слід писати на електронну пошту викладача.
<b>Інформація про дисципліну</b>	Дисципліна «Прикладне програмування на NodeJS» є дисципліною вільного вибору зі спеціальності А4.09 Середня освіта (Інформатика) для освітньої програми Середня освіта (Інформатика), яку викладають у шостому семестрі в обсязі 4,5 кредитів (за Європейською кредитно-трансферною системою ECTS)
<b>Коротка анотація дисципліни</b>	Фокус уваги курсу спрямовано на вивчення основ програмування JavaScript. Використання об'єктів та ланцюжків прототипів, асинхронні виклики функцій. Розглянуто відмінності мови JavaScript від строго типізованих мов програмування. Курс пов'язаний з дисциплінами "Програмування", "Теорія алгоритмів", "Алгоритми і структури даних".
<b>Мета та цілі дисципліни</b>	Головною метою дисципліни «Прикладне програмування на NodeJS» є отримання теоретичних та практичних знань використання мови програмування JavaScript. Написання програм з використанням підходів функціонального програмування. Освоєння основ асинхронного виконання коду у середовищі Node.js
<b>Література для вивчення дисципліни</b>	<i>Основна література</i> 1. MDN Web Docs [Електронний ресурс] – Режим доступу: <a href="https://developer.mozilla.org/en-US/docs/Web/JavaScript">https://developer.mozilla.org/en-US/docs/Web/JavaScript</a> 2. Документація Node.js - Node.js [Електронний ресурс] – Режим доступу: <a href="https://nodejs.org/uk/docs">https://nodejs.org/uk/docs</a> 3. Dr. Axel Rauschmayer: Deep JavaScript: Theory and techniques [Електронний ресурс] – Режим доступу: <a href="https://exploringjs.com/deep-js/index.html">https://exploringjs.com/deep-js/index.html</a>  <i>Додаткова література</i> 1. Dr. Axel Rauschmayer: Exploring ES6 [Електронний ресурс] – Режим доступу: <a href="https://exploringjs.com/es6.html">https://exploringjs.com/es6.html</a> 2. Dr. Axel Rauschmayer: Exploring ES2016 and ES2017 [Електронний ресурс] – Режим доступу: <a href="https://exploringjs.com/es2016-es2017.html">https://exploringjs.com/es2016-es2017.html</a> 3. Dr. Axel Rauschmayer: Exploring ES2018 and ES2019 [Електронний ресурс] – Режим доступу: <a href="https://exploringjs.com/es2018-es2019/index.html">https://exploringjs.com/es2018-es2019/index.html</a>
<b>Обсяг курсу</b>	Загальний обсяг: 135 годин. Аудиторних занять: 64 год. з них 32 години лекцій, 32 години лабораторних занять та 71 годин самостійної роботи
<b>Очікувані результати навчання</b>	Після завершення цього курсу студент буде : <b>Знати</b> - синтаксис мови JavaScript - засоби відлагодження та середовище виконання Node.js; - особливості роботи асинхронних методів у JavaScript.

