

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
ХАРКІВСЬКИЙ НАЦІОНАЛЬНИЙ ЕКОНОМІЧНИЙ УНІВЕРСИТЕТ
ІМЕНІ СЕМЕНА КУЗНЕЦЯ

ЗАТВЕРДЖЕНО

на засіданні кафедри
інформаційних систем.

Протокол № 8 від 19.01.2024 р.

ПОГОДЖЕНО

Проректор з навчально-методичної роботи

Каріна НЕМАШКАЛО



ПРОГРАМУВАННЯ ІНТЕРНЕТ

робоча програма навчальної дисципліни (РПНД)

Галузь знань
Спеціальність
Освітній рівень
Освітня програма

12 "Інформаційні технології"
121 "Інженерія програмного забезпечення"
перший (бакалаврський)
"Інженерія програмного забезпечення"

Статус дисципліни

Мова викладання, навчання та оцінювання

обов'язкова

українська

Розробник:

к.т.н., с.н.с.

підписано КЕП

Юрій ПАРФЬОНОВ

Завідувач кафедри
інформаційних систем

Дмитро БОНДАРЕНКО

Гарант програми

Олег ФРОЛОВ

Харків
2024

ВСТУП

У теперішній час найбільш розповсюдженими програмними системами є такі, що розроблені з використанням інтернет-технологій. Це потребує від фахівців з інженерії програмного забезпечення чіткого уявлення загальних концепцій інтернет-програмування та використання сучасних засобів розробки серверної частини веб-застосунків.

Навчальна дисципліна "Програмування Інтернет" є обов'язковою навчальною дисципліною, яку вивчають, згідно з навчальним планом підготовки фахівців освітнього ступеню "бакалавр" за спеціальністю 121 "Інженерія програмного забезпечення".

Необхідним елементом успішного засвоєння навчального матеріалу дисципліни є самостійна робота здобувачів із технічною літературою та сучасними середовищами програмування.

Метою навчальної дисципліни "Програмування Інтернет" є формування компетентностей щодо використання сучасних засобів інтернет-програмування та технологій розроблення серверної частини веб-застосунків.

Завданнями навчальної дисципліни є володіння сучасними технологіями інтернет-програмування; удосконалення навичок розроблення серверних веб-застосунків та використання середовищ програмування.

Об'єктом навчальної дисципліни є розроблення інтернет-застосунків.

Предметом навчальної дисципліни є сучасні технології розроблення серверних веб-застосунків.

Результати навчання та компетентності, які формує навчальна дисципліна визначено в табл. 1.

Таблиця 1

Результати навчання та компетентності, які формує навчальна дисципліна

Результати навчання	Компетентності, якими повинен оволодіти здобувач вищої освіти
PH5	СК14
PH7	ЗК2, ЗК5, СК10, СК13
PH8	ЗК5, ЗК6
PH14	СК4
PH15	ЗК2, СК10, СК13
PH16	ЗК7
PH17	ЗК2, ЗК7, СК10, СК12, СК13

де, PH5. Знати і застосовувати відповідні математичні поняття, методи доменного, системного і об'єктно-орієнтованого аналізу та математичного моделювання для розробки програмного забезпечення.

PH7. Знати і застосовувати на практиці фундаментальні концепції, парадигми і основні принципи функціонування мовних, інструментальних і обчислювальних засобів інженерії програмного забезпечення.

PH8. Вміти розробляти людино-машинний інтерфейс.

PH14. Застосовувати на практиці інструментальні програмні засоби доменного аналізу, проектування, тестування, візуалізації, вимірювань та документування програмного забезпечення.

PH15. Мотивовано обирати мови програмування та технології розробки для розв'язання завдань створення і супроводження програмного забезпечення.

PH16. Мати навички командної розробки, погодження, оформлення і випуску всіх видів програмної документації.

PH17. Вміти застосовувати методи компонентної розробки програмного забезпечення.

ЗК02. Здатність застосовувати знання у практичних ситуаціях.

ЗК05. Здатність вчитися і оволодівати сучасними знаннями.

ЗК06. Здатність до пошуку, оброблення та аналізу інформації з різних джерел.

ЗК07. Здатність працювати в команді.

СК04. Здатність формулювати та забезпечувати вимоги щодо якості програмного забезпечення у відповідності з вимогами замовника, технічним завданням та стандартами.

