

Міністерство освіти і науки України
Національний технічний університет
«Дніпровська політехніка»

Інститут електроенергетики
(інститут)

Факультет інформаційних технологій
(факультет)

Кафедра Програмного забезпечення комп'ютерних систем
(повна назва)

ПОЯСНЮВАЛЬНА ЗАПИСКА
кваліфікаційної роботи ступеня
бакалавра

(назва освітньо-кваліфікаційного рівня)

студента Стрюк Олег Андрійович
(ПІБ)

академічної групи 122-19-1
(шифр)

спеціальності 122 Комп'ютерні науки
(код і назва спеціальності)

освітньої програми Комп'ютерні науки
(назва освітньої програми)

на тему: Розробка комп'ютерної відеоігри для мобільного
пристрою на платформі Android

Керівники	Прізвище, ініціали	Оцінка за шкалою		Підпис
		рейтинговою	інституційною	
кваліфікаційної роботи	доц. Реута. О.В.			
розділів:				
спеціальний	доц. Реута О.В.			
економічний	проф. Вагонова О.Г.			
Рецензент	проф. Гнатушенко В.В.			
Нормоконтролер	доц. Гуліна І.Г.			

Дніпро
2023

Міністерство освіти і науки України
НТУ «Дніпровська політехніка»

ЗАТВЕРДЖЕНО:

завідувач кафедри
програмного забезпечення комп'ютерних систем

(повна назва)

М.О. Алексєєв

(підпис)

(прізвище, ініціали)

« » 2023 року

ЗАВДАННЯ

на кваліфікаційну роботу
бакалавра

(назва освітньо-кваліфікаційного рівня)

студента 122-19-1
(група)

Стрюк Олег Андрійович
(прізвище та ініціали)

тема кваліфікаційної роботи
пристрою на платформі Android

Розробка ком'юторні відео гри для мобільног

затверджена наказом ректора НТУ «ДП» від 16.05.2023р. № 350-с

Розділ	Зміст виконання	Термін виконання
Спеціальний	На основі матеріалів проєктно-технологічної практики та інших науково-технічних джерел провести аналіз стану рішення проблеми та постановку задачі. Обґрунтувати вибір та здійснити реалізацію методів вирішення проблеми	13.05.2023 р.
Економічний	Провести розрахунок трудомісткості розробки програмного забезпечення, витрат на створення ПЗ й тривалості його розробки	27.05.2023 р.

Завдання видав

(підпис)

доц. Реута О.В.

(посада, прізвище, ініціали)

Завдання прийняв до виконання

(підпис)

Стрюк О.А.

(прізвище, ініціали)

Дата видачі завдання: 14.01.2023 р.

Термін подання кваліфікаційної роботи до ЕК: 12.06.2023 р.

РЕФЕРАТ

Пояснювальна записка: 125 с., 46 рис., 3 дод., 22 джерел.

Об'єкт розробки: комп'ютерна гра-вікторина.

Мета кваліфікаційної роботи: розробка комп'ютерної відеоігри для мобільного пристроя на платформі Android.

У вступі розглядається аналіз, та сучасний стан проблеми, конкретизується мета кваліфікаційної роботи та галузь її застосування, наведено обґрунтування актуальності теми та уточняється постановка завдання.

У першому розділі буде проаналізовано актуальність гри у нинішній час, загальні відмінності галузі та застосування, підстави для розробки програми, вимоги та технічні параметри як програми так і допоміжних засобів.

У другому розділі буде описане функціональні призначена, обрана платформа розробки, опис використаних технологій та мов програмування. Користування середовищем розробки та редактором зображення. Структування системи та алгоритм функціонування. Використання технічних та програмних засобів, виклик та завантаження програми, опис інтерфейсу.

В економічному розділі визначено трудомісткість розробленої інформаційної системи, проведений підрахунок вартості роботи по створенню програми та розраховано час на її створення.

Практичне значення гри, має мету викликати у користувача інтерес до науки, історії держави та суспільства, також дає можливість перевірити свої знання у тих користувачів ,які працюють у сферах ,чи проявляють інтерес до тем ,які містяться у рівнях гри. Користувач буде перебувати не тільки в сфері науки але й у сфері культури, яка оточує кожного.

Список ключових слів: СЕРЕДОВИЩЕ, НАУКИ, СУСПІЛЬСТВО, ANDROID, ФОТОРЕДАКТОР, ІНТЕРФЕЙС, ГРА.

ABSTRACT

Explanatory note: 125 p., 46 fig., 3 app., 22 refs.

Object of development: quiz game.

The purpose of the qualification work: development of a computer video game for a mobile device on the Android platform.

The introduction examines the analysis and the current state of the problem, specifies the purpose of the qualification work and the field of its application, provides a rationale for the relevance of the topic, and clarifies the statement of the task.

The first section will analyze the relevance of the game at the present time, the general differences of the industry and application, the basis for the development of the program, the requirements and technical parameters of both the program and the aids.

The second section will describe the functionality of the selected development platform, the description of the used technologies and programming languages. Using the development environment and image editor. System structure and functioning algorithm. Use of technical and software tools, calling and downloading the program, description of the interface.

In the economic section, the labor intensity of the developed information system is determined, the cost of work on creating the program is calculated, and the time for its creation is calculated.

The practical meaning of the game aims to arouse the user's interest in science, the history of the state and society, and also provides an opportunity to test their knowledge with those users who work in the fields or show interest in the topics contained in the levels of the game. The user will be not only in the field of science, but also in the field of culture that surrounds everyone.

List of keywords: ENVIRONMENT, SCIENCES, SOCIETY, ANDROID, PHOTO EDITOR, INTERFACE, GAME.

ЗМІСТ

РЕФЕРАТ	3
ABSTRACT	4
ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ.....	7
ВСТУП	8
РОЗДІЛ 1. АНАЛІЗ ПРЕДМЕТНОЇ ГАЛУЗІ ТА ПОСТАНОВКА ЗАДАЧІ	10
1.1. Загальні відомості з предметної галузі	10
1.2. Призначення розробки та галузь застосування	15
1.3. Підстави для розробки	15
1.4. Постановка завдання	16
1.5. Вимоги до програми або програмного виробу	18
1.5.1. Вимоги до функціональних характеристик.....	18
1.5.2. Вимоги до інформаційної безпеки	19
1.5.3. Вимоги до складу та параметрів технічних засобів.....	19
РОЗДІЛ 2. ПРОЕКТУВАННЯ ТА РОЗРОБКА ІНФОРМАЦІЙНОЇ СИСТЕМИ.....	21
2.1. Функціональне призначення системи	21
2.2. Опис застосованих математичних методів.....	22
2.3. Опис використаних технологій та мов програмування.....	25
2.3.1. Опис мови Java.....	25
2.3.2. Опис фодоредактора Photoshop	26
2.3.3. Опис програми Android Studio.....	28
2.4. Опис структури системи та алгоритмів її функціонування	29
2.5. Обґрунтування та організація вхідних та вихідних даних програми	43
2.6. Опис роботи розробленої системи	43
2.6.1. Використані технічні засоби.....	43
2.6.2. Використані програмні засоби	44
2.6.3. Виклик та завантаження програми.....	44
2.6.4. Опис інтерфейсу користувача	45
РОЗДІЛ 3. ЕКОНОМІЧНИЙ РОЗДІЛ	52
3.1. Розрахунок трудомісткості та вартості розробки програмного продукту.....	52
3.2. Розрахунок витрат на створення програми	56

ВИСНОВКИ	59
СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ	60
ДОДАТОК А. КОД ПРОГРАМИ	63
ДОДАТОК Б. ВІДГУК КЕРІВНИКА ЕКОНОМІЧНОГО РОЗДІЛУ	124
ДОДАТОК В. ПЕРЕЛІК ФАЙЛІВ НА ДИСКУ	125

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ

FIFA – Fédération Internationale de Football Association;

MMORPG - Massively multiplayer online role-playing game;

AAA - triple-A (an informal term for a class of high-budget computer games.);

UX - user experience;

UI - user interface;

WORA - Write Once and Run Anywhere;

JVM - Java Virtual Machine;

API - Application Programming Interface;

iOS – iPhone Operating System;

IDE - Integrated Development Environment;

ОС – Операційна система;

APK - Android Package Kit;

XML - eXtensible Markup Language;

EPM - Enterprise Performance Management.

ВСТУП

Розробка мобільних додатків на сьогоднішній день є дуже популярною темою. У наш час є багато компаній, які випускають свої продукти, будь це гра, додаток доставки їжі чи аудіо-помічник, GPS та інші корисні додатки. Кожен з них має власну мету, але з часом усі вони об'єднуються навколо однієї – Заробіток.

Розробка ігор почалася ще в 1940 році, а в 1952 році Олександр Дуглас розробив першу гру – ОХО(Хрестики – Нолики). Потім у 1962 році Стивом Расселом була розроблена одна з найпопулярніших ігор для двох гравців – Spacewar. Але у 1970 році почався дуже великий зріст та прогрес у сфері ігор, почали з'являтися ігри нових жанрів та у більшому доступі. У 1984 року, коли ігрова індустрія розвивалася великими кроками завдяки новим комп'ютерам, які мали більшу потужність та появи інтернету який допомагав поширювати серед суспільства ще більшу популярність до ігрової індустрії. Ігри почали розвивати не тільки для комп'ютерних систем. У світі консолей спочатку на ринок вийшла PlayStation від Sony, а 2002 року - від Microsoft з Xbox. Тим часом Nintendo випустила Wii, яка тепер також підтримує головоломки, судоку, музику, теніс чи бокс.[1]

Сфера онлайн-ігор та консолей постійно вдосконалюється з розвитком цифрових технологій - так створюється безліч оновленого та різноманітного ігрового контенту. Зараз існують тисячі онлайн-ігор із абсолютно різними історіями, персонажами та цілями. На додаток до графічного дизайну, музика також є важливою частиною ігор: вона створює напругу, залишається у вашій голові та сприяє створенню атмосфери гри – ви граєте кількома почуттями і таким чином легше поринаєте у цифровий світ. Також не потрібно забувати про локалізацію ігор, це дуже потрібна частина для сприйняття гри користувачем. Тому що, якщо ви не розумієте англійської мови, то не зможете зануритись у атмосферу гри. [2]

Але треба не забувати, те що ігри можуть нести в собі характер не тільки позитивний. У нашій сучасності багато компаній розробляють та випускають ігри з прямою цілю наживи на гравцях. Додавання спеціальних комплектів ,купивши які ти будеш мати перевагу над суперником, комплекти в яких ми зможете піймати одну з речей маючих рідкість, від «звичайного» до «легендарного», багато країн у світі стежить за такими іграми та забороняє їх. Наглядний приклад це – футбольний симулятор FIFA ,який заборонили у Нідерландах, тому що «Якщо продукт виглядає і відчувається як азартна гра, він повинен регулюватися як азартна гра».[3] Також є компанії розробників ,які роблять гру безкоштовною, але увесь сюжет буде доступний тільки після покупки доповнень ,приклад – Destiny 2. Тому треба обережно обирати ігри та розуміти , чи готови ви витратити час на гру.

Тому я вирішив розробити свою гру, щоб зрозуміти розробника з боку гравця, але ця гра має бути локалізована на українській мові, тому що увесь час розробники не бачили наш ринок і вважали, що ігор локалізованих російською мовою достатньо. Це все було до 2022 року. Тепер потроху розробники починають добавляти українську локалізацію та нашу валюту у свої сервіси, прикладі цьому гра - Alan Wake яка буде мати українську локалізацію, та американська компанія Xbox сервіси якою згодом з'являться в Україні.[4]

У грі вікторині буде:

- Українська локалізація(усі тексти будуть українською);
- Рівні будуть містити в собі питання завдяки яким буде можливість дізнатись про українських діячів;
- Кожен рівень матиме пояснення виконання та підказки;
- У кінці рівня на користувача буде чикати цікавий факт з тему рівня якого він проходив;
- У гру зможуть грати діти від п'яти років та гравці прихильного віку.

РОЗДІЛ 1

АНАЛІЗ ПРЕДМЕТНОЇ ГАЛУЗІ ТА ЇЇ СТВОРЕННЯ

1.1 Загальні відомості з предметної галузі

Ігри не лише показують себе у сфері розваг чи заробітку, ще ігри можуть також допомагати у навчанні. Наприклад, Minecraft Education Edition [5]. Розроблений спеціально для використання у школі, проект застосовується у навчанні різних предметів. За допомогою гри учні вивчають, наприклад, хімію, історію, літературу та географію. На навчанні учні можуть спостерігати різні природні процеси, брати участь у історичних подіях, проводити досліди. У грі є також функція програмування, що значно розширює можливості гравця. І нехай все це відбувається у кубічному світі, масштаб застосування платформи вражає.

На сайті проекту можна завантажити готові світи та знайти план уроку. Звичайно, для застосування гри в навчанні потрібний обладнаний клас та тренінг викладачів. Але таке навчання принесе новий досвід як учням, і вчителям. Адже це не лише інформатика, а й багато інших шкільних предметів. Рішення, по суті, геніальне у своїй простоті – взяти те, що люблять діти та подавати через це знання. У процесі гри ми можемо отримати не лише приємне, а й корисне. Це може бути розвиток якогось навички чи розширення кругозору. Хтось вивчив англійську мову завдяки іграм, хтось освоїв навички правильного та швидкого друку на комп'ютері.

Також ми повинні не забувати, що кожна гра, має свій жанр. Ігрова сфера перенасичена такими жанрами[6]:

Покрокові стратегії - ігри, в яких гравці роблять свої дії по черзі. Поділ ігрового процесу на ходи відриває його від реального життя та позбавляє гру динамізму, внаслідок чого ці ігри не такі популярні, як стратегії в реальному часі.

Варгейм - не жанр, а характеристика конкретної гри, яка говорить, що вам не доведеться рахувати в ній гроші, більшу частину ігрового часу ви воюватимете всіма доступними засобами. Незважаючи на повну відсутність економіки, гравці без проблем підвищують якість та кількість своїх загонів,

вміло розпоряджаючись двома найважливішими ігровими ресурсами – територією та часом.

Шутер - жанр комп'ютерних ігор, в яких успіх гравця великою мірою залежить від швидкості реакції і здатності швидко приймати тактичні рішення. Дія таких ігор розвивається дуже динамічно і вимагає напруження уваги і швидкої реакції на події, що відбуваються в грі. При цьому як основний засіб прогресу в грі, як правило, використовується будь-яка зброя.

MMORPG — розрахована на багато користувачів рольова онлайн-гра, (зазвичай у жанрі фентазі), в якій гравці можуть взаємодіяти один з одним в ігровому світі. Вони можуть допомагати один одному у виконанні завдань, обмінюватися речами, торгуватися, битися у дуелях. Гравцю пропонується вигаданий герой, яким можна керувати.

Квест - пригодницька гра, один з основних жанрів ігор, що вимагають від гравця вирішення розумових завдань для просування сюжету. Сюжет може бути зумовленим або давати безліч результатів, вибір яких залежить від дій гравця.

Симулятори — це програмне забезпечення або пристрої, що імітують керування будь-яким пристроєм. Можуть застосовуватися як для навчання, так і для розваг.

Файтінг - жанр відеоігор, що імітують рукопашний бій малої кількості персонажів у межах обмеженого простору, званого ареною.

Економічні симулятори - стратегії, в яких військова сторона відсутня в принципі, тому від гравця потрібно лише налагоджувати економіку (і зробити це набагато складніше, ніж у неекономічних стратегіях). Іноді в них присутні суперники (конкуренти), але воєнних дій із ними все одно не відбувається. Економічні симулятори дуже специфічний жанр, це ігри «на любителя».

Інді-гра - комп'ютерна гра, створена окремим розробником або невеликим колективом без фінансової підтримки видавця комп'ютерних ігор. Масштаб явищ, пов'язаних з інді-іграми, відчутно зростає з другої половини 2000-х років, в основному через розвиток нових способів цифрової дистрибуції та засобів розробки. Більше того, що менше колектив, то яскравіше виражається

індивідуальність конкретного розробника. Невеликі колективи, широкі можливості та відсутність меж для творчості створили умови, в яких інді-ігри можуть бути інноваційними, креативними, з великим художнім виразом. Обмежені можливостями створення технологічної графіки, розробники змушені робити ставку на інноваційний геймплей. Втім, серед інді-ігор є як інноваційні ігри, так і ігри класичних жанрів. Таким чином, приналежність до «інді» не передбачає, що гра має нести інновації.[7]

Статистика популярності жанрів відеоігор за 2022 р. за даними порталу Statista.com показана на рис. 1.1.

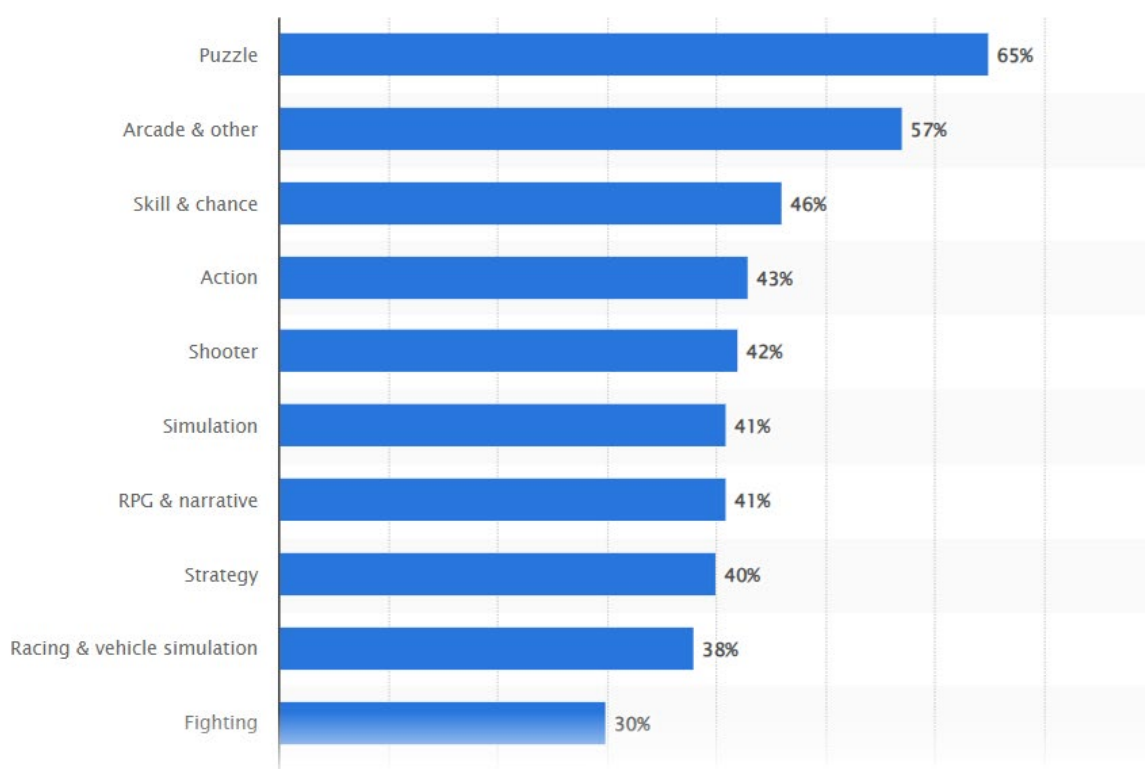


Рис. 1.1. Статистика популярності жанрів відеоігор за 2022р. від Statista.com

Сама розробка ігор містить в собі багато важливих речей. Людині для розробки великого та сучасного продукту треба займатися не лише написанням

коду. Тому розробкою великих проектів займаються підготовленні та маючи опит компанії.

У сфері розробки ігор практично немає поділу на «фронт» та «бек», усі розробники ігор — це дуже різнобічно розвинені програмісти. За специфікою галузі потрібно все робити швидко, і завдання сформульовані не завжди зрозуміло, а то й не завжди сформульовані — але це має виглядати красиво, грати захоплююче та приносити бізнесу багато грошей.

Крім уміння добре програмувати та розбиратися в архітектурах, потрібно мати комплексні знання у різних галузях.

- 3D-моделювання;
- растрових та векторних графічних редакторів;
- анімація;
- інших суміжних напрямів, які пов'язані безпосередньо з професією розробника.

Потрібно розбиратися в графіці та ефектах, розуміти, що саме потребує ресурси і як оптимізувати гру, щоб вона могла запускатися на цільових пристроях з високим показником FPS, при цьому ще й мати графіку, яка буде на рівні цього проекту. Потрібно часто вирішувати завдання геймдизайну та ігрової логіки, які з'явилися у процесі розробки.

Тому розробки хорошої гри потребує великих ресурсів самий головний з яких — це людський.

Гра — це не лише знання мови програмування, дизайну чи анімації, в перш за все потрібна розуміти я її побудувати. Кожен проект будується, як і тіло людини, кожна зі складових якого життєво необхідна, то: архітектура - це скелет і нервова система гри, геймплей - його м'язи, оточення - його канали комунікації, ігровий движок - мозок, без розуміння цього, створити гру буде важко.

Заробіток мобільних ігор за період 2017 – 2022 рр. показаний на рис. 1.2.

Багато компаній використовують різні «движки» для розробки своїх проектів, але є засоби котрі не можна отримати вільно, тому що це може бути розробка компанії, яку користуються тільки вони.[8] Тоді вибір звужується до

декількох привабливих ігрових «движків»: Unity, Unreal Engine, Game Maker, Godot, Cry Engine. Також є похідний від нього Amazon Lumberyard та парочка менш відомих. Але самі відомі та більш прості: це Unity і Unreal Engine. Обидва дуже гарні, але кожен по-своєму. Обидва движки мають велике і розвинене ком'юніті, тому у їх вивченні у вас не буде великих проблем. У цьому їх різниця.



Рис. 1.2. Заробіток мобільних ігор за період 2017 – 2022рр.

Unity - неймовірно популярний ігровий движок, дуже простий у розумінні, зручний, приємний у використанні. Найкраще на ньому вдається робити мобільні ігри. Більше половини ігор, які ви знайдете у себе в мобільному телефоні (якщо не всі), швидше за все, розроблено саме на ньому. Комп'ютерні ігри теж роблять: Subnautica, Wasteland 2, Rust, Hearthstone, Pillars of Eternity, Humankind та інші.

Unreal Engine - авангард технологій, відкритий ігровий движок, що володіє однією з найкращих у наш час систем рендерингу, відмінним фізичним движком і безліччю різних крутих штук, таких як realtime-мультиплеер, який працює "з коробки". Добре оптимізований, чудово підходить для AAA-графіки, особливо

для шутерів. На ньому було зроблено багато легендарних ігор, таких як Borderlands 3, Bioshock 2, Fortnite, Mass Effect , PUBG, Mortal Kombat.

У весь час розробки відеоігор викликала інтерес не тільки до мов програмування ,дизайну ,вивчення мови розробника, але ще і до заробітку [9]. На даний час розробка ігор приносить дуже великі гроші, але це тільки при тому ,що гра повністю окупила свою розробку та продовжує працювати продовж декілька років приносячи прибуток компанії. Якщо брати приклад мобільних ігор, то це: Clash of Clans, Shadow Fight, Mario, Mortal Combat.

1.2 Призначення розробки та галузь застосування

Метою розробки є – гра вікторина, яка має українську локалізацію ,та містить в собі цікаві факти з науки, історії та українського суспільства, будь то спорт, або українські знаменитості. Основую є визвати у користувача інтерес до української культури.

Чому саме вікторина ,а не інші види ігор:

- Несе в собі розважальний характер;
- Містить інтересні питання, складність яких стає важчою з кожним пройденим рівнем;
- Має рівні з різними темами у різних сферах;
- Розвиває розумові здібності.

1.3 Підстави для розробки

Відповідно до освітньої програми, згідно навчального плану та графіків навчального процесу, в кінці навчання студент виконує кваліфікаційну роботу. Тема роботи узгоджується з керівником проекту, випускаючою кафедрою, та затверджується наказом ректора.[10]

Отже, підставами для розробки «виконання кваліфікаційної роботи» є:

- освітня програма 122 «Комп'ютерні науки»;
- навчальний план та графік навчального процесу;
- завдання на кваліфікаційну роботу на тему «Розробка комп'ютерної відеоігри для мобільного пристрою на платформі Android».

1.4 Постановка завдання

Основним завданням кваліфікаційної роботи є розробка комп'ютерної відеоігри для мобільного пристрою на платформі Android.

Для кожної гри та додатка одним з найважливіших пунктів є – Дизайн. Початкового меню та рівнів, які повинні бути зроблені по стандартам UX та UI дизайну[11].

Спочатку були розроблені тестовий дизайн головного меню, так щоб людина могла побачити кнопку для запуску гри та переходу до рівнів. При створенні рівні вибирали приємні кольори, щоб користувач не відчував дискомфорт.

Кожен рівень має свій особистий стиль, який має одну тему з завданням цього рівня. Приклад рис.1.3 та рис 1.4. Це робиться для того, щоб користувач міг більш асоціювати та розуміти завдання. Також потрібно не забувати, що у кожної людини свої пристрої, які мають різні розміри екранів. Екрани телефонів за розміром поділяються на 4 види по густині та роздільній здатності(рис.1.5)

При натисканні на кнопку відбувається невеличка анімація, а саме вона починає заповнюватись білим кольором .

Для досягнення поставленої мети необхідно:

- вивчити предметну галузь розв'язуваної задачі;
- проаналізувати усі можливі технології та алгоритми для виконання поставлених задач;
- обрати технології, мову та архітектуру застосунку.



Рис 1.3. Особистий стиль рівня, варіант 1

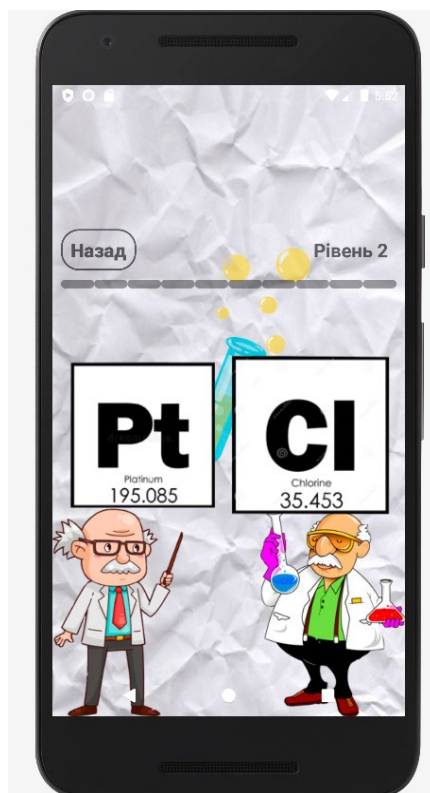


Рис.1.4. Особистий стиль рівня, варіант 2



Рис.1.5. Види густоти та розмірів екрану телефон

1.5 Вимоги до програми або програмного виробу

1.5.1 Вимоги до функціональних характеристик

Основною ціллю є розробка гри вікторина, але вона повинна містити в собі багато функцій та вимог для досягнення мети.

