Міністерство освіти і науки України

Харківський національний університет імені В.Н. Каразіна

Кафедра моделювання систем і технологій

“ЗАТВЕРДЖУЮ”

Проректор з науково-педагогічної роботи

_____ Антон ПАНТЕЛЕЙМОНОВ

“_____” _____ 2020 р.

РОБОЧА ПРОГРАМА НАВЧАЛЬНОЇ ДИСЦИПЛІНИ

«Розробка та супровід проблемно-орієнтованих програмних систем»

рівень вищої освіти	другий (магістерський)
галузь знань	12 Інформаційні технології
спеціальність	123 Комп’ютерна інженерія
освітня програма	Комп’ютерна інженерія
спеціалізація	
вид дисципліни	Обов’язкова
факультет	Комп’ютерних наук

2020 / 2021 навчальний рік

Програму рекомендовано до затвердження Вченою радою факультету комп'ютерних наук

« 31 » серпня 2020 року, протокол № 12

РОЗРОБНИКИ ПРОГРАМИ: доктор технічних наук, професор кафедри моделювання систем і технологій **Ткачук Микола Вячеславович**.

Програму схвалено на засіданні кафедри моделювання систем і технологій

Протокол від « 28 » серпня 2020 року № 1

Завідувач кафедри моделювання систем і технологій

_____ Микола ТКАЧУК

Програму погоджено з гарантом освітньо-професійної програми 123 «Комп'ютерна інженерія»

Гарант освітньо-професійної програми «Комп'ютерна інженерія»

_____ Олена ТОСТОЛУЗЬКА

Програму погоджено методичною комісією факультету комп'ютерних наук

Протокол від « 31 » серпня 2020 року № 1

Голова методичної комісії факультету комп'ютерних наук

_____ Анатолій БЕРДНІКОВ

ВСТУП

Програма навчальної дисципліни “**Розробка та супровід проблемно-орієнтованих програмних систем**” складена відповідно до освітньо-професійної програми підготовки другого (магістерського) рівня спеціальності 123 Комп’ютерна інженерія.

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 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;

В результаті вивчення цієї дисципліни студенти мають набути наступні компетентності.

Загальні компетентності (ЗК):

- вміння виявляти, ставити та вирішувати проблеми за професійним спрямуванням (ЗК1);
- здатність до пошуку, оброблення та аналізу інформації з різних джерел(ЗК4);
- здатність приймати обґрунтовані рішення (ЗК6);
- здатність оцінювати та забезпечувати якість виконуваних робіт (ЗК8);

Спеціальні (фахові,предметні) компетентності (ФК):

- здатність обґрунтовано обирати та застосовувати фундаментальні знання і моделі, а також технології створення та використання прикладного і спеціалізованого програмного забезпечення для розв’язування складних професійних задач і проблем комп’ютерної інженерії (ФК1);
- здатність до дослідження, системного аналізу та забезпечення безперервності бізнес/операційних процесів, концепцій, теорій, принципів і методів нових технологій, включаючи технології розумних, мобільних, зелених і безпечних обчислень (ФК3);
- здатність застосовувати системний підхід до вирішення науково -технічних завдань комп’ютерної інженерії (ФК4);
- здатність застосовувати комплексний підхід до вирішення експериментальних завдань модернізації та реконструкції комп’ютерних систем та мереж,

різноманітних вбудованих і розподілених додатків, зокрема з метою підвищення їх ефективності (ФК7);

- здатність аргументувати вибір методів розв'язування складних спеціалізованих задач і проблем, критично оцінювати отримані результати, обґрунтовувати та захищати прийняті рішення (ФК11);
- здатність перетворювати формальні моделі в напрямку отримання практично необхідної комп'ютерної моделі та ставити задачі збереження і обробки даних (ФК13);

1.3. Кількість кредитів–10

1.4. Загальна кількість годин–300

1.5. Характеристика навчальної дисципліни	
Обов'язкова / за вибором	
Денна форма навчання	Денна форма навчання
Рік підготовки	
1-й	1-й
Семестр	
1-й	2-й
Лекції	
32 год.	32 год.
Практичні, семінарські заняття	
16 год.	16 год.
Лабораторні заняття	
год.	год.
Самостійна робота	
72 год.	132 год.
у тому числі індивідуальні завдання	
год.	

1.6. Заплановані результати навчання

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

знати:

- фактори ризику та критерії успіху виконання 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.

вміти:

- 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 (ECA): автономні (stand-alone architecture) ECA, ECA 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.

