

Черкаський державний технологічний університет
Факультет інформаційних технологій і систем

«ЗАТВЕРДЖУЮ»

Голова вченої ради
факультету інформаційних
технологій і систем

І. Б. Трегубенко

Протокол № 2

« 28 » 09 2020р.

РОБОЧА ПРОГРАМА

навчальної дисципліни

(СИЛАБУС)

«Основи програмної інженерії»

Шифр за ОПП – ОПП-4

Рівень вищої освіти	бакалаврський
Галузь знань –	12 «Інформаційні технології»
Спеціальність –	126 «Інформаційні системи та технології»
Освітня програма –	«Web-технології, Web-дизайн»

2020-2021 навчальний рік

Робоча програма навчальної дисципліни (силабус) «Основи програмної інженерії» підготовки здобувачів освітнього рівня «бакалавр» за спеціальністю 126 «Інформаційні системи та технології», освітня програма «Web-технології, Web-дизайн» – 12 стор.

Робоча програма навчальної дисципліни (силабус) складений на основі програми навчальної дисципліни «Основи програмної інженерії», шифр (за ОПП) – ОПП-4.

Розробник:

Лавданська О. В., к.т.н., доцент кафедри інформаційних технологій проектування
(прізвище та ініціали, наук. ст., вчене звання, посада ІППІ кафедри, що розробив силабус)

Робоча програма навчальної дисципліни (силабус) затверджена на засіданні кафедри інформаційних технологій проектування
(найменування кафедри)

Протокол № 2 від « 14 » вересня 20 20р.

Обговорено та рекомендовано до затвердження методичною комісією факультету інформаційних технологій і систем

« 21 » вересня 20 20р., протокол № 2

Голова методичної комісії
факультету інформаційних технологій і систем


(підпис)

(А. Р. Карапетян)

1 ІНФОРМАЦІЯ ПРО ВИКЛАДАЧА

Прізвище, ім'я, по батькові	Лавданська Ольга В'ячеславівна
Науковий ступінь	кандидат технічних наук
Наукове звання	
Посада	доцент кафедри інформаційних технологій проектування
Місце роботи	Черкаський державний технологічний університет, Кафедра інформаційних технологій проектування
Адреса кафедри	к.603, бул. Шевченка, 460, м. Черкаси, 18006
Контактний телефон	+38 099 648 75 66
Профайл викладача	https://itp.chdtu.edu.ua/staff/lavdanska-olga-vyacheslavivna/
e-mail	yegorovaov@gmail.com
Профайл дисципліни	http://fitis.moodle.chdtu.edu.ua/course/view.php?id=646
Розклад консультацій	щовівторка, 8.30-9.50, а. 603-1

2 ОПИС НАВЧАЛЬНОЇ ДИСЦИПЛІНИ

Галузь знань, спеціальність, освітня програма, освітній рівень	Загальні характеристики		Навчальне навантаження з дисципліни	
			денна форма навчання	заочна форма навчання
<u>Галузь знань</u> 12 «Інформаційні технології»	Обов'язкова		Курс підготовки	
			1	-
<u>Спеціальність</u> 126 «Інформаційні системи та технології»	Загальна кількість кредитів ЄКТС	4	Семестр підготовки	
	Загальна кількість годин	120	1	-
<u>Освітня програма</u> «Web-технології, Web-дизайн»	Кількість аудиторних годин	48	Лекції	
			16 год.	-
	Кількість годин самостійної роботи	72	Практичні, семінарські	
			-	-
<u>Освітній рівень</u> бакалаврський	Мова навчання – українська		Лабораторні	
			32 год.	-
			Самостійна робота	
			72 год.	-
			Форма підсумкового контролю	
		залік	-	

3 МЕТА І ЗАВДАННЯ НАВЧАЛЬНОЇ ДИСЦИПЛІНИ

Мета викладання дисципліни	Теоретична та практична підготовка здобувачів освітнього рівня бакалавра у напрямку використання знань, методів та практичного досвіду для розробки програмного забезпечення.
Завдання вивчення дисципліни	Забезпечити у здобувачів освітнього рівня бакалавра розуміння підходів до аналізу, проектування, реалізації, тестування та документування програмного забезпечення і сформувати вміння їх використовувати.

