CS Capstone Final Report

May 30, 2020

Boxeur - 3D Case Designer

Prepared by

Group 35

Boxeur

Yuxiao Huang

Peng Zhang

Evan Hopper-Moore

Yu Chuan Tey

Drake Evans

Abstract

Boxeur is a simplistic online 3D case designer targeted primarily at users untrained in 3D design software. It allows users to create cases to their specifications for use in 3D printing or laser cutting. This document summarizes the project and the progress we made implementing it. In addition, this document aims to provide the resources necessary to continue work on the project including online resources and code sections.

CONTENTS

1 Foreword

Our project is implemented in PHP, MySQL, JS, HTML, and CSS. The bulk of our web application's features are in the editor part of the site which is coded in Javascript using the three.js library. The features that are left to implement include exporting the 3D model to a .dxf file type, correctly implementing edge types (such as interlocking fingers to allow for the case to open), and adding account options (like delete account or download all projects). Our client also mentioned stretch goals of adding more hole shapes, like USB or Ethernet ports, or a system to upload and download cases on a public board. There are also some slight visual issues in the 3D rendering and overall styling issues that could be fixed for a more streamlined product.

2 Introduction

The TekBots maker space at Oregon State University provides access to 3D printers and laser cutter for students, but many potential users are inexperienced with 3D rendering software and don't have the necessary skills to create enclosures for their projects. Professor Donald Heer, the coordinator of the TekBots maker space, has tasked us with creating a tool that can generate a 3D model that can fit students' specifications to be used for fabrication. The tool will visualize the output as a 3D model that reacts live to the users input. Users will also be able to specify holes in the case to accommodate the needs of their project. Our final product provides greater levels of freedom than currently available tools to support development of student projects at Oregon State University.

Our group has five members and we decided our roles during the first term. Drake Evans handled most of the 3D code, working in js and the three.js library. Evan Hopper-Moore acted as the team leader as well as working on back-end integration in PHP and front-end styling. Yu Chuan Tey worked with Javascript in the editor and styling of the homepage. Yuxiao Huang worked on styling the library page. Peng Zhang worked on front-end integration of the homepage. The client did not directly help with development, but helped shape the features and layout of the editor through responding to our early design documents.

Due to the changes in Spring term we weren't able to catch up weekly on development work in a physical space like we had been doing prior. Along with heavy online class loads for all of us, these changes led to a loss of productivity in our team. This led to our final version of the project missing a few features and our code freeze score of 3.5 out of 4.

This final report collects the documents we wrote describing project and includes resources to point a potential future group in the right direction. The Tech Review document also includes research and recommendations for technologies for implementation. In the Appendix there are sections of our code which highlight the core parts of our web application as well as what features are left to implement.

3 Documents

All the documents pertinent to the development of our project are included below. First the requirements document which describes what the complete product would look like, then the design document which outlines the technologies and techniques used for implementation. Finally, each of our technology reviews are included which has our individual research on possible options for technologies.

Requirements Document

for

Boxeur - Case Designer

Prepared by

Evan Hopper-Moore

Yuxiao Huang

Peng Zhang

Drake Evans

Yu Chuan Tey

Oregon State University

CS 46X - Senior Software Engineer Project

Kirsten Winters

Scott Fairbanks

October 18th, 2019

CONTENTS

1 INTRODUCTION

1.1 Purpose

Many students at OSU seek to take advantage of the equipment that is provided at OSU's makerspaces, such as 3D printers and laser cutters, but don't have the required background knowledge in 3D modeling. Our web based application will provide an easy way for students to generate 3D models of enclosures for their projects that are ready for fabrication with no prior knowledge of 3D CAD software.

1.2 Scope

The purpose of this document is to outline the requirements for the case designer from a technical viewpoint.

1.3 Definitions and Abbreviations

CS	Computer Science
ECE	Electrical and Computer Engineering
OSU	Oregon State University
CAD	Computer-Aided Design
Makerspace	A place in which people with interests in computing or technology can gather to work on projects while sharing ideas, equipment, and knowledge.

2 OVERALL DESCRIPTION

2.1 Product Perspective

The main goal of this project is to create a web based application that lets students define exact measurements for a case which can be exported to be fabricated. This lets students create exactly what they need with no background knowledge of the intricacies of 3D printers and laser cutters and with no experience in 3D auto computer-aided design.

2.2 Product Functions

2.2.1 Production function:

1. Provide a variety of enclosures templates
2. Users can modify the size of the enclosures and other related data.
3. Calculate whether the relevant data is accurate enough

4. Read data and provide solutions based on user input

5. After the production is completed, the output production model

2.2.2 Graphic function:

1. Provide available graphics and record based on user input

2. Graphics can be dragged, zoomed in or out correctly.

3. Record the final model and optimize the solutions.

2.2.3 Website function:

1. User's input and production model will be stored for next use

2. The model is allowed to be downloaded, and the user can also upload the model to complete the design.

3. Provide instructions for use and prompt users how to better complete the design

2.3 User Characteristics

The intended users of this product will be the ECE and CS students involved in the design of enclosures. Considering that many students do not have experience in 3D printing and laser cutting, this application simplifies the design method to the maximum to meet the needs of more students, so the application is equally applicable to students without any design experience. In addition, the application helps anyone who wants to design enclosures, even though its target user is students, which is easy to use.

2.4 Constraints

2.4.1 Program languages:

All programming languages should implement the functionality of a website application based on Html, CSS, and JavaScript. Some functions need help with databases or other programming languages such as JAVA.

2.4.2 Information safety:

The application will ensure the user's personal information security, it does not make the user's information leak, and ensures the user's data security and avoid loss or damage.

2.4.3 Data Storage:

When the user's network or browser encounters an unexpected situation, the application will ensure that their data and design will not be lost. For example, when the user's network is interrupted, the application automatically saves the user's work in advance for the next use.

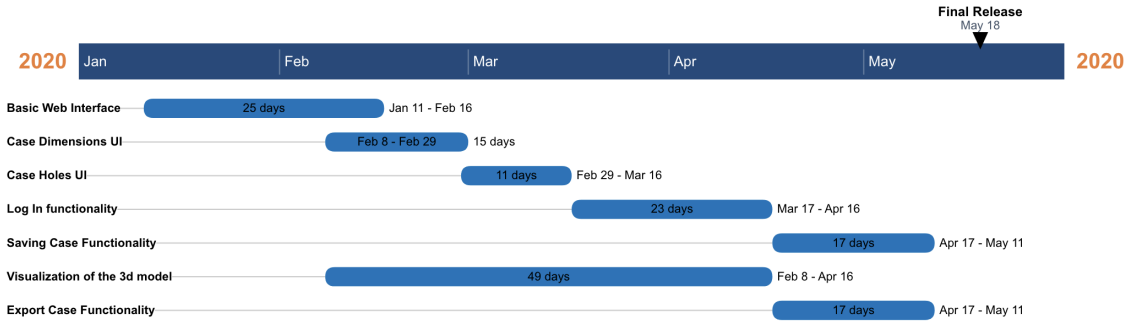
2.4.4 Device Constraint:

The application can adapt to a variety of browsers as well as screen sizes and output files correctly. It should be allowed to run successfully on a variety of computers.

2.5 Assumptions and Dependencies

- We assume the users will speak English
- We assume the users will have experience in using websites

3 Gantt Chart



CS Capstone Design Document

May 29, 2020

Online Case Designer

Prepared by

Group 35

Boxeur

Yuxiao Huang
Peng Zhang
Evan Hopper-Moore
Yu Chuan Tey
Drake Evans

Abstract

This document describes the various techniques, tools and technologies that we have selected to build the online case designer, Boxeur. The technologies described cover the creation of the web app, front end and back end, and the visualization of the case's 3D model. Because this project is not necessarily large in nature, we have placed value in finding lightweight solutions that still offer robust technology to support the core features of Boxeur.

CONTENTS

1 Overview

1.1 Purpose

The purpose of this document is to outline the technologies that will be implemented to achieve the core functionalities of Boxeur.

1.2 Scope

This document covers the technologies behind the front end, back end and 3D visualization of the web app. Each technology choice has an explanation of what it does for our application and how it will be used.

2 Website

As our project at it's core is a design tool, designing a responsive and easy to use interface is crucial for our project. Also because of this, there most likely won't be a large amount of pages on the site, so we prioritized light weight design in selecting the chosen technologies.

2.1 Technology Implemented

2.1.1 InVision

Simply put, InVision is a comprehensive UI design software that supports designing user interfaces for mobile applications and other similar interactive products. Prototyping is a key stage in the design process when creating a user interface for a web site. According to the survey, prototyping is the stage where testing can save time, effort and money. The advantage of the InVision tool is that it is very easy to apply to a prototype because uploading still images of the screen is not time-consuming and prototyping with them. Besides, InVision has good navigation and is very friendly to different types of users.

