

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ  
Харківський національний університет радіоелектроніки

Центр післядипломної освіти  
Кафедра Програмної інженерії

**КВАЛІФІКАЦІЙНА РОБОТА**

**Пояснювальна записка**

другий (магістерський)

(рівень вищої освіти)

Дослідження методів гейміфікації розробки систем навчання

Виконала:

студентка 2 курсу групи ІІЗЗдм-21-2

Суровікіна К.С.

(прізвище, ініціали)

Спеціальність 121 – Інженерія програмного  
забезпечення

Тип програми Освітньо-наукова

Керівник доц. Каук В.І.

(посада, прізвище, ініціали)

Допускається до захисту

Зав. Кафедри \_\_\_\_\_ З.В. Дудар

2023

Харківський національний університет радіоелектроніки  
Центр післядипломної освіти  
Кафедра Програмної інженерії  
Рівень вищої освіти другий (магістерський)  
Спеціальність 121– Інженерія програмного забезпечення  
(код і повна назва)  
Тип програми Освітньо-наукова програма  
Освітня програма Інженерія програмного забезпечення

ЗАТВЕРДЖУЮ:

Зав. кафедри \_\_\_\_\_

(підпис)

«03» квітня 2023 р.

## ЗАВДАННЯ

### НА КВАЛІФІКАЦІЙНУ РОБОТУ

студентки Суровікіної Катерині Сергіївні  
(прізвище, ім'я, по батькові)

1. Тема роботи «Дослідження методів гейміфікації розробки систем навчання» затверджена наказом університету від «03» квітень 2023 р. № 83Стз
2. Термін подання студентом роботи до екзаменаційної комісії «23» травня 2023 р.
3. Вихідні дані до роботи моделі та методи гейміфікації, розробка методів та системи навчання, мова програмування JavaScript, бібліотека React, середовище розробки Visual Studio Code.
4. Перелік питань, що потрібно опрацювати в роботі мета роботи, аналіз предметної галузі і постановка задачі, дослідження предметної галузі, огляд існуючих методів гейміфікації, розробка гейміфікованої систем навчання програмування, вибір відповідних інструментів для розробки рішення.

## КАЛЕНДАРНИЙ ПЛАН

№	Назва етапів роботи	Терміни виконання етапів роботи	Примітка
1.	Аналіз предметної галузі	03.04.2023	виконано
2.	Аналіз основних моделей та методів гейміфікації, їх впливу на мотивацію студентів у навчанні	06.04.2023	виконано
3.	Проектування системи	19.04.2023	виконано
4.	Вибір відповідних інструментів для розробки рішення	22.04.2023	виконано
5.	Розробка додатка для навчання та підвищення рівня вмінь програмістів	02.05.2023	виконано
6.	Підготовка пояснювальної записки, презентації та доповіді	13.05.2023	виконано
7.	Нормоконтроль	17.05.2023	виконано
8.	Рецензування	18.05.2023	виконано
9.	Занесення диплома в електронний архів	20.05.2023	виконано
10.	Попередній захист	21.05.2023	виконано
11.	Допуск до захисту	22.05.2023	виконано

Дата видачі завдання «03» квітня 2023р.

Студент

\_\_\_\_\_

(підпис)

Керівник роботи

\_\_\_\_\_

(підпис) (посада, прізвище, ініціали)

## РЕФЕРАТ / ABSTRACT

Кваліфікаційна робота магістра містить: 71с., 5 рис., 51 джер.

АРХІТЕКТУРА, ГЕЙМІФІКАЦІЯ, МОТИВАЦІЯ, СИСТЕМИ НАВЧАННЯ,  
ФРОНТ-ЕНД ТЕХНОЛОГІЇ, JAVASCRIPT, REACT.

Об'єктом дослідження є методи гейміфікації в розробці систем навчання.

Метою роботи є вивчення ефективності та впливу гейміфікаційних методів на процес навчання та підвищення якості роботи програмістів у технічних компаніях.

Методи розробки базуються на таких технологіях, як JavaScript, React, NodeJS, Visual Studio, Git.

В результаті роботи було досліджено, що використання методів гейміфікації в розробці систем навчання має значний потенціал для поліпшення процесу навчання та досягнення кращих результатів. Виявлені основні переваги і можливості. Було реалізовано систему, яка дозволяє інтегрувати гейміфікаційні елементи в процес навчання програмування.

ARCHITECTURE, GAMIFICATION, MOTIVATION, LEARNING  
SYSTEMS, FRONT-END TECHNOLOGIES, JAVASCRIPT, REACT.

The object of research is the methods of gamification in the development of educational systems.

The aim of this study is to investigate the effectiveness and impact of gamification methods on the learning process and to enhance the performance of programmers in technical companies.

The development methods are based on technologies such as JavaScript, React, NodeJS, Visual Studio, and Git.

As a result of the qualification work, it has been found that gamification methods

in educational system development have significant potential for improving the learning process and achieving better results. The main advantages and opportunities have been identified. A system integrating gamification elements into the programming learning process has been implemented.

Я, Суровікіна Катеріна Сергіївна, студентка групи ІПЗздм-21-2, здобувачка вищої освіти на другому (магістерському) рівні кафедри «Програмна інженерія», заявляю: моя кваліфікаційна робота на тему «Дослідження методів гейміфікації розробки систем навчання», що буде представлена до ЕК для публічного захисту, виконана самостійно, в ній не містяться елементи плагіату і вона може бути опублікована в електронному архіві відкритого доступу EIArKhNURE. Всі запозичення з друкованих та електронних джерел мають відповідні посилання.

Я ознайомена з діючим положенням «Про протидію академічному плагіату в ХНУРЕ», згідно з яким виявлення плагіату є підставою для відмови в допуску кваліфікаційної роботи до захисту та застосування дисциплінарних заходів.

## ЗМІСТ

Вступ.....	7
1. Огляд предметної галузі.....	9
1.1. Історія гейміфікації.....	9
1.2. Основні моделі гейміфікації.....	12
1.3. Основні методи гейміфікації.....	14
1.3.1. Метод гейміфікації "Гра".....	15
1.3.2. Метод гейміфікації "Бали та досягнення".....	17
1.3.3. Метод гейміфікації "Ранги та рейтинги".....	19
1.3.4. Метод гейміфікації "Місії та завдання".....	19
1.3.5. Метод гейміфікації "Персонажі та історії".....	21
1.3.6. Метод гейміфікації " Соціальна інтеракція ".....	22
1.4. Проблеми в гейміфікації: аналіз та вирішення.....	24
2. Підхід в розробці гейміфікованої системи навчання.....	28
2.1. Вибір основних підходів та методів гейміфікації.....	28
2.2. Аналіз існуючих додатків для навчання програмування.....	30
2.3. Опис проектування додатка для підвищення якості роботи програмістів.....	35
3. Програмна реалізація системи навчання.....	38
3.1. Опис використовуваних засобів розробки.....	38
3.2. Опис архітектури системи.....	43
3.3. Опис роботи застосунку.....	46
Висновки.....	50
Перелік джерел посилання.....	52
Додаток А Тези доповідей V Міжнародної наукової конференції «Актуальні питання розвитку галузей науки».....	57
Додаток Б Звіт результатів перевірки на унікальність тексту в мережі інтернет та базі ХНУРЕ.....	62
Додаток В Експертний висновок нормоконтроль.....	63

	7
Додаток Г Слайди презентації.....	64

## ВСТУП

Сучасний світ заснований на постійному розвитку та інноваціях, і навчання не є винятком. У наш час компанії та організації намагаються підвищити продуктивність працівників, впроваджуючи нові технології та методи навчання. Застосування інноваційних методів та технологій у системах навчання стає все більш актуальним і перспективним. Одним зі способів, які застосовують для покращення навчального процесу, є гейміфікація.

Гейміфікація – це підхід, який використовує елементи гри та гейміфікаційні механізми в навчальних процесах з метою стимулювання мотивації та залучення учнів до активного навчання. Цей підхід базується на ідеї, що елементи гри, такі як бали, досягнення, рівні, лідерські дошки, можуть зробити процес навчання більш цікавим, захоплюючим та мотивуючим для учнів. Гейміфікація включає в себе застосування ігрових елементів та механік у навчальному процесі з метою збільшення мотивації учнів та покращення їхнього сприйняття матеріалу. Цей підхід є досить новим та не дослідженим достатньо, тому проведення дослідження методів гейміфікації розробки систем навчання є актуальним.

Результати такого дослідження можуть допомогти виявити ефективні методи гейміфікації, які можна застосовувати у навчанні, а також використовувати їх у практиці, оскільки гейміфікація може допомогти підвищити ефективність навчального процесу та покращити результати. Існують питання щодо оптимального вибору гейміфікаційних елементів, їхньої інтеграції в навчальну систему, а також впливу гейміфікації на навчальні результати та мотивацію учнів.

Традиційні методи навчання можуть бути недостатньо ефективними та нудними для більшості. Застосування ігрових елементів та механік гейміфікації може допомогти зробити навчання більш захоплюючим та ефективним. Під час дослідження метою є зосередитися на аналізі різних підходів та технологій, що використовуються у гейміфікаційних системах навчання. Дослідити можливості

використання балів, досягнень, рівнів, лідерських дошок та інших гейміфікаційних елементів для залучення учнів та підвищення їхньої мотивації.

Проте, однією з проблем гейміфікації є те, що не всі ігри є придатними для використання в освітніх цілях, або не всі ігри можуть бути адаптовані для використання в системах навчання. Крім того, існує ризик того, що ігрові елементи можуть бути використані неефективно, або що вони можуть завадити навчанню, якщо їх використовувати неправильно. Важливим є аналіз можливих перешкод, таких як недостатня участь учнів, залежність від зовнішніх мотиваторів, важкість управління гейміфікаційною системою та інші.

Необхідно розширити область застосування гейміфікації, вивчити її вплив на різні типи навчання та цільові аудиторії. Крім того, важливо розглянути можливості інтеграції штучного інтелекту, віртуальної реальності та інших новітніх технологій у практику гейміфікації. Дослідження в області гейміфікації в системах навчання є неперервним процесом, оскільки розвиток технологій та новітні підходи вимагають постійного оновлення та вдосконалення методів. Є великий потенціал гейміфікації як інноваційного підходу до покращення навчальних процесів. Він сприяє залученню, мотивації та підвищенню якості навчання. Однак, для успішної імплементації гейміфікації в системи навчання необхідно враховувати контекст, потреби та особливості цільової аудиторії.

Таким чином, основною метою цієї роботи є дослідження та визначення ефективних методів гейміфікації та розробка системи навчання для програмістів, яка буде відповідати потребам галузі та забезпечувати оптимальний рівень засвоєння знань, використовуючи найбільш оптимальні методи.

## 1. ОГЛЯД ПРЕДМЕТНОЇ ГАЛУЗІ

У наш час компанії та організації намагаються підвищити продуктивність працівників, впроваджуючи нові технології та методи навчання. Одним зі способів, які застосовують для покращення навчального процесу, є гейміфікація.

Гейміфікація включає в себе застосування ігрових елементів та механік у навчальному процесі з метою збільшення мотивації студентів та покращення їхнього сприйняття матеріалу. Цей підхід є досить новим та не дослідженим достатньо, тому проведення дослідження методів гейміфікації розробки систем навчання є актуальним.

Гейміфікація – це використання ігрових елементів та механік у негеймових контекстах, таких як навчання, з метою збільшення мотивації, зацікавленості та ефективності навчання [1].

У сучасному світі програмування є однією з найбільш швидко розвиваючихся галузей, тому важливо розвивати ефективні методи навчання програмування, які відповідали б вимогам сучасного ринку праці. Застосування гейміфікації в розробці систем навчання може забезпечити підвищення зацікавленості та мотивації студентів, зниження втрати уваги під час навчання та підвищення ефективності засвоєння знань [2].

