

ХМЕЛЬНИЦЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ

Факультет інформаційних технологій
Кафедра комп'ютерної інженерії та інформаційних систем



ЗАТВЕРДЖУЮ
 Тетяна ГОВОРУЩЕНКО
 6 вересня 2024 р.

Навчальна дисципліна Технології проєктування програмних систем

Освітньо-наукова програма Комп'ютерна інженерія та програмування

Рівень вищої освіти другий (магістерський)

Загальна інформація

Позиція	Зміст інформації
Викладач(і)	Говорушенко Тетяна Олександрівна
Профайл викладача	http://kiis.khmn.u.edu.ua/personnel/hovorushchenko
E-mail викладача(ів)	tat_yana@ukr.net
Контактний телефон	заповнюється за домовленістю
Сторінка дисципліни в ІСУ	https://msn.khnu.km.ua/course/view.php?id=3974
Навчальний рік	2024-2025
Консультації	Очні: вівторок, 16.10-17.30, ауд. 1-207 Онлайн: за необхідністю та попередньою домовленістю

Характеристика дисципліни

Статус дисципліни	Форма здобуття освіти	Курс	Семестр	Загальний обсяг		Кількість годин							Форма семестрового контролю		
				Кредити ЄКТС	Години	Аудиторні заняття				Індивідуальна робота студента	Самостійна робота, в т.ч. ІРС	Курсовий проєкт	Курсова робота	Залік	Іспит
						Разом	Лекції	Лабораторні роботи	Практичні заняття						
О	Д	1	1	5	150	51	17	34			99	+			+

Анотація дисципліни

Дисципліна "Технології проєктування програмних систем" є однією зі спеціальних профілюючих дисциплін і тому займає провідне місце у підготовці магістрів комп'ютерної інженерії.

Дисципліна викладається для здобувачів другого (магістерського) рівня вищої освіти денної форми навчання спеціальностей галузі інформаційних технологій. При викладанні дисципліни використовуються активні і творчі форми проведення занять, зокрема, методи проблемного навчання.

Пререквізити: Інженерія програмного забезпечення; **кореквізити:** Теорія алгоритмів та обчислювальних процесів, Іноземна мова за професійним спрямуванням.

Мета і завдання дисципліни

Метою дисципліни є: 1) формування компетентностей, необхідних при проєктуванні програмних систем у складі колективу; 2) ознайомити студентів з сучасними парадигмами та технологіями проєктування програмних систем; 3) надати глибокі та міцні знання з проєктування програмних систем, необхідні для подальшої практичної інженерної та наукової діяльності; 4) ознайомити студентів з теоретичною базою, що використовується при вирішенні задач

проектування програмних систем; 5) виробити у студентів вміння використовувати набуті знання при проектуванні програмних систем; 6) підготувати студентів до проектування програмних систем у складних і непередбачуваних умовах, що потребує застосування нових підходів та породження нових ідей (креативності), самостійного пошуку помилок, оцінювання своєї поведінки та результатів мислення і постійного самовдосконалення; 7) підготувати студентів до провадження дослідницької та/або інноваційної діяльності в галузі проектування програмних систем.

Завдання дисципліни. Надати студентам знання і практичні навички з проектування програмних систем; підготувати студентів до провадження дослідницької та/або інноваційної діяльності в галузі проектування програмних систем, сформувати компетентності, необхідні при проектуванні програмних систем у складі колективу.

Очікувані результати навчання.

Студент, який успішно завершив вивчення дисципліни, повинен: вміло застосовувати програмні засоби управління проектами, засоби проектування програмного забезпечення; засоби тестування програмного забезпечення, обирати оптимальні технологію та мову для проектування та розробки програмного забезпечення, аналізувати отримані результати, користуватись довідковою літературою, самостійно опрацьовувати окремі питання дисципліни; застосовувати знання технічних характеристик, конструктивних особливостей, призначення і правил експлуатації програмно-технічних засобів комп'ютерних систем та мереж для вирішення технічних задач спеціальності; розробляти програмне забезпечення для вбудованих і розподілених застосувань, мобільних і гібридних систем, експлуатувати його; проектувати та розробляти системне та прикладне програмне забезпечення, включаючи інтерфейси користувача програмних систем, враховуючи його призначення, необхідний рівень якості та зручність використання (usability).