4 РЕЗУЛЬТАТИ НАВЧАННЯ

№ з/п	Результати навчання
1	ПР 5. 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.
2	ПР 8. 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.
3	ПР 13. Демонструвати знання сучасних тенденцій інженерії програмного забезпечення та практичних навичок використання практик, процесів та методологій розробки програмного забезпечення у професійній діяльності.

5 ПРЕРЕКВІЗИТИ

Відсутні

6 ПОСТРЕКВІЗИТИ

«UX Development», «Об'єктно-орієнтоване програмування».

7 ПРОГРАМА НАВЧАЛЬНОЇ ДИСЦИПЛІНИ

Змістовий модуль №1 <i>Життєвий цикл та процеси розробки програмного забезпечення</i>
Тема 1. Вступ до програмної інженерії.
Предмет, мета та задачі дисципліни. Результати навчання. Поняття програмного забезпечення. Історія розвитку програмного забезпечення. Місце

та значення інженерії програмного забезпечення. Показники якості програмного забезпечення.

Тема 2. Життєвий цикл програмного забезпечення.

Поняття життєвого циклу програмного забезпечення. Процеси життєвого циклу програмного забезпечення. Моделі життєвого циклу програмного забезпечення: водоспадна, ітеративна, спіральна, V-подібна.

Тема 3. Управління проектом

Функції та основні обов'язки проектного менеджера в сфері ІТ. Типи ІТ-компаній. Основні поняття проектного менеджменту: проект, портфель проектів, програма. Класифікація проектів. Життєвий цикл ІТ-проекту. Моделі життєвого циклу ІТ-проекту. Учасники проекту. Середовище проекту. Области знань проектного менеджменту. Процеси управління проектами. Міжнародні стандарти в області управління проектами.

Тема 4. Інженерія вимог до програмного забезпечення.

Поняття вимоги до програмного забезпечення. Функції, які виконують вимоги до програмного забезпечення. Джерела та підходи до визначення вимог до програмного забезпечення. Рівні та типи вимог до програмного забезпечення. Властивості якісних вимог. Техніки тестування вимог. Приклади аналізу та тестування вимог. Типові помилки при аналізі та тестуванні вимог.

Тема 5. Аналіз та проектування програмного забезпечення.

Рівні проектування. Розробка концепції системи. Моделювання бізнес-процесів. Аналіз предметної області. Моделювання поведінки системи. Стратегії проектування програмного забезпечення: структурна, функціональна, об'єктно-орієнтована, процесно-орієнтована. Архітектура програмного забезпечення.

Тема 6. Реалізація програмного забезпечення.

Структурне програмування. Функціональне програмування. Стили програмування. Стандарти програмування. Інструменти та засоби програмування. Складність розробки програмного забезпечення. Програмна документація. Проблеми впровадження програмного забезпечення.

Тема 7. Тестування програмного забезпечення.

Поняття тестування програмного забезпечення. Забезпечення якості програмного забезпечення (quality assurance). Контроль якості програмного забезпечення (quality control). Аудит програмного забезпечення (quality audit). Hard skills тестувальника програмного забезпечення. Життєвий цикл тестування програмного забезпечення. Валідація. Верифікація. Види і напрямки тестування. Рівні тестування. Тестова документація. Автоматизація тестування.

Тема 8. Обслуговування програмного забезпечення.

Види обслуговування програмного забезпечення. Вартість обслуговування програмного забезпечення. Процеси обслуговування програмного забезпечення. Реінжиніринг програмного забезпечення. Повторне використання компонентів.

