

**ДЕРЖАВНИЙ УНІВЕРСИТЕТ ТЕЛЕКОМУНІКАЦІЙ**

**НАВЧАЛЬНО-НАУКОВИЙ ІНСТИТУТ  
ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ**

Кафедра інженерії програмного забезпечення

**ПОЯСНЮВАЛЬНА ЗАПИСКА**

до бакалаврської роботи

на ступінь вищої освіти бакалавр

на тему «РОЗРОБКА МОБІЛЬНОГО ДОДАТКУ ДЛЯ РОЗВИТКУ ТА  
ТЕСТУВАННЯ ЛОГІКИ ТА МАТЕМАТИЧНИХ ЗДІБНОСТЕЙ ПРОГРАМІСТА  
МОВОЮ C#»

Виконав: студент 4 курсу, групи ПД-42  
спеціальності 121 Інженерія програмного  
забезпечення

Івченко Д.Є.

Керівник Довженко Т.П

Рецензент \_\_\_\_\_

**ДЕРЖАВНИЙ УНІВЕРСИТЕТ ТЕЛЕКОМУНІКАЦІЙ**  
**Навчально-науковий інститут інформаційних технологій**

Кафедра Інженерії програмного забезпечення

Ступінь вищої освіти - «Бакалавр»

Спеціальність - 121 Інженерія програмного забезпечення

ЗАТВЕРДЖУЮ

Завідувач кафедри

Інженерія програмного забезпечення

\_\_\_\_\_ Негоденко О.В.

« \_\_\_\_ » \_\_\_\_\_ 2021 року

**З А В Д А Н Н Я**

**НА БАКАЛАВРСЬКУ РОБОТУ СТУДЕНТУ**

1. Тема роботи: «Розробка мобільного додатку для розвитку та тестування логіки та математичних здібностей програміста мовою С#»

Керівник роботи Довженко Т.П. к.т.н., доцент кафедри ІІЗ

затверджені наказом вищого навчального закладу від — «12» березня 2021 року №65.

2. Строк подання студентом роботи «01» червня 2021 року

Вхідні дані до роботи:

Розробка математичних і логічних завдань.

Алгоритм розробки мобільного додатку.

Науково-технічна література, пов'язана з розробкою мобільного додатку для навчання та тестування.

4. Зміст розрахунково-пояснювальної записки (перелік питань, які потрібно розробити).

4.1 Аналіз вимог до мобільного додатку.

4.2 Аналіз та порівняння існуючих додатків та програм для навчання та розвитку логіки та математики.

4.3 Дослідження програмних засобів для розробки мобільного додатку.

4 3. Перелік графічного матеріалу

1. \_\_\_\_\_

2. \_\_\_\_\_

3. \_\_\_\_\_

4. \_\_\_\_\_

5. Дата видачі завдання «19» квітня 2021

### КАЛЕНДАРНИЙ ПЛАН

№ з/п	Назва етапів бакалаврської роботи	Строк виконання етапів роботи	Примітка
1	Підбір науково-технічної літератури	19.04-23.04	Виконано
2	Аналіз вимог до мобільного додатку.	23.04-26.04	Виконано
3	Аналіз існуючих додатків з розвитку та тестування логіки та математики.	26.04-28.04	Виконано
4	Дослідження програмних засобів	28.04-30.04	Виконано
5	Моделювання об'єкту проектування	01.05-05.05	Виконано
6	Розробка функціоналу застосунку	05.05-15.05	Виконано
7	Вступ, висновки, реферат	15.05-17.05	Виконано
8	Розробка презентації застосунку	18.05	Виконано

Студент

Керівник роботи







## РЕФЕРАТ

### РОЗРОБКА МОБІЛЬНОГО ДОДАТКУ ДЛЯ РОЗВИТКУ ТА ТЕСТУВАННЯ ЛОГІКИ ТА МАТЕМАТИЧНИХ ЗДІБНОСТЕЙ ПРОГРАМІСТА МОВОЮ C#

Текстова частина бакалаврської роботи 56с, 36 рис, 15 джерел

Ключові слова: мобільний додаток, логіка, проектування задач, Android, Unity, C#.

*Об'єкт дослідження* – процес навчання та тестування логіки та математичних здібностей програміста.

*Предмет дослідження* – програмний додаток для навчання і розвитку логіки і математики.

*Мета роботи* – підвищення ефективності праці програміста шляхом тренування його логічних і математичних здібностей для покращення його роботи з використанням мобільного додатку.

*Методи дослідження* – методи створення математичних і логічних задач на основі готових або створених формул, методи структурного аналізу і проектування, методи створення програмного додатку.

Наукова новизна даної роботи полягає в наступному:

1. Розроблено алгоритм створення логічних задач для навчання програмістів.
2. Встановлено що використання мови C# та середовища розробки Unity це вдале рішення для створення мобільного додатку .
3. На основі результатів виконаних досліджень та створених алгоритмів було розроблено програмний додаток для тестування та навчання логіки та математики у програмістів.

Впровадження розробленого програмного додатку допоможе оптимізувати процес навчання логіки і математики для програміста за рахунок створених унікальних алгоритмів навчання і створених зашифрованих формул. Використання

програмного продукту передбачає використання мобільного пристрою на операційній системі Android.



## ЗМІСТ

ВСТУП.....	10
1. АНАЛІЗ І ХАРАКТЕРИСТИКА СТВОРЕННЯ ДОДАТКУ.....	12
1.1. Актуальність програми для розвитку логіки та математики .....	12
1.2. Огляд та аналіз існуючих додатків для розвитку логіки та математики.....	13
1.3. Аналіз платформи розробки .....	19
2. ВИБІР ПРОГРАМНИХ ЗАСОБІВ ДЛЯ СТВОРЕННЯ ДОДАТКУ .....	23
2.1. Вибір мови програмування.....	23
2.2. Вибір середовища розробки .....	26
2.3. Огляд середовища розробки Unity .....	32
2.4. Аналіз процесу створення додатку на Unity.....	37
3. ПРОЕКТУВАННЯ ТА СТВОРЕННЯ ДОДАТКУ.....	42
3.1. Огляд програм для створення макету додатку .....	42
3.2. Моделювання об'єкту проектування .....	48
3.3. Створення макету мобільного додатку .....	52
3.4. Створення мобільного додатку в Unity.....	54
4 ІНСТРУКЦІЯ ВИКОРИСТАННЯ МОБІЛЬНОГО ДОДАТКУ .....	63
4.1. Процес створення задач для додатку .....	63
4.2. Інструкція використання мобільного додатку.....	64
ВИСНОВКИ .....	66
СПИСОК ЛІТЕРАТУРИ .....	67
Додаток А .....	69
Додаток Б.....	71
Додаток В .....	80

## ВСТУП

*Об'єкт дослідження* – процес навчання та тестування логіки там математичних здібностей програміста.

*Предмет дослідження* – програмний додаток для навчання і розвитку логіки і математики.

*Мета роботи* – підвищення ефективності праці програміста шляхом тренування його логічних і математичних здібностей для покращення його роботи з використанням мобільного додатку.

*Методи дослідження* – методи створення математичних і логічних задач на основі готових або створених формул, методи структурного аналізу і проектування, методи створення програмного додатку.

У відповідності до поставленої мети визначено основні завдання дослідження:

- проаналізувати поточний стан задач;
- дослідити можливості використання середовища розробки Unity і мови програмування C#;
- розробити структуру даних;
- розробити програмний додаток під операційну систему android;
- встановити що створення скриптів за допомогою мови програмування c# повністю задовольняють вимогам;
- створити необхідні задачі для подальшої реалізації в навчанні та програмного продукту.

Практична значущість полягає в тому, що результати дослідження можуть бути використані для навчання програміста логіки та математики.

Наукова новизна даної роботи полягає в наступному:

- розроблено алгоритм створення логічних задач для навчання програмістів;
- встановлено що використання мови c# та середовища розробки unity це вдале рішення для створення мобільного додатку;

- на основі результатів виконаних досліджень та створених алгоритмів було розроблено програмний додаток для тестування та навчання логіки та математики у програмістів.

# 1. АНАЛІЗ І ХАРАКТЕРИСТИКА СТВОРЕННЯ ДОДАТКУ

## 1.1. Актуальність програми для розвитку логіки та математики

В наш час зростає актуальність програм, відео та ігор. Але вся інформацію яку можна отримати від інтернету, зазвичай є пасивною, яка не несе в собі розвиваючий характер. Наш мозок кожен день отримує безліч різноманітної інформації яка на нас впливає та змінює нас кожен день. Кожен користувач смартфона використовує свій девайс для перегляду фото, відео, та текстового виду інформації, що безумовно є великим плюсом для розвитку, але актуальність саме цільового розвитку логіки – це є запорука правильного розвитку мислення, логіки, і прийняття логічних і раціональних рішень на протязі всього життя. Зараз потрібно задумуватися над тим, що потрібно розвивати свою логіку, мати мислення, робити відбір, опираючись на попередній досвід.

Насамперед мобільні та комп'ютерні ігри, розробники роблять для того, щоб розважатися. Є дуже багато видів ігор. Але не всі вони зможуть розвинути знання, логіку та уміння мислити. Незалежно від віку – це може бути дитина яка ходить в садочок, чи доросла людина, кожному потрібно розвивати свої мислення.

Гра в якій є логічні задачі, зможуть розвинути логічні здібності, людина зможе більше і правильніше міркувати. Насправді, в таких іграх, не потрібні ніякі великі знання, там потрібно мати всього лише кмітливість. Мозок потрібно також стимулювати його роботу, тренувати, давати йому подумати. В цьому допоможуть саме ігри на логіку, адже саме вони є дуже ефективним засобом для розвитку логічного і просторового мислення, додатково до цього гра на логіку розвиває евристичне мислення - здатності знаходити рішення нетипових задач.

Логічні ігри дають хороший розвиток в математичних здібностях. Так же за такими іграми, можна нестандартно мислити, та виходи за рамки стандартних відповідей, ці вміння допоможуть в життєвих задач.

Мобільна гра на логіку, буде завжди з собою під рукою, коли не буде що робити, то цей час можна буде провести з користю для самого ж себе.

Математично-логічна гра - це гра в якій є задача, яку потрібно розв'язати, але ці задачі не прості, в них є скрите розв'язування. Потрібно знайти або зрозуміти метод вирішення задачі або прикладу.

## **1.2. Огляд та аналіз існуючих додатків для розвитку логіки та математики**

На даний момент створено не мало програм для розвитку логіки, серед них можна зустріти і застосунки які мають дуже безмежний спектр логічних задач в різних формах, наприклад, текстові логічні задачі, аудіо файлові, відео задачі, задачі на числах і тому подібних. Але більш досконаліші це саме програми, які мають простий інтерфейс, які дозволяють сконцентруватися, адже це є процес навчання, так як вирішуючи логічні задачі ми пристосовуємось до неординарного мислення для пошуку правильної відповіді. Додатків для розвитку математичних здібностей набагато менше, тому ще математичні задачі прирівнюють до більш слабких методів розвитку, але вони також є невід'ємними з задачами на логіку Застосунків даного напрямку не так багато, розглянемо основні з них, їхні переваги та недоліки для користувача[1].

Логічна гра LogicLike – це розвиваюча гра на логіку, розгадування задач, ребусів, математичні загадки та головоломки. Можна грати як на сайті у комп'ютері так і в програмі на мобільному телефоні.

Гра має декілька категорій, а саме:

- математичні ребуси з буквами та цифрами;
- правда чи брехня;
- завдання на просторове мислення;
- математика;
- різні настільні задачі, шахи;

- головоломки;
- sudoku, какуро.

В цю гру можуть грати як діти, так і дорослі, тому що в ній є категорії віку. Цікаві головоломки, загадки, математичні задачі, ребуси, вони допоможуть для розвитку мислення, розвинути інтелект пам'яті та розвитку дедукції. В них є три рівня тяжкості: новачок, опитний та експерт. Регулярно доповнюють рівні. Яркий інтерфейс програми, можна заробляти зірки, вигравати призи та отримувати нових героїв. Так же на цій платформі можна брати участь у олімпіаді та змагатися з іншими людьми і вигравати нагороди.

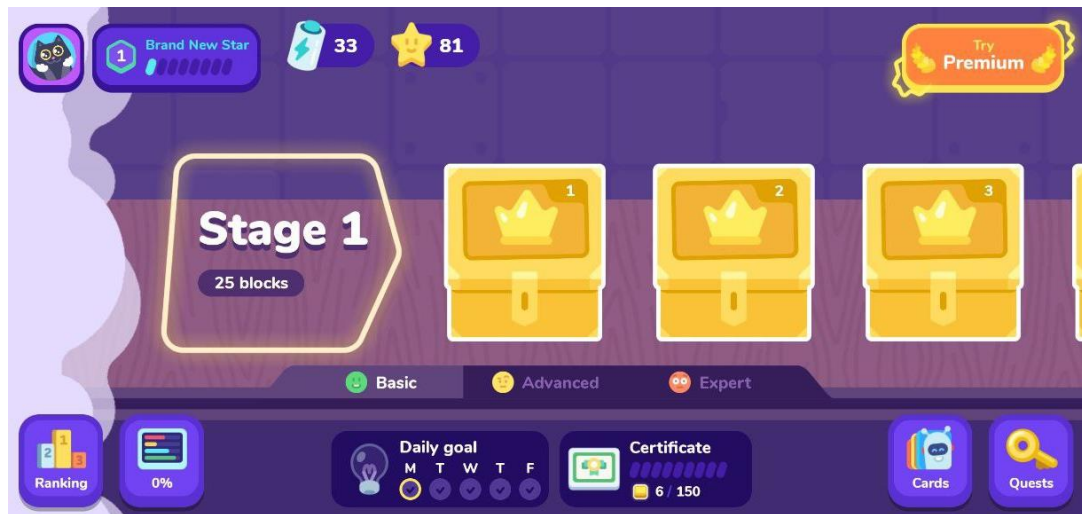


Рисунок 1.1. - Мобільний додаток logiclike

Гра має таку функцію, як грати з друзями, тобто змагатися з другом розумом. Завдяки такій грі можна провести цікаво та корисно час з друзями або наодинці. Плюси даної гри це насамперед різноманіття рівнів з різними видами логіки, а також можливість вирішувати задачі незалежно від девайсу, гра доступна на платформі Android, IOS, а також на комп'ютері з використання будь якого браузеру.

Math Riddles – Популярна гра на логіку, яка презентує користувачеві простий інтерфейс з обмеженим в кількості рівнями для розвитку задачами, які формуються на

основі математичних дій та задач, які необхідно вирішити методом підбору або рішення. Гра включає в себе суміш математичних і логічних задач тільки з числами, але більшість рівнів має ідентичний алгоритм, що для проходження всієї гри достатньо розуміння логіки декілька десятків вирішених головоломок. Гра на англійській мові, Але інтерфейс програми дозволяє інтуїтивно розуміти що до чого.

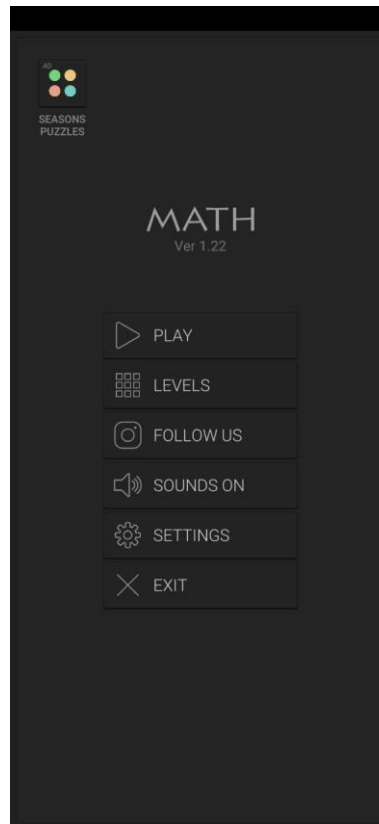


Рисунок 1.2. - Мобільний додаток Math Riddles

Brain Out – мобільний додаток для розвитку логіки та нестандартного мислення з методом пошуку речей та впровадження механік можливостей телефону. Гра створена на основі вивчення логіки нестандартними способами включаючи в себе підходи адаптивного пошуку відповіді, тобто задачі представленні у вигляді активної фотографії на пошук потрібних речей, цей додаток є основоположником ігор для розвитку пошукових здібностей та нестандартного методу пошуку правильного рішення.

Наприклад щоб знайти необхідних предмет потрібно не тільки взаємодіяти за переметами в додатку, але ще включати фізичні фактори взаємодії, такі як перевернути телефон, чи дмухати на нього. Але не вада цього підходу полягає в тому що більшість задач не має логічного зв'язних деталей, що викликає дуже великий резонанс серед користувачів які не бачать сенс в багатьох не логічних задачах, які не мають в собі фундаментально правильної відповіді

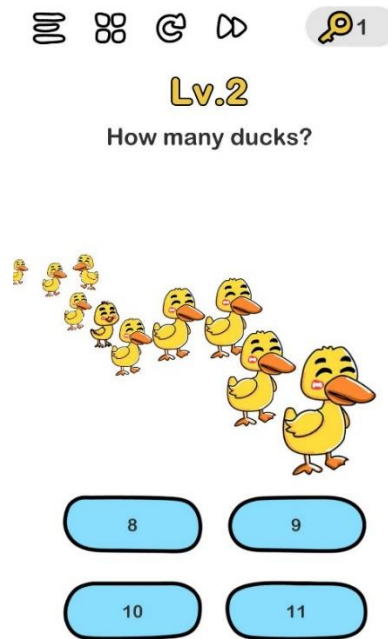


Рисунок 1.3. - Мобільний додаток Brain Out

Matches puzzle game -логічна, класична головоломка зі сирниками. Щоб пройти рівень, потрібно видаляти, двигати чи доповнювати сирники, потрібно сирники переміщати так, щоб вийшла потрібна кількість квадратів або трикутників, поставити, забрати чи перемістити правильно сирник так, щоб вийшла правильна відповідь. Ця гра захоплююча, має нестандартне мислення, розвиває моторику уважності.



Графіка додатку на високому рівні, потягне навіть на слабкий телефон і буде працювати стабільно. В грі більше тисячі головоломок та десять епізодів. Також можна зрівняти ваш рейтинг з іншими гравцями.



Рисунок 1.4. - Мобільний додаток Matches puzzle game

В грі matches puzzle game є два види головоломок: математичні головоломки та фігурні головоломки. Якщо не виходить пройти рівень, то можна використати підказку. Щоб грати в цю гру потрібно мати гарну кмітливість.

Всі представлені варіанти програм є дуже популярні серед своєї категорії, мають свої унікальні задачі, які перетинаються з розвитком не тільки логіки, а також тренування мислення, неординарний підхід до вирішення задач і створення нового процесу аналізу задач, але і мають свої недоліки. Представлена таблиця показує основні плюси і мінуси даних програм. Для перегляду підказок, або якщо ви хочете переглянути наступний рівень, потрібне підключення до інтернету.

