

МІНІСТЕРСТВО НАУКИ І ОСВІТИ УКРАЇНИ  
ЖИТОМИРСЬКИЙ ДЕРЖАВНИЙ ТЕХНОЛОГІЧНИЙ УНІВЕРСИТЕТ  
Кафедра комп'ютерної інженерії

**“ЗАТВЕРДЖУЮ”**

Завідувач кафедри комп'ютерної  
інженерії

\_\_\_\_\_ Єфіменко А.А.

“ \_\_\_\_\_ ” \_\_\_\_\_ 20\_\_ року

## **РОБОЧА ПРОГРАМА НАВЧАЛЬНОЇ ДИСЦИПЛІНИ**

### **«Node.js»**

напряму підготовки: 6.050103 «Програмна інженерія»

Факультет інформаційно-комп'ютерних технологій

Робоча програма дисципліни «Node.js» для студентів напрямку 6.050103 «Програмна інженерія».

Розробники:

Оринчак Іван Андрійович, старший викладач кафедри комп'ютерної інженерії

Робочу програму схвалено на засіданні кафедри комп'ютерної інженерії

Протокол від “ \_\_\_\_ ” \_\_\_\_\_ 20\_\_ року № \_\_\_\_

Завідувач кафедри комп'ютерної інженерії

\_\_\_\_\_ Єфіменко А.А.

### 1. Опис навчальної дисципліни

Найменування показників	Галузь знань, напрям підготовки, освітньо-кваліфікаційний рівень	Характеристика навчальної дисципліни	
		денна форма навчання / заочна форма навчання	
Кількість кредитів – 6	Галузь знань 0501 „Інформатика та обчислювальна техніка”	Варіативна	
	Напрямок підготовки 6.050103 «Програмна інженерія»		
Модулів – 1		Рік підготовки	
Змістових модулів – 1		2016-й	2017-й
Загальна кількість годин – 216		Семестр	
		1-й	2-й
Тижневих годин для денної форми навч.: аудиторних – 5 самостійної роботи студента – 8,5	Освітньо-кваліфікаційний рівень: бакалавр	Лекції	
		16 год.	- год.
		Практичні, семінарські	
		- год.	- год.
		Лабораторні	
		64 год.	- год.
		Самостійна робота	
		136 год.	- год.
		Індивідуальні завдання:	
		- год.	
Вид контролю:			
залік – 1 сем.			

Примітка.

Співвідношення кількості годин аудиторних занять до самостійної і індивідуальної роботи становить (%): 80/136  
для заочної форми навчання – 12/204

## 2. Мета та завдання навчальної дисципліни

**Метою** викладання навчальної дисципліни є навчити студентів самостійно проектувати та реалізовувати додатки різної складності за допомогою Node.js з використанням структурно-модульного методу програмування.

**Основними завданнями** вивчення дисципліни, які стоять перед студентами - це вивчення принципів використання платформи Node.js та засвоєння практичних аспектів побудови базових алгоритмів та програм різного рівня складності.

У результаті вивчення навчальної дисципліни студент повинен

**знати:**

- передумови і історію виникнення предмету програмування;
- методи структурного програмування;
- основні синтаксичні конструкції мови Node.js;
- основні методології розробки програмного забезпечення;
- основи забезпечення та контролю якості програмного забезпечення.

**вміти:**

- складати програми мовою Node.js;
- застосувати грамотний стиль програмування;
- побудувати структурований алгоритм обробки базових структур даних;
- програмно реалізувати алгоритм у вигляді окремих додатків;
- виділити загальні методи обробки даних у окремі процедурні блоки та запрограмувати їх.

## 3. Програма навчальної дисципліни

### Змістовий модуль 1. Робота з Node.js.

Тема 1. Встановлення та запуск Node.js. Створення середовища розробки для Node.

Тема 2. Інтерактивний режим роботи з Node з використання REPL. Переваги REPL. REPL-команди.

Тема 3. Ядро Node. Глобальні об'єкти. Сервери, потоки вводу-виводу, сокети.

