

Міністерство освіти і науки України
Харківський національний університет імені В.Н. Каразіна

Кафедра моделювання систем і технологій

«ЗАТВЕРДЖУЮ»

Проректор з
науково-педагогічної роботи



А.В.Пантелеймонов

2019 року

РОБОЧА ПРОГРАМА НАВЧАЛЬНОЇ ДИСЦИПЛІНИ
«ПРОЕКТУВАННЯ ІНФОРМАЦІЙНИХ СИСТЕМ»

рівень вищої освіти	перший (бакалаврський)
галузь знань	12 Інформаційні технології
напрямок підготовки	122 Комп'ютерні науки
освітня програма	Комп'ютерні науки
спеціалізація	
вид дисципліни	Обов'язкова
факультет	Комп'ютерних наук

2019/2020 навчальний рік

Програму рекомендовано до затвердження вченою радою факультету комп'ютерних наук

Протокол від « 27 » червня 2019 року № 2

РОЗРОБНИК ПРОГРАМИ: кандидат технічних наук, доцент кафедри моделювання систем і технологій Гамзаєв Рустам Олександрович

Програму схвалено на засіданні кафедри моделювання систем і технологій

Протокол від « 30 » травня 2019 року № 15

Завідувач кафедри моделювання систем і технологій



М. В. Ткачук

Програму погоджено методичною радою факультету комп'ютерних наук

Протокол від « 20 » червня 2019 року № 9

Голова методичної комісії



А. Г. Бердніков

ВСТУП

Програма навчальної дисципліни «Проектування Інформаційних систем» складена відповідно до освітньо-професійної програми підготовки першого (бакалаврського) рівня за спеціальністю 122 Комп'ютерні науки.

1. Опис навчальної дисципліни

1.1. Мета викладання навчальної дисципліни

Метою вивчення курсу «Проектування Інформаційних систем» є формування у студентів теоретичних знань та практичних навичок із аналізу, моделювання, проектування сучасних інформаційних систем з використанням об'єктно орієнтованих методів, типових шаблонів для розробки програмних систем, а також використання типових архітектурних рішень

1.2. Основні завдання вивчення дисципліни:

- ознайомлення з принципами побудови сучасних інформаційних систем;
- формування у студентів знань за дисципліною, пов'язаних з процесом створення інформаційних систем та методами аналізу програмних систем;
- ознайомлення та вивчення понять шаблонів проектування програмного забезпечення та їх використання для підвищення якості архітектури систем;
- набуття практичних навичок роботи в колективі, вмінь знаходити правильні рішення на всіх етапах життєвого циклу програмних продуктів, використання сучасних інструментальних засобів та методологічних підходів;
- викладення основних стилів розробки програмної архітектури.

1.3. Кількість кредитів - 7

1.4. Загальна кількість годин - 210

1.5. Характеристика навчальної дисципліни.

1.5. Характеристика навчальної дисципліни	
Нормативна / за вибором	
Денна форма навчання	Денна форма навчання
Рік підготовки	
3-й	3-й
Семестр	
5-й	6-й
Лекції	
32 год.	год.
Практичні заняття	
год.	год.
Лабораторні заняття	
32 год.	год.
Самостійна робота	
146 год.	год.
Індивідуальні завдання	

1.6. Заплановані результати навчання.

Згідно з вимогами освітньо-професійної програми студенти повинні досягти результатів навчання.

У результаті вивчення даного курсу студент повинен **знати**: існуючі життєві цикли програмних систем та пов'язані з цим процеси розробки інформаційних систем; атрибути якості програмного забезпечення, типові програмні архітектури інформаційних систем, шаблони проектування (design patterns).

У результаті вивчення даного курсу студент повинен **вміти**: використовувати інструментальні, технологічні та технічні CASE засоби для проектування, інформаційних систем; аналізувати вимоги до програмних систем та будувати модельні та технологічні рішення з урахуванням потреб кінцевих користувачів; виконувати розрахунки оцінки ефективності та продуктивність експлуатаційних показників мережі.

2. Тематичний план навчальної дисципліни.

Розділ 1. Етапи та процеси побудови інформаційних систем

Тема 1.1 «Етапи та процеси розробки програмних продуктів». Загальні відомості про основні етапи розробки програмних продуктів та існуючі методології такі як

водоспадна, ітеративна (спіральна, інкрементна), уніфікований процеси розробки програмних продуктів, гнучкі методології тощо.

