

**ЖИТОМИРСЬКИЙ ДЕРЖАВНИЙ УНІВЕРСИТЕТ ІМЕНІ ІВАНА ФРАНКА**  
**Фізико-математичний факультет**  
**Кафедра комп'ютерних наук та інформаційних технологій**

**МЕТОДИЧНІ ВКАЗІВКИ ДО ПРОХОДЖЕННЯ НАВЧАЛЬНОЇ ПРАКТИКИ**  
**З РОЗРОБКИ ЦИФРОВИХ ОСВІТНІХ РЕСУРСІВ**

**для підготовки здобувачів**  
**першого (бакалаврського) рівня вищої освіти**

<b>Галузь знань</b>	<i>А Освіта/Педагогіка</i>
<b>Спеціальність</b>	<i>А5 Професійна освіта</i>
<b>Спеціалізація</b>	<i>А5.39 Цифрові технології</i>
<b>Освітня програма</b>	<i>Професійна освіта (Цифрові технології)</i>
<b>Факультет</b>	<i>фізико-математичний</i>

Укладач: асистент Гуменюк Вікторія  
Розглянуто та схвалено  
на засіданні кафедри комп'ютерних наук та  
інформаційних технологій  
Протокол від «17» червня 2026 року № 22  
Завідувач кафедри \_\_\_\_\_ Олена УСАТА

Рекомендовано до друку Вченою радою  
Житомирського державного університету імені Івана Франка  
(протокол №12 від 26.06.2026 р.)

Рецензенти:

- Юлія МЕЛЬНИЧУК кандидат педагогічних наук, доцент, доцент кафедри цифрових освітніх технологій Луцького національного технічного університету.
- Майя КОВАЛЬЧУК кандидат педагогічних наук, доцент, завідувач кафедри комп'ютерних технологій та моделювання систем Поліського національного університету.
- Олена УСАТА кандидат педагогічних наук, доцент, завідувач кафедри комп'ютерних наук та інформаційних технологій Житомирського державного університету імені Івана Франка.

Методичні вказівки до проходження навчальної практики з розробки цифрових освітніх ресурсів для здобувачів першого (бакалаврського) рівня вищої освіти спеціальності А5 Професійна освіта спеціалізації А5.39 Цифрові технології освітньо-професійної програми Професійна освіта (Цифрові технології) / Укладач: В. Гуменюк. Житомир : Вид-во ЖДУ ім. І. Франка, 2026. 59 с.

Методичні вказівки розроблено для здобувачів першого (бакалаврського) рівня вищої освіти спеціальності А5 Професійна освіта спеціалізації А5.39 Цифрові технології освітньої програми Професійна освіта (Цифрові технології).

Видання містить теоретичні відомості, практичні та індивідуальні завдання до навчальної практики з розробки цифрових освітніх ресурсів для 4-го та 5-го семестрів, які узгоджені з програмою навчальної практики. Також матеріали включають критерії оцінювання роботи здобувачів вищої освіти під час практики, вимоги до захисту практики, підготовки презентацій, а також детальні вказівки та додатки щодо оформлення звітної документації.

© Гуменюк В.В., 2026  
© ЖДУ імені Івана Франка, 2026

## Зміст

Вступ.....	5
Критерії оцінювання роботи здобувачів вищої освіти під час практики.....	7
Завдання практики на 4-й семестр.....	8
Теоретичні відомості.....	8
Проектування логіки цифрового ресурсу й сценаріїв його використання.....	8
Візуалізація даних. Створення інтерактивних презентацій, інфографік та постерів. ....	9
Структурування інформації. Розробка ментальних карт та організація спільної роботи. ....	10
Гейміфікація. Розробка простих ігор та інтерактивних навчальних вправ (встановлення відповідностей, кросворди тощо). Створення навчальних квестів, вікторин та тестів для перевірки знань.....	11
Аудіопродукти. Основи запису, монтажу та обробки звуку. Створення освітніх подкастів. Навчальна анімація. Створення анімаційних роликів для пояснення процесів чи явищ.....	12
Відеопродукти. Запис екрана, основи відеомонтажу, створення коротких відеоуроків. ....	14
Електронні видання. Створення макетів електронних посібників з інтеграцією мультимедіа. ....	15
Вебагрегація. Створення простих HTML-сторінок/мінісайтів для об'єднання матеріалів.....	16
Практичне завдання №1.....	17
Практичне завдання №2.....	18
Практичне завдання №3.....	19
Індивідуальне завдання на практику .....	20
Орієнтовна тематика індивідуальних завдань .....	20
Порядок виконання індивідуального завдання.....	21
Захист практики та вимоги до звітної документації.....	26
Підготовка та презентація індивідуальних практичних завдань .....	26
Оформлення звітної документації .....	27
Завдання практики на 5-й семестр.....	28
Теоретичні відомості.....	28
Основи веброзробки для створення навчальних міні сайтів. Інтерактивні вебсторінки. ....	28
Створення комплексних електронних підручників та навчальних модулів. Мультимедійні посібники та їх інтеграція. ....	29
Вступ до розробки освітніх ігор. Інтерактивні текстові ігри та квести. Ігрові рушії в освітньому процесі.....	30
Технології віртуальної реальності. 3D-моделювання навчальних об'єктів.....	32
Проектування складної логіки ЦОР.....	33
Практичне завдання №1.....	34
Практичне завдання №2.....	37
Індивідуальне завдання на практику .....	43
Порядок виконання індивідуального завдання.....	43
Захист практики та вимоги до звітної документації.....	50

Підготовка та презентація індивідуальних практичних завдань .....	50
Оформлення звітної документації .....	50
Список використаних джерел та літератури.....	51
Додаток А .....	54

## ВСТУП

Підготовка сучасних компетентних фахівців та запит суспільства істотно підвищують рівень вимог до майбутніх бакалаврів з професійної освіти, змінюючи зміст і структуру завдань, які здобувач вищої освіти має вирішувати самостійно. Навчальна практика є невід'ємною складовою процесу підготовки фахівців у закладі вищої освіти та обов'язковим компонентом освітньо-професійної програми для підготовки фахівців першого (бакалаврського) рівня.

Навчальна практика з розробки цифрових освітніх ресурсів проходить у 4 та 5 семестрах.

Семестр	Загальний обсяг/ кредитів	Лабораторних годин	Самостійна робота	Тривалість (к-ть тижнів)
4	90/3	40	50	2
5	90/3	40	50	2

Базою для проведення даної практики визначено комп'ютерні лабораторії ЖДУ.

**Метою** навчальної практики є формування у здобувачів вищої освіти професійних умінь і навичок створення сучасних цифрових освітніх ресурсів, здатних підвищити ефективність навчального процесу в закладах професійної освіти. Це передбачає застосування сучасних технологій програмування, мультимедіа, гейміфікації, інтерактивного дизайну тощо.

Під час проходження навчальної практики здобувачі мають виконати такі основні **завдання**:

- Поглибити знання про цифрові освітні ресурси, інструменти їх розробки та способи інтеграції в освітнє середовище.
- Розвинути навички застосування інструментів мультимедійного та графічного дизайну, веброзробки та ігрових платформ.
- Формувати цифрову компетентність щодо створення електронних підручників, освітніх ігор, інтерактивних тренажерів та навчальних модулів.
- Удосконалити вміння працювати в команді, презентувати цифрові продукти, критично оцінювати їх дидактичну ефективність.
- Дотримуватись етичних, правових та технічних стандартів під час створення цифрових освітніх ресурсів.

За час проходження практики здобувач вищої освіти має набути таких **компетентностей**:

ЗК5. Здатність приймати обґрунтовані рішення.

ЗК6. Навички використання інформаційних і комунікаційних технологій.

ЗК7. Здатність вчитися і оволодівати сучасними знаннями.

ЗК 8. Здатність працювати в команді.

ЗК10. Здатність виявляти ініціативу та підприємливість.

СК5. Здатність використовувати сучасні інформаційні технології та спеціалізоване програмне забезпечення та інтегрувати їх в освітнє середовище.

СК8. Здатність використовувати відповідне програмне забезпечення для вирішення професійних завдань, відповідно до спеціалізації.

СК14. Здатність збирати, аналізувати та інтерпретувати інформацію (дані) відповідно до спеціалізації.

СК17. Здатність застосовувати комп'ютерні методи побудови зображень та графічної подачі візуального матеріалу макетно-модельними матеріалами і технологіями їх опрацювання.

СК18. Здатність до застосування вебтехнологій та інструментів про'єктування для вирішення спеціалізованих задач.

**Програмні результати:**

<b>Програмні результати</b>	<b>Методи навчання</b>	<b>Форми та методи оцінювання</b>
ПР8. Самостійно планувати й організувати власну професійну діяльність і діяльність здобувачів освіти і підлеглих.	Пояснювально-ілюстративний, аналітичний, проектний, частково-пошуковий, дослідницькі, моделювання.	Поточний контроль на заняттях, захист індивідуальних завдань практики, залік.
ПР9. Відшукувати, обробляти, аналізувати та оцінювати інформацію, що стосується професійної діяльності, користуватися спеціалізованим програмним забезпеченням та сучасними засобами зберігання та обробки інформації.	Пояснювально-ілюстративний, аналітичний, проектний, частково-пошуковий, дослідницькі, моделювання.	Поточний контроль на заняттях, захист індивідуальних завдань практики, залік.
ПР27 Застосовувати цифрові технології, технічні засоби навчання в управлінні й організації освітнього процесу закладу освіти та науково-дослідній роботі.	Пояснювально-ілюстративний, аналітичний, проектний, частково-пошуковий, дослідницькі, моделювання.	Поточний контроль на заняттях, захист індивідуальних завдань практики, залік.
ПР29. Використовувати методи, засоби, технологічні рішення у розробці вебзастосунків, у тому числі навчального призначення.	Пояснювально-ілюстративний, аналітичний, проектний, частково-пошуковий, дослідницькі, моделювання.	Поточний контроль на заняттях, захист індивідуальних завдань практики, залік.
ПР30. Застосовувати методи та алгоритми комп'ютерної графіки у процесі розробки графічних застосунків, проектувати та створювати системи мультимедіа і графічного моделювання.	Пояснювально-ілюстративний, аналітичний, проектний, частково-пошуковий, дослідницькі, моделювання.	Поточний контроль на заняттях, захист індивідуальних завдань практики, залік.

## КРИТЕРІЇ ОЦІНЮВАННЯ РОБОТИ ЗДОБУВАЧІВ ВИЩОЇ ОСВІТИ ПІД ЧАС ПРАКТИКИ

### Загальні принципи оцінювання

- Під час навчальної практики, що проводиться в університеті, здобувачі отримують оцінки за кожне заняття за 100-бальною шкалою.
- Підсумкова оцінка з навчальної практики виставляється за 100-бальною шкалою на підставі оцінок за кожне окреме навчальне заняття та визначається як їх середньоарифметичне значення.
- Під час проходження практики відбувається захист індивідуальних практичних завдань, а здобувачі вищої освіти надають на кафедру звітну документацію.
- Результатом проходження практики є залік.

### Розподіл підсумкової оцінки за видами діяльності

При виставленні підсумкової оцінки за результатами навчальної практики враховується такий розподіл у відсотках

№	Вид діяльності здобувача вищої освіти	Розподіл підсумкової оцінки.
1.	Виконання завдань практики	30%
2.	Виконання індивідуальних завдань	50%
3.	Оформлення звітної документації	15%
4.	Своєчасність подачі звітної документації	5%
Всього:		100%

### Шкала оцінювання результатів навчальної практики здобувачів вищої освіти

Оцінка за університетською шкалою	100-бальна шкала оцінювання	Оцінка ЄКТС
<i>Зараховано</i>	90 – 100	A
	82 – 89	B
	74 – 81	C
	64 – 73	D
	60 – 63	E
<i>Незараховано</i>	35 – 59	FX
	1 – 34	F

## ЗАВДАННЯ ПРАКТИКИ НА 4-Й СЕМЕСТР

*Загальний обсяг: 90 годин, з яких 40 годин аудиторної та 50 годин самостійної роботи.*

У цьому семестрі здобувачі зосереджуються на опануванні базових інструментів для створення візуального, аудіовізуального контенту та простих інтерактивних форм.

- Проектування логіки ресурсу, базових сценаріїв використання та простої інтерактивної структури.
- Створення інтерактивних презентацій, інфографік, ментальних карт та постерів за допомогою таких сервісів як Canva, Genially, Migo.
- Розробка простих ігор, навчальних вправ, квестів та тестів з використанням LearningApps, Wordwall, Quizizz, Google Forms, Kahoot.
- Створення аудіовізуальних продуктів, зокрема подкастів, відеоуроків та анімацій у середовищах Audacity, Clipchamp, Powtoon, Animaker.
- Розробка макетів електронних посібників та простих HTML-сторінок за допомогою Canva, Book Creator, Google Sites.

Навчальну практику розбито на дві логічні частини: вивчення інструментів (заняття 1-10) та створення комплексного індивідуального проєкту (заняття 11-20).

У першій частині здобувачі вивчають функціонал платформ, а у другій частині здобувачі працюють над власною темою з фахової дисципліни, розробляючи для неї повноцінний комплекс ЦОР.

### ТЕОРЕТИЧНІ ВІДОМОСТІ

#### **Проектування логіки цифрового ресурсу й сценаріїв його використання.**

Створення ефективного цифрового освітнього ресурсу (ЦОР) розпочинається не з безпосередньої програмної розробки чи дизайну, а з етапу проектування. Цей етап є фундаментом, який визначає, наскільки зручним, зрозумілим та дидактично доцільним буде кінцевий продукт для здобувачів освіти. Проектування охоплює розробку логіки ресурсу, його інтерактивної структури та визначення сценаріїв використання.

##### *Проектування логіки цифрового ресурсу*

Логіка ЦОР — це внутрішня архітектура продукту, яка визначає алгоритми та взаємозв'язки між усіма його елементами (інформаційними блоками, мультимедіа, тестовими завданнями, ігровими механіками). Проектування логіки передбачає відповіді на питання: як навчальний матеріал буде поділено на порції (модулі, рівні, екрани), в якій послідовності користувач отримуватиме інформацію та за яких умов він зможе перейти до наступного етапу.

В освітніх ресурсах логіка тісно пов'язана з педагогічним дизайном (Instructional Design). Наприклад, логіка навчального тренажера може передбачати, що доступ до практичного завдання відкривається лише після перегляду теоретичного відеоролика (жорстка, послідовна логіка). Натомість електронний довідник чи мікросайт дозволяє користувачеві вільно переміщуватися та самостійно обирати розділи для вивчення (гнучка логіка).

##### *Інтерактивна структура ресурсу*

Структура визначає спосіб організації контенту та навігацію. Під час проектування ЦОР найчастіше використовують такі типи інтерактивних структур:

- Лінійна. Користувач рухається послідовно від першого кроку до останнього без можливості зміни маршруту (наприклад, відеоурок, базова презентація, подкаст).
- Ієрархічна (деревоподібна). Наявність головного меню, з якого користувач переходить до підрозділів і може в будь-який момент повернутися назад (наприклад, освітній мінісайт, електронний підручник з навігаційним меню).
- Розгалужена (нелінійна). Подальший шлях користувача залежить від його вибору або результатів попередніх дій (наприклад, навчальний квест або тестування з адаптивним рівнем складності, де у разі помилки користувача спрямовують на повторення певного правила).

Для візуалізації структури на етапі проектування найкраще використовувати інструменти для створення ментальних карт (Mind Maps) або блок-схем.

## Сценарії використання (Use Cases) та шлях користувача (User Flow)

Сценарій використання — це детальний опис того, як користувач взаємодіє з цифровим ресурсом для досягнення конкретної навчальної мети. Проектування сценаріїв дозволяє розробнику поглянути на продукт очима користувача та уникнути логічних помилок в інтерфейсі (UI) і користувацькому досвіді (UX).

Якісний сценарій використання враховує такі аспекти:

Точка входу	з чого починається взаємодія (екран привітання, інструкція до гри, авторизація)
Основний шлях	ідеальна послідовність дій, за якої користувач безпомилково проходить навчальний матеріал і успішно виконує підсумкове завдання
Альтернативні шляхи та обробка помилок	що відбувається, якщо користувач дає неправильну відповідь, не розуміє завдання або натискає не ту кнопку. Розробник має передбачити наявність підказок, можливості повернутися на крок назад та обов'язкового зворотного зв'язку (фідбеку) на кожну дію

### Етапи проектування ЦОР

Процес проектування доцільно розбити на такі послідовні кроки:

1. Аналіз аудиторії та формулювання дидактичної мети (що саме ресурс має навчити, які компетентності сформулювати).
2. Збір та структурування контенту (написання текстів, підбір зображень, аудіо, відео).
3. Розробка блок-схеми або ментальної карти (візуалізація логіки та навігаційних зв'язків між елементами).
4. Створення сценаріїв взаємодії (прописування алгоритмів роботи кнопок, тригерів та інтерактивних елементів).
5. Створення прототипу (вайрфрейму) — схематичного чорнового зображення екранів майбутнього ресурсу для перевірки зручності розміщення елементів до початку роботи над візуальним дизайном.

Грамотне проектування на початковому етапі дозволяє уникнути логічних хиб, суттєво економить час під час безпосередньої розробки ЦОР у програмному середовищі та гарантує високу дидактичну ефективність готового освітнього продукту.

## Візуалізація даних. Створення інтерактивних презентацій, інфографік та постерів.

### Поняття візуалізації даних та її роль в освіті

Візуалізація даних — це процес перетворення складної, абстрактної або об'ємної інформації у графічний формат (схеми, графіки, діаграми, інфографіки) для її швидкого та ефективного сприйняття.

У контексті створення цифрових освітніх ресурсів (ЦОР) візуалізація виконує ключову роль у зниженні когнітивного навантаження на здобувачів освіти. Зважаючи на те, що мозок людини обробляє візуальну інформацію значно швидше за текстову, грамотна візуалізація дозволяє фокусувати увагу на головному та виявляти приховані закономірності. Наприклад, пояснення етапів графічного конвєсера або процесу багатопрхідного рендерингу сприймається здобувачами набагато ефективніше, якщо замість суцільного тексту використати структурну інфографіку з чіткою послідовністю.

### Види візуальних освітніх матеріалів

Під час навчальної практики здобувачі мають опанувати створення таких візуальних продуктів:

- Навчальна інфографіка. Графічне подання складної інформації, даних або знань. Вона може бути статистичною (діаграми, графіки), хронологічною (таймлайни для відображення історичних подій), процесуальною (покрокові інструкції) або порівняльною.
- Освітній постер. Статичний візуальний інструмент, що містить стислий виклад ключової інформації з певної теми. Постери часто використовуються як опорні конспекти або пам'ятки (наприклад, правила безпеки, формули, ключові терміни).
- Інтерактивна презентація. На відміну від класичних лінійних слайдів, інтерактивна презентація містить елементи нелінійної навігації (кнопки, гіперпосилання,

спливаючі вікна, вбудовані аудіо- та відеоматеріали). Це перетворює її зі звичайного засобу супроводу лекції на повноцінний інструмент для самостійного вивчення матеріалу.

#### *Базові принципи дизайну ЦОР*

Для того, щоб створені візуальні матеріали були дидактично доцільними та естетично привабливими, необхідно дотримуватись кількох фундаментальних правил дизайну:

- Ієрархія та композиція. Найважливіша інформація має виділятися розміром, кольором або розташуванням. Елементи слід групувати за змістом, залишаючи достатньо "повітря" (вільного простору), щоб уникнути візуального шуму.

- Типографіка. Для освітніх ресурсів рекомендується використовувати не більше 2-3 шрифтів (наприклад, один акцидентний для заголовків та один класичний без зарубок для основного тексту). Текст має бути контрастним щодо фону і легко читатися з будь-якого пристрою.

