

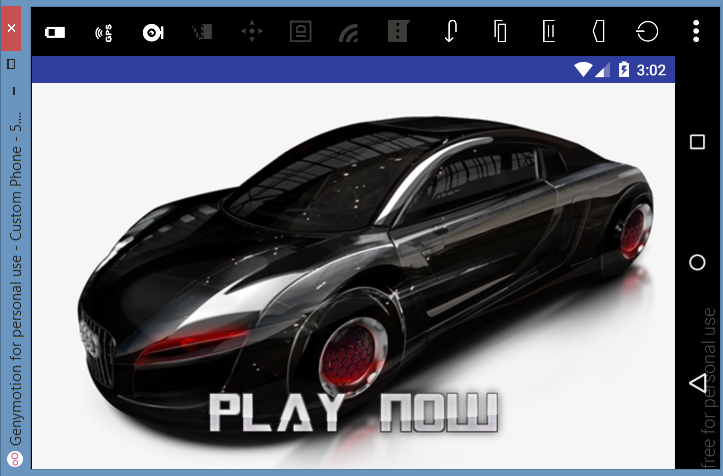
Lovely Professional University

Department of Computer Science Engineering

**Android Game Development Project**

on

“**Car Game**”



**Submitted to**

Department of Computer Science Engineering

Lovely Professional University

01-12-2017

**by**

Name - Chetan Chauhan

Reg. No- 11408101

Course - Btech( CSE)

**Table of Contents**

|  |  |  |  |
| --- | --- | --- | --- |
| 1 | Acknowledgment................................................................................................................................ | | 3 |
| 2 | Abstract................................................................................................................................................. | | 3 |
| 3 | Objectives.............................................................................................................................................. | | 3 |
| 4 | Programming Environment................................................................................................................... | | 3 |
| 5 | Functionality.......................................................................................................................................... | | 3 |
|  | 5.1. Player Car......................................................................................................................... | | 3 |
|  | 5.2. | Enemy Cars............................................................................................................................ | 4 |
| 6 | Methodology......................................................................................................................................... | | 4 |
|  | 6.1. Canvas.............................................................................................................................. | | 5 |
|  | 6.2. | Paint....................................................................................................................................... | 5 |
|  | 6.3. | Bitmap................................................................................................................................... | 5 |
|  | 6.4. | SurfaceView......................................................................................................................... | 5 |
|  |  |  |  |
| 7 | Project Management........................................................................................................................... | | 5 |
| 8 | Screen-Shots....................................................................................................................................... | | 6 |
| 9 | Future Enhancements.......................................................................................................................... | | 8 |
| 10 Code................................................................................................................................................. | | | 8 |
| 11 References........................................................................................................................ | | | 20 |

1. **Acknowledgment**

I would like to thank Mr. Roshan Srivastav Sir who replied to our endless queries without any complain. We offer gratitude to our teachers for guiding us throughout the project.

1. **Abstract**

This project aims to bring the fun and simplicity of Car Game with some new features. This is based on the car moving towards a direction and some other enemy cars are trying to stop that car. Every time when the score will increase the difficulty level will increase and the cars will keep on changing until the game is over.

This project explores a new dimension in the traditional Car game to make it more interesting and challenging. The simplicity of this game makes it an ideal candidate for a minor project as we can focus on advanced topics like Canvas, Actions, Threads, Bitmap, Paint, SurfaceView.

1. **Objectives**

This game aims to change the way people think of traditional car game. It will offer the experience of new type of car game to the player retaining the simplicity of traditional car game.

The major objectives of this project are:

* Create a car game that will have all the functionality of traditional car games.
* It should be able to give the score after the game is over .
* The moving of enemy car left ,right also make game more challenging and interesting. The movement and action of the car will be controlled by the player whose aim will keep saving the car from the enemy cars coming in way of player car.
* There will be many enemy car will be coming in the way of player car. The player has to tap on the screen to move it left or right. By default the car will be moving to the right the when the player will tap on the screen then it will move to left then again to right, this will keep on going.
* The enemy car which are coming in the way of player car will not overlap with each other.
* The score will increase when player car completed the round till the end of the screen.

**4. Programming Environment**

* **Android Studio**

I have used Android Studio IDE for the development of this project.

* **GenyMotion**

I have used GenyMotion Emulator to run the game application in Android Studio IDE for the development of this project.

**5. Functionality**

The complete car game application is divided into two major components:

* Player Car
* Enemy Cars

**5.1 Player Car**

The player car have the following functionality.

* The Player Car will move in a single direction. By default the car will move in the right direction.
* The Player have to tap on the screen to move it to right. The player have to continuously tap on the screen to move it to the right.
* If the Player Car collides with the boundary of the game then the game will be over and the score will be printed on the screen.
* If the Player car collides with the enemy-car then also the game will over.
* The main purpose of player-car is to move without colliding with the boundary and the enemy cars.
* After completing the one round the car will automatically come to the initial position and then again start moving in the same direction.

**5.2 Enemy Car**

The player car have the following functionality:

* The Enemy cars will be coming towards the player car in the opposite direction of the player car.
* There will be always two enemy-cars on the players screen. And enemy cars can come at any random position of the screen
* The one of the enemy car be keep on moving in zig-zag motion which will increase the difficulty of the game. And the other enemy car will be moving in a straight path.
* The Enemy cars will never overlap with each other because, I have given the limits on the areas of the cars.
* If the Enemy car will collide with the player car then the game will over.

