

Міністерство освіти і науки України
Харківський національний університет імені В.Н. Каразіна
Кафедра штучного інтелекту та програмного забезпечення

“ЗАТВЕРДЖУЮ”

Проректор
з науково-педагогічної роботи

Олександр ГОЛОВКО

_____ 2022 р.



РОБОЧА ПРОГРАМА НАВЧАЛЬНОЇ ДИСЦИПЛІНИ

Технологія створення програмних продуктів

рівень вищої освіти	<u>перший (бакалаврський) рівень</u>
галузь знань	<u>12 Інформаційні технології</u>
спеціальність	<u>122 Комп'ютерні науки</u>
освітня програма	<u>Комп'ютерні науки</u>
спціалізація	_____
вид дисципліни	<u>обов'язкова</u>
факультет	<u>комп'ютерних наук</u>

Програму рекомендовано до затвердження Вченою радою факультету комп'ютерних наук
«29» серпня 2022 року, протокол № 14

РОЗРОБНИКИ ПРОГРАМИ:

Старший викладач кафедри штучного інтелекту та програмного забезпечення **Горбань
Анатолій Михайлович**

Програму схвалено на засіданні кафедри штучного інтелекту та програмного забезпечення
Протокол від «29» серпня 2022 року № 1

Завідувач кафедри штучного інтелекту та програмного забезпечення


Володимир КУКЛІН

Програму погоджено з гарантом освітньої програми «Комп'ютерні науки»

Гарант освітньої програми «Комп'ютерні науки»


Сергій БОГУЧАРСЬКИЙ

Програму погоджено науково-методичною комісією факультету комп'ютерних наук

Протокол від «29» серпня 2022 року № 1

Голова науково-методичної комісії факультету комп'ютерних наук


Анатолій БЕРДНІКОВ

ВСТУП

Програма навчальної дисципліни “Технологія створення програмних продуктів” складена відповідно до освітньо-професійної програми підготовки (бакалаврського) рівня вищої освіти, спеціальності 122 Комп'ютерні науки.

1. Опис навчальної дисципліни

1.1. Основною метою є надання уявлення про розробку ПС як промисловий процес, що базується на системі інженерних принципів та технологічних процесах, орієнтованих на досягнення максимальної продуктивності середніх та великих груп розробників; засвоєння теоретичних та практичних знань та навичок в області промислової розробки ПС.

1.2. Формування у студентів знань про парадигми процесів розробки програмного забезпечення, організацію процесу управління проектом, методи забезпечення та оцінки якості програм, технологічні процеси етапів аналізу, проектування, тестування та супроводження ПС.

Студенти мають оволодіти знаннями базових понять, принципів, методів та процедур прикладного системного аналізу.

В ході вивчення дисципліни у студента повинні формуватися наступні компетентності.

Загальні компетентності (ЗК).

- ЗК1. Здатність до абстрактного мислення, аналізу та синтезу.
- ЗК2. Здатність застосовувати знання у практичних ситуаціях.
- ЗК3. Знання та розуміння предметної області та розуміння професійної діяльності.
- ЗК6. Здатність вчитися й оволодівати сучасними знаннями.
- ЗК7. Здатність до пошуку, оброблення та аналізу інформації з різних джерел.
- ЗК8. Здатність генерувати нові ідеї (креативність).
- ЗК9. Здатність працювати в команді.
- ЗК10. Здатність бути критичним і самокритичним.
- ЗК11. Здатність приймати обґрунтовані рішення.
- ЗК12. Здатність оцінювати та забезпечувати якість виконуваних робіт.

Спеціальні (фахові, предметні) компетентності (ФК)

- ФК3. Здатність до логічного мислення, побудови логічних висновків, використання формальних мов і моделей алгоритмічних обчислень, проектування, розроблення й аналізу алгоритмів, оцінювання їх ефективності та складності, розв'язності та нерозв'язності алгоритмічних проблем для адекватного моделювання предметних областей і створення програмних та інформаційних систем.
- ФК8. Здатність проектувати та розробляти програмне забезпечення із застосуванням різних парадигм програмування: узагальненого, об'єктно-орієнтованого, функціонального, логічного, з відповідними моделями, методами й алгоритмами обчислень, структурами даних і механізмами управління.