Гра призначена для розвитку розважального та захоплюючого характеру, але не потрібно забувати про приємний для користувача дизайн. Тому додаток повинен надавати користувачу наступне:

- Зрозумілий для користувача UI дизайн;
- Також сучасний та зрозумілий UX дизайн;
- Можливість зберігати прогрес;
- Зручний перехід між основним меню та рівнями;
- Змога обирати рівні;
- При кожному виборі рівня пояснювати ціль завдання;
- При кожному проходженні рівня, гравець має отримати інтересний факт;
- Змога повернутись до минулого рівня;
- При проходженні рівня перехід на інший.

Для досягнення мети у розробці гри потрібно:

- Дослідити предметну галузь;
- Дослідити галузі пов'язані з тематикою рівнів;
- Створення основного «скелету» для додавання нових рівнів;
- Розуміння у роботі з фоторедактором для створення стилів рівнів;
- Можливість зберігати прогрес.

1.5.2. Вимоги до інформаційної безпеки

Так як гра створена в ході кваліфікаційної роботи, то вимог то інформаційної безпеки не має. Тому що гра не зберігає жодної особистої інформації про користувача, не містить реклами та не просить давати доступ до інших програм мобільного пристрою..

Єдине що може врятувати гру від плагіату – це відсилки на розробника. У коді гри є декілька відсилок на розробника, які містяться у коментарях.

1.5.3. Вимоги до складу та параметри технічних засобів

Для використання гри вікторина нам буде потрібна мобільна система, тому що гра не підтримує розширення для комп'ютерних систем. Тому обираємо мобільну систему до якої відносяться телефони та планшети. Через це гра повинна бути сумісна з багатьма пристроями ,які є найбільш популярними у суспільства.(рис. 1.6)

Відповідно для використання гри мобільна система повинна бути не нижче версії Android – гра користується версією 8.0, цю версію підтримують 80% сучасних пристроїв. Також гра може працювати на більш нових версіях, новою останньою версією є, Android 13.0.[12]

Гра не забирає багато ресурсів у пристрою користувача, тому знадобляться лише мінімальні витрати. А саме:

- Процесор – 4 ядра (Max 2.30GGz);
- Оперативна пам'ять – 1 GB;

- Версія android – від 8.0 до 13.0;
- Потрібно місця – 126 МВ.

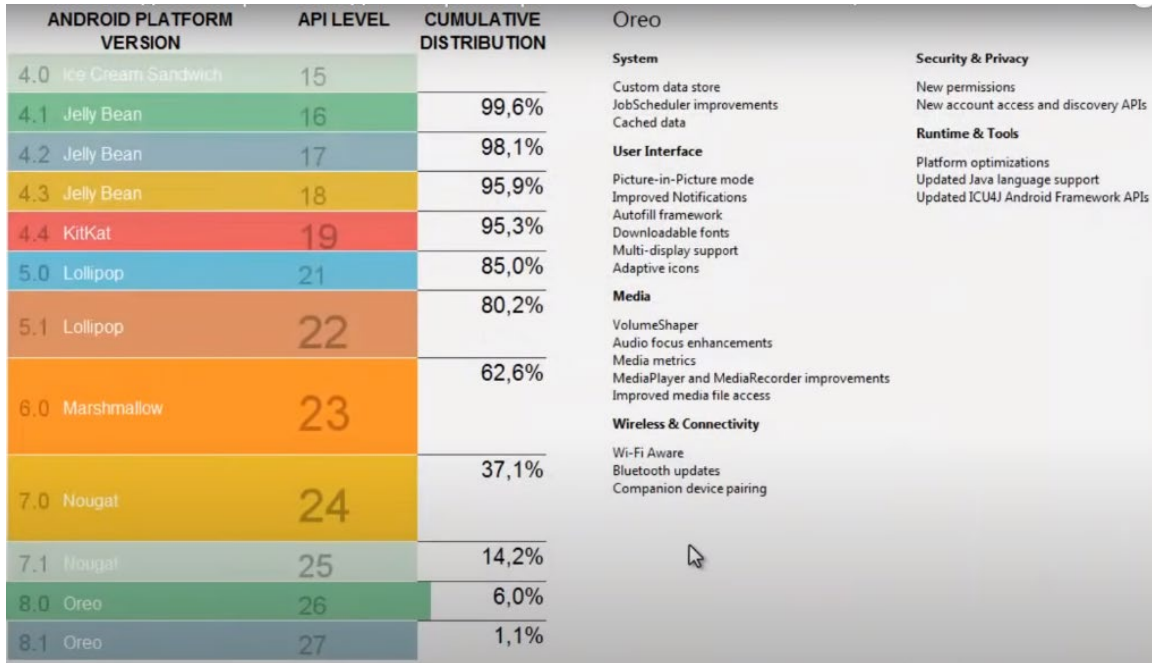


Рис.1.6. Статистика версій Android, які підтримують сучасні проекти.

РОЗДІЛ 2 ПРОЕКТУВАННЯ ТА РОЗРОБКА ІНФОРМАЦІЙНОЇ СИСТЕМИ

2.1. Функціональне призначення системи

За темою кваліфікаційної роботи є розробка гри на платформі андроїд. Ця гра має жанр – Вікторина.

Призначення розробленої гри:

- Викликати інтерес до української культури та всесвітньої науки;
- Перевірити знання у людини ,яка має інтерес до галузі ,які наданні у рівнях.

Для досягнення поставленої задачі , гра містить в собі наступне:

- Кожен рівень має свій дизайн;
- На початку рівня, з'являється діалогове вікно ,яке пояснює завдання для користувача;
- У рівні знаходяться 10 запитань;
- При виборі відповіді зображення буде показувати, чи є ваш вибір правильним;
- З проходженням рівня на гравця чекає інтересних факт з тему у стилі якої був рівень;
- Поки гравець не пройде рівень усі інші будуть заблоковані;
- Прогрес рівнів буде зберігатися.

2.2. Опис застосованих математичних методів

При розробці гри математичні методи не застосовувалися. Тому використовувалися програмні методи.

Було використано два методи по вибори правильної відповіді:

1. Перший метод полягає у тому, що завдання буде правильною відповідь на числі ,яке найбільше. Кожне зображення має нумерацію від 0 до 9.(рис. 2.1.)

```
public class Array {  
  
    //Масив для першого рівня - початок  
    6 usages  
    final int[] images1 = {  
        R.drawable.first_level_0,  
        R.drawable.first_level_1,  
        R.drawable.first_level_2,  
        R.drawable.first_level_3,  
        R.drawable.first_level_4,  
        R.drawable.first_level_5,  
        R.drawable.first_level_6,  
        R.drawable.first_level_7,  
        R.drawable.first_level_8,  
        R.drawable.first_level_9,  
    };  
  
    final int[] power1 = {0,1,2,3,4,5,6,7,8,9};  
  
    //Масив для першого рівня - кінець
```

Рис.2.1. Написання коду – масиву для першого метода

```

numLeft = random.nextInt( bound: 10);
img_left.setImageResource(array.images1[numLeft]); // з масиву зображення

numRight = random.nextInt( bound: 10);

//Цикл перевіряючий рівність чисел - початок
while (numLeft==numRight){
    numRight = random.nextInt( bound: 10);
}
//Цикл перевіряючий рівність чисел - кінець

img_right.setImageResource(array.images1[numRight]); // з масиву зображення

```

Рис.2.2. Код для вилучення зображення з масиву

2. Другий метод полягає у тому, щоб обрати вже не найбільше зображення. Нам потрібно обрати зображення, яке ми самі обрали вірним. Гра буде обирати зображення, які ми зробили більшими за інші та поділили їх між собою.(рис 2.3 та 2.4.). Також потрібно робити код для кожного зображення

```

final int[] images3={
R.drawable.chef_1,
R.drawable.job_1,
    R.drawable.chef_2,
    R.drawable.job_2,
    R.drawable.chef_3,
    R.drawable.job_3,
    R.drawable.chef_4,
    R.drawable.job_4,
    R.drawable.chef_5,
    R.drawable.job_5,
    R.drawable.chef_6,
    R.drawable.job_6,
    R.drawable.chef_7,
    R.drawable.job_7,
    R.drawable.chef_8,
    R.drawable.job_8,
    R.drawable.chef_9,
    R.drawable.job_9,
    R.drawable.chef_10,
    R.drawable.job_10,
};

42 usages
final int[] strong = {
    1,0,1,0,1,0,1,0,1,0,1,0,1,0,1,0,1,0,
};

```

Рис.2.3. Написання коду-масиву для другого рівня

```

}
//Якщо доторкнувся до зображення - кінець
if (count==10){
    //ВИХІД З РІВНЯ
    dialogEnd.show();
}else {
    numLeft = random.nextInt( bound: 20); //Генеруємо число
    img_left.setImageResource(array.images5[numLeft]); // З масиву зображення

    numRight = random.nextInt( bound: 20);

    //Цикл перевіряючий рівність чисел - початок
    while (array.strong[numLeft]==array.strong[numRight]){
        numRight = random.nextInt( bound: 20);
    }
    //Цикл перевіряючий рівність чисел - кінець

    img_rihgt.setImageResource(array.images5[numRight]); // З масиву зображення
    img_rihgt.setEnabled(true); //Включаем обратно правую картинку
}

```

Рис.2.4.. Код для вилучення зображення з масиву.

2.3. Опис технологій та мов програмування

2.3.1. Опис мови програмування Java

Так як гра створювалась у програмі AndroidStudio для її написання було доступно два види мов програмування. Один з них – це Java, а другий – Kotlin. (рис 2.5)

Для написання гри, було обрано мову програмування Java [13]. Тому що вона допомагає більш детально розібратися у кодї гри, та вивчення цієї мови дуже сильно допоможе у розумінні інших мов програмування.

Java — це об'єктно-орієнтована мова комп'ютерного програмування загального призначення, призначена для роботи в широкому діапазоні цифрових систем і пристроїв. Java-програми користуються майже всюди: центри обробки даних, веб-сайти, ноутбуки, ігрові консолі, суперкомп'ютери, смартфони та планшети.

Не забуваймо, що Java також є платформою. [14] Це означає, що це, по суті, набір програм, які допомагають розробникам ефективно програмувати та запускати програми Java на будь-якому цифровому пристрої, на якому є віртуальна машина Java (JVM).

Окрім використання для розробки програмного забезпечення для комп'ютерів і мобільних пристроїв, Java також використовується для програмування інших електронних апаратних пристроїв, таких як: кондиціонери, телевізори, холодильники, пральні машини та багато інших.

Найважливіше те, що Java — це мова програмування з відкритим вихідним кодом, яка надає велику колекцію безкоштовних бібліотек кодування та інструментів, які істотно скорочують час та інвестиційні витрати на розробку програмних додатків.

Мова Java часто називають WORA (Write Once and Run Anywhere) що робить його ідеальним для децентралізованих хмарних додатків. Постачальники хмарних послуг вибирають мову Java для запуску програм на

широкому спектрі базових платформ. Мова Java використовується для механізмів обробки даних, які можуть працювати зі складними наборами даних та великими обсягами даних у режимі реального часу.

Гра вікторина, яка була розроблена на мові Java не має бази даних. Це тому що ресурси програми зберігаються ,як файли тому в наявності бази даних немає необхідності. У Android є своя вбудованна БД – sqlite, але у розробці гри, вона нам не знадобиться.

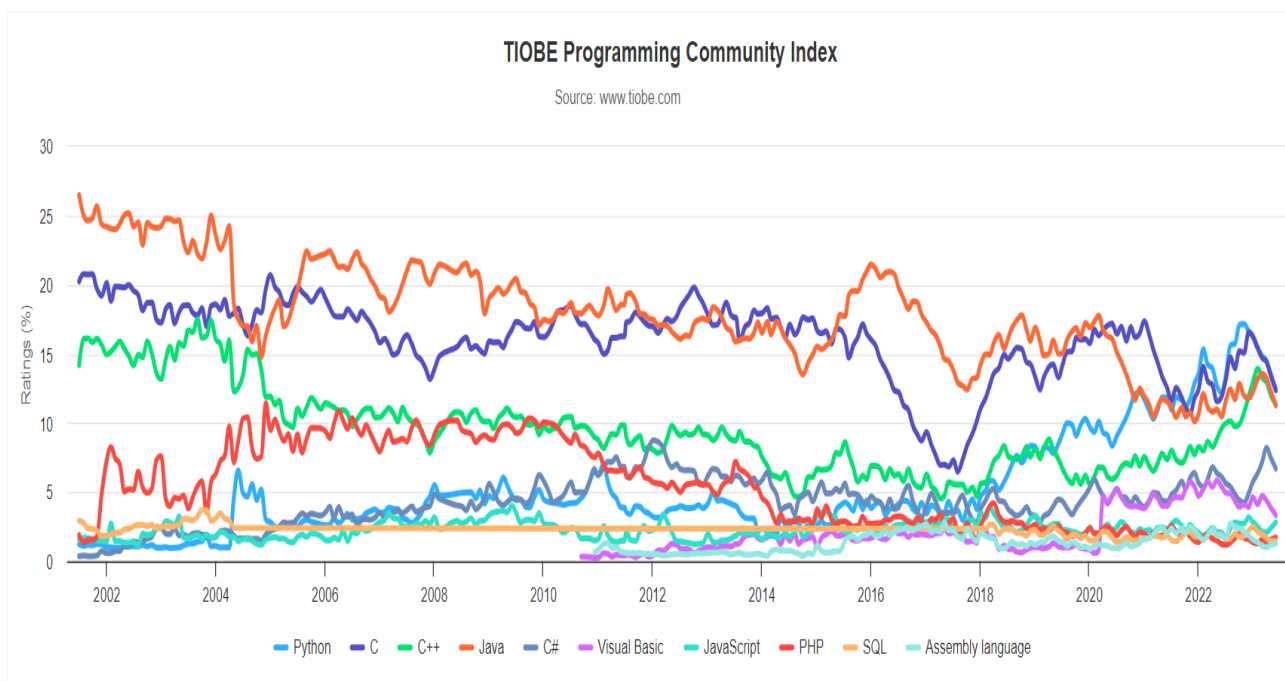


Рис.2.5. Графік користування мовами програмування[15]

2.3.2. Опис фоторедактора Photoshop

Для того щоб гра приємна виглядала для користувача, було обрано програму Photoshop. Цю програму може отримати кожна людина на сім днів з усім функціоналом та іншими програмами фірми Adobe. У створенні гри знадобився лише Photoshop. Завдяки ньому була змога робити дуже якісні та власні зображення. Кожне зображення, від стартового меню до усіх доступних рівнів гри, було зроблено завдяки цій програмі.

Adobe Photoshop – це багатофункціональний редактор, який дозволяє працювати із зображеннями та відеофайлами[16]. Продукт підходить для комп'ютерів з операційними системами Windows та MacOS. Є й мобільні версії фотошопу для Android, iOS, Windows Phone.

Adobe Photoshop знайомий практично кожному. Графічний редактор активно використовується як особистих, так комерційних цілей. Його застосовують дизайнери, кінематографісти, творці ігор та рекламних роликів. Фотошоп регулярно оновлюється, дбаючи про комфорт користувачів, при цьому адаптуватися до нововведень зазвичай не складно.

Незважаючи на те, що спочатку програма була розроблена для редагування зображень для друку на папері (насамперед, для поліграфії), зараз вона широко використовується в веб-дизайні.

Photoshop тісно пов'язаний з іншими програмами обробки медіафайлів, анімації та іншої творчості. Спільно з такими програмами, як Adobe Illustrator, Adobe Premiere, Adobe After Effects, він може використовуватися для створення професійних PSD, може бути експортований та імпортований у низку цих програмних продуктів.

Через високу популярність Photoshop, підтримка його формату файлів, PSD, була реалізована в його основних конкурентах, таких як Macromedia Fireworks, Corel Photo-Paint, Pixel image editor, WinImages,(рис 2.6)

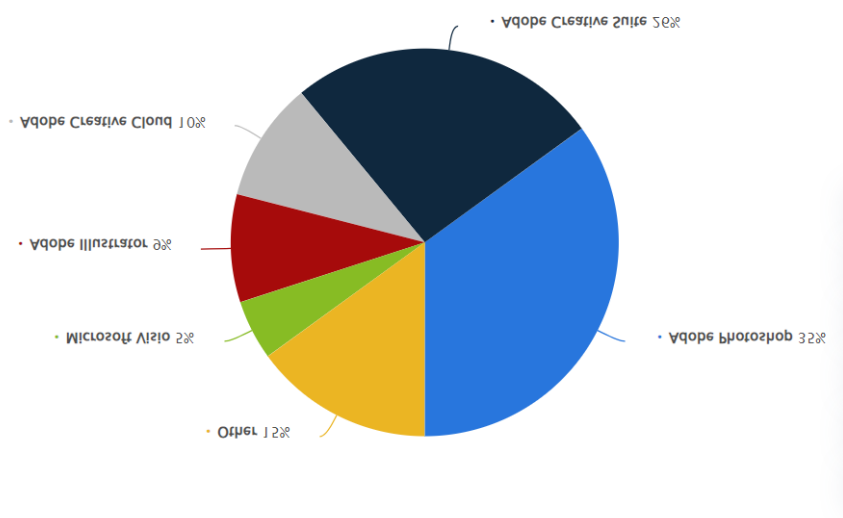


Рис.2.6. Схема використання Adobe Photoshop у 2022р.[17]

2.3.3. Опис програми Android Studio

Середовище розробки було вибрано програму Android Studio. Тому що ця програма зарекомендувала себе ,як дуже просте та зручне місце для створення відеогри.

Android Studio — це потужне та складне середовище розробки, розроблене спеціально для розробки, тестування та упаковки додатків Android. Його можна завантажити разом із Android SDK як єдиний пакет. Це набір інструментів і компонентів. Багато таких інструментів встановлюються та оновлюються незалежно один від одного.[18]

Як і багато програм, котрі мають свої аналоги та схожі по функціоналу продукти розробки, Android Studio також бориться за популярність серед інших програм з способом розробки програм для Android. Існують інші IDE, такі як Eclipse і NetBeans, і навіть можна розробити повну програму, використовуючи лише блокнот і командний рядок.[19]

Створена з певною метою, Android Studio приваблює все більше сторонніх плагінів, які надають великий набір цінних функцій, недоступних безпосередньо через IDE. До них входять плагіни для пришвидшення часу створення, налагодження проекту через Wi-Fi та багато іншого.

Незважаючи на те, що це, можливо, кращий інструмент, є кілька вагомих причин зупинитися на іншій IDE, наприклад Eclipse. Багато розробників розробляють для кількох платформ, що робить Eclipse гарним вибором інструменту. У кожного розробника є дедлайни, і знайомство з незнайомим програмним забезпеченням може спочатку значно сповільнити його роботу. Але Android studio є офіційною IDE для Android studio, і кожен розробник програм для Android повинен бути обережним щодо відмінностей між ними, щоб вони могли з'ясувати подібності та відмінності та побачити, що їм підходить.

2.4. Опис структури системи та алгоритм її функціонування

Структура гри – це показник того на скільки гравцю приємне грати на знаходитись у грі. Показник того чи розбирається він у інтерфейсі без усяких проблем. Звичайно гра вікторина повинна змушувати гравця думати , але тільки у момент проходження рівня. Якщо гравцю складно орієнтуватися у меню, то це дуже поганий показник. Тому чим простіша структура ,тим більш привабливий продукт.(рис.2.7.)

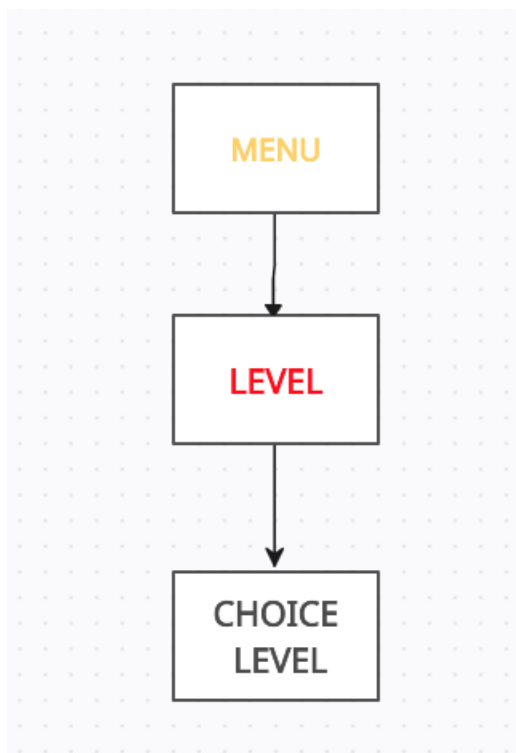


Рис.2.7. Схема пересування в інтерфейсі

Проект знаходиться у папці «app» ,що містить 3 основник папки в яких знаходиться гра та матеріали до неї.(рис. 2.8.)

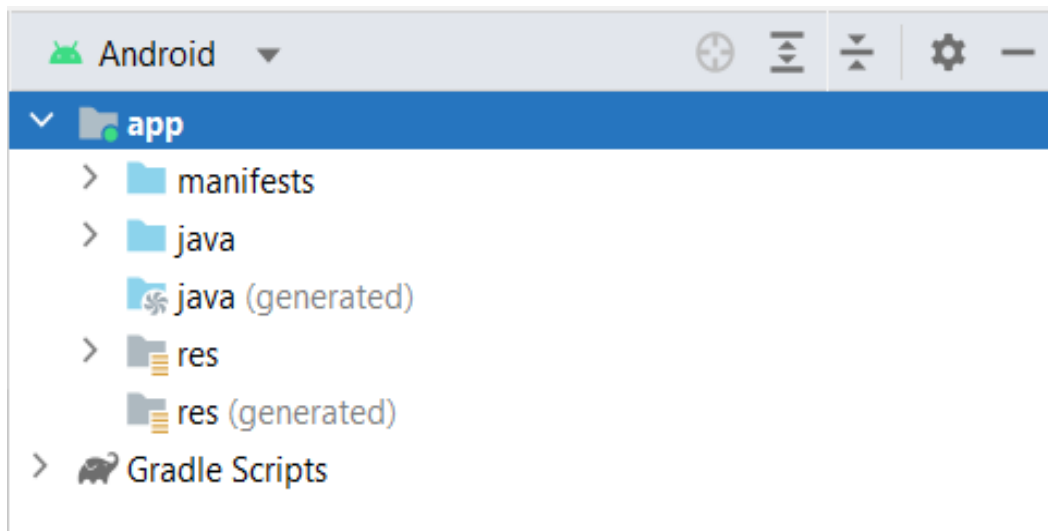


Рис.2.8. Файлова структура гри

Каталог «manifests» містить в собі файл «AndroidManifest.xml» - це основний файл завдяки якому програма розуміє, що перше повинно запуститися. У цьому файлі прописаний порядок запуску гри та умови переходу з меню гри у меню вибору рівнів з меню рівнів до їх вибору. Також у цьому файлі є можливість задавати умови запуску рівня, або основного меню. (рис 2.9.)

```

1 <?xml version="1.0" encoding="utf-8"?>
2 <manifest xmlns:android="http://schemas.android.com/apk/res/android"
3     xmlns:tools="http://schemas.android.com/tools">
4
5     <application
6         android:allowBackup="true"
7         android:dataExtractionRules="@xml/data_extraction_rules"
8         android:fullBackupContent="@xml/backup_rules"
9         android:icon="@mipmap/ic_launcher"
10        android:label="@string/QuizV2"
11        android:roundIcon="@mipmap/ic_launcher_round"
12        android:supportRtl="true"
13        android:theme="@style/Theme.QuizV2"
14        tools:targetApi="31">
15
16        <activity
17            android:name=".MainActivity"
18            android:exported="true"
19            android:screenOrientation="portrait">
20            <intent-filter>
21                <action android:name="android.intent.action.MAIN" />
22
23                <category android:name="android.intent.category.LAUNCHER" />
24            </intent-filter>
25        </activity>
26        <activity
27            android:name=".GameLevels"
28            android:exported="true"
29            android:screenOrientation="portrait">
  
```

Рис.2.9. Розгорнутий вид файлу «AndroidManifest.xml»

Розширювана мова розмітки (XML) – це мова розмітки, яка надає правила визначення будь-яких даних. На відміну від інших мов програмування, XML не може виконувати обчислювальні операції сам собою.

Папка «Java» містить в собі ще одну папку, а саме одну в назв проекту. На даний момент папка носить ім'я «space.stryuk.quizv2»(рис.2.10.)

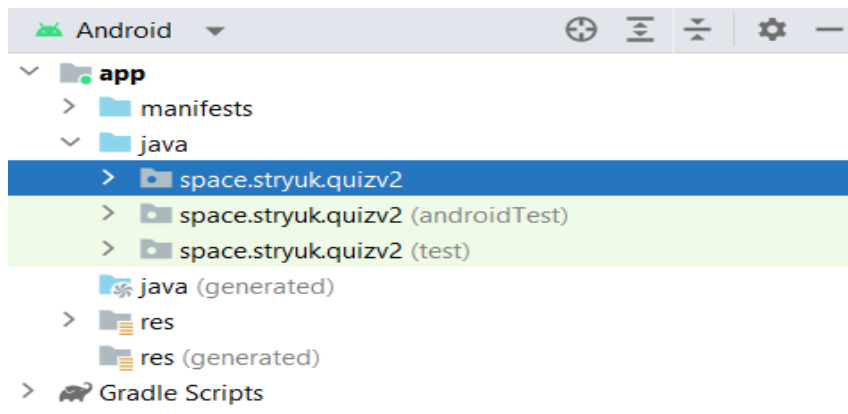


Рис.2.10. Файли ,які містить папка «Java»

Папка «space.stryuk.quizv2» містить в собі декілька файлів типу «Java» (рис.2.11.), а саме:

- Array;
- GameLevels;
- MainActivity;
- Level*.