Таблиця 1.1 – Порівняння додатків для розвитку логіки та математики

Назва додатку	Переваги	Недоліки
logiclike	<ul style="list-style-type: none"> <li>- велика кількість рівнів;</li> <li>- багатоплатформність (Android, IOS, PC);</li> <li>- єдиний обліковий запис для збереження прогресу;</li> <li>- гра з друзями.</li> </ul>	<ul style="list-style-type: none"> <li>- недоступність класифікації рівнів;</li> <li>- потрібне інтернет підключення;</li> <li>- реклама;</li> <li>- умовно безкоштовна.</li> </ul>
Math Riddles	<ul style="list-style-type: none"> <li>- зрозумілий інтерфейс;</li> <li>- математичні і логічні задачі.</li> </ul>	<ul style="list-style-type: none"> <li>- обмежений запас рівнів;</li> <li>- відсутність роз'яснення задач,</li> <li>- реклама;</li> <li>- тільки на Android.</li> </ul>
Brain Out	<ul style="list-style-type: none"> <li>- нестандартні задачі на логіку;</li> <li>- неординарний підхід використання телефону;</li> <li>- велика кількість рівнів.</li> </ul>	<ul style="list-style-type: none"> <li>- задачі не мають логічних відповідей;</li> <li>- одноманіття наступних задач;</li> <li>- реклама і платний функціонал підказок.</li> </ul>
Matches puzzle game	<ul style="list-style-type: none"> <li>- головоломка на логіку мають широкий спектр задач;</li> <li>- два види логічних задач.</li> </ul>	<ul style="list-style-type: none"> <li>- інтерфейс містить в собі дуже багато посилань на сторонні ресурси;</li> <li>- реклама;</li> <li>- відсутність деяких функцій (пропустити рівень, отримати підказку).</li> </ul>

### 1.3. Аналіз платформи розробки

Розглянувши всі існуючі додатки для розвитку логіки математики і пам'яті на просторах інтернету, а саме для мобільних пристроїв, програми для платформи Android розміщені в Play Market, також для IOS Для пристроїв Apple, можна зробити висновки, що всі програми мають або одноманітні задачі, вирішивши декілька таких задач розумієш логіку всіх подальших рівнів, або мають велику кількість відволікаючих факторів, таких як реклами, досягнення, непотрібна інформація. Всі ігри мають спільну проблему, ще складний інтерфейс і відсутність моменту навчання, тобто відсутність концентрації на вирішення поставленої задачі, яку дають користувачу[2].

Порівняння Google Android та Apple iOS – це операційні системи, перш за все мобільних технологій.

В наш час Android це найпопулярніша платформа для смартфонів в усьому світі, їм користуються велика кількість моделей смартфонів. iOS використовується лише на одній платформі Apple, а саме iPhone.

#### Дисплей

В сучасних телефонах Android та iOS використовують дисплеї IPS LCD.

#### Камери

Насправді в наш час роблять Android камери не гірше ніж на iOS. Але все рівно iPhone камера має свої плюси, так як і Android.

Цінова категорія програм Android та iOS майже не відрізняються один від одного, але складність проходження сертифікації для пристроїв IOS робить майже всі проекти та програми платними для користувачів Iphone, тому платформа Android має більше програм і додатків.

Насправді коли завантажуєш якусь програму, то хочеться спочатку протестувати її, але в iOS такого немає, є базові програми безкоштовні та все. В Android все навпаки, більшість програм в цій операційній системі безкоштовні.

## Оновлення

В iOS кожний пристрій отримує п'ять років постійного оновлення операційної системи, а в Android навпаки всі оновлення операційної системи проходить дуже повільно, із – за того, що деякі пристрої використовують модифіковану версію операційної системи.

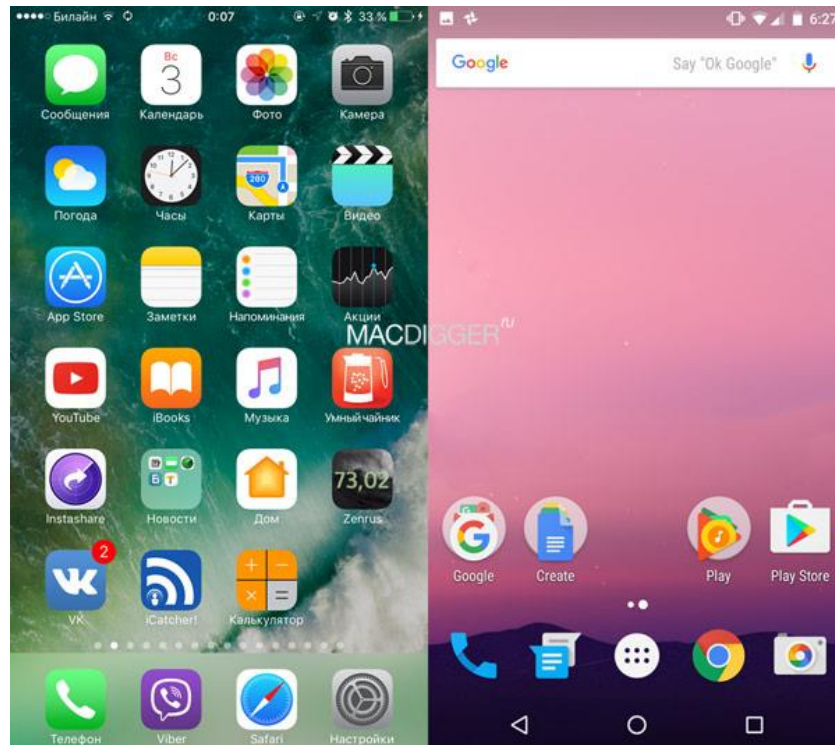


Рисунок 1.5. - Інтерфейс Android і iOS

### Висновок порівняння:

1. iOS
  - a. чудова продуктивність роботи;
  - b. довгострокові оновлення операційної системи;
  - c. дуже добра цінність в довгостроковій перспективі;
  - d. високоякісні додатки та магазин додатків.
2. Android
  - a. налаштовується інтерфейс;

- b. велика різноманітність програм;
- c. набагато більше вільних рухів;
- d. більш доступніший телефон.

Порівняння Android та iOS в кожній операційній системі є свої плюси та недоліки. Але саме iPhone пропонує кращу продуктивність, та довше прослужать.

Для закріплення потенційного розвитку, потрібне систематичне тренування логіки, але задачі, які подаються повинні мати незалежний характер, тобто не повинні повторюватись, це дозволяє зробити процес навчання унікальним. Створений додаток має назву «Math and Logic» Використання різнопланових задач дає можливість вивчати логіку всім користувачам додатку студентам навчальних закладів, школярів, а також програмістів, для яких необхідно розвивати свої логічні здібності, тому що логіка – це невід’ємна частина будь якого напрямку в програмуванні.

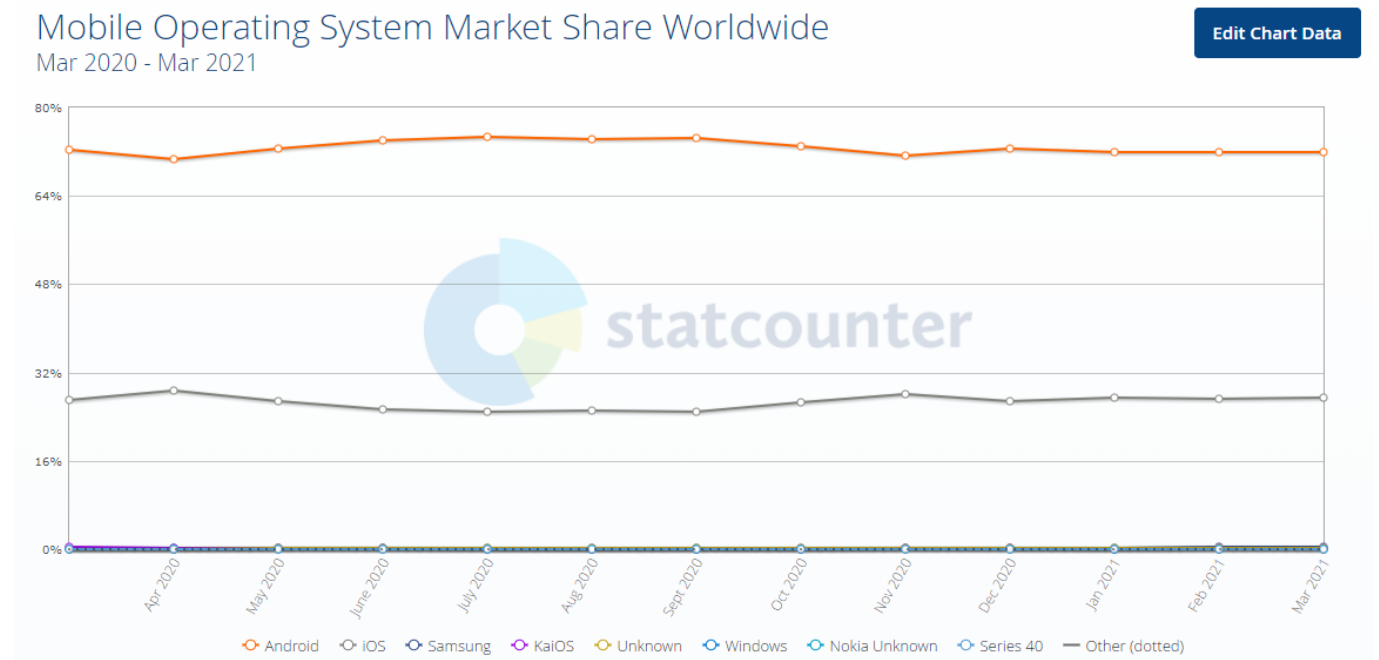


Рисунок 1.6. - Таблиця використання ОС в світі

Головна особливість розроблюваного додатку – це можливість отримати не лише теоретичну базу знань з логіки и розв’язування задач, а і застосовувати їх в

повсякденному житті, використовуючи алгоритми дослідження та аналізу поставлених задач, які тісно пов'язані з роботою програміста, адже процес програмування включає дуже багато сфер діяльності, такі як математична, розумова, логічна сфера діяльності.

Вибираючи платформу, на якій можна створювати додаток я використовував ресурс[3], який показав що розробка під операційну систему Android охопить найбільшу аудиторію користувачів, що зможе використовувати програмний додаток. IOS система вважається закритою системою, що не дозволяє встановлювати сторонні додатки на пристрої, і для того, щоб встановити будь який сторонній додаток, йому необхідно пройти спеціальну сертифікацію для вільного доступу.

Статистика користувачів мобільних пристроїв на квітень 2021 року виглядає наступним чином:

- Android - 71.83%;
- iOS - 27.41%;
- інші платформи мають менше 0.5 відсотків долі ринку.

Спираючись на дану інформацію можна зробити висновок що розробка яка буде зроблена на платформі Android буде доцільною розробкою, що охопить велику кількість аудиторії.

## 2. ВИБІР ПРОГРАМНИХ ЗАСОБІВ ДЛЯ СТВОРЕННЯ ДОДАТКУ

### 2.1. Вибір мови програмування

Розробка програмного додатку під Android це дуже нелегкий процес, охоплює в собі декілька алгоритмів створення додатку, а саме графічно частина і сам код. На даній момент існують певні популярні і зручні мови, які дуже доцільно підходять для програмування додатку для мобільних пристроїв, а саме Java, C#.

```

1 package com.rabbitmq.consumers;
2
3 import com.rabbitmq.client.Channel;
4 import com.rabbitmq.client.Connection;
5 import com.rabbitmq.client.ConnectionFactory;
6 import com.rabbitmq.client.DeliverCallback;
7
8 public class ConsumerA
9 {
10     private final static String QUEUE_NAME = "Queue_PC_A";
11
12     public static void main(String[] argv) throws Exception
13     {
14         ConnectionFactory factory = new ConnectionFactory();
15         factory.setHost("localhost");
16         Connection connection = factory.newConnection();
17         Channel channel = connection.createChannel();
18
19         channel.queueDeclare(QUEUE_NAME, false, false, false, null);
20         System.out.println(" [*] Waiting for messages. To exit press CTRL+C");
21
22         DeliverCallback deliverCallback = (consumerTag, delivery) -> {
23             String message = new String(delivery.getBody(), "UTF-8");
24             System.out.println(" [x] Received '" + message + "'");
25         };
26         channel.basicConsume(QUEUE_NAME, true, deliverCallback, consumerTag -> { });
27     }
28 }

```

Рисунок 2.1. - Приклад синтаксису Java

Java – це об'єктно орієнтована мова програмування, який використовується як універсальна мова програмування, яка може виконувати майже всі задачі. Архітектура цієї мови програмування унікальна, вона не залежить від платформи розробки, можна створювати програми, під різні платформи і задачі. Наприклад розробка додатків під управлінням Android, промислові програми,

програми для комп'ютерів і навіть веб сервіси. На мові Java можна програмувати програми під андроїд, використовуючи тільки код і компілятор[8].

C# – це об'єктно орієнтована мова програмування, відноситься до роду «С» подібних мов програмування, синтаксис його дуже схожий с мовою програмування Java. Невідмінну від попередньої мови програмування ця мова не є крос платформною, але також за допомогою її можна виконувати багато задач, особливо великих програмних систем, веб застосунків, а також мобільних і десктопних додатків[4].

```

6 public class DemoScript: MonoBehaviour {
7
8     public Light myLight;
9
10    void Awake () {
11        int myVar = AddTwo(9,2);
12        Debug.Log(myVar);
13    }
14
15    void Update () {
16        if (Input.GetKeyDown ("space")) {
17            MyFunction ();
18        }
19    }
20
21    }
22
23    void MyFunction () {
24
25        myLight.enabled = !myLight.enabled;
26    }
27
28    int AddTwo (int var1, int var2) {
29        int returnValue = var1 + var2;
30        return returnValue;
31    }
32    }
33
34 }

```

Рисунок 2.2. - Приклад синтаксису C# в Unity

Порівнюючи синтаксис двох мов програмування можна побачити, що код на C# вигадає більш компактним, функціонал більш ширший і використання методів об'єктно орієнтовного програмування реалізується більш доцільніше ніж в Java.

Також для збереження інформації необхідно використовувати JSON – це формат збереження даних текстового характеру. Повна аббревіатура розшифровується як JavaScript Object Notation. Незважаючи на походженні від Java цей формат запису даних є повноцінною і незалежною мовою програмування. Це



допоможе зберігати прогрес, до якого дійшов користувач програми і записувати необхідну інформацію.

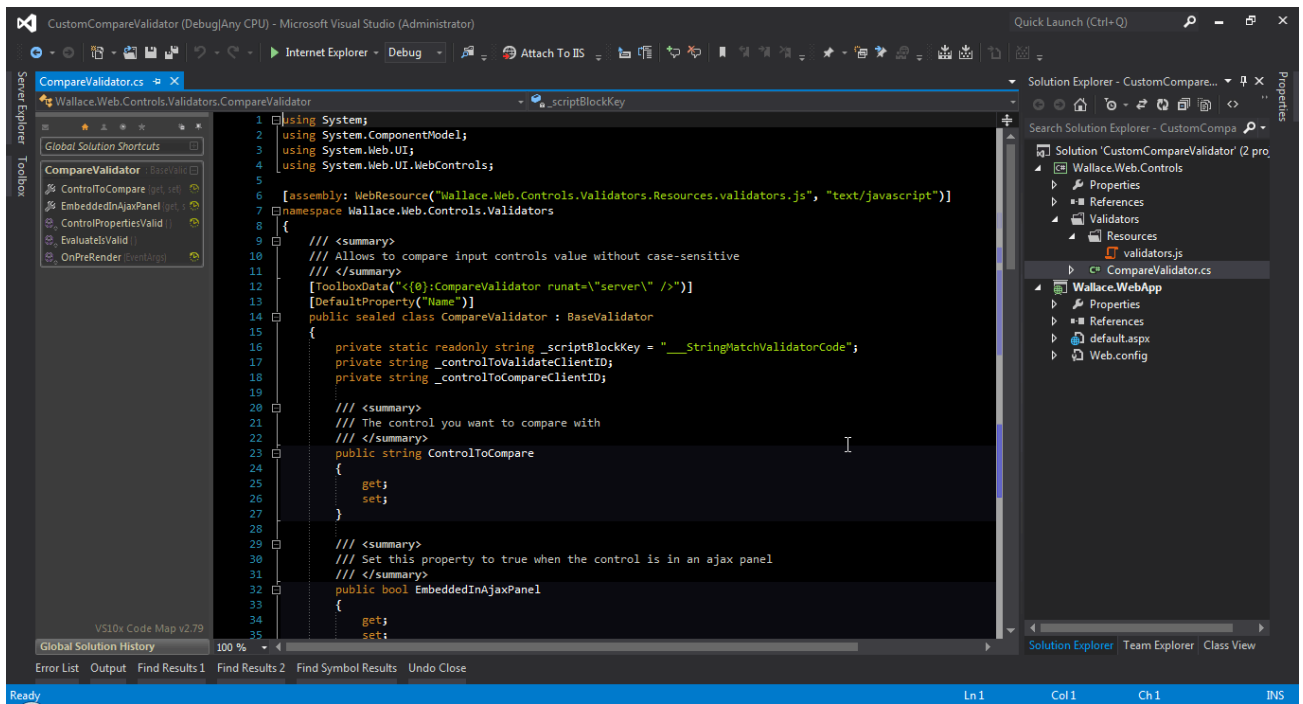


Рисунок 2.3. - Середовища розробки Microsoft Visual Studio

Завдяки детальному вивченню інформацію про мови програмування і створення додатку було обрана мова програмування C#, яка має дуже велику підтримку багатьма програмами і середовищами розробки. Використання середовища розробки Microsoft Visual Studio це доцільне рішення, так як лінійка продуктів даної програми дає можливість не писати код, а ще створювати консольні додатки, ігри, програми, також включає в себе редагування вихідного коду з повною можливістю подальшого рефакторинга. Дуже зручний і потужний інструмент для будь якого розробника, так як містить в собі безліч інструментів а саме містить конструктор форм для створення додатків з графічним інтерфейсом, конструкторів баз даних, веб дизайн і розробка[9].

Створення мобільних додатків і програм має сенс, так як в наш час майже кожна людина має смартфон, тому велика аудиторія Android користувачів мають можливість встановлювати прості за об'ємом пам'яті ігри і додатки, в той час як комп'ютерні ігри

потребують набагато більше місця на жорсткому диску, так як більш складна розробка графіки, що потребує багато місця. Але існують мінуси створення додатку під телефон це розмір екрану який не перевищує 5-6 дюймів. Але це є водночас великим плюсом, так як при розробці програмного продукту іде акцент на налаштування алгоритмів дії, і працездатності ресурсів самого додатку потрібно менше часу витратити на налагодження готового програмного продукту.

