

Uniting Twist Developer Documentation

Nick Werle

March 24, 2017

1 Activities

Uniting twist contains two activities: `MainActivity` and `StartActivity`. Each has an associated layout of the same name.

1.1 StartActivity

The start activity is the "main menu" of sorts - where the player "begins" the game. Displays the high score and starts the `MainActivity`

1.2 MainActivity

The main activity is the main portion of the app - the activity where the game is played. Provides logic for setting up renderer, losing.

2 Classes

2.1 MyGLRenderer

Implementation for `GLSurfaceView.Renderer`. Provides methods to set up the shader, debug methods for open GL errors, etc.

2.2 Point

Basic XYZ point implementation.

2.3 Scene

Holds the scene to be drawn, as well as update and pause/resume methods to handle ticks.

2.4 Hexagon

Object that handles the rendering an individual hexagon. Not for game logic.

2.5 GyroscopeListener

Listens to the Gyroscope for `onSensorChanged` events, updating the angle and time of each event.

2.6 GraphGameHexagon

Handles the graph that controls which hexagons are connected to which.

2.7 GameHexagonContainer

Holds a `GameHexagon` and the point it exists at. Provides a method to draw the hexagon.

2.8 GameHexagon

Object containing each hexagon as an object of the game (as opposed to one for rendering). Provides logic for moving, rotating, and drawing itself on screen.

2.9 GameField

Handles the hexagons that fly about the screen and the graph of connected hexagons too. Spawns hexagons on a timer, detects collisions of hexagons, and updates hexagons each tick.

2.10 FlyingGameHexagon

A `GameHexagonContainer` for hexagons that are still in flight. Updates it's position based on the angle and time since last update.

2.11 FieldGraph

Closely related to the `GameGraphHexagon`. Handles the game graph in relation to the hexagons on the field. Handles logic for removing clusers, rotating the field, and drawing.

2.12 Engine

Handles tick logic, updating the scene and then rendering each tick. Accomplishes this by recording the time of the last tick and then attempting to rerender when a minimum amount of time has elapsed.

2.13 ColorUtils

Color related utilities.

2.14 Circle

Draws the circle which defines the radius the player's placed tiles may not exceed.