//Notes to self: The purpose of this page is in reality not to give credit but to explain how the project was created, please write accordingly.

This projects could not have been completed with the absence of any one of the following software which would be mentioned.

MATLAB – This is the programming language which created the dodecahedron in the first place.

Pixabey – All images taken in Pixabey were black and white and for the sole purpose of engraving them into the dodecahedron. All of them are labelled "free for commercial reuse" and "no attribution required", as seen here.

https://pixabay.com/p-160494/?no\_redirect

MeshLab – A very helpful software for viewing 3D/STL files. The background image particularly was made possible with the help of this software and hundreds of trials of snipping tools...and on top of that the help of a free photo editing tool on the internet called Pixlr. MeshLab also helped with anti-aliasing and smoothing through an idea called Taubin Smoothing.

https://graphics.stanford.edu/courses/cs468-01-fall/Papers/taubin-smoothing.pdf

Three.js - This is a specific JavaScript file available online but is also a JavaScript library/API hosted in GitHub, a very popular website for programmers to share their code. Most of the code in this library, if not all, are MIT Licensed and copyrighted "free of charge...without limitations the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software...". All of the JavaScript programs taken and used in this project are licensed this way. This program can be credited to put all 3D models onto this website, such as loading stl files onto the canvas, displaying the canvas, color schemes, lightings, camera position and angle of display, object rotation (on rotating manually or by entering the birthdate), zooming, displaying a dynamic loading scene etc.

Hammer.js - This is an open-source JavaScript library/API used to detect special patterns specifically in touch screen data by the users (where users can use more than one finger for motions as described below). This project used three of the six possible pattern recognitions: rotation, pinching, and tapping.

\*insert picture\*