1.3. Кількість кредитів – 4

1.4. Загальна кількість годин – 120

1.5. Характеристика навчальної дисципліни	
Нормативна / за вибором	
денна форма навчання	Заочна (дистанційна) форма навчання
Рік підготовки	
4-й	
Семестр	
7-й	
Лекції	
32 год.	_____ год.
Практичні, семінарські заняття	
_____ год.	_____ год.
Лабораторні заняття	
32 год.	_____ год.
Самостійна робота	
56 год.	_____ год.
В т.ч. індивідуальні завдання	
5 год.	_____ год..

1.6. Заплановані результати навчання

знати:

- виявлення та встановлення вимог до системи;
- вибір методів та засобів аналізу систем.

вміти:

- розрахунку проектних метрик та оцінки на їх основі параметрів проекту,
- планування типових проектних завдань,
- структурування і модульної оптимізації ПС,
- планування і розробки тестових варіантів.

В результаті вивчення дисципліни у студента повинні формуватися наступні програмні результати навчання (ПРН).

- ПРН 5. Проектувати, розробляти та аналізувати алгоритми розв'язання обчислювальних та логічних задач, оцінювати ефективність та складність алгоритмів на основі застосування формальних моделей алгоритмів та обчислюваних функцій.
- ПРН 7. Розуміти принципи моделювання організаційно-технічних систем і операцій; використовувати методи дослідження операцій, розв'язання одно- та багатокритеріальних оптимізаційних задач лінійного, цілочисельного, нелінійного, стохастичного програмування.

2. Тематичний план навчальної дисципліни

Розділ 1. Організація процесу розробки ПС

Тема 1. Основні поняття.

Визначення технології розробки програмного забезпечення. Проблеми надійності ПЗ та моделі якості процесів розробки.

Тема 2. Парадигми розробки

Парадигми класичного життєвого циклу та макетування.

Тема 3. Стратегії.

Поняття стратегії розробки програмного забезпечення. Стратегії на основі класичного життєвого циклу — інкрементна модель та модель швидкої розробки (RAD).

Тема 4. Еволюційні моделі розробки

Моделі розробки на основі макетування — спіральна модель, компонентно-орієнтована модель та XP-процес

Розділ 2. Керівництво програмними проектами

Тема 1. Процес керівництва проектом

Завдання та роль процесу керівництва на різних етапах проекту, роль вимірювань, та метрик. Оцінка проекту, аналіз ризиків, трасування та контроль.

Тема 2. Планування

Планування проекту та планування проектних задач у відповідності до вибраної парадигми розробки.

Тема 3. LOC-метрики

Розмірно-орієнтовані метрики — їх розрахунок та виконання оцінки проекту на їх основі. Конструктивна модель вартості COSOMO та її використання.

Тема 4. FP-метрики

Функціонально-орієнтовані метрики — їх розрахунок та виконання оцінки проекту на їх основі.

Розділ 3. Основи проектування ПС

Тема 1. Особливості етапу проектування

Особливості процесу синтезу ПС, основні завдання підетапів: структурування системи, моделювання управління, декомпозиція підсистем.

Тема 2. Характеристики модульної структури ПС

Модульність, інформаційна закритість. Метрики зв'язності та зчеплення.

Тема 3. Класичні методи проектування

Метод структурного проектування, використання шаблонів проектування та компонентний підхід.

Тема 4. Складність ПС

Поняття складності системи, метрики складності та характеристики ієрархічної структури системи.

Розділ 4. Тестування ПС

Тема 1. Основні поняття.

Основні поняття, принципи та види тестування програмного забезпечення.

Тема 2. Структурне тестування

Особливості тестування “білого ящика”, способи тестування базового шляху, умов, потоків даних та циклів.