Однак, застосування гейміфікації в освіті має свої обмеження та виклики, які потрібно враховувати при розробці систем навчання. Основними обмеженнями є необхідність розробки спеціалізованих програм та інструментів, що вимагає великих затрат часу та коштів, а також необхідність підбору відповідних гейміфікованих елементів та механік, які будуть ефективними для конкретної групи програмістів.

### 1.1 Історія гейміфікації

Методи гейміфікації використовуються в різних галузях, включаючи бізнес, маркетинг, освіту та здоров'я. Історія методів гейміфікації починається з 1950-х

років, коли інженерія та системне мислення були введені в освіту як спосіб вирішення проблем.

У 1970-х роках інтерактивні комп'ютерні ігри стали популярними серед широкої аудиторії. Ігри стали не тільки розважальними, але й популярними інструментами навчання та тренування [3].

Один із найперших прикладів гейміфікації можна віднести до 1890-х років, коли психолог на ім'я Едвард Торндайк провів дослідження поведінки тварин за допомогою коробки-головоломки. Він виявив, що тварини навчилися тікати з коробки швидше та ефективніше, коли отримували за це винагороду, наприклад їжу. Це дослідження заклало основу для концепції оперантного кондиціонування, яка є ключовим елементом гейміфікації.

Ще однією важливою віхою в історії гейміфікації стала розробка першої програми лояльності в 1896 році компанією Sperry & Hutchinson, яка випускала «Зелені марки» як винагороду за покупку певних продуктів. Успіх цієї програми надихнув багато інших компаній на впровадження подібних програм на основі винагород, таких як милі для часто літаючих пасажирів і бали за кредитними картками [4].

На початку 2000-х років, внаслідок зростання інтернет-технологій та зростання популярності соціальних мереж, гейміфікація стала все більш поширеною як засіб залучення і утримання уваги користувачів. Гейміфікація почала привертати більше уваги, оскільки дослідники та професіонали галузі почали досліджувати її потенціал для підвищення залученості користувачів, мотивації та навчання в різних контекстах.

Термін «геймофікація» був введений у 2002 році Ніком Пеллінгом, британським програмістом і винахідником, який використовував його для опису процесу застосування ігрової механіки до неігрового контексту [5]. З тих пір гейміфікація була застосована в широкому діапазоні сфер, включаючи освіту, охорону здоров'я, маркетинг і навчання співробітників.

У 2010-х роках гейміфікація стала популярною в бізнесі, де використовується для підвищення мотивації працівників та клієнтів, збільшення продуктивності та покращення взаємодії з брендом. Також вона знаходить застосування в здоров'ї та фітнесі, де використовується для підвищення мотивації до здорового способу життя та досягнення фітнес-цілей.

У 21 столітті цифрові технології дозволили гейміфікації стати більш складною та доступною, ніж будь-коли раніше. Сьогодні такі ігрові елементи, як бали, значки, таблиці лідерів і квести, зазвичай використовуються в мобільних додатках, онлайн-платформах і освітньому програмному забезпеченні, щоб мотивувати користувачів і покращити їхній досвід навчання. Зі швидким розвитком технологій віртуальної та доповненої реальності очікується, що в найближчі роки гейміфікація стане ще більш захоплюючою та ефективною.

Сьогодні гейміфікація застосовується в освіті як спосіб залучення та мотивації учнів, підвищення їхньої активності та зацікавленості у навчанні. Вона використовується в проектах розробки програмного забезпечення для забезпечення кращої взаємодії між користувачем та програмним забезпеченням.

З поширенням комп'ютерів і розвитком галузі інформаційних технологій з'явилася можливість реалізувати гейміфікацію в електронних форматах. Перші ігри, які були гейміфіковані для навчання, були в основному спрямовані на молодших учнів, вони були веселі, легкі і мали графічний дизайн, який дуже подобався молодшим школярам [6].

Однак з часом гейміфікація стала використовуватися для навчання в більш старших вікових групах. Для цього з'явилися нові інструменти, які дозволяють розробляти більш складні інтерактивні системи навчання з використанням ігрових елементів. Вона є ефективним інструментом для підвищення мотивації, залучення уваги і збільшення зацікавленості користувачів у різних діях.

У наш час механіки геймінгу можна знайти майже всюди. Гейміфікацію застосовують практично в усіх сферах життя, але трендсеттерами як зазвичай є великі айті-компанії.

Популярний динозавр в Google Chrome, який з'являється щоразу, коли браузер не може отримати доступ до інтернету.

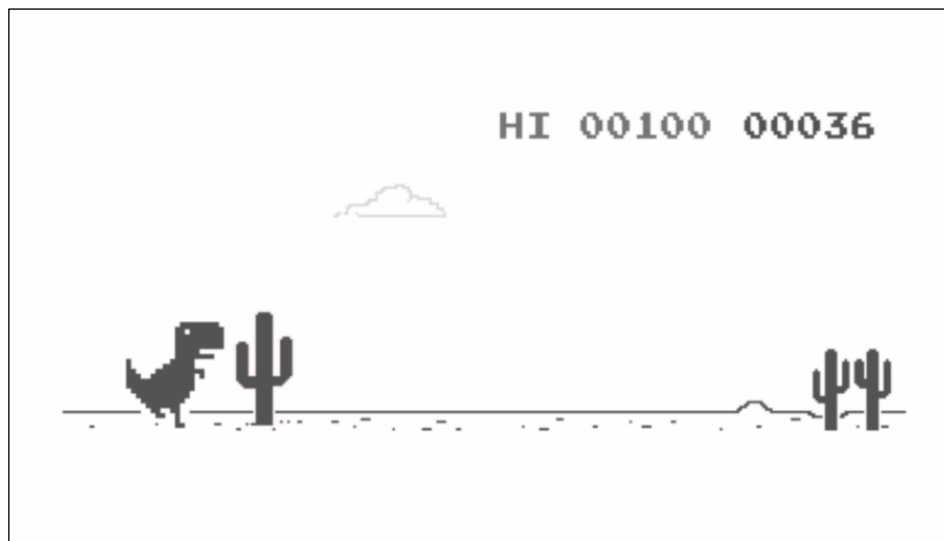


Рисунок 1.1 – Приклад гейміфікація [7].

## 1.2 Основні моделі гейміфікації

Існують різні моделі гейміфікації, які використовуються для проектування та реалізації гейміфікованих досвідів. Деякі з найбільш відомих моделей включають:

- бейджі – це нагороди, які надаються користувачеві за виконання певних завдань або досягнення певних результатів. Ці нагороди можуть відображатися на профілі користувача та давати йому статус або престиж в середині спільноти;
- рейтинги – це системи, які оцінюють успішність користувачів в різних аспектах, таких як результати рішення завдань або рівень активності в середині спільноти. Рейтинги можуть надихати користувачів до змагання та поліпшення своїх результатів;
- лідерборди – це списки користувачів, які досягли найкращих результатів в певних завданнях або метриках. Лідерборди можуть надихати користувачів до змагання за першість та допомагати створювати здорову конкуренцію в середині спільноти;

- виклики – це завдання або ігрові елементи, які вимагають від користувача певних зусиль або рішення певних проблем. Виклики можуть надихати користувачів до активної участі в діяльності та розвивати їх навички та знання;
- прогрес-бари – це інтерактивні елементи, які відображають прогрес користувача виконання завдань або досягнення метрик. Прогрес-бари можуть мотивувати користувачів до досягнення цілей та створювати відчуття задоволення від успіху;
- сюжет – це ігровий елемент, який надає користувачеві контексту та історії навколо завдання або діяльності. Сюжет може надихати користувачів до більш глибокої уваги до діяльності та досягнення цілей;
- соціальна гейміфікація – це використання ігрових елементів та механізмів в соціальних мережах, щоб надихати користувачів до взаємодії та співпраці. Соціальна гейміфікація може надати користувачам відчуття спільноти та допомогти створити позитивне середовище для взаємодії.

Моделі гейміфікації визначають основні принципи та стратегії, які застосовуються для досягнення певної мети, такої як наприклад "навчання" (education), "здоров'я" (health) або "економічних стимулів" (economic incentives).

Використання моделей гейміфікації може покращити мотивацію користувачів та збільшити їхню активність в середині додатку. При розробці додатку для навчання програмістів, моделі гейміфікації можуть допомогти залучити користувачів до активної участі в навчанні, надати їм контекст та структуру вивчення, а також надихнути до пошуку нових знань та вдосконалення своїх навичок [8].

### 1.3 Основні методи гейміфікації

Методи гейміфікації прагнуть залучити природні людські інстинкти: конкуренція, досягнення, статус, самовираження, альтруїзм, розв'язання задач.

В основі стратегії гейміфікації лежить винагородження за виконані завдання.

Методи гейміфікації відображають шляхи застосування ігрових механік та елементів гри для покращення певного процесу, продукту або послуги. Це можуть бути, наприклад, бейджі, рівні, належність до кланів, лідерборди, винагороди за досягнення певних цілей тощо. Методи гейміфікації використовуються для залучення користувачів до виконання бажаної дії, підвищення мотивації, зниження відчуття монотонності та нудності [9].

Основні методи гейміфікації в освіті включають:

- гра: Використання ігрових елементів для викладання нових знань та навичок. Наприклад, використання ігрових симуляцій для вивчення науки, історії або математики.
- бали та досягнення: Система нагородження учнів за виконання певних дій або досягнення цілей. Наприклад, створення системи балів за виконання домашніх завдань, відповіді на питання або виконання додаткових завдань.
- ранги та рейтинги: Система порівняння учнів за результатами їхньої діяльності. Наприклад, створення рейтингу класу за кількістю балів, отриманих за домашні завдання, відповіді на питання та інші дії.
- мисії та завдання: Система виконання певних дій для отримання бонусів або досягнення цілей. Наприклад, створення завдань на кожен урок, виконання яких допоможе учням закріпити отримані знання та навички.
- персонажі та історії: Система створення персонажів або історій для залучення учнів до навчального процесу. Наприклад, створення історії про винахідника, яка допоможе учням краще зрозуміти історію розвитку науки.
- соціальна інтеракція: Система взаємодії між учнями та вчителем, що допомагає створювати комунітет та залучати нових учнів. Наприклад,

використання форумів для обговорення матеріалу, викладеного на уроці або спільного виконання завдань у групі.

Згідно з дослідженням, проведеним Корпорацією IBM, 90% працівників, які взяли участь в програмі навчання, що використовувала елементи гейміфікації, заявили, що такий підхід допоміг їм краще зрозуміти матеріал та підвищити рівень мотивації [10].

Також, згідно зі звітом компанії TalentLMS, використання гейміфікації в навчальних програмах збільшує участь учасників на 40%, підвищує рівень розуміння матеріалу на 55%, а також збільшує шанси успішного завершення курсу на 50% [11].

### 1.3.1 Метод гейміфікації "Гра"

Метод гейміфікації "Гра" включає використання ігрових елементів у процесі навчання, що дозволяє зробити процес більш захопливим та привабливим для учнів.

У цьому методі навчання можуть брати участь різні ігрові елементи, такі як графіка, звук, механіка гри, анімація, відео та інші. Наприклад, можна використовувати ігрові симулятори для вивчення наукових понять, створювати ігрові завдання для закріплення вивченого матеріалу або використовувати ігрову механіку для вирішення навчальних завдань.