Є також інші мови програмування, але вони не підходять під конкретно поставлені задачі, наприклад мовою програмування Swift можна розробляти мобільні додатки але виключно тільки під операційну систему IOS і macOS. На мові програмування PHP і JavaScript можна створювати лише браузерні ігри які повноцінно працюють лише на великих моніторах комп'ютерів або ноутбуків.

## **2.2. Вибір середовища розробки**

Android Studio призначена для розробки мобільних додатків. Використовують мову програмування Java. Дана середовище розробки призначена для Windows, macOS та для Linux. На цій платформі не можна повністю відключити автозбереження документа, це є один із плюсів даної програми. Багато нових функцій які з'являються з новими версіями Android Studio які додають нові класи і методики розробки мобільних додатків.

Параметри Android Studio являються великий набір функцій, можна поставити зручний процес розробки, також є редактор коду, з яким легко працювати, достатньо велика бібліотека з готовими файлами та шаблонами для розробки ПО – програмного забезпечення, перевірка помилок.

Властивістю програми є те, що в неї вбудований емулятор, завдяки цьому, розробник може перевірити, як його програма буде виглядати на екранах різних пристроїв. В останній версії може розібратися навіть не початківець Ця програма дуже

зручна за роботою. В програмі зроблені всі сучасні методи для пакування коду та його маркування.

Не дивлячись на те, що в програмі є емулятор, для нього потрібний сильний персональний комп'ютер. Під час створення програми або утиліту, користувач Android Studio може дивитися за змінами проекту в реальному часі.

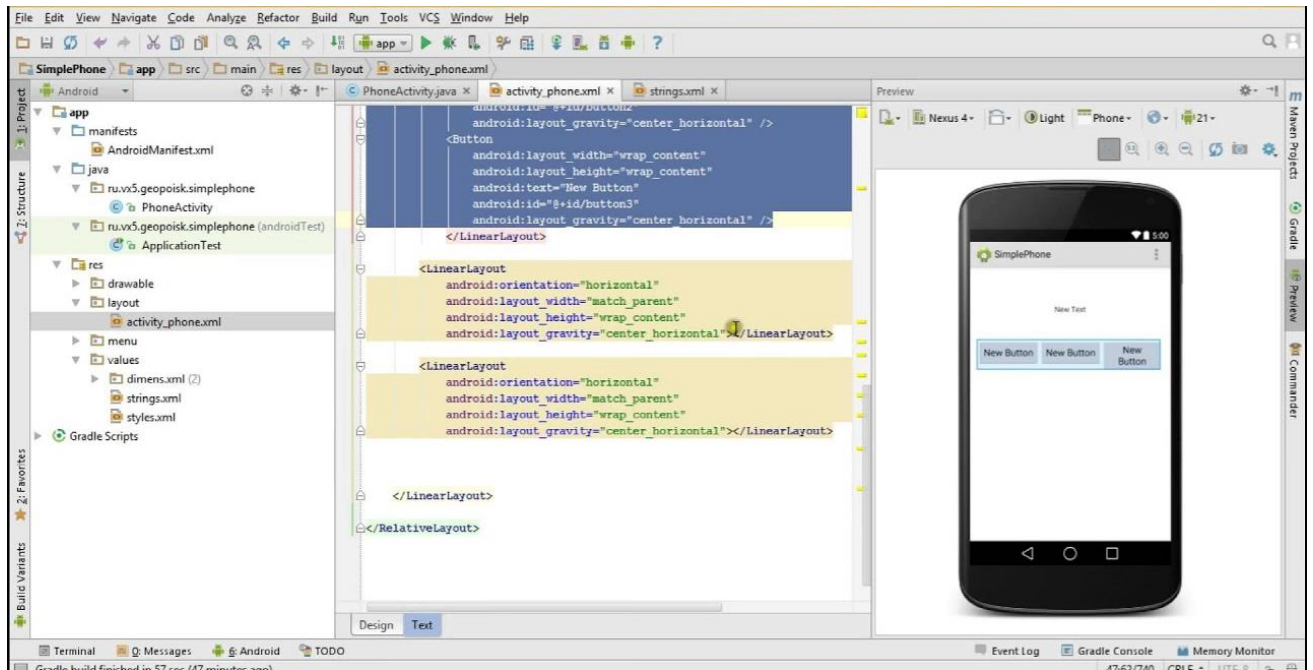


Рисунок 2.4. - інтерфейс середовища Android Studio

Недоліком даного програмного продукту є те, що кожний модуль розробляється індивідуально під потреби користувача, відсутність налагодження елементів дизайну шляхом технології drag and drop, це дає деякі незручності для створення, і змушує розробника створювати Android додаток виключно написанням коду. Цей спосіб важкий і не виправданий відносно проекту який я створюю.

Створення мобільного додатку за допомогою Android Studio передбачає знання мови програмування Java. Також при створенні мобільного додатку виникають складнощі при вивченні бібліотек які використовуються разом з середовищем розробки.

Unreal Engine – популярна платформа для створення ігор. Програмне забезпечення дозволяє створювати ігри для більшості платформ та операційної системи, також можна використовувати на різних портативних пристроїв. Ця платформа написана мовою програмування C++.

Для роботи з простими та нерухомими елементами ігрового простору, використовують двійний простір, всі вони поділяються на пуста та заповнене. Відкритий код дозволить додати свої файли, завдяки цьому майже немає обмежень по створенню гри. Ця платформа дозволяє розробникам відтворювати будь які ідеї в створенню і прокатуванню ігор и додатків будь якої складності і масштабу.

Щоб зробити на цьому движку гру, потрібно мати знання та навички, якщо не знати основної мови програмування C++, то навіть не потрібно братися за неї. Так на цій платформі є багато можливостей, багато функцій, інструментів, але вони всі не допоможуть, якщо не знати самим мови програмування.

Є багато плюсів: тут багата кількість інструментів, багато відео уроків, великий рівень графіки, майже немає обмежень по створенню гри та відкритий код програмування.

З мінусів: якщо слабкий комп'ютер то може не потягнути, важкий інтерфейс, в якому прийдеться довго вивчати та для початківця не підходить.

В Unreal Engine є багато версій:

- Unreal Engine 1;
- Unreal Engine 1.5;
- Unreal Engine 2;
- Unreal Engine 2.5;
- Unreal Engine 2 Runtime;
- Unreal Engine 2X;
- Unreal Engine 3;
- Unreal Development Kit (UDK);

- Unreal Engine 3.5;
- Unreal Engine 4.



Рисунок 2.5. - інтерфейс Unreal Engine

Важливо те, що всі вони використовуються і оновлюються незалежно один від одного, тому якщо створювати проект на одній версії, то при переході на іншу версію можуть виникнути складнощі з несумісністю даних версій.

Кожна версія має свої унікальності, що дає змогу розробникам комп'ютерних і мобільних вибрати саме ту версію, під якою буде реалізований максимальний потенціал майбутньої гри.

Також для ігор по мережі підтримують технології, можуть включати в себе не більше шести десяти чотирьох гравців водночас. Перше середовище розробки було створено з підтримкою лише двох платформ: Windows та Macintosh.

Складність та направленість даної програми в сторону розробки десктопних додатків, що не дозволяє повноцінно зробити на даному програмний додаток для операційної системи Android.

Платформа для розробки ігор Unity, дозволяє створювати програми більше ніж на двадцяти п'яти платформах таких як комп'ютері , приставках, смартфоні та планшетах. В Unity є багато різних компонентів, но за допомогою скриптів можна створити самим компоненти. Компоненти на основі скриптів дозволяють запускати ігрові події, використовувати фізичні властивості, перевіряти предмет на об'єкт зіткнень[5].

Перевага програми – це велика кількість бібліотеки асетів та плагінов, завдяки ним можна прискорити роботу про розробку ігор. Так же їх можна імпортувати на експортувати, добавляти в гру вже готові рівні та героїв, робити фото реалістичну графіку.

Unity доступний вже безкоштовно але так же є і платні деякі моменти.

Але навіть з тим, що він такий гарний та зручний, звичайно ж є і мінуси, для того щоб створити складнішу гру на цій платформі, потрібно мати навиків та знання з мови програмування C#. Якщо сцени будуть тяжкі, там буде дуже багато компонентів, то саме із - за цього, гра може бути повільною.

Unity гарно підходить для початківців. Unity дозволяє швидко створювати об'єкт, розташувати та зв'язати компоненти між собою. Цей движок має дуже велику аудиторію користувачів, при цьому якщо будуть якісь питання, як, що і де зробити, то це можна знайти на сайті, саме тому ця платформа буде корисна навіть новачку.

Із мінусів Unity є те, що не має підтримки, посилань на зовнішню бібліотеку, програмісту прийдеться налаштовувати самостійно. Але якщо у вас не має навиків програмування, то вам прийдеться навчитись. Не можна доповнювати в движок сторонню фізику, але можна створювати власну методом математичного прорахунку і надання їм необхідних параметрів

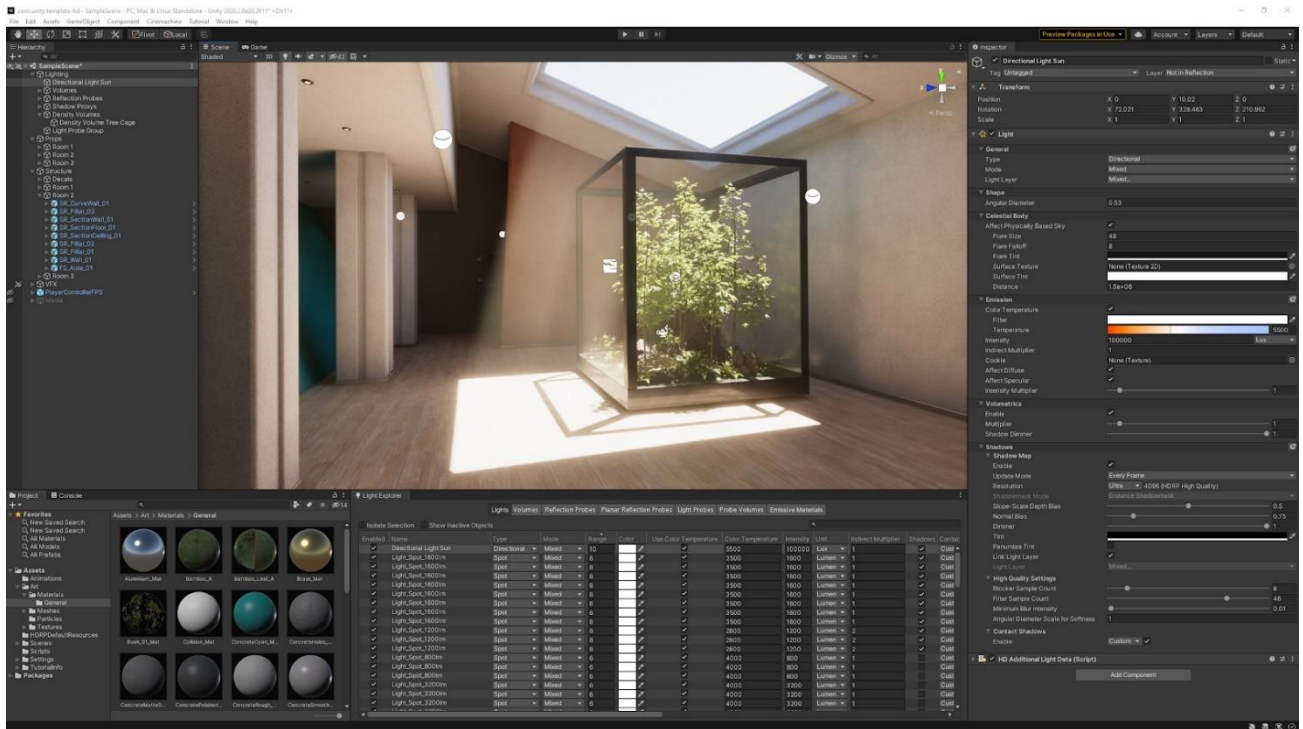


Рисунок 2.6. – робоче вікно Unity

Також на Unity можна робити не тільки комп'ютерні та мобільні ігри, а створювати анімований фільми та мультфільми. Редагувати роботу в Unity можна навіть під час перевірки програми, двигати елементи можна при запусненій грі. Також є модульна система компонентів, яка використовується для конструювання ігрових об'єктів.

Процес створення 2д гри поділяється на два основних етапи: створення дизайну через Unity та написання коду через мову програмування C#[6].

В Unity є офіційний сайт, де можна перевірити правильність написання свого коду. А також інтеграція з Microsoft Visual Studio дозволяє підключати бібліотеки Unity, що дозволяє проводити рефакторинг коду а також отримувати підказки та автоматичну підміну коду в випадках помилок. Зручний інтерфейс програми дозволяє створювати програму навіть без знань в програмуванні, використовуючи систему створення потрібних об'єктів і змінюючи її шляхом можливостей програми. Для створення потрібних взаємодій між об'єктами на сцені використовують спеціальні C#

скрипти, які дають можливість взаємодіяти між об'єктами так структурами програми. Також підтримка проекту та створення резервних копій на Git дозволяє одночасно створювати декількома розробниками один проект[7].

### 2.3. Огляд середовища розробки Unity

Unity – це програма для створення ігор, яка розроблена американською компанією Unity Technologies. За допомогою цього движка, можна створити ігри, які будуть запускатися на персональному комп'ютері, мобільному телефоні та приставці.

Якщо не проводити налаштування вікна програми, то за замовчуванням воно виглядає наступним чином зображеним на рис. 2.7.

В центрі головне вікно гри або сцена, в якому можна, змінити або перемістити об'єкт. Також міститься вкладка емулятора гри, на якому можна витрати розмір ігрового екрану і також орієнтацію ( вертикальну чи горизонтальну). Центральне вікно також містить магазин плагінів і текстур а також вікно створення анімації і сполучення анімацій між собою.

Знизу консоль для виведення багів та помилок, також знизу знаходяться системні файли для створення гри. Знизу лівого боку показуються всі файли сцен, файли програмного коду текстур і всього що пов'язано з програмним додатком. Також можна віднайти додатково встановленні пакети даних, що використовує Unity для роботи зі створенням майбутнього програмного додатку. В лівому верхньому вікні відображаються всі об'єкти які знаходяться на сцені, на цьому місці можна працювати з об'єктами, копіювати їх, переміщати та міняти вид. при натисканні на будь який об'єкт відкривається меню інспектора, за яким можна додавати властивість вибраного об'єкту. Наприклад додавання скриптів, фізики, методів взаємодій, параметрів розміру, анімації.



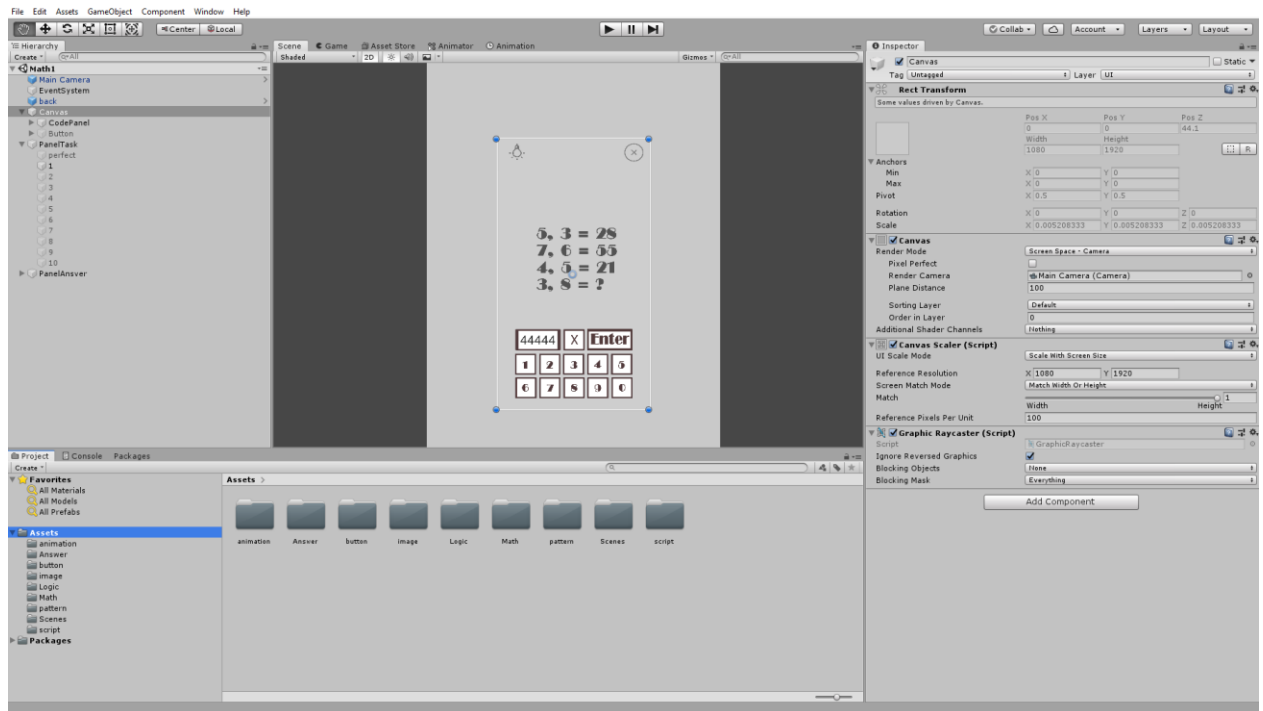


Рисунок 2.7. - Головний інтерфейс програми

Unity – це багато платформна середовища розробки програмних додатків і ігор, тому вона абсолютно універсальна і не залежить від проектів та операційних систем під яку створюється графічне програмне забезпечення[5]. Перелік популярних ОС, під які використовується програма Unity для створення програмних продуктів:

- Android;
- IOS;
- macOS;
- Windows;
- PS4;
- TvOS;
- Web;
- Xbox;
- Linux.

Але саме популярних платформ і зручних для розробки.