2.1.2 Sketch

Sketch is a vector graphics editor most commonly used for designing web user interfaces.Sketch's powerful and vibrant community conditions drive a lot of tracking. As a result, the software is constantly updated, plug-ins are added, and most of the content is free. Most users feel that Sketch has a short learning time. In other words, this software is very easy for users to experience the convenience of using this software. Sketch is designed specifically for UI designers, which simplifies the work of designers, shortens the process and saves valuable time. The symbols and preset templates of this software are very creative, and the design experience brought to users is different from other software. The highlight of this software is its resizing function, which can control the stretching, fixing, changing the size and buoyancy of elements. Plugins, which are updated every day, constantly provide designers with fresh concepts, so that designers can always get different creative ideas.

2.1.3 LESS

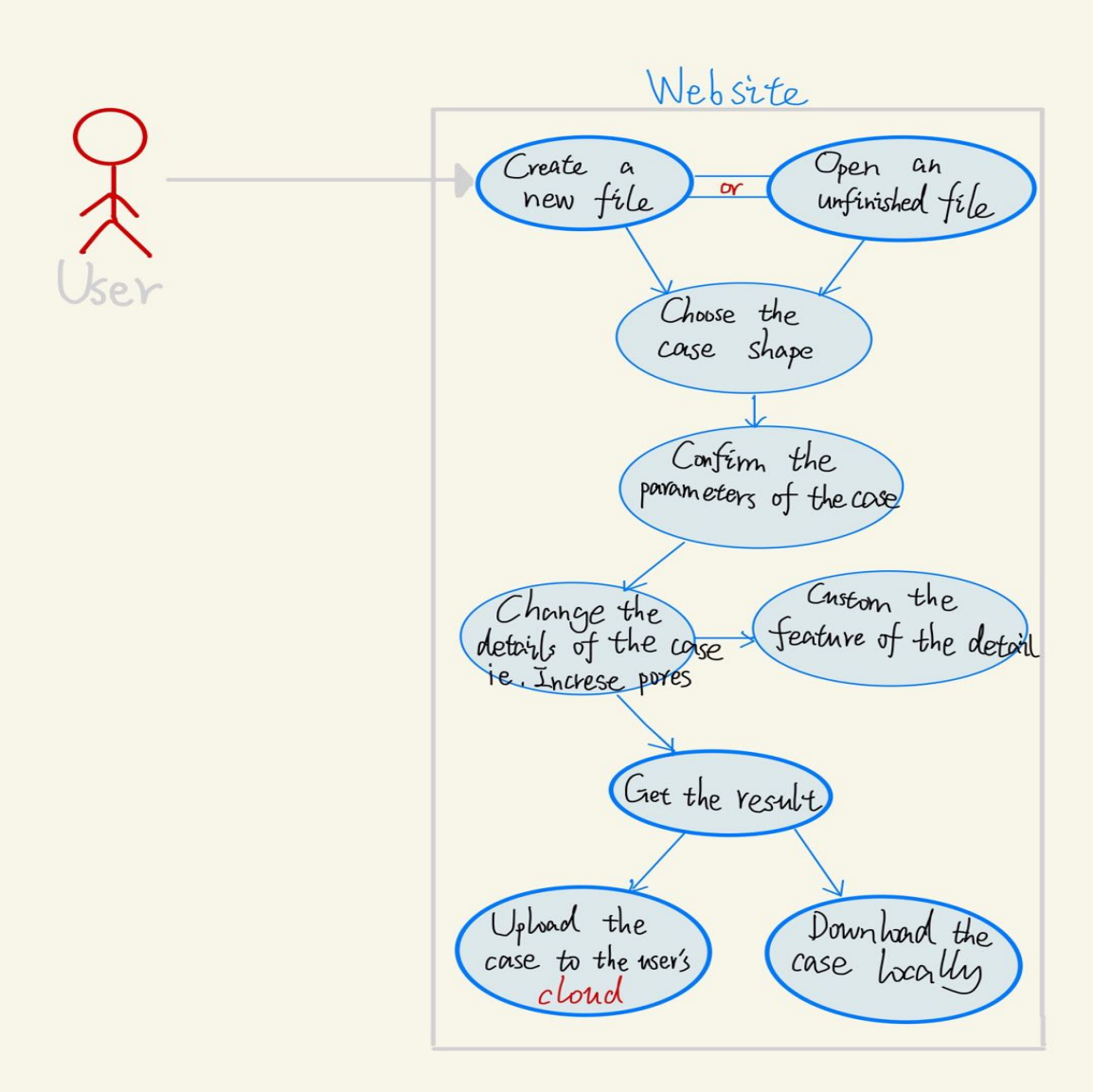
Leaner Style Sheets (LESS) is a backwards compatible language extension for CSS. LESS improves on CSS mainly by adding the capability for nesting. Nesting leads to cleaner and easy to read code as the more standard curly-brace blocks are made possible. LESS also adds the capability for variables, functions, mixins and importing other `.less` files. Variables make it easier to maintain a website, for example a color used for buttons across the site could be changed in

one line without having to search and replace in multiple files. Mixins provide similar functionality but with chunks of code, making it faster to write code that instead would have to be copied in multiple places. Mixins could be especially useful with browser specific properties (ie `-moz-*` and `-webkit-*`). LESS files are compiled into CSS either at run time on the browser using JavaScript or using a Node.js command line tool.

2.1.4 React

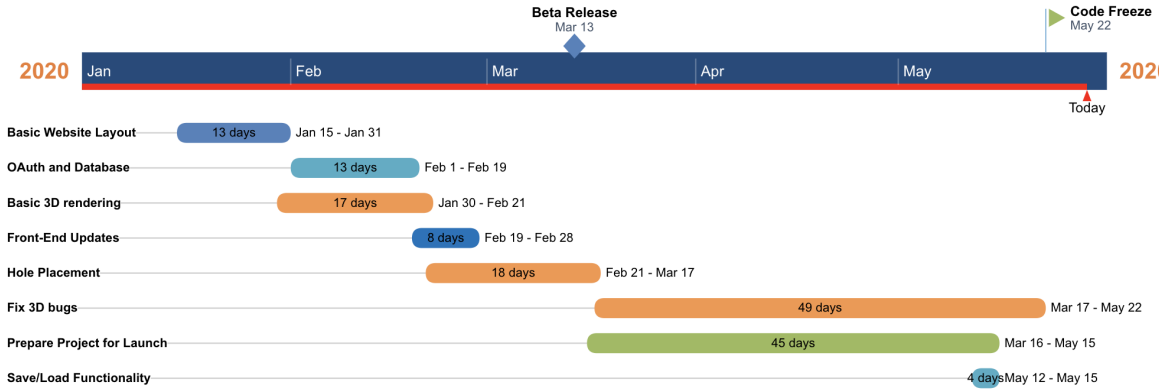
React is a very popular open source front-end framework, which is mainly used for web application development and provides developers with a very simple and fast solution. The basic syntax for React is JSX, which allows developers to mix JavaScript and HTML code in React code. Finally, React will convert JSX to JavaScript to run the code and display on the page. In addition, React improves the overall development efficiency by introducing Virtual DOM to solve complex DOM operations. For the development of WEB applications, React is a very good choice, which will bring more benefits to our development team.

2.2 User Use-Case Chart



User use case chart. This shows that the operation that users can do. Users can create a enclosures by using this tool.

2.3 Development Gantt Chart



3 3D Graphics

One of the core functionalities of the Boxeur app is presenting the three dimensional model of the case, updating as the user changes the specifications. The technology chosen must be able to be changed at run time and preferably will interface well with web languages.

3.1 Technology implemented

3.1.1 WebGL and 3D Model Interaction

The method that we will use to implement an interactive 3D model into the webpage will be WebGL, a javascript implementation of OpenGL. With the toolkit provided by WebGL, we will be able to implement an interactive model that the user will be able to change according to the specifications they give on the first page of the application. The next page of the document will be more interactive, allowing the user to choose a shape for a hole to add the case and the size of the hole. They will then get to choose where on the case the hole will go by interacting with the model itself. As the user moves the mouse around the model, ruler lines will follow the mouse. To move the model, the user will right-click the model and drag the mouse. Clicking will place the hole where the user currently has the mouse. The shape for the hole will have the mouse at its center. The user will also have the option to turn on or off snap measurements, where the shape will snap to specific common measurements (center, edge, or other simple lines of measurements) or to the measurements of previously placed shapes on the case. Once the user has placed a shape, they will have the option to place more. Holes may overlap if the user to chooses to get more complex shapes than the ones provided. If templates are implemented, then the user will also be able to choose saved templates and pre-made templates.

