

Інженерія вимог до програмного забезпечення

СИЛАБУС

Шифр і назва спеціальності	121 – Інженерія програмного забезпечення	Інститут / факультет	Комп'ютерних наук і програмної інженерії
Назва програми	«Інженерія програмного забезпечення»	Кафедра	Програмної інженерії та інформаційних технологій управління
Тип програми	Освітньо-професійна	Мова навчання	українська

Викладач

Москаленко Валентина Володимирівна

Valentyna.Moskalenko@khpі.edu.ua



Д.т.н., доцент, професор кафедри ПІІТУ. Кількість наукових та навчальних публікацій – більше 90.
(<https://publons.com/researcher/1588564/valentyna-moskalenko/>; Web of Science ResearcherID R-9960-2018;
<https://scholar.google.com.ua/citations?user=eUidJHIAAAAJ&hl=ru>; <https://www.scopus.com/authid/detail.uri?authorId=36021571200>;
<https://orcid.org/0000-0002-9994-5404>).

Курси, що викладає: «Теорія ймовірності та математична статистика», «Основи комп'ютерних наук та методів штучного інтелекту», «Основи інформаційних систем та технологій», «Інженерія вимог до програмного забезпечення», «Основи бізнес-аналізу», «Аналітика бізнес-систем», «Методи обчислювального інтелекту», «Методи обчислювального інтелекту та інтелектуальний аналіз», «Machine Learning»

Загальна інформація про курс

Анотація	<p>Дисципліна «Інженерія вимог до програмного забезпечення» є навчальною дисципліною з циклу фахової обов'язкової підготовки за спеціальністю 121 “ Інженерія програмного забезпечення ”. Вона викладається у четвертому семестрі в обсязі 90 год.(3 кредиту ECTS), зокрема: лекції – 32год., лабораторні – 16 год., самостійна робота – 42 год. У курсі передбачено два змістових модулі та дві модульні контрольні роботи. Завершується дисципліна заліком.</p> <p>Предметом вивчення навчальної дисципліни є основні аспекти управління вимогами до програмного забезпечення (ПЗ) на протязі всього циклу розробки ПЗ. Завданням курсу є отримання знань щодо видів користувацьких вимог, особливостей їх формування та прийоми формулювання вимог бізнес-аналітиками, методів аналізу і атестації вимог для різних типів програмних систем, отримання навичок розв'язання конфліктів між вимогами різних видів, визначення пріоритетності вимог та їх документування; отримання знань щодо прийомів управління вимогами до програмного забезпечення; формування вмінь застосування інструментальних засобів для розробки користувацьких вимог; отримання знань щодо можливих ризиків, пов'язаних з вимогами.</p>
Цілі курсу	Мета вивчення дисципліни є теоретична та практична підготовка студентів у галузі інженерії вимог до програмного забезпечення, отримання студентами практичних навичок аналізу та моделювання проблемної області, розробки специфікацій вимог, а також навичок управління вимогами у продовж всього життєвого циклу ПЗ
Формат	Лекції, лабораторні роботи, консультації. Підсумковий контроль – залік.
Семестр	4

