**Maya Custom Smart Shelf – MCSS V 1.00**

**The MCSS project introduce a new way to manage a shelf object into the Maya environment.**

**The purpose of this custom element is to give to the IT – TD a more flexible way to insert and distribute new Scripts/Tools into the pipeline.**

**Setup**

**Local Machine:**

**In your userSetup.py**

**import maya.cmds as cmds**

**import runpy**

**cmds.evalDeferred("runpy.run\_path('/path\_to\_root\_folder/MCSS\_Root/MCSS\_Project/execute.py')")**

**Pipeline:**

**( . . . )**

**General Overview**

**The shelf object is divided in 4 areas:**

1. **Reload Button:**

* **Fetch and reload the shelf with the last version of the script/tool files.**

1. **Pop ups:**
   * + **Set of procedures collected and organized by its type of function. Can be**

**Skin operation, Toggle operations, Create operations.**

1. **Buttons:**
   * + **Call and execute single script files non related to each other**
2. **Tools:**
   * + **Every Tool created that comes with a user interface**

**Way to go**

**Step One >> User develops a new Script or Tool**

**-**

**Step Two >> Categorization. User goes to the project tree and stores the new piece of code or Tool Structure inside of its own category. Each developed project will have a different purpose in the shelf and have to meet some requirements to work with it.**

**If is a function and is related with a pop up menu:**

**The code have to be wrapped inside of a function with its name in camel case and setting “\*args, \*\*kwargs” parameters by default. Eg:**

**def MyNewFunction(\*args, \*\*kwargs):**

**This function have to be inserted into its own pop up category file, after the last function.**

**If the script is a single file and have nothing related with the pop up menus:**

**The file have to be named in camel case and inserted into scripts module folder**

**RootProject**

**- scripts**

**MyScriptFile.py**

**If is a Tool:**

**The tool root folder have to be stored into the “tools” module folder**

**RootProject**

**- tools**

**-MyNewTool**

**execute.py (This is important, must have this file in it)**

**Each tool have to have a file called “execute.py” , this file is the responsible for launching the tool and the shelf will look for it to make the tool init.**

**Icons**

**To set and icon with it respectively button:**

**Icon must be inserted into the icon folder format .PNG with the name exactly as**

**has been setup in the Function, script file or tool root folder.**

**-**

**Stage Three >> Reload shelf.**

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**