

СИЛАБУС НАВЧАЛЬНОЇ ДИСЦИПЛІНИ «МОДЕЛЮВАННЯ ТА ПРОЕКТУВАННЯ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ»



Ступінь освіти	бакалавр
Спеціальність	121 Інженерія програмного забезпечення
Освітня програма	Інженерія програмного забезпечення
Тривалість викладання	5-й семестр (9, 10 чверть)
Загальний обсяг	4 кредити ЄКТС (120 годин)
Заняття:	
лекції:	2 години/тиждень
лабораторні заняття:	2 години /тиждень
Мова викладання	українська

Сторінка курсу в СДО НТУ «ДП»: <https://do.nmu.org.ua/course/view.php?id=4534>

Кафедра, що викладає Програмного забезпечення комп'ютерних систем



Викладач:

Приходченко Сергій Дмитрович
Доцент, к.т.н.

Персональна сторінка

<https://pzks.nmu.org.ua/ua/teachers/prykhodchenkosd.php>

E-mail: Prykhodchenko.s.d@nmu.one

1. Анотація до курсу

Цей курс є важливим кроком для студентів, які бажають розвивати навички розуміння та оптимізації програмного забезпечення. Він спрямований на надання студентам глибокого розуміння та навичок у галузі моделювання та аналізу програмних систем. Курс ставить перед собою наступні головні завдання:

Розуміння основних концепцій моделювання ПЗ: Студенти отримають можливість вивчити та використовувати різні методики та інструменти для створення моделей програмного забезпечення. Вони ознайомляться з різними видами моделей, такими як структурні, функціональні, динамічні, тощо.

Аналіз та валідація програмних систем: Студенти навчаться застосовувати техніки аналізу для виявлення помилок, уразливостей та покращення якості програмного забезпечення. Вони будуть розглядати аспекти ефективності, надійності та безпеки систем.

Моделювання процесів та архітектур: Студенти досліджують архітектурні шаблони, проектування систем, інтеграцію компонентів, а також моделювання процесів розробки та взаємодії програмних систем.

Застосування інструментів моделювання: Студенти матимуть можливість використовувати сучасні інструменти для створення та аналізу моделей програмного забезпечення, такі як UML, CASE-системи та інші популярні інструменти.

Вирішення реальних проблем: Курс акцентує на практичному використанні навичок у розв'язанні реальних завдань, зокрема в розробці та оптимізації програмного забезпечення для різних областей застосування.

Цей курс сприяє підготовці студентів до роботи в галузі розробки програмного забезпечення, де розуміння моделювання та аналізу програмних систем є ключовими навичками.

2. Мета та завдання курсу

Мета дисципліни – ознайомити студентів з основними підходами до методів та засобів моделювання та аналізу програмних систем; сформулювати принципи побудови та функціонування процесу моделювання та аналізу програмного забезпечення; сформулювати компетентності та надати сучасні практики щодо автоматизованого моделювання додатків.

Завданнями дисципліни є:

- опанування теоретико-понятійної бази курсу;
- ознайомлення здобувачів з основними підходами до методів та засобів моделювання та аналізу програмних систем, принципами побудови та функціонування процесу моделювання та аналізу програмного забезпечення;
- ознайомлення здобувачів з сучасними практиками та інструментальними засобами щодо автоматизованого моделювання та аналізу програмного забезпечення.

3. Результати навчання

Дисциплінарні результати навчання:

- застосовувати різні методики та інструменти для створення моделей програмного забезпечення такими як структурні, функціональні, динамічні, тощо;
- застосовувати техніки аналізу для виявлення помилок, уразливостей та покращення якості програмного забезпечення;
- застосовувати на практиці ефективні підходи щодо проектування програмного забезпечення, а також моделювання процесів розробки та взаємодії програмних систем;
- вибирати та використовувати відповідну задачі методологію створення програмного забезпечення.

Дисциплінарні результати навчання сформовано на основі ПРН освітньо-професійної програми «Інженерія програмного забезпечення» першого (бакалаврського) рівня вищої освіти (ПР3, ПР11, ПР12, ПР16).

