

Розробка звукової доріжки	Звукова доріжка включає в себе музику, звуки оточення, діалоги персонажів та інші аудіо елементи, які додають атмосферу і глибину грі	<ul style="list-style-type: none"> <li>• <i>Композиція музики.</i> Створення музичних композицій, які підкреслюють настрій атмосфери гри.</li> <li>• <i>Звукові ефекти.</i> Створення музичних композицій, які підкреслюють настрій атмосфери гри</li> <li>• <i>Діалоги.</i> Запис та редагування голосових ліній для персонажів та ігрових ситуацій</li> </ul>
Тестування та відлагодження	На цьому етапі гра піддається ретельному тестуванню для виявлення помилок, багів та недоліків. Після виявлення проблем проводяться відповідні корекції	<ul style="list-style-type: none"> <li>• <i>Функціональне тестування.</i> Перевірка функціональності гри, виявлення та виправлення помилок</li> <li>• <i>Бета-тестування.</i> Проведення тестування залученням зовнішніх тестерів або гравців для отримання зворотного зв'язку</li> </ul>
Вип. гри	Коли всі попередні етапи завершені і гра пройшла успішне тестування, вона готова до випуску. Гра розповсюджується серед гравців через різноманітні канали, такі як цифрові магазини, диски, платформи для завантаження тощо	<ul style="list-style-type: none"> <li>• <i>Маркетинг та просування</i> Підготовка матеріалів для просування гри рекламні кампанії та комунікація з громадськістю</li> <li>• <i>Легалізація.</i> Забезпечення всіх необхідних ліцензій та правових документів для випуску гри</li> <li>• <i>Реліз.</i> Публікація гри на різних платформах або в магазинах, підтримка після релізу та взаємодія з гравцями</li> </ul>

Запропонований авторами алгоритм може стати корисним для розробки власної гри.

### Список використаних джерел

1. Розробка відеоігор. URL: [https://en.wikipedia.org/wiki/Video\\_game\\_development](https://en.wikipedia.org/wiki/Video_game_development).
2. Туранський П. В., Лешук С. О. Навчально-ігрові середовища // Сучасні інформаційні технології та інноваційні методики навчання: досвід, тенденції, перспективи: матеріали IV Міжнародної науково-практичної інтернет-конференції, м. Тернопіль, 7–8 листопада, 2019 р. Тернопіль : ТНПУ ім. В. Гнатюка, 2019. С. 18–20.
3. Skaskiv A. Game study technology // Perspectives of world science and education: the 4th International scientific and practical conference (December 25–27, 2019). Osaka. Japan : CPN Publishing Group, 2019. P. 193–196.

## GODOT ENGINE – ІНСТРУМЕНТ ДЛЯ ПІДГОТОВКИ ФАХІВЦІВ У СФЕРІ ІНЖЕНЕРІЇ ІГРОВИХ ПРОЄКТІВ

**Джуга Денис Євгенійович**

студент спеціальності 122 Комп'ютерні науки (Інженерія ігрових проєктів),  
Тернопільський національний педагогічний університет імені Володимира Гнатюка,  
zhuga\_dy@fizmat.tnpu.edu.ua

**Мартинюк Сергій Володимирович**

кандидат фізико-математичних наук, доцент кафедри інформатики та методики її навчання,  
Тернопільський національний педагогічний університет імені Володимира Гнатюка,  
sergmart65@tnpu.edu.ua

Ігрова індустрія сьогодення стрімко розвивається, що створює постійну потребу задіяння нових професійних спеціалістів. В ігровій індустрії задіяні фахівці багатьох спеціальностей, починаючи від програмістів, які прописують механіку гри, та закінчуючи композиторами, які створюють звуковий супровід.

Майже кожен, хто причетний до розробки ігрових проєктів, повинен мати базові знання створення ігор для того, щоб розуміти всю специфіку роботи, яку має виконувати. Для цього майбутнім спеціалістам потрібно ознайомитися та попрактикуватися з інструментами створення ігор. Одним з таких програмних продуктів є Godot Engine.

Godot Engine – сучасний багатоплатформний ігровий рушій з відкритим кодом, який має інструменти для роботи з 2D та 3D графікою [1]. Програмування відбувається з використання мов програмування C++, C#, GDScript (високорівнева динамічно типізована мова програмування, створена спеціально для Godot Engine). Працює під керівництвом операційних системах Windows, Linux, macOS, BSD та Haiku. Програма дозволяє експортувати проєкти для використання на персональні комп'ютери, мобільні та вебплатформи [2].