4 Back End Design

The web application must have the functionality to restore previous sessions which will be implemented through a system of logins and storing user data. We will design the architecture for the database in our back end language, Node.js. These technologies will help us design this architecture well and implement it easily.

4.1 Technology Implemented

4.1.1 Node.js

Node.js is a JavaScript runtime for building scalable network applications. Node.js functions similar to a web server and lets developers define exacly how files are served, making it as lightweight as we define it to be. Node.js uses an event-driven model and focuses on controlling processes through callbacks in JavaScript. To handle connections, Node uses events to spawn new processes for connections and follows the defined instructions for serving files, making a simple one-file server possible.

4.1.2 MySQL

The data can be stored and processed using Oregon State University's server storage, and using Mysql in conjunction with HTML, JavaScript to read and call data. Files can be stored and called in CSV format and allow users to manipulate and use the les. For example, users can upload data, download data, and delete data on the website. These operations need to call Mysql, get from the database, and store the data. We will connect the HTML to Mysql so that the user can directly retrieve the target le. In addition, this allows each user to sign up as a member and view recently available les for easier and faster design.

4.1.3 REST API

The Representational State Transfer(REST) API is an architectural style that defines how data is sent and the interaction between the server-side and the client-side. A RESTful design for an API is based on using Uniform Resource Identifiers (URI) and the JSON data type to efficiently transfer data. JSON is a highly browser compatible data type which can be interpreted easily by JavaScript. REST APIs focus on simplicity by using the standard HTML protocols and keeping the interaction stateless, meaning all of the information required to get or send data is included in the request. REST API defines other principles such as having chacheable data and a hierarchical structure which can increase speed and security, respectively. Using REST API as a guideline to designing our architecture in Node.js will lead to a clean and scalable application.

Tech Review

for

Boxeur - Case Designer

Prepared by

Evan Hopper-Moore

Oregon State University

CS 46X - Senior Software Engineer Project

Kirsten Winters

Scott Fairbanks

November 8th, 2019

Abstract

The purpose of this document is to outline some options for solutions for the web interface of Boxeur, a tool for students to create 3D enclosures. The options investigated include for the choice of styling language, architectural style, and web framework to be used. Along with a short analysis and comparison of options, recommendations for final choices of technologies are included to be considered by the Boxeur team.

CONTENTS

1 INTRODUCTION

Boxeur is a web based tool for designing 3D enclosures for fabrication through 3D printing and laser cutting. The website's interface will be simplistic in design, only spanning a few pages for creation of cases, logging in, and recovering previously saved sessions. Because this project is not necessarily large in nature, I have placed value in finding lightweight solutions that still offer robust technology to support the required features.

2 TECHNOLOGIES

2.1 Styling Language

2.1.1 Description

CSS is the basis for almost all website styling and is a core element of web design. There are a few alternatives that offer extra features on top of CSS that could result in an easier process for our group. These extension languages could be worth using to keep our project concise, readable, and scalable.

2.1.2 Options

1) CSS
2) LESS
3) SCSS

2.1.3 CSS

Cascade Styling Sheets (CSS) is the basis for styling in web development and describes how HTML elements and content should look and behave [1]. CSS solved the issue of inline styling in HTML which was not scalabale as it was hard to manage on larger sites. All browsers support CSS as it is now the standard language for styling HTML. It's impossible to escape using CSS in modern day web development but there are some language extensions that add more features on top of and compile down to CSS.

2.1.4 LESS

Leaner Style Sheets (LESS) is a ”backwards compatible language extension for CSS” [2]. LESS improves on CSS mainly by adding the capability for nesting. Nesting leads to cleaner and easy to read code as the more standard curly-brace blocks are made possible. LESS also adds the capability for variables, functions, mixins and importing other `.less` files. Variables make it easier to maintain a website, for

example a color used for buttons across the site could be changed in one line without having to search and replace in multiple files. Mixins provide similar functionality but with chunks of code, making it faster to write code that instead would have to be copied in multiple places. Mixins could be especially useful with browser specific properties (ie `-moz-*` and `-webkit-*`). LESS files are compiled into CSS either at run time on the browser using JavaScript or using a Node.js command line tool. Depending on what framework we choose for the site we could use either method.

2.1.5 SCSS

Sassy Cascade Styling Sheets (SCSS), also known as Sass, is "the most mature, stable, and powerful professional grade CSS extension language in the world" [5]. It is based on Ruby and compiles down to CSS, but unlike LESS it cannot be compiled in the browser. SCSS provides the same features of LESS such as variables, nesting, mixins, and functions while also supporting more functionality such as loops and control directives. Control directives in SCSS add functionality for the traditional statements `if/else`, `for`, `each`, and `while` which can also increase scalability and readability in our stylesheets.

2.1.6 Comparison

CSS is the basis for styling HTML across the web and is supported in all browsers. However, CSS can be improved upon by using CSS extension languages such as LESS and SCSS which compile down to CSS either in the browser or manually on the command line. The differences between LESS and SCSS are slight but it seems like SCSS is more popular, especially as it was just adopted as the base for Bootstrap 4 [4]. Both languages offer similar functionality such as nesting, variables, mixins, and functions but SCSS adds a few more features such as control directives. Even though LESS has less functionality than SCSS, it can be compiled using a JavaScript file included in the in HTML. This also means that when writing in LESS, every time the page is reloaded the most current styling is used as opposed to SCSS which could be out of date and could lead to confusion with team members.

2.1.7 Recommendation

CSS is certainly capable of doing anything we would need for our project but the added benefits of LESS or SCSS would make our project easier to read and write. SCSS boasts more features than LESS, however it requires the extra work of installing Ruby in our project. The language I recommend we use is LESS because it has less overhead than SCSS as it simply requires including a JavaScript script, but still supports the main features of nesting, mixins and more.

2.2 Architectural Style

2.2.1 Description

Throughout the implementation of our project we will have to deal with sending data back to the server side of our application, for example when implementing a log-in feature or fetching case data for restoring sessions. There are a few different popular methods to design our API and how we retrieve and send data to the server.

2.2.2 Options

1) REST API
2) SOAP

2.2.3 REST API

The Representational State Transfer(REST) API is an architectural style that defines how data is sent and the interaction between the server-side and the client-side [6]. A RESTful design for an API is based on using Uniform Resource Identifiers (URI) and the JSON data type to efficiently transfer data. JSON is a highly browser compatible data type which can be interpreted easily by JavaScript. REST APIs focus on simplicity by using the standard HTML protocols and keeping the interaction stateless, meaning all of the information required to get or send data is included in the request. REST API defines other principles such as having chacheable data and a hierarchical structure which can increase speed and security, respectively.

2.2.4 SOAP

Simple Object Access Protocol (SOAP) is a standards-based Web services access protocol which relies on XML to transfer messages. Developed by Microsoft, it fixed a lot of problems that developers had transferring data on the internet in its early days and became fairly standard in API designs. The SOAP envelope is a definition for how XML data is organized. The envelope contains headers, which can contain information such as credentials or definitions of the data type in the message, and a body block which contains the actual message. Using this structure, SOAP also provides built in security features through its standard called WS-Security, ideal for enterprise level usage.

2.2.5 Comparison

Because of SOAPs design structure, naturally it requires more overhead to support the XML envelope's headers and security protocols while the REST API values a slimmer design. This translates to using

more bandwidth when using SOAP. REST API's architecture also supports caching and is widely known for excellent performance and scalability.

2.2.6 Recommendation

My recommendation between the two approaches for sending and receiving data is the RESTful API structure. Our project most likely won't need the extra security provided by SOAP, so it seems the lightweight choice of REST is right for us. We can use the guidelines in the REST API description on their website to keep our architecture clean and efficient while minimizing bandwidth usage.

2.3 Framework Choice

2.3.1 Description

Web Frameworks are APIs or software designed to support the deployment and creation of web services. For our project we need a framework that can support rendering dynamic content from databases for planned functionality such as restoring sessions, pulling user created case models, and publishing new models for sharing.

2.3.2 Options

1) Django
2) Node.js

2.3.3 Django

Django is a "high-level Python Web framework that encourages rapid development and clean, pragmatic design" [3]. Django follows a design pattern called Model Template View (MTV) that acts as a guideline for clean design. The models act as the data access layer and handle validating, interacting with, and relating data. The template layer, also referred to as the presentation layer, handles what is displayed on the page. The view layer is also called the business layer as it accesses models and populates templates, acting as the connection between the two layers. The MTV layout guides projects into a clean design with well separated layers. This high level design can also limit us in a way because we have less control over how we design our application.

