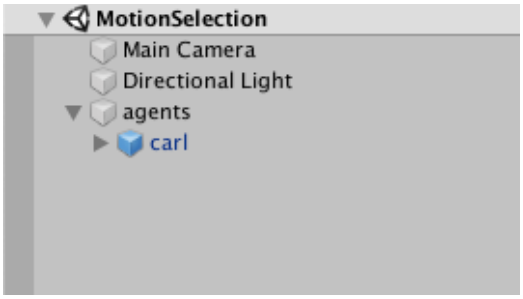
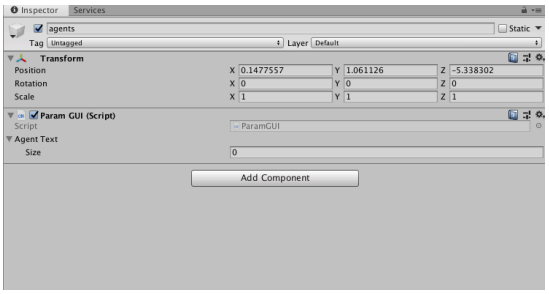
Scene Hierarchy

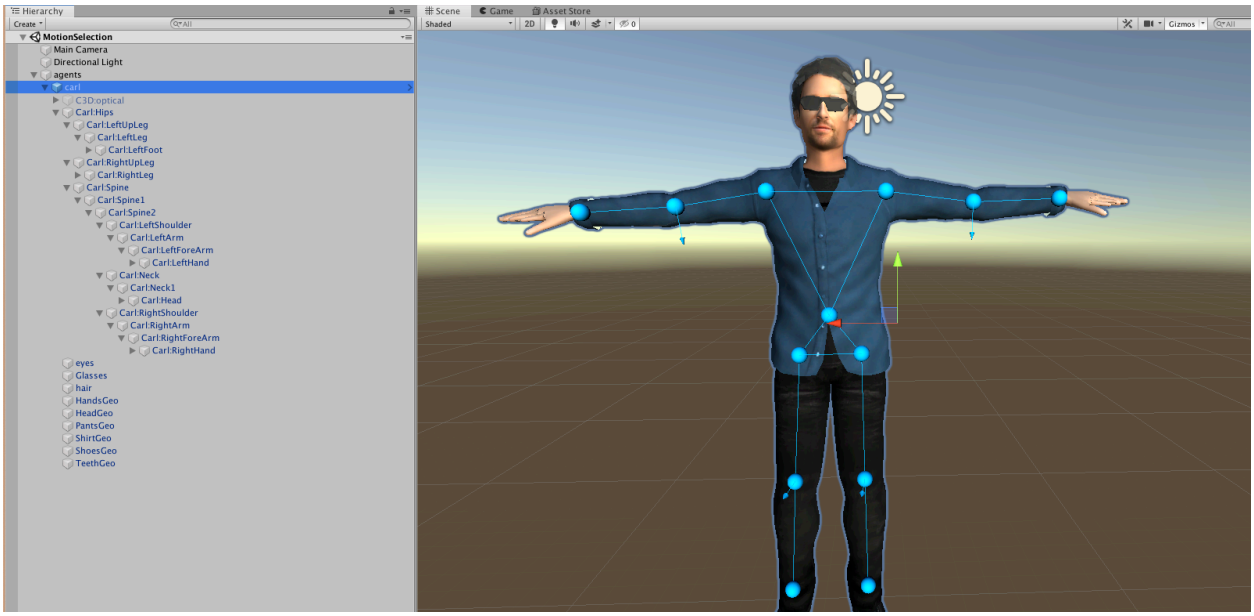
All the characters must be placed under a parent GameObject. In this case, the parent game object is called “agents”, but it can take any name.