4. Структура курсу

Види та тематика навчальних занять	Внесок в загальну оцінку, %
ЛЕКЦІЇ	40
1. Моделювання та Аналіз ПЗ Введення. Мета навчання. Моделювання та абстракція. Висновки.	
2. Що таке система Розуміння, як ми дивимося на системи. Розуміння, що система побудована з будівельних блоків, які несуть відповідальність. Введення об'єктів і орієнтація об'єкта.	
3. Проблемна область (аналіз) Аналіз домену. Контекстні діаграми. Зацікавлені сторони. Варіанти використання.	
4. Аналіз доменів, відповідальність [Моделювання поведінки] Розуміння, що таке аналіз домену. Розуміння, що таке діаграми послідовностей. CRC-карти і як їх можна використовувати в аналізі та проектуванні. Виконання аналізу домену для випадку, описаного природною мовою. Створення діаграм послідовностей	
5. Діаграми класів; Діаграми послідовностей; Діаграми активностей Документування дизайну програмного забезпечення. Діаграма активності. Діаграма класів. Діаграма послідовності.	
6. Підсумок фази аналізу, CRC вправи Зв'язки в діаграмах класів. Діаграми послідовностей. Діаграма активності (токени). Приклад: від опису дії до моделі домену, діаграми прецеденту та прецедентів. Вправа з CRC картою	
<i>Тестова контрольна робота №1 (за темами 1-6).</i>	20
7. Лекція 7 – Поведінка (Стан машини) Моделювання поведінки. Діаграми автоматів станів. Поведінкові. Протоколи	
8. Перехід від аналізу до дизайну Підсумок за призначенням моделювання. Що таке проектування програмного забезпечення? Перехід від аналізу до проектування. Принципи проектування.	
9. Шаблони Шаблони проектування. Шаблон стану. Singleton. Challenge. шаблон «Спостерігач». Фасад. Адаптер [структурний]. Абстрактна фабрика. Декоратор. Багаторівнева архітектура. MVC. Клієнт – сервер.	
10. Приховування інформації та повторення. Принципи проектування. Приховування інформації. Розширення. Агрегація і склад. Підсумок	
<i>Тестова контрольна робота №2 (за темами 7-10).</i>	20
ЛАБОРАТОРНІ ЗАНЯТТЯ	60
Лабораторна робота 1 Завдання 1 – Моделювання процесів інформаційної системи, використовуючи відповідний підхід	
<i>Звіт з роботи № 1 та захист лабораторної роботи.</i>	20
Лабораторна робота 2 Завдання 2 – Аналіз домену з діаграмою класів і аналіз вимог з діаграмами прецедентів	
<i>Звіт з роботи № 2 та захист лабораторної роботи.</i>	20
Лабораторна робота 3 Завдання 3 – Поведінка станів агрегату.	

Види та тематика навчальних занять	Внесок в загальну оцінку, %
<i>Звіт з роботи № 3 та захист лабораторної роботи.</i>	20
РАЗОМ	100

5. Технічне обладнання та/або програмне забезпечення

Використовуються лабораторії кафедри програмного забезпечення комп'ютерних систем (комп'ютерне та мультимедійне обладнання). Дистанційна платформа Moodle, MS Office 365, Microsoft Teams, GIT, Modeio Open Source v5.

6. Система оцінювання та вимоги

6.1. Навчальні досягнення здобувачів вищої освіти за результатами вивчення курсу оцінюватимуться за шкалою, що наведена нижче:

Рейтингова шкала	Інституційна шкала
90 – 100	відмінно
74 – 89	добре
60 – 73	задовільно
0 – 59	незадовільно

6.2. Здобувачі вищої освіти можуть отримати **підсумкову оцінку** з навчальної дисципліни **на підставі поточного оцінювання знань** за умови, якщо набрана кількість балів з поточного тестування та виконання і захисту лабораторних робіт складатиме не менше 60 балів.

Теоретична частина оцінюється за результатами здачі двох контрольних тестових робіт, кожна з яких містить тестові закриті запитання з однією вірною відповіддю (максимальна кількість – 20 балів за кожною тестовою роботою). Загалом за дві контрольні тестові роботи отримується **максимум 40 балів**, тобто 40% від оцінки за дисципліну.

Лабораторні роботи (три роботи – у вигляді індивідуального завдання з кожної, розподіл % див. в таблиці розділу 4) виконуються у письмовому вигляді (звіт з кожної роботи оцінюється в межах балів, представлених в таблиці розділу 4, загалом лабораторні враховуються як 60% (максимум 60 балів). При несвоєчасному здаванні роботи оцінка знижується вдвічі. Лабораторні роботи захищаються у вигляді опитування за звітом, і захист враховується, як 50% від оцінки за роботу. У сумі за лабораторну частину курсу при поточному оцінюванні отримується **максимум 60 балів**.

