

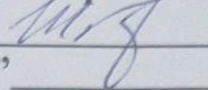
ЗАКЛАД ВИЩОЇ ОСВІТИ
«УНІВЕРСИТЕТ КОРОЛЯ ДАНИЛА»

Факультет суспільних і прикладних наук

Кафедра інформаційних технологій

ЗАТВЕРДЖУЮ

Проректор з методичної роботи

 Ярослав ШТАНЬКО

“ ” 2024 р.

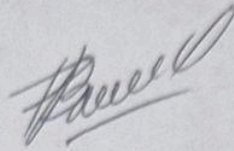
КОНСТРУЮВАННЯ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ

СИЛАБУС НАВЧАЛЬНОЇ ДИСЦИПЛІНИ

Галузь знань:	12 Інформаційні технології
Спеціальність:	121 Інженерія програмного забезпечення
Освітньо-професійна (освітньо-наукова) програма:	Розробка та тестування програмного забезпечення
Освітній рівень:	(перший) бакалаврський
Статус дисципліни:	обов'язкова
Мова викладання, навчання та оцінювання:	українська

РОЗРОБНИК:

асистент кафедри ІТ



Роман ВИТВИЦЬКИЙ

ЗАТВЕРДЖЕНО:

на засіданні кафедри інформаційних технологій,
протокол № 1 від 28.08.2024 р.

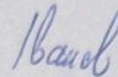
Завідувач кафедри



Сергій ВАЩИШАК

УЗГОДЖЕНО:

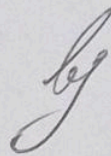
Гарант ОПП



Олександр ІВАНОВ

на засіданні кафедри інформаційних технологій,
протокол № 1 від 28.08.2024 р.

Завідувач кафедри



Сергій ВАЩИШАК

СХВАЛЕНО:

на засіданні Науково-методичної ради, протокол № 1 від 30.08.2024 р.

e-mail	roman.vytvytskyi@ukd.edu.ua
Номер аудиторії чи кафедри	206
Посилання на сайт	https://ukd.edu.ua/person/roman-vytvytskyi-k
Сторінка курсу в СДО	https://online.ukd.edu.ua/course/

ВСТУП

Анотація навчальної дисципліни

Навчальна дисципліна "Конструювання програмного забезпечення" є складовою освітньо-професійної програми підготовки фахівців за освітнім ступенем "бакалавр" галузі знань 12 "Інформаційні технології" спеціальності 121 "Інженерія програмного забезпечення", освітньої програми "Розробка та тестування програмного забезпечення".

Слухачі дисципліни повинні одержати теоретичні знання та практичні навички з використання методів та засобів конструювання програмного забезпечення в систематизованому вигляді для їх застосування на процесах розробки програмних систем на основі мови програмування Python, а також основам роботи в команді.

Мета навчальної дисципліни

Мета дисципліни – формування у студентів бази теоретичних знань та умінь щодо сучасних методів та засобів конструювання програмних систем які розвивають здатність розв'язувати складні спеціалізовані завдання і практичні проблеми інженерії програмного забезпечення.

В результаті вивчення дисципліни студент повинен **знати**:

- основи поведження в команді;
- інструменти для роботи в команді;
- синтаксис мови програмування Python;
- процеси та інструменти для відлагодження програм;
- як працювати з фреймворками;
- інструментальні засоби конструювання програмного забезпечення.

В результаті вивчення дисципліни студент повинен **вміти**:¹

- вибирати мови програмування для створення програмного забезпечення;
- виконувати розгортання та збірку програм;
- керувати процесом конструювання програм;
- обирати методику конструювання;
- застосовувати набуті знання на практиці.

¹ поняття вміти і знати повинні співвідноситися з програмними результатами навчання

Компетентності та результати навчання, яких набувають здобувачі освіти внаслідок вивчення навчальної дисципліни “Конструювання програмного забезпечення” (шифри та зміст компетентностей та програмних результатів навчання вказано відповідно до ОПП/ОНП «Розробка та тестування програмного забезпечення»).