- Колористика. Кольори повинні мати не лише естетичну, а й смислову функцію (наприклад, виділення акцентів). Бажано використовувати єдину кольорову палітру (3-4 кольори), уникаючи надто яскравих, "кричущих" поєднань, які швидко втомлюють зір.

#### *Інструменти для створення візуальних ЦОР*

Для виконання завдань з візуалізації даних, зокрема створення інтерактивних презентацій, інфографік, ментальних карт та постерів, використовуються такі сучасні хмарні сервіси, як Canva, Genially, Miro.

Canva є універсальним графічним редактором із великою бібліотекою готових шаблонів, що дозволяє швидко генерувати якісні постери та інфографіки.

Genially спеціалізується на інтерактивному контенті, надаючи потужні інструменти для створення презентацій із розгалуженою навігацією, анімованими елементами та інтерактивними мітками (спливаючими вікнами з додатковим текстом чи медіа).

### **Структурування інформації. Розробка ментальних карт та організація спільної роботи.**

#### *Структурування інформації та сутність ментальних карт*

Структурування інформації — це процес організації розрізнених даних у логічну, взаємопов'язану систему для полегшення їх сприйняття, аналізу та запам'ятовування. В епоху інформаційного перевантаження вміння правильно структурувати матеріал є ключовим для створення ефективних цифрових освітніх ресурсів.

Одним із найпопулярніших інструментів структурування є ментальні карти (Mind Maps, карти думок). Це візуальні ієрархічні схеми, що відображають зв'язки між концептами, ідеями, процесами або завданнями. Класична ментальна карта будується навколо єдиної центральної теми (ідеї), від якої відходять гілки з ключовими підтемами, що, своєю чергою, можуть розгалужуватися на ще дрібніші деталі.

#### *Дидактичний потенціал ментальних карт*

В освітньому процесі ментальні карти виконують кілька важливих функцій:

- Систематизація знань (допомагають здобувачам побачити загальну картину (макроструктуру) теми та зрозуміти логічні зв'язки між її окремими елементами).

- Проєктування ресурсів (ментальні карти є незамінними під час розробки нових продуктів. Вони використовуються для проєктування логіки ресурсу, сценаріїв використання та загальної інтерактивної структури).

- Опорні конспекти (візуалізована таким чином інформація легше відтворюється в пам'яті під час підготовки до контрольних заходів).

#### *Організація спільної роботи та командна взаємодія*

Створення сучасних цифрових освітніх продуктів найчастіше є не індивідуальним, а командним процесом. Одним із завдань навчальної практики є удосконалення вміння здобувачів працювати в команді.

Спільна робота (Collaboration) у цифровому середовищі передбачає одночасну взаємодію кількох розробників над одним проєктом у реальному часі. Це вимагає від здобувачів здатності використовувати сучасні інформаційні технології та інтегрувати їх в освітнє середовище. Сучасні платформи для спільної роботи забезпечують можливості для

проведення "мозкових штурмів" (Brainstorming), делегування завдань, спільного редагування схем та миттєвого обміну коментарями.

#### *Інструменти для створення ментальних карт*

Для візуального структурування інформації та організації командної взаємодії використовують спеціалізовані хмарні сервіси. Зміст діяльності здобувачів вищої освіти на навчальній практиці передбачає створення ментальних карт за допомогою сервісу Miro (або аналогічних інструментів, таких як Canva чи Genially).

Miro — це нескінченна віртуальна онлайн-дошка, яка ідеально підходить для спільної роботи команди. Вона дозволяє створювати складні ментальні карти, блок-схеми, додавати стікери (нотатки), медіафайли та візуалізувати будь-які навчальні чи проєктні процеси. Кожен учасник команди має власний курсор і може вносити зміни, які миттєво відображаються у всіх інших користувачів, що робить цей інструмент еталонним для віддаленої командної роботи.

### **Гейміфікація. Розробка простих ігор та інтерактивних навчальних вправ (встановлення відповідностей, кросворди тощо). Створення навчальних квестів, вікторин та тестів для перевірки знань.**

Гейміфікація (ігрофікація) — це застосування ігрових механік, елементів дизайну та ігрового мислення в неігрових ситуаціях (зокрема, у навчанні) для підвищення мотивації, залученості та ефективності засвоєння матеріалу.

Застосування технологій гейміфікації є одним із ключових напрямів формування у здобувачів умінь і навичок створення сучасних цифрових освітніх ресурсів (ЦОР), здатних підвищити ефективність навчального процесу. Гейміфікація не означає створення повноцінної розважальної гри; її суть полягає у додаванні ігрових елементів (балів, рівнів, таймерів, таблиць лідерів, нагород, миттєвого зворотного зв'язку) до звичайних навчальних завдань, що робить рутинну роботу захопливою.

#### *Дидактичні особливості інтерактивних навчальних вправ*

Формування цифрової компетентності здобувачів охоплює створення освітніх ігор та інтерактивних тренажерів. Прості інтерактивні вправи є базовим рівнем гейміфікації. До найпоширеніших типів таких вправ належать:

- Встановлення відповідностей (Matching). Завдання, де необхідно з'єднати пов'язані елементи (наприклад, поняття та його визначення, дату та подію, інструмент та його функцію). Це чудово розвиває логічне мислення.
- Кросворди, філворди та анаграми. Ефективні інструменти для запам'ятовування фахової термінології та перевірки засвоєння нових понять.
- Сортування та класифікація. Вправи, що вимагають розподілу об'єктів за групами (наприклад, правильні/неправильні твердження).
- Заповнення пропусків. Завдання на відновлення цілісності тексту, коду або формули.

Ці інструменти дозволяють реалізувати принцип миттєвого зворотного зв'язку: здобувач одразу бачить свої помилки та має можливість їх виправити, що знижує страх перед оцінюванням.

#### *Платформи для розробки простих ігор та вправ*

Зміст діяльності на навчальній практиці передбачає розробку простих ігор та вправ із використанням таких цифрових інструментів, як Scratch, LearningApps, Wordwall та Quizizz. Ці сервіси не вимагають глибоких знань із програмування та базуються на використанні готових шаблонів.

- LearningApps – один із найпопулярніших безкоштовних сервісів для створення інтерактивних навчальних модулів. Він пропонує десятки готових шаблонів (кросворди, пазли, класифікація, вікторини), які легко наповнити власним фаховим контентом.
- Wordwall – потужна платформа для створення як цифрових, так і друкованих дидактичних матеріалів. Її головна перевага — можливість змінити ігрову механіку в один клік. Наприклад, ввівши список запитань, розробник може представити його у вигляді класичного тесту, ігрового лабіринту або вікторини в стилі телешоу.

- Scratch – візуальне середовище блочного програмування, яке дозволяє створювати власні анімовані історії та розробляти інтерактивні ігри з унікальною логікою та сценаріями.
- Quizizz – сервіс, що фокусується на проведенні вікторин із яскраво вираженими елементами гейміфікації (аватари, бонуси за серію правильних відповідей, меми при правильних/неправильних відповідях).

#### *Гейміфікація у системі контролю знань*

Класичний контроль знань часто викликає у здобувачів освіти стрес та зниження мотивації. Застосування технологій гейміфікації під час оцінювання дозволяє перетворити рутинне тестування на захопливий процес змагання або дослідження. Інтеграція ігрових елементів (нарахування балів за швидкість, таблиці лідерів, командні змагання, нагороди) сприяє підвищенню ефективності навчального процесу та залученості аудиторії.

#### *Навчальні квести (Escape Rooms / Breakout Games)*

Навчальний квест — це інтерактивна ігрова форма роботи, що передбачає вирішення серії головоломок, завдань або тестів для досягнення певної мети (наприклад, "вибратися з кімнати" або "розшифрувати код").

Квести базуються на нелінійній (розгалуженій) структурі, де перехід до наступного етапу можливий лише після правильного виконання попереднього завдання. Вони чудово розвивають критичне мислення та вміння працювати з інформацією. Створення навчальних квестів є одним з основних напрямів діяльності здобувачів на навчальній практиці з розробки цифрових освітніх ресурсів.

#### *Вікторини та інтерактивні тести*

На відміну від квестів, вікторини зазвичай мають лінійну структуру та спрямовані на швидку перевірку фактів, термінології чи розуміння концепцій. Інтерактивні тести дозволяють викладачу здійснювати формувальне оцінювання в режимі реального часу, миттєво виявляючи прогалини у знаннях аудиторії, а здобувачам — відразу бачити свої результати та правильні відповіді.

#### *Платформи для створення тестів та квестів*

Для створення навчальних квестів та тестів під час практики використовуються такі інструменти: Google Forms, Kahoot, Genially.

- Google Forms (Google Форми). Хоча цей інструмент найчастіше використовується для класичного тестування або опитувань, його функціонал «переходу до розділу на основі відповіді» дозволяє створювати повноцінні розгалужені текстові квести. Крім того, інструмент дозволяє налаштувати автоматичне оцінювання та надання зворотного зв'язку (фідбеку) на кожну відповідь.

- Kahoot! Одна з найпопулярніших платформ для створення вікторин (квізів), орієнтована на синхронну роботу в аудиторії (або онлайн). Вона використовує потужні механіки гейміфікації: яскравий інтерфейс, динамічну музику, нарахування балів за швидкість та точність, а також виведення турнірної таблиці на екран після кожного запитання.

- Genially. Потужний сервіс для створення візуальних освітніх продуктів, який пропонує готові шаблони у категорії «Гейміфікація» (Gamification). За допомогою Genially можна створювати складні візуальні квести (Escape Rooms) з клікабельними елементами, прихованими підказками та інтерактивними завданнями, які органічно інтегруються у графічний сюжет гри.

**Аудіопродукти. Основи запису, монтажу та обробки звуку. Створення освітніх подкастів. Навчальна анімація. Створення анімаційних роликів для пояснення процесів чи явищ.**

#### *Роль аудіопродуктів у сучасному освітньому процесі*

Сучасні цифрові освітні ресурси охоплюють широкий спектр мультимедійних форматів, серед яких аудіопродукти займають особливе місце. Використання звуку в навчанні дозволяє урізноманітнити подачу матеріалу, знизити зорове навантаження на здобувачів та адаптувати освітній процес для людей, які найкраще сприймають інформацію на слух.

Зміст діяльності здобувачів вищої освіти на навчальній практиці з розробки цифрових освітніх ресурсів безпосередньо передбачає створення аудіовізуальних продуктів, до яких належать подкасти, відеоуроки та анімації.

#### *Освітній подкаст як цифровий ресурс*

Освітній подкаст — це серія цифрових аудіофайлів навчального або науково-популярного характеру, які користувач може слухати онлайн або завантажити на власний пристрій.

Переваги подкастів як цифрових освітніх ресурсів:

- Мобільність (можливість слухати матеріал під час прогулянки, поїздки чи виконання інших справ, що розширює межі освітнього середовища).
- Ефект присутності (голос викладача створює відчуття особистого спілкування, що позитивно впливає на мотивацію).
- Доступність (легкість у поширенні (через освітні мікросайти або електронні посібники) та відтворенні на будь-яких пристроях).

#### *Основи запису, монтажу та обробки звуку*

Створення якісного аудіопродукту вимагає розуміння базових принципів роботи зі звуком і складається з таких етапів:

1. Підготовка та запис. Написання сценарію (тексту), вибір приміщення з мінімальною луною (ехо), використання зовнішнього мікрофона або якісної гарнітури. Запис дикторського голосу.

2. Монтаж (редагування). Видалення невдалих дублів, тривалих пауз, обмовок або шумів дихання. Компонування різних фрагментів в єдину аудіодоріжку.

3. Обробка звуку. Застосування спеціальних ефектів для покращення якості звучання. Базові інструменти обробки включають:

- *Шумозаглушення* — видалення рівномірного фонового шуму (гудіння комп'ютера, шум вентилятора).
- *Еквалізація* — коригування частот для надання голосу більшої глибини або чіткості.
- *Нормалізація та компресія* — вирівнювання гучності, щоб тихі слова були розбірливими, а гучні не викликали дискомфорту.

4. Зведення (мікшування). Накладання фонової музики (з обов'язковим дотриманням авторських прав — використання треків з ліцензією Creative Commons) та звукових ефектів для створення відповідного емоційного фону.

#### *Програмне забезпечення для створення аудіопродуктів*

Для запису та обробки звукових файлів під час створення подкастів використовується спеціалізоване програмне забезпечення. Згідно з програмою навчальної практики, базовим інструментом для розробки аудіовізуальних продуктів визначено програму Audacity.

Audacity — це безкоштовний, багатоплатформний аудіоредактор із відкритим вихідним кодом. Він дозволяє записувати звук наживо, здійснювати багатодоріжковий монтаж, вирізати та склеювати фрагменти, а також містить велику бібліотеку вбудованих ефектів для професійної обробки звуку.

#### *Дидактичний потенціал навчальної анімації*

Навчальна анімація — це візуалізація динамічних процесів, явищ або концепцій шляхом швидкої зміни кадрів (створення ілюзії руху), яка супроводжується текстовим, звуковим або музичним поясненням.

Зміст діяльності здобувачів вищої освіти на навчальній практиці передбачає розробку різних аудіовізуальних продуктів, серед яких окреме місце займає створення анімацій. На відміну від звичайного відео (скринкасту чи запису викладача), анімація має унікальні дидактичні переваги:

- Візуалізація невидимого дозволяє наочно показати процеси, які неможливо побачити в реальному житті (наприклад, рух електронів, історичні битви, макроекономічні моделі).
- Управління часом — здатність уповільнити надшвидкі процеси або прискорити надто повільні для їх кращого розуміння.

- Абстрагування – можливість прибрати зайві фонові деталі, які відвертають увагу у звичайному відео, і сфокусувати увагу здобувачів виключно на головному об'єкті чи алгоритмі.

#### *Етапи створення анімаційного ролика та сторіборд*

Процес створення анімації значно відрізняється від класичного відеомонтажу і потребує ретельного попереднього проєктування. Основні етапи:

1. Формулювання ідеї та написання сценарію. Визначення дидактичної мети, цільової аудиторії та ключового повідомлення ролика.

2. Створення сторіборда (розкадрування). Це візуальний план майбутньої анімації у вигляді серії ескізів (подібно до коміксу). Сторіборд визначає, що саме буде відбуватися в кадри на кожній секунді, які персонажі чи об'єкти з'являться та який текст (або голос диктора) супроводжуватиме цю сцену.

3. Вибір графічного стилю та озвучування. Підбір фонів, персонажів, інфографіки та запис дикторського голосу (або додавання фонові музики).

4. Анімування (програмна реалізація). Налаштування рухів об'єктів на монтажній шкалі (таймлайні).

#### *Принципи мультимедійного навчання в анімації*

Щоб анімація не перетворилася на розважальний мультфільм, а виконувала саме навчальну функцію, слід дотримуватися принципів когнітивної теорії мультимедійного навчання (за Р. Майером):

- Принцип надмірності (краще використовувати анімацію + голос диктора, ніж анімацію + голос + дублювання цього ж тексту на екрані (щоб не перевантажувати зоровий канал)).

- Принцип просторової суміжності (текстові підписи мають з'являтися безпосередньо поруч із відповідними об'єктами, а не в іншому кутку екрана).

- Принцип сегментації (складну анімацію краще розбити на короткі логічні частини з можливістю зупинки або повтору).

#### *Хмарні сервіси для створення навчальної анімації*

Сучасні хмарні платформи дозволяють створювати якісні анімаційні ролики без навичок професійного художника-аніматора. Для виконання завдань практики застосовуються такі сервіси, як Powtoon та Animaker.

- Powtoon. Популярний сервіс для створення відеопрезентацій та анімаційних роликів. Пропонує велику бібліотеку готових сцен, персонажів і переходів. Його інтерфейс нагадує розширену версію PowerPoint із додаванням таймлайну, що робить його інтуїтивно зрозумілим для освітян.

- Animaker. Потужна платформа, яка фокусується на 2D-анімації персонажів. Дозволяє налаштувати міміку (lip-sync — синхронізацію руху губ із записаним голосом), емоції та складні траєкторії руху об'єктів.

### **Відеопродукти. Запис екрана, основи відеомонтажу, створення коротких відеоуроків.**

#### *Навчальне відео як цифровий освітній ресурс*

У сучасному освітньому середовищі відеоконтент є одним із найефективніших засобів передачі інформації. Відео поєднує в собі візуальні (динамічне зображення, текст, графіка) та аудіальні (голос диктора, музика) канали сприйняття, що дозволяє пояснювати складні процеси, алгоритми чи явища значно доступніше, ніж за допомогою статичного тексту.

Зміст діяльності здобувачів вищої освіти на навчальній практиці безпосередньо охоплює створення аудіовізуальних продуктів, зокрема відеоуроків. Розробка такого ресурсу вимагає не лише технічних навичок, але й здатності збирати, аналізувати та інтерпретувати інформацію відповідно до спеціалізації.

#### *Формати навчальних відеоуроків*

Для вирішення різних дидактичних завдань використовуються різні формати відео:

- Скринкаст (Screencast) або запис екрана. Відеозапис того, що відбувається на екрані комп'ютера користувача, який зазвичай супроводжується голосовими коментарями. Це

ідеальний формат для створення інструкцій щодо роботи з програмним забезпеченням або демонстрації алгоритмів розв'язання задач.

- «Голова, що говорить» (Talking Head). Класичний формат, де викладач знаходиться в кадрі та звертається безпосередньо до аудиторії. Створює ефект особистої присутності та добре підходить для мотиваційних вступів чи пояснення теоретичних концепцій.

- Відеопрезентація. Поєднання слайдів презентації з голосовим супроводом та, за потреби, невеликим вікном із зображенням викладача з вебкамери.

### 3. Етапи створення відеоуроку та основи монтажу

Процес розробки якісного відеоуроку складається з трьох ключових етапів:

1. Препродакшн (Підготовка). Написання детального сценарію. Текст для відео має бути лаконічним, розмовним та чітко структурованим. Важливо заздалегідь підготувати всі візуальні матеріали (презентації, зображення), які будуть демонструватися на екрані.

2. Продакшн (Зйомка/Запис). Безпосередній запис відео з екрана та/або вебкамери разом із записом голосу. Для забезпечення якості важливо слідкувати за освітленням (якщо використовується камера) та відсутністю сторонніх шумів.

3. Постпродакшн (Монтаж). Обробка відзнятого матеріалу у відеоредакторі. Базові операції відеомонтажу включають:

- Обрізання (Trimming) та Розділення (Splitting) - видалення невдалих дублів, пауз, обмовок або зайвих фрагментів на початку та в кінці відео.

- Кадрування (Cropping) - зміна масштабу зображення, щоб сфокусувати увагу на важливій частині екрана.

- Накладання (Overlay) - додавання текстових плашок із термінами, вбудовування додаткових зображень або графіки поверх основного відео.

- Переходи (Transitions) - візуальні ефекти, що згладжують зміну різних фрагментів відео (наприклад, плавне розчинення).

*Інструменти для створення відеоуроків*

Для створення відеоуроків та монтажу під час практики використовується сервіс Clipchamp (поряд з іншими інструментами для аудіовізуальних продуктів).

Clipchamp — це сучасний, інтуїтивно зрозумілий відеоредактор (доступний як вебдодаток та як вбудована програма у Windows), що поєднує в собі функції запису та багатодоріжкового монтажу. Він дозволяє одночасно записувати екран і вебкамеру, легко розрізати відео на таймлайні, додавати текст, застосовувати фільтри та експортувати готовий продукт у високій якості.

## **Електронні видання. Створення макетів електронних посібників з інтеграцією мультимедіа.**

*Електронні видання як комплексні цифрові ресурси*

Електронний посібник (або підручник) — це комплексний цифровий освітній ресурс, що містить систематизований навчальний матеріал і забезпечує інтерактивну взаємодію користувача з контентом.