В Результаті вивчення дисципліни у студента повинні формуватися наступні програмні результати навчання (ПРН):

- знати і розуміти наукові і математичні положення, що лежать в основі функціонування програмних і програмно-технічних комп’ютерних засобів, систем та мереж, Інтернету речей, систем для оброблення великих даних (ПРН2);
- знати і розуміти принципи системного аналізу та забезпечення безперервності бізнес/операційних процесів, концепцій, теорій, принципів і методів нових технологій, включаючи технології розумних, мобільних, зелених і безпечних обчислень (ПРН4);
- мати фундаментальні знання і розуміння моделей, а також технологій створення та використання прикладного і спеціалізованого програмного забезпечення розв’язування професійних задач і проблем комп’ютерної інженерії (ПРН6);

- знати засоби автоматизації проектування до розробки компонентів комп'ютерних систем та мереж, Інтернет додатків, кіберфізичних систем тощо (ПРН7);
- вміти застосовувати знання для аналізу інженерних продуктів, процесів і систем за встановленими критеріями, ідентифікації, формулювання і розв'язування науково - технічних задач комп'ютерної інженерії, використовуючи методи, що є найбільш придатними для досягнення поставлених цілей (ПРН9);
- вміти формулювати та розв'язувати задачі у галузі комп'ютерної інженерії, що пов'язані з процедурами спостереження об'єктів, вимірювання, контролю, діагностування і прогнозування з урахуванням загальнолюдських цінностей, суспільних, державних та виробничих інтересів (ПРН10);
- мати навички автономного і самостійного навчання у сфері комп'ютерної інженерії і дотичних галузей знань, аналізувати власні освітні потреби та об'єктивно оцінювати результати навчання (ПРН11);
- вміти застосовувати системний підхід до вирішення науково - технічних завдань комп'ютерної інженерії (ПРН14);
- вміти досліджувати, розробляти та впроваджувати засоби і системи автоматизації проектування до розробки компонентів комп'ютерних систем та мереж, Інтернет додатків, кіберфізичних систем тощо (ПРН16);
- застосовувати, інтегрувати, розробляти, впроваджувати та удосконалювати сучасні інформаційні технології, науково - технічні методи і моделі, фізичні та математичні фундаментальні знання в галузі комп'ютерної інженерії (ПРН17);
- здатність аргументувати вибір методів розв'язування складних спеціалізованих задач і проблем, критично оцінювати отримані результати, обґрунтовувати та захищати прийняті рішення (ПРН18);
- зрозуміло і недвозначно доносити власні висновки з проблем комп'ютерної інженерії, а також знання та пояснення, що їх обґрунтовують (ПРН21);
- здатність адаптуватись до нових ситуацій, обґрунтовувати, приймати та реалізовувати у межах компетенції рішення (ПРН23);
- усвідомлювати необхідність навчання впродовж усього життя з метою поглиблення набутих та здобуття нових фахових знань, удосконалення креативного мислення (ПРН24).

2. Тематичний план навчальної дисципліни

ВСТУП.

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

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 (ІТ) 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 об'єктно-орієнтоване 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) – основні 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). Розгорнутий приклад розробки еталонної архітектури розподіленої Web-базованої ППС для регіональної інформаційної системи моніторингу об'єктів 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 побудови еталонної архітектури розподіленої Web-базованої ППС для регіональної інформаційної системи моніторингу об'єктів 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.

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					
	усього	у тому числі				
		л	п	лаб.	інд.	с. р.
1	2	3	4	5	6	7
1 семестр						
РОЗДІЛ 1. Основи сучасної методології проектування складних ППС						
Тема 1. Методологічні та технологічні проблеми проектування складних ППС	10	2				8
Тема 2. Огляд сучасних методологій проектування ППС. Методологія RUP	16	4	2			10
Тема 3. Гнучкі технології (agile-technologies) проектування ППС	11	4	2			5
РОЗДІЛ 2. Методи та засоби інженерії вимог в процесі проектування ППС						
Тема 4. Класифікація вимог до програмного забезпечення ППС	15	4	2			9
Тема 5. Засоби специфікації вимог	22	6	4			12
Підготовка до контрольної роботи	5					5
РОЗДІЛ 3. Архітектурне проектування ППС						
Тема 6. Проектування системної архітектури ППС	25	8	4			13
Тема 7. Шаблони (патерни) проектування в процесі розробки архітектури ППС	16	4	2			10
Разом за 1 семестр	**120**	**32**	**16**			**72**
2 семестр						
РОЗДІЛ 4. Моделі та методи розробки гетерогенних ресурсів даних в ППС						
Тема 8. Онтології як засіб моделювання даних і знань в розподілених гетерогенних системах	26	4	2			20
Тема 9. Технології інтеграції та федералізації ресурсів даних в розподілених ППС	19	4	2			13
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	30	4	2			24
Тема 11. Платформа JEE для проектування КПР	21	4	2			15
РОЗДІЛ 6. Архітектурні моделі та технології розробки розподілених ППС						
Тема 12. Архітектурна модель та технологія JMS для розробки розподілених ППС	21	4	2			15
Тема 13. Архітектурна модель та технологія REST-сервісів для розробки розподілених ППС	23	6	2			15
Підготовка до контрольної роботи	5					5
РОЗДІЛ 7. Методи та технології реінжинірингу в процесах супроводу ППС						
Тема 14. Моделі, методи та технології проведення реінжинірингу	21	4	2			15
Тема 15. Комплексний приклад проведення реінжинірингу реальної УПС	14	2	2			10
Разом за 2 семестр	**180**	**32**	**16**			**132**
Усього годин	**300**	**64**	**32**			**204**