Шифр та назва компетентності	Шифр та назва програмних результатів навчання
ЗК2. Здатність застосовувати знання у практичних ситуаціях.	ПРН4. Знати і застосовувати професійні стандарти і інші нормативно-правові документи в галузі інженерії програмного забезпечення.
ЗК3. Здатність спілкуватися державною мовою як усно, так і письмово.	ПРН9. Знати та вміти використовувати методи та засоби збору, формулювання та аналізу вимог до програмного забезпечення.
ЗК4. Здатність спілкуватися іноземною мовою як усно, так і письмово.	ПРН14. Застосовувати на практиці інструментальні програмні засоби доменного аналізу, проектування, тестування, візуалізації, вимірювань та документування програмного забезпечення.
ЗК10. Здатність діяти соціально відповідально та свідомо.	ПРН20. Знати підходи щодо оцінки та забезпечення якості програмного забезпечення.
ФК1. Здатність ідентифікувати, класифікувати та компетентності формулювати вимоги до програмного забезпечення.	ПРН23. Вміти документувати та презентувати результати розробки програмного забезпечення.
ФК2. Здатність брати участь у проектуванні (ФК) програмного забезпечення, включаючи проведення моделювання (формальний опис) його структури, поведінки та процесів функціонування.	ПРН24. Вміти проводити розрахунок економічної ефективності програмних систем.

ФК4. Здатність формулювати та забезпечувати вимоги щодо якості програмного забезпечення у відповідності з вимогами замовника, технічним завданням та стандартами.	
ФК5. Здатність дотримуватися специфікацій, стандартів, правил і рекомендацій в професійній галузі при реалізації процесів життєвого циклу.	
ФК12. Здатність здійснювати процес інтеграції системи, застосовувати стандарти і процедури управління змінами для підтримки цілісності, загальної функціональності і надійності програмного забезпечення.	
ФК13. Здатність обґрунтовано обирати та освоювати інструментарій з розробки та супроводження програмного забезпечення.	

ОПИС НАВЧАЛЬНОЇ ДИСЦИПЛІНИ

Курс	2		
Семестр	4		
Кількість кредитів ЄКТС	6		
Аудиторні навчальні заняття		денна форма	заочна форма
	лекції	30 (в годинах)	6 (в годинах)
	семінари, практичні	30 (в годинах)	6 (в годинах)
Самостійна робота		120 (в годинах)	168 (в годинах)
Форма підсумкового контролю	Екзамен	4	2

Структурно-логічна схема вивчення навчальної дисципліни²:

Пререквізити	Постреквізити
Інформаційно-комунікаційні технології	Групова динаміка і комунікації
	Паралельні та розподілені обчислення
	Менеджмент проектів програмного забезпечення
	Якість програмного забезпечення та тестування

ПРОГРАМА НАВЧАЛЬНОЇ ДИСЦИПЛІНИ

Перелік тем лекційного матеріалу

Змістовий модуль 1

Ознайомлення з фундаментальними основами програмування та роботи в команді

Тема 1. One day of Software engineer (2 год.).

Вступ в предмет. Опис робочого дня програміста. Ознайомлення з Trello та методологіями розробки. Поняття про meeting, task, release.

Завдання для самостійної роботи. Опис вимог з використанням User Story (10 год.).

Тема 2. Робота в команді (2 год.).

Основи роботи з системою контролю версій Git та Docker. Документація в проекті.

Тема 3. Підготовка до розробки програмного забезпечення (2 год.).

Знайомство з інтегрованим середовищем розробки. Поняття про пакети та модулі, дебагінг коду, віртуальне середовище.

Завдання для самостійної роботи. Методології розробки (10 год.).

Тема 4. Основи мови програмування Python (4 год.).

Типи даних, оператори, умовні вирази, цикли, ітератори, функції, генератори, контекстні менеджери, робота з ОС та файлами.