На відміну від звичайних текстових документів чи сканованих копій друкованих книг (наприклад, у форматі PDF), сучасні електронні видання мають нелінійну навігацію та підтримують інтеграцію мультимедійних об'єктів (фото, відео, звук, інтерактивні тести). Формування цифрової компетентності щодо створення таких електронних підручників та навчальних модулів є одним із ключових завдань навчальної практики.

Електронний посібник часто виступає своєрідним "контейнером", у якому агрегуються цифрові продукти, розроблені на попередніх етапах (інфографіки, подкасти, відеоуроки, посилання на ігрові вправи).

*Принципи проєктування та макетування електронних посібників*

Створення електронного видання розпочинається з розробки його макета — візуальної схеми розміщення текстових, графічних та навігаційних елементів на сторінці (екрані). Під час макетування слід застосовувати комп'ютерні методи побудови зображень та графічної подачі візуального матеріалу, дотримуючись таких правил:

- Адаптивність та екранне читання. Текст на екрані сприймається на 25% повільніше, ніж на папері. Тому сторінки не повинні бути перевантажені суцільним текстом. Обов'язково використання абзаців, маркованих списків, підзаголовків та вільного простору ("повітря").

- Типографіка. Для основного тексту в електронних виданнях рекомендується використовувати шрифти без зарубок (Sans Serif), оскільки вони краще відображаються на екранах із різною роздільною здатністю.

- Інтерактивна навігація. Посібник повинен мати клікабельний зміст (гіперпосилання), кнопки повернення на головну сторінку та зручний перехід між розділами.

#### *Інтеграція мультимедіа*

Головна перевага макета електронного посібника — можливість збагачення тексту медіафайлами. Зміст діяльності здобувачів вищої освіти на практиці передбачає розробку електронних посібників із фото, відео та звуком.

- Візуальний супровід. Інтеграція 3D-моделей, схем та графіків для наочного пояснення.

- Аудіовізуальний контент. Вбудовування відеоуроків та подкастів безпосередньо на сторінку підручника, що дозволяє здобувачу не переходити на сторонні ресурси.

- Контрольний блок. Додавання вбудованих інтерактивних модулів (наприклад, створених у H5P) для самоперевірки після кожного розділу.

#### *Інструменти для створення електронних посібників*

Для розробки макетів електронних посібників та підручників під час практики використовуються такі інструменти, як Canva та Book Creator.

- Book Creator. Спеціалізований хмарний сервіс для створення інтерактивних електронних книг. Він дозволяє легко комбінувати текст, зображення, аудіо- та відеофайли на сторінках віртуальної книги з ефектом перегортання сторінок. Інтерфейс інтуїтивно зрозумілий, що робить його ідеальним для освітян.

- Canva. Цей графічний редактор дозволяє створювати високоякісні візуальні макети електронних посібників, використовуючи готові шаблони. Хоча він більше орієнтований на візуальну подачу (дизайн сторінок), Canva підтримує вбудовування гіперпосилань, відео та базову анімацію, що дозволяє експортувати готовий продукт як інтерактивний PDF-файл або мікросайт.

### **Вебагрегація. Створення простих HTML-сторінок/мінісайтів для об'єднання матеріалів.**

#### *Поняття вебагрегації в освітньому процесі*

Вебагрегація — це процес збору, систематизації та об'єднання різноманітних інформаційних і мультимедійних ресурсів у єдиному цифровому середовищі (на одній вебсторінці або сайті).

Під час розробки цифрових освітніх ресурсів виникає потреба структурувати розрізнені матеріали (інфографіки, відеоуроки, подкасти, інтерактивні ігри та тести) у єдиний логічний комплекс. Найкращим рішенням для цього є створення освітнього мікросайту. Такий підхід дозволяє здобувачу освіти отримати єдину "точку входу" до навчального матеріалу, де всі елементи пов'язані єдиною навігацією та сценарієм використання.

#### *Дидактичне значення освітніх мінісайтів*

Формування здатності до застосування вебтехнологій та інструментів проєктування для вирішення спеціалізованих задач є важливою складовою професійної підготовки здобувачів. Створення мікросайтів дозволяє:

- Організувати нелінійну траєкторію навчання (здобувач може вільно переміщуватися між модулями сайту).

- Використовувати методи, засоби та технологічні рішення у розробці вебзастосунків навчального призначення.

- Застосовувати цифрові технології в управлінні й організації освітнього процесу.

#### *Інструменти для веброзробки та агрегації матеріалів*

Зміст діяльності на навчальній практиці передбачає веброзробку навчальних мінісайтів, тренажерів, інтерактивних сторінок із застосуванням базових вебтехнологій (HTML, CSS,

JavaScript). Однак для швидкої агрегації матеріалів (на початковому рівні) найчастіше використовують візуальні конструктори:

- Google Sites (Google Сайти). Інтуїтивно зрозумілий хмарний сервіс, який дозволяє створювати прості HTML-сторінки та макети електронних посібників без написання програмного коду. Платформа ідеально підходить для вебагрегації, оскільки підтримує безшовне вбудовування (Embed) відео, аудіо, презентацій та інтерактивних вправ (наприклад, з LearningApps або Wordwall) безпосередньо на сторінку мікросайту.

## Практичне завдання №1

### Проектування логіки та сценарію використання цифрового освітнього ресурсу

**Інструменти:** Miro (або інший сервіс для створення ментальних карт/блок-схем, наприклад, diagrams.net), текстовий редактор.

#### Порядок виконання завдання:

##### 1. Вибір теми та формулювання мети

1. Оберіть тему з запропонованих нижче. Цей список є орієнтовним та не є вичерпним. Здобувачі вищої освіти можуть запропонувати власну тему в межах своєї фахової дисципліни. Обрана тематика буде застосовуватися до завдань 1-3.

2. Сформулюйте цільову аудиторію (вік, рівень підготовки) та головну дидактичну мету (наприклад: «засвоїти нові терміни», «відпрацювати алгоритм дій», «перевірити знання з теми»).

##### 2. Розробка інтерактивної структури (Ментальна карта / Блок-схема)

1. Зареєструйтеся або увійдіть у сервіс Miro (або аналогічний).

2. Створіть нову дошку та спроектуйте структуру вашого майбутнього ресурсу у вигляді блок-схеми.

3. Ваша схема повинна містити:

- Точку входу: стартовий екран (назва, кнопка «Почати», коротка інструкція).
- Теоретичний блок: екрани з інформацією (текст, відео, аудіо).
- Практичний/Контрольний блок: мінімально 2-3 завдання або запитання.
- Точку виходу: екран результатів та зворотний зв'язок (фідбек).

4. З'єднайте блоки стрілками, щоб показати навігаційні зв'язки (лінійні, ієрархічні або розгалужені).

**3. Опис сценарію використання (Use Case)** У текстовому документі опишіть шлях користувача під час взаємодії з вашим ресурсом.

1. Основний шлях. Опишіть покроково ідеальну ситуацію. *Приклад: Користувач натискає «Почати» → переглядає відео → переходить до тесту → правильно відповідає на 3 питання → отримує повідомлення «Вітаємо, ви пройшли рівень!».*

2. Альтернативний шлях (Обробка помилки). Опишіть, що станеться, якщо користувач помилиться. *Приклад: Користувач дає неправильну відповідь на 2 питання → система показує повідомлення «Спробуй ще раз» і кнопку підказки → користувач натискає підказку → читає правило → повертається до питання.*

#### Приклади тематики для завдання.

1. Правила безпечної роботи за комп'ютером.
2. Апаратне забезпечення: складові персонального комп'ютера.
3. Еволюція носіїв інформації: від перфокарт до хмарних сховищ.
4. Клавіатурні скорочення (гарячі клавіші) ОС Windows/macOS.
5. Типи комп'ютерних вірусів та базовий антивірусний захист.
6. Базові типи даних у програмуванні.
7. Популярні мови програмування та їхнє призначення.
8. Що таке фішинг і як розпізнати шахрайські листи.

9. Правила створення та зберігання надійних паролів.
10. Растрова vs векторна графіка: ключові відмінності. .
11. Що таке хмарні технології: просто про складне.
12. Основні формати графічних файлів (JPEG, PNG, GIF, SVG). .
13. Історія розвитку мережі Інтернет.
14. Базові теги HTML для початківців.
15. Поняття алгоритму та його основні властивості.
16. Що таке штучний інтелект і нейромережі.
17. Топології комп'ютерних мереж (зірка, кільце, шина).
18. Пристрої введення та виведення інформації.
19. Одиниці вимірювання інформації (біти, байти, мегабайти).
20. Правила цифрового етикету (нетикету) в мережі.
21. Як працює пошукова система: алгоритми Google.
22. Огляд найпопулярніших веббраузерів та їхні особливості.
23. Основні професії в сучасному IT-секторі.
24. Принципи роботи електронної пошти та протоколи (POP3, IMAP, SMTP).
25. Вільне (Open Source) та пропріетарне програмне забезпечення.

### **Форма звітності**

Здобувач має здати на перевірку викладачу:

- Посилання на створену інтерактивну структуру ресурсу (блок-схему в Міро з відкритим доступом для перегляду).
- Текстовий файл із короткою інформацією про обрану тему, дидактичну мету та розписаними сценаріями використання (основний та альтернативний).

## **Практичне завдання №2**

### **Створення візуальних цифрових освітніх ресурсів**

**Інструменти:** Хмарні сервіси Canva або Genially.

#### **Порядок виконання завдання:**

##### **1. Підготовчий етап**

1. Використовується обрана тематика у завданні №1.
2. Проаналізуйте матеріал та визначте, який формат підійде найкраще: постер для узагальнення правил, інфографіка для відображення статистики/процесів чи інтерактивна презентація для покрокового пояснення.
3. Виділіть ключові тези та підберіть графічні елементи. Зменште обсяг суцільного тексту до мінімуму.

##### **2. Розробка ресурсу (на вибір здобувача)**

###### **Варіант А: Створення постера або інфографіки (Canva).**

1. Авторизуйтеся у сервісі Canva.
2. Оберіть формат «Освітня інфографіка» або «Плакат».
3. Розробіть структуру (заголовок, основні інформаційні блоки, висновки або контактна інформація).
4. Оформіть матеріал, використовуючи не більше 3-х базових кольорів та 2-х шрифтів. Додайте іконки для візуального підкріплення тексту.

###### **Варіант Б: Створення інтерактивної презентації (Genially).**

1. Авторизуйтеся у сервісі Genially.
2. Створіть нову презентацію (рекомендований обсяг — 3-5 слайдів).
3. Додайте інтерактивні елементи (налаштуйте кнопки навігації (переходи між слайдами не за порядком) або спливаючі вікна (Tooltip/Window) для розміщення додаткової інформації без перевантаження слайда).

### **Форма звітності**

Здобувач має здати на перевірку викладачу:

- Інтерактивний ресурс (презентацію) у вигляді відкритого вебпосилання або статичний плакат/інфографіку у форматі PDF чи зображення.

### Практичне завдання №3

#### Розробка інтерактивних навчальних вправ та міні ігор. Створення навчальних квестів та інтерактивних вікторин для контролю знань

**Інструменти:** Сервіси для створення інтерактивних вправ LearningApps, Wordwall (за бажанням — Scratch або Quizizz). Хмарні сервіси Google Forms, Kahoot, Genially (на вибір здобувача).

#### Порядок виконання завдання:

##### 1. Підготовка контенту

1. Використовується обрана тематика на занятті №1.
2. Підготуйте текстовий або візуальний матеріал для двох різних типів ігрових вправ (на вибір здобувача).

##### 2. Розробка цифрового освітнього ресурсу (один варіант на вибір)

###### Варіант А: Розгалужений квест у Google Forms

1. Створіть нову форму в Google Forms та увімкніть режим "Тести" в налаштуваннях (для автоматичного нарахування балів).
2. Розбийте ваші запитання на окремі "Розділи".
3. Для кожного запитання налаштуйте функцію "Перейти до розділу на основі відповіді": правильна відповідь переводить користувача до наступного завдання квесту, а неправильна — до розділу з підказкою або штрафним завданням.
4. Додайте обов'язковий зворотний зв'язок (коментарі) для правильних та неправильних відповідей.

###### Варіант Б: Динамічна вікторина у Kahoot!

1. Авторизуйтеся на платформі Kahoot!.
2. Створіть новий квіз та перенесіть туди підготовлені запитання, додавши ілюстрації.
3. Налаштуйте ігрові параметри для кожного запитання: ліміт часу (наприклад, 20-30 секунд) та кількість балів (стандартні або подвійні за складні питання).
4. Збережіть вікторину та згенеруйте посилання або PIN-код для приєднання учасників.

###### Варіант В: Візуальний квест (Escape Room) у Genially

1. Увійдіть у сервіс Genially та оберіть категорію "Gamification".
2. Знайдіть шаблон типу "Escape Game" або "Breakout".
3. Замініть шаблонний текст на ваші запитання та відповіді, зберігаючи закладену в шаблон логіку переходів між слайдами.
4. Налаштуйте інтерактивні кнопки та спливаючі вікна-підказки.

###### Варіант Г: Розробка інтерактивного тренажера у LearningApps

1. Перейдіть на платформу LearningApps та авторизуйтеся.
2. Натисніть "Створити вправу" та оберіть шаблон «Знайди пару» (Matching) або «Класифікація» (Group assignment).
3. Заповніть поля шаблону підготовленим матеріалом (комбінуйте текст та зображення для кращого візуального сприйняття).
4. Обов'язково налаштуйте мотиваційний зворотний зв'язок (фідбек) — текст, який побачить користувач після успішного виконання завдання.
5. Збережіть вправу та налаштуйте доступ за посиланням.

###### Варіант Д: Розробка мінігри у Wordwall

1. Перейдіть на платформу Wordwall та авторизуйтеся.
2. Натисніть "Створити активність" і оберіть яскраво виражений ігровий шаблон, наприклад, «Кросворд» (Crossword), «Лабіринт» (Maze Chase) або «Літак» (Airplane).

3. Введіть підготовлені запитання та правильні/неправильні варіанти відповідей.
4. Збережіть гру та протестуйте її самостійно.
5. Зверніть увагу на функцію перемикання шаблонів: спробуйте змінити ігрову механіку за допомогою панелі праворуч (наприклад, перетворіть "Лабіринт" на "Вікторину") без повторного введення даних. Згенеруйте фінальне посилання для доступу.

### **Форма звітності**

Для отримання поточної оцінки за заняття (за 100-бальною шкалою) здобувачі мають:

- Надати викладачу посилання на створений ресурс.

## **ІНДИВІДУАЛЬНЕ ЗАВДАННЯ НА ПРАКТИКУ**

Здобувач вищої освіти має обрати одну комплексну тему з дисципліни своєї майбутньої фахової спеціалізації. Для цієї теми необхідно розробити повноцінний навчальний модуль, який охоплюватиме всі етапи засвоєння матеріалу (мотивацію, вивчення теорії, практичне закріплення та контроль знань). Усі створені матеріали мають бути інтегровані в єдиний освітній мікросайт.

### **Обов'язкові складові індивідуального комплексу ЦОР**

Здобувачі здають підготовлені за час практики цифрові освітні ресурси, серед яких обов'язково мають бути такі продукти:

1. Проектна документація. Створена у сервісі Migo ментальна карта, що відображає проєктування логіки ресурсу, сценаріїв використання та інтерактивної структури комплексу.
2. Візуальний супровід. Інтерактивний ресурс (презентація, плакат або інфографіка), розроблений у Canva або Genially для пояснення теоретичних аспектів теми.
3. Практичний блок. Навчальна гра (мінігра, тренажер) або серія інтерактивних вправ, створених у LearningApps чи Wordwall.
4. Блок контролю знань. Навчальний квест, вікторина або інтерактивний тест, розроблений за допомогою Google Forms, Kahoot або Genially.
5. Аудіовізуальні продукти. Комплекс медіафайлів для супроводу теми, що включає **(на вибір здобувача, можуть поєднуватись)**:
  - Навчальне відео (скрінкаст-відеоурок), змонтоване у Clipchamp.
  - Навчальний подкаст (аудіозапис), оброблений в Audacity.
  - Короткий анімаційний ролик, створений у Powtoon або Animaker.
6. Електронне видання. Макет електронного посібника (створений у Book Creator або Canva), який містить текст, фото, відео та звук. **(виконується за бажанням)**
7. Вебагрегатор. Освітній мікросайт (створений на базі Google Sites), який містить логічно структуровані сторінки з вбудованими (інтегрованими) ресурсами, переліченими у пунктах 2–6.

### **Орієнтовна тематика індивідуальних завдань**

1. Інформаційні системи
2. Історія розвитку комп'ютерної техніки
3. Базові елементи мови C++
4. Програмне забезпечення інформаційної системи
5. Типи пам'яті комп'ютера. Оперативний запам'ятовуючий пристрій. Визначення характеристик
6. Колірні моделі
7. Елементна база комп'ютерної системи (Процесор, оперативна пам'ять, відеокарта, материнська плата). Основні та периферійні пристрої
8. Команда розгалуження, вибору в мові C++
9. Технології тестування комп'ютерних комплектуючих
10. Дослідження відеокарти. Основні характеристики, визначення моделі, типу роз'єму та додаткових характеристик

11. Системи опрацювання текстової інформації в MS Word
12. Основи створення та обробки векторних зображень
13. Цикли з передумовою та цикли з післяумовою
14. Створення комп'ютерних презентацій та публікацій
15. Одновимірні та двовимірні масиви
16. Засоби введення-виведення графічної інформації
17. Технології опрацювання даних у електронних таблицях MS Excel
18. Поняття HTML та CSS
19. Формати файлів для збереження графічної інформації.
20. Основи JS
21. Спеціалізоване програмне забезпечення для тестування комплектуючих
22. Поняття комп'ютерної системи. Класифікація комп'ютерів: офісні, домашні, ігрові
23. Накопичувачі. Жорсткі диски та твердотільні накопичувачі. Порівняння характеристик
24. Основи створення та обробки растрових зображень
25. Операційні системи Windows. Особливості роботи, переваги та недоліки. Процедура встановлення та налаштування
26. Накопичувачі. Жорсткі диски та твердотільні накопичувачі. Порівняння характеристик
27. Дослідження центрального процесору. Основні характеристики, визначення моделі, типу сокету
28. Накопичувачі. Жорсткі диски та твердотільні накопичувачі. Порівняння характеристик

**Зверніть увагу:** Цей список є орієнтовним та не є вичерпним. Здобувачі вищої освіти можуть запропонувати власну тему в межах своєї фахової дисципліни. За потреби індивідуальний вибір теми та її деталізація можуть додатково обговорюватись та затверджуватись із викладачем-керівником практики.

### **Порядок виконання індивідуального завдання**

**Створення ментальної карти (Miro) зі структурою майбутнього комплексу ЦОР та сценарієм заняття.**

#### *1) Вибір теми індивідуального завдання*

Оберіть комплексну тему з запропонованого списку або запропонуйте свою.

#### *2) Формування сценарію заняття*

Здійсніть проєктування логіки ресурсу та сценаріїв його використання цільовою аудиторією. Розбийте обрану тему на логічні блоки:

- Мотивація/Вступ. Як ви зацікавите користувача?
- Теорія. Через які ресурси подаватиметься основний матеріал?
- Практика. Які інтерактивні вправи допоможуть закріпити навички?
- Контроль. Як відбуватиметься перевірка знань?

#### *3) Створення ментальної карти в Miro*

Перейдіть до сервісу Miro та створіть нову дошку для проєктування інтерактивної структури вашого комплексу. Побудуйте ієрархію:

1. Центральний вузол. Назва вашої теми.
2. Гілки першого рівня. Основні етапи заняття (Вступ, Теорія, Практика, Тестування).
3. Гілки другого рівня. Конкретні типи ЦОР, які ви будете створювати для кожного етапу (наприклад, інтерактивна презентація в Canva, відеоурок у Clipchamp, гра у Wordwall, квест у Google Forms).
4. Гілки третього рівня (деталізація). Коротко пропишіть, яка саме інформація або які ключові запитання будуть у кожному ресурсі.

