



СИЛАБУС НАВЧАЛЬНОЇ ДИСЦИПЛІНИ

КОНСТРУЮВАННЯ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ

Галузь знань	12 Інформаційні технології
Шифр та назва спеціальності	121 Інженерія програмного забезпечення
Назва освітньо-професійної програми	Освітньо-професійна програма підготовки «Інженерія програмного забезпечення»
Рівень вищої освіти	Перший (бакалаврський)
Факультет	Інформаційних технологій та кібербезпеки
Кафедра	Інженерії програмного забезпечення
Статус навчальної дисципліни	ОК-21 ОПП Освітньо-професійна програма підготовки «Інженерія програмного забезпечення»
Форма навчання	Денна

Викладачі

Глазунова Людмила Володимирівна
lvglazun@gmail.com



Доцент кафедри Інженерії програмного забезпечення, кандидат фізико-математичних наук, доцент

Загальна інформація про дисципліну

Анотація до дисципліни	<p>Дисципліна «Конструювання програмного забезпечення» повинна сприяти формуванню у студентів професійних практичних знань, оволодіння основними принципами конструювання програмного забезпечення, набуття практичних навичок самостійного написання якісного програмного коду для розв'язку задач у практичній діяльності. Предметом вивчення дисципліни «Конструювання програмного забезпечення» є:</p> <ol style="list-style-type: none">1) застосування специфікації вимог до проектування програмного продукту;2) застосування абстрагування для виділення об'єктів та класів;3) побудова моделі концептуальних класів;4) застосування діаграми взаємодії для розподілу обов'язків між класами;
------------------------	--

	<p>5) вибір мови об'єктно орієнтованого програмування та інтеграції модулів;</p> <p>6) застосування сучасних інструментів реалізації сервісної частини програмного забезпечення;</p> <p>7) інтеграція програмних та апаратних засобів.</p>
Мета дисципліни	Надбання студентами знань з методів конструювання програмного забезпечення, оволодіння методикою та технологією об'єктно орієнтованого конструювання, одержання навичок створення якісного програмного коду та знань з налагодження модулів програмного продукту.
Компетентності, формуванню яких сприяє дисципліна	<p>СК-14. Здатність брати участь у проектуванні програмного забезпечення, включаючи проведення моделювання (формальний опис) його структури, поведінки та процесів функціонування.</p> <p>СК-15. Здатність розробляти архітектури, модулі та компоненти програмних систем.</p> <p>СК-17. Здатність дотримуватися специфікацій, стандартів, правил і рекомендацій в професійній галузі при реалізації процесів життєвого циклу.</p> <p>СК-23. Здатність реалізовувати фази та ітерації життєвого циклу програмних систем та інформаційних технологій на основі відповідних моделей і підходів розробки програмного забезпечення.</p> <p>СК-25. Здатність обґрунтовано обирати та освоювати інструментарій з розробки та супроводження програмного забезпечення.</p>
Результати навчання	<p>ПРН-3. Знати основні процеси, фази та ітерації життєвого циклу програмного забезпечення.</p> <p>ПРН-4. Знати і застосовувати професійні стандарти та інші нормативно-правові документи в галузі інженерії програмного забезпечення.</p> <p>ПРН-5. Знати і застосовувати відповідні математичні поняття, методи доменного, системного й об'єктно-орієнтованого аналізу та математичного моделювання для розробки програмного забезпечення.</p> <p>ПРН-6. Уміння вибирати та використовувати відповідну задачі методологію створення програмного забезпечення.</p> <p>ПРН-12. Застосовувати на практиці ефективні підходи щодо проектування програмного забезпечення.</p> <p>ПРН-17. Вміти застосовувати методи компонентної розробки програмного забезпечення.</p>
Обсяг дисципліни	Загальний обсяг дисципліни: 6 кредитів ЄКТС (180 годин). Для денної форми навчання: лекції – 26 години, практичні заняття – 26 годин, лабораторні заняття – 14, самостійна робота – 114 годин.
Форма підсумкового контролю	Екзамен
Терміни викладання дисципліни	Дисципліна викладається у 1-му семестрі 3-го року навчання

Програма дисципліни

Тема 1.	<p><i>Моделі конструювання ПЗ: основні поняття.</i></p> <p>Місце конструювання у процесі розробки ПЗ. Порівняння структурного і об'єктно-орієнтованого підходів моделювання ПЗ. Уніфікований процес розробки ПЗ та інші технології. Інструменти моделювання ПЗ.</p>
----------------	---

