

Технології розроблення програмного забезпечення

Робоча програма навчальної дисципліни (Силабус)

Реквізити навчальної дисципліни

Рівень вищої освіти *Перший (бакалаврський)*

Галузь знань	<i>12 Інформаційні технології</i>
Спеціальність	<i>126 Інформаційні системи та технології</i>
Освітня програма	<i>Інформаційне забезпечення робототехнічних систем</i>
Статус дисципліни	<i>Нормативна</i>
Форма навчання	<i>очна(денна)</i>
Рік підготовки, семестр	<i>3 курс, весняний семестр</i>
Обсяг дисципліни	
Семестровий контроль/ контрольні заходи	<i>залік</i>
Розклад занять	
Мова викладання	<i>Українська</i>

Інформація про керівника курсу / викладачів	Лектор: <i>к.т.н., доцент, Мелкумян Катерина Юріївна</i> , e.melkumyan@ukr.net Комп'ютерний практикум: асистент, Коваль Олександр Сергійович, t70000s@gmail.com
Розміщення курсу	В електронному кампусі

Програма навчальної дисципліни

1. Опис навчальної дисципліни, її мета, предмет вивчення та результати навчання

Мета викладання навчальної дисципліни є формування теоретичних знань і практичних навичок щодо здатності студента професійно застосовувати сучасні веб-технології та засоби розробки програмного забезпечення при створенні веб-застосунків.

Навчальна дисципліна призначена для отримання студентом:

Знання

- Сучасних технологій організації робочого середовища веб-систем;
- Патерни проектування архітектури веб-додатків;
- Особливості сучасних технологій програмування веб-додатків;
- Веб-технології та стандарти передачі та збереження інформації

Вміння

- Самостійно налаштувати робоче середовище для розробки та функціонування Веб-системи незалежно від операційної системи;
- Проектувати та реалізувати архітектуру веб-системи;
- Проектувати та реалізувати інтерфейс користувача під будь-які задачі та платформу, створювати landing page, використовувати сучасні технології CSS3, HTML5, шаблони дизайну на основі Bootstrap;
- використання каскадних таблиць стилів;
- проектувати та реалізувати серверну частину веб-системи, розробки серверних сценаріїв;
- розміщення веб-застосунків на сервері.

Згідно з вимогами освітньо-професійної програми студенти після засвоєння навчальної дисципліни мають продемонструвати такі результати:

Компетенції

1. Здатність розробляти та управляти проектами
2. Здатність застосовувати стандарти в області інформаційних систем та технологій при розробці функціональних профілів, побудові та інтеграції систем, продуктів, сервісів і елементів інфраструктури

3. Здатність до проектування, розробки, налагодження та вдосконалення системного, комунікаційного та програмно-апаратного забезпечення інформаційних систем та технологій, Інтернету речей (IoT), комп'ютерно-інтегрованих систем та системної мережної структури, управління ними
4. Здатність використовувати сучасні інформаційні системи та технології (виробничі, підтримки прийняття рішень, інтелектуального аналізу даних та інші), методики захисту інформації та кібербезпеки під час виконання функціональних завдань та обов'язків.
5. Здатність застосовувати інформаційні технології у ході створення, впровадження та експлуатації системи менеджменту якості та оцінювати витрати на її розроблення та забезпечення.

Програмні результати навчання

2. Використовувати базові знання інформатики й сучасних інформаційних систем та технологій, навички програмування, технології безпечної роботи в комп'ютерних мережах, методи створення баз даних та інтернет-ресурсів, технології розроблення алгоритмів і комп'ютерних програм мовами високого рівня із застосуванням об'єктно-орієнтованого програмування для розв'язання задач проектування і використання інформаційних систем та технологій
3. Демонструвати знання сучасного рівня технологій інформаційних систем, практичні навички програмування та використання прикладних і спеціалізованих комп'ютерних систем та середовищ з метою їх запровадження у професійній діяльності
4. Аргументувати вибір програмних та технічних засобів для створення інформаційних систем та технологій на основі аналізу їх властивостей, призначення і технічних характеристик з урахуванням вимог до системи і експлуатаційних умов; мати навички налагодження та тестування програмних і технічних засобів інформаційних систем та технологій
5. Обґрунтовувати структури вибір технічної та розробляти відповідне програмне забезпечення, що входить до складу інформаційних систем та технологій
6. Застосовувати правила оформлення проектних матеріалів інформаційних систем та технологій, знати склад та послідовність виконання проектних робіт з урахуванням вимог відповідних нормативно-правових документів для запровадження у професійній діяльності.
7. Здійснювати системний аналіз архітектури підприємства та його IT-інфраструктури, проводити розроблення та вдосконалення її елементної бази і структури.
8. Знати сучасні мови програмування та технології створення програмного забезпечення інформаційних систем та технологій
9. Вміти розробляти програмне забезпечення для задач управління і обробки сенсорної інформації робототехнічних систем