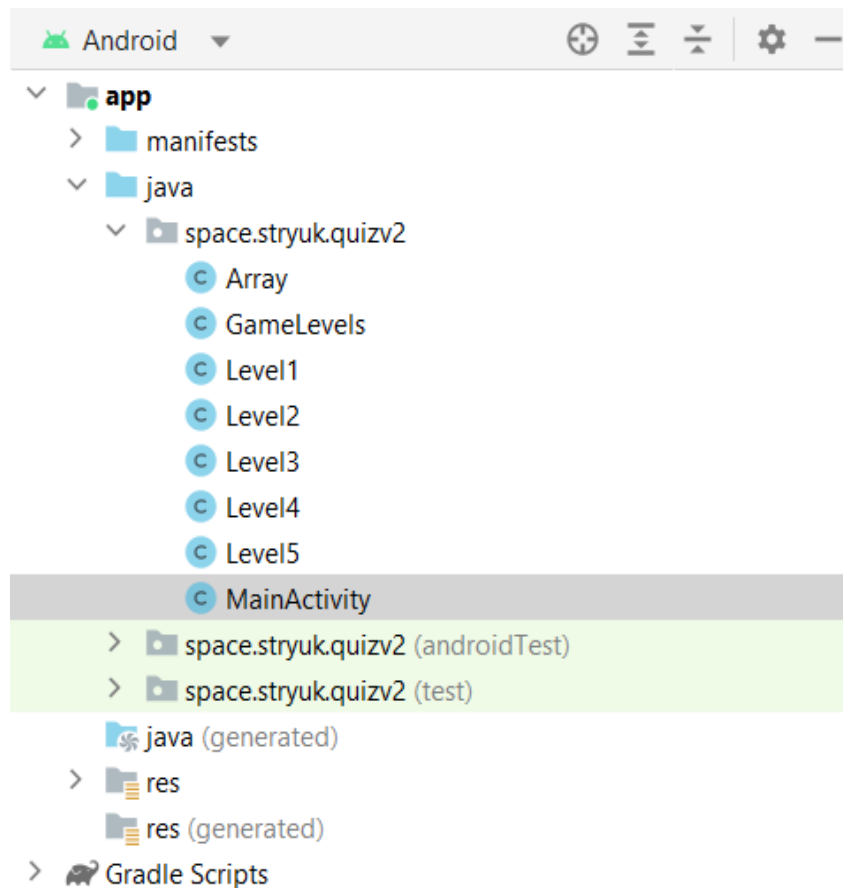


Рис.2.11. Файли, які містить папка «space.stryuk.quizv2»

MainActivity – цей файл відповідає за головне меню. Завдяки ньому є змога додати нові функції у головне меню(рис.2.12.). Зараз доступні такі функції:

- Функція кнопки «Початок» завдяки якій є можливість перейти у меню рівнів;
- Функція виходу з гри при подвійному натисканні на кнопку телефону «Назад».


```

MainActivity.java x
1 package space.stryuk.quizv2;
2
3 import ...
12
5 usages
13 public class MainActivity extends AppCompatActivity {
14     private long backPressedTime;
15     private Toast backToast;
16
17     @Override
18     protected void onCreate(Bundle savedInstanceState) {
19         super.onCreate(savedInstanceState);
20         setContentView(R.layout.activity_main);
21
22         Button buttonStart = (Button)findViewById(R.id.buttonStart);
23         buttonStart.setOnClickListener(new View.OnClickListener() {
24             @Override
25             public void onClick(View v) {
26                 try {
27                     Intent intent = new Intent( packageContext: MainActivity.this, GameLevels.class);
28                     startActivity(intent);finish();
29                 }catch (Exception e){
30
31                 }//кінець будування
32             }
33         });
34         Window w = getWindow();
35         w.setFlags(WindowManager.LayoutParams.FLAG_FULLSCREEN,WindowManager.LayoutParams.FLAG_FULLSCREEN);
36     }

```

Рис.2.12. Код з папки «MainActivity»

GameLevels – відповідає за налаштування рівнів гри, а саме за їх послідовністю та дозволяє закрити рівні до того моменту доки ви не пройшли рівень. (рис.2.13)

```

TextView textView1 = (TextView) findViewById(R.id.textView1);
textView1.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {
        try{
            if (level>=1) {
                Intent intent = new Intent( packageContext: GameLevels.this, Level1.class);
                startActivity(intent);
                finish();
            }else {
                //порожне
            }
        }catch (Exception e){
            //Порожне
        }
    }
}

```

Рис.2.13. Код переходу першого рівня

Array – файл містить в собі масиви завдяки яким ми можемо створювати математичні методи та користуватися за ради розробки рівнів. (рис.2.14)



```
1 package space.stryuk.quizv2;
2
3 public class Array {
4
5     //Масив для першого рівня - початок
6     6 usages
7     final int[] images1 = {
8         R.drawable.first_level_0,
9         R.drawable.first_level_1,
10        R.drawable.first_level_2,
11        R.drawable.first_level_3,
12        R.drawable.first_level_4,
13        R.drawable.first_level_5,
14        R.drawable.first_level_6,
15        R.drawable.first_level_7,
16        R.drawable.first_level_8,
17        R.drawable.first_level_9,
18    };
19
20    final int[] power1 = {0,1,2,3,4,5,6,7,8,9};
```

Рис.2.14. Приклад створення масиву для першого рівня.

Level* - цей файл містить в собі код рівня. Рівень будується з багатьох аспектів. Дозволяє запускати рівень через меню. Також відкриває діалогове вікно із зображенням теми рівням, дотриманням кольору приємному користувачу, а також з поясненням завданням. Також використовуються математичні методи, які наводили у розділі (Розділ 2.2). Передбачено баг з тим що можливо натиснути обидва варіанти відповіді, через це прогрес рівня буде стояти на місці, або буде змога дізнатися правильну відповідь. Під кожную кнопку підставлене зображення та анімація для розуміння, чи правильну відповідь ви обрали.

Ресурси (папка «res») — це додаткові файли та статичний вміст, який використовує наш код, наприклад растрові зображення, визначення макета, рядки інтерфейсу користувача, інструкції щодо анімації тощо.(рис.2.15.)

Завжди використовує зовнішні ресурси програми, такі як зображення та рядки, із вашого коду, щоб була змога підтримувати їх незалежно. Крім того, надання альтернативних ресурсів для певних конфігурацій пристроїв, згрупувавши їх у спеціально названих каталогах ресурсів. Під час виконання Android використовує відповідний ресурс на основі поточної конфігурації.

Наприклад, можливість надати інший макет інтерфейсу користувача залежно від розміру екрана або різні рядки залежно від налаштувань мови.

Після того, як виведете ресурси своєї програми назовні, ми зможемо отримати до них доступ за допомогою ідентифікаторів ресурсів, згенерованих у класі R проєкті. У цьому документі показано, як згрупувати ресурси у проєкті Android. Тут також показано, як надати альтернативні ресурси для певних конфігурацій пристроїв, а потім отримати доступ до них із коду програми чи інших XML-файлів.

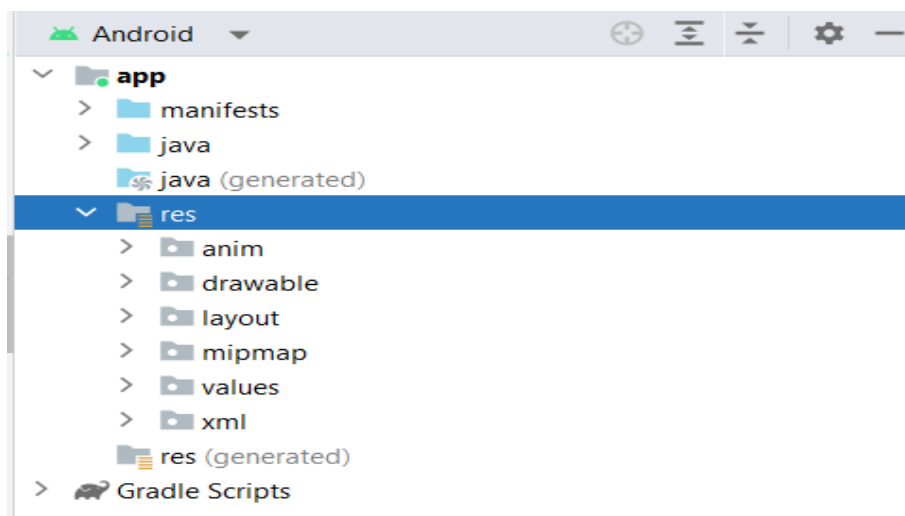


Рис.2.15. Папка «res»

У цій папці знаходяться ще папки, які містять файли для особистого призначення. Папка «res» містить такі файли:

- Папка «anim»;
- Папка «drawable»;
- Папка «layout»;
- Папка «mipmap»;
- Папка «values»;
- Папка «xml».

Ресурс «anim» може визначати один із двох типів анімації(рис 2.16):

Анімація власності – створює анімацію, змінюючи значення властивостей об’єкта протягом встановленого періоду часу за допомогою аніматора.

Переглянути анімацію у якій існує два типи анімації, які можна створювати за допомогою фреймворку перегляду анімації:

- Анімація Tween: створює анімацію, виконуючи серію перетворень на одному зображенні за допомогою анімації;
- Рамкова анімація: або створює анімацію, показуючи послідовність зображень у порядку за допомогою AnimationDrawable.

Анімація, визначена в XML, яка змінює властивості цільового об’єкта, як-от колір фону чи значення альфа-каналу, протягом встановленого періоду часу.

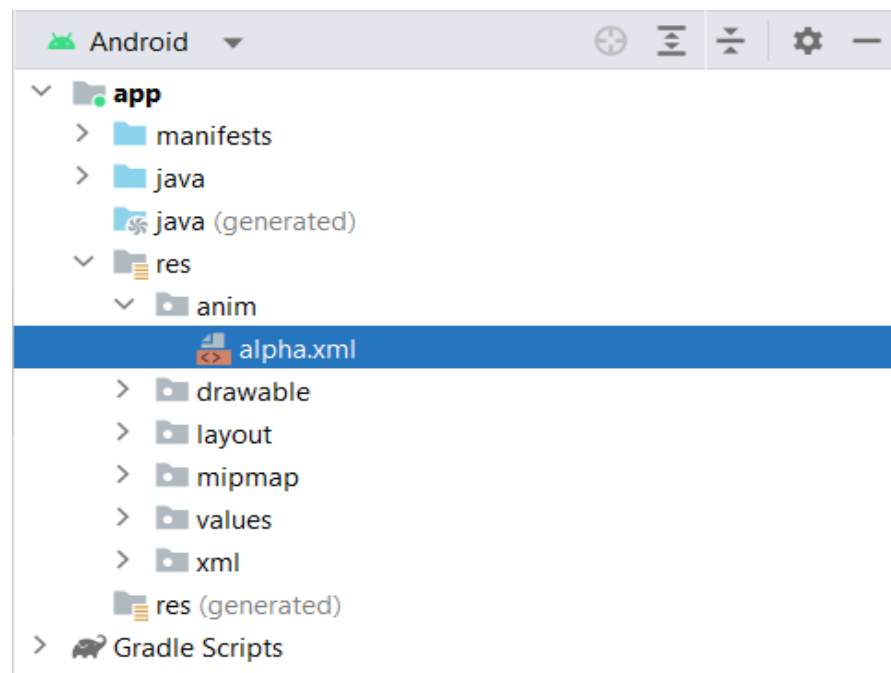


Рис.2.16. Що міститься в папці «anim»

Ресурс із можливістю малювання(drawable) — це загальна концепція графіки, яку можна намалювати на екрані, яку можна отримати за допомогою API, наприклад «getDrawable(int)» або застосувати до іншого XML-ресурсу з такими атрибутами, як «android:drawable» та «android:icon».

У папці мого проекту цей файл містить зображення з яких будується завдання. Зображення несе в собі теми рівнів, які користувач повинен пройти.

Також у папці знаходяться файли стилів(рис.2.18) – це файли які відповідають за стилі кнопок та приймають участь у невеличкій анімації при натисканні. Містить в собі логотип гри, якій відображується у меню та на іконці у меню мобільного пристрою.(рис. 2.17.)

Layout - Ресурс макета визначає архітектуру інтерфейсу користувача в дії або компоненті інтерфейсу користувача(рис.2.19). У папці «layout» знаходяться основні файли завдяки яким побудовані рівні та зроблені стилі з потрібними параметрами. Папка містить:

- activity_main.xml;
- dialog_end.xml;
- gamelevels.xml;
- previewdialog.xml;
- universal.xml.

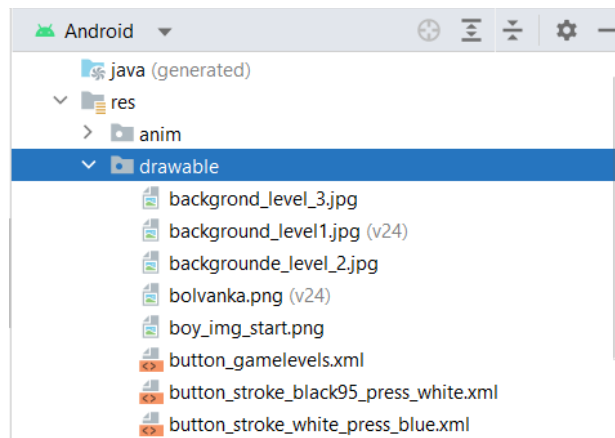


Рис. 2.17. Файли які містить папка «drawable»

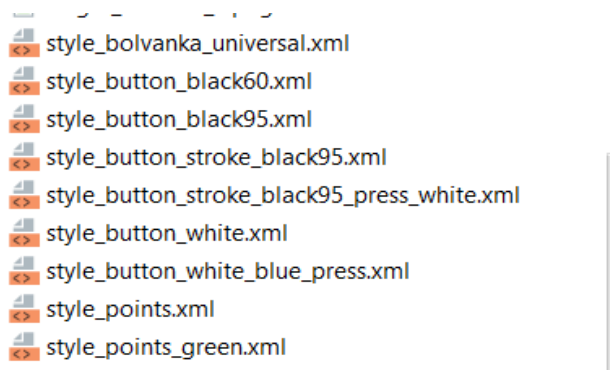


Рис.2.18. Стилi кнопок.

Файл «activity_main» - відповідає за стиль стартового меню. Саме у цьому файлі ми зможемо надавати параметри до кнопок та стилів, а саме до логотипа (logo) та заднього фону (background).

Файл «dialog_end» - відповідаю за налаштування стилю у діалоговому вікні при проходженні рівня. Містить в собі зображення (стиль) та текст з цікавим фактом по темі рівня.

Файл «gemalevels» - відповідає за вигляд меню вибору рівнів. В ньому міститься зображення заднього фону. Також розташовані кнопки до яких застосовуються стильові параметри з папка «drawable».

Файл «previewdialog» - відповідає за стиль діалогового вікна ,яке пояснює гравці ,що він потрібен робити для того щоб пройти рівень. Містить в собі кнопку «Далі» та «X» - закриття діалогового вікна та перехід у меню вибору рівня.

Файл «universal» - містить в собі побудову рівня.

- Перше – це стиль зображення якого обиралося під тему рівня;
- Функція кнопки «Назад» завдяки якій є змога повернутися у меню вибору рівня;
- Показник «Рівень» який зображую число рівня на якому знаходиться гравець;
- Строку прогресу , завдяки якій є можливість дізнатися скільки ще рівнів залишилося , до проходження рівня;

- Два прозорих зображення для виділення картки рівня.

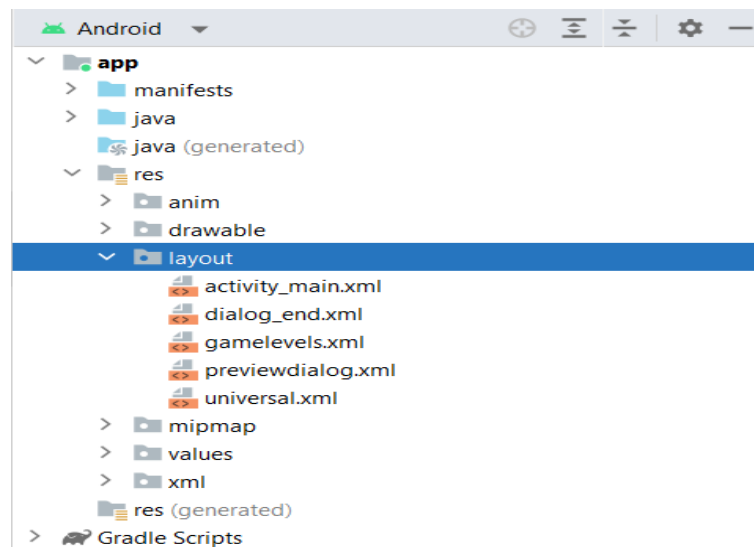


Рис.2.19. Файли які містить папка «layout»

Папка «тірмар» - містить в собі зображення для налаштування логотипу гри у меню телефону.

Папка «» - містить в собі базу кольорів та тексту, які використовуються у будованні рівнів в усіх інших параметрах. Якщо точніше:

- colors.xml – це база кольорів які були застосовані при написанні коди до стилів кнопок та анімації. (рис.2.20);
- descriptionslevel – це база яка містить в собі текст до діалогового вікна на початку та в кінці рівня. (рис.2.21);
- string.xml – містить в собі базу тексту, яка відповідає надання значення до кнопок та нумерації рівнів. (рис.2.22)

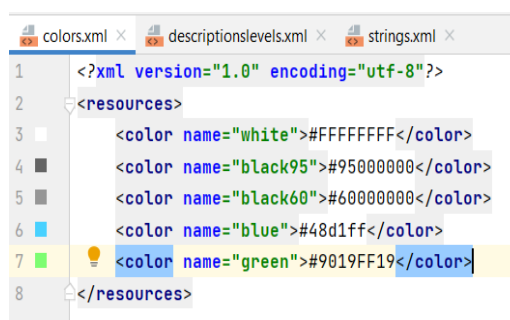
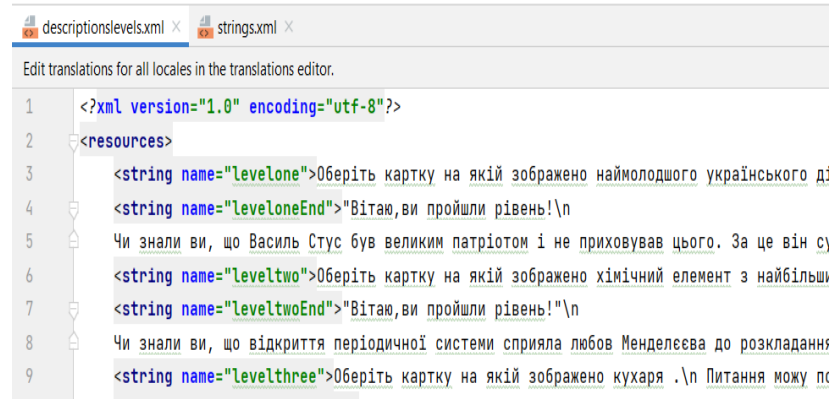



Рис. 2.20. Кольори ,які використовуються у створенні рівнів та стилів



```
<?xml version="1.0" encoding="utf-8"?>
<resources>
  <string name="levelone">Оберіть картку на якій зображено наймолодшого українського ді
  <string name="leveloneEnd">"Вітаю,ви пройшли рівень!\n
  Чи знали ви, що Василь Стус був великим патріотом і не приховував цього. За це він с)
  <string name="leveltwo">Оберіть картку на якій зображено хімічний елемент з найбільш
  <string name="leveltwoEnd">"Вітаю,ви пройшли рівень!\n
  Чи знали ви, що відкриття періодичної системи сприяла любов Менделєєва до розкладанн)
  <string name="levelthree">Оберіть картку на якій зображено кухаря .\n Питання можу пс
```

Рис.2.21. Тексти ,які використовуються у діалогових вікнах



```
<resources>
  <string name="app_name">QuizV2</string>
  <string name="start">Старт</string>
  <string name="back">Назад</string>
  <string name="level1">Рівень:</string>
  <string name="level_one">Рівень 1</string>
  <string name="level_two">Рівень 2</string>
  <string name="level_three">Рівень 3</string>
  <string name="level_four">Рівень 4</string>
  <string name="level_five">Рівень 5</string>
  <string name="textcontinue">Далі</string>
</resources>
```

Рис.2.22. Найменування рівнів ,назви проекту та кнопок

Для того щоб побудувати рівень,ми повинні розуміти з яких саме шматочків пазлу ми побудуємо рівень гри. Це можна побачити на рис. 2.23.

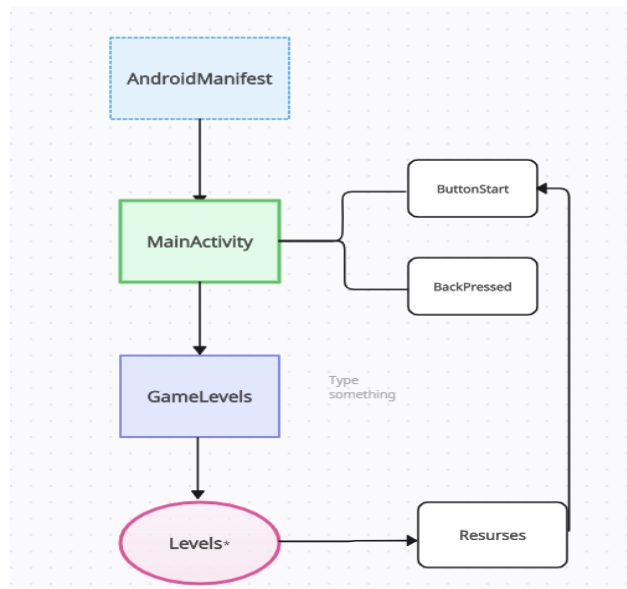


Рис.2.23. Зображення циклу створення рівнів

З побудування рівнів ми розуміємо ,що кожен проєкт має свою файлову структуру. Структура гри вікторини зображена на рис.2.24.

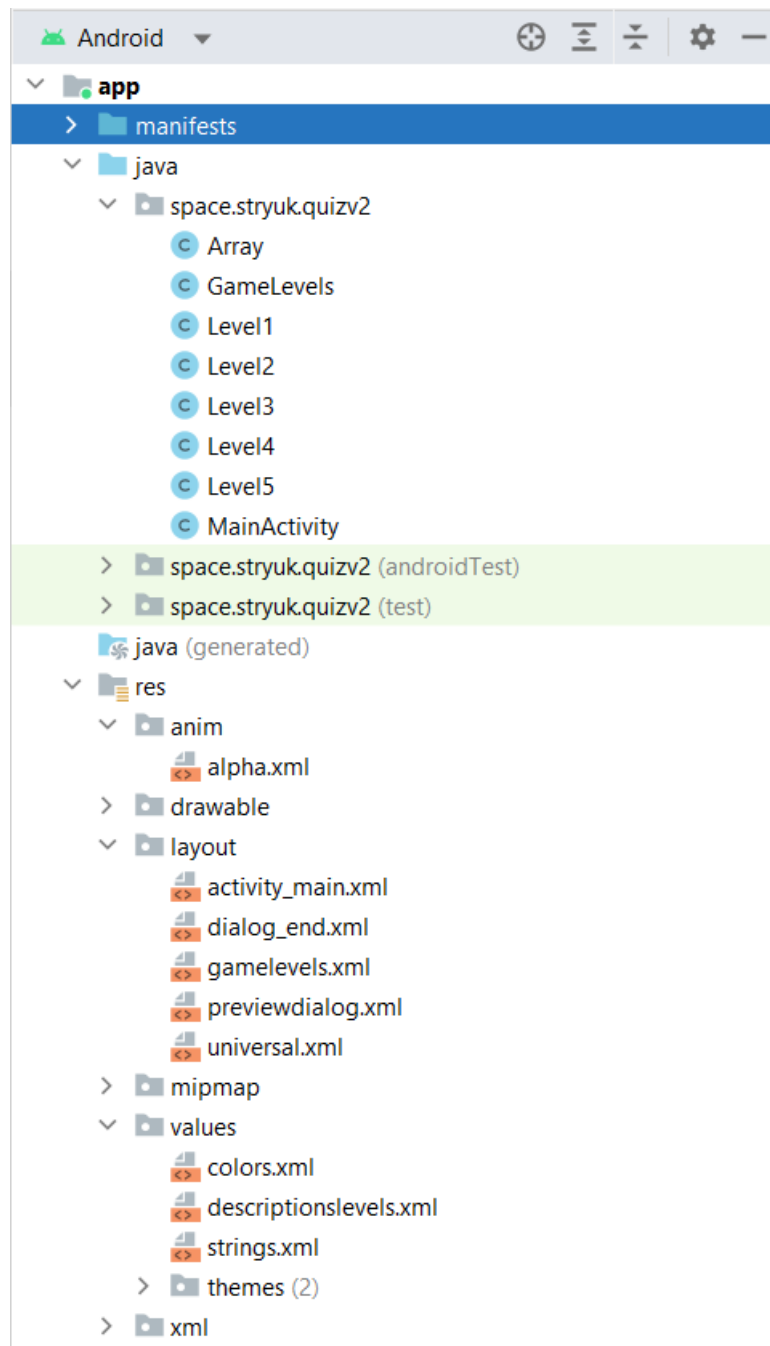


Рис.2.24. Структура файлової системи

2.5. Обґрунтування та організація вхідних та вихідних даних програми

Програмне забезпечення отримає вхідні дані при внесенні їх гравцем. Завантаження інформації з файлів.

Вхідні дані:

- Числа з масиву;
- Зображення .

Вихідні дані:

- Масив з числом від 0 до 9;
- Зображення з даними поверх нього.

Файли зображення зберігаються тільки у самій грі. У користувача не має можливості додавати свої особисті зображення чи математичні методи.

2.6. Опис роботи розробленої системи

2.6.1. Використані технічні засоби

Пристрій на якому проводилася розробка та тестування:

- Операційна система Windows 10;
- Засоби клавіатура та комп'ютерна миша;
- Охолодження;
- Монітор розміром 1920 x 1080, 60 hertz;
- Оперативна пам'ять 16 GB;
- Доступ до мережі Internet (Google) ;
- Процесор – intel Core i5-9300H Coffe Lake 2.40 GHz.

Нижче на рисю 2.27 буде показані технічні заміри працюючого застосунку. Також до уваги треба взяти те, що використані характеристики є рекомендованими. Тому що може статися так, що гра не запуститься на інших пристроях для редагування коду.

Имя	Состояние	7% ЦП	74% Память	0% Диск	0% Сеть
Приложения (7)					
Android Studio (7)		3,3%	2 801,7 МБ	0,1 МБ/с	0 Мбит/с
crashpad_handler		0%	0,9 МБ	0 МБ/с	0 Мбит/с
emulator		0%	0,9 МБ	0 МБ/с	0 Мбит/с
Filesystem events processor		0%	0,1 МБ	0 МБ/с	0 Мбит/с
OpenJDK Platform binary		0%	455,5 МБ	0,1 МБ/с	0 Мбит/с
qemu-system-i386		3,1%	454,0 МБ	0,1 МБ/с	0 Мбит/с
QuizV2		0,2%	1 884,6 МБ	0 МБ/с	0 Мбит/с
Хост окна консоли		0%	5,7 МБ	0 МБ/с	0 Мбит/с

Рис.2.27. Технічні заміри застосунку

2.6.2. Використані програмні засоби

Програма написана за допомогою мови програмування Java. З використанням фоторедактора Photoshop завдяки якому було зроблені усі зображення для рівнів ,окрім анімації натискання кнопок і самий їх вид.

В якості IDE було обрано – AndroidStudio. Так як це середовище кожен рік має по декілька оновлень, допомагає вивчити нову мову програмування таку як Java та Kotlin. Дає можливість запустити різні види пристроїв від мобільного до телевізора та перевірити сумісність гри з різними екранними розширеннями.