В меню підпунктів, ми можемо побачити вікна:

1. File.
2. Edit.
3. Assets.
4. GameObject.
5. Component.
6. Window.
7. Help.

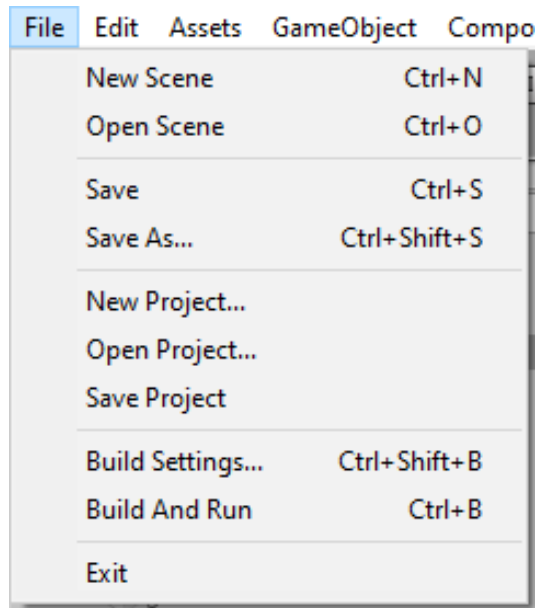


Рисунок 2.8. - Перелік функцій у вкладці File

Вкладка «Edit» - призначена, щоб виконувати прості операції з програмним кодом. В цій вкладці є такі функції: програмний код чи будь який елемент сцени можна вирізати, вставити, скопіювати та видалити у виділеному відрізці, також певний відрізок можна реверсувати, дублювати та відсортувати. Також можна запустити емулятор гри, натиснути на паузу і вибрати усі типи відтворення графіки, які є доступними на вибраній сцені. Можна додавати до проекту нові файли і створювати нові сцени або відкривати старі, зберігати зміни в проекті, також можна

створити новий проект, збереження проекту з подальшим створенням його під вибрану платформу операційну систему розробки, а далі експортувати файл у форматі HTML для веб застосунків, EXE для формату комп'ютерних ігор, APK файл для андроїд версій, інші. А також вихід з програми.

Вкладка «Assets» - призначена для створення компонентів, які представляють собою, графіку, звукові файли та скрипти, а також все інше що пов'язана за будь яким впливом чи взаємодію між об'єктами на просторі створення мобільного додатку.

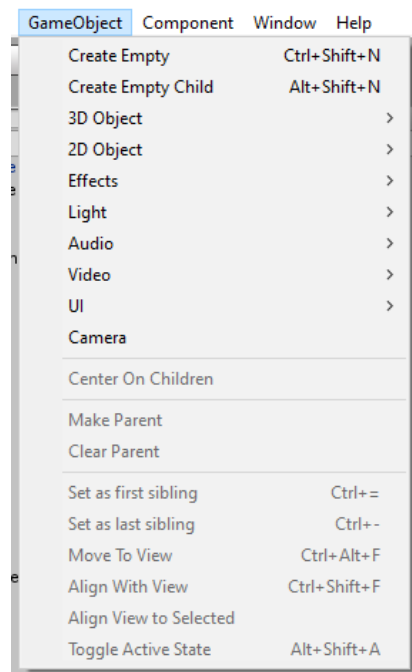


Рисунок 2.9. - Перелік функцій вкладки GameObject

Вкладка «GameObject», Рис.2.9. - це вкладка створення усіх можливих об'єктів і ефектів в Unity, а саме:

- трьох вимірні об'єкти( куб, сфера, капсула, циліндр, простір);
- двовимірні об'єкти;
- візуальні ефекти;
- світло;
- аудіо;

- відео;
- елементи інтерфейсу( текст, кнопки, панелі, перемикачі).

Вкладка «Component» - він визначає поведінку об'єктів у грі. Через цю функцію можна додавати, редагувати, копіювати, вставляти, перевіряти властивості, скинути, видалити компоненти.

Вкладка «Window», Рис.2.10. - призначена для налаштування активних вікон, та для редагування всіх вікон. Створення вікна користувача редактора включає в себе наступні дії:

- створення скрипта;
- використання коду для включення відображення вікна;
- реалізація коду для інструменту.

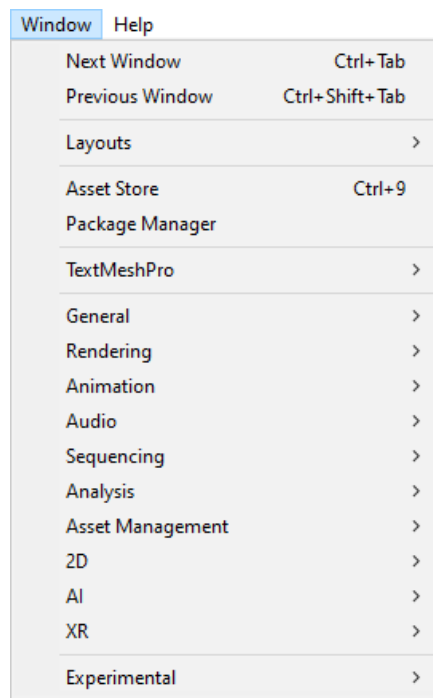


Рисунок 2.10. - Вкладка window

Вкладка «Help» - призначена для відкриття веб-сторінки з документацією, саме там програміст зможе знайти відповідь на те чи інше питання. Також у цій вкладці є опція онлайн запиту, який може дати програміст прямо на лінію підтримки.

## 2.4. Аналіз процесу створення додатку на Unity

Для створення власного програмного продукту на Unity спочатку необхідно зареєструватися та отримати необхідний перелік програм, а саме потрібний пакет файлів для створення Android додатку, додаткові бібліотеки і об'єкти які будуть в подальшому використовуватись. Для коректного створення додатку також важливо встановити наступні програми:

1. Microsoft visual studio з компонентами C#.
2. Android Studio для підтримки створення проекту під Android.

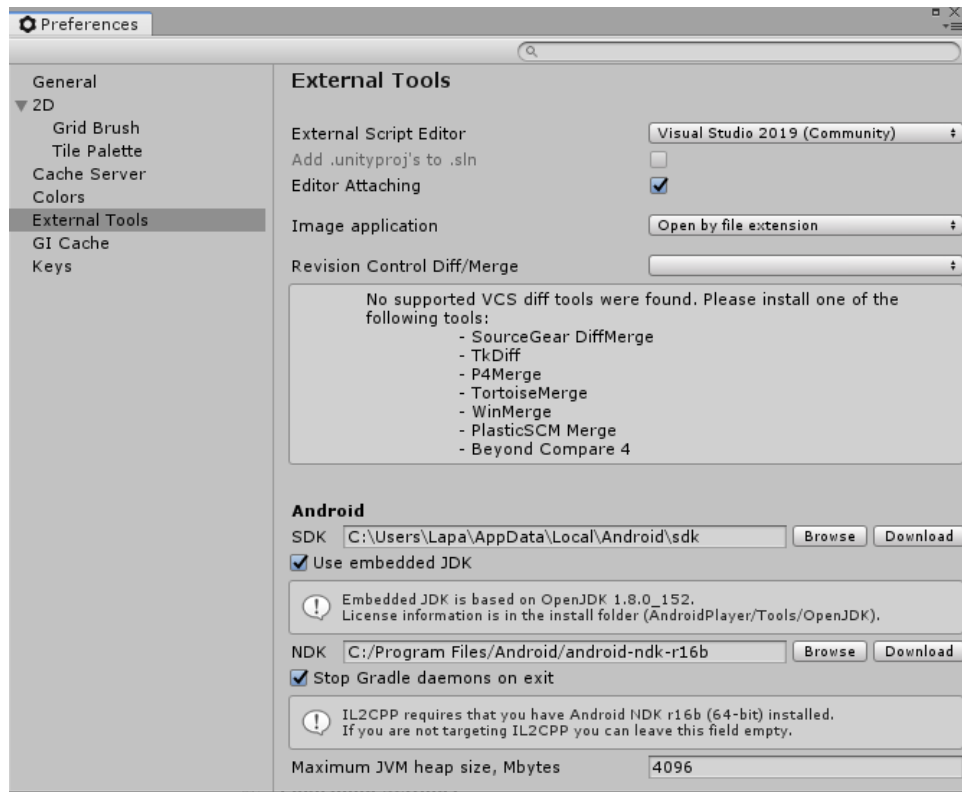


Рисунок 2.11. - Підключення додаткових програм

Підключення Android Studio необхідне для правильно створення проекту під мобільний девайс на операційній системі Android. Данна програма дає можливість встановити в Unity важливий компонент Android SDK. Цей компонента дає можливість збирати весь проект та імпортувати на мобільний телефон. Також додаючи необхідні елементи для створення мобільних додатків і дрібні налаштування проекту.

Після встановлення цих програм необхідно їх підключити до Unity, і тоді можна повноцінно займатися створенням і розробкою програмного продукту на Unity. Для створення даного додатку потрібно використати створення 2D сцени, так як елементи 3D використовуватися не будуть.



Рисунок 2.12. - Налаштування додатку

Окрім шаблонів, які містяться в програмі, потрібно створити заготовчі шаблоні для відтворення задач на математику і логіку для тестування програміста.

Програма дозволяє інтегрувати власні шрифти, але це додає зайву вагу програмі, тому було вирішено створювати інтерфейс вручну, запобігаючи інтегрування сторонніх шрифтів для полегшення процесу і зробити менше навантаження на процесор телефону, оптимізуючи при цьому задачі и весь інтерфейс програми. Після повного налаштування і підключення необхідних програм потрібно переключитися на розробку саме мобільного додатку, і змінити деякі налаштування проекту, а саме налаштування самого додатку.

«Icon» дозволяє змінити іконку головного виду самої програми. Для зміни іконки відведений великий пункт, тому що типи екранів та типів Android девайсів багато, і для коректного відображення на будь якому екрані потрібно створити якомога більше іконок додатку з різними співвідношенням сторін і розмірів головної іконки. Це також створено для того, щоб при завантаженні мобільного додатку на Play Market, сервіс автоматично зможе налаштовувати показ реклами вашого додатку та просувати її в тій категорії, в якому завантажено додаток.

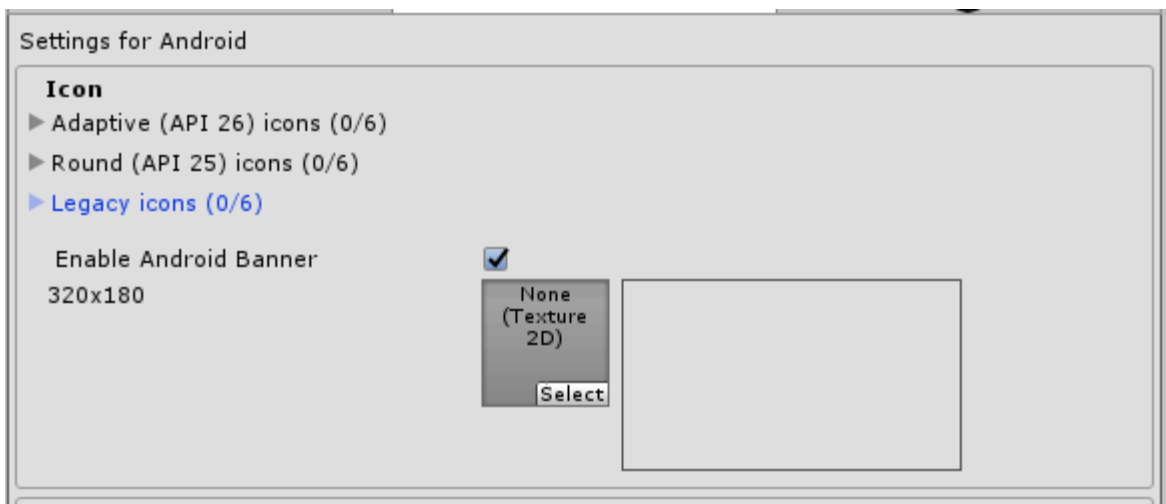


Рисунок 2.13. - Налаштування іконки додатку

«Resolution and Presentation» можна встановити відповідні налаштування системи, а саме відображення позиції додатку, також автоматичного повороту, використання 32/64 bit системи і налаштування під потрібні індикатори.

В «Splash Image» можна налаштувати заставку яка буде відображатися на початку за включення самої програми, додати також додаткові ефекти які можна використовувати. Якщо використовувати стандартну версію програми, то перша анімація включення буде завжди логотип компанії Unity, а потім власна анімація логотипу, видалити логотип компанії можна лише при купівлі комерційної версії продукту.

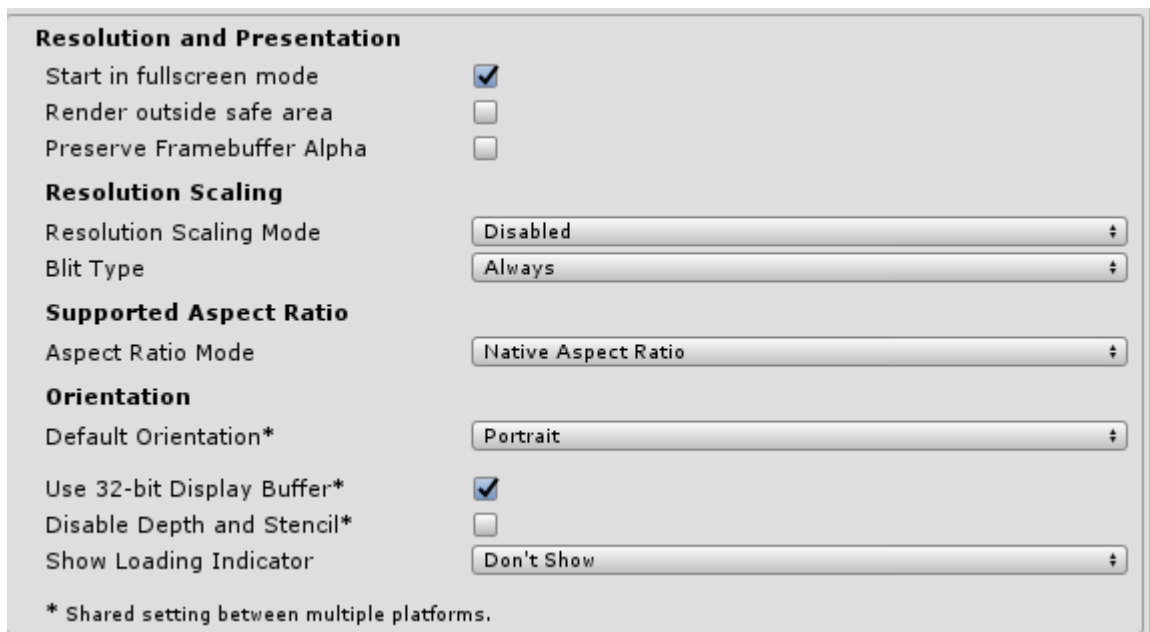


Рисунок 2.114. - Налаштування дозволу і презентації додатку

«Other Setting» наступні налаштування створені для більш конкретного налаштування архітектури і підтримки під конкретні задачі створюваного продукту а саме вибір версії Android версію ядра і архітектуру графічного відтворювача телефону і палітру відображення кольорів. Ці налаштування потрібні для підтримки старих девайсів, які мають проблеми зі сумісністю нових моделей програмного забезпечення. Одним із елементів налаштування є підтримка автоматичного відправлення помилок що дозволяє взаємодіяти між користувачем та розробником. Це дає можливість в подальшому оновлювати та модернізувати додаток. Якщо розробляти достатньо велико ресурсний додаток, то завдяки цій вкладці можна додати обмеження, які не



дадуть можливість встановлювати додаток на слабкі девайси, які не підтримують конкретні параметри створеного програмного продукту. Параметри налаштування також є і для створення та відслідковування дій користувача, наприклад взаємодію з програмним продуктом тільки якщо телефон підключений до інтернету, чи при виникненні помилки її можна автоматично відправити розробнику, для подальшого збирання інформації від користувачів, формувати звіт щодо помилок та в подальшому оновлювати та оптимізувати додаток.

«Publishing settings» - вкладка відповідає за підключення та інтеграції додатку в Play Market, підключення до Unity Hub і додатку відбувається за допомогою спеціального ключа які можна отримати якщо ви станете розробником на плей маркеті. Додається можливість випускати модифіковані версії додатку для вже старшого програмного продукту, чи повертати попередню версію, якщо нова не коректно працює. За цим параметром можна створити спеціальний окремий програмний додаток, що дає змогу тестувальникам та розробникам програмного продукту перевіряти додаток та отримувати файли використання і основних дій програмного продукту при тестуванні.

Після правильного налаштування створеного додатку з'являється можливість доступу магазину Unity, де можна скачувати і вибирати готові сцени, проекти, плагіни, які інтегрують і реалізують багатогранні можливості даного середовища розробки, вони є безкоштовні і умовно безкоштовні. Використання їх доцільне лише у випадку створення великих і масштабних проектів, я у своєму проекті використовую лише свої і унікальні скрипти, завдання і механіки. Також з'являється можливість інтегрувати рекламу в додаток с подальшою монетизацією створеної програми і просування її на Play Market, що дасть можливість розповсюджувати мобільний додаток для розвитку логіки і математики з назвою «Math and Logic».

### 3. ПРОЕКТУВАННЯ ТА СТВОРЕННЯ ДОДАТКУ

#### 3.1. Огляд програм для створення макету додатку

Adobe Photoshop – це редактор для обробки фото, відео файлів а також графічний дизайнер, можна створити не лише графічну графіку але й 3д. Ця програма популярна, її навіть використовують професіональні фотографи та дизайнери. В ньому дуже велика кількість інструментів.

Є готові корекції для редагування фото. Можна затемнити чи засвітити ділянки зображення. Завдяки темному інтерфейсу можна краще зосередитись на роботі.

На цій платформі є свої плюси. Переваги Adobe Photoshop: можливість зробити для себе зручний інтерфейс, є безкоштовна версія, також можна скачати доповнені інструменти до тих що є, багато відеороликів про користування Adobe Photoshop, по відеороликам легко навчитися користуватися Adobe Photoshop.

Із мінусів, можна зазначити те, що офіційна підписка платна, не всі комп'ютери зможуть потягнути Adobe Photoshop, новачкам буде тяжко навчитися володіти цим редактором. Програма тяжка для освоєння новому користувачу.

В наш час, всі графічні дизайнери використовують саме Adobe Photoshop. Майже всі макети сайтів створюються в цій програмі. Для різних задач інтерфейс Photoshop може виглядати по різному, його можна зробити так щоб початківцю було зрозуміло, а можна зробити для зручне положення вікон для професійного дизайнера.

Використання даної програми, може зекономити час, так як дозволяє створювати макети та шаблони, і інтегрувати з іншими програмами, наприклад якщо використовувати зв'язок між Adobe Photoshop Adobe Illustrator можна імпортувати заготовленні макети з однієї ж програми і створювати прототип у векторному форматі в іншій програмі. Але така схема моделювання має недолік, так як використання даних програм потребує великий ресурс комп'ютерного заліза і не підходить початківцям.