Тема 5. Об'єктно-орієнтоване програмування в Python (2 год.).

Основні поняття ООП, класи, абстракція, скрапінг та асинхронність. Змістовий модуль II. Основи веб-програмування та фреймворк Django

Тема 6. Веб та фреймворки (2 год.).

CGI/WSGI/ASGI, патерни, MVC, роутинг, ORM, зв'язки в базі даних.

Тема 7. Основи веб-фреймворку Django (8 год.).

Знайомство з фреймворком Django, його основами. Екземпляри класу. Робота з базами даних, запити до бази даних, фільтри, сортування, пошук агрегація та

² тільки для обов'язкових дисциплін

міграція даних. Зв'язок з сервером, файлові сховища, автентифікація та авторизація. Відображення даних, форми, віджети, валідація даних та медіафайли.

Тема 8. Веб-розробка (4 год.).

Основи HTML, CSS та Bootstrap. TypeScript та node.js.

Завдання для самостійної роботи. Верстка інтернет сторінок (10 год.).

Протокол передачі http та https, робота з ssh та сокетами (10 год.). Розміщення веб-ресурсів на web hosting service (10 год.)

Тема 9. Розширені можливості фреймворку Django (4 год.).

Backend розробка. Власні команди Django, оптимізація навантаження, панель налагодження, захист даних.

Зміст практичних занять

Змістовий модуль 1

Тема 1. Робота з Trello, створення дошок. Робота з Jira (2 год.).

Тема 2. Використання системи контролю версій GIT. Розгортання додатків в Docker (2 год.).

Тема 3. Налаштування середовища розробки. Зв'язування проекту та GitHub (2 год.).

Тема 4. Основи програмування на Python. Створення змінних, робота з умовними операторами, циклами, функціями. Робота з файловою системою на програмному рівні (4 год.).

Тема 5. Об'єктно-орієнтоване програмування. Створення класів, об'єктів, абстрактні класи (2 год.).

Змістовий модуль II.

Тема 6. Створення архітектури на основі шаблону MVC. Підключення до бази даних (2 год.).

Тема 7. Робота з фреймворком Django. Створення класів. Робота з базами даних, запити до бази даних, фільтри, сортування, пошук агрегація і міграція. Підключення до сервера, автентифікація та авторизація. Відображення даних, створення форм, віджетів, валідація даних та робота з медіафайлами (8 год.).

Тема 8. Веб-розробка. HTML теги, стилізація за допомогою CSS. Підключення Bootstrap і node.js (4 год.).

Тема 9. Backend розробка. Оптимізація навантаження, панель налагодження, захист даних. (4 год.).

Зміст самостійної роботи здобувачів

Розподіл годин, виділених на вивчення дисципліни:

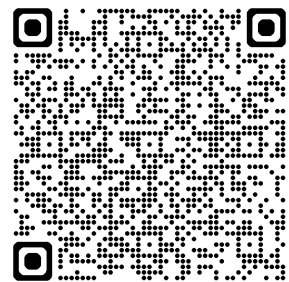
Найменування видів робіт	Розподіл годин за формами навчання	
	денна	заочна
Самостійна робота, год, у т.ч.:	120	168
Опрацювання матеріалу, викладеного на лекціях	10	10
Підготовка до практичних занять та контрольних заходів	10	40
Підготовка звітів з практичних робіт	30	40
Підготовка до поточного контролю	20	20
Опрацювання матеріалу, винесеного на самостійне вивчення	50	58

ПОЛІТИКА КУРСУ

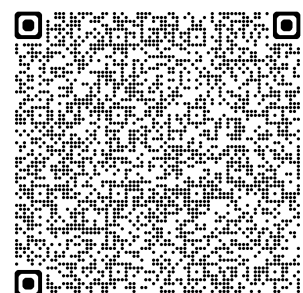
Коротко, з покликанням на відповідну нормативну базу УКД, висвітлити питання.³

1) щодо системи поточного і підсумкового контролю

Організація поточного та підсумкового семестрового контролю знань студентів, проведення практик та атестації, переведення показників академічної успішності за 100-бальною шкалою в систему оцінок за національною шкалою здійснюється згідно з “Положенням про систему поточного і підсумкового контролю, оцінювання знань та визначення рейтингу здобувачів освіти”. Ознайомитись з документом можна за [покликанням](#).



