



Технології розробки web-орієнтованих систем

Робоча програма навчальної дисципліни (Силабус)

Реквізити навчальної дисципліни

Рівень вищої освіти	<i>Перший (бакалаврський)</i>
Галузь знань	<i>G Інженерія, виробництво та будівництво</i>
Спеціальність	<i>G5 Електроніка, електронні комунікації, приладобудування та радіотехніка</i>
Освітня програма	<i>Інформаційно-комунікаційні технології</i>
Статус дисципліни	<i>Вибіркова</i>
Форма навчання	<i>очна(денна)/заочна</i>
Рік підготовки, семестр	<i>4 курс, осінній семестр</i>
Обсяг дисципліни	<i>4 кредити</i>
Семестровий контроль/ контрольні заходи	<i>залік</i>
Розклад занять	
Мова викладання	<i>Українська</i>
Інформація про керівника курсу / викладачів	Лектор: доцент каф. Суліма С.В., +380934310943, itssulima@gmail.com Практичні доцент каф. Суліма С.В.
Розміщення курсу	https://classroom.google.com/c/ODc2NDgyMDk3ODY2?cjc=tiyrowir

Програма навчальної дисципліни

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

Зазначена дисципліна включена до циклу професійної та практичної підготовки. У структурно-логічній схемі навчання дисципліна розміщена у 7-му семестрі.

Вивчення дисципліни базується на знанні студентами матеріалів, передбачуваних учбовими планами дисциплін:

- інформатика;
- спец. розділи математики;
- операційні системи;
- бази даних;
- технології програмування та інші.

1.1. Мета дисципліни

Метою вивчення дисципліни є формування у студентів здатностей:

- планувати проект по розробці програмного забезпечення (web-додатку);
- аналізувати ризики, які виникають при розробці та функціонуванні програмного забезпечення (web-додатку);
- організовувати процес розробки програмного забезпечення відповідно до обраної методології.

1.2. Предмет навчальної дисципліни

Навчальна дисципліна «Технології побудови web-орієнтованих систем» відноситься до дисциплін спеціальної підготовки студентів освітньої програми «Інформаційно-комунікаційні технології» та розкриває основні етапи та підходи до створення програмних web-додатків, принципи їхнього функціонування, та питання взаємодії між їх розробниками. Матеріал, що викладається, дозволяє отримати теоретичні основи та відпрацювати на практиці принципи планування процесу розробки та ефективної організації роботи щодо створення програмних додатків. При цьому в навчальній дисципліні розглядається широкий спектр аспектів створення та функціонування web-додатків із застосуванням сучасних інформаційних технологій.

1.3 Основні завдання дисципліни

Згідно з вимогами програми навчальної дисципліни студенти після її засвоєння мають продемонструвати такі результати навчання:

знання:

- про етапи та принципи планування розробки програмного забезпечення (web-додатків);
- про проектні ризики та ризики функціонування програмного забезпечення (web-додатків);
- про принципи організації роботи розробника програмного забезпечення (web-додатків);
- про технології та програмне забезпечення, які використовуються на кожному етапі розробки програмного забезпечення (web-додатків);
- про проблеми взаємодії користувача з web-додатками та принципи їх вирішення;
- про принципи побудови та тестування сучасних web-додатків;

уміння:

- виконувати вибір апаратного і програмного забезпечення для рішення конкретних прикладних задач;
- розробляти і тестувати прототипи web-додатків;
- організовувати роботу по розробці програмного забезпечення для вирішення прикладних задач згідно з підходом Agile;
- використовувати системи контролю версій.

досвід:

- планування розробки сучасного web-додатку;
- розробки web-додатку з використанням систем контролю версій;
- командної розробки web-додатку.

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

Дисципліна базується на знаннях спеціальних розділів математики-1. Дискретної математики; Інформатики, програмуванні та інші.