#### *4) Звітність та погодження*

Згенеруйте посилання на вашу дошку в Miro (з правами для перегляду) та надайте його викладачу.

**Візуальний супровід. Розробка інтерактивної презентації або детальної інфографіки (Canva/Genially) для пояснення теоретичної частини обраної теми.**

*1) Вибір формату та підготовка контенту*

- Визначте, який формат краще розкриє вашу тему: лінійна інтерактивна презентація чи довга детальна інфографіка (плакат).

- Відберіть ключові теоретичні поняття з вашої ментальної карти (Miro).

*Приклад:* Для пояснення архітектури графічного конвеєра доцільно розробити інтерактивну інфографіку, де кожен блок ілюструє окрему стадію перетворення тривимірної сцени (від обробки вершин до фінального піксельного рендерингу), а натискання на елементи відкриває детальні пояснення алгоритмів.

*2) Робота у візуальному редакторі (Canva або Genially)*

- Авторизуйтеся в обраному хмарному сервісі.
- Оберіть відповідний освітній шаблон або створіть дизайн з нуля.
- Розподіліть підготовлений текст та ілюстрації (схеми, графіки, моделі) по слайдах або блоках інфографіки.

- Дотримуйтеся принципів типографіки та екранного читання: використовуйте шрифти без зарубок (Sans Serif), контрастні кольори та залишайте достатньо "повітря" (вільного простору).

*3) Налаштування інтерактивності*

- Додайте гіперпосилання або кнопки для зручної нелінійної навігації між розділами матеріалу.

- Використайте функцію інтерактивності (наприклад, спливаючі вікна чи підказки у Genially), щоб сховати розгорнуті визначення чи додаткові приклади. Це дозволить не перевантажувати основний екран суцільним текстом.

- Застосуйте доречну анімацію появи елементів для фокусування уваги користувача.

*4) Публікація*

- Перевірте роботу всіх кнопок, посилань та спливаючих вікон у режимі попереднього перегляду.

- Опублікуйте вашу роботу та згенеруйте публічне вебпосилання (Share link).

**Практикум. Створення набору з 1-2 інтерактивних вправ чи міні ігор (LearningApps/Wordwall) для закріплення термінології або процесів.**

*1) Підготовка навчального контенту та вибір механік*

1. Проаналізуйте теоретичний матеріал вашого проєкту та виділіть ключові терміни, класифікації або алгоритми, які користувач має запам'ятати.

2. Оберіть 1-2 різні ігрові механіки (наприклад, "встановлення відповідностей", "сортування за групами", "правильна послідовність" або "кросворд").

3. Складіть текстові пари (термін — визначення) або підберіть тематичні зображення.

*Приклад:* Для теми тривимірної графіки доцільно розробити вправу на встановлення правильної послідовності етапів графічного конвеєра, гру на сортування властивостей об'єктів (наприклад, планарні та непланарні сітки), або вікторину щодо налаштування параметрів шорсткості матеріалів у Blender.

*2) Розробка вправ у хмарних сервісах (LearningApps або Wordwall)*

1. Авторизуйтеся в обраному сервісі та створіть нові активності за обраними шаблонами.

2. Заповніть шаблони підготовленим текстом та візуальними матеріалами.

3. Обов'язково налаштуйте зворотний зв'язок (фідбек) — мотивуючий текст або підказку, яку побачить користувач після успішного завершення вправи чи у разі помилки.

*3) Тестування та налаштування доступу*

1. Пройдіть створені міні ігри самостійно, щоб перевірити їхню логіку, складність та відсутність двозначних відповідей.

2. Збережіть ігри та змініть налаштування приватності так, щоб вони були доступні іншим користувачам.

3. Згенеруйте вебпосилання (або збережіть код вбудовування — Embed/iframe, який знадобиться вам на фінальному етапі для розміщення вправ на мікросайті).

**Контроль знань. Розробка підсумкового тесту, вікторини або квесту (Kahoot/Quizizz/Google Forms) за матеріалами теми.**

*1) Проектування бази запитань*

- Проаналізуйте теоретичний та практичний блоки вашого проєкту. Сформулюйте 10-15 запитань різного рівня складності, які комплексно охоплюють тему.

- До кожного запитання підготуйте одну правильну відповідь та 2-3 правдоподібні неправильні варіанти (дистрактори).

*Приклад:* Для оцінювання знань із тривимірного моделювання можна додати питання на впізнавання алгоритмів рендерингу за описом, порівняння планарних і непланарних сіток або визначення правильних параметрів шорсткості матеріалів.

*2) Вибір платформи та формату оцінювання*

- Оберіть інструмент, який найкраще відповідає логіці вашого комплексу:

- Google Forms (ідеально для створення розгалужених текстових квестів (де неправильна відповідь веде на сторінку з підказкою) або класичних тестів із детальним зворотним зв'язком).

- Kahoot! / Quizizz (оптимальний вибір для створення динамічних вікторин із таймером, нарахуванням балів за швидкість та елементами змагання).

*3) Програмна реалізація та налаштування*

- Авторизуйтеся на обраній платформі та створіть новий тест/вікторину.

- Перенесіть підготовлені запитання. Обов'язково додайте візуальний супровід (зображення, схеми чи короткі відеофрагменти) до тих питань, де це доцільно.

- Налаштуйте параметри оцінювання: кількість балів за правильну відповідь, ліміт часу та автоматичний показ правильних відповідей/пояснень (фідбек) після завершення.

*4) Тестування та звітність*

- Пройдіть створений тест самостійно в режимі попереднього перегляду, щоб переконатися у відсутності помилок та коректності нарахування балів.

- Опублікуйте ресурс та згенеруйте публічне посилання (для Google Forms переконайтеся, що вимкнено обмеження доступу для зовнішніх користувачів).

**Аудіосупровід. Написання сценарію та запис короткого навчального подкасту (Audacity) з додатковою чи цікавою інформацією до теми.**

*1) Вибір теми та написання сценарію*

Оберіть вузьку підтему, цікавий факт або історичну довідку, яка доповнює основний теоретичний матеріал вашого проєкту, але не дублює його. Напишіть детальний текст для аудіозапису (орієнтовний обсяг — на 2–3 хвилини звучання). Наприклад, якщо основною темою вашого проєкту є тривимірне моделювання та етапи графічного конвеєра, у подкасті можна розповісти слухачам про еволюцію алгоритмів рендерингу тривимірних сцен або цікаві нюанси налаштування фізично коректних матеріалів (зокрема, параметрів шорсткості) у Blender.

*2) Запис дикторського голосу в Audacity*

Запустіть програму Audacity та перевірте підключення мікрофона. Натисніть кнопку запису і начитайте підготовлений текст. Намагайтеся говорити чітко та в одному темпі. Якщо

ви зробили помилку або обмовилися, не зупиняйте запис — просто зробіть коротку паузу (щоб потім це місце було легко візуально знайти на аудіодоріжці) і повторіть фразу знову.

### 3) *Монтаж та обробка аудіо*

Використовуючи інструмент виділення, виріжте всі невдалі дублі, довгі паузи та зайві звуки (кашель, гучне дихання). Виділіть невеликий фрагмент "тиші" на записі та застосуйте ефект «Зниження шуму», після чого очистьте всю доріжку від фонового гудіння. Далі застосуйте ефект «Нормалізація» (Normalize), щоб вирівняти загальну гучність вашого голосу.

### 4) *Зведення з музикою та експорт*

Завантажте легку фонову інструментальну музику з відкритою ліцензією (Creative Commons) та імпортуйте її на нову доріжку в проєкті. За допомогою інструмента «Огинаюча» (Envelope Tool) або повзунка гучності треку зменште звук музики так, щоб вона створювала приємну атмосферу, але не перекривала дикторський голос. Налаштуйте плавне затухання треку в кінці. Екпортуйте готовий проєкт як аудіофайл у форматі MP3.

**Відеосупровід. Зйомка та монтаж відеоуроку (Clipchamp), де здобувач особисто пояснює найскладніший аспект обраної теми.**

### 1) *Підготовка сценарію та середовища*

- Виділіть найскладніший або найважливіший аспект вашої теми.
- Підготуйте матеріал для демонстрації: відкрийте необхідне програмне забезпечення, презентацію або код.

- Складіть короткий план виступу, щоб мова була лаконічною та структурованою.

### 2) *Запис відео*

- Відкрийте Clipchamp і оберіть функцію «Запис і створення» (Record & create), «Екран і камера» (Screen & camera).

- Налаштуйте вивід мікрофона та розташування вікна камери (якщо ви плануєте бути в кадрі — це створює ефект присутності).

- Розпочніть запис. Чітко озвучте тему, поясніть логіку ваших дій і продемонструйте процес на екрані. Після завершення запису зупиніть його, і він автоматично додасться на таймлайн.

### 3) *Монтаж та доповнення*

- Очищення. Використовуйте інструмент «Розділити» (Split) та «Видалити», щоб вирізати невдалі дублі, паузи чи обмовки.

- Акценти. Додайте у вкладці «Текст» текстові плашки (Lower Thirds) із назвами ключових термінів або формул у моменти, коли ви їх озвучуєте.

- Візуальна динаміка. Використовуйте переходи (Transitions) між фрагментами та, за потреби, накладіть фонову музику на низькій гучності (через інструмент «Гучність» на панелі властивостей), щоб надати відео професійного вигляду.

### 4) *Експорт та звітність*

- Натисніть «Експортувати» (Export) і оберіть роздільну здатність 1080p для чіткого відображення деталей інтерфейсу.

- Збережіть готовий файл у форматі MP4 або завантажте його на хмарний диск.

**Анімація. Створення короткого анімаційного ролика (Powtoon/Animaker) для візуалізації певного явища або правила з теми.**

### 1) *Вибір об'єкта анімації та підготовка сценарію*

- Виберіть процес з вашої теми, що має динаміку (наприклад, рух електронів, етапи роботи алгоритму, цикл виконання програми або взаємодія компонентів системи).

- Напишіть сценарій на 30–60 секунд. Опишіть, що буде відбуватися в кадрі (які персонажі або об'єкти з'являться і куди будуть рухатися).

### 2) *Робота в сервісі (Powtoon або Animaker)*

- Авторизуйтеся в обраному сервісі та оберіть шаблон «Навчання» (Education) або почніть проєкт із «Чистого аркуша».

- Створення сцени. Розмістіть на робочій області фонові об'єкти та персонажів, що відповідають вашій темі.
- Анімація об'єктів. Виберіть ефекти для появи та руху об'єктів. Налаштуйте траєкторії руху (Motion Path), щоб процес виглядав логічно.
- Текст. Додайте мінімальну кількість тексту — лише для виділення головних понять.

### 3) Робота з таймлайном та озвучуванням

- Розподіліть події на часовій шкалі (Timeline) так, щоб рух об'єктів відповідав швидкості викладу матеріалу.
- Додайте озвучку (запишіть дикторський голос безпосередньо в сервісі або завантажте вже готовий аудіофайл) та синхронізуйте рухи персонажів з голосом (функція Lip-sync, якщо доступна).

### 4) Фіналізація та звітність

- Перегляньте результат у режимі Preview. Переконайтеся, що анімація не перевантажена зайвими деталями і чітко розкриває суть явища (принцип когнітивної теорії мультимедійного навчання).
- Опублікуйте проєкт і згенеруйте публічне посилання на нього.

## **Пакування в електронний посібник. Верстка матеріалів (Book Creator) у єдиний посібник, додавання текстів, зображень та мультимедіа. (виконується за бажанням)**

### 1) Структурування книги

- Створіть нову книгу в Book Creator. Оберіть формат сторінок (рекомендується альбомний для кращого відображення на екранах).
- Спроектуйте структуру (мінімум 5-6 сторінок):
  - Обкладинка. Назва вашої теми та ім'я автора.
  - Зміст. Створіть клікабельні посилання на ключові розділи.
  - Теоретичні розділи. Послідовне розміщення текстових блоків, зображень та схем.
  - Мультимедійні сторінки. Вбудовування створених раніше відео, подкастів та анімацій.
  - Блок самоперевірки. Сторінка з посиланнями на ваші інтерактивні вправи чи тести.

### 2) Верстка та інтеграція контенту

- Текст та зображення. Додайте основний навчальний текст. Використовуйте шрифти без зарубок для кращого читання. Вставте раніше створену інфографіку або зображення з вашого проєкту.
- Мультимедіа. Вставте посилання на ваш відеоролик, подкаст або анімаційний ролик. Переконайтеся, що об'єкти коректно відображаються у вікні перегляду.
- Інтерактив. Додайте кнопки-гіперпосилання для переходу на зовнішні ресурси (ваші тести в Google Forms або вправи у Wordwall).

### 3) Дизайн та логіка навігації

- Дотримуйтесь єдиного стилю (кольорова гама, шрифти) для всіх сторінок.
- Перевірте, щоб "повітря" (вільного простору) було достатньо для комфортного читання.
- Налаштуйте навігацію: додайте на кожну сторінку кнопку «На головну» або «До змісту».

### 4) Фінальна перевірка та публікація

- Перейдіть у режим «Read book» і клацайте кожне посилання, відтворіть кожне вбудоване відео. Переконайтеся, що все працює без помилок.
- Натисніть кнопку «Share» + «Publish online». Переконайтеся, що книга має статус Public (для можливості перегляду викладачем).
- Скопіюйте посилання на опубліковану книгу для подальшого розміщення на вашому вебсайті (Google Sites).

## **Інтеграція ресурсів. Створення навчального мінісайту (Google Sites). Розміщення на ньому всіх розроблених на заняттях матеріалів у логічній послідовності.**

### *1) Структурування сайту (Навігація)*

- Перейдіть до Google Sites та створіть новий проєкт.
- У меню «Сторінки» (Pages) створіть структуру, що відповідає вашому сценарію заняття з ментальної карти (Miro):
  - Головна. Назва теми, мета, опис для кого цей курс.
  - Теоретичний блок. Сторінка з інтерактивною презентацією/інфографікою та вбудованим відеоуроком.
  - Анімація та Додаткові матеріали. Сторінка для вашого анімаційного ролика та подкасту.
  - Практика. Сторінка з вбудованими мінііграми (LearningApps/Wordwall).
  - Контроль. Сторінка з підсумковим тестом/квестом.

### *2) Інтеграція (Вебагрегація) контенту*

- Для розміщення кожного ресурсу використовуйте функцію «Вбудувати» (Embed), «За вбудованим кодом» (Embed code). Використовуйте <iframe> коди, надані на попередніх заняттях, для коректного відображення (безшовна інтеграція).
- Якщо прямий код недоступний, використовуйте функцію «Вбудувати» через URL або вставляйте як посилання з кнопкою (Button), але забезпечте, щоб зовнішній вигляд був професійним та читабельним.

### *3) Дизайн та логіка викладу*

- Дотримуйтесь єдиного візуального стилю (заголовки, шрифти, кольори).
- Забезпечте логічну послідовність: контент має рухатися від теорії до практики, а потім до контролю знань.
- Перевірте «адаптивність» — перегляньте, як ваш сайт відображається на різних пристроях (кнопка «Попередній перегляд»).

### *4) Фінальне тестування та публікація*

- Проклікайте кожну сторінку, перевірте роботу кожного відео, аудіо та тесту. Не повинно бути неробочих посилань (404 error).
- Натисніть «Опублікувати» (Publish). Задайте адресу сайту (наприклад, yourname-course-topic).

## **ЗАХИСТ ПРАКТИКИ ТА ВИМОГИ ДО ЗВІТНОЇ ДОКУМЕНТАЦІЇ**

Виконання індивідуальних завдань є найважливішим етапом роботи здобувача і становить 50% від загального розподілу підсумкової оцінки за практику.

На всіх етапах проходження практики здобувачі отримують індивідуальні консультації у своїх викладачів-керівників з метою виявлення недоліків та надання практичної допомоги у процесі виконання завдання.

Під час проходження практики обов'язково відбувається захист індивідуальних практичних завдань. Під час захисту (на 20-му занятті) здобувач має презентувати свій освітній мікросайт, продемонструвати роботу всіх вбудованих ЦОР та обґрунтувати їх дидактичну ефективність і відповідність обраній фаховій темі. Після цього здобувачі надають на кафедру відповідну звітну документацію (Додаток А).

### **Підготовка та презентація індивідуальних практичних завдань**

Підготуйте коротку усну доповідь (на 3–5 хвилин), яка включає:

- Актуальність. Чому обрана вами тема є важливою для вивчення?
- Структура. Які саме ЦОР ви розробили та як вони пов'язані між собою (демонстрація логіки ментальної карти з Miro).
- Демонстрація. Короткий показ "шляху користувача" на вашому мікросайті (від вступу до контролю знань).

○ Ефективність. Як ваші цифрові інструменти допомагають краще засвоїти складні аспекти теми?

Презентація та дискусія:

• Під час заняття презентуйте свої результати виконання індивідуального завдання.

• Будьте готові відповісти на запитання колег та викладача щодо технічних рішень (наприклад, чому обрали саме цей інструмент для тестування) та дидактичного наповнення.

• Візьміть активну участь в обговоренні проєктів інших здобувачів — конструктивна критика є частиною вашої професійної підготовки.

### **Оформлення звітної документації**

Переконайтеся, що всі посилання (на Miro, сайти, інтерактивні вправи, відео) є робочими та відкритими для перегляду. Сформууйте фінальний перелік (або звітний файл), де впорядковано наведені всі посилання на ваші напрацювання за заняттями. Підготуйте звіт за шаблоном (Додаток А).

## ЗАВДАННЯ ПРАКТИКИ НА 5-Й СЕМЕСТР

*Загальний обсяг: 90 годин, з яких 40 годин аудиторної та 50 годин самостійної роботи.*

У цьому семестрі увага приділяється веброзробці, складному мультимедіа, ігровим рушіям та технологіям віртуальної/доповненої реальності.

- Проектування складної логіки ресурсу, розгалужених сценаріїв використання та комплексної інтерактивної структури.
- Веброзробка навчальних мінісайтів, тренажерів та інтерактивних сторінок із застосуванням HTML, CSS, JavaScript.
- Розробка повноцінних освітніх ігор на платформах Construct 3, Twine, Unity WebGL.
- Створення комплексних електронних підручників або навчальних модулів з мультимедіа і тестами в H5P та Book Creator.
- Створення віртуальних виставок, 3D-турів та навчальних об'єктів за допомогою Artsteps, Vectary, Tinkercad.

Навчальну практику розбито на дві логічні частини: практичне ознайомлення з інструментарієм (заняття 1–10) та створення комплексного індивідуального проєкту (заняття 11–20).

У першій частині здобувачі вивчають функціонал платформ та інструментарій, а у другій частині здобувачі працюють над власним індивідуальним проєктом за обраною темою.

### ТЕОРЕТИЧНІ ВІДОМОСТІ

#### **Основи веброзробки для створення навчальних міні сайтів. Інтерактивні вебсторінки.**

*Навчальний мінісайт як цифровий освітній ресурс*

Навчальний мінісайт — це спеціалізований веб-ресурс із невеликою кількістю сторінок (або односторінковий застосунок — Single Page Application), який присвячений певній вузькій освітній темі, дисципліні або окремому навчальному модулю. Основна перевага такого формату полягає у можливості структурувати мультимедійний контент, об'єднавши текстові, графічні, аудіо- та відеоматеріали з інтерактивними завданнями в єдиному керованому середовищі.

Для розробки таких ресурсів з нуля використовується класична тріада вебтехнологій: HTML, CSS та JavaScript.

*HTML (HyperText Markup Language.: Каркас вебсторінки*

HTML є базовою мовою розмітки гіпертексту, що визначає логічну структуру та вміст вебсторінки.

- Елементи та теги. Будівельними блоками HTML є теги (наприклад, `<h1>` — `<h6>` для заголовків, `<p>` для текстових абзаців, `<a>` для гіперпосилань, `<img>` для інтеграції зображень).