Отримані бали за теоретичну частину та лабораторні роботи додаються і є підсумковою оцінкою за вивчення навчальної дисципліни. Максимально за поточною успішністю здобувач вищої освіти може набрати 100 балів.

Максимальне оцінювання поточного контролю в балах:

Теоретична частина	Практична частина	Разом
40	60	100

6.3. Критерії оцінювання підсумкової роботи. У випадку якщо здобувач вищої освіти за поточною успішністю отримав менше 60 балів та/або прагне поліпшити оцінку проводиться **підсумкове оцінювання**.

Екзамен проводиться у вигляді комплексної контрольної роботи, яка включає запитання з теоретичної та практичної частини курсу. Білет складається з **30 тестових завдань** з чотирма варіантами відповідей, одна правильна відповідь оцінюється в 2 бали (**разом 60 балів**) та **2 завдань** з практичної частини, кожне з запитань оцінюється максимум у 20 балів (**разом 40 балів**).

Отримані бали за тестові завдання та завдання з практичної частини додаються і є підсумковою оцінкою за вивчення навчальної дисципліни. Максимально за підсумковою роботою здобувач вищої освіти може набрати 100 балів.

7. Політика курсу

7.1. Політика щодо академічної доброчесності. Академічна доброчесність здобувачів вищої освіти є важливою умовою для опанування результатами навчання за дисципліною і отримання задовільної оцінки з поточного та підсумкового контролів. Академічна доброчесність базується на засудженні практик списування (виконання письмових робіт із залученням зовнішніх джерел інформації, крім дозволених для використання), плагіату (відтворення опублікованих текстів інших авторів без зазначення авторства), фабрикації (вигадування даних чи фактів, що використовуються в освітньому процесі). Політика щодо академічної доброчесності регламентується положенням "Положення про систему запобігання та виявлення плагіату у Національному технічному університеті "Дніпровська політехніка".

У разі порушення здобувачем вищої освіти академічної доброчесності (списування, плагіат, фабрикація), робота оцінюється незадовільно та має бути виконана повторно. При цьому викладач залишає за собою право змінити тему завдання.

7.2. Комунікаційна політика. Здобувачі вищої освіти повинні мати активовану корпоративну університетську пошту.

Усі письмові запитання до викладачів стосовно курсу мають надсилатися на університетську електронну пошту.

7.3. Політика щодо перескладання. Роботи, які здаються із порушенням термінів без поважних причин оцінюються на нижчу оцінку. Перескладання підсумкового оцінювання відбувається із дозволу деканату за наявності поважних причин (наприклад, лікарняний).

7.4. Політика щодо оскарження оцінювання. Якщо здобувач вищої освіти не згоден з оцінюванням його знань він може опротестувати виставлену викладачем оцінку у встановленому порядку.

7.5. Відвідування занять. Для здобувачів вищої освіти денної форми відвідування занять є обов'язковим. Поважними причинами для неявки на заняття є хвороба, участь в університетських заходах, академічна мобільність, які необхідно підтверджувати документами. Про відсутність на занятті та причини відсутності здобувач вищої освіти має повідомити викладача або особисто, або через старосту.

8. Рекомендовані джерела інформації

1. Петрик М.Р., Петрик О.Ю. Моделювання програмного забезпечення: науково-методичний посібник. Тернопіль: Вид-во ТНТУ імені Івана Палія, 2015. 200 с.

2. Лимаренко Ю. О. Моделювання та аналіз програмного забезпечення: навч.-метод. посібник / Ю. О. Лимаренко – Запоріжжя: ЗДІА, 2014. – 88 с.
3. Табунщик Г. В. Проектування та моделювання програмного забезпечення сучасних інформаційних систем: навчальний посібник/ Г. В. Табунщик, Т.І. Каплієнко, О.А. Петрова – Запоріжжя: Дике Поле, 2016. – 250 с.
4. Грицюк Ю. І. Аналіз вимог до програмного забезпечення: навчальний посібник. / Ю. І. Грицюк – Львів: Видавництво Львівської політехніки, 2018. –456 с.
5. Дудзяний І.М. Об'єктно-орієнтоване моделювання програмних систем: навчальний посібник / І.М. Дудзяний. – Львів, Видавничий центр ЛНУ ім. І. Франка, 2007. – 108 с.
6. ISO/IEC TR 19759:2015 Software Engineering – Guide to the software engineering body of knowledge (SWEBOK).