



Модельно-орієнтоване проектування програмних систем

Робоча програма навчальної дисципліни (Силабус)

1. Реквізити навчальної дисципліни

Рівень вищої освіти	<i>Третій (PhD)</i>
Галузь знань	<i>12 Інформаційні технології</i>
Спеціальність	<i>121 Інженерія програмного забезпечення</i>
Освітня програма	<i>Інженерія програмного забезпечення</i>
Статус дисципліни	<i>Нормативна</i>
Форма навчання	<i>Очна (денна, вечірня), заочна</i>
Рік підготовки, семестр	<i>2 курс, осінній семестр</i>
Обсяг дисципліни	<i>4 кредити ЕКТС (120 годин)</i>
Семестровий контроль/ контрольні заходи	<i>Екзамен, модульна контрольна робота, блиц-опитування, календарний контроль</i>
Розклад занять	<i>Згідно розкладу на весняний семестр поточного навчального року (rozklad.kpi.ua)</i>
Мова викладання	<i>Українська, англійська</i>
Інформація про керівника курсу / викладачів	<i>Лектор: к.т.н., доцент Коваль Олександр Васильович, o.koval@kpi.ua, тел. 067-249-82-46 Практика: : к.т.н., доцент Коваль Олександр Васильович, o.koval@kpi.ua, тел. 067-249-82-46</i>
Розміщення курсу	<i>Google classroom, Електронний кампус, платформа дистанційного навчання «Сікорський»</i>

2. Програма навчальної дисципліни

1. Опис навчальної дисципліни, її мета, предмет вивчення та результати навчання

Вивчення дисципліни «Модельно-орієнтоване проектування програмних систем» дозволяє сформувати у здобувачів освіти компетенцій, необхідні для розв'язання складних задач професійної діяльності, пов'язаної із проектуванням програмних систем з застосуванням стандартів та підходів модельно-орієнтованого проектування, зокрема на прикладі розробки семантичної моделі предметної області програмної системи.

Предметом дисципліни «Модельно-орієнтоване проектування програмних систем» є елементи технології проектування семантичної моделі предметної області програмної системи та її використання для побудови сценаріїв діяльності користувача програмної системи.

Метою навчальної дисципліни є формування у здобувачів вищої освіти таких компетентностей:

- Здатність розробляти нові моделі та наукові методи проектування, розроблення та дослідження ефективності програмного забезпечення.

- Здатність приймати стратегічні рішення, що передбачають та формулюють майбутні напрямки розвитку модельно-орієнтованих процесів, нових бізнес-продуктів та сервісів.
- Здатність до виконання попередньої обробки даних з використанням загальних засобів добування даних; до пошуку нових корисних даних та їх взаємозв'язків.
- Здатність до розроблення нових та вдосконалення існуючих моделей, методів, засобів, процесів у сфері інженерії програмного забезпечення, які забезпечують розвиток або надають нові можливості технологіям розроблення та використання програмного забезпечення.
- Здатність критично переосмислювати наявні технології інженерії програмного забезпечення та відстежувати тенденції їх розвитку.
- Здатність забезпечувати безперервний саморозвиток і самовдосконалення, відповідальність за розвиток інших у професійній галузі, дотримуючись педагогічної етики, правил академічної доброчесності у науково-педагогічній діяльності.
- Здатність використовувати адекватні методи ефективної взаємодії з представниками різних груп (соціальних, культурних і професійних).
- Здатність працювати в команді, формувати позитивні відносини з колегами, спілкуватися з широкою науковою спільнотою та громадськістю в сфері інженерії програмного забезпечення.
- Здатність розширювати межі знань використовуючи результати оригінальних досліджень.

В результаті вивчення навчальної дисципліни здобувачі вищої освіти набудуть таких програмних результатів навчання:

- Уміти розробляти нові та вдосконалювати існуючі моделі, методи, засоби у сфері інженерії програмного забезпечення, які забезпечують розвиток технологій розроблення та використання програмного забезпечення.
- Уміти використовувати мову моделювання для візуалізації, специфікації, конструювання й документування артефактів програмних систем.
- Уміти застосовувати, розробляти та удосконалювати методи верифікації програмного забезпечення.
- Знати методи аналізу великих обсягів даних.
- Уміти розробляти та удосконалювати методи модельно-орієнтованого проектування інформаційних систем для вирішення теоретичних і прикладних задач за умови створення об'єктних, сценарних моделей.
- Вміти досліджувати робочі параметри процесів життєвого циклу програмного забезпечення, а також здійснювати аналіз вибраних методів та засобів підтримки цих процесів та бути спроможним обґрунтувати свій вибір.
- Розуміти теоретичні засади, що лежать в основі методів досліджень інформаційних систем та програмного забезпечення, методології проведення досліджень та обчислювальних експериментів.
- Глибоко розуміти загальні принципи та методи інженерії програмного забезпечення, а також методологію наукових досліджень, застосувати їх у власних дослідженнях у сфері інженерії програмного забезпечення та у викладацькій практиці.
- Мати передові концептуальні та методологічні знання з інженерії програмного забезпечення і на межі предметних галузей, а також дослідницькі навички, достатні для

