

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
Львівський національний університет імені Івана Франка
Факультет прикладної математики та інформатики
Кафедра теорії оптимальних процесів

Затверджено

На засіданні кафедри теорії оптимальних процесів факультету прикладної математики та інформатики Львівського національного університету імені Івана Франка

Протокол № 1 від 28.08.2024 р.)



Завідувач кафедри Шахно С.М.

Силабус з навчальної дисципліни

“Системний підхід до розробки програмного забезпечення”,
що викладається в межах ОПП Системний аналіз
другого (магістерського) рівня вищої освіти для здобувачів з
спеціальності 124 – системний аналіз

Львів 2024 р.

Назва дисципліни	Системний підхід до розробки програмного забезпечення
Адреса викладання дисципліни	Головний корпус ЛНУ ім. І. Франка м. Львів, вул. Університетська 1
Факультет та кафедра, за якою закріплена дисципліна	Факультет прикладної математики та інформатики Кафедра теорії оптимальних процесів
Галузь знань, шифр та назва спеціальності	12 – інформаційні технології 124 – системний аналіз
Викладачі дисципліни	Чипурко Андрій Іванович, доцент кафедри теорії оптимальних процесів
Контактна інформація викладачів	andrii.chypurko@lnu.edu.ua ; https://ami.lnu.edu.ua/employee/chypurko-andriy-ivanovych ; Головний корпус ЛНУ ім. І. Франка, ауд. 269. м. Львів, вул. Університетська, 1
Консультації з питань навчання по дисципліні відбуваються	Консультації в день проведення лекцій/лабораторних занять (за попередньою домовленістю).
Сторінка курсу	https://ami.lnu.edu.ua/course/dv3-top-systemnyy-pidkhid-do-rozrobky-prohramnoho-zabezpechennia
Інформація про дисципліну	У курсі розглядаються сучасні SDLC підходи до розробки програмного забезпечення
Коротка анотація дисципліни	Дисципліна “ Системний підхід до розробки програмного забезпечення” є нормативною дисципліною зі спеціальності 124 – системний аналіз для магістерської освітньої програми, яка викладається в 1-му семестрі в обсязі 4-ох кредитів (за Європейською Кредитно-Трансферною Системою ECTS).
Мета та цілі дисципліни	<p>Мета дисципліни:</p> <p>Ознайомити студентів із системним підходом до розробки програмного забезпечення, розглядаючи сучасні підходи життєвого циклу розробки програмного забезпечення (SDLC). Забезпечити засвоєння необхідних знань і навичок для ефективної розробки, тестування та впровадження програмних продуктів у різних середовищах.</p> <p>Цілі дисципліни:</p> <ol style="list-style-type: none"> 1. Надати студентам теоретичні знання про системний підхід до розробки програмного забезпечення, включаючи основні концепції та термінологію SDLC. 2. Ознайомити студентів з різними підходами до SDLC, такими як Waterfall, Agile, Scrum, Kanban, та їх перевагами і недоліками. 3. Вивчити етапи SDLC, включаючи вимоги, проектування, розробку, тестування, впровадження та обслуговування програмного забезпечення.

	<p>4. Навчити студентів використовувати сучасні інструменти та методи для управління проектами розробки програмного забезпечення.</p> <p>5. Сприяти розвитку навичок командної роботи, комунікації, критичного мислення та прийняття рішень, необхідних для успішної розробки програмного забезпечення.</p> <p>6. Ознайомити студентів з методами забезпечення якості програмного забезпечення, включаючи тестування, верифікацію та валідацію.</p> <p>7. Підготувати студентів до професійної кар'єри в галузі розробки програмного забезпечення та забезпечити їх необхідними навичками для адаптації до змін у технологіях та методах розробки.</p>
<p>Література для вивчення дисципліни</p>	<ol style="list-style-type: none"> 1. https://www.tutorialspoint.com/sdlc/sdlc_overview.htm 2. https://phoenixnap.com/blog/software-development-life-cycle 3. https://softserve.udemy.com/course/sdlc-software-development-life-cycle-lifecycle-process/?src=sac&kw=sdl 4. https://softserve.udemy.com/course/sdlc-phases/learn/lecture/14385348#overview 5. https://www.pmi.org/learning/library/agile-project-management-scrum-6269 6. https://www.scaledagileframework.com/
<p>Обсяг курсу</p>	<p>Загальний обсяг: 120 годин. Аудиторних занять: 48 год., з них 32 год. лекції, 16 год. лабораторних занять. Самостійної роботи: 72 год.</p>
<p>Очікувані результати навчання</p>	<p>Після завершення цього курсу студент буде :</p> <p>Знати:</p> <ol style="list-style-type: none"> 1. Основні концепції та термінологію системного підходу до розробки програмного забезпечення. 2. Різні підходи до життєвого циклу розробки програмного забезпечення (SDLC), такі як Waterfall, Agile, Scrum, Kanban, та їхні переваги і недоліки. 3. Етапи SDLC, включаючи збір вимог, проектування, розробку, тестування, впровадження та обслуговування програмного забезпечення. 4. Методи та інструменти управління проектами розробки програмного забезпечення. 5. Методики забезпечення якості програмного забезпечення, включаючи тестування, верифікацію та валідацію. 6. Тенденції та сучасні практики в галузі розробки програмного забезпечення. <p>Вміти:</p> <ol style="list-style-type: none"> 1. Аналізувати вимоги до програмного забезпечення та розробляти відповідну документацію. 2. Використовувати сучасні інструменти та методи для планування, розробки, тестування та впровадження програмного забезпечення. 3. Застосовувати різні підходи до SDLC залежно від специфіки проекту та вимог замовника. 4. Ефективно працювати в команді, комунікувати з замовниками та іншими зацікавленими сторонами проекту. 5. Забезпечувати якість програмного забезпечення, проводити тестування, верифікацію та валідацію. 6. Управляти проектами розробки програмного забезпечення, використовуючи методи та інструменти управління проектами.