Компетентності, на формування яких спрямовано ОК:

Інтегральна - Здатність розв'язувати складні задачі і проблеми в галузі комп'ютерної інженерії або у процесі навчання, що передбачає проведення досліджень та/або здійснення інновацій та характеризується невизначеністю умов і вимог..

ЗК3. Здатність проводити дослідження на відповідному рівні

ЗК6. Здатність виявляти, ставити та вирішувати проблеми

ЗК8. Здатність спілкуватися іноземною мовою

ЗК9. Міжособистісні навички і вміння

ЗК10. Базові дослідницькі навички і вміння

СК1. Здатність до визначення технічних характеристик, конструктивних особливостей, застосування і експлуатації програмних, програмно-технічних засобів, комп'ютерних систем та мереж різного призначення

СК2. Здатність розробляти алгоритмічне та програмне забезпечення, компоненти комп'ютерних систем та мереж, Інтернет додатків, кіберфізичних систем з використанням сучасних методів і мов програмування, а також засобів і систем автоматизації проектування

СК5. Здатність будувати архітектуру та створювати системне і прикладне програмне забезпечення комп'ютерних систем та мереж

СК7. Здатність досліджувати, розробляти та обирати технології створення великих і надвеликих систем

СК8. Здатність забезпечувати якість продуктів і сервісів інформаційних технологій на протязі їх життєвого циклу

СК9. Здатність представляти результати власних досліджень та/або розробок у вигляді презентацій, науково-технічних звітів, статей і доповідей на науково-технічних конференціях

СК10. Здатність ідентифікувати, класифікувати та описувати роботу програмно-технічних засобів, комп'ютерних систем, мереж та їхніх компонентів

СК11. Здатність обирати ефективні методи розв'язування складних задач комп'ютерної інженерії, критично оцінювати отримані результати та аргументувати прийняті рішення

СК12. Здатність використовувати методи аналізу, ідентифікації й синтезу комп'ютерних систем та мереж, кіберфізичних систем, засобів Інтернету речей та ІТ-інфраструктур

СК13. Здатність проектувати системи та їхні компоненти з урахуванням усіх аспектів їх життєвого циклу та поставленої задачі, включаючи створення, налаштування, експлуатацію, технічне обслуговування та утилізацію.

СК14. Здатність проектувати та розробляти інтерфейси користувача програмних систем, враховуючи їх призначення та зручність використання.

СК17. Здатність досліджувати проблему в галузі комп'ютерних та інформаційних технологій, визначати їх обмеження.

Програмні результати навчання, на забезпечення яких спрямовано ОК:

ПРН1. Застосовувати загальні підходи пізнання, методи математики, природничих та інженерних наук до розв'язання складних задач комп'ютерної інженерії

ПРН4. Застосовувати спеціалізовані концептуальні знання, що включають сучасні наукові здобутки у сфері комп'ютерної інженерії, необхідні для професійної діяльності, оригінального мислення та проведення досліджень, критичного осмислення проблем інформаційних технологій та на межі галузей знань.

ПРН5. Розробляти і реалізовувати проекти у сфері комп'ютерної інженерії та дотичні до неї міждисциплінарні проекти з урахуванням інженерних, соціальних, економічних, правових та інших аспектів.

ПРН6. Аналізувати проблематику, ідентифікувати та формулювати конкретні проблеми, що потребують вирішення, обирати ефективні методи їх вирішення.

ПРН7. Вирішувати задачі аналізу та синтезу комп'ютерних систем та мереж.

ПРН8. Застосовувати знання технічних характеристик, конструктивних особливостей, призначення і правил експлуатації програмно-технічних засобів комп'ютерних систем та мереж для вирішення складних задач комп'ютерної інженерії та дотичних проблем.

ПРН9. Розробляти програмне забезпечення для вбудованих і розподілених застосувань, мобільних і гібридних систем

ПРН12. Вільно спілкуватись усно і письмово українською мовою та однією з іноземних мов (англійською, німецькою, італійською, французькою, іспанською) при обговоренні професійних питань, досліджень та інновацій в галузі інформаційних технологій.

ПРН13. Зрозуміло і недвозначно доносити власні знання, висновки та аргументацію з питань інформаційних технологій і дотичних міжгалузевих питань до фахівців і нефахівців, зокрема, до осіб, які навчаються.

