Main\_activity.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:layout\_width="match\_parent"

android:layout\_height="match\_parent"

tools:context=".MainActivity"

android:background="@drawable/background">

<ImageButton

android:id="@+id/imageButton"

android:layout\_width="85dp"

android:layout\_height="85dp"

android:layout\_centerInParent="true"

android:background="@drawable/play2"

android:onClick="startgame" />

<TextView

android:id="@+id/textView"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:layout\_centerHorizontal="true"

android:text="PLAY"

android:textSize="34dp"

android:layout\_below="@+id/imageButton"

android:textColor="@android:color/holo\_red\_dark"/>

</RelativeLayout>

Main Activity.java:

package com.example.flappybird;

import androidx.appcompat.app.AppCompatActivity;

import android.content.Context;

import android.content.Intent;

import android.os.Bundle;

import android.util.Log;

import android.view.View;

import android.widget.ImageButton;

import java.util.ArrayList;

import java.util.List;

import java.util.Random;

public class MainActivity extends AppCompatActivity {

@Override

protected void onCreate(Bundle savedInstanceState) {

super.onCreate(savedInstanceState);

setContentView(R.layout.activity\_main);

AppConstants.initialization(this.getApplicationContext());

}

public void startgame(View view){

String tag;

String msg;

// Log.i(tag="ImageButton", msg="clicked");

Intent intent=new Intent(this, GameActivity.class);

startActivity(intent);

finish();

}

}

Background.java:

package com.example.flappybird;

public class Backgroundimage {

private int backgroundImageX, backgroundImageY,backgroundImageVelocity;

public Backgroundimage(){

backgroundImageX=0;

backgroundImageY=0;

backgroundImageVelocity=3;

}

//getter method for getting x-coordinate

public int getX(){

return backgroundImageX;

}

//getter method for getting the Y-coordinate

public int getY(){

return backgroundImageY;

}

//Setter method for setting the X-coordinate

public void setX(int backgroundImageX){

this.backgroundImageX=backgroundImageX;

}

//setter method for setting Y-coordinate

public void setY(int backgroundImageY){

this.backgroundImageY=backgroundImageY;

}

//getter method for getting the velocity

public int getVelocity(){

return backgroundImageVelocity;

}

}

BitmapBank.java:

package com.example.flappybird;

import android.content.res.Resources;

import android.graphics.Bitmap;

import android.graphics.BitmapFactory;

public class BitmapBank {

Bitmap big\_background;

Bitmap[] bird;

public BitmapBank(Resources res) {

big\_background= BitmapFactory.decodeResource(res,R.drawable.big\_background);

big\_background=scaleImage(big\_background);

bird=new Bitmap[4];

bird[0]=BitmapFactory.decodeResource(res, R.drawable.yellowbird);

bird[1]=BitmapFactory.decodeResource(res, R.drawable.yellowbird);

bird[2]=BitmapFactory.decodeResource(res, R.drawable.yellowbird);

bird[3]=BitmapFactory.decodeResource(res, R.drawable.yellowbird); }

//return bird bitmap of frame

public Bitmap getBird(int frame){

return bird[frame]; }

public int getBirdWidth(){

return bird[0].getWidth();}

public int getBirdHeight(){

return bird[0].getHeight(); }

//return background bitmap

public Bitmap getBig\_background(){

return big\_background;}

//return background width

public int getBackgroundWidth(){

return big\_background.getWidth();}

//return background height

public int getBackgroundHeight(){

return big\_background.getHeight();}

public Bitmap scaleImage(Bitmap bitmap){

float widthHeightRatio = getBackgroundWidth() / getBackgroundHeight();

//we will multiply widthHeightRatio with screenHeight to get scaled width of the bitmap.

