НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ БІОРЕСУРСІВ І ПРИРОДОКОРИСТУВАННЯ УКРАЇНИ

КАФЕДРА КОМП'ЮТЕРНИХ НАУК

«ЗАТВЕРДЖУЮ»
Декан факультету інформаційних технологій
_____ Олена ГЛАЗУНОВА
«___» _____ 20 23 р.

«СХВАЛЕНО»
на засіданні кафедри комп'ютерних наук
Протокол № 12 від «01» 06 20 23 р.
Завідувач кафедри
_____ Белла ГОЛУБ

«РОЗГЛЯНУТО»
Гарант ОП_«Інженерія програмного забезпечення»_
Гарант ОП
_____ Белла ГОЛУБ

РОБОЧА ПРОГРАМА НАВЧАЛЬНОЇ ДИСЦИПЛІНИ

КОНСТРУЮВАННЯ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ

Спеціальність : 121 «Інженерія програмного забезпечення»
Освітня програма: «Інженерія програмного забезпечення»
Факультет: інформаційних технологій
Розробник: старший викладач кафедри комп'ютерних наук Бородкін Г. О.

Київ 2023

1 Опис навчальної дисципліни

«КОНСТРУЮВАННЯ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ»

Галузь знань, спеціальність, освітній ступінь		
Освітній ступінь	Бакалавр	
Спеціальність	121 «Інженерія програмного забезпечення»	
Освітня програма	«Інженерія програмного забезпечення»	
Характеристика навчальної дисципліни		
Вид	Обов’язкова	
Загальна кількість годин	120	
Кількість кредитів ECTS	5	
Кількість змістових модулів	2	
Курсовий проект (робота) (за наявності)	так	
Форма контролю	*іспит*	
Показники навчальної дисципліни для денної та заочної форм навчання		
	денна форма навчання	заочна форма навчання
Курс (рік підготовки)	3	
Семестр	5	
Лекційні заняття	*30 год.*	
Практичні, семінарські заняття	-	
Лабораторні заняття	*30 год.*	
Самостійна робота	*90 год.*	
Індивідуальні завдання	-	
Кількість тижневих аудиторних годин для денної форми навчання	*4 год.*	

1 Мета, завдання та компетентності навчальної дисципліни

Метою вивчення курсу «Конструювання програмного забезпечення» є формування бази теоретичних знань та умінь щодо сучасних методів та засобів конструювання програмних систем.

Завдання дисципліни “Конструювання програмного забезпечення” - надати студентам знання в сфері реалізації задач автоматизації обробки інформації, автоматизації керування об'єктами, в тому числі, сільськогосподарськими, за допомогою комп'ютерної техніки. Такі знання майбутній спеціаліст зможе застосовувати як при подальшому навчанні, так і після отримання вищої освіти у своїй професійній діяльності.

У даній дисципліні висвітлюються питання, які виникають під час конструювання програмного забезпечення та процесів, пов’язаних із ним. У курсі розглянуто базові підходи до конструювання, різні способи керування ним та практичні аспекти, що виникають у процесі конструювання програмного забезпечення. Також увага приділяється проектуванню за допомогою «шаблонів проектування» та UML, як засобу їх візуального подання.

Набуття компетентностей:

інтегральна компетентність (ІК):

Здатність розв’язувати складні задачі і проблеми інженерії програмного забезпечення, що передбачає проведення досліджень з елементами наукової новизни та/або здійснення інновацій в умовах невизначеності вимог.

загальні компетентності (ЗК):

К05. Здатність вчитися і оволодівати сучасними знаннями.

фахові (спеціальні) компетентності (ФК):

К14. Здатність брати участь у проектуванні програмного забезпечення, включаючи проведення моделювання (формальний опис) його структури, поведінки та процесів функціонування.

К17. Здатність дотримуватися специфікацій, стандартів, правил і рекомендацій в професійній галузі при реалізації процесів життєвого циклу.

К23. Здатність реалізовувати фази та ітерації життєвого циклу програмних систем та інформаційних технологій на основі відповідних моделей і підходів розробки програмного забезпечення.

К25. Здатність обґрунтовано обирати та освоювати інструментарій з розробки та супроводження програмного забезпечення.

Програмні результати навчання (ПРН):

ПР03. Знати основні процеси, фази та ітерації життєвого циклу програмного забезпечення.

ПР04. Знати і застосовувати професійні стандарти і інші нормативно-правові документи в галузі інженерії програмного забезпечення.

ПР06. Уміння вибирати та використовувати відповідну задачі методологію створення програмного забезпечення, враховуючи специфіку природоохоронної галузі.

ПР08. Вміти розробляти людино-машинний інтерфейс.

ПР09. Знати та вміти використовувати методи та засоби збору, формулювання та аналізу вимог до програмного забезпечення.