**6. Methodology**

This game application includes the concepts like Canvas, Actions, Threads, Bitmap, Paint, SurfaceView, and some basic logic to move the objects and some programming.

**6.1 Canvas**

You can meet the requirements of an app that needs specialized drawing and/or control of the graphics animation by drawing to a canvas, which is represented by the Canvas class. A canvas serves as a pretense, or interface, to the actual surface upon which your graphics are drawn—you can perform your *draw* operations to the canvas. Via the canvas, your app draws to the underlying [Bitmap](https://developer.android.com/reference/android/graphics/Bitmap.html) object, which is placed into the window.

If you're drawing within the [onDraw(Canvas)](https://developer.android.com/reference/android/view/View.html" \l "onDraw(android.graphics.Canvas)) callback, the canvas is already provided and you only need to place your drawing calls upon it. If you're using a [SurfaceView](https://developer.android.com/reference/android/view/SurfaceView.html) object, you can acquire a canvas from [lockCanvas()](https://developer.android.com/reference/android/view/SurfaceHolder.html" \l "lockCanvas()). Both of these scenarios are discussed in the following sections.

If you need to create a new [Canvas](https://developer.android.com/reference/android/graphics/Canvas.html) object, then you must define the underlying [Bitmap](https://developer.android.com/reference/android/graphics/Bitmap.html) object that is required to place the drawing into a window. The following code example shows how to set up a new canvas from a bitmap:

Bitmap b = Bitmap.createBitmap(100, 100, Bitmap.Config.ARGB\_8888);  
Canvas c = new Canvas(b);

It's possible to use the bitmap in a different canvas by using one of the [drawBitmap()](https://developer.android.com/reference/android/graphics/Canvas.html" \l "drawBitmap(android.graphics.Bitmap, android.graphics.Matrix, android.graphics.Paint)) methods. However, we recommend that you use a canvas provided by the [onDraw(Canvas)](https://developer.android.com/reference/android/view/View.html" \l "onDraw(android.graphics.Canvas)) callback or the [lockCanvas()](https://developer.android.com/reference/android/view/SurfaceHolder.html" \l "lockCanvas()) method. For more information, see [Drawing on a View](https://developer.android.com/guide/topics/graphics/2d-graphics.html#on-view) and [Drawing on a SurfaceView](https://developer.android.com/guide/topics/graphics/2d-graphics.html#on-surfaceview).

The [Canvas](https://developer.android.com/reference/android/graphics/Canvas.html) class has its own set of drawing methods, including [drawBitmap()](https://developer.android.com/reference/android/graphics/Canvas.html" \l "drawBitmap(android.graphics.Bitmap, android.graphics.Matrix, android.graphics.Paint)), [drawRect()](https://developer.android.com/reference/android/graphics/Canvas.html" \l "drawRect(android.graphics.Rect, android.graphics.Paint)), [drawText()](https://developer.android.com/reference/android/graphics/Canvas.html" \l "drawText(char[], int, int, float, float, android.graphics.Paint)), and many more. Other classes that you might use also have draw() methods. For example, you probably have some [Drawable](https://developer.android.com/reference/android/graphics/drawable/Drawable.html) objects that you want to put on the canvas. The Drawable class has its own [draw(Canvas)](https://developer.android.com/reference/android/graphics/drawable/Drawable.html#draw(android.graphics.Canvas)) method that takes your canvas as an argument.

**6.2 Paint**

The Paint class holds the style and color information about how to draw geometries, text and bitmaps.

Here is the Syntax of the paint class in java.

Paint p=**new** Paint(); // creating paint class.

p.setColor(Color.***RED***); // setting the color of the text.

p.setTextSize(100f); //setting the size of the text.

canvas.drawText(**"Text"**,520,400,p); //drawing the paint object on the canvas .

**6.3 Bitmap**

The Bitmap class represents a bitmap image. You create bitmaps via the BitmapFactory class.

Using a BitmapFactory, you can create bitmaps in three common ways: from a resource, a file, or an InputStream. To create a bitmap from a resource, you use the BitmapFactory method decodeResource():

Bitmap **bp**= BitmapFactory.*decodeResource*(ct.getResources(),R.drawable.car5); /\* creating bimap object and storing the image in the bitmap \*/

canvas.drawBitmap(**cp**.getBp(),**cp**.getX(),**cp**.getY(),**null**); // drawing bitmap on the canvas.

**6.4 SurfaceView**

SurfaceViews contain a nice rendering mechanism that allows threads to update the surface's content without using a handler (good for animation).Surfaceviews cannot  be transparent, they can only appear behind other elements in the view hierarchy.I've found that they are much faster for animation than rendering onto a View.

SurfaceHolder is contained in the SurfaceView Class.

Here is the Syntax:

SurfaceHolder **sh**;

sh.addCallback(**new** SurfaceHolder.Callback() {  
 @Override  
 **public void** surfaceCreated(SurfaceHolder holder) {  
  
 **mt**.isrunning(**true**); // mt is object of thread class.  
 **mt**.start();  
  
 }

@Override  
**public void** surfaceChanged(SurfaceHolder holder, **int** format, **int** width, **int** height) {  
  
}  
  
@Override  
**public void** surfaceDestroyed(SurfaceHolder holder) {  
  
}

});