Змістовий модуль №2 <i>Методології та практики розробки програмного забезпечення</i>	
Тема 1. Agile технологія розробки програмного забезпечення.	
Поняття гнучкого методу. Гнучкі методи. Інструменти гнучкої розробки. Гнучке управління проектами. Масштабування гнучких методів. Використання користувачької історії в agile технології розробки програмного забезпечення.	
Тема 2. RUP технологія розробки програмного забезпечення.	
Основні поняття RUP технології розробки програмного забезпечення. Основні положення RUP технології розробки програмного забезпечення. Життєвий цикл розробки програмного забезпечення за технологією RUP. Новації RUP технології розробки програмного забезпечення.	
Тема 3. Екстремальна технологія розробки програмного забезпечення.	
Основні поняття екстремальної технології розробки програмного забезпечення. Цінності та принципи екстремальної технології. Практики екстремальної технології розробки програмного забезпечення. Життєвий цикл розробки програмного забезпечення за екстремальною технологією. Інструменти екстремальної розробки.	
Тема 4. Практики розробки програмного забезпечення	
Стандарти оформлення коду. Процес інспекції коду. Стратегії випуску програмного забезпечення. Підтримка програмного забезпечення. Юніті та API тестування. Технічне обслуговування програмного забезпечення.	
Тема 5. Основи системи контролю версій.	
Основи Git. Розгалуження в Git. Git на сервері. Розподілений Git. GitHub. Інструменти Git. Налаштування Git.	

8 ТЕМАТИЧНИЙ ПЛАН НАВЧАЛЬНОЇ ДИСЦИПЛІНИ

№ теми	Назва модулів і тем	Форми організації навчання, кількість годин						Література, інформаційні ресурси
		Денна форма			Заочна форма			
		Лекції	Лабораторні роботи	Самостійна робота	Лекції	Лабораторні роботи	Самостійна робота	
Змістовий модуль № 1. <i>Життєвий цикл та процеси розробки програмного забезпечення</i>								
1	Вступ до програмної інженерії	1		4				1,2
2	Життєвий цикл програмного забезпечення	1		4				2,3
3	Управління проектом	1	4	5				1,3
4	Інженерія вимог до програмного забезпечення	1	4	5				1,2,3
5	Аналіз та проектування програмного	1	4	5				1,2,3

	забезпечення							
6	Реалізація програмного забезпечення	1	4	4				1,2,3
7	Тестування програмного забезпечення	1		4				1,2,3
8	Обслуговування програмного забезпечення	1		5				1,2,3
Змістовий модуль № 2.								
<i>Методології та практики розробки програмного забезпечення</i>								
9	Agile технологія розробки програмного забезпечення	2	4	7				1,2
10	RUP технологія розробки програмного забезпечення	1		7				1,2
11	Екстремальна технологія розробки програмного забезпечення	1		7				2,3
12	Практики розробки програмного забезпечення	1	4	7				2,3
13	Основи системи контролю версій	3	8	8				1,2
	Разом	16	32	72				

9 ЛАБОРАТОРНІ РОБОТИ

№ з/п	Назва лабораторної роботи	Кількість годин	
		Денна	Заочна
1	Розробка концепції програмної системи	4	
2	Управління проектом розробки програмної системи	4	
3	Розробка use case diagram	4	
4	Розробка user story	4	
5	Створення баг-репорту	4	
6	Основи Git	4	
7	Галуження в git	4	
8	Створення та налаштування облікового запису в GitHub	4	

МЕТОДИЧНЕ ЗАБЕЗПЕЧЕННЯ

https://drive.google.com/file/d/1-obNBaffD_9Rw-gHZabPOIov7jmJTfEC/view?usp=sharing

10 САМОСТІЙНА РОБОТА

Рекомендації до самостійної роботи здобувачів вищої освіти

МЕТОДИЧНЕ ЗАБЕЗПЕЧЕННЯ

<https://drive.google.com/file/d/1Mf-wvAcuIoNOIpzQv0REYv8Y4OdDhSsh/view?usp=sharing>

11 СИСТЕМА ОЦІНЮВАННЯ НАВЧАЛЬНИХ ДОСЯГНЕНЬ

11.1 МЕТОДИ КОНТРОЛЮ

В організації навчального процесу застосовуються контрольні заходи у формі вхідного, поточного, модульного, рейтингового і підсумкового контролю.