ПР12. Застосовувати на практиці ефективні підходи щодо проектування програмного забезпечення.

ПР17. Вміти застосовувати методи компонентної розробки програмного забезпечення.

2 Програма та структура навчальної дисципліни для:

– повного терміну денної (заочної) форми навчання;

– скороченого терміну денної (заочної) форми навчання.

Назви змістових модулів і тем	Кількість годин											
	денна форма						Заочна форма					
	усього	у тому числі					усього	у тому числі				
		л	п	лаб	інд	с.р.		л	п	лаб	інд	с.р.
1	2	3	4	5	6	7	8	9	10	11	12	13
Змістовий модуль 1. Загальні відомості про конструювання програмного забезпечення												
Тема 1. Вступ до дисципліни «Конструювання програмного забезпечення»	10	2		2		6						
Тема 2. Введення в конструювання програмного забезпечення.	10	2		2		6						
Тема 3. Загальні підходи до проектування програмних систем	10	2		2		6						
Тема 4. Техніка моделювання архітектури ПЗ.	10	2		2		6						
Тема 5. Концепції сервіс-орієнтованої архітектури ПЗ.	10	2		2		6						
Тема 6. RESTful та SOAP Web-сервіси.	10	2		2		6						
Тема 7. Документування архітектури ПЗ.	10	2		2		6						
Разом за змістовим модулем 1	**70**	**14**		**14**		**42**						
Змістовий модуль 2. Керування конструюванням програмного забезпечення												
Тема 8. Методи об'єктного аналізу і моделювання.	10	2		2		6						
Тема 9. Рівнева організація прикладу.	10	2		2		6						
Тема 10. Дизайн рівня представлення.	10	2		2		6						
Тема 11. Дизайн рівня бізнес-логіки.	10	2		2		6						
Тема 12. Дизайн рівня даних.	10	2		2		6						
Тема 13. Дизайн рівня сервісів.	10	2		2		6						
Тема 14. Інтерфейси, взаємодія, еволюція програм і даних	10	2		2		6						
Тема 15. Моделі якості та надійності програмних систем	10	2		2		6						
Разом за змістовим модулем 2	**80**	**16**		**16**		**48**						

Усього годин	**150**	**30**		**30**		**90**						
Курсовий проект (робота) з ***(якщо є в робочому навчальному плані)***	194	**-**	**-**	**-**		**-**		**-**	**-**	**-**		**-**
Усього годин	**344**	**30**		**30**		**90**						

5 Теми лабораторних занять

№ з/п	Назва теми	Кількість годин
1	Тема 1. Вступ до дисципліни «Конструювання програмного забезпечення»	2
2	Тема 2. Введення в конструювання програмного забезпечення.	2
3	Тема 3. Загальні підходи до проектування програмних систем	2
4	Тема 4. Техніка моделювання архітектури ПЗ.	2
5	Тема 5. Концепції сервіс-орієнтованої архітектури ПЗ.	2
6	Тема 6. RESTful та SOAP Web-сервіси.	2
7	Тема 7. Документування архітектури ПЗ.	2
8	Тема 8. Методи об'єктного аналізу і моделювання.	2
9	Тема 9. Рівнева організація прикладу.	2
10	Тема 10. Дизайн рівня представлення.	2
11	Тема 11. Дизайн рівня бізнес-логіки.	2
12	Тема 12. Дизайн рівня даних.	2
13	Тема 13. Дизайн рівня сервісів.	2
14	Тема 14. Інтерфейси, взаємодія, еволюція програм і даних	2
15	Тема 15. Моделі якості та надійності програмних систем	2
	Всього	**30**

6 Теми самостійної роботи

№ з/п	Назва теми	Кількість годин
1	Тема 1. Вступ до дисципліни «Конструювання програмного забезпечення»	6
2	Тема 2. Введення в конструювання програмного забезпечення.	6
3	Тема 3. Загальні підходи до проектування програмних систем	6
4	Тема 4. Техніка моделювання архітектури ПЗ.	6
5	Тема 5. Концепції сервіс-орієнтованої архітектури ПЗ.	6
6	Тема 6. RESTful та SOAP Web-сервіси.	6
7	Тема 7. Документування архітектури ПЗ.	6
8	Тема 8. Методи об'єктного аналізу і моделювання.	6
9	Тема 9. Рівнева організація прикладу.	6
10	Тема 10. Дизайн рівня представлення.	6
11	Тема 11. Дизайн рівня бізнес-логіки.	6
12	Тема 12. Дизайн рівня даних.	6
13	Тема 13. Дизайн рівня сервісів.	6
14	Тема 14. Інтерфейси, взаємодія, еволюція програм і даних	6
15	Тема 15. Моделі якості та надійності програмних систем	6
	Всього	**90**

7 Зразки контрольних питань, тестів для визначення рівня засвоєння знань студентами.

