МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РОССИЙСКОЙ ФЕДЕРАЦИИ

ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ УЧРЕЖДЕНИЕ

КАЛИНИНГРАДСКОЙ ОБЛАСТИ

ПРОФЕССИОНАЛЬНАЯ ОБРАЗОВАТЕЛЬНАЯ ОРГАНИЗАЦИЯ

«КОЛЛЕДЖ ИНФОРМАЦИОННЫХ ТЕХНОЛОГИЙ И СТРОИТЕЛЬСТВА»

(ГБУ КО ПОО «КИТиС»)

|  |  |  |
| --- | --- | --- |
| Курсовой проект  допущен к защите  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  (дата)  Зам.директора по УМР  \_\_\_\_\_\_\_\_\_\_\_\_\_Павленко Г.Я.  (подпись) |  | Курсовой проект  защищен с оценкой  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  (оценка)  Руководитель работы  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  (подпись, дата) |

Пояснительная записка к курсовому проекту

по дисциплине: МДК 01.01 Разработка программных модулей

Тема: Разработка мобильной игры под Android «Крестики-нолики»

Специальность 09.02.07 «Информационные системы и программирование»

|  |  |
| --- | --- |
| Выполнил: | студент 3 курса,  группы ИСп19-1  Арипджанова Малика Талатовна  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  (подпись) |
| Руководитель: | Большакова-Стрекалова Анна Викторовна, преподаватель ГБУ КО ПОО КИТиС  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  (подпись) |

Калининград

2022

Содержание

|  |  |
| --- | --- |
| Введение | 3 |
| 1. Описание предметной области | 4 |
| * 1. Аналоги разрабатываемого приложения | 4 |
| * 1. Техническое задание | 7 |
| * 1. Описание структуры приложения | 8 |
| 1. Описание разработки приложения | 10 |
| * 1. Обоснование средств разработки | 10 |
| * 1. Разработка интерфейса | 10 |
| * 1. Разработка логики работы приложения. Схема взаимодействия компонентов проекта | 13 |
| * 1. Описание переменных, компонентов, классов и подпрограмм | 14 |
| 1. Тестирование и установка приложения | 16 |
| Заключение | 18 |
| Список использованной литературы | 19 |
| Приложение | 20 |

Введение

Крестики-нолики — логическая игра между двумя противниками на квадратном поле 3 на 3 клетки или большего размера (вплоть до “бесконечного поля”). Один из игроков играет “крестиками”, второй — “ноликами”. В традиционной китайской игре используются черные и белые камни [3].

Данная работа посвящена разработке мобильного приложения “Крестики-нолики”.

Актуальность работы заключается в том, чтобы протестировать аналоги разрабатываемого приложения, научиться разрабатывать собственное приложение и закрепить знания. Основной задачей является создание приложения для пользователей. Разрабатываемое приложение будет служить для развлечения пользователей, для того, чтобы скрасить время ожидания пользователей, ведь в любой работе нужен отдых, а данное приложение поможет расслабиться и отвлечься от повседневных дел.

Цель работы является разработка мобильной игры под Android «Крестики-нолики».

Задачи:

1. Ознакомление с аналогами приложений.
2. Описание структуры индивидуального приложения.
3. Описание платформы для разрабатываемого приложения.
4. Разработка интерфейса.
5. Запуск и тестирование приложения.
6. Описание предметной области
   1. Аналоги разрабатываемого приложения

Игра Крестики-Нолики 3х3 (таблица 1, рисунок 1, 2) [1].

Таблица 1 – Описание игры

|  |  |
| --- | --- |
| Категория | Пояснение |
| Описание | Классическая головоломка для двоих снова возвращается на ваши экраны, кстати перебралась она из школьных тетрадей, черновиков и в конце концов заборов. Именно на этих поверхностях происходили настоящие баталии за право быть лучшим в крестиках-ноликах. |
| Управление в игре | Мышкой |
| Достоинства | Можно играть против компьютера |
| Недостатки | Играть можно только в полноэкранном режиме |
| Интерфейс | Просто интерфейс |
| Работа программы | Работает исправно. |

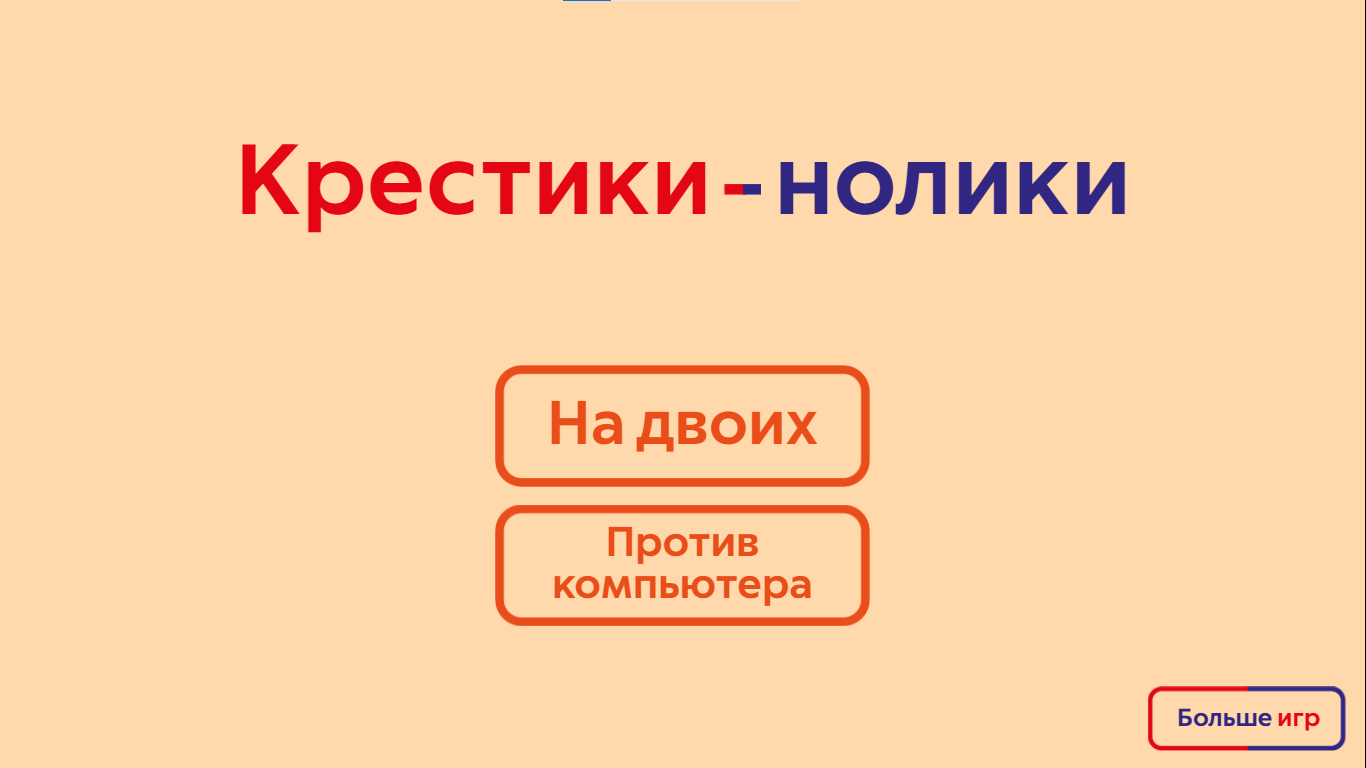


Рисунок 1 - Меню программы



Рисунок 2 – Поле ходов

Игра Гигантские Крестики Нолики (таблица 2, рисунок 3,4) [2].