2) щодо оскарження результатів контрольних заходів

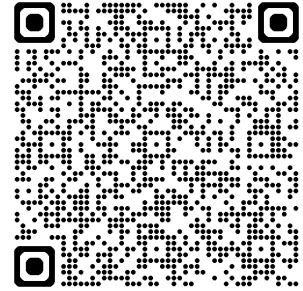


³ зміст пунктів може редагуватись з огляду на особливості курсу

Здобувачі вищої освіти мають право на оскарження оцінки з дисципліни отриманої під час контрольних заходів. Апеляція здійснюється відповідно до «Положення про політику та врегулювання конфліктних ситуацій». Ознайомитись з документом можна за [покликанням](#).

3) щодо відпрацювання пропущених занять

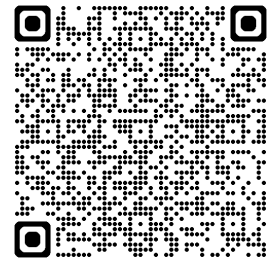
Згідно “Положення про організацію освітнього процесу” здобувач допускається до семестрового контролю конкретної навчальної дисципліни (семестрового екзамену, диференційованого заліку), якщо він виконав усі види робіт, передбачені на семестр навчальним планом та силабусом/робочою програмою навчальної дисципліни, підтвердив опанування на мінімальному рівні результатів навчання (отримав ≥ 35 бали), відпрацював визначені індивідуальним навчальним планом всі лекційні, практичні, семінарські та лабораторні заняття, на яких він був відсутній. Ознайомитись з документом можна за [покликанням](#).



3

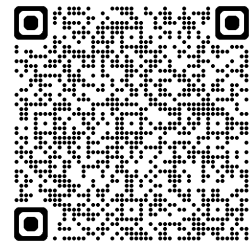
4) щодо дотримання академічної доброчесності

“Положення про академічну доброчесність” закріплює моральні принципи, норми та правила етичної поведінки, позитивного, сприятливого, доброчесного освітнього і наукового середовища, професійної діяльності та професійного спілкування спільноти Університету, викладання та провадження наукової (творчої) діяльності з метою забезпечення довіри до результатів навчання. Ознайомитись з документом можна за [покликанням](#).



5) щодо використання штучного інтелекту

“Положення про академічну доброчесність” визначає політику щодо використання технічних засобів на основі штучного інтелекту в освітньому процесі. Ознайомитись з документом можна за [покликанням](#).⁴ “Положення про систему запобігання та виявлення академічного плагіату, самоплагіату, фабрикації та фальсифікації академічних творів” містить рекомендації щодо використання в академічних текстах генераторів на основі штучного інтелекту. Ознайомитись з документом можна за [покликанням](#).



⁴ визначається політика використання ШІ в навчальній дисципліні - дозволене/заборонене, правила використання

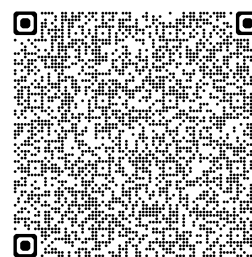
б) щодо використання технічних засобів в аудиторії та правила комунікації

Використання мобільних телефонів, планшетів та інших гаджетів під час лекційних та практичних занять дозволяється виключно у навчальних цілях (для уточнення певних даних, перевірки правопису, отримання довідкової інформації тощо). На гаджетах повинен бути активований режим «без звуку» до початку заняття. Під час занять заборонено надсилання текстових повідомлень, прослуховування музики, перевірка електронної пошти, соціальних мереж тощо. Під час виконання заходів контролю використання гаджетів заборонено (за винятком, коли це передбачено умовами його проведення). У разі порушення цієї заборони результат анулюється без права перескладання.