Тема 3. Функціональне тестування

Особливості тестування “чорного ящика”, способи розбиття по еквівалентності, аналізу граничних умов, діаграм причин-наслідків.

Тема 4. Організація процесу тестування

Методика тестування елементів ПС, інтеграції, правильності та системне тестування.

3. Структура навчальної дисципліни

Назви розділів і тем	Кількість годин					
	денна форма					
	усьог о	у тому числі				
л		п	лаб.	інд.	с. р.	
1	2	3	4	5	6	7
Розділ 1. Організація процесу розробки ПС						
Тема 1. Основні поняття	4	2				2
Тема 2. Парадигми розробки	6	2				4
Тема 3. Стратегії	6	2				4
Тема 4. Еволюційні моделі розробки	6	2				4
Разом за розділом 1	22	8				14
Розділ 2. Керівництво програмними проектами						
Тема 1. Процес керівництва проектом	6	2	2			2
Тема 2. Планування	8	2	2			4
Тема 3. LOC-метрики	10	2	4			4
Тема 4. FP-метрики	10	2	4			4
Разом за розділом 2	34	8	12			14
Розділ 3. Основи проектування ПС						
Тема 1. Особливості етапу проектування	7	2	1			4
Тема 2. Характеристики модульної структури ПС	8	2	2			4
Тема 3. Класичні методи проектування	7	2	1			4
Тема 4. Складність ПС	8	2	2			4
Разом за розділом 3	30	8	6			16
Розділ 4. Тестування ПС						
Тема 1. Основні поняття	8	2	4			2
Тема 2. Структурне тестування	6	2	2			2
Тема 3. Функціональне тестування	10	2	4			4
Тема 4. Організація процесу тестування	10	2	4			4
Разом за розділом 4	34	8	14			12
Усього годин	120	32	32			56

4. Теми практичних (лабораторних) занять

№ з/п	Назва теми	Кількість годин
Розділ 2. Керівництво програмними проектами		
1	Оцінка проекту на основі LOC-оцінок	2
2	Оцінка проекту за допомогою конструктивної моделі вартості	2
3	Розрахунки показників зв'язності програм	4
4	Розрахунки показників зчеплення програм	4
Розділ 3. Основи проектування ПС		
5	Оцінки характеристик ієрархічної структури програм	6
Розділ 4. Тестування ПС		
6	Підрахунок показника цикломатичної складності для різних мов програмування	4
7	Формування тестових варіантів по способу тестування базового шляху	2
8	Формування тестових варіантів по способу тестування потоків даних	2
9	Формування тестових варіантів по способу розбиття по еквівалентності	2
10	Формування тестових варіантів по способу аналізу граничних значень	4
Разом:		32

5. Завдання для самостійної роботи

№ з/п	Види, зміст самостійної роботи	Кількість годин
Розділ 1. Організація процесу розробки ПС		
1	Знайомство з основними положеннями дисципліни - за джерелами з списку рекомендованої літератури	14
Розділ 2. Керівництво програмними проектами		
2	Вивчення методології використання моделі СОСОМО II для етапів композиції ПС, раннього етапу проектування, етапу постархітектури та аналізу чутливості проекту.	14
Розділ 3. Основи проектування ПС		
3	Напрацювання навичок оцінки метрик якості модулів на основі аналізу кодів відкритого ПЗ.	16
Розділ 4. Тестування ПС		
4	Планування тестових варіантів та планів тестування власних (в т. ч. кваліфікаційних) проектів.	12
Разом		56

6. Індивідуальні завдання

2 контрольні роботи

7. Методи навчання

На досягнення освітніх цілей спрямовані такі методи навчання студентів:

– практичні (використовують для пізнання дійсності, формування навичок і вмінь, поглиблення знань. Під час їх застосування використовуються такі прийоми: планування

виконання завдання, постановка завдання, оперативне стимулювання, контроль і регулювання, аналіз результатів, визначення причин недоліків);