3. Зміст навчальної дисципліни

Назви розділів і тем	Кількість годин			
	Всього	у тому числі		
		Лекції	Практичні	СРС
1	2	3	4	6
Тема 1. Історія розвитку інформаційних систем.	7	4	1	2
Тема 2. Основи організації роботи розробника програмного забезпечення.	58	18	3	38
Тема 3. Методології розробки програмного забезпечення.	39	10	7	22
Тема 4. Забезпечення успішної розробки та вимог до програмного забезпечення.	15	4	7	4
Разом за розділом 1	120	36	18	66
Всього годин	120			

4. Навчальні матеріали та ресурси

4.1. Базова література

1. Електронний конспект лекцій.

4.2. Додаткова література

1. Sutherland J. Scrum. The Art of Doing Twice the Work in Half the Time / Jeffrey Sutherland., 2015.
2. Rigby D. Doing Agile Right: Transformation Without Chaos / D. Rigby, S. Elk, S. Berez., 2020.
3. Schwaber K. The Scrum Guide / K. Schwaber, J. Sutherland., 2017.
4. Derby E. Agile Retrospectives: Making Good Teams Great 1st Edition / E. Derby, D. Larsen., 2006.

4.3. Інформаційні ресурси

1. GIT Documentation [Електронний ресурс] – Режим доступу до ресурсу: <https://git-scm.com/doc>.
2. How to Write a Software Requirements Specification (SRS Document) [Електронний ресурс]. – 2021. – Режим доступу до ресурсу: <https://www.perforce.com/blog/alm/how-write-software-requirements-specification-srs-document>.

Навчальний контент

5. Методика опанування навчальної дисципліни (освітнього компонента)

5.1 Лекційні заняття

№ з/п	Назва теми лекції та перелік основних питань
1	Огляд курсу. Базові технології фронтенду та бекенду. <u>Основні питання:</u> 1. Ключові технології та поняття 2. HTML: ключова фронтенд-технологія

	3. CSS: стилізація 4. DOM: структура веб-сторінки 5. JavaScript 6. Основні фреймворки та бібліотеки JavaScript 7. Більше практик, які використовуються у фронтенд-розробці 8. Бекенд. Програмування серверної частини 9. Дії у серверній частині
2	Системи контролю версій. <u>Основні питання:</u> 1. Призначення систем контролю версій. 2. Функції та сфери застосування систем контролю версій. 3. Типи систем контролю версій. 4. Базова термінологія, яка використовується в системах контролю версій. 5. Розгалуження, типи гілок та їх призначення. 6. Принципи використання систем контролю версій для розробки програмного забезпечення. 7. Нумерація версій програмного забезпечення 8. Налаштування git 9. Основи git 10. Розгалуження в git 11. Вказівники в git 12. Внесення виправлень 13. Тегування 14. Git у Visual Studio 15. GitHub - Створення та налаштування облікового запису 16. Додаткові функції Git сервісів
3	Управління проєктами. Системи управління проєктами. <u>Основні питання:</u> 1. Сутність та поняття проєкту 2. Управління проєктами 3. Життєвий цикл проєкту 4. 10 принципів ефективного управління проєктами 5. Життєвий цикл програмного забезпечення 6. Моделі життєвого циклу ПЗ 7. Призначення систем управління проєктами. 8. Особливості підходів замовника та розробника до розуміння проєктів. 9. Типи систем управління проєктами. 10. Базова термінологія, яка використовується в системах управління проєктами. 11. Життєвий цикл задачі, та стани в яких вона може перебувати. 12. Особливості інтерфейсу робочого місця в залежності від ролі. 13. Приклади інструментів управління проєктами
4	Командна розробка програмного забезпечення. Методології розробки програмного забезпечення. SCRUM та Kanban. <u>Основні питання:</u> 1. Основні поняття 2. Основні ролі у команді розробників 3. Ролі у координації команди: Project Manager і Scrum Master