проведення наукових і прикладних досліджень на рівні сучасних світових досягнень з відповідного напрямку, отримання нових знань та/або здійснення інновацій.

2. Пререквізити та постреквізити дисципліни (місце в структурно-логічній схемі навчання за відповідною освітньою програмою)

Успішному вивченню дисципліни «Модельно-орієнтоване проектування програмних систем» передують вивчення дисциплін «Формальні методи програмної інженерії» навчального плану підготовки докторів філософії за спеціальністю 121 Інженерія програмного забезпечення.

Дисципліна «Модельно-орієнтоване проектування програмних систем» передують дисципліні «Методи реінжинірингу програмного забезпечення». Отримані в результаті засвоєння дисципліни «Модельно-орієнтоване проектування програмних систем» теоретичні знання та практичні уміння можуть бути корисні для проведення наукових досліджень за темою дисертації.

3. Зміст навчальної дисципліни

Дисципліна «Модельно-орієнтоване проектування програмних систем» передбачає вивчення таких тем:

Розділ 1. Вступ до модельно-орієнтованого проектування програмних систем.

Тема 1.1. Основні підходи до моделювання програмного забезпечення.

Тема 1.2. Моделювання програмного забезпечення з використанням мови моделювання.

Розділ 2. Модельно-орієнтований підхід до побудови моделі предметної області ПС.

Тема 2.1. Побудова моделі предметної області ПС.

Тема 2.2. Метадані як елемент опису предметної області ПС.

Розділ 3. Моделювання предметної області програмної системи з застосуванням семантичного модулювання управління знаннями.

Тема 3.1. Методологія семантичного моделювання побудови системи управління знаннями.

Тема 3.2. Побудова концептуальної моделі онтології предметної області.

Тема 3.3. Моделювання відносин між концептами предметної області та їх властивостями.

Тема 3.4. Мови логічного висновку.

Тема 3.5. Оцінка якості моделі онтології.

Тема 3.6. Побудова запитів до опису онтологічної моделі Про ПС.

Тема 3.7. Методи і технології інтеграції онтологічної моделі з реляційними даними..

Розділ 4. Моделювання діяльності користувача ПС на основі онтологічної моделі Про.

Тема 4.1. Сценарно-цільовий підхід до діяльності користувача ПС та побудова сценаріїв діяльності користувача ПС.

Тема 4.2. Побудова сценаріїв діяльності користувача та інтерфейсу користувача ПС на базі онтологічної моделі Про.

4. Навчальні матеріали та ресурси

Базова література:

1. John D. Poole. Model-Driven Architecture: Vision, Standards And Emerging Technologies. – Режим доступу: URL: https://www.omg.org/mda/mda_files/Model-Driven_Architecture.pdf

2. Alberto Rodrigues da Silva. Model-driven engineering: A survey supported by the unified conceptual model. Computer Languages, Systems & Structures, Volume 43, October 2015, Pages 139-155
3. Додонов О. Г., Коваль О. В., Глоба Л. С., Бойко Ю. Д. Комп'ютерне моделювання інформаційно-аналітичних систем: монографія. Київ: ІПРІ НАН України, 2017. 239 с.
4. А.Г. Додонов, В.Р. Сенченко, А.В. Коваль Аналитика и знания в компьютерных системах. Монография. Институт проблем регистрации информации НАН Украины. Национальный технический университет Украины «Киевский политехнический институт имени Игоря Сикорского», Киев, 2020, 315 с.
5. Web Protege User Guide. URL: https://protegewiki.stanford.edu/wiki/Main_Page