Таблица 2 – Описание игры

|  |  |
| --- | --- |
| Категория | Пояснение |
| Описание | Гигантские Крестики Нолики – это простая и всем известная игра, в которой можно развлечь себя на многие часы. Для этого вам предлагается сыграть с компьютером, который достаточно смышленый в подобных игрушках. Если вы никогда не играли в крестики нолики, то лучше не начинать игру на большом поле. Начните с маленького, чтобы приспособиться к правилам. |
| Управление в игре | Мышкой |
| Достоинства | Можно играть против компьютера, выбрать размер поля, регулировать звук |
| Недостатки | Присутствует реклама |
| Интерфейс | Просто интерфейс |
| Работа программы | Работает исправно. |

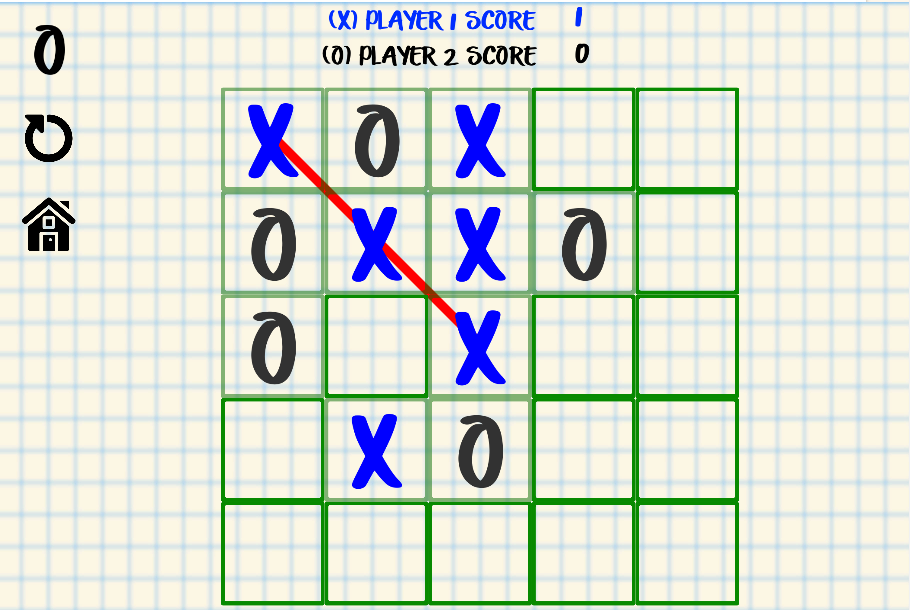


Рисунок 3 – Подсчет выигрышных баллов

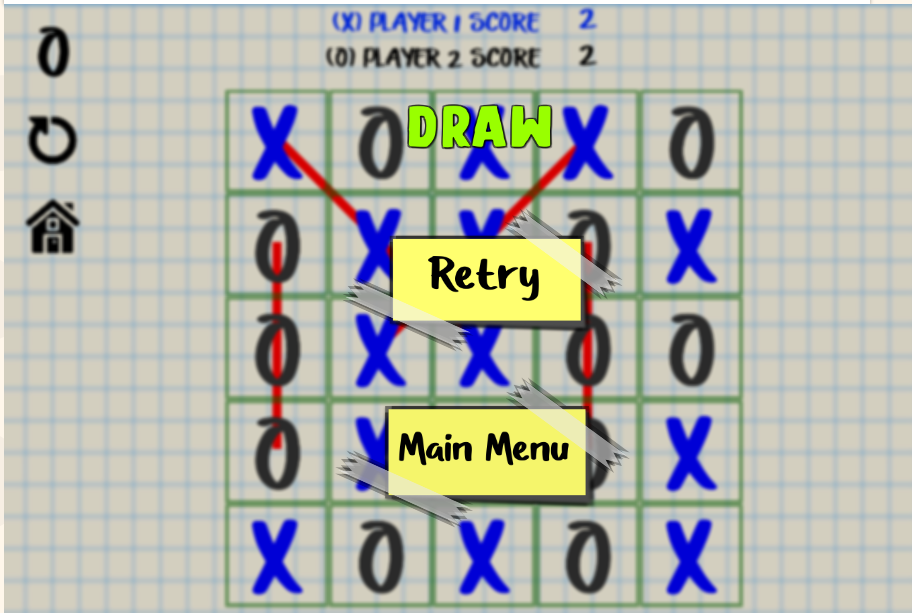


Рисунок 4 – Результат победителей

* 1. Техническое задание

Наименование проекта – “TIC TAC TOE”

Краткая характеристика области применения:

Приложение разработано для развлечения пользователей, которые хотят скрасить время ожидания, расслабится или отвлечься от повседневных дел.

Основания для разработки [4]:

Гост 19.201-78, Гост 7.32-2001, Гост 2.105–2019

Эксплуатационное назначение:

Для развлечения пользователей

Требования к программе или программному изделию

Требование к составу выполняемых функций

Программа должна обеспечивать возможность выполнения ниже перечисленных пунктов:

1. Переход между активностями по нажатию на кнопку.
2. Возможность ввода имён игроков.
3. Возможность делать ходы на поле.
4. Объявлять победителя.
5. Возможность начать все сначала по нажатию на кнопку.

Технико-экономические показатели:

Ориентировочная экономическая эффективность не рассчитывается

Стадии и этапы разработки:

* разработка технического задания;
* рабочее проектирование;
* внедрение;

На этапе разработки технического задания должны быть выполнены перечисленные ниже работы [5]:

* постановка задачи;
* определение и уточнение требований к техническим средствам;
* определение требований к программе;
* определение стадий, этапов и сроков разработки программы и документации на неё;
* выбор языков программирования;
* согласование и утверждение технического задания.
  1. Описание структуры приложения

Данное приложение позволяет пользователям вводить имена игроков, а также начать игру сначала. Само приложение меняет очередь игроков для ввода и объявляет победителя.

Визуальное представление структуры приложения:



Рисунок 5 - Описание структуры приложения

1. Описание разработки приложения
   1. Обоснование средств разработки

Для разработки приложения крестики-нолики есть возможность использования таких сред разработки, как: Python, Visual Studio (C++, C#), Delphi и т.п.

В данной работе используется среда разработки Android Studio, и язык программирования Java, так как этот язык достаточно удобный.

* 1. Разработка интерфейса

При запуске приложения, в первую очередь, появляется главный экран (Рисунок 6).

