

АНОТАЦІЯ ВИБІРКОВОЇ ДИСЦИПЛІНИ

| | |
|--|---|
| Назва дисципліни / освітнього компонента | «Методи об'єктного аналізу і моделювання» |
| Освітня програма | «Інженерія програмного забезпечення» |
| Компонент освітньої програми | Вибірковий |
| Загальна кількість кредитів та кількість годин для вивчення дисципліни | 3,0 кредити / 90 годин |
| Вид підсумкового контролю | Залік |
| Мова викладання | Українська |
| Викладач | Мащенко Володимир Андрійович , докт. тех. наук, доцент, професор кафедри інформаційних технологій та моделювання |
| CV викладача на сайті кафедри | https://kitm.rshu.edu.ua |
| E-mail викладача | v.a.mashchenko@nuwm.edu.ua |
| Консультації | Згідно з графіком консультацій |

МЕТА ТА ЗАВДАННЯ НАВЧАЛЬНОЇ ДИСЦИПЛІНИ

Навчальна дисципліна «Методи об'єктного аналізу і моделювання» належить до вибіркового компонента циклу професійної підготовки для першого (бакалаврського) рівня вищої освіти за спеціальністю 121 Інженерія програмного забезпечення. Вона вивчається у 3 семестрі після освоєння таких дисциплін: «Дискретний аналіз», «Програмування», «Алгоритми і структури даних», «Математична логіка». Вибірковий компонент логічно поєднується з обов'язковим компонентом «Об'єктно-орієнтоване програмування», забезпечуючи методологічний фундамент для його практичного застосування та закладаючи основу для подальших дисциплін з моделювання й проектування програмних систем.

Об'єктний підхід є ключовим у сучасній інженерії програмного забезпечення, оскільки дозволяє створювати масштабовані, гнучкі та зрозумілі програмні системи. Дисципліна спрямована на формування цілісної системи теоретичних знань та практичних умінь і навичок у сфері *об'єктно-орієнтованого аналізу (ООА)*, *об'єктно-орієнтованого проектування (ООПр)* та *об'єктно-орієнтованого програмування (ООП)*. Вона забезпечує студентів методологічною основою для розуміння складних програмних систем як сукупності взаємодіючих об'єктів, що володіють унікальними властивостями, реалізують індивідуальну поведінку та мають зв'язки. Подолання проблеми надмірної складності задачі аналізу, моделювання та проектування складних систем здійснюється через різні види декомпозиції – структурну, функціональну, об'єктну, що дозволяє переходити від абстрактних моделей до конкретних архітектурних рішень і програмної реалізації.

Мета викладання дисципліни. Метою викладання є формування у здобувачів здатності мислити об'єктами, аналізувати предметні області, виділяти сутності та їхні взаємозв'язки, будувати моделі складних систем і трансформувати їх у архітектурні рішення. Дисципліна спрямована на розвиток системного мислення, що дозволяє поєднувати теоретичні знання з практичними навичками, а також забезпечує підготовку до роботи з великими та складними програмними продуктами.

Цілі та завдання дисципліни:

- оволодіння визначальними принципами та методами об'єктного аналізу і проектування;
- вивчення принципів та методів побудови моделей складних систем із застосуванням різних видів декомпозиції (структурна, функціональна, об'єктна);
- формування практичних умінь і навичок виділення в предметній області ключових об'єктів, встановлення взаємозв'язків між ними та побудови моделей предметних областей;
- розвиток здатності інтегрувати результати об'єктного аналізу та проектування у процес розробки програмного забезпечення;
- розвиток уміння застосовувати методи аналізу, моделювання і проектування для вирішення прикладних задач в сфері інформаційних технологій та програмної інженерії;
- розвиток здатності до архітектурного мислення та системного бачення.

Очікувані результати навчання. Оволодіння навчальною дисципліною забезпечить формування у здобувачів вищої освіти цілісної системи знань, вмінь та навичок у сфері об'єктно-орієнтованих аналізу, проектування та програмування при розробленні складних програмних систем різного призначення.

Студенти мають знати:

- основи об'єктного аналізу та проектування;
- принципи побудови моделей складних систем;
- методи архітектурного мислення;
- підходи до перевірки та валідації моделей.

Студенти мають вміти:

- аналізувати предметні області, виділяючи ключові об'єкти, та формалізувати їх у вигляді об'єктних моделей;
- будувати моделі взаємодії об'єктів;
- будувати архітектурні рішення на основі об'єктного підходу;
- формалізувати вимоги та узгоджувати їх із архітектурними рішеннями;
- застосовувати методи ООА та ООПр для моделювання і проектування програмних систем;
- застосовувати об'єктний підхід у програмній реалізації.
- інтегрувати результати моделювання у процес програмної реалізації;
- демонструвати здатність до командної роботи та міждисциплінарної комунікації.

ЗМІСТ НАВЧАЛЬНОЇ ДИСЦИПЛІНИ

Тема 1. Основи об'єктного аналізу і моделювання. Об'єктне мислення як методологія інженерії програмного забезпечення. Поняття об'єкта, класу, властивостей і поведінки. Взаємозв'язки між об'єктами та їхня роль у системі. Декомпозиція як спосіб подолання складності. Порівняння структурного, функціонального та об'єктного підходів. Приклади застосування об'єктного мислення у C++.

Тема 2. Аналіз предметної області. Визначення меж предметної області. Виділення сутностей та їхніх характеристик. Встановлення взаємозв'язків між об'єктами. Формалізація вимог у вигляді об'єктних моделей. Узгодження вимог із майбутньою архітектурою. Приклади аналізу предметної області у C++.

Тема 3. Об'єктно-орієнтоване проектування. Принципи інкапсуляції, наслідування та поліморфізму. Побудова класів та ієрархій. Об'єктна декомпозиція системи. Шаблони проектування як інструмент вирішення типових задач. Узгодження архітектури з вимогами предметної області. Приклади архітектурних рішень у C++.

Тема 4. Моделювання взаємодії об'єктів. Сценарії використання системи. Моделювання динаміки системи. Ролі та відповідальність об'єктів. Статичні та динамічні моделі. Взаємодія об'єктів у процесі виконання. Реалізація взаємодії об'єктів у C++.

Тема 5. Архітектурні рішення. Поняття архітектури програмної системи. Архітектурні стилі та шаблони. Декомпозиція системи на підсистеми та модулі. Масштабованість і гнучкість архітектури. Перевірка відповідності архітектури вимогам. Приклади архітектурних рішень у C++.

Тема 6. Інтеграція аналізу та проєктування у процес розробки. Перехід від моделі до програмної реалізації. Роль об'єктно-орієнтованої концепції у підтримці результатів аналізу та проєктування. Узгодженість між моделями та кодом. Інтеграція моделей у процес розробки. Організація командної роботи на основі об'єктних моделей. Приклади інтеграції у C++-проєктах.

Тема 7. Практичні аспекти моделювання складних систем. Моделювання інформаційних систем. Моделювання бізнес-процесів. Об'єктна декомпозиція великих систем. Приклади застосування ООА та ООПр у промислових проєктах. Перевірка та валідація моделей. Реалізація прикладів моделювання у C++.

Тема 8. Перехід від моделі до програмної реалізації. Перехід від об'єктних моделей до програмного коду. Забезпечення відповідності між моделлю та реалізацією. Застосування шаблонів проєктування у процесі програмування. Проблеми і труднощі при переході від аналізу та проєктування до реалізації. Підтримка та розвиток системи після реалізації. Приклади переходу від моделі до реалізації у C++.