

СЛАБУС НАВЧАЛЬНОЇ ДИСЦИПЛІНИ

«Веб-застосунки з Java/Spring»



Ступінь освіти	Магістр
Галузь знань	12 Інформаційні технології
Спеціальність	всі спеціальності галузі
Тривалість викладання	1 семестр
Заняття:	весняний семестр (3 та 4 чверть)
Лекції	2 год/тижд
Практичні заняття	1 год/тижд
Мова викладання	українська

Передумови для вивчення: мати базові навички програмування в середовищі Java.

Сторінка курсу в СДО НТУ «ДП»: <https://do.nmu.org.ua/course/view.php?id=3574>

Консультації: за окремим розкладом, що попередньо погоджений зі здобувачами освіти.

Онлайн-консультації: MS Teams, електронна пошта, Telegram

Інформація про викладача:



Викладач:

Мінєєв Олександр Сергійович

к.т.н, доцент каф. САУ

Посилання на профілі:

[Сторінка кафедри САУ](#)

Scopus ID: [57209981391](#)

ORCID ID: [0000-0002-8814-1355](#)

[Google Scholar](#)

1. Анотація курсу

Java та Spring мають ключове значення для сучасної розробки веб-застосунків завдяки своїй гнучкості, надійності та масштабованості. Java — це одна із найпопулярніших мов програмування, яка забезпечує високу продуктивність і кросплатформеність. Spring, як фреймворк, спрощує створення складних додатків, забезпечуючи інструменти для швидкої розробки, інтеграції з базами даних і захисту систем.

Курс "Веб-застосунки на Java/Spring" призначений для розробників, які бажають поглибити свої знання у сфері створення сучасних веб-додатків з використанням одного з найпопулярніших Java-фреймворків — Spring. Цей курс підходить як для новачків, так і для тих, хто вже має базовий досвід роботи з Java.

Що ви дізнаєтеся на курсі:

1. **Основи Spring Framework:** розуміння архітектури Spring, включаючи модулі Spring Core, Spring MVC, Spring Boot та інші.
2. **Створення RESTful API:** навчитися створювати API з використанням Spring MVC, обробляти HTTP-запити та відповідати на них JSON-даними.
3. **Робота з базами даних:** інтеграція баз даних за допомогою Spring Data JPA, налаштування підключень та виконання CRUD-операцій.
4. **Тестування та налагодження:** використання інструментів для тестування Java-програм, таких як JUnit та Mockito, а також основні принципи дебагінгу.
5. **Робота з шаблонами:** інтеграція таких шаблонізаторів, як Thymeleaf, для створення динамічних веб-сторінок.

2. Мета та завдання навчальної дисципліни

Мета дисципліни – навчити здобувачів розробляти сучасні, масштабовані та безпечні веб-додатки. Курс охоплює основи роботи з Java та Spring Framework, забезпечуючи знання про створення RESTful API, роботу з базами даних, впровадження безпеки та тестування. Студенти набудуть практичних навичок роботи з ключовими компонентами Spring, що дозволить їм будувати стабільні та ефективні веб-застосунки, готові до реальних проектів та викликів у сфері розробки програмного забезпечення.

3. Результати навчання

Після вивчення дисципліни студенти мають досягти таких результатів навчання:

- розуміти структуру сучасних веб-додатків, зокрема клієнтсько-серверну архітектуру, принципи REST і MVC, а також роль фреймворку Spring у розробці Java-додатків;
- налаштовувати проекти Spring Boot, створювати контролери, сервіси та репозиторії, а також застосовувати принципи dependency injection;
- створювати та документувати REST API, які підтримують CRUD-операції, і використовувати ці API для взаємодії між різними частинами додатку або з зовнішніми сервісами;
- вміти інтегрувати застосунок з базою даних за допомогою Spring Data JPA, виконувати запити, зберігати та отримувати дані, а також розуміти принципи транзакцій та ORM;
- застосовувати шаблони проєктування (наприклад, Singleton, Factory, DAO) у веб-додатках на Java, що сприятиме створенню структурованого, підтримуваного й масштабованого коду;
- розгорнути веб-додатки, створені з використанням Spring Boot, на різних серверах та хмарних платформах (наприклад, Heroku, AWS або Docker), а також зможуть використовувати інструменти CI/CD для автоматизації процесу розгортання та оновлення застосунків.