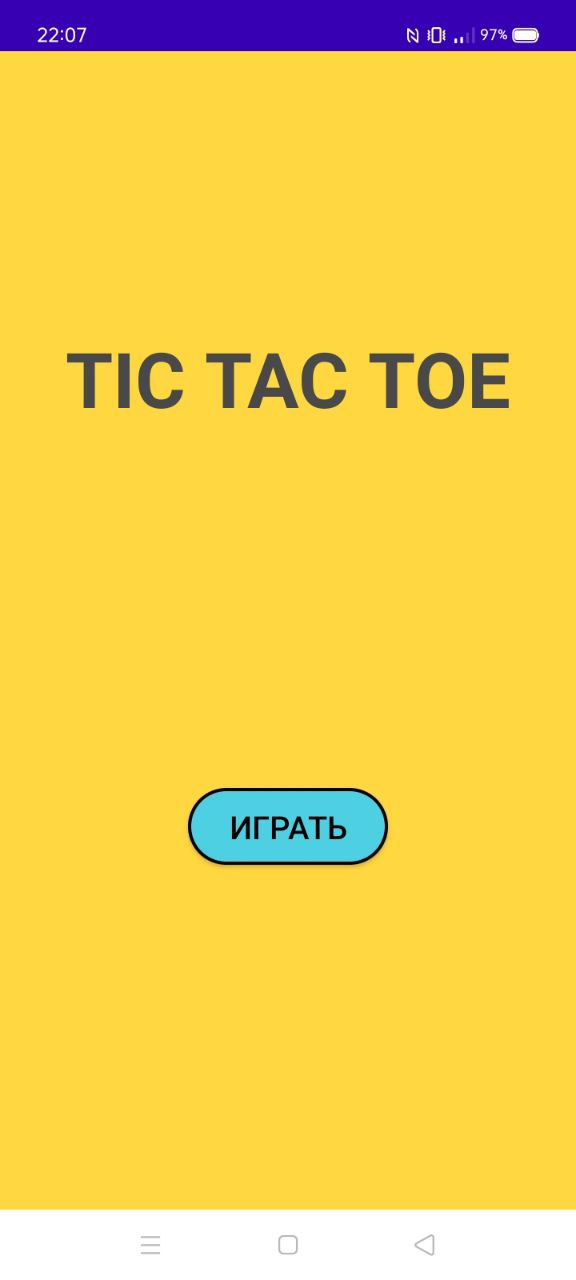


Рисунок 6 – Главное меню

На экране показано главное окно с кнопкой “Играть” и название приложения. Сама кнопка выполняет функцию перехода на другой activity (Рисунок 7).

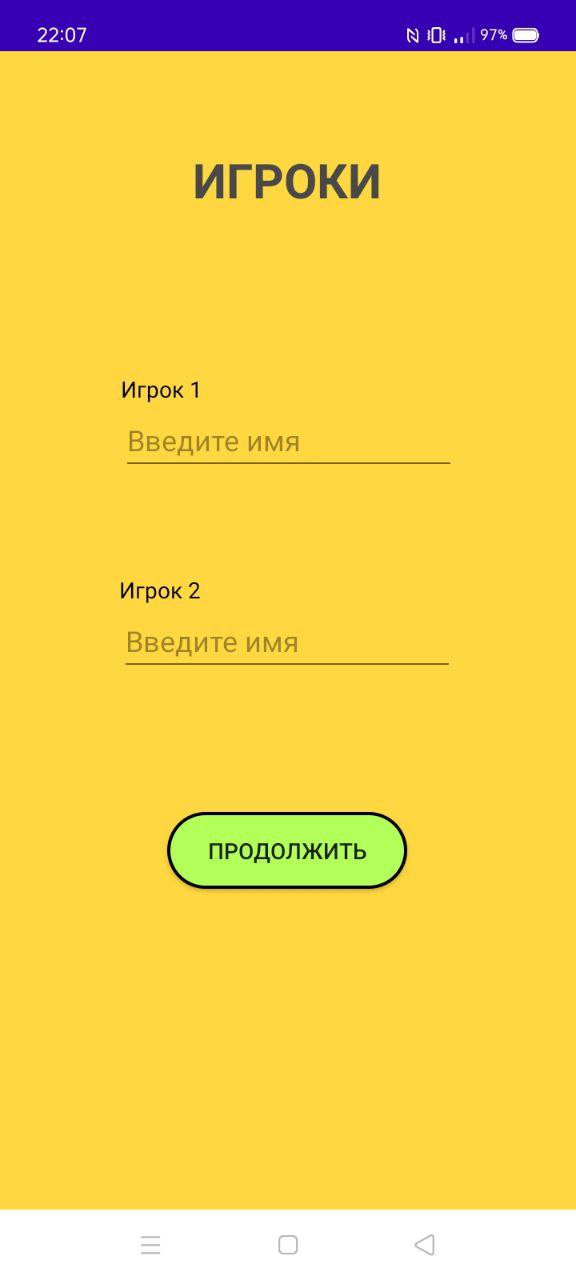


Рисунок 7 – Меню ввода имени игроков

По переходе в другое activity появляется меню для ввода имени игроков с кнопкой “Продолжить”. По нажатию на кнопку происходит переход на поле ходов с размером 3 на 3 (Рисунок – 8).

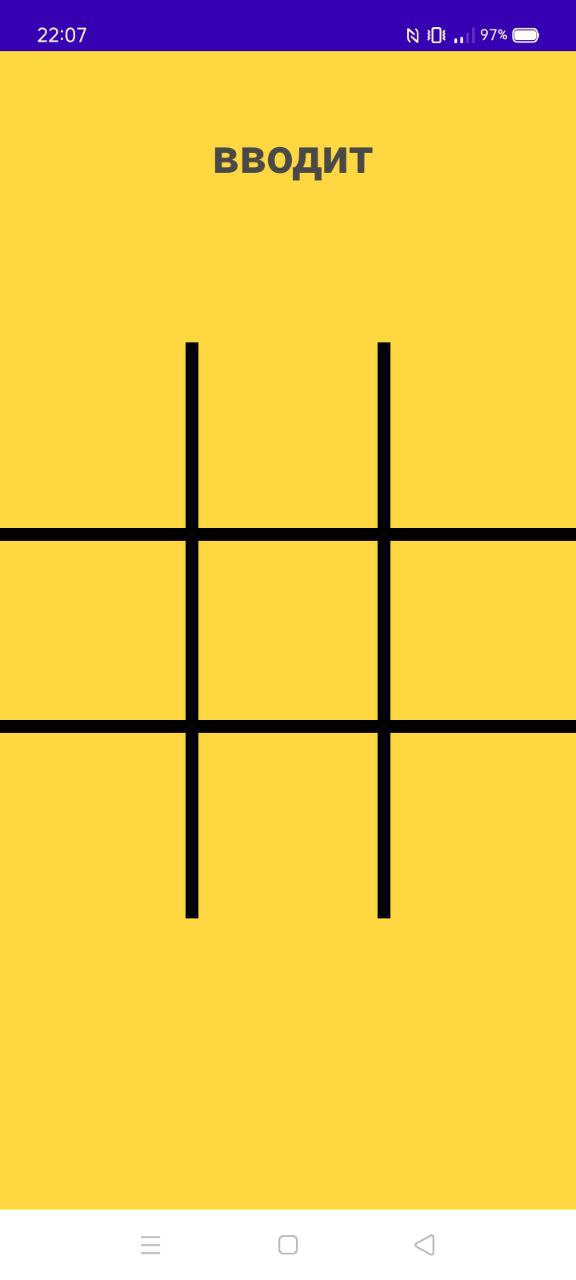


Рисунок 8 – Поле для ввода X и O

При выполнении нескольких ходов и объявления результата в activity появляется две кнопки: “Начать заново” – кнопка для того, чтобы начать игру сначала, не меняя при этом игроков; “Выйти” – кнопка для перехода в главное окно (Рисунок 9).