ПРН14. Планувати і виконувати наукові дослідження в сфері комп'ютерної інженерії, формулювати і перевіряти гіпотези, обирати методики та інструменти, аналізувати результати, обґрунтовувати висновки.

ПРН15. Вміти виконувати експериментальні дослідження за професійною тематикою

ПРН17. Вміти проєктувати та розробляти системне та прикладне програмне забезпечення, включаючи інтерфейси користувача програмних систем, враховуючи його призначення, необхідний рівень якості та зручність використання (usability)

ПРН 19. Вміти ефективно працювати як індивідуально, так і у складі команди

Тематичний і календарний план вивчення дисципліни

№ тижня	Тема лекції*	Тема практичного заняття*	Тема лабораторної роботи*	Самостійна робота студентів		
				Зміст	Год.	Література
1-2	Сучасний стан сфери виробництва програмних засобів. Порівняльний аналіз та вибір життєвого циклу розроблення програмного забезпечення. Розповсюджені процеси та етапи розроблення програмних систем		Розроблення технічного завдання. Методологія об'єктно-орієнтованого аналізу й проєктування	Опрацювання лекційного матеріалу. Підготовка до ЛР1. Робота над КП	11	[1-5]
3-4	Основні фази, стандарти та засоби розроблення програмного забезпечення. Сучасні технології проєктування програмного забезпечення		Розроблення діаграм варіантів використання і діаграм класів. Розроблення діаграм станів і діаграм діяльності	Опрацювання лекційного матеріалу. Підготовка до захисту ЛР1. Підготовка до ЛР2. Робота над КП	11	[1-5]
5-6	Етап визначення вимог до програмної системи		Розроблення діаграм послідовності і діаграм кооперації. Розроблення діаграм	Опрацювання лекційного матеріалу. Підготовка до захисту ЛР2. Підготовка до ЛР3. Робота над КП	11	[1-5]

			компонентів і діаграм розгортання			
7-8	Архітектура програмних систем. Управління ризиками при розробленні програмного забезпечення		Управління розробкою програмних проєктів	Опрацювання лекційного матеріалу. Підготовка до захисту ЛР3. Підготовка до ЛР4. Робота над КП	11	[1-5]
9-10	Формальні специфікації програмного забезпечення. Сертифікація й оцінювання процесів створення ПЗ		Командна розробка програмних проєктів у Microsoft Team Foundation Server	Підготовка до ТК Т1-4. Опрацювання лекційного матеріалу. Підготовка до захисту ЛР4. Підготовка до ЛР5. Робота над КП	14	[1-5]
11-12	Методи та засоби колективного розроблення програмного забезпечення		Командна розробка програмних проєктів у Microsoft Team Foundation Server	Опрацювання лекційного матеріалу. Підготовка до ЛР5. Робота над КП	11	[1-5]
13-14	Оцінка трудомісткості розроблення програмного забезпечення		Верифікація вихідного коду програмного забезпечення (статичний аналіз)	Опрацювання лекційного матеріалу. Підготовка до захисту ЛР5. Підготовка до ЛР6. Робота над КП	11	[1-5]
15-16	Методи та засоби оцінки якості програмного забезпечення на етапі проєктування		Верифікація вихідного коду програмного забезпечення (динамічний аналіз)	Опрацювання лекційного матеріалу. Підготовка до захисту ЛР6. Підготовка до ЛР7. Підготовка до захисту КП	11	[1-5]
17	Підсумкове заняття		Підсумкове заняття	Опрацювання лекційного матеріалу. Підготовка до захисту ЛР7. Підготовка до ТК Т5-8. Підготовка до захисту КП	8	[1-5]

Примітка: * Послідовність проведення занять визначається розкладом (може не відповідати нумерованим тижням)

Політика дисципліни.

