Додаток 7

Міністерство освіти і науки України

Харківський національний університет імені В.Н. Каразіна

Кафедра моделювання систем і технологій

"ЗАТВЕРДЖУЮ"

Проректор з науково-педагогічної роботи

____

"______" ______________ 20 18 р.

Робоча програма навчальної дисципліни

Розробка та супровід проблемно-орієнтованих програмних систем

рівень вищої освіти	другий (магістерський) рівень
галузь знань	12 Інформаційні технології 15 Автоматизація та приладобудування
	(шифр і назва)
спеціальність	122 Комп'ютерні науки, 125 Кібербезпека, 151 Автоматизація та комп'ютерно-інтегровані технології
	(шифр і назва)
освітня програма	Інформаційні управляючі системи і технології, Безпека інформаційних і комунікаційних систем Комп'ютеризовані системи автоматизації та управління
	(шифр і назва)
спеціалізація	
	(шифр і назва)
вид дисципліни	обов'язкова
	(обов'язкова / за вибором)
факультет	комп'ютерних наук

2018 / 2019 навчальний рік

Програму обговорено та рекомендовано до затвердження вченою радою факультету комп'ютерних наук

"29" серпня 2018 року, протокол № 9

РОЗРОБНИК ПРОГРАМИ: д.т.н., проф. Ткачук Микола Вячеславович

Програму схвалено на засіданні кафедри моделювання систем і технологій

"11"червня 2018 року , протокол № 14

Завідувач кафедри моделювання систем і технологій

_________________________ Краснобаєв В.А.
(підпис) (прізвище та ініціали)

Програму погоджено методичною комісією факультету комп'ютерних наук

"27"червня 2018 року, протокол № 7

Голова методичної комісії

_________________________ Васильєва Л. В.
(підпис) (прізвище та ініціали)

Програма навчальної дисципліни "**Розробка та супровід проблемно-орієнтованих програмних систем**" складена відповідно до освітньо-професійної (освітньо-наукової) програми підготовки

другий (магістерський) рівень____

(назва рівня вищої освіти, освітньо-кваліфікаційного рівня)

спеціальності (напряму) 122 Комп'ютерні науки, 151 Автоматизація та комп'ютерно-інтегровані технології, 125 Кібербезпека_______________________________________

спеціалізації

____

Предметом вивчення є сучасні методології розробки, супроводу та технології реалізації проблемно-орієнтованих програмних систем (ППС).

1. Опис навчальної дисципліни

1.1. Мета викладання навчальної дисципліни

Метою викладання навчальної дисципліни є є поглиблення та розширення теоретичних та практичних знань студентів в області сучасних методологій проектування та технологій реалізації складних ППС.

1.2. Основні завдання вивчення дисципліни

- ознайомлення з характерними особливостями сучасного етапу розвитку інформаційних технологій (ІТ) та індустрії розробки програмного забезпечення (ПЗ);
- вивчення основних методологічних принципів сучасної програмної інженерії;
- вивчення основ розробки ПЗ за методологією RUP;
- з'ясування переваг гнучких (agile-) методологій розробки ПЗ;
- засвоєння принципів архітектурного проектування ПЗ ППС та застосування патернів проектування;

1.3. Кількість кредитів

122,125- 7 кредитів ECTS

151- 10 кредитів ECTS

1.4. Загальна кількість годин

122,125- 210 годин

151- 300 годин.

1.5. Характеристика навчальної дисципліни	
Нормативна / за вибором	
Денна форма навчання	Заочна (дистанційна) форма навчання
Рік підготовки	
1-й	1-й
Семестр	
1-й	1-й/ 2-й
Лекції	
32 год.- 151,125,122	____ год.
Практичні, семінарські заняття	
_____ год.	_____ год.
Лабораторні заняття	
__16__ год. 151,125,122	_____ год.
Самостійна робота	
__42__ год. 122,125 72 год. 151	______ год.
Індивідуальні завдання	
___ год.	
Семестр 2	
2-й	2-й
Лекції	
_32 год -151,125,122	____ год.
Практичні, семінарські заняття	
_____ год.	____ год.
Лабораторні заняття	
_16_ год - 151,125,122	____ год.
Самостійна робота	
__72_ год. 122,125 132 год. 151	____ год.
Індивідуальні завдання	
____ год.	____ год.