**7. Project Management**

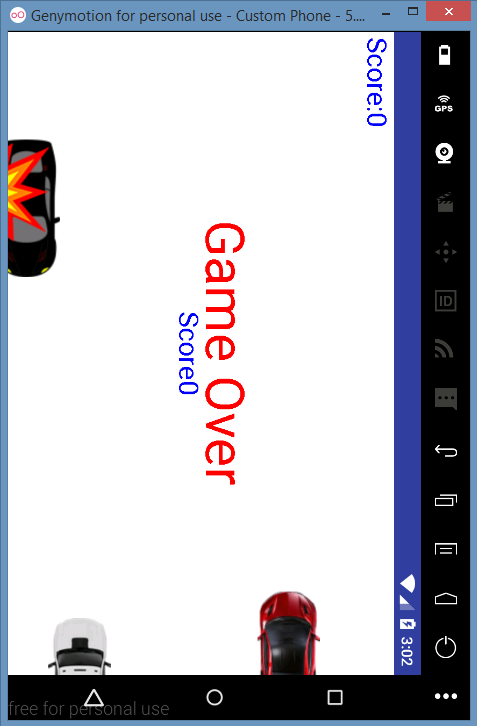
I have managed the versions of the car game project in the GitHub.

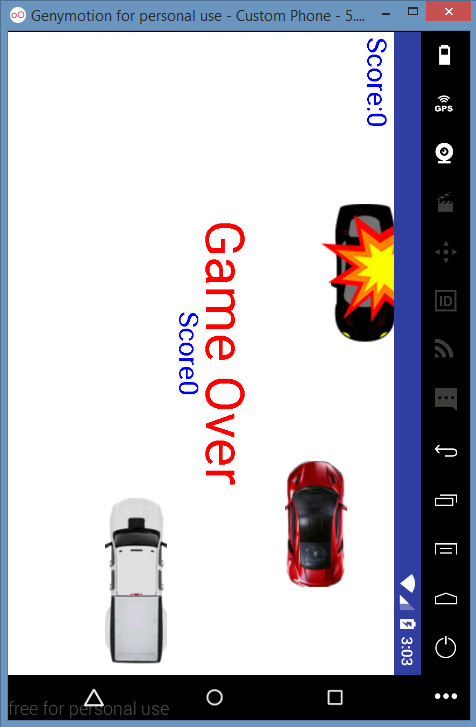
Here is the link of my Project on Github:

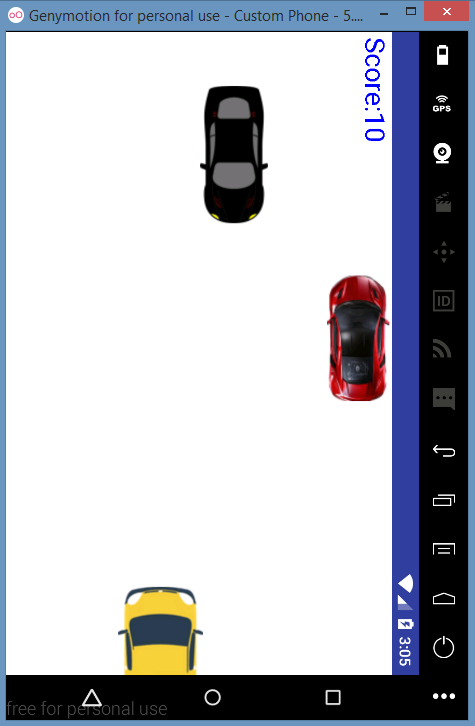
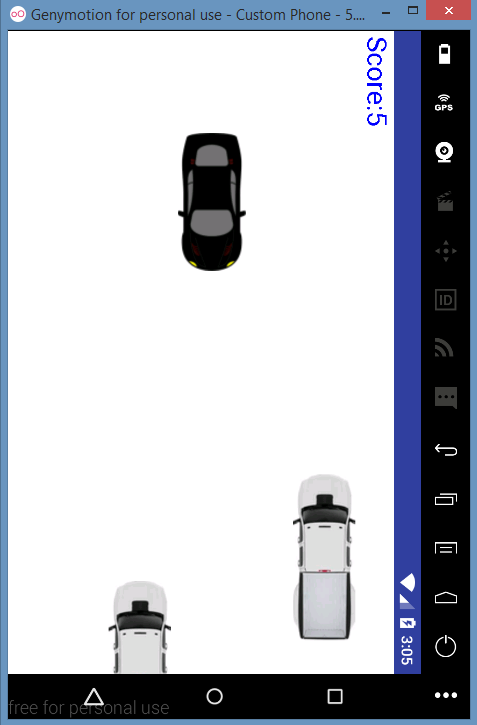
**https://github.com/chetanchauhan4321/CarGame**

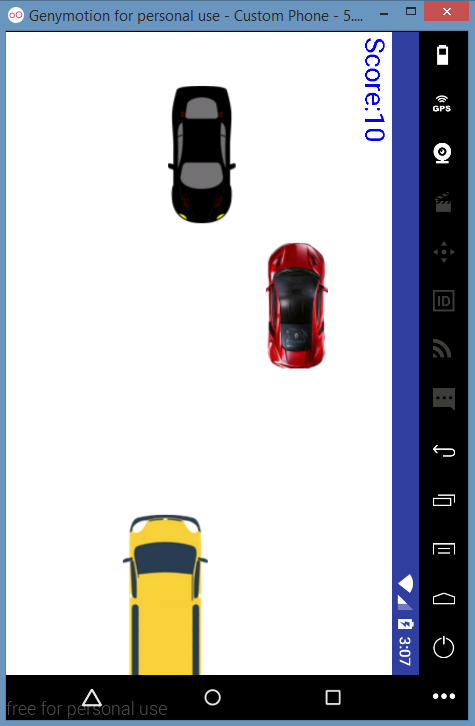
There I have maintain the versions the game some if I want to have the functionality which was before then I can use Github for that.

