

**Національний університет кораблебудування
імені адмірала Макарова
Херсонський навчально-науковий інститут**

Кафедра інформаційних технологій
та фізико-математичних дисциплін

T8125



ЗАТВЕРДЖЕНО
Заступник директора з
навчальної роботи

к.т.н., проф. Дудченко О.М.

РОБОЧА ПРОГРАМА НАВЧАЛЬНОЇ ДИСЦИПЛІНИ

Program of the Discipline

CASE-ЗАСОБИ РОЗРОБКИ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ

CASE TOOLS FOR SOFTWARE DEVELOPMENT

рівень вищої освіти *другий (магістерський)*

тип дисципліни *обов'язкова*

мова викладання *українська*

Херсон – 2024

Робоча програма навчальної дисципліни «CASE-засоби розробки програмного забезпечення» є однією із складових комплексної підготовки фахівців галузі знань 12 «Інформаційні технології» спеціальності 122 «Комп'ютерні науки» освітня професійна програма «Інформаційні управляючі системи та технології»

«26» серпня 2024 р. – 25с.

Розробники: Дудченко О.М. професор НУК кафедри інформаційних технологій та фізико-математичних дисциплін, канд. техн. наук; Карпова С.О., ст. викладач кафедри інформаційних технологій та фізико-математичних дисциплін.

Проект робочої програми навчальної дисципліни «CASE-засоби розробки програмного забезпечення» узгоджено з гарантом освітньої програми

Гарант освітньої програми «Інформаційні управляючі системи та технології»

д.т.н., доц.



П.Й. Гучек

Проект робочої програми навчальної дисципліни «CASE-засоби розробки програмного забезпечення» розглянуто на засіданні кафедри інформаційних технологій та фізико-математичних дисциплін

Протокол № 09 від «26» серпня 2024 р.

Завідувач кафедри



П.Й. Гучек

Робоча програма навчальної дисципліни «CASE-засоби розробки програмного забезпечення» затверджена методичною радою ХННІ НУК.

Протокол №01 від «28» травня 2024 р.

Голова МР ХННІ НУК



О.М. Дудченко

ЗМІСТ

| | |
|--|----|
| Вступ..... | 4 |
| 1. Опис навчальної дисципліни..... | 6 |
| 2. Мета навчальної дисципліни..... | 7 |
| 3. Передумови для вивчення дисципліни | 7 |
| 4. Очікувані результати навчання..... | 7 |
| 5. Програма навчальної дисципліни..... | 8 |
| 6. Методи навчання, засоби діагностики результатів навчання та методи їх демонстрування | 17 |
| 7. Форми поточного та підсумкового контролю..... | 17 |
| 8. Критерії оцінювання результатів навчання..... | 20 |
| 9. Засоби навчання..... | 21 |
| 10. Рекомендовані джерела інформації..... | 22 |
| Додаток..... | 24 |

Вступ

Анотація

Дисципліною «CASE-засоби розробки програмного забезпечення» передбачено набуття студентами знань про сучасні методології та засоби автоматизації розробки програмного забезпечення, про тестування програмного забезпечення, кодогенерації й реінжинірингу у CASE – засобах.

Програма навчальної дисципліни «CASE-засоби розробки програмного забезпечення» розрахована на студентів, які вивчили математичні основи, алгоритмічні принципи в моделюванні, проектуванні, розробці та супроводі інформаційних систем і технологій.

Програма навчальної дисципліни «CASE-засоби розробки програмного забезпечення» передбачає комплексне застосування набутих компетенцій для розв'язання прикладних задач, розробку, впровадження і супровід інтелектуальних систем аналізу та обробки даних в організаційних, технічних, природничих та соціально-економічних системах. Опанування курсу надає професійні компетенції для подальшого вивчення дисциплін професійної підготовки.

Для вивчення дисципліни «CASE-засоби розробки програмного забезпечення» необхідні знання з таких дисциплін: «Основи програмної інженерії», «Об'єктно-орієнтоване програмування», «Організація та технології передачі даних у комп'ютерних мережах», «Аналіз вимог до програмного забезпечення», «Конструювання програмного забезпечення», «Архітектура та проектування програмного забезпечення», «Якість програмного забезпечення та тестування», «Моделювання програмного забезпечення», «Безпека програм та даних».

Дисципліна «CASE-засоби розробки програмного забезпечення» носить міждисциплінарний характер, вона забезпечує підготовку студентів до вивчення навчальних дисциплін «Кваліфікаційна атестація», «Наукове стажування», «Управління проектами», та до написання магістерської роботи.

Ключові слова: вимоги, діаграми, кодогенерація, об'єктно-орієнтоване моделювання, програмне забезпечення, проектування, реінжиніринг, тестування, CASE – засоби, CASE – технології, UML.

Annotation

The discipline «CASE-software development tools» provides students with knowledge about modern methodologies and tools for automating software development, software testing, code generation and reengineering in CASE - tools.

The program of the discipline «CASE-software development tools» is designed for students who have studied the mathematical foundations and algorithmic principles

in modelling, design, development and maintenance of information systems and technologies.