	<p><b>Вміти</b></p> <ul style="list-style-type: none"> <li>- створювати програмні застосунки JavaScript;</li> <li>- працювати з об'єктами (ланцюжками прототипів) JavaScript;</li> <li>- написати асинхронні методи у JavaScript.</li> </ul>															
<b>Компетентності</b>	<p><i>Інтегральна:</i> Здатність розв'язувати складні спеціалізовані задачі та практичні проблеми у галузі комп'ютерних наук або у процесі навчання, що передбачають застосування теорій та методів інформаційних технологій і характеризується комплексністю та невизначеністю умов.</p> <p><i>Загальні (ЗК):</i>  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.  ЗК7. Здатність до пошуку, оброблення та аналізу інформації з різних джерел.  ЗК12. Здатність оцінювати та забезпечувати якість виконуваних робіт.</p> <p><i>Спеціальні (фахові, предметні) компетентності (СК):</i>  СК3. Здатність до логічного мислення, побудови логічних висновків, використання формальних мов і моделей алгоритмічних обчислень, проектування, розроблення й аналізу алгоритмів, оцінювання їх ефективності та складності, розв'язності та нерозв'язності алгоритмічних проблем для адекватного моделювання предметних областей і створення програмних та інформаційних систем.  СК7. Здатність застосовувати теоретичні та практичні основи методології та технології моделювання для дослідження характеристик і поведінки складних об'єктів і систем, проводити обчислювальні експерименти з обробкою й аналізом результатів.  СК8. Здатність проектувати 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.</p>															
<b>Програмні результати навчання</b>	<p>ПР1. Застосовувати знання основних форм і законів абстрактно-логічного мислення, основ методології наукового пізнання, форм і методів вилучення, аналізу, обробки та синтезу інформації в предметній області комп'ютерних наук.</p> <p>ПР9. Розробляти програмні моделі предметних середовищ, вибирати парадигму програмування з позицій зручності та якості застосування для реалізації методів та алгоритмів розв'язання задач в галузі комп'ютерних наук.</p> <p>ПР10. Використовувати інструментальні засоби розробки клієнт-серверних застосунків, проектувати 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.</p>															
<b>Ключові слова</b>	Програмування на NodeJS, Node.js, JavaScript, JS, асинхронні функції у JavaScript, JS prototype															
<b>Формат курсу</b>	Очний: проведення лекцій, лабораторних робіт та консультацій; можливий також онлайн режим викладання з використанням засобів відеозв'язку та розміщенням навчальних матеріалів у хмарному сховищі.															
<b>Теми</b>	<table border="1"> <thead> <tr> <th>Тиж.</th> <th>Тема, план, короткі тези</th> <th>Форма діяльності (заняття)* *лекція, самостійна, дискусія, групова робота)</th> <th>Завдання, год</th> <th>Термін виконання</th> </tr> </thead> <tbody> <tr> <td>1-2</td> <td>Програмування на JavaScript: історія, базові типи, об'єкти та функції. Базові оператори та цикли.</td> <td>Лекція</td> <td>4</td> <td></td> </tr> <tr> <td>1-2</td> <td>Середовище програмування. Відлагодження програм. Використання базових типів та операторів.</td> <td>Лабораторне, групова робота</td> <td>4</td> <td>Наступне ЛЗ</td> </tr> </tbody> </table>	Тиж.	Тема, план, короткі тези	Форма діяльності (заняття)* *лекція, самостійна, дискусія, групова робота)	Завдання, год	Термін виконання	1-2	Програмування на JavaScript: історія, базові типи, об'єкти та функції. Базові оператори та цикли.	Лекція	4		1-2	Середовище програмування. Відлагодження програм. Використання базових типів та операторів.	Лабораторне, групова робота	4	Наступне ЛЗ
Тиж.	Тема, план, короткі тези	Форма діяльності (заняття)* *лекція, самостійна, дискусія, групова робота)	Завдання, год	Термін виконання												
1-2	Програмування на JavaScript: історія, базові типи, об'єкти та функції. Базові оператори та цикли.	Лекція	4													
1-2	Середовище програмування. Відлагодження програм. Використання базових типів та операторів.	Лабораторне, групова робота	4	Наступне ЛЗ												

	3-4	Об'єкт та ланцюжок прототипів. Класи в JavaScript.	Лекція	4	
	3-4	Створення класів та їх використання.	Лабораторне, групова робота	4	Наступне ЛЗ
	5-6	Функції та їхнє використання в JavaScript.	Лекція	4	
	5-6	Особливості роботи з функціями.	Лабораторне, групова робота	4	Наступне ЛЗ
	7-8	Використання менеджера пакетів.	Лекція	4	
	7-8	Ознайомлення з NPM та YARN. Створення проекту з встановленими пакетами.	Лабораторне, групова робота	4	Наступне ЛЗ
	9-10	Асинхронні функції в JavaScript. Callbacks. Promises.	Лекція	4	
	9-10	Використання підходів callback, promise. Робота з файловою системою.	Лабораторне, групова робота	4	Наступне ЛЗ
	11-12	Асинхронні функції в JavaScript. Async functions.	Лекція	4	
	11-12	Використання підходів async functions. Порівняння різних підходів асинхронних функцій.	Лабораторне, групова робота	4	Наступне ЛЗ
	13-14	Створення API сервера. Основи підходів REST.	Лекція	4	
	13-14	Побудова REST API сервера. Відлагодження API.	Лабораторне, групова робота	4	Наступне ЛЗ
	15-16	Робота з базою даних.	Лекція	4	
	15-16	Створення комплексних серверних застосунків.. Залік	Лабораторне, групова робота	4	Наступне ЛЗ
<b>Підсумковий контроль, форма</b>	залік в кінці семестру				
<b>Пререквізити</b>	Для вивчення курсу студенти потребують базових знань з дисциплін "Програмування", "Алгоритми і структури даних", "Теорія алгоритмів" та вмінь знаходити необхідні документи в мережі інтернет, інсталювати та налаштовувати вільно поширювані середовища програмування.				