2.3.4 Node.js

Node.js is a "JavaScript runtime [for building] scalable network applications" [7]. Node.js functions similar to a web server and lets developers define exacly how files are served, making it as lightweight

as the developers define it to be. Node.js uses an event-driven model and focuses on controlling processes through callbacks in JavaScript. To handle connections, Node uses events to spawn new processes for connections and follows the defined instructions for serving files, making a simple one-file server possible.

2.3.5 Comparison

Django has a fair amount of overhead and setup required compared to Node.js, however, it does automatically support a MTV design pattern. In Node.js we would have to set up our own database access design and populating templates with content could be a hassle. Node.js, however, is integrated closely with JavaScript which makes accessing and changing HTML elements easy.

2.3.6 Recommendation

Because Node.js uses lower level design compared to Django, we are given the choice of just how robust the server will be. There is very little overhead involved in starting a project in Node.js and for Boxeur, which only has a handful of pages, I believe the correct framework choice is Node.js.

REFERENCES

[1] w3schools.com. ”CSS Introduction.” W3Schools. https://www.w3schools.com/css/css_intro.asp

[2] The Core Less Team. ”Overview.” Getting Started - Less.js, http://lesscss.org/.

[3] Django Software Foundation. ”The Web Framework for Perfectionists with Deadlines.” Django, https://www.djangoproject.com/.

[4] Otto, Mark, and Jacob Thornton. ”Introduction.” Bootstrap, https://getbootstrap.com/docs/4.0/getting-started/introduction/.

[5] Catlin, Hampton, et al. “CSS with Superpowers.” Sass, https://sass-lang.com/.

[6] Fielding, Roy Thomas. ”REST API Tutorial.” Learn REST, https://restfulapi.net/.

[7] Node.js Foundation. “About Node.js.” Node.js, https://nodejs.org/en/about/.

Technology Review

Peng Zhang

November 7, 2019

Team 35: Boxcur

CS 461 - Fall 2019

Abstract

This article will provide a detailed analysis of the technical plan and implementation. When the user implements interaction with our project, these technologies implement data storage and recall to facilitate user upload and download operations. In addition, these technologies add effectiveness and usability to the user experience, and we need to ensure the correct output of the 3D model. This document consists of three parts, UI visualization and usability, Data storage and processing, and User interface design and organization.

1 Introduction

My role in our team is to organize the UI and implement user interaction. Online Case Designer is a tool that can generate a 3D model that can fit students' specifications to be used for fabrication. The tool will visualize the output as a 3D model that reacts live to the users input. We are going to implement a website application that can help students complete the design of enclosures without restrictions. Our ultimate goal is a simplistic and suitable application for most students, although they don't have the design experience or access to 3D printers and laser cutters, we can help them. In addition, Users will also be able to specify holes in the case to accommodate the needs of their project. Our final product will provide greater levels of freedom than currently available tools to support development of student projects at Oregon State University.

2 Piece 1: UI visualization and usability

In the process of user interaction, the project needs to ensure availability. Each button can perform its intended functions, such as uploading and downloading files, which requires consideration of the type and format of the file, which may use JavaScript to implement reading and downloading functions and provide constraints. The project will correctly generate 3D models to ensure the visualization of the project. This may include the following technologies: HTML5 (Tool: JSFiddle), WebGL in JavaScript, Algorithm in C++.

2.1 HTML 5

HTML5 is the latest version of the website language, which is suitable for large multi-function calls and supports user interaction. Our project is a web application, and HTML5 is the primary method of user interaction. For example, the normal display of all functions, including the generation of 3D models, requires the help of HTML5. In addition, the use of some APIs also requires HTML5 support, like WebGL to generate 3D models. JSFiddle is a HTML5 design tool that can be compiled and helps us develop and improve web page layout, we can try to use HTML code on this.

2.2 WebGL:Web Graphics Library

WebGL is an API in JavaScript that helps us implement interactive 3D models. WebGL will be used in conjunction with JavaScript, and the visualization of the project will be improved and invoked in JavaScript. For example, we need to make the 3D model arbitrarily dragged and zoomed in and out. This may use the function jQuery to get the user's actions and improve the output of the image according to the user's preferences.

2.3 Algorithm in C++

Algorithm is a library in C++ that can help project optimization algorithms to help users get more accurate data, such as the length and width of the graphics, volume, area, and more. The Algorithm library can integrate the data provided by the user to get the coordinates and position of the graphic, so as to correctly generate the 3D graphics. And the Algorithm library can calculate the availability of data, so that the user's graphic design is best improved. A good example is that the Algorithm library can treat user input as visual data and analyze them to generate a suitable coordinate system to display a 3D model or provide an effective solution based on user input.

3 Piece 2: Data storage and processing

The reading and processing of data can realize the main functions of user interaction, such as user-designed uploading and downloading. This helps the user's data not be lost and guarantees that they can continue to work after interrupting the design. In addition, this project can process the user's data to facilitate the accuracy of the data and the correct generation of the graphics. The main technologies that may be used are Mysql, PHP, Struct in C/C++. In addition, we may use html2canvas.js to implement a user's web page capture to get a quick overview of the design.

3.1 MySql server to store data

The data can be stored and processed using Oregon State University's server storage, and using Mysql in conjunction with HTML, JavaScript to read and call data. Files can be stored and called in CSV format and allow users to manipulate and use the files. For example, users can upload data, download data, and delete data on the website. These operations need to call Mysql, get from the database, and store the data. We will connect

the HTML to Mysql so that the user can directly retrieve the target file. In addition, this allows each user to sign up as a member and view recently available files for easier and faster design.

3.2 PHP

PHP is the current popular WEB development language, which can be used in conjunction with HTML and Mysql to ensure data storage and processing. We need PHP to do multi-file processing, such as user uploading more than three files, which requires PHP and JavaScript to generate a queue and layered processing, and finally use Mysql storage. In addition, PHP and Mysql, HTML calls can provide visibility into the reading of the target file to increase the processing efficiency of the application. The main goal of our project is to generate 3D design models, and PHP applications can effectively help the rendering of WEB programs.

3.3 Struct of C/C++

A struct in C/C++ programs are extremely helpful for reading and processing data. For example, when a user uploads graphical data information, we need a C/C++ program to help analyze the usability of the data and generate and represent them graphically. The program can create a Struct and analyze each data as a Struct. The final result will be a 3D model in HTML and output as a coordinate. These outputs can be read and implemented by PHP and JavaScript, and combined with the use of WebGL to generate 3D models, increasing the effectiveness and usability of the project.

4 Piece 3: User interface design and organization

The user interaction design will be designed as a prototype and concept map. This project will provide visual targets for the students, and we will ensure the usability of the design and systematic design with a simple and easy to understand design. A good user interface can enhance the user experience. We want every button and application to be easier to use, and they should be able to learn quickly when they are used for the first time. we will use the following technologies: Balsamiq cloud, CSS: Cascading Style Sheets and Mockplus.

4.1 Balsamiq cloud

Balsamiq is a web application focused on user interaction design that will help us design and implement designs more easily. When we create a new project on Balsamiq, we can clearly build computer screens, create menus, buttons, and various web features. In addition, we can add a link to each function to jump to the page that should be. This is easy to modify, and we can use this as a prototype to let the target users interact and improve the final solution based on their feedback.

4.2 CSS: Cascading Style Sheets

CSS is the main language for designing user interaction interfaces. It can edit the display of HTML and modify font, spacing, size, position, etc. It must be used in conjunction with HTML. For example, we can modify the font color to maintain design consistency, or modify the position of the button/graphic to visualize the user interface and make it easier to learn. In addition, the use of CSS can see the changes of the page more directly, making our design easy to modify. We will focus on typography and formatting to make all features easier to find, all designs support user interaction.

4.3 Mockplus

Mockplus is also a user interaction design application that enables the design of application prototypes. It provides a large number of icons and graphics, and allows the user to design the UI interface in a drag-and-drop format without the need for code participation. This makes it easy for designers without any experience to complete the design. In addition, it allows users to collaborate on design, and users can invite team members to complete the design. This is a great help for user interaction design.