Рисунок 1.2 – Складові концепти метода гри [12]

При використанні методу гейміфікації "Гра" учні стають активними учасниками навчального процесу, бо вони займаються не просто слуханням лекцій або читанням матеріалу, а активно взаємодіють з ігровими елементами, вирішують ігрові завдання та отримують миттєву зворотну інформацію про свої успіхи.

Крім того, метод гейміфікації "Гра" допомагає збільшити мотивацію до навчання та зацікавленість учнів у вивченні матеріалу, оскільки ігрові елементи допомагають зробити процес навчання більш захоплюючим та цікавим.

Чотири концепти, в основі яких лежить ідея гри: гейміфікація, серйозні ігри, забавки та ігровий дизайн [13]. Відмінності між ними лежать у двох вимірах:

- ігри/забавки (gaming/playing) – свідчить про спрямування та регульованість діяльності;
- повністю/частково (whole/parts) – вказує на міру інтеграції ігрових елементів у процес.

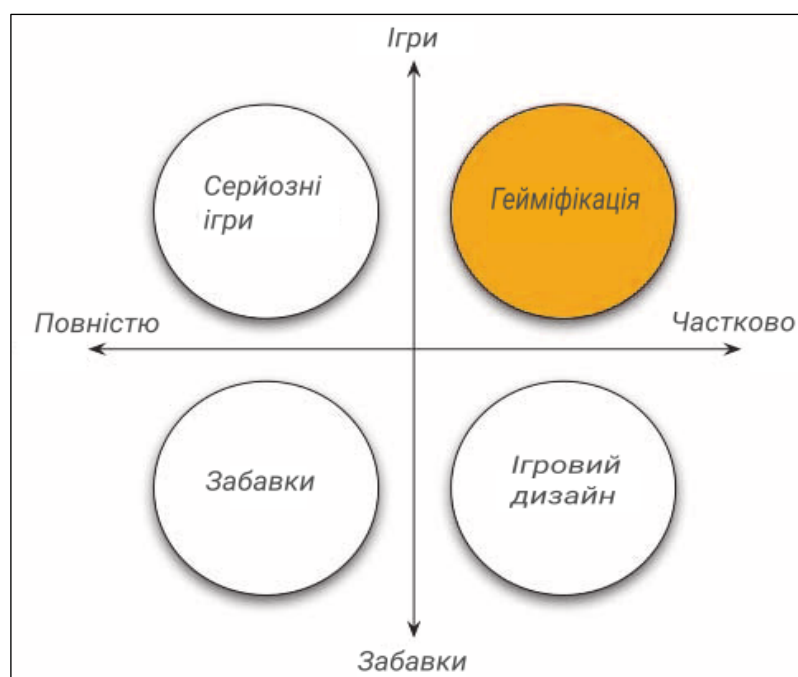


Рисунок 1.3 – Концепти, в основі яких лежить ідея гри [14]

Серйозні ігри (serious games) мають конкретну мету, націлену на розв'язання реальних життєвих ситуацій.

Забавки (toys) – це ігри, що не мають чітких правил, а також не націлені на конкретний результат чи мету; вони орієнтовані лише на переживання позитивних емоцій або невимушене дослідження.

Ігровий дизайн (playful design) теж не має конкретної мети, які підтримується правилами; його використовують, щоби зробити процес більш людським, приємним та легким для сприйняття.

Гейміфікація використовує елементи гри, але основа процесу залишається незмінною. Втім, межі між цими концептами є радше теоретичними, натомість, на практиці вони можуть успішно поєднуватись [15].

### 1.3.2 Метод гейміфікації "Бали та досягнення"

Метод гейміфікації "Бали та досягнення" полягає у використанні системи нагород та досягнень для стимулювання та мотивації користувачів до активного навчання та досягнення поставлених цілей.

Зазвичай користувач отримує бали за успішне виконання завдань та проходження різних рівнів складності. За накопичені бали користувач може отримати досягнення та піднятися на наступний рівень.

Наприклад, користувач може отримувати бали за правильне вирішення задач, за час, витрачений на їх вирішення, за виконання додаткових завдань та інші дії, пов'язані з процесом навчання. Кожен набраний бал може додати користувачеві певну кількість досвіду та рейтингу.

Отримання досягнень може стати додатковим стимулом для користувачів. Наприклад, користувач може отримати досягнення за успішне виконання завдань певного рівня складності, за проходження курсу, за виконання додаткових завдань тощо. Кожне досягнення може мати свій власний опис та зображення, що зробить навчання більш захопливим та інтерактивним. На рисунку 1.4 наведено приклади таких досягнень[16].

Крім того, додаток може містити список досягнень, які студент може отримати при досягненні певних мет. Наприклад, "Майстер програмування" - за

успішне виконання великої кількості завдань різної складності, або "Експерт" - за успішне виконання всіх завдань на певному рівні складності.

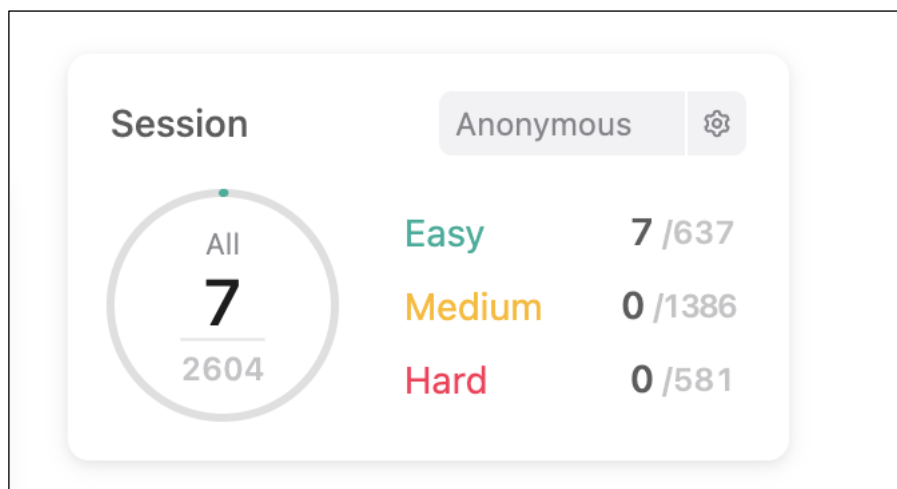


Рисунок 1.4 – Накопичені бали користувача leetcode

Отже, метод гейміфікації "Бали та досягнення" може допомогти зробити процес навчання програмування більш захопливим та ефективним для користувачів. Використання системи нагород та досягнень може стимулювати користувачів до більш активного та цілеспрямованого навчання [17].

### 1.3.3 Метод гейміфікації "Ранги та рейтинги"

Метод гейміфікації "Ранги та рейтинги" полягає у використанні системи позиційного ранжування користувачів за їхнім досягненням та успіхами в процесі навчання.

Цей метод може бути реалізований шляхом введення рейтингу користувачів, який буде відображати їхній рівень знань та успішність у вирішенні завдань. Користувачі, які більше успішно пройшли курс, отримують вищі позиції у рейтингу.

Рейтинг можна відображати як загальний, так і в рамках конкретної категорії (наприклад, мови програмування, рівня складності завдань тощо). Також можна ввести систему рангів, які будуть надавати користувачам певні привілеї та можливості.

Крім того, введення рейтингу може сприяти взаємодії та конкуренції між користувачами. Вони зможуть порівнювати свій рівень знань та успішність з іншими користувачами, стимулювати один одного до активного навчання та досягнення більш високих результатів.

Отже, метод гейміфікації "Ранги та рейтинги" може бути ефективним інструментом для підвищення мотивації та зацікавленості користувачів у процесі навчання програмування. Використання системи рангів та рейтингів може допомогти створити конкурентне середовище, що стимулюватиме користувачів до досягнення кращих результатів та підвищення їхнього рівня знань[18].

#### 1.3.4 Метод гейміфікації "Місії та завдання"

Метод гейміфікації "Місії та завдання" використовується для створення відчуття гри та пригоди в процесі навчання програмування. Основна ідея полягає в тому, щоб перетворити навчання на пригоду з набором завдань, які необхідно виконати, щоб здобути нові знання та навички.

Цей метод може бути реалізований через створення різноманітних місій та завдань для користувачів. Наприклад, користувач може отримувати завдання на написання певного блоку коду, вирішення складної задачі, створення власного проекту тощо. За кожне успішно виконане завдання користувач отримує бали, які додаються до його рахунку.

Для того, щоб користувач відчував себе у грі, можна використовувати сюжетну лінію, яка відобразить його прогрес у навчанні. Наприклад, кожне нове завдання може представляти наступний етап пригоди, в якій герой (користувач) отримує нові знання та навички для подолання наступних викликів.

Крім того, в додатку можна використовувати бонусні завдання, які дають додаткові бали та навички, а також можуть мати певні призи за їх успішне виконання.

Таким чином, метод гейміфікації "Місії та завдання" може стати відмінним інструментом для підвищення мотивації та зацікавленості користувачів у навчанні

програмування. Створення пригодницької сюжетної лінії та різноманітних завдань може допомогти користувачам забути про монотонність навчання та зануритися у світ гри, де їхні успіхи будуть визнані та винагороджені. Крім того, ранги та рейтинги можуть бути також використані для стимулювання користувачів до досягнення нових цілей та досягнень у навчанні.

Також можна використовувати ранги для визначення рівня володіння користувачем певними навичками програмування. Кожен новий рівень може відображати певні вимоги щодо знань та навичок, які потрібно володіти, щоб досягти наступного рівня. Це може стати чудовою мотивацією для користувачів вдосконалювати свої знання та навички, адже вони будуть мати явну мету та ціль для досягнення.

Крім того, рейтинги можуть стимулювати користувачів до більш активної участі в навчальному процесі та виконанні завдань. За кожне успішно виконане завдання користувач отримує певну кількість балів, які додаються до його рахунку. Ці бали можуть використовуватися для створення рейтингу, в якому будуть відображатися найактивніші та найкращі учасники додатку. Це може створити здорову конкуренцію між користувачами, що дозволить їм не тільки підвищити свій рівень знань та навичок, але й отримати додаткову мотивацію досягати нових результатів у навчанні програмування.

Отже, метод гейміфікації "Місії та завдання" може бути дуже ефективним інструментом для навчання програмування. Він допоможе зробити навчання більш захоплюючим та мотивуючим для користувачів, що дозволить їм досягти кращих результатів у навчанні[19].

### 1.3.5 Метод гейміфікації "Персонажі та історії"

Метод гейміфікації "Персонажі та історії" може бути використаний для навчання програмування, якщо створити додаток, який буде містити історії та персонажів, пов'язаних з програмуванням. Цей метод полягає в тому, що

користувачі можуть отримувати інформацію про програмування через історії та персонажів, які допомагають зрозуміти складні концепції та процеси.

Наприклад, можна створити персонажів, які будуть представляти різні аспекти програмування. Уявимо, персонаж "Кодер" може розповісти про основні принципи програмування, а персонаж "Тестувальник" може розповісти про тестування програмного забезпечення. Кожен персонаж може мати власну історію, яка допоможе користувачам зрозуміти складні концепції та процеси, пов'язані з програмуванням.

Крім того, персонажі можуть бути використані для створення завдань та викликів, пов'язаних з програмуванням. Наприклад, персонаж "Кодер" може запропонувати користувачеві завдання, пов'язані з написанням простого коду, а персонаж "Тестувальник" може запропонувати завдання, пов'язані з тестуванням програмного забезпечення. Ці завдання можуть мати різні рівні складності, що дозволить користувачам поступово підвищувати свій рівень знань та навичок у програмуванні.