- Семантика. Сучасний стандарт HTML5 передбачає використання семантичних тегів (таких як `<header>`, `<nav>`, `<article>`, `<section>`, `<footer>`), які чітко вказують браузерам та допоміжним технологіям на призначення певного блоку. Це є критично важливим для забезпечення інклюзивності освітніх ресурсів.

- DOM (Document Object Model). Після завантаження HTML-документа браузер перетворює його на об'єктну модель документа (DOM) — деревоподібну структуру, де кожен елемент є окремим вузлом. Саме до цієї структури згодом звертаються стилі та скрипти.

*CSS (Cascading Style Sheets). Візуальне оформлення*

Каскадні таблиці стилів (CSS) відповідають за зовнішній вигляд, форматування та просторове розташування HTML-елементів.

- Стилізація. CSS дозволяє керувати кольорами, типографікою, відступами, розмірами та візуальними ефектами.

- Селектори та правила. За допомогою селекторів (за тегом, класом `.classname`, або ідентифікатором `#idname`) розробник "звертається" до конкретного HTML-елемента в DOM-дереві і застосовує до нього набір візуальних правил.

- Адаптивність (Responsive Web Design). Завдяки гнучким сіткам (Flexbox, Grid) та медіазапитам (`@media`), CSS забезпечує коректне відображення навчального міні сайту на

різних пристроях (смартфонах, планшетах, десктопах), що є обов'язковою вимогою до сучасних цифрових освітніх продуктів.

*JavaScript (JS). Інтерактивність та логіка*

JavaScript — це високорівнева мова програмування, яка перетворює вебсторінку зі статичного документа на динамічний та інтерактивний застосунок.

- Маніпуляція DOM. JS дозволяє програмно змінювати HTML-вміст та CSS-стилі (додавати або видаляти елементи, змінювати їхні атрибути) у режимі реального часу, без необхідності перезавантажувати сторінку.

- Обробка подій (Event Handling). JS "слухає" та реагує на дії користувача — кліки мишкою (onclick), введення тексту з клавіатури, наведення курсора, перетягування об'єктів (Drag-and-Drop) тощо.

- Логіка навчальних інструментів. За допомогою JavaScript реалізується дидактичний потенціал ресурсу: перевірка відповідей у тестах та тренажерах, підрахунок балів, миттєвий зворотний зв'язок (виведення підказок або повідомлень про помилку), робота з таймерами, анімація графіків та створення повноцінних браузерних навчальних міні ігор.

*Синергія технологій у створенні ЦОР*

Для успішної розробки освітнього міні сайту необхідна тісна взаємодія всіх трьох компонентів, кожен з яких виконує свою специфічну роль:

- HTML формує вміст і відповідає на питання "Що це за елемент?" (наприклад, кнопка для перевірки знань).

- CSS формує дизайн і визначає "Як він виглядає?" (кнопка має синій колір, заокруглені кути та тінь).

- JavaScript формує поведінку і керує тим, "Що відбувається, коли з ним взаємодіють?" (при натисканні на кнопку скрипт зчитує введену відповідь, порівнює її з еталоном і забарвлює елемент у зелений або червоний колір).

### **Створення комплексних електронних підручників та навчальних модулів.**

#### **Мультимедійні посібники та їх інтеграція.**

*Сутність комплексних електронних підручників та мультимедійних посібників*

Сучасний електронний підручник — це не просто оцифрована копія друкованого видання (наприклад, звичайний PDF-файл), а комплексний дидактичний засіб, що поєднує структурований навчальний матеріал із мультимедійними елементами та інструментами інтерактивної взаємодії.

Мультимедійні посібники відрізняються високим рівнем візуалізації та наявністю різноманітних медіаформатів (аудіо, відео, 3D-моделі, анімація), що дозволяє задіяти різні канали сприйняття інформації та значно підвищити ефективність навчання.

*Структура комплексного навчального модуля*

Для забезпечення повноцінного циклу навчання цифровий модуль зазвичай складається з таких компонентів:

- Інформаційний блок. Теоретичний матеріал, поданий у вигляді гіпертексту, графічних схем, відео- чи аудіолекцій.

- Практичний блок. Завдання для відпрацювання навичок (наприклад, віртуальні симуляції або інтерактивні кейси).

- Контрольно-оцінювальний блок. Інтерактивні тести, опитування та завдання для самоперевірки з миттєвим автоматичним зворотним зв'язком.

- Навігаційний блок. Зручна та інтуїтивно зрозуміла система меню, пошуку та внутрішніх гіперпосилань для переміщення між розділами ресурсу.

*Сучасні інструменти для розробки (H5P та Book Creator)*

Для створення електронних підручників та навчальних модулів з мультимедіа і тестами сьогодні використовується низка спеціалізованих платформ:

- H5P – потужний відкритий фреймворк, який дозволяє створювати, ділитися та повторно використовувати інтерактивний HTML5-контент. Він ідеально підходить для розробки інтерактивних відео, де на певних хронометражах з'являються запитання, а також для створення розгалужених сценаріїв (Branching Scenarios) та повноцінних тестових модулів.

- Book Creator – простий та інтуїтивно зрозумілий хмарний інструмент для створення макетів електронних посібників та мультимедійних книг. Дозволяє легко комбінувати текст, зображення, аудіо та відео, а також малювати безпосередньо на сторінках, що робить його універсальним для візуалізації навчального контенту.

#### *Ергономічні та дидактичні вимоги до ЦОР*

Під час розробки мультимедійних посібників необхідно враховувати психолого-педагогічні закономірності сприйняття інформації:

- Дозування інформації. Уникнення когнітивного (розумового) перевантаження шляхом розбиття матеріалу на невеликі, логічно завершені фрагменти (принцип мікронавчання).

- Принцип мультимедійності. Пояснення матеріалу за допомогою комбінації тексту та релевантних зображень є ефективнішим, ніж лише за допомогою суцільного тексту.

- Адаптивність та інклюзивність. Контент має коректно відображатися на пристроях з різною діагоналлю екрана (смартфонах, планшетах, комп'ютерах) та бути доступним для здобувачів з особливими освітніми потребами (наявність субтитрів до відео, альтернативного тексту для зображень).

#### *Інтеграція в освітнє середовище*

Створений мультимедійний продукт потрібно правильно інтегрувати в загальний освітній процес закладу:

- Вбудовування (Embedding). Використання спеціальних кодів вставки (iframe) для інтеграції створених в H5P чи Book Creator об'єктів безпосередньо на сторінки навчальних мінісайтів або у блоги викладачів.

- Інтеграція в LMS (Системи управління навчанням). Використання стандартів SCORM або LTI (Learning Tools Interoperability) дозволяє завантажувати готові модулі в системи на кшталт Moodle, зберігаючи при цьому можливість автоматично збирати аналітику, відстежувати прогрес і формувати оцінки здобувачів.

### **Вступ до розробки освітніх ігор. Інтерактивні текстові ігри та квести. Ігрові рушії в освітньому процесі.**

#### *Сутність освітніх ігор (Serious Games)*

Освітні, або «серйозні» ігри (Serious Games) — це цифрові інтерактивні застосунки, розроблені з первинною метою навчання, тренування або моделювання певних процесів, де розважальний компонент відіграє допоміжну роль. На відміну від класичних відеоігор, орієнтованих виключно на відпочинок, освітні ігри мають чітко визначену дидактичну мету та вимірювані результати навчання. Вони застосовуються для симуляції складних професійних завдань, розвитку критичного мислення та безпечного відпрацювання практичних навичок.

#### *Гейміфікація та ігрове навчання: ключові відмінності*

У проектуванні цифрових освітніх ресурсів важливо розмежовувати ці два суміжні поняття:

- Гейміфікація (Gamification) - це впровадження окремих ігрових механік (бали, рівні, таблиці лідерів, бєджі, таймери) у традиційний, неігровий освітній контекст. Наприклад, нарахування здобувачам балів досвіду за вчасно здану лабораторну роботу.

- Ігрове навчання (Game-Based Learning) - це використання повноцінної гри як основного середовища та інструменту для засвоєння матеріалу. Здобувач взаємодіє з віртуальним світом, правилами та наративом гри, отримуючи знання безпосередньо через ігровий процес.

#### *Дидактичний потенціал та моделювання складних систем*

Освітні ігри дозволяють створити безпечне віртуальне середовище для експериментів, де помилка не має реальних негативних наслідків, а слугує тригером для аналізу та зміни стратегії.

Особливо ефективним цей підхід є в інженерних та ІТ-спеціальностях. Наприклад, у викладанні комп'ютерних наук високим дидактичним потенціалом володіють ігрові симулятори, що візуалізують внутрішні етапи графічного конвеєра. У такій освітній грі здобувач може інтерактивно взаємодіяти з алгоритмами рендерингу, досліджувати процеси

багатопрхідного рендерингу (multi-pass rendering) у тривимірній сцені або власноруч змінювати параметри шорсткості матеріалів на непланарних сітках. Це дозволяє наочно, через ігрову взаємодію, засвоїти складні технічні концепції побудови зображень.

#### *Базові етапи розробки освітньої гри*

Створення освітньої гри є комплексним процесом, який об'єднує педагогічний дизайн та класичну розробку програмного забезпечення (GameDev):

1. Концептуалізація та визначення освітньої мети. Формування ідеї гри, визначення цільової аудиторії та конкретних навичок або знань, які гравець має здобути.
2. Проектування ігрових механік (Game Design). Розробка правил гри, умов перемоги та поразки, системи винагород і балансу між складністю завдання та навичками гравця (стан «поток»).
3. Створення контенту та розробка. Написання сценарію, розробка візуального стилю (2D/3D графіка, інтерфейс), підбір звукового супроводу та безпосереднє програмування логіки в ігровому рушії.
4. Тестування та дидактична валідація. Перевірка гри на наявність програмних помилок (багів) та оцінка її реальної освітньої ефективності на фокус-групі здобувачів.
5. Реліз та інтеграція. Публікація цифрового продукту та його органічне впровадження у навчальну програму або курс.

#### *Сутність інтерактивних текстових ігор та навчальних квестів*

Інтерактивні текстові ігри (Interactive Fiction) та освітні квести — це формат цифрових освітніх ресурсів, у яких ключовим елементом взаємодії є робота з текстовою інформацією та прийняття рішень. На відміну від пасивного читання електронного підручника, здобувач виступає активним суб'єктом середовища: він розв'язує загадки, обирає репліки в діалогах або визначає послідовність дій для вирішення конкретної професійної чи навчальної задачі.

#### *Архітектура та нелінійний наратив (розгалужений сценарій)*

Основою більшості текстових ігор та складних квестів є нелінійна структура подачі інформації.

- Вузли та зв'язки. Сценарій розбивається на окремі фрагменти (вузли, сцени або сторінки), які з'єднані між собою логічними зв'язками або гіперпосиланнями.
- Багатоваріантність. Залежно від вибору користувача на поточному етапі, система перенаправляє його на відповідну гілку сюжету. Це дозволяє створювати кілька варіантів фіналу гри або різні шляхи до правильного вирішення квесту.
- Управління станом. Більш складні текстові ігри можуть запам'ятовувати вибори користувача (через змінні), формуючи унікальний досвід проходження (наприклад, накопичення "балів репутації" чи збір "інвентарю").

#### *Дидактичний потенціал формату*

Впровадження інтерактивних квестів та текстових ігор в освітній процес забезпечує виконання низки дидактичних завдань:

- Розвиток критичного мислення. Здобувач вчиться аналізувати вхідні дані, прогнозувати наслідки своїх рішень та вибудовувати причинно-наслідкові зв'язки.
- Моделювання ситуацій. Формат ідеально підходить для створення симуляцій (наприклад, алгоритм спілкування з клієнтом, порядок дій у нештатній ситуації, розслідування кіберінциденту).
- Підвищення мотивації. Елементи загадки, дослідження та наявність сюжету стимулюють пізнавальний інтерес та утримують увагу значно краще, ніж класичні тести.

#### *Інструментарій для розробки квестів та текстових ігор*

Для створення інтерактивних завдань такого типу використовується різноманітне програмне забезпечення, що не вимагає глибоких знань мов програмування:

- Платформи для квестів та тестів. Для створення лінійних та базових нелінійних навчальних квестів і тестів ефективно застосовуються такі інструменти, як Google Forms, Kahoot та Genially. За допомогою Genially, зокрема, можна створювати інтерактивні презентації-квести у стилі "Escape Room".
- Спеціалізовані середовища для текстових ігор. Для розробки повноцінних освітніх ігор із розгалуженим сюжетом використовується платформа Twine. Цей інструмент

дозволяє візуалізувати структуру текстової гри у вигляді графа (схеми), де розробник може легко керувати логікою переходів між текстовими блоками та створювати комплексний інтерактивний нарратив.

#### *Поняття ігрового рушія в контексті освіти*

Ігровий рушій (Game Engine) — це базове програмне середовище (фреймворк), яке містить набір інструментів для розробки відеоігор та інтерактивних застосунків. Сучасна освітня парадигма активно запозичує ці технології для створення навчальних симуляцій, віртуальних лабораторій, інтерактивних тренажерів та освітніх ігор. Використання таких середовищ дозволяє викладачам та розробникам не писати базовий код з нуля (наприклад, для відтворення графіки чи звуку), а зосередитися безпосередньо на проектуванні логіки ресурсу, сценаріїв використання та інтерактивної структури.

#### *Базова архітектура ігрового рушія*

Більшість сучасних ігрових рушіїв складаються з таких ключових підсистем, які виконують важливі функції в освітніх продуктах:

- **Графічний рушій (Рендерер).** Відповідає за виведення 2D або 3D графіки на екран. Дозволяє створювати реалістичні віртуальні виставки, 3D-тури та складні навчальні об'єкти.
- **Фізичний рушій.** Математично моделює закони фізики (гравітацію, зіткнення, пружність, динаміку рідин). Це критично важливо для розробки достовірних віртуальних лабораторних робіт з природничих дисциплін.
- **Система скриптингу.** Середовище для написання програмного коду (або візуального програмування), яке визначає реакцію програми на дії користувача, перевіряє правильність виконання навчального завдання та забезпечує зворотний зв'язок.
- **Аудіопідсистема.** Керує відтворенням звукових ефектів та фонові музики, що підвищує рівень занурення у навчальний процес.

#### *Популярні ігрові рушії для створення ЦОР*

- **Construct 3.** Потужний рушій для створення 2D-ігор, який базується на системі візуального програмування (Event-Driven Programming). Він не вимагає глибокого знання текстових мов програмування, що робить його ідеальним для швидкої розробки освітніх мініігор та інтерактивних вправ.
- **Unity (з підтримкою WebGL).** Один із найпопулярніших у світі професійних ігрових рушіїв для розробки 2D та 3D проєктів. Збірка під формат WebGL дозволяє запускати повноцінні просторові освітні ігри та симуляції безпосередньо у вікні веббраузера, що робить їх легкодоступними для здобувачів без необхідності встановлення додаткового програмного забезпечення.
- **Середовища для 3D-моделювання та віртуальної реальності.** Такі платформи як Artsteps, Vectary та Tinkercad не є класичними ігровими рушіями, проте вони використовують схожі технології вебрендерингу для створення просторових навчальних об'єктів та інтерактивних 3D-турів, які можна інтегрувати в загальний освітній процес.

#### *Дидактичні переваги використання рушіїв*

- **Безпека та доступність.** Можливість моделювання процесів, які в реальному житті є небезпечними (наприклад, хімічні реакції), дорогими або недоступними.
- **Мультиmodalність.** Поєднання візуального, аудіального та кінестетичного (через інтерактивну взаємодію) каналів сприйняття інформації, застосування технологій мультимедіа, гейміфікації та інтерактивного дизайну.
- **Підвищення мотивації.** Забезпечення високого рівня залученості здобувачів завдяки інтерактивності, можливості дослідження та миттєвому зворотному зв'язку в ігровому середовищі.

#### **Технології віртуальної реальності. 3D-моделювання навчальних об'єктів.**

#### *Віртуальна та доповнена реальність в освітньому середовищі*

Технології віртуальної реальності (VR) створюють повністю штучне, імерсивне цифрове середовище, тоді як доповнена реальність (AR) накладає цифрові об'єкти на реальний світ. В освітньому процесі ці технології дозволяють:

- Забезпечити ефект повної присутності під час вивчення складних просторових або історичних тем.
- Створювати безпечні умови для проведення віртуальних лабораторних робіт чи симуляцій.

- Проектувати комплексні цифрові продукти на виході, такі як 3D-тур, 3D-панорама або продукт з використанням технології доповненої реальності.

#### *Основи 3D-моделювання навчальних об'єктів*

3D-моделювання є ключовим етапом у підготовці віртуального контенту, що вимагає здатності застосовувати комп'ютерні методи побудови зображень та графічної подачі візуального матеріалу. Основні концепції включають:

- Полігональна сітка (Mesh). Базова структура 3D-моделі, що складається з вершин, ребер та граней. Для освітніх вебзастосунків критично важливо оптимізувати кількість полігонів, щоб створений ресурс швидко та стабільно завантажувався у браузері користувача.

- Матеріали та текстурування. Надання об'єкту фізично коректного та реалістичного вигляду за допомогою налаштування базового кольору, відбиття світла, шорсткості та рельєфу.

- Рендеринг. Процес перетворення тривимірної моделі або сцени на плоске зображення чи інтерактивний простір для відображення на екрані пристрою здобувача.

#### *Огляд інструментарію для розробки просторових ЦОР*

Для створення віртуальних виставок, 3D-турів та навчальних об'єктів у межах навчальної практики використовуються платформи, орієнтовані на швидку веброзробку (Artsteps, Vectary, Tinkercad):

- Tinkercad. Базове середовище від компанії Autodesk, яке використовує підхід конструювання з простих примітивів. Цей інструмент ідеально підходить для початкового етапу 3D-моделювання навчальних об'єктів та підготовки моделей до 3D-друку.

- Vectary. Потужний онлайн-інструмент для 3D-дизайну та доповненої реальності. Дозволяє створювати складні непланарні об'єкти, налаштовувати освітлення та миттєво генерувати AR-контент, який можна переглянути за допомогою камери смартфона (шляхом сканування QR-коду).

- Artsteps. Спеціалізована платформа, головним призначенням якої є створення віртуальних виставок. Здобувачі можуть конструювати архітектуру приміщення, розміщувати на стінах 2D-матеріали (фотографії, інфографіку, відео) та додавати тривимірні навчальні об'єкти в простір, створюючи повноцінні 3D-тури.

#### *Етапи створення віртуальної навчальної виставки*

Розробка комплексного просторового ресурсу зазвичай складається з таких логічних кроків:

1. Формування дидактичної концепції та збір необхідних мультимедійних матеріалів (зображень, текстів, аудіогідів).
2. Розробка з нуля або імпорт вже існуючих 3D-моделей навчальних об'єктів.
3. Побудова архітектури віртуального простору.
4. Розміщення контенту, налаштування інтуїтивної навігації та точок огляду.
5. Тестування працездатності, оцінка дидактичної ефективності та публікація ресурсу для доступу здобувачів.

### **Проектування складної логіки ЦОР.**

#### *Архітектура комплексного цифрового освітнього ресурсу*

Проектування складної логіки вимагає переходу від лінійного подання інформації (як у звичайних презентаціях чи відео) до створення динамічних систем, здатних реагувати на дії користувача. Архітектура такого ЦОР базується на моделі станів: система постійно відстежує поточний прогрес здобувача, аналізує введені ним дані та визначає, який контент або функціонал необхідно надати на наступному етапі.

#### *Розробка сценаріїв використання (Use Cases)*

Сценарій використання описує послідовність кроків, які здійснює здобувач для досягнення певної навчальної мети в межах ресурсу.

- Лінійні сценарії. Користувач проходить матеріал послідовно (Крок 1 -> Крок 2 -> Крок 3). Вони використовуються для базового ознайомлення з теорією.

