**Cs405 Project 3**

**Task 1:**

In draw function, change reasoning is very critical for keeping the scene's hierarchy. That is how we figure out the transformedMvp variable: matrix multiply the MVP that comes with function by current node's local transformation matrix.getTransformationMatrix() on trs. That step will ensure that the global MVP matrix is changed in ways that are unique to current node. Those changes could be translations, rotations or scaling. Other variables linked to transformation transformedModelView, transformedNormals and transformedModel are directly taken from the parent so we don't need to be changed at that level. That means that these matrices can stay same.  
  
Later, The recursive drawing is made via loop that goes to the children of each node in the function. So changed matrices sent to draw function for each child. That recursive method ensures that all descendant nodes are drawn in the html with the right transformations. That allows us to keep scene graph's structure united and let parent changes make change all child nodes.

**Task 2:**

In Task 2, lighting parts are calculated with using the both diffuse and specular effects so that the interaction of light with the surfaces of planets correctly modeled. In order to find the diff variable, take the largest value of the dot product between the (normal) and normalized light direction that is lightdir. That ensures that the diffuse lighting is only positive during light hits the surface directly.

Specular component is calculate via putting together the direction of reflection and the direction of view. Standardized direction toward the camera is shown by the variable trees. That is found by taking the inverse of the fragment position. This is what the reflectionval variable uses to find the direction of the light that is reflected from surface. For spec value, power of the dot product between trees and reflectionval got, then values that are not negative extracted and got pow of power of phongExp. So this calculations gave values for diffuse and specular effects.

**Task 3:**

In task 3, In order to add Mars to solar system, we define new MeshDrawer(); that will be Mars planet. With setTranslation(-6, 0, 0) and setScale(0.35, 0.35, 0.35), we set its translation and scale values. Also, new SceneNode(NewNodeformars, mrsTRS,sunNode) with that child set of sunnode. Moreover, with setTextureImg we pulled the texture for the mars. Also in renderLoop function we set rotation as (0, 0, zRotation \* 1.5) that will rotate z axis 1.5 times the sun’s rotation.

**ScreenShot Of Program:**

Planets in the solar system

Description automatically generated