	<p>7. Впроваджувати та підтримувати програмні продукти, здійснювати їх модернізацію та оновлення.</p> <p>8. Адаптуватися до змін у технологіях та методах розробки, застосовувати нові підходи та інструменти в професійній діяльності.</p>
Ключові слова	<p>SDLC (Software Development Life Cycle), визначення проекту, Waterfall, Agile, Scrum, Kanban, проектний менеджмент, планування спринтів, оцінка часу, Scaled Agile, метрики продуктивності, метрики якості, інструменти SDLC, вибір методології, Story Points, Planning Poker, Velocity, Burndown Charts, SAFe (Scaled Agile Framework), LeSS (Large Scale Scrum), DAD (Disciplined Agile Delivery), Cycle Time, Lead Time, Throughput, Defect Density, Test Coverage, Customer Satisfaction, Mean Time to Repair (MTTR), JIRA, Trello, GitHub, Jenkins, Docker, Selenium, верифікація, валідація, командна робота, критичне мислення, тестування програмного забезпечення, життєвий цикл розробки.</p>
Формат курсу	<p>Очний, дистанційний .</p> <p>Проведення практичних занять, заслуховування реальних наукових доповідей, виголошених на наукових конференціях, презентації магістерських робіт і консультацій.</p>
Теми	<p>Тема 1. Основи SDLC, визначення</p> <ol style="list-style-type: none"> 1. Збір вимог 2. Аналіз вимог 3. Проектування системи 4. Реалізація (кодинг) 5. Тестування 6. Впровадження 7. Обслуговування та підтримка <p>Тема 2. SDLC методології</p> <ol style="list-style-type: none"> 1. Waterfall 2. Agile (Scrum, Kanban) 3. V-Model 4. Spiral Model 5. Incremental Model 6. RAD (Rapid Application Development) <p>Тема 3. Waterfall</p> <ol style="list-style-type: none"> 1. Визначення вимог 2. Проектування системи 3. Реалізація (кодинг) 4. Тестування 5. Впровадження 6. Обслуговування <p>Тема 4. Agile Scrum</p> <ol style="list-style-type: none"> 1. Scrum команди 2. Ролі (Product Owner, Scrum Master, Development Team) 3. Артефакти (Product Backlog, Sprint Backlog, Increment) 4. Події (Sprint Planning, Daily Scrum, Sprint Review, Sprint Retrospective) <p>Тема 5. Agile Kanban</p> <ol style="list-style-type: none"> 1. Візуалізація роботи 2. Обмеження незавершеної роботи 3. Управління потоком 4. Постійні покращення 5. Тема 6. Проектний менеджмент з Agile Scrum 6. Планування спринтів

	<ul style="list-style-type: none"> 7. Визначення ролей та відповідальностей 8. Виконання спринтів 9. Відстеження прогресу 10. Ретроспективи для аналізу та покращення <p>Тема 7. Планування і оцінка часу в Agile</p> <ul style="list-style-type: none"> 1. Story Points 2. Planning Poker 3. Velocity 4. Burndown Charts <p>Тема 8. Scaled Agile</p> <ul style="list-style-type: none"> 1. SAFe (Scaled Agile Framework) 2. LeSS (Large Scale Scrum) 3. DAD (Disciplined Agile Delivery) <p>Тема 9. Метрики продуктивності</p> <ul style="list-style-type: none"> 1. Velocity 2. Cycle Time 3. Lead Time 4. Throughput <p>Тема 10. Метрики якості</p> <ul style="list-style-type: none"> 1. Defect Density 2. Test Coverage 3. Customer Satisfaction 4. Mean Time to Repair (MTTR) <p>Тема 11. Інструменти SDLC</p> <ul style="list-style-type: none"> 1. JIRA 2. Trello 3. GitHub 4. Jenkins 5. Docker 6. Selenium <p>Тема 12. Вибір SDLC методології</p> <ul style="list-style-type: none"> 1. Тип проекту 2. Розмір команди 3. Строки виконання 4. Бюджет 5. Технічні вимоги
Підсумковий контроль, форма	Комбінований залік у кінці семестру.
Пререквізити	Для вивчення курсу студенти потребують базових знань з управління проектами, інформаційних технологій та економіки.
Навчальні методи та техніки, які будуть використовуватися під час викладання курсу	Лекції, презентації, лабораторні заняття. Індивідуальні завдання.
Необхідне обладнання	Комп'ютер із стандартним програмним забезпеченням, Internet доступ.