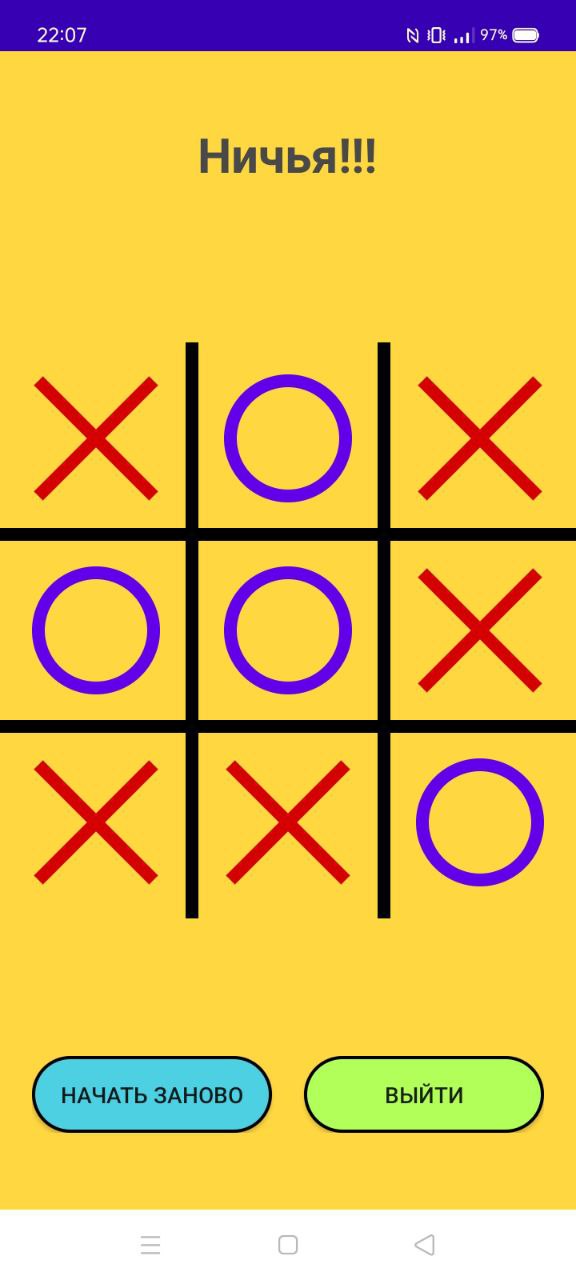


Рисунок 9 – Окно с результатом игры

* 1. Разработка логики работы приложения. Схема взаимодействия компонентов проекта

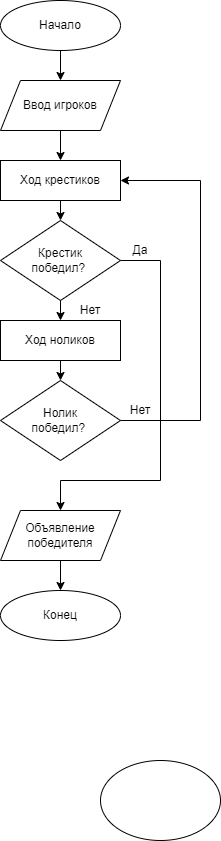


Рисунок 10 – Описание функционала мобильной игры крестики-нолики

* 1. Описание переменных, компонентов, классов и подпрограмм

Таблица 3 – Методы

|  |  |
| --- | --- |
| Название | Назначение |
| onCreate | Привязка к активностям. |
| homePress | Переход между активностями MainActivity |
| submitButtonClick | Переход между активностями Game |
| playButtonClick | Переход между активностями Player |

Таблица 4 – Классы

|  |  |
| --- | --- |
| Название | Назначение |
| TicTacToe | Отвечает за поле с размером 3 на 3 |
| MainActivity | Класс по умолчанию. Главное меню и действия в нем. |
| GameLogic | Отвечает за определения победителя |
| Player | Отвечает за ввод имени игроков |
| Game | Отвечает за действие кнопок “Начать заново” и “Выйти” |

1. Тестирование и установка приложения

Данное разработанное приложение будет устанавливаться на реальное устройство – смартфон с OS Android. Для этого надо:

1. Подключить телефон к компьютеру через USB-провод.
2. Через настройки телефона, в отделе для разработчиков включить отладчик USB.

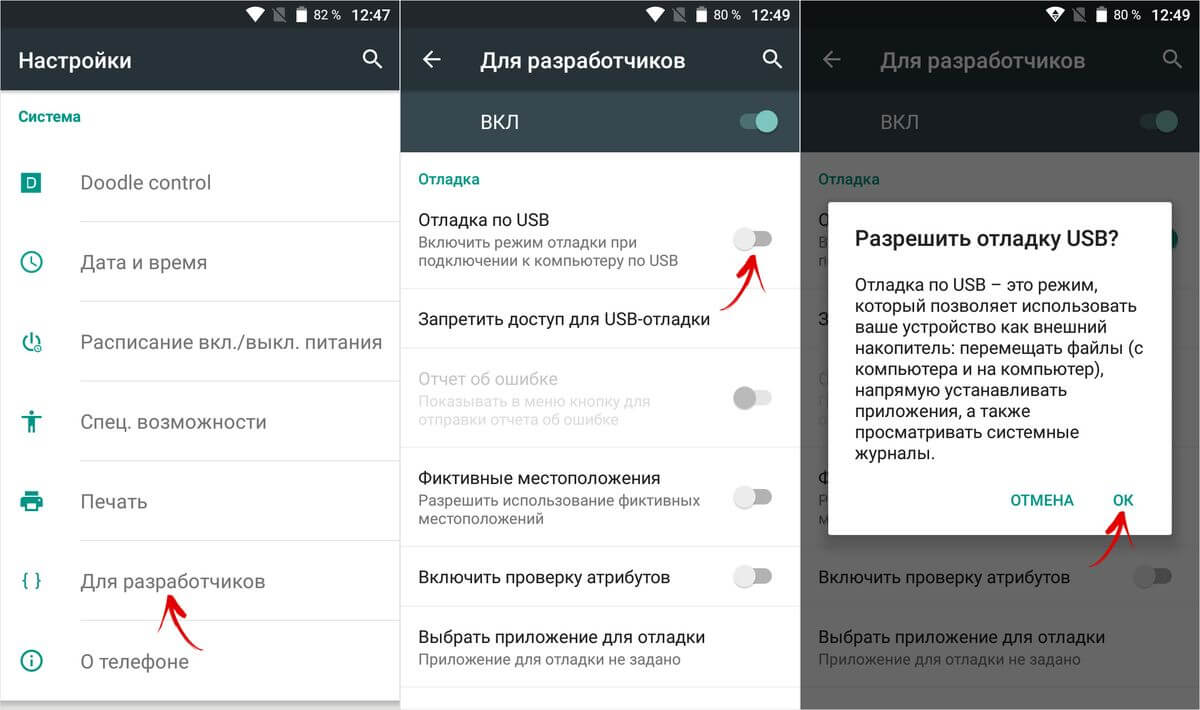


Рисунок 11 – Включение отладки по USB

1. Далее запускаем проект.