Крім персонажів та історій, метод гейміфікації "Персонажі та історії" також може використовувати бали та досягнення для стимулювання користувачів до досягнення нових цілей та досягнень у навчанні. Кожен успішно виконаний виклик або завдання може приносити користувачеві бали, які він може витратити на отримання нової інформації, підвищення рівня складності завдань або отримання інших бонусів. Крім того, досягнення можуть бути використані як додатковий стимул для користувачів до виконання завдань та досягнення успіху у навчанні.

Застосування методу гейміфікації "Персонажі та історії" у навчанні програмування може допомогти зробити процес навчання більш захоплюючим та цікавим для користувачів. Це може бути особливо корисно для новачків, які можуть відчувати себе загубленими в складних концепціях та процесах програмування. Використання персонажів та історій може допомогти їм зрозуміти

складні концепції та процеси у контексті, що більш зрозумілий та доступний для них[20].

Загалом, метод гейміфікації "Персонажі та історії" може бути корисним інструментом для навчання програмування, оскільки він дозволяє зробити процес навчання більш захоплюючим та цікавим для користувачів. Використання цього методу може допомогти користувачам зрозуміти складні концепції та процеси програмування, а також збільшити їх мотивацію до вивчення цієї галузі.

### 1.3.6 Метод гейміфікації "Соціальна інтеракція"

Цей метод гейміфікації включає в себе використання елементів ігор та змагань, що спрямовані на взаємодію між користувачами. Це може включати в себе використання рейтингів, змагань та співпраці між користувачами.

У контексті навчання програмування, метод гейміфікації "Соціальна інтеракція" може бути використаний, наприклад, для створення спільноти програмістів, де користувачі можуть обговорювати технічні проблеми та рішення, а також допомагати один одному у вирішенні завдань. Така спільнота може включати в себе рейтинги та систему нагород, що дозволить користувачам отримувати бали та досягнення за активну участь у спільноті.

Крім того, метод гейміфікації "Соціальна інтеракція" може бути використаний для проведення змагань між користувачами з різних країн або навіть між різними університетами. Наприклад, можна провести змагання зі створення найкращого програмного продукту, або виконання певних завдань програмування. Користувачі можуть конкурувати між собою за місця в рейтингу та отримувати нагороди за перемогу.

Застосування методу гейміфікації "Соціальна інтеракція" у навчанні програмування може допомогти збільшити мотивацію та зацікавленість користувачів до вивчення цієї галузі, а також дозволить залучити до навчання більше людей та створити активну спільноту програмістів.

I, нарешті, останній метод гейміфікації - соціальна інтеракція. Цей метод передбачає використання в додатку соціальних функцій, що дає можливість користувачам спілкуватися між собою, ділитися своїми досягненнями та отримувати поради від інших програмістів[21].

Наприклад, у додатку можна створити форум, де користувачі зможуть обговорювати свої успіхи та проблеми, запитувати поради від інших користувачів. Також можна розробити систему "наставників", де досвідчені програмісти можуть допомагати новачкам у вирішенні складних завдань.

Цей метод також можна поєднати з попередніми методами гейміфікації, наприклад, створити гру, де користувачі можуть спільно виконувати завдання та змагатися між собою.

В цілому, застосування методів гейміфікації у додатку для навчання програмування може значно збільшити мотивацію користувачів до навчання, зробити процес навчання більш захоплюючим та ефективним.

#### 2.4 Проблеми в гейміфікації: аналіз та вирішення

Основних проблеми, пов'язані з використанням методів гейміфікації у навчанні та інших сферах, можуть бути різними в залежності від конкретної ситуації та додатку, що використовує гейміфікацію. Однак, розуміння цих проблем може допомогти розробникам додатків зробити більш обдумані та ефективні рішення.

Слід зазначити, що ефективність гейміфікованих систем навчання залежить від багатьох факторів, таких як конкретні завдання та цілі навчання, мотивація та індивідуальні особливості студентів. Також необхідно враховувати недоліки гейміфікації, які можуть вплинути на її ефективність, такі як можливість штучного збільшення результатів та ризик втрати інтересу студентів після відчуття перенасиченості гейміфікованої системи навчання [22].

Основні проблем:

- перевірка ефективності: Існує питання щодо того, наскільки ефективні і результативні гейміфіковані системи порівняно з традиційними системами навчання та розвитку. Наприклад вибір неправильних метрик може призвести до того, що користувачі будуть фокусуватись на досягненні певних цілей, а не на розвитку навичок, які є дійсно важливими для їх роботи. Необхідно ретельно вибирати метрики, щоб вони відображали реальну продуктивність користувача;[23]
- мотивація: Хоча гейміфікація може стати потужним інструментом для стимулювання навчання, вона може бути неефективною, якщо користувачі не мають достатньої мотивації для досягнення поставленої мети. Важливо знайти способи збільшення мотивації, щоб забезпечити успішну імплементацію гейміфікації. Гейміфікація може стати невдалим, якщо не вдасться належним чином мотивувати учасників. Це може статися, якщо завдання, які пропонуються в гейміфікованій системі, є надто складними або надто простими;
- монотонність: Гейміфіковані системи можуть стати монотонними, якщо завдання не будуть варіюватися належним чином, що може призвести до втрати інтересу учасників;
- ризик залежності: Однією з основних проблем може бути залежність користувачів від гейміфікованої системи. Як і в інших випадках, занадто часта гра або використання гейміфікованих систем може призвести до залежності, що може бути проблемою з точки зору здоров'я та продуктивності. Відчуття залежності може знизити мотивацію користувачів до самостійного навчання, оскільки вони будуть очікувати нових гейміфікованих елементів для отримання задоволення від процесу навчання. Бейджі або нагороди можуть бути ефективними на початку, але не сприяти внутрішній мотивації та саморозвитку;[24]
- недостатня затребуваність: Навіть якщо гейміфікована система є ефективною, вона може стати недостатньо затребуваною через

відсутність змістовного контексту для користувачів, або через те, що вона не відповідає їхнім потребам. Також можливою проблемою є відсутність індивідуалізації гейміфікованої системи. Якщо система не враховує індивідуальні потреби та рівні користувачів, це може призвести до нерівності між користувачами та знизити їхню мотивацію до навчання;[25]

- етичні проблеми: Існує ризик, що гейміфіковані системи можуть бути використані для маніпулювання або підкупу учасників, що може бути неетичним;
- фінансові проблеми: Впровадження гейміфікації може бути витратним процесом, особливо для менших компаній або організацій. Деякі можуть не мати достатньої фінансової можливості для реалізації.

Ці проблеми не є неодмінними наслідками використання методів гейміфікації, але можуть виникати при неправильному або недбалому підході до їхньої імплементації. Щоб уникнути цих проблем, важливо враховувати наступні аспекти:

**Чітка спрямованість:** Методи гейміфікації повинні бути прямо пов'язані з основною метою навчання або бізнес-задачею. Вони повинні сприяти досягненню конкретних цілей і підвищувати результативність учнів або користувачів.

**Варіативність:** Різноманітність гейміфікаційних елементів і завдань допомагає уникнути монотонності та забезпечити цікавість і різноманітність учнів або користувачів.

**Індивідуалізація:** Варто враховувати індивідуальні потреби, навички та інтереси учнів або користувачів. Гейміфікація повинна бути адаптована до конкретних особливостей і надавати можливості для індивідуального прогресу та розвитку.

**Внутрішня мотивація:** Націлитися на розвиток внутрішньої мотивації, яка базується на самозадоволенні, саморозвитку та задоволенні від самого процесу

навчання чи роботи. Гейміфікація може стати стимулом, але вона не повинна бути єдиним або основним джерелом мотивації.

Баланс: Важливо знайти правильний баланс між гейміфікаційними елементами та основними завданнями навчання або роботи. Надмірне використання гейміфікації може викликати залежність від зовнішніх стимулів та зробити навчання або роботу залежними від гейміфікаційних елементів. Тому важливо забезпечити, щоб гейміфікація була лише одним з інструментів, що сприяють навчанню чи роботі, а не основним джерелом мотивації.

Окрім того, реалізація гейміфікації може вимагати додаткових ресурсів, таких як час, фінанси та технічні можливості. Необхідно забезпечити належну підтримку та ресурси для розробки, впровадження та підтримки гейміфікаційної системи.

Однак, важливо пам'ятати, що гейміфікація не є універсальним рішенням для всіх ситуацій. В деяких випадках, особливо у більш серйозних навчальних або професійних контекстах, важливість серйозного підходу та глибокого розуміння предметної області можуть переважати гейміфікаційні елементи.[26]

Крім того, існують етичні аспекти, які повинні бути враховані при використанні гейміфікації. Наприклад, створення здорової конкуренції серед учнів або користувачів може бути корисним, але необхідно уникати створення ворожих або нездорових атмосфер. Також слід забезпечувати рівномірний доступ до гейміфікованого навчання або роботи, щоб уникнути нерівності.

Усі ці проблеми можуть бути вирішені через глибоке дослідження, тестування та вдосконалення методів гейміфікації. Важливо проводити збалансовану оцінку переваг і недоліків, а також слухати повернення від учнів, користувачів та інших зацікавлених сторін.

Використання методів гейміфікації може стати потужним інструментом для покращення навчання та стимулювання учнів або користувачів. Проте, варто розглядати його як один із компонентів ширшого підходу до навчання та мотивації, враховуючи контекст, потреби та особливості цільової аудиторії.[27]

У підсумку, використання методів гейміфікації в навчанні та інших сферах може бути ефективним і стимулюючим, але вимагає обережного підходу та урахування різних факторів. Правильне використання гейміфікації може покращити мотивацію, залученість та результативність учнів або користувачів, створюючи стимулююче та захоплююче навчальне середовище.

## 2. ПІДХІД В РОЗРОБЦІ ГЕЙМІФІКОВАНОЇ СИСТЕМИ НАВЧАННЯ

У цій роботі пропонується підхід до вирішення завдання щодо гейміфікації у додатку для навчання програмування в компанії. Зокрема, вивчаються різні методи гейміфікації та їх вплив на користувачів, а також аналізується ефективність використання гейміфікації для навчання програмування.

Завдання полягає в ідентифікації найбільш поширених проблем, пов'язаних з використанням гейміфікації для навчання програмування, та визначенні шляхів їх вирішення. Крім того, метою є запропонувати певні підходи до використання гейміфікації, які можуть бути більш ефективними для навчання програмування в компанії. Робота має на меті дослідження впливу гейміфікації на мотивацію користувачів та їхню успішність у навчанні програмування. Досліджуються, які елементи гейміфікації можуть бути найбільш стимулюючими для користувачів та можуть допомагати їм зберігати мотивацію та зацікавленість у процесі навчання.

Отже, головною метою системи навчання з гейміфікацією є підвищення якості роботи програмістів. Це досягається за допомогою створення цікавих та захоплюючих завдань з програмування, які мають вигляд ігор. Цільовою аудиторією системи є співробітники технічних компаній, які бажають покращити свої знання та навички в галузі програмування. Для досягнення цієї мети, розглядається використання різних підходів та методів гейміфікації у системі навчання, включаючи вибір гейміфікаційних елементів та їхню інтеграцію в систему.[28]

### 2.1 Вибір основних підходів та методів гейміфікації

Для ефективної гейміфікації системи навчання програмування можна використовувати такі підходи та методи:

Використання нагород та бейджів за досягнення певних меток та завдань. Наприклад, учень може отримувати бейджі за успішне виконання завдань певного рівня складності або за досягнення певного рівня знань у певній темі. Це

допомагає стимулювати мотивацію та підтримувати інтерес учнів до навчання [29].

Використання візуалізації та графіки. Цей підхід полягає у використанні графічних елементів для показу прогресу користувача, завдань та досягнень. Візуалізація допомагає користувачам бачити свій прогрес, що мотивує їх до подальших зусиль.