Тема 1.2 «Аналіз та опрацювання вимог». Вимоги в різних моделях життєвого циклу програмного забезпечення їх класифікація, роль, місце і загальна схема роботи з ними. Основні способи виявлення, аналізу та специфікації вимог. Основні функції та можливості системи управління вимогами.

Тема 1.3 «Мова UML». Мова UML її структура та можливості використання. UML діаграми концептуального рівня моделювання ПЗ. Основні елементи та правила застосування. CASE засоби для автоматизування процесу проектування.

Тема 1.4 «Аналіз структури системи». Структурні UML діаграми, діаграма класів, діаграма пакетів, діаграма розгортання та інші, принципи відображення діаграми класів на похідний код.

Тема 1.5 «Визначення програмної архітектури». Основні поняття програмної архітектури. Способи опису програмної архітектури. Типові архітектурні рішення: клієнт – серверна архітектура, багаторівнева архітектура, розподілена архітектура.

Тема 1.6 «Стилі та тактики програмної Архітектури». Визначення поняття стилів та тактик архітектури програмного забезпечення. Детальний опис тактик модифікації, ефективності, надійності та інших.

Тема 1.7 «Представлення бізнес логіки». Основні підходи та принципи для проектування бізнес логіки, використання підходів сценарій транзакцій, доменна модель, таблична модель.

Тема 1.8 «Інверсія контролю». Основні поняття інверсії контролю та приклад використання підходу для зменшення кількості зав'язків та підвищення рівня модифікації програмних систем. Визначення основних принципів Аспектно – орієнтованого програмування та використання для вирішення завдання локалізації наскрізної функціональності.

Розділ 2. Принципи та шаблони проектування

Тема 2.1 «Шаблони архітектури програмного забезпечення». Поняття шаблону та їх опис та класифікація. Введення поняття шаблону архітектури та приклади, на

прикладі багаторівневої архітектури визначення кожного рівня та принципів їх побудови.

Тема 2.2 «GRASP шаблони». Поняття GRASP шаблонів та їх використання для створення високоякісного об'єктно орієнтованого програмного забезпечення.

Тема 2.3 «Шаблони проектування програмного забезпечення».

Визначення 23х GOF шаблонів, опис структури кожного шаблону, та приклади їх використання.

3. Структура навчальної дисципліни

Назви розділів і тем	Кількість годин					
	денна форма					
	усього	у тому числі				
л		п	лаб.	інд.	с. р.	
1	2	3	4	5	6	7
Розділ 1. Етапи та процеси побудови інформаційних систем						
Тема 1.1 «Етапи та процеси розробки програмних продуктів».	20	2				18
Тема 1.2 «Аналіз та опрацювання вимог».	18	4		4		10
Тема 1.3 «Мова UML».	21	2		8		11
Тема 1.4 «Аналіз структури системи».	14	4				10
Тема 1.5 «Визначення програмної архітектури».	21	4		4		13
Тема 1.6 «Стили та тактики програмної Архітектури».	19	2				17
Тема 1.7 «Представлення бізнес логіки».	14	2				12
Тема 1.8 «Інверсія контролю».	16	2				14
Разом за розділом 1	143	22		16		105
Розділ 2. Принципи та шаблони проектування						
Тема 2.1 «Шаблони архітектури програмного забезпечення».	26	4		6		16
Тема 2.2 «GRASP шаблони».	16	2		4		10
Тема 2.3 «Шаблони проектування програмного забезпечення».	25	4		6		15
Разом за розділом 2	67	10		16		41
Усього годин	210	32		32		146

4. Теми семінарських (практичних, лабораторних) занять

№ з/п	Назва теми	Кількість годин
1	Аналіз та опрацювання вимог	4
2	Використання мови UML 2.0	8
3	Типові програмні архітектури	4
4	Принципи SOLID та шаблони GRASP	4
5	Шаблони проектування GoF (твірні та структурні)	6
6	Шаблони проектування GoF (поведінки)	6
Разом		32