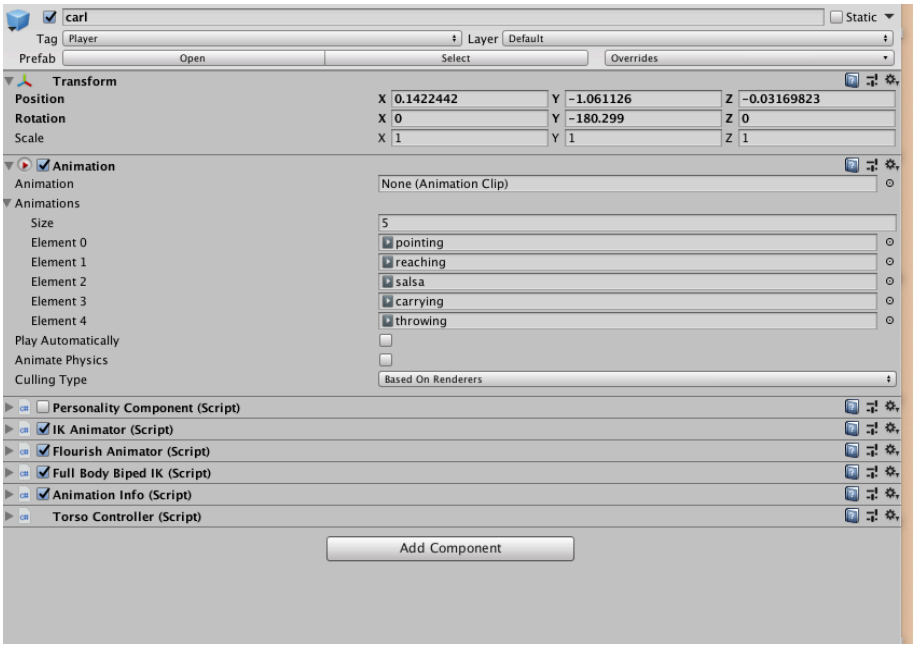
The GUI script to assign the personality parameters is attached to “agents”.



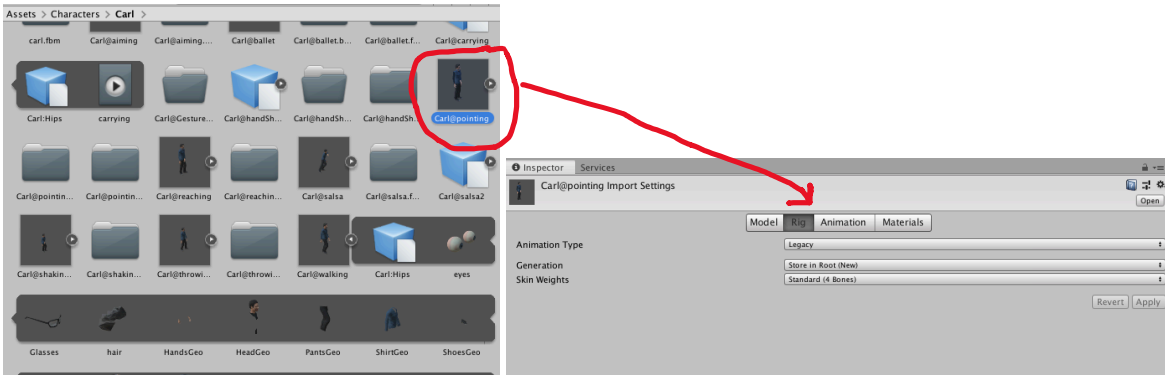
In this scene, we have one character, named as “carl”.



The character should have the following components and scripts:
-Transform
-Animation
-Personality Component
-IK Animator
-Flourish Animator
-Full Body Biped IK
-Animation Info
-Torso Controller