Комплект тестів для визначення рівня засвоєння знань студентами знаходиться за посиланням:
https://elearn.nubip.edu.ua/course/view.php?id=2686

8 Методи навчання

При викладанні навчальної дисципліни використовуються такі методи навчання:

М1. Лекція (проблемна, інтерактивна).

М2. Лабораторна робота – для використання набутих знань до розв’язування практичних завдань.

М3. Проблемне навчання – створення проблемної ситуації для зацікавленого і активного сприйняття матеріалу.

М4. Проектне навчання (індивідуальне).

М8. Дослідницький метод.

9 Форми контролю

МК1. Тестування.
МК4. Методи усного контролю.
МК5. Екзамен.
МК7. Звіт

10 Розподіл балів, які отримують студенти. Оцінювання знань студента відбувається за 100-бальною шкалою і переводиться в національні оцінки згідно з табл. 1 «Положення про екзамени та заліки у НУБіП України» (наказ про уведення в дію від 26 квітня 2023 р. протокол № 10)

Рейтинг здобувача вищої освіти, бали	**Оцінка національна за результати складання**	
	екзаменів	**заліків**
90-100	**відмінно**	**зараховано**
74-89	**добре**	
60-73	**задовільно**	
0-59	**незадовільно**	**не зараховано**

Для визначення рейтингу студента із засвоєння дисципліни $\mathbf{R_{ДИС}}$ (до 100 балів) одержаний рейтинг з атестації $\mathbf{R_{AT}}$ (до 30 балів) додається до рейтингу студента (слухача) з навчальної роботи $\mathbf{R_{HP}}$ (до 70 балів): $\mathbf{R_{ДИС} = R_{HP} + R_{AT}}$.

11 Навчально-методичне забезпечення

1. Бородкін Г.О., Бородкіна І.Л. Інженерія програмного забезпечення. / Посібник для студентів вищих навчальних закладів – К. Центр навчальної літератури. – 2018. – 204 с.

12 Рекомендовані джерела інформації

Базова

1. Лавріщева К.М. Програмна інженерія. Підручник –К., «НАНУ» – 2008.–319 с
2. S. McConnell, Code Complete, 2nd ed., Microsoft Press, 2004.
3. D. Budgen, Software Design, 2nd ed., Addison-Wesley, 2003.
4. Guckkenheimer S., Peter J. Software Engineering With Microsoft Visual Studio. Team System. – Adison Wesley, 2006. – 273 p.
5. Бабенко Л.П., Лавріщева К.М. Основи програмної інженерії.– Навч. посібник.–К.: Знання, 2001. –269 с.

6. Лавріщева К.М. Основні напрямки досліджень в програмній інженерії і шляхи їхнього розвитку // Проблеми програмування. – 2003. – № 3–4. – С. 44–58.
7. R.S. Pressman, Software Engineering: A Practitioner's Approach, McGraw-Hill, 2010.
8. I. Sommerville, Software Engineering, 9th ed., Addison-Wesley, 2011.
9. A. Silberschatz, P.B. Galvin, and G. Gagne, Operating System Concepts, 8th ed., Wiley, 2008.
10. INCOSE, Systems Engineering Handbook: A Guide for System Life Cycle Processes and Activities, version 3.2.2, International Council on Systems Engineering, 2012.
11. S. Friedenthal, A. Morre, and R. Steiner, A Practical Guide to SysML: The Systems Modeling Language, 2nd ed., Morgan Kaufmann, 2012.

Допоміжна

1. Zave P., Jackson M. Four Dark Corners of Requirements Engineering // ACM Transactions on Software Engineering, January 1997.– № 1.
2. I. Jacobson, G. Booch, and J. Rumbaugh, The Unified Software Development Process, Addison-Wesley Professional, 1999.
3. J. Nielsen, Usability Engineering, Morgan Kaufmann, 1993.
4. G. Booch, J. Rumbaugh, and I. Jacobson, The Unified Modeling Language User Guide, Addison-Wesley, 1999.
5. E. Gamma et al., Design Patterns: Elements of Reusable Object-Oriented Software, 1st ed., Addison-Wesley Professional, 1994.

Інформаційні ресурси

1. ЕНК по дисципліні знаходиться за електронною адресою: **https://elearn.nubip.edu.ua/enrol/index.php?id=2686**
2. http://uk.wikipedia.org/wiki/Життєвий_цикл_програмного_забезпечення
3. http://uk.wikipedia.org/wiki/Методологія_розробки_програмного_забезпечення
4. http://uk.wikipedia.org/wiki/Гнучка_розробка_програмного_забезпечення
5. http://uk.wikipedia.org/wiki/Програмна_інженерія
6. IEEE Std., 12207-2008 (a.k.a. ISO/IEC 12207:2008) 'Standard for Systems and Software Engineering—Software Life Cycle Processes, IEEE, 2008.
7. IEEE, IEEE Std. 1069-2009 Standard for Information Technology—Systems Design—Software Design Descriptions, IEEE, 2009.
8. ISO/IEC/IEEE., 24765:2010 Systems and Software Engineering—Vocabulary, ISO/IEC/IEEE, 2010.
9. IEEE, "IEEE Std. 1517-2010 Standard for Information Technology—System and Software Life Cycle Processes—Reuse Processes", IEEE, 2010.