Тема 2.	<i>Розробка архітектури ПЗ на основі аналізу вимог до ПЗ.</i> Реалізація нефункціональних вимог до системи (надійність, безпека, зберігання даних у конкретному середовищі, інтерфейси із зовнішніми системами і т.д.) за допомогою розробки архітектури системи. Принципи побудови архітектури програмного забезпечення. Основні архітектурні стилі. Архітектурні рівні проектування інформаційної системи. Основні сучасні типи застосунків.
Тема 3.	<i>Конвенції програмування.</i> Поширені стандарти кодування. Знайомство з інтегрованим середовищем розробки. Поняття про пакети та модулі, дебагінг коду, віртуальне середовище. Специфікація мов програмування та задач. Стандартизація мов програмування. Сумісна розробка програмного забезпечення.
Тема 4.	<i>Принципи і шаблони створення якісного програмного коду.</i> Застосування принципів S.O.L.I.D для створення якісного програмного коду. Шаблони проектування GoF.
Тема 5.	<i>Інструменти для реалізації програмного забезпечення .</i> Знайомство з фреймворком ASP.NET MVC, його основами. Екземпляри класу. Робота з базами даних, запити до бази даних, фільтри, сортування, пошук агрегація та міграція даних. Зв'язок з сервером, файлові сховища, автентифікація та авторизація.
Тема 6.	<i>Інтеграції програмного забезпечення.</i> Способи забезпечення надійності системи при інтеграції програмних і апаратних засобів. Сучасні технології інтеграції інформаційних ресурсів. Роль інтеграційних шаблонів проектування. Типи інтеграційних задач. Слабке зв'язування. Компоненти слабкозв'язаного інтеграційного рішення.
Тема 7.	<i>Методологія DevOps.</i> Методологія DevOps: безперервна інтеграція коду (CI), безперервна доставка та розгортання коду (CD), безперервне тестування (CT). Переваги взаємної інтеграції CI/CD/CT.

Список рекомендованих джерел

1. Larman C., Applying UML and patterns. An Introduction to Object-Oriented analysis and Design and Iterative Development/ 3rd edition/ Prentice Hall PTR Upper Saddle River, NJ 07458. 2005. - 736 p.
2. Керниган В., Pike Р. The Practice of Programming / Addison-Wesley, New York, 2004. – 288 p.
3. Табунщик Г.В., Каплієнко Т.І., Петрова О.А., Проектування та моделювання програмного забезпечення сучасних інформаційних систем./ Навчальний посібник./ Запоріжжя: Дике Поле ЗНТУ 2016. – 250 с. ISBN 978-966-2752-07-0
4. Лавріщева К.М. Програмна інженерія: Підручник / К.М. Лавріщева. – ВНУ, 2008. – 435 с.
5. Трофименко О. Г., Манаков С. Ю., Ларін Д. Г. Основи програмної інженерії : навч.-метод. посібник [Електронне видання] / О. Г. Трофименко, С. Ю. Манаков, Д. Г. Ларін. – Одеса : Фенікс, 2022. – 194 с. ISBN 978-966-928-808-0
6. Моделювання та аналіз програмного забезпечення: методичні вказівки до практичних та лабораторних занять [Електронне видання] / Укладач: Л.В. Глазунова – Одеса:ДУІТЗ, 2021., с. 92, <https://metod.suitt.edu.ua/>

Інформація про консультації

Щосереди на протязі семестру з 14¹⁵ до 15¹⁵ год., конференція ZOOM доцента Глазунова Л.В.

Загальна схема оцінювання

Сума балів за всі види навчальної діяльності	Шкала ЄКТС	Оцінка за національною шкалою		Нарахування балів	Бали нараховуються таким чином:
		для іспиту	для заліку		
90-100	A	Відмінно	зараховано		<i>Оцінювання знань здобувачів вищої освіти здійснюється за 100-бальною шкалою і становить: за поточну успішність (участь у практичних заняттях, виконання практичних завдань та контрольних робіт) – до 60 балів, за результати екзамену – до 40 балів.</i>
82-89	B	Добре			
74-81	C				
64-73	D				
60-63	E	Задовільно			
35-59	FX	Незадовільно з можливістю повторного складання	Не зараховано з можливістю повторного складання		
0-34	F	Незадовільно з обов'язковим повторним вивченням дисципліни	Не зараховано з обов'язковим повторним вивченням дисципліни		

Політика опанування дисципліни

Відвідування: Присутність на лекціях практичних, лабораторних заняттях та контрольних заходах (екзамен/залік) є обов'язковою. Відсутність на заняттях може бути лише з поважних причин.

Дотримання принципів академічної доброчесності: Підготовка усіх завдань, письмових робіт і т. ін., що виконуються в межах дисципліни, здійснюється здобувачем вищої освіти самостійно, на засадах академічної доброчесності. Викладач має право для перевірки робіт застосовувати програму **Unicheck**.

Умови зарахування пропущених занять: Зарахування пропущених практичних та лабораторних занять проводяться під час консультацій.

Інші умови: Навчально-методичні матеріали дисципліни розміщені на платформі Moodle, за посиланням <http://e-learning2.suitt.edu.ua/course/view.php?id=838>