10. Пререквізити та постреквізити дисципліни (місце в структурно-логічній схемі навчання за відповідною освітньою програмою)

Для успішного засвоєння дисципліни «Технології розроблення програмного забезпечення - 2» студенту необхідні наступні навчальні дисципліни: «Бази даних», «Програмування», «Об'єктно-орієнтоване програмування».

На даній навчальній дисципліні базуються дисципліни: «Проектування інформаційних систем», «Управління проектами».

11. Зміст навчальної дисципліни

1. Основи веб-робробки
 - a. Web-програмування як приклад архітектури “клієнт-сервер”
 - b. Open System Interface (OSI) Model
 - c. Етапи Web-розробки
2. Поняття Web-серверу
 - a. Налаштування локального веб-серверу та запуск додатку
 - b. Хмарні платформи та служби
3. Передача даних між клієнтом та сервером
 - a. Формати передачі даних: JSON vs XML
 - b. Структура URL, використання для передачі параметрів
 - c. Використання cookies
4. Поняття BackEnd розробки. Введення у ASP.Net Core
5. Технологія MVC ASP.NET
 - a. Архітектура MVC. Огляд складових.
 - b. Модель MVC ASP.NET з використанням Entity Framework
 - c. Виконання CRUD операцій над моделлю у MVC ASP.NET з використанням Entity Framework
6. Введення у WEB API Application
 - a. RESTful Architecture
 - b. WEB API
 - c. Роутінг API контроллера
7. Поняття контейнеризації
 - a. Загальна архітектура Docker
 - b. Огляд компонент: образ, реєстр, контейнер

12. Навчальні матеріали та ресурси

1. ASP.NET Documentation. Електронний ресурс:
<https://docs.microsoft.com/en-us/aspnet/core/?view=aspnetcore-5.0>
2. Дуглас Крокфорд Как устроен JavaScript — СПб.: Питер, 2019. — 304.
3. Никольский А. П. JavaScript на примерах СПб.: Наука и Техника, 2017. - 272 с.
4. Bootstrap. Електронний ресурс:
<https://www.tutorialrepublic.com/twitter-bootstrap-tutorial/>

13. Методика опанування навчальної дисципліни (освітнього компонента)

Під час вивчення дисципліни передбачені наступні форми та методи навчання і викладання, що сприяють досягнення результатів, закладених в робочу програму: лекції, що передбачають викладення теоретичного матеріалу та узагальнення знань; самостійна робота студента представлена у вигляді оформлених комп'ютерних практикум за допомогою яких формуються вміння, набувається досвід практичного використання конкретних, обраних програмних інструментів для вирішення поставлених задач.

14. Самостійна робота студента

Практика. Частина 1:

1. Ознайомлення та набуття практичних навичок у роботі з технологією MVC ASP.NET.
 - a. Створення моделі.
 - b. Створення контролеру.
 - c. Створення представлення.
2. Створення моделі MVC ASP.NET з використанням Entity Framework.
 - a. Створення нової бази даних за допомогою Entity Framework.
 - b. Створення моделі бази даних.
 - c. Використання даних отриманих з БД.
3. Виконання CRUD операцій над моделлю у MVC ASP.NET з використанням Entity Framework.
 - a. Створення нової сутності за допомогою Entity Framework.
 - b. Оновлення вже існуючої у БД сутності.
 - c. Валідація даних.
 - d. Видалення сутності з БД.
4. Створення додатку MVC ASP.NET з використанням WebApi.
 - a. Підключення вже існуючої бази даних за допомогою Entity Framework.
 - b. Створення прошарку Api.
 - c. Налаштування взаємодії між Api прошарком та Entity Framework.
5. Налаштування роутингу у WebApi MVC ASP.NET та виконання використанням WebApi.
 - a. Налаштування роутингу API контролеру .
 - b. Реалізація CRUD операцій за допомогою WebAPI.

