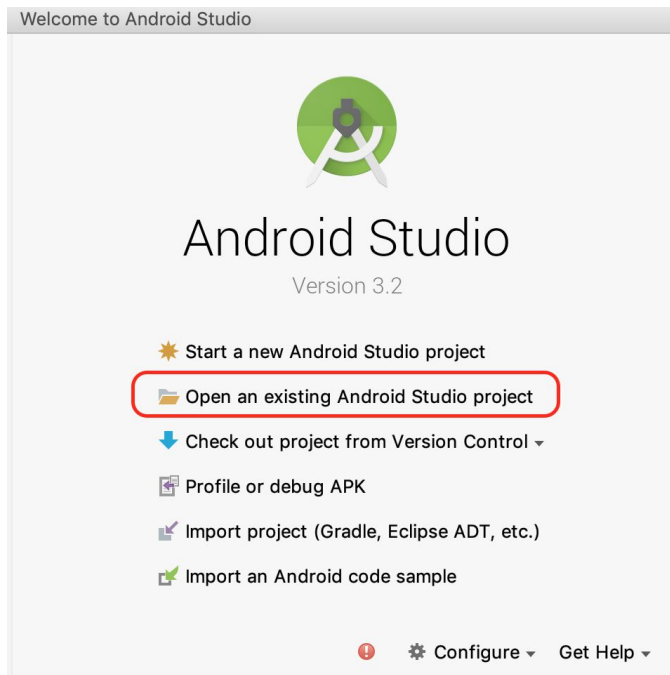


TRABAJO EN CONJUNTO

1. Abrir Android studio
2. Abrir el proyecto



3. Conectar el teléfono y correr la aplicación
4. En **app > assets** abrir la imagen de la tierra
5. Con la aplicación observar la tierra y se debe de enmarcar la imagen

PRIMERA ACTIVIDAD

1. En la carpeta app > assets agregar los archivos del laberinto en la carpeta green maze
2. Modificar la clase **AugmentedImageRenderer.java** y agregar el nuevo modelo a agregar. (se pueden eliminar los anteriores)

```
private final ObjectRenderer mazeRenderer = new ObjectRenderer();
```

3. En el método que crea los modelos, agregar esta nueva textura:

```
public void createOnGUIThread(Context context) throws IOException {
```

```
    mazeRenderer.createOnGUIThread(
```

```
        context, "models/green-maze/GreenMaze.obj", "models/frame_base.png");
```

```
    mazeRenderer.setMaterialProperties(0.0f, 3.5f, 1.0f, 6.0f);
```

```
}
```

4. Por último, dibujamos el laberinto:

```
public void draw(
```

```
    float[] viewMatrix,
```

```

float[] projectionMatrix,
AugmentedImage augmentedImage,
Anchor centerAnchor,
float[] colorCorrectionRgba) {
float[] tintColor =
    convertHexToColor(TINT_COLORS_HEX[augmentedImage.getIndex() %
TINT_COLORS_HEX.length]);

final float maze_edge_size = 492.65f; // Magic number of maze size
final float max_image_edge = Math.max(augmentedImage.getExtentX(),
augmentedImage.getExtentZ()); // Get largest detected image edge size

Pose anchorPose = centerAnchor.getPose();

float mazzScaleFactor = max_image_edge / maze_edge_size; // scale to set Maze to
image size
float[] modelMatrix = new float[16];

// OpenGL Matrix operation is in the order: Scale, rotation and Translation
// So the manual adjustment is after scale
// The 251.3f and 129.0f is magic number from the maze obj file
// We need to do this adjustment because the maze obj file
// is not centered around origin. Normally when you
// work with your own model, you don't have this problem.
Pose mozeModelLocalOffset = Pose.makeTranslation(
    -251.3f * mazzScaleFactor,
    0.0f,
    129.0f * mazzScaleFactor);
anchorPose.compose(mozeModelLocalOffset).toMatrix(modelMatrix, 0);
mazeRenderer.updateModelMatrix(modelMatrix, mazzScaleFactor,
mazzScaleFactor/10.0f, mazzScaleFactor);
mazeRenderer.draw(viewMatrix, projectionMatrix, colorCorrectionRgba, tintColor);
}

```

5. Debemos corregir el error que tenemos, esto lo logramos agregando el siguiente método en la clase **ObjectRenderer.java**

```

public void updateModelMatrix(float[] modelMatrix, float scaleFactorX, float scaleFactorY,
float scaleFactorZ) {
float[] scaleMatrix = new float[16];
Matrix.setIdentityM(scaleMatrix, 0);
scaleMatrix[0] = scaleFactorX;
scaleMatrix[5] = scaleFactorY;
scaleMatrix[10] = scaleFactorZ;
Matrix.multiplyMM(this.modelMatrix, 0, modelMatrix, 0, scaleMatrix, 0);
}

```

6. Volvemos a correr la aplicación y ahora, al cargar la tierra, veremos un pequeño laberinto sobre esta

SEGUNDA ACTIVIDAD

Agreguemos un muñeco al laberinto

1. Modificar la clase **AugmentedImageRenderer.java** y agregar el nuevo modelo a agregar.
`private final ObjectRenderer andyRenderer = new ObjectRenderer();`
2. En el método que crea los modelos, agregar esta nueva textura (No borrar la anterior, pues borramos el laberinto) :

```
public void createOnGThread(Context context) throws IOException {
```

```
    andyRenderer.createOnGThread(
```

```
        context, "models/andy.obj", "models/andy.png");
```

```
    andyRenderer.setMaterialProperties(0.0f, 3.5f, 1.0f, 6.0f);
```

```
}
```

3. Por último, dibujamos el muñeco agregando esto al final del método de dibujo hecho anteriormente:

```
Pose andyModelLocalOffset = Pose.makeTranslation(
```

```
    0.0f,
```

```
    0.1f,
```

```
    0.0f);
```

```
    anchorPose.compose(andyModelLocalOffset).toMatrix(modelMatrix, 0);
```

```
    andyRenderer.updateModelMatrix(modelMatrix, 0.05f); // 0.05f is a Magic number to scale
```

```
    andyRenderer.draw(viewMatrix, projectionMatrix, colorCorrectionRgba, tintColor);
```

4. Nuevamente corremos nuestra aplicación y veremos el muñequito dentro del laberinto