Міжнародний гуманітарний університет

Факультет кібербезпеки, програмної інженерії та комп'ютерних наук

Кафедра комп'ютерної інженерії та інноваційних технологій

СИЛАБУС НАВЧАЛЬНОЇ ДИСЦИПЛІНИ

ТЕХНОЛОГІЇ СТВОРЕННЯ ПРОГРАМНИХ ПРОДУКТІВ

Галузь знань 12 Інформаційні технології
Спеціальність 121 Інженерія програмного забезпечення
Назва освітньої програми Інженерія програмного забезпечення
Рівень вищої освіти перший (бакалаврський)

Розробники і викладачі	Контактний телефон	Електронна адреса
доцент кафедри комп'ютерної інженерії та інноваційних технологій к.т.н., доцент **Лебедєва Олена Юріївна**	067-485-86-19	lebedeva.onmu@gmail.com

1. АНОТАЦІЯ ДО КУРСУ

Дисципліна «Технології створення програмних продуктів» спрямована на формування системного розуміння процесів розробки програмного забезпечення від етапу формування вимог до введення програмного продукту в експлуатацію та його супроводження. Курс охоплює моделі життєвого циклу програмного забезпечення, сучасні методології розробки, принципи архітектурного проєктування, компонентний підхід, використання систем контролю версій та інструментальних засобів автоматизації. Значна увага приділяється забезпеченню якості програмного забезпечення, тестуванню, верифікації та валідації, а також управлінню змінами та інтеграції програмних компонентів.

У межах дисципліни розглядаються питання організації процесу розробки, документування, застосування сучасних технологій та стандартів програмної інженерії. Практична складова курсу орієнтована на набуття здобувачами досвіду проєктування архітектури, модульної реалізації, інтеграції компонентів, застосування систем контролю версій та процедур погодження змін, що дає змогу сформувати навички професійної взаємодії в процесі створення програмного продукту.

МЕТА ДИСЦИПЛІНИ. Метою викладання навчальної дисципліни є формування у здобувачів здатності реалізовувати фази та ітерації життєвого циклу програмного забезпечення із застосуванням сучасних методологій, технологій та інструментальних засобів, проєктувати архітектуру програмних систем, забезпечувати якість програмного продукту та організовувати процес його інтеграції й супроводження.

Опанування дисципліни забезпечує розвиток умінь обґрунтовано обирати технології розробки, застосовувати методи компонентної реалізації, здійснювати тестування та контроль якості, планувати етапи виконання проєкту, а також використовувати системи контролю версій і засоби автоматизації процесів розробки.

ПРЕРЕКВІЗИТИ. Необхідною умовою для опанування дисципліни є розуміння предметної області інженерії програмного забезпечення, основних видів професійної діяльності фахівця, загальних етапів створення програмних систем та їх місця в інформаційних технологіях, сформоване під час вивчення дисципліни «Вступ до спеціальності», а також у процесі проходження навчальної практики. Здобувач повинен орієнтуватися в структурі програмного продукту, мати уявлення про вимоги до якості програмного забезпечення та володіти базовими навичками роботи з програмними інструментами.

ПОСТРЕКВІЗИТИ. Знання та вміння, отримані під час вивчення дисципліни, є основою для подальшого опанування навчальних компонент: «Управління ІТ-проєктами», «Організація баз даних та знань», «Якість та тестування програмного забезпечення», «Економічна ефективність програмних систем», а також для проходження технологічної, виробничої та переддипломної практик і виконання кваліфікаційної роботи. Набуті компетентності забезпечують здатність планувати етапи створення програмного продукту, обґрунтовувати вибір технологій і методів розробки, здійснювати інтеграцію та супроводження програмних систем.

2. ОЧІКУВАНІ КОМПЕТЕНТНОСТІ, ЯКІ ПЛАНУЄТЬСЯ СФОРМУВАТИ, ТА ДОСЯГНЕННЯ РЕЗУЛЬТАТІВ НАВЧАННЯ

У процесі реалізації програми дисципліни формуються наступні компетентності із передбачених освітньою програмою:

Інтегральна компетентність

Здатність розв’язувати складні спеціалізовані завдання або практичні проблеми інженерії програмного забезпечення, що характеризуються комплексністю та невизначеністю умов, із застосуванням теорій та методів інформаційних технологій.

Загальні компетентності

К07. Здатність працювати в команді.

Спеціальні (фахові, предметні) компетентності

К15. Здатність розробляти архітектури, модулі та компоненти програмних систем.