	4. Ролі в бізнес-менеджменті: Product Manager, Product Owner і Business Analyst 5. Стратегії керівництва 6. Стадії формування ефективності у команді 7. Мотивація 8. Історія розвитку підходів до розробки програмного забезпечення. 9. Прогнозовані та адаптивні методології. 10. Моделі процесів розробки програмного забезпечення. 11. Гнучка методологія розробки програмного забезпечення. Основні ідеї, основні принципи успішності, основні принципи. 12. Методики в рамках методології Agile. 13. Засади Scrum. 14. Ролі в команді при використанні Scrum. 15. Базові поняття Scrum. 16. Основні дії при використанні Scrum. 17. Методологія Kanban
5	Принципи об'єктно-орієнтованого програмування і проектування SOLID. <u>Основні питання:</u> 1. S: Принцип єдиної відповідальності (Single Responsibility Principle). 2. O: Принцип відкритості-закритості (Open-Closed Principle). 3. L: Принцип підстановки Лісков (Liskov Substitution Principle). 4. I: Принцип розділення інтерфейсу (Interface Segregation Principle). 5. D: Принцип інверсії залежностей (Dependency Inversion Principle).
6	Основи патернів проектування. <u>Основні питання:</u> 1. Що дає застосування патернів? 2. Патерни, що породжують 3. Структурні патерни 4. Поведінкові патерни 5. Патерни класів 6. Патерни об'єктів 7. Як вибрати потрібний патерн
7	Frontend технології: React фреймворк. <u>Основні питання:</u> 1. Вступ до React 2. Компоненти та JSX 3. Props та State 4. Event Handling 5. Практичний приклад
8	Контейнеризація та Docker. <u>Основні питання:</u> 1. Вступ та мотивація 2. Теоретичні основи контейнеризації 3. Знайомство з Docker 4. Практична робота з Docker 5. Docker Compose та оркестрація
9	Хмарні технології та розгортання web-додатків <u>Основні питання:</u> 1. Cloud Computing 2. IaaS, PaaS, SaaS 3. Основні хмарні платформи

	4. Контейнеризація у хмарі 5. Kubernetes: основні поняття 6. Serverless Functions 7. Автоматичне масштабування застосунків 8. Моніторинг і логування 9. Практичні сценарії деплою сучасних web-додатків
10	DevOps інтеграція: CI/CD pipeline. <u>Основні питання:</u> 1. Вступ та мотивація 2. Основні поняття CI/CD 3. Компоненти CI/CD pipeline 4. Інструменти та технології 5. Практичні приклади 6. Найкращі практики
11	Розгортання та супровід web-орієнтованих систем на платформі Microsoft Azure. <u>Основні питання:</u> 1. Хмарні обчислення та платформа Microsoft Azure 2. Архітектура web-застосунків у Microsoft Azure 3. Розгортання web-додатків за допомогою Azure App Service 4. Хмарні бази даних та сховища даних 5. Автентифікація та захист web-додатків 6. Автоматизація розгортання та DevOps 7. Моніторинг та масштабування web-застосунків 8. Безсерверні технології Microsoft Azure 9. Практичний приклад побудови web-системи на Microsoft Azure
12	Аналіз ризиків. Ризики в сфері інформаційної безпеки. <u>Основні питання:</u> 1. Поняття ризику. Ризики та проектна діяльність. Причини ризиків. 2. Базові поняття управління ризиками. 3. Базовий цикл управління ризиками. 4. Поняття загрози та вразливості інформаційної системи. 5. Рівні зрілості керування ризиками в сфері інформаційної безпеки. 6. Життєвий цикл забезпечення безпеки інформаційної системи. 7. Модель багаторівневого захисту.
13	Тестування програмного забезпечення. <u>Основні питання:</u> 1. Якість ПЗ. Забезпечення якості ПЗ 2. Цілі і завдання процесу тестування 3. Класифікація видів тестування 4. Вимоги до програмного забезпечення 5. Базові поняття, фази і артефакти процесу тестування 6. Тест дизайн та його техніки 7. Баг/Дефект Репорти. Системи баг трекінгу. Градація серйозності та пріоритетності. 8. Класифікація видів тестування 9. Автоматизація тестування

5.2. Практичні заняття (комп'ютерний практикум)

Комп'ютерний практикум №1 Розробка простого веб-сайту зі стилізацією. HTML. CSS.