Організація освітнього процесу з дисципліни відповідає вимогам положень про організаційне і навчально-методичне забезпечення освітнього процесу, освітній програмі та навчальному плану. Студент зобов'язаний відвідувати лекції, лабораторні заняття згідно з розкладом, не запізнюватися на заняття, завдання виконувати відповідно до графіка. Пропущене лабораторне заняття студент зобов'язаний опрацювати самостійно у повному обсязі і відзвітувати перед викладачем не пізніше, ніж за тиждень до чергової атестації. До лабораторних занять студент має підготуватися за відповідною темою і проявляти активність. Набуті особою знання з дисципліни або її окремих розділів у неформальній освіті зараховуються відповідно до Положення про порядок перезарахування результатів навчання у ХНУ.

Критерії оцінювання результатів навчання.

Кожний вид роботи з дисципліни оцінюється за **чотирибальною** шкалою. Семестрова підсумкова оцінка визначається як середньозважена з усіх видів навчальної роботи, виконаних і зданих позитивно з врахуванням

коефіцієнта вагомості. Вагові коефіцієнти змінюються залежно від структури дисципліни і важливості окремих її видів робіт. При оцінюванні знань студентів використовуються різні засоби контролю, зокрема: усне опитування перед допуском до виконання лабораторної роботи – здійснюється на її початку; засвоєння теоретичного матеріалу з тем перевіряється тестовим контролем; якість виконання, набуття теоретичних знань і практичних навичок перевіряється шляхом захисту кожної лабораторної роботи згідно з робочою програмою дисципліни і робочим навчальним планом.

Структурування дисципліни за видами робіт і оцінювання результатів навчання студентів денної форми навчання у семестрі за ваговими коефіцієнтами

Аудиторна робота							Самостійна, індивідуальна робота	Семестровий контроль (іспит)	
I семестр									
Лабораторні роботи №:							Контроль:	Підсумковий контрольний захід	
1	2	3	4	5	6	7	ТК Т 1-4		
ВК:							0,4	0,2	0,4

Умовні позначення: ТК – тестовий контроль; Т – тема дисципліни; ВК – ваговий коефіцієнт.

Структурування курсового проєктування для дисципліни за видами робіт і оцінювання результатів курсового проєктування студентів у семестрі за ваговими коефіцієнтами

Самостійна, індивідуальна робота			Підсумковий контроль	
I семестр				
Роботи (контрольні точки) №			Захист	
1	2	3		
ВК:			0,8	0,2

Умовні позначення: ВК – ваговий коефіцієнт.

Оцінювання тестових завдань. Тематичний тест для кожного студента складається з двадцяти тестових завдань, кожне з яких оцінюється одним балом. Максимальна сума балів, яку може набрати студент, складає 20. На тестування відводиться 20 хвилин. Тестування проводиться з використанням модульного середовища для навчання MOODLE.

Оцінювання здійснюється за чотирибальною шкалою.

Відповідність набраних балів за тестове завдання оцінці, що виставляється студенту, представлена у нижченаведеній таблиці.

Сума балів за тестове завдання	1–11	12–14	15–18	19–20
Оцінка	2	3	4	5

Співвідношення інституційної шкали оцінювання і шкали оцінювання ЄКТС

Оцінка ECTS	Інституційна шкала балів	Інституційна оцінка	Критерії оцінювання
A	4,75-5,00	5	Зараховано Відмінно – глибоке і повне опанування навчального матеріалу і виявлення відповідних умінь та навичок. Добре – повне знання навчального матеріалу з кількома незначними помилками. Добре – в загальному правильна відповідь з двома-трьома суттєвими помилками. Задовільно – неповне опанування програмного матеріалу, але достатнє для практичної діяльності за професією. Задовільно – неповне опанування програмного матеріалу, що задовольняє мінімальні критерії оцінювання
B	4,25-4,74	4	
C	3,75-4,24	4	
D	3,25-3,74	3	
E	3,00-3,24	3	
FX	2,00-2,99	2	Незараховано Незадовільно – безсистемність одержаних знань і неможливість продовжити навчання без додаткових знань з дисципліни Незадовільно – необхідна серйозна подальша робота і повторне вивчення дисципліни.
F	0,00-1,99	2	

Питання для підсумкового контролю з дисципліни

1. Огляд та порівняльний аналіз моделей життєвого циклу ПЗ.
2. Вибір прийнятної моделі життєвого циклу ПЗ.