1.6. Заплановані результати навчання

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

повинні знати

- фактори ризику та критерії успіху виконяння проектів по розробці складних ППС;
- основи сучасних методологій проектування ППС, включаючи індустріальний стандарт RUP (Rational Unified Process);
- переваги застосування гнучких методологій проектування, зокрема, таких як XP та Scrum;
- методи накопичення та аналізу вимог до ПЗ;
- класифікацію еталонних програмних архітектур, критерії їх вибору із урахуванням специфіки власних проектів;

- класифікацію патернів проектування та принципи їх використання в проектуванні ППС.
- знати переваги використання патернів проектування (design patterns) із колекції патернів GoF.
- моделі та методи розробки гетерогенних ресурсів даних в ППС
- принципи розробки та супроводу компонентних програмних рішень для ППС (на прикладі платформи JEE та стандарту CORBA)
- архітектурну модель та технологію JMS для розробки розподілених ПОПС
- архітектурну модель та технологію REST-сервісів для розробки розподілених ПОПС
- методи та технології реінжинірингу в процесах супроводу ПОПС

<u>повинні вміти</u>

- визначати та аналізувати фактори ризику та успіху при плануванні та виконанні відповідного IT-проекту;
- планувати та виконувати основні процеси і дисципліни на всіх фазах RUP-проекту: Inception, Elaboration, Construction та Transition;
- застосовувати основні практики організації гнучкої розробки ПЗ за методологіями XP та Scrum;
- виконувати накопичення та аналіз вимог із застосуванням таких методів / нотацій як: RUP/UML, SADT/IDEF0/IDEFX1, DFD;
- визначати необхідність та можливість застосування в процесах розробки ПЗ ППС еталонних системних архітектур (ЕСА): автономні (stand-alone architecture) ЕСА, ЕСА типу ”файл-сервер” (file-server architecture), 2-х, 3-х рівневий ”клієнт-сервер” (client-server architecture);
- застосовувати патерни проектування (design patterns) із колекції патернів GoF.
- використовувати моделі та методи розробки гетерогенних ресурсів даних в ППС
- застосовувати на практиці принципи розробки та супроводу компонентних програмних рішень для ППС (на прикладі платформи JEE та стандарту CORBA)
- застосовувати на практиці архітектурну модель та технологію JMS для розробки розподілених ПОПС
- застосовувати на практиці архітектурну модель та технологію REST-сервісів для розробки розподілених ПОПС
- знати та використовувати методи та технології реінжинірингу в процесах супроводу ПОПС

2. Тематичний план навчальної дисципліни

Вступ.

Мета та задачі дисципліни. Структура лекційного материалу. Форми занять та контролю засвоення материалу. Стислий огляд інформаційних джерел, що рекомендовані до дисципліни. Вхідний контроль знань студентів щодо базових питань програмування та інформаційних технологій (список контрольних запитань).

1 семестр

Розділ 1. Основи сучасної методології проектування складних ППС

Тема 1. Методологічні та технологічні проблеми проектування складних ППС

Деякі характерні особливості сучасного етапу розвитку інформаційних технологій (IT) та індустрії розробки програмного забезпечення (ПЗ). Вимоги щодо ефективності процесу проектування ПЗ ППС: «фактори ризику» та «критерії успіху». Приклади реальних проектів розробки ППС. Постановка задачі побудови інтегрованого модельно-інструментального комплексу для адаптивного проектування та реінжинірингу складних ППС і необхідність міждисциплінарного підходу для її ефективного вирішення. Поняття про програмну кібернетику (software cybernetics): принципи, методи та застосування.

Тема 2. Огляд сучасних методологій проектування ППС. Методологія RUP

Базові поняття процесу проектування ППС. Основні моделі життєвого циклу (ЖЦ) розробки та експлуатації ППС. Класифікація основних традиційних методологій проектування ППС: структурно-функціональний підхід (SADT - Structured Analysis Design Technique) і обєктно-оріеитоване проектування (ООП), їх переваги та недоліки. Методологія проектування RUP (Rational Unified Process) як індустріальний стандарт ООП консорціума OMG (Object Management Group). Характеристика основних фаз та процесів RUP. CASE-засоби підтримки RUP. Класифікація та технологія застосування засобів мови моделювання систем UML (Unified Modeling Language 2.0) на основних етапах проектування ПЗ ППС.

Тема 3. Гнучкі технології (agile-technologies) проектування ППС

Недоліки та обмеження традиційних методологій проектування та реалізації ППС. Основні принципи розробки та характерні особливості застосування гнучких (agile-) методологій розробки ПЗ ППС. Методологія розробки ПЗ XP (eXtreme Programming) – основны принципи та переваги. Схема виконання проекту розробки ППС за методологією Scrum. Порівняльна характеристика методологій Scrum та XP. Критерії вибору та ефективності застосування гнучких (agile-) методологій розробки ПЗ ППС.