К16. Здатність формулювати та забезпечувати вимоги щодо якості програмного забезпечення у відповідності з вимогами замовника, технічним завданням та стандартами.

К19. Володіння знаннями про інформаційні моделі даних, здатність створювати програмне забезпечення для зберігання, видобування та опрацювання даних.

К22. Здатність накопичувати, обробляти та систематизувати професійні знання щодо створення і супроводження програмного забезпечення та визнання важливості навчання протягом всього життя.

К23. Здатність реалізовувати фази та ітерації життєвого циклу програмних систем та інформаційних технологій на основі відповідних моделей і підходів розробки програмного забезпечення.

К24. Здатність здійснювати процес інтеграції системи, застосовувати стандарти і процедури управління змінами для підтримки цілісності, загальної функціональності і надійності програмного забезпечення.

Результати навчання

ПР03. Знати основні процеси, фази та ітерації життєвого циклу програмного забезпечення.

ПР06. Уміння вибирати та використовувати відповідну задачі методологію створення програмного забезпечення.

ПР12. Застосовувати на практиці ефективні підходи щодо проектування програмного забезпечення.

ПР13. Знати і застосовувати методи розробки алгоритмів, конструювання програмного забезпечення та структур даних і знань.

ПР15. Мотивовано обирати мови програмування та технології розробки для розв’язання завдань створення і супроводження програмного забезпечення.

ПР17. Вміти застосовувати методи компонентної розробки програмного забезпечення.

ПР19. Знати та вміти застосовувати методи верифікації та валідації програмного забезпечення.

ПР20. Знати підходи щодо оцінки та забезпечення якості програмного забезпечення.

ПР22. Знати та вміти застосовувати методи та засоби управління проектами.

Заплановані результати навчання за навчальною дисципліною

Знання:

– основні процеси, фази та ітерації життєвого циклу програмного забезпечення;
– класичні та гнучкі методології розробки програмних систем, їх принципи та сфери застосування;
– принципи формування, аналізу та документування вимог до програмного забезпечення;
– підходи до проєктування архітектури програмних систем та компонентної реалізації;
– сучасні інструментальні засоби розробки, системи контролю версій та засоби автоматизації;
– атрибути якості програмного забезпечення, стандарти та моделі оцінювання якості;
– методи тестування, верифікації та валідації програмного забезпечення;
– основи управління ІТ-проєктами, управління змінами та конфігурацією програмних систем.

Уміння:

– обирати та обґрунтовувати модель життєвого циклу і методологію розробки залежно від задачі;
– формулювати функціональні та нефункціональні вимоги до програмного продукту;
– проєктувати архітектуру програмної системи та визначати структуру її компонентів;
– застосовувати системи контролю версій для організації змін і інтеграції модулів;
– реалізовувати компонентний підхід під час розробки програмного забезпечення;
– розробляти тестові сценарії та здійснювати модульне й інтеграційне тестування;
– оцінювати якість програмного забезпечення та аналізувати результати перевірки;
– планувати етапи виконання програмного проєкту та оцінювати трудомісткість робіт.

Навички:

– системного аналізу процесів створення програмного забезпечення;
– застосування інструментальних засобів розробки та автоматизації збирання програмного продукту;
– організації роботи з репозиторіями та узгодження змін у процесі розробки;
– документування вимог, архітектурних рішень і результатів тестування;
– інтеграції програмних компонентів у цілісну систему;
– контролю якості та супроводження програмного продукту на різних етапах його життєвого циклу;
– професійної взаємодії під час обговорення та погодження технічних рішень.

3. ОБСЯГ ТА ОЗНАКИ КУРСУ

Загалом		**Вид заняття** (денна форма/заочна форма)				**Ознаки курсу**		
Кредитів ЄКТС	**Годин**	**Лекційні заняття**	**Практичні заняття**	**Лабораторні роботи**	**Самостійна робота**	**Курс, (рік навчання)**	**Семестр**	**Обов'язкова / вибіркова**
5	150	26/12	38/8	0/0	86/130	2	3	Обов'язкова

4. СТРУКТУРА НАВЧАЛЬНОЇ ДИСЦИПЛІНИ

Назви змістових модулів і тем	**Кількість годин**									
	Денна форма					**Заочна форма**				
	Заг.	**у тому числі**				**Заг.**	**у тому числі**			
		Лекц.	**Практ.**	**Лаб.**	**Сам. роб.**		**Лекц.**	**Практ.**	**Лаб.**	**Сам. роб.**
Тема 1. Життєвий цикл програмного забезпечення та методології розробки	30	6	8	0	16	30	2	2	0	26
Тема 2. Проєктування та архітектура програмних систем	34	6	10	0	18	34	4	2	0	28
Тема 3. Технології реалізації програмного забезпечення	32	6	10	0	16	32	2	2	0	28