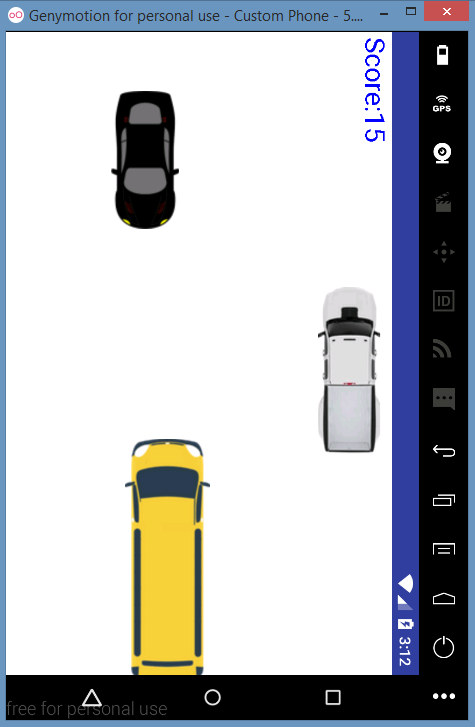
**8. Screen-Shots**



****





****

**9. Future Enhancements**

There will be some future Enhancement which is need to be made:

* Sounds: In the update of the game I will include the sounds in the game.
* I will use different level of difficulties.
* I will use some other objects in the game in future updates.

**10. Code**

* 1. **activity\_main.xml**

*<?***xml version="1.0" encoding="utf-8"***?>*<**RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"  
 xmlns:app="http://schemas.android.com/apk/res-auto"  
 xmlns:tools="http://schemas.android.com/tools"  
 android:id="@+id/activity\_main"  
 android:layout\_width="match\_parent"  
 android:layout\_height="match\_parent"  
 android:paddingBottom="@dimen/activity\_vertical\_margin"  
 android:paddingLeft="@dimen/activity\_horizontal\_margin"  
 android:paddingRight="@dimen/activity\_horizontal\_margin"  
 android:paddingTop="@dimen/activity\_vertical\_margin"  
 tools:context="com.example.user.cargame.MainActivity"  
 android:background="@drawable/car"**>  
  
 <**ImageButton  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:id="@+id/but1"  
 android:background="@drawable/playnow"  
 android:layout\_centerHorizontal="true"  
 android:layout\_alignParentBottom="true"  
 android:onClick="dothis"**/>  
</**RelativeLayout**>

* 1. **MainActivity.java**

**package** com.example.user.cargame;  
  
**import** android.content.pm.ActivityInfo;  
**import** android.support.v7.app.AppCompatActivity;  
**import** android.os.Bundle;  
**import** android.util.DisplayMetrics;  
**import** android.view.View;  
**import** android.view.Window;  
  
**public class** MainActivity **extends** AppCompatActivity {  
 GameView **gv**;  
 @Override  
 **protected void** onCreate(Bundle savedInstanceState) {  
 **super**.onCreate(savedInstanceState);  
 supportRequestWindowFeature(Window.***FEATURE\_NO\_TITLE***);  
 setContentView(R.layout.***activity\_main***);  
 setRequestedOrientation(ActivityInfo.***SCREEN\_ORIENTATION\_LANDSCAPE***);  
 DisplayMetrics dm=getResources().getDisplayMetrics();  
 **gv**=**new** GameView(**this**,dm.**widthPixels**,dm.**heightPixels**);  
}  
 **public void** dothis(View v){ setContentView(**gv**);}  
}

* 1. **GameView.java**

**package** com.example.user.cargame;  
  
**import** android.annotation.SuppressLint;  
**import** android.content.ContentValues;  
**import** android.content.Context;  
**import** android.content.SharedPreferences;  
**import** android.graphics.Canvas;  
**import** android.graphics.Color;  
**import** android.graphics.Paint;  
**import** android.graphics.Rect;  
**import** android.view.MotionEvent;  
**import** android.view.SurfaceHolder;  
**import** android.view.SurfaceView;  
**import** android.widget.Button;  
  
**import static** android.content.Context.***MODE\_WORLD\_WRITEABLE***;  
  
*/\*\*  
 \* Created by user on 23-11-2017.  
 \*/***public class** GameView **extends** SurfaceView{  
 SurfaceHolder **sh**;  
 **static** SharedPreferences *pref*;  
 **static** SharedPreferences.Editor *editor*;  
 CPlayer **cp**;  
 MyThread **mt**;  
 Boom **boom**;  
 Rect **r**,**r1**,**r2**;  
 EnemyCar **ec1**;  
 EnemyCar2 **ec2**;  
 EnemyTruck **et**;  
 EnemyTruck2 **et2**;  
 **int tx**,**ty**;  
 **int btmy**,**topy**;  
 Button **b1**;  
 Context **cc**;  
 @SuppressLint(**"WrongCall"**)  
 **public** GameView(Context ct,**int** x,**int** y)  
 {  
 **super**(ct);  
 **cc**=ct;  
 **tx**=x;  
 **ty**=y;  
 **cp**=**new** CPlayer(ct,x,y);  
 **boom** =**new** Boom(ct);  
 **sh**=getHolder();  
 **btmy**=y;  
 **topy**=0;  
 **ec1**=**new** EnemyCar(ct,x,y);  
 **ec2**=**new** EnemyCar2(ct,x,y);  
 **et**=**new** EnemyTruck(ct,x,y);  
 **et2**=**new** EnemyTruck2(ct,x,y);  
 **mt** = **new** MyThread(**this**);  
 *pref*= ct.getSharedPreferences(**"MyPref"**, 0); *// 0 - for private mode  
 editor* = *pref*.edit();  
  