Розділ 2. Методи та засоби інженерії вимог в процесі проектування ППС

Тема 4. Класифікація вимог до програмного забезпечення ППС

Інженерія вимог до ПЗ (software requirements engineering) як складова частина процесу доменого моделювання в проектуванні ППС: основні задачі, методи та інструментальні засоби для їх вирішення (приклади систем управління вимогами - requirements management systems). Класифікація вимог до ПЗ: функціональні, системні вимог та атрибути якості (нефункціональні вимоги).

Тема 5. Засоби специфікації вимог

Особливості роботи з вимогами в гнучких процесах проектування ПЗ (на прикладі методології Scrum). Абстракції для моделювання предметної області (домену -) проектування ППС. Засоби мови UML для специфікації вимог до ПЗ: діаграми варіантів використання та діаграми послідовностей. Альтернативні підходи для специфікації вимог: SADT/IDEF0 та DFD (Data Flow Diagram): графічні нотації та приклади їх застосування.

Розділ 3. Архітектурне проектування ППС

Тема 6. Проектування системної архітектури ППС

Визначення поняття: системна архітектура (СА) – system architecture. Проблема вибору СА в процесі проектування ППС. Архитектурна модель «4+1» проекцій за Ф. Крачтеном (Ph. Kruchten) та її застосування.

Класифікація та особливості окремих типів СА: автономні СА, СА типу ”файл-сервер”, ”клієнт-сервер” (2-х, 3-х рівневий). Розгонутий приклад розробки еталонної архітектури розподіленої Web-базованої ППС для регіональної інформаційної системи моніторингу обєктів нафто-газовидобування.

Тема 7. Шаблони (патерни) проектування в процесі розробки архітектури ППС

Визначення поняття: шаблон (патерн – pattern) проектування, мета та переваги застосування патернів в процесі розробки архітектури ППС. Загальна класифікація патернів з колекції GoF.

Структурні патерни та патерні поведінки. Архітектурний патерн MVC (Model-View-Controller): визначення, загальний принцип побудови та переваги застосування.

Розгонутий приклад розробки патрнів для побулови еталонної архітектури розподіленої Web-базованої ППС для регіональної інформаційної системи моніторингу обєктів нафто-газовидобування.

Розділ 4. Моделі та методи розробки гетерогенних ресурсів даних в ППС

Тема 8. Онтології як засіб моделювання даних і знань в розподілених гетерогенних системах

Проблеми обробки даних в розподілених системах. Конфлікти визначення даних: структурні, семантичні та розмірності. Визначення поняття онтології. Лінгвістичні терміни опису онтологій. Приклади побудови доменних онтологій.

Тема 9. Технології інтеграції та федералізації ресурсів даних в розподілених ППС

Схема інтеграції ресурсів даних: принципи побудови, технології реалізації, переваги та недоліки. Схема федералізації ресурсів даних: принципи побудови, технології реалізації, переваги та недоліки.

2 семестр

Розділ 5. Принципи розробки та супроводу компонентних програмних рішень (КПР) для ППС (на прикладі платформи JEE та стандарту CORBA)

Тема 10. Основні поняття компонентних технологій розробки ПЗ

Загальні поняття та принципи побудови КПР: класи, модулі, компоненти та транзакції. Синхронна та асинхронна схеми взаємодії в КПР. Архітектурний стандарт CORBA для створення розподілених ППС.

Тема 11. Платформа JEE для проектування КПР

Основні версії платформи: JME, JSE, JEE. Поняття контейнера компонентів як основи платформи. Класифікація компонентів EJB. Особливості реалізації та застосування JEE-компонентів.

Розділ 6. Архітектурні моделі та технології розробки розподілених ППС

Тема 12. Архітектурна модель та технологія JMS для розробки розподілених ППС

Різниця між архітектурними моделями традиційних клієнт-серверних і розподілених програмних систем, які використовують технологію JMS. Моделі обміну повідомленнями в технології JMS. Склад компонентів та схему їх взаємодії у JMS. Типи JMS - повідомлень та їх стислий опис

Тема 13. Архітектурна модель та технологія REST-сервісів для розробки розподілених ППС

Поняття Веб-сервісу. HTTP-запити REST-застосуванні. Специфікація REST API. Мови програмування для реалізації REST API. Файл "package.json" Процес створення API для RESTful застосування.

Розділ 7. Методи та технології реінжинірингу в процесах супроводу ППС

Тема 14. Моделі, методи та технології проведення реінжинірингу

Визначення поняття успадкованої програмної системи (УПС). Методи аналізу та оцінки УПС. Методи та процедури проведення реінжинірингу УПС

Тема 15. Комплексний приклад проведення реінжинірингу реальної УПС.