Тема 4. Забезпечення якості програмного забезпечення	30	4	6	0	20	30	2	2	0	26
Тема 5. Управління проєктами та супроводження програмних продуктів	24	4	4	0	16	24	2	0	0	22
Загалом	**150**	**26**	**38**	**0**	**86**	**150**	**12**	**8**	**0**	**130**
Підсумковий контроль – екзамен										

5. ТЕХНІЧНЕ Й ПРОГРАМНЕ ЗАБЕЗПЕЧЕННЯ / ОБЛАДНАННЯ

Здобувачі отримують теми та питання курсу, основну і додаткову літературу, рекомендації, завдання та оцінки за їх виконання як традиційним шляхом, так і з використанням платформ онлайн навчання на основі інформаційних систем Moodle або Google Classroom. Практичні навички у пошуку та аналізу інформації за курсом, з оформлення індивідуальних завдань, тощо, здобувачі можуть отримувати як за допомогою власного обладнання, так і користуючись університетськими комп'ютерними класами та бібліотекою.

Для виконання практичних робіт з дисципліни використовується безоплатне програмне забезпечення: середовище розробки Visual Studio Code або PyCharm Community Edition з інтерпретатором Python (версія 3.x), система контролю версій Git та хмарний сервіс GitHub (або GitLab) для організації репозиторіїв і інтеграції змін. Для реалізації модульного тестування застосовуються вбудований модуль unittest або фреймворк pytest. За потреби використовується GitHub Actions для автоматизації збирання та тестування програмного продукту.

Зазначені інструменти забезпечують розробку, налагодження, тестування, інтеграцію та документування програмних компонентів відповідно до сучасних технологій створення програмного забезпечення. За потреби можуть використовуватися онлайн-середовища програмування для розробки та тестування програм без встановлення локального програмного забезпечення..

6. САМОСТІЙНА РОБОТА

До самостійної роботи здобувачів відносяться наступні види діяльності:

1. Ознайомлення з науковою та навчальною літературою відповідно зазначених у програмі тем.
2. Опрацювання лекційного матеріалу.
3. Підготовка до практичних занять.
4. Консультації з викладачем протягом семестру.
5. Самостійне опрацювання окремих питань навчальної дисципліни.
6. Підготовка та виконання індивідуальних завдань у вигляді есе, рефератів тощо.
7. Підготовка до підсумкового контролю.

Назва теми	Кількість годин	
	Денна форма	Заочна форма
Тема 1. Життєвий цикл програмного забезпечення та методології розробки 1. Порівняння моделей життєвого циклу програмного забезпечення та сфери їх застосування. 2. Аналіз принципів Agile та їх відмінність від класичних моделей. 3. Ролі та артефакти Scrum-процесу. 4. Методи збору та формалізації вимог до програмного забезпечення. 5. Нефункціональні вимоги та їх вплив на архітектуру системи. 6. Управління змінами вимог у процесі розробки.	16	26
Тема 2. Проєктування та архітектура програмних систем 1. Архітектурні стилі програмних систем та критерії їх вибору. 2. Принципи модульності та слабкої зв’язності компонентів. 3. Використання UML-діаграм у процесі проєктування. 4. Проєктування програмних інтерфейсів та контрактів взаємодії. 5. Компонентна розробка та повторне використання коду. 6. Проблеми інтеграції програмних модулів. 7. Забезпечення масштабованості та розширюваності програмної системи.	18	28
Тема 3. Технології реалізації програмного забезпечення 1. Організація роботи з системами контролю версій у процесі розробки. 2. Стратегії гілкування та управління змінами в Git. 3. Інструментальні засоби автоматизації збирання програмного продукту. 4. Принципи безперервної інтеграції. 5. Інформаційні моделі даних у програмних продуктах. 6. Методи забезпечення коректності та валідації даних.	16	28
Тема 4. Забезпечення якості програмного забезпечення 1. Поняття якості програмного забезпечення та її характеристики. 2. Міжнародні стандарти якості програмного забезпечення.	20	26

