# High Level Base Pseudocode

## GameObject

**Method** GameObject() **is** Class Constructor

**Input:** *None.*

**Output:** *None.*

*Initialise a particular instance of this class*

**End Method**

**function** GetLocation() **is** a ‘getter’ function

**Input:** *None.*

**Output:** *Vector3 Location: This GameObject’s current location.*

*Return this GameObject’s Location member property*

**Return** Location

**End function**

**function** GetRotation() **is** a ‘getter’ function

**Input:** *None.*

**Output:** *Vector3 Rotation: This GameObject’s current rotation.*

*Return this GameObject’s Rotation member property*

**Return** Rotation

**End function**

**function** GetScale() **is** a ‘getter’ function

**Input:** *None.*

**Output:** *Vector3 Scale: This GameObject’s current scale.*

*Return this GameObject’s Scale member property*

**Return** Scale

**End function**

**Method** MoveObject() **is**

**Input:** *None.*

**Output:** *None*.

*Handle translation of this object*

**End Method**

**Method** RotateObject() **is**

**Input:** *None.*

**Output:** *None*.

*Handle rotation of this object*

**End Method**

**Method** ScaleObject() **is**

**Input:** *None.*

**Output:** *None*.

*Handle scaling of this object*

**End Method**

## MoveableObject

**Method** MoveableObject() **is** Class Constructor

**Input:** *None.*

**Output:** *None.*

*Initialise a particular instance of this class*

**End Method**

**Method** OnCollisionBegin() **is** a class event handler

**Input:** *None.*

**Output:** *None*.

*Handle the event of this object beginning collision with another object*

**End Method**

**Method** OnCollisionEnd() **is** a class event handler

**Input:** *None.*

**Output:** *None*.

*Handle the event of this object Ending collision with another object*

**End Method**

## EntityControlledObject

**Method** EntityControlledObject() **is** Class Constructor

**Input:** *None.*

**Output:** *None.*

I*nitialise a particular instance of this class*

**End Method**

**function** GetAIControlled **is** a ‘getter’ function

**Input:** *None.*

**Output:** *bool AIControlled: This flag indicates whether this object is controlled by the Player, or by an AI system.*

*Return this object’s AIControlled member property*

**Return** AIControlled

**End function**

## InputInterpretor

**Method** InputInterpretor() **is** Class Constructor

**Input:** *None.*

**Output:** *None.*

I*nitialise a particular instance of this class*

**End Method**

**function** InputProvidedEvent() **is** an event dispatcher

**Input:** *None.*

**Output:** *InputStruct InputEvent: The input event that has been identified*

R*eturn an input event corresponding to the input provided*

**Return:** *InputEvent*

**End function**

**Virtual Method** UpdateCurrentInput() **is** overridable

**Input:** *None.*

**Output:** *None.*

*Update the current handling of input, for the given input Method*

**End Method**

## InputMethod

**Method** InputMethod() **is** Class Constructor

**Input:** *None.*

**Output:** *None.*

I*nitialise a particular instance of this class*

**End Method**

**function** GetInputInterpretationSystem() **is**  a ‘getter’ function

**Input:** *None.*

**Output:** *InputInterpretor InputInterpretationSystem: The system this InputMethod utilises for receiving input, for checking on discrite input, that may not necessarily fire an InputProvidedEvent().*

**Return** *InputInterpretationSystem*

**End function**

**Method** UpdateCurrentInput() **is**

**Input:** *None.*

**Output:** *None.*

*Update the current handling system for input, as per this particular InputMethod*

**End Method**

### PlayerEntity

**Method** PlayerEntity() **is** Class Constructor

**Input:** *None.*

**Output:** *None.*

I*nitialise a particular instance of this class*

**End Method**

**Function** GetCollectiblesPickedUp() **is** a ‘getter’ function

**Input:** *None.*

**Output:** *int CollectiblesPickedUp: The number of collectibles that the Player has picked-up on this current level.*

**Return:** CollectiblesPickedUp

**End function**

**Method** InitiateControlSystem() **is**

**Input:** *None.*

**Output:** *None.*

*Initialise the control system of the input Method, given the control system the Player is using*

**End Method**

## AIEntity

**Method** AIEntity() **is** Class Constructor

**Input:** *None.*

**Output:** *None.*

I*nitialise a particular instance of this class*

**End method**

**Method** InitiateAILogicSystem **is**

**Input:** *None.*

**Output:** *None.*

*Initialise the logic system (FSM, Behaviour Tree etc.) that this AIEntity utilises*

**End method**

## AILogicSystem

**Method** AILogicSystem() **is** Class Constructor

**Input:** *None.*

**Output:** *None.*

I*nitialise a particular instance of this class*

**End method**

## GameState

**Method** GameState() **is** Class Constructor

**Input:** *None.*

**Output:** *None.*

I*nitialise a particular instance of this class*

**End method**

**Function** GetWorldGameObjects() **is** a ‘getter’ function

**Input:** *None.*

**Output:** *GameObject[] WorldGameObjects: An array of all of the GameObjects in the current level (world).*

**Return:** WorldGameObjects

**End function**

**Method** UpdateGameWorld() **is**

**Input:** *None.*

**Output:** *None.*

*Handle the update process for the current world’s logic (for all GameObjects)*

**End method**

## PlayerHUD

**Method** PlayerHUD() **is** Class Constructor

**Input:** *None.*

**Output:** *None.*

I*nitialise a particular instance of this class*

**End method**

## Renderer

**Method** Renderer() **is** Class Constructor

**Input:** *None.*

**Output:** *None.*

I*nitialise a particular instance of this class*

**End method**

## CollisionManager

**Method** CollisionManager() **is** Class Constructor

**Input:** *None.*

**Output:** *None.*

I*nitialise a particular instance of this class*

**End method**

## AILogicSystem

**Method** AILogicSystem() **is** Class Constructor

**Input:** *None.*

**Output:** *None.*

I*nitialise a particular instance of this class*

**End method**

# References

1. Wikipedia, 2017, Wikipedia: WikiProject Computer science/Manual of Style [Viewed on the 14/10/2017]. Available from: <https://en.wikipedia.org/wiki/Wikipedia:WikiProject_Computer_science/Manual_of_style#General_guidelines_for_writing_pseudocode>