2.6.3. Виклик та завантаження програми

Для виклику та завантаження програми потрібно виконати наступне:

- Завантаження ресурсів гри ,таких як: зображення,стилі,логотип, кольорів;
- Виклин через натискання на іконку програми для виклику стартового меню.

2.6.4. Опис інтерфейсу користувача

Інтерфейс у грі доволі простий. Спочатку користувача зустріне початкове меню з логотипом та фоновим зображенням.(рис.2.25)



Рис. 2.25. Зображення початкового меню

Користувач при запуску гри бачить стартове меню. Для того щоб почалась гра потрібно натиснути на кнопку «Почати»(рис2.26). При натисканні пройде невелика анімація ,кнопка заповниться білим кольором.

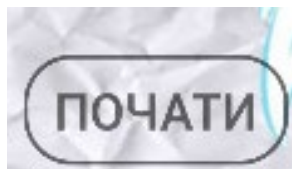


Рис. 2.26. Кнопка «Почати»

При натисканні на кнопку «Почати» гравець переходить у меню рівнів(рис.2.27). Де йому потрібно пройти рівень ,щоб відкрити інший. Доки гравець не пройде рівень інші будуть заблоковані, що характеризується малюнком «X». (рис.2.28)



Рис.2.27. Меню рівнів гри



Рис.2.28. Кнопка розблокованого та заблокованого рівнів

Якщо гравець хоче повернутися у стартове меню, він може скористатися кнопкою «Назад» за чим піде анімація натискання у меню рівнів (рис.2.29) або скористатися кнопкою «Назад» мобільного. (рис.2.30)

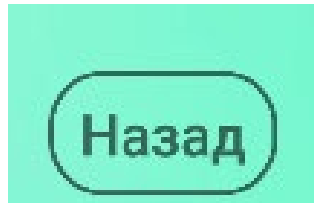


Рис. 2.29. Кнопка «Назад» меню вибору рівнів



Рис.2.30. Кнопка «Назад» мобільного пристрою

При виборі рівня для гравця з'явиться діалогове вікно на якому буде опис завдання та кумедне зображення, яке підходить до стилю рівня. (рис.2.31)

Також діалогове вікно має дві функціональні кнопки. Кнопка «Далі» відповідає за те що гравець зрозумів зміст завдання та готовий його вирішити, при натисканні проходить анімація.(рис.2.32).

У діалоговому вікні є функціональна кнопка «X» завдяки якій є змога повернутися у меню рівнів.(рис.2.33)



Рис.2.31. Діалогове вікно про запуску рівня



Рис.2.32. Кнопка «Далі»



Рис.2.33. Кнопка, яка повертає у меню рівня

Після натискання кнопки «Далі» гравцю відкривається рівень гри де йому потрібно обрати правильну відповідь.(рис.2.34). Також у рівні їде нумерація «Рівень 2» (рис.2.35). В кожному меню є кнопка «Назад» щоб повернутися у попереднє меню (рис.2.36).

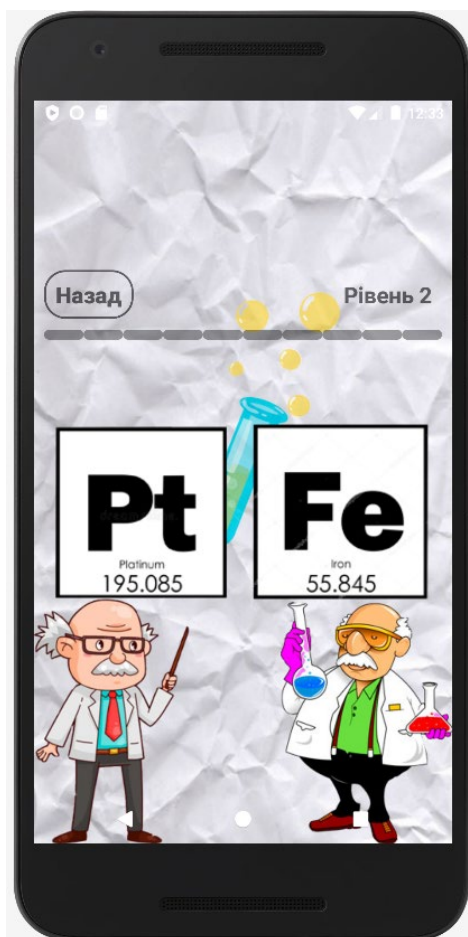


Рис.2.34. Рівень



Рис.2.35. Нумерація рівня



Рис. 2.36. Кнопка «Назад» у рівні гри

Якщо гравець правильно обрав картинку, то зображення підсвітиться зеленим зображенням (рис.2.37) та стрічка прогресу (рис.2.38) буде

заповнюватись, але якщо гравець обрав не правильну відповідь, то прогрес скинеться на 2 маркера.



Рис. 2.37. Анімація вибору картинки

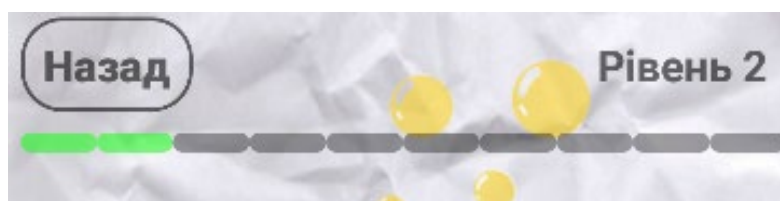


Рис.2.38. Стрічка прогресу рівня

Після проходження рівня з'явиться нове діалогове вікно, яке повідомить гравця про проходження рівня та відкриття нового (рис.2.39) Діалогове вікно містить в собі цікавий фак з теми рівня із функцією «скролу», а також кнопку «<>»(рис.2.40), яка повертає гравця у меню.

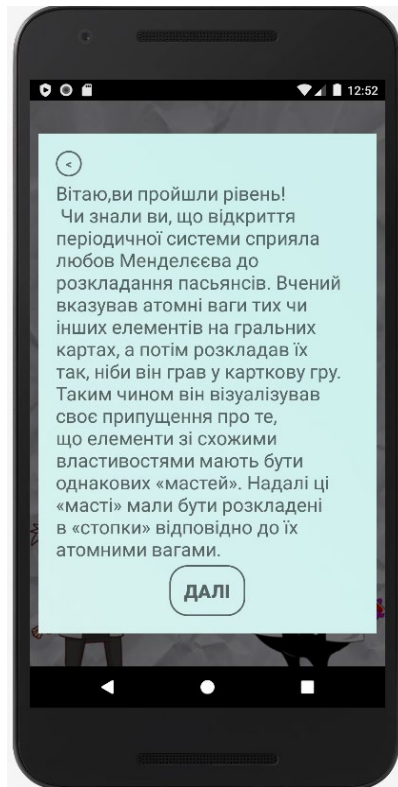


Рис. 2.39. Діалогове вікно по завершенню рівня



Рис.2.40. Кнопка повернутися у меню

Після проходження рівнях, наступний рівень буде розблоковано. Так буде доки гравець не пройде усі рівні гри. Прогрес проходження буде збережено.

РОЗДІЛ 3 ЕКОНОМІЧНИЙ РОЗДІЛ

3.1 Розрахунок трудомісткості та вартості розробки програмного продукту

Початкові дані:

1. передбачуване число операторів програми – 3530;
2. коефіцієнт кореляції програми в ході її розробки – 0.05;
3. коефіцієнт складності програми – 1.25;
4. годинна заробітна плата програміста– 104 грн/год;

Завдяки сайту (jobs.dou.ua), який вносить статистику з опитуванні користувачів та розташованих на ньому вакансій, по статистиці на грудень 2022 року, програміст Intern Software Engineer заробляє 450\$ у місяць праці[20].

Наразі курс долару до гривня становить 36.6 грн. За чотири тижні людина відпрацьовує 160 годин. Працюючи по 8 годин кожні 5 днів.

5. коефіцієнт збільшення витрат праці в наслідок недостатнього опису задачі – 1.25;
6. коефіцієнт кваліфікації програміста, обумовлений від стажу роботи з даної спеціальності – 1;
7. вартість машино-години ЕОМ – 22грн/год.

Комп'ютер моєї моделі (Nitro 5 AN515-45-R5S3) при 90% потужності витрачає 95 вват. Сходячи з того, що мій ПК працює увесь день та навантажується через постійні процеси пов'язані з іншими програмами за 10 годин він тратить 1 140вват. Також якщо рахувати користування опаленням завдяки електричному котлу (у холодні часи) ,чи кондиціонеру (у спекотні дні) споживання зростає до 8кВат[21]. За годину у комфортному середовищі уходить 8,095кВат. За даними Міністерства Фінансів курс складає 2.64грн за 1кВат[22].

Нормування праці в процесі створення ПЗ істотно ускладнено в силу творчого характеру праці програміста. Тому трудомісткість розробки ПЗ може бути розрахована на основі системи моделей з різною точністю оцінки.

Трудомісткість розробки ПЗ можна розрахувати за формулою:

$$t = t_o + t_u + t_a + t_n + t_{oml} + t_\delta, \text{ людино-годин} \quad (3.1)$$

де t_o - витрати праці на підготовку й опис поставленої задачі (приймається 50 людино-годин);

t_u - витрати праці на дослідження алгоритму рішення задачі;

t_a - витрати праці на розробку блок-схеми алгоритму;

t_n -витрати праці на програмування по готовій блок-схемі;

t_{oml} -витрати праці на налагодження програми на ЕОМ;

t_δ - витрати праці на підготовку документації.

Складові витрати праці визначаються через умовне число операторів у програмному забезпеченні, яке розробляється.

Умовне число операторів (підпрограм):

$$Q = q \cdot C \cdot (1 + p) \quad (3.2)$$

де q - передбачуване число операторів (3530);

C - коефіцієнт складності програми (1,25);

p - коефіцієнт кореляції програми в ході її розробки (0,05).

Звідси умовне число операторів в програмі:

$$Q = 1,25 \cdot 3530 \cdot (1 + 0,05) = 4633$$

Витрати праці на вивчення опису задачі t_u визначається з урахуванням уточнення опису і кваліфікації програміста:

$$t_u = \frac{Q \cdot B}{(75..85) \cdot k}, \text{ люДИНО-ГОДИН} \quad (3.3)$$

де B - коефіцієнт збільшення витрат праці внаслідок недостатнього опису задачі;

k - коефіцієнт кваліфікації програміста, обумовлений від стажу роботи з даної спеціальності. При стажі роботи в один рік він складає 1.

З урахуванням коефіцієнта кваліфікації $k = 1$, отримуємо витрати праці на вивчення опису завдання:

$$t_u = (4633 \cdot 1.25) / (75 \cdot 1) = 77 \text{ люДИНО-ГОДИН}$$

Витрати праці на розробку алгоритму рішення задачі визначаються за формулою:

$$t_a = \frac{Q}{(20...25) \cdot k} \text{ люДИНО-ГОДИН} \quad (3.4)$$

де Q – умовне число операторів програми;

k – коефіцієнт кваліфікації програміста.

Підставивши відповідні значення в формулу (3.4), отримаємо:

$$t_a = 4633 / (20 \cdot 1) = 231,65 \text{ люДИНО-ГОДИН.}$$

Витрати на складання програми по готовій блок-схемі:

$$t_n = \frac{Q}{(20...25) \cdot k}, \text{ люДИНО-ГОДИН} \quad (3.5)$$

$$t_n = 4633 / (25 \cdot 1) = 185,32 \text{ людино-годин.}$$

Витрати праці на налагодження програми на ЕОМ:

–за умови автономного налагодження одного завдання:

$$t_{oml} = \frac{Q}{(4..5) \cdot k}, \text{ людино-годин} \quad (3.6)$$

$$t_{oml} = 4633 / (5 \cdot 1) = 926,6 \text{ людино-годин.}$$

–за умови комплексного налагодження завдання:

$$t_{oml}^k = 1,5 \cdot t_{oml}, \text{ людино-годин} \quad (3.7)$$

$$t_{oml}^k = 1,5 \cdot 926,6 = 1389,9 \text{ людино-годин.}$$

Витрати праці на підготовку документації визначаються за формулою:

$$t_d = t_{dp} + t_{do}, \text{ людино-годин} \quad (3.8)$$

де t_{dp} - трудомісткість підготовки матеріалів і рукопису:

$$t_{dp} = \frac{Q}{(15..20) \cdot k}, \text{ людино-годин} \quad (3.9)$$

t_{do} - трудомісткість редагування, печатки й оформлення документації:

$$(3.10)$$

$$t_{\partial o} = 0,75 \cdot t_{\partial p}, \text{ людино-годин}$$

Підставляючи відповідні значення, отримаємо:

$$t_{\partial p} = 4633 / (18 \cdot 1) = 257,38 \text{ людино-годин.}$$

$$t_{\partial o} = 0,75 \cdot 257,38 = 185,31 \text{ людино-годин.}$$

$$t_{\partial} = 257,38 + 185,31 = 442,69 \text{ людино-годин.}$$

Повертаючись до формули (3.1), отримаємо повну оцінку трудомісткості розробки програмного забезпечення:

$$t = 50 + 77 + 231,65 + 185,32 + 926,6 + 442,69 = 1913 \text{ людино-годин.}$$

3.2 Розрахунок витрат на створення програми

Витрати на створення ПЗ $K_{\text{ПО}}$ включають витрати на заробітну плату виконавця програми $Z_{\text{ЗП}}$ і витрат машинного часу, необхідного на налагодження програми на ЕОМ:

$$K_{\text{ПО}} = Z_{\text{ЗП}} + Z_{\text{МВ}}, \text{ грн} \quad (3.11)$$

Заробітна плата виконавців визначається за формулою:

$$Z_{\text{ЗП}} = t \cdot C_{\text{ПР}}, \text{ грн} \quad (3.12)$$

де t - загальна трудомісткість, людино-годин,

$C_{\text{ПР}}$ - середня годинна заробітна плата програміста, грн/година

З урахуванням того, що середня годинна зарплата програміста становить 125 грн / год, отримуємо:

$$Z_{\text{ЗП}} = 1913 \cdot 104 = 198\,952 \text{ грн.}$$

Вартість машинного часу, необхідного для налагодження програми на ЕОМ, визначається за формулою:

$$Z_{мв} = t_{отл} \cdot C_{мч}, \text{ грн}, \quad (3.13)$$

де $t_{отл}$ - трудомісткість налагодження програми на ЕОМ, год;

$C_{мч}$ - вартість машино-години ЕОМ, грн/год (12 грн/год).

Підставивши в формулу (3.14) відповідні значення, визначимо вартість необхідного для налагодження машинного часу:

$$Z_{мв} = 926,6 \cdot 22 = 20\,385,2 \text{ грн.}$$

Звідси витрати на створення програмного продукту:

$$K_{ПО} = 198\,952 + 20\,385,2 = 219\,337,2 \text{ грн.}$$

Очікуваний період створення ПЗ:

$$T = \frac{t}{B_k \cdot F_p}, \text{ міс.} \quad (3.14)$$

де B_k - число виконавців (дорівнює 1);

F_p - місячний фонд робочого часу (при 40 годинному робочому тижні $F_p=176$ годин).

Звідси, за формулою (3.14) витрати на створення програмного продукту:

$$T = 1913 / (1 \cdot 176) \approx 10 \text{ міс.}$$

Висновок: Гру розроблено для поширення української культури у суспільстві та виклик людям, які цікавляться наукою. Вартість даного програмного забезпечення становить близько 219 337,2грн. Очікуваний час розробки становить 1913 годин, тобто орієнтовно десять місяців. Цей термін пов'язаний зі значним числом операторів, і включає час на написання документації-опису, необхідної для створення додатку, вивчення інших програм та IDE, дослідження і розробку алгоритму вирішення поставленого завдання, програмування за готовим алгоритмом, налагодження і розгортання програми, та підготовку вторинної документації, що допомагає у підтримці розробленого ПЗ.

ВИСНОВКИ

Головною метою кваліфікаційної роботи було розробка комп'ютерної відеоігри для мобільного пристроїв на платформі Android.

У продовж часу була обрана гра вікторина, яка повинна була містити в собі цікавий дизайн ,українську локалізацію та викликати у користувача інтерес до української культури починаючи з історії закінчуючи художніми витворами мистецтва. Саме на це була основна тема, тому що дуже малий процент людей може назвати популярних діячів 20-го століття, а що казати про інші роки.

Практичне значення гри полягала у тому ,щоб зацікавити гравця завдяки темам рівнів, які буде цікаво проходити та нагадувати собі те що забули, дізнатися того , що не знали. У гри були намагання зробити приємний та зрозумілий дизайн, завдання які становилися складнішими з кожнім проходженням та приємний бонус у вигляді цікавого факту з теми рівня.

Підчас виконання роботи були виконані наступні задачі:

- Прийняте рішення про жанр гри для розробки;
- Вивчення предметної галузі;
- Створення дизайну з дотриманням усіх пунктів;
- Написання функціоналу до різних видів мобільних пристроїв;
- Взаємодія з іншими програмами ,такими як Photoshop;
- Розбір IDE у якому була створена гра.

Розроблений застосунок дозволяє:

- Перевірити знання користувача;
- Розвиток мислення;
- Багато рівнів дозволять скоротити час.

Гру було створено у середовищі AndroidStudio за допомогою мови Java. При написанні кодів гри, був створений код завдяки якому є змога добавляти власні рівні на місці розробник, з власними умовами, зображеннями, стилями та фактами.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Video Game History [Електронний ресурс] - Режим доступу до ресурсу: <https://www.history.com/topics/inventions/history-of-video-games>
2. How gaming industry has evolved over the last 20 years [Електронний ресурс] - Режим доступу до ресурсу: <https://techxmedia.com/how-gaming-industry-has-evolved-over-the-last-20-years/>
3. Casino elements in games [Електронний ресурс] - Режим доступу до ресурсу: <https://thefootballlovers.com/football-questions/what-is-the-fifa-gambling-scandal>
4. Українська локалізація відомої гри Alan Wake 2 [Електронний ресурс] - Режим доступу до ресурсу: <https://itc.ua/ua/novini/ne-tilku-ukrayinska-lokalizatsiya-alan-wake-2-remedy-takozh-pereklala-ukrayinskoyu-ofitsijnyj-sajt-gry/>
5. Minecraft Education [Електронний ресурс] - Режим доступу до ресурсу: <https://education.minecraft.net/en-us/discover/what-is-minecraft>
6. Statistics on the most popular game genres [Електронний ресурс] - Режим доступу до ресурсу: <https://www.statista.com/statistics/1263585/top-video-game-genres-worldwide-by-age/>
7. The State of Independent Game Development in 2022 Режим доступу до ресурсу: <https://indiebandits.com/2022/10/17/the-state-of-independent-game-development-in-2022/>
8. Types of game engines [Електронний ресурс] - Режим доступу до ресурсу: <https://www.perforce.com/blog/vcs/most-popular-game-engines>
9. The global mobile gaming market [Електронний ресурс] - Режим доступу до ресурсу: <https://headphonesaddict.com/mobile-gaming-statistics/>
10. Buku Mobile Programming for Students - <https://repository.unmul.ac.id/bitstream/handle/123456789/4926/Buku%20Mobile%20Programming%20for%20Students.pdf?sequence=1>

11. UX UI Design Rules [Електронний ресурс] - Режим доступу до ресурсу: <https://topuser.pro/en/15-rules-ui-ux-design/>
12. Top Android OS versions - <https://www.appbrain.com/stats/top-android-sdk-versions>
13. Description of the Java programming language [Електронний ресурс] - Режим доступу до ресурсу: <https://www.ibm.com/topics/java>
14. Why and How is Java Platform Independent [Електронний ресурс] - Режим доступу до ресурсу: <https://www.simplilearn.com/why-and-how-is-java-platform-independent-article>
15. Programming language statistics [Електронний ресурс] - Режим доступу до ресурсу: <https://www.tiobe.com/tiobe-index/>
16. Description of Photoshop [Електронний ресурс] - Режим доступу до ресурсу: <https://www.techtarget.com/whatis/definition/Photoshop#:~:text=Photoshop%20is%20an%20image%20creation,raster%20graphics%20and%20vector%20graphics.>
17. Computer graphics and photo editing software market share worldwide in 2022, by product [Електронний ресурс] - Режим доступу до ресурсу: <https://www.statista.com/statistics/1223620/graphics-photo-editing-software-market-share/>
18. Description of Android Studio [Електронний ресурс] - Режим доступу до ресурсу: <https://developer.android.com/studio/intro>
19. Most Popular Development Environments [Електронний ресурс] - Режим доступу до ресурсу: <https://www.itwriting.com/blog/11471-stack-overflow-survey-shows-leap-in-popularity-for-visual-studio-code.html>
20. Сайт для пошуку статистики по ЗП в Україні [Електронний ресурс] - Режим доступу до ресурсу: <https://jobs.dou.ua/salaries/?period=2022-12&position=Intern/Trainee%20SE&technology=Java&experience=0>
21. Інструкція по витратам у електричних котлів [Електронний ресурс] - Режим доступу до ресурсу: <https://gradusplus.com/kotly/elektricheskie/skolko-elektroenergii-potreblyayet-elektrokotel-otopleniya/>

22.Сайт Міністерства Фінансів [Електронний ресурс] - Режим доступу до ресурсу: <https://index.minfin.com.ua/tariff/electric/index.php/>

КОД ПРОГРАМИ

AndroidManifest.xml

```
<?xml version="1.0" encoding="utf-8" ?>
<manifest xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:tools="http://schemas.android.com/tools">

    <application
        android:allowBackup="true"
        android:dataExtractionRules="@xml/data_extraction_rules"
        android:fullBackupContent="@xml/backup_rules"
        android:icon="@mipmap/ic_launcher"
        android:label="@string/app_name"
        android:roundIcon="@mipmap/ic_launcher_round"
        android:supportsRtl="true"
        android:theme="@style/Theme.QuizV2"
        tools:targetApi="31">

        <activity
            android:name=".MainActivity"
            android:exported="true"
            android:screenOrientation="portrait">
            <intent-filter>
                <action android:name="android.intent.action.MAIN" />

                <category android:name="android.intent.category.LAUNCHER" />
            </intent-filter>
        </activity>
        <activity
            android:name=".GameLevels"
            android:exported="true"
            android:screenOrientation="portrait">
            <intent-filter>
                <category android:name="android.intent.category.LAUNCHER" />
            </intent-filter>
        </activity>

        <activity
            android:name=".Level1"
            android:exported="true"
            android:screenOrientation="portrait">
            <intent-filter>
                <category android:name="android.intent.category.LAUNCHER" />
            </intent-filter>
        </activity>

        <activity
            android:name=".Level2"
            android:exported="true"
            android:screenOrientation="portrait">
            <intent-filter>
                <category android:name="android.intent.category.LAUNCHER" />
            </intent-filter>
        </activity>
        <activity
            android:name=".Level3"
            android:exported="true"
            android:screenOrientation="portrait">
```

```

        <intent-filter>
            <category android:name="android.intent.category.LAUNCHER" />
        </intent-filter>
    </activity>
    <activity
        android:name=".Level4"
        android:exported="true"
        android:screenOrientation="portrait">
        <intent-filter>
            <category android:name="android.intent.category.LAUNCHER" />
        </intent-filter>
    </activity>
    <activity
        android:name=".Level5"
        android:exported="true"
        android:screenOrientation="portrait">
        <intent-filter>
            <category android:name="android.intent.category.LAUNCHER" />
        </intent-filter>
    </activity>
</application>
</manifest>

```

MainActivity

```

package space.stryuk.quizv2;

import androidx.appcompat.app.AppCompatActivity;

import android.content.Intent;
import android.os.Bundle;
import android.view.View;
import android.view.Window;
import android.view.WindowManager;
import android.widget.Button;
import android.widget.Toast;

public class MainActivity extends AppCompatActivity {
    private long backPressedTime;
    private Toast backToast;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);

        Button buttonStart = (Button) findViewById(R.id.buttonStart);
        buttonStart.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                try {
                    Intent intent = new Intent(MainActivity.this,
GameLevels.class);
                    startActivity(intent); finish();
                } catch (Exception e) {

                } //кінець будування
            }
        });
        Window w = getWindow();

```



```

w.setFlags(WindowManager.LayoutParams.FLAG_FULLSCREEN,WindowManager.LayoutParams
.FLAG_FULLSCREEN);
}
//Система кнопка "Назад" - початок

@Override
public void onBackPressed() {
    super.onBackPressed();

    if (backPressedTime + 2000 > System.currentTimeMillis()) {
        backToast.cancel();
        super.onBackPressed();
        return;
    } else {
        backToast = Toast.makeText(getApplicationContext(),"Нажміть ще раз, щоб
вийти з гри", Toast.LENGTH_SHORT);
        backToast.show();
    }

    backPressedTime = System.currentTimeMillis();
}

//Системна кнопка "Назад" - кінець
}

```

GameLevels

```

package space.stryuk.quizv2;

import android.content.Intent;
import android.content.SharedPreferences;
import android.os.Bundle;
import android.view.View;
import android.view.Window;
import android.view.WindowManager;
import android.widget.Button;
import android.widget.TextView;

import androidx.appcompat.app.AppCompatActivity;

public class GameLevels extends AppCompatActivity {

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.gamelevels);

        SharedPreferences save = getSharedPreferences("Save", MODE_PRIVATE);
        final int level = save.getInt("Level1", 1);

        Window w = getWindow();

w.setFlags(WindowManager.LayoutParams.FLAG_FULLSCREEN,WindowManager.LayoutParams
.FLAG_FULLSCREEN);

        Button button_back = (Button)findViewById(R.id.button_back);
        button_back.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {

```

```

        //команда для програмування кнопки

        //початок будування
        try {
            Intent intent = new Intent(GameLevels.this,
MainActivity.class);
            startActivity(intent);finish();
        } catch (Exception e){
        }
        //кінець будування
    }
});

//Кнопка переходу на 1 рівень - початок
TextView textView1 = (TextView) findViewById(R.id.textView1);
textView1.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {
        try{
            if (level>=1) {
                Intent intent = new Intent(GameLevels.this,
Level1.class);

                startActivity(intent);
                finish();
            }else {
                //порожне
            }
        }catch (Exception e){
            //Порожне
        }
    }
});

//Кнопка переходу на 1 рівень - кінець

//Кнопка переходу на 2 рівень - початок
TextView textView2 = (TextView) findViewById(R.id.textView2);
textView2.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {
        try{
            if (level>=2) {
                Intent intent = new Intent(GameLevels.this,
Level2.class);

                startActivity(intent);
                finish();
            }else {
                //писто
            }
        }catch (Exception e){
            //Порожне
        }
    }
});

//Кнопка переходу на 2 рівень - кінець

//Кнопка переходу на 3 рівень - початок
TextView textView3 = (TextView) findViewById(R.id.textView3);
textView3.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {
        try{
            if (level>=3) {