4. Структура курсу

Види та тематика навчальних занять	Внесок в загальну оцінку, %
ЛЕКЦІЇ	40
1. Короткий екскурс в JavaCore. Згадати все. Що таке JVM. JDK. JRE. JVM memory model. Maven. ООП в Java. Collections. Junit. SQL. JDBC. Json. Xml	10
2. Apache POI. Створення Знайомство POI. Створення та читання EXcell та Word files.	8
3. Jsoup. Розробка бота-парсера Тестування та створення ботів. Мій перший бот. Особливості. Пришвидшення. Робота с HTML. Вивчення XPath	10
4.HTTP.REST. Web Server Розуміння Веб-Сервера. Архітектура Клієнт-Серверу. Модель OSI. Основні методи HTTP(s). .	8
5. SpringMVC.Dependency injection Inversion of control. Сімейство Spring. Навіщо? Чи можна інакше?	8
6. Spring Boot. Веб сервер за 5 хвилин. Безпека. Базові анотації. Головні недоліки.	8
7. Spring data. Object Relationship Model. Особливості. Прив'язка до СУБД. Чи це смерть SQL?	8
8. Messaging Мікросервісна архітектура.Кілька слів про ZooKeeper та Apache Kafka.	8
9. Optimization. GOF patterns. SOLID. Шаблони проектування. Найпоширеніші проблеми.	8
ПРАКТИЧНІ ЗАНЯТТЯ	60
Практична частина проекту № 1. Hello world. Мета: закріпити теоретичні знання і розвинути практичні навички зі створення додатків на java.	6
Практична частина проекту № 2 ADD save/read data to excel file. Мета: закріпити теоретичні знання і розвинути практичні навички з використання Apache POI.	4
Практична частина проекту № 3 My first Spring App. Мета: закріпити теоретичні знання і розвинути практичні навички із використання фреймворку SPRING.	6
Практична частина проекту № 4 My first Spring boot App Мета: закріпити теоретичні знання і розвинути практичні навички із використання фреймворку SPRING-BOOT.	6
Практична частина проекту № 5 Selenium bot . Мета: закріпити теоретичні знання і розвинути практичні навички з використання Selenium.	4

Практична частина проекту № 6. Spring data and MySQL ORM Мета: закріпити теоретичні знання і розвинути практичні навички зі використання Hibernate.	6
Практична частина проекту № 7. Testing.Finishing Deploying. Мета: закріпити теоретичні знання і розвинути практичні навички з кінцевим тестуванням перевіркою та запуском серверу.	6
РАЗОМ	100

5. Технічне обладнання та/або програмне забезпечення

Технічні засоби навчання: мультимедійні та комп'ютерні пристрої.

Засоби дистанційної освіти: Moodle, MS Teams, Telegram.

Пакети приладних програм: Inteliji IDEA ULTIMATE (безкоштовна для навчання, після реєстрації з пошти університету), Git (безкоштовний), JDK (безкоштовний).

6. Система оцінювання та вимоги

6.1. Навчальні досягнення здобувачів вищої освіти за результатами вивчення курсу оцінюватимуться за шкалою, що наведена нижче:

Рейтингова шкала	Інституційна шкала
90 – 100	відмінно
75-89	добре
60-74	задовільно
0-59	незадовільно

6.2. Здобувач вищої освіти може отримати підсумкову оцінку з навчальної дисципліни на підставі поточного оцінювання знань за умови, якщо набрана кількість балів з поточного тестування та самостійної роботи складатиме не менше 60 балів.

Поточна успішність складається з успішності за теоретичну частину курсу (максимум – 30 балів) та оцінок за виконання практичних робіт (максимум 10 балів за кожну роботу та максимальною сумарною оцінкою за всі роботи – 70 балів). Отримані бали за теоретичну частину курсу та практичні роботи додаються і є підсумковою оцінкою за вивчення навчальної дисципліни. Максимально за поточною успішністю здобувач вищої освіти може набрати 100 балів.

Шкала оцінювання (зазначено максимально можливі бали):

Теоретична частина	Практичні роботи		Разом
	При своєчасному складанні	При несвоєчасному складанні	
30	70	45	100

6.3 Критерії оцінювання поточного та підсумкового контролю:

- підсумкове оцінювання відбувається у формі диференційованого заліку у форматі тесту, який містить запитання запитання із вибором відповіді і відкриті запитання у вигляді задач;

- поточне оцінювання практичних робіт відбувається шляхом захисту звіту з індивідуальної роботи (максимальний бал – 10, який формується наступним чином: 50 %

правильність і повнота викладення матеріалу в звіті, 50 % – захист індивідуальної роботи шляхом відповіді на контрольні питання).