– пояснювальне-ілюстративний (використовують для викладання й засвоєння нового навчального матеріалу, фактів, підходів, оцінок, висновків тощо);

– репродуктивний (для застосування студентами вивченого на основі зразка або правила, алгоритму, що відповідає інструкціям, правилам, в аналогічних до представленого зразка ситуаціях);

Як правило лекційні та практичні заняття проводяться аудиторне. В умовах дії карантину заняття проводяться відповідно до Наказу ректора Харківського національного університету імені В.Н. Каразіна (аудиторне або дистанційно за допомогою платформ Google Meet або Zoom).

8. Методи контролю

На протязі усього терміну викладання означеної дисципліни проводиться поточний контроль засвоєння лекційного матеріалу (контроль знань) та контроль здобуття практичних навиків (контроль вмінь). Підсумковий семестровий контроль також дозволяє контролювати як одержані знання так і набуті вміння:

- Після закінчення викладання кожного модуля курсу контроль знань проводиться у вигляді тестування по матеріалам розділа.
- Після викладання курсу здійснюється підсумковий контроль.

9. Схема нарахування балів

Розподіл балів для підсумкового семестрового контролю в формі заліку без виконання залікової роботи

Поточний контроль та самостійна робота					
Розділ 1	Розділ 2	Розділ 3	Розділ 4	Контрольні роботи, передбачені навчальним планом	Разом
20	20	20	20	20	100

Критерії оцінювання

Контрольні роботи

Контрольна робота складається з 10 питань з відкритою множиною відповідей. Кожна відповідь оцінюється в 0 балів (невірна або відсутня), 0,5 бала (часткова чи неточна) чи 1 бал (вичерпна).

Практикум

Практичні заняття з курсу організовані як виконання ІТ проекту (в частині аналізу) групою студентів в складі до 10 осіб з зазначенням вкладу кожного члена команди в виконання етапів роботи. За результатами роботи формується фінальний звіт що включає також робочу документацію проекту – протоколи інтерв'ю, специфікації вимог, сценарії, моделі рамок системи, структур даних, бізнес-прецедентів та бізнес-класів. Фінальний звіт рецензується викладачем на відповідність стандартним (в ІТ індустрії) вимогам до проектної документації та за результатами оцінюється в обсязі від 0 до 40 балів. Оцінка кожного члена команди обраховується за формулою

$$\text{ОЦінд} = \text{ОЦзвіт} * \text{КС} * \text{КУінд} * \text{ППінд},$$

Де ОЦінд – оцінка студента за практикум (при ОЦінд >40 приймається 40), ОЦзвіт – оцінка фінального звіту (від 0 до 40), КС – кількість студентів в команді, КУінд – коефіцієнт «трудової участі» (вкладу студента в виконання проекту в цілому в діапазоні від 0 до 1) встановлюється спільно самими членами команди таким чином щоб в сумі на всю команду набиралось значення 1 (тобто 100 %), ППінд – показник присутності студента на аудиторних заняттях як відношення «був присутній» до «всього занять» (від 0 до 1).

Так, наприклад, для групи з 9 студентів при оцінці фінального звіту в 32 бали студент з КУінд = 15 % що пропустив половину занять (ППінд = 0,5) отримає оцінку $OЦінд = 32 * 9 * 0,15 * 0,5 = 21,6 \approx 22$ бали, а маючи КУінд = 7 % і ППінд = 1,0 отримає $OЦінд \approx 20$ балів і, нарешті, при вкладі в спільну роботу над проектом 20 % пропустивши при цьому четверть занять матиме $OЦінд = 43,2 \approx 40$ балів.