```

```

        Intent intent = new Intent(GameLevels.this,
Level3.class);
        startActivity(intent);
        finish();
    }else {
        //pysto
    }
}catch (Exception e){
    //Порожне
}
}
});

//Кнопка переходу на 3 рівень - кінець

//Кнопка переходу на 4 рівень - початок
TextView textView4 = (TextView) findViewById(R.id.textView4);
textView4.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {
        try{
            if (level>=4) {
                Intent intent = new Intent(GameLevels.this,
Level4.class);
                startActivity(intent);
                finish();
            }else {
                //pysto
            }
        }catch (Exception e){
            //Порожне
        }
    }
});

//Кнопка переходу на 4 рівень - кінець

//Кнопка переходу на 5 рівень - початок
TextView textView5 = (TextView) findViewById(R.id.textView5);
textView5.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {
        try{
            if (level>=5) {
                Intent intent = new Intent(GameLevels.this,
Level5.class);
                startActivity(intent);
                finish();
            }else {
                //pysto
            }
        }catch (Exception e){
            //Порожне
        }
    }
});

//Кнопка переходу на 5 рівень - кінець
final int[] x={
    R.id.textView1,
    R.id.textView2,
    R.id.textView3,
    R.id.textView4,
    R.id.textView5,

```

```

    };
    for (int i=1; i<level; i++){
        TextView tv = findViewById(x[i]);
        tv.setText(""+(i+1));
    }

}
//Системна кнопка "Назад" - початок
@Override
public void onBackPressed(){
    //початок будування
    try {
        Intent intent = new Intent(GameLevels.this, MainActivity.class);
        startActivity(intent);finish();
    } catch (Exception e){
    }
    //кінець будування
}
//Системна кнопка "Назад" - Кінець
}

```

Array

```

package space.stryuk.quizv2;

public class Array {

    //Масив для першого рівня - початок
    final int[] images1 = {
        R.drawable.first_level_0,
        R.drawable.first_level_1,
        R.drawable.first_level_2,
        R.drawable.first_level_3,
        R.drawable.first_level_4,
        R.drawable.first_level_5,
        R.drawable.first_level_6,
        R.drawable.first_level_7,
        R.drawable.first_level_8,
        R.drawable.first_level_9,
    };

    final int[] power1 = {0,1,2,3,4,5,6,7,8,9,};

    //Масив для першого рівня - кінець

    //Масив для 2 рівня - початок
    final int[] images2 = {
        R.drawable.secondlevel_0,
        R.drawable.secondlevel_1,
        R.drawable.secondlevel_2,
        R.drawable.secondlevel_3,
        R.drawable.secondlevel_4,
        R.drawable.secondlevel_5,
        R.drawable.secondlevel_6,
        R.drawable.secondlevel_7,
        R.drawable.secondlevel_8,
        R.drawable.secondlevel_9,
    };
}

```

```

final int[] power2 = {0,1,2,3,4,5,6,7,8,9,};

//Масив для 2 рівня - кінець

//Масив для 3 рівня - початок
final int[] images3={
R.drawable.chef_1,
R.drawable.job_1,
    R.drawable.chef_2,
    R.drawable.job_2,
    R.drawable.chef_3,
    R.drawable.job_3,
    R.drawable.chef_4,
    R.drawable.job_4,
    R.drawable.chef_5,
    R.drawable.job_5,
    R.drawable.chef_6,
    R.drawable.job_6,
    R.drawable.chef_7,
    R.drawable.job_7,
    R.drawable.chef_8,
    R.drawable.job_8,
    R.drawable.chef_9,
    R.drawable.job_9,
    R.drawable.chef_10,
    R.drawable.job_10,
};

final int[] strong = {
    1,0,1,0,1,0,1,0,1,0,1,0,1,0,1,0,1,0,
};

//Масив для 3 рівня - кінець

//Масив для 4 рівня - початок
final int[] images4={
    R.drawable.footballer_ukr_1,
    R.drawable.enother_footballer_1,
    R.drawable.footballer_ukr_2,
    R.drawable.enother_footballer_2,
    R.drawable.footballer_ukr_3,
    R.drawable.enother_footballer_3,
    R.drawable.footballer_ukr_4,
    R.drawable.enother_footballer_4,
    R.drawable.footballer_ukr_5,
    R.drawable.enother_footballer_5,
    R.drawable.footballer_ukr_6,
    R.drawable.enother_footballer_6,
    R.drawable.footballer_ukr_7,
    R.drawable.enother_footballer_7,
    R.drawable.footballer_ukr_8,
    R.drawable.enother_footballer_8,
    R.drawable.footballer_ukr_9,
    R.drawable.enother_footballer_9,
    R.drawable.footballer_ukr_10,
    R.drawable.enother_footballer_10,
};

final int[] strong2 = {
    1,0,1,0,1,0,1,0,1,0,1,0,1,0,1,0,
};

//Масив для 4 рівня - кінець

```

```

//Масив для 5 рівня - початок
final int[] images5={
    R.drawable.level5_singer_ukr_1,
    R.drawable.singer_enother_1,
    R.drawable.level5_singer_ukr_2,
    R.drawable.singer_enother_2,
    R.drawable.level5_singer_ukr_3,
    R.drawable.singer_enother_3,
    R.drawable.level5_singer_ukr_4,
    R.drawable.singer_enother_4,
    R.drawable.level5_singer_ukr_5,
    R.drawable.singer_enother_5,
    R.drawable.level5_singer_ukr_6,
    R.drawable.singer_enother_6,
    R.drawable.level5_singer_ukr_7,
    R.drawable.singer_enother_7,
    R.drawable.level5_singer_ukr_8,
    R.drawable.singer_enother_8,
    R.drawable.level5_singer_ukr_9,
    R.drawable.singer_enother_9,
    R.drawable.level5_singer_ukr_10,
    R.drawable.singer_enother_10,
};

final int[] strong3 = {
    1,0,1,0,1,0,1,0,1,0,1,0,1,0,1,0,
};

//Масив для 5 рівня - кінець
}

```

Level_1

```

package space.stryuk.quizv2;

import androidx.appcompat.app.AppCompatActivity;

import android.app.Dialog;
import android.content.Intent;
import android.content.SharedPreferences;
import android.graphics.Color;
import android.graphics.drawable.ColorDrawable;
import android.os.Bundle;
import android.view.MotionEvent;
import android.view.View;
import android.view.Window;
import android.view.WindowManager;
import android.view.animation.Animation;
import android.view.animation.AnimationUtils;
import android.widget.Button;
import android.widget.ImageView;
import android.widget.TextView;

import java.util.Random;

public class Level1 extends AppCompatActivity {

    Dialog dialog;
    Dialog dialogEnd;

    public int numLeft; //Для лівого зображення
    public int numRight; //Для правого зображення
    Array array = new Array();

```

```

Random random = new Random(); //Генерація чисел
public int count = 0; //Правельні відповіді

@Override
protected void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
    setContentView(R.layout.universal);

    //Робимо text_levels
    TextView text_levels = findViewById(R.id.text_levels);
    text_levels.setText(R.string.level_one); //Встановили текст.

    final ImageView img_left = (ImageView) findViewById(R.id.img_left);
    //код Округлення кутів лівого зображення
    img_left.setClipToOutline(true);

    final ImageView img_rihgt = (ImageView) findViewById(R.id.img_right);
    //код Округлення кутів правого зображення
    img_rihgt.setClipToOutline(true);

    //Путь до лівою Textview

    //Розгорнути гру у весь екран - початок
    Window w = getWindow();
    w.setFlags(WindowManager.LayoutParams.FLAG_LAYOUT_NO_LIMITS,
WindowManager.LayoutParams.FLAG_LAYOUT_NO_LIMITS);
    //Розгорнути гру у весь екран - Кінець

    //Виклик діалогового вікна на початку рівня
    dialog = new Dialog(this); //Робимо вікно діалога
    dialog.requestWindowFeature(Window.FEATURE_NO_TITLE); //забираємо
заголовог
    dialog.setContentView(R.layout.previewdialog); //путь до діалогового
вікна
    dialog.getWindow().setBackgroundDrawable(new
ColorDrawable(Color.TRANSPARENT)); //прозорий фон вікна
    dialog.setCancelable(false); //вікно не можна зачинити кнопкою "назад"

    //Кнопка для закриття вікна діалогу - початок
    TextView buttonclose = (TextView) dialog.findViewById(R.id.buttonclose);
    buttonclose.setOnClickListener(new View.OnClickListener() {
        @Override
        public void onClick(View v) {
            //натискання кнопки - початок
            try {
                //повернутись до рівня - початок
                Intent intent = new Intent(Level1.this,
GameLevels.class); //перехід
                startActivity(intent);
                finish(); // закрити вікно
                //повернутись до рівня - кінець
            } catch (Exception e) {
                //У цьому місті не має коду. А ну геть!
            }
            dialog.dismiss(); //Закриття діалогового вікна

            //натискання кнопки - кінець
        }
    });
    //Кнопка для закриття вікна діалогу - кінець

```

```

        //Кнопка "Далі" - початок
        Button buttoncontinue =
        (Button) dialog.findViewById(R.id.buttoncontinue);
        buttoncontinue.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                dialog.dismiss(); //Закрити діалогове вікно
            }
        });

        //Кнопка "Далі" - кінець

        dialog.show(); //показати діалогове вікно

//
//
//Виклик діалогового вікна у кінці рівня
dialogEnd = new Dialog(this); //Робимо вікно діалога
dialogEnd.requestWindowFeature(Window.FEATURE_NO_TITLE); //забираємо
заголовок
dialogEnd.setContentView(R.layout.dialog_end); //путь до діалогового
вікна
dialogEnd.getWindow().setBackgroundDrawable(new
ColorDrawable(Color.TRANSPARENT)); //прозорий фон вікна
dialogEnd.getWindow().setLayout(WindowManager.LayoutParams.MATCH_PARENT,
WindowManager.LayoutParams.MATCH_PARENT);
dialogEnd.setCancelable(false); //вікно не можна зачинити кнопкою "назад"

//Кнопка для закриття вікна діалогу - початок
TextView buttonclose2 =
(TextView) dialogEnd.findViewById(R.id.buttonclose);
buttonclose2.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {
        //натискання кнопки - початок
        try {
            //повернутись до рівня - початок
            Intent intent = new Intent(Level1.this,
GameLevels.class); //перехід
            startActivity(intent);
            finish(); // закрити вікно
            //повернутись до рівня - кінець
        } catch (Exception e) {
            //У цьому місті не має коду. А ну геть!
        }
        dialogEnd.dismiss(); //Закриття діалогового вікна

        //натискання кнопки - кінець

    }
});
//Кнопка для закриття вікна діалогу - кінець

//Кнопка "Далі" - початок
Button buttoncontinue2 =
(Button) dialogEnd.findViewById(R.id.buttoncontinue);
buttoncontinue2.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {
        try {
            Intent intent = new Intent(Level1.this, Level2.class);
            startActivity(intent);
        }
    }
});

```



```

        finish();
    }catch(Exception e){
        //Кода не має
    }
    dialogEnd.dismiss();//Закрити діалогове вікно
}
});

//Кнопка "Далі" - кінець

//


---


//Кнопка назад - Початок
Button btn_back = (Button)findViewById(R.id.button_back_level1);
btn_back.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {
        //Натискання кнопки "Назад" - початок
        try {
            //Повернутись до вибору рівня - початок
            Intent intent = new Intent(Level1.this,
GameLevels.class);//Перехід
            startActivity(intent);
            finish();
            //Повернутись до вибору рівня - кінець
        }catch (Exception e){
            //Я то кажу, що кудо не має. Що ти від мене хочеш?
        }
        //Натискання кнопки "Назад" - кінець
    }
});

//Кнопка назад - Кінець

//Масив для прогресу гри - початок
final int[] progress = {
R.id.point1,R.id.point2,R.id.point3,R.id.point4,R.id.point5,R.id.point6,
    R.id.point7,R.id.point8,R.id.point9,R.id.point10,
};
//Масив для прогресу гри - початок

//Застосування анімації - початок
final Animation a =
AnimationUtils.loadAnimation(Level1.this,R.anim.alpha);
//Застосування анімації - кінець

numLeft = random.nextInt(10);
img_left.setImageResource(array.images1[numLeft]); // З масиву зображення

numRight = random.nextInt(10);

//Цикл перевіряючий рівність чисел - початок
while (numLeft==numRight){
    numRight = random.nextInt(10);
}
//Цикл перевіряючий рівність чисел - кінець

```

```

img_rihgt.setImageResource(array.images1[numRight]); // З масиву
зображення

//Обробна натискання на ліве зображення - початок
img_left.setOnTouchListener(new View.OnTouchListener() {
    @Override
    public boolean onTouch(View v, MotionEvent event) {
        //Натискання зображення - початок
        if (event.getAction()==MotionEvent.ACTION_DOWN) {
            //Якщо дотокнувся до зображення - початок
            img_rihgt.setEnabled(false); //Блокування правого зображення
            if (numLeft>numRight) {
                img_left.setImageResource(R.drawable.img_true);
            }else {
                img_left.setImageResource(R.drawable.img_false);
            }
            //Якщо дотокнувся до зображення - кінець
        }else if (event.getAction()==MotionEvent.ACTION_UP) {
            //Якщо дотокнувся до зображення - початок
            if (numLeft>numRight) {
                //Якщо ліве зображення більше
                if (count<10) {
                    count=count+1;
                }
                //Фарбуємо прогрес сірим кольором - початок
                for (int i=0; i<10; i++) {
                    TextView tv = findViewById(progress[i]);
                    tv.setBackgroundResource(R.drawable.style_points);
                }
                //Фарбуємо прогрес сірим кольором - кінець
                //Правельні відповіді фарбуємо зеленим - початок
                for (int i=0; i<count; i++) {
                    TextView tv = findViewById(progress[i]);
                    tv.setBackgroundResource(R.drawable.style_points_green);
                }
                //Правельні відповіді фарбуємо зеленим - кінець
            }else {
                //Якщо ліве зображення менше
                if (count>0) {
                    if (count==1) {
                        count=0;
                    }else {
                        count=count-2;
                    }
                }
                //Фарбуємо прогрес сірим кольором - початок
                for (int i=0; i<9; i++) {
                    TextView tv = findViewById(progress[i]);
                    tv.setBackgroundResource(R.drawable.style_points);
                }
                //Фарбуємо прогрес сірим кольором - кінець
                //Правельні відповіді фарбуємо зеленим - початок
                for (int i=0; i<count; i++) {
                    TextView tv = findViewById(progress[i]);
                    tv.setBackgroundResource(R.drawable.style_points_green);
                }
                //Правельні відповіді фарбуємо зеленим - кінець
            }
            //Якщо доторкнувся до зображення - кінець
            if (count==10) {

```

```

        //ВИХІД З РІВНЯ
        SharedPreferences save = getSharedPreferences("Save",
MODE_PRIVATE);

        final int level = save.getInt("Level1", 1);
        if (level>1){
            //порожне
        }else {
            SharedPreferences.Editor editor = save.edit();
            editor.putInt("Level1", 2);
            editor.commit();
        }
        dialogEnd.show();
    }else {
        numLeft = random.nextInt(10);
        img_left.setImageResource(array.images1[numLeft]); // 3
масиву зображення

        img_left.startAnimation(a);

        numRight = random.nextInt(10);

        //Цикл перевіряючий рівність чисел - початок
        while (numLeft==numRight){
            numRight = random.nextInt(10);
        }
        //Цикл перевіряючий рівність чисел - кінець

        img_rihgt.setImageResource(array.images1[numRight]); // 3
масиву зображення

        img_rihgt.startAnimation(a);
        img_rihgt.setEnabled(true); //Включаем обратно правую
картинку

    }
}
//Натискання зображення - кінець
return true;
}
});
//Обробна натискання на ліве зображення - кінець

//Обробна натискання на праве зображення - початок
img_rihgt.setOnTouchListener(new View.OnTouchListener() {
    @Override
    public boolean onTouch(View v, MotionEvent event) {
        //Натискання зображення - початок
        if (event.getAction()==MotionEvent.ACTION_DOWN){
            //Якщо дотокнувся до зображення - початок
            img_left.setEnabled(false); //Блокування ліве зображення
            if (numLeft<numRight){
                img_rihgt.setImageResource(R.drawable.img_true);
            }else {
                img_rihgt.setImageResource(R.drawable.img_false);
            }
            //Якщо дотокнувся до зображення - кінець
        }else if (event.getAction()==MotionEvent.ACTION_UP){
            //Якщо дотокнувся до зображення - початок
            if (numLeft<numRight){
                //Якщо праве зображення більше
                if (count<10){
                    count=count+1;
                }
                //Фарбуємо прогрес сірим кольором - початок
                for (int i=0; i<10; i++){
                    TextView tv = findViewById(progress[i]);

```

```

        tv.setBackgroundResource(R.drawable.style_points);
    }
    //Фарбуємо прогрес сірим кольором - кінець
    //Правельні відповіді фарбуємо зеленим - початок
    for (int i=0; i<count; i++){
        TextView tv = findViewById(progress[i]);

tv.setBackgroundResource(R.drawable.style_points_green);
    }
    //Правельні відповіді фарбуємо зеленим - кінець
}else {
    //Якщо праве зображення менше
    if (count>0){
        if (count==1){
            count=0;
        }else {
            count=count-2;
        }
    }
    //Фарбуємо прогрес сірим кольором - початок
    for (int i=0; i<9; i++){
        TextView tv = findViewById(progress[i]);
        tv.setBackgroundResource(R.drawable.style_points);
    }
    //Фарбуємо прогрес сірим кольором - кінець
    //Правельні відповіді фарбуємо зеленим - початок
    for (int i=0; i<count; i++){
        TextView tv = findViewById(progress[i]);

tv.setBackgroundResource(R.drawable.style_points_green);
    }
    //Правельні відповіді фарбуємо зеленим - кінець
}
//Якщо доторкнувся до зображення - кінець
if (count==10){
    //ВИХІД З РІВНЯ
    SharedPreferences save = getSharedPreferences("Save",
MODE_PRIVATE);

    final int level = save.getInt("Level1", 1);
    if (level>1){
        //порожнє
    }else {
        SharedPreferences.Editor editor = save.edit();
        editor.putInt("Level1", 2);
        editor.commit();
    }
    dialogEnd.show();
}else {
    numLeft = random.nextInt(10);
    img_left.setImageResource(array.images1[numLeft]); // 3
    масиву зображення

    img_left.startAnimation(a);

    numRight = random.nextInt(10);

    //Цикл перевіряючий рівність чисел - початок
    while (numLeft==numRight){
        numRight = random.nextInt(10);
    }
    //Цикл перевіряючий рівність чисел - кінець

    img_rihgt.setImageResource(array.images1[numRight]); // 3
    масиву зображення

    img_rihgt.startAnimation(a);

```

```

        img_left.setEnabled(true); //Включаем обратно ліву
картинку
    }
    }
    //Натискання зображення - кінець
    return true;
}
});
//Обробна натискання на праве зображення - кінець

}
//Системна кнопка "Назад" - початок
@Override
public void onBackPressed(){
    //Натискання кнопки "Назад" - початок
    try {
        //Повернутись до вибору рівня - початок
        Intent intent = new Intent(Level1.this, GameLevels.class); //Перехід
        startActivity(intent);
        finish();
        //Повернутись до вибору рівня - кінець
    } catch (Exception e){
        //Я то кажу, що кудо не має. Що ти від мене хочеш?
    }
    //Натискання кнопки "Назад" - кінець
}
//Системна кнопка "Назад" - кінець
}
}

```

Level_2

```

package space.stryuk.quizv2;

import android.app.Dialog;
import android.content.Intent;
import android.content.SharedPreferences;
import android.graphics.Color;
import android.graphics.drawable.ColorDrawable;
import android.os.Bundle;
import android.view.MotionEvent;
import android.view.View;
import android.view.Window;
import android.view.WindowManager;
import android.view.animation.Animation;
import android.view.animation.AnimationUtils;
import android.widget.Button;
import android.widget.ImageView;
import android.widget.LinearLayout;
import android.widget.TextView;

import androidx.appcompat.app.AppCompatActivity;

import java.util.Random;

public class Level2 extends AppCompatActivity {

    Dialog dialog;
    Dialog dialogEnd;
}

```

```

public int numLeft; //Для лівого зображення
public int numRight; //Для правого зображення
Array array = new Array();
Random random = new Random(); //Генерація чисел
public int count = 0; //Правельні відповіді

@Override
protected void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
    setContentView(R.layout.universal);

    //Робимо text_levels
    TextView text_levels = findViewById(R.id.text_levels);
    text_levels.setText(R.string.level_two); //Встановили текст.

    final ImageView img_left = (ImageView)findViewById(R.id.img_left);
    //код Округлення кутів лівого зображення
    img_left.setClipToOutline(true);

    final ImageView img_rihgt = (ImageView)findViewById(R.id.img_right);
    //код Округлення кутів правого зображення
    img_rihgt.setClipToOutline(true);

    //Путь до лівою Textview

    //Розгорнути гру у весь екран - початок
    Window w = getWindow();
    w.setFlags(WindowManager.LayoutParams.FLAG_LAYOUT_NO_LIMITS,
WindowManager.LayoutParams.FLAG_LAYOUT_NO_LIMITS);
    //Розгорнути гру у весь екран - Кінець

    //Встановлюємо фон - початок
    ImageView background = (ImageView)findViewById(R.id.background);
    background.setImageResource(R.drawable.background_level_2);
    //Встановлюємо фон - початок

    //Виклик діалогового вікна на початку рівня
    dialog = new Dialog(this); //Робимо вікно діолога
    dialog.requestWindowFeature(Window.FEATURE_NO_TITLE); //забираємо
заголовоч
    dialog.setContentView(R.layout.previewdialog); //путь до діалогового
вікна
    dialog.getWindow().setBackgroundDrawable(new
ColorDrawable(Color.TRANSPARENT)); //прозорий фон вікна
    dialog.setCancelable(false); //вікно не можна зачинити кнопкою "назад"

    //Встановимо зображення в діалогове віко - початок
    ImageView previewimg = (ImageView)dialog.findViewById(R.id.previewimg);
    previewimg.setImageResource(R.drawable.preview_secondlevel_img1);
    //Встановимо зображення в діалогове віко - кінець

    //встановлюємо фон діалогового вікна - початок
    LinearLayout dialogdon =
(LinearLayout)dialog.findViewById(R.id.dialogdon);
    dialogdon.setBackgroundResource(R.drawable.second_level_preview);
    //встановлюємо фон діалогового вікна - кінець

    //Встановлюємо опис завдання - початок
    TextView textdescription =
(TextView)dialog.findViewById(R.id.textdescription);
    textdescription.setText(R.string.leveltwo);

```

```

//Встановлюємо опис завдання - кінець

//Кнопка для закриття вікна діалогу - початок
TextView buttonclose = (TextView)dialog.findViewById(R.id.buttonclose);
buttonclose.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {
        //натискання кнопки - початок
        try {
            //повернутись до рівня - початок
            Intent intent = new Intent(Level2.this,
GameLevels.class); //перехід
            startActivity(intent);
            finish(); // закрити вікно
            //повернутись до рівня - кінець
        } catch (Exception e) {
            //У цьому місті не має коду. А ну геть!
        }
        dialog.dismiss(); //Закриття діалогового вікна

        //натискання кнопки - кінець
    }
});
//Кнопка для закриття вікна діалогу - кінець

//Кнопка "Далі" - початок
Button buttoncontinue =
(Button)dialog.findViewById(R.id.buttoncontinue);
buttoncontinue.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {
        dialog.dismiss(); //Закрити діалогове вікно
    }
});

//Кнопка "Далі" - кінець

dialog.show(); //показати діалогове вікно

//
-----
//Виклик діалогового вікна у кінці рівня
dialogEnd = new Dialog(this); //Робимо вікно діалога
dialogEnd.requestWindowFeature(Window.FEATURE_NO_TITLE); //забираємо
заголовок
dialogEnd.setContentView(R.layout.dialog_end); //путь до діалогового
вікна
dialogEnd.getWindow().setBackgroundDrawable(new
ColorDrawable(Color.TRANSPARENT)); //прозорий фон вікна
dialogEnd.getWindow().setLayout(WindowManager.LayoutParams.MATCH_PARENT,
WindowManager.LayoutParams.MATCH_PARENT);
dialogEnd.setCancelable(false); //вікно не можна зачинити кнопкою "назад"

//Встановлюємо фон діалогово вікна - початок
LinearLayout dialogfonEnd =
(LinearLayout)dialogEnd.findViewById(R.id.dialogfon);
dialogfonEnd.setBackgroundResource(R.drawable.second_level_preview);
//Встановлюємо фон діалогово вікна - кінець

//цікавий факт - початок

```

```

        TextView textdescriptionEnd =
        (TextView) dialogEnd.findViewById(R.id.textdescriptionEnd);
        textdescriptionEnd.setText(R.string.leveltwoEnd);

        //цікавий факт - кінець

        //Кнопка для закриття вікна діалогу - початок
        TextView buttonclose2 =
        (TextView) dialogEnd.findViewById(R.id.buttonclose);
        buttonclose2.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                //натискання кнопки - початок
                try {
                    //повернутись до рівня - початок
                    Intent intent = new Intent(Level2.this,
GameLevels.class); //перехід
                    startActivity(intent);
                    finish(); // закрити вікно
                    //повернутись до рівня - кінець
                } catch (Exception e) {
                    //У цьому місті не має коду. А ну геть!
                }
                dialogEnd.dismiss(); //Закриття діалогового вікна

                //натискання кнопки - кінець
            }
        });
        //Кнопка для закриття вікна діалогу - кінець

        //Кнопка "Далі" - початок
        Button buttoncontinue2 =
        (Button) dialogEnd.findViewById(R.id.buttoncontinue);
        buttoncontinue2.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                try {
                    Intent intent = new Intent(Level2.this, Level3.class);
                    startActivity(intent);
                    finish();
                } catch (Exception e) {
                    //Кода не має
                }
                dialogEnd.dismiss(); //Закрити діалогове вікно
            }
        });

        //Кнопка "Далі" - кінець

//


---


        //Кнопка назад - Початок
        Button btn_back = (Button) findViewById(R.id.button_back_level1);
        btn_back.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                //Натискання кнопки "Назад" - початок
                try {
                    //Повернутись до вибору рівня - початок
                    Intent intent = new Intent(Level2.this,