<b>Навчальні методи та техніки, які будуть використані під час викладання курсу</b>	Самостійна робота з вивченням оприлюднених електронних матеріалів. Проведення лекцій з розглядом детальним розглядом основ мови програмування JavaScript. Лабораторні заняття у вигляді виконання практичних завдань та обговорення написаних програм.				
<b>Необхідне обладнання</b>	Для проведення лекцій: комп'ютер, проектор, доступ до мережі інтернет. Для проведення лабораторних та виконання завдань: комп'ютер, ОС Windows/Linux, доступ до інтернету, безкоштовні середовища програмування Node.js, Visual Studio Code.				
<b>Критерії оцінювання (окремо для кожного виду навчальної діяльності)</b>	<p>Виконання та захист студентами індивідуальних завдань та проектів.</p> <p>Оцінювання проводиться за 100-бальною шкалою. 60 балів нараховують за виконання лабораторних завдань: по 10 балів за кожну з 6 лабораторних робіт. Ще 40 балів – за створення серверного застосунку.</p> <p>Кожна робота оцінюється максимально за такою схемою:</p> <ul style="list-style-type: none"> <li>• 100% – умови завдання виконано повністю, автор відповідає на всі запитання щодо використаних підходів, чітко інтерпретує отримані результати, немає ознак недоброчесності;</li> <li>• 80% – наведено логічно правильну послідовність розв'язування, алгоритми складено правильно, бракує окремих коментарів, автор не досить повно пояснює використані підходи, немає ознак недоброчесності;</li> <li>• 60% – у правильній послідовності розв'язування допущено окремі помилки, які автор уміє виправити після зауваження викладача, бракує коментарів, на запитання щодо використаних підходів автор відповідає з помилками, немає ознак недоброчесності;</li> <li>• 40% – у правильній послідовності розв'язування пропущено окремі етапи, завдання виконано частково, автор не розуміє недоліків поданої роботи, не вміє їх виправити, немає ознак недоброчесності;</li> </ul>				

	<ul style="list-style-type: none"> <li>• 20% – завдання виконано частково, автор не може самостійно інтерпретувати отримані результати, виправити помилки, немає ознак недоброчесності;</li> <li>• 0% – завдання не виконано, або ж виявлено ознаки недоброчесності: запозичення, автор не володіє відповідним теоретичним матеріалом тощо;</li> </ul> <p>Активна участь у обговоренні лекційного матеріалу, самостійне відшукування та використання додаткової інформації для виконання робіт є підставою для нарахування додаткових балів.</p> <p><b>Відвідання занять</b> є важливою складовою навчання. Студенти мають інформувати викладача про неможливість відвідати заняття. У будь-якому випадку студенти зобов'язані дотримуватися усіх строків визначених для виконання робіт, передбачених курсом. Виконані роботи завантажують у відповідне хмарне сховище (Microsoft Teams).</p> <p>Альтернативою відвідування лабораторних занять в університеті може бути дистанційна онлайн робота за розкладом проведення занять. Активність на лекціях і лабораторних враховують при оцінюванні відповідного лабораторного завдання.</p> <p>Студенти заохочуються до використання також й іншої літератури та джерел, яких немає серед рекомендованих.</p>
<b>Опитування</b>	Анкету-оцінку з метою оцінювання якості курсу буде надано по завершенню курсу.