Комп'ютерний практикум №2 Розробка веб-сайту з базою даних. RESTful API

Комп'ютерний практикум №3 Тестування програмного забезпечення

6. Самостійна робота студента

Зазначаються види самостійної роботи (підготовка до аудиторних занять, проведення розрахунків за первинними даними, отриманими на лабораторних заняттях, розв'язок задач, написання реферату, виконання розрахункової роботи, виконання домашньої контрольної роботи тощо) та терміни часу, які на це відводяться.

№ з/п	Назва теми, що виноситься на самостійне опрацювання	Кількість годин СРС
1	Історія розвитку інформаційних систем.	2
2	Основи організації роботи розробника програмного забезпечення.	38
3	Методології розробки програмного забезпечення.	22
4	Забезпечення успішної розробки та вимог до програмного забезпечення.	4

7. Індивідуальні завдання

Індивідуальні завдання робочим навчальним планом не передбачені.

8. Контрольні роботи

У відповідності з навчальним планом в кредитному модулі передбачено модульну контрольну роботу (МКР).

Ціль МКР – перевірка ступеня засвоєння студентами навчального матеріалу, викладеного на лекціях, а також, перевірка практичних навиків, яких вони набувають на практичних заняттях.

МКР проводиться по мірі засвоєння студентами навчального матеріалу.

Результати виконання МКР враховуються при рейтинговій оцінці успішності студентів, а також при семестровому контролі.

МКР присвячена створенню веб-додатку, що включає в себе:

- презентація ідеї проекту
- створення документації
- безпосередньо готовий проект
- презентація готового проекту
- використання системи контролю версій при роботі над проектом
- використання систем управління проектом (jira, trello)

Політика та контроль

9. Політика навчальної дисципліни (освітнього компонента)

1. Практичні заняття

За доповідь на теоретичне питання до практичного заняття:

- бал за вчасну та правильну відповідь на питання – 10;
- бал за неточну відповідь на питання – 8;
- бал за неповну відповідь на питання – 6.

Максимальна кількість балів: 20.

2. Комп'ютерний практикум

Оцінюється рівень виконання, використані технології. За виконання комп'ютерного практикуму:

- бал за вчасну та правильну відповідь – 10;
- бал за неточну відповідь – 8;
- бал за неповну відповідь – 6.

Максимальна кількість балів: 30.

3. Модульна контрольна робота

Ваговий бал – 50:

- бал за вчасну та правильну відповідь на питання – 50;
- бал за неточну відповідь на питання – 30;
- бал за неповну відповідь на питання – 10.

Максимальна кількість балів: 50.

4. Критерії залікового оцінювання

Ваговий бал – 20, у разі повних відповідей на теоретичні запитання та вірно зробленого практичного завдання.

Бали за білет формуються наступним чином:

1. Теоретичні питання 3 шт.:

- правильна відповідь на питання – 5;
- відповідь та питання зі незначними помилками – 4;
- неповна відповідь на питання – 3;
- відповідь та питання зі значними помилками – 2;
- невірна відповідь, яка містить вірні міркування студента – 1;
- немає відповіді на питання – 0.

2. Практичне завдання:

- правильно виконане практичне завдання – 5;
- практичне завдання виконане з помилками – від 1 до 4;
- не виконане практичне завдання – 0.

Формула для розрахунку балів за екзамен:

$$\text{Бали} = \sum_{i=1,4} (\text{бал за питання})_i, \text{ де } (\text{бал за питання})_i \in [0;5]$$

Штрафні бали:

- за несвоєчасний (більш ніж 1 тиждень) захист практичних робіт – «-0,5 бала».