Використання лідербордів та групової роботи. Цей підхід заснований на конкуренції між користувачами та їхньої спільної роботи. Користувачі можуть брати участь у турнірах, де вони змагаються між собою за кращі результати, а також працювати у групах, де вони спільно вирішують завдання.

Також можливість різних форм фідбеку та оцінювання. Учні повинні отримувати зворотний зв'язок щодо своїх досягнень, дізнаватися про свої помилки та отримувати підказки, як їх виправити. Це допомагає підтримувати мотивацію та розвивати навички учнів.

Використання історій та квестів. Цей підхід допомагає користувачам зануритися у світ програмування, створивши історії та квести, пов'язані з програмуванням. Користувач може проходити квести та виконувати завдання, які розкривають історію та культуру програмування [30].

Або, ігровий світ або квест з обмеженням часу, що потрібно виконати завдання, або використання елементів "битви" або змагань, щоб стимулювати змагання між учнями. Ці методи допомагають створити більш захоплюючу та цікаву навчальну атмосферу. Крім того, ефективним є індивідуальний підхід до учнів, врахування їхніх особливостей та інтересів. Наприклад, створення персоналізованих завдань.

Важливо враховувати, що додаток повинен бути легким та зручним у використанні, а також мати зрозумілу систему мотивації та нагород.

Основні причини, чому це важливо:

Швидкість роботи: Легкий додаток має менші вимоги до ресурсів пристрою та швидше працює, що є важливим для стабільної роботи та задоволення користувачів.

Зручність використання: Легкий додаток легше використовувати на різних пристроях, включаючи ті, що мають обмежені обсяги пам'яті та обробки даних.

Економія ресурсів: Легкий додаток використовує менше ресурсів пристрою, таких як батарея, що забезпечує більш тривалу роботу пристрою без підзарядки. Легкий додаток має менші вимоги до розробки та тестування, що зменшує загальну вартість розробки та підтримки додатка. Вартість ресурсів для підтримки додатків може значно відрізнятись в залежності від багатьох факторів, таких як розмір додатка, кількість користувачів, необхідність серверних потужностей, необхідність використання додатком зовнішніх API тощо. Окрім того, вартість підтримки додатків може змінюватись з часом, оскільки вона залежить від необхідності в поновленнях програмного забезпечення, виправленнях помилок, підтримці безпеки та інших чинників.

Отже, легкий додаток для навчання програмування є важливим для забезпечення ефективної та зручної роботи користувачів, економії ресурсів пристрою та зменшення вартості розробки.

## 2.2 Аналіз існуючих додатків для навчання програмування

Існує багато додатків для навчання програмування.

CodinGame – це платформа, яка пропонує різні головоломки та завдання з програмування, оформлені у вигляді ігор. Ви можете вибрати завдання з різних мов програмування та виконувати їх, отримуючи бали та зрівні.

Один з найцікавіших аспектів CodinGame – це мультиплеєрні турніри. Ви можете створювати своїх роботів або програми та змагатися з іншими користувачами у різних типах ігор. Це випробування вашого програмування та стратегічного мислення в реальному часі.

CodinGame також пропонує можливості для розвитку навичок програмування через практику та взаємодію з іншими користувачами. Ви можете ділитися своїми рішеннями, отримувати зворотній зв'язок, а також досліджувати та вдосконалювати свої навички, працюючи над складнішими завданнями.

Додаток відомий своєю високою якістю завдань та ігрових сценаріїв, а також активною спільнотою програмістів. Ви зможете відчути справжній дух змагань, розвивати свої навички та насолоджуватися навчанням програмування через гейміфікований підхід.

Codecademy – це один з найбільш популярних онлайн-курсів програмування, який пропонує навчання на різних мовах програмування, таких як Python, JavaScript та інших. Він має добре структурований матеріал і навчання відбувається шляхом виконання завдань. Однак, цей додаток може бути корисним тільки для початківців, оскільки він не надає детального пояснення того, як працює код.

LeetCode – це онлайн-платформа для практики алгоритмів та задач програмування, яка є популярною серед програмістів, які готуються до технічних співбесід. Цей додаток надає велику колекцію завдань з різних областей програмування, таких як алгоритми, структури даних, математика, бази даних та багато іншого.

Основна мета LeetCode полягає в тому, щоб допомогти програмістам підготуватися до складних технічних співбесід та вдосконалити свої навички програмування. Завдання на платформі LeetCode орієнтовані на реальні сценарії, з якими можна зіткнутися у вирішенні реальних проблем у програмуванні.

У LeetCode ви можете вибирати завдання різного рівня складності, від початкових до дуже високих, залежно від вашого рівня знань та досвіду. Кожне завдання супроводжується поясненням умови, прикладами вхідних та вихідних даних, а також можливими підказками для вирішення.

Один з основних функціональних елементів LeetCode – це онлайн-інтерактивне середовище, де ви можете писати свій код прямо на платформі та

перевіряти його правильність на вході-виході. Після написання коду ви можете перевірити його на правильність та ефективність, і LeetCode автоматично перевірить ваше рішення та надасть результат.

Додаток також має можливість перегляду та обговорення рішень інших користувачів, що дозволяє вам вивчити різні підходи до вирішення задач та покращити свої вміння програмування.

Udemy – це інший онлайн-курс для навчання програмування. У ньому є велика кількість курсів з різних мов програмування, фреймворків та інших технологій. Він має високу якість викладання та детальні матеріали, що дозволяє навчитись від базових до високорівневих тем. Однак, в ньому є нестача інтерактивності та можливості отримувати фідбек.

Також додаток Udemy має різноманіття викладачів з різними підходами та гнучкість в плануванні навчання. У користувачів є можливість вчитися у своєму власному темпі та вибирати курси, які найкраще відповідають їхнім потребам. Однак, є і недоліки, які можуть включати обмежену інтерактивність курсів та відсутність особистого підходу до навчання. Деякі користувачі також вказують на нерівномірну якість викладачів та курсів на платформі. Враховуючи ці фактори, користувачам слід ретельно вибирати курси та викладачів, щоб забезпечити якісне навчання.

Coursera – це платформа онлайн-навчання, яка співпрацює з університетами та організаціями з усього світу, щоб пропонувати безкоштовні та платні курси з програмування. Він має високу якість викладання та можливість взаємодії з іншими студентами. Однак, він може бути дорогим, і для взаємодії потрібно використовувати форуми, що не забезпечує таку ж інтерактивність, як і пряма комунікація.

edX – це платформа онлайн-навчання, заснована на співпраці між університетами та організаціями. Вона пропонує безкоштовні та платні курси з програмування та інших предметів. EdX пропонує широкий вибір курсів з різних галузей з викладанням від провідних університетів та організацій. Це дозволяє

користувачам знайти курси, які відповідають їхнім інтересам і навчальним потребам.

Однією з головних переваг EdX є висока якість навчання. Курси розроблені провідними експертами відповідних галузей, що гарантує доступ до найновішої інформації та кращих практик.

EdX також надає гнучкість і самостійність. Користувачі можуть вчитися у власному темпі та вибирати розклад навчання, що найбільш підходить їм.

Крім того, EdX часто пропонує сертифікати після успішного завершення курсів, що може бути корисним для покращення професійних перспектив.

Один з основних недоліків EdX полягає в тому, що деякі курси вимагають плати за отримання сертифікату або доступу до певного матеріалу. Це може бути обмеженням для деяких користувачів, особливо з фінансово обмеженими можливостями.

Також варто враховувати, що хоча EdX надає великий вибір курсів, можуть виникати обмеження щодо доступності певних курсів або мов в різних регіонах.

Деякі користувачі також можуть вважати, що самостійне навчання на платформі EdX вимагає більшої самодисципліни і мотивації, оскільки немає жорстких дедлайнів або викладачів, які нагляда

SoloLearn – це мобільний додаток для навчання програмування, доступний для iOS та Android. Він пропонує безкоштовні курси з багатьох мов програмування, а також можливість взаємодії з іншими користувачами. Додаток використовує систему балів, нагород та досягнень для мотивації користувачів до навчання та досягнення нових результатів.

FreeCodeCamp – це онлайн-курс з програмування, який пропонує навчання на різних мовах програмування, фреймворків та інших технологій. Окрім того, він надає можливість вирішувати завдання та проходити випробування, щоб отримувати сертифікати. Переваги FreeCodeCamp:

- безкоштовний доступ до усіх матеріалів та ресурсів;

- підтримка великої спільноти користувачів, що допомагає розв'язати будь-які труднощі;
- зосередженість на практичних завданнях, що дозволяє студентам навчитися розробляти додатки в реальному світі та додавати ці проекти у свої резюме;
- пропонує навчання не тільки програмуванню, але і іншим навичкам, таким як розробка веб-сайтів, робота з базами даних та технології машинного навчання.

Недоліками є те, що цей додаток потребує великої самодисципліни та мотивації, оскільки весь процес навчання відбувається онлайн та самотійно. І саме гейміфікація вирішує ці проблеми.

### 2.3 Опис проектування додатка для підвищення якості роботи програмістів

Після аналізу існуючих додатків та аналізу методів гейміфікації, що розглядаються в попередньому розділі, результатом є проектування додатку, який за рахунок методів гейміфікації буде підвищувати якість роботи програмістів - це ігровий додаток тренажер.

Назва додатку "CodeJourney".

Основною метою CodeJourney є підвищення рівня вмінь програмістів та покращення навичок програмування, а також мотивування програмістів на постійне покращення. Користувачі зможуть виконувати різні ігрові завдання, пов'язані з програмуванням, що допоможе їм покращити розуміння кодування, алгоритмів та програмних мов.

Додаток зосереджений на оцінці якості роботи програмістів за допомогою гейміфікованих методів. Він буде містити завдання, які можна розв'язувати у вигляді головоломок або ігрових рівнів, що оцінюють рівень знань та навичок програмістів.

Також додаток у майбутньому буде мати можливість створювати групові завдання, щоб сприяти співпраці та комунікації між програмістами, що сприятиме

кращій якості роботи. Крім того, додаток буде мати функцію, що надає зворотний зв'язок та рекомендації щодо покращення роботи програмістів, що допоможе їм розвиватися та покращувати свої навички.

Основну проблемою, яку буде вирішувати додаток, це використання ефективних методів гейміфікації для розвитку програмістів, які будуть відповідати потребам різних груп програмістів та забезпечувати оптимальний рівень засвоєння знань у галузі програмування. Важливим елементом є адаптивність додатка, що дозволяє пристосовуватися до потреб та рівня навчання кожного окремого користувача. Це може бути досягнуто за допомогою індивідуальної настройки навчальних програм, оцінювання рівня знань та автоматичного вибору оптимального рівня складності завдань.

Технології.

Frontend:

- ReactJS
- React Router
- Material UI

Backend:

- Node.js
- GraphQL
- Firebase

Гейміфікація:

- бали та досягнення за успішне виконання завдань;
- рейтинг користувачів;
- лідерборди;
- профіль користувача з відстеженням його прогресу.

Головні функції:

- завдання для програмування на різних мовах програмування, таких як Python, Java, JavaScript тощо;

- різні рівні складності, що дають можливість гравцям починати зі зручного для них рівня та поступово підвищувати рівень складності;
- показник успішності, який дозволить гравцям відслідковувати свій прогрес та дізнаватись про свої слабкі місця в програмуванні;
- функція "бойових арен", де гравці можуть змагатися один з одним в програмуванні та заробляти додаткові бали та винагороди;
- можливість створення груп, де гравці можуть допомагати один одному та спільно виконувати завдання.

"CodeJourney" може бути використаний як для самостійного навчання, так і для організації навчальних курсів та тренінгів для компаній та університетів. Завдяки цьому додаток може бути корисним для різних категорій користувачів - від початківців до досвідчених програмістів, які хочуть покращити свої навички в певних мовах програмування або в певних областях.