Проверка приложения на перегрузку, на различных эмуляторах.

Разработанное приложение выдерживает перегрузку на различных эмуляторах:

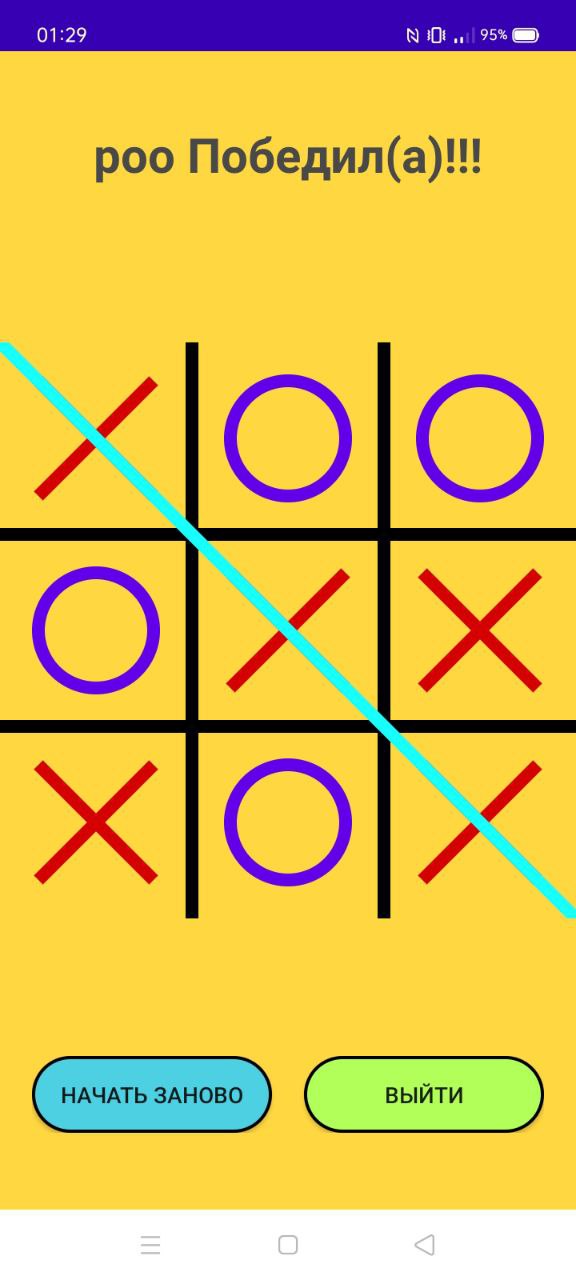


Рисунок 12 – Проверка на перегрузку

Проверка приложения на удобство пользователя во время использования:

В разработанном приложении удобнее всего было бы, если кнопка “Выйти” полностью выходила бы из приложения, а не возвращалась в главное окно. Но такой возможности нет.

Заключение

Игровое приложение предназначено для людей любого возраста. Игра не требует особых умственных способностей, определенного умственного развития, программа очень легка в использовании. Чтобы запустить и использовать приложение не требуется наличие особых навыков пользования смартфоном. Программа поможет скоротать время в перерыве между работой, а также можно использовать как полноценную мобильную игру для развлечения с друзьями и коллегами.

Игра «Крестики-нолики» может быть установлена в учебных компьютерных классах различного рода и поможет неопытным пользователям овладеть навыками работы с персональным компьютером.

Несмотря на внешнюю простоту, программа представляет собой сложную систему взаимодействующих друг с другом функций. Потребовалось немало сил и времени на разработку этого приложения. Кроме того, потребовалось привлечение большого объема дополнительной информации.

В процессе выполнения курсовой работы были выполнены все поставленные задачи и разработка популярной игры «Крестики-нолики» прошла успешно.

Список использованной литературы

1. [Игра Крестики-Нолики 3х3 - Играть Онлайн! (igroutka.ru)](https://igroutka.ru/igry-krestiki-noliki/39384-krestiki-noliki-3h3.html).
2. [Игра Гигантские Крестики Нолики - Играть Онлайн! (igroutka.ru)](https://igroutka.ru/igry-krestiki-noliki/30599-gigantskie-krestiki-noliki.html).
3. [Крестики-нолики. Правила игры в крестики-нолики. (pravilaigr.ru)](https://www.pravilaigr.ru/xo.php?).
4. [Техническое задание на программу по ГОСТ 19.201-78 - Информационные технологии в профессиональной деятельности (google.com)](https://www.sites.google.com/site/ktnosciencergpu/Home/zan/z5/z55/z551).
5. [ГОСТ 19.102-77 Единая система программной документации (ЕСПД). Стадии разработки от 20 мая 1977 - docs.cntd.ru](https://docs.cntd.ru/document/1200007628).
6. Java. Методы программирования", Блинов, Романчик, издательство Четыре четверти, 2013 год.
7. Java. Справочник разработчика", Бенджамин Дж. Эванс, Дэвид Флэнаган, издательство Диалектика-Вильямс, 2019 год.
8. "Java. Руководство для начинающих", Герберт Шилдт.
9. "Java для чайников", Барри Бёрд.
10. "Java. Полное руководство", Герберт Шилдт.

Приложение А

activity\_main.xml

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

<androidx.constraintlayout.widget.ConstraintLayout 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:layout\_width="match\_parent"

android:layout\_height="match\_parent"

tools:context=".MainActivity"

android:background="#FFD740">

<TextView

android:id="@+id/homePageTitle"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:layout\_marginTop="172dp"

android:text="@string/appTitle"

android:textColor="#494949"

android:textSize="48sp"

android:textStyle="bold"

app:layout\_constraintEnd\_toEndOf="parent"

app:layout\_constraintHorizontal\_bias="0.498"

app:layout\_constraintStart\_toStartOf="parent"

app:layout\_constraintTop\_toTopOf="parent" />

<Button

android:id="@+id/playBTN"

android:layout\_width="125dp"

android:layout\_height="wrap\_content"

android:layout\_marginTop="224dp"

android:background="@drawable/green\_button"

android:onClick="playButtonClick"

android:text="@string/playButton"

android:textColor="#000000"

android:textSize="20sp"

app:layout\_constraintEnd\_toEndOf="parent"

app:layout\_constraintStart\_toStartOf="parent"

app:layout\_constraintTop\_toBottomOf="@+id/homePageTitle" />

</androidx.constraintlayout.widget.ConstraintLayout>

game.xml

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

<androidx.constraintlayout.widget.ConstraintLayout 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"

xmlns:custom="http://schemas.android.com/apk/res-auto"

android:layout\_width="match\_parent"

android:layout\_height="match\_parent"

android:background="#FFD740"

tools:context=".Game">

<TextView

android:id="@+id/displayTurn"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:layout\_marginTop="44dp"

android:textColor="#494949"

