

контроль доступу до ресурсів. Також враховано рекомендації OWASP щодо захисту вебзастосунків від найбільш поширених загроз [5]. Як зазначають вітчизняні дослідники, зокрема наукові представники ТНПУ, забезпечення безпеки інформаційних систем є важливою складовою сучасних програмних рішень.

Інтерфейс користувача розроблено з урахуванням сучасних принципів UX/UI, що забезпечує інтуїтивність та зручність використання системи. Користувач має можливість керувати ботами, переглядати статистику та аналізувати результати торгівлі.

Проведене тестування застосунку підтвердило його стабільність, коректність роботи та відповідність поставленим вимогам. Було перевірено функціональність основних модулів, продуктивність системи та її стійкість до навантажень.

Наукова новизна роботи полягає у поєднанні сучасних вебтехнологій із методами автоматизованого аналізу ринку криптовалют, що дозволяє створити ефективний інструмент для автоматизованої торгівлі.

Практичне значення роботи полягає у створенні прототипу вебзастосунку, який може бути використаний для подальшого розвитку та впровадження у реальних умовах.

У результаті виконання роботи розроблено вебзастосунок для автоматизованої торгівлі криптовалютами, що включає реалізацію алгоритмів роботи торгових ботів, зручного інтерфейсу користувача та серверної логіки. Система забезпечує виконання торгових операцій відповідно до заданих параметрів, підтримує аналіз результатів та керування процесами торгівлі. Проведене тестування підтвердило стабільність і коректність роботи застосунку. Практичне значення роботи полягає у створенні прототипу, який може бути використаний для подальшого розвитку та впровадження у реальних умовах.

#### Список використаних джерел

1. Angular Documentation. URL: <https://angular.io> (дата звернення: 27.03.2026).
2. Freeman A. Pro ASP.NET Core MVC. New York: Apress, 2017. 1017 p.
3. Nakamoto S. Bitcoin: A Peer-to-Peer Electronic Cash System. URL: <https://bitcoin.org/bitcoin.pdf> (дата звернення: 26.03.2026).
4. Sommerville I. Software Engineering. 10th ed. Boston: Pearson, 2016. 816 p.
5. OWASP Web Security Testing Guide. URL: <https://owasp.org/www-project-web-security-testing-guide/> (дата звернення: 27.03.2026).

## РОЗРОБКА МІЖПЛАТФОРМНОЇ КОМП'ЮТЕРНОЇ ГРИ ЖАНРУ IDLE CLICKER «CANDY CLICKER» НА БАЗІ РУШІЯ GODOT ENGINE

### Гаврилюк Андрій Васильович

здобувач першого рівня вищої освіти спеціальності Комп'ютерні науки  
Тернопільський національний педагогічний університет імені Володимира Гнатюка  
[gavrylyuk\\_av@fizmat.tnpu.edu.ua](mailto:gavrylyuk_av@fizmat.tnpu.edu.ua)

### Габрусєв Валерій Юрійович

кандидат педагогічних наук, доцент кафедри інформатики та методики її навчання  
Тернопільський національний педагогічний університет імені Володимира Гнатюка  
[gabrusev@fizmat.tnpu.edu.ua](mailto:gabrusev@fizmat.tnpu.edu.ua)

Сучасна ігрова індустрія є одним із найдинамічніших сегментів глобального ринку цифрових технологій, що за обсягом доходів перевершує кіноіндустрію та музичну галузь разом узяті. Кількість активних гравців у світі перевищує три мільярди осіб, а серед різноманіття ігрових жанрів особливе місце посідають інкрементальні ігри (Idle/Clicker). За останнє десятиліття цей жанр перетворився з

нішевого експериментального формату на комерційно успішний сегмент із глобальним ринком обсягом 2,5 мільярда доларів США у 2024 році та прогнозованим зростанням до 6,1 мільярда доларів до 2033 року [3]. Низький поріг входу для розробників, мінімальні вимоги до апаратного забезпечення та ефективні механіки залучення гравців роблять цей жанр привабливою платформою для дослідження принципів ігрового проектування та програмної інженерії.

Стрімкий розвиток вільних ігрових рушіїв, зокрема Godot Engine із відкритим вихідним кодом та ліцензією MIT, створює нові можливості для розробки повнофункціональних ігрових застосунків без фінансових зобов'язань. Актуальність даної роботи обумовлена необхідністю систематизації досвіду застосування рушія Godot Engine для розробки ігор жанру Idle/Clicker, дослідження архітектурних патернів, специфічних для інкрементальних ігор, та практичної реалізації збалансованої ігрової системи з урахуванням психологічних механізмів залучення гравців і сучасних вимог до кросплатформних застосунків [1].

Метою дослідження є розробка повнофункціонального ігрового застосунку «Candy Clicker» з використанням рушія Godot Engine версії 4.x та мови програмування GDScript, що реалізує класичні механіки інкрементального жанру в оригінальному тематичному оформленні кондитерської тематики та забезпечує збалансований ігровий досвід із можливістю збереження прогресу між сесіями [5].