References

[1] “Balsamiq cloud.” [Online]. Available: https://balsamiq.com/

[2] “Mockplus.” [Online]. Available: https://idoc.mockplus.com/

[3] “Jsfiddle.” [Online]. Available: https://jsfiddle.net/

Team 35 - Online Case Design

Name: Yu Chuan Tey
Class: CS 461
Term: Fall
Tech Review

***Abstract*—The basic part of this Online Case Design is to create a webpage to design some 3D object and to send to companies for laser cut. User can design some object they want with some tools provided such as some calculation of the width, Height, Depth. Furthermore, in the webpage also provided an image that the user input from the information that provided. The purpose of this Online Case Design is to help user easy to use the webpage and to design something they want. After user finish design, then they are able to download and save the object that they design.**

I. Introduction & accomplish

This is an Online Case Design and the purpose of this project is to help user easy to design an object and to create themselves. After design they create, then they can download it and send the file to some company to laser cut. Some of the knowledge require high skills, for example, some of the material require specific cuts, or some design require some very specific measurement. Which mean that through this Online Case Design, will be a lot easier for user to use it and does not require high skills.

The minimum requirement for this Online Case Design is to have a webpage with width, height, depth for measurement the size of the box. After that we also have to create the material thickness and what kind of material. Furthermore, the other requirement is to have edge join such as flat, finger and T-slot join. Lastly, also have a 3D image after input all the requirement and user can have some knowledge about what the object is. Some expectation for this Online Case Design from our client is to create some design such as making some hole between the object or some other useful design.

One of the things our project trying to accomplish is to make a full website that let non-high skills people to build a container on our website with some specific design that require high skills. User can use their design for laser cut.

II. Pieces & responsible

1. The first piece to accomplish our project, I think the website is very important. This is because user will go to internet and design using our website. Therefore, having a clean and easy use website will be one of the important things to accomplish. In addition, to accomplish our project, we have to let the website to be very easy to use. For example, we need to have some basic container and then user can add some additional design according to what they want. Furthermore, the website will also need an 3D image to let the user know what their design look like.

   To build a website, it requires knowledge of HTLM, CSS, JavaScript, and other languages. We need JavaScript or other language to have user interface. This is because when user want to insert their design, and JavaScript is one of the easy languages to use.

   CSS
   Pros : the advantage of using CSS is very easy to use and easy to understand. In addition, CSS support most of the browser and which mean that don't need to worry about some browser couldn't work.[1]

   Cons: one of the main problems is that HTML need to take longer time to load CSS, which mean that if you have slower internet, then you need more time for that.[1]

2. The other piece to accomplish our project is database. Database is one of the important for this project. This is because user would have to save their design, and this will need database for it.

   To build a database, it requires knowledge of MySQL, PHP or other languages. MySQL is one of the easy languages to use for making database and easy to connect with the webpage. On the other hand, there is other database like MongoDB, NoSQL. This tools also help user to create database easily.

   MySQL
   Pros: one of the advantage of MySQL is that it is design with the focus on the Web.[2] Which mean that mySQL is one of the best for doing website.

   Cons: MySQL is not as mature as another relational database management.[2] what this mean is that it will be complicated to use that other relational database management.

   MongoDB
   Pros: one of the advantages of using MongoDB is that faster to turn around in development.[3] In addition,

MongoDB also flexibility to find out schema design when we change the design or our mind.[3]

Cons: one of the disadvantages is that MongoDB nesting is not always possible. What this mean is that sometime there is no foreign key constraints to enforce consistency [3].

III. Conclusion

In conclusion, this Online Case Design will help a lot of user to create 3D object without require high skills. Furthermore, this project also includes some simple tools that other case maker does not have it. Therefore, this will be a simple and easy to use case maker webpage. In addition, there is some piece that we need to accomplish is making website and creating a database for user. There is a lot of technology for us to use, such as html, CSS, JavaScript, MongoDB, MySQL. All these tools have their advantages and disadvantages. To be success to our project, we have to decided which tools is the best for our project and have to learn how to use all of the tools.

Reference

[1] The Pros and Cons of CSS. Stanford-Brown Blogs. https://www.sanfordbrown.edu/Student-Life/blog/January-2015/Pros-and-Cons-of-CSS

[2] The Pros and Cons of MySQL, SMARTFILE. https://www.smartfile.com/blog/the-pros-and-cons-of-mysql/

[3] MongoDB Pros and Cons. https://www.slideshare.net/johnrjenson/mongodb-pros-and-cons

CS CAPSTONE TECH REVIEW

NOVEMBER 8, 2019

ONLINE CASE DESIGNER

PREPARED BY

GROUP 35

BOXEUR

YUXIAO HUANG

Abstract

The whole article describes many technical details about how to complete the team's project. First, learn how to improve our product by using some existing 3D modeling software, and then quickly get the design prototype of our product by using UI design software like InVision.In the end, the paper discusses how to grab useful network data through some computer languages. This paper is divided into four parts: introduction, interaction modes, tools for generating the UI and data generation and capture. Through the analysis of my positioning in the team, the role of the article is to help me better complete the contribution in the team.

CONTENTS

1 Introduction

In this team, I was responsible for adjusting the UI design and user interaction content of the whole project. The main function of our product is to provide customers with satisfactory 3D models, so we pay special attention to the UI design of the product. Beautiful UI design and interaction will enhance the user's desire to use our products. Our ultimate goal is to design production tools that can provide detailed parameters for all types of users, and our products will save users as much time as possible in the process of using the products. Similarly, to make a correct model accurately, our project also attaches great importance to data capture. Improving the accuracy and speed of data capture is the way to improve the user experience.

2 Piece 1: Interaction modes

2.1 3D Slash

Our project is an operation service based on 3D modeling, so everyone, including users, needs to be familiar with the simple operation process of 3D modeling.3D Slash proves that 3D modeling is not only fun but also very productive.3D modeling can be a difficult process for people who are not in the design of the computer field, but 3D Slash will soon enable users to adapt to both online and offline use. Many interesting animations are designed in this software to attract more users, so this software is a good educational software to attract ordinary users' interest in 3D modeling. And 3D Slash has also released a virtual reality mode that allows users to view their designs on VR devices. This technology can more truly feel the fruits of the creator's labor. Also, the user can use the photos as the basis of the model and add layers as they are added, eventually converting the initial flat image into a 3d one. This makes the whole software seem easier to use.

2.2 Tinkercad

Thinkercad is a simple online 3D design tool for the general public. The conceptual design of this tool is very similar to our project. Tinkercad allows both professional and amateur designers to create 3D models of toys, prototypes, jewelry, etc. It is worth mentioning that this is a free, good user interface and a very short learning time to use the modeling tool. Although this software has never been listed as a professional application, it can play an important role in simple primary use. And for most first-time users, learning from the work of others is the best way. Tinkercad helps users share their work or access models and examples Shared by other volunteers. The biggest similarity between this tool and our project is that we are both online tools, so the portability of the two tools is much better than other modeling software. Although it is an online tool, the functionality of the tool is so rich that most beginning users will choose it as the learning software for learning to model.

2.3 Vectary

The function of Vectary is more complete than the above two, so it has a free trial plan, but after the planned time is up, it still needs the user to decide whether to continue to spend money to use it. Vectary has a lot of learning resources attached. Users can access these video resources for free to learn how to use the online tool more efficiently. It also provides interactive help so that the user knows exactly what it does each time. Users can save some steps in the design process by using one-click buttons to generate shapes or objects. Because it is an online tool, it can easily send files by

sharing a URL. Also as a sharing function, Vectary invites other members of the group to view and edit the current design project together. Finally, rendering is also integrated into the 3D tools, so users do not need to re-render the 3D model using other software.

3 Piece 2: Tools of generating the UI

3.1 Javascript

Javascript is an object-based and event-driven client-side scripting language with relative security. It is also a scripting language widely used for client-side Web development and is often used to add dynamic functions to HTML pages, such as responding to various user operations. Like other scripting languages, JavaScript is an interpreted language that provides a very convenient development process. To facilitate user operations, the language is interpreted line by line during program execution.JavaScript makes it easy for us to manipulate HTML objects and supports distributed computing. But the downside of the language is that JavaScript support varies from browser to browser. In other words, sometimes there will be a certain gap in the display effect, or even sometimes it will not show.

3.2 InVision

Simply put, InVision is a comprehensive UI design software that supports designing user interfaces for mobile applications and other similar interactive products. Prototyping is a key stage in the design process when creating a user interface for a web site. According to the survey, prototyping is the stage where testing can save time, effort and money. The advantage of the InVision tool is that it is very easy to apply to a prototype because uploading still images of the screen is not time-consuming and prototyping with them. Besides, InVision has good navigation and is very friendly to different types of users.

3.3 Sketch

Sketch's powerful and vibrant community conditions drive a lot of tracking. As a result, the software is constantly updated, plug-ins are added, and most of the content is free. Most users feel that Sketch has a short learning time. In other words, this software is very easy for users to experience the convenience of using this software. The sketch is designed specifically for UI designers, which simplifies the work of designers, shortens the process and saves valuable time. The symbols and preset templates of this software are very creative, and the design experience brought to users is different from other software. The highlight of this software is its resizing function, which can control the stretching, fixing, changing the size and buoyancy of elements. Plugins, which are updated every day, constantly provide designers with fresh concepts, so that designers can always get different creative ideas.

