**Technical Requirements**

The technologies or software that we used to develop this application are: (1) Android Studio. Android Studio provides the fastest tools for building apps on every type of Android device. We use Android Studio for code editing, debugging, performance tooling, a flexible build system, and an instant build/deploy system. (2) Also, we use Photoshop for creating image and Word Document for creating background.

****USE CASE DIAGRAM

SYSTEM FLOW

****

****

**JAVA CODES SNIPPET**

1. When button is clicked, new activity will appear.

private void settingButton() {  
 Button setting = (Button) findViewById(R.id.btnSetting);  
 setting.setOnClickListener(new View.OnClickListener() {  
 public void onClick(View v) {  
 Intent i = new Intent(MainActivity.this, SettingActivity.class);  
 startActivity(i);  
 finish();  
 }  
 });  
}

1. When user wants to go back (Back button) using shared preference

public void backButton() {  
 Button back = (Button) findViewById(R.id.btnBack);  
 back.setOnClickListener(new View.OnClickListener() {  
 public void onClick(View v) {  
 bgdCheck = 1;  
 Intent i = new Intent(SettingActivity.this, MainActivity.class);  
 startActivity(i);  
 finish();  
 SharedPreferences sp = getSharedPreferences("GreenyDroid.txt", MODE\_PRIVATE);  
 SharedPreferences.Editor spsave = sp.edit();  
 spsave.commit();  
 }  
 });  
}

1. Turning on and off of music background.

public static int bgdCheck = 0;  
static CheckBox music;  
  
@Override  
protected void onCreate(@Nullable Bundle savedInstanceState) {  
 super.onCreate(savedInstanceState);  
 setContentView(R.layout.activity\_setting);  
 backButton();  
  
 music = (CheckBox) findViewById(R.id.checkBox);  
 if (MainActivity.bgdMusic.isPlaying()) {  
 SettingActivity.music.setChecked(true);  
 } else {  
 SettingActivity.music.setChecked(false);  
 }  
  
 music.setOnCheckedChangeListener(new CompoundButton.OnCheckedChangeListener() {  
 public void onCheckedChanged(CompoundButton buttonView, boolean isChecked) {  
 if (isChecked) {  
 SharedPreferences.Editor editor = getSharedPreferences("GreenyDroid.txt", MODE\_PRIVATE).edit();  
 editor.putBoolean("Resume", true);  
 editor.commit();  
 music.setChecked(true);  
 MainActivity.bgdMusic.start();  
 } else {  
 SharedPreferences.Editor editor = getSharedPreferences("GreenyDroid.txt", MODE\_PRIVATE).edit();  
 editor.putBoolean("Pause", false);  
 editor.commit();  
 music.setChecked(false);  
 MainActivity.bgdMusic.pause();  
 }  
 }  
 });

1. Getting answer from the user using the radio button.

static RadioButton A, B, C, D;  
  
protected void onCreate(Bundle savedInstanceState) {  
 super.onCreate(savedInstanceState);  
 setContentView(R.layout.activity\_acupcakequestion);  
  
 A = (RadioButton)findViewById(R.id.radioButtonA);  
 A.setOnCheckedChangeListener(new CompoundButton.OnCheckedChangeListener() {  
 @Override  
 public void onCheckedChanged(CompoundButton buttonView, boolean isChecked) {  
 if(isChecked){  
 nextButton();  
 }else{  
 nextButton();  
 }  
 }  
 });  
 B = (RadioButton)findViewById(R.id.radioButtonB);  
 B.setOnCheckedChangeListener(new CompoundButton.OnCheckedChangeListener() {  
 @Override  
 public void onCheckedChanged(CompoundButton buttonView, boolean isChecked) {  
 if(isChecked){  
 nextButton();  
 }else{  
 nextButton();  
 }  
 }  
 });  
 C = (RadioButton)findViewById(R.id.radioButtonC);  
 C.setOnCheckedChangeListener(new CompoundButton.OnCheckedChangeListener() {  
 @Override  
 public void onCheckedChanged(CompoundButton buttonView, boolean isChecked) {  
 if(isChecked){  
 SharedPreferences sp = getSharedPreferences("QuestionCupcake",MODE\_PRIVATE);  
 SharedPreferences.Editor spsave = sp.edit();  
 spsave.commit();  
 nextButton();  
 }else{  
 nextButton();  
 }  
 }  
 });  
}

1. Putting delay before the fruits would drop in mini game

class Timer implements Runnable{

@Override

public void run() {

while(count != 5){

SystemClock.sleep(1000);

count++;

}

}

}

1. Displaying high score

SharedPreferences sp = getSharedPreferences("highscore.txt", MODE\_PRIVATE);

lastHighScore = sp.getInt("highscorebeat", highScore);

highScore = lastHighScore;

7. Adding score

int c = 0;

String name = StartActivity.name;

if(CCupcakeQuestionActivity.D.isChecked()){

c+=1;

}

if(BCupCakeQuestionActivity.C.isChecked()){

c+=1;

}

if(ACupcakeQuestionActivity.C.isChecked()){

c+=1;

}

8. If the user’s score is equals to three, he/she may proceed to the next activity, else if less than three, user should try again.

switch(c){

case 1:

if(StartActivity.name.isEmpty()){

}else{

TextView Name = (TextView)findViewById(R.id.congratulation);

Name.setText("Try again " + name);

TextView Score = (TextView)findViewById(R.id.score);

Score.setText("" + c);

}

retryButton();

break;

case 2:

if(StartActivity.name.isEmpty()){

}else{

TextView Name = (TextView)findViewById(R.id.congratulation);

Name.setText("Almost there " + name);

TextView Score = (TextView)findViewById(R.id.score);

Score.setText("" + c);

}

retryButton();

break;

case 3:

if(StartActivity.name.isEmpty()){

}else{

TextView Name = (TextView)findViewById(R.id.congratulation);

Name.setText("Congratulation " + name);

TextView Score = (TextView)findViewById(R.id.score);

Score.setText("" + c);

}

nextButton();

break;

default:

if(StartActivity.name.isEmpty()){

}else{

TextView Name = (TextView)findViewById(R.id.congratulation);

Name.setText("Please try again " + name);

TextView Score = (TextView)findViewById(R.id.score);

Score.setText("" + c);

}

retryButton();

break;

}