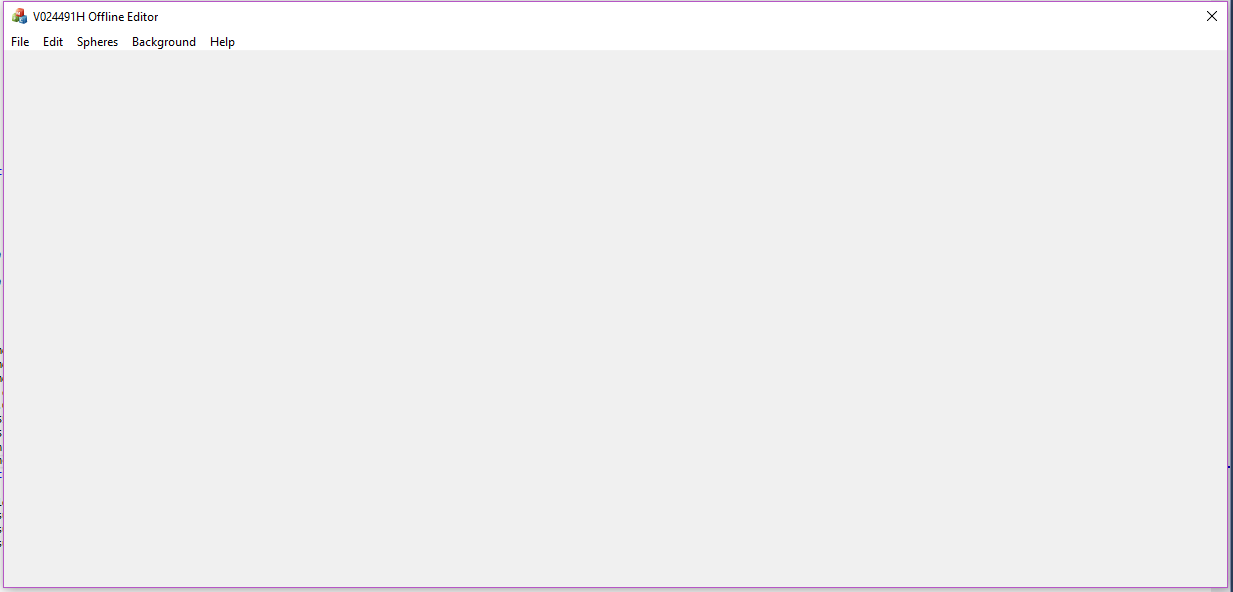
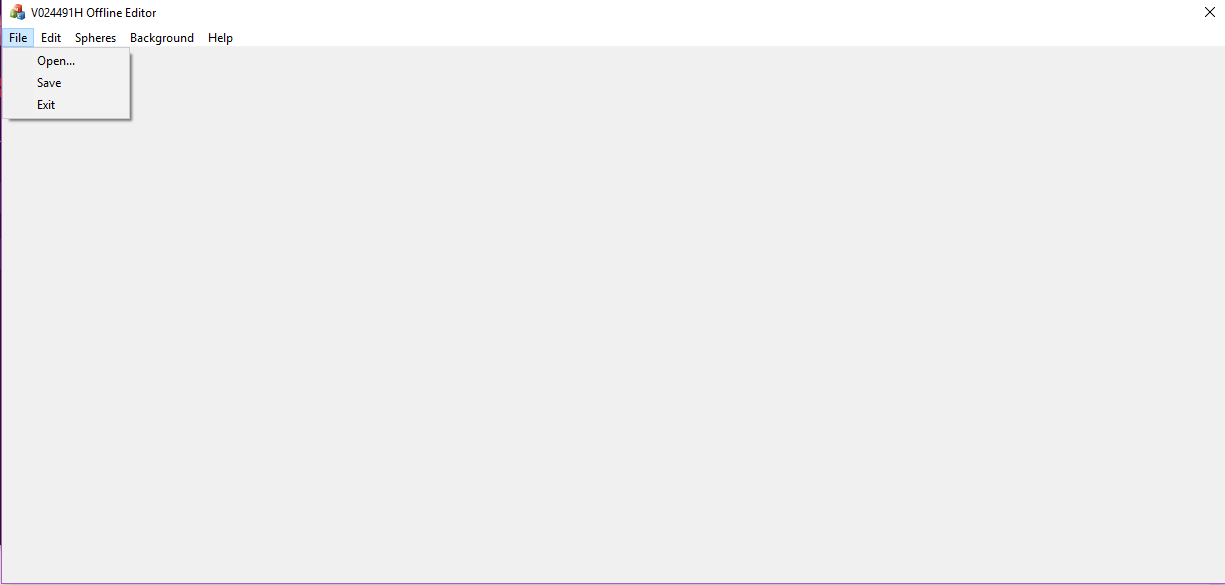
* **Offline editor:**

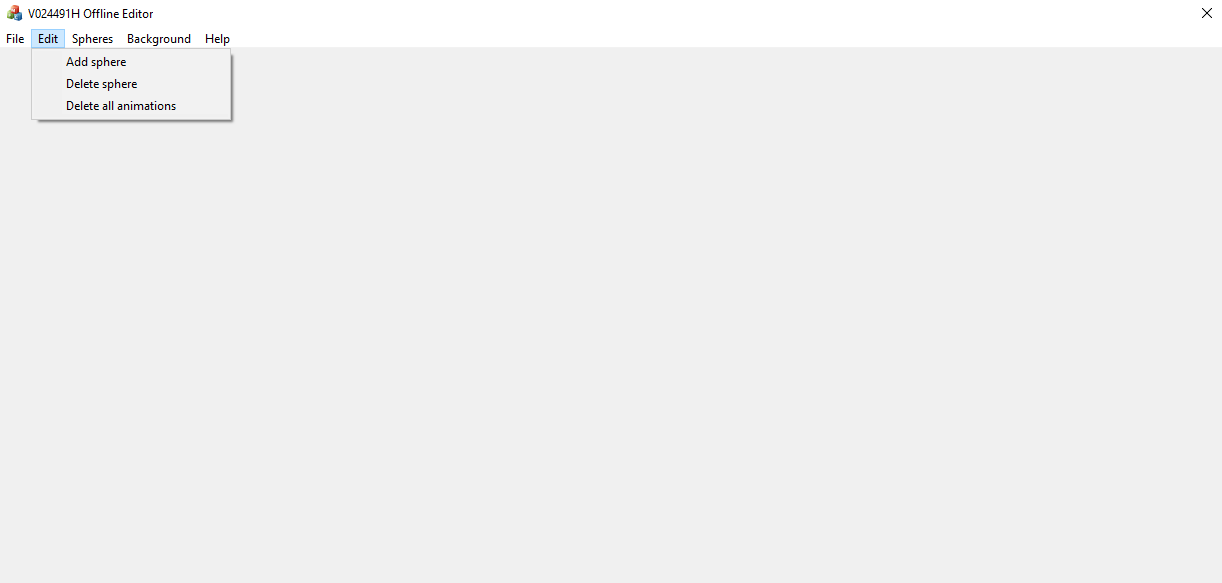
An offline editor has been created with the purpose of generating a *.json* file that the main framework can read, this is just a quick guide:

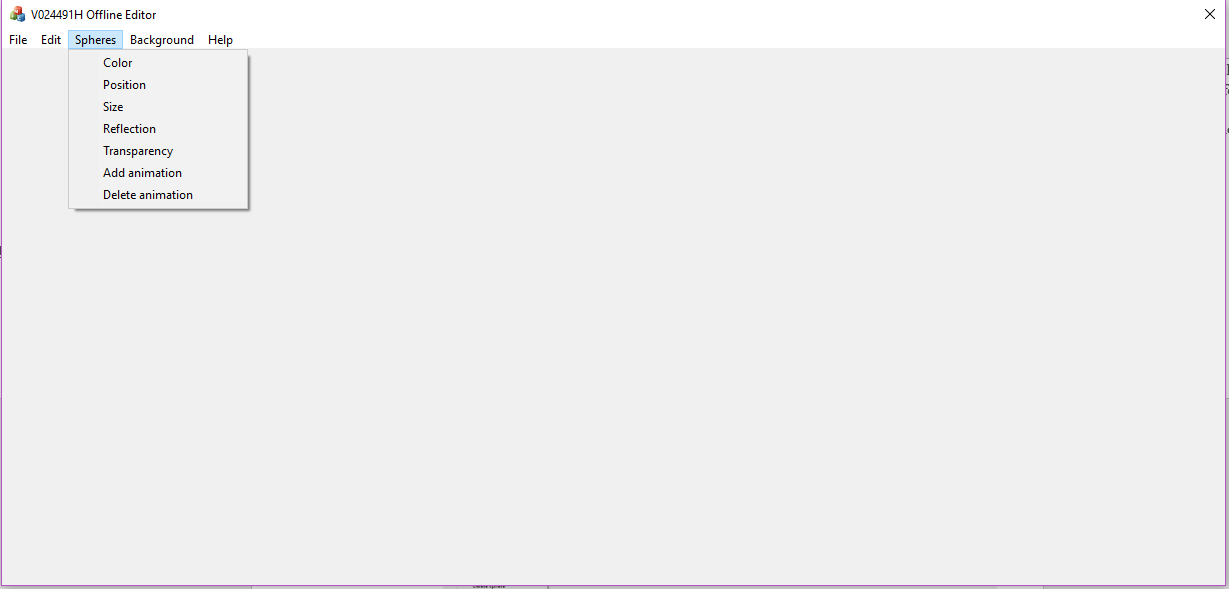
Figure 1: this offline editor was made with windows dialogs, it has a menu with multiple options. Install the required extensions to run the source code.

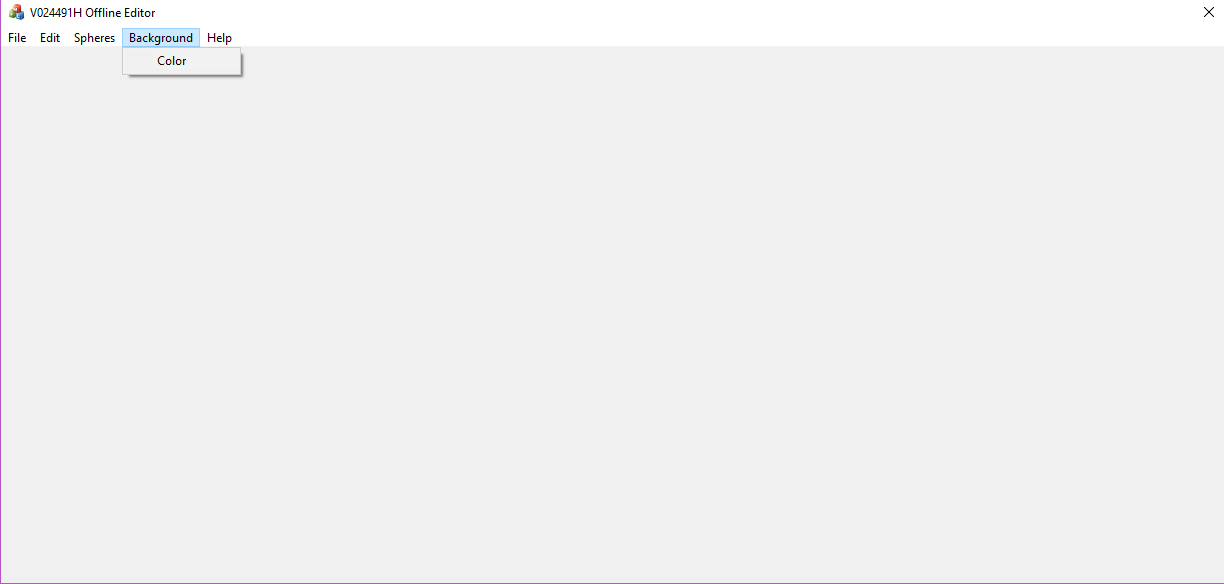
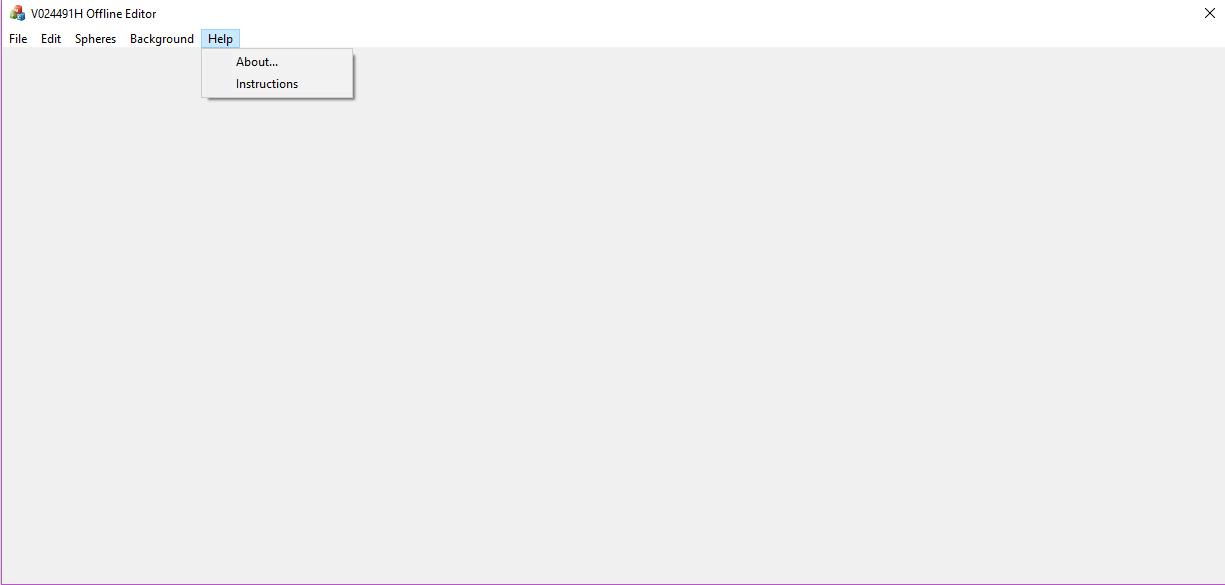


* **File:**



* **Open:** It currently does nothing; it was intended to have the function of searching for a file inside the drive to load all the information into the editor.