СК10. Здатність накопичувати, обробляти та систематизувати професійні знання щодо створення і супроводження програмного забезпечення та визнання важливості навчання протягом всього життя.

СК12. Здатність здійснювати процес інтеграції системи, застосовувати стандарти і процедури управління змінами для підтримки цілісності, загальної функціональності і надійності програмного забезпечення.

СК13. Здатність обґрунтовано обирати та освоювати інструментарій з розробки та супроводження програмного забезпечення.

СК14. Здатність до алгоритмічного та логічного мислення.

ПРОГРАМА НАВЧАЛЬНОЇ ДИСЦИПЛІНИ

Зміст навчальної дисципліни

Змістовий модуль 1. Використання засобів Інтернет-програмування мовою Python

Тема 1. Вступ до дисципліни

1.1. Розподілені програмні системи. Програмні компоненти. Архітектура розподілених програмних систем. Проміжне програмне забезпечення.

1.2. Підтримка інтернет-протоколів та суміжних технологій у Python.

Тема 2. Використання інтернет-протоколів у Python

2.1. Використання протоколів SMTP, POP3, IMAP тощо. Підтримка інтернет-протоколів у Python.

Тема 3. Використання Web Scraping у Python

3.1. Основи Web Scraping. Бібліотеки для Web Scraping.

Змістовий модуль 2. Основи розроблення серверної частини веб-застосунків на базі фреймворку Django

Тема 4. Основи фреймворку Django

4.1. Веб-застосунки. Шаблон MVC. Веб-фреймворки. Серверні веб-фреймворки.

4.2. Загальні відомості про фреймворк Django. Реалізація шаблону MVT.

4.3. Інсталяція Django. Структура веб-застосунку Django. Маршрутизація HTTP-запитів.

Тема 5. Моделі Django

5.1. Загальні відомості про доступ до баз даних. Об'єктно-реляційне відображення даних. Поняття про Django Object Relational Mapping.

5.2. Вступ до моделей. Клас моделі. Типи полів моделі. Відношення між моделями. Налаштування моделей. Метакласи. Міграції бази даних.

Тема 6. Види та шаблони у Django

6.1. Розроблення видів. Зіставлення URL та видів.

6.2. Структура шаблону. Обмін даними між шаблоном та видом.

6.3. Основи мови шаблонів. Використання тегів у шаблонах. Часткові шаблони. Успадкування шаблонів.

Тема 7. Форми

7.1. Класи форм. Процес опрацювання форм у Django. Поля форми. Валідація. Стандартні форми для моделей.

Тема 8. Види, що базуються на класах

8.1. Загальні відомості. Використання узагальнених видів. Успадкування видів. Використання mixins.

Тема 9. Використання сайту адміністратора

9.1. Загальні відомості про сайт адміністратора. Реєстрація моделей. Створення суперкористувача. Налаштування інтерфейсу сайту адміністратора.

Тема 10. Сесії

10.1. Загальні відомості про сесії. Використання сесій. Збереження даних сесії. Налаштування сесій.

Тема 11. Авторизація та автентифікація

11.1. Система авторизації та автентифікації у Django. Активування автентифікації. Створення користувачів та груп.

Тема 12. Інтернаціоналізація та локалізація

12.1. Інтернаціоналізація в вихідному коді застосунку, в коді шаблону та в URL. Основи локалізації у Django.

Перелік лабораторних занять, а також питань та завдань до самостійної роботи наведено в табл. 2.

Таблиця 2

Перелік лабораторних занять

Назва теми	Зміст
Тема 2. Лабораторне заняття 1	Використання інтернет-протоколів у Python
Тема 3. Лабораторне заняття 2	Використання Web Scraping у Python
Тема 4. Лабораторне заняття 3	Основи Django
Теми 5 - 7. Лабораторне заняття 4	Використання баз даних у веб-застосунках на базі фреймворку Django
Тема 8. Лабораторне заняття 5	Використання видів, що базуються на класах

Тема 9. Лабораторне заняття 6	Використання сайту адміністратора Django
Теми 10, 11. Лабораторне заняття 7	Використання сесій, автентифікації та авторизації у Django
Тема 12. Лабораторне заняття 8	Застосування інтернаціоналізації та локалізації у Django

Перелік самостійної роботи за навчальною дисципліною наведено в табл. 3.