4 Piece 3: Data generation and capture

4.1 Python

Python is very suitable for the development of web crawler programming language, provides such as urllib, re, JSON, pyquery and other modules, but also has a lot of forming framework, such as Scrapy framework, PySpider crawler system, itself is very simple and convenient, so it is the first choice of web crawler programming language. Python's

interface to crawl web documents is cleaner; Compared to other dynamic scripting languages, Python's urllib2 package provides a more complete API for accessing web documents. Python has other advantages. It integrates well with most cloud and platform as a service provider.It brings unique advantages in ensuring large-scale performance in data science and machine learning when supporting parallel computing for multiple processes. You can also extend Python with modules written in C/C ++.

4.2 DocuPhase

To be able to extract data from different sites for comparison, and to use natural search results data for a more comprehensive analysis of other similar products.DocuPhase products reformat data extracted from scanned or digital documents into searchable and editable text using optical character recognition (OCR) techniques. In other words, the product can extract text-based information from digital images. The product's automation platform USES automated OCR to transform documents of any format and complexity into data available for the business.DocuPhase's advanced capture and identification tools make it easy to put documents and data into a user's document repository. Whether the file is emailed or entered digitally on the platform, it can be processed quickly and made available to people with access rights. The DocuPhase greatly simplifies the task of extracting and entering data, saving users a lot of effort when they need to automate data collection and synchronization.

4.3 Java

The Java language's syntax is very close to the C and C++ languages, making it easy for most programmers to learn and use.Java, on the other hand, has discarded the little-used, hard-to-understand, and confusing features of C++, such as operator overloading, multiple inheritance, and automatic casts. In particular, the Java language does not use Pointers, but references. It also provides automatic waste collection so that programmers don't have to worry about memory management.Java is commonly used in network environments, for which it provides a security mechanism against malicious code. In addition to the many security features of the Java language, Java has a security mechanism for classes downloaded over the network. One of the design goals of the Java language is to adapt to a dynamically changing environment. Classes needed by Java programs can be loaded dynamically into the runtime environment or over the network. This is also good for software upgrades. Besides, classes in Java have a runtime representation that allows runtime type checking. For exception handling, Java is also divided into checking exceptions, runtime exceptions, and errors. Through exception handling, the accuracy and speed of data can be increased more efficiently in the process of data capture.

Drake Evans
CS 46X
Fall Term
Group 35 Case Designer

Abstract
The TekBots makerspace at Oregon State University provides access to 3D printers and laser cutter for students, but many potential users are inexperienced with 3D rendering software and don't have the necessary skills to create enclosures for their projects. Online Case Designer is a tool that can generate a 3D model that can fit students' specifications to be used for fabrication. The tool will visualize the output as a 3D model that reacts to the users input. Users will also be able to specify holes in the case to accommodate the needs of their project.Successful completion of the project is a working and simple web application that can design basic cases with holes for whatever ports, cables, etc. Our final product will provide greater levels of freedom than currently available tools to support development of student projects at Oregon State University.

Role:

My primary role in the project shall be graphics programming/display and ui related to the 3d graphics that the client wants. I will also be aiding in other aspects of the project as needed.

Responsibilities:

I will primarily be responsible for the 3d graphics aspects of the projects. This is a key component of the project and must be executed correctly for the project to succeed. The 3d graphics portion of the project has 2 primary things that must be accomplished: choosing the best method of 3d graphics display for the project and proper backend coding methods. Good web interface for the 3d graphics is a secondary task, and one that will likely receive a lot of aid from the web design side of the group.

Methods for Graphics Display:

There are two methods of display that will be examined: WebGL and Sketchfab. WebGL is an application of OpenGL for use on websites. Sketchfab is a platform that allows users to publish, shop for, and share/embed 3d models.

Sketchfab is an online platform that user can register on to publish 3d models you've made as well as find other models that other users have published. The website also allows you to embed the models into webpages, which is why it's of possible interest. The website also allows the purchase of 3d models to be used however the purchaser pleases, as they are royalty free. Use of free content is subject to whatever creative commons license the uploading user applies to it. [1]

WebGL is a library that is used in javascript web programming to create 3d models in webpage's. It is more akin to OpenGL than Sketchfab, whereas Sketchfab is just a sharing platform that has a proprietary model viewer that can be embedded into webpage's. WebGL allows us to code in an interactive model that we can mold to our needs. It only requires us to learn WebGL code as opposed to designing a model in another program. As we only require a simple shape, this will be simple. WebGL is a simple yet powerful tool to have, and it will allow us to easily achieve our goals. It also is supported by all modern browsers and doesn't require any specialized programs from the user [2].

The choice between these two of which to use is simple unfortunately, as Sketchfab does not allow for one of the criteria of the project to be realized, user modification. WebGL also lines up better for accessibility, ease of use, and likely the coding language we will be using for the web interface, javascript. The choice for this project will have to be WebGL.

Coding Methods:

WebGL has plenty of coding tools for displaying 3d graphics, as well as an extensive documentation that will allow us to achieve a very strong code base for this project. As we only need to start with drawing a box, that is an easy task. The library utilizes javascript, so it can be easily integrated into the website. Utilizing various libraries in conjunction with WebGL, an interactive model should be implemented to the specification of the client. One website demonstrates the capabilities of WebGL really well, showing how WebGL can create a very competent interactive 3d display [3]. One of the greatest strengths of WebGL is its compatibility with most modern web browsers without the need for any form of plugins to be installed, allowing for increased accessibility.

Starting out with adding the spot for the model, the "Canvas" element of HTML will allow us to add a spot for our 3D model. Following that, we can enable the webgl context for the canvas element by passing the string, "webgl" to the getContext method, and then the rest of the coding comes from the WebGL tools. The tools follow a similar organization to the standard C++ OpenGL interface, making them easier for me to understand, as I have worked with OpenGL. The procedure starts with setting a lot of stuff up, such as shaders by setting up two sets of shader programs, one for the vertex shader and one for the fragment shader. The shaders don't need to be very complex, given the nature of the task we wish to achieve. Next, we set up the objects we wish to use (the cube) and the viewing options for the scene such as viewing type

(perspective or orthographic, likely orthographic for easier visualization of measurements), viewing angle (where we're looking), depth buffer (how far away from the screen is the center of the scene), colors, viewing window (the size of the canvas element), etc. Then, we draw the objects and animate them if necessary. As the user needs to be able to interact with the model, animation of the scene will come mostly from mouse click and drags on the canvas element of the webpage. The user will need to be able to modify the model, so changing options for the parameters of the object will be accomplished through the use of drop down menus and data entry fields that modify variables in the javascript, such as size and material thickness.

The second page for adding holes will be tricky, and will utilize an almost entirely different set of graphics coding, as the interaction with the model will be different. Users will need to be able to move the cube around at will, as well as be able to visualize the holes they wish to add. Visualization of the holes will be a simple css toggle, but the snap measurements will be slightly tricky. Measurements will need to be kept track of with variables that change with the mouse position, as well as checks against the current hole measurements and coordinates. Then, checks for which measurement the mouse cursor is nearest will happen. Clicking on a location will add a hole to a list of holes with all of the measurements needed for machining.

3D Interactive UI:

There will be a lot of collaboration on this element of the 3D display, in order to achieve the desired look and feel of how the user interface functions. This interface will come mostly out of HTML and CSS coding, likely utilizing drop down menus and entry fields to offer the user options to modify the shape and other facets of the model. The following page will be simple, the same canvas element along with a new set of options. As holes are placed, they will be added to a list of holes under the options on the side of the screen, and will be highlighted if the user clicks on a hole. Users will also be able to remove holes. There's not much to discuss in terms of technical review for these items, as they are mostly discussed in the other reviews for web development solutions.

Bibliography:

[1] Gamedev, D. Cao, J. van Aelst, and Oleaf, “Publish & find 3D models online,” *Sketchfab*. [Online]. Available: https://sketchfab.com/. [Accessed: 04-Nov-2019].

[2] “WebGL: 2D and 3D graphics for the web,” *MDN Web Docs*. [Online]. Available: https://developer.mozilla.org/en-US/docs/Web/API/WebGL_API. [Accessed: 04-Nov-2019].

[3] C. Delmon, “WebGL, the wow effect,” *JoliCode*, 16-Apr-2018. [Online]. Available: https://jolicode.com/blog/webgl-the-wow-effect. [Accessed: 04-Nov-2019].