Тема 4. Модульна система Node. Зовнішні модулі. Завантаження модулів.

Тема 5. Маршрутизація трафіку, службові файли. Маршрутизатори. Проксі-сервери.

Тема 6. Платформа Express. Встановлення. Обробка помилок. Маршрутизація.

Тема 7. Express, системи шаблонів та CSS. Вбудований JavaScript-код.

Тема 8. Отримання структурованих даних в Node та Redis.

Тема 9. Node та прив'язка до реляційних баз даних. Робота з db-mysql

Тема 10. Веб-сокети. Модуль Socet-IO.

Тема 11. Тестування та відладка Node-додатків. Блочне тестування. Порівняльне тестування.

Тема 12. Захист Node-додатків. Шифрування додатків. Протидія атакам.

#### 4. Структура навчальної дисципліни

Назви змістових модулів і тем	Кількість годин					
	денна форма					
	усього	у тому числі				
л		п	лаб.	інд.	с. р.	
1	8	9	10	11	12	13
<b>Модуль 1. Робота з Node.js.</b>						
<b>Змістовий модуль 1. Робота з Node.js.</b>						
Тема 1. Встановлення та запуск Node.js. Створення середовища розробки для Node.	16	2		4		10
Тема 2. Інтерактивний режим роботи з Node з використання REPL. Переваги REPL. REPL-команди.	14			4		10
Тема 3. Ядро Node. Глобальні об'єкти. Сервери, потоки вводу-виводу, сокети.	16	2		4		10
Тема 4. Модульна система Node. Зовнішні модулі. Завантаження модулів.	22	2		8		12
Тема 5. Маршрутизація трафіку, службові файли. Маршрутизатори. Проксі-сервери.	16			4		12
Тема 6. Платформа Express. Встановлення. Обробка помилок. Маршрутизація.	22	2		8		12
Тема 7. Express, системи шаблонів та CSS. Вбудований JavaScript-код.	22	2		8		12
Тема 8. Отримання структурованих даних в Node та Redis.	18	2		4		12
Тема 9. Node та прив'язка до реляційних баз даних. Робота з db-mysql.	22	2		8		12
Тема 10. Веб-сокети. Модуль Socet-IO.	18	2		4		12
Тема 11. Тестування та відладка Node-додатків. Блочне тестування. Порівняльне тестування.	16			4		12
Тема 12. Захист Node-додатків. Шифрування додатків. Протидія атакам.	14			4		10
Разом за змістовим модулем 1	216	16		64		136
<b>Усього годин</b>						

#### 4. Теми семінарських занять

Семінарські заняття не передбачено навчальним планом

#### 5. Теми практичних занять

Практичні заняття не передбачено навчальним планом

### 7. Теми лабораторних занять

№ з/п	Назва теми	Кількість годин
1.	Тема 1. Встановлення та запуск Node.js. Створення середовища розробки для Node.	4
2.	Тема 2. Інтерактивний режим роботи з Node з використання REPL. Переваги REPL. REPL-команди.	4
3.	Тема 3. Ядро Node. Глобальні об'єкти. Сервери, потоки вводу-виводу, сокети.	4
4.	Тема 4. Модульна система Node. Зовнішні модулі. Завантаження модулів.	8
5.	Тема 5. Маршрутизація трафіку, службові файли. Маршрутизатори. Проксі-сервери.	4
6.	Тема 6. Платформа Express. Встановлення. Обробка помилок. Маршрутизація.	8
7.	Тема 7. Express, системи шаблонів та CSS. Вбудований JavaScript-код.	8
8.	Тема 8. Отримання структурованих даних в Node та Redis.	4
9.	Тема 9. Node та прив'язка до реляційних баз даних. Робота з db-mysql.	8
10.	Тема 10. Веб-сокети. Модуль Socet-IO.	4
11.	Тема 11. Тестування та відладка Node-додатків. Блочне тестування. Порівняльне тестування.	4
12.	Тема 12. Захист Node-додатків. Шифрування додатків. Протидія атакам.	4