Таблиця 3

Перелік самостійної роботи

Назва теми	Зміст
Теми 1 - 12	Вивчення лекційного матеріалу
Теми 1 - 12	Підготовка до лабораторних занять
Теми 1 - 12	Підготовка до екзамену

Кількість годин лекційних, лабораторних занять та годин самостійної роботи наведено в робочому плані (технологічній карті) з навчальної дисципліни.

МЕТОДИ НАВЧАННЯ

У процесі викладання навчальної дисципліни для набуття визначених результатів навчання, активізації освітнього процесу передбачено застосування таких методів навчання, як:

Словесні (лекція (Теми 1, 3, 4 - 11), проблемні лекції (Теми 2, 12)). Наочні (демонстрація (Теми 1 - 12)).

Практичні (лабораторні заняття, інтерактивне дистанційне навчання, коучинг, презентація, рольові ігри (Теми 1 - 12); кейс-метод (Теми 11, 12)).

ФОРМИ ТА МЕТОДИ ОЦІНЮВАННЯ

Університет використовує 100 бальну накопичувальну систему оцінювання результатів навчання здобувачів вищої освіти.

Поточний контроль здійснюється під час проведення лекційних та лабораторних занять і має на меті перевірку рівня підготовленості здобувача вищої освіти до виконання конкретної роботи і оцінюється сумою набраних балів:

– для дисциплін з формою семестрового контролю екзамен (іспит): максимальна сума – 60 балів; мінімальна сума, що дозволяє здобувачу вищої освіти скласти екзамен (іспит) – 35 балів.

Підсумковий контроль включає семестровий контроль та атестацію здобувача вищої освіти.

Семестровий контроль проводиться у формі семестрового екзамену (іспиту). Складання семестрового екзамену (іспиту) здійснюється під час екзаменаційної сесії.

Максимальна сума балів, яку може отримати здобувач вищої освіти під час екзамену (іспиту) – 40 балів. Мінімальна сума, за якою екзамен (іспит) вважається складеним – 25 балів.

Підсумкова оцінка за навчальною дисципліною визначається сумуванням балів за поточний та підсумковий контроль.

Під час викладання навчальної дисципліни використовуються наступні контрольні заходи:

Поточний контроль: захист лабораторних робіт (50 балів), письмова контрольна робота (10 балів).

Семестровий контроль: Екзамен (40 балів)

Більш детальну інформацію щодо системи оцінювання наведено в робочому плані (технологічній карті) з навчальної дисципліни.

Приклад екзаменаційного білета та критерії оцінювання для навчальної дисципліни.

Приклад екзаменаційного білета

ЕКЗАМЕНАЦІЙНИЙ БІЛЕТ № 1

Завдання 1

Розробіть Web-застосунок на базі фреймворку Django для роботи з моделлю даних Student, об'єкти котрої зберігаються в базі даних SQLite. Клас моделі Student має явно визначені атрибути Прізвище, Факультет, Курс.

Web-застосунок має містити:

1. HTML-сторінку з формою Django, що дозволяє в її окремих полях ввести прізвище, факультет та курс студента та відправити ці дані на Function-based View (FBV).
2. FBV, який за допомогою розробленого сервісного класу має записувати введені дані до бази даних та зчитувати їх з бази даних шляхом виклику відповідних методів сервісного класу, а також в HTTP-відповіді сервера - передавати зчитані дані на ще одну HTML-сторінку для відображення (кожне значення має знаходитися в окремому рядку).
3. Сервісний клас, що містить методи для додавання даних до бази даних та зчитування даних з бази даних з використанням Django ORM API.

Завдання 2

Створіть суперкористувача для доступу на сайт адміністратора веб-застосунку з Завдання 1. Зареєструйте модель даних Student цього застосунку для використання її сумісно з сайтом адміністратора. Виконайте програмне налаштування веб-інтерфейсу сайту адміністратора для забезпечення відображення власного заголовку веб-сторінок замість «Django administration» та фільтрації записів, що відображуються в списку, за прізвищем та факультетом.