  
 **sh**.addCallback(**new** SurfaceHolder.Callback() {  
 @Override  
 **public void** surfaceCreated(SurfaceHolder holder) {  
  
 **mt**.isrunning(**true**);  
 **mt**.start();  
  
 }  
  
 @Override  
 **public void** surfaceChanged(SurfaceHolder holder, **int** format, **int** width, **int** height) {  
  
 }  
  
 @Override  
 **public void** surfaceDestroyed(SurfaceHolder holder) {  
  
 }  
 });  
 }  
  
 @Override  
 **protected void** onDraw(Canvas canvas){  
 **int** score=**cp**.getScore();  
 *editor*.putInt(**"k1"**,score);  
 Paint ps=**new** Paint();  
 ps.setColor(Color.***BLUE***);  
 ps.setTextSize(50f);  
  
  
 **int** btm=**cp**.getY();  
 canvas.drawColor(Color.***WHITE***);  
  
 **r**=**new** Rect(**cp**.getX(),**cp**.getY(),**cp**.getX()+**cp**.getBp().getWidth(),**cp**.getY()+**cp**.getBp().getHeight());  
 canvas.drawText(**"Score:"**+score,10,50,ps);  
 canvas.drawBitmap(**cp**.getBp(),**cp**.getX(),**cp**.getY(),**null**);  
  
 *// canvas.drawBitmap(ec1.getBitmap(),ec1.getX(),ec1.getY(),null);  
 // canvas.drawBitmap(ec2.getBitmap(), ec2.getX(), ec2.getY(), null);* **if**(score<5) {  
 canvas.drawBitmap(**ec1**.getBitmap(),**ec1**.getX(),**ec1**.getY(),**null**);  
 canvas.drawBitmap(**et**.getBitmap(), **et**.getX(), **et**.getY(), **null**);  
  
 }  
 **else if**(score<10){  
 canvas.drawBitmap(**ec2**.getBitmap(), **ec2**.getX(), **ec2**.getY(), **null**);  
 canvas.drawBitmap(**et**.getBitmap(),**et**.getX(),**et**.getY(),**null**);  
 }  
 **else if**(score<15)  
 {  
 canvas.drawBitmap(**ec1**.getBitmap(), **ec1**.getX(), **ec1**.getY(), **null**);  
 canvas.drawBitmap(**et2**.getBitmap(), **et2**.getX(), **et2**.getY(), **null**);  
  
  
 }  
 **else** {  
 canvas.drawBitmap(**ec2**.getBitmap(), **ec2**.getX(), **ec2**.getY(), **null**);  
 canvas.drawBitmap(**et2**.getBitmap(), **et2**.getX(), **et2**.getY(), **null**);  
 }  
  
 *// r2 = new Rect(ec2.getX(), ec2.getY(), ec2.getX() + ec2.getBitmap().getWidth(), ec2.getY() + ec2.getBitmap().getHeight());* **if**(score<5) {  
 **r1**=**new** Rect(**ec1**.getX(),**ec1**.getY(),**ec1**.getX()+**ec1**.getBitmap().getWidth(),**ec1**.getY()+**ec1**.getBitmap().getHeight());  
  
 **r2** = **new** Rect(**et**.getX(), **et**.getY(), **et**.getX() + **et**.getBitmap().getWidth(), **et**.getY() + **et**.getBitmap().getHeight());  
 }  
 **else if**(score<10)  
 {  
 **r1**=**new** Rect(**ec2**.getX(),**ec2**.getY(),**ec2**.getX()+**ec2**.getBitmap().getWidth(),**ec2**.getY()+**ec2**.getBitmap().getHeight());  
  
 **r2** = **new** Rect(**et**.getX(), **et**.getY(), **et**.getX() + **et**.getBitmap().getWidth(), **et**.getY() + **et**.getBitmap().getHeight());  
 }  
 **else if**(score<15)  
 {  
 **r1**=**new** Rect(**ec1**.getX(),**ec1**.getY(),**ec1**.getX()+**ec1**.getBitmap().getWidth(),**ec1**.getY()+**ec1**.getBitmap().getHeight());  
  
 **r2** = **new** Rect(**et2**.getX(), **et2**.getY(), **et2**.getX() + **et2**.getBitmap().getWidth(), **et2**.getY() + **et2**.getBitmap().getHeight());  
  
 }  
 **else** {  
 **r1**=**new** Rect(**ec2**.getX(),**ec2**.getY(),**ec2**.getX()+**ec2**.getBitmap().getWidth(),**ec2**.getY()+**ec2**.getBitmap().getHeight());  
  
 **r2** = **new** Rect(**et2**.getX(), **et2**.getY(), **et2**.getX() + **et2**.getBitmap().getWidth(), **et2**.getY() + **et2**.getBitmap().getHeight());  
 }  
  