Обсяг (кредити) / Тип курсу (обов'язковий / вибіркового)	3/ Обов'язковий	Лекції (години)	32	Лабораторні заняття (години)	16	Самостійна робота (години)	42
Програмні компетентності	<p>K05. Здатність вчитися і оволодівати сучасними знаннями.</p> <p>K06. Здатність до пошуку, оброблення та аналізу інформації з різних джерел.</p> <p>K13. Здатність ідентифікувати, класифікувати та формулювати вимоги до програмного забезпечення.</p> <p>K14. Здатність брати участь у проектуванні програмного забезпечення, включаючи проведення моделювання (формальний опис) його структури, поведінки та процесів функціонування.</p> <p>K16. Здатність формулювати та забезпечувати вимоги щодо якості програмного забезпечення у відповідності з вимогами замовника, технічним завданням та стандартами.</p> <p>K17. Здатність дотримуватися специфікацій, стандартів, правил і рекомендацій в професійній галузі при реалізації процесів життєвого циклу.</p> <p>K18. Здатність аналізувати, вибирати і застосовувати методи і засоби для забезпечення інформаційної безпеки (в тому числі кібербезпеки).</p> <p>K23. Здатність реалізовувати фази та ітерації життєвого циклу програмних систем та інформаційних технологій на основі відповідних моделей і підходів розробки програмного забезпечення.</p> <p>K24. Здатність здійснювати процес інтеграції системи, застосовувати стандарти і процедури управління змінами для підтримки цілісності, загальної функціональності і надійності програмного забезпечення.</p> <p>K25. Здатність обґрунтовано обирати та освоювати інструментарій з розробки та супроводження програмного забезпечення.</p> <p>K26. Здатність до алгоритмічного та логічного мислення.</p>						
	<p>Результати навчання</p> <p>ПРО1. Аналізувати, цілеспрямовано шукати і вибирати необхідні для вирішення професійних завдань інформаційно-довідникові ресурси і знання з урахуванням сучасних досягнень науки і техніки.</p>	<p>Методи викладання та навчання</p> <p>Інтерактивні лекції з презентаціями, дискусії, практичні заняття, командна робота, кейс-метод, метод зворотного зв'язку з боку студентів, проблемне навчання</p>	<p>Форми оцінювання (поточне оцінювання CAS, підсумкове оцінювання FAS)</p> <p>Письмові індивідуальні завдання до лабораторних робіт (CAS), оцінювання знань на лабораторних заняттях (CAS), експрес-опитування(CAS), онлайн -тести (CAS), підсумковий/семестровий контроль у формі семестрового заліку, відповідно до графіку навчального процесу (FAS)</p>				
<p>ПРО3. Знати основні процеси, фази та ітерації життєвого циклу програмного забезпечення.</p>	<p>Інтерактивні лекції з презентаціями, дискусії, практичні заняття, командна робота, кейс-метод, метод зворотного зв'язку з боку студентів, проблемне навчання</p>	<p>Письмові індивідуальні завдання до лабораторних робіт (CAS), оцінювання знань на лабораторних заняттях (CAS), експрес-опитування(CAS), онлайн -тести (CAS), підсумковий/семестровий контроль у формі семестрового заліку, відповідно до графіку навчального процесу (FAS)</p>					
<p>ПРО9. Знати та вміти використовувати методи та засоби збору, формулювання та аналізу вимог до програмного забезпечення.</p>	<p>Інтерактивні лекції з презентаціями, дискусії, практичні заняття, командна робота, кейс-метод, метод зворотного зв'язку з боку студентів, проблемне навчання</p>	<p>Письмові індивідуальні завдання до лабораторних робіт (CAS), оцінювання знань на лабораторних заняттях (CAS), експрес-опитування(CAS), онлайн -тести (CAS), підсумковий/семестровий контроль у формі семестрового заліку, відповідно до графіку навчального процесу (FAS)</p>					