5. Завдання для самостійної роботи

№ з/п	Види, зміст самостійної роботи	Кількість годин
1	Етапи та процеси розробки програмних продуктів . <i>Загальні відомості про основні етапи розробки програмних продуктів та існуючі методології такі як водоспадна, ітеративна (спіральна, інкрементна).</i>	18
2	Аналіз та опрацювання вимог. <i>Вимоги в різних моделях життєвого циклу програмного забезпечення їх класифікація, роль, місце і загальна схема роботи з ними.</i>	10
3	Мова UML Використання мови UML 2.0 UML діаграми концептуального рівня моделювання ПЗ.	11
4	Аналіз структури системи. <i>Структурні UML діаграми, діаграма класів, діаграма пакетів, діаграма розгортання та інші, принципи відображення діаграми класів на похідний код.</i>	10
5	Визначення програмної архітектури. <i>Типові архітектурні рішення: клієнт – серверна архітектура, багаторівнева архітектура, розподілена архітектура.</i>	13
6	Стили та тактики програмної Архітектури. <i>Детальний опис тактик модифікації, ефективності, надійності та інших.</i>	17
7	Представлення бізнес логіки. <i>Основні підходи та принципи для проектування бізнес логіки, використання підходів сценарій транзакцій, доменна модель, таблична модель.</i>	12
8	Інверсія контролю. <i>Основні поняття інверсії контролю та приклад використання підходу для зменшення кількості зав'язків та підвищення рівня модифікації програмних систем.</i>	14
9	Шаблони архітектури програмного забезпечення. <i>Поняття шаблону та їх опис та класифікація.</i>	16
10	GRASP шаблони. <i>Поняття GRASP шаблонів та їх використання для створення високоякісного об'єктно орієнтованого програмного забезпечення.</i>	10
11	Спроекувати та запрограмувати задану систему використовуючи принципи та шаблони проектування.	15
Разом		146

6. Індивідуальні завдання

Курсова робота, контрольна робота

7. Методи контролю

Протягом навчального семестру проводиться поточний контроль знань, який складається з виконання лабораторних робіт, контрольної роботи та курсової роботи. Загальна сума балів, яку студент може набрати, складає – 60 балів.

Максимальна оцінка за лабораторну роботу – 3 бали.

Максимальна оцінка за контрольну роботу – 12 балів.

Максимальна оцінка за курсову роботу – 30 балів.

Підсумковий контроль – екзамен у письмовій формі.

Допуск до складання екзамену студент отримує, якщо він виконав усі лабораторні роботи, склав контрольну роботу та захистив курсову роботу з сумарною оцінкою не менше 50 балів . В іншому випадку студент не допускається до складання екзамену.

Екзамен складається з чотирьох питань (два теоретичних та два практичних).

Максимальна кількість балів за екзамен – 40 балів.

8. Схема нарахування балів

Поточний контроль та самостійна робота						Контрольна робота	Курсова робота	Разом	Екзамен	Сума
ЛР 1	ЛР 2	ЛР 3	ЛР 4	ЛР 5	ЛР 6	12	30	60	40	100
3	3	3	3	3	3					

Критерії поточної оцінки знань студентів (лабораторна робота, крок оцінювання 1 бал)

Кількість балів	Критерії оцінки
0	Студент не володіє навчальним матеріалом та не розуміє змісту теоретичних питань
1	Студент має початковий рівень знань, може виконати практичне лабораторне завдання, але з великою кількістю помилок; фрагментарно викладає окремі питання.
2	Студент достатньо повно володіє навчальним матеріалом; може пояснити основні процеси, що відбуваються під час роботи інформаційної системи та наводити власні приклади на підтвердження деяких тверджень; практично без помилок виконує практичні лабораторні завдання.
3	Студент в повному обсязі володіє навчальним матеріалом; вміє вільно використовувати нові інформаційні технології для поповнення власних знань та розв'язування задач; без помилок виконує практичні лабораторні завдання.