4 Blog Posts

4.1 Fall

4.1.1 Evan Hopper-Moore

Week	Progress	Problems	Plans
1	Individually we worked on our resumes and searched for projects on the project portal.	There were many projects available on the portal.	We all submitted our top ten choices for projects and got assigned to the same project.
2	We wrote the Team Standards and got to know each other from talking online and after class.	We needed more information for the Problem Statement document due soon.	We reached out to our client to set up a meeting time.
3	We met with our client Professor Donald Heer on the OSU campus.	We gained plenty of insight into what the project would encompass.	We talked online and compared notes of the meeting to make sure we were all on the same page.
4	We completed the draft for the requirements document and got it fairly close to the final version.	The Gantt chart was not totally representative of our expected project timeline.	We searched and found new Gantt chart software to find one that could support our needs.
5	We separated the features of our app into different sections for the Tech Review assignments.	We had to be sure of what features and technologies we wanted to focus on in our app to accurately split up the work.	We consulted with our TA to get some examples of different types of technologies.
6	We individually completed our Tech Review papers focusing on different aspects of the project.	We needed to share our new knowledge about different technology options.	We talked together online and presented our Tech Reviews so we could all learn more about the options we chose.
7	We discussed online which technology options would be best for our application.	Final decisions about the options were still needed to be made.	Our team met up on campus to discuss and came to a decision.
8	We made decisions on what technologies we would go forward with for the web app.	The Requirements Document needed work to be filled out and accurate.	We discussed more online to fill out the document with more information on the technology choices.

9	We the draft of the Design Document with a solid list of the options we would be using.	Some of the options had overlap or we decided we weren't going to use.	Some editing needs to be done on the document to accurately reflect the options we'll be using.
10	We edited the Design Document and some of the charts from previous documents.	We have to send a collection of our documents to the client for approval.	We will meet online to finish the documents and write an email to the client.

4.1.2 Peng Zhang

Week	Progress	Problems	Plans
1	We worked on the resume and to select a project we want to do.	Confused about choosing project and find a project that most fit to me.	Pick the top ten choice for our project.
2	Meet with team and wrote Team Standards and talk with each other about the project	We need to know more about the project we need to do and don't know how to start it.	We plan to contact our client to get more information about this.
3	We met with our client Professor Donald Heer to ask for some information.	We confused about what the project looks like and everyone has their own understand about the project.	We talked about this and summarize everyone's idea to one idea.
4	we drafted the requirements document and prepared to complete the second draft.	Each member still has a different understanding of the application or this field.	We plan to unify everyone's opinions in the form of group meetings so that our goals are the same and clear.
5	Completed the second draft of the requirement document and start the Tech Review assignments.	We had problem that how to determine the details and plans of the project, including how to organize the upcoming plan.	Discuss issues in the form of group meetings and assign individual tasks in detail to ensure that everyone understands our own personal responsibilities.
6	We summarized the first draft of the Tech Review and discussed what everyone thought in the group meeting.	We hadn't do much communication about the Teach Review, lack of communication.	We held a meeting on Slack and talked about this, then shared the document each other.

7	Analyzed our tasks and explained the technologies and tools we will use, and prepare the Design document.	Our problem is to determine which tool are more helpful to us.	We met on campus and discuss which tools are more helpful to us, we tend to solve the problems together.
8	Prepared the final draft of the Design document, and made sure the technologies we need..	Many technical tools may not guarantee our work efficiency.	We discussed more online and plan to omit unnecessary tooling techniques to increase efficiency.
9	We completed and summarized all the documents, and we did the final upgrade to launch the client.	Some of the technical tools had overlap.	We talked about it online and decide the final choice.
10	Improved all the document and complete the first term of the Capstone class.	We still need to send all the document to our client.	Determined the final draft of all the document together and sent the documents to the client.

4.1.3 Yuxiao Huang

Week	Progress	Problems	Plans
1	We finished each own resume and chose our own favorite project	No idea how to choose a suitable project to me	Picked ten projects as an alternative.
2	Met with team and wrote Team Standards and talked with each other about the project	We had no idea how to start constructing our project and what we need to do for it	We were going to meet with our client professor for knowing more details about our project next week
3	This week was the first time to meet our client Professor Donald Heer on the OSU campus	After meeting with our client, we receive enough ideas about constructing our website. But we also get more questions	We created a Slack team online and communicate each other people. And we introduced the ability of each one
4	We accomplished project requirement document and the finished version is very close to the final version	After chekcing our project timeline, we found the Gantt chart had lots of discordant points with our project timeline	We found a software of creating Gantt chart to help us to finish a better chart now

5	We divided the project to several parts for finishing each feature better	Make sure each part is corresponding to each feature and each feature is satisfied with the project what the client needs	Our TA provides plenty of technical examples for helping us to understand the project deeper
6	As an individual assignment, Tech Review report asks us to introduce all aspects of the project. Each of us provides different point of view.	We need to share our own technical options for the Tech Review	We deal with the issues by communicating with other people on the Slack. We make certain of each aspect.
7	We received all suitable technical options for our project after communicating online	The final version of all technical options still need us to consider	After the end of the class on Thursday, we met each other and discussed the technical options
8	Make sure the technical options what we should follow for advancing the project	We still have some issues about the Requirement Document and we have to fix them at this weekend	On the Slack, each one provides different opinions about the Requirement Document. At the end of the meeting, we identified all methods
9	At this week, we have to fill out the Design Document and talk about it with our TA	We noticed some technical options are not suitable for project and we decided to fix the mistake	The mistake about the technical option performed we still have some shortage on finishing the tasks. But we found this at the beginning phase of the project
10	The Design Document got further modified and we changed some charts from the document	The document should be approved by the client, so we have to make an appointment with him	This is the last week of this term, so we meet online to end the documents and tell the client about our opinions

4.1.4 Yu Chuan Tey

Week	Progress	Problems	Plans
1	This week we work on resume and sign up for projects that we like.	Didn't meet any problem.	Submit 10 choices from a bunch of projects.

2	We meet after class and wrote team standard.	No idea what we need to do for our project.	Plan to meet our client.
3	We have our first meeting with our client Professor Donald Heer on the OSU campus.	We there is some problem of how to implement after met with our client.	We create slack group and gather our information.
4	Started draft 2 for requirement document.	There is a lot of suggestion for requirement document, and we need to gather them.	Gather all the information we need.
5	Finished requirement draft 2 and started Tech review draft.	Need to figure what tools we going to use for our project.	We discuss in our group meeting and talk with TA about this.
6	Finished final draft of Tech Review.	Lack of communication about Tech review and there is a bit conflict between this.	We talk about this issues and communicate in group meeting and slack to improve Tech review and everyone aspect.
7	Start our design document and discuss about what technology we are going to use.	We haven't decided about the final option for which technology we are going to use.	Meet up and communicate about this problem.
8	Final draft of design document due, and we need to improve our work from first draft.	The design document needed to be complete.	We communicate online finish the design document with some improvement from draft 1.
9	We finish all the documents and start preparing to let our client know about our document.	Some technology we are not able to use, and also some are overlapping.	We discuss about this and improve our document.
10	We improve our documents that require some changes.	We send our documents to the client for approval.	Client talk in email about what we need to make some changes for our documents.

4.1.5 Drake Evans

Week	Progress	Problems	Plans
1	No progress report for this week		
2	No progress report for this week		

3	Progress is a little behind the ideal milestone at the moment. The requirements document is a bit bare, and we need to meet with our client probably one more time before its exactly where it should be in terms of progress.	We're not really having any problems at the moment, now that we've all met and have an open channel of communication.	Plans for the coming week include finishing the group problem statement, and working more on our requirements document. Also, possibly starting a github repo for the project and adding all our current documentation to a wiki/folder.
4	Week 4 went fairly well. Requirements document made it in, as well as the final problem statement.	Problems at this stage were minimal, aside from time conflicts for working on stuff.	Looking forward, the next time to start the tech review and look at who's going to review what.
5	Progress is good, albeit slow. The requirements doc was successfully submitted, and we're slow to getting to our first tech review draft.	No problems have really arisen thus far.	Plans for the future include finishing tech reviews, and looking forward to the design document.
6	Progress this week has not been great, and I'll discuss the problems in a moment. My tech review is not complete yet, and I'll be finishing it as soon as possible.	Problems are a bit strong at the moment. Communication from the team is really bad at the moment, seeing as no one discussed how to split up tech review components, aside from me and one other. So I have no clue as to whether the rest of the team has even done anything with it.	Plans for the week include finishing up the tech review asap, and attempting to fix team communication. Oh, and can't forget about getting the design document draft done.
7	Week 7 went ok, the design document draft 1 was successfully submitted, though it does need a lot of work.	Biggest issue at the moment is poor group communication. We agreed to use slack, and for a while that worked well, but now it seems like no one pays attention to it unless a group assignment is due.	Plans for the week include finishing the second draft of the design document.