- Розгалужені (нелінійні) сценарії. Шлях користувача залежить від його рішень. Наприклад, якщо здобувач робить помилку в тестовому блоці, логіка ЦОР автоматично перенаправляє його на додаткову сторінку з роз'ясненням теорії, а при правильній відповіді — відкриває доступ до складнішого рівня.

#### *Проектування інтерактивної структури*

Інтерактивна структура визначає, як саме користувацький інтерфейс взаємодіє з алгоритмами «під капотом» ресурсу. У складних ЦОР це вимагає ретельного планування змінних, тригерів та умов:

- Змінні (Variables). Використовуються для збереження даних під час сесії (наприклад, кількість набраних балів, обраний персонаж, зібрані елементи або час, витрачений на завдання).

- Умови (Conditions). Логічні оператори (If/Else), які перевіряють стан змінних. Наприклад, при розробці інтерактивного симулятора тривимірного графічного конвеєра умова може перевіряти введені дані: якщо здобувач змінює параметр шорсткості матеріалу для непланарної сітки, система має автоматично перерахувати відповідні етапи багатопрохідного рендерингу та оновити візуалізацію сцени.

- Тригери (Triggers). Події, що запускають виконання певних умов чи дій (натискання кнопки, зіткнення об'єктів у просторі, завершення таймера).

#### *Системи зворотного зв'язку (Feedback Loops)*

Важливою складовою складної логіки є проектування глибокого та змістовного зворотного зв'язку. На відміну від простої констатації «Правильно/Неправильно», комплексна логіка має аналізувати характер помилки. Наприклад, у віртуальному тренажері з написання коду алгоритм може розпізнати синтаксичну помилку та вивести підказку, що спрямовує користувача до правильного форматування, не даючи при цьому прямої відповіді.

#### *Інструменти для візуалізації логіки (Wireframing та Flowcharting)*

Перш ніж розпочинати безпосередню розробку або програмування в рушіях, логіку необхідно візуалізувати. Для цього використовують блок-схеми та ментальні карти (наприклад, у середовищах Miro, Draw.io або Figma). На таких схемах кожен екран або стан ресурсу позначається блоком, а можливі переходи між ними — стрілками, що супроводжуються описом умов переходу. Це дозволяє виявити логічні тупики, нескінченні цикли або бракуючі елементи навігації ще на етапі планування.

### **Практичне завдання №1**

#### **Верстка базової структури сторінки з використанням HTML та CSS; застосування основ JavaScript для створення простих інтерактивних елементів і тренажерів.**

**Інструментарій:** Редактор коду (Visual Studio Code, Sublime Text) або онлайн-середовище, сучасний веббраузер.

#### **Порядок виконання завдання:**

##### **1. Підготовка робочого середовища та структури файлів**

1. Створіть на комп'ютері нову папку для вашого проєкту (наприклад, edu\_practice\_task).

2. У цій папці створіть три порожні файли:
  - index.html (для структури);
  - style.css (для оформлення);
  - script.js (для логіки тренажера).

3. Відкрийте створену папку у вашому редакторі коду.

##### **2. Верстка базової HTML-структури**

Відкрийте файл index.html та створіть семантичний каркас сторінки. Зверніть увагу на підключення файлів стилів та скриптів.

## HTML

```
<!DOCTYPE html>
<html lang="uk">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Навчальний мінісайт-тренажер</title>
  <link rel="stylesheet" href="style.css">
</head>
<body>
  <header>
    <h1>Основи комп'ютерної графіки</h1>
    <p>Навчальний тренажер для перевірки знань</p>
  </header>

  <main>
    <section class="theory-block">
      <h2>Теоретичні відомості</h2>
      <p>Рендеринг – це процес отримання зображення за моделлю з
      допомогою комп'ютерної програми.</p>
    </section>

    <section class="practice-block">
      <h2>Тренажер: Перевірка термінології</h2>
      <div class="question-container">
        <p>Який термін означає процес побудови зображення за 3D-
        моделлю?</p>
        <input type="text" id="userAnswer" placeholder="Введіть
        відповідь...">
        <button onclick="checkAnswer()">Перевірити</button>
        <p id="feedbackMessage"></p>
      </div>
    </section>
  </main>

  <footer>
    <p>Розроблено в рамках навчальної практики</p>
  </footer>

  <script src="script.js"></script>
</body>
</html>
```

## Крок 3. Візуальне оформлення сторінки (CSS)

Відкрийте файл style.css та додайте стилі, щоб зробити мінісайт привабливим та зручним для читання.

## CSS

```
body {
  font-family: Arial, sans-serif;
  line-height: 1.6;
  margin: 0;
  padding: 0;
  background-color: #f4f4f9;
  color: #333;
}

header {
  background-color: #2c3e50;
  color: #fff;
  text-align: center;
  padding: 2rem 0;
}

main {
```

```

    max-width: 800px;
    margin: 20px auto;
    padding: 20px;
    background: #fff;
    border-radius: 8px;
    box-shadow: 0 4px 6px rgba(0,0,0,0.1);
}

.question-container {
    background-color: #e8f4f8;
    padding: 20px;
    border-left: 5px solid #3498db;
    border-radius: 4px;
    margin-top: 20px;
}

input[type="text"] {
    padding: 10px;
    width: 60%;
    border: 1px solid #ccc;
    border-radius: 4px;
}

button {
    padding: 10px 20px;
    background-color: #3498db;
    color: white;
    border: none;
    border-radius: 4px;
    cursor: pointer;
}

button:hover {
    background-color: #2980b9;
}

#feedbackMessage {
    font-weight: bold;
    margin-top: 15px;
}

```

#### 4. Програмування логіки тренажера (JavaScript)

Відкрийте файл `script.js` та напишіть функцію, яка зчитуватиме відповідь користувача, порівнюватиме її з правильною та виводитиме результат на екран.

##### JavaScript

```

function checkAnswer() {
    // Отримуємо значення, яке ввів користувач, та переводимо його в нижній
    реєстр
    let userAnswer = document.getElementById("userAnswer").value.toLowerCase().trim();

    // Отримуємо елемент для виведення результату
    let feedback = document.getElementById("feedbackMessage");

    // Правильна відповідь
    let correctAnswer = "рендеринг";

    // Логіка перевірки
    if (userAnswer === correctAnswer) {
        feedback.textContent = "Правильно! Відмінна робота.";
        feedback.style.color = "green";
    } else if (userAnswer === "") {
        feedback.textContent = "Будь ласка, введіть відповідь у поле.";
        feedback.style.color = "orange";
    } else {
        feedback.textContent = "Неправильно. Спробуйте ще раз!";
    }
}

```

```
        feedback.style.color = "red";
    }
}
```

### 5. Тестування та налагодження

1. Відкрийте файл index.html у браузері.
2. Перевірте коректність відображення всіх блоків.
3. Введіть правильну відповідь у текстове поле та натисніть "Перевірити".

Переконайтеся, що повідомлення зелене.

4. Введіть неправильну відповідь і перевірте, чи повідомлення стало червоним.

### Завдання для самостійного опрацювання

Після успішного виконання базової частини, модифікуйте ресурс:

- Змініть тематику сайту та тренажера на ту, яка стосується обраної вами фахової дисципліни.
- Додайте до HTML-коду зображення, що ілюструє ваше питання (використовуйте тег `<img>`).
- Ускладніть логіку JavaScript: додайте друге питання з вибором варіантів відповіді (використовуючи `<input type="radio">`) та налаштуйте підрахунок загальної кількості балів.

### Форма звітності

Здобувач має здати на перевірку викладачу:

- архів з файлами проєкту (HTML, CSS, JS) або посилання на збережений проєкт у хмарному середовищі.

## Практичне завдання №2

**Виконайте одне з завдань на вибір.**

### Варіант А.

**Тема:** Створення базової механіки мінігри з використанням візуального програмування у Construct 3.

**Інструментарій:** Сучасний веббраузер, доступ до мережі Інтернет (робота виконується у безкоштовній вебверсії рушія на сайті editor.construct.net).

### Порядок виконання завдання:

#### 1. Підготовка робочого простору

1. Відкрийте редактор Construct 3 у браузері.
2. Натисніть кнопку New Project (Новий проєкт).
3. Задайте ім'я проєкту (наприклад, Edu\_Platformer), оберіть стандартні налаштування (Viewport size: 1920x1080, Orientation: Landscape) та натисніть Create. Перед вами з'явиться робоча сцена (Layout 1).

#### 2. Створення ігрових об'єктів (Спрайтів)

Для прототипу створимо просту гру, де головний герой (наприклад, маг Рубік) має досліджувати рівень та збирати магічні руни.

1. Двічі клікніть лівою кнопкою миші на порожньому місці сцени та оберіть тип об'єкта Sprite (Спрайт).
2. Клікніть на сцені, щоб розмістити його. Відкриється вбудований графічний редактор. Намалуйте простий квадрат потрібного кольору або завантажте готове зображення персонажа. Закрийте вікно редактора.
3. У лівій панелі властивостей (Properties) перейменуйте цей об'єкт на Player.
4. Повторіть процедуру, щоб створити ще два спрайти:
  - Ground (Платформа) — зафарбуйте його та розтягніть внизу сцени як основу, по якій буде переміщуватися герой.
  - Rune (Предмет для збору) — об'єкт, який гравець має зібрати. Розмістіть кілька копій цього об'єкта у різних місцях на сцені.

5. Додайте текстовий інтерфейс: двічі клікніть на сцені, оберіть тип об'єкта Text. Переіменуйте його на ScoreText і розмістіть у верхньому лівому куті екрана.

### 3. Налаштування фізики та поведінок (Behaviors)

У Construct 3 більшість фізичних взаємодій реалізується без написання коду — за допомогою вбудованих поведінок.

1. Виділіть об'єкт Player. У панелі Properties знайдіть розділ Behaviors і натисніть *Add new behavior*.

2. У вікні, що з'явиться, оберіть Platform (це автоматично дасть персонажу гравітацію та керування стрілками клавіатури) та Scroll To (щоб камера завжди слідувала за гравцем).

3. Виділіть об'єкт Ground, перейдіть у Behaviors та додайте поведінку Solid (Твердий об'єкт). Це перетворить спрайт на фізичну перешкоду і не дозволить гравцю провалюватися крізь землю.

### 4. Програмування логіки (Event Sheet)

Тепер налаштуємо систему зникнення зібраних предметів та підрахунок балів.

1. У правій панелі Project двічі клікніть на Event sheet 1 (Аркуш подій).

2. Натисніть праву кнопку миші на порожньому місці та оберіть Add global variable (Додати глобальну змінну). Назвіть її Score із початковим значенням 0.

3. Створіть подію зіткнення:

- Натисніть Add event -> оберіть Player -> знайдіть умову On collision with another object -> виберіть об'єкт Rune.

4. Додайте дії (Actions) до цієї події:

- Натисніть Add action навпроти створеної події -> оберіть Rune -> дія Destroy (Знищити).

- Знову натисніть Add action -> оберіть System -> дія Add to -> виберіть змінну Score та впишіть значення 1.

- Натисніть Add action -> оберіть ScoreText -> дія Set text -> у полі тексту впишіть формулу: "Бали: " & Score.

### 5. Тестування та налагодження

1. Натисніть кнопку Preview (іконка Play у верхньому меню середовища).

2. Перевірте працездатність механік: персонаж повинен рухатися за допомогою клавіатурних стрілок, стрибати, стояти на платформі, а при торканні до рун — вони мають зникати з екрана, одночасно збільшуючи рахунок балів.

### Форма звітності

Здобувач має здати на перевірку викладачу:

- прототип гри безпосередньо у вікні браузера або надати експортований файл проекту у форматі .сзр.

### Варіант Б.

**Тема:** Розробка нелінійного освітнього сценарію у середовищі Twine.

**Інструментарій:** Сучасний веббраузер, безкоштовне онлайн-середовище розробки (сайт [twinery.org](http://twinery.org)).

#### Порядок виконання завдання:

##### 1. Підготовка робочого простору та створення проекту

1. Відкрийте браузер та перейдіть на сайт [twinery.org](http://twinery.org).

2. Натисніть кнопку Use in your browser (Використовувати в браузері).

3. На головному екрані натисніть + New (або + Story), щоб створити новий проект.

4. Задайте назву вашого освітнього квесту (наприклад, CyberSecurity\_Quest) та натисніть Create.

5. Перед вами відкриється робоча область (граф сцени), на якій вже створено перший стартовий вузол — Untitled Passage (Параграф).

##### 2. Формування стартового вузла та створення розгалужень

Twine автоматично створює нові сторінки (параграфи), коли ви додаєте гіперпосилання у квадратних дужках.

1. Двічі клікніть на стартовий вузол Untitled Passage. Переіменуйте його на Start.
2. У текстовому полі введіть вступну інформацію для здобувача. *Наприклад: "Ви — системний адміністратор. На сервер компанії здійснюється хакерська атака. Що ви зробите в першу чергу?"*
3. Створіть варіанти відповідей (гіперпосилання), взявши їх у подвійні квадратні дужки:
  - [[Відключити сервер від мережі Інтернет]]
  - [[Запустити антивірусне сканування]]
4. Закрийте вікно редагування параграфа. Ви побачите, що Twine автоматично створив два нових вузли з відповідними назвами та з'єднав їх зі стартовим вузлом стрілками.

### **3. Наповнення гілок сценарію контентом**

1. Двічі клікніть на новостворений параграф Відключити сервер від мережі Інтернет.
2. Додайте текст наслідків цього вибору: *"Сервер ізольовано. Атаку зупинено, але компанія втрачає гроші через простій. Переходимо до аналізу логів."*
3. Додайте посилання для продовження: [[Переглянути системні журнали->Аналіз логів]] (конструкція -> дозволяє зробити текст посилання одним, а назву наступного параграфа — іншою, більш зручною для вас).
4. Заповніть таким чином інші параграфи, формуючи дерево рішень.

### **4. Додавання ігрової логіки (Змінні та Умови)**

Щоб зробити квест складнішим, додамо систему підрахунку балів або інвентар (використовується стандартний формат Harlowe).

1. Поверніться до параграфа Start. У самому початку тексту задайте стартове значення змінної (наприклад, балів репутації) за допомогою макроса (set.):

(set: \$reputation to 10)

2. Перейдіть до параграфа Відключити сервер від мережі Інтернет і змініть значення змінної (віднімемо 2 бали за простій):

(set: \$reputation to it - 2)

3. Створіть новий фінальний параграф (наприклад, Final). Виведіть результат користувача на екран:

Ваша підсумкова репутація: \$reputation

4. Додайте у фінальний параграф умову (if:) для різного тексту залежно від балів:  
(if: \$reputation > 5)[Ви успішно впоралися з кризою!]  
(if: \$reputation <= 5)[Керівництво незадоволене вашими діями. Вам призначено перееатестацію.]

### **5. Тестування та публікація**

1. Щоб протестувати квест, натисніть кнопку Play (Відтворити) у нижньому правому куті вікна (або у верхньому меню, залежно від версії). Пройдіть квест різними шляхами, щоб перевірити коректність переходів та підрахунку змінних.
2. Для збереження результату натисніть на назву історії (в лівому нижньому або верхньому куті) і оберіть Publish to File (Опублікувати у файл).
3. Збережіть згенерований .html файл на свій комп'ютер.

### **Форма звітності**

Здобувач має здати на перевірку викладачу:

- готовий та протестований інтерактивний ресурс (освітню гру / квест) у вигляді збереженого .html файлу, який запускається автономно в будь-якому браузері.

### **Варіант В.**

**Тема:** Ознайомлення з інтерфейсом та створення простої сцени в Unity WebGL.

**Інструментарій:** Персональний комп'ютер, встановлений Unity Hub та Unity Editor із доданим модулем WebGL Build Support.

## **Порядок виконання завдання:**

### **1. Підготовка робочого середовища та створення проєкту**

1. Відкрийте програму Unity Hub.
2. Перейдіть у вкладку Projects та натисніть кнопку New project.
3. У списку шаблонів (Templates) оберіть 3D Core (або просто 3D). Це базовий шаблон, який ідеально підходить для розуміння роботи сучасного графічного конвеєра без зайвих надбудов.
4. У правій панелі задайте назву проєкту (наприклад, Edu\_Unity\_Scene) та оберіть папку для його збереження. Натисніть Create project. Процес ініціалізації може зайняти кілька хвилин.

### **2. Огляд інтерфейсу редактора**

Після завантаження перед вами з'явиться стандартне вікно Unity Editor, що складається з кількох ключових панелей:

- Hierarchy (Ієрархія) - відображає список усіх об'єктів, які зараз знаходяться на сцені. За замовчуванням тут є Main Camera (камера) та Directional Light (джерело світла).
- Scene (Сцена - основне вікно для візуального редагування 3D-простору. Використовуйте праву кнопку миші (ПКМ) + клавіші WASD для польоту по сцені.
- Game (Гра) - Відображає сцену так, як її буде бачити користувач через об'єкт Main Camera.
- Inspector (Інспектор) - Панель праворуч, яка показує властивості (компоненти) виділеного об'єкта.
- Project (Проєкт) - Файлова система вашого проєкту (текстури, моделі, скрипти).

### **3. Створення геометрії сцени**

1. У панелі Hierarchy натисніть ПКМ на порожньому місці, оберіть 3D Object -> Plane (Площина). Вона слугуватиме підлогою для нашої сцени.
2. Виділіть створений Plane. В панелі Inspector знайдіть компонент Transform. Переконайтеся, що координати Position встановлені на X: 0, Y: 0, Z: 0 (за необхідності натисніть іконку трьох крапок біля Transform та оберіть Reset).
3. Знову натисніть ПКМ у Hierarchy -> 3D Object та додайте кілька базових фігур: Cube (Куб), Sphere (Сферу) та Cylinder (Циліндр).
4. Використовуючи інструменти на верхній панелі (Move, Rotate, Scale) або гарячі клавіші W, E, R, розмістіть об'єкти на площині так, щоб вони утворили композицію і не перетиналися між собою.

### **4. Налаштування матеріалів та параметрів рендерингу**

Для того, щоб об'єкти не були однаково сірими, необхідно створити матеріали.

1. У панелі Project (внизу екрана) перейдіть у папку Assets. Натисніть ПКМ -> Create -> Material. Назвіть його, наприклад, Mat\_Red.
2. Виділіть Mat\_Red. В Inspector знайдіть параметр Albedo (колір) і змініть його на червоний.
3. Зверніть увагу на параметри Metallic та Smoothness у налаштуваннях матеріалу. Параметр Smoothness визначає гладкість поверхні і є математично оберненим до параметру шорсткості (roughness). Змінюючи його повзунок від 0 до 1, ви можете спостерігати, як змінюється відблиск на матеріалі від матового до дзеркального. Базове налаштування цих PBR-матеріалів є основою для розуміння етапів рендерингу освітлення.
4. Перетягніть створений матеріал Mat\_Red безпосередньо на один з об'єктів у вікні Scene або на його назву в Hierarchy.
5. Створіть ще кілька матеріалів з різними кольорами та різними показниками гладкості/шорсткості і застосуйте їх до інших об'єктів.

### **5. Налаштування освітлення та камери**

1. Виділіть Directional Light в Hierarchy. В Inspector за допомогою інструменту Rotate (E) змініть кут нахилу світла, спостерігаючи, як змінюються тіні від ваших об'єктів.
2. Виділіть Main Camera. Спозиціонуйте її так, щоб ваша композиція виглядала виразно.

*Підказка:* Ви можете налаштувати ідеальний ракурс у вікні Scene, виділити камеру, а потім у верхньому меню натиснути GameObject -> Align With View (гарячі клавіші Ctrl+Shift+F). Камера автоматично переміститься туди, куди ви зараз дивитесь.

## **6. Збірка (Експорт) під WebGL**

Щоб вашу сцену можна було відкрити у звичайному браузері, необхідно змінити цільову платформу.

