

Gyroscope and Accelerometer for WebGL

This code package provides a Unity script, SK_DeviceSensor, designed to facilitate the integration of gyroscope and accelerometer data into Unity projects, specifically tailored for WebGL builds. The package includes functionality to start and stop listeners for both gyroscope and accelerometer events.

Key Features:

Gyroscope Integration: Start and stop listeners for gyroscope data.

Conversion methods to translate gyroscope readings into Quaternion rotations.

Accelerometer Integration: Start and stop listeners for accelerometer data.

Conversion methods to process accelerometer readings and rotations.

Ease of Use: Simple static Start and stop listeners for accelerometer data. Conversion methods to process accelerometer readings and rotations eObject in the Unity scene.

Instructions:

UnityEvent and Data Classes:UnityEvent<GyroscopeData> and UnityEvent<AccelerometerData> are Unity's event system used to handle gyroscope and accelerometer data, respectively.

GyroscopeData and AccelerometerData are data classes that hold the relevant sensor readings.

Initialization:The script contains a static class SK_DeviceSensor with static methods and events.

The Initialize() method initializes the gyroscope and sets up the JavaScript library (SK_GyroscopeJsLib) for WebGL.

Start and Stop Listeners: StartGyroscopeListener and StartAccelerometerListener methods start listening for gyroscope and accelerometer data, respectively. They take UnityAction parameters for handling the data.

Corresponding Stop methods stop listening for data.

Platform Checks: There are checks for the platform using UNITY_EDITOR and UNITY_WEBGL. If the platform is not WebGL or in the Unity Editor, warning messages are displayed.

Reading Events: The methods OnGyroscopeReading and OnAccelerometerReading are callback methods that handle incoming sensor readings.

They convert the JSON payload into GyroscopeReadingEvent or AccelelometerReadingEvent objects and then convert the data into GyroscopeData or AccelerometerData.

SensorRotationToQuaternion: SensorRotationToQuaternion methods convert gyroscope readings to Quaternion rotations.

The readings are converted from Euler angles to Quaternion rotations.

Instructions for Use:

Integration: Attach the script SK_DeviceSensor to a GameObject in your Unity scene. (For debugging purposes)

Gyroscope: To start listening to gyroscope data, call SK_DeviceSensor.StartGyroscopeListener(yourCallbackMethod) and pass a callback method that takes a GyroscopeData parameter.

To stop listening, call SK_DeviceSensor.StopGyroscopeListener().

Accelerometer: To start listening to accelerometer data, call SK_DeviceSensor.StartAccelerometerListener(yourCallbackMethod) and pass a callback method that takes an AccelerometerData parameter.

To stop listening, call SK_DeviceSensor.StopAccelerometerListener().

Callback Methods: Implement your callback methods to handle the gyroscope and accelerometer data as needed.

Platform Checks: Be aware of the platform limitations mentioned in the warning messages for non-WebGL or Unity Editor platforms.

Make sure to test the integration in a WebGL build to ensure compatibility and proper functioning of gyroscope and accelerometer features. Adjust your application's behavior based on the sensor data received in your callback methods.