 *// if(Rect.intersects(r,r1))* **if**(Rect.*intersects*(**r**,**r1**)|| Rect.*intersects*(**r**,**r2**))  
 {  
 **if**(score>*pref*.getInt(**"k1"**,0)){  
 *editor*.putInt(**"k1"**,score);  
 }  
 **boom**.setX(**cp**.getX()+**cp**.getBp().getWidth()-150);  
 **boom**.setY(**cp**.getY()-45);  
  
  
 canvas.drawBitmap(**boom**.getBitmap(),**boom**.getX(),**boom**.getY(),**null**);  
 **mt**.isrunning(**false**);  
 Paint p=**new** Paint();  
 Paint p1=**new** Paint();  
 Paint p2=**new** Paint();  
 p.setColor(Color.***RED***);  
 p1.setColor(Color.***BLUE***);  
 p2.setColor(Color.***BLUE***);  
 p.setTextSize(100f);  
 p1.setTextSize(50f);  
 p2.setTextSize(50f);  
  
 canvas.drawText(**"Game Over"**,350,350,p);  
 canvas.drawText(**"Score:"**+score,520,400,p1);  
 *// canvas.drawText("Max. Score:"+pref.getInt("k1",0),600,500,p2);* }  
  
 **else if**(btm>=**btmy**-150)  
 {  
 **if**(score>*pref*.getInt(**"k1"**,0)){  
 *editor*.putInt(**"k1"**,score);  
 }  
  
 **boom**.setX(**cp**.getX());  
 **boom**.setY(**cp**.getY()+20);  
 canvas.drawBitmap(**boom**.getBitmap(),**boom**.getX(),**boom**.getY(),**null**);  
 **mt**.isrunning(**false**);  
  
 Paint p=**new** Paint();  
 Paint p1=**new** Paint();  
 p.setColor(Color.***RED***);  
 p1.setColor(Color.***BLUE***);  
 p.setTextSize(100f);  
 p1.setTextSize(50f);  
  
 canvas.drawText(**"Game Over"**,350,350,p);  
 canvas.drawText(**"Score"**+score,520,400,p1);  
  
  
  
 }  
 **else if**(btm<=**topy**)  
 {  
 **if**(score>*pref*.getInt(**"k1"**,0)){  
 *editor*.putInt(**"k1"**,score);  
 }  
 **boom**.setX(**cp**.getX()+20);  
 **boom**.setY(**cp**.getY()-80);  
 canvas.drawBitmap(**boom**.getBitmap(),**boom**.getX(),**boom**.getY(),**null**);  
 **mt**.isrunning(**false**);  
  
 Paint p=**new** Paint();  
 Paint p1=**new** Paint();  
 p.setColor(Color.***RED***);  
 p1.setColor(Color.***BLUE***);  
 p.setTextSize(100f);  
 p1.setTextSize(50f);  
  
 canvas.drawText(**"Game Over"**,350,350,p);  
 canvas.drawText(**"Score"**+score,520,400,p1);  
  
  
  
  
  
 }  
  
 **ec1**.change();  
  
 **ec2**.change();  
 **et**.change();  
 **et2**.change();  
  
  
 **cp**.change();  
  
 }  
 **public boolean** onTouchEvent(MotionEvent motionEvent){  
  
  
 **switch**(motionEvent.getAction() & motionEvent.***ACTION\_MASK***){  
 **case** MotionEvent.***ACTION\_UP***:  
 **cp**.moveLeft();  
 **break**;  
 **case** MotionEvent.***ACTION\_DOWN***:  
 **cp**.moveRight();  
 *//cp.setY(cp.getY()-40);* **break**;  
  
 }  
 **return true**;  
 }  
}

* 1. **MyThread.java**

**package** com.example.user.cargame;  
  
**import** android.annotation.SuppressLint;  
**import** android.graphics.Canvas;  
  
*/\*\*  
 \* Created by user on 23-11-2017.  
 \*/***public class** MyThread **extends** Thread{  
 **boolean flag**=**false**;  
 GameView **gameView**;  
  
 **public** MyThread(GameView gv)  
 {  
 **gameView**=gv;  
  
 }  
  
 **void** isrunning(**boolean** flag)  
 {  
 **this**.**flag**=flag;  
 }  
 @SuppressLint(**"WrongCall"**)  
 @Override  
 **public void** run() {  
 **while** (**flag**){  
 Canvas c=**null**;  
 **try** {  
 Thread.*sleep*(200);  
 **synchronized** (**gameView**.getHolder()){  
 c = **gameView**.getHolder().lockCanvas();  
 **gameView**.onDraw(c);  
 }  
 } **catch** (Exception e) {  
 }  
 **finally** {  
 **gameView**.getHolder().unlockCanvasAndPost(c);  
 }  
 }  
 }  
}

* 1. **CPlayer.java**

**package** com.example.user.cargame;  
  
**import** android.content.Context;  
**import** android.graphics.Bitmap;  
**import** android.graphics.BitmapFactory;  
  
*/\*\*  
 \* Created by user on 23-11-2017.  
 \*/***public class** CPlayer {  
  
 Bitmap **bp**;  
 **int g**;  
 **int x**,**x1**,**y**,**y1**,**score**=0;  
 **boolean b**;  
 CPlayer(Context ct,**int** screenx,**int** screeny)  
 {  
 **bp**= BitmapFactory.*decodeResource*(ct.getResources(),R.drawable.***car5***);  
 **x**=0;  
 **x1**=screenx-**bp**.getWidth(); *// -bp.getWidth* **y**=screeny/2-**bp**.getHeight()/2;  
 **y1**=**y**;  
 *// btmy=screeny;* **b**=**false**;  
 *// gravity=10;* **g**=15;  
  
 }  
 **void** moveLeft(){**b**=**false**;}  
 **void** moveRight(){  
 **b**=**true**;  
 }  
  