```



```

GameLevels.class);//Перехід
    startActivity(intent);
    finish();
    //Повернутись до вибору рівня - кінець
}catch (Exception e){
    //Я то кажу, що куди не має. Що ти від мене хочеш?
}
//Натискання кнопки "Назад" - кінець

}
});

//Кнопка назад - Кінець

//Масив для прогресу гри - початок
final int[] progress = {
R.id.point1,R.id.point2,R.id.point3,R.id.point4,R.id.point5,R.id.point6,
    R.id.point7,R.id.point8,R.id.point9,R.id.point10,
};
//Масив для прогресу гри - початок

//Застосування анімації - початок
final Animation a =
AnimationUtils.loadAnimation(Level2.this,R.anim.alpha);
//Застосування анімації - кінець

numLeft = random.nextInt(10);
img_left.setImageResource(array.images2[numLeft]);// З масиву зображення

numRight = random.nextInt(10);

//Цикл перевіряючий рівність чисел - початок
while (numLeft==numRight){
    numRight = random.nextInt(10);
}
//Цикл перевіряючий рівність чисел - кінець

img_rihgt.setImageResource(array.images2[numRight]);// З масиву зображення

//Обробна натискання на ліве зображення - початок
img_left.setOnTouchListener(new View.OnTouchListener() {
    @Override
    public boolean onTouch(View v, MotionEvent event) {
        //Натискання зображення - початок
        if (event.getAction()==MotionEvent.ACTION_DOWN){
            //Якщо доткнувся до зображення - початок
            img_rihgt.setEnabled(false);//Блокування правого зображення
            if (numLeft>numRight){
                img_left.setImageResource(R.drawable.img_true);
            }else {
                img_left.setImageResource(R.drawable.img_false);
            }
            //Якщо доткнувся до зображення - кінець
        }else if (event.getAction()==MotionEvent.ACTION_UP){
            //Якщо доткнувся до зображення - початок
            if (numLeft>numRight){
                //Якщо ліве зображення більше
                if (count<10){

```

```

        count=count+1;
    }
    //Фарбуємо прогрес сірим кольором - початок
    for (int i=0; i<10; i++){
        TextView tv = findViewById(progress[i]);
        tv.setBackgroundResource(R.drawable.style_points);
    }
    //Фарбуємо прогрес сірим кольором - кінець
    //Правельні відповіді фарбуємо зеленим - початок
    for (int i=0; i<count; i++){
        TextView tv = findViewById(progress[i]);

tv.setBackgroundResource(R.drawable.style_points_green);
    }
    //Правельні відповіді фарбуємо зеленим - кінець
}else {
    //Якщо ліве зображення менше
    if (count>0){
        if (count==1){
            count=0;
        }else {
            count=count-2;
        }
    }
    //Фарбуємо прогрес сірим кольором - початок
    for (int i=0; i<9; i++){
        TextView tv = findViewById(progress[i]);
        tv.setBackgroundResource(R.drawable.style_points);
    }
    //Фарбуємо прогрес сірим кольором - кінець
    //Правельні відповіді фарбуємо зеленим - початок
    for (int i=0; i<count; i++){
        TextView tv = findViewById(progress[i]);

tv.setBackgroundResource(R.drawable.style_points_green);
    }
    //Правельні відповіді фарбуємо зеленим - кінець
}
//Якщо доторкнувся до зображення - кінець
if (count==10){
    //ВИХІД З РІВНЯ
    SharedPreferences save = getSharedPreferences("Save",
MODE_PRIVATE);

    final int level = save.getInt("Level1", 1);
    if (level>2){
        //порожнє
    }else {
        SharedPreferences.Editor editor = save.edit();
        editor.putInt("Level1", 3);
        editor.commit();
    }
    dialogEnd.show();
}else {
    numLeft = random.nextInt(10);
    img_left.setImageResource(array.images2[numLeft]); // 3
    масиву зображення
    img_left.startAnimation(a);

    numRight = random.nextInt(10);

    //Цикл перевіряючий рівність чисел - початок
    while (numLeft==numRight){
        numRight = random.nextInt(10);
    }
}

```

```

//Цикс перевіряючий рівність чисел - кінець
img_rihgt.setImageResource(array.images2[numRight]); // 3
масиву зображення
img_rihgt.startAnimation(a);
картинку
img_rihgt.setEnabled(true); //Включаем обратно правую

    }
}
//Натискання зображення - кінець
return true;
}
});
//Обробна натискання на ліве зображення - кінець

//Обробна натискання на праве зображення - початок
img_rihgt.setOnTouchListener(new View.OnTouchListener() {
    @Override
    public boolean onTouch(View v, MotionEvent event) {
        //Натискання зображення - початок
        if (event.getAction()==MotionEvent.ACTION_DOWN) {
            //Якщо дотокнувся до зображення - початок
            img_left.setEnabled(false); //Блокування ліве зображення
            if (numLeft<numRight) {
                img_rihgt.setImageResource(R.drawable.img_true);
            }else {
                img_rihgt.setImageResource(R.drawable.img_false);
            }
            //Якщо дотокнувся до зображення - кінець
        }else if (event.getAction()==MotionEvent.ACTION_UP) {
            //Якщо дотокнувся до зображення - початок
            if (numLeft<numRight) {
                //Якщо праве зображення більше
                if (count<10) {
                    count=count+1;
                }
                //Фарбуємо прогрес сірим кольором - початок
                for (int i=0; i<10; i++){
                    TextView tv = findViewById(progress[i]);
                    tv.setBackgroundResource(R.drawable.style_points);
                }
                //Фарбуємо прогрес сірим кольором - кінець
                //Правельні відповіді фарбуємо зеленим - початок
                for (int i=0; i<count; i++){
                    TextView tv = findViewById(progress[i]);
                    tv.setBackgroundResource(R.drawable.style_points_green);
                }
                //Правельні відповіді фарбуємо зеленим - кінець
            }else {
                //Якщо праве зображення менше
                if (count>0) {
                    if (count==1) {
                        count=0;
                    }else {
                        count=count-2;
                    }
                }
                //Фарбуємо прогрес сірим кольором - початок
                for (int i=0; i<9; i++){
                    TextView tv = findViewById(progress[i]);
                    tv.setBackgroundResource(R.drawable.style_points);
                }
            }
        }
    }
});

```

```

        //Фарбуємо прогрес сірим кольором - кінець
        //Правельні відповіді фарбуємо зеленим - початок
        for (int i=0; i<count; i++){
            TextView tv = findViewById(progress[i]);

tv.setBackgroundResource(R.drawable.style_points_green);
        }
        //Правельні відповіді фарбуємо зеленим - кінець
    }
    //Якщо доторкнувся до зображення - кінець
    if (count==10){
        //ВИХІД З РІВНЯ
        SharedPreferences save = getSharedPreferences("Save",
MODE_PRIVATE);

        final int level = save.getInt("Level1", 1);
        if (level>2){
            //порожне
        }else {
            SharedPreferences.Editor editor = save.edit();
            editor.putInt("Level1", 3);
            editor.commit();
        }
        dialogEnd.show();
    }else {
        numLeft = random.nextInt(10);
        img_left.setImageResource(array.images2[numLeft]); // з
масиву зображення

        img_left.startAnimation(a);

        numRight = random.nextInt(10);

        //Цикл перевіряючий рівність чисел - початок
        while (numLeft==numRight){
            numRight = random.nextInt(10);
        }
        //Цикл перевіряючий рівність чисел - кінець

        img_rihgt.setImageResource(array.images2[numRight]); // з
масиву зображення

        img_rihgt.startAnimation(a);
        img_left.setEnabled(true); //Включаем обратно ліву
картинку

    }
    }
    //Натискання зображення - кінець
    return true;
}
});
//Обробна натискання на праве зображення - кінець

}
//Системна кнопка "Назад" - початок
@Override
public void onBackPressed(){
    //Натискання кнопки "Назад" - початок
    try {
        //Повернутись до вибору рівня - початок
        Intent intent = new Intent(Level2.this, GameLevels.class); //Перехід
startActivity(intent);
finish();
        //Повернутись до вибору рівня - кінець
    }
}

```

```

    }catch (Exception e){
        //Я то кажу, що кудо не має. Що ти від мене хочеш?
    }
    //Натискання кнопки "Назад" - кінець
}
//Системна кнопка "Назад" - кінець
}

```

Level_3

```

package space.stryuk.quizv2;

import android.app.Dialog;
import android.content.Intent;
import android.content.SharedPreferences;
import android.graphics.Color;
import android.graphics.drawable.ColorDrawable;
import android.os.Bundle;
import android.view.MotionEvent;
import android.view.View;
import android.view.Window;
import android.view.WindowManager;
import android.view.animation.Animation;
import android.view.animation.AnimationUtils;
import android.widget.Button;
import android.widget.ImageView;
import android.widget.LinearLayout;
import android.widget.TextView;

import androidx.appcompat.app.AppCompatActivity;

import java.util.Random;

public class Level3 extends AppCompatActivity {

    Dialog dialog;
    Dialog dialogEnd;

    public int numLeft; //Для лівого зображення
    public int numRight; //Для правого зображення
    Array array = new Array();
    Random random = new Random(); //Генерація чисел
    public int count = 0; //Правельні відповіді

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.universal);

        //Робимо text_levels
        TextView text_levels = findViewById(R.id.text_levels);
        text_levels.setText(R.string.level_three); //Встановили текст.

        final ImageView img_left = (ImageView) findViewById(R.id.img_left);
        //код Округлення кутів лівого зображення
        img_left.setClipToOutline(true);

        final ImageView img_rihgt = (ImageView) findViewById(R.id.img_right);
        //код Округлення кутів правого зображення
        img_rihgt.setClipToOutline(true);
    }
}

```

```

//Путь до лівою Textview

//Розгорнути гру у весь екран - початок
Window w = getWindow();
w.setFlags(WindowManager.LayoutParams.FLAG_LAYOUT_NO_LIMITS,
WindowManager.LayoutParams.FLAG_LAYOUT_NO_LIMITS);
//Розгорнути гру у весь екран - кінець

//Встановлюємо фон - початок
ImageView background = (ImageView) findViewById(R.id.background);
background.setImageResource(R.drawable.new_background_level3_version2);
//Встановлюємо фон - початок

//Виклик діалогового вікна на початку рівня
dialog = new Dialog(this); //Робимо вікно діалога
dialog.requestWindowFeature(Window.FEATURE_NO_TITLE); //забираємо
заголовок
dialog setContentView(R.layout.previewdialog); //путь до діалогового
вікна
dialog.getWindow().setBackgroundDrawable(new
ColorDrawable(Color.TRANSPARENT)); //прозорий фон вікна
dialog.setCancelable(false); //вікно не можна зачинити кнопкою "назад"

//Встановимо зображення в діалогове вікно - початок
ImageView previewimg = (ImageView) dialog.findViewById(R.id.previewimg);
previewimg.setImageResource(R.drawable.preview_level3_img1);
//Встановимо зображення в діалогове вікно - кінець

//встановлюємо фон діалогового вікна - початок
LinearLayout dialogdon =
(LinearLayout) dialog.findViewById(R.id.dialogdon);
dialogdon.setBackgroundResource(R.drawable.new_preview_level3_v2);
//встановлюємо фон діалогового вікна - кінець

//Встановлюємо опис завдання - початок
TextView textdescription =
(TextView) dialog.findViewById(R.id.textdescription);
textdescription.setText(R.string.levelthree);
//Встановлюємо опис завдання - кінець

//Кнопка для закриття вікна діалогу - початок
TextView buttonclose = (TextView) dialog.findViewById(R.id.buttonclose);
buttonclose.setOnClickListener(new View.OnClickListener() {
@Override
public void onClick(View v) {
//натискання кнопки - початок
try {
//повернутись до рівня - початок
Intent intent = new Intent(Level3.this,
GameLevels.class); //перехід
startActivity(intent);
finish(); // закрити вікно
//повернутись до рівня - кінець
}catch (Exception e){
//У цьому місті не має коду. А ну геть!
}
dialog.dismiss(); //Закриття діалогового вікна

//натискання кнопки - кінець
}
}
}

```

```

});
//Кнопка для закриття вікна діалогу - кінець

//Кнопка "Далі" - початок
Button buttoncontinue =
(Button) dialog.findViewById(R.id.buttoncontinue);
buttoncontinue.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {
        dialog.dismiss(); //Закрити діалогове вікно
    }
});

//Кнопка "Далі" - кінець

dialog.show(); //показати діалогове вікно

//
//Виклик діалогового вікна у кінці рівня
dialogEnd = new Dialog(this); //Робимо вікно діалога
dialogEnd.requestWindowFeature(Window.FEATURE_NO_TITLE); //забираємо
заголовок
dialogEnd.setContentView(R.layout.dialog_end); //путь до діалогового
вікна
dialogEnd.getWindow().setBackgroundDrawable(new
ColorDrawable(Color.TRANSPARENT)); //прозорий фон вікна
dialogEnd.getWindow().setLayout(WindowManager.LayoutParams.MATCH_PARENT,
WindowManager.LayoutParams.MATCH_PARENT);
dialogEnd.setCancelable(false); //вікно не можна зачинити кнопкою "назад"

//Встановлюємо фон діалогово вікна - початок
LinearLayout dialogfonEnd =
(LinearLayout) dialogEnd.findViewById(R.id.dialogfon);
dialogfonEnd.setBackgroundResource(R.drawable.new_preview_level3_v2);
//Встановлюємо фон діалогово вікна - кінець

//цікавий факт - початок
TextView textdescriptionEnd =
(TextView) dialogEnd.findViewById(R.id.textdescriptionEnd);
textdescriptionEnd.setText(R.string.levelthreeEnd);

//цікавий факт - кінець

//Кнопка для закриття вікна діалогу - початок
TextView buttonclose2 =
(TextView) dialogEnd.findViewById(R.id.buttonclose);
buttonclose2.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {
        //натискання кнопки - початок
        try {
            //повернутись до рівня - початок
            Intent intent = new Intent(Level3.this,
GameLevels.class); //перехід
            startActivity(intent);
            finish(); // закрити вікно
            //повернутись до рівня - кінець
        } catch (Exception e) {
            //У цьому місті не має коду. А ну геть!
        }
    }
});

```

```

        dialogEnd.dismiss(); //Закриття діалогового вікна

        //натискання кнопки - кінець
    }
});
//Кнопка для закриття вікна діалогу - кінець

//Кнопка "Далі" - початок
Button buttoncontinue2 =
(Button) dialogEnd.findViewById(R.id.buttoncontinue);
buttoncontinue2.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {
        try {
            Intent intent = new Intent(Level3.this, Level4.class);
            startActivity(intent);
            finish();
        } catch (Exception e) {
            //Кода не має
        }
        dialogEnd.dismiss(); //Закрити діалогове вікно
    }
});

//Кнопка "Далі" - кінець

//


---


//Кнопка назад - Початок
Button btn_back = (Button) findViewById(R.id.button_back_level1);
btn_back.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {
        //Натискання кнопки "Назад" - початок
        try {
            //Повернутись до вибору рівня - початок
            Intent intent = new Intent(Level3.this,
GameLevels.class); //Перехід
            startActivity(intent);
            finish();
            //Повернутись до вибору рівня - кінець
        } catch (Exception e) {
            //Я то кажу, що кудо не має. Що ти від мене хочеш?
        }
        //Натискання кнопки "Назад" - кінець
    }
});

//Кнопка назад - Кінець

//Масив для прогресу гри - початок
final int[] progress = {
R.id.point1, R.id.point2, R.id.point3, R.id.point4, R.id.point5, R.id.point6,
R.id.point7, R.id.point8, R.id.point9, R.id.point10,
};
//Масив для прогресу гри - початок

```



```

//Застосування анімації - початок
final Animation a =
AnimationUtils.loadAnimation(Level3.this,R.anim.alpha);
//Застосування анімації - кінець

numLeft = random.nextInt(20);//Гинеруємо число
img_left.setImageResource(array.images3[numLeft]);// З масиву зображення

numRight = random.nextInt(20);

//Цикл перевіряючий рівність чисел - початок
while (array.strong[numLeft]==array.strong[numRight]){
    numRight = random.nextInt(20);
}
//Цикл перевіряючий рівність чисел - кінець

img_rihgt.setImageResource(array.images3[numRight]);// З масиву зображення

//Обробна натискання на ліве зображення - початок
img_left.setOnTouchListener(new View.OnTouchListener() {
    @Override
    public boolean onTouch(View v, MotionEvent event) {
        //Натискання зображення - початок
        if (event.getAction()==MotionEvent.ACTION_DOWN){
            //Якщо доткнувся до зображення - початок
            img_rihgt.setEnabled(false);//Блокування правого зображення
            if (array.strong[numLeft]>array.strong[numRight]){
                img_left.setImageResource(R.drawable.img_true);
            }else {
                img_left.setImageResource(R.drawable.img_false);
            }
            //Якщо доткнувся до зображення - кінець
        }else if (event.getAction()==MotionEvent.ACTION_UP){
            //Якщо доткнувся до зображення - початок
            if (array.strong[numLeft]>array.strong[numRight]){
                //Якщо ліве зображення більше
                if (count<10){
                    count=count+1;
                }
                //Фарбуємо прогрес сірим кольором - початок
                for (int i=0; i<10; i++){
                    TextView tv = findViewById(progress[i]);
                    tv.setBackgroundResource(R.drawable.style_points);
                }
                //Фарбуємо прогрес сірим кольором - кінець
                //Правельні відповіді фарбуємо зеленим - початок
                for (int i=0; i<count; i++){
                    TextView tv = findViewById(progress[i]);
                    tv.setBackgroundResource(R.drawable.style_points_green);
                }
                //Правельні відповіді фарбуємо зеленим - кінець
            }else {
                //Якщо ліве зображення менше
                if (count>0){
                    if (count==1){
                        count=0;
                    }else {
                        count=count-2;
                    }
                }
            }
        }
    }
});

```

```

    }
    //Фарбуємо прогрес сірим кольором - початок
    for (int i=0; i<9; i++){
        TextView tv = findViewById(progress[i]);
        tv.setBackgroundResource(R.drawable.style_points);
    }
    //Фарбуємо прогрес сірим кольором - кінець
    //Правельні відповіді фарбуємо зеленим - початок
    for (int i=0; i<count; i++){
        TextView tv = findViewById(progress[i]);

tv.setBackgroundResource(R.drawable.style_points_green);
    }
    //Правельні відповіді фарбуємо зеленим - кінець
}
//Якщо доторкнувся до зображення - кінець
if (count==10){
    //ВИХІД З РІВНЯ
    SharedPreferences save = getSharedPreferences("Save",
MODE_PRIVATE);

    final int level = save.getInt("Level1", 1);
    if (level>3){
        //порожнє
    }else {
        SharedPreferences.Editor editor = save.edit();
        editor.putInt("Level1", 4);
        editor.commit();
    }
    dialogEnd.show();
}else {
    numLeft = random.nextInt(20); //Гинеруємо число
    img_left.setImageResource(array.images3[numLeft]); // з
масиву зображення

    numRight = random.nextInt(20);

    //Цикл перевіряючий рівність чисел - початок
    while (array.strong[numLeft]==array.strong[numRight]){
        numRight = random.nextInt(20);
    }
    //Цикл перевіряючий рівність чисел - кінець

    img_rihgt.setImageResource(array.images3[numRight]); // з
масиву зображення

    img_rihgt.setEnabled(true); //Включаем обратно правую
картинку

}
}
//Натискання зображення - кінець
return true;
}
});
//Обробна натискання на ліве зображення - кінець

//Обробна натискання на праве зображення - початок
img_rihgt.setOnTouchListener(new View.OnTouchListener() {
    @Override
    public boolean onTouch(View v, MotionEvent event) {
        //Натискання зображення - початок
        if (event.getAction()==MotionEvent.ACTION_DOWN){
            //Якщо доторкнувся до зображення - початок
            img_left.setEnabled(false); //Блокування ліве зображення
            if (array.strong[numLeft]<array.strong[numRight]){

```

```

        img_rihgt.setImageResource(R.drawable.img_true);
    }else {
        img_rihgt.setImageResource(R.drawable.img_false);
    }
    //Якщо доторкнувся до зображення - кінець
}else if (event.getAction()==MotionEvent.ACTION_UP){
    //Якщо доторкнувся до зображення - початок
    if (array.strong[numLeft]<array.strong[numRight]){
        //Якщо праве зображення більше
        if (count<10){
            count=count+1;
        }
        //Фарбуємо прогрес сірим кольором - початок
        for (int i=0; i<10; i++){
            TextView tv = findViewById(progress[i]);
            tv.setBackgroundResource(R.drawable.style_points);
        }
        //Фарбуємо прогрес сірим кольором - кінець
        //Правельні відповіді фарбуємо зеленим - початок
        for (int i=0; i<count; i++){
            TextView tv = findViewById(progress[i]);

tv.setBackgroundResource(R.drawable.style_points_green);
        }
        //Правельні відповіді фарбуємо зеленим - кінець
    }else {
        //Якщо праве зображення менше
        if (count>0){
            if (count==1){
                count=0;
            }else {
                count=count-2;
            }
        }
        //Фарбуємо прогрес сірим кольором - початок
        for (int i=0; i<9; i++){
            TextView tv = findViewById(progress[i]);
            tv.setBackgroundResource(R.drawable.style_points);
        }
        //Фарбуємо прогрес сірим кольором - кінець
        //Правельні відповіді фарбуємо зеленим - початок
        for (int i=0; i<count; i++){
            TextView tv = findViewById(progress[i]);

tv.setBackgroundResource(R.drawable.style_points_green);
        }
        //Правельні відповіді фарбуємо зеленим - кінець
    }
    //Якщо доторкнувся до зображення - кінець
    if (count==10){
        //ВИХІД З РІВНЯ
        SharedPreferences save = getSharedPreferences("Save",
MODE_PRIVATE);

        final int level = save.getInt("Level1", 1);
        if (level>3){
            //порожнє
        }else {
            SharedPreferences.Editor editor = save.edit();
            editor.putInt("Level1", 4);
            editor.commit();
        }
        dialogEnd.show();
    }else {
        numLeft = random.nextInt(20); //Гинеруємо число
    }
}

```

```

        img_left.setImageResource(array.images3[numLeft]); // 3
масиву зображення

        numRight = random.nextInt(20);

        //Цикл перевіряючий рівність чисел - початок
        while (array.strong[numLeft]==array.strong[numRight]){
            numRight = random.nextInt(20);
        }
        //Цикл перевіряючий рівність чисел - кінець

        img_rihgt.setImageResource(array.images3[numRight]); // 3
масиву зображення

        img_left.setEnabled(true); //Включаем обратно ліву
картинку

    }
    }
    //Натискання зображення - кінець
    return true;
}
});
//Обробна натискання на праве зображення - кінець

}
//Системна кнопка "Назад" - початок
@Override
public void onBackPressed(){
    //Натискання кнопки "Назад" - початок
    try {
        //Повернутись до вибору рівня - початок
        Intent intent = new Intent(Level3.this, GameLevels.class); //Перехід
        startActivity(intent);
        finish();
        //Повернутись до вибору рівня - кінець
    } catch (Exception e){
        //Я то кажу, що кудо не має. Що ти від мене хочеш?
    }
    //Натискання кнопки "Назад" - кінець
}
//Системна кнопка "Назад" - кінець
}
}

```

Level_4

```

package space.stryuk.quizv2;

import android.app.Dialog;
import android.content.Intent;
import android.content.SharedPreferences;
import android.graphics.Color;
import android.graphics.drawable.ColorDrawable;
import android.os.Bundle;
import android.view.MotionEvent;
import android.view.View;
import android.view.Window;
import android.view.WindowManager;
import android.view.animation.Animation;
import android.view.animation.AnimationUtils;
import android.widget.Button;

```

```

import android.widget.ImageView;
import android.widget.LinearLayout;
import android.widget.TextView;

import androidx.appcompat.app.AppCompatActivity;

import java.util.Random;

public class Level4 extends AppCompatActivity {

    Dialog dialog;
    Dialog dialogEnd;

    public int numLeft; //Для лівого зображення
    public int numRight; //Для правого зображення
    Array array = new Array();
    Random random = new Random(); //Генерація чисел
    public int count = 0; //Правельні відповіді

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.universal);

        //Робимо text_levels
        TextView text_levels = findViewById(R.id.text_levels);
        text_levels.setText(R.string.level_four); //Встановили текст.

        final ImageView img_left = (ImageView) findViewById(R.id.img_left);
        //код Округлення кутів лівого зображення
        img_left.setClipToOutline(true);

        final ImageView img_rihgt = (ImageView) findViewById(R.id.img_right);
        //код Округлення кутів правого зображення
        img_rihgt.setClipToOutline(true);

        //Путь до лівою Textview

        //Розгорнути гру у весь екран - початок
        Window w = getWindow();
        w.setFlags(WindowManager.LayoutParams.FLAG_LAYOUT_NO_LIMITS,
        WindowManager.LayoutParams.FLAG_LAYOUT_NO_LIMITS);
        //Розгорнути гру у весь екран - Кінець

        //Встановлюємо фон - початок
        ImageView background = (ImageView) findViewById(R.id.background);
        background.setImageResource(R.drawable.level4_main_background);
        //Встановлюємо фон - початок

        //Виклик діалогового вікна на початку рівня
        dialog = new Dialog(this); //Робимо вікно діалога
        dialog.requestWindowFeature(Window.FEATURE_NO_TITLE); //забираємо
заголовог
        dialog.setContentView(R.layout.previewdialog); //путь до діалогового
вікна
        dialog.getWindow().setBackgroundDrawable(new
ColorDrawable(Color.TRANSPARENT)); //прозорий фон вікна
        dialog.setCancelable(false); //вікно не можна зачинити кнопкою "назад"

        //Встановимо зображення в діалогове віко - початок
        ImageView previewimg = (ImageView) dialog.findViewById(R.id.previewimg);

```



```

        dialogEnd.getWindow().setBackgroundDrawable(new
ColorDrawable(Color.TRANSPARENT));//прозорий фон вікна
        dialogEnd.getWindow().setLayout(WindowManager.LayoutParams.MATCH_PARENT,
WindowManager.LayoutParams.MATCH_PARENT);
        dialogEnd.setCancelable(false);//вікно не можна зачинити кнопкою "назад"

        //Встановлюємо фон діалогово вікна - початок
        LinearLayout dialogfonEnd =
        (LinearLayout) dialogEnd.findViewById(R.id.dialogfon);
        dialogfonEnd.setBackgroundResource(R.drawable.level4_preview_end);
        //Встановлюємо фон діалогово вікна - кінець

        //цікавий факт - початок
        TextView textdescriptionEnd =
        (TextView) dialogEnd.findViewById(R.id.textdescriptionEnd);
        textdescriptionEnd.setText(R.string.levelfourEnd);

        //цікавий факт - кінець

        //Кнопка для закриття вікна діалогу - початок
        TextView buttonclose2 =
        (TextView) dialogEnd.findViewById(R.id.buttonclose);
        buttonclose2.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                //натискання кнопки - початок
                try {
                    //повернутись до рівня - початок
                    Intent intent = new Intent(Level4.this,
GameLevels.class);//перехід
                    startActivity(intent);
                    finish();// закрити вікно
                    //повернутись до рівня - кінець
                } catch (Exception e) {
                    //У цьому місті не має коду. А ну геть!
                }
                dialogEnd.dismiss();//Закриття діалогового вікна

                //натискання кнопки - кінець
            }
        });
        //Кнопка для закриття вікна діалогу - кінець

        //Кнопка "Далі" - початок
        Button buttoncontinue2 =
        (Button) dialogEnd.findViewById(R.id.buttoncontinue);
        buttoncontinue2.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                try {
                    Intent intent = new Intent(Level4.this, Level5.class);
                    startActivity(intent);
                    finish();
                } catch (Exception e) {
                    //Кода не має
                }
                dialogEnd.dismiss();//Закрити діалогове вікно
            }
        });
        //Кнопка "Далі" - кінець