Опис архітектури та особливостей функціонування реальної УПС в предметній галузі АСУ ТП на об’єктах нафто- та газовидобування. Оцінка стану УПС та функціональні вимоги до нової версії системи. Розробка нової архітектури та структури даних для модернізованої АСУ ТП.

3. Структура навчальної дисципліни

Назви розділів 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
1 семестр												
Розділ 1. Основи сучасної методології проектування складних ППС												
Тема 1. Вступ: Методологічні та технологічні проблеми проектування складних ППС		2										
Тема 2. Огляд сучасних методологій проектування ППС. Методологія RUP		4										
Тема 3. Гнучкі технології (agile-technologies) проектування ППС		4										
Разом за розділом 1		*10*		*6*		*10/ 18*						
Розділ 2. Методи та засоби інженерії вимог в процесі проектування ППС												
Тема 4. Визначення та класифікація вимог до ПЗ ППС		4										
Тема 5. Засоби специфікації вимог		4										
Разом за розділом 2		*8*		*4*		*10/18*						
Розділ 3. Архітектурне пректування ППС												
Тема 6. Проектування системної архітектури ППС		4										
Тема 7. Шаблони (патерни) проектування в процесі розробки архітектури ППС		4										
Разом за розділом 3		*8*		*4*		*10/18*						
Розділ 4. Моделі та методи розробки гетерогенних ресурсів даних в ППС												
Тема 8. Онтології як засіб моделювання даних і знань в розподілених гетерогенних системах		4										
Тема 9. Технології інтеграції та федералізації ресурсів даних в розподілених ППС		4										
Разом за розділом 4		*8*		*4*		*12/18*						

Назви розділів 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
2 семестр												
Розділ 5. Принципи розробки та супроводу компонентних програмних рішень (КПР) для ППС (на прикладі платформи JEE та стандарту CORBA)												
Тема 10. Основні поняття компонентних технологій розробки ПЗ		4										
Тема 11. Платформа JEE для проектування КПР		4										
Разом за розділом 5		*8*		*6*		*18/33*						
Розділ 6. Архітектурні моделі та технології розробки розподілених ППС												
Тема 12. Архітектурна модель та технологія JMS для розробки розподілених ППС		4										
Тема 13. Архітектурна модель та технологія REST-сервісів для розробки розподілених ППС		6										
Разом за розділом 6		*10*		*4*		*18/33*						
Розділ 7. Методи та технології реінжинірингу в процесах супроводу ППС												
Тема 14. Моделі, методи та технології проведення реінжинірингу		6										
Тема 15. Комплексний приклад проведення реінжинірингу реальної УПС		6										
Разом за розділом 7		*12*		*4*		*36/66*						
Усього годин		64		32		114/204						

4. Теми лабораторних занять

№ з/п	Назва теми	Кількість годин
1	Основні фукціональні можливості CASE-засобу проектування ПЗ Visual Paradigm	2
2	Розробка UML діаграм концептуального рівня проектування ПЗ	2
3	Розробка UML діаграм логічного рівня проектування ПЗ: статичні діаграми	2
4	Розробка UML діаграм логічного рівня проектування ПЗ: динамічні діаграми	2

№ з/п	Назва теми	Кількість годин
5	Розробка UML діаграм фізичного рівня проектування ПЗ	2
6	Налаштування середовища розробки для створення трьох-рівня компонентного програмного рішення (кпр) з тонким клієнтом на платформі JEE	2
7	Проектування і розробка трехуровнего додатки з тонким клієнтом в JEE	2
8	Проектування шару домен із застосуванням типового рішення «модель предметної області» (Domain Model) з рівнем служб (Service Layer)	2
9	Проектування і розробка КПР з використанням технології EJB на платформі JEE	2
10	Проектування і розробка розподіленого КПР з використанням Web-служб в JEE	3
11	Технологія JMS (Java Messaging Services)	3
12	Розробка програмного застосування для обміну повідомленнями з використанням технології JMS 1.1.	2
13	Розробка Web-застосування для обміну повідомленнями з використанням JMS 2.0.	2
14	Розробка серверних компонентів RESTful застосувань з використанням Node.js, Spring MVC та PHP	2
15	Тестування серверних компонентів RESTful застосувань з використанням JMeter	2
	Разом	**32**