Аналіз жанру Idle/Clicker показав, що він ґрунтується на п'яти ключових механіках: активний клік (безпосередня взаємодія гравця з ігровим елементом), система генераторів пасивного доходу, ієрархічна система покращень, система престижу та офлайн-прогрес. Феноменальна популярність жанру пояснюється здатністю ефективно активувати системи винагороди мозку через «дофаміновий цикл» та задовольняти фундаментальну людську потребу у відчутті прогресу. Порівняльний аналіз сучасних ігрових рушіїв для двовимірної розробки — Unity, Unreal Engine, GameMaker, Defold та Construct — дозволив обґрунтувати вибір Godot Engine як оптимального інструменту завдяки відкритому вихідному коду, ліцензії MIT, відсутності роялті, вбудованій мові GDScript з Python-подібним синтаксисом, а також нативній підтримці експорту на платформи Windows, Linux, Android та HTML5 [2].

На етапі проектування розроблено концепцію гри в кондитерській тематиці з 11 типами генераторів пасивного доходу — Желейка, Мармеладка, Кексик, Ледяник, Пончик, Ріжок та ін. — кожен з яких має унікальну базову вартість і продуктивність. Архітектура програмного забезпечення побудована на компонентному підході із централізованим головним скриптом `main.gd` та системою сигналів Godot для слабкозв'язаної комунікації між модулями. Формалізовано математичну модель ігрової економіки з експоненційною функцією вартості генераторів  $C(n) = C_0 \times r^n$  із коефіцієнтом зростання  $r = 1.12$  та лінійною продуктивністю  $P(n) = P_0 \times n$ , що забезпечує збалансований темп прогресії з поступовим уповільненням на пізніх етапах гри [3]. Спроектовано систему персистентного збереження на основі JSON-серіалізації з записом у директорію `user://`, механізмом валідації даних за принципом *graceful degradation* та стратегією автозбереження, що поєднує періодичний (кожні 30 секунд) та подієвий підходи [4].

У процесі реалізації розроблено алгоритм офлайн-прогресу з обмеженням накопичення, модуль форматування великих чисел та основні ігрові механіки засобами GDScript. Інтерфейс створено на базі системи вузлів Godot із забезпеченням адаптивності та зручності використання. Проведене тестування підтвердило стабільність роботи застосунку на різних платформах та його відповідність заданим вимогам. Виконано оптимізацію продуктивності, що забезпечило ефективну роботу та низьке енергоспоживання. Наукова новизна

полягає у поєднанні архітектурних підходів Godot із механіками інкрементальних ігор, а практичне значення — у створенні готового ігрового продукту, придатного до використання та подальшого розвитку.

### Список використаних джерел

1. Мельник П. П., Василенко Я. П. Особливості використання рушія Godot та C# для розробки ігрових застосунків. Сучасні цифрові технології та інноваційні методики навчання: досвід, тенденції, перспективи: матеріали XV Міжнародної науково-практичної інтернет-конференції, м. Тернопіль, 10 квітня, 2025 р. Тернопіль: ТНПУ ім. Володимира Гнатюка, 2025. С. 156–159.
2. Alybaev A. Comparative Analysis of Unity and Godot for 2D Game Development. Preprints. 2025. URL: <https://www.preprints.org/manuscript/202511.1981> (дата звернення: 24.03.2026).
3. Buergi J. Idle Yet Engaged: How Idle Games Satisfy Our Needs for Competence and Autonomy: diss. Rensselaer Polytechnic Institute, 2025. 120 p. URL: <https://www.proquest.com/openview/daa4dd324eab2af807e492f8c945122a/1> (дата звернення: 26.03.2026).
4. Heidari A. Gaming with Emojis: A Look at Different Strategies of Emoji Inclusion in the Design of Digital Games. Acta Ludologica. 2024. Vol. 7, No. 2. P. 38–51.
5. Vanhove S. Learning GDScript by Developing a Game with Godot 4: A Fun Introduction to Programming in GDScript 2.0 and Game Development Using the Godot Engine. Packt Publishing, 2024. 350 p.

## РОЗГЛЯД ОПТИМІЗАЦІЙНИХ ЗАДАЧ В АСПЕКТІ МІЖПРЕДМЕТНИХ ЗАНЯТЬ

### Грод Інна Миколаївна

кандидат фізико-математичних наук, доцент кафедри інформатики та методики її навчання  
Тернопільський національний педагогічний університет імені Володимира Гнатюка  
[grodin@tnpu.edu.ua](mailto:grodin@tnpu.edu.ua)

### Грод Іван Миколайович

доктор фізико-математичних наук, професор кафедри математики та методики її навчання  
Тернопільський національний педагогічний університет імені Володимира Гнатюка  
[grodiv@tnpu.edu.ua](mailto:grodiv@tnpu.edu.ua)

Актуальним є питання формування у студентів різних напрямів загальних, математичних та професійних компетентностей засобами математики, що, у свою чергу, демонструє їх практичну значущість. Інтеграція сучасних технологій у навчальний процес є необхідною умовою формування професійних компетентностей студентів [2].

У сучасній освіті, зокрема в математичній галузі, існують різноманітні практичні завдання: рівняння, текстові задачі, нерівності, системи рівнянь і нерівностей, функції різних типів тощо. Завдання оптимізації — це окремий тип завдань. Вони приваблюють викладачів і студентів тим, що розв'язування таких задач може бути розглянуте в аспекті міжпредметних занять. Наприклад, математика і фізика, математика та економіка тощо.

Історично людство стикалося з проблемами оптимізації ще в давнину. У давнину розв'язувалися геометричні задачі, пов'язані з властивостями різних геометричних фігур. З появою диференціального та інтегрального числення стало можливим вивчати складніші задачі. У середині минулого століття народився новий етап розвитку методів оптимізації. Початком цього етапу були завдання, орієнтовані на практику. З'явилися лінійне та динамічне програмування, теорія ігор тощо.