The curriculum of the discipline «CASE-software development tools» provides a comprehensive application of the acquired competencies for solving applications, development, implementation and maintenance of intelligent systems, which are advanced software systems capable of autonomous decision-making and learning, for analysis and data processing in organizational, technical, natural and socio-economic systems. Mastering the course provides professional competencies for further study of training disciplines.

To study the discipline «CASE-software development tools» requires knowledge of the following disciplines: «Fundamentals of Software Engineering», «Object-Oriented Programming», «Organization and technology of data transmission in computer networks», «Analysis of software requirements», «Software design», «Software architecture and design», «Software quality and testing», «Software modeling», «Software and data security». These disciplines are essential as they provide the foundational knowledge and skills necessary for understanding and effectively applying the concepts and tools in the CASE-software development tools course.

The discipline «CASE-software development tools» is interdisciplinary; it prepares students to study the disciplines «Qualification Certification», «Scientific Internship», and «Project Management», and to write a master's thesis.

Keywords: *requirements, diagrams, code generation, object-oriented modelling, software, design, reengineering, testing, CASE - tools, CASE - technologies, UML.*

1. Опис навчальної дисципліни

| Найменування показників | Галузь знань, спеціальність (освітня програма), освітній рівень | Характеристика навчальної дисципліни | |
|---|---|--|-----------------------|
| | | денна форма навчання | заочна форма навчання |
| Кількість кредитів – 4 | Галузь знань 12 - «Інформаційні технології» | Обов'язкова | |
| Модулів - 6 | | Рік підготовки | |
| Змістових модулів - 6 | | 1-й | 1-й |
| Електронний адрес на сайті ХННІ НУК: http://kb.nuos.edu.ua/Licensing%20and%20accreditation%20specialties/software-engineering_.html | Спеціальність 122 - «Комп'ютерні науки» Освітня програма «Інформаційні управляючі системи та технології» | Семестри | |
| | | 1-й | 1-й |
| Індивідуальне науково-дослідне завдання - нема | | Лекції | |
| | | 1-й семестр – 30 год. | 14 год. |
| Загальна кількість годин - 120 | | Лабораторні | |
| | | 1-й семестр – 15 год. | 6 год. |
| Тижневих годин для денної форми навчання: аудиторних: 1-й семестр – 3 самостійної роботи студента: 1-й семестр – 5 | Освітній рівень: другий (магістерський) | Самостійна робота | |
| | | 1-й семестр – 75 год. | 100 год. |
| | | Індивідуальні завдання: год. | |
| | | - | - |
| | | Види контролю: 1-й семестр – залік | |
| Форма контролю: комбінована (письмовий контроль, тестовий контроль) | | | |

2. Мета навчальної дисципліни

2.1 Метою вивчення навчальної дисципліни «CASE-засоби розробки програмного забезпечення» є формування у студентів відповідно до освітньої програми таких компетентностей:

Інтегральна компетентність

– Здатність розв'язувати задачі дослідницького та/або інноваційного характеру у сфері комп'ютерних наук;

Загальні компетентності

ЗК01. Здатність до абстрактного мислення, аналізу та синтезу.

ЗК02. Здатність застосовувати знання у практичних ситуаціях.

ЗК05. Здатність вчитися й оволодівати сучасними знаннями.

СК02. Здатність формалізувати предметну область певного проекту у вигляді відповідної інформаційної моделі.

3. Передумови для вивчення дисципліни

Передумовами для вивчення даної дисципліни є дисципліни: математичне моделювання систем і процесів та методи оптимізації, сучасні засоби колективної розробки інформаційних систем.

4. Очікувані результати навчання

Вивчення навчальної дисципліни передбачає формування та розвиток у студентів таких результатів навчання:

РН06. Розробляти концептуальну модель інформаційної або комп'ютерної системи.

РН10. Проектувати архітектурні рішення інформаційних та комп'ютерних систем різного призначення.

РН12. Проектувати та супроводжувати бази даних та знань.

РН18. Збирати, формалізувати, систематизувати і аналізувати потреби та вимоги до інформаційної або комп'ютерної системи, що розробляється, експлуатується чи супроводжується

5. Програма навчальної дисципліни

1-й семестр

Модуль 1.

Змістовий модуль 1. Сучасні CASE – технології. Методологічні основи CASE – технологій

Тема 1. Вступ. Мета та задачі вивчення дисципліни. Методологія об'єктно-орієнтованого програмування. CASE- засоби та технології розробки програмного забезпечення.

Джерела інформації: [3] – стор. 135-154; [11].

Тема 2. Введення до мови UML. Моделювання у процесі розробки програмного забезпечення. Об'єктне моделювання. Основні концепції UML.

Джерела інформації: [4] – стор. 26-92; [5] – стор. 26 – 58, 166-200; [7] – стор. 450-465.

Модуль 2.

Змістовий модуль 2. Уніфікована мова моделювання програмного забезпечення в об'єктній методології UML

Тема 3. Моделювання поведінки системи діаграмами Use Case (варіантів використання). Актори й елементи Use Case. Відношення на діаграмах Use Case. Специфікація елементів Use Case. Розробка вимог.