* **Save:** Saves the current information inside a *.json* file to be used for the framework.
* **Exit:** Closes the editor.
* **Edit:** 
* **Add sphere:** Opens a dialog window that allows you to input information about the sphere, creates a sphere and saves it inside a vector of spheres.
* **Delete sphere:** Opens a dialog box that has a list of the spheres, the sphere selected will be deleted after you press “delete”. This dialog box won’t open if there are no spheres to delete.
* **Remove all animations:** Removes the animations of all the spheres.
* **Spheres:**



* **Color:** Opens a dialog box that has a list of spheres and modifiable boxes/sliders that store the new R/G/b values of the sphere, after pressing OK, the changes will apply to the sphere selected.
* **Position:** Opens a dialog box that has a list of spheres and modifiable boxes that store the new X/Y/Z value of the sphere, after pressing OK, the changes will apply to the sphere selected.
* **Size:** Same behaviour of “Position” with the radius value.
* **Reflection:** Same behaviour of “Size” with the reflection value.
* **Transparency:** Same behaviour of “Size” with the transparency value.
* **Add animation:** Opens a dialog box that contains a list of spheres and modifiable boxes, the boxes on the left have with the same behaviour as “Position” showing the position of the sphere at the start of the animation, the boxes on the right side store the final position of the sphere after the animation; the box “Frames” is used to store the frames the animation will take. After pressing OK, the program will pass the values to create an animation struct that stores the change every frame the sphere will have on the different axis and the number of frames it will be animated.
* **Delete animation:** Same behaviour as “Delete spheres”, but it only deletes the animation of the sphere.
* **Background:** 
* **Color:** Opens a dialog box with the behaviour of “Spheres: color”, it changes the color of the background of the image that will be generated and the main dialog window.
* **Help:** 
* **About:** Opens a dialog box with information about the main dialog.
* **Instructions:** Opens a dialog box with a few FAQ of the editor.