Також, "CodeJourney" може мати великий потенціал для розвитку та вдосконалення за рахунок постійного оновлення завдань та додавання нових функцій, що дозволить залучати більше користувачів та зберігати їх зацікавленість у навчанні програмування.

### 3. ПРОГРАМНА РЕАЛІЗАЦІЯ СИСТЕМИ НАВЧАННЯ

#### 3.1 Опис використовуваних засобів розробки

В процесі розробки програмного продукту, включаючи гейміфіковані системи навчання, вибір правильних засобів програмної реалізації є одним з ключових аспектів. Обрані засоби визначають не лише ефективність та продуктивність розробки, але й якість та функціональність самого продукту.

При обиранні засобів програмної реалізації важливо враховувати такі фактори:

**Функціональність:** Засоби повинні задовольняти потреби проекту та забезпечувати необхідну функціональність. Наприклад, якщо проект передбачає використання фронтенд-фреймворка, важливо обрати той, який надає потрібні можливості для розробки інтерактивних інтерфейсів [31].

**Відкритість та підтримка:** Вибрані засоби повинні мати активну спільноту розробників та бути підтримуваними. Це дозволить отримати допомогу в разі потреби, а також забезпечити актуальні версії, патчі та нові функції.

**Досвід команди:** Важливо врахувати досвід та навички команди розробників. Якщо у команди вже є експерти з певними засобами, це може бути перевагою при обиранні [32].

**Масштабованість:** Засоби повинні бути гнучкими та масштабованими, щоб вони могли відповідати потребам розробки із зростанням проекту. Наприклад, обрані фреймворки повинні підтримувати легку інтеграцію нових функціональностей та забезпечувати оптимальну продуктивність навіть при великому обсязі даних.

**Вартість:** Важливо враховувати вартість використання засобів програмної реалізації. Це може включати вартість ліцензій, підтримки, навчання та інфраструктури. Для багатьох проектів, особливо стартапів або освітніх ініціатив, важливо знайти доступні засоби, які задовольняють потреби проекту, не перевищуючи бюджет.

Обирання засобів програмної реалізації вимагає збалансованого підходу і врахування різних факторів. Не існує універсального рішення, яке підходило б для всіх проектів. Тому, залежно від потреб та контексту, може знадобитись детальний аналіз, порівняння та вибір найбільш підходящих засобів програмної реалізації.

Засоби розробки, які були використані під час розробки гейміфікованої системи навчання програмування:

Редактор коду – Visual Studio Code (VS Code). В якості основного редактора коду був використаний Visual Studio Code. VS Code є популярним та потужним редактором коду, який надає широкий набір функцій та розширень для комфортної розробки програмного забезпечення. Він забезпечує підсвічування синтаксису, автодоповнення коду, інтегроване керування версіями та багато інших корисних функцій.

Мова програмування – JavaScript. JavaScript є однією з найпоширеніших мов програмування для веб-розробки. Використання мови JavaScript в розробці додатку CodeJourney було обумовлено кількома факторами.

По-перше, JavaScript є широко поширеною мовою програмування, яка використовується як на стороні клієнта (веб-браузері), так і на стороні сервера (за допомогою Node.js). Це означає, що ми зможемо створити додаток, який працює як на веб-сторінці, так і на сервері, забезпечуючи зручну інтерактивну навчальну платформу для програмістів.

По-друге, JavaScript є мовою з динамічним типом даних, що надає гнучкість у роботі з даними та забезпечує легкість розробки. Вона також підтримує функціональне та об'єктно-орієнтоване програмування, що дозволяє реалізовувати складні логічні операції та структури даних.

По-третє, JavaScript має широкий набір бібліотек та фреймворків, які полегшують розробку додатків. Наприклад, ми використовуємо React, який надає ефективну систему компонентів та стану, спрощуючи процес розробки і підтримки додатку.

По-четверте, JavaScript є мовою, яку багато програмістів вже знають або вивчають, оскільки вона є основою веб-розробки. Використання JavaScript дозволяє залучити більше розробників до проекту і забезпечити більшу доступність та прийняття додатку.

Загалом, використання JavaScript у розробці CodeJourney дозволяє створити потужну, гнучку та доступну платформу для навчання програмування, яка підтримує різні платформи та забезпечує комфортну розробку та використання для програмістів різного рівня.

Фреймворк – React. Для розробки фронтенд-частини системи навчання був використаний фреймворк React. React є популярним JavaScript-фреймворком, який дозволяє створювати динамічні та інтерактивні веб-інтерфейси. Фреймворк React був обраний для розробки застосунку CodeJourney з кількох причин.

По-перше, React є одним з найпопулярніших та широко використовуваних фреймворків для розробки користувацьких інтерфейсів у веб-додатках. Він має велику спільноту розробників, багато ресурсів та документації, що спрощує роботу з ним.

По-друге, React пропонує ефективну та просту модель компонентів, яка дозволяє розбити інтерфейс на невеликі, перевикористовувані частини. Це сприяє покращенню організації коду, забезпечує легку розширюваність та підтримуваність додатку.

По-третє, React має вбудовану підтримку віртуального DOM (Document Object Model), що дозволяє ефективно оновлювати інтерфейс залежно від змін у даних. Це робить додаток швидким і масштабованим, особливо при великій кількості компонентів та динамічних елементів.

По-четверте, React має багатий екосистему та підтримує велику кількість сторонніх бібліотек та плагінів. Це дозволяє легко інтегрувати додаткові функціональності та розширювати можливості застосунку.

Обрання React для розробки CodeJourney було обґрунтовано його популярністю, простотою використання, ефективністю та широким спектром

можливостей, які він надає. Це дозволяє нам забезпечити якісний та потужний досвід навчання програмування користувачам додатку.

Крім перерахованих переваг, вибір React також зумовлений його здатністю працювати з комплексними додатками із складним станом та взаємодією. За допомогою React, ми можемо легко керувати станом додатку та забезпечити швидке та ефективно оновлення інтерфейсу користувача при зміні даних або взаємодії з користувачем. React також надає можливість розробки компонентів, які можуть бути повторно використані у різних частинах додатку. Це дає нам можливість створювати модульні та масштабовані компоненти, що прискорює розробку та полегшує підтримку [33]. Ще однією перевагою React є наявність розширеного набору інструментів та бібліотек для розробки. Наприклад, ми можемо використовувати бібліотеку React Router для управління маршрутизацією додатку, що дає змогу реалізувати багатосторінкові додатки зі зручною навігацією.

Розробка за допомогою React дозволяє використовувати сучасні практики розробки, такі як компонентна архітектура, односторонній потік даних та використання функціональних компонентів зі станом за допомогою хуків. Це сприяє покращенню якості коду, забезпечує більшу зручність у розробці та зрозумілість кодової бази.

Загалом, вибір React для розробки CodeJourney дозволяє нам створити потужний, гнучкий та ефективний додаток для навчання програмування з використанням сучасних підходів та засобів.

Інші інструменти та бібліотеки. Під час розробки були використані різні інструменти та бібліотеки для поліпшення ефективності та якості розробки. Наприклад, були використані npm (Node Package Manager) для управління залежностями проєкт. Npm (Node Package Manager) було використано з декількох причин. Керування залежностями: npm дозволяє легко керувати залежностями проєкту. Завдяки npm можна швидко встановлювати, оновлювати та видаляти пакети, необхідні для розробки і функціонування додатку. Це допомагає

забезпечити консистентність середовища розробки та зменшити можливі проблеми, пов'язані зі сумісністю версій пакетів[34]. Широкий вибір пакетів: npm має один з найбільших репозиторіїв пакетів у світі програмування. Це означає, що розробники можуть використовувати готові пакети, які надають різноманітні функціональні можливості, такі як робота з базами даних, аутентифікація, валідація форм, графічні ефекти та багато іншого. Використання пакетів з репозиторію npm прискорює процес розробки, оскільки не потрібно писати все з нуля. Спільнота розробників: npm має велику та активну спільноту розробників. Це означає, що є багато ресурсів, документації, підтримки та прикладів використання пакетів. Розробники можуть легко знайти відповіді на свої питання, вирішити проблеми та використовувати кращі практики, які спільнота пропонує. Легка інтеграція з іншими інструментами: npm добре інтегрується з іншими інструментами розробки, такими як збирачі модулів (наприклад, Webpack або Rollup).

Роутінг – React Router. Для забезпечення навігації та переходу між різними сторінками в системі навчання була використана бібліотека React Router. React Router є потужним інструментом для управління маршрутизацією в React-додатках. Вона дозволяє визначати маршрути, зв'язувати їх з відповідними компонентами та забезпечує зручні методи для навігації та передачі параметрів між сторінками.

Інструменти розробника. Під час розробки були використані різні інструменти для поліпшення продуктивності та якості коду. Наприклад, інструменти для відладки (наприклад, Chrome Developer Tools) використовувалися для виявлення та виправлення помилок у коді. Також були використані системи контролю версій (наприклад, Git) для ефективного керування та спільної роботи над проектом. Ці засоби розробки були обрані з урахуванням їх популярності, функціональності та підтримки спільнотою розробників. Вони відповідають вищезазначеним критеріям та допомогли

забезпечити ефективну та зручну розробку гейміфікованої системи навчання з використанням Java Script, React та GraphQL.

### 3.2 Опис архітектури системи

Архітектура системи базується на моделі клієнт-сервер. Додаток розроблений з використанням фреймворку React для фронтенду і Node.js для бекенду.

На стороні клієнта, фронтенд додатку побудований з використанням компонентної архітектури React. Компоненти розбиті на логічні блоки, що спрощує розробку, тестування та підтримку коду. Компоненти взаємодіють між собою та з сервером, обмінюючись даними за допомогою HTTP-запитів.

На стороні сервера, використовується середовище Node.js для реалізації серверної логіки. Бекенд додатку побудований з використанням Express.js - популярного фреймворку для створення веб-додатків на Node.js. Express.js надає зручний і простий у використанні спосіб обробки HTTP-запитів, роботу з маршрутами, обробку даних та багато іншого. Запити до сервера оброблюються за допомогою контролерів, які містять бізнес-логіку додатку.

Додаток використовує базу даних для зберігання інформації про користувачів, їх прогрес, досягнення та інші дані. Для забезпечення доступу до бази даних використовується ORM (Object-Relational Mapping) - технологія, що дозволяє зручно працювати з базою даних за допомогою об'єктно-орієнтованого підходу.

Також в системі використовується механізм аутентифікації та авторизації користувачів. Для цього використовується JWT (JSON Web Tokens), який дозволяє генерувати та перевіряти токени для авторизації користувачів.

Загалом, архітектура системи CodeJourney ретельно спроектована з урахуванням принципів модульності, розширюваності та повторного використання коду. Розділення на фронтенд і бекенд дозволяє розподілити логіку додатку і забезпечити більшу масштабованість. Компонентна архітектура React

сприяє зручному управлінню та реюзабельності компонентів, що покращує швидкість розробки та зберігає чистоту коду.

Додаток, створений за допомогою команди `create-react-app` використовує популярну архітектурну модель для розробки веб-додатків з використанням React. Вона надає зручний стартовий набір, що дозволяє швидко розпочати проект із підтримкою основних функціональних можливостей.