5. Завдання для самостійної робота 122,125

№ з/п	Види, зміст самостійної роботи	Кількість годин
1	**Методологія RUP:** Надати опис основні завдання / проектні документи (артефакти), які повинні бути вирішені / створені на фазі RUP / Inception /Elaboration/Construction/ Transition. Показати, як методика виконання цієї фази RUP може бути застосована у вашій курсової / дипломної роботи.	10
2	**Гнучкі технології (agile-technologies) проектування ПЗ: Scrum and XP:** Покажіть, як методологія Scrum може бути застосована у вашому проекті, зокрема, розгляньте при цьому такі поняття як: а) як виконати розподіл Scrum-ролей у вашому проекті? б) що таке Product Backlog і Sprint Backlog в вашому проекті? в) чим може бути для вас Daily Scrum Meeting і Sprint Review Meeting?	10
3	**Засоби специфікації вимог:** Розробка контекстних діаграм SADT / IDEF0, з прикладами для своєї курсової / дипломної роботи. Проведіть її декомпозицію на 2-3 функціональних блоку наступного логічного рівня.	10
4	**Проектування системної архітектури ППС**: 1) Модель «4 + 1» - уявлення системної архітектури (СА) з Ф. Кручтену (Ph. Kruchten): проекція логічної структури (Logical View): призначення, артефакти, актори. Наведіть приклад можливої інтерпретації цієї проекції опису СА в контексті своєї курсової / дипломної роботи. (для цього необхідно привести смислове назва теми своєї д / р). 2) Еталонна системна архітектура (ЕСА) типу «файл-сервер (file-server)»: її UML-діаграма, переваги і недоліки. Який тип ЕСА і чому (мотивація цього вибору) ви плануєте використовувати в своєму проекті, для цього:	12

№ з/п	Види, зміст самостійної роботи	Кількість годин
	а) привести список основних вимог, к-які важливі для вибору цієї архітектури; b) UML-діаграма цієї архітектури із зазначенням деталей реалізації (технології програмування + протоколи мережевої взаємодії); c) перерахуйте переваги та можливі недоліки цієї ЕСА. 3) ЕСА типу «клієнт-сервер» / «товстий клієнт» (thick client): її UML-діаграма, переваги і недоліки. Який тип ЕСА і чому (мотивація цього вибору) ви плануєте використовувати в своїй курсовій / дипломній роботі, для цього: а) привести список основних вимог, к-які важливі для вибору цієї архітектури; b) UML-діаграма цієї архітектури із зазначенням деталей реалізації (технології програмування + протоколи мережевої взаємодії); c) перерахуйте переваги та можливі недоліки цієї ЕСА.	
5	**Шаблони (патерни) проектування в процесі розробки ПЗ ППС:** У контексті своєї курсової / дипломної роботи. і з урахуванням обраної ЕСА покажіть можливість застосування одного (або кількох) з патернів проектування з GoF-колекції	18
6	**Онтології як засіб моделювання даних і знань в розподілених гетерогенних системах** 1. У контексті своєї дипломної роботи розробити її онтологію, для чого: a) перерахувати основні концепти ПрО (7-10 елементів) і визначити їх відносини в термінах: синонім / антонім / омонім, гіпонім / гипероним, холонім / мероним b) розробити соотв. UML-діаграми класів цієї онтології. 2. Представити цю онтологію у вигляді EER - діаграми (не менше 4-5 сутностей!), Для кожної сутності задати по 3-4 основних смислових атрибута. Визначити також всі розмірності і типи відповідних зв'язків між цими сутностями і задати для кожної з них можливі асоційовані атрибути (з урахуванням специфіки своєї ПрО). Рішення про склад цих атрибутів прийняти самостійно.	18
7	**Технології інтеграції та федералізації ресурсів даних в розподілених ППС:** Представити архітектурні схеми для реалізації технології інтеграції та федералізації ресурсів даних в розподілених ППС, навести конкретні приклади СКБД та програмних засобів для їх реалізації	18
8	**Платформа JEE для проектування КПР:** Розглянути наступні питання програмної реалізації CORBA-об'єктів 1. Що таке ORB і IIOP? Яким чином з їх допомогою взаємодію CORBA-об'єкти? 2. Що таке IDL? Яким чином обробляються IDL-опису CORBA-об'єктів? 3. Які етапи повинні бути програмно реалізовані для створення серверного CORBA-об'єкта? 4. Які інтерфейси повинен мати IDL-модуль CORBA-сервера? 5. Які інтерфейси повинен мати IDL-модуль CORBA-клієнта? 6. Який метод забезпечує знаходження віддаленого CORBA-об'єкта? 7. Якими альтернативними способами можна виконати запуск CORBA-додатки клієнтським застосуванням?	18
	Разом	**114**