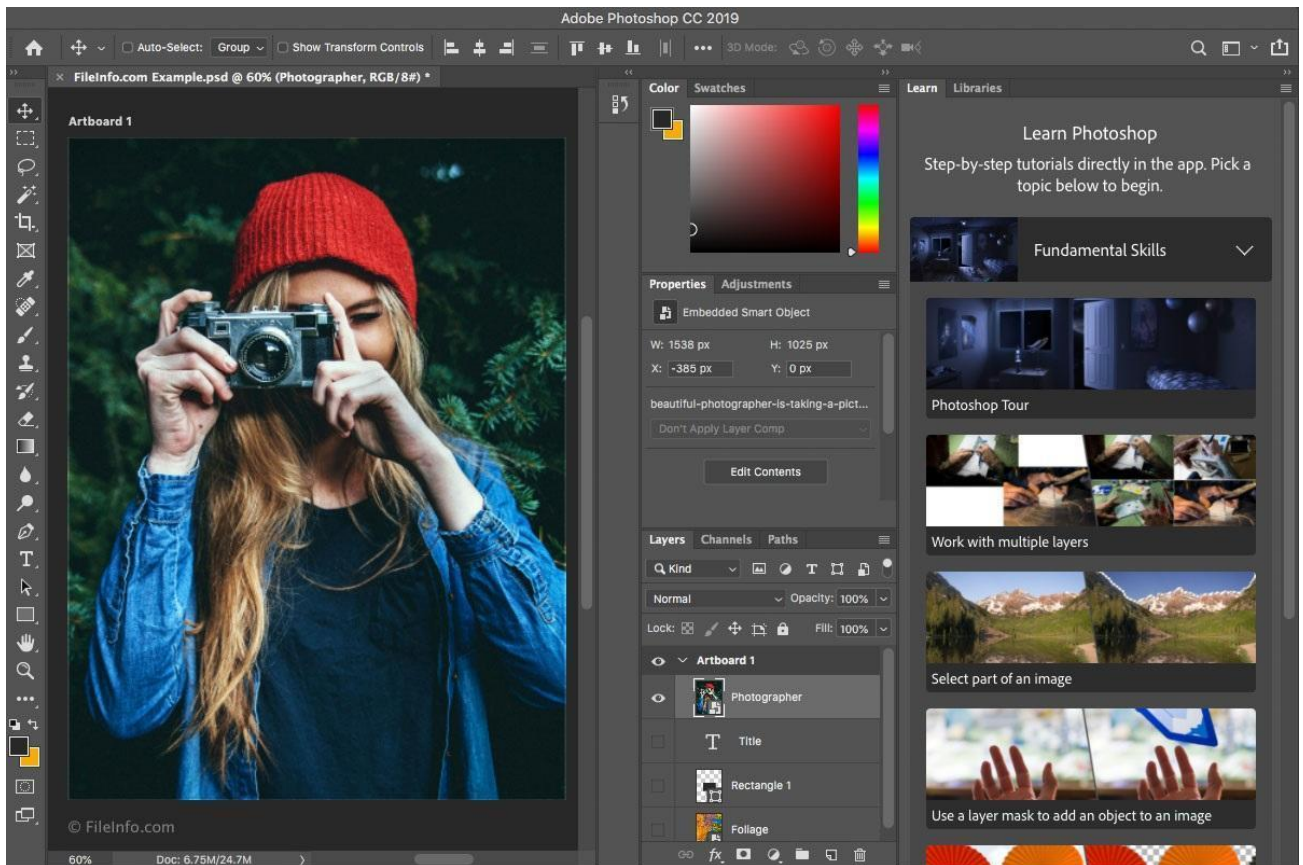


Рисунок 3.1. - вікно програми Adobe Photoshop

Саме в Adobe Photoshop було зроблено гру «Math and Logic». Завдяки зручному та великій можливості інструментів, робити подібні ігри цікаво та зручно. Великий вибір літер, які можна також редагувати, палітра кольорів, розміри та види[10].

Переваги використання програми для створення макетів:

1. Візуалізація ідеї при створенні шаблону майбутнього додатку.
2. Чітке структурування вигляду всіх елементів інтерфейсу.
3. Імпорт окремих елементів дизайну.
4. Можливість працювати з ескізами та малюнками.

Огляд інших програм для створення макетів які використовувалися для створення Дипломного проекту.

Adobe Illustrator – це векторний графічний редактор. Її використовують художники, ілюстратори та дизайнери для векторної графіки. В Adobe Illustrator можна не тільки малювати але й редагувати роботу.

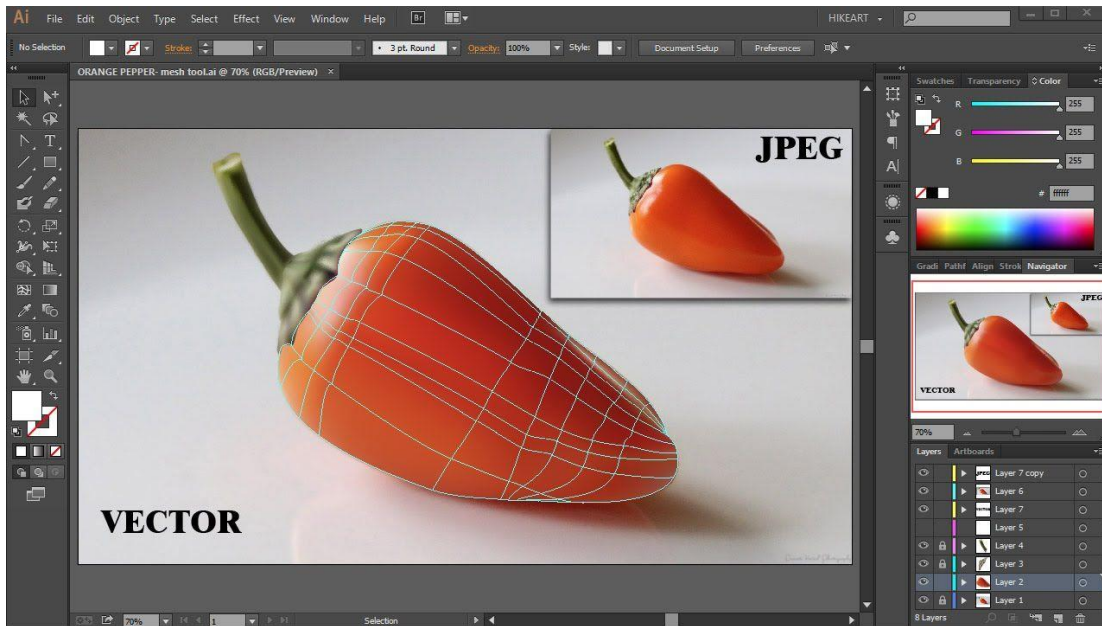


Рисунок 3.2. - вікно програми Adobe Illustrator

Функціонування програми полягає в різноманітному виборі кистей, різних операцій та ефектів, створення об'єктів із застосуванням оригінальних кистей, редагування заливок і контурів об'єктів, створення тривимірних спец ефектів, застосування градієнтів, змішання кольорів, багата можливість редагування текстової інформації, растрування об'єктів, а також трасування растрової графіки, регулювання прозорості, застосування різноманітних художніх ефектів, малювання векторні об'єкти довільної форми, підтримка символів і їх копій, створення багатосторінкових файлів з інформацією про шарі[11].

Створення векторних зображень дозволяє створювати чисті та бездоганні ілюстрації, саме в цій програмі всі зроблені роботи будуть залишатися такими ж, та не будуть втрачати свою якість, навіть тоді коли гортати зображення в верх та в низ.

В Adobe Illustrator не потрібні навіть навички малювати, так як завдяки векторному зображенню, та тому як воно виправляє криві лінії.

Також можна зробити свій звичайний малюнок та анімацію, для цього потрібно імпортувати у Flash, та можете дивитися, як ваш малюнок оживає. Ще можна додати свій логотип та ілюстрацію до відео, або оживити його через After Effects.

Якщо ілюстрація створена як растрові зображення не можуть бути перетворені у векторний формат.

Цю програму використовують, як для особистих потреб так і по роботі у великих компаніях. Система надає безліч інструментів для створення логотипів, ілюстрацій, типографіки, візиток та проектів, також можна використовувати для різних декорацій, відеоігор та одягу.

Adobe Illustrator достатньо важкий для новачка, щоб вивчити всі функції ілюстратора знадобиться не мало часу. Також, всі файли Adobe можуть переходити з однієї програми до іншої, оскільки компанія забезпечила їх сумісність[12].

Figma – це графічний редактор для веб-дизайнера. В ньому можна створювати: Інтерактивні прототипи сайтів та мобільних програм; елементи інтерфейсу – вікна, меню, іконки, кнопки та форми зворотного зв'язку.

В Figma всі документи зберігаються на хмарі. Завдяки цьому, можна працювати над макетами за посиланнями, без загрузок. Інструменти для власного користування всі безкоштовні. Можна давати доступ до документи, ця функція також безкоштовна. Ціна за підписку на Figma, залежить від можливостей для роботи. Якщо буде безкоштовний план, то туди входять: прототипи Figma, атрибути коду, імпорт документів із Sketch та інших форматів, зберігання документів без обмежень та доступ для перегляду та коментування. Але не можна буде спільно редагувати документи та історія на хмарі зберігається лише 30 днів.

Після реєстрування, на пошту прийде повідомлення з відео уроками, форумами та допомогою.

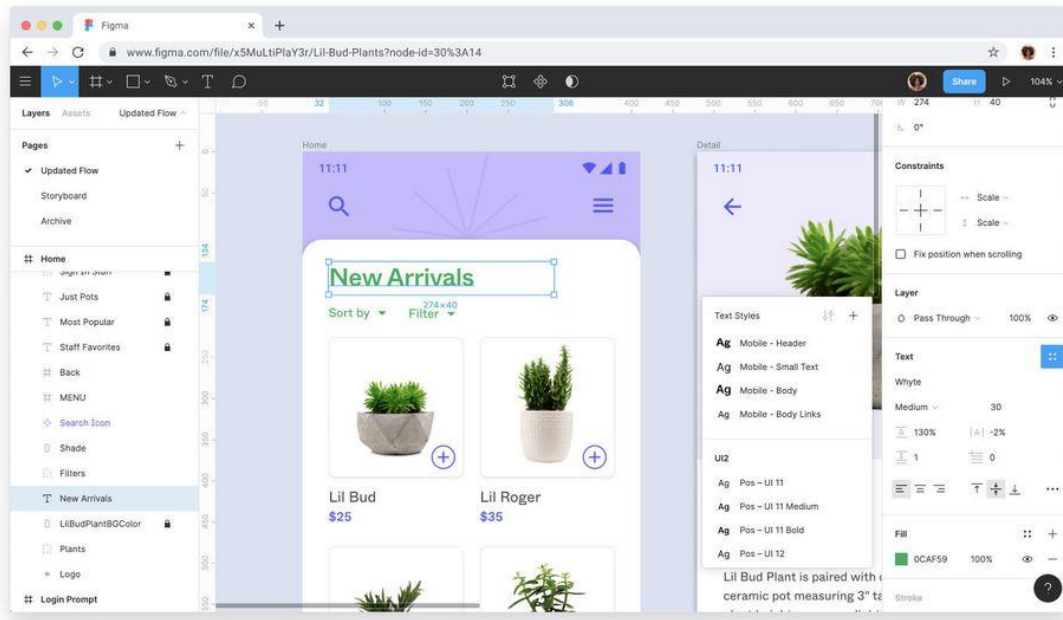


Рисунок 3.3. - інтерфейс ресурсу Figma

Figma дозволяє зробити версію посилання або мобільної програми, робивши безліч екранів. Всі екрани розміщені на одній сторінці, завдяки цьому не потрібно буде клацати між екранами.

Компоненти Figma – це повторні елементи у проекті. Можна створювати компоненти із кнопок, заголовків, меню, іконок, полів, форм та блоків.

Фрейми дуже схожі на артборди у Photoshop, но мають дуже велику гнучкість у роботі, вони добре масштабуються та легко можуть використовуватися у інших фреймах.

Figma – це взагалі онлайн сервіс, але також є десктопна версія Figma. Щоб працювати на цьому сервері, потрібно мати завжди інтернет підключення. Всі файли зберігаються локально, але якщо потрібно, то можна їх імпортувати в версію інтернет браузера.

Figma можна користуватися безкоштовно, але буде небагато функцій, а саме будуть такі:

- максимум можна робити три проекта;

- максимум всього лише два редактора;
- історія версій зберігається лише 30 днів.

У Figma шрифти підтягуються з Google Fonts або можна автоматично завантажувати у проект.

Таблиця 3.1. – Порівняння додатків для розвитку логіки та математики

Назва програми	Adobe Photoshop	Adobe Illustrator	Figma
Передбачуване використання	Створення макетів сайтів, логотипів, дизайну додатків, імпорт в будь який формат, 3D моделювання	Векторна графіка, створення дизайну логотипів, створення анімації будь якої складності	Тільки створення дизайну веб застосунків і мобільних додатків і відтворення взаємодії між макетами
Векторна графіка	Так	Так	Ні
Растрова графіка	Так	Ні	Так
Імпорт створених шаблонів	Так	Так	Ні
Вибір формату зображення	Так	Так	Так
Шаблони додатку	Ні	Ні	Так
Створення Дизайну	Так	Так	Так

Виконавши аналіз всіх можливих програм для створення шаблону мобільного додатку, можна зробити висновок що саме використання програми Adobe Photoshop

дає більш широкі можливості для цього і весь функціонал повністю забезпечить доцільну розробку. Спираючись на данні з табл. 3.1. можна зробити висновок, що Adobe Illustrator для використання та створення логотипів а також створення макетів з векторної графіки. Figma реалізована для візуалізацію взаємодії між створеним макетом, що пришвидшує сприйняття готового продукту, але має недолік в тому, що не можливо імпортувати будь який елемент до формату, який надалі можна використовувати для створення мобільного додатку.

Створення шаблону додатку для вивчення та розвитку математичних та логічних здібностей у програміста буде виконуватись саме в програмі Photoshop. Всі елементи меню і логічних задач виконується в програмі, щоб зменшити навантаження на програму Unity.

### **3.2. Моделювання об'єкту проектування**

Моделювання програмного продукту дозволяє уявити та повністю відобразити роботу програмного продукту, а також розбити на етапи створення для послідовного проектування. Для створення моделі проекту, або програмного продукту використовують UML. Це мова графічного опису створюваного програмного продукту, що дозволяє візуалізувати будь який процес.

Для візуалізації процесу роботи мобільного або показу всіх основних складових процесів використання додатку використовують блок-схему, діаграму станів а також діаграму діяльності. Прямолінійне розгалуження для будь якого програмного продукту означає, зо програма має тільки один шлях для роботи з нею і не має розгалужень чи варіацій використання. Такий програмний продукт не має значущого і корисного використання.

Загальні характеристики блок схеми мають свої стандарти, що дає змогу будь якому аналізувати чи створювати проект за готовою схемою.



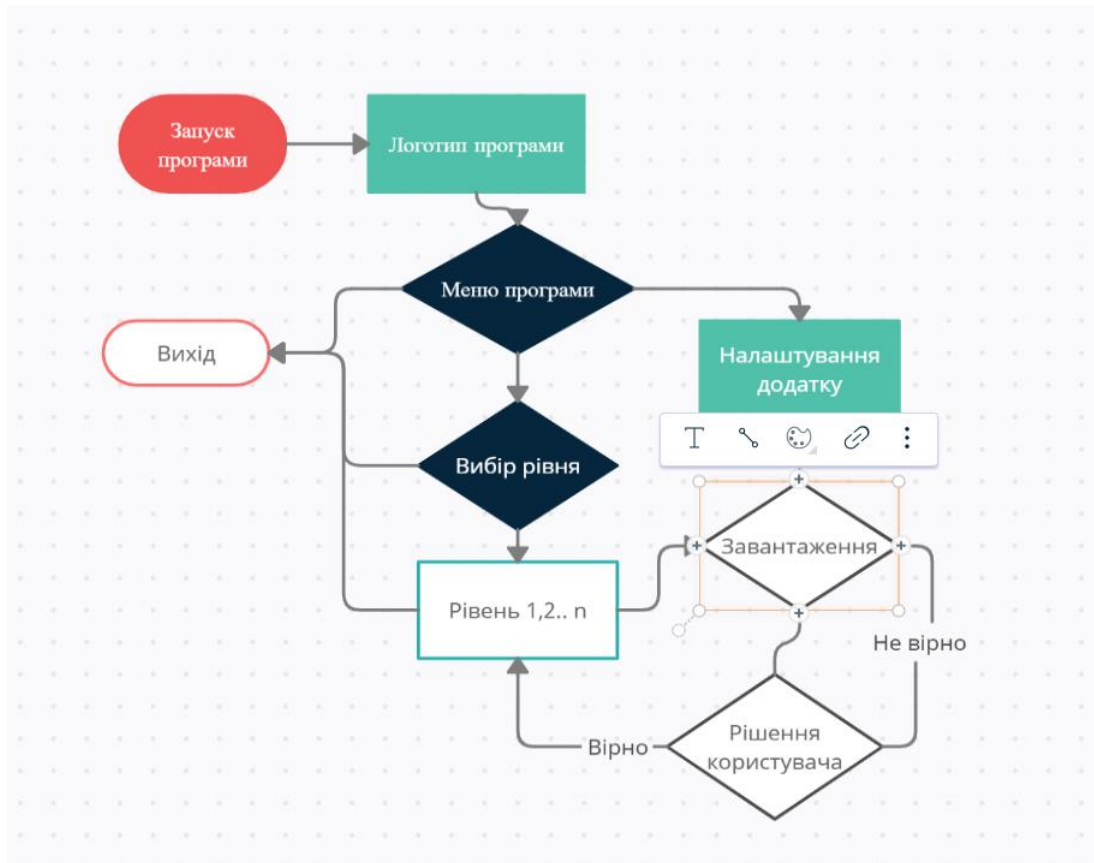


Рисунок 3.4. - Блок-схема роботи програми

Основні складові блок-схеми з таких фігур:

1. Закруглений прямокутник. Відображає початок або вихід з програми.
2. Прямокутник. Елемент відображає процес який виконується.
3. Паралелограм, відображає всі вхідні та вихідні данні що дає програмний продукт.
4. Ромб, означає умову, яка відбувається у програмі. Вихідні значення можуть бути розгалужені, так як умова може бути як підтверджена так і спростована.
5. Прямокутник з полями, цей елемент відображає виконання будь якого процесу, що складається з декількох або однієї операції.
6. Шестикутник, відображає елемент функцію циклічного характеру. Використовується для обробки вхідних або вихідних значень, або елементів, які повинні відбуватися лише тоді, коли виконується умова.

Для відображення послідовного зв'язку між фігурами блок схеми використовують ліній з стрілкою, яка позначає з який порядок дій та етапи блок-схеми.