1. У верхньому меню оберіть File -> Build Settings.
2. У списку Platforms оберіть WebGL.
3. Натисніть кнопку Switch Platform (у правому нижньому куті вікна). Процес конвертації ресурсів може тривати певний час.
4. Після завершення натисніть кнопку Add Open Scenes, щоб додати поточну сцену до збірки.
5. Натисніть Build. Unity попросить обрати або створити нову папку на комп'ютері для збереження готових файлів (назвіть її Build\_WebGL).
6. Після завершення компіляції відкрийте цю папку. Ви побачите файл index.html та супутні папки, які формують ваш інтерактивний вебзастосунок.

## **Форма звітності**

Здобувач має здати на перевірку викладачу:

- експортовану сцену, що працює у веббраузері (запустити файл index.html локально або завантажити на безкоштовний хостинг), а також вміти пояснити призначення основних панелей інтерфейсу редактора Unity.



## ІНДИВІДУАЛЬНЕ ЗАВДАННЯ НА ПРАКТИКУ

Самостійне проектування та розробка одного комплексного цифрового освітнього ресурсу (ЦОР) для обраної теми з профільної фахової дисципліни.

### Порядок виконання та вимоги:

1. Вибір фахової теми. Здобувач освіти самостійно аналізує програму свого напряму підготовки, обирає конкретну тему з галузі інформаційних технологій та визначає дидактичну мету майбутнього цифрового продукту (подання нового матеріалу, перевірка знань, відпрацювання практичних навичок).

2. Вибір формату ресурсу. Залежно від поставленої мети та специфіки обраної теми, здобувач має обрати один із таких форматів для розробки:

1. веброзробка навчальних мінісайтів, тренажерів, інтерактивних сторінок (HTML, CSS, JavaScript);
2. розробка освітніх ігор (Construct 3, Twine, Unity WebGL);
3. створення електронних підручників або навчальних модулів з мультимедіа і тестами (H5P, Book Creator);
4. віртуальні виставки, 3D-тури, навчальні об'єкти (Artsteps, Vectary, Tinkercad).

3. Проектування та розробка. Здобувач здійснює проектування логіки ресурсу, сценаріїв використання та інтерактивної структури. Після підготовки контенту (текстів, графіки, аудіо, відео чи 3D-моделей) відбувається безпосередня технічна реалізація ресурсу в обраному програмному середовищі.

### Приклад реалізації завдання:

Для теми, що стосується алгоритмів багатопрхідного рендерингу та функціонування графічного конвеєра, здобувач може розробити інтерактивний 3D-тренажер. У такому ресурсі користувач матиме змогу власноруч змінювати параметри шорсткості матеріалів для непланарних сіток і в реальному часі спостерігати, як ці налаштування впливають на фінальну візуалізацію тривимірної сцени. Альтернативним форматом для цієї ж фахової теми може стати комплексний електронний підручник із вбудованими тестами для перевірки знань про етапи рендерингу.

### Порядок виконання індивідуального завдання

**Аналітика та планування. Вибір теми з галузі інформаційних технологій. Визначення типу ресурсу, цільової аудиторії та дидактичної мети. Формування технічного завдання.**

#### 1) Аналіз та вибір теми

1. Вибір теми з галузі інформаційних технологій  
2. Оберіть одну вузьку, але змістовну тему, яка, на вашу думку, потребує кращої візуалізації, інтерактивного відпрацювання або гейміфікації для легшого засвоєння.

#### 2) Визначення цільової аудиторії та дидактичної мети

- Цільова аудиторія. Чітко опишіть, для кого створюється цей ресурс (наприклад: здобувачі 1-го курсу, учні старших класів, фахівці для підвищення кваліфікації). Врахуйте їхній базовий рівень знань.

- Дидактична мета. Дайте відповідь на запитання: «Що саме має засвоїти, зрозуміти або навчитися робити користувач після взаємодії з вашим ЦОР?» (наприклад: завчити термінологію, відпрацювати алгоритм дій, зрозуміти просторову будову об'єкта).

#### 3) Вибір типу цифрового освітнього ресурсу

Спираючись на сформульовану мету, оберіть найбільш релевантний формат ЦОР:

- Для подання теорії та комплексної перевірки: електронний підручник, навчальний мінісайт.

- Для запам'ятовування термінів чи фактів: інтерактивна гра-вікторина, вебтренажер.

- Для вивчення просторових об'єктів: 3D-тур, віртуальна виставка.

- Для моделювання ситуацій: текстовий квест з розгалуженим сценарієм.

#### 4) Формування технічного завдання)

1. Створіть текстовий документ (у форматі Word або Google Docs), який міститиме такі пункти:

2. Назва проєкту (робоча назва вашого ресурсу).

3. Тема (наприклад: Тривимірна графіка, тема "Базові примітиви та рендеринг").

4. Цільова аудиторія та мета (опис з Кроку 2).

5. Формат та інструментарій (наприклад: 2D-платформер, розроблений у Construct 3).

6. Орієнтовна структура ресурсу (короткий перелік розділів сайту, ігрових рівнів або сторінок посібника).

7. Перелік необхідного контенту (тексти, які потрібно написати; зображення/моделі, які потрібно знайти або створити).

#### 5) Вимоги до звітності

Готове технічне завдання має бути погоджене з викладачем-керівником практики. Затверджене ТЗ є допуском до наступного етапу роботи — проєктування архітектури та підготовки бази знань.

### **Архітектура проєкту. Проєктування структури обраного ресурсу: створення ескізів, блок-схем, макетів інтерфейсу (Wireframing) або розкадрування сценарію.**

#### 1) Вибір інструментарію для візуалізації

- Для створення блок-схем та графів логіки оберіть зручне цифрове середовище (наприклад, Miro, Draw.io).

- Для розробки макетів інтерфейсу підготуйте профільне програмне забезпечення (Figma) або використовуйте звичайний папір для чорнових ескізів від руки.

#### 2) Побудова алгоритмічної блок-схеми або графа логіки

1. Створіть стартовий блок, що символізує головний екран або початок роботи з ресурсом.

2. Промалюйте всі можливі шляхи користувача. Наприклад, якщо проєктується інтерактивний симулятор роботи графічного конвеєра, блок-схема має відображати логічні умови та переходи між етапами обробки тривимірної сцени, багатопрохідного рендерингу та фінального виведення зображення.

3. Позначте на схемі точки прийняття рішень (тестові запитання, вибір сюжетної гілки) та їхні наслідки для подальшої навігації.

#### 3) Створення макетів інтерфейсу

1. Розробіть структурні ескізи для кожного ключового екрана вашого майбутнього продукту.

2. Схематично розмістіть основні елементи: навігаційне меню, текстові блоки, місця для мультимедійного контенту, кнопки управління та інтерактивні зони.

3. Не використовуйте на цьому етапі кольори, складну графіку чи фінальні шрифти — зосередьтеся виключно на ергономіці та логіці розташування елементів.

#### 4) Розкадрування сценарію (для ігор, квестів та відео)

1. Якщо обраний формат передбачає розгорнутий сюжет (наприклад, текстовий квест, анімація або освітня гра), створіть розкадрування (Storyboard).

2. Замалюйте послідовність ключових сцен, супроводивши кожен кадр описом того, що відбувається на екрані та які дії очікуються від користувача.

#### 5) Вимоги до звітності

Результатом виконання завдання є підготовлений пакет проєктної документації, який необхідно погодити з керівником практики.

### **Підготовка бази знань. Збір, генерація та структурування контенту: написання текстів, підбір графіки, створення чи пошук мультимедіа та необхідних моделей.**

#### 1) Збір та структурування текстового контенту

1. Спираючись на затверджене технічне завдання та розроблену архітектуру, зберіть необхідний теоретичний матеріал.

2. Адаптуйте текст для сприйняття з екрана (принцип мікронавчання): розбийте суцільний текст на короткі абзаци, використовуйте марковані списки та виділяйте ключові терміни.

Приклад: Якщо ресурс стосується комп'ютерної графіки, підготуйте лаконічні текстові блоки з описом стадій графічного конвеєра, алгоритмів багатопрохідного рендерингу або правил роботи з непланарними сітками.

#### *2) Підбір та створення 2D-графіки*

1. Створіть власні графічні матеріали (інтерактивні презентації, інфографіки, ментальні карти, постери) за допомогою таких інструментів, як Canva, Genially або Miro.

2. Якщо ви використовуєте готові зображення або фотографії з Інтернету, переконайтеся, що вони мають відповідну ліцензію (Creative Commons) для використання в освітніх цілях.

3. Оптимізуйте розмір та роздільну здатність зображень для швидкого завантаження у вебсердовищі.

#### *3) Підготовка аудіовізуальних продуктів (за потреби)*

1. Для створення мультимедійного супроводу запишіть власні аудіофайли (подкасти, голосові пояснення) та обробіть їх у редакторі Audacity.

2. За необхідності змонтуйте навчальні відеоуроки чи анімації, використовуючи середовища Clipchamp, Powtoon або Animaker.

#### *4) Пошук та розробка 3D-моделей (для віртуальних середовищ та ігор)*

1. Якщо ваш ЦОР передбачає роботу у тривимірному просторі, підготуйте необхідні просторові об'єкти. Ви можете створити їх власноруч у базових редакторах (Tinkercad, Vectary) або у професійних пакетах на кшталт Blender.

2. Під час створення моделей звертайте увагу на оптимізацію полігональної сітки та налаштування базових матеріалів (колір, шорсткість) для коректного експорту у вебрушії.

3. Альтернативно, знайдіть готові оптимізовані 3D-моделі на спеціалізованих стоках.

#### *5) Систематизація бази знань*

1. Створіть на своєму комп'ютері або у хмарному сховищі (Google Drive) єдину директорію проєкту.

2. Розподіліть зібраний контент за логічними папками (наприклад: Тексти, Зображення, Аудіо, Моделі).

3. Назвіть файли зрозуміло і латинськими літерами (наприклад, rendering\_scheme.png замість Знімок екрана 1.png). Це попередить помилки під час їхньої інтеграції в HTML-код чи ігрові рушії.

#### *б) Вимоги до звітності*

Для отримання оцінки за заняття здобувач має продемонструвати викладачу-керівнику систематизовану базу знань (папку з підготовленими файлами та текстовими документами), яка повністю покриває потреби розробленої на попередньому етапі архітектури проєкту.

**Розробка прототипу (Альфа-версія). Налаштування базового середовища: створення каркаса мінісайту, базової сцени в рушії, основи електронного підручника чи віртуального простору.**

#### *1) Ініціалізація проєкту та налаштування середовища*

1. Запустіть обране середовище розробки (редактор коду, ігровий рушії, онлайн-платформу).

2. Створіть новий проєкт і задайте базові налаштування: роздільну здатність екрана, орієнтацію (альбомна/книжкова), цільову платформу (наприклад, WebGL для 3D-проєктів).

3. Створіть структуру папок для майбутнього контенту (якщо розробка ведеться локально), щоб забезпечити порядок в асетах.

## 2) Створення базового каркаса (відповідно до формату ЦОР)

Залежно від обраного типу ресурсу, реалізуйте його «чорнову» структуру (Blockout):

- Для мінісайтів та вебтренажерів. Створіть базові файли (HTML, CSS, JS). Пропишіть семантичний каркас HTML-сторінки (теги <header>, <main>, <nav>, <footer>) та створіть порожні блоки (<div>) для майбутнього тексту, зображень і тестових форм.

- Для електронних підручників (Book Creator, H5P). Створіть титульну сторінку, зміст та необхідну кількість порожніх розділів/слайдів згідно з вашим розкладуванням.

- Для ігрових рушіїв та 3D-середовищ (Unity, Construct 3, Artsteps). Створіть стартову сцену. Розмістіть базові примітиви (куби, площини), що позначатимуть межі локації або інтерактивні зони. Налаштуйте головну камеру та базове освітлення. Наприклад, якщо ваш ресурс моделює роботу графічного конвеєра, на цьому етапі достатньо розмістити прості форми, які позначатимуть вузли проходження алгоритму, без застосування багатопрохідного рендерингу, деталізації сіток чи складних матеріалів.

### 3) Налаштування базової навігації та переходів

1. Зв'яжіть створені порожні сторінки або сцени між собою.

2. Додайте тимчасові кнопки-заглушки ("Вперед", "Назад", "Меню") та налаштуйте для них найпростіші тригери переходу.

3. Ваша мета на цьому етапі — зробити так, щоб користувач міг "проклікати" весь ресурс від початку до кінця, перевіривши логіку розгалужень, навіть якщо самі сторінки ще порожні.

### 4) Тестування альфа-версії

1. Запустіть створений каркас у режимі попереднього перегляду або скомпілюйте тестовий білд.

2. Перевірте, чи коректно працюють переходи між усіма вузлами ресурсу та чи не виникає критичних помилок під час завантаження базової сцени.

### 5) Вимоги до звітності

Для отримання оцінки за поточне заняття здобувач має продемонструвати викладачу-керівнику практики працездатний "скелет" (альфа-версію) свого проекту, де реалізована структура та базові переходи, готові до подальшої інтеграції текстового, графічного та мультимедійного контенту.

**Інтеграція контенту. Наповнення розробленої оболонки підготовленим матеріалом. Налаштування візуального відображення та базового дизайну сторінок чи сцен.**

#### 1) Робота з текстовим наповненням

1. Відкрийте ваш робочий проєкт (каркас мінісайту, базову сцену в рушії або макет електронного підручника).

2. Перенесіть підготовлені тексти з вашої бази знань у відповідні блоки розробленої оболонки.

3. Здійсніть базове форматування тексту (виділіть заголовки ієрархічними тегами (для веброзробки) або відповідними стилями, налаштуйте читабельні шрифти, міжрядкові інтервали та відступи).

#### 2) Інтеграція мультимедіа та 2D-графіки

1. Додайте у проєкт підготовлені зображення, інфографіку, схеми та ментальні карти.

2. Інтегруйте аудіовізуальні продукти (навчальні відео, подкасти, анімації) у відповідні розділи ресурсу.

3. Обов'язково пропишіть альтернативний текст (alt-текст) для всіх графічних елементів, щоб забезпечити інклюзивність та доступність вашого освітнього продукту.

#### 3) Налаштування 3D-сцен та віртуальних об'єктів (залежно від формату ЦОР)

1. Якщо ваш ресурс реалізується у тривимірному просторі, імпортуйте оптимізовані 3D-моделі у створену базову сцену.

2. Налаштуйте матеріали об'єктів для коректного графічного відображення. Зверніть особливу увагу на параметри шорсткості (roughness) та відбиття світла, що є критично важливим при роботі з непланарними сітками для досягнення реалістичності навчальних об'єктів.

3. Спозиціонуйте джерела світла та налаштуйте камери для оптимального огляду інтегрованих моделей.

#### *4) Базовий дизайн та стилізація*

1. Застосуйте технологічні рішення для візуального оформлення ресурсу. Для мінісайтів це передбачає роботу з CSS-стилями (налаштування кольорової палітри, фонів, рамок, тіней).

2. Для ігрових рушіїв налаштуйте елементи інтерфейсу користувача (UI Canvas): кнопки меню, панелі прогресу, вікна з текстом.

3. Перевірте контрастність тексту відносно фону та загальну естетичну привабливість сторінок або сцен.

#### *5) Вимоги до звітності*

Для отримання оцінки за поточне заняття здобувач має продемонструвати викладачу-керівнику практики Бета-версію проєкту — повністю наповнений контентом та візуально оформлений ресурс, який готовий до фінального етапу програмування складної логіки та інтерактивності.

**Налаштування інтерактивності. Програмування логіки роботи ресурсу: додавання тригерів, гіперпосилань, анімацій, тестових модулів або ігрових механік.**

#### *1) Реалізація внутрішньої та зовнішньої навігації*

1. Замініть тимчасові кнопки-заглушки, створені на етапі альфа-версії, на повноцінні навігаційні елементи.

2. Налаштуйте гіперпосилання для переходу між сторінками мінісайту, параграфами текстового квесту або сценами ігрового рушія.

3. Переконайтеся, що користувач має можливість у будь-який момент повернутися до головного меню або на попередній крок (унікайте "навігаційних тупиків").

#### *2) Додавання тригерів та обробка подій*

1. Визначте ключові події (дії користувача), на які має реагувати ваш ресурс: клік мишкою, наведення курсора (hover), натискання клавіші або перетягування об'єкта (Drag-and-Drop).

2. Запрограмуйте відповідні тригери. Наприклад, за допомогою JavaScript додайте обробники подій для кнопок, а в Construct 3 або Unity налаштуйте події зіткнення об'єктів (Collisions) чи взаємодії з UI-елементами.

#### *3) Інтеграція контрольно-оцінювальних модулів*

1. Якщо ваш ЦОР містить тестовий блок або тренажер, налаштуйте логіку перевірки відповідей користувача.

2. Реалізуйте систему підрахунку балів, використовуючи змінні.

3. Налаштуйте алгоритм зворотного зв'язку: виведення повідомлень про правильну чи неправильну відповідь, надання дидактичних підказок або автоматичне перенаправлення на сторінку з теорією у разі допущення помилки.

#### *4) Налаштування анімацій та специфічних ігрових механік*

1. Для підвищення рівня залученості додайте мікроанімації: плавна поява тексту, зміна кольору або масштабу кнопки при наведенні, анімація правильної відповіді.

2. Якщо ви розробляєте освітню гру, налаштуйте ключові механіки: систему "спроб", таймери на виконання завдання, збір навчальних артефактів на рівні або відкриття нових локацій.

#### *5) Проміжне налагодження*

1. Протестуйте створену логіку взаємодії: свідомо введіть неправильні відповіді в тести, спробуйте перейти за всіма можливими посиланнями та взаємодіяти з усіма об'єктами.

2. Відстежте та виправте логічні помилки (наприклад, коли бали не додаються, перехід здійснюється на хибну сторінку або кнопка не реагує на натискання).

#### *б) Вимоги до звітності*

Для отримання оцінки за поточне заняття здобувач має продемонструвати викладачу-керівнику практики інтерактивну версію свого проєкту, де реалізована базова логіка взаємодії: працюють переходи, кнопки реагують на дії користувача, а тестові чи ігрові модулі коректно обробляють вхідні дані.

**Внутрішнє тестування. Перевірка працездатності всіх елементів ресурсу. Виявлення логічних помилок, невідповідностей у сценарії, некоректних посилань або багів.**

#### *1) Функціональне тестування*

1. Запустіть ваш цифровий освітній ресурс у робочому середовищі (браузері).
2. Послідовно перевірте всі інтерактивні елементи: кожна кнопка має натискатися, кожне вікно — відкриватися та закриватися.
3. Здійсніть перевірку всіх внутрішніх та зовнішніх гіперпосилань. Жоден перехід не повинен вести на неіснуючу сторінку (помилка 404) або спричиняти зависання ресурсу.
4. Якщо ви розробили просторовий ресурс, перевірте коректність завантаження всіх 3D-моделей, відображення текстур та роботу камери (відсутність провалів камери крізь текстури чи геометрію).

#### *2) Перевірка логіки та розгалужених сценаріїв*

1. Пройдіть ресурс від початку до кінця кількома різними шляхами. Це особливо важливо для нелінійних квестів або інтерактивних симуляторів.
2. Перевірте систему підрахунку балів або змінних: чи коректно нараховуються бали за правильні відповіді та чи віднімаються/не зараховуються за хибні.
3. Перевірте логічні умови (If/Else). Наприклад: якщо здобувач у симуляторі графічного конвеєра ввів хибне значення для параметру шорсткості матеріалу, система має видати відповідну підказку, а не просто закрити програму чи пропустити його на наступний рівень.

#### *3) Тестування граничних випадків*

1. Спробуйте поведися як "непередбачуваний користувач". Введіть у текстові поля для відповідей нетипові дані (літери замість цифр, спецсимволи, занадто довгий текст, порожній рядок).
2. Швидко клікайте по кнопках переходу або відповідях. Переконайтеся, що система захищена від багаторазових спрацьовувань одного й того ж тригера.
3. Протестуйте механіку "відміни" або повернення на попередній крок — чи не збивається при цьому загальний прогрес і таймери.