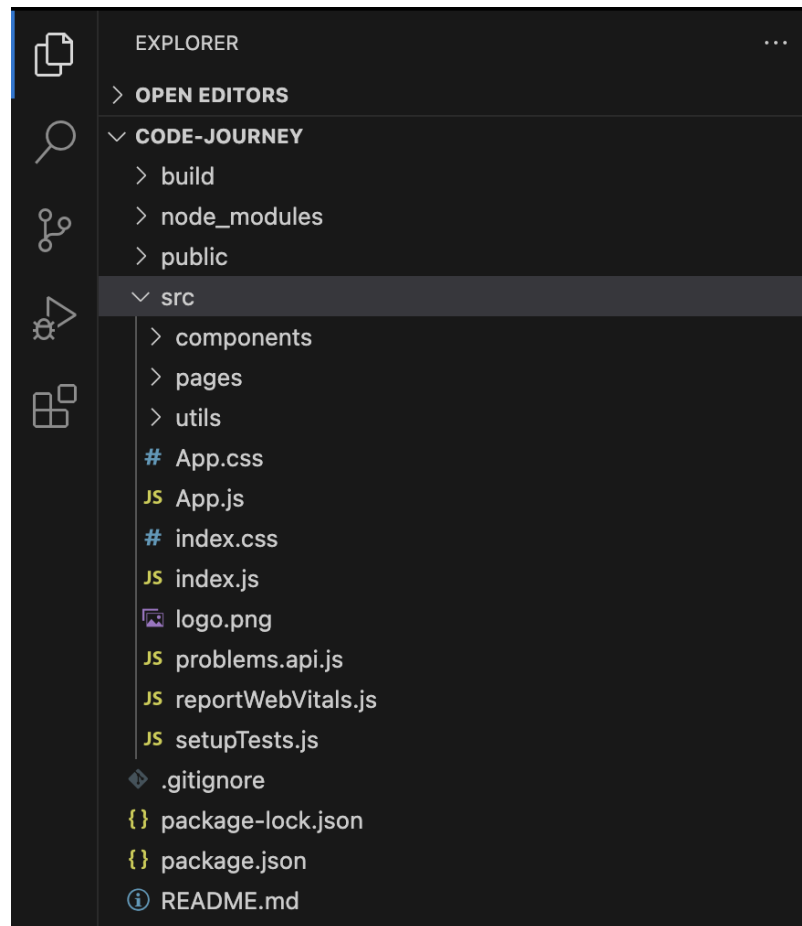


Рисунок 3 – Архітектура додатку [Рисунок виконаний самостійно]

Основна архітектура додатку створеного за допомогою `create-react-app` має наступну структуру:

- папка `public`: В цій папці знаходяться статичні ресурси, такі як HTML-файли, зображення, шрифти тощо. Файли, розміщені в цій папці, будуть доступні для веб-браузера;

- папка `src`: Це основна папка, де розміщений весь вихідний код додатку. Файл `.index.js`: Цей файл є вхідним для додатку і відповідає за рендеринг кореневого компонента в DOM. Файл `App.js`: У цьому файлі зазвичай розміщений головний компонент додатку, який включає інші компоненти та відповідає за логіку та структуру додатку;
- папка `components`: В цій папці можна створювати підпапки та розміщувати компоненти, які будуть використовуватись у додатку;
- папка `styles`: Ця папка містить файли CSS або інші файли стилів, які використовуються для вигляду компонентів;
- папка `utils`: В цій папці можна розміщувати допоміжні функції, утиліти або константи, які використовуються в додатку;
- конфігураційні файли: У кореневій папці додатку також можуть знаходитися конфігураційні файли. Файл `package.json`. Цей файл містить інформацію про проект, залежності та скрипти для розробки та збірки додатку. В файлі `package.json` вказується назва пакету та його версія. При наявності цього файлу, `npm` може автоматично завантажувати та встановлювати необхідні пакети зі зазначеними версіями. Більш того, `npm` надає можливість використовувати скрипти, які можуть бути визначені в файлі `package.json`. Це дозволяє виконувати різні операції, такі як збірка проекту, запуск тестів, літінг коду та багато іншого. Використання скриптів спрощує автоматизацію рутинних завдань та полегшує розробку та підтримку додатку. Файл `package-lock.json` містить точну версію всіх залежностей, встановлених у проекті, для забезпечення однакових версій при повторній установці. Файл `package-lock.json` використовується для забезпечення точної репродукції залежностей та їх версій під час розгортання та виконання додатку. Коли ви встановлюєте пакети, використовуючи `npm` або `yarn`, вони шукають інформацію про версії пакетів у файлі `package-lock.json` та встановлюють ці самі версії. Це гарантує, що всі члени команди, що працюють над проектом,

використовують однакові версії залежностей. Файл `package-lock.json` також забезпечує стабільність та незмінність залежностей. Він фіксує конкретні версії пакетів, які були встановлені на момент створення файлу `package-lock.json`, та забезпечує їх повторне відтворення при переустановці залежностей. Цей файл також містить контрольні суми для кожного пакету, що дозволяє виявити будь-які зміни або пошкодження в пакетах. Це допомагає попередити проблеми, які можуть виникнути внаслідок некоректної залежності або пошкодження файлів пакету. Загалом, використання файлу `package-lock.json` у CodeJourney допомагає забезпечити стабільність та незмінність залежностей під час розгортання та виконання додатку. Файл `.env`. Цей файл містить змінні середовища, такі як ключі API, URL-адреси бази даних тощо;

- файл `public/index.html`. Цей файл використовується як шаблон HTML для веб-сторінки. Він містить основну структуру HTML-документу та посилання на скомпільовані JavaScript та CSS файли;
- іконка `public/favicon.ico`. Це значок, який відображається у вкладці браузера або ярлику додатку.

За потреби можна створювати додаткові папки та розміщувати в них відповідні файли.

Архітектура додатку, створеного за допомогою `create-react-app`, спрощує розробку веб-додатків, оскільки надає початкову структуру та налаштування. Використання React, разом із підтримкою React Router для роутингу та інших додаткових бібліотек та інструментів, дозволяє створювати потужні та гнучкі додатки для навчання програмування на JavaScript та Node.js з використанням GraphQL.

### 3.3 Опис роботи застосунку

Застосунок CodeJourney – це інтерактивний тренажер для вивчення програмування на JavaScript з використанням гейміфікації. Цей додаток

призначений для навчання новачків у програмуванні та поліпшення навичок досвідчених розробників.

Основна мета CodeJourney – зробити процес навчання програмування захопливим, цікавим і взаємодійним. Він використовує методи гейміфікації, такі як бали, досягнення, ранги, мисії та завдання, персонажі та історії, а також соціальну інтеракцію, для створення стимулюючого навчального досвіду.

Користувачі CodeJourney можуть вивчати програмування, розв'язувати практичні завдання, писати код, вирішувати головоломки і отримувати миттєвий зворотний зв'язок. Додаток надає інтерактивні уроки з покроковими поясненнями, які допомагають зрозуміти концепції програмування та вирішувати задачі.

Окрім того, CodeJourney пропонує спільноту користувачів, де вони можуть обговорювати питання, ділитися своїм прогресом та співпрацювати в проектах. Це сприяє соціальній інтеракції і збільшує мотивацію до навчання та вдосконалення навичок програмування.

Застосунок CodeJourney розроблений з використанням технологій React, Node.js та GraphQL. Його інтуїтивний і легкий у використанні інтерфейс дозволяє користувачам швидко навчитися програмувати, отримуючи задоволення від процесу. Кожен елемент додатку – від дизайну до функціональності – ретельно продуманий, щоб забезпечити комфортне та ефективне навчання.

У майбутньому планується розширення функціональності CodeJourney. Перш за все, планується додавання додаткових розділів і уроків, що охоплюють різні аспекти програмування на JavaScript і Node.js. Користувачі зможуть продовжувати вивчати складніші теми, такі як робота з базами даних, розробка веб-додатків, використання різних фреймворків тощо.

Також планується розширити підтримку інших мов програмування. Користувачі зможуть вивчати і практикувати розробку на інших мовах, таких як Python, Java, C++, що відкриє нові можливості для розвитку навичок програмування в різних областях.

CodeJourney прагне стати не просто інтерактивним тренажером, але й активною спільнотою програмістів. Планується розширити можливості співпраці та проектної роботи між користувачами. Це дозволить створювати спільні проекти, вирішувати складні задачі та навчатися один від одного.

Загалом, CodeJourney – це інноваційний додаток, що поєднує гейміфікацію та навчання програмування, створюючи захопливе і ефективне середовище для розвитку навичок у сфері програмування. Розширення функціональності і підтримка інших мов програмування роблять його універсальним і корисним для широкого спектру користувачів – від початківців до досвідчених розробників.

Крім того, CodeJourney надає користувачам можливість взаємодіяти з реальними проектами та завданнями. Це дає можливість отримати практичний досвід і розвинути навички роботи з реальними сценаріями програмування. Користувачі можуть брати участь у віртуальних командних проектах, спільно вирішувати завдання та робити внесок у процес розробки.

Основна функціональність CodeJourney включає інтерактивні уроки, практичні завдання, кодові виклики та проекти, оцінювання та статистику продуктивності. Кожен урок представлений в зручній формі, з можливістю перегляду теоретичного матеріалу, демонстрації прикладів та виконання практичних завдань.

Заснована на підході гейміфікації, CodeJourney пропонує систему балів, досягнень та рейтингів, що стимулюють користувачів досягати поставлених цілей та прогресувати в навчанні. Користувачі можуть збирати досягнення за виконані завдання, отримувати бали за правильно розв'язані виклики та підніматися в рейтингу спільноти.

Однією з особливостей CodeJourney є візуалізація процесу навчання. Користувачі можуть спостерігати свій прогрес, переглядати статистику своїх досягнень, аналізувати свої сильні та слабкі сторони та визначати області, в яких потрібно більше практики.

В цілому, CodeJourney є потужним інструментом для навчання програмування, який поєднує гейміфікацію, інтерактивність та реальний досвід. Він спрощує процес навчання, роблячи його цікавим, захоплюючим та ефективним. Завдяки широкому спектру уроків, практичних завдань та проєктів, користувачі можуть вивчати програмування на різних рівнях складності і поглиблювати свої знання по мірі просування.

Описані вище функціональні можливості CodeJourney дозволяють студентам, початківцям і досвідченим програмістам розвивати свої навички, експериментувати з новими технологіями та покращувати свої професійні навички. Додаток також забезпечує можливість спілкування із спільнотою, обміну знаннями та досвідом з іншими користувачами.

Застосунок є інтуїтивно зрозумілим та легким у використанні. Він надає можливість вивчати програмування в зручній для кожного темпі і режимі. Користувачі мають можливість переглядати свій прогрес, виконувати завдання на різних рівнях складності, взаємодіяти з реальними проєктами та отримувати зворотний зв'язок та оцінку своєї роботи.

Загалом, CodeJourney є потужним інструментом для навчання програмування, який поєднує в собі навчальний матеріал, інтерактивні завдання та гейміфікацію. Він спрощує процес навчання та забезпечує мотивацію для досягнення успіху у програмуванні.

## ВИСНОВКИ

У результаті роботи було проведено дослідження методів гейміфікації розробки систем навчання для розвитку програмістів. Основною метою роботи було вивчення ефективності та впливу гейміфікаційних методів на процес навчання та підвищення якості роботи програмістів у технічних компаніях. З метою досягнення поставленої мети, були проаналізовані різні підходи та методи гейміфікації, вибрані технології для розробки системи навчання та реалізовано додаток, що інтегрує гейміфікаційні елементи.

Дослідження показало, що використання методів гейміфікації в розробці систем навчання має значний потенціал для поліпшення процесу навчання та досягнення кращих результатів. Було виявлено, що гейміфікаційні елементи, такі як лідерборди, досягнення, бейджі та винагороди, сприяють залученню та мотивації учнів. Вони стимулюють активну участь, конкуренцію, співпрацю та самостійне навчання.