<p>ПР10. Проводити передпроектне обстеження предметної області, системний аналіз об'єкта проектування.</p>	<p>Інтерактивні лекції з презентаціями, дискусії, практичні заняття, командна робота, кейс-метод, метод зворотного зв'язку з боку студентів, проблемне навчання</p>	<p>Письмові індивідуальні завдання до лабораторних робіт (CAS), оцінювання знань на лабораторних заняттях (CAS), експрес-опитування(CAS), онлайн -тести (CAS), підсумковий/семестровий контроль у формі семестрового заліку, відповідно до графіку навчального процесу (FAS)</p>
<p>ПР11. Вибирати вихідні дані для проектування, керуючись формальними методами опису вимог та моделювання.</p>	<p>Інтерактивні лекції з презентаціями, дискусії, практичні заняття, командна робота, кейс-метод, метод зворотного зв'язку з боку студентів, проблемне навчання</p>	<p>Письмові індивідуальні завдання до лабораторних робіт (CAS), оцінювання знань на лабораторних заняттях (CAS), експрес-опитування(CAS), онлайн -тести (CAS), підсумковий/семестровий контроль у формі семестрового заліку, відповідно до графіку навчального процесу (FAS)</p>
<p>ПР14. Застосовувати на практиці інструментальні програмні засоби доменного аналізу, проектування, тестування, візуалізації, вимірювань та документування програмного забезпечення.</p>	<p>Інтерактивні лекції з презентаціями, дискусії, практичні заняття, командна робота, кейс-метод, метод зворотного зв'язку з боку студентів, проблемне навчання</p>	<p>Письмові індивідуальні завдання до лабораторних робіт (CAS), оцінювання знань на лабораторних заняттях (CAS), експрес-опитування(CAS), онлайн -тести (CAS), підсумковий/семестровий контроль у формі семестрового заліку, відповідно до графіку навчального процесу (FAS)</p>
<p>ПР19. Знати та вміти застосовувати методи верифікації та валідації програмного забезпечення.</p>	<p>Інтерактивні лекції з презентаціями, дискусії, практичні заняття, командна робота, кейс-метод, метод зворотного зв'язку з боку студентів, проблемне навчання</p>	<p>Письмові індивідуальні завдання до лабораторних робіт (CAS), оцінювання знань на лабораторних заняттях (CAS), експрес-опитування(CAS), онлайн -тести (CAS), підсумковий/семестровий контроль у формі семестрового заліку, відповідно до графіку навчального процесу (FAS)</p>
<p>ПР23. Вміти документувати та презентувати результати розробки програмного забезпечення.</p>	<p>Інтерактивні лекції з презентаціями, дискусії, практичні заняття, командна робота, кейс-метод, метод зворотного зв'язку з боку студентів, проблемне навчання</p>	<p>Письмові індивідуальні завдання до лабораторних робіт (CAS), оцінювання знань на лабораторних заняттях (CAS), експрес-опитування(CAS), онлайн -тести (CAS), підсумковий/семестровий контроль у формі семестрового заліку, відповідно до графіку навчального процесу (FAS)</p>

СИСТЕМА ОЦІНЮВАННЯ

Розподіл балів для оцінювання успішності аспіранта	Сума балів за всі види навчальної діяльності	Оцінка ECTS	Оцінка за національною шкалою	Нарахування балів
	90-100	A	відмінно	
	82-89	B	добре	
	74-81	C		
	64-73	D	задовільно	
	60-63	E		
	35-59	FX		
	0-34	F	незадовільно з обов'язковим повторним вивченням дисципліни	

100% підсумкове оцінювання у вигляді заліку (10%) та поточного оцінювання (90%).
10% семестровий залік, відповідно до графіку навчального процесу;
90% поточне оцінювання:

- 80% оцінювання виконання завдань на лабораторних заняттях;
- 10% проміжний контроль (2 модульні контрольні роботи)

Політика курсу Дотримуватися правил внутрішнього розпорядку університету. Брати активну участь у навчальному процесі. Студенти зобов'язані відвідувати заняття згідно розкладу та дотримуватися етики поведінки. У разі відсутності студентам необхідно буде виконати всі завдання, щоб компенсувати пропущені заняття. Виконання лабораторних робіт вимагає попередньої підготовки та завчасного опрацювання всіх необхідних матеріалів. Письмові завдання повинні бути подані до встановлених строків.

Структура та зміст курсу

Лекція 1	Основні положення інженерії вимог до програмного забезпечення			Самостійна робота	Галузі знань за SWEBOK
Лекція 2	Визначення вимог як етап розробки програмного забезпечення				Життєвий цикл програмного забезпечення
Лекція 3	Класифікація вимог. Рівні вимог за К. Вігерсом; класифікація в рамках концепції SWEBOK; класифікація RUP (FURPS +) та ін.				
Лекція 4	Процес формування вимог				
Лекція 5	Проблеми формування вимог та підходи щодо їх вирішення				
Лекція 6	Традиційні методи виявлення вимог та технології виявлення вимог	Лабораторна робота 1	Виявлення бізнес-вимог до ПЗ (Формування Business Requirements Specification) (Innovation Campus: ПЗ25)		Мозковий штурм та інші методики генерації ідей
Лекція 7	Сучасні методи виявлення вимог				Види прототипування ПЗ
Лекція 8	Процес аналізу вимог	Лабораторна робота 2	Формування функціональних та нефункціональних вимог до ПЗ (Innovation Campus: ПЗ12)		
Лекція 9	Розширений аналіз вимог.				Бізнес-процесний підхід управління підприємством
Лекція 10	Використання методів моделювання для аналізу вимог.	Лабораторна робота 3	Формування Software Requirements Specification (Innovation Campus: ПЗ12)		Розробка діаграм UML
Лекція 11					Методологія моделювання SADT та стандарти IDEF
Лекція 12	Управління вимогами до ПЗ	Лабораторна робота 4	Управління змінами вимог до ПЗ		
Лекція 13	Управління змінами вимог до ПЗ				
Лекція 14	Атрибути якості ПЗ				
Лекція 15	Тестування вимог ПЗ				
Лекція 16	Особливості управління вимогами за різними моделями розробки ПЗ				Особливості моделей розробки ПЗ