Вхідний контроль проводиться перед вивченням нового курсу з метою визначення рівня підготовки здобувачів вищої освіти з дисциплін, які забезпечують цей курс. За результатами вхідного контролю розробляються заходи з надання індивідуальної допомоги здобувачам вищої освіти, коригування навчального процесу з відповідного курсу.

Поточний контроль здійснюється під час проведення лекцій та лабораторних занять і має на меті перевірку рівня підготовленості здобувача вищої освіти до виконання конкретних видів навчальної діяльності.

Модульний контроль успішності здобувачів вищої освіти здійснюється для перевірки рівня засвоєння навчального матеріалу в кінці кожного навчального модуля.

Рейтинговий контроль є інструментом комплексного оцінювання якості навчальної роботи здобувача вищої освіти з усіх кредитних модулів на певному етапі навчання. Рейтинговий контроль успішності здобувачів вищої освіти проводиться на 8-9 навчальних тижнях.

Семестровий контроль з дисципліни проводиться відповідно до навчального плану у вигляді заліку в терміни, встановлені графіком навчального процесу, та в обсязі навчального матеріалу, визначеному робочою програмою дисципліни.

Залік – це вид підсумкового контролю, за якого засвоєння здобувачем вищої освіти навчального матеріалу з дисципліни оцінюється на підставі результатів поточного, проміжного контролів (тестування, поточного опитування, виконання індивідуальних завдань та певних видів робіт на лабораторних заняттях) протягом семестру і модульного контролю.

11.2 ПИТАННЯ ДО ЗАЛІКУ

1. Історія розвитку програмного забезпечення.
2. Місце та значення інженерії програмного забезпечення.
3. Показники якості програмного забезпечення.
4. Поняття життєвого циклу програмного забезпечення.
5. Процеси життєвого циклу програмного забезпечення.
6. Водоспадна модель життєвого циклу програмного забезпечення.
7. Ітеративна модель життєвого циклу програмного забезпечення.
8. Спіральна модель життєвого циклу програмного забезпечення.
9. V-подібна модель життєвого циклу програмного забезпечення.
10. Функції та основні обов'язки проектного менеджера в сфері ІТ.

11. Типи ІТ-компаній.
12. Основні поняття проектного менеджменту: проект, портфель проектів, програма.
13. Класифікація проектів.
14. Життєвий цикл ІТ-проекту.
15. Моделі життєвого циклу ІТ-проекту.
16. Учасники проекту. Середовище проекту.
17. Області знань проектного менеджменту. Процеси управління проектами.
18. Міжнародні стандарти в області управління проектами.
19. Функції, які виконують вимоги до програмного забезпечення.
20. Джерела та підходи до визначення вимог до програмного забезпечення.
21. Рівні та типи вимог до програмного забезпечення.
22. Властивості якісних вимог.
23. Техніки тестування вимог.
24. Приклади аналізу та тестування вимог.
25. Типові помилки при аналізі та тестуванні вимог.
26. Рівні проектування програмного забезпечення.
27. Розробка концепції системи.
28. Моделювання бізнес-процесів.
29. Аналіз предметної області.
30. Моделювання поведінки системи.
31. Стратегії проектування програмного забезпечення: структурна, функціональна, об'єктно-орієнтована, процесно-орієнтована.
32. Архітектура програмного забезпечення.
33. Структурне програмування.
34. Функціональне програмування.
35. Стили програмування.
36. Стандарти програмування.
37. Інструменти та засоби програмування.
38. Складність розробки програмного забезпечення.
39. Програмна документація.
40. Проблеми впровадження програмного забезпечення.
41. Поняття тестування програмного забезпечення.
42. Забезпечення якості програмного забезпечення (quality assurance).
43. Контроль якості програмного забезпечення (quality control).
44. Аудит програмного забезпечення (quality audit).
45. Hard skills тестувальника програмного забезпечення.
46. Життєвий цикл тестування програмного забезпечення.
47. Валідація. Верифікація.
48. Види і напрямки тестування.
49. Рівні тестування.
50. Тестова документація.
51. Автоматизація тестування.
52. Види обслуговування програмного забезпечення.
53. Вартість обслуговування програмного забезпечення.