Джерела інформації: [1] – стор. 202-220; [4] – стор. 93-117; [5] – стор. 229 – 244; [7] – стор. 62 – 71, 158 -199;.

Тема 4. Діаграми взаємодії. Діаграми послідовності. Об'єкти, фокуси управління. Діаграми кооперації. Потоки управління у часі, структура потоків управління.

Джерела інформації: [1] – стор. 221-233; [4] – стор. 183 -214; [5] – стор. 222 – 229.

Тема 5. Діаграми класів. Класифікатори. Видимість, Область дії. Відношення у діаграмах класів. Шаблони. Екземпляри, Діаграми об'єктів.

Джерела інформації: [1] – стор. 59 – 70, 119-133, 150 – 160, 172 - 187; [4] – стор. 118 -144; [5] – стор. 201 – 214.

Тема 6. Діаграми станів. Діаграми діяльності.

Джерела інформації: [1] – стор. 234 - 307; [4] – стор. 145 -182; [5] – стор. 215 – 221.

Тема 7. Діаграми кооперації. Паттерни й їх класифікація. Каркаси. Об'єкти. Основні пакети мови UML. Зв'язки. Повідомлення.

Джерела інформації: [1] – стор. 161 – 171, 340 - 350; [5] – стор. 244 – 256; [7] – стор. 101 -103.

Тема 8. Моделі реалізації. Діаграми компонентів і розгортання.

Джерела інформації: [1] – стор. 309 – 329, 351 – 371; [5] – стор. 256 – 267, 277 - 282.

Змістовий модуль 3. Особливості моделювання на UML у середовищі CASE – засобу Rational Rose

Тема 9. Особливості моделювання на UML у середовищі CASE – засобу Rational Rose.

Джерела інформації: [4] – стор. 235 – 256; [5] – стор. 387 – 420; [6] – стор. 25 – 36.

Тема 10. Організація процесу розробки програмного забезпечення у середовищі Rational Rose.

Джерела інформації: [8] – стор. 21 – 64.

Модуль 3.

Змістовий модуль 4. Раціональний уніфікований процес розробки програмного забезпечення (RUP)

Тема 11. Rational Unified Process (RUP) як методологія. Структура життєвого циклу проекту.

Джерела інформації: [5] – стор. 315-352; [7] – стор. 32-136.

Тема 12. Робочі потоки RUP. Розробка моделі варіантів використання у середовищі CASE – засобу Rational Rose. Конструювання класів. Моделювання взаємодії об'єктів.

Джерела інформації: [5] – стор. 112-121; [7] – стор. 140-341; [8] – стор. 65-70, 103-117.

Тема 13. Кодогенерація та реінженірінг у CASE – засобі Rational Rose

Джерела інформації: [7] – стор. 25-70, [8] – стор. 287-296, стор. 519-524, [9]

Модуль 4.

Змістовий модуль 5. CASE – засоби розробки та тестування програмного забезпечення

Тема 14. Тестування програмного забезпечення. CASE – засоби тестування. Джерела інформації: [5] – стор. 121- 169, 358-389.

Змістовий модуль 6. Сучасний стан розвитку CASE – засобів

Тема 15. Тенденції розвитку та стан ринку об'єктно-орієнтованих CASE – засобів. Моделювання програмних систем. Проблеми та процес впровадження CASE – засобів. Очікувані та несподівані результати. Вибір CASE – засобів. Джерела інформації: [10]

5.1 Тематичний план навчальної дисципліни

| 3. Назви змістових модулів і тем | Кількість годин | | | | | | | | | |
|--|-----------------|--------------|---|----------|-----------|--------------|--------------|---|----------|-----------|
| | денна форма | | | | | заочна форма | | | | |
| | усього | у тому числі | | | | усього | у тому числі | | | |
| | | л | п | лаб | с.р. | | л | п | лаб | с.р. |
| 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 |
| 1-й семестр | | | | | | | | | | |
| Модуль 1 | | | | | | | | | | |
| Змістовий модуль 1. Вступ. Сучасні CASE – технології. Методологічні основи CASE – технологій | | | | | | | | | | |
| Тема 1. Вступ. Мета та задачі вивчення дисципліни. Методологія об'єктно-орієнтованого програмування. CASE - засоби та технології розробки програмного забезпечення. | 15 | 2 | - | - | 12 | - | 1 | - | - | 14 |
| Тема 2. Введення до мови UML. Моделювання у процесі розробки програмного забезпечення. Об'єктне моделювання. Основні концепції UML. | 15 | 2 | - | - | 14 | - | - | - | - | 15 |
| Разом за модулем 1 | 30 | 4 | | - | 26 | 30 | 1 | | - | 29 |
| Модуль 2 | | | | | | | | | | |
| Змістовий модуль 2. Уніфікована мова моделювання програмного забезпечення в об'єктній методології UML | | | | | | | | | | |
| Тема 3. Моделювання поведінки системи діаграмами Use Case (варіантів використання). Актори й елементи Use Case. Відношення на діаграмах Use Case. Специфікація елементів Use Case. Розробка вимог. | 3 | 2 | - | 1 | 2 | - | 5 | - | 2 | 2 |
| Тема 4. Діаграми взаємодії. Діаграми послідовності. Об'єкти, фокуси управління. Діаграми кооперації. Потоки управління у | 3 | 2 | - | 1 | 1 | - | - | - | - | 2 |