```

```

//


---


//Кнопка назад - Початок
Button btn_back = (Button)findViewById(R.id.button_back_level1);
btn_back.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {
        //Натискання кнопки "Назад" - початок
        try {
            //Повернутись до вибору рівня - початок
            Intent intent = new Intent(Level4.this,
GameLevels.class); //Перехід
            startActivity(intent);
            finish();
            //Повернутись до вибору рівня - кінець
        } catch (Exception e) {
            //Я то кажу, що кудо не має. Що ти від мене хочеш?
        }
        //Натискання кнопки "Назад" - кінець
    }
});

//Кнопка назад - Кінець

//Масив для прогресу гри - початок
final int[] progress = {
R.id.point1,R.id.point2,R.id.point3,R.id.point4,R.id.point5,R.id.point6,
R.id.point7,R.id.point8,R.id.point9,R.id.point10,
};
//Масив для прогресу гри - початок

//Застосування анімації - початок
final Animation a =
AnimationUtils.loadAnimation(Level4.this,R.anim.alpha);
//Застосування анімації - кінець

numLeft = random.nextInt(20); //Гинеруємо число
img_left.setImageResource(array.images4[numLeft]); // З масиву зображення

numRight = random.nextInt(20);

//Цикл перевіряючий рівність чисел - початок
while (array.strong[numLeft]==array.strong[numRight]){
    numRight = random.nextInt(20);
}
//Цикл перевіряючий рівність чисел - кінець

img_rihgt.setImageResource(array.images4[numRight]); // З масиву
зображення

//Обробна натискання на ліве зображення - початок
img_left.setOnTouchListener(new View.OnTouchListener() {
    @Override
    public boolean onTouch(View v, MotionEvent event) {
        //Натискання зображення - початок

```



```

if (event.getAction()==MotionEvent.ACTION_DOWN) {
    //Якщо дотокнувся до зображення - початок
    img_rihgt.setEnabled(false); //Блокування правого зображення
    if (array.strong[numLeft]>array.strong[numRight]) {
        img_left.setImageResource(R.drawable.img_true);
    }else {
        img_left.setImageResource(R.drawable.img_false);
    }
    //Якщо дотокнувся до зображення - кінець
} else if (event.getAction()==MotionEvent.ACTION_UP) {
    //Якщо дотокнувся до зображення - початок
    if (array.strong[numLeft]>array.strong[numRight]) {
        //Якщо ліве зображення більше
        if (count<10) {
            count=count+1;
        }
        //Фарбуємо прогрес сірим кольором - початок
        for (int i=0; i<10; i++){
            TextView tv = findViewById(progress[i]);
            tv.setBackgroundResource(R.drawable.style_points);
        }
        //Фарбуємо прогрес сірим кольором - кінець
        //Правельні відповіді фарбуємо зеленим - початок
        for (int i=0; i<count; i++){
            TextView tv = findViewById(progress[i]);
            tv.setBackgroundResource(R.drawable.style_points_green);
        }
        //Правельні відповіді фарбуємо зеленим - кінець
    } else {
        //Якщо ліве зображення менше
        if (count>0) {
            if (count==1) {
                count=0;
            } else {
                count=count-2;
            }
        }
        //Фарбуємо прогрес сірим кольором - початок
        for (int i=0; i<9; i++){
            TextView tv = findViewById(progress[i]);
            tv.setBackgroundResource(R.drawable.style_points);
        }
        //Фарбуємо прогрес сірим кольором - кінець
        //Правельні відповіді фарбуємо зеленим - початок
        for (int i=0; i<count; i++){
            TextView tv = findViewById(progress[i]);
            tv.setBackgroundResource(R.drawable.style_points_green);
        }
        //Правельні відповіді фарбуємо зеленим - кінець
    }
    //Якщо доторкнувся до зображення - кінець
    if (count==10) {
        //ВИХІД З РІВНЯ
        SharedPreferences save = getSharedPreferences("Save",
MODE_PRIVATE);

        final int level = save.getInt("Level1", 1);
        if (level>4) {
            //порожне
        } else {
            SharedPreferences.Editor editor = save.edit();
            editor.putInt("Level1", 5);
            editor.commit();
        }
    }
}

```

```

    }
    dialogEnd.show();
} else {
    numLeft = random.nextInt(20); //Гинеруємо число
    img_left.setImageResource(array.images4[numLeft]); // З
масиву зображення

    numRight = random.nextInt(20);

    //Цикс перевіряючий рівність чисел - початок
    while (array.strong[numLeft]==array.strong[numRight]){
        numRight = random.nextInt(20);
    }
    //Цикс перевіряючий рівність чисел - кінець

    img_rihgt.setImageResource(array.images4[numRight]); // З
масиву зображення

    img_rihgt.setEnabled(true); //Включаем обратно правую
картинку

}
}
//Натискання зображення - кінець
return true;
});
//Обробна натискання на ліве зображення - кінець

//Обробна натискання на праве зображення - початок
img_rihgt.setOnTouchListener(new View.OnTouchListener() {
    @Override
    public boolean onTouch(View v, MotionEvent event) {
        //Натискання зображення - початок
        if (event.getAction()==MotionEvent.ACTION_DOWN){
            //Якщо дотокнувся до зображення - початок
            img_left.setEnabled(false); //Блокування ліве зображення
            if (array.strong[numLeft]<array.strong[numRight]){
                img_rihgt.setImageResource(R.drawable.img_true);
            } else {
                img_rihgt.setImageResource(R.drawable.img_false);
            }
            //Якщо дотокнувся до зображення - кінець
        } else if (event.getAction()==MotionEvent.ACTION_UP){
            //Якщо дотокнувся до зображення - початок
            if (array.strong[numLeft]<array.strong[numRight]){
                //Якщо праве зображення більше
                if (count<10){
                    count=count+1;
                }
                //Фарбуємо прогрес сірим кольором - початок
                for (int i=0; i<10; i++){
                    TextView tv = findViewById(progress[i]);
                    tv.setBackgroundResource(R.drawable.style_points);
                }
                //Фарбуємо прогрес сірим кольором - кінець
                //Правельні відповіді фарбуємо зеленим - початок
                for (int i=0; i<count; i++){
                    TextView tv = findViewById(progress[i]);
tv.setBackgroundResource(R.drawable.style_points_green);
                }
                //Правельні відповіді фарбуємо зеленим - кінець
            } else {
                //Якщо праве зображення менше

```

```

        if (count>0){
            if (count==1){
                count=0;
            }else {
                count=count-2;
            }
        }
        //Фарбуємо прогрес сірим кольором - початок
        for (int i=0; i<9; i++){
            TextView tv = findViewById(progress[i]);
            tv.setBackgroundResource(R.drawable.style_points);
        }
        //Фарбуємо прогрес сірим кольором - кінець
        //Правельні відповіді фарбуємо зеленим - початок
        for (int i=0; i<count; i++){
            TextView tv = findViewById(progress[i]);

tv.setBackgroundResource(R.drawable.style_points_green);
        }
        //Правельні відповіді фарбуємо зеленим - кінець
    }
    //Якщо доторкнувся до зображення - кінець
    if (count==10){
        //ВИХІД З РІВНЯ
        SharedPreferences save = getSharedPreferences("Save",
MODE_PRIVATE);

        final int level = save.getInt("Level1", 1);
        if (level>4){
            //порожнє
        }else {
            SharedPreferences.Editor editor = save.edit();
            editor.putInt("Level1", 5);
            editor.commit();
        }
        dialogEnd.show();
    }else {
        numLeft = random.nextInt(20); //Гинеруємо число
        img_left.setImageResource(array.images4[numLeft]); // з
масиву зображення

        numRight = random.nextInt(20);

        //Цикл перевіряючий рівність чисел - початок
        while (array.strong[numLeft]==array.strong[numRight]){
            numRight = random.nextInt(20);
        }
        //Цикл перевіряючий рівність чисел - кінець

        img_rihgt.setImageResource(array.images4[numRight]); // з
масиву зображення

        img_left.setEnabled(true); //Включаем обратно ліву
картинку

    }
}
//Натискання зображення - кінець
return true;
}
});
//Обробна натискання на праве зображення - кінець
}
}

```

```

//Системна кнопка "Назад" - початок
@Override
public void onBackPressed(){
    //Натискання кнопки "Назад" - початок
    try {
        //Повернутись до вибору рівня - початок
        Intent intent = new Intent(Level4.this, GameLevels.class); //Перехід
        startActivity(intent);
        finish();
        //Повернутись до вибору рівня - кінець
    } catch (Exception e){
        //Я то кажу, що куди не має. Що ти від мене хочеш?
    }
    //Натискання кнопки "Назад" - кінець
}
//Системна кнопка "Назад" - кінець
}

```

Level_5

```

package space.stryuk.quizv2;

import android.app.Dialog;
import android.content.Intent;
import android.graphics.Color;
import android.graphics.drawable.ColorDrawable;
import android.os.Bundle;
import android.view.MotionEvent;
import android.view.View;
import android.view.Window;
import android.view.WindowManager;
import android.view.animation.Animation;
import android.view.animation.AnimationUtils;
import android.widget.Button;
import android.widget.ImageView;
import android.widget.LinearLayout;
import android.widget.TextView;

import androidx.appcompat.app.AppCompatActivity;

import java.util.Random;

public class Level5 extends AppCompatActivity {

    Dialog dialog;
    Dialog dialogEnd;

    public int numLeft; //Для лівого зображення
    public int numRight; //Для правого зображення
    Array array = new Array();
    Random random = new Random(); //Генерація чисел
    public int count = 0; //Правельні відповіді

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.universal);

        //Робимо text_levels
        TextView text_levels = findViewById(R.id.text_levels);
    }
}

```

```

text_levels.setText(R.string.level_five); //Встановили текст.

final ImageView img_left = (ImageView) findViewById(R.id.img_left);
//код Округлення кутів лівого зображення
img_left.setClipToOutline(true);

final ImageView img_rihgt = (ImageView) findViewById(R.id.img_right);
//код Округлення кутів правого зображення
img_rihgt.setClipToOutline(true);

//Путь до лівою Textview

//Розгорнути гру у весь екран - початок
Window w = getWindow();
w.setFlags(WindowManager.LayoutParams.FLAG_LAYOUT_NO_LIMITS,
WindowManager.LayoutParams.FLAG_LAYOUT_NO_LIMITS);
//Розгорнути гру у весь екран - Кінець

//Встановлюємо фон - початок
ImageView background = (ImageView) findViewById(R.id.background);
background.setImageResource(R.drawable.level5_main_background);
//Встановлюємо фон - початок

//Виклик діалогового вікна на початку рівня
dialog = new Dialog(this); //Робимо вікно діалога
dialog.requestWindowFeature(Window.FEATURE_NO_TITLE); //забираємо
заголовог
вікна
dialog.setContentView(R.layout.previewdialog); //путь до діалогового
вікна
dialog.getWindow().setBackgroundDrawable(new
ColorDrawable(Color.TRANSPARENT)); //прозорий фон вікна
dialog.setCancelable(false); //вікно не можна зачинити кнопкою "назад"

//Встановимо зображення в діалогове віко - початок
ImageView previewimg = (ImageView) dialog.findViewById(R.id.previewimg);
previewimg.setImageResource(R.drawable.level5_smail_preview_img5);
//Встановимо зображення в діалогове віко - кінець

//встановлюємо фон діалогового вікна - початок
LinearLayout dialogdon =
(LinearLayout) dialog.findViewById(R.id.dialogdon);
dialogdon.setBackgroundResource(R.drawable.level5_preview);
//встановлюємо фон діалогового вікна - кінець

//Встановлюємо опис завдання - початок
TextView textdescription =
(TextView) dialog.findViewById(R.id.textdescription);
textdescription.setText(R.string.levelfive);
//Встановлюємо опис завдання - кінець

//Кнопка для закриття вікна діалогу - початок
TextView buttonclose = (TextView) dialog.findViewById(R.id.buttonclose);
buttonclose.setOnClickListener(new View.OnClickListener() {
@Override
public void onClick(View v) {
//натискання кнопки - початок
try {
//повернутись до рівня - початок
Intent intent = new Intent(Level5.this,
GameLevels.class); //перехід
startActivity(intent);
finish(); // закрити вікно
}
}
}
}

```

```

        //повернутись до рівня - кінець
    }catch (Exception e){
        //У цьому місті не має коду. А ну геть!
    }
    dialog.dismiss();//Закриття діалогового вікна

    //натискання кнопки - кінець
}
});
//Кнопка для закриття вікна діалогу - кінець

//Кнопка "Далі" - початок
Button buttoncontinue =
(Button)dialog.findViewById(R.id.buttoncontinue);
buttoncontinue.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {
        dialog.dismiss();//Закрити діалогове вікно
    }
});

//Кнопка "Далі" - кінець

dialog.show();//показати діалогове вікно

//
//Виклик діалогового вікна у кінці рівня
dialogEnd = new Dialog(this);//Робимо вікно діалога
dialogEnd.requestWindowFeature(Window.FEATURE_NO_TITLE);//забираємо
заголовков
dialogEnd.setContentView(R.layout.dialog_end); //путь до діалогового
вікна
dialogEnd.getWindow().setBackgroundDrawable(new
ColorDrawable(Color.TRANSPARENT));//прозорий фон вікна
dialogEnd.getWindow().setLayout(WindowManager.LayoutParams.MATCH_PARENT,
WindowManager.LayoutParams.MATCH_PARENT);
dialogEnd.setCancelable(false);//вікно не можна зачинити кнопкою "назад"

//Встановлюємо фон діалогово вікна - початок
LinearLayout dialogfonEnd =
(LinearLayout)dialogEnd.findViewById(R.id.dialogfon);
dialogfonEnd.setBackgroundResource(R.drawable.level5_preview);
//Встановлюємо фон діалогово вікна - кінець

//цікавий факт - початок
TextView textdescriptionEnd =
(TextView)dialogEnd.findViewById(R.id.textdescriptionEnd);
textdescriptionEnd.setText(R.string.levelfiveEnd);

//цікавий факт - кінець

//Кнопка для закриття вікна діалогу - початок
TextView buttonclose2 =
(TextView)dialogEnd.findViewById(R.id.buttonclose);
buttonclose2.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {
        //натискання кнопки - початок
        try {

```

```

        //повернутись до рівня - початок
        Intent intent = new Intent(Level5.this,
GameLevels.class); //перехід
        startActivity(intent);
        finish(); // закрити вікно
        //повернутись до рівня - кінець
    } catch (Exception e) {
        //У цьому місті не має коду. А ну геть!
    }
    dialogEnd.dismiss(); //Закриття діалогового вікна

    //натискання кнопки - кінець
}
});
//Кнопка для закриття вікна діалогу - кінець

//Кнопка "Далі" - початок
Button buttoncontinue2 =
(Button) dialogEnd.findViewById(R.id.buttoncontinue);
buttoncontinue2.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {
        try {
            Intent intent = new Intent(Level5.this, GameLevels.class);
            startActivity(intent);
            finish();
        } catch (Exception e) {
            //Кода не має
        }
        dialogEnd.dismiss(); //Закрити діалогове вікно
    }
});

//Кнопка "Далі" - кінець

//


---


//Кнопка назад - Початок
Button btn_back = (Button) findViewById(R.id.button_back_level1);
btn_back.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {
        //Натискання кнопки "Назад" - початок
        try {
            //Повернутись до вибору рівня - початок
            Intent intent = new Intent(Level5.this,
GameLevels.class); //Перехід
            startActivity(intent);
            finish();
            //Повернутись до вибору рівня - кінець
        } catch (Exception e) {
            //Я то кажу, що кудо не має. Що ти від мене хочеш?
        }
        //Натискання кнопки "Назад" - кінець
    }
});

//Кнопка назад - Кінець

```

```

//Масив для прогресу гри - початок
final int[] progress = {
R.id.point1,R.id.point2,R.id.point3,R.id.point4,R.id.point5,R.id.point6,
    R.id.point7,R.id.point8,R.id.point9,R.id.point10,
};
//Масив для прогресу гри - початок

//Застосування анімації - початок
final Animation a =
AnimationUtils.loadAnimation(Level5.this,R.anim.alpha);
//Застосування анімації - кінець

numLeft = random.nextInt(20);//Гинеруємо число
img_left.setImageResource(array.images5[numLeft]); // З масиву зображення

numRight = random.nextInt(20);

//Цикл перевіряючий рівність чисел - початок
while (array.strong[numLeft]==array.strong[numRight]){
    numRight = random.nextInt(20);
}
//Цикл перевіряючий рівність чисел - кінець

img_rihgt.setImageResource(array.images5[numRight]); // З масиву
зображення

//Обробна натискання на ліве зображення - початок
img_left.setOnTouchListener(new View.OnTouchListener() {
    @Override
    public boolean onTouch(View v, MotionEvent event) {
        //Натискання зображення - початок
        if (event.getAction()==MotionEvent.ACTION_DOWN){
            //Якщо дотокнувся до зображення - початок
            img_rihgt.setEnabled(false); //Блокування правого зображення
            if (array.strong[numLeft]>array.strong[numRight]){
                img_left.setImageResource(R.drawable.img_true);
            }else {
                img_left.setImageResource(R.drawable.img_false);
            }
            //Якщо дотокнувся до зображення - кінець
        }else if (event.getAction()==MotionEvent.ACTION_UP){
            //Якщо дотокнувся до зображення - початок
            if (array.strong[numLeft]>array.strong[numRight]){
                //Якщо ліве зображення більше
                if (count<10){
                    count=count+1;
                }
                //Фарбуємо прогрес сірим кольором - початок
                for (int i=0; i<10; i++){
                    TextView tv = findViewById(progress[i]);
                    tv.setBackgroundResource(R.drawable.style_points);
                }
                //Фарбуємо прогрес сірим кольором - кінець
                //Правельні відповіді фарбуємо зеленим - початок
                for (int i=0; i<count; i++){
                    TextView tv = findViewById(progress[i]);
                    tv.setBackgroundResource(R.drawable.style_points_green);
                }
            }
        }
    }
});

```



```

        //Правельні відповіді фарбуємо зеленим - кінець
    }else {
        //Якщо ліве зображення менше
        if (count>0){
            if (count==1){
                count=0;
            }else {
                count=count-2;
            }
        }
        //Фарбуємо прогрес сірим кольором - початок
        for (int i=0; i<9; i++){
            TextView tv = findViewById(progress[i]);
            tv.setBackgroundResource(R.drawable.style_points);
        }
        //Фарбуємо прогрес сірим кольором - кінець
        //Правельні відповіді фарбуємо зеленим - початок
        for (int i=0; i<count; i++){
            TextView tv = findViewById(progress[i]);
            tv.setBackgroundResource(R.drawable.style_points_green);
        }
        //Правельні відповіді фарбуємо зеленим - кінець
    }
    //Якщо доторкнувся до зображення - кінець
    if (count==10){
        //ВИХІД З РІВНЯ
        dialogEnd.show();
    }else {
        numLeft = random.nextInt(20); //Гинеруємо число
        img_left.setImageResource(array.images5[numLeft]); // 3
        масиву зображення

        numRight = random.nextInt(20);

        //Цикс перевіряючий рівність чисел - початок
        while (array.strong[numLeft]==array.strong[numRight]){
            numRight = random.nextInt(20);
        }
        //Цикс перевіряючий рівність чисел - кінець

        img_rihgt.setImageResource(array.images5[numRight]); // 3
        масиву зображення

        img_rihgt.setEnabled(true); //Включаем обратно правую
        картинку
    }
}
//Натискання зображення - кінець
return true;
}
});
//Обробна натискання на ліве зображення - кінець

//Обробна натискання на праве зображення - початок
img_rihgt.setOnClickListener(new View.OnClickListener() {
    @Override
    public boolean onTouch(View v, MotionEvent event) {
        //Натискання зображення - початок
        if (event.getAction()==MotionEvent.ACTION_DOWN){
            //Якщо доторкнувся до зображення - початок
            img_left.setEnabled(false); //Блокування ліве зображення
            if (array.strong[numLeft]<array.strong[numRight]){
                img_rihgt.setImageResource(R.drawable.img_true);
            }
        }
    }
});

```

```

    }else {
        img_rihgt.setImageResource(R.drawable.img_false);
    }
    //Якщо дотокнувся до зображення - кінець
}else if (event.getAction()==MotionEvent.ACTION_UP) {
    //Якщо дотокнувся до зображення - початок
    if (array.strong[numLeft]<array.strong[numRight]) {
        //Якщо праве зображення більше
        if (count<10){
            count=count+1;
        }
        //Фарбуємо прогрес сірим кольором - початок
        for (int i=0; i<10; i++){
            TextView tv = findViewById(progress[i]);
            tv.setBackgroundResource(R.drawable.style_points);
        }
        //Фарбуємо прогрес сірим кольором - кінець
        //Правельні відповіді фарбуємо зеленим - початок
        for (int i=0; i<count; i++){
            TextView tv = findViewById(progress[i]);

tv.setBackgroundResource(R.drawable.style_points_green);
        }
        //Правельні відповіді фарбуємо зеленим - кінець
    }else {
        //Якщо праве зображення менше
        if (count>0){
            if (count==1){
                count=0;
            }else {
                count=count-2;
            }
        }
        //Фарбуємо прогрес сірим кольором - початок
        for (int i=0; i<9; i++){
            TextView tv = findViewById(progress[i]);
            tv.setBackgroundResource(R.drawable.style_points);
        }
        //Фарбуємо прогрес сірим кольором - кінець
        //Правельні відповіді фарбуємо зеленим - початок
        for (int i=0; i<count; i++){
            TextView tv = findViewById(progress[i]);

tv.setBackgroundResource(R.drawable.style_points_green);
        }
        //Правельні відповіді фарбуємо зеленим - кінець
    }
    //Якщо доторкнувся до зображення - кінець
    if (count==10) {
        //ВИХІД З РІВНЯ
        dialogEnd.show();
    }else {
        numLeft = random.nextInt(20); //Гинеруємо число
        img_left.setImageResource(array.images5[numLeft]); // з
масиву зображення

        numRight = random.nextInt(20);

        //Цикл перевіряючий рівність чисел - початок
        while (array.strong[numLeft]==array.strong[numRight]) {
            numRight = random.nextInt(20);
        }
        //Цикл перевіряючий рівність чисел - кінець

```

```

        img_rihgt.setImageResource(array.images5[numRight]); // 3
масиву зображення
        img_left.setEnabled(true); //Включаем обратно ліву
картинку
    }
}
//Натискання зображення - кінець
return true;
}
});
//Обробна натискання на праве зображення - кінець

}
//Системна кнопка "Назад" - початок
@Override
public void onBackPressed(){
//Натискання кнопки "Назад" - початок
try {
//Повернутись до вибору рівня - початок
Intent intent = new Intent(Level5.this, GameLevels.class); //Перехід
startActivity(intent);
finish();
//Повернутись до вибору рівня - кінець
}catch (Exception e){
//Я то кажу, що куди не має. Що ти від мене хочеш?
}
//Натискання кнопки "Назад" - кінець
}
//Системна кнопка "Назад" - кінець
}
}

```

colors.xml

```

<?xml version="1.0" encoding="utf-8"?>
<resources>
    <color name="white">#FFFFFF</color>
    <color name="black95">#95000000</color>
    <color name="black60">#60000000</color>
    <color name="blue">#48d1ff</color>
    <color name="green">#9019FF19</color>
</resources>

```

descriptionslevels.xml

```

<?xml version="1.0" encoding="utf-8"?>
<resources>
    <string name="levelone">Оберіть картку на якій зображено наймолодшого
українського діяча .\n Питання могу повторюватися доки ви не заробите 10 очок.
</string>
    <string name="leveloneEnd">"Вітаю, ви пройшли рівень!\n
    Чи знали ви, що Василь Стус був великим патріотом і не приховував цього. За
це він сумарно отримав 23 роки примусових робіт або позбавлення волі від
радянського режиму. Також Стус принципово розмовляв українською мовою і одного
разу побився в університеті за те, що йому сказали "говорити нормальною
мовою"."</string>
    <string name="leveltwo">Оберіть картку на якій зображено хімічний елемент з

```

найбільшим атомним номером .\n Питання могу повторюватися доки ви не заробите 10 очок. </string>

<string name="leveltwoEnd">"Вітаю,ви пройшли рівень!"\n

Чи знали ви, що відкриття періодичної системи сприяла любов Менделєєва до розкладання пасьянсів. Вчений вказував атомні ваги тих чи інших елементів на гральних картах, а потім розкладав їх так, ніби він грав у карткову гру. Таким чином він візуалізував своє припущення про те, що елементи зі схожими властивостями мають бути однакових «мастей». Надалі ці «масті» мали бути розкладені в «стопки» відповідно до їх атомними вагами.</string>

<string name="levelthree">Оберіть картку на якій зображено кухаря .\n Питання могу повторюватися доки ви не заробите 10 очок. </string>

<string name="levelthreeEnd">"Вітаю,ви пройшли рівень!"\n

- Білий кухарський ковпак на роботі стали носити у 18 столітті на прохання англійського короля Георга II. Цей головний убір захищав страви від попадання в них волосся з голови кухарів. Сьогодні вимога до його носіння закріплена у санітарних нормах багатьох країн світу.</string>

<string name="levelfour">Оберіть картку на якій зображено українського гравця.</string>

<string name="levelfourEnd">"Вітаю,ви пройшли рівень!"\n "Кар'єру Шевченка мало не обірвала аварія на Чорнобильській АЕС, тоді як дітей назавжди вивозили із зон, заражених радіацією, Андрій повернувся та продовжив грати."</string>

<string name="levelfive">Оберіть картку на якій зображено українського співака.</string>

<string name="levelfiveEnd">Вітаю,ви пройшли рівень!\n

"Чи знали ви, що першою композицією Скрябіна, яка підштовхнула його до меломанії, стала пісня легендарних The Beatles - Lady Madonna."</string>

</resources>

strings.xml

<resources>

<string name="app_name">QuizV2</string>

<string name="start">ПОЧАТИ</string>

<string name="back">Назад</string>

<string name="level1">Рівень:</string>

<string name="level_one">Рівень 1</string>

<string name="level_two">Рівень 2</string>

<string name="level_three">Рівень 3</string>

<string name="level_four">Рівень 4</string>

<string name="level_five">Рівень 5</string>

<string name="textcontinue">Далі</string>

</resources>

button_gamelevel.xml

<?xml version="1.0" encoding="utf-8" ?>

<selector xmlns:android="http://schemas.android.com/apk/res/android">

<item android:state_pressed="false"

android:drawable="@drawable/style_button_black95"></item>

<item android:state_pressed="true"

android:drawable="@drawable/style_button_black60"></item>

</selector>

button_stroke_black95_preес_white.xml

```

<?xml version="1.0" encoding="utf-8" ?>
<selector xmlns:android="http://schemas.android.com/apk/res/android">
<item android:state_pressed="false"
android:drawable="@drawable/style_button_stroke_black95"/>
  <item android:state_pressed="true"
android:drawable="@drawable/style_button_stroke_black95_press_white"/>
</selector>