5. Завдання для самостійної робота 151

№ з/п	Види, зміст самостійної роботи	Кількість годин
1	**Методологія RUP:** Надати опис основні завдання / проектні документи (артефакти), які повинні бути вирішені / створені на фазі RUP / Inception /Elaboration/Construction/ Transition. Показати, як методика виконання цієї фази RUP може бути застосована у вашій курсової / дипломної роботи.	18
2	**Гнучкі технології (agile-technologies) проектування ПЗ: Scrum and XP:** Покажіть, як методологія Scrum може бути застосована у вашому проекті, зокрема, розгляньте при цьому такі поняття як: а) як виконати розподіл Scrum-ролей у вашому проекті? б) що таке Product Backlog і Sprint Backlog в вашому проекті? в) чим може бути для вас Daily Scrum Meeting і Sprint Review Meeting?	18
3	**Засоби специфікації вимог:** Розробка контекстних діаграм SADT / IDEF0, з прикладами для своєї курсової / дипломної роботи. Проведіть її декомпозицію на 2-3 функціональних блоку наступного логічного рівня.	18
4	**Проектування системної архітектури ППС**: 1) Модель «4 + 1» - уявлення системної архітектури (СА) з Ф. Кручтену (Ph. Kruchten): проекція логічної структури (Logical View): призначення, артефакти, актори. Наведіть приклад можливої інтерпретації цієї проекції опису СА в контексті своєї курсової / дипломної роботи. (для цього необхідно привести смислове назва теми своєї д / р). 2) Еталонна системна архітектура (ЕСА) типу «файл-сервер (file-server)»: її UML-діаграма, переваги і недоліки. Який тип ЕСА і чому (мотивація цього вибору) ви плануєте використовувати в своєму проекті, для цього: a) привести список основних вимог, к-які важливі для вибору цієї архітектури; b) UML-діаграма цієї архітектури із зазначенням деталей реалізації (технології програмування + протоколи мережевої взаємодії); c) перерахуйте переваги та можливі недоліки цієї ЕСА. 3) ЕСА типу «клієнт-сервер» / «товстий клієнт» (thick client): її UML-діаграма, переваги і недоліки. Який тип ЕСА і чому (мотивація цього вибору) ви плануєте використовувати в своїй курсовій / дипломній роботі, для цього: a) привести список основних вимог, к-які важливі для вибору цієї архітектури; b) UML-діаграма цієї архітектури із зазначенням деталей реалізації (технології програмування + протоколи мережевої взаємодії); c) перерахуйте переваги та можливі недоліки цієї ЕСА.	18
5	**Шаблони (патерни) проектування в процесі розробки ПЗ ППС:** У контексті своєї курсової / дипломної роботи. і з урахуванням обраної ЕСА покажіть можливість застосування одного (або кількох) з патернів проектування з GoF-колекції	33
6	**Онтології як засіб моделювання даних і знань в розподілених гетерогенних системах** 1. У контексті своєї дипломної роботи розробити її онтологію, для чого: a) перерахувати основні концепти ПрО (7-10 елементів) і визначити їх відносини в термінах: синонім / антонім / омонім, гіпонім / гипероним, холонім / мероним b) розробити соотв. UML-діаграми класів цієї онтології. 2. Представити цю онтологію у вигляді EER - діаграми (не менше 4-5 сутностей!), Для кожної сутності задати по 3-4 основних смислових	33

№ з/п	Види, зміст самостійної роботи	Кількість годин
	атрибута. Визначити також всі розмірності і типи відповідних зв'язків між цими сутностями і задати для кожної з них можливі асоційовані атрибути (з урахуванням специфіки своєї ПрО). Рішення про склад цих атрибутів прийняти самостійно.	
7	**Технології інтеграції та федералізації ресурсів даних в розподілених ППС:** Представити архітектурні схеми для реалізації технології інтеграції та федералізації ресурсів даних в розподілених ППС, навести конкретні приклади СКБД та програмних засобів для їх реалізації	33
8	**Платформа JEE для проектування КПР:** Розглянути наступні питання програмної реалізації CORBA-об'єктів 1. Що таке ORB і IIOP? Яким чином з їх допомогою взаємодію CORBA-об'єкти? 2. Що таке IDL? Яким чином обробляються IDL-опису CORBA-об'єктів? 3. Які етапи повинні бути програмно реалізовані для створення серверного CORBA-об'єкта? 4. Які інтерфейси повинен мати IDL-модуль CORBA-сервера? 5. Які інтерфейси повинен мати IDL-модуль CORBA-клієнта? 6. Який метод забезпечує знаходження віддаленого CORBA-об'єкта? 7. Якими альтернативними способами можна виконати запуск CORBA-додатки клієнтським застосуванням?	33
	Разом	204

6. Контрольні роботи

1. Покажіть, як методологія XP може бути застосована у вашому проекті, зокрема, розгляньте при цьому такі поняття як: а) сформулюйте метафору для вашого проекту; б) розробка на основі тестування; в) включення замовника в команду розробників проекту

2 Побудуйте приклад UML / Use Case - та SADT / IDEF0 - діаграм для одного і того ж підмножини функціональніх вимог для курсової / дипломної роботи.