Структура проєктів Godot Engine містить такі основні складові:

- **Сцена** – це один або певна кількість (структура) вузлів, які утворюють дерево і зберігаються окремим файлом;
- **Скрипт** – код, написаний на доступних для використання мовах рушія, під'єднаний до вузла сцени, інтегрований у файл сцени або збережений окремо;
- **Вузол** – це об'єкт, наділений власними властивостями й унікальним функціоналом. Їх можна поділити на такі класи:
  - «Node2D» – вузли для створення 2D ігрових сцен;
  - «Node3D» – вузли для створення 3D ігрових сцен;
  - «Control» – вузли для створення користувацького інтерфейсу;
  - інші вузли загального використання (використовують з будь-яким типом, наведеним вище).

Хоча Godot Engine є доволі простим в опануванні, проте для ефективнішого вивчення потрібно мати структурований план послідовного оволодіння.

1. Ознайомлення з інтерфейсом програмного забезпечення. Це потрібно зробити в першу чергу, щоб надалі легко орієнтуватися в програмі й не витратити зайвого часу на пошуки потрібних панелей, вікон, закладок тощо.

2. Вивчення основ і синтаксису мови програмування GDScript:

- змінні й операції над ними;
- розгалуження;
- цикли;
- функції;
- класи.

3. Основи роботи з 2D вузлами. Для кращого вивчення доцільно спробувати створити прості ігрові проєкти такі як рух персонажу, підбирання предметів, постріл зі зброї тощо.

4. Вивчення та практика вузлів для створення користувацького інтерфейсу. Приклад елементів для здобуття практичних навичок є розробка лічильника зібраних монет, відображення кількості життів персонажа тощо.

5. Вивчення та використання на практиці вузли, які не підлягають під жодну категорію (додання звуків, створення анімації, таймер та інші). Застосування цих вузлів допоможе у створенні складніших елементів, покращить ігровий процес та атмосферу ігрового світу [3].

Зразок плану опанування Godot, поданий вище, є тільки невеликою частиною того, чого можна навчитися та використовувати при розробці ігор. Проте варто зауважити, що цих знань вистачить для створення простих ігор. Наприклад:

- **Платформер.** Жанр ігор, у якому ігровий світ складається з платформ. Однією з найвідоміших представників цього жанру є серія ігор від японської компанії Nintendo «Super Mario»;
- **«Top-down».** Жанр ігор, у якому перегляд робочого поля здійснюється з камери, встановленої зверху (рис. 1).



Рис. 1. Гра Stardew Valley

Опановуючи Godot Engine, студенти набудуть необхідних знань і розуміння процесу створення ігрових проєктів. Рушій Godot Engine є доволі перспективним, щоб стати основним для деяких ігрових компаній і студій у майбутньому, оскільки вже зараз він є одним з найуживаніших рушіїв серед інді-розробників та інді-студій. Крім цього, майбутнім розробникам значно простіше буде освоїти інші рушії та розробляти більш якісні й оптимізовані складові частини ігор.

### Список використаних джерел

1. Wikipedia: Godot. URL: <https://uk.wikipedia.org/wiki/Godot> (дата звернення: 26.03.2024).
2. Perforce: What Are the Best Game Engines? URL: <https://www.perforce.com/blog/vcs/most-popular-game-engines#five-01> (дата звернення: 27.03.2024).
3. DOU: Сучасний ІТ-ринок: тенденції, рівень зарплат, поради новачкам і досвідченим розробникам. URL: <https://dou.ua/forums/topic/46109/> (дата звернення: 31.03.2024).

## ОСОБЛИВОСТІ ФОРМУВАННЯ НАВИЧОК ТА ПРЕДМЕТНОЇ КОМПЕТЕНЦІЇ З ПРОЕКТУВАННЯ МІКРОКОНТРОЛЕРІВ У СЕРЕДОВИЩІ PROTEUS ДЛЯ МАЙБУТНІХ ІНФОРМАЦІЙНО-ТЕХНОЛОГІЧНИХ СПЕЦІАЛІСТІВ

### Заєць Олександр Юрійович

здобувач третього (освітньо-наукового) рівня вищої освіти спеціальності  
011 Освітні, педагогічні науки (інформаційно-комунікаційні технології в освіті),  
Український державний університет імені Михайла Драгоманова,  
[o.yu.zaiets@npu.edu.ua](mailto:o.yu.zaiets@npu.edu.ua)

Формування навичок та предметної компетенції з проєктування мікроконтролерів у середовищі Proteus є невід'ємною складовою підготовки майбутніх інформаційно-технологічних спеціалістів. Дані навички дозволяють студентам здобути практичний досвід у створенні електронних пристроїв та використанні мікроконтролерів для управління різноманітними системами. Під час навчання у середовищі Proteus студенти отримують можливість експериментувати з різними аспектами проєктування, від алгоритмів