| | | | | | | | | | | |
|--|-----------|-----------|----------|----------|-----------|-----------|----------|----------|----------|-----------|
| часі, структура потоків управління. | | | | | | | | | | |
| Тема 5. Діаграми класів. Класифікатори. Видимість, Область дії. Відношення у діаграмах класів. Шаблони. Екземпляри, Діаграми об'єктів. | 3 | 2 | - | 1 | 1 | | | | | 3 |
| Тема 6. Діаграми станів. Діаграми діяльності. | 3 | 2 | - | 0,5 | 1 | | | | | 3 |
| Тема 7. Діаграми кооперації. Паттерни й їх класифікація. Каркаси. Об'єкти. Основні пакети мови UML. Зв'язки. Повідомлення. | 3 | 2 | - | 0,5 | 1 | | | | | 3 |
| Тема 8. Моделі реалізації. Діаграми компонентів і розгортання. | 3 | 2 | - | 0,5 | 1 | | | | | 3 |
| Змістовий модуль 3. Особливості моделювання на UML у середовищі CASE – засобу Rational Rose | | | | | | | | | | |
| Тема 9. Особливості моделювання на UML у середовищі CASE – засобу Rational Rose. | 3 | 2 | - | 0,5 | 1 | | | | | 2 |
| Тема 10. Організація процесу розробки програмного забезпечення у середовищі Rational Rose | 4 | 2 | - | | 1 | | | - | 1 | 3 |
| Разом за модулем 2 | 30 | 16 | - | 5 | 9 | 30 | 6 | | 3 | 21 |
| Модуль 3 | | | | | | | | | | |
| Змістовий модуль 4. Раціональний уніфікований процес розробки програмного забезпечення (RUP) | | | | | | | | | | |
| Тема 11. Rational Unified Process (RUP) як методологія. Структура життєвого циклу проекту. | 5 | 2 | - | 1 | 6 | | | | | 8 |
| Тема 12. Робочі потоки RUP. Розробка моделі варіантів використання у середовищі CASE – засобу Rational Rose. Конструювання класів. Моделювання взаємодії об'єктів. | 10 | 2 | - | 1 | 6 | | 3 | - | 1 | 8 |
| Тема 13. Кодогенерація та реінженірінг у CASE – засобі Rational Rose | 15 | 2 | - | 2 | 8 | | | | | 10 |
| Разом за модулем 3 | 30 | 6 | - | 4 | 20 | 30 | 3 | - | 1 | 26 |

| Модуль 4 | | | | | | | | | | |
|---|------------|-----------|----------|-----------|-----------|------------|-----------|---|----------|------------|
| Змістовий модуль 5. CASE – засоби розробки та тестування програмного забезпечення | | | | | | | | | | |
| Тема 14. Тестування програмного забезпечення. CASE – засоби тестування. | 15 | 2 | - | 6 | 10 | | 4 | - | 2 | 14 |
| Змістовий модуль 6. Сучасний стан розвитку CASE – засобів | | | | | | | | | | |
| Тема 15. Тенденції розвитку та стан ринку об'єктно-орієнтованих CASE – засобів. Моделювання програмних систем. Проблеми та процес впровадження CASE – засобів. Очікувані та несподівані результати. Вибір CASE – засобів. | 15 | 2 | - | - | 10 | | - | - | - | 10 |
| Разом за модулем 4 | 30 | 4 | - | 6 | 20 | 30 | 4 | | 2 | 24 |
| Разом | 120 | 30 | - | 15 | 75 | 120 | 14 | | 6 | 100 |

Примітка. Для студентів заочної форми навчання викладаються оглядові лекції за темами змістових модулів в обсягах відповідно до таблиці.

5.2 Теми лабораторних занять

| № з/п | Назва теми | Кількість годин | |
|--------------------|---|-----------------|--------------|
| | | Денна форма | Заочна форма |
| 1-й семестр | | | |
| 1 | Робота у середовищі CASE – засобу Visual Paradigm for uml. Розробка діаграм Use Case у середовищі CASE – засобу Visual Paradigm for UML. Джерела інформації: [1,2,4,5,7], робота 1. | 1 | 0,5 |
| 2 | Розробка діаграм взаємодії у середовищі CASE – засобу Visual Paradigm for UML. Джерела інформації: [1,2,4,5], робота 2. | 1 | 0,5 |
| 3 | Розробка діаграм класів у середовищі CASE – засобу Visual Paradigm for UML. Джерела інформації: [1,2,4,5], робота 3. | 1 | 1 |
| 4 | Розробка діаграм станів і діяльності у середовищі CASE – засобу Visual Paradigm for UML. Джерела інформації: [1,2,5], робота 4. | 1 | 1 |
| 5 | Розробка діаграм реалізації у середовищі CASE – засобу Visual Paradigm for UML. Джерела інформації: [1,2,5,9], робота 5. | 1 | 1 |
| 6 | Робота у середовищі CASE – засобу Rational Rose. Генерація програмного коду C++, Java IDL. Джерела інформації: [2, 4-9], робота 6. | 4 | 1 |
| 7 | Тестування програмного забезпечення. Джерела інформації: [2,5,10], робота 7. | 6 | 1 |
| Разом | | 15 | 6 |