3. Сучасний стан сфери виробництва програмних засобів.
4. Розповсюджені процеси та етапи розроблення програмних систем.
5. Сертифікація процесів створення ПЗ.
6. Оцінювання процесів створення ПЗ.
7. Визначення технології проектування програмного забезпечення (ТППЗ).
8. Загальні вимоги, пропоновані до ТППЗ.
9. Приклади ТППЗ.
10. Моделі програмних систем.
11. Прототипування програмних систем.
12. Основні фази розроблення ПЗ: формулювання вимог, формулювання цілей проекту, аналіз прикладної галузі, створення функційної специфікації, проектування, реалізація.
13. Стандарти в галузі розроблення ПЗ.
14. Програмні засоби підтримки життєвого циклу.
15. Методи визначення вимог.
16. Планування етапу визначення вимог.
17. Формалізація вимог: виділення вимог за допомогою прецедентів.
18. Формалізація вимог: псевдокод, кінцеві автомати, графічні дерева рішень, візуальне подання вимог за допомогою діаграм UML.
19. Завдання та результати етапу аналізу вимог.
20. Планування архітектури.
21. Проектування архітектури.
22. Документування архітектури.
23. Аналіз архітектури.
24. Поняття ризику.
25. Управління ризиками.
26. Формальні специфікації як засіб підвищення якості ПЗ.
27. Специфікування інтерфейсів.
28. Специфікація поведінки систем.
29. Мови розроблення формальних специфікацій.
30. Методологія розроблення ПЗ Microsoft Solutions Framework (MSF).
31. Принципи створення бібліотеки MSF.
32. Модель команди в MSF, ролеві кластери, масштабованість команд та керування компромісами у MSF.
33. Гнучкий підхід до створення ПЗ, основні принципи гнучкого розроблення.
34. Реалізація концепції керування програмним проектом на всіх етапах життєвого циклу у Visual Studio.
35. Функціональні можливості та архітектура TeamFoundationServer (TFS).
36. Способи розгортання TFS на одному або декількох серверах, в одному домені, робочі групі або в декількох доменах.
37. Шаблони командних проектів TFS, області керування командними проектами.
38. Питання створення командного проекту, зміст програмної інфраструктури проекту, склад і призначення робочих елементів.
39. Аналіз методології Scrum, робочі елементи шаблону MicrosoftVisualStudioScrum. Організація колективу у методології Scrum.
40. Методика оцінки трудомісткості розробки ПЗ на основі функціональних точок.
41. Алгоритмічне моделювання трудомісткості розробки ПЗ.
42. Поняття та модель якості ПЗ.
43. Основні принципи метричного аналізу.
44. Вибір метрик, придатних до використання на етапі проектування ПЗ.
45. Дослідження результатів метричного аналізу.
46. Засоби статичного аналізу програмного коду.
47. Дослідження результатів статичного аналізу.

9. МЕТОДИЧНЕ ЗАБЕЗПЕЧЕННЯ

Навчальний процес з дисципліни забезпечений необхідними навчально-методичними розробками в модульному середовищі.

10. РЕКОМЕНДОВАНА ЛІТЕРАТУРА

1. Говорушенко Т. О. Аналіз, дослідження та оцінювання програмних систем: навчальний посібник. – Хмельницький: Хмельницький національний університет, 2019. 358 с.
2. Говорушенко Т.О. Технологія проектування програмних систем: Методичні вказівки для виконання лабораторних робіт для магістрів спеціальності 123 “Комп’ютерна інженерія” денної форми навчання. – Хмельницький: ХНУ, 2021. – 90 с.
3. Говорушенко Т.О. Технології проектування програмних систем: Методичні вказівки до курсового проектування для здобувачів вищої освіти за освітньо-науковою програмою “Комп’ютерна інженерія та програмування” спеціальності 123 “Комп’ютерна інженерія” другого (магістерського) рівня вищої освіти. – Хмельницький: ХНУ, 2022. – 65 с.
4. C. Hu. An Introduction to Software Design: Concepts, Principles, Methodologies, and Techniques 1st ed. 2023 Edition
5. M. Chemuturi. Software Design: A Comprehensive Guide to Software Development Projects 1st Edition, 2018.

Розробник:



д.т.н., проф. Говорушенко Т.О.

Погоджено:

В.о. зав. каф. КПС:



к.т.н., доц. Засорнова І.О

Гарант ОНП «КІП»:



д.т.н., проф. Савенко О.С.