android:textSize="30sp"

android:textStyle="bold"

app:layout\_constraintEnd\_toEndOf="parent"

app:layout\_constraintHorizontal\_bias="0.498"

app:layout\_constraintStart\_toStartOf="parent"

app:layout\_constraintTop\_toTopOf="parent" />

<androidx.constraintlayout.widget.ConstraintLayout

android:id="@+id/constraintLayout"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:layout\_marginBottom="48dp"

app:layout\_constraintBottom\_toBottomOf="parent"

app:layout\_constraintEnd\_toEndOf="parent"

app:layout\_constraintHorizontal\_bias="0.497"

app:layout\_constraintStart\_toStartOf="parent">

<Button

android:id="@+id/playAgainButton"

android:layout\_width="150dp"

android:layout\_height="wrap\_content"

android:background="@drawable/green\_button"

android:onClick="playAgainPress"

android:text="@string/playAgainButton"

app:layout\_constraintBottom\_toBottomOf="parent"

app:layout\_constraintStart\_toStartOf="parent"

app:layout\_constraintTop\_toTopOf="parent" />

<Button

android:id="@+id/homeButton"

android:layout\_width="150dp"

android:layout\_height="wrap\_content"

android:layout\_marginStart="20dp"

android:background="@drawable/color\_button"

android:onClick="homePress"

android:text="@string/homeButton"

app:layout\_constraintBottom\_toBottomOf="parent"

app:layout\_constraintEnd\_toEndOf="parent"

app:layout\_constraintStart\_toEndOf="@+id/playAgainButton"

app:layout\_constraintTop\_toTopOf="parent" />

</androidx.constraintlayout.widget.ConstraintLayout>

<com.example.tictactoe.TicTacToe

android:id="@+id/TicTacToe"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

app:layout\_constraintBottom\_toBottomOf="parent"

app:layout\_constraintEnd\_toEndOf="parent"

app:layout\_constraintStart\_toStartOf="parent"

app:layout\_constraintTop\_toTopOf="parent"

custom:OColor="#6200EA"

custom:XColor="#D50000"

custom:boardColor="@color/black"

custom:winningLineColor="#18FFFF" />

</androidx.constraintlayout.widget.ConstraintLayout>

player.xml

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

<androidx.constraintlayout.widget.ConstraintLayout 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:layout\_width="match\_parent"

android:layout\_height="match\_parent"

tools:context=".Player"

android:background="#FFD740">

<TextView

android:id="@+id/playerSetupTitle"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:layout\_marginTop="60dp"

android:text="@string/playerSetup"

android:textColor="#494949"

android:textSize="30sp"

android:textStyle="bold"

app:layout\_constraintEnd\_toEndOf="parent"

app:layout\_constraintHorizontal\_bias="0.498"

app:layout\_constraintStart\_toStartOf="parent"

app:layout\_constraintTop\_toTopOf="parent" />

<EditText

android:id="@+id/player1Name"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:layout\_marginTop="120dp"

android:ems="10"

android:inputType="textPersonName"

android:hint="@string/NameHint"

app:layout\_constraintEnd\_toEndOf="parent"

app:layout\_constraintHorizontal\_bias="0.502"

app:layout\_constraintStart\_toStartOf="parent"

app:layout\_constraintTop\_toBottomOf="@+id/playerSetupTitle"

android:importantForAutofill="no"/>

<EditText

android:id="@+id/player2Name"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:layout\_marginTop="80dp"

android:ems="10"

android:inputType="textPersonName"

android:hint="@string/NameHint"

app:layout\_constraintEnd\_toEndOf="parent"

app:layout\_constraintHorizontal\_bias="0.497"

app:layout\_constraintStart\_toStartOf="parent"

app:layout\_constraintTop\_toBottomOf="@+id/player1Name"

android:importantForAutofill="no"/>

<TextView

android:id="@+id/player1Label"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:text="@string/Label1"

android:textColor="#000000"

app:layout\_constraintBottom\_toTopOf="@+id/player1Name"

app:layout\_constraintStart\_toStartOf="@+id/player1Name" />

<TextView

android:id="@+id/player2Label"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:text="@string/Label2"

android:textColor="#000000"

app:layout\_constraintBottom\_toTopOf="@+id/player2Name"

app:layout\_constraintStart\_toStartOf="@+id/player2Name" />

<Button

android:id="@+id/submitBTN"

android:layout\_width="150dp"

android:layout\_height="wrap\_content"

android:layout\_marginTop="84dp"

android:background="@drawable/color\_button"

android:onClick="submitButtonClick"

android:text="@string/submitButton"

app:layout\_constraintEnd\_toEndOf="parent"

app:layout\_constraintHorizontal\_bias="0.498"

app:layout\_constraintStart\_toStartOf="parent"

app:layout\_constraintTop\_toBottomOf="@+id/player2Name" />

</androidx.constraintlayout.widget.ConstraintLayout>

color\_button.xml/green\_button.xml

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

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

<item>

<shape android:shape="rectangle">

<solid android:color="#B2FF59"/> //для зеленой кнопки меняем цвет на #4DD0E1

<corners android:radius="100dp"/>

<stroke android:color="@color/black" android:width="2dp"/>

</shape>

</item>

</selector>

art.xml

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

<resources>

<declare-styleable name="TicTacToe">

<attr name="boardColor" format="color"/>

<attr name="XColor" format="color"/>

<attr name="OColor" format="color"/>

<attr name="winningLineColor" format="color"/>

</declare-styleable>

</resources>

strings.xml

<resources>  
 <string name="app\_name">TiC tAc ToE</string>  
 <string name="title\_activity\_game">GameDisplay</string>  
 <string name="appTitle">TIC TAC TOE</string>  
 <string name="playButton">ИГРАТЬ</string>  
 <string name="playerSetup">ИГРОКИ</string>  
 <string name="Label1">Игрок 1</string>  
 <string name="Label2">Игрок 2</string>  
 <string name="submitButton">ПРОДОЛЖИТЬ</string>  
 <string name="NameHint">Введите имя</string>  
 <string name="playAgainButton">НАЧАТЬ ЗАНОВО</string>  
 <string name="homeButton">ВЫЙТИ</string>  
 <string name="tieGame">Ничья!!!</string>  
</resources>

themes.xml(values\night)

...

parent="Theme.AppCompat.Light.NoActionBar">

...