3. Рівні та види тестування програмного забезпечення. 4. Модульне та інтеграційне тестування. 5. Верифікація та валідація програмного забезпечення. 6. Документування результатів тестування та управління дефектами.		
Тема 5. Управління проєктами та супроводження програмних продуктів 1. Основи управління ІТ-проєктами та життєвий цикл проєкту. 2. Методи оцінювання трудомісткості програмних проєктів. 3. Управління ризиками у процесі розробки програмного забезпечення. 4. Управління конфігурацією та реліз-менеджмент. 5. Види супроводження програмного забезпечення. 6. Професійна документація програмного продукту та її структура.	16	22
Загалом	**86**	**130**

7. ВИДИ ТА МЕТОДИ КОНТРОЛЮ

Види контролю	Складові оцінювання
Поточний контроль, який здійснюється під час проведення практичних (лабораторних) занять, виконання самостійної роботи, індивідуальних завдань, дослідної робота здобувача тощо	50%
Підсумковий контроль, який здійснюється у ході проведення екзамену	50%

Методи діагностики знань (контролю)	Фронтальне опитування, розв`язання практичних завдань, тестування здобувачів, наукові доповіді, реферати, усні повідомлення, екзамен

8. ОЦІНЮВАННЯ ПОТОЧНОЇ, САМОСТІЙНОЇ ТА ІНДИВІДУАЛЬНОЇ РОБОТИ ЗДОБУВАЧІВ

Види роботи	Планові терміни виконання	Форми контролю та звітності	Максимальний відсоток оцінювання
Систематичність і активність роботи на практичних заняттях			
1.1. Підготовка до практичних занять	Відповідно до робочої програми та розкладу занять	Перевірка обсягу та якості засвоєного матеріалу під час практичних занять	**30**
Виконання завдань для самостійного опрацювання			
1.2. Індивідуальне тестування	Відповідно до робочої програми та розкладу занять	Перевірка результатів індивідуального тестування, перевірка конспектів лекцій, перевірка рефератів	**10**
Виконання індивідуальних завдань (дослідна робота здобувача)			
1.3. Інші види індивідуальних завдань, в т.ч. підготовка наукових публікацій, участь у роботі круглих столів, конференцій тощо.	Відповідно до робочої програми та розкладу занять	Обговорення результатів проведеної роботи під час аудиторних занять, наукових конференцій та круглих столів.	**10**
Разом балів за поточний контроль			**50**
Підсумковий контроль – екзамен			**50**
Загальний результат оцінювання			**100**

Поточний контроль знань здобувачів при підсумковому контролі у формі екзамену (максимум 50 балів) оцінюється за шкалою («2», «3», «4», «5»), з обов'язковим переведенням балів за кожний вид роботи таким чином:

– за підготовку до аудиторних занять (максимум 30 балів) здобувачу освіти необхідно мати не менше 1/2 оцінок від загальної кількості практичних (лабораторних) занять, для переводу балів за підготовку до аудиторних занять середньоарифметична оцінка множиться на коефіцієнт 6;

– за виконання завдань для самостійного опрацювання (тестування, завдання, підготовка реферату за заданою тематикою) (максимум 10 балів) здобувачу освіти необхідно мати не менше 1/2 оцінок від загальної кількості програмного матеріалу, що виноситься на самостійне вивчення, для переводу зазначених балів середньоарифметична оцінка множиться на коефіцієнт 2;

– за інші види індивідуальних завдань, в т.ч. підготовку наукових публікацій, участь у роботі круглих столів, конференцій тощо (максимум 10 балів) здобувачу освіти необхідно мати не менше 1/2 оцінок від загальної кількості інших видів індивідуальних завдань, для переводу балів за підготовку до аудиторних занять середньоарифметична оцінка множиться на коефіцієнт 2.

Критерії оцінювання поточного контролю

– **«2»** – виставляється за неправильну відповідь (письмову роботу), яка не відповідає змісту теми та свідчить про нерозуміння її основних положень;

– **«3»** – виставляється за відповідь (письмову роботу), яка відповідає змісту теми, але є неповною і неточною у визначенні понять, свідчить про неповне розуміння основних положень навчального матеріалу, свідчить про те, що знання здобувача мають фрагментарний, поверховий характер;

– **«4»** – оцінюється правильна і обґрунтована відповідь (письмова робота), з якої видно, що здобувач знає й розуміє теоретичний матеріал в обсязі навчальної теми, володіє необхідними навичками, не допускає суттєвих помилок.

