- C++ is really good for it if we do end up having to
- One possibility is OpenGL
    - Good for various things, modelling and CAD are two of those things so we good
    - Low level rendering, need to put together libraries for higher level stuff
    - Quite a complex system - so hard to learn, but also definitely applicable to what we might want it for
    - **Will definitely be time consuming, no easy way to render complex 3D, lines are hard enough**
    - OpenGL doesn't remember information about where things are drawn, so we would have to implement our own system **(tiiiimmmeeeeee)**
    - If we used this it would doubtless take a very long time, it'd be grounds up from polygons, so might not even work with existing graphics formats idk
- Could also use a higher level 3D system - Scene graph systems
    - Arranges a graphical in a graph/tree structure
    - Relationships are logical / spatial
    - When using a tree parents / children are specifically respected (often manipulation on a parent will apply to trees)
    - There are various versions of this, some are compatible with various lower level graphics systems
- An actual implementation would likely involve using OpenGL to draw individual components, and some sort of scene graph system to render the entire scene
- It would still be time consuming, not as bad as doing the whole thing in OpenGL but still would eat up a large amount of project time
- Similar system seems to take up to a full year with 2 devs - without any of the simulation / control work
- Therefore Making our own 3D environment should only be done if absolutely necessary