GO Directions – Directions plugin for Go Map

By Alan Grant

Documentation

GO Directions is a plugin for Go Map that will provide an easy access to road directions, meaning that **you will need GO Map** already in your project before importing this plugin.

GO Directions is really easy to use and to customize, if you are already skilled with GoMap inspector interfaces you will find it really similar.

This plugin will make your life easier if you want to add this features to Go Map:

- Directions from point A to point B using coordinates/address
- Directions from current location to destination using coordinates/address
- Driving, walking, transit, bicycling travel modes supported.
- Mixed directions like walking+transit+walking.
- Directions via waypoints.
- Complete route mesh customization via unity inspector.
- Full Google Directions API wrapper, data it's converted in objects and components of the unity engine.

The Following documentation is based on the "GODirections - Demo" scene. After reading this document you can safely delete anything inside the "demo" folder and make your own scene.

*The data used comes from Google directions API, please refer to their conditions about traffic limit and pricing. *
https://developers.google.com/maps/documentation/directions/

The scene contains 4 game objects:

- Location Manager
- Map
- Avatar
- GODirections

This documentation covers the GODirections object and the scripts used to get directions.

GO Directions
GODirections.cs
This is the core class of this plugin, allowing you to set the main parameters for your direction meshes.
When a route mesh is created you'll see it as a child of the GODirections gameobject.

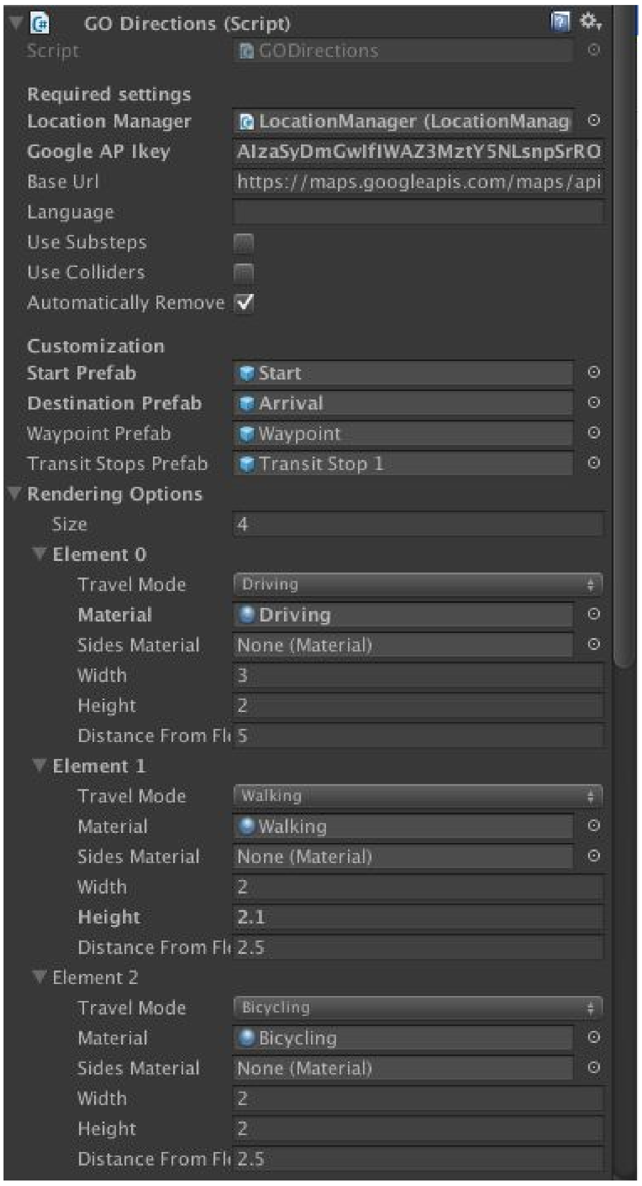
Parameters:

- (script) LocationManager: a link to the Location Manager script in the scene. Be sure to make this link or the directions won't work.
- (string) Google API key: the api key to be used for your google requests. Be sure to make your api key **here**.
- (string) Base Url: Don't change this url, it's the one used by Google to host api services.
- (string) Language: This set the language used in the turn-by-turn directions (GODirectionsRoute.cs). Leave it blank to make it localized automatically.
- (bool) Use substeps: Enabling this flag will split the route children (steps) in many gameobjects as are the turns of the returned route. Sometimes it's unnecessary to do that as it reduces the graphic precision of the rendered objects.
- (bool) Automatically Remove Routes: Enabling this flag will clear any previous route when a new it's returned.
- (GODirectionsRendering[]) a list of rendering options for each of the travel modes available.