5.3 Самостійна робота

| № з/п | Назва теми | Кількість годин | |
|-------|--|-----------------|--------------|
| | | денна форма | заочна форма |
| 1 | Методологія процедурно-орієнтованого програмування. | 0,5 | 2 |
| 2 | Методологія об'єктно-орієнтованого програмування. | 0,5 | 2 |
| 3 | Методологія об'єктно-орієнтованого аналізу та проектування. | 0,5 | 2 |
| 4 | Методологія системного аналізу та системного моделювання. | 0,5 | 2 |
| 5 | Теорія множин. Теорія графів. Семантичні мережі | 0,5 | 2 |
| 6 | Діаграми структурного системного аналізу. Діаграми «сутність-зв'язок». | 0,5 | 2 |
| 7 | Діаграми функціонального моделювання. Діаграми потоків даних. | 0,5 | 2 |
| 8 | Основні етапи розвитку UML. Предмети в UML. Відношення в UML. Відношення асоціації, розширення, узагальнення, включення. | 0,5 | 2 |
| 9 | Об'єкти, зв'язки, повідомлення в мові UML. Компоненти, інтерфейси, залежності в мові UML. Вузол і з'єднання в мові UML. | 0,5 | 2 |
| 10 | Ім'я, атрибути й операції класу. Відношення залежності, асоціації, агрегації, композиції й узагальнення | 0,5 | 2 |
| 11 | Ім'я стану. Список внутрішніх дій. Початковий та кінцевий стан. Події(event), сторожева умова(guard condition), вираз дії (action expression). Послідовні підстани (sequential substates), паралельні підстани (concurrent substates). Переходи між паралельними станами. Переходи між складовими станами. Синхронізуючі стани | 0,5 | 2 |
| 12 | Лінія життя об'єкта. Фокус управління. Галуження потоку управління. Стереотипи повідомлень. Тимчасові обмеження на діаграмах послідовності | 0,5 | 2 |
| 13 | Патерни проектування в нотації мови UML. Патерн «Фасад» та його позначення в нотації мови UML. Патерн «Спостерігач» та його позначення в нотації мови UML. Мультиоб'єкт. Активний та составний об'єкт. Стереотипи зв'язків. Формат запису повідомлень | 0,5 | 2 |
| 14 | Ім'я компоненту. Види компонентів. | 0,5 | 2 |
| 15 | Загальна характеристика CASE-засоба Rational Rose. Особливості робочого інтерфейсу Rational Rose. | 0,5 | 2 |

| | | | |
|--------------|---|-----------|-----------|
| 16 | Стапи й ітерації. Ітеративність. Орієнтація на архітектуру. Процес, керований прикладами. | 0,5 | 2 |
| 17 | Тестування методом чорного ящика. Структурне тестування. | 2 | 4 |
| 18 | Тестування гілок. Низхідне та висхідне тестування. | 2 | 4 |
| 19 | Підготовка моделі для генерації програмного кода. Вибір мови програмування та редагування властивостей. генерації програмного кода. | 1 | 2 |
| 20 | Визначення потреб в CASE-засобах. Оцінка та вибір CASE-засобів. Приклад підходу до визначення критеріїв вибору CASE-засобів. | 0,5 | 2 |
| 21 | Перехід до практичного використання CASE-засобів. Приклади комплексів CASE-засобів. | 0,5 | 2 |
| 22 | Генерація програмного коду C++ і Visual C++, Java, Visual Basic, PowerBuilder, CORBA/IDL, DDL, Oracle | 0,5 | 2 |
| 23 | Обернене проектування програмного коду C++, Java, Visual Basic, PowerBuilder, Oracle | 0,5 | 2 |
| Разом | | 15 | 50 |

5.4 Розподіл годин самостійної роботи

| № з/п | Вид роботи | Кількість годин | |
|--------------------|--|-----------------|--------------|
| | | денна форма | заочна форма |
| 1-й семестр | | | |
| 1 | Підготовка до лекцій | 15 | 10 |
| 2 | Підготовка до лабораторних робіт | 15 | 10 |
| 3 | Підготовка до поточного модульного контролю | 15 | - |
| 4 | Виконання контрольної роботи | - | 15 |
| 5 | Самостійне вивчення тем, що не входять до лекційного курсу | 15 | 50 |
| 6 | Підготовка до підсумкового контролю | 15 | 15 |
| Разом | | 75 | 100 |

6. Методи навчання, засоби діагностики результатів навчання та методи їх демонстрування

Методи навчання:

для всіх видів занять:

- робота з літературою - опрацювання різних видів джерел, спрямоване на формування нових знань, їх закріплення, вироблення вмінь і навичок та реалізацію контрольної-корекційної функції в умовах формальної освіти;

для лекційних занять:

- лекція - усний виклад навчального матеріалу, який характеризується великим обсягом, складністю логічних побудов, сконцентрованістю розумових образів, доведень і узагальнень;

- відеометод - використання відеоматеріалів для активізації наочно-чуттєвого сприймання; забезпечує більш легке і міцне засвоєння знань в їх образно-понятійній цілісності та емоційній забарвленості;

для лабораторних занять:

- лабораторна робота - метод поглиблення і закріплення теоретичних знань шляхом створення програм і отримання результатів роботи програми з використанням комп'ютерів;

- інструктаж - ознайомлення зі способами виконання завдань, інструментами, матеріалами, технікою безпеки та організацію робочого місця.

Засобами оцінювання та методами демонстрування результатів навчання є:

- звіти з виконання лабораторної роботи та презентації результатів виконаних лабораторних робіт на комп'ютері (або письмовий контроль результатів);

- усні відповіді на лабораторних заняттях;

- поточні модульні контрольні роботи у формі тестування (тестовий контроль);

- залік.

7. Форми поточного та підсумкового контролю

Досягнення студента оцінюються за 100-бальною системою Університету.

Підсумкова оцінка навчального курсу включає в себе оцінки з поточного контролю і оцінки заключного іспиту.

Питома вага заключного іспиту в загальній системі оцінок – **40 балів**. Право здавати заключний іспит дається студенту, якій з урахуванням максимальних балів проміжних оцінок і заключного іспиту набирає не менше **60 балів**. Підсумкова оцінка навчального курсу є сумою проміжних оцінок і оцінки іспиту.

Поточний контроль проводиться на кожному лабораторному занятті та за результатами виконання завдань самостійної роботи. Він передбачає оцінювання теоретичної підготовки здобувачів вищої освіти із зазначеної теми (у тому числі, самостійно опрацьованого матеріалу) під час виконання завдань лабораторних робіт.

Зарахування кредитів навчального курсу можливо тільки після досягнення результатів, запланованих РПНД, що виражається в одній з позитивних оцінок, передбачених чинним законодавством.

7.1 Форми контролю результатів навчальної діяльності студентів та їх оцінювання

Критерії оцінювання лабораторних робіт

| Бал | Критерії оцінювання |
|-----|---|
| 5 | Робота виконана у встановлений термін. Виконана самостійно, чітко сформульовані цілі, завдання та гіпотеза досліджень. Застосовувалися коректні методи обробки отриманих результатів. У висновках проведена коректна інтерпретація результатів. |
| 4 | Робота виконана у встановлений термін. Студент виконує лабораторну роботу згідно з інструкцією, іноді після консультації викладача; описує спостереження; в цілому правильно складає звіт та робить висновки. |
| 3 | Робота виконана з порушенням встановлених термінів. Студент виконує лабораторну роботу згідно з інструкцією, іноді після консультації викладача; описує спостереження; складає звіт, що містить неточності у висновках та помилки. |
| 2 | Робота виконана з порушенням встановлених термінів. Студент виконує лабораторну згідно з інструкцією; складений звіт містить неточності у висновках та помилки. |
| 1 | Робота виконана з порушенням встановлених термінів. Студент виконує лабораторну під керівництвом викладача; складений звіт містить неточності у висновках та помилки. |
| 0 | Робота не виконувалася |

Критерії оцінювання поточного модульного контролю знань у формі тестування

| | | | | | | | | | | |
|--------------------------|-----|----|----|----|----|----|----|----|----|----|
| Правильних відповідей, % | 100 | 90 | 80 | 70 | 60 | 50 | 40 | 30 | 20 | 10 |
| 10-й семестр | | | | | | | | | | |
| Бал | 20 | 18 | 16 | 14 | 12 | 10 | 8 | 6 | 4 | 2 |

Критерії оцінювання контрольної роботи (для заочної форми)

| Бал | Критерії оцінювання |
|-----|---|
| 25 | Робота виконана у встановлений термін. Матеріал викладено у достатньому обсязі, аргументовано і у правильній послідовності. Під час захисту роботи студент вільно орієнтується в матеріалах. |
| 15 | Робота виконана у встановлений термін. Матеріал викладено у достатньому обсязі, але частка програм наведена без результатів розрахунків. Під час захисту роботи студент вільно орієнтується в матеріалах. |
| 10 | Робота виконана з порушенням встановлених термінів. Матеріал викладено у правильній послідовності, але недостатньо повно, більша частка програм наведена без результатів розрахунків. Під час захисту роботи студент слабо орієнтується в матеріалах. |
| 0 | Роботу не виконано. |