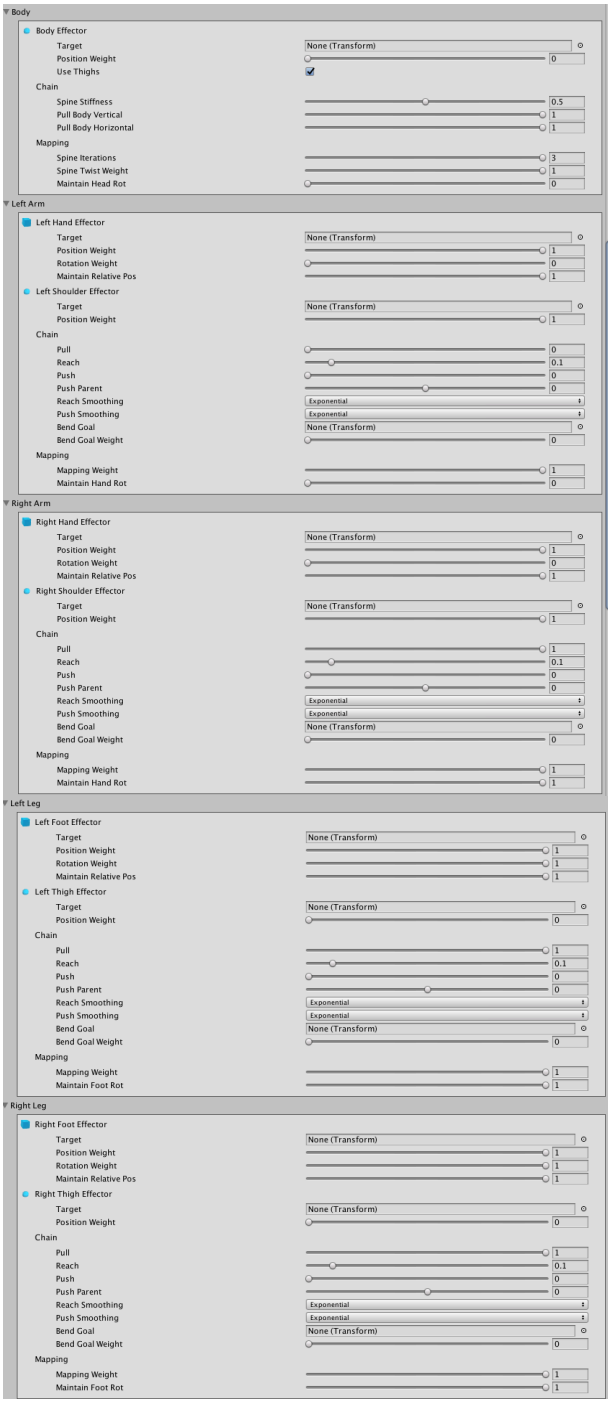
Animation
Animations can be any size. Just add them to the animations list. However, note that the animation rigs must be marked as legacy.



Full Body Biped IK