54. Процеси обслуговування програмного забезпечення.
55. Реінжиніринг програмного забезпечення.
56. Повторне використання компонентів.
57. Гнучкі методи розробки програмного забезпечення.
58. Інструменти гнучкої розробки.
59. Гнучке управління проектами. Масштабування гнучких методів.
60. Використання користувацької історії в agile технології розробки програмного забезпечення.
61. Основні поняття RUP технології розробки програмного забезпечення.
62. Основні положення RUP технології розробки програмного забезпечення.
63. Життєвий цикл розробки програмного забезпечення за технологією RUP.
64. Новації RUP технології розробки програмного забезпечення.
65. Основні поняття екстремальної технології розробки програмного забезпечення.
66. Цінності та принципи екстремальної технології.
67. Практики екстремальної технології розробки програмного забезпечення.
68. Життєвий цикл розробки програмного забезпечення за екстремальною технологією.
69. Інструменти екстремальної розробки.
70. Стандарти оформлення коду.
71. Процес інспекції коду.
72. Стратегії випуску програмного забезпечення.
73. Підтримка програмного забезпечення.
74. Юніті та API тестування.
75. Технічне обслуговування програмного забезпечення.
76. Основи Git.
77. Розгалуження в Git.
78. Git на сервері.
79. Розподілений Git.
80. GitHub.
81. Інструменти Git.
82. Налаштування Git.

11.3 КРИТЕРІЇ ОЦІНЮВАННЯ

ДЕННА ФОРМА

Для здобувачів денної форми навчання	
Вид навчальної роботи	Кількість балів максимум
<u>Постійна частина</u>	
ЗМІСТОВИЙ МОДУЛЬ № 1	
«Життєвий цикл та процеси розробки програмного забезпечення» – 60 годин	
Захист лабораторної роботи № 1	10
Захист лабораторної роботи № 2	10

Захист лабораторної роботи № 3	10
Захист лабораторної роботи № 4	10
Модульна контрольна робота № 1	10
<i>Всього за змістовим модулем № 1</i>	50
ЗМІСТОВИЙ МОДУЛЬ № 2 «Методології та практики розробки програмного забезпечення» – 60 годин	
Захист лабораторної роботи № 5	10
Захист лабораторної роботи № 6	10
Захист лабораторної роботи № 7	10
Захист лабораторної роботи № 8	10
Модульна контрольна робота № 2	10
<i>Всього за змістовим модулем № 2</i>	50
<u>Додаткова частина</u>	
Підготовка та захист реферату за індивідуальною темою	20
Участь у Днях студентської науки	20
Участь у науковій конференції чи семінарі за темою дисципліни	20
Оформлення наочного стенда за індивідуальною темою	20
<u>Штрафна частина</u>	
Пропуск одного заняття без поважної причини	-5
Несвоєчасний захист звіту з лабораторної роботи	-5
ПІДСУМКОВА СЕМЕСТРОВА ОЦІНКА	100

ЗАОЧНА ФОРМА

Вид навчальної роботи	Кількість балів максимум
<i>Контрольна робота з дисципліни (відповідно до отриманого завдання)</i>	60
Залік	40
Разом	100

12 РЕКОМЕНДОВАНА ЛІТЕРАТУРА

Основна

1. Sommerville I. Software engineering; tenth edition. Harlow: Pearson Education Limited, 2016. 811 p.
2. Куликов С. С. Тестирование программного обеспечения. Базовый курс. Минск: Четыре четверти, 2017. 312 с.
3. Табунщик Г. В., Каплієнко Т. І., Петрова О. А. Проектування та моделювання програмного забезпечення сучасних інформаційних систем. Запоріжжя : Дике Поле, 2016. 250 с.