Критерії оцінювання (окремо для кожного виду навчальної діяльності)	Оцінювання проводиться за 100-бальною шкалою.					
	Оцінка за шкалою ECTS		Оцінка в балах	Оцінка за національною шкалою		
				Екзамен, диференційований залік	залік	
		Відмінно		Відмінно		зараховано
		Дуже добре		Добре		
		Добре				
		Задовільно		Задовільно		
	Достатньо					
	Незадовільно		Незадовільно		не зараховано	
	<p>Поточне оцінювання: впродовж семестру студент може отримати 50 балів.</p> <p>Академічна доброчесність: Очікується, що роботи студентів будуть їх оригінальними дослідженнями чи міркуваннями. Відсутність посилань на використані джерела, фабрикування джерел, списування, втручання в роботу інших студентів становлять, але не обмежують, приклади можливої академічної недоброчесності. Виявлення ознак академічної недоброчесності в письмовій роботі студента є підставою для її незарахування викладачем, незалежно від масштабів плагіату чи обману.</p> <p>Відвідання занять є важливою складовою навчання. Очікується, що всі студенти відвідають усі практичні заняття курсу. Студенти повинні інформувати викладача про неможливість відвідати заняття. У будь-якому випадку студенти зобов'язані дотримуватися термінів визначених для виконання рефератів та та доповідей, передбачених курсом.</p> <p>Література. Уся література, яку студенти не зможуть знайти самостійно, буде надана викладачем виключно в освітніх цілях без права її передачі третім особам. Студенти заохочуються до використання також й іншої літератури та джерел, яких немає серед рекомендованих.</p> <p>Політика виставлення балів. Враховуються бали набрані за виконання рефератів, на семінарських заняттях (презентація доповідей) та бали підсумкового оцінювання. При цьому обов'язково враховуються присутність на заняттях та активність студента під час семінарського заняття; недопустимість пропусків та запізнь на заняття; користування мобільним телефоном, планшетом чи іншими мобільними пристроями під час заняття в цілях не пов'язаних з навчанням; списування та плагіат; несвоєчасне виконання поставленого завдання і т. ін.</p> <p>Жодні форми порушення академічної доброчесності не толеруються.</p>					
Питання до заліку чи екзамену.	Залік виставляється за результатами виконаних робіт, описаних вище.					
Опитування	Анкету-оцінку з метою оцінювання якості курсу буде надано по завершенню курсу.					

Схема курсу «Науковий семінар»

Тиждень	Тема, план, короткі тези	Форма діяльності (заняття)	Література. Ресурси в інтернеті	Завдання, год.	Термін виконання
1	Тема 1. Основи SDLC, визначення Тема 2. SDLC методології	лекція (4 год.)	[1,2,3]	Опрацювання лекційного матеріалу (3 год.) Вивчення літератури за темою. (4 год.)	1 тиждень
2	Тема 3. Waterfall Тема 4. Agile Scrum	лекція (4 год.)	[1,2]	Опрацювання лекційного матеріалу (6 год.) Вивчення літератури за темою. (8 год.)	1 тиждень
3	Тема 5. Agile Kanban Тема 6. Проектний менеджмент з Agile Scrum	лекція (4 год.)	[1,2,3]	Опрацювання лекційного матеріалу (6 год.) Вивчення літератури за темою. (8 год.)	1 тиждень
4	Тема 7. Планування і оцінка часу в Agile Тема 8. Scaled Agile Тема 9. Метрики продуктивності	лекція (4 год.)	[1,2]	Опрацювання лекційного матеріалу (6 год.) Вивчення літератури за темою. (8 год.)	1 тиждень
5-8	Тема 10. Метрики якості Тема 11. Інструменти SDLC Тема 12. Вибір SDLC методології	Лабораторні заняття (8 год.)	[1,2,3]	Опрацювання лекційного матеріалу (6 год.) Вивчення літератури за темою. (16 год.)	4 тижні
9	Залікова робота (тест). Підведення підсумків курсу.	Лабораторні			На занятті.

		заняття			
--	--	---------	--	--	--