4. Теми семінарських (практичних, лабораторних) занять

№ з/п	Назва теми	Кількість годин
1 семестр		
ПР1	Основні функціональні можливості CASE-засобу проектування ПЗ Visual Paradigm	2
ПР2	Розробка UML діаграм концептуального рівня проектування ПЗ	2
ПР3	Розробка UML діаграм логічного рівня проектування ПЗ: статичні діаграми	2
ПР4	Розробка UML діаграм логічного рівня проектування ПЗ: динамічні діаграми	4
ПР5	Розробка UML діаграм фізичного рівня проектування ПЗ	4
ПР6	Проектування і розробка 3-х рівневого додатки з тонким клієнтом в JEE	2
Разом за 1 семестр		**16**
2 семестр		
ПР7	Проектування шару домен із застосуванням типового рішення «модель предметної області» (Domain Model) з рівнем служб (Service Layer)	2
ПР8	Проектування і розробка КПР з використанням технології EJB на платформі JEE	2
ПР9	Проектування і розробка розподіленого КПР з використанням Web-служб в JEE	2
ПР10	Технологія JMS (Java Messaging Services)	2
ПР11	Розробка програмного застосування для обміну повідомленнями з використанням технології JMS 2.0.	2
ПР12	Розробка Web-застосування для обміну повідомленнями з використанням JMS 2.0.	2
ПР13	Розробка серверних компонентів RESTful застосувань з використанням Node.js, Spring MVC та PHP	2
ПР14	Тестування серверних компонентів RESTful застосувань з використанням JMeter	2
Разом за 2 семестр		**16**
Усього годин		**32**

5. Завдання для самостійної роботи

№ з/п	Види, зміст самостійної роботи	Кількість годин
1 семестр		
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.	11
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

№ з/п	Види, зміст самостійної роботи	Кількість годин
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.	14
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.	24
5	Підготовка до контрольної роботи	5
Разом за 1 семестр		**72**
	2 семестр	
6	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	30
7	**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.	40
8	**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	25

№ з/п	Види, зміст самостійної роботи	Кількість годин
9	**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?	32
10	Підготовка до контрольної роботи	5
Разом за 2 семестр		**132**
Усього годин		**204**

6. Індивідуальні завдання

Контрольна робота.

7. Методи навчання

Пояснювально- ілюстративні, репродуктивні, практичні методи навчання. Як правило лекційні та практичні заняття проводяться аудиторне. А в умовах дії карантину заняття проводяться відповідно до Наказу ректора Харківського національного університету імені В.Н. Каразіна (аудиторне або дистанційне за допомогою платформ Google Meet або Zoom).

8. Методи контролю

Протягом першого та другого навчальних семестрів проводиться поточний контроль знань, який складається з виконання практичних робіт, контрольної роботи та підсумкового контролю у формі екзамену.

Максимальна оцінка за виконання практичного завдання – 6 балів.

За підсумками навчального семестру за умови, що студент виконав усі практичні роботи та склав контрольну роботу з результатом – позитивний бал та отримав за підсумками семестру не менше 50 балів – отримує допуск до складання екзамену. В іншому випадку студент не допускається до складання екзамену.

Контрольна робота складається з 3 питань (2 теоретичних та 1 практичного).

Відповіді на кожне питання контрольної роботи оцінюються у 6 балів.

Максимальна оцінка за виконання контрольної роботи 1 семестр – 24 балів,2 семестр –12 балів.

Загальна сума балів, яку студент може набрати при поточному контролі – 60 балів.

Умовами допуску до екзамену є виконання усіх контрольних точок протягом семестру.

Екзамен складається з чотирьох питань (два теоретичних та два практичних). Відповіді на кожне питання оцінюються у 10 балів.

Форма складання екзамену – письмова робота. Максимальна кількість балів за екзамен – 40 балів.