FullBodyBipedIK.cs is under Assets/RootMotion/IK Components/

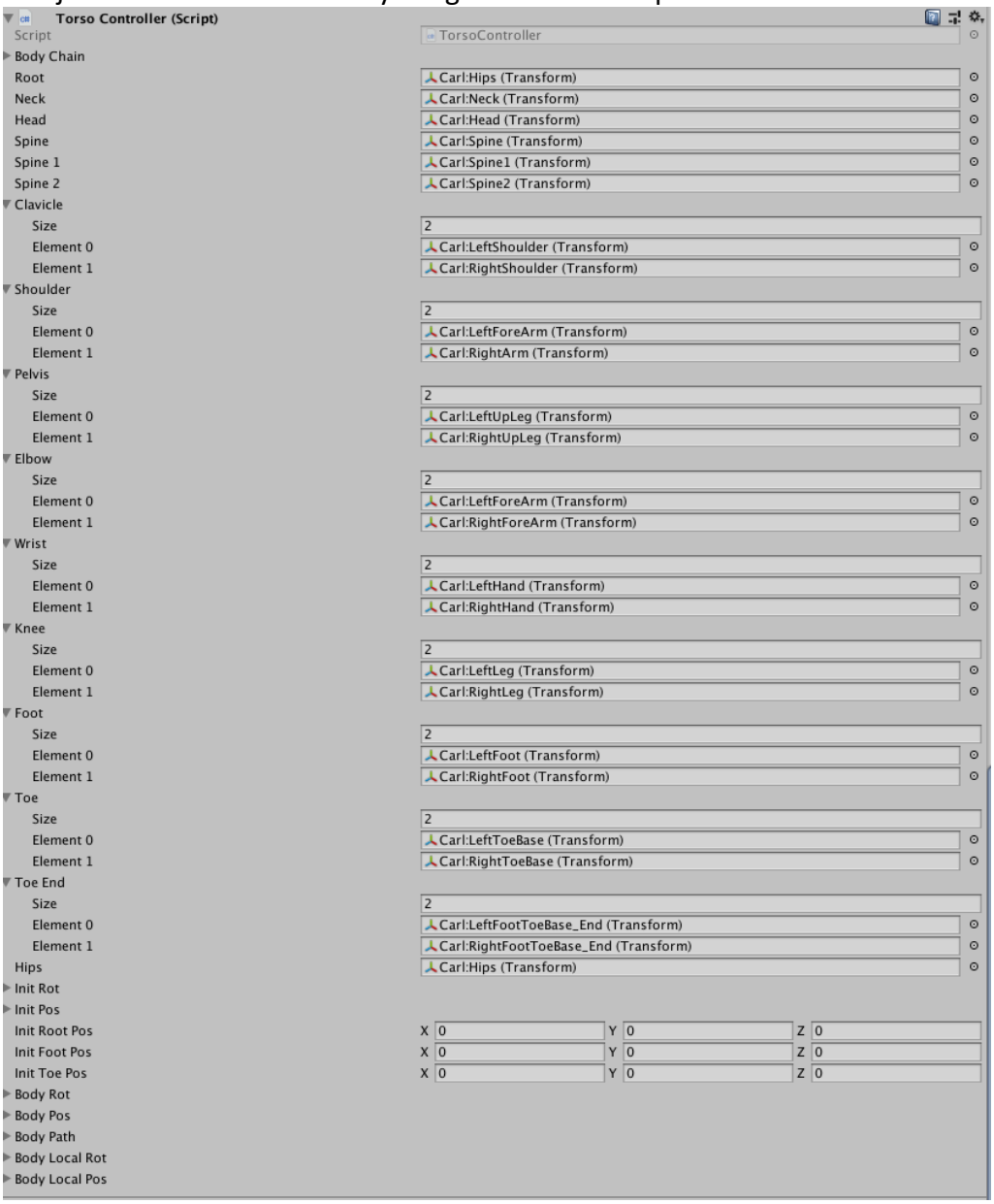
The joints are automatically filled in; however, some parameters should be manually updated as follows:



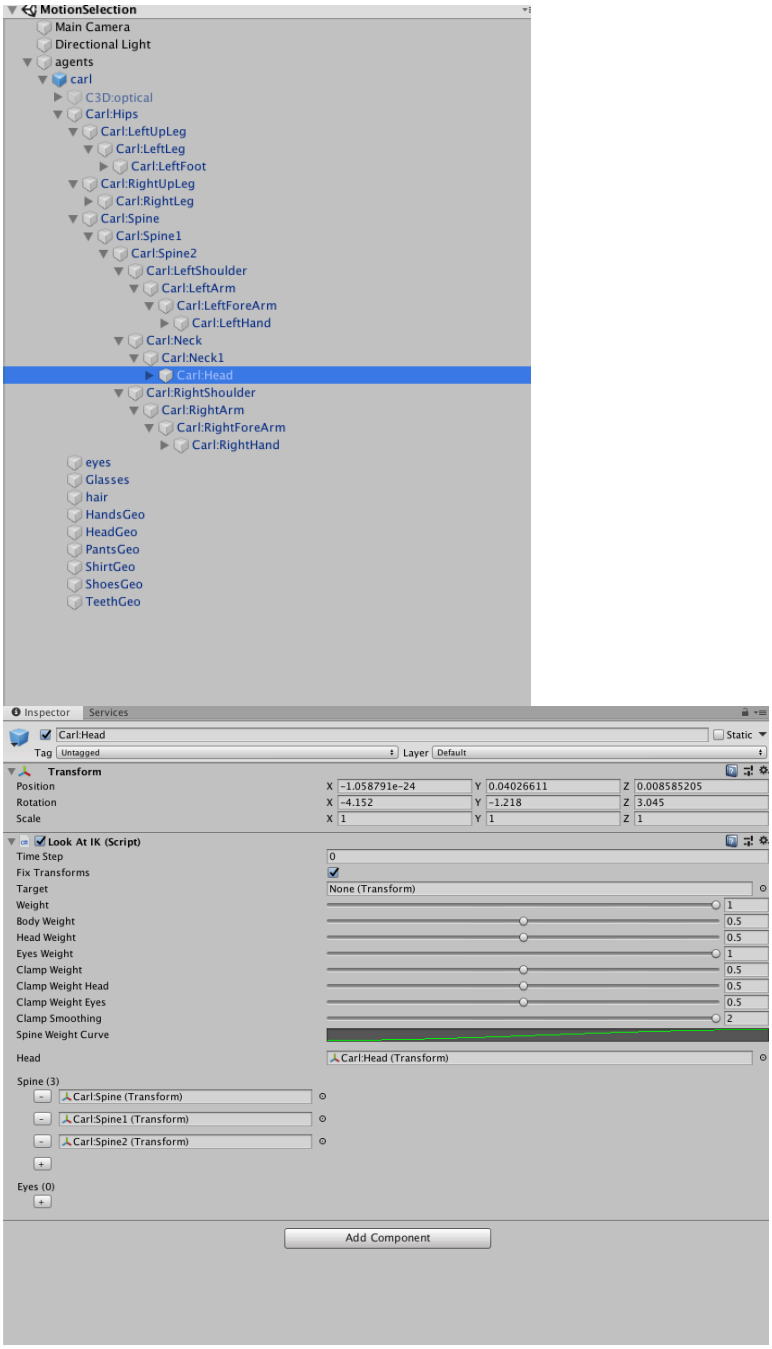
Torso Controller

TorsoController.cs is under Assets/Scripts

The joints need to be manually assigned for this script:



In addition to FullBodyBipedIK, we assign LookAtIK to control the character's head oritentation. LookATIK.cs is under Assets/RootMotion/IK Components/ LookAtIK script should be added to the character's head as:

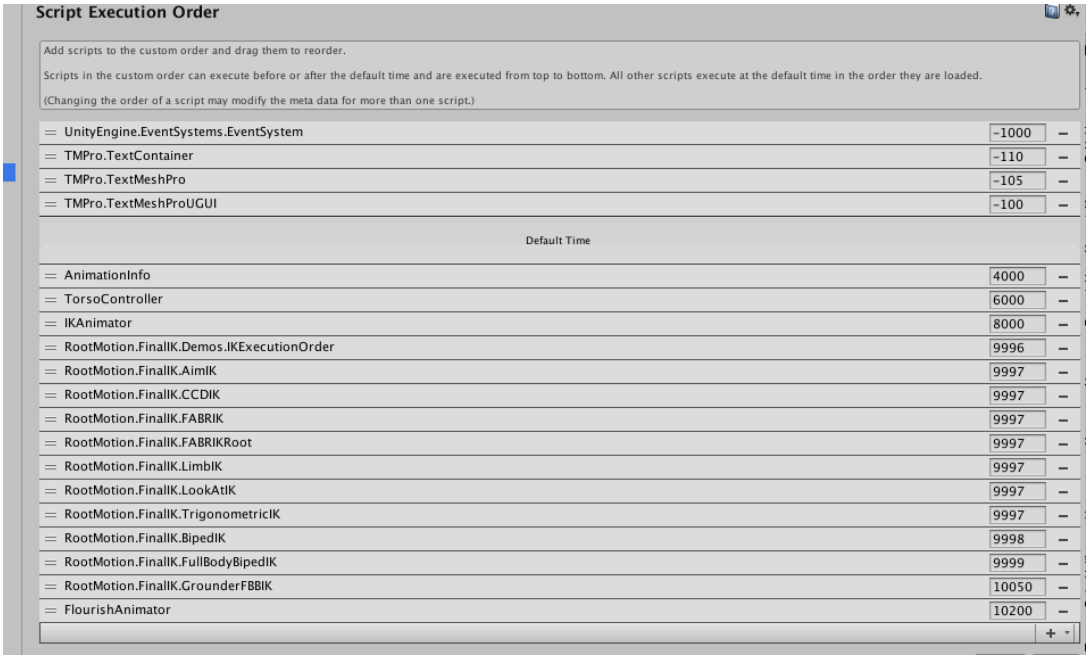


Project Structure and Other Dependencies

Assets/Libs:
An older version of Meta.Numerics package is used in the project. The required dll is under Assets/Libs/
Assets/Scripts:
All the scripts used in the project.
Assets/Scenes:
The demos are under this directory.
Assets/Characters:
All the characters, including their materials, animations and prefabs are under individual folders
Assets/Resources:
Text files used to read parameters from
Assets/RootMotion:
RootMotion IK libraries

Script Execution Order

Editor→ Project Settings→Script Execution order



The order of the scripts should be as AnimationInfo < TorsoController < IKAnimator < FinalIK (both LookAtIK and FullBodyBipedIK) < FlourishAnimator