### 8. Самостійна робота

№ з/п	Назва теми	Кількість годин
1.	Тема 1. Встановлення та запуск Node.js. Створення середовища розробки для Node.	10
2.	Тема 2. Інтерактивний режим роботи з Node з використання REPL. Переваги REPL. REPL-команди.	10
3.	Тема 3. Ядро Node. Глобальні об'єкти. Сервери, потоки вводу-виводу, сокети.	10
4.	Тема 4. Модульна система Node. Зовнішні модулі. Завантаження модулів.	12
5.	Тема 5. Маршрутизація трафіку, службові файли. Маршрутизатори. Проксі-сервери.	12
6.	Тема 6. Платформа Express. Встановлення. Обробка помилок. Маршрутизація.	12

7.	Тема 7. Express, системи шаблонів та CSS. Вбудований JavaScript-код.	12
8.	Тема 8. Отримання структурованих даних в Node та Redis.	12
9.	Тема 9. Node та прив'язка до реляційних баз даних. Робота з db-mysql.	12
10.	Тема 10. Веб-сокети. Модуль Socet-IO.	12
11.	Тема 11. Тестування та відладка Node-додатків. Блочне тестування. Порівняльне тестування.	12
12.	Тема 12. Захист Node-додатків. Шифрування додатків. Протидія атакам.	10
	Разом	136

### 9. Методи навчання

Основними видами занять, які проводяться під керівництвом викладача, є лабораторні роботи і самостійна робота.

При виконанні лабораторних робіт зміцнюються знання, отримані на лекціях, набуваються первинні навички роботи з програмування “front-end” та “back-end” додатків. Розглядаються типові засоби та методи роботи з середовищем Node.js.

При самостійній роботі студенти набувають навички самостійного освоєння інструментарію Node.js, які не використані в навчальному процесі та поглиблюються свої знання щодо технологій створення Node-додатків.

### 10. Методи контролю

Під час вивчення дисципліни використовуються наступні методи контролю: поточне тестування за теоретичним матеріалом, захист лабораторних робіт у формі співбесіди.

### 11. Розподіл балів, які отримують студенти

Залік (6 семестр)

Поточне тестування та сам. робота	Підсумковий тест (залік)	Сума
Змістовий модуль 1		
Захист лаб.р.		
80	20	100

### Шкала оцінювання: національна та ЄКТС

Сума балів за всі види навчальної діяльності	Оцінка за національною шкалою	
	для екзамену, курсового проекту (роботи), практики	для заліку
90-100	відмінно	зараховано
82-89	добре	
74-81		
64-73	задовільно	
60-63		
35-59	незадовільно з можливістю повторного складання	не зараховано з можливістю повторного складання
1-34	незадовільно з обов'язковим повторним вивченням дисципліни	не зараховано з обов'язковим повторним вивченням дисципліни

### 13. Методичне забезпечення

1. Внутрішній веб-сайт кафедри комп'ютерної інженерії;
2. Методичні вказівки з лабораторного практикуму (в електронному вигляді);
3. Методичні вказівки для самостійної роботи студентів (в електронному вигляді).

### 14. Рекомендована література

#### Основна

1. Шелли Пауэрс, Изучаем Node.js - Санкт-Петербург: «Питер», 2014 – С.-402.
2. Хэррон Дэвид. Node.js Разработка серверных веб-приложений на javascript. – ДМК Пресс, 2012 – С.-146.

#### Допоміжна

1. Brown Ethan. Web Development with Node and Express PDF, O'Reilly Media, 2013 - 330
2. Cantelon M. at al. Node.js in Action (+ source code) PDF, Manning Publications Co., 2013. — 417 p.
3. Howard D. Node.js for PHP Developers PDF, O'Reilly, 2012. - 269 p.
4. Ihrig C.J. Pro Node.js for Developers EPUB, Apress, 2013. — 350p.



## **15. Інформаційні ресурси**

1. <http://uk.wikipedia.org>.
2. [www.intuit.ru](http://www.intuit.ru) – Національний відкритий університет