9. Схема нарахування балів

1 семестр

Поточний контроль та самостійна робота						Контрольна робота, передбачена навчальна планом	Разом	Екзамен	Разом
ПР1	ПР2	ПР3	ПР4	ПР5	ПР6	24	60	40	100
6	6	6	6	6	6				

2 семестр

Поточний контроль та самостійна робота								Контрольна робота, передбачена навчальна планом	Разом	Екзамен	Разом
ПР 7	ПР 8	ПР 9	ПР 10	ПР 11	ПР 12	ПР 13	ПР 14	12	60	40	100
6	6	6	6	6	6	6	6				

Критерії поточної оцінки знань студентів
(практична робота, крок оцінювання 1 бал, усього 6 балів)

Кількість балів	Критерії оцінки
1	Студент має фрагментарні знання при незначному загальному їх обсязі (менше половини навчального матеріалу) при відсутності сформованих умінь та навичок.
2	Студент має початковий рівень знань, значну (більше половини) частину навчального матеріалу може відтворити репродуктивно; може з допомогою викладача виконати просте навчальне завдання; має елементарні, нестійкі навички необхідні для виконання завдання.
3	Студент вміє аналізувати навчальний матеріал, в цілому самостійно застосовувати його на практиці; контролювати власну діяльність; самостійно визначити спосіб розв’язування навчальної задачі.
4	Студент вміє застосовувати вивчений матеріал у стандартних ситуаціях; може пояснити основні процеси, що відбуваються під час роботи інформаційної системи та наводити власні приклади на підтвердження деяких тверджень; вміє виконувати навчальні завдання.
5	Студент володіє міцними знаннями, 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.
6	Студент має 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
(контрольна робота, крок оцінювання 4 бала, усього 24 балів)
1 семестр

Кількість балів	Критерії оцінки
4	Студент демонструє фрагментарні знання при незначному загальному їх обсязі (менше половини навчального матеріалу).
8	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.
12	Студент знайомий з основними поняттями навчального матеріалу; може самостійно відтворити значну частину навчального матеріалу і робити певні узагальнення; вміє виконати просте навчальне завдання.
16	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.
20	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.
24	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
(контрольна робота, крок оцінювання 2 бала, усього 12 балів)
2 семестр

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 підсумкової оцінки знань студентів
(одне практичне питання екзаменаційного білету,
крок оцінювання 10 балів)

Кількість балів	Критерії оцінки
0	Робота виконана не в повному обсязі. Допущені грубі помилки. Робота виконаний не самостійно.
10	Студент має фрагментарні знання при незначному загальному їх обсязі (менше половини навчального матеріалу) за відсутності сформованих умінь та навичок.
20	Студент має рівень знань вищий, 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.
30	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.
40	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

10. Рекомендована література

Основна література

1. *Эванс, Э.* Предметно-ориентированное проектирование (Domain-Driven Development - DDD): Структуризация сложных программных систем: Пер. с англ. – М. «ООО И.Д. Вильямс», 2011. – 448с.
2. *Лаврищева К.М.* Електронний підручник «Програмна інженерія» Київського національного університету ім. Т. Г. Шевченка [Электронный ресурс] – Режим доступа: http://csc.knu.ua/uk/library/books/lavrishcheva-6.pdf
3. Ian Sommerville SoftwareEngineering / 9edition. - Addison Wesley, 2011 (access on URL: http://faculty.mu.edu.sa/public/uploads/1429431793.203Software%20Engineering%20by%20Somerville.pdf
4. Андон Ф.И. и др. Основы инженерии качества программных систем. – 2-е изд. - К.: «Академпериодика», 2007. – 632с..

5. *С. Амблер*. Гибкие технологии: экстремальное программирование и унифицированный процесс разработки. Библиотека программиста. – СПб.: Питер, 2005. – 412 с.
6. *Ларман, К.* Применение UML 2.0 и шаблонов проектирования: практическое руководство - : Пер. с англ. – М. ООО «И.Д. Вильямс», 2009. – 736 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. 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.
5. 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

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

1. Ткачук М.В., Гамзаєв Р.О., Мартінкус І.О., Нагорний К.А., Хруслов Методичні вказівки щодо виконання лабораторних робіт з навчальної дисципліни «Розробка та супровід проблемно-орієнтованих програмних систем» - ХНУ імені В.Н. Каразіна, 2020. - 77 с
2. Хруслов М.М., Ткачук М.В., Гамзаєв Р.О. Застосування уніфікованої мови моделювання UML для аналізу та проектування програмного забезпечення інформаційних систем - Дистанційний курс, **прот**. №3 від 27.02.2020 Науково-методична рада ХНУ імені В.Н. Каразіна, Сертифікат № 205/2020, 2020р.
3. 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. // http://repository.kpi.kharkov.ua/handle/KhPI-Press/31734.
4. http://www.computer.org/portal/web/swebok/html/contents
5. http://www.omg.org
6. http://www.intuit.ru
7. http://www.conference-service.com/conferences/software-engineering.html