Комунікація відбувається через електронну пошту і сторінку дисципліни в Moodle.

7) щодо зарахування результатів навчання, здобутих шляхом формальної/інформальної освіти

Процедури визнання результатів навчання, здобутих шляхом формальної/інформальної освіти визначаються «Положенням про порядок визнання результатів навчання, здобутих шляхом неформальної та / або інформальної освіти». Ознайомитись з документом можна за [покликанням](#).⁵



МЕТОДИ НАВЧАННЯ

При вивченні дисципліни застосовується комплекс методів для організації навчання студентів з метою розвитку їх логічного та абстрактного мислення, творчих здібностей, підвищення мотивації до навчання та формування особистості майбутнього фахівця.

Програмний результат навчання⁶	<u>Метод навчання</u>	Метод оцінювання
ПРН4. Знати і застосовувати професійні стандарти і інші нормативно-правові	МН 1.1 - лекція МН 2.4 - комп'ютерні і мультимедійні методи МН 11 - конкретизація МН 7 - аналітичний	МФО 1 - іспит МФО 4 - поточний контроль МФО 5 - усний контроль

⁵ визначається перелік електронних та інших ресурсів та умови перезарахування

⁶ для вибіркового навчання дисциплін вказується результат навчання

документи в галузі інженерії програмного забезпечення.	МН 19 - робота під керівництвом викладача МН 20.1 - кейс-метод МН 20.5 - тренінгові заняття	
ПРН9. Знати та вміти використовувати методи та засоби збору, формулювання та аналізу вимог до програмного забезпечення.	МН 1.1 - лекція МН 1.2 – розповідь -пояснення МН 1.3 - бесіда МН 2.4 - комп'ютерні і мультимедійні методи МН 5 - дедуктивний МН 7 - аналітичний МН 9 - порівняння МН 19 - робота під керівництвом викладача МН 20.1 - кейс-метод МН 20.5 - тренінгові заняття	іспит поточний контроль МФО 5 - усний контроль
ПРН14. Застосовувати на практиці інструментальні програмні засоби доменного аналізу, проектування, тестування, візуалізації, вимірювань та документування програмного забезпечення.	МН 1.1 - лекція МН 2.4 - комп'ютерні і мультимедійні методи МН 7 - аналітичний МН 9 - порівняння МН 19 - робота під керівництвом викладача МН 20.1 - кейс-метод МН 20.5 - тренінгові заняття	іспит поточний контроль усний контроль
ПРН20. Знати підходи щодо оцінки та забезпечення якості програмного забезпечення.	МН 1.1 - лекція МН 1.2 – розповідь -пояснення МН 2.4 - комп'ютерні і мультимедійні методи МН 3.4 - практичні роботи МН 5 - дедуктивний МН 7 - аналітичний МН 9 - порівняння	іспит поточний контроль усний контроль

	<p>МН 15 - проблемно-пошуковий</p> <p>МН 19 - робота під керівництвом викладача</p> <p>МН 20.1 - кейс-метод</p> <p>МН 20.2 - дискусія, диспут</p> <p>МН 20.7 - бесіда-діалог</p>	
<p>ПРН23. Вміти документувати та презентувати результати розробки програмного забезпечення.</p>	<p>МН 1.1 - лекція</p> <p>МН 1.2 – розповідь -пояснення</p> <p>МН 2.4 - комп'ютерні і мультимедійні методи</p> <p>МН 3.4 - практичні роботи</p> <p>МН 7 - аналітичний</p> <p>МН 9 - порівняння</p> <p>МН 19 - робота під керівництвом викладача</p> <p>МН 20.2 - дискусія, диспут</p> <p>МН 20.3 - мозковий штурм</p> <p>МН 20.7 - бесіда-діалог</p>	<p>іспит</p> <p>поточний контроль</p> <p>усний контроль</p>
<p>ПРН24. Вміти проводити розрахунок економічної ефективності програмних систем.</p>	<p>МН 1.1 - лекція</p> <p>МН 2.4 - комп'ютерні і мультимедійні методи</p> <p>МН 3.4 - практичні роботи</p> <p>МН 8 - синтетичний</p> <p>МН 19 - робота під керівництвом викладача</p> <p>МН 20.1 - кейс-метод</p> <p>МН 20.7 - бесіда-діалог</p>	<p>іспит</p> <p>поточний контроль</p> <p>усний контроль</p>