 **void** change(){  
 **y**=**y**+**g**;  
 **if**(**b**){  
  
 **x**=**x**+10;  
 **y**=**y**-30;  
 }  
  
 **else**{  
 **x**=**x**+10;  
 }  
  
 **if**(**x**>=**x1**){  
 **x**=0;  
 *//y=y1;* **score**=**score**+5;  
 }  
  
  
 }  
  
 **public** Bitmap getBp() {  
 **return bp**;  
 }  
  
 **public void** setBp(Bitmap bp) {  
 **this**.**bp** = bp;  
 }  
 **public int** getScore(){**return score**;}  
  
 **public int** getX() {  
 **return x**;  
 }  
  
 **public void** setX(**int** x) {  
 **this**.**x** = x;  
 }  
  
 **public int** getY() {  
 **return y**;  
 }  
  
 **public void** setY(**int** y) {  
 **this**.**y** = y;  
 }  
}

* 1. **EnemyCar.java**

**package** com.example.user.cargame;  
  
**import** android.content.Context;  
**import** android.graphics.Bitmap;  
**import** android.graphics.BitmapFactory;  
  
**import** java.util.Random;  
  
*/\*\*  
 \* Created by user on 23-11-2017.  
 \*/***public class** EnemyCar {  
  
 **private** Bitmap **bitmap**;  
 **private int x**,**x1**,**sy**,**midx**;  
 **private int y**,**z**,**y1**;  
  
 **public** EnemyCar(Context context, **int** screenx, **int** screeny){  
 **bitmap**= BitmapFactory.*decodeResource*(context.getResources(),R.drawable.***car1***);  
 **x**=screenx+**bitmap**.getWidth();  
  
 **x1**=**x**;  
 **z**=screeny;  
 **midx**=**z**/2;  
 Random r=**new** Random();  
 *// y=1+r.nextInt(screeny-bitmap.getWidth()-screeny/2);  
 //y=screeny/2-r.nextInt(screeny-bitmap.getWidth());* **y**=r.nextInt(screeny/2-**bitmap**.getWidth());  
 **sy**=screeny;  
 *//y1=y;* }  
  
 **public void** change()  
 {  
 Random r=**new** Random();  
 **int** k=5+r.nextInt(30);  
 **x**=**x**-k;  
 */\* if(x<=midx+100)  
 {  
 y=y-2;  
 }  
 else if(y<=0)  
 {  
 y=y+2;  
 }  
 else y=y+2;  
 \*/* **if**(**x**<=0)  
 {  
 **x**=**x1**;  
 Random r2=**new** Random();  
 *// y=1+r2.nextInt(z-bitmap.getWidth());* **y**=r2.nextInt(**sy**/2-**bitmap**.getWidth());  
  
 }  
 }  
  
 **public** Bitmap getBitmap(){ **return bitmap**;}  
 **public int** getX(){**return x**;}  
 **public int** getY(){**return y**;}  
 **public void** setY(**int** y) {  
 **this**.**y** = y;  
 }  
 **public void** setX(**int** x) {  
 **this**.**x** = x;  
 }  
  
  
  
}

* 1. **EnemyCar2.java**

**package** com.example.user.cargame;  
  
**import** android.content.Context;  
**import** android.graphics.Bitmap;  
**import** android.graphics.BitmapFactory;  
  
**import** java.util.Random;  
  
*/\*\*  
 \* Created by user on 24-11-2017.  
 \*/***public class** EnemyCar2 {  
  
 **private** Bitmap **bitmap**;  
 **private int x**,**x1**,**sy**,**midx**;  
 **private int y**,**z**,**y1**;  
  
 **public** EnemyCar2(Context context, **int** screenx, **int** screeny){  
 **bitmap**= BitmapFactory.*decodeResource*(context.getResources(),R.drawable.***truck2***);  
 **x**=screenx+**bitmap**.getWidth();  
  
 **x1**=**x**;  
 **z**=screeny;  
 **midx**=**z**/2;  
 Random r=**new** Random();  
 *// y=1+r.nextInt(screeny-bitmap.getWidth()-screeny/2);  
 //y=screeny/2-r.nextInt(screeny-bitmap.getWidth());* **y**=r.nextInt(screeny/2-**bitmap**.getWidth());  
 **sy**=screeny;  
 *//y1=y;* }  
  
 **public void** change()  
 {  
 Random r=**new** Random();  
 **int** k=5+r.nextInt(30);  
 **x**=**x**-k;  
 */\* if(x<=midx+100)  
 {  
 y=y-2;  
 }  
 else if(y<=0)  
 {  
 y=y+2;  
 }  
 else y=y+2;  
 \*/* **if**(**x**<=0)  
 {  
 **x**=**x1**;  
 Random r2=**new** Random();  
 *// y=1+r2.nextInt(z-bitmap.getWidth());* **y**=r2.nextInt(**sy**/2-**bitmap**.getWidth());  
  
 }  
 }  
  
 **public** Bitmap getBitmap(){ **return bitmap**;}  
 **public int** getX(){**return x**;}  
 **public int** getY(){**return y**;}  
 **public void** setY(**int** y) {  
 **this**.**y** = y;  
 }  
 **public void** setX(**int** x) {  
 **this**.**x** = x;  
 }  
}