Додаткова література

3. Marco Brambilla, Jordi Cabot, Manuel Wimmer. Model-Driven Software Engineering in Practice: Second Edition. Morgan & Claypool, 2017. 168 p. – Режим доступу: URL: (https://www.slideshare.net/mbrambil/modeldriven-software-engineering-in-practice-chapter-1-introduction?from_action=save)
4. Volter M. et al. Model-Driven Software Development: Technology, Engineering, Management. John Wiley & Sons, 2006
5. Alberto Rodrigues da Silva. Model-driven engineering: A survey supported by the unified conceptual model. Computer Languages, Systems & Structures, Volume 43, October 2015, Pages 139-155.
6. Петрик М.Р. Моделювання програмного забезпечення : науково-методичний посібник / М.Р. Петрик, О.Ю. Петрик – Тернопіль : Вид-во ТНТУ імені Івана Пулюя, 2015. – 200 с.
7. Scenario-Based Task Analysis Practice / Kentaro Go, John M. Carrol // Yamanashi University and Virginia Tech https://www.researchgate.net/publication/228690465_Scenario-based_task_analysis. [Електронний ресурс]. – Режим доступу: URL: www.researchgate.net/publication/228690465_Scenario-based_task_analysis.
8. Marvin Minsky. A Framework for Representing Knowledge. URL: <http://web.media.mit.edu/~minsky/papers/Frames/frames.html>
9. Novograduska R.L., Globa L.S., Koval O.V., Senchenko V.R. Ontology model of intelligent modeling system for marine facilities identification. Proceedings of International Conference on Information and Telecommunication Technologies and Radio Electronics (UkrMiCo), 2017. DOI:10.1109/UkrMiCo.2017.8095426. IEEE Digital Library, 8095426 (Scopus).
10. Novograduska R.L., Globa L.S., Koval O.V., Senchenko V.R. Ontology for Applications Development. Chapter 2 «Ontology in Information Science»; Book edited by Ciza Thomas. Print ISBN 978-953-51-3887-7. Published: March 8, 2018. P. 29–53. URL: <http://dx.doi.org/10.5772/intechopen.74042>
9. Novograduska R.L., Globa L.S., Koval O.V., Senchenko V.R. Examples of Ontology Model Usage in Engineering Fields. Chapter 3 «Ontology in Information Science»; Book edited by Ciza Thomas. Print ISBN 978-953-51-3887-7. Published: March 8, 2018. P. 54–81. URL: <http://dx.doi.org/10.5772/intechopen.74369>
10. Senchenko V.R., Koval A.V. The technology of semantic modeling for knowledge management system in environment Protege. «Информационные технологии и безопасность». Материалы XVII международной научно-практической конференции. Київ, 2017. Вип. 17. С. 211–234.
11. An introduction to the owl web ontology language. URL: <http://www.cse.lehigh.edu/~heflin/IntroToOWL.pdf>
12. Лавріщева К.М. Програмна інженерія: підручник. Київ: НАН України, 2008. 319 с.

11. Навчальний контент

5. Методика опанування навчальної дисципліни(освітнього компонента)

№ з/п	Тип навчального заняття	Опис навчального заняття
Розділ 1. Вступ до модельно-орієнтованого проектування програмних систем.		
1	Лекція 1. Основні підходи до моделювання програмного забезпечення	Огляд методів проектування програмного забезпечення. Основні підходи та концепції моделювання програмного забезпечення. Принципи моделювання. Об'єкти моделювання. Модельно-орієнтований підхід. Концепція модельно-орієнтованого підходу.
2	Лекція 2. Моделювання програмного забезпечення з використанням мови моделювання.	1. Мова моделювання артефактів програмного забезпечення. Концептуальна модель мови моделювання UML. Правила мови моделювання UML
3	Комп'ютерний практикум 1.	Завдання: Описати актуальність, об'єкт та предмет дослідження, мету та завдання та дисертаційної роботи. Провести функціональну та об'єктну декомпозицію обраної предметної області ПС, що розробляється в дисертаційній роботі. Побудувати діаграму прецедентів та визначити характеристики елементів діаграми прецедентів. Побудувати діаграми класів та об'єктів, процесів. Навести визначення статичного класу. Навести опис концептуальної схеми БД. Побудувати діаграми компонентів, розташування. Побудуйте та опишіть архітектуру ПС.
Розділ 2. Модельно-орієнтований підхід до побудови моделі предметної області ПС.		
4	Лекція 3. Побудова моделі предметної області ПС за використанням модельно-орієнтованого підходу	Використання семантичної моделі до побудови моделі предметної області (ПрО) ПС. Засоби побудови моделі ПрО. Основні напрямки моделювання ПрО. Метадані як елемент опису онтологічної моделі ПрО.
Розділ 3. Моделювання предметної області програмної системи з застосуванням семантичного модулювання управління знаннями.		
5	Лекція 4. Семантичне моделювання побудови системи управління знаннями.	Методологія семантичного моделювання предметної області. Визначення глибини і масштабу моделі предметної області. Побудова концептуальної моделі онтології предметної області.
6	Лекція 5. Моделювання відносин між концептами предметної області та їх властивостями.	Механізм зв'язування понять. Формування відносин між екземплярами класів. Формування аксіом через зв'язування класів. Графічне відображення моделі онтології