ОЦІНЮВАННЯ РЕЗУЛЬТАТІВ НАВЧАННЯ

Контрольні заходи⁷

(в разі потреби - розділити за семестрами)

<i>Вид</i>	<i>Зміст⁸</i>	<i>% від загальної оцінки</i>	<i>Бал</i>	
			<i>min</i>	<i>max</i>
Поточні контрольні заходи	всього	60	35	60
Підсумкові контрольні заходи	проект	40	25	40
Всього:		100	60	100

Процедура проведення контрольних заходів, а саме поточного контролю знань протягом семестру та підсумкового семестрового контролю, регулюється «Положенням про систему поточного та підсумкового контролю оцінювання знань та визначення рейтингу студентів».

Фіксація **поточного** контролю здійснюється в “Електронному журналі обліку успішності академічної групи” на підставі чотирибальної шкали - “2”; “3”; “4”; “5”. У разі відсутності студента на занятті виставляється “н”. За результатами поточного контролю у Журналі, автоматично визначається підсумкова оцінка, здійснюється підрахунок пропущених занять.

Усі пропущені заняття, а також негативні оцінки студенти зобов'язані відпрацювати впродовж трьох наступних тижнів. У випадку недотримання цієї норми, замість “н” в журналі буде виставлено “0” (нуль балів), без права перездачі. Відпрацьоване лекційне заняття в електронному журналі позначається літерою «в».⁹

Критерії оцінювання¹⁰ (за необхідності, поточного та/або підсумкового контролю)

До підсумкового контролю допускаються студенти які за результатами поточного контролю отримали не менше 35 балів. Усі студенти, що отримали 34 балів і менше, не допускаються до складання підсумкового контролю і на підставі укладання додаткового договору, здійснюють повторне вивчення дисципліни впродовж наступного навчального семестру. За результатами підсумкового контролю

⁷ зміст редагується залежно від наповнення дисципліни

⁸ у випадку наявності видів роботи, які оцінюються окремо (проект, завдання тощо) прописується в окремому рядку; за відсутності - одним рядком визначається вся сукупність аудиторної роботи (опитування, поточні контрольні тощо) та визначається стандартне значення балів (35/60)

⁹ можна вказати теми чи завдання, які є обов'язковими до виконання, а також особисті підходи до оцінювання рівня знань здобувачів під час аудиторної роботи

¹⁰ критерії вказуються згідно з особливостями дисципліни.

(диференційований залік/екзамен) студент може отримати 40 балів. Студенти, які під час підсумкового контролю отримали 24 бали і менше, вважаються такими, що не здали екзамен/диференційований залік і повинні йти на перездачу. Підсумковим контролем вивчення дисципліни є проєкт. Проєкт виконується згідно тематик та вимог.

Загальна семестрова оцінка з дисципліни, яка виставляється в екзаменаційних відомостях оцінюється в балах (згідно з **Шкалою оцінювання знань за ЄКТС**) і є сумою балів отриманих під час поточного та підсумкового контролю.

