Interactive Von Neumann Architecture

Seth Childers, Tyler Scott, Shem Sedrick

<https://tscott8.github.io/VNAness>

**Goal:**

Most people view a computer as a magic black box that pulls up Google, Twitter, and Facebook. The goal of this project is to create a 3D interactive model of the inside of a laptop. This project’s goal is to create a viable teaching tool for exploring and explaining the integral parts of the modern computer.



**Challenges:**

* Creating models and adding the textures
* Loading .obj + .mtl files
* Positioning the parts, then adding them to one full object
* Setting up the orbiting camera, lights and scene
* Expansion of the individual parts
* Controls to handle expansion, reset, zoom, etc.
* Object selection on mouse click
* Displaying descriptions based on the selection
* Highlighting the selected objects texture
* Animating selected objects





**Resources:**

* WebGl – uses web browser to access local graphics hardware acceleration
* Three.js framework – interaction with WebGL
* Dat.GUI framework – controls interface works together with Three.js
* Blender – created/textured 3D models



**How:**

The Three.js framework provides key tools for setting up a 3D scene. We took advantage of key components (cameras, lights, and object loaders) to populate the scene with our 3D models and add the lighting. This framework took care of panning and zooming based on the mouse movements and button presses. However, mouse clicks and highlighting the clicked part were troublesome to correctly implement.

The selecting feature was by far the most challenging problem we solved. It was especially hard to visualize a 2d mouse interacting with 3d space. We solved this problem by shooting (projecting) a ray from the mouse click location, in relation to the camera location, into 3d space. Whatever that ray intersects with, is selected.