7	Лекція 6. Мови логічного висновку.	Дескриптивна логіка. Виразність мов онтології. Оцінка якості моделі онтології.
8	Комп'ютерний практикум 2.	Завдання: Побудувати онтологічну модель ПрО ПС, що розробляється в дисертаційній роботі.
9.	Лекція 7. Побудова запитів до опису онтологічної моделі ПрО ПС.	Базові положення мови запитів SPARQL. Механізми формування SPARQL-запитів. Методи і технології інтеграції онтологічної моделі з реляційними даними.
10.	Комп'ютерний практикум 3.	Завдання: Визначити та побудувати запити до опису онтологічної моделі ПрО ПС, що розробляється в дисертаційній роботі.
Розділ 4. Моделювання діяльності користувача ПС на основі онтологічної моделі ПрО.		
11	Лекція 8. Сценарно-цільовий підхід до діяльності користувача ПС та побудова сценаріїв діяльності користувача ПС.	Сценарно-цільовий підхід до діяльності користувача ПС. Сценарій діяльності користувача ПС. Об'єктна модель сценарію діяльності користувача. Модель функціонування порталу знань на основі сценарного підходу. Формування сценаріїв діяльності користувача в веб-середовищі.
12	Лекція 9. Побудова інтерфейсу користувача ПС на базі сценаріїв діяльності користувача та онтологічної моделі ПрО.	Інтерфейс користувача ПС при формуванні сценаріїв діяльності користувача. Модель користувача інтегрованої платформи побудови сценаріїв діяльності користувача. Використання онтологічної моделі ПрО ПС для побудови інтерфейсу користувача
13	Комп'ютерний практикум 4.	Завдання: Побудувати фрагмент сценарію діяльності користувача та інтерфейсу користувача ПС, що розробляється в дисертаційній роботі, на базі онтологічної моделі ПрО.
14	Тематична контрольна робота	

1. Самостійна робота студента/аспіранта

Дисципліна «Модельно-орієнтоване проектування програмних систем» ґрунтується на самостійній підготовці до аудиторних занять на теоретичні та практичні теми.

№ з/п	Назва теми, що виноситься на самостійне опрацювання	Кількість годин	Література
1	Підготовка до лекції 1	1	1, стор. 4–13; 2, стор. 61–66.
2	Підготовка до лекції 2	2	4, стор. 18–46, 107–146.
	Підготовка до лекції 3	2	2, стор. 36–53, 59–72.
4	Підготовка до лекції 4	2	3, стор. 128–145; 5.
5	Підготовка до лекції 5	2	3, стор. 146–154.
6	Підготовка до лекції 6	2	3, стор. 154–160, 177–182.
7	Підготовка до лекції 7	2	3, стор. 160–176, 183–193.
8	Самостійне опрацювання матеріалів лекції 8	6	2, стор. 12–17, 78-124;

			3, стор. 226–235.
9	Самостійне опрацювання матеріалів лекції 9	6	2, стор. 126–129. 3, стор. 215–230
10	Підготовка до комп'ютерного практикуму 1	7	
11	Підготовка до комп'ютерного практикуму 2	12	
12	Підготовка до комп'ютерного практикуму 3	12	
13	Підготовка до комп'ютерного практикуму 4	12	
	Підготовка до тематичної контрольної роботи	4	
14	Підготовка до екзамену	9	

12. Політика та контроль

2. Політика навчальної дисципліни (освітнього компонента)

- Відвідування занять з комп'ютерного практикуму може бути епізодичним та за потреби захисту робіт комп'ютерного практикуму.
- Правила поведінки на заняттях: активність, повага до присутніх, відключення телефонів.
- Дотримання політики академічної доброчесності.
- Правила захисту робіт комп'ютерного практикуму: роботи повинні бути зроблені згідно з завданням та темою дисертаційної роботи здобувача освіти, якій повинен визначитися з фрагментом програмного забезпечення (програмної системи), якій планується розробляти в рамках дисертаційної роботи та відповідно якого будуть виконуватися завдання комп'ютерного практикуму.
- Правила призначення заохочувальних та штрафних балів є наступними.