* 1. **EnemyTruck.java**

**package** com.example.user.cargame;  
  
**import** android.content.Context;  
**import** android.graphics.Bitmap;  
**import** android.graphics.BitmapFactory;  
  
**import** java.util.Random;  
  
*/\*\*  
 \* Created by user on 25-11-2017.  
 \*/***public class** EnemyTruck {  
 **private** Bitmap **bitmap**;  
 **private int x**,**x1**,**xmid**;  
 **private int y**,**z**,**y1**;  
 **private int pwidth**;  
 EnemyCar **enemyCar**;  
 CPlayer **cPlayer**;  
  
 **public** EnemyTruck(Context context, **int** screenx, **int** screeny){  
 **bitmap**= BitmapFactory.*decodeResource*(context.getResources(),R.drawable.***truck2***);  
 **x**=screenx+**bitmap**.getWidth();  
 **xmid**=screenx/2;  
 **x1**=**x**;  
 **z**=screeny;  
 Random r=**new** Random();  
 **y**=screeny/2+r.nextInt(screeny-getBitmap().getWidth()-270);  
 *// cPlayer=new CPlayer(context,screenx,screeny);  
 // enemyCar=new EnemyCar(context,screenx,screeny);  
 // pwidth=cPlayer.getBp().getWidth();  
 //y=enemyCar.getY()+pwidth+50;  
 // y1=y;* }  
  
 **public void** change()  
 {  
 Random r=**new** Random();  
 **int** k=5+r.nextInt(30);  
 **x** = **x** - k;  
 **if**(**x**<=**xmid**+100) {  
  
 **y** = **y** + 2;  
 }  
  
 **else if**(**y**<=**z**-180)  
 {  
 **y**=**y**-2;  
 }  
 **else y**=**y**-2;  
  
 **if**(**x**<=0)  
 {  
 **x**=**x1**;  
 Random r2=**new** Random();  
 **y**=**z**/2+r2.nextInt(**z**-**bitmap**.getWidth()-270);  
 *//y=y1;* }  
 }  
  
 **public** Bitmap getBitmap(){ **return bitmap**;}  
 **public int** getX(){**return x**;}  
 **public int** getY(){**return y**;}  
 **public void** setY(**int** y) {  
 **this**.**y** = y;  
 }  
 **public void** setX(**int** x) {  
 **this**.**x** = x;  
 }  
}

* 1. **EnemyTruck2.java**

**package** com.example.user.cargame;  
  
**import** android.content.Context;  
**import** android.graphics.Bitmap;  
**import** android.graphics.BitmapFactory;  
  
**import** java.util.Random;  
  
*/\*\*  
 \* Created by user on 26-11-2017.  
 \*/***public class** EnemyTruck2 {  
 **private** Bitmap **bitmap**;  
 **private int x**,**x1**,**xmid**;  
 **private int y**,**z**,**y1**;  
 **private int pwidth**;  
 EnemyCar **enemyCar**;  
 CPlayer **cPlayer**;  
  
 **public** EnemyTruck2(Context context, **int** screenx, **int** screeny){  
 **bitmap**= BitmapFactory.*decodeResource*(context.getResources(),R.drawable.***bus***);  
 **x**=screenx+**bitmap**.getWidth();  
 **xmid**=screenx/2;  
 **x1**=**x**;  
 **z**=screeny;  
 Random r=**new** Random();  
 **y**=screeny/2+r.nextInt(screeny-getBitmap().getWidth()-270);  
 *// cPlayer=new CPlayer(context,screenx,screeny);  
 // enemyCar=new EnemyCar(context,screenx,screeny);  
 // pwidth=cPlayer.getBp().getWidth();  
 //y=enemyCar.getY()+pwidth+50;  
 // y1=y;* }  
  
 **public void** change()  
 {  
 Random r=**new** Random();  
 **int** k=5+r.nextInt(30);  
 **x** = **x** - k;  
 **if**(**x**<=**xmid**+100) {  
  
 **y** = **y** + 2;  
 }  
  
 **else if**(**y**<=**z**-180)  
 {  
 **y**=**y**-2;  
 }  
 **else y**=**y**-2;  
  
 **if**(**x**<=0)  
 {  
 **x**=**x1**;  
 Random r2=**new** Random();  
 **y**=**z**/2+r2.nextInt(**z**-**bitmap**.getWidth()-270);  
 *//y=y1;* }  
 }  
  
 **public** Bitmap getBitmap(){ **return bitmap**;}  
 **public int** getX(){**return x**;}  
 **public int** getY(){**return y**;}  
 **public void** setY(**int** y) {  
 **this**.**y** = y;  
 }  
 **public void** setX(**int** x) {  
 **this**.**x** = x;  
 }  
  
  
}

**10.10. Boom.java**

**package** com.example.user.cargame;  
  
**import** android.content.Context;  
**import** android.graphics.Bitmap;  
**import** android.graphics.BitmapFactory;  
  
*/\*\*  
 \* Created by user on 23-11-2017.  
 \*/***public class** Boom {  
 **private** Bitmap **bitmap**;  
 **private int x**;  
 **private int y**;  
 **public** Boom(Context context)  
 {  
 **bitmap**= BitmapFactory.*decodeResource*(context.getResources(),R.drawable.***boom***);  
 **x**=-250;  
 **y**=-250;  
 }  
  
 **public void** setX(**int** x){**this**.**x**=x;}  
 **public void** setY(**int** y){**this**.**y**=y;}  
  
 **public** Bitmap getBitmap(){**return bitmap**;}  
 **public int** getX(){**return x**;}  
 **public int** getY(){  
 **return y**;  
 }  
}

1. **References:**

Fighter Plane Game by Mr. Roshan Srivastav.

****