7. Політика курсу

7.1. Політика щодо академічної доброчесності. Академічна доброчесність студентів є важливою умовою для опанування результатами навчання за дисципліною і отримання задовільної оцінки з поточного та підсумкового контролів. Академічна доброчесність базується на засудженні практик списування (виконання письмових робіт із залученням зовнішніх джерел інформації, крім дозволених для використання), плагіату (відтворення опублікованих текстів інших авторів без зазначення авторства), фабрикації (вигадування даних чи фактів, що використовуються в освітньому процесі). У НТУ «Дніпровська політехніка» політика щодо академічної доброчесності регламентується положенням "Положення про систему запобігання та виявлення плагіату у Національному технічному університеті "Дніпровська політехніка": http://www.nmu.org.ua/ua/content/activity/us_documents/System_of_prevention_and_detection_of_plagiarism.pdf.

У разі порушення студентом академічної доброчесності (списування, плагіат, фабрикація), робота оцінюється незадовільно та має бути виконана повторно. При цьому викладач залишає за собою право змінити тему завдання.

7.2. Комунікаційна політика. Студенти повинні мати активовану університетську (корпоративну на домені @nmu.one) пошту. Усі письмові запитання до викладачів стосовно курсу мають надсилатися на університетську електронну пошту.

7.3. Політика щодо перескладання. Роботи, які здаються із порушенням термінів без поважних причин оцінюються на нижчу оцінку. Перескладання відбувається із дозволу деканату за наявності поважних причин (наприклад, лікарняний).

7.4. Відвідування занять. Для студентів денної форми відвідування занять є обов'язковим. Поважними причинами для неявки на заняття є хвороба, участь в університетських заходах, відрядження, які необхідно підтверджувати документами у разі тривалої (два тижні) відсутності. Про відсутність на занятті та причини відсутності студент має повідомити викладача або особисто, або через старосту. Якщо студент захворів, ми рекомендуємо залишатися вдома і навчатися за допомогою дистанційної платформи. Студентам, чий стан здоров'я є незадовільним і може вплинути на здоров'я інших студентів, буде пропонуватися залишити заняття (така відсутність вважатиметься пропуском з причини хвороби). Практичні заняття не проводяться повторно, ці оцінки неможливо отримати під час консультації. **За об'єктивних причин (наприклад, міжнародна мобільність) навчання може відбуватись в он-лайн формі за погодженням з керівником курсу.**

7.5. Участь в анкетуванні. Наприкінці вивчення курсу та перед початком сесії студентам буде запропоновано анонімно заповнити електронні анкети (MS Office 365), які буде розіслано на ваші університетські поштові скриньки. Заповнення анкет є важливою складовою вашої навчальної активності, що дозволить оцінити дієвість застосованих методів викладання та врахувати ваші пропозиції стосовно покращення змісту навчальної дисципліни.

8. Рекомендовані джерела інформації:

Базові:

1. Walls, Craig. *Spring in Action*. 5th ed., Manning Publications, 2018.
2. Bauer, Christian, King, Gavin, Gregory, Gary. *Java Persistence with Hibernate*. 2nd ed., Manning Publications, 2015.
3. Walls, Craig. *Spring Boot in Action*. Manning Publications, 2016.
4. Шабалин, Вячеслав. *Spring 5 для професіоналів. Розробка Java-додатків у корпоративному середовищі*. Видавництво "К.С.Д.", 2020.
5. Флерко, Олександр. *Java для початківців. Підручник для самонавчання*. Видавництво "Фактор", 2019.

Додаткові:

1. Richardson, Leonard, Ruby, Sam. *RESTful Web Services*. O'Reilly Media, 2007.
2. Bloch, Joshua. *Effective Java*. 3rd ed., Addison-Wesley, 2018.
3. Carnell, John. *Spring Microservices in Action*. Manning Publications, 2017.
4. Кевін, Браун. *Веб-програмування на Java: від теорії до практики*. Видавництво "Мандрівець", 2020
5. Сковронський, Ігор. *Java та Spring для професіоналів*. Видавництво "ІнфоБіз", 2021