GODirectionsRendering
Use this classes to customize the style of each travel mode.

Parameters:

- (enum) Travel Mode: select the travel mode you want to customize.
- (Meterial) Material: the material used for the top side of the step object.

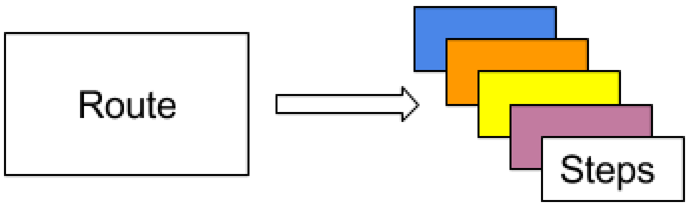


GO Directions (Script)

Script	GODirections
Required settings	
Location Manager	LocationManager (LocationManag
Google AP Ikey	AIzaSyDmGwlfIWAZ3MztY5NLsnpSrRO
Base Url	https://maps.googleapis.com/maps/api
Language	
Use Substeps	☐
Use Colliders	☐
Automatically Remove	✓
Customization	
Start Prefab	Start
Destination Prefab	Arrival
Waypoint Prefab	Waypoint
Transit Stops Prefab	Transit Stop 1
Rendering Options	
Size	4
Element 0	
Travel Mode	Driving
Material	Driving
Sides Material	None (Material)
Width	3
Height	2
Distance From Fl	5
Element 1	
Travel Mode	Walking
Material	Walking
Sides Material	None (Material)
Width	2
Height	2.1
Distance From Fl	2.5
Element 2	
Travel Mode	Bicycling
Material	Bicycling
Sides Material	None (Material)
Width	2
Height	2
Distance From Fl	2.5

- (Meterial) Sides Material: the material used for the sides of the step object. When left to null the step object is created only with the main material.
- (float) Width: The width of the step object.
- (float) Height: The height of the step object.
- (float) Distance from floor: The y value for the step objects.

Directions structure



(GODirections, GODirectionsRoute, GODirectionsSteps)

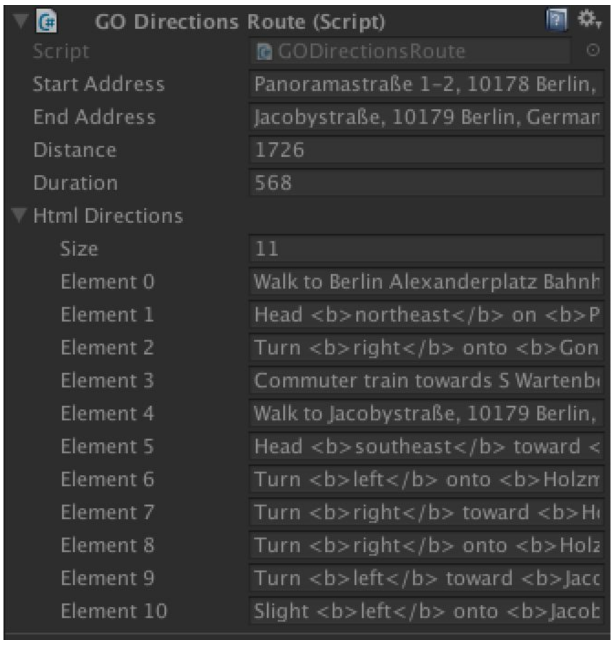
GO Directions Route

GODirectionsRoutes.cs

This class is attached to the route GameObjects and contains all the information about the returned route.

Parameters:

- (string) Start Address: the geocoded address of the starting point.
- (string) End Address: the geocoded address of the destination point.
- (int) Distance: the distance of the whole route in meters.
- (int) Duration: the duration of the whole route in seconds.
- (List<string>) Html Directions: the turn by turn directions in html format. You can use this information to show the user an overview of the complete route or some step by step directions. (this strings are localized in the language set on the GODirections class.)