Критерії оцінювання

Екзаменаційний білет містить два комплексних евристичних завдання. Результатам виконання кожного з них присвоюється рейтинг від 0 до 6 відповідно до наступної шкали:

6	Завдання виконано в повному обсязі. Програма працює правильно. Вихідний код програми задовольняє вимогам, вказаним в завданні.
5	Завдання виконано. Програма працює правильно, але вона розроблена з порушенням однієї з вимог, вказаних в завданні.
4	Завдання головним чином виконано. Програма працює, але вона розроблена з порушенням двох з вимог, вказаних в завданні.
3	Завдання виконано з суттєвими недоліками. Програма працює, але не виконано одну з вимог, вказаних в завданні, або три з таких вимог реалізовано з порушеннями
2	Завдання практично не виконано. Програма запускається, але не виконано дві з вимог, вказаних в завданні, або чотири з таких вимог реалізовано з порушеннями
1	Завдання не виконано. Програма запускається та частково відповідає постановці завдання, але не виконано більше двох з вимог, вказаних в завданні, або більш чотирьох з таких вимог реалізовано з порушенням. Чи програма не запускається або завершується аварійно, але є програмний код, розроблений студентом, який відповідає постановці завдання.
0	Завдання не виконано. Програма відсутня або програма не містить вихідного коду, розробленого студентом, або програма не відповідає постановці завдання або програма має явні технічні ознаки несамостійності її розробки.

Невиконання або суттєве порушення будь-якої з загальних вимог до програми знижує рейтинг результатів виконання кожного такого завдання на 1 (один).

Загальні вимоги до програм

1. Інтерфейс користувача має відповідати постановці завдання та містити текстові мітки (повідомлення) українською (англійською) мовою, які відносяться до введення та виведення основних та допоміжних даних, їхньої валідації тощо. Використання транслітерації заборонено.

2. Дотримання вимог PEP 8 - Style Guide for Python Code (<https://www.python.org/dev/peps/pep-0008/>).

3. Вихідний код кожного з класів програми мовою Python має знаходитися в окремому файлі.

Завдання 1 екзаменаційного білету є обов'язковим, за його виконання можна отримати до 90% від максимальної оцінки. Завдання 2 екзаменаційного білету є додатковим, за його виконання можна отримати до 10% від максимальної оцінки. Максимальна оцінка за 40-бальною шкалою за виконання кожного з завдань визначається таким чином:

Рейтинг результатів виконання завдання	Оцінка за виконання Завдання 1	Оцінка за виконання Завдання 2
6	36	4
5	33 - 35	3
4	30 - 32	2
3	26 - 29	1
2	20 - 25	0
1	1 - 19	
0	0	

РЕКОМЕНДОВАНА ЛІТЕРАТУРА

Основна

1. Mele A. Django by Example / A. Mele. - Packt Publishing Limited, Birmingham 2020 - 568 p.
2. Vincent W. Django for Beginners: Build websites with Python and Django / W. Vincent. - WelcomeToCode, 2023 - 349 p.

Додаткова

3. Feldroy D. Two Scoops of Django 3.X: Best Practices for Django / D. Feldroy. - Two Scoops Press , 2020 - 477 p.

Інформаційні ресурси

4. The Python Standard Library [Electronic resource]. – Access mode : <https://docs.python.org/3/library/index.html>
5. Interactive Python tutorial [Electronic resource]. – Access mode: <https://www.learnpython.org/>
6. Django documentation [Electronic resource]. – Access mode : <https://docs.djangoproject.com/en/5.0/>
7. Django forum [Electronic resource]. – Access mode : <https://forum.djangoproject.com/>
8. DjangoSites: Shiny websites, powered by Django [Electronic resource]. – Access mode : <https://djangosites.org/>
9. Django Tutorial: The Local Library website [Electronic resource]. – Access mode: https://developer.mozilla.org/en/docs/Learn/Server-side/Django/Tutorial_local_library_website
10. Django Admin Cookbook [Electronic resource]. – Access mode : <https://books.agiliq.com/projects/django-admin-cookbook/en/latest/>
11. Django Tutorial [Electronic resource]. – Access mode : <https://www.w3schools.com/django/>
12. Персональна навчальна система "Програмування Інтернет" [Електронний ресурс]. - Режим доступу: <https://pns.hneu.edu.ua/course/view.php?id=4650>
13. Парфьонов Ю. Е., Колгатін О. Г. Вибір служби вебхостингу для застосунків на базі фреймворку Django / Ю. Е. Парфьонов, О. Г. Колгатін // Вісник Харківського національного автомобільно-дорожнього університету: збірник наукових праць. - Харків: ХНАДУ, 2022. – Вип. 96. – С. 66 – 70. – Режим доступу: <http://repository.hneu.edu.ua/handle/123456789/27384>