Література

Обов'язкова

- 1 A Guide to the Business Analysis Body of Knowledge® (BABOK® Guide). International Institute of Business Analysis, Toronto, Ontario, Canada. Version 3.0 published 2015.
- 2 Guide to Software Engineering Body of Knowledge (SWEBOK). IEEE Computer Society, 2004
- 3 Wiegers, K.E.(2018) Software Requirements, 3rd ed., Microsoft Press,
- 4 Грицюк Ю. І. Аналіз вимог до програмного забезпечення. Навчальний посібник.
- 5 Klaus Pohl Requirements Engineering Fundamentals, 2nd Edition: A Study Guide for the Certified Professional for Requirements Engineering Exam (2016) - Foundation Level IREB compliant. Rocky Nook, Inc.
- 6 Berenbach, B., Paulish, D., Katzmeier, J., & Rudorfer, A. (2009). Software & Systems Requirements Engineering: In Practice. New York: McGraw-Hill Professional.
- 7 Лавріщева К.М. Програмна інженерія.(2008) Київ
- 8 IEEE Recommended Practice for Software Requirements Specifications. IEEE Std 830-1998.
- 9 Dick, J., Hull, E., & Jackson, K. (2017). Requirements Engineering. (4rd ed.) Springer.Retrieved from: <https://doi.org/10.1007/978-3-319-61073-3>
- 10 Авраменко А.С., Авраменко В.С., Косенюк Г.В. (2017) Тестування програмного забезпечення. Черкаси: ЧНУ імені Богдана Хмельницького.

Додаткова

1. Klaus Pohl (2010) Requirements Engineering: Fundamentals, Principles, and Techniques. pringer Publishing Company, Incorporated,
2. Leffingwell D., Widrig D. (2004) Managing software requirements: a unified approach Addison-Wesley Longman Publishing Co., Inc. Boston, MA, USA ©20001.
3. Cohn Mike. User Stories Applied. For Agile Software Development . Addison-Wesley; Pearson Education,
4. Rumpe, Bernhard (2016) Modeling with UML. Language, Concepts, Methods Springer. Retrieved from:https://nibmehub.com/opac-service/pdf/read/Modeling%20with%20UML_%20Language-%20Concepts-%20Methods.pdf
5. Сценарій управління змінами// Retrieved from <https://oberemokii.com/uk/additional-scenarios/managing-the-content-and-timing/the-scenario-of-change-management>
6. Навчальний ресурс з тестування програмного забезпечення. Особливості вимог програмного забезпечення. Методи тестування. Фази тестування. Класи еквівалентності. Retrieved from: <https://qlearning.com.ua/theory/lectures/material/requirements-testing-methods-equivalence/>
7. Karl Weigers and Joy Beatty.(2013) Software Requirements (3rd Edition): Microsoft Press,
8. Mall R. (2014) Fundamentals Of Software Engineering, 4Th Ed., Phi,

Норми академічної етики

Студент повинен дотримуватися «Кодексу етики академічних взаємовідносин та доброчесності НТУ «ХПІ»: виявляти дисциплінованість, вихованість, доброзичливість, чесність, відповідальність

Силабус за змістом повністю відповідає робочій програмі курсу.