Екзамен

Проводиться в письмовій формі і полягає в наданні стислих та вичерпних відповідей що не мають явних ознак плагіату на два тематичних питання вибраного білету. Орієнтовний об'єм кожної відповіді складає одну-дві рукописні сторінки. Кожна відповідь оцінюється максимум в 20 балів. Більш детальні критерії:

Оцінка в балах	Пояснення
15 – 20	Відповідь на питання надана повністю, відмінна робота без помилок або з однією-двома незначними помилками.
10 – 14	Відповідь на питання надана повністю, робота з декількома незначними помилками, або з однією – двома значними помилками.
5 – 9	Відповідь на питання надана не повністю, але прогалини не носять істотного характеру, необхідний об'єм знань матеріалу у студента в основному сформований, робота з трьома і більше значними помилками
0–4	Відповідь не надавалась або відповідь не має відношення до поставленого питання.

Шкала оцінювання

Сума балів за всі види навчальної діяльності протягом семестру	Оцінка	
	для чотирирівневої шкали оцінювання	для дворівневої шкали оцінювання
90 – 100	відмінно	зараховано
70-89	добре	
50-69	задовільно	
1-49	незадовільно	не зараховано

10. Рекомендована література

Основна література

1. Sommerville I. Software Engineering. .: Addison-Wesley, 2001.
2. Rumbaugh J. R., Blaha M. R. Object-Oriented Modeling and Design with UML. 2 ed. : Prentice Hall PTR, 2005.
3. Maciaszek L.A. Requirements Analysis and System Design, 3rd ed. : Addison-Wesley, 2007.
4. Maciaszek, L.A. and Liong, B.L. Practical Software Engineering. A Case-Study Approach, Addison-Wesley, 2005.

Допоміжна література

1. Larman C. Applying UML and Patterns. 2 ed. : Prentice Hall PTR, 2002.
2. Ambler, S. W. The Object Primer. 2nd ed. Cambridge University Press, 2001.
3. Beck, K, Fowler, M. Planning Extreme Programming. Addison-Wesley, 2001.
4. Booch, G. Object-Oriented analysis and design. 2 ed. Addison-Wesley, 1994.
5. Graham, I. Object-Oriented Methods. Principles & Practice. 3 ed. Addison-Wesley, 2001.
6. Jacobson, I., Booch, G., Rumbaugh, J. The Unified Software Development Process. Addison-Wesley, 1999.
7. Yourdon, E., and Constantine, L. Structured Design: fundamentals of a discipline of computer program and systems design. Englewood Cliffs, NJ: Prentice-Hall, 1979.
8. K. Beck. Test-Driven Development by Example. : Addison-Wesley, 2003.

11. Посилання на інформаційні ресурси в Інтернеті, відео-лекції, інше методичне забезпечення

1. Software Engineering Body of Knowledge: [Електронний ресурс] // 2005. URL: http://www.swebok.org/ironman/pdf/SWEBOK_Guide_2004.pdf
2. F. Bachmann, L. Bass, C. Buhman, S. Comella-Dorda, F. Long, J. Robert, R. Seacord, K. Wallnau. Volume II: Technical Concepts of Component-Based Software Engineering.: [Електронний ресурс] // 2nd Edition/ Technical Report CMU/SEI-2000-TR-008. URL: <http://www.sei.cmu.edu/pub/documents/00.reports/pdf/00tr008.pdf>
3. G. Alonso, F. Casati, H. Kuno, V. Machiraju. Web Services. Concepts, Architectures and Applications: [Електронний ресурс] // Springer-Verlag, 2004. URL: <http://www.inf.ethz.ch/personal/alonso/WebServicesBook>
4. Capability Maturity Model Integration (CMMI), Version 1.1. CMMI for Systems Engineering, Software Engineering, Integrated Product and Process Development, and Supplier Sourcing (CMMI-SE/SW/IPPD/SS, V1.1). Staged Representation.: [Електронний ресурс] // SEI Technical Report CMU/SEI-2002-TR-012, Software Engineering Institute, Pittsburgh, March 2002. URL: <http://www.sei.cmu.edu/pub/documents/02.reports/pdf/02tr012.pdf>
5. Портал про автоматизоване тестування ПЗ: [Електронний ресурс] // URL: <http://automated-testing.info>.