Крім того, було встановлено, що використання гейміфікації в розробці систем навчання сприяє підвищенню якості роботи програмістів у технічних компаніях. Гейміфікаційні елементи спонукають до постійного самовдосконалення, розвитку нових навичок та збільшення продуктивності. В результаті, компанії можуть отримати кваліфікованіших та більш мотивованих співробітників.

Отже, можна зробити висновок, що гейміфікація є потужним інструментом для покращення процесу навчання та розробки систем навчання. Вона сприяє залученню учнів, створює мотивацію до активної участі, підвищує зацікавленість та стимулює досягнення результатів. Гейміфікаційні елементи, такі як бейджі, рівні, лідерборди та винагороди, сприяють змагальності, співпраці та взаємодії між учнями. Це сприяє покращенню взаємодії та соціального аспекту навчання.

Було проведено аналіз сучасних підходів до гейміфікації навчання та визначено, що вони можуть бути ефективними при розробці систем навчання для програмістів. Також було розглянуто проблеми, що можуть виникнути при їх

використанні. А саме було виявлено певні недоліки та проблеми, пов'язані з використанням гейміфікації, такі як можливість занадто сильного спрощення матеріалу або зосередження уваги на грі, замість основної мети навчання. Також, варто враховувати що не всі гейміфікаційні елементи можуть бути підходящими для кожного типу навчання або аудиторії. Деякі учні можуть відчувати негативний тиск або конкуренцію, що може вплинути на їхню мотивацію та самооцінку. Важливо враховувати баланс між гейміфікаційними елементами та академічним змістом навчання, щоб забезпечити належну освіту.

Дослідження методів гейміфікації в розробці систем навчання виявило їхню значну ефективність у покращенні процесу навчання та підвищенні мотивації учнів. На основі проведеного дослідження було розроблено прототип системи навчання з використанням гейміфікації.

Також за результатами роботи були написані тези доповіді для участі у міжнародній науковій конференції «Актуальні питання розвитку галузей науки» [51].

У подальшому планується розширення функціоналу розробленої системи навчання та вдосконалення методів гейміфікації з метою поліпшення результатів навчання та забезпечення максимальної ефективності використання гейміфікації в навчальних процесах.

**ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ**

1. Burke, B. (2014). Gamify: How gamification motivates people to do extraordinary things. Bibliomotion.
2. Deterding, S., Dixon, D., Khaled, R., & Nacke, L. (2011). From game design elements to gamefulness: Defining "gamification". In Proceedings of the 15th International Academic MindTrek Conference: Envisioning Future Media Environments (pp. 9-15). ACM.
3. Hamari, J., & Sjöklint, M. (2015). Does gamification work?—A literature review of empirical studies on gamification. In Proceedings of the 2015 48th Hawaii International Conference on System Sciences (pp. 3025-3034). IEEE.
4. Landers, R. N., & Landers, A. K. (2014). An empirical test of the theory of gamified learning: The effect of leaderboards on time-on-task and academic performance. *Simulation & Gaming*, 45(6), 769-785.
5. Lee, J. J., & Hammer, J. (2011). Gamification in education: What, how, why bother?. *Academic Exchange Quarterly*, 15(2), 1-5.
6. Mekler, E. D., Brühlmann, F., Tuch, A. N., & Opwis, K. (2017). Towards understanding the effects of individual gamification elements on intrinsic motivation and performance. *Computers in Human Behavior*, 71, 525-534.
7. Seaborn, K., & Fels, D. I. (2015). Gamification in theory and action: A survey. *International Journal of Human-Computer Studies*, 74, 14-31.
8. Werbach, K., & Hunter, D. (2012). Gamification: How game elements motivate behavior. In *The gamification of learning and instruction: Game-based methods and strategies for training and education* (pp. 3-31). Pfeiffer.
9. Kapp, K. M. (2012). *The gamification of learning and instruction fieldbook: Ideas into practice*. John Wiley & Sons.
10. Nacke, L. E., Bateman, C., & Mandryk, R. L. (2014). BrainHex: A neurobiological gamer typology survey. *Entertainment Computing*, 5(1), 55-62.
11. Pivec, M., & Dziabenko, O. (2015). Game-based learning and gamification in higher education: Impact study on lecturers. *Journal of Interactive Learning*

Research, 26(1), 27-48.

12. Sailer, M., Hense, J. U., Mayr, S. K., & Mandl, H. (2017). How gamification motivates: An experimental study of the effects of specific game design elements on psychological need satisfaction. *Computers in Human Behavior*, 69, 371-380.

13. Sigala, M. (2017). Gamification in tourism: A review of the literature. *Tourism Management*, 60, 244-254.

14. Vlachopoulos, D., & Makri, A. (2017). The effect of games and simulations on higher education: A systematic literature review. *International Journal of Educational Technology in Higher Education*, 14(1), 1-22.

15. Zichermann, G., & Linder, J. (2010). *Game-based marketing: Inspire customer loyalty through rewards, challenges, and contests*. John Wiley & Sons.

16. Werbach, K., & Hunter, D. (2012). *For the win: How game thinking can revolutionize your business*. Wharton Digital Press.

17. Reeves, B., & Read, J. L. (2009). *Total engagement: Using games and virtual worlds to change the way people work and businesses compete*. Harvard Business Press.

18. Barata, G., Gama, S., Jorge, J., & Gonçalves, D. (2013). Improving participation and learning with gamification. In *Proceedings of the First International Conference on Gameful Design, Research, and Applications* (pp. 10-17).

19. Deterding, S., Dixon, D., Khaled, R., & Nacke, L. (2011). From game design elements to gamefulness: Defining “gamification”. In *Proceedings of the 15th International Academic MindTrek Conference: Envisioning Future Media Environments* (pp. 9-15). ACM.

20. Hamari, J., Koivisto, J., & Sarsa, H. (2014). Does gamification work?—A literature review of empirical studies on gamification. In *Proceedings of the 47th Hawaii International Conference on System Sciences* (pp. 3025-3034). IEEE.

21. Nicholson, S. (2012). A user-centered theoretical framework for meaningful gamification. In *Proceedings of Games+ Learning+ Society* (Vol. 8).

22. Reeves, B., Malone, T. W., & O'Driscoll, T. (2008). Leadership's online labs. *Harvard Business Review*, 86(12), 62-70.
23. Landers, R. N., Bauer, K. N., Callan, R. C., & Armstrong, M. B. (2017). Psychological theory and gamification: The study of motivation and learning. In *Gamification in Education and Business* (pp. 25-49). Springer.
24. Morschheuser, B., Werder, K. P., & Hamari, J. (2019). How to design gamification? A method for engineering gamified software. *Information and Software Technology*, 109, 13-31.
25. Kapp, K. M. (2012). *The gamification of learning and instruction: Game-based methods and strategies for training and education*. John Wiley & Sons.
26. Johnson, L., Adams Becker, S., Estrada, V., Freeman, A., & Hall, C. (2015). *NMC horizon report: 2015 higher education edition*. The New Media Consortium.
27. Kiili, K. (2005). Digital game-based learning: Towards an experiential gaming model. *The Internet and Higher Education*, 8(1), 13-24.
28. Csikszentmihalyi, M. (1991). *Flow: The psychology of optimal experience*. HarperCollins.
29. Zichermann, G., & Cunningham, C. (2011). *Gamification by design: Implementing game mechanics in web and mobile apps*. O'Reilly Media, Inc.
30. Hunicke, R., LeBlanc, M., & Zubek, R. (2004). MDA: A formal approach to game design and game research. In *Proceedings of the AAAI Workshop on Challenges in Game AI* (pp. 1-5).
31. Hoffman, J. A., & Duarte, P. (2014). Learning through game design: An investigation of age and gender differences in game design activities. *Computers & Education*, 73, 30-45.
32. Kohn, A. (1993). *Punished by rewards: The trouble with gold stars, incentive plans, A's, praise, and other bribes*. Houghton Mifflin Harcourt.
33. Hamari, J., Koivisto, J., & Sarsa, H. (2014). Does gamification work?-A literature review of empirical studies on gamification. In *Proceedings of the 47th*

Hawaii International Conference on System Sciences (pp. 3025-3034).

34. Landers, R. N., Bauer, K. N., & Callan, R. C. (2017). Gamification of task performance with leaderboards: A goal setting experiment. *Computers in Human Behavior*, 71, 508-515.

35. Dicheva, D., Dichev, C., Agre, G., & Angelova, G. (2015). Gamification in education: A systematic mapping study. *Journal of Educational Technology & Society*, 18(3), 75-88.

36. Lister, M., West, J. H., Cannon, B., Sax, T., Brodegard, D., & Just, E. (2014). Gamification of learning in statistics and analytics. *International Journal of Teaching and Learning in Higher Education*, 26(3), 349-359.

37. Nicholson, S. (2015). A user-centered theoretical framework for meaningful gamification. In *Proceedings of the 2015 Annual Symposium on Computer-Human Interaction in Play* (pp. 201-214).

38. Lee, J. J., & Hammer, J. (2011). Gamification in education: What, how, why bother? *Academic Exchange Quarterly*, 15(2), 1-5.

39. Huang, Y. M., & Backman, S. J. (2011). Gamification and user engagement. In *Proceedings of the 2011 International Conference on Advances in Social Networks Analysis and Mining* (pp. 273-276).

40. Zichermann, G., & Linder, J. (2013). *The gamification revolution: How leaders leverage game mechanics to crush the competition*. McGraw Hill Professional.

41. Werbach, K., & Hunter, D. (2012). *For the win: How game thinking can revolutionize your business*. Wharton Digital Press.

42. Plass, J. L., Homer, B. D., & Kinzer, C. K. (2015). Foundations of game-based learning. *Educational Psychologist*, 50(4), 258-283.

43. Barata, G., Gama, S., Jorge, J., & Gonçalves, D. (2013). Improving participation and learning with gamification. In *Proceedings of the First International Conference on Gameful Design, Research, and Applications* (pp. 10-17).

44. Ren, Y., & Lu, Y. (2019). Gamification of education: A review of literature. *Eurasia Journal of Mathematics, Science and Technology Education*, 15(6), em1716.

45. Hainey, T., Connolly, T. M., Stansfield, M., & Boyle, E. A. (2011). Evaluation of a game to teach requirements collection and analysis in software engineering at tertiary education level. *Computers & Education*, 56(4), 1219-1230.
46. Hung, H. T. (2017). Gamification for educational systems development: An exploratory study. *International Journal of Human-Computer Interaction*, 33(11), 877-889.
47. Scheepers, R. A., Schneider, J., & van der Meijden, H. (2019). Can game-based learning enhance STEM education? A systematic review of randomized controlled trials. *Journal of Educational Psychology*, 111(3), 385-404.
48. Fornasiero, R. A., Pontarolo, R., & Todesco, J. L. (2017). The impact of gamification on software engineering: A systematic literature review. *Journal of Systems and Software*, 132, 131-146.
49. Anderson, E. L., Steenburgh, T. J., & Aime, F. (2016). Gamification of task-based assignments to motivate student effort and learning. *Journal of Education for Business*, 91(7), 365-371.
50. Zare, M., & Rahmani, R. (2017). Gamification in education: A systematic review of literature. *International Journal of Information and Learning Technology*, 34(4), 423-436.
51. Суровікіна К. С. Використання методів гейміфікації в розробці систем навчання. Тези доповідей V Міжнародної наукової конференції «Актуальні питання розвитку галузей науки» 12.05.2023, Чернігів, Україна. Секція XII Комп'ютерна та програмна інженерія. С. 114. Доступно за адресою: <https://archive.mcnd.org.ua/index.php/conference-proceeding/issue/view/12.05.2023>