```

button_stroke_white_press_blue.xml

```

<?xml version="1.0" encoding="utf-8" ?>
<selector xmlns:android="http://schemas.android.com/apk/res/android">
<item android:state_pressed="false"
android:drawable="@drawable/style_button_white"/>
  <item android:state_pressed="true"
android:drawable="@drawable/style_button_white_blue_press"/>
</selector>

```

style_bolvanka_univeral.xml

```

<?xml version="1.0" encoding="utf-8" ?>
<shape xmlns:android="http://schemas.android.com/apk/res/android">
<corners android:radius="15dp"></corners>
</shape>

```

style_button_black60.xml

```

<?xml version="1.0" encoding="utf-8" ?>
<shape xmlns:android="http://schemas.android.com/apk/res/android">
<corners android:radius="15dp"></corners>
  <solid android:color="@color/black60"></solid>
</shape>

```

style_button_black95.xml

```

<?xml version="1.0" encoding="utf-8" ?>
<shape xmlns:android="http://schemas.android.com/apk/res/android">
<corners android:radius="15dp"></corners>
  <solid android:color="@color/black95"></solid>
</shape>

```

style_button_stroke_black95.xml

```

<?xml version="1.0" encoding="utf-8" ?>
<shape xmlns:android="http://schemas.android.com/apk/res/android">
<stroke android:width="2dp" android:color="@color/black95" />
  <corners android:radius="20dp" />
</shape>

```

style_button_stroke_black95_press_white.xml

```

<?xml version="1.0" encoding="utf-8" ?>
<shape xmlns:android="http://schemas.android.com/apk/res/android">
<stroke android:width="2dp" android:color="@color/black95" />
  <corners android:radius="20dp" />

```

```
<solid android:color="@color/white"/>
</shape>
```

style_button_white.xml

```
<?xml version="1.0" encoding="utf-8" ?>
<shape xmlns:android="http://schemas.android.com/apk/res/android">
<stroke android:width="2dp" android:color="@color/white"/>
    <corners android:radius="20dp"/>
</shape>
```

style_button_white_blue_press.xml

```
<?xml version="1.0" encoding="utf-8" ?>
<shape xmlns:android="http://schemas.android.com/apk/res/android">
<stroke android:width="2dp" android:color="@color/white"/>
    <corners android:radius="20dp"/>
    <solid android:color="@color/blue"/>
</shape>
```

style_points.xml

```
<?xml version="1.0" encoding="utf-8" ?>
<shape xmlns:android="http://schemas.android.com/apk/res/android">
<corners android:radius="10dp"/>
    <solid android:color="@color/black60"/>
</shape>
```

style_points_green.xml

```
<?xml version="1.0" encoding="utf-8" ?>
<shape xmlns:android="http://schemas.android.com/apk/res/android">
<corners android:radius="10dp"/>
    <solid android:color="@color/green"/>
</shape>
```

Alpha.xml

```
<?xml version="1.0" encoding="utf-8" ?>
<set xmlns:android="http://schemas.android.com/apk/res/android">
<alpha
    android:duration = "200"
    android:fromAlpha="0.0"
    android:toAlpha="1.0">
</alpha>
</set>
```

activity_main.xml

```
<?xml version="1.0" encoding="utf-8" ?>
<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:id="@+id/RelativeLayout"
```

```
android:layout_width="match_parent"
android:layout_height="match_parent"
tools:context=".MainActivity">
```

```
<ImageView
    android:id="@+id/main_background"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:scaleType="centerCrop"
    android:src="@drawable/new_main_backgroundv3" />
```

```
<ImageView
    android:id="@+id/imageView2"
    android:layout_width="275dp"
    android:layout_height="376dp"
    android:layout_centerHorizontal="true"
    android:layout_marginStart="50dp"
    android:layout_marginTop="100dp"
    android:layout_marginEnd="50dp"
    android:layout_marginBottom="200dp"
    android:src="@drawable/main_img_boyv2" />
```

```
<LinearLayout
    android:id="@+id/startLayout"
    android:layout_width="136dp"
    android:layout_height="wrap_content"
    android:layout_alignParentBottom="true"
    android:layout_centerHorizontal="true"
    android:orientation="horizontal"
    android:padding="10dp">
```

```
<Button
    android:id="@+id/buttonStart"
    android:layout_width="116dp"
    android:layout_height="wrap_content"
    android:layout_weight="1"
    android:background="@drawable/button_stroke_black95_press_white"
    android:paddingBottom="-10dp"
    android:text="@string/start"
    android:textAllCaps="true"
    android:textColor="@color/black95"
    android:textSize="22sp" />
```

```
</LinearLayout>
</RelativeLayout>
```

dialog_end.xml

```
<?xml version="1.0" encoding="utf-8" ?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:orientation="vertical"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:id="@+id/dialogfon"
    android:background="@drawable/new_preview_backgrounde_level1"
    android:layout_gravity="center"
    android:scaleType="centerCrop"
    android:layout_margin="10dp">
    <LinearLayout
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:orientation="vertical"
        android:padding="20dp"
```

```

        android:gravity="center">
        <TextView
            android:layout_width="30dp"
            android:layout_height="30dp"
            android:id="@+id/buttonclose"
            android:text="&#60;"
            android:textStyle="bold"
            android:gravity="center"
            android:background="@drawable/button_stroke_black95_press_white"
            android:layout_gravity="start"
        />

```

```

        <ScrollView
            android:layout_width="wrap_content"
            android:layout_height="0dp"
            android:layout_weight="88"
            android:layout_marginTop="10dp">

```

```

            <TextView
                android:layout_width="match_parent"
                android:layout_height="match_parent"
                android:id="@+id/textdescriptionEnd"
                android:text="@string/leveloneEnd"
                android:textColor="@color/black95"
                android:textSize="22sp" />

```

```

        </ScrollView>

```

```

        <Button

```

```

            android:layout_width="wrap_content"
            android:layout_height="0dp"
            android:id="@+id/buttoncontinue"
            android:background="@drawable/button_stroke_black95_press_white"
            android:layout_weight="12"
            android:text="@string/textcontinue"
            android:textColor="@color/black95"
            android:textAllCaps="true"
            android:textSize="22sp"
            android:textStyle="bold"
            android:layout_marginTop="10dp"
        />

```

```

    </LinearLayout>

```

```

</LinearLayout>

```

gamelevels.xml

```

<?xml version="1.0" encoding="utf-8" ?>
<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:orientation="vertical">

    <ImageView
        android:id="@+id/imageView"
        android:layout_width="match_parent"
        android:layout_height="match_parent"
        android:scaleType="centerCrop"
        app:srcCompat="@drawable/main2_background" />

    <LinearLayout
        android:id="@+id/container1"

```



```
android:layout_width="match_parent"
android:layout_height="match_parent"
android:gravity="center"
android:orientation="vertical">
```

```
<LinearLayout
```

```
    android:id="@+id/container2"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:orientation="vertical"
    android:padding="10dp">
```

```
<LinearLayout
```

```
    android:id="@+id/container3"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:orientation="horizontal">
```

```
<Button
```

```
    android:id="@+id/button_back"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="@string/back"
```

```
android:background="@drawable/button_stroke_black95_press_white"
    android:textSize="22sp"
    android:paddingLeft="5dp"
    android:paddingRight="5dp"
    android:textColor="@color/black95"
    android:textAllCaps="false"
/>
```

```
</LinearLayout>
```

```
<ScrollView
```

```
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:layout_marginTop="10dp">
```

```
<LinearLayout
```

```
    android:id="@+id/container4"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:orientation="vertical" >
```

```
<LinearLayout
```

```
    android:id="@+id/line1"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:orientation="horizontal">
```

```
<TextView
```

```
    android:id="@+id/textView1"
    android:layout_width="50dp"
    android:layout_height="50dp"
    android:layout_marginRight="10dp"
    android:background="@drawable/button_gamelevels"
    android:gravity="center"
    android:text="1"
    android:textColor="@color/white"
    android:textSize="24sp"
    android:textStyle="bold" />
```

```
<TextView
```

```
    android:id="@+id/textView2"
```

```
android:layout_width="50dp"
android:layout_height="50dp"
android:layout_marginRight="10dp"
android:background="@drawable/button_gamelevels"
android:gravity="center"
android:text="X"
android:textColor="@color/white"
android:textSize="24sp"
android:textStyle="bold" />
```

```
<TextView
    android:id="@+id/textView3"
    android:layout_width="50dp"
    android:layout_height="50dp"
    android:layout_marginRight="10dp"
    android:background="@drawable/button_gamelevels"
    android:gravity="center"
    android:text="X"
    android:textColor="@color/white"
    android:textSize="24sp"
    android:textStyle="bold" />
```

```
<TextView
    android:id="@+id/textView4"
    android:layout_width="50dp"
    android:layout_height="50dp"
    android:layout_marginRight="10dp"
    android:background="@drawable/button_gamelevels"
    android:gravity="center"
    android:text="X"
    android:textColor="@color/white"
    android:textSize="24sp"
    android:textStyle="bold" />
```

```
<TextView
    android:id="@+id/textView5"
    android:layout_width="50dp"
    android:layout_height="50dp"
    android:layout_marginRight="10dp"
    android:background="@drawable/button_gamelevels"
    android:gravity="center"
    android:text="X"
    android:textColor="@color/white"
    android:textSize="24sp"
    android:textStyle="bold" />
```

```
</LinearLayout>
```

```
<LinearLayout
    android:id="@+id/line2"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:orientation="horizontal"
    android:layout_marginTop="10dp">
```

```
<TextView
    android:id="@+id/textView6"
    android:layout_width="50dp"
    android:layout_height="50dp"
    android:text="X"
    android:layout_marginRight="10dp"
    android:background="@drawable/button_gamelevels"
    android:textColor="@color/white"
    android:textSize="24sp"
    android:textStyle="bold"
```

```

        android:gravity="center"
    />
<TextView
    android:id="@+id/textView7"
    android:layout_width="50dp"
    android:layout_height="50dp"
    android:text="X"
    android:layout_marginRight="10dp"
    android:background="@drawable/button_gamelevels"
    android:textColor="@color/white"
    android:textSize="24sp"
    android:textStyle="bold"
    android:gravity="center"
    />
<TextView
    android:id="@+id/textView8"
    android:layout_width="50dp"
    android:layout_height="50dp"
    android:text="X"
    android:layout_marginRight="10dp"
    android:background="@drawable/button_gamelevels"
    android:textColor="@color/white"
    android:textSize="24sp"
    android:textStyle="bold"
    android:gravity="center"
    />
<TextView
    android:id="@+id/textView9"
    android:layout_width="50dp"
    android:layout_height="50dp"
    android:text="X"
    android:layout_marginRight="10dp"
    android:background="@drawable/button_gamelevels"
    android:textColor="@color/white"
    android:textSize="24sp"
    android:textStyle="bold"
    android:gravity="center"
    />
<TextView
    android:id="@+id/textView10"
    android:layout_width="50dp"
    android:layout_height="50dp"
    android:text="X"
    android:layout_marginRight="10dp"
    android:background="@drawable/button_gamelevels"
    android:textColor="@color/white"
    android:textSize="24sp"
    android:textStyle="bold"
    android:gravity="center"
    />
</LinearLayout>

<LinearLayout
    android:id="@+id/line3"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:orientation="horizontal"
    android:layout_marginTop="10dp">

    <TextView
        android:id="@+id/textView11"
        android:layout_width="50dp"
        android:layout_height="50dp"
        android:text="X"

```

```

        android:layout_marginRight="10dp"
        android:background="@drawable/button_gamelevels"
        android:textColor="@color/white"
        android:textSize="24sp"
        android:textStyle="bold"
        android:gravity="center"
    />
<TextView
    android:id="@+id/textView12"
    android:layout_width="50dp"
    android:layout_height="50dp"
    android:text="X"
    android:layout_marginRight="10dp"
    android:background="@drawable/button_gamelevels"
    android:textColor="@color/white"
    android:textSize="24sp"
    android:textStyle="bold"
    android:gravity="center"
/>
<TextView
    android:id="@+id/textView13"
    android:layout_width="50dp"
    android:layout_height="50dp"
    android:text="X"
    android:layout_marginRight="10dp"
    android:background="@drawable/button_gamelevels"
    android:textColor="@color/white"
    android:textSize="24sp"
    android:textStyle="bold"
    android:gravity="center"
/>
<TextView
    android:id="@+id/textView14"
    android:layout_width="50dp"
    android:layout_height="50dp"
    android:text="X"
    android:layout_marginRight="10dp"
    android:background="@drawable/button_gamelevels"
    android:textColor="@color/white"
    android:textSize="24sp"
    android:textStyle="bold"
    android:gravity="center"
/>
<TextView
    android:id="@+id/textView15"
    android:layout_width="50dp"
    android:layout_height="50dp"
    android:text="X"
    android:layout_marginRight="10dp"
    android:background="@drawable/button_gamelevels"
    android:textColor="@color/white"
    android:textSize="24sp"
    android:textStyle="bold"
    android:gravity="center"
/>
</LinearLayout>

<LinearLayout
    android:id="@+id/line4"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:orientation="horizontal"
    android:layout_marginTop="10dp">

```

```

<TextView
    android:id="@+id/textView16"
    android:layout_width="50dp"
    android:layout_height="50dp"
    android:text="X"
    android:layout_marginRight="10dp"
    android:background="@drawable/button_gamelevels"
    android:textColor="@color/white"
    android:textSize="24sp"
    android:textStyle="bold"
    android:gravity="center"
/>
<TextView
    android:id="@+id/textView17"
    android:layout_width="50dp"
    android:layout_height="50dp"
    android:text="X"
    android:layout_marginRight="10dp"
    android:background="@drawable/button_gamelevels"
    android:textColor="@color/white"
    android:textSize="24sp"
    android:textStyle="bold"
    android:gravity="center"
/>
<TextView
    android:id="@+id/textView18"
    android:layout_width="50dp"
    android:layout_height="50dp"
    android:text="X"
    android:layout_marginRight="10dp"
    android:background="@drawable/button_gamelevels"
    android:textColor="@color/white"
    android:textSize="24sp"
    android:textStyle="bold"
    android:gravity="center"
/>
<TextView
    android:id="@+id/textView19"
    android:layout_width="50dp"
    android:layout_height="50dp"
    android:text="X"
    android:layout_marginRight="10dp"
    android:background="@drawable/button_gamelevels"
    android:textColor="@color/white"
    android:textSize="24sp"
    android:textStyle="bold"
    android:gravity="center"
/>
<TextView
    android:id="@+id/textView20"
    android:layout_width="50dp"
    android:layout_height="50dp"
    android:text="X"
    android:layout_marginRight="10dp"
    android:background="@drawable/button_gamelevels"
    android:textColor="@color/white"
    android:textSize="24sp"
    android:textStyle="bold"
    android:gravity="center"
/>
</LinearLayout>

<LinearLayout
    android:id="@+id/line5"

```

```
android:layout_width="match_parent"  
android:layout_height="wrap_content"  
android:orientation="horizontal"  
android:layout_marginTop="10dp">
```

```
<TextView
```

```
    android:id="@+id/textView21"  
    android:layout_width="50dp"  
    android:layout_height="50dp"  
    android:text="X"  
    android:layout_marginRight="10dp"  
    android:background="@drawable/button_gamelevels"  
    android:textColor="@color/white"  
    android:textSize="24sp"  
    android:textStyle="bold"  
    android:gravity="center"  
/>
```

```
<TextView
```

```
    android:id="@+id/textView22"  
    android:layout_width="50dp"  
    android:layout_height="50dp"  
    android:text="X"  
    android:layout_marginRight="10dp"  
    android:background="@drawable/button_gamelevels"  
    android:textColor="@color/white"  
    android:textSize="24sp"  
    android:textStyle="bold"  
    android:gravity="center"  
/>
```

```
<TextView
```

```
    android:id="@+id/textView23"  
    android:layout_width="50dp"  
    android:layout_height="50dp"  
    android:text="X"  
    android:layout_marginRight="10dp"  
    android:background="@drawable/button_gamelevels"  
    android:textColor="@color/white"  
    android:textSize="24sp"  
    android:textStyle="bold"  
    android:gravity="center"  
/>
```

```
<TextView
```

```
    android:id="@+id/textView24"  
    android:layout_width="50dp"  
    android:layout_height="50dp"  
    android:text="X"  
    android:layout_marginRight="10dp"  
    android:background="@drawable/button_gamelevels"  
    android:textColor="@color/white"  
    android:textSize="24sp"  
    android:textStyle="bold"  
    android:gravity="center"  
/>
```

```
<TextView
```

```
    android:id="@+id/textView25"  
    android:layout_width="50dp"  
    android:layout_height="50dp"  
    android:text="X"  
    android:layout_marginRight="10dp"  
    android:background="@drawable/button_gamelevels"  
    android:textColor="@color/white"  
    android:textSize="24sp"  
    android:textStyle="bold"  
    android:gravity="center"
```

```
    />  
</LinearLayout>
```

```
<LinearLayout  
    android:id="@+id/line6"  
    android:layout_width="match_parent"  
    android:layout_height="wrap_content"  
    android:orientation="horizontal"  
    android:layout_marginTop="10dp">
```

```
<TextView  
    android:id="@+id/textView26"  
    android:layout_width="50dp"  
    android:layout_height="50dp"  
    android:text="X"  
    android:layout_marginRight="10dp"  
    android:background="@drawable/button_gamelevels"  
    android:textColor="@color/white"  
    android:textSize="24sp"  
    android:textStyle="bold"  
    android:gravity="center"  
/>
```

```
<TextView  
    android:id="@+id/textView27"  
    android:layout_width="50dp"  
    android:layout_height="50dp"  
    android:text="X"  
    android:layout_marginRight="10dp"  
    android:background="@drawable/button_gamelevels"  
    android:textColor="@color/white"  
    android:textSize="24sp"  
    android:textStyle="bold"  
    android:gravity="center"  
/>
```

```
<TextView  
    android:id="@+id/textView28"  
    android:layout_width="50dp"  
    android:layout_height="50dp"  
    android:text="X"  
    android:layout_marginRight="10dp"  
    android:background="@drawable/button_gamelevels"  
    android:textColor="@color/white"  
    android:textSize="24sp"  
    android:textStyle="bold"  
    android:gravity="center"  
/>
```

```
<TextView  
    android:id="@+id/textView29"  
    android:layout_width="50dp"  
    android:layout_height="50dp"  
    android:text="X"  
    android:layout_marginRight="10dp"  
    android:background="@drawable/button_gamelevels"  
    android:textColor="@color/white"  
    android:textSize="24sp"  
    android:textStyle="bold"  
    android:gravity="center"  
/>
```

```
<TextView  
    android:id="@+id/textView30"  
    android:layout_width="50dp"  
    android:layout_height="50dp"  
    android:text="X"  
    android:layout_marginRight="10dp"
```

```

        android:background="@drawable/button_gamelevels"
        android:textColor="@color/white"
        android:textSize="24sp"
        android:textStyle="bold"
        android:gravity="center"
    />
</LinearLayout>

</LinearLayout>
</ScrollView>
</LinearLayout>
</LinearLayout>
</RelativeLayout>

```

previewdialog.xml

```

<?xml version="1.0" encoding="utf-8" ?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:orientation="vertical"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:id="@+id/dialogdon"
    android:background="@drawable/new_preview_backgrounde_level1"
    android:layout_gravity="center"
    android:scaleType="centerCrop"
    android:layout_margin="10dp">
    <LinearLayout
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:orientation="vertical"
        android:padding="20dp"
        android:gravity="center">
        <TextView
            android:layout_width="40dp"
            android:layout_height="40dp"
            android:id="@+id/buttonclose"
            android:text="X"
            android:textStyle="bold"
            android:gravity="center"
            android:background="@drawable/button_stroke_black95_press_white"
            android:layout_gravity="end"
        />
        <ImageView
            android:layout_width="match_parent"
            android:layout_height="0dp"
            android:id="@+id/previewimg"
            android:src="@drawable/new_preview_img1"
            android:layout_weight="60"
            android:layout_marginTop="10dp" />
        <TextView
            android:layout_width="wrap_content"
            android:layout_height="0dp"
            android:id="@+id/textdescription"
            android:text="@string/levelone"
            android:layout_weight="30"
            android:textColor="@color/black95"
            android:textSize="22sp"
            android:textStyle="bold"
            android:layout_marginTop="10dp" />
        <Button
            android:layout_width="wrap_content"

```



```

        android:layout_height="0dp"
        android:id="@+id/buttoncontinue"
        android:background="@drawable/button_stroke_black95_press_white"
        android:layout_weight="10"
        android:text="@string/textcontinue"
        android:textColor="@color/black95"
        android:textAllCaps="true"
        android:textSize="22sp"
        android:textStyle="bold"
        android:layout_marginTop="10dp"
    />

```

```
</LinearLayout>
```

```
</LinearLayout>
```

universal.xml

```

<?xml version="1.0" encoding="utf-8" ?>
<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:id="@+id/img_Left"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:orientation="vertical">

    <ImageView
        android:id="@+id/background"
        android:layout_width="match_parent"
        android:layout_height="match_parent"
        android:scaleType="centerCrop"
        android:src="@drawable/new_back_grounde_level1" />

    <LinearLayout
        android:id="@+id/container1"
        android:layout_width="match_parent"
        android:layout_height="match_parent"
        android:gravity="center"
        android:orientation="vertical">

        <LinearLayout
            android:id="@+id/container2"
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:orientation="vertical"
            android:padding="10dp">

            <LinearLayout
                android:id="@+id/line1"
                android:layout_width="match_parent"
                android:layout_height="match_parent"
                android:orientation="horizontal">

                <Button
                    android:id="@+id/button_back_level1"
                    android:layout_width="wrap_content"
                    android:layout_height="wrap_content"
                    android:layout_marginRight="10dp"
                    android:background="@drawable/button_stroke_black95_press_white"
                    android:text="@string/back"

```

```
android:textAllCaps="false"  
android:textColor="@color/black95"  
android:textSize="22sp"  
android:textStyle="bold" />
```

```
<TextView  
    android:id="@+id/text_levels"  
    android:layout_width="wrap_content"  
    android:layout_height="wrap_content"  
    android:layout_weight="1"  
    android:gravity="right"  
    android:padding="10dp"  
    android:text="@string/level1"  
    android:textColor="@color/black95"  
    android:textStyle="bold"  
    android:textSize="22sp"
```

```
/>
```

```
</LinearLayout>
```

```
<LinearLayout  
    android:id="@+id/line2"  
    android:layout_width="match_parent"  
    android:layout_height="wrap_content"  
    android:layout_marginTop="10dp"  
    android:orientation="horizontal">
```

```
<TextView  
    android:id="@+id/point1"  
    android:layout_width="wrap_content"  
    android:layout_height="10dp"  
    android:layout_weight="1"  
    android:background="@drawable/style_points" />
```

```
<TextView  
    android:id="@+id/point2"  
    android:layout_width="wrap_content"  
    android:layout_height="10dp"  
    android:layout_weight="1"  
    android:background="@drawable/style_points" />
```

```
<TextView  
    android:id="@+id/point3"  
    android:layout_width="wrap_content"  
    android:layout_height="10dp"  
    android:layout_weight="1"  
    android:background="@drawable/style_points" />
```

```
<TextView  
    android:id="@+id/point4"  
    android:layout_width="wrap_content"  
    android:layout_height="10dp"  
    android:layout_weight="1"  
    android:background="@drawable/style_points" />
```

```
<TextView  
    android:id="@+id/point5"  
    android:layout_width="wrap_content"  
    android:layout_height="10dp"  
    android:layout_weight="1"  
    android:background="@drawable/style_points" />
```

```
<TextView
```

```

        android:id="@+id/point6"
        android:layout_width="wrap_content"
        android:layout_height="10dp"
        android:layout_weight="1"
        android:background="@drawable/style_points" />

        <TextView
            android:id="@+id/point7"
            android:layout_width="wrap_content"
            android:layout_height="10dp"
            android:layout_weight="1"
            android:background="@drawable/style_points" />
        <TextView
            android:id="@+id/point8"
            android:layout_width="wrap_content"
            android:layout_height="10dp"
            android:layout_weight="1"
            android:background="@drawable/style_points" />
        <TextView
            android:id="@+id/point9"
            android:layout_width="wrap_content"
            android:layout_height="10dp"
            android:layout_weight="1"
            android:background="@drawable/style_points" />
        <TextView
            android:id="@+id/point10"
            android:layout_width="wrap_content"
            android:layout_height="10dp"
            android:layout_weight="1"
            android:background="@drawable/style_points" />
    </LinearLayout>
    <LinearLayout
        android:id="@+id/line3"
        android:layout_width="match_parent"
        android:layout_height="match_parent"
        android:layout_marginTop="10dp"
        android:orientation="horizontal">
        <ImageView
            android:id="@+id/img_left"
            android:layout_width="match_parent"
            android:layout_height="320dp"
            android:layout_marginRight="5dp"
            android:scaleType="centerCrop"
            android:background="@drawable/style_bolvanka_universal"
            android:layout_weight="1"
            android:src="@drawable/new_bolvanka"
            />
        <ImageView
            android:id="@+id/img_right"
            android:layout_width="match_parent"
            android:layout_height="320dp"
            android:layout_marginRight="5dp"
            android:scaleType="centerCrop"
            android:background="@drawable/style_bolvanka_universal"
            android:layout_weight="1"
            android:src="@drawable/new_bolvanka"
            />
    </LinearLayout>
</LinearLayout>
</LinearLayout>
</LinearLayout>

```

ВІДГУК
керівника економічного розділу
на кваліфікаційну роботу бакалавра
на тему:
«Розробка комп'ютерної відеоігри для мобільного пристроя на платформі
Android»
студента групи 122-19-1 Стрюка Олега Андрійовича.

ПЕРЕЛІК ФАЙЛІВ НА ДИСКУ

Ім'я файлу	Опис
Пояснювальні документи	
Кваліфікація-робота (122-19-1 Стрюк О.А.).docx	Пояснювальна записка до кваліфікаційної роботи. Документ Word.
Кваліфікація-робота (122-19-1 Стрюк О.А.).pdf	Пояснювальна записка до кваліфікаційної роботи в форматі PDF.
Програма	
qualification_work_quiz.zip	Архів. Містить файли програми.
Презентація	
Презентація Стрюк.pptx	Презентація кваліфікаційної роботи.