GO Directions Route (Script)

Script	GODirectionsRoute
Start Address	Panoramastraße 1-2, 10178 Berlin,
End Address	Jacobystraße, 10179 Berlin, German
Distance	1726
Duration	568
Html Directions	
Size	11
Element 0	Walk to Berlin Alexanderplatz Bahnh
Element 1	Head <b>northeast</b> on <b>P
Element 2	Turn <b>right</b> onto <b>Gon
Element 3	Commuter train towards S Wartenb
Element 4	Walk to Jacobystraße, 10179 Berlin,
Element 5	Head <b>southeast</b> toward <
Element 6	Turn <b>left</b> onto <b>Holzm
Element 7	Turn <b>right</b> toward <b>H
Element 8	Turn <b>right</b> onto <b>Holz
Element 9	Turn <b>left</b> toward <b>Jacc
Element 10	Slight <b>left</b> onto <b>Jacob

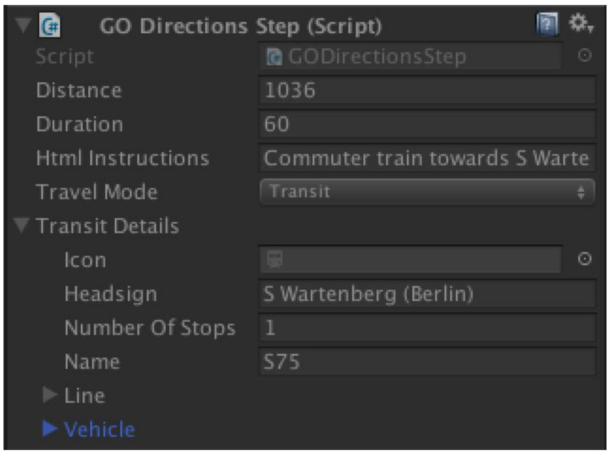
GO Directions Step

GODirectionsStep.cs

This class is attached to step GameObjects and contains all the information about the route portion and it's travel mode.

Parameters:

- (int) Distance: the distance of the step in meters.
- (int) Duration: the duration of the step in seconds.
- (string) Html Directions: the directions in html of the step.
- (enum) travel mode: the travel mode of this step.



GO Directions Step (Script)

Script	GODirectionsStep
Distance	1036
Duration	60
Html Instructions	Commuter train towards S Warte
Travel Mode	Transit
Transit Details	
Icon	
Headsign	S Wartenberg (Berlin)
Number Of Stops	1
Name	S75
Line	
Vehicle	

Transit details
When selecting “transit” as the initial travel mode the returned route can be a composition of walking and transit steps. The transit steps are made by bus, train, ecc and have more information about the transportation service.

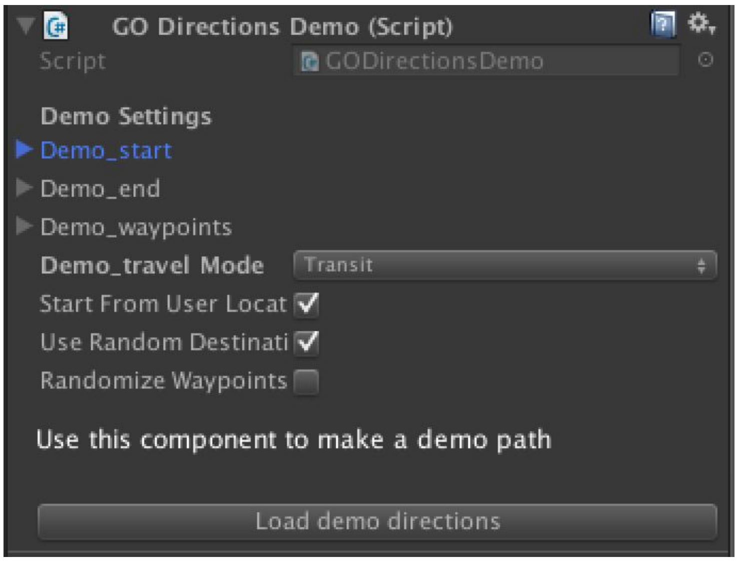
Parameters:

- (Texture2D) Icon: The icon used for the transportation kind.
- (string) headsign: text of the headsign.
- (int) number of stops of this transit step.
- (string) Name: line number of transportation custom name.

