Samuel Barish
Partners:
Nicholas Forester
Shawn Edmond
Eduardo Gaona

My partners for the final project are Eduardo Gaona, Nicholas Forester, Shawn Edmond. For our project we decided that we are going to go in depth on the topic of Interior Mapping. Basically interior mapping is mapping textures and objects into buildings using a real-time pixel shader. This technique allows for rooms inside of buildings viewed from outside to be rendered into interesting and realistic looking rooms. This uses raycasting in the fragment shader to calculate the positions of floors and walls behind these viewed windows, thus creating a realistic styled window. This prospect could be used for city styled video games like Saints Row, Call of Duty, and many other games. It also adds depth and details to these rooms that can be seen through the windows of generated buildings. These rooms can also be lit, contain furniture, and contain animated characters.

Interior Mapping starts with the creation of ceilings. The ceiling is decided by using the function ceiling(y/d) * d, where d is the distance between ceilings and y is the y position of the pixel that is viewed from the camera tilted upward towards the building. While the camera is tilted downwards this simply changes to (ceiling(y/d)-1) * d. The result of this calculation is the position of the intersection of the ray and the ceiling. It then continues to use the x and z coordinates of the intersection as the uv coordinate plane for the texture. Texture size can also be changed through multiplying by some constant. Walls are added in a very similar way. They use the same calculation that was used for ceiling but instead use the xy plane and the xz plane. The way extra details are added into the room is by using a technique that adds an extra plane that is parallel to the surface of the building. It is intersected with the camera ray and if it is closer than the walls or ceiling then the furniture will show up otherwise it won’t. This is accomplished by using the pixel shader to show them instead of geometry.

I would implement this using GLSL shaders in a very similar manner. The formulas would stay the same and we would only change the constants to get the desired image. We will get the assets off of google poly. The rest will come down to experimentation and reading more in-depth about the smaller details into how interior mapping works.