– **«5»** – оцінюється повна, правильна, ґрунтовна відповідь (письмова робота) на запитання теми, що передбачає логічність і послідовність викладу матеріалу, володіння термінологією, показує вміння повно й глибоко використовувати теоретичні знання в практичній площині.

Критерії оцінювання підсумкового контролю у формі екзамену

Максимальна оцінка підсумкового контролю у формі екзамену не може перевищувати 50 балів. При цьому рівень знань оцінюється:

– **від 40 до 50 балів** – здобувач виявляє особливі творчі здібності, вміє самостійно знаходити та опрацьовувати необхідну інформацію, демонструє знання матеріалу, проводить узагальнення і висновки;

– **від 30 до 40 балів** – здобувач володіє знаннями матеріалу, але допускає незначні помилки у формуванні термінів, категорій, проте за допомогою викладача швидко орієнтується і знаходить правильні відповіді;

– **від 20 до 30 балів** – здобувач відтворює значну частину теоретичного матеріалу, виявляє знання і розуміння основних положень, з допомогою викладача може аналізувати навчальний матеріал, але дає недостатньо обґрунтовані, невичерпні відповіді, допускає помилки.

– **від 10 до 20 балів** – здобувач володіє навчальним матеріалом на середньому рівні, допускає помилки, серед яких є значна кількість суттєвих;

– від 0 до 10 балів – здобувач володіє навчальним матеріалом на рівні, вищому за початковий, значну частину його відтворює на репродуктивному рівні, на всі запитання дає необґрунтовані, невичерпні відповіді, допускає помилки.

9. ОЦІНЮВАННЯ ЗАГАЛЬНИХ РЕЗУЛЬТАТІВ ЗНАНЬ ЗДОБУВАЧІВ

Загальні результати знань здобувачів по даній дисципліні оцінюються наступним чином:

– **«Відмінно»/«зараховано» (A)** – від 90 до 100 балів. Здобувач виявляє особливі творчі здібності, вміє самостійно знаходити та опрацьовувати необхідну інформацію, демонструє знання матеріалу, проводить узагальнення і висновки. Був присутній на лекціях та семінарських заняттях, під час яких давав вичерпні, обґрунтовані, теоретично і практично правильні відповіді, має конспект з виконаними завданнями до самостійної роботи, презентував реферат за заданою тематикою, проявляє активність і творчість у науково-дослідній роботі;

– **«Добре»/«зараховано» (B)** – від 82 до 89 балів. Здобувач володіє знаннями матеріалу, але допускає незначні помилки у формуванні термінів, категорій, проте за допомогою викладача швидко орієнтується і знаходить правильні відповіді. Був присутній на лекціях та семінарських заняттях, має конспект з виконаними завданнями до самостійної роботи, презентував реферат за заданою тематикою, проявляє активність і творчість у науково-дослідній роботі;

– **«Добре»/«зараховано» (C)** – від 74 до 81 балів. Здобувач відтворює значну частину теоретичного матеріалу, виявляє знання і розуміння основних положень, з допомогою викладача може аналізувати навчальний матеріал, але дає недостатньо обґрунтовані, невичерпні відповіді, допускає помилки. При цьому враховується наявність конспекту з виконаними завданнями до самостійної роботи, реферату та активність у науково-дослідній роботі;

– **«Задовільно»/«зараховано» (D)** – від 64 до 73 балів. Здобувач був присутній не на всіх лекціях та семінарських заняттях, володіє навчальним матеріалом на середньому рівні, допускає помилки, серед яких є значна кількість суттєвих. При цьому враховується наявність конспекту з виконаними завданнями до самостійної роботи та рефератів;

– **«Задовільно»/«зараховано» (E)** – від 60 до 63 балів. Здобувач був присутній не на всіх лекціях та семінарських заняттях, володіє навчальним матеріалом на рівні, вищому за початковий, значну частину його відтворює на репродуктивному рівні, на всі запитання дає необґрунтовані, невичерпні відповіді, допускає помилки, має неповний конспект з завданнями до самостійної роботи.

– **«Незадовільно з можливістю повторного складання»/«не зараховано» (FX)** – від 35 до 59 балів. Здобувач володіє матеріалом на рівні окремих фрагментів, що становлять незначну частину навчального матеріалу.

– **«Незадовільно з обов'язковим повторним вивченням дисципліни»/«не зараховано» (F)** – від 0 до 34 балів. Здобувач не володіє навчальним матеріалом.