//then call createScaleBitmap() to create a new Bitmap,scaled from an existing bitmap,when possible.

int backgroundScaledWidth=(int) widthHeightRatio \* AppConstants.SCREEN\_HEIGHT;

return Bitmap.createScaledBitmap(bitmap, backgroundScaledWidth,AppConstants.SCREEN\_HEIGHT,false);

}

}

GameThread.java:

package com.example.flappybird;

import android.graphics.Canvas;

import android.os.SystemClock;

import android.util.Log;

import android.view.SurfaceHolder;

public class GameThread extends Thread {

SurfaceHolder surfaceHolder; //surfaceholder object reference

boolean isRunning; //Flag to detect whether the thread is running or not

long startTime, loopTime; //loop start time and loop duration

long DELAY = 33; //delay in milliseconds between screen refreshes

public GameThread(SurfaceHolder surfaceHolder) {

this.surfaceHolder = surfaceHolder;

isRunning = true;}

@Override

public void run() {

//looping until the boolean is false

while (isRunning){

startTime= SystemClock.uptimeMillis();

//locking the canvas

Canvas canvas=surfaceHolder.lockCanvas(null);

if (canvas!=null){

synchronized (surfaceHolder){

AppConstants.getGameEngine().updateAndDrawBackgroundImage(canvas);

AppConstants.getGameEngine().updateAndDrawBird(canvas);

//unlocking the canvas

surfaceHolder.unlockCanvasAndPost(canvas);}

}

//loop time

loopTime=SystemClock.uptimeMillis()-startTime;

//pausing here to make sure we update right amount per second

if(loopTime<DELAY){

try{

Thread.sleep(DELAY-loopTime);

}catch(InterruptedException e){

Log.e("Interrupted","Interrupted while sleeping");} }

}

}

//return whether the thread is running

public boolean isRunning(){

return isRunning;}

//Sets the thread state,false=stopped,true=running

public void setIsRunning(boolean state){

isRunning=state;} }

AppConstants.java:

package com.example.flappybird;

import android.content.Context;

import android.util.DisplayMetrics;

import android.view.Display;

import android.view.WindowManager;

public class AppConstants {static BitmapBank bitmapBank; //Bitmap object reference

static GameEngine gameEngine; //GameEngine object reference

static int SCREEN\_WIDTH,SCREEN\_HEIGHT;

static int gravity;

static int VELOCITY\_WHEN\_JUMPED;

public static void initialization(Context context){ setScreenSize(context);

bitmapBank=new BitmapBank(context.getResources());

gameEngine=new GameEngine();

AppConstants.gravity=3;

AppConstants.VELOCITY\_WHEN\_JUMPED = -40;}

//Return BitmapBank instance

public static BitmapBank getBitmapBank(){

return bitmapBank; }

//retutn GameEngine instance

public static GameEngine getGameEngine(){

return gameEngine; }

private static void setScreenSize(Context context){

WindowManager wm=(WindowManager) context.getSystemService(Context.WINDOW\_SERVICE);

Display display=wm.getDefaultDisplay();

DisplayMetrics metrics=new DisplayMetrics();

display.getMetrics(metrics);

int width=metrics.widthPixels;

int height=metrics.heightPixels;

AppConstants.SCREEN\_WIDTH=width;

AppConstants.SCREEN\_HEIGHT=height; }}

Bird.java:

package com.example.flappybird;

public class Bird {

private int birdX,birdY,currentFrame,velocity;

public static int maxFrame;

public Bird(){

birdX=AppConstants.SCREEN\_WIDTH/2 - AppConstants.getBitmapBank().getBirdWidth()/2;

birdY=AppConstants.SCREEN\_HEIGHT/2 - AppConstants.getBitmapBank().getBirdHeight()/2;

currentFrame=0;

maxFrame=3;

velocity=0; }

//getter method for velocity

public int getVelocity(){

return velocity; }

//setter method for velocity

public void setVelocity(int velocity){

this.velocity=velocity;}

//getter method for current frame

public int getCurrentFrame(){

return currentFrame;}

//setter method for current frame

public void setCurrentFrame(int currentFrame){

this.currentFrame=currentFrame;}

//getter method for getting X-coordinate of bird

public int getX(){

return birdX;}

//getter method for getting Y-coordinate of bird

public int getY(){

return birdY;}

//setter method for setting the X-coordinate

public void setX(int birdX){

this.birdX=birdX; }

//setter method for setting the Y-coordinate

public void setY(int birdY){

this.birdY=birdY;}}

GameActivity.java:

package com.example.flappybird;

import android.app.Activity;

import android.os.Bundle;

import androidx.annotation.Nullable;

public class GameActivity extends Activity { GameView gameView;

