



Силабус освітнього компонента

Програма навчальної дисципліни



Web-додатки на основі NodeJS

Шифр та назва спеціальності

121 – Інженерія програмного забезпечення
122 – Комп'ютерні науки

Інститут

ННІ Комп'ютерних наук та інформаційних технологій

Освітня програма

Інженерія програмного забезпечення
Комп'ютерні науки та інтелектуальні системи

Кафедра

Програмна інженерія та інтелектуальні технології управління (321)

Рівень освіти

Бакалавр

Тип дисципліни

Вибіркова

Семестр

6

Мова викладання

Українська, англійська

Викладачі, розробники



Мельник Каріна Володимирівна

Karina.Melnyk@khp.edu.ua

Доктор філософії в технічних науках, к.т.н. доцент, доцент кафедри програмної інженерії та інформаційних технологій управління

Підготувала і опублікувала понад 70 публікацій, 5 колективних монографій, 10 статей у виданнях, індексованих у Scopus і Web of Science. (h-index = 5, i10-index = 1 у Google Академії -

<https://scholar.google.com.ua/citations?user=xCU7GMgAAAAJ&hl=ru> ;

ідентифікатор ORCID <https://orcid.org/0000-0001-9642-5414> ;

ідентифікатор автора Scopus Author

<https://www.scopus.com/authid/detail.uri?authorId=57195074119>).

Провідний лектор з курсів: Основи інженерії програмного забезпечення (бакалаври) (англійською мовою), Методи обробки емпіричної інформації (бакалаври) (англійською та українською мовами), Основи проектування інтелектуальних систем (магістри) (англійською та українською мовами).

[Детальніше про викладача на сайті кафедри](#)

Загальна інформація

Анотація

Дисципліна «Web-додатки на основі NodeJS» є навчальною дисципліною вільного вибору студента профільної підготовки за спеціальностями 121 «Інженерія програмного забезпечення», 122 «Комп'ютерні науки». Навчальна дисципліна спрямована на формування цілісної картини знань по створенню мережевих додатків мовою JavaScript, які можуть бути запущені поза браузером клієнта.

Мета та цілі дисципліни

Мета вивчення навчальної дисципліни є формування у фахівців теоретичних знань і практичних навичок у питаннях проектування та створення серверних додатків мовою Javascript.

Формат занять

Лекції, лабораторні роботи, самостійна робота, консультації. Підсумковий контроль – залік.

Компетентності

121 – Інженерія програмного забезпечення:

K13. Здатність ідентифікувати, класифікувати та формулювати вимоги до програмного забезпечення.

K14. Здатність брати участь у проектуванні програмного забезпечення, включаючи проведення моделювання (формальний опис) його структури, поведінки та процесів функціонування.

K15. Здатність розробляти архітектури, модулі та компоненти програмних систем.

K19. Володіння знаннями про інформаційні моделі даних, здатність створювати програмне забезпечення для зберігання, видобування та опрацювання даних.

K25. Здатність обґрунтовано обирати та освоювати інструментарій з розробки та супроводження програмного забезпечення.

122 – Комп'ютерні науки

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.

СК9. Здатність реалізувати багаторівневу обчислювальну модель на основі архітектури клієнт-сервер, включаючи бази даних, знань і сховища даних, виконувати розподілену обробку великих наборів даних на кластерах стандартних серверів для забезпечення обчислювальних потреб користувачів, у тому числі на хмарних сервісах.

СК13. Здатність до розробки мережевого програмного забезпечення, що функціонує на основі різних топологій структурованих кабельних систем, використовує комп'ютерні системи і мережі передачі даних та аналізує якість роботи комп'ютерних мереж.

СК20. Здатність розробляти архітектуру програмних систем та їх окремих компонент при побудові інтелектуальних систем управління у різних галузях, управляти процесами життєвого циклу програмного забезпечення інтелектуальних систем управління.

Результати навчання

121 – Інженерія програмного забезпечення:

ПР12. Застосовувати на практиці ефективні підходи щодо проектування програмного забезпечення.

ПР14. Застосовувати на практиці інструментальні програмні засоби доменного аналізу, проектування, тестування, візуалізації, вимірювань та документування програмного забезпечення.

ПР15. Мотивовано обирати мови програмування та технології розробки для розв'язання завдань створення і супроводження програмного забезпечення.

ПР18. Знати та вміти застосовувати інформаційні технології обробки, зберігання та передачі даних.

122 – Комп'ютерні науки

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.

ПР10. Використовувати інструментальні засоби розробки клієнт-серверних застосувань, 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.

ПР20. Розробляти архітектуру програмних систем та їх окремих компонент при побудові інтелектуальних систем управління у різних галузях, а також управляти процесами життєвого циклу програмного забезпечення інтелектуальних систем управління.

Обсяг дисципліни

Загальний обсяг дисципліни в обсязі 150 год.(5 кредитів ECTS), зокрема: лекції – 32 год., лабораторні – 32 год., самостійна робота – 86 год. Вона викладається у шостому семестрі. У курсі передбачено два змістових модулі та одна модульна контрольна робота. Завершується дисципліна заліком.