MainActivity.java

package com.example.tictactoe;  
import androidx.appcompat.app.AppCompatActivity;  
import android.content.Intent;  
import android.os.Bundle;  
import android.view.View;  
public class MainActivity extends AppCompatActivity {  
 @Override  
 protected void onCreate(Bundle savedInstanceState) {  
 super.onCreate(savedInstanceState);  
 setContentView(R.layout.activity\_main);  
 }  
 public void playButtonClick(View view){  
 Intent intent = new Intent(this, Player.class);  
 startActivity(intent);  
 }  
}

TicTacToe.java

package com.example.tictactoe;  
import android.annotation.SuppressLint;  
import android.content.Context;  
import android.content.res.TypedArray;  
import android.graphics.Canvas;  
import android.graphics.Paint;  
import android.util.AttributeSet;  
import android.view.MotionEvent;  
import android.view.View;  
import android.widget.Button;  
import android.widget.TextView;  
import androidx.annotation.Nullable;  
class TicTacToe extends View {  
 private final GameLogic game;  
 private final int gameBoardColor;  
 private final int XColor;  
 private final int OColor;  
 private final int diagonalColor;  
 private boolean winningLine = false;  
 private int cellSize = getWidth() / 3;  
 private final Paint paint = new Paint();  
 public TicTacToe(Context context, @Nullable AttributeSet attrs) {  
 super(context, attrs);  
 game = new GameLogic();  
 TypedArray a = context.getTheme().obtainStyledAttributes(attrs,  
 R.styleable.TicTacToe, 0, 0);  
 try {  
 gameBoardColor = a.getInteger(R.styleable.TicTacToe\_boardColor, 0);  
 XColor = a.getInteger(R.styleable.TicTacToe\_XColor, 0);  
 OColor = a.getInteger(R.styleable.TicTacToe\_OColor, 0);  
 diagonalColor = a.getInteger(R.styleable.TicTacToe\_winningLineColor, 0);  
 }finally {  
 a.recycle();  
 }  
 }  
 @Override  
 protected void onMeasure(int width, int height){  
 super.onMeasure(width, height);  
 int dimension = Math.min(getMeasuredWidth(), getMeasuredHeight());  
 cellSize = dimension/3;  
 setMeasuredDimension(dimension, dimension);  
 }  
 @SuppressLint("ClickableViewAccessibility")  
 @Override  
 public boolean onTouchEvent(MotionEvent event){  
 float x = event.getX();  
 float y = event.getY();  
 int action = event.getAction();  
 if (action == MotionEvent.ACTION\_DOWN){  
 int row = (int) Math.ceil(y/cellSize);  
 int col = (int) Math.ceil(x/cellSize);  
 if (!winningLine){  
 if (game.updateGameBoard(row, col)){  
 invalidate();  
 if (game.winnerCheck()){  
 winningLine = true;  
 invalidate();  
 }  
 //updates the actual game code  
 if (game.getPlayer() % 2 == 0){  
 game.setPlayer(game.getPlayer() - 1);  
 }  
 else {  
 game.setPlayer(game.getPlayer() + 1);  
 }  
 }  
 }  
 return true;  
 }  
 else{  
 return false;  
 }  
 }  
 @Override  
 protected void onDraw(Canvas canvas){  
 paint.setStyle(Paint.Style.STROKE);  
 paint.setAntiAlias(true);  
 drawGameBoard(canvas);  
 drawMarkers(canvas);  
 if (winningLine){  
 paint.setColor(diagonalColor);  
 drawWinningLine(canvas);  
 }  
 }  
 private void drawGameBoard(Canvas canvas){  
 paint.setColor(gameBoardColor);  
 paint.setStrokeWidth(16);  
 for (int c=1; c<3; c++){  
 canvas.drawLine(cellSize\*c, 0, cellSize\*c, canvas.getHeight(), paint);  
 }  
 for (int r=1; r<3; r++){  
 canvas.drawLine(0, cellSize\*r, canvas.getWidth(), cellSize\*r, paint);  
 }  
 }  
 private void drawMarkers(Canvas canvas){  
 //will draw an X or O onto the board  
 for (int r=0; r<3; r++){  
 for (int c=0; c<3; c++){  
 if (game.getGameBoard()[r][c] != 0){  
 if (game.getGameBoard()[r][c] == 1){  
 drawX(canvas, r, c);  
 }  
 else {  
 drawO(canvas, r, c);  
 }  
 }  
 }  
 }  
 }  
 private void drawX(Canvas canvas, int row, int col){  
 paint.setColor(XColor);  
 //positive sloping line  
 canvas.drawLine((float) ((col+1)\*cellSize - cellSize\* 0.2),  
 (float) (row\*cellSize + cellSize\* 0.2),  
 (float) (col\*cellSize + cellSize\* 0.2),  
 (float) ((row+1)\*cellSize - cellSize\* 0.2),  
 paint);  
 //negative sloping line  
 canvas.drawLine((float) (col\*cellSize + cellSize\* 0.2),  
 (float) (row\*cellSize + cellSize\* 0.2),  
 (float) ((col+1)\*cellSize - cellSize\* 0.2),  
 (float) ((row+1)\*cellSize - cellSize\* 0.2),  
 paint);  
 }  
 private void drawO(Canvas canvas, int row, int col){  
 paint.setColor(OColor);  
 canvas.drawOval((float)(col\*cellSize + cellSize\* 0.2),  
 (float)(row\*cellSize + cellSize\* 0.2),  
 (float) ((col\*cellSize+cellSize) - cellSize\* 0.2),  
 (float) ((row\*cellSize+cellSize) - cellSize\* 0.2),  
 paint);  
 }  
 private void drawWinningLine(Canvas canvas){  
 int row = game.getWinType()[0];  
 int col = game.getWinType()[1];  
 switch (game.getWinType()[2]){  
 case 1:  
 //draw Horizontal Line  
 drawHorizontalLine(canvas, row, col);  
 break;  
 case 2:  
 //draw Vertical Line  
 drawVerticalLine(canvas, row, col);  
 break;  
 case 3:  
 //draw Diagonal Line Neg  
 drawDiagonalLineNeg(canvas);  
 break;  
 case 4:  
 //draw Diagonal Line Pos  
 drawDiagonalLinePos(canvas);  
 break;  
 }  
 }  
 private void drawHorizontalLine(Canvas canvas, int row, int col){  
 canvas.drawLine(col, row\*cellSize + (float)cellSize/2,  
 cellSize\*3, row\*cellSize + (float)cellSize/2,  
 paint);  
 }  
 private void drawVerticalLine(Canvas canvas, int row, int col){  
 canvas.drawLine(col\*cellSize + (float)cellSize/2, row,  
 col\*cellSize + (float)cellSize/2, cellSize\*3,  
 paint);  
 }  
 private void drawDiagonalLinePos(Canvas canvas){  
 canvas.drawLine(0, cellSize\*3, cellSize\*3, 0, paint);  
 }  
 private void drawDiagonalLineNeg(Canvas canvas){  
 canvas.drawLine(0, 0, cellSize\*3, cellSize\*3, paint);  
 }  
 public void setUpGame(Button playAgain, Button home, TextView playerDisplay, String[] names){  
 game.setPlayAgainBTN(playAgain);  
 game.setHomeBTN(home);  
 game.setPlayerTurn(playerDisplay);  
 game.setPlayerNames(names);  
 }  
 public void resetGame(){  
 game.resetGame();  
 winningLine = false;  
 }  
}