Допоміжна

4. Гагарина Л.Г., Кокарева Е.В., Сидорова-Виснадул Б.Д. Технология разработки программного обеспечения. М.: ИД Форум, Инфра-М, 2018. 400 с.

5. Гниденко И.Г., Павлов Ф.Ф., Федоров Д.Ю. Технология разработки программного обеспечения: учеб. пособие для СПО. М.: Издательство Юрайт, 2018. 235 с.
6. Орлов С.А. Программная инженерия; 5-е изд., обновленное и доп. СПб: Издательский дом «Питер», 2016. 640 с.

13 ІНФОРМАЦІЙНІ РЕСУРСИ

1. Software Engineering | Introduction to Software Engineering. URL: <https://www.geeksforgeeks.org/software-engineering-introduction-to-software-engineering/> (date of the request: 15.08.2020).
2. Software Case Tools Overview. URL: https://www.tutorialspoint.com/software_engineering/case_tools_overview.htm. (date of the request: 15.08.2020).
3. When to use Scrum? Waterfall vs. Scrum vs. Kanban vs. Scrumban. URL: <https://dusted.codes/when-to-use-scrum-waterfall-vs-scrum-vs-kanban-vs-scrumban> (date of the request: 15.08.2020).
4. Git--everything-is-local. URL: <https://git-scm.com/book/uk/v2> (date of the request: 15.08.2020).

14 ПЕРЕЛІК НОРМАТИВНИХ ДОКУМЕНТІВ

1. Кодекс академічної доброчесності Черкаського державного технологічного університету (зі змінами 23.12.2019), затверджений рішенням Вченої ради ЧДТУ, протокол № 7 від 23.12.2019 р.
2. Положення про перевірку академічних і наукових робіт на плагіат, затверджене рішенням Вченої ради ЧДТУ, протокол № 5 від 18.11.2019 р.
3. Положення про організацію освітнього процесу в Черкаському державному технологічному університеті, затверджене рішенням Вченої ради ЧДТУ, протокол № 7 від 18.12.2017 р., зі змінами та доповненнями, внесеними Вченою радою ЧДТУ, протокол № 8 від 15.04.2019 р.
4. Положення про організацію контролю та оцінювання якості навчання студентів, затверджене рішенням Вченої ради ЧДТУ, протокол № 2 від 17.09.2012 р.
5. Порядок проведення семестрового контролю та атестації здобувачів вищої освіти з використанням технологій дистанційного навчання в Черкаському державному технологічному університеті, затверджений рішенням Вченої ради ЧДТУ, протокол № 13 від 18.05.2020 р.
6. ДСТУ 3008:2015. Звіти у сфері науки і техніки. Структура та правила оформлювання / Нац. Стандарт України. Вид. офіц. [На заміну ДСТУ 3008-95; чинний від 2017-07-01]. Київ : ДП «УкрНДНЦ», 2016. 31 с. (Інформація та документація).
7. ДСТУ 8302:2015. Бібліографічне посилання. Загальні положення та правила складання / Нац. Стандарт України. Вид. офіц. [Уведено вперше; чинний від 2016-07-01]. Київ : ДП «УкрНДНЦ», 2016. 17 с. (Інформація та документація). З внесеними правками.

15 ПОЛІТИКА ДИСЦИПЛІНИ

Для успішного вивчення дисципліни та проходження контрольних заходів здобувачі вищої освіти зобов'язані:

- не запізнюватися на заняття;
- не пропускати заняття (у разі хвороби надати довідку або її ксерокопію);
- своєчасно і самостійно виконувати всі передбачені програмою завдання до лабораторних робіт;
- брати очну участь у контрольних заходах;
- оволодіти навчальним матеріалом для самостійного вивчення з дисципліни у вільний від обов'язкових занять час;
- підтримувати зворотній зв'язок з викладачем на всіх етапах вивчення дисципліни;
- дотримуватися принципів академічної доброчесності.