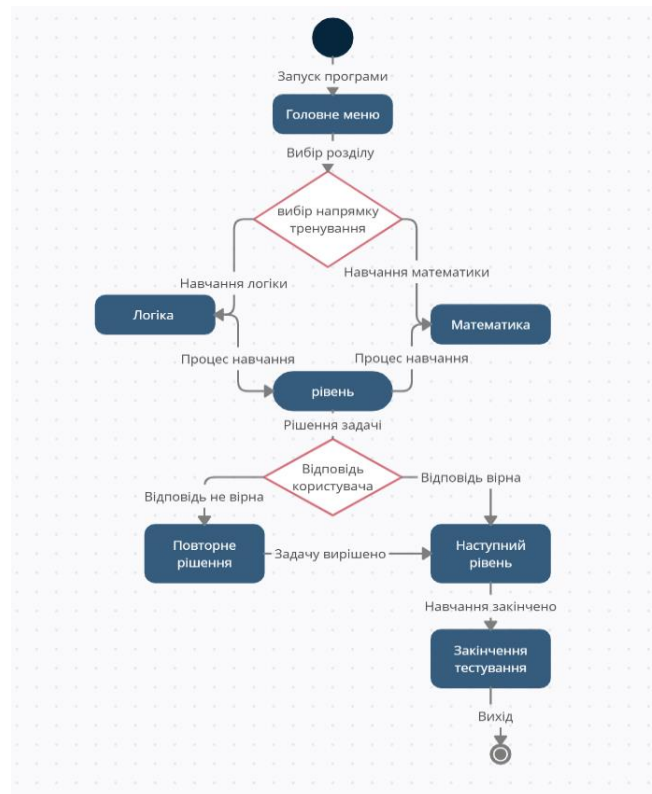


Рисунок 3.5. – Діаграма станів

Діаграма стану – це діаграма, яка використовується для опису поведінки системи, якщо відбувається подія, то враховуються всі можливі стани об'єкта.

Бувають різні діаграми стану, які трохи відрізняються один від одного.

Елементами діаграми стану є округлі поля. Кожна діаграма стану має початок, в якому створюють об'єкт. Після початку стану об'єкти змінюють свій стан. Наступний стан визначається умовами, які засновані на діяльності.

Також є перехід у діаграму стану – це перехід, від одного стану до іншого. Перехід може здійснюватися тоді, коли будуть отримані об'єктом повідомлень або сигналом прийому та закінчення виконання діяльності.

Діаграми станів частіше використовують для описання поведінки різних та окремих екземплярів об'єктів. Діаграми станів також можуть використовувати для функціональності моделей, а саме:

- операцій;
- методів;
- варіантів використання;
- підсистеми;
- перехід діаграми станів відображається цілою лінією та стрілкою.

Діаграми станів використовують для того, щоб можна було пояснити, як працюють складні об'єкти. Вони працюють для того, щоб моделювати динамічні аспекти систем.

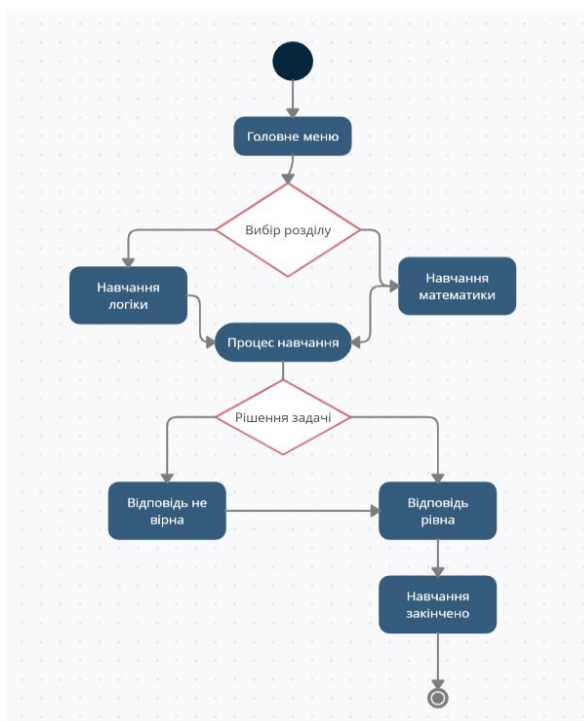


Рисунок 3.6. – Діаграма діяльності

Діаграма діяльності – це візуальне відображення графу діяльності. Головною вершиною графу є коректні та певні дії, переходи цього відбуваються при завершенні

дій. Діаграма діяльності це окремий вид діаграми станів. Кожен стан можуть бути виконання операції деякої частини або деякої операції класу.

При розробці якихось складних проектів, проблема вибору оптимальних рішень стає дуже актуальною. Кожна діаграма діяльності має мати свій початок та кінець. Потрібно розташовувати діаграму діяльності так, щоб всі виконуючі дії, виконувалися зверху до низу.

Коли будується діаграма діяльності, там може використовуватися лише нетригерні переходи. На діаграмі діяльності галуження зображено невеликим розміром ромбом, без тексту в середині. У діаграмі діяльності важливою сферою використання є те, що їх можна застосовувати для моделювання різних бізнес-процесів.

### **3.3. Створення макету мобільного додатку**

Проаналізувавши програмні засоби створення макетів, було вирішено скористуватися програмою Adobe Photoshop так як вона по функціоналу більш зручна там має необмежену кількість налаштувань для створення дизайну.

Для початку створення шаблону дизайну було створено та намальований прототип цільової програми і в подальшому всі ескізи були імпортовані та відцифровані для подальшого редагування в програмі[13].

Розробка макету проходила за наступним сценарієм:

1. Створення документу потрібного розміру, використовував розмір стандартного мобільного телефону з розширенням 1080 на 1920 пікселів в вертикальному положенню робочого аркуша.
2. Створення оригінальної заставки і анімація її методом розширення кадрів статичного тексту.
3. Вибір і збереження палітри кольорів які використовуються, для подальшого редагування.

4. Створення форми усіх кнопок що використовуються для загального меню що складаються з цифр, кнопок за назвою, кнопок підменю.
5. Створення задач під одною стилістикою.
6. Створення відповідей під одну стилістику.

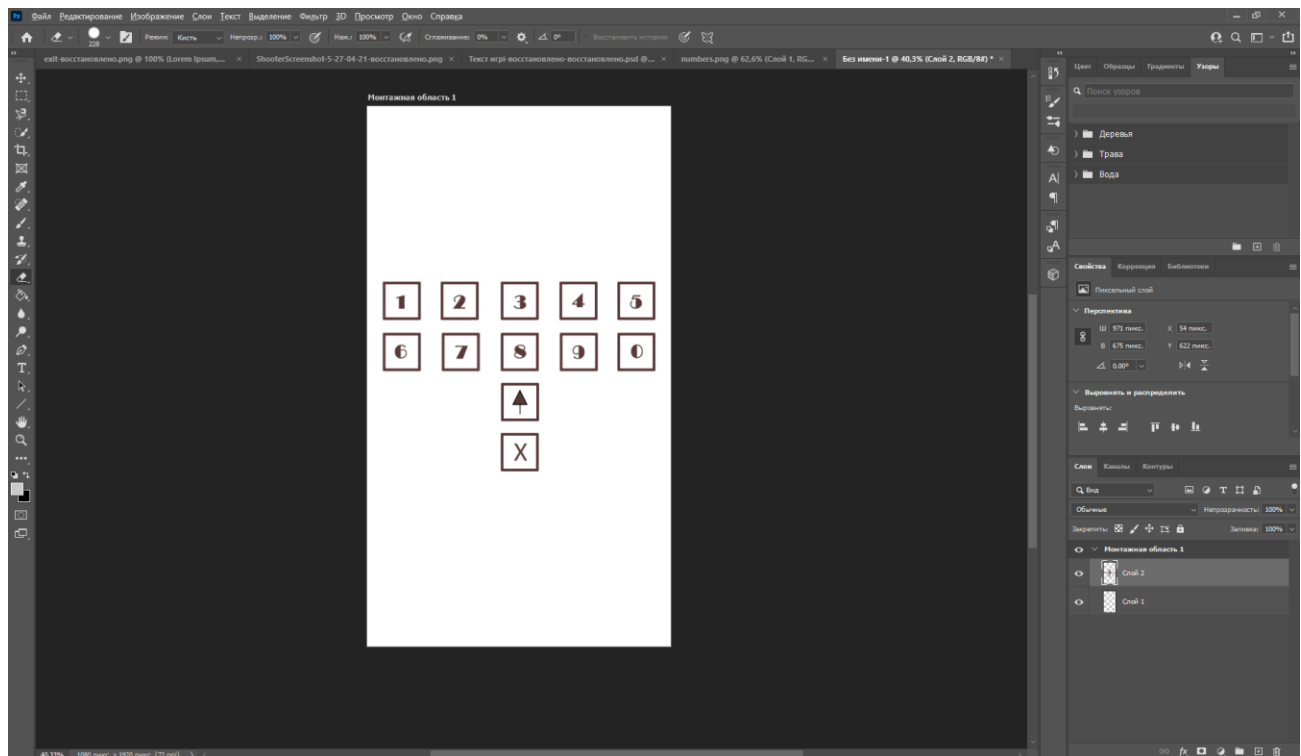


Рисунок 3.7. - Розробка кнопок для додатку

Зручність використання програми фотошоп полягає в тому що можна зразу побачити як воно буде виглядати на екрані телефону. Створюючи шаблон інтерфейсу його зразу можна зберігати окремим файлом для подальшого редагування в програмі Unity, спеціальний розділ автоматично знаходить та виділяє необхідні елементи, тобто створивши макет програмного додатку можна зразу отримати з нього всі необхідні шматочки інтерфейсу програми. Тому після створення необхідних шаблонів дизайну їх можна зразу відкривати і використовувати як об'єкти взаємодії при створенні додатку.

Для коректного відображення всіх елементів в програмі для створення додатку

потрібно кожний файл окремо імпортувати в розширенні PNG, цей формат дає змогу вставляти текстові елемент без заднього фону, а також створювати прозорі елементи на сцені.

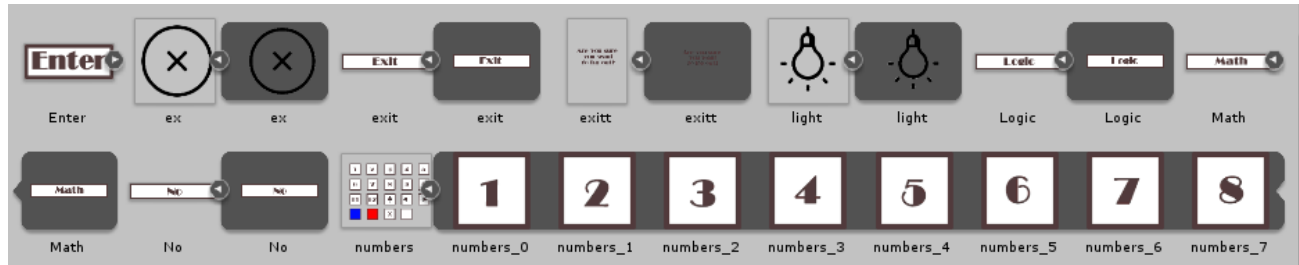


Рисунок 3.8. - Unity розгорнув кнопки на окремі елементи

При використанні програми Photoshop було створено і розроблено унікальний інтерфейс програми і також підібрано шрифт, що дає можливість уніфікувати мобільний додаток і зробити його неповторним. Також з метою покращення сприйняття задач для логіки і математики було виключено елементи декору для того щоб користувач міг повністю сконцентруватися на своєму навчанні та розвитку.

### 3.4. Створення мобільного додатку в Unity

Для початку розробки необхідно встановити головну програму Unity на персональний комп'ютер, для цього потрібно на офіційному сайті завантажити обрану версію. Доступні дві головні версії – безкоштовна і платна. Для звичайних користувачів або студентів доступна безкоштовна версія, яка має трохи обмежений функціонал, але для звичайного користувача чи невеликої компанії цього функціоналу буде достатньо. Для компаній, які серйозно займаються розробкою створення не тільки мобільних версій додатків а також розробкою на інші платформи, які займаються комерцією, потрібно встановлювати платну версію середовища розробки[14].

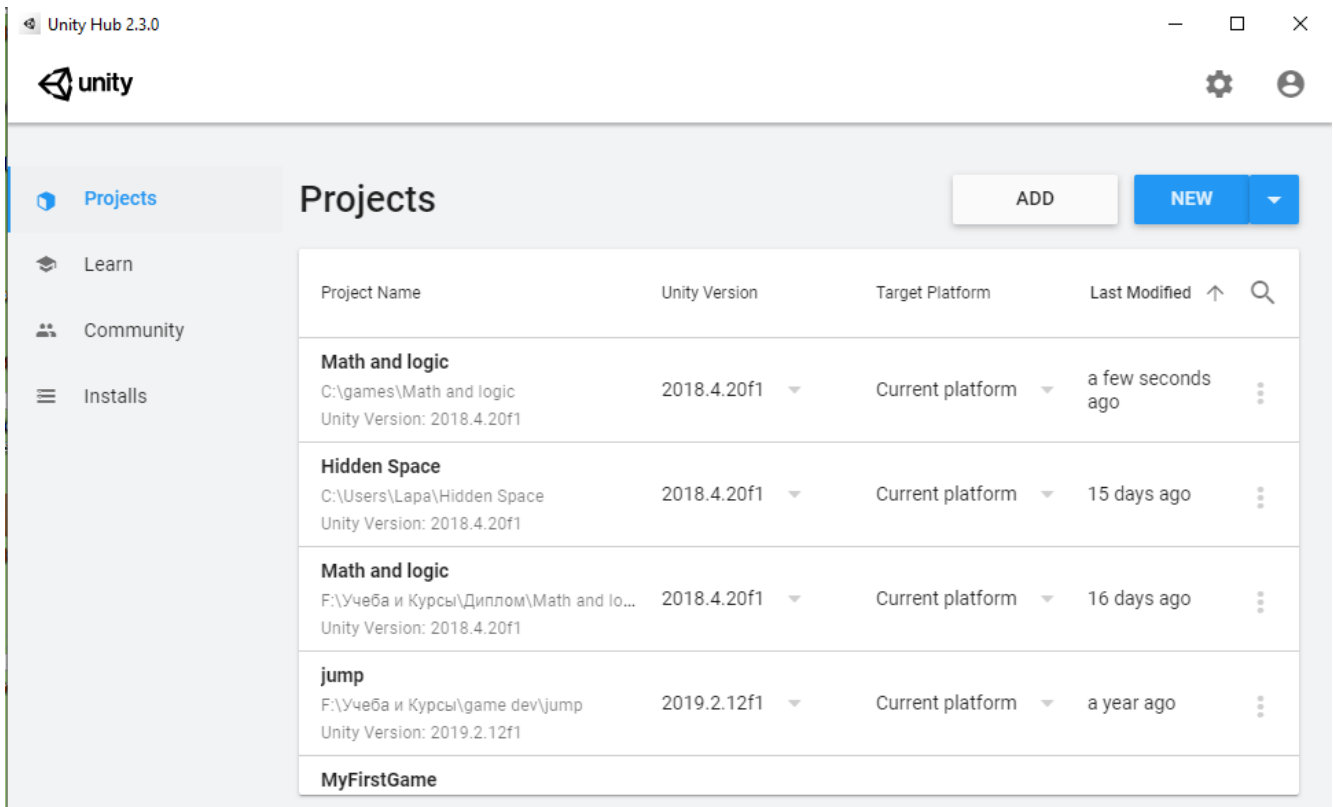


Рисунок 3.9. - Меню Unity Hub

Після того як пройде встановлення відкривається лаунчер Unity Hub- це програма яка дозволяє створювати та налаштовувати проект під будь яку з доступних платформ, а також а також встановлювати і завантажувати потрібну версію Unity. Також відкривається доступ до перегляду простих навчаючих проектів, які показують початківцю з чого по-перше починати розробку, які існують механіки взаємодії.

В Unity є доступ до спільноти, це може буди актуальним як для початківців так і для професіоналів. Цей розділ створено для того, щоб односторонні розробники мобільних і не тільки додатків мали можливість обмінюватися і ділитися своїми створеними прожектами, а також допомагали один одному, якщо з'являються помилки або потрібна допомога за написання коду або створення взаємодії з об'єктами.

Інтерфейс головного меню, в якому можна завантажувати різні компоненти, вибирати поточний проект та вибрати платформу для створення гри.

Для створення додатку необхідно першим чином створити проект самого додатку, для цього необхідно в програмі Unity Hub потрібно вибрати специфікацію самої гри, а саме версію Unity, допоміжні компоненти, тип програмного додатку, назву, і місце куди зберігати додаток. Головний параметр створеного додатку- це щоб місце збереження проекту не містив назви кирилівськими буквами, тільки англійською мовою, це є один нюанс при створенні проекту.

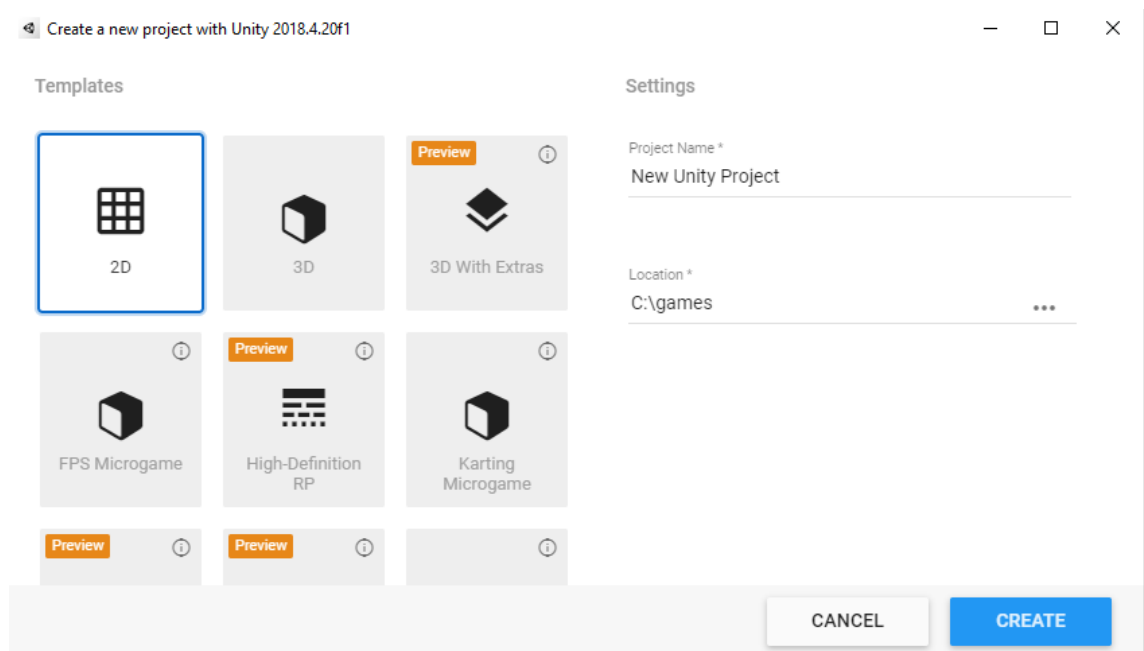


Рисунок 3.10. - Меню створення проекту

Створення починається із розміщення необхідних об'єктів на сцені. Для зручного використанні можна створювати об'єктні копії, для того щоб їх дублювати. Дублікати копій дублюють з материнського об'єкту повністю всі властивості. Тому при зміні будь якого дочірнього об'єкту можна змінити початковий елемент. Якщо використовувати основні компоненти і надавати їм властивості, які додаються автоматично, то це призведе до не коректної взаємодії між об'єктами.