Критерії оцінювання підсумкового контролю

| Бал | Критерії оцінювання |
|-----|--|
| 40 | Студент зробив роботу самостійно без помилок та відповідає на теоретичні питання без помилок |
| 30 | Студент зробив роботу самостійно без помилок, але відповіді на теоретичні питання не повні |
| 20 | Студент зробив роботу з незначними помилками, але відповідає на теоретичні питання без помилок |
| 10 | Студент зробив роботу з суттєвими помилками, але відповідає на теоретичні питання без помилок |
| 0 | Студент не зробив роботу і не відповідає на теоретичні питання без помилок |

Узагальнюючі результати поточного контролю знань

| Форма контролю | Максимальна кількість балів | |
|------------------------------|-----------------------------|------------------------------|
| | Денна форма | Заочна форма |
| 1-й семестр | | |
| Виконання лабораторних робіт | 7 роб. × 5 балів = 35 балів | 7 роб. × 5 балів = 35 балів |
| Поточний модульний контроль | 1 МКР × 25 балів = 25 балів | - |
| Виконання контрольних робіт | - | 1 роб. × 25 балів = 25 балів |
| Всього | 60 | 60 |

8. Критерії оцінювання результатів навчання

| Змістовий модуль | Тема | Денна форма | | Заочна форма | |
|----------------------|---------|-----------------------------|------------|------------------------|------------|
| | | Вид роботи | Бали | Вид роботи | Бали |
| 1 | 2 | 3 | 4 | 5 | 6 |
| 1-й семестр | | | | | |
| ЗМ 1-ЗМ 2 | T1-T3 | Лабораторна робота № 1 | 5 | Лабораторна робота № 1 | 5 |
| | T4 | Лабораторна робота № 2 | 5 | Лабораторна робота № 2 | 5 |
| | T5 | Лабораторна робота № 3 | 5 | Лабораторна робота № 3 | 5 |
| | T6 | Лабораторна робота № 4 | 5 | Лабораторна робота № 4 | 5 |
| | T7-T8 | Лабораторна робота № 5 | 5 | Лабораторна робота № 5 | 5 |
| | T1-T8 | Поточний модульний контроль | 25 | - | - |
| ЗМ 3 | T9-T13 | Лабораторна робота № 6 | 5 | Лабораторна робота № 6 | 5 |
| ЗМ 4 | T14-T15 | Лабораторна робота № 7 | 5 | Лабораторна робота № 7 | 5 |
| | | - | - | Контрольна робота | 25 |
| Підсумковий контроль | | Тест | 40 | Тест | 40 |
| Сума | | | 100 | | 100 |

9. Засоби навчання

Технічні засоби навчання: мультимедійний проектор, персональні комп'ютери з підключенням до мережі Інтернет.

При проведенні занять за дистанційною формою навчання (у період карантину) використовуються дистанційні платформи й інформаційно-комунікаційні технології (Moodle, Google Classroom, DingTalk, ZOOM Cloud Meetings, Skype, Viber, WeChat, Telegram, соціальні мережі тощо).

10. Рекомендовані джерела інформації

Основна література

1. Буч Г., Рамбо Дж. и др. Язык UML. Руководство пользователя: Пер. с англ. – М: ДМК, 2000. –432 с., ил.
2. Дудченко О. М., Карпова С. О. CASE-засоби розробки та тестування програмного забезпечення: методичні вказівки до виконання лабораторних робіт. Миколаїв: НУК, 2018. - 84 с.
3. Калянов Г.Н. CASE. Структурный системный анализ М.: «Лори», 1996. – 241 с. ил.
4. Леоненков А.В. Самоучитель UML. СПб: БХВ, Питербург, 2002, -304 с. ил.
5. Орлов С.А. Технологии разработки программного обеспечения: Учебник/ СПб: Питер, 2002. 464 с. ил.
6. Трофимов С.А. CASE –технологии: практическая работа в Rational Rose. Изд. 2-е. - М.: Бинوم - Пресс, 2002 г.. –208 с. ил.
7. Якобсон А., Буч Г., Рамбо Дж. Унифицированный процесс разработки программного обеспечения. -СПб: Питер, 2002. 496 с. ил.
8. Уэнди Боггс, Майкл Боггс UML и Rational Rose- М.: Лори, 2001. - 582 с.
9. Viktor Farcic, Alex Garcia Test-Driven Java Development.: Published by Packt Publishing Ltd. Livery Place 35 Livery Street Birmingham B3 2PB, UK., 2015. - 258 pages
10. <http://www.citforum.ru/database/case/index.shtml> - CASE-технологии. Современные методы и средства проектирования информационных систем.
11. <http://www.citforum.ru/programming/rational/index.shtml> - Средства тестирования приложений для разработчиков. Инструменты от компании Rational Software.