Практика. Частина 2:

1. Робота з Front end, вивчення технології Bootstrap.
 - a. Знайомство з технологією Bootstrap.
 - b. Створення простої Web сторінки за допомогою Bootstrap.
2. Робота з Front end, вивчення технології LESS.
 - a. Знайомство з LESS.

- b. Ознайомлення і видозміна прикладу роботи LESS.
- 3. Робота з Git і deploy проектів за допомогою GitHub.
 - a. Робота з Git.
 - b. Робота з GitHub.
 - c. Deploy проекту за допомогою GitHub.
- 4. Знайомство з мовою програмування TypeScript що широко використовується в Angular.
 - a. Установка компілятора TypeScript та вибір текстового редактору.
 - b. Вивчення необхідної теоретичної інформації.
 - c. Створення простого тестового прикладу на TypeScript.
- 5. Системи для проведення збірки проекту Grunt і Gulp.
 - a. Робота з Grunt.
 - b. Робота з Gulp.
- 6. Знайомство з платформою Angular.
 - a. Установка Angular.
 - b. Автоматичне створення проекту та запуск вбудованого серверу.
 - c. Створення проекту вручну.
- 7. Нескладний приклад додатку на Angular.
 - a. Побудова типової форми для роботи з БД.

Політика та контроль

8. Політика навчальної дисципліни (освітнього компонента)

Вимоги, які виставляються перед студентом:

- відвідування лекційних занять та комп'ютерних практикумів є обов'язковою складовою вивчення матеріалу;
- обов'язковою складовою вивчення матеріалу також є самостійне опрацювання матеріалу, написання модульних контрольних робіт, виконання передбачених комп'ютерних практикумів;
- заохочувальні бали виставляються за: активну участь на лекціях, допомогу в оновленні методичних матеріалів;
- штрафні бали передбачені за несвоєчасну здачу комп'ютерних практикумів між граничними атестаційними контролями.

9. Види контролю та рейтингова система оцінювання результатів навчання (PCO)

Рейтинг студента з дисципліни складається з білав, що він отримує за:

- 1) Виконання комп'ютерних практикумів - 60 балів
- 2) Написання модульних контрольних робіт - 40 балів

Якщо сума балів менша за 60, але модульні контрольні роботи зараховані, студент виконує залікову контрольну роботу. В такому випадку сума балів переводиться до підсумкової оцінки згідно з таблицею нижче:

Кількість балів	Оцінка
100-95	Відмінно
94-85	Дуже добре
84-75	Добре
74-65	Задовільно
64-60	Достатньо
Менше 60	Незадовільно
Не виконані умови допуску	Не допущено

10. Додаткова інформація з дисципліни (освітнього компонента)

Перелік питань на залік

1. Платформа ASP.NET Core, основні переваги її використання, відмінності від попередніх версій.
2. Патерн MVC, визначення та призначення його основних компонентів.
3. Механізм роботи патерну MVC в цілому, порядок виклику компонентів. Переваги цього підходу.
4. Структура проекту ASP.NET Core MVC, призначення його частин.
5. Точка входу у додаток ASP.NET Core, клас Startup та метод Configure.
6. Компонент middleware. Визначення, особливості роботи.
7. Конвеєр обробки запиту. Визначення, особливості роботи.
8. Сервіси MVC. Призначення. Механізм роботи і додавання.
9. Модель у патерні MVC. Особливості побудови, роботи з БД.
10. Механізм прив'язки моделей.
11. Контролери у патерні MVC, дії контролера. Визначення, принцип роботи та виклику.
12. Передача даних у контролер. Атрибути методів контролеру.
13. Представлення у патерні MVC. Визначення, принцип роботи та виклику.
14. Об'єкт ViewResult та метод View. Принцип роботи та виклику.
15. Майстер-сторінки. Визначення та принцип роботи.

16. Двигун представлень Razor. Основні можливості та конструкції програмування що він реалізує.
17. WEB API. Визначення. Механізм роботи.
18. Методи API контролеру. Які є, механізм роботи.

Робочу програму навчальної дисципліни (силабус):

Складено доцент, к.т.н., доцент Мелкумян Катериною Юрївною

Ухвалено кафедрою технічної кібернетики (протокол № 10 від 29.04.2020)

Погоджено Методичною комісією факультету (протокол № 10 від 21.05.2020)