Заохочувальні бали нараховуються за:

- точні та повні відповіді під час опитувань за матеріалами лекцій. Протягом семестру на лекціях відбувається **бліц-опитування** за темами минулих лекцій. Максимальна кількість балів за блиц-опитування: $1 \text{ ба} \times 7_{\text{л.}} = 7 \text{ балів}$.

- творчий підхід у виконанні робіт комп'ютерного практикуму. Максимальна кількість балів за всі роботи: $4 \text{ бали} \times 3 \text{ комп. практи.} + 11 \text{ балів} \times 1 \text{ комп. практи.} = 23 \text{ балів}$.

Штрафні бали нараховуються за:

- плагіат(код програми не відповідає варіанту завдання, ідентичність коду програми серед різних робіт) у роботах комп'ютерного практикуму: -5 балів за кожну спробу.

3. Види контролю та рейтингова система оцінювання результатів навчання (PCO)

Протягом семестру аспіранти виконують 4 комп'ютерних практикумів. Максимальна кількість балів за кожний комп'ютерний практикум: 15 балів.

Бали нараховуються за:

- якість виконання комп'ютерного практикуму: 0-5 бали;
- відповідь під час захисту комп'ютерного практикуму: 0-5 бали;
- своєчасне представлення роботи до захисту: 0-5 бали.

Критерії оцінювання якості виконання:

5 бали – робота виконана якісно, в повному обсязі;

3 бали – робота виконана в повному обсязі, але містить незначні помилки;
1 бал – робота виконана не в повному обсязі, або містить суттєві помилки.

Критерії оцінювання відповіді:

5 бали – відповідь повна, добре аргументована;
3 бали – в цілому відповідь правильна, але має недоліки або незначні помилки;
0 балів – немає відповіді або відповідь неправильна.

Критерії оцінювання своєчасності представлення роботи до захисту:

5 бали – робота представлена до захисту не пізніше вказаного терміну;
1 бал – робота представлена до захисту пізніше вказаного терміну.

Максимальна кількість балів за виконання та захист комп'ютерних практикумів:

15 балів × 4 комп. практ. = 60 балів.

Протягом семестру на лекціях відбувається **опитування за темою поточного заняття**.

Максимальна кількість балів за опитування, яку можна отримати протягом семестру: 2 бали.

Завдання на **тематичну контрольну роботу** складається з 2 теоретичних запитань. Відповідь на кожне теоретичне запитання оцінюється 5 балами.

Критерії оцінювання кожного теоретичного запитання контрольної роботи:

5 балів – відповідь правильна, повна, добре аргументована;
4 бали – відповідь правильна, розгорнута, але не дуже добре аргументована;
3 бали – в цілому відповідь правильна, але має недоліки;
2 бали – у відповіді є незначні помилки;
1 бал – у відповіді є суттєві помилки;
0 балів – немає відповіді або відповідь неправильна.

Максимальна кількість балів за модульну контрольну роботу:

5 балів × 2 теоретичні запитання = 10 балів.

Рейтингова шкала з дисципліни дорівнює:

$R_c = R_{\text{ком.практ}} + R_{\text{опитув}} + R_{\text{тв. підхід}} + R_{\text{ТКР}} = 60 \text{ балів} + 7 \text{ бали} + 23 \text{ бали} + 10 \text{ балів} = 100 \text{ балів}$.

Календарний контроль: провадиться двічі на семестр як моніторинг поточного стану виконання вимог силабусу.

На першій атестації (8-й тиждень) студент отримує «зараховано», якщо його поточний рейтинг не менше 50 % від максимальної кількості балів, яку може отримати студент до першої атестації.

На другій атестації (14-й тиждень) студент отримує «зараховано», якщо його поточний рейтинг не менше 50 % від максимальної кількості балів, яку може отримати студент до другої атестації.

Семестровий контроль: **екзамен**.

Таблиця відповідності рейтингових балів оцінкам за університетською шкалою:

Кількість балів	Оцінка
100-95	Відмінно
94-85	Дуже добре
84-75	Добре
74-65	Задовільно
64-60	Достатньо
Менше 60	Незадовільно
Не виконані умови допуску	Не допущено

4. Додаткова інформація з дисципліни (освітнього компонента)

Перелік запитань, які виносяться на семестровий контроль, наведено у Додатку 1.

Робочу програму навчальної дисципліни (силабус):

Складено к.т.н., доц. Коваль О.В.

Ухвалено кафедрою АПЕПС (протокол № 1 від 27.08.2020 р)

Погоджено Методичною комісією (протокол № 1 від 02.09.2020 р.)