10. ISO 9000 Системи керування якістю — Основні положення і словник.
11. ISO 9000-1 Стандарти з керування якістю та забезпечення якості. Ч.1. Настанови щодо вибору та застосування.
12. ISO 9000-2 Стандарти з керування якістю та забезпечення якості. Ч.2. Настанови щодо застосування ISO 9001, ISO 9002, ISO 9003.
13. ISO 9000-3 Стандарти з керування якістю та забезпечення якості – Ч.3. Настанови щодо застосування ISO 9001 під час розроблення, постачання та супроводження програмного забезпечення.
14. ISO 9000-4 Стандарти з керування якістю та забезпечення якості – Ч.4. Настанови щодо керування програмою надійності.
15. ISO 9001-2001 Системи керування якістю. Вимоги.
16. ISO 9003 Система якості. Модель забезпечення якості в процесі контролю готової продукції та її випробуваннях.
17. ISO/IEC 9126-1:2001 Програмна інженерія. Якість продукту.Ч.1. Модель якості.
18. ISO/IEC TR 9126-2:2003 Програмна інженерія. Якість продукту.Ч.2. Зовнішні метрики
19. ISO/IEC TR 9126-3:2003 Програмна інженерія. Якість продукту.Ч.3. Внутрішні метрики.
20. ISO/IEC TR 9126-4:2004 Програмна інженерія. Якість продукту.Ч.4. Метрики якості при використанні.
21. ISO/IEC 14598-2:2000 Програмна інженерія. Оцінювання програмного продукту. Ч.2. Планування та керування.
22. ISO/IEC 14598-3:2000 Програмна інженерія. Оцінювання програмного продукту. Ч.3. Процес для розробників.
23. ISO/IEC 14598-4:1999 Програмна інженерія. Оцінювання продукту. Ч.4. Процес для замовників.
24. ISO/IEC 14598-6:2001 Програмна інженерія. – Оцінка продукту – Ч.6. Документація модулів оцінювання.
25. ISO/IEC 19761:2003 Програмна інженерія. COSMIC-FFP. Метод
26. ISO/IEC 25000:2005 Програмна інженерія. Оцінювання і вимоги до якості програмного продукту(SQuaRE). Настанова щодо оцінювання і вимоги до якості програмного продукту.
27. ISO/IEC 25001:2007 Програмна інженерія. Оцінювання і вимоги до якості програмного продукту(SQuaRE). Планування та керування.
28. ISO/IEC 25020:2007 Програмна інженерія. Оцінювання і вимоги до якості програмного продукту(SQuaRE). Настанова та еталонна модель вимірювання.
29. ISO/IEC TR 25021:2007 Програмна інженерія. Оцінювання і вимоги до якості програмного продукту(ОВЯПП - SQuaRE). Елементи вимірювання якості.

30. ISO/IEC 25030:2007 Програмна інженерія. Оцінювання і вимоги до якості програмного продукту(SQuaRE). Вимоги якості.
31. ISO/IEC 25051:2006 Програмна інженерія. Оцінювання і вимоги до якості програмного продукту(SQuaRE). Вимоги до якості готових комерційних продуктів програмних засобів та інструкції по тестуванню.
32. ISO/IEC 25062:2006 Програмна інженерія. Оцінювання і вимоги до якості програмного продукту(SQuaRE). Загальний промисловий формат, який використовується при тестуванні звітів.
33. IEEE, "IEEE Std. 1517-2010 Standard for Information Technology—System and Software Life Cycle Processes—Reuse Processes", IEEE, 2010.
34. IEEE, "IEEE Std. 12207-2008 (a.k.a. ISO/IEC 12207:2008) Standard for Systems and Software Engineering—Software Life Cycle Processes", IEEE, 2008.
35. ДСТУ ISO/IEC TR 9126-2 Програмна інженерія. Якість продукту.Ч.2 Зовнішні метрики.
36. ДСТУ ISO/IEC TR 9126-3 Програмна інженерія. Якість продукту.Ч.3 Внутрішні метрики.
37. ДСТУ ISO/IEC TR 9126-4 Програмна інженерія. Якість продукту.Ч.4 Метрики якості при використанні.
38. ДСТУ ISO/IEC 15939 Інженерія систем і програмних засобів. Процес вимірювання.