GO Directions Demo
GODirectionsDemo.cs
This class allows you to generate routes from A to B or from the user location to destination, randomly or not, as a test device for this plugin. Use this to try the demo and make your own customization style.

Parameters:

- (Coordinates) Demo_start: the route starting point.
- (Coordinates) Demo_start: the route destination.
- (Coordinates[]) Demo_start: the route waypoints.
- (enum) Demo_ travel mode: the travel mode for the route.
- (bool) Start from user location: flag this to override starting point.
- (bool) Use random Destination: flag this to override the destination point and make it random.
- (bool) Randomize waypoints: flag this to generate random waypoints in the demo route.
- (Button) Load demo directions: press this button while the application is running to request a demo route.


GO Directions Demo (Script)
Script: GODirectionsDemo
Demo Settings
Demo_start
Demo_end
Demo_waypoints
Demo_travel Mode: Transit
Start From User Locat ✓
Use Random Destinati ✓
Randomize Waypoints ☐
Use this component to make a demo path

Demo scene UI
The GO Directions demo scene contains various examples of Unity UI to show directions html information. You can use this UI example directly in your project if you like the way they are made or replace them with whatever UI you want (or just delete them!).
The purpose of this paragraph is not only to show which UI elements are used in the scene but to make you understand how GoDirections exposes data and how you can show it in your UI pattern.

Prefabs oriented pattern

With Go Directions you can choose some prefabs that will be placed in the Start location, the Arrival, every waypoint, and transit stops. It's **up to these prefabs to implement the logic** that gets the information from the route object and display it.

For example in the demo scene, once added, **the prefab for transit stops** (bus, train, etc) has both the logic to build up text and icon placed on the 3D model and an entire canvas that will slide up when you tap on the prefab. The canvas will show a description for the stop name, transportation number, and departure time.
If you take a look to the GOTransitPrefabDemo.cs component that's attached to the transit prefab you will see how it retrieves information from a component named GOStopDetails. This component will be attached to every transit prefab you want to use with Go Directions and contains every transportation stop information you'll need.

The start/stop/waypoints prefabs unlike the transit one won't have any specific component added to them. As you can see from the GOStartStopPrefab.cs component (attached to this prefabs) you can get all the directions info by just getting the GODirectionsRoute component from the hierarchy.
The GOStartStopPrefab.cs also handles a sliding canvas that will show the information about the entire route.

Directions through code

public IEnumerator RequestDirectionsWithCoordinates(Coordinates start, Coordinates end, Coordinates [] waypoints, GOTravelModes travelmode)

- Use this method to request directions between gps coordinates.
- Add waypoint coordinates if you want, nullable if you don't need it.

public IEnumerator RequestDirectionsFromUserLocation(Coordinates end, Coordinates [] waypoints, GOTravelModes travelmode)

- Use this method to request directions between current user location and a coordinate.
- Add waypoint coordinates if you want, nullable if you don't need it.

public IEnumerator RequestDirectionsWithAddresses(string start, string end, string [] waypoints, GOTravelModes travelmode)

- Use this method to request directions between adresses.
- Add waypoint addresses if you want, nullable if you don't need it.

public IEnumerator RequestDirectionsFromUserLocationToAddress(Coordinates start, string end, GOTravelModes travelmode)

- Use this method to request directions between current user location and an address.

First start guide

This part is going to explain how to add this plugin the GOMap demo scene.

1. Start a new project.

2. Import GoMap asset and open the demo scene.
3. Import GoDirections asset.
4. Drag into the scene the GODirections prefab.
5. In the GODirections.cs inspector link the location manager that's in the scene.
6. Make a Google API Key **here** and put it in the GODirections.cs inspector.
7. Run the scene and press the button to make a demo route.

FAQ

- **No questions yet**: Do you need support? write me at alangrant.unity@gmail.com