Передумови вивчення дисципліни (пререквізити)

Курс «Web-додатки на основі NodeJS» базується на дисциплінах «Фреймворки JavaScript», «Поглиблений курс веб-розробки».

Особливості дисципліни, методи та технології навчання

Методи викладання та навчання:

інтерактивні лекції з презентаціями, дискусії, лабораторні заняття, командна робота, кейс-метод, метод зворотного зв'язку з боку студентів, проблемне навчання.

Форми оцінювання:

письмові індивідуальні завдання до лабораторних робіт (CAS), оцінювання знань на лабораторних заняттях (CAS), експрес-опитування (CAS), онлайн-тести (CAS), підсумковий/семестровий контроль у формі семестрового заліку, відповідно до графіку навчального процесу (FAS).

Програма навчальної дисципліни

Теми лекційних занять

Тема 1. Серверна розробка.

Що таке серверна розробка. Серверна розробка мовою JavaScript.

Тема 2. Основи Node.js.

Чому ми повинні використовувати Node.js. Як працює Node.js. Цикл подій та шаблон реактора. Як інсталиувати Node.js

Тема 3. Структура програми на Node.js.

Node.js Модулі. Node.js NPM.

Тема 4. Додаткові можливості.

Потоки. EventEmitter. Обробка винятків.

Тема 5. Шаблони проектування. Кращі практики.

Спостерігач. Зворотні дзвінки. Події. Організація коду.

Тема 6 Node.js MySQL

Node.js MySQL створення бази даних. Node.js MySQL CRUD таблиць.

Тема 7 Node.js MongoDB

Node.js MongoDB створення бази даних. Node.js MongoDB CRUD колекцій.

Тема 8. Node.js додатки

Спілкування в режимі реального часу. Потік даних. Панелі моніторингу системи. REST API.

Теми практичних занять

Практичні заняття в рамках дисципліни не передбачені.

Теми лабораторних робіт

Тема 1. Створення простого застосунку Node.js.

Тема 2. Створення веб-додатку Node.js + MySQL/PostgreSQL

Тема 3. Створення веб-додатку Node.js + MongoDB

Тема 4. Створення веб-додатку за обраною темою.

Самостійна робота

Студентам рекомендуються додаткові матеріали (відео, статті) для самостійного вивчення та опрацювання.

Література та навчальні матеріали

1. Lim, Greg. Beginning Node.js, Express & MongoDB Development. Сінгапур, Greg Lim, 2020.
2. Herron D. Node.js Web Development. Third ed. Birmingham: Packt Publishing; 2016. <https://public.ebookcentral.proquest.com/choice/publicfullrecord.aspx?p=4594292>
3. Nandaa A. Beginning Api Development with Node. Js : Build Highly Scalable Developer-Friendly Apis for the Modern Web with Javascript and Node. Js. Birmingham: Packt Publishing; 2018. <https://search.ebscohost.com/login.aspx?direct=true&scope=site&db=nlebk&db=nlabk&AN=1856582>.
4. Hibbard J Reyes C Wanyoike M et al. Node. Js. Sebastopol: SitePoint Pty Limited; 2018. <https://public.ebookcentral.proquest.com/choice/publicfullrecord.aspx?p=5611119>
5. Rappi F. Modern Frontend Development with Node. Js a Compendium for Modern Javascript Web Development Within the Node.js Ecosystem. 1st ed. Birmingham: Packt Publishing Limited; 2022. <https://public.ebookcentral.proquest.com/choice/PublicFullRecord.aspx?p=30254093>.

Система оцінювання

Критерії оцінювання успішності студента та розподіл балів

100% підсумкове оцінювання у вигляді заліку (10%) та поточного оцінювання (90%).

10% залік: семестровий залік, відповідно до графіку навчального процесу

90% поточне оцінювання:

70% оцінювання завдань на лабораторних роботах;

20% проміжний контроль (2 самостійні індивідуальні роботи)

Шкала оцінювання

Сума балів	Національна оцінка	ECTS
90–100	Відмінно	A
82–89	Добре	B
75–81	Добре	C
64–74	Задовільно	D
60–63	Задовільно	E
35–59	Незадовільно (потрібне додаткове вивчення)	FX
1–34	Незадовільно (потрібне повторне вивчення)	F

Норми академічної етики і політика курсу

Студент повинен дотримуватися «Кодексу етики академічних взаємовідносин та доброчесності НТУ «ХПІ»: виявляти дисциплінованість, вихованість, доброзичливість, чесність, відповідальність. Конфліктні ситуації повинні відкрито обговорюватися в навчальних групах з викладачем, а при неможливості вирішення конфлікту – доводитися до відома співробітників дирекції інституту.

Нормативно-правове забезпечення впровадження принципів академічної доброчесності НТУ «ХПІ» розміщено на сайті: <http://blogs.kpi.kharkov.ua/v2/nv/akademichna-dobrochesnist/>

Погодження

Силабус погоджено

08.06.2023

Завідувач кафедри
Ігор ГАМАЮН

08.06.2023

Гаранти ОП
Андрій КОПП
Юлія ЛІТВИНОВА