#### *4) Тестування контенту та UI*

1. Вичитайте всі тексти на предмет орфографічних, пунктуаційних та стилістичних помилок.
2. Перевірте верстку: текст не має виходити за межі блоків або наповзати на зображення при зміні розміру вікна браузера.
3. Оцініть читабельність контенту та контрастність шрифтів.

#### *5) Документування знайдених помилок*

1. У процесі тестування не намагайтеся відразу виправляти все. Спочатку зафіксуйте знайдені проблеми.
2. Створіть текстовий документ та записуйте помилки за таким алгоритмом:
  - Де знаходиться помилка (екран/сцена/розділ).
  - Опис проблеми (що саме не працює).
  - Кроки для відтворення (як повторити цю помилку).
  - Очікуваний результат (як це мало б працювати в ідеалі).

#### *б) Вимоги до звітності*

Для отримання оцінки за поточне заняття здобувач має продемонструвати викладачу-керівнику практики заповнений журнал тестування із переліком знайдених під час самоперевірки помилок та запропонувати шляхи їх усунення на наступному етапі оптимізації.

### **Оптимізація та адаптація. Виправлення виявлених недоліків. Оптимізація ресурсу для стабільної роботи та коректного відображення на цільових платформах.**

#### *1) Аналіз журналу тестування та виправлення помилок*

1. Відкрийте журнал тестування, заповнений на попередньо.
2. Пріоритезуйте завдання: розпочніть роботу з усунення критичних помилок (непрацюючі гіперпосилання, фатальні збої під час завантаження, порушення логіки підрахунку балів).
3. Після виправлення критичних багів перейдіть до візуальних та текстових недоліків (виправлення друкарських помилок, налаштування відступів).
4. Після кожного вагомого виправлення обов'язково перевіряйте працездатність пов'язаних елементів, щоб переконатися, що внесені зміни не зламали інший функціонал (регресійне тестування).

#### *2) Оптимізація контенту (2D, аудіовізуального та 3D)*

- Для вебресурсів. Перевірте "вагу" вашого ресурсу. Стисніть усі зображення (рекомендовані формати WebP або оптимізований JPEG) та аудіофайли. Навчальні відеоролики краще не інтегрувати безпосередньо у проєкт, а підвантажувати через посилання з відеохостингів (наприклад, YouTube), щоб зменшити час завантаження сторінки.
- Для просторових та ігрових ЦОР. Оптимізуйте ресурсомісткі елементи. Наприклад, якщо ваш проєкт — це інтерактивний тренажер для вивчення графічного конвесра, зверніть особливу увагу на оптимізацію геометрії: зменште кількість полігонів для складних непланарних сіток, стисніть розмір текстур матеріалів та оптимізуйте алгоритми багатопрохідного рендерингу. Видаліть зі сцени всі приховані об'єкти та скрипти, які не використовуються у фінальній роботі.

#### *3) Адаптація інтерфейсу (UI/UX Scaling)*

- Мінісайти та електронні посібники. Перевірте адаптивність верстки за допомогою інструментів розробника у браузері (F12 -> Device Toolbar). За необхідності допишіть CSS-медіазапити (@media), щоб текст, зображення та кнопки коректно масштабувалися і не виходили за межі екрана на мобільних телефонах і планшетах.
- Ігрові рушії. Налаштуйте параметри масштабування користувацького інтерфейсу (наприклад, Canvas Scaler в Unity). Переконайтеся, що тестові вікна, панелі з балами та кнопки навігації не перекривають важливий навчальний контент при зміні пропорцій екрана.

#### *4) Фінальний білд та профілювання продуктивності*

1. Скомпілюйте оновлену версію вашого ЦОР.
2. Запустіть фінальну збірку на цільовій платформі (у браузері локально або на хостингу).
3. Оцініть швидкість завантаження та частоту кадрів (FPS) під час активної взаємодії з ресурсом. Продукт має працювати плавно, без відчутних затримок.

#### *5) Вимоги до звітності*

Для отримання оцінки за поточне заняття здобувач має продемонструвати викладачу-керівнику практики оптимізовану версію ресурсу, яка працює стабільно, швидко завантажується та має адаптивний дизайн. Також необхідно надати звіт про виправлені недоліки.

### **Фінальне доопрацювання проєкту.**

#### *1) Фінальне доопрацювання та пакування ресурсу*

1. Здійсніть контрольний перегляд вашого цифрового освітнього ресурсу (ЦОР). Переконайтеся, що всі виявлені під час внутрішнього тестування помилки усунуті, а продукт працює стабільно.

2. Підготуйте фінальну версію для здачі: здобувачі здають підготовлені за час практики цифрові освітні ресурси у форматі, передбаченому технічним завданням (посилання на опублікований вебсайт, експортований архів 3D-туру, скомпільований файл гри або підготовлений мультимедійний проєкт).

## **ЗАХИСТ ПРАКТИКИ ТА ВИМОГИ ДО ЗВІТНОЇ ДОКУМЕНТАЦІЇ**

Виконання індивідуальних завдань є найважливішим етапом роботи здобувача і становить 50% від загального розподілу підсумкової оцінки за практику.

На всіх етапах проходження практики здобувачі отримують індивідуальні консультації у своїх викладачів-керівників з метою виявлення недоліків та надання практичної допомоги у процесі виконання завдання.

Під час проходження практики обов'язково відбувається захист індивідуальних практичних завдань. Під час захисту (на 20-му занятті) здобувач має презентувати свій освітній мікросайт, продемонструвати роботу всіх вбудованих ЦОР та обґрунтувати їх дидактичну ефективність і відповідність обраній фаховій темі. Після цього здобувачі надають на кафедрі відповідну звітну документацію (Додаток А).

### **Підготовка та презентація індивідуальних практичних завдань**

Підготуйте коротку усну доповідь (на 3–5 хвилин), яка включає:

- Актуальність. Чому обрана вами тема є важливою для вивчення?
- Структура. Які саме ЦОР ви розробили та як вони пов'язані між собою (демонстрація логіки ментальної карти з Miro).
- Демонстрація. Короткий показ "шляху користувача" на вашому мікросайті (від вступу до контролю знань).
- Ефективність. Як ваші цифрові інструменти допомагають краще засвоїти складні аспекти теми?

Презентація та дискусія:

- Під час заняття презентуйте свої результати виконання індивідуального завдання.
- Під час демонстрації необхідно запустити свій продукт і показати його ключові можливості. Наприклад, якщо презентується розроблений 3D-тренажер або віртуальна лабораторія, варто наживо продемонструвати логіку взаємодії: як користувач може змінювати параметри шорсткості матеріалів на непланарних сітках та як це впливає на багатопрохідний рендеринг сцени.
- Будьте готові відповісти на запитання колег та викладача щодо технічних рішень (наприклад, чому обрали саме цей інструмент для тестування) та дидактичного наповнення.
- Візьміть активну участь в обговоренні проєктів інших здобувачів — конструктивна критика є частиною вашої професійної підготовки.
- Акцентуйте увагу на тому, як застосовані сучасні технології програмування, мультимедіа та інтерактивного дизайну здатні підвищити ефективність навчального процесу.

### **Оформлення звітної документації**

Переконайтеся, що всі посилання (на Miro, сайти, інтерактивні вправи, відео) є робочими та відкритими для перегляду. Сформуйте фінальний перелік (або звітний файл), де впорядковано наведені всі посилання на ваші напрацювання за заняттями. Підготуйте звіт за шаблоном (Додаток А).

## СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ ТА ЛІТЕРАТУРИ

1. Allen M. W. Michael Allen's guide to e-learning: Building interactive, fun, and effective learning programs for any company. John Wiley and Sons Ltd, 2016. 406 p.
2. Авраменко В.С. Технологія програмування та створення програмних продуктів. Електронний конспект лекцій. ЧНУ, 2017. 192 с.
3. Алексенко О. В. Технології програмування та створення програмних продуктів: конспект лекцій. Суми : Сумський державний університет, 2020. 133 с.
4. Амеліна О. С., Цуркан О. В. Дистанційне та змішане навчання. Досвід, поради, інструменти. Київ : Основа, 2022. 128 с.
5. Басюк Т.М., Думанський Н.О., Пасічник О.В. Основи інформаційних технологій. Серія «Комп'ютинг» : навч. посібник. Львів : Новий Світ 2000. 2020. 392 с.
6. Биков В. Ю. Цифрова трансформація суспільства і розвиток комп'ютерно-технологічної платформи освіти і науки України. Інформаційно-цифровий освітній простір України: трансформаційні процеси і перспективи розвитку : методологічний семінар НАПН України (м. Київ, 4 квітня 2019 р.). 2019. С. 20-26.
7. Буйницька О. Інформаційні технології та технічні засоби навчання: навч. посібник. К: Центр навчальної літератури. 2018. 240 с.
8. Буйницька О. П. Інформаційні технології та технічні засоби навчання: навчальний посібник / Оксана Буйницька ; М-во освіти і науки, молоді та спорту України, Київський ун-т ім. Бориса Грінченка. Київ : Центр учбової літератури, 2012. 239 с.
9. Вовкодав, О. В. Сучасні інформаційні технології : навч. посіб. / О. В. Вовкодав, Х.В. Лип'яніна. Тернопіль : ТНЕУ, 2017. 550 с.
10. Войтович Н.В., Найдьонова А.В. Використання хмарних технологій Google та сервісів web 2.0 в освітньому процесі. Методичні рекомендації. Дніпро: ДПТНЗ «Дніпровський центр ПТОТС», 2017. 113 с.
11. Глинський Я. М. Інформатика: практикум з інформаційних технологій. Тернопіль : Підруч. і посіб., 2014. 302 с.
12. Гуревич Р.С., Кадемія М. Ю., Шевченко Л. С. Інформаційні технології навчання: інноваційний підхід : навчальний посібник. Вінниця : ТОВ фірма «Планер», 2013. 348 с.
13. Деякі питання організації дистанційного навчання : наказ Міністерства освіти і науки від 08.09.2020 № 1115. URL: <https://zakon.rada.gov.ua/laws/show/z0941-20#Text>.
14. Дистанційне навчання: дидактика, методика, організація: монографія / В. Г. Гетта, та ін. Чернівці, 2017. 286 с.
15. Дистанційне та змішане навчання в школі. Путівник / упоряд. І. П. Воротникова. Київ : Київ. ун-т ім. Б. Грінченка, 2020. 48 с.
16. Інформаційні технології: метод. реком. до практ. та лаб. занять / Уклад.: Я. Р. Сікора, А. Л. Федорчук. Житомир: Вид-во ЖДУ ім. І. Франка, 2023. 90 с.
17. Інформаційно-освітнє середовище професійно-технічних навчальних закладів: посібник / Карташова Л. А., Юрженко В. В., Гуралюк А. Г., Липська Л. В., Гуменна Л. С., Зуєва А. Б., Шупік І. М., Ростока М. Л., Шевченко В. Л. За наук. ред. Лузана П. Г. - Київ: ПІТО НАПН, 2017. - 124 с.  
[https://lib.iitta.gov.ua/709621/1/%D0%9F%D0%BE%D1%81%D1%96%D0%B1%D0%BD%D0%B8%D0%BA\\_%D0%86%D0%9E%D0%A1\\_%D0%9F%D0%A2%D0%9D%D0%97\\_%D0%BD%D0%BE%D0%B2%D0%B8%D0%B9.pdf](https://lib.iitta.gov.ua/709621/1/%D0%9F%D0%BE%D1%81%D1%96%D0%B1%D0%BD%D0%B8%D0%BA_%D0%86%D0%9E%D0%A1_%D0%9F%D0%A2%D0%9D%D0%97_%D0%BD%D0%BE%D0%B2%D0%B8%D0%B9.pdf)
18. Кадемія М. Ю., Шестопалюк О. В., Кобися В.М. Використання сервісів соціальних медіа в навчальному процесі ВНЗ: Блоги, Веб-квести, Блог-квести: навчально-методичний посібник (видання 2-е, доповнене). Вінниця : ТОВ «Ландо ЛТД», 2014. 236 с.
19. Карпенко М. Ю., Манакова Н. О., Гавриленко І. О. Технології створення програмних продуктів та інформаційних систем : навч. посібник. Харків : ХНУМГ ім. О. М. Бекетова, 2017. 93 с
20. Катерняк І. Посібник з підготовки та організації електронного навчання. К.: ТОВ «Фарбований лист», 2016. 48 с.
21. Комп'ютери та комп'ютерні технології: навч. посіб. / Ю. Б. Бродський, К. В.

Молодецька, О. Б. Борисюк, І. Ю. Гринчук. Житомир : Вид-во «Житомирський національний агроекологічний університет», 2016. 186 с.

22. Нужній Є.М., Клименко І.В., Акімов О.О. Інструментальні засоби електронного офісу: навч. посібник. К: Центр навчальної літератури. 2017. 296 с.

23. Основи інформаційних технологій і систем: підручник / В.А. Павлишин, Л.К. Гліненко, Н.Б. Шаховська. Львів: Видавництво Львівської політехніки, 2018. 620 с.

24. Прокопенко А. І., Підчасов Є. В., Москаленко В. В., Доценко С. О., Лебедева В.В. Технології дистанційного навчання: методологія створення та супроводу навчальних курсів. Навч. посібн. Х.: ХНПУ імені Г.С. Сковороди; «Мітра», 2019. 81 с.

25. Спірін О. М. Критерії і показники якості інформаційно-комунікаційних технологій навчання. Інформаційні технології і засоби навчання: електр. наук.-пед. журнал / ред. В. Ю. Биков. 2013. № 1 (33). URL : <http://journal.iitta.gov.ua>

26. Сучасні інформаційно-комунікаційні технології: Навчальний посібник. / Г.Г.Швачич, В.В.Толстой, Л.М.Петречук, Ю.С.Івашенко, О.А.Гуляєва, О.В. Соболенко О.В. Дніпро: НМетАУ, 2017. 230 с. URL\$ [https://nmetau.edu.ua/file/ikt\\_tutor.pdf](https://nmetau.edu.ua/file/ikt_tutor.pdf)

27. Сучасні інформаційно-комунікаційні технології: навчальний посібник. Г. Г. Швачич, В. В. Толстой, Л. М. Петречук, Ю. С. Івашенко, О. А. Гуляєва, О. В. Соболенко. Дніпро: НМетАУ, 2017. 230 с.

28. Твердохліб І.А. Особливості програмно-технічного забезпечення дистанційного навчання в умовах воєнного стану: методичні рекомендації. [Електронне видання]. Київ: Педагогічна думка, 2023. 44 с.

29. Технології дистанційного професійного навчання: метод. посіб. / О. В. Базелюк, О. М. Спірін, Л. М. Петренко, А. А. Каленський та ін. Житомир: «Полісся», 2018. 160 с. URL: <https://cutt.ly/GfBwWm1>.

30. Уманець, В. Цифровізація освітніх та управлінських процесів у закладах професійної (професійно-технічної) освіти. Наука і техніка сьогодні. 2023. № 2 (16). URL: <http://perspectives.pp.ua/index.php/nts/article/view/3822/3843>

31. Шишкіна М.П. Класифікація програмних засобів навчального призначення// URL: <http://lib.iitta.gov.ua/352/1/Classif-Kir.pdf>.

#### **Інтернет ресурси:**

1. About – USDLA. USDLA – United States Distance Learning Association. URL: <https://usdla.org/about/>.

2. Book Creator – створення цифрових інтерактивних книг. <https://bookcreator.com>

3. Canva for Education. URL: <https://www.canva.com/education/>

4. Edmodo – освітня мережа для управління класом, спілкування та оцінювання. <https://new.edmodo.com>

5. Genially. URL: <https://genial.ly/>

6. Google Classroom – безкоштовна платформа для організації навчання. <https://classroom.google.com>

7. Google for Education Teacher Center – інтерактивні модулі для підготовки вчителів до сертифікації Google. <https://teachercenter.withgoogle.com>

8. Google Forms – просте створення тестів, форм опитування з автоматичним підрахунком. <https://forms.google.com>

9. H5P – інструмент для створення інтерактивного контенту, тестів, презентацій. <https://h5p.org>

10. ISTE – International Society for Technology in Education – світові стандарти цифрової компетентності викладачів. URL: <https://www.iste.org/standards>

11. Kahoot! – платформа для вікторин, ігор, опитувань у реальному часі. <https://kahoot.com>

12. LearningApps. URL: <https://learningapps.org/>

13. Lucidpress – створення макетів для публікацій, брошур, бюлетенів. <https://www.marq.com> (раніше Lucidpress)

14. Mentimeter – інтерактивні презентації з вбудованими опитуваннями. <https://www.mentimeter.com>

15. Microsoft Learn Educator Center – онлайн-академія цифрових рішень Microsoft в освіті. <https://learn.microsoft.com/en-us/training/educator-center>
16. Mozaik education. URL: <https://www.mozaweb.com/uk/index.php>.
17. Piktochart – інфографіка, графічні звіти, постери. <https://piktochart.com>
18. Prezi – анімовані, нелінійні презентації. <https://prezi.com>
19. Quizizz – створення інтерактивних тестів із гейміфікацією. <https://quizizz.com>
20. Slido – створення запитань, опитувань, голосувань у режимі презентацій. <https://www.slido.com>
21. Venngage – генератор інфографіки з шаблонами. <https://venngage.com>
22. Virtual Classroom Software. BigBlueButton – Open Source Web Conferencing. URL: <https://bigbluebutton.org/>.
23. Wordwall – генератор дидактичних ігор і вправ (вікторини, лабіринти, відповідність). <https://wordwall.net>
24. Інститут інформаційних технологій і засобів навчання НАПН України. URL: <https://iitlt.gov.ua/>
25. Про авторське право і суміжні права. Офіційний вебпортал парламенту України. URL: <https://zakon.rada.gov.ua/laws/show/3792-12>.

**ДОДАТОК А**

Міністерство освіти і науки України  
Житомирський державний університет імені Івана Франка

Фізико-математичний факультет  
Кафедра комп'ютерних наук та інформаційних технологій

**ЗВІТ**  
**з навчальної практики з розробки цифрових**  
**освітніх ресурсів**

УЗГОДЖЕНО

керівник практики від кафедри  
комп'ютерних наук та інформаційних  
технологій

посада,

Прізвище, ім'я

ВИКОНАВЕЦЬ

студент(ка) 2-го (3-го) курсу

\_\_\_\_\_ групи,

Спеціальність *A5 Професійна освіта*

Спеціалізація *A5.39 Цифрові технології*

Прізвище, ім'я, по-батькові

м. Житомир, 2026 р.

## Зміст

1. ЗАВДАННЯ НАВЧАЛЬНОЇ ПРАКТИКИ З ВИГОТОВЛЕННЯ МУЛЬТИМЕДІЙНИХ ПРОГРАМНИХ ЗАСОБІВ, ЯК ЕЛЕМЕНТІВ МЕТОДИЧНОГО ЗАБЕЗПЕЧЕННЯ	56
2. СТАДІЇ ТА ЕТАПИ РОЗРОБКИ	57
3. ВИКОНАННЯ ІНДИВІДУАЛЬНОГО ЗАВДАННЯ	58
4. ВИСНОВКИ	59

## **1. ЗАВДАННЯ НАВЧАЛЬНОЇ ПРАКТИКИ З РОЗРОБКИ ЦИФРОВИХ ОСВІТНІХ РЕСУРСІВ**

Практичне завдання №1. ...

Практичне завдання №2. ..

Практичне завдання №3. ..

Індивідуальне завдання на практику

...



### **3. ВИКОНАННЯ ЗАВДАННЯ НА ПРАКТИКУ**

Опис виконання практичних та індивідуальних завдань.

## **4. ВИСНОВКИ**