3. Модель «4 + 1» - уявлення системної архітектури (СА) з Ф. Кручтену (Ph. Kruchten): проекція процесів розробки (Development View): призначення, артефакти, актори. Наведіть приклад можливої інтерпретації цієї проекції опису СА в контексті свого проекту (для цього необхідно привести смислове назва теми своєї д / р).

4. Розглянути ЕСА типу «клієнт-сервер» / «тонкий клієнт (thin client)»: її UML-діаграма, переваги і недоліки. Який тип ЕСА і чому (мотивація цього вибору) ви плануєте використовувати для курсової / дипломної роботи., для цього:
   a) привести список основних вимог, к-які важливі для вибору цієї архітектури;
   b) UML-діаграма цієї архітектури із зазначенням деталей реалізації (технології програмування + протоколи мережевої взаємодії);
   c) перерахуйте переваги та можливі недоліки цієї ЕСА.

5. Розглянути ЕСА типу «клієнт-сервер» / «Сервер додатків (application server)»: її UML-діаграма, переваги і недоліки. Який тип ЕСА і чому (мотивація цього вибору) ви плануєте використовувати для курсової / дипломної роботи, для цього:
   a) привести список основних вимог, к-які важливі для вибору цієї архітектури;
   b) UML-діаграма цієї архітектури із зазначенням деталей реалізації (технології

програмування + протоколи мережевої взаємодії);
c) перерахуйте переваги та можливі недоліки цієї ЕСА.

6. Поясніть різницю між такими поняття компонентних програмних рішень (КПР) як: клас, компонент, модуль? Наведіть приклади визначення цих понять на будь-якій мові програмування: Java, C #, C ++.

7. Які основні поняття включає в себе компонентна модель (КМ) розробки ПО? Назвіть відомі вам КМ, яку з них ви використовуєте в своїй курсовій / дипломній роботі
8. Перелічіть і коротко охарактеризуйте загальні принципи побудови розподілених КПР. Які з них мають значення для ПО у вашій курсовій / дипломній роботі?

9. Розробити архітектурну модель JMS для можливої реалізації ПЗ в своїй курсовій / дипломній роботі

10. Розробити архітектурну модель на основі REST-сервісів для можливої реалізації ПЗ в своїй курсовій / дипломній роботі

7. Методи контролю

Поточне тестування кожного студента проводиться на лекційних і самостійних роботах для оцінки ступеня засвоєння матеріалу, підсумкове тестування на екзамені.

8. Схема нарахування балів

Приклад для підсумкового семестрового контролю при проведенні семестрового екзамену

1 семестр

Поточний контроль, самостійна робота, індивідуальні завдання											Екзамен	Сума
Розділ 1			Розділ 2		Розділ 3		Розділ 4		Контрольна робота, передбачена навчальним планом	Разом		
Т1	Т2	Т3	Т4	Т5	Т6	Т7	Т8	Т9				
2	2	2	3	2	3	2	2	2	40	60	40	100

2 семестр

Поточний контроль, самостійна робота, індивідуальні завдання								Екзамен	Сума
Розділ 5		Розділ 6		Розділ 7		Контрольна робота, передбачена навчальним планом	Разом		
Т4	Т5	Т6	Т7	Т8	Т9				
3	3	3	4	3	4	40	60	40	100