**Критерії поточної оцінки знань студентів
(контрольна робота, крок оцінювання 2 бали)**

Кількість балів	Критерії оцінки
2	Студент демонструє фрагментарні знання при незначному загальному їх обсязі (менше половини навчального матеріалу).
4	Студент демонструє, значну (більше половини) частину навчального матеріалу може відтворити репродуктивно; може викладача виконати просте навчальне завдання; має елементарні, нестійкі навички необхідні для виконання завдань.
6	Студент знайомий з основними поняттями навчального матеріалу; може самостійно відтворити значну частину навчального матеріалу і робити певні узагальнення; вміє виконати просте навчальне завдання.
8	Студент демонструє вивчений матеріал у стандартних ситуаціях; пояснює основні процеси, що відбуваються під час роботи інформаційної системи та наводить власні приклади на підтвердження деяких тверджень; вміє виконувати навчальні завдання.
10	Студент демонструє міцні знання, самостійно визначає проміжні цілі власної навчальної діяльності, оцінює нові факти, явища; вміє самостійно знаходити додаткові відомості та використовує їх для реалізації поставлених перед ним навчальних цілей, судження його (її) логічні і достатньо обґрунтовані; має певні навички управління інформаційною системою.
12	Студент демонструє стійкі системні знання та продуктивно їх використовує; вміє вільно використовувати нові інформаційні технології для поповнення власних знань та розв'язування задач; має стійкі навички управління інформаційною системою у нестандартних ситуаціях.

**Критерії поточної оцінки знань студентів
(курсова робота, крок оцінювання 10 бали)**

Кількість балів	Критерії оцінки
0	Студент демонструє фрагментарні знання при незначному загальному їх обсязі (менше половини навчального матеріалу).
10	Студент демонструє, значну (більше половини) частину роботи; пояснює елементарні процеси, відображені при моделюванні інформаційної системи
20	Студент демонструє роботу у повному обсязі, але з незначними помилками сит; пояснює основні процеси, відображені при моделюванні інформаційної системи та наводить власні приклади на підтвердження деяких тверджень;
30	Студент демонструє роботу у повному обсязі; пояснює основні процеси, відображені при моделюванні інформаційної системи та наводить власні приклади на підтвердження деяких тверджень;

**Критерії підсумкової оцінки знань студентів
(кожне питання екзамену, крок оцінювання 2 бали)**

Кількість балів	Критерії оцінки
2	Студент розпізнає окремі об'єкти, явища і факти предметної галузі; знає і виконує базові технологічні застосування.
4	Студент має фрагментарні знання при незначному загальному їх обсязі (менше половини навчального матеріалу) за відсутності сформованих умінь та навичок.
6	Студент має рівень знань вищій, ніж початковий; може з допомогою викладача

	відтворити значну частину навчального матеріалу з елементами логічних зв'язків; має стійкі навички виконання елементарних технологічних застосувань та їх опрацювання.
8	Студент вільно володіє навчальним матеріалом, застосовує знання на практиці; вміє узагальнювати і систематизувати навчальну інформацію; самостійно виконує передбачені програмою навчальні завдання; самостійно знаходить і виправляє допущені помилки; може аргументовано обрати раціональний спосіб виконання навчального завдання.
10	Студент має стійкі системні знання та продуктивно їх використовує, стійкі навички керування інформаційною системою в нестандартних ситуаціях; уміє вільно використовувати нові інформаційні технології для поповнення власних знань та розв'язування задач.

Шкала оцінювання

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

9. Рекомендована література

Основна література

1. Рудаков А.В., Федорова Г.Н. Технология разработки программных продуктов. Практикум. – М.: Академия, 2014. – 192 с.
2. Лаврищева К.М. Електронний підручник «Програмна інженерія» Київського національного університету ім. Т. Г. Шевченка [Електронний ресурс] – Режим доступа: <http://cyb.univ.kiev.ua/library/books/lavrishcheva-6.pdf>
3. Э. Гамма, Р. Хелм, Р. Джонсон, Дж. Валсидес. Приемы объектно-ориентированного проектирования. Паттерны проектирования. – СПб.: Питер, 2001. – 366 с.
4. Мартин Фаулер, Дейвид Райс, Мэттью Фоммел, Эдвард Хайет, Роберт Ми, Рэнди Стаффорд. Шаблоны корпоративных приложений. – М.: Вильям, 2009. – 544 с.
5. Крэг Ларман. Применение UML 2.0 и шаблонов проектирования. Введение в объектно-ориентированный анализ, проектирование и итеративную разработку. – М.: Вильям, 2013. – 736 с

Додаткова література

1. Ткачук Н.В и др. Архитектуры, модели и технологии программного обеспечения информационно-управляющих систем. – Харьков: НТУ «ХПИ», 2005. - 546 с.
2. Стандарт UML 2.2. http://www.omg.org/technology/documents/modeling_spec_catalog.htm #UML

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

1. Каталог патернів проектування <https://refactoring.guru/ru/design-patterns>
2. Лекція з якості програмного забезпечення www.intuit.ru/studies/courses/2190/237/lecture/6136