Таблиця відповідності результатів контролю знань за різними шкалами

100-бальна шкала	Шкала ECTS	Національна шкала	
		екзамен	залік
90-100	A	Відмінно	Зараховано
82-89	B	Добре	Зараховано
74-81	C		
64-73	D	Задовільно	Зараховано
60-63	E		
35-59	FX	Незадовільно	Не зараховано
0-34	F		

10. ДОТРИМАННЯ ВИМОГ АКАДЕМІЧНОЇ ДОБРОЧЕСНОСТІ

Викладання та засвоєння навчальної дисципліни передбачає неухильне дотримання вимог академічної доброчесності, яка регулюється Законом України «Про освіту» (Відомості Верховної Ради (ВВР), 2014, № 37-38, ст. 2004), «Методичними рекомендаціями для закладів вищої освіти з підтримки принципів академічної доброчесності» (Лист МОН № 1/9-650 від 23.10.2018), іншими нормативними документами. Зокрема, основні засади дотримання академічної доброчесності визначено у «Положенні про академічну доброчесність у Міжнародному гуманітарному університеті», «Кодексі академічної доброчесності Міжнародного гуманітарного університету» та «Інструкції щодо процедури технічної перевірки на наявність текстових запозичень (академічного плагіату)», затвердженої наказом Міжнародного гуманітарного університету № 708а від 03.05.2024.

Відповідно до ст. 42 Закону України «Про освіту» академічна доброчесність – це сукупність етичних принципів та визначених законом правил, якими мають керуватися учасники освітнього процесу під час навчання, викладання та провадження наукової (творчої) діяльності з метою забезпечення довіри до результатів навчання та/або наукових (творчих) досягнень.

Дотримання академічної доброчесності передбачає:

– самостійне виконання навчальних завдань, завдань поточного та підсумкового контролю результатів навчання (для осіб з особливими освітніми потребами ця вимога застосовується з урахуванням їхніх індивідуальних потреб і можливостей);

– посилання на джерела інформації у разі використання ідей, розробок, тверджень, відомостей;

– дотримання норм законодавства про авторське право і суміжні права;

– надання достовірної інформації про результати власної навчальної (наукової, творчої) діяльності, використані методики досліджень і джерела інформації.

11. РЕКОМЕНДОВАНА ЛІТЕРАТУРА

Основна

1. Pressman, R. S., Maxim, B. R. Software Engineering: A Practitioners Approach. McGraw-Hill Education. 2019. 1408p. ISBN: 9781260548006, 1260548007.
2. Sommerville, I. Software Engineering, Global Edition. Pearson Education. 2016. 816p. ISBN: 9781292096148, 1292096144.
3. Martin, R.C. Clean Architecture: A Craftsman's Guide to Software Structure and Design. Prentice Hall. 2018. 404p. ISBN: 9780134494166, 0134494164.
4. Liang, Q. Continuous Delivery 2.0: Business-leading DevOps Essentials. CRC Press. 2021. 362p. ISBN: 9781000474763, 1000474763.
5. Chacon, Scott. Pro Git: Everything You Need to Know About Git. Apress, 2022. 497p.

Допоміжна

6. Ahmed, Ashfaque, and Prasad, Bhanu. Foundations of Software Engineering. CRC Press, 2016. 475p. ISBN: 9781498737609, 1498737609
7. Adewole, Ayobami. C# and .NET Core Test-Driven Development: Dive Into TDD to Create Flexible, Maintainable, and Production-ready .NET Core Applications. Packt Publishing, 2018. 300p. ISBN: 9781788299923, 1788299922
8. Peters, Lawrence J.. Software Project Management: Methods and Techniques. CRC Press. 244p. ISBN: 9781040035238, 104003523X.
9. Gechman, Marvin. Project Management of Large Software-Intensive Systems. CRC Press, 2019. 392p. ISBN: 9780429648168, 0429648162.
10. Singh, Shweta. Role of Project Management Software in the Manufacturing Sector. Германия, GRIN Verlag, 2022. 102p. ISBN: 9783346776754, 3346776751
11. ISO/IEC 25010:2011 Systems and software engineering – Software product quality requirements and evaluation (SQuaRE) – System and software quality models.
12. ISO/IEC/IEEE 12207:2017 Systems and software engineering – Software life cycle processes.

Інформаційні ресурси

13. Scrum Guide. URL: https://scrumguides.org
14. Git Documentation. URL: https://git-scm.com/docs
15. GitHub Docs. URL: https://docs.github.com
16. Atlassian Agile Coach. URL: https://www.atlassian.com/agile
17. Software Engineering Institute (SEI). URL: https://www.sei.cmu.edu