Допоміжна література

1. <http://www.citforum.ru/programming/rational/index.shtml> - Средства тестирования приложений для разработчиков. Инструменты от компании Rational Software.
2. <http://www.citforum.ru/programming/rational/rcrm.shtml> - Управление тестированием, разработкой и конфигурацией на основе Rational Change Request Management.
3. <http://www.citforum.ru/programming/rational/rationaltest.shtml> - Средства тестирования от компании Rational.
4. http://www.citforum.ru/programming/digest/ucm_rational.shtml - Управление изменениями, тестированием и документированием с использованием технологий Rational
5. <http://www.citforum.ru/database/case/index.shtml> - CASE-технологии. Современные методы и средства проектирования информационных систем.
6. <https://habrahabr.ru/company/mailru/blog/277543/>
7. <https://msdn.microsoft.com/ru-ru/library/hh212233.aspx>

8. http://storage.piter.com/upload/contents/978549602091/978549602091_p.pdf
<http://www.slideshare.net/vyacheslavlyalkin/ss-27124432>

Інформаційні ресурси в Інтернет

1. Сайт ХННІ НУК: <http://kb.nuos.edu.ua>
2. <https://courses.prometheus.org.ua/>
3. <https://eduhub.in.ua/courses>
4. <https://ru.coursera.org/>

Розробник
к.т.н., професор НУК



Дудченко О.М.

Розробник
ст. викладач

Карпова С.О.

Питання для модульного контролю

Контрольні питання до 1-го модуля

1. Поясніть аббревіатуру CASE.
2. Розкрийте поняття CASE – технології та CASE – засобів.
3. Методологічна основа CASE – технологій.
4. Наведіть класифікацію CASE – технологій.
5. Склад, структура та функції CASE – засобів.
6. Наведіть класифікацію CASE – засобів.
7. Призначення UML.
8. Призначення діаграмам варіантів використання. З яких елементів складаються діаграми Use Case?

Контрольні питання до 2-го модуля

1. Який порядок побудови моделі вимог?
2. Яке призначення діаграм взаємодії?
3. Що є спільне у діаграм послідовності та кооперації? Які відмінності між ними?
4. Яке призначення діаграм класів? Які відношення використовуються на діаграмах класів?
5. Елементи діаграм станів. Яке призначення діаграм станів? Елементи діаграм діяльності. Призначення цих діаграм станів?
6. Що таке кооперація в UML? Які складові кооперації?
7. Що таке паттерн?
8. Яке призначення пакетів в UML? Як визначається видимість змісту пакетів?
9. Що таке каркаси в UML?
10. Яке призначення моделей реалізації?
11. Які елементи складають діаграми компонентів?
12. Які елементи складають діаграми розташування?

Контрольні питання до 3-го модуля

1. Що являє собою CASE – засіб Rational Rose?
2. Які можливості надає CASE – засіб Rational Rose?
3. Які види діаграм можуть використовуватись у середовищі CASE – засобу Rational Rose?
4. Яка методологія розробки ПЗ у середовищі CASE – засобу Rational Rose?
5. Яку структуру має раціональний уніфікований процесу розробки ПЗ?
6. Які етапи уніфікованого процесу розробки ПЗ?
7. Які робочі потоки входять до складу уніфікованого процесу розробки ПЗ?
8. Які моделі розробляються в уніфікованому процесі розробки ПЗ?
9. Яка сутність управління ризиками?

10. Яка роль визначення вимог у життєвому циклі ПЗ? Які цілі процесу визначення вимог?
11. Що таке модель застосовної області? Як використовується модель застосовної області?
12. Що таке бізнес-модель?
13. Яка роль аналізу у життєвому циклі ПЗ? Що складає модель аналізу?
14. Які види діяльності складають робочий процес аналізу?
15. Яка роль проектування у життєвому циклі ПЗ? Які цілі процесу проектування? Які моделі створюються у процесі проектування?
16. Які види діяльності складають робочий процес проектування?
17. Яка роль реалізації у життєвому циклі ПЗ? Які моделі створюються у процесі реалізації? Які види діяльності складають робочий процес реалізації?
18. Як створюється нова модель? Як змінювати формат діаграм? Які доступні можливості зміни виду діаграм? Як налаштовується робочий стіл Rational Rose?
19. Як використовується менеджер Add-In?

Контрольні питання до 4-го модуля

1. Що таке тестування? Що таке тестування ПО?
2. CASE – засоби тестування?
3. Що являє собою CASE – засіб OBJECTEERING? Які можливості надає CASE – засіб OBJECTEERING?
4. Які види діаграм можуть використовуватись у середовищі CASE – засобу OBJECTEERING ?
5. Що являє собою CASE – засіб QUANTIFY? Які можливості надає CASE – засіб QUANTIFY?
6. Які види тестувань можуть виконуватись у середовищі CASE – засобу QUANTIFY ?
7. Що являє собою CASE – засіб PURIFY? Які можливості надає CASE – засіб PURIFY?
8. Які види тестувань можуть використовуватись у середовищі CASE – засобу PURIFY ?
9. Що являє собою CASE – засіб PURECOVERAGEY? Які можливості надає CASE – засіб PURECOVERAGE?
10. Які види інформації з тестувань може отримуватись у середовищі CASE – засобу QUANTIFY ?
11. Які CASE – засоби можуть бути застосовані зараз в процесі розробки ПЗ?
12. Які проблеми впровадження CASE – засобів?
13. Процес впровадження CASE – засобів?
14. Які напрями розвитку CASE – засобів ?
15. Перспективи розвитку CASE – засобів?