Заохочувальні бали:

- за участь у факультетській олімпіаді з дисципліни, модернізації лабораторних робіт, виконання завдань із удосконалення дидактичних матеріалів з дисципліни надається +5 - +10 (заохочувальних) балів.

Розрахунок шкали (R) рейтингу:

Сума вагових балів контрольних заходів протягом семестру становить:

$R_n=100$ балів.

Рейтингова оцінка студента RD з кредитного модуля формується як сума всіх рейтингових балів r_k , а також заохочувальних/ штрафних балів r_s .

$$RD = \sum_k r_k + \sum_s r_s$$

Студенти, які набрали протягом семестру кількість балів $RD_n \geq 0,6R_n=60$ балів, мають можливість:

- отримати залікову оцінку «автоматом» відповідно до набраного рейтингу, переведеного в оцінку згідно з таблицею;
- виконувати залікову роботу з метою підвищення оцінки;
- у разі отримання оцінки, більшої ніж «автоматом» з рейтингу, студент отримує оцінку за результатами контрольної роботи;
- у разі отримання оцінки меншої ніж «автоматом» з рейтингу, попередній рейтинг студента з дисципліни скасовується і він отримує оцінку тільки за результатами залікової контрольної роботи.

Студенти, які набрали протягом семестру рейтинг з кредитного модуля менше балів $0,6R_n=60$ балів, зобов'язані виконувати залікову контрольну роботу.

Значення рейтингу з кредитного модулю	Оцінка ECTS	Традиційна оцінка
95...100	A	зараховано
85...94	B	
75..84	C	
65..74	D	
60...64	E	
40...59	F _x	не зараховано
$r_c < 40$ або не виконані інші умови допуску до заліку	F	не допущений

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

Головна частина рейтингу студента формується через активну участь у практичних заняттях та отримання результатів модульної контрольної роботи.

Вид контролю	Спосіб контролю
Поточний контроль	Частина 1. Перевірка підготовки до практичних занять Частина 2. Перевірка виконання практичних занять відповідно до розкладу занять, модульні контрольні роботи
Календарний контроль	Проводиться двічі на семестр як моніторинг поточного стану виконання вимог силабусу
Семестровий контроль	Залік

Таблиця відповідності рейтингових балів оцінкам за університетською шкалою:

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

11. Додаткова інформація з дисципліни (освітнього компонента)

Теми дисципліни взаємозв'язані, матеріал вивчається в логічній послідовності, закріплення матеріалу здійснюється на практичних заняттях та під час самостійної підготовки студентів. Завершується вивчення навчальної дисципліни екзаменом.

На лекціях розкриваються найбільш суттєві теоретичні питання, які дозволяють забезпечити студентам можливість глибокого самостійного вивчення всього програмного матеріалу.

На практичних заняттях студенти закріплюють пройдений теоретичний матеріал на практиці.

Теоретичні знання поглиблюються шляхом самостійної роботи з використанням ресурсів глобальної мережі Internet.

Додатковий матеріал, або той, що не вимагає керівництва викладача, виноситься на самостійні заняття.

Вивчення всіх тем здійснюється загально прийнятою методикою: основи знань викладаються на лекціях, технічні принципи використання та розробки спеціального програмного забезпечення, а також алгоритми його роботи засвоюються в процесі практичних занять.

На заняттях використовуються презентації Power Point, слайди, навчальні схеми, стенди, технічні засоби навчання, матеріальна частина, обчислювальна техніка.

Контроль засвоєння навчального матеріалу здійснюється індивідуальним опитуванням, письмовими відповідями на поставлені питання з використанням джерел інформації (відповіді творчого плану) і без їх використання (доповіді матеріалу, який вивчається), співбесідою в індивідуальному порядку на консультаціях, а також на екзамені.

Робочу програму навчальної дисципліни (силабус):

Складено доцент Суліма С.В.

Ухвалено кафедрою ІТТ (протокол № 14 від 27.05.2026 р.)

Погоджено Методичною комісією НН ІТС (протокол № 7 від 18.06.2026 р.)