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. *Эванс, Э.* Предметно-ориентированное проектирование (DDD): структуризация сложных программных систем: Пер. с англ. – М. «ООО И.Д. Вильямс», 2011. – 448с.
2. *Ларман, К.* Применение UML 2.0 и шаблонов проектирования: практическое руководство - : Пер. с англ. – М. ООО «И.Д. Вильямс», 2009. – 736 с.
3. Лаврищева Е.М., Грищенко В.Н. Сборочное программирование. Основы индустрии программных продуктов. – К., Наукова думка, 2009. – 370с.
4. Андон Ф.И., Коваль Г.И., Коротун Т.М. и др. Основы инженерии качества программных систем. – 2-е изд. - К.: «Академпериодика», 2007. – 632с..
5. *Басс Л., Клементс П., Кацман Р.* Архитектура программного обеспечения на практике. - СПб.: Питер, 2006. -340 с.
6. *С. Амблер*. Гибкие технологии: экстремальное программирование и унифицированный процесс разработки. Библиотека программиста. – СПб.: Питер, 2005. – 412 с.
7. *Ткачук Н.В., Шеховцов В.А., Кукленко Д.В., Сокол В.Е.* Архитектуры, модели и технологии программного обеспечения информационно-управляющих систем. – Харьков: НТУ «ХПИ», 2005. - 546 с.

Допоміжна література

1. Ткачук М.В., Мартінкус І.О., Нагорний К.А., Гамзаев Р. О. Про один підхід до оцінки ефективності застосування методів доменного моделювання при розробці сімейств програмних систем // Збірка наук. праць ХУПС, № 5(54), 2017. – С. 127-134.
2. M. Tkachuk, A. Vekshin, and R. Gamzayev A Model-Based Framework for Adaptive Resource Management in Mobile Augmented Reality System // Proceedings of the ICTERI-2016: 12th International Conference on ICT in Education, Research and Industrial Applications: Integration, Harmonization and Knowledge Transfer, Kyiv, Ukraine, June 21-24, 2016, CEUR-WS.org/Vol-1614, pp.41-56.
3. Ткачук М.В. Алгоритмічна модель адаптивного управління ресурсами мобільної системи доповненої реальності / М.В. Ткачук, О.В. Векшин, Д.М. Місюта // // Зб. наук. Праць «Системи обробки інформації», ХУПС ім. Івана Кожедуба. – Харків, – 2015. - № 12 (137), с.151- 155.
4. Сокол В.Є. Моделі та алгоритми опрацювання інцидентів в системах автоматизації корпоративних IT-послуг на основі використання бази даних управління конфігураціями / Сокол В.Є., Ткачук М.В., Кузнєцов М // Бионика интеллекта. - ХНУРЄ – Х. 2014. № 1(82). – С. 74-79.
5. Ткачук Н.В. Концепция и информационная технология повышения эффективности применения систем управления корпоративной ИТ-инфраструктурой / Ткачук Н.В.,

Сокол В.Е. // // Вестник Национального технического университета «ХПИ». Сборник научных трудов. Серия: Новые решения в современных технологиях. – Харьков: НТУ «ХПИ». – 2014. – № 17 (1060). – С. 85-95.

6. Гамзаев Р. О., Мета-модель процесса трассировки требований при разработке программного обеспечения / Гамзаев Р. О., Ткачук Н. В., Мартинкус И. О. // Вісник НТУ «ХПІ». Серія: Нові рішення в сучасних технологіях. – Х: НТУ «ХПІ», – 2014. – 26 (1069). – С.121-128
7. Ткачук Н.В. Онтологический подход к интеграции систем управления ИТ-ресурсами в корпоративную архитектуру предприятий / Ткачук Н.В., Сокол В.Е., Глуховцова Е.В. // Інформаційні технології: наука, техніка, технологія, освіта, здоров'я: Програма XXI міжнародної науково-практичної конференції, (29-31 травня 2013р., Харків) / за ред. проф. Товажнянського Л.Л. – Харків, НТУ «ХПІ». – С.8.
8. Ткачук Н. В. Лекции по проблемам программной кибернетики в Клагенфуртском университете (Австия) октябрь 2011 – январь 2012 (англ..) – *Prof. Dr. Nikolay Tkachuk Selected Topics in Application Engineering Research: Cybernetics Approaches in Modern Software Engineering*(электр. ресурс https://campus.aau.at/studien/lvkarte.jsp?sprache_nr=35&rlvkey=71887)

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

1. Методичні вказівки до виконання лабораторних робіт з дисципліни "Архітектура та проектування програмного забезпечення" : за розд. "Мова моделювання UML як засіб аналізу та проектування програмних систем" : для студ. з напряму підготовки 6.050103 "Програмна інженерія" / уклад. М. В. Ткачук [та ін.] ; "Харківський політехнічний ін-т", Нац. техн. ун-т. – Харків : НТУ "ХПІ", 2017. – 60 с. //
2. http://repository.kpi.kharkov.ua/handle/KhPI-Press/31734.
3. http://www.computer.org/portal/web/swebok/html/contents
4. http://www.omg.org
5. http://www.intuit.ru
6. http://www.conference-service.com/conferences/software-engineering.html