По перше після створення проекту потрібно переконатися, яка саме версія Unity встановлена, так як від версії залежить підтримка будь якого функціоналу. Наприклад старі версії не підтримують нововведені модулі взаємодії, або навпаки, біль нова версія вже має деякі елементи оптимізації, і старі методи створенні на ранніх версіях вже не підтримуються, або мають інший оптимальний вигляд.

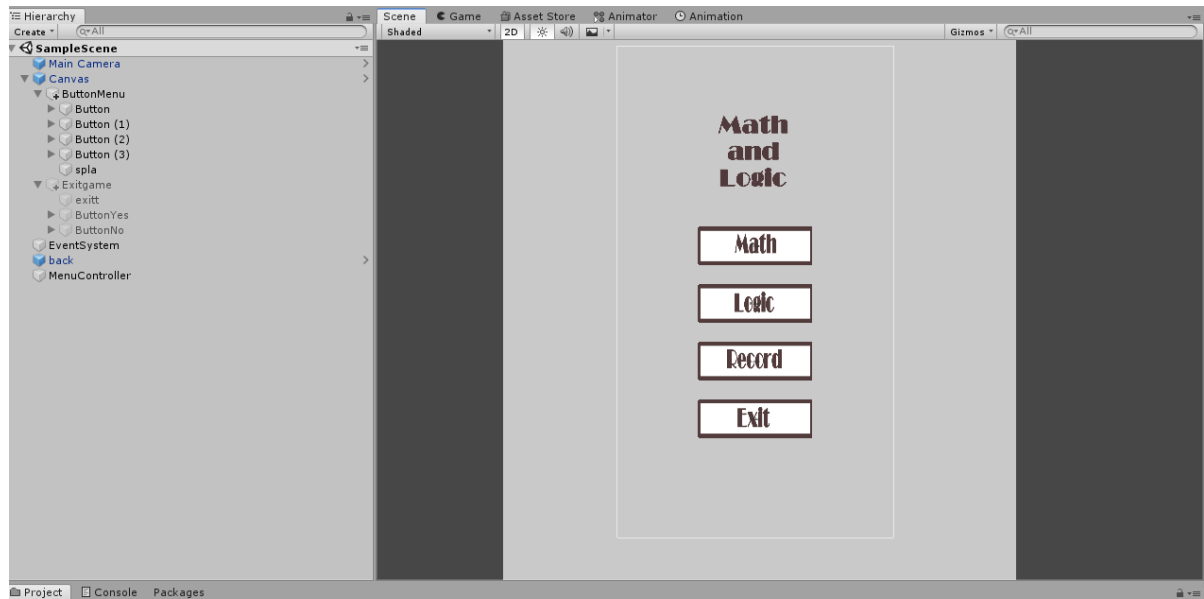


Рисунок 3.11. - Розміщення об'єктів на сцені додатку

Опираючись на це потрібно ознайомитися с документацією версії на якій буде розробка програмного додатку з вивчення та тренування логіки та математики програміста. Починаючи створювати мобільний додаток, потрібно порівняти аналоги і проаналізувати, з чого повинен починатися мобільний додаток. Візуальна частина починається з екрану завантаження. В Unity можна створити власний екран завантаження з вибором потрібною анімацією і впровадження музики або відео афектами за бажанням. Окрім стандартної заставки Unity, можна таким чином більш уніфікувати мобільний додаток.

Перша сцена яку потрібно створити це головне меню додатку, на якому розмішені всі необхідні елементи, для взаємодії з користувачем. Користувач має право

на вибір між режимами тренування чи навчання, перегляд власних досягнень в мобільному додатку, а також на вихід з додатку за бажанням. На рис.3.ттт. зображено головне меню додатку розмішене на сцені, задній фон розмішений таким чином, щоб він заходив за межі головного екрану, це створено для того, якщо користувач буде використовувати телефон з великим розширенням екрану, це дасть можливість йому не бачити нічого іншого окрім фону.

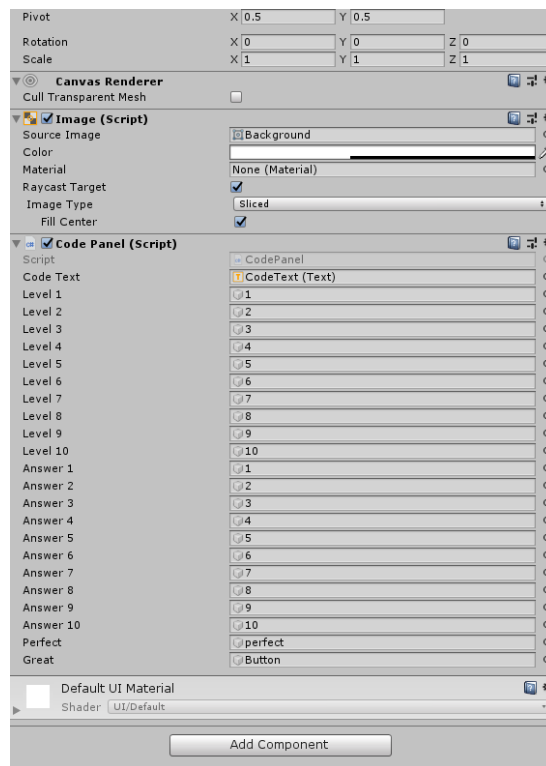


Рисунок 3.12. - Підключення скрипту взаємодії

Також для коректного відображення елементів меню і компонентів додатку потрібно обов'язково налаштувати положення екрану. Для зручного сприйняття вирішено використовувати вертикальне положення. Це дає можливість охопити більш важливу область фокусування користувача, так як центрально розміщення об'єктів можливе тільки при вертикальному розміщенні телефона.

Після повного налаштування інтерфейсу головного меню потрібно також створити необхідні сцени що дають можливість навчатися і розвивати математичні і

логічні навички у програміста. Для цього окрім розміщення на сцені об'єктів і кнопок, а також доповнюючи елементів додатку, потрібно створити взаємодію між об'єктами, зв'язати послідовність дій між ігровими елементами. Для цього потрібно долучитися до написання скрипту взаємодії і правильно підключати до потрібних об'єктів.

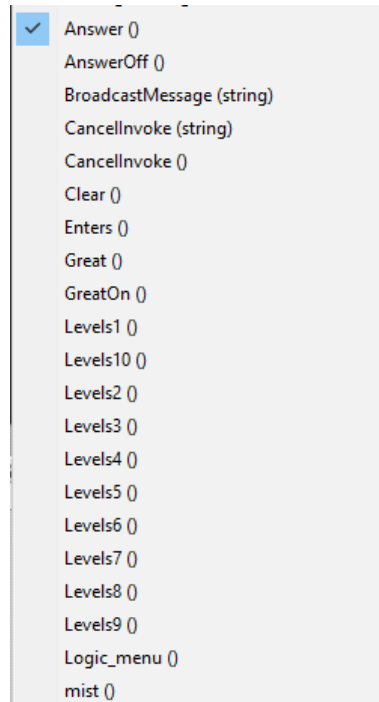


Рисунок 3.13. - Класи та методи створенні для взаємодії між об'єктами

Після коректного підключення до об'єкту взаємодії під назвою «Code panel» я створюю додаткові взаємо зв'язки між всіма елементами, прописуючи їх в кодї та використовуючи систему drag and drop підключаю об'єкти.

Кожному рівню в мобільному додатку я присвоїв унікальний номер, за яким закріплюється задача а також підказка для того, щоб користувач зміг самостійно подумати і з підказкою вирішити будь яку математичну або логічну задачу.

Під час створення додатку його можна запускати і перевіряти як додаток буде функціонувати на мобільному пристрої, це дає змогу без компіляції переглядати взаємодію між об'єктами і без скачування і підключення андроїд девайсу вирішувати проблеми або задачі програмного продукту, це є дуже зручно.

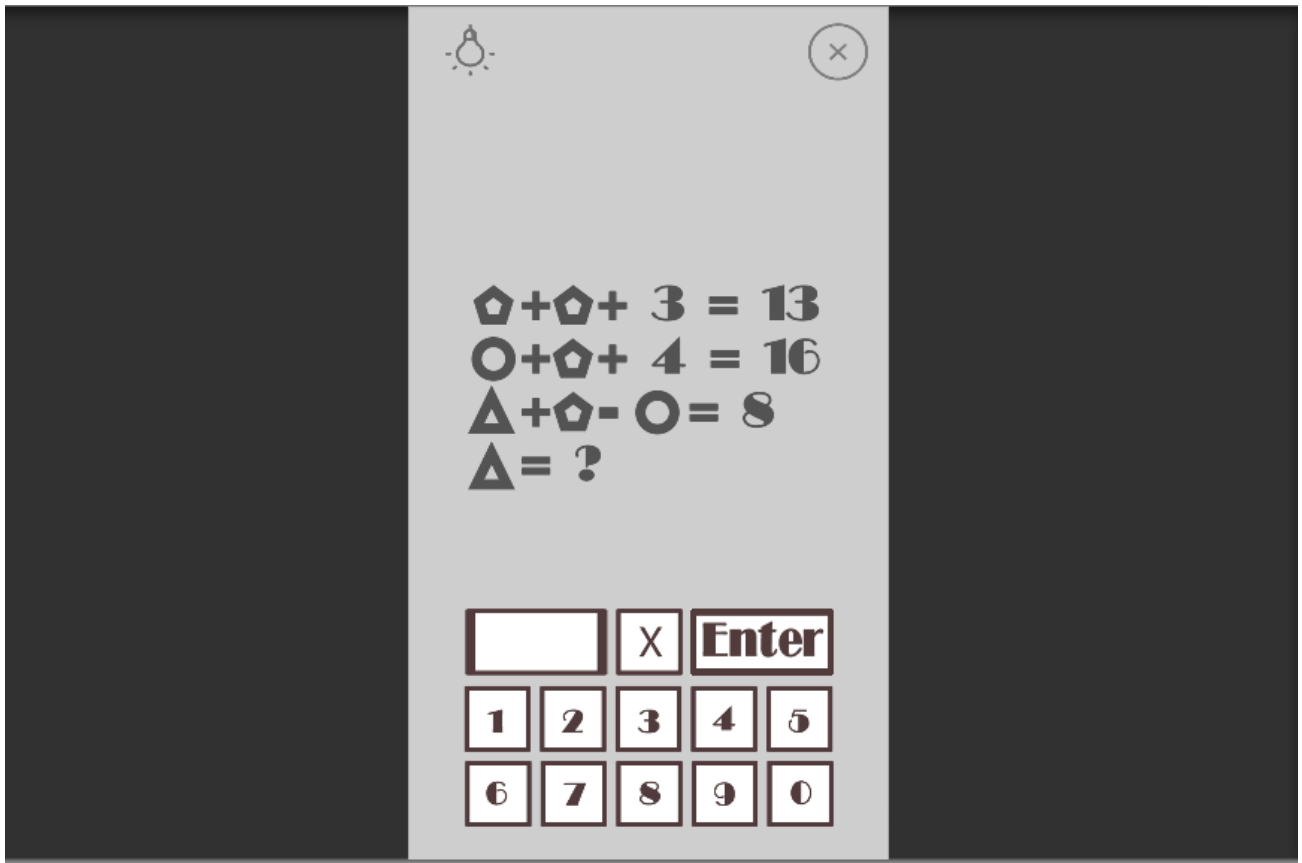


Рисунок 3.14. - Створена сцена мобільного додатку

Для кожного елемента дизайну додатку біло доданий унікальний скрипт, багаторівневий код дозволяє незалежно від кількості рівнів додавати безліч кількостей рівнів, це створено для того, щоб можна було в подальшому змінювати гру без втручання в код, і оновлювати старі логічні загадки і математичні головоломки.

Також створено візуальні ефекти перемоги або помилки, якщо користувач схибив з відповіддю або правильно надав коректну інформацію. Це надихає користувача більше тренуватися і не відволікатися від по сторонніх магнітів інтерфейсу.

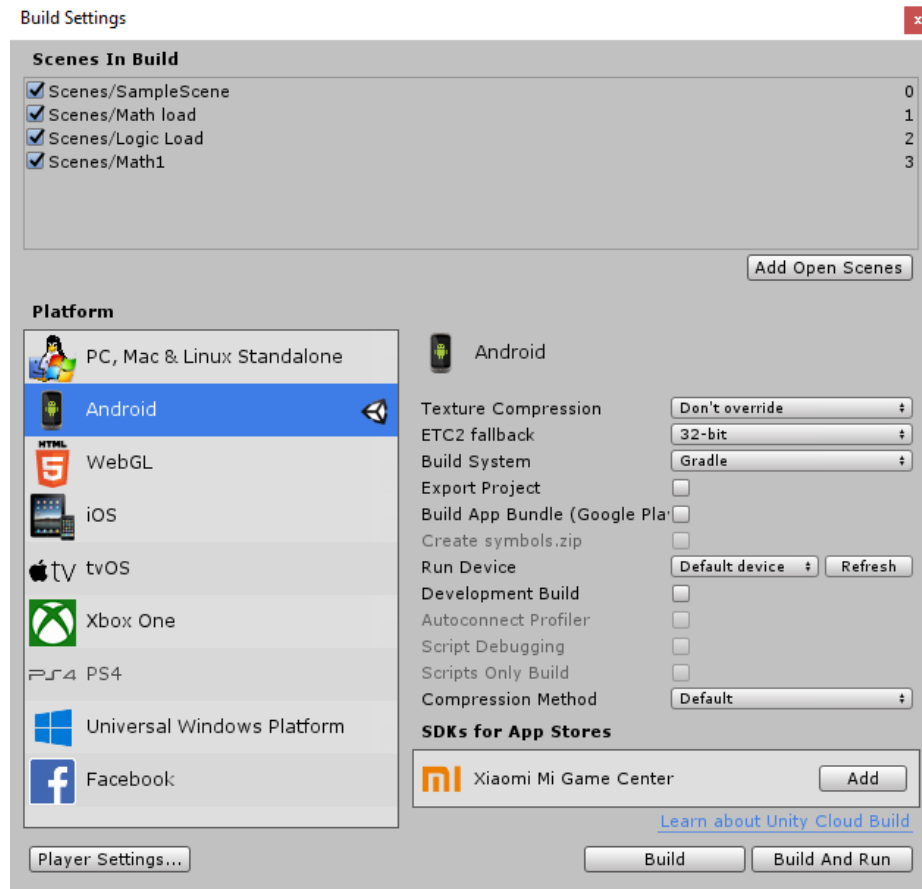


Рисунок 3.15. - Сторінка імпорту мобільного додатку

Після створення мобільного додатку його необхідно імпортувати на мобільний пристрій на операційній системі Android. Для цього потрібно імпортувати проект під вибрану платформу і вибрати необхідні параметри архітектури системи і вибрати необхідні сцени. Які будуть доступні в мобільному додатку. Після створення на комп'ютері з'явиться програмний продукт з розширенням «APK». Для його можна завантажити на будь який Android пристрій для перевірки працездатності готового мобільного додатку.

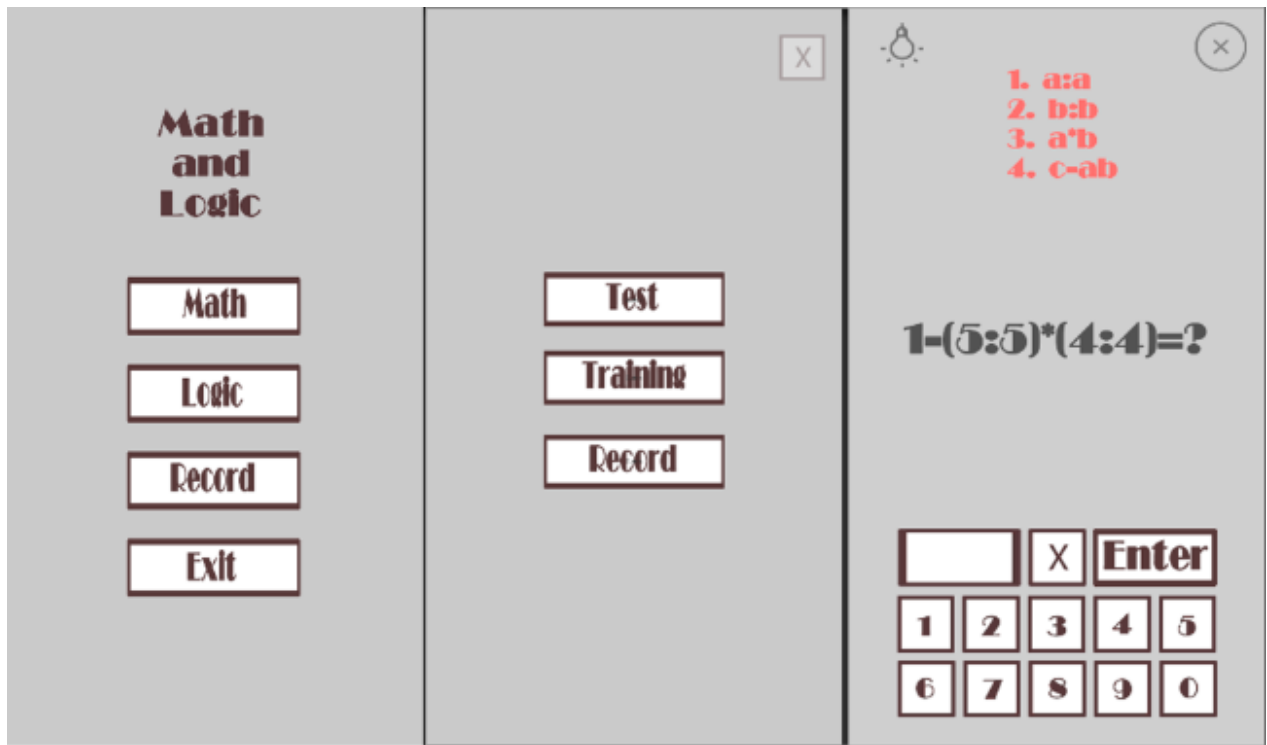


Рисунок 3.16. – Скріншоти програми

Після створення додатку та завантаження його на мобільний телефон, потрібно перевірити додаток методом фактичного тестування, і також перевірити відповідність усіх рівнім і підказок, що використовувалися, щоб переконатися в правильній працездатності програми для навчання та тестування логіки та математики у програмістів.

## 4 ІНСТРУКЦІЯ ВИКОРИСТАННЯ МОБІЛЬНОГО ДОДАТКУ

### 4.1. Процес створення задач для додатку

Щоб створити один рівень, потрібно провести аналіз існуючих прикладів на логіку та математику, і створити власний приклад, використовуючи формули, зашифровуючи відповідь на створену задачу[15].