Шкала оцінювання знань за ЄКТС:

Оцінка за національною шкалою	Рівень досягнень, %	Шкала ECTS
Національна диференційована шкала		
Відмінно	90 – 100	A
Добре	83 – 89	B
	75 – 82	C
Задовільно	67 – 74	D
	60 – 66	E
Незадовільно	35 – 59	FX
	0 – 34	F
Національна недиференційована шкала		
Зараховано	60 – 100	-
Не зараховано	0 – 59	-

Студенти, які не з'явилися на заліки/екзамени без поважних причин, вважаються такими, що одержали незадовільну оцінку.

РЕКОМЕНДОВАНІ ДЖЕРЕЛА ІНФОРМАЦІЇ¹¹

Основна література

1. Маттес Е. Пришвидшений курс Python. Львів : ВСЛ, 2021.
2. Селіверстов Р., Мельничин А. Основи програмування мовою Python: навч. посібник. Львів : ЛНУ імені Івана Франка, 2020.

¹¹ обов'язково: враховувати вимоги [ДСТУ 8302:2015](#) (відповідно до [Наказу № 65, від 4.03. 2016](#)), [рекомендації](#) Національного агентства з забезпечення якості вищої освіти, використовувати літературу за останні 5-7 років, наводити власні публікації за змістом навчальної дисципліни.

3. Роберт М. Чиста архітектура: мистецтво розробки програмного забезпечення". Роберт Мартін, Фабула, 2019. 416 с.
4. Бородкіна І., Бородкин Г. Інженерія програмного забезпечення. Посібник для студентів вищих навчальних закладів. М:Центр навчальної літератури, 2018. 204 с.
5. Lambert К. А. Fundamentals of Python: First Programs, 2nd Edition. Cengage, 2019.
6. Мартін Р. Чистий код, Р.Мартін М.: Фабула, 2019. 416 с.

Електронні інформаційні ресурси

1. Авраменко В.С. Проектування інформаційних систем : навч. посіб. Черкаси: Черкас. нац. ун-т імені Б. Хмельницького, 2017. 434 с. URL: <http://eprints.cdu.edu.ua/1481/1/pro.pdf> (Дата звернення: 02.07. 2022).
2. Корнієнко С.К. Проектування інформаційного забезпечення автоматизованих систем : навч. посіб. Запоріжжя: ЗНТУ, 2015. 210 с. URL: <http://eir.zntu.edu.ua/handle/123456789/835> (Дата звернення: 03.07. 2022).
3. Дегтярьова Л. М. Навчальний посібник з дисципліни «Технології розробки програмного забезпечення» для студентів спеціальності 123 «Комп'ютерна інженерія». Полтава: ПолтНТУ, 2017. 218 с. URL: <http://reposit.nupp.edu.ua/handle/PolNTU/4461> (Дата звернення: 03.07.2022).
4. Проектування інформаційних систем. Загальні питання теорії проектування ІС : конспект лекцій. уклад.: О. С. Коваленко, Л. М. Добровська. : КПІ імені Ігоря Сікорського, 2020. 192 с. URL: https://ela.kpi.ua/bitstream/123456789/33651/1/PIS_KL.pdf (Дата звернення: 12.07. 2022).
5. Табунщик Г. В. Проектування та моделювання програмного забезпечення сучасних інформаційних систем : підручник. Запоріжжя: Дике Поле, 2016. 250 с. URL: http://eir.zntu.edu.ua/bitstream/123456789/1824/1/Tabunshchik_Software_Design.pdf
6. Цюцюра С. В. Сучасні методології проектування та розробка програмного забезпечення : конспект лекцій К.: КНУБА, 2020. 131 с. URL: <http://www.knuba.edu.ua/ukr/wpcontent/uploads/2021/01/%D0%A1%D0%9C%D0%9F-%D1%82%D0%B0-%D0%A0%D0%9F%D0%97-%D0%9A%D0%BE%D0%BD%D1%81%D0%BF%D0%B5%D0%BA%D1%82-%D0%BB%D0%B5%D0%BA%D1%86%D1%96%D0%B9.pdf> (Дата звернення: 22.07.2022).