@Override

protected void onCreate(@Nullable Bundle savedInstanceState) {

super.onCreate(savedInstanceState);

gameView=new GameView(this);

setContentView(gameView);}}

GameEngine.java:

package com.example.flappybird;

import android.graphics.Canvas;

public class GameEngine { Backgroundimage backgroundimage;

Bird bird;

static int gameState;

public GameEngine(){

backgroundimage=new Backgroundimage();

bird=new Bird();

//0=not started

//1=playing

//2=gameover

gameState=0;}

public void updateAndDrawBackgroundImage(Canvas canvas){

backgroundimage.setX(backgroundimage.getX() - backgroundimage.getVelocity());

if (backgroundimage.getX()< -AppConstants.getBitmapBank().getBackgroundWidth()){

backgroundimage.setX(0);}

canvas.drawBitmap(AppConstants.getBitmapBank().getBig\_background(),backgroundimage.getX(),backgroundimage.getY(),null);

if (backgroundimage.getX() < -(AppConstants.getBitmapBank().getBackgroundWidth() - AppConstants.SCREEN\_WIDTH)){

canvas.drawBitmap((AppConstants.getBitmapBank().getBig\_background()), backgroundimage.getX() + AppConstants.getBitmapBank().getBackgroundWidth(),backgroundimage.getY(),null);}}

public void updateAndDrawBird(Canvas canvas){ if (gameState==1) {

if (bird.getY() < (AppConstants.SCREEN\_HEIGHT - AppConstants.getBitmapBank().getBirdHeight()) || bird.getVelocity()<0) {bird.setVelocity(bird.getVelocity() + AppConstants.gravity);

bird.setY(bird.getY() + bird.getVelocity());}}

int currentFrame=bird.getCurrentFrame();

canvas.drawBitmap(AppConstants.getBitmapBank().getBird(currentFrame),bird.getX(),bird.getY(),null);currentFrame++;

//if it exceeds maxframe re-initiate to 0

if (currentFrame > bird.maxFrame) {

currentFrame = 0;}

bird.setCurrentFrame(currentFrame);}}

GameView.java:

package com.example.flappybird;

import android.annotation.SuppressLint;

import android.content.Context;

import android.graphics.Canvas;

import android.graphics.Paint;

import android.util.AttributeSet;

import android.view.MotionEvent;

import android.view.SurfaceHolder;

import android.view.SurfaceView;

import android.view.View;

import androidx.annotation.Nullable;

import java.util.ArrayList;

import java.util.List;

public class GameView extends SurfaceView implements SurfaceHolder.Callback{

GameThread gameThread;

public GameView(Context context) {

super(context);

initView(); }

@Override

public void surfaceCreated(@Nullable SurfaceHolder holder) {

if (!gameThread.isRunning()){

gameThread=new GameThread(holder);

gameThread.start(); }else {

gameThread.start();}}

@Override

public void surfaceChanged(@Nullable SurfaceHolder holder, int format, int width, int height);}

@Override

public void surfaceDestroyed(@Nullable SurfaceHolder holder) {if(gameThread.isRunning()){

gameThread.setIsRunning(false);

boolean retry=true;

while(retry){try{gameThread.join();

retry=false;}catch (InterruptedException e{}}}}

void initView(){

SurfaceHolder holder=getHolder();

holder.addCallback(this);

setFocusable(true);

gameThread=new GameThread(holder); }

@Override

public boolean onTouchEvent(MotionEvent event) {

int action = event.getAction();

//tap is detected

if (action==MotionEvent.ACTION\_DOWN){

AppConstants.getGameEngine().gameState=1;

AppConstants.getGameEngine().bird.setVelocity(AppConstants.VELOCITY\_WHEN\_JUMPED); }

return true;}}