Спочатку потрібно створити формулу для нашої головоломки, придумати логічне рішення, зашифрувати його так, щоб розуміння вирішення задачі було неоднозначним але мало закономірну спільність між попередніми прикладами, що наведено вище. Також є рівні де потрібно думати тільки математично, але й тут може зустрітися логіка.

Візьмемо приклад із гри:

$$12=23$$

$$36=47$$

$$19=?$$

Формула тут така:  $ab$  ( це число ), потім  $a+1$ ,  $b+1$ .

Тобто потрібно до кожної цифри додати один, і вийде правильна відповідь. Аналогічно створюються наступні приклади з використанням додавання, множення, і інших математичних формул. Перед самою задачею додається приклади, які можуть наштовхнути на рішення задачі, тобто задачі з правильною відповіддю. Прослідковуючи логіку можна побачити в формулу по якій створений приклад логічної задачі.

Після того, як придумали приклад і формулу, можна переходити до графічної частини роботи. Весь дизайн гри було зроблено в Adobe Photoshop, після того як визначилися з кольором для фону, потрібно зробити кнопки, для цього потрібно взяти інструмент квадрат, вибрали колір, і в середині написати цифру від 0 до 9, також кнопки «math», «logic», «record», «exit», «training», «enter», «wrong». Також зробити

титульну сторінку гри, її також робити у Adobe Photoshop, беремо фон такий, як у рівнях, після цього робимо назву гри та хто її розробник. В грі є підказки, щоб подивитися на неї, потрібно натиснути на лампочку, після цього на екрані зверху з'явиться підказка.

Потім задачі потрібно зберегти в PNG, після цього можна перенести файл в Unity. Процес створення нових сцен з рівнями доволі простий, по готовому шаблону і написаного коду можна створювати безліч сцен та задач, оновлювати і розширювати можливості програмного додатку, це створення для зручності масштабування. Навіть не знаючи коду можна в подальшому розвивати Android додаток.

## **4.2. Інструкція використання мобільного додатку**

Для використання мобільного додатку з тестування логіки та математики:

1. Зайти в гру в Play Market, та встановити додаток на будь якому Android пристрої.
2. На головному екрані телефону знайти відповідний додаток та зайти до нього.
3. Після наведення на відкриття мобільного додатку буде відображено логотип Unity а після анімований логотип назви гри «Math and Logic» і розробника даного додатку.
4. На головному меню будуть кнопки, на яких буде написано «Math», «Logic», «Record», «Exit».
5. Після натискання на кнопку «Math» буде ще три кнопки «Test», «Training», «Record»
6. Кнопка «Record» відповідає за відображення рекорду часу за яке користувач пройшов випробування.



Кнопка «Exit» відповідає за вихід з програмного додатку, в випадку випадкового натиснення на кнопку, підпрограма перепитає, чи впевненні що ви хочете вийти з додатку.

Кнопка «Math» відповідає за підменю задач з математики.

Кнопка «Logic» відповідає за підменю задач з логіки.

7. Натискаємо на кнопку «Test»
8. Нас переносе вже в саму гру, і починається перший рівень.
9. В самому рівні є кнопки з цифрами від 0 до 9, а також кнопка «Enter», її потрібно натискати після того, як напишете відповідь на математично – логічний приклад.
10. Якщо користувач ввів більше 6 дозволених символів то поле вводу автоматично анулюється.
11. При не правильній відповіді на задачу впливає помилка що сигналізує про не правильну відповідь, яка згодом зникає і дає можливість спробувати ще раз.
12. Якщо користувач не в змозі вирішити задачу, він може використати підказку, яка покаже формулу чи відповідь, по якій можна вирішити математичну або логічну задачу.
13. Коректна відповідь на задачу відкриває вікно успішної відповіді і дає можливість перейти на наступну задачу.
14. Після завершення проходження тестування можна перейти до інших розділів, тренування, або пройти тест наново чи вийти з мобільного додатку.

## ВИСНОВКИ

Робота присвячена розробці мобільного додатку для розвитку логіки та математичних здібностей програміста. в процесі дослідження теми та мети було проведено:

1. Проведено дослідження, які обґрунтовують наукову новизну та актуальність роботи. А саме необхідність використання додаткових програм для розвитку математичних та логічних здібностей програміста, аналіз існуючих програмних додатків, їх переваги та недоліки. Було визначено, що існуючі застосунки розраховані лише для надання звичайних логічних задач, не спрямованих для розвитку логіки у програмістів.
2. Враховуючи переваги та недоліки існуючих програм було розроблено альтернативні задачі для розвитку та тестування логіки та математики за допомогою мобільного додатку. створений застосунок відповідає всім і функціональним можливостям, що відповідає вимогам.
3. Описано програмні засоби, середовища розробки, і програми для створення макету мобільного додатку. Визначено що оптимальним рішенням є використання середовища розробки Unity та мови програмування C#. встановлено що C# має переваги при створенні скриптів, не поступається якістю створення програмного забезпечення.
4. Застосунок буде ефективним для програмістів з будь якої сфери розробки, адже логіка там математика це невід'ємна частина даної професії. Створені задачі в мобільному додатку оптимізовані як для користувача податківця в програмуванні, так і для більш досвідчених програмістів.

Мобільний додаток може значно спростити процес навчання логіки та математики для тих хто хоче розвивати свої навички, використовуючи мобільний телефон.

## СПИСОК ЛІТЕРАТУРИ

1. Латеральна логіка, Головоломний шлях до нестандартного мислення. Автор - Гарет Мур 2018. 205с.
2. Питання юзабіліті. Мобільний UX для розробників та інших випадкових дизайнерів 2018. 392с.
3. <https://gs.statcounter.com/> [Електронний ресурс] : [Інтернет-портал] - Електронні дані. Режим доступу <https://gs.statcounter.com/> (дата звернення 15.04.2021). - Назва з екрана.
4. Прайс. С # 7 і .NET Core. Крос-платформна розробка для професіоналів. Марк Дж. 2017. 640с.
5. <https://unity.com/ru> [Електронний ресурс] : [Інтернет-портал] - Електронні дані - Режим доступу <https://unity.com/ru> (дата звернення 16.04.2021). - Назва з екрана.
6. Джозеф Хокинг, підручник «Unity в дії. Мультиплатформна розробка на C #. 2-е межд. видання » 2021.352с.
7. Unity і C #. Геймдев від ідеї до реалізації. Бонд Д. 2019. 928с.
8. Роберт Мартін, Чиста архітектура: мистецтво розробки програмного забезпечення. // Фабула. - 2019. 416с.
9. Грицюк Юрій, Аналіз вимог до програмного забезпечення. //Львівська політехніка. -2018. 456с.
10. Підручник Adobe Photoshop CC. Офіційний навчальний курс./ Ексмо. 2014. 456с.
11. <https://illustrator.demiart.ua> [Електронний ресурс] : [Інтернет-портал] - Електронні дані. [San Jose, Публічна організація “Adobe Inc”]. - Режим доступу [/book-adobe-illustrator/](https://illustrator.demiart.ua/book-adobe-illustrator/) (дата звернення 15.04.2021). - Назва з екрана.
12. <https://www.adobe.com/> [Електронний ресурс] : [Інтернет-портал] - Електронні дані. [San Jose, Публічна організація “Adobe Inc”]. - Режим доступу <https://www.adobe.com/> (дата звернення 15.04.2021). - Назва з екрана.

13. Akash Khandavilli, 100 Dos and Dont's UI/UX Design Tips eBook- Professional research based // Akash Khandavilli – 2020, 78с
14. Корі Сандлер, Гленфорд Майерс, Том Баджетт, Мистецтво тестування програм // Діалектика – 2016, 272с.
15. 327 математичних задач. Автор - Борис Анастасьевич Кордемский . 464с.

## Додаток А

Програмний код файлу Main\_menu.php

```
using System.Collections;
using System.Collections.Generic;
using UnityEngine;

public class ControlMenu : MonoBehaviour
{
    public GameObject buttonMenu;
    public GameObject buttonExit;

    void Start()
    {

    }

    void Update()
    {

    }

    public void ShowExitButton()
    {
        buttonMenu.SetActive(false);
        buttonExit.SetActive(true);
    }

    public void BackMenu()
    {
        buttonMenu.SetActive(true);
        buttonExit.SetActive(false);
    }

    public void Quit()
    {
        Application.Quit();
    }

    public void Math_menu()
    {
        SceneManager.LoadScene(1);
    }
}
```

```
public void Logic_menu()
{
    SceneManager.LoadScene(2);
}
public void Main_Menu()
{
    SceneManager.LoadScene(0);
}
public void LoadMath()
{
    SceneManager.LoadScene(3);
}
}
```

## Додаток Б

Програмний код файлу Control\_Panel.php

```
using System.Collections;
using System.Collections.Generic;
using UnityEngine;
using UnityEngine.UI;
using UnityEngine.SceneManagement;
public class CodePanel : MonoBehaviour
{
    // Start is called before the first frame update
    [SerializeField]
    Text codeText;
    string codeTextValue = "";
    public GameObject level1;
    public GameObject level2;
    public GameObject level3;
    public GameObject level4;
    public GameObject level5;
    public GameObject level6;
    public GameObject level7;
    public GameObject level8;
    public GameObject level9;
    public GameObject level10;
    public GameObject answer1;
    public GameObject answer2;
    public GameObject answer3;
    public GameObject answer4;
    public GameObject answer5;
    public GameObject answer6;
    public GameObject answer7;
    public GameObject answer8;
    public GameObject answer9;
    public GameObject answer10;
    public GameObject perfect;
    public GameObject great;
    int Keys = 1;
    void Start()
    {
```

```
}

// Update is called once per frame
void Update()
{

    codeText.text = codeTextValue;

    if (codeTextValue.Length >= 6)
        codeTextValue = "";

}

public void Enters()
{

    switch (Keys)
    {
        case 1:
            if (codeTextValue == "17")
            {
                GreatOn();
                Levels1();
                AnswerOff();
            }
            else
            {
                mist();
            }
            break;

        case 2:
            if (codeTextValue == "16")
            {
                GreatOn();
                Levels2();
                AnswerOff();
            }
            else
            {
```



```
        mist();
    }
    break;

case 3:
    if (codeTextValue == "10")
    {
        GreatOn();
        Levels3();
        AnswerOff();
    }
    else
    {
        mist();
    }
    break;

case 4:
    if (codeTextValue == "21094")
    {
        GreatOn();
        Levels4();
        AnswerOff();
    }
    else
    {
        mist();
    }
    break;

case 5:
    if (codeTextValue == "0")
    {
        GreatOn();
        Levels5();
        AnswerOff();
    }
    else
    {
        mist();
    }
}
```

```
break;

case 6:
  if (codeTextValue == "9")
  {
    GreatOn();
    Levels6();
    AnswerOff();
  }
  else
  {
    mist();
  }
  break;

case 7:
  if (codeTextValue == "12")
  {
    GreatOn();
    Levels7();
    AnswerOff();
  }
  else
  {
    mist();
  }
  break;

case 8:
  if (codeTextValue == "50")
  {
    GreatOn();
    Levels8();
    AnswerOff();
  }
  else
  {
    mist();
  }
  break;
```

```

    case 9:
        if (codeTextValue == "15")
        {
            GreatOn();
            Levels9();
            AnswerOff();
        }
        else
        {
            mist();
        }
        break;
    case 10:
        if (codeTextValue == "40")
        {
            GreatOn();
            Levels10();
            AnswerOff();
        }
        else
        {
            mist();
        }
        break;
    }

}

public void mist()
{

    codeTextValue = "Wrong";
    Invoke("Clear", 1);
}

public void Levels1()
{
    Keys = 2;
    level1.SetActive(false);
    level2.SetActive(true);
    codeTextValue = "";
}

```

```
public void Levels2()
{
    Keys = 3;
    level2.SetActive(false);
    level3.SetActive(true);
    codeTextValue = "";
}
public void Levels3()
{
    Keys = 4;
    level3.SetActive(false);
    level4.SetActive(true);
    codeTextValue = "";
}
public void Levels4()
{
    Keys = 5;
    level4.SetActive(false);
    level5.SetActive(true);
    codeTextValue = "";
}
public void Levels5()
{
    Keys = 6;
    level5.SetActive(false);
    level6.SetActive(true);
    codeTextValue = "";
}
public void Levels6()
{
    Keys = 7;
    level6.SetActive(false);
    level7.SetActive(true);
    codeTextValue = "";
}
public void Levels7()
{
    Keys = 8;
    level7.SetActive(false);
    level8.SetActive(true);
    codeTextValue = "";
```

```
}  
public void Levels8()  
{  
    Keys = 9;  
    level8.SetActive(false);  
    level9.SetActive(true);  
    codeTextValue = "";  
}  
public void Levels9()  
{  
    Keys = 10;  
    level9.SetActive(false);  
    level10.SetActive(true);  
    codeTextValue = "";  
}  
public void Levels10()  
{  
    SceneManager.LoadScene(1);  
}  
public void Great()  
{  
    perfect.SetActive(false);  
    great.gameObject.SetActive(false);  
}  
public void GreatOn()  
{  
    perfect.SetActive(true);  
    great.gameObject.SetActive(true);  
}  
  
public void Answer()  
{  
  
    switch (Keys)  
    {  
        case 1:  
            answer1.SetActive(true);  
            break;  
        case 2:  
            answer2.SetActive(true);  
            break;  
    }  
}
```

```
    case 3:
        answer3.SetActive(true);
        break;
    case 4:
        answer4.SetActive(true);
        break;
    case 5:
        answer5.SetActive(true);
        break;
    case 6:
        answer6.SetActive(true);
        break;
    case 7:
        answer7.SetActive(true);
        break;
    case 8:
        answer8.SetActive(true);
        break;
    case 9:
        answer9.SetActive(true);
        break;
    case 10:
        answer10.SetActive(true);
        break;
}
}
public void AnswerOff()
{
    answer1.SetActive(false);
    answer2.SetActive(false);
    answer3.SetActive(false);
    answer4.SetActive(false);
    answer5.SetActive(false);
    answer6.SetActive(false);
    answer7.SetActive(false);
    answer8.SetActive(false);
    answer9.SetActive(false);
    answer10.SetActive(false);
}
```

```
public void Clear()
{
    codeTextValue = "";
}
public void AddDigit(string digit)
{
    codeTextValue += digit;
}
public void Logic_menu()
{
    SceneManager.LoadScene(1);
}
}
```

## Додаток В

ДЕРЖАВНИЙ УНІВЕРСИТЕТ ТЕЛЕКОМУНІКАЦІЙ

НАВЧАЛЬНО-НАУКОВИЙ ІНСТИТУТ ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ



КАФЕДРА ІНЖЕНЕРІЇ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ



«РОЗРОБКА МОБІЛЬНОГО ДОДАТКУ ДЛЯ РОЗВИТКУ ТА  
ТЕСТУВАННЯ ЛОГІКИ ТА МАТЕМАТИЧНИХ ЗДІБНОСТЕЙ  
ПРОГРАМІСТА МОВОЮ C#»

Виконав студент 4 курсу

Групи ПД-42  
Івченко Дмитро Євгенович

Київ -2021

### МЕТА, ОБ'ЄКТ ТА ПРЕДМЕТ ДОСЛІДЖЕННЯ

**Мета роботи** – підвищення ефективності праці програміста шляхом тренування його логічних і математичних здібностей для покращення його роботи з використанням мобільного додатку.

**Об'єкт дослідження** – процес навчання та тестування логіки та математичних здібностей програміста.

**Предмет дослідження** – програмний додаток для навчання і розвитку логіки і математики.



## АНАЛОГИ



Мобільний додаток logiclike



Мобільний додаток  
Matches puzzle game



Мобільний додаток Brain Out



Мобільний додаток Math Riddles

3

## ТЕХНІЧНЕ ЗАВДАННЯ

1. Створення мобільного додатку для тренування та розвитку логіки та математичних здібностей.
2. Реалізація моделі програмного продукту.
3. Оптимізація роботи з пристроями на Android.
4. Редагування та створення рівнів та категорій задач.

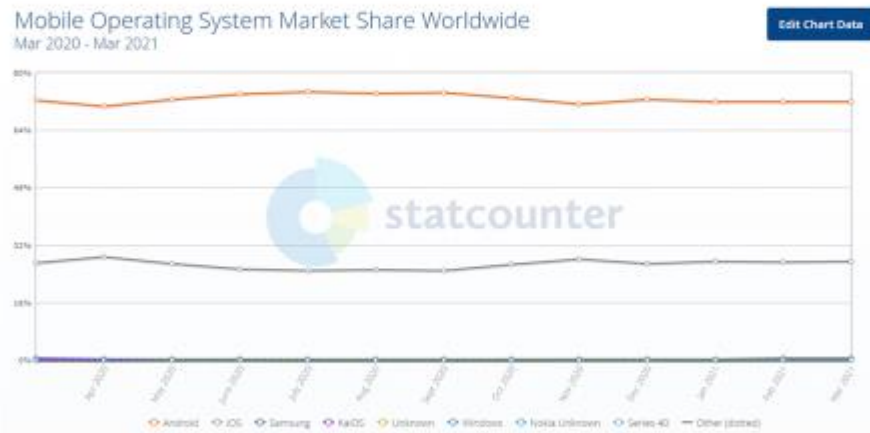
4

***Статистика користувачів мобільних пристроїв на квітень 2021 року виглядає наступним чином:***

- Android - 71.83%
- iOS - 27.41%
- інші платформи мають менше 0.5 відсотків долей ринку

5

**Таблиця використання ОС в світі**



6

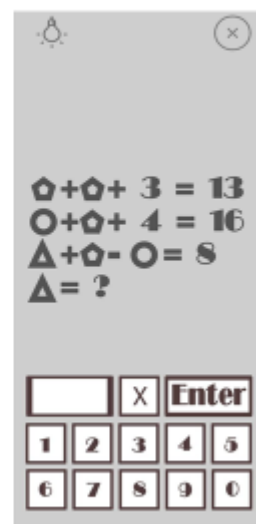
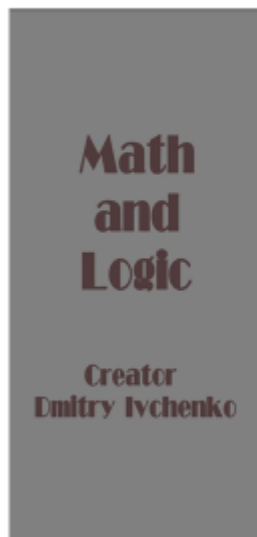
## Програми завдяки яким була зроблена гра



unity

7

## Візуальна частина роботи



8

## ВИСНОВКИ

Під час розробки мобільного додатку для розвитку та тренування логічним та математичних здібностей програміста було проведено аналіз вимог до інформаційної системи, виявлено і усунуто ряд недоліків на етапі створення макетів та створення задач математичного і логічного характеру. Мобільний додаток було протестовано за допомогою методів ручного тестування. Мобільний додаток готовий до використання програмістами будь якого рівня кваліфікація.