Game.java

package com.example.tictactoe;  
import androidx.appcompat.app.AppCompatActivity;  
import android.content.Intent;  
import android.os.Bundle;  
import android.view.View;  
import android.widget.Button;  
import android.widget.TextView;  
public class Game extends AppCompatActivity {  
 private TicTacToe ticTacToe;  
 private Button playAgainButton;  
 private Button homeButton;  
 private TextView playerTurn;  
 private String[] playerNames;  
 @Override  
 protected void onCreate(Bundle savedInstanceState) {  
 super.onCreate(savedInstanceState);  
 setContentView(R.layout.game);  
 playAgainButton = findViewById(R.id.playAgainButton);  
 homeButton = findViewById(R.id.homeButton);  
 playerTurn = findViewById(R.id.displayTurn);  
 playAgainButton.setVisibility(View.GONE);  
 homeButton.setVisibility(View.GONE);  
 ticTacToe = findViewById(R.id.TicTacToe);  
 playerNames = getIntent().getStringArrayExtra("PLAYER\_NAMES");  
 if (playerNames != null) {  
 playerTurn.setText((playerNames[0] + " вводит"));  
 }  
 ticTacToe.setUpGame(playAgainButton, homeButton, playerTurn, playerNames);  
 }  
 public void playAgainPress(View view){  
 ticTacToe.resetGame();  
 ticTacToe.invalidate();  
 }  
 public void homePress(View view){  
 Intent intent = new Intent(this, MainActivity.class);  
 startActivity(intent);  
 }  
}

GameLogic.java

package com.example.tictactoe;  
import android.view.View;  
import android.widget.Button;  
import android.widget.TextView;  
class GameLogic {  
 private final int[][] gameBoard;  
 //first element = row, seconded element = col, third element = line type  
 private int[] winType = {-1, -1, -1};  
 private int player = 1;  
 String[] playerNames = {"Игрок 1", "Игрок 2"};  
 private Button playAgainBTN;  
 private Button homeBTN;  
 private TextView playerTurn;  
 GameLogic(){  
 //sets up game board  
 gameBoard = new int[3][3];  
 for (int r = 0; r < 3; r++){  
 for (int c = 0; c < 3; c++){  
 gameBoard[r][c] = 0;  
 }  
 }  
 }  
 public boolean updateGameBoard(int row, int col){  
 //An X will be a 1  
 //An O will be a 2  
 //Player will be a 1 or 2. if 1 --> X. if 2 --> O  
 if (gameBoard[row-1][col-1] == 0){  
 gameBoard[row-1][col-1] = player;  
 //happens in the opposite order because the turn is flipped after updateGameBoard  
 if (player == 1){  
 playerTurn.setText((playerNames[1] + " вводит"));  
 }  
 else{  
 playerTurn.setText((playerNames[0] + " вводит"));  
 }  
 return true;  
 }  
 else {  
 return false;  
 }  
 }  
 public boolean winnerCheck(){  
 boolean isWinner = false;  
 //Horizontal check  
 for (int r = 0; r<3; r++){  
 //player 1 win check  
 if (gameBoard[r][0] == gameBoard[r][1] && gameBoard[r][0] == gameBoard[r][2] &&  
 gameBoard[r][0] != 0){  
 winType = new int[]{r, 0, 1};  
 isWinner = true;  
 break;  
 }  
 }  
 //Vertical check  
 for (int c=0; c<3; c++){  
 //player 1 win check  
 if (gameBoard[0][c] == gameBoard[1][c] && gameBoard[2][c] == gameBoard[0][c] &&  
 gameBoard[0][c] != 0){  
 winType = new int[]{0, c, 2};  
 isWinner = true;  
 break;  
 }  
 }  
 //negative diagonal check  
 if (gameBoard[0][0] == gameBoard[1][1] && gameBoard[2][2] == gameBoard[0][0] &&  
 gameBoard[0][0] != 0){  
 winType = new int[]{0, 2, 3};  
 isWinner = true;  
 }  
 //positive diagonal check  
 if (gameBoard[2][0] == gameBoard[1][1] && gameBoard[0][2] == gameBoard[2][0] &&  
 gameBoard[2][0] != 0){  
 winType = new int[]{2, 2, 4};  
 isWinner = true;  
 }  
 int boardFilled = 0;  
 for (int row=0; row<3; row++){  
 for (int col=0; col<3; col++){  
 if(gameBoard[row][col] != 0){  
 boardFilled += 1;  
 }  
 }  
 }  
 if (isWinner){  
 playAgainBTN.setVisibility(View.VISIBLE);  
 homeBTN.setVisibility(View.VISIBLE);  
 playerTurn.setText((playerNames[player-1] + " Победил(а)!!!"));  
 return true;  
 }  
 else if (boardFilled == 9){  
 playAgainBTN.setVisibility(View.VISIBLE);  
 homeBTN.setVisibility(View.VISIBLE);  
 playerTurn.setText("Ничья!!!");  
 return true;  
 }  
 return false;  
 }  
 public void setPlayer(int player) {  
 this.player = player;  
 }  
 public int getPlayer() {  
 return player;  
 }  
 public int[][] getGameBoard() {  
 return gameBoard;  
 }  
 public void setPlayAgainBTN(Button playAgainBTN) {  
 this.playAgainBTN = playAgainBTN;  
 }  
 public void setHomeBTN(Button homeBTN) {  
 this.homeBTN = homeBTN;  
 }  
 public void setPlayerTurn(TextView playerTurn) {  
 this.playerTurn = playerTurn;  
 }  
 public int[] getWinType() {  
 return winType;  
 }  
 public void setPlayerNames(String[] playerNames) {  
 this.playerNames = playerNames;  
 }  
 public void resetGame(){  
 for (int r = 0; r < 3; r++){  
 for (int c = 0; c < 3; c++){  
 gameBoard[r][c] = 0;  
 }  
 }  
 winType[0] = -1;  
 winType[1] = -1;  
 winType[2] = -1;  
 player = 1;  
 playAgainBTN.setVisibility(View.GONE);  
 homeBTN.setVisibility(View.GONE);  
 playerTurn.setText((playerNames[0] + " вводит"));  
 }  
}

Player.java

package com.example.tictactoe;  
import androidx.appcompat.app.AppCompatActivity;  
import android.content.Intent;  
import android.os.Bundle;  
import android.view.View;  
import android.widget.EditText;  
public class Player extends AppCompatActivity {  
 EditText player1;  
 EditText player2;  
 @Override  
 protected void onCreate(Bundle savedInstanceState) {  
 super.onCreate(savedInstanceState);  
 setContentView(R.layout.player);  
 player1 = findViewById(R.id.player1Name);  
 player2 = findViewById(R.id.player2Name);  
 }  
 public void submitButtonClick(View view){  
 String player1Name = player1.getText().toString();  
 String player2Name = player2.getText().toString();  
 Intent intent = new Intent(this, Game.class);  
 intent.putExtra("PLAYER\_NAMES", new String[] {player1Name, player2Name});  
 startActivity(intent);  
 }  
}

AndroidManifest.xml

<activity android:name=".Game">  
 <intent-filter>  
 <action android:name="android.intent.action.MAIN" />  
 <category android:name="android.intent.category.LAUNCHER" />  
 </intent-filter>  
 </activity>  
 <activity android:name=".Player" />  
 </application>  
</manifest>

Рецензия

Протестировав данное приложение, можно сказать, что оно простое, понятное в функциях, интерфейс простой без анимации. С помощью кнопок можно управлять переходом между окнами.

Из недостатков, невозможно полностью выйти из приложения после нажатия кнопки “Выйти”, что весьма неудобно.

В итоге можно сказать, что для данного приложения нужны правки и доработки, если это все изменить. То оно приобретет интерес к пользователям.