8	Progress is good, we have a second draft of the design document in, and communication is slightly improving.	Problems are that communication is still not great, but otherwise the project is going fine.	Plans for the future are to get client verification for our design document.
9	Progress has not really changed since last week, though it is time to start seeking client verification.	Problems have not changed.	Plans for this next week include getting client verification and editing our design document in any way we may need to get client verification.
10	No progress report for this week.		

4.2 Winter

4.2.1 Evan Hopper-Moore

Week	Progress	Problems	Plans
1	We completed our team critique and had good time to review and assess our accomplishments last term.	Our client let us know that our choice of back-end language wouldn't work with the OSU server's hosting so we'll have to investigate using PHP and Javascript.	We plan to meet up in person to start development on the project and get on the same page.
2	We started looking into implementing the project, I made the first commits adding a basic PHP project file structure and some instructions for running the project on the OSU servers.	Our mentor let us know that the project has to run on OSU servers so we are limited to certain versions of software and command line tools. So far it hasn't been too much of an issue, mostly a roadblock.	We will continue setting up the project, hopefully we can get to a point where we're experimenting with 3D graphics soon. We plan to talk online about new ideas we have and meet up with design prototypes.

3	We have decided on a design for our web page after sharing some prototypes and agreeing as a group.	We don't have total access to what is installed on the OSU server so we may run into problems in how our site is hosted.	We plan to email our client to get our design approved and get any information we can about how hosting the website and database will work. We also plan to begin setting up the website's UI once our client approves.
4	I restructured the project's codebase to better suit the layout of our project after the design review meeting.	We might be falling behind in implementation.	I plan to better explain the changes I have made in PHP in person to help implementation go smoother and faster.
5	This week I integrated login functionality through Google OAuth, letting users log in and create accounts through the external Google account sign in.	Progress has been slower than I thought on the project. It feels like we're not all quite on the same page as far as how fast development needs to be happening.	We're meeting again this weekend to catch up on what we've accomplished this week and to spend some development time together to make sure we're on the same page.
6	The functionality of the editor's input fields is coming along, soon it will be ready to integrate with the 3D aspect of the project.	The 3D side of the project is a bit behind the web application, mostly because it's the hardest part.	I plan to see if there are opportunities to do parts of the work to complete the 3d side faster, as well as completing work with database functionality and project saving.
7	We met with our client and he approved of what we have so far and made some suggestions going forward.	The 3D side of the app is still a little buggy and it needs some work.	We plan to get to a point where we can split up the tasks on the 3D part of the app.

8	We completed the design review and I'm excited to receive some feedback on how we've designed it so far.	We might be lagging a bit behind in the implementation schedule, hopefully we can pick up the slack this weekend and divvy up tasks for 3D development.	We will continue working on integration, getting the 3D feature branch up to date with the style changes on our master branch as well as developing more functionality.
9	I finished my individual reviews for other teams. In the design review I learned a lot about the other projects and the ways people are using libraries and technologies to complete their goals.	We might be a bit behind Beta functionality, we need to do more work on integrating the 3D elements of the project into the web development side.	We'll collaborate online to finish implementation of the 3D elements we have currently. Hopefully soon we can get hole placement fully implemented and accurate.
10	Very little progress has been made, mostly because our group has been preeoccupied with other classes at the end of the term. We did make some plans to get together to finish the app for beta functionality though.	We have been lacking some communication, but mostly because of other work as mentioned before.	We will finish up the features we want included in the Beta release, hopefully we can finish a few aspects with the 3D simulation and also complete the styling of the whole site.

4.2.2 Peng Zhang

Week	Progress	Problems	Plans
1	We did an assessment for each other this week, and this let us know how to fix our method of teamwork	Confused about how to start work or start coding.	We plan to meet with group and talked about this, planed to start coding next week.
2	We had a meeting this week and started coding now. We assigned issues to each group member on GitHub. Everyone is responsible for one part.	Our project has to work on OSU server, that is a limit to us.	We will continue to complete the task that everyone is assigned, and will attend a group meeting next week will discuss other solutions.

3	We shared the idea of the website layout and determined the final version of the website layout this week. And sent it to our client.	We got the client's feedback, and there are some problems about the website layout we need to fix.	We plan to fix our design when we get feedback from the client and planed to start coding on the website layout and some features like 3D graph.
4	We fixed our website layout based on the client's feedback.	We were stuck on the implement of the editor page, like how to create the 3D model.	We talked about the changes of the website layout, and plan to look for some information about the implementation.
5	I worked on the homepage styling, and we still working on the 3D and login functionality.	Our progress is a bit slower than our plan.	We met online and talked about everyone's work, and we planed to have a basically complete User Interface before week
6	We completed the User interface of the home, editor, library page and the login function connected to the database.	The problem is we cannot sure all the project can be completed before the presentation day, the 3D rendering still have something to be done	We plan to fasten the process, assigned tasks to everyone, like someone works on the 3D rendering, and others works on the optimize the whole project.
7	We met with our client and 3D model can be displayed successfully.	The 3D part still need more work because it has some bugs.	We plan to split the page and optimize the whole page to make sure it is good to present to other groups.
8	We just finished merging all the things we've done and did the design review with other groups.	We still have more work need to do for our project.	We plan to do the left things like the homepage logo, merge the 3D function to the Editor page, continue to work on the library page and import/export function.

9	We just completed the design review feedback this week and our project is also complete 80%.	We still need more things to do to finish the beta function.	We plan to look at all the feedback from other groups and draft a plan to modify or fix something according to the feedback. And for next term or next week, we will optimize something like UI, functions to enhance user experience.
10	Just done some optimization this week, and we still working on the beta functionality.	We have been lacking some communication.	We plan to finish the beta functionality before the weekend submits it on time. We talked about the communication problem, and we will improve this next term.

4.2.3 Yuxiao Huang

Week	Progress	Problems	Plans
1	We reviewed our work progress last semester and made some plans for the semester	Our client told us we should care about the choice of the programming language. We realized we should use PHP and Javascript to finish our website	We plan to meet each one and separate our work tasks
2	We created our first phase of the PHP files as the beginning part	Our mentors let us know that the project had to run on the OSU server.So far, this hasn't been a big deal, but it's not clear how you're going to run on an OSU server	We will continue to build this project and we are experimenting with 3D graphics.We plan to discuss our new ideas online and discuss how to solve this difficult point of 3D graphics construction

3	We are planing to share each own prototype for the website and share our own opinions	We cannot fully access the content installed on the OSU server, so we may need to figure out how to keep our content on the server first	We are going to send an email to our client in order to get permission to host our database on the server.Once our customer approves, we will start building the first UI version
4	In order to better adapt to our project layout, we made some corresponding plans	We have some mistakes on the implementation	I plan to have an in-depth understanding with other team members of the way PHP is used in our project, so as to facilitate the work requirements in the future
5	I built the first initial version of the library page with the help of Evan	The progress of the project was slower than we all expected, so we needed more communication and Shared working hours	We discussed the details of our work together in the library at the weekend, so as to reduce the unnecessary waste of time in the future work
6	According to Drake, more than half of the work on the 3D model is done	More optimization is needed in the construction of 3D model, probably because this part is the most difficult in our project	I plan to optimize the library page by searching more materials online
7	We met with the client this week and got some guidance	There are still some problems with building 3D models, but they are getting closer to the goal	We plan to break down the tasks more specifically in the 3D model building section
8	We have completed the design review and are satisfied with the current progress	There are still some difficulties in the 3D model development section, which we plan to split up this weekend	We intend to interface the development style of the 3D model section with the main style of our web pages
9	I have completed individual reviews of other groups.After finishing this assignment, I analyzed other projects and studied their data application methods.	We need to make the construction of 3D models more complete in the web pages	We plan to improve the construction of the 3D model on the network, and finally further develop the accuracy of the model

10	At the end of the term, the progress of the project slowed down obviously.Because each of our members set out to review other subjects	We lack some communication, but it's not a big problem, because it will be solved in future communication.	We are about to complete the beta version of our website, which requires us to take a deeper look at the details of our website

4.2.4 Yu Chuan Tey

Week	Progress	Problems	Plans
1	We completed our peer evaluation for last term and complete our team critique.	Our client let us know that our back end language are not able to work on OSU server and we have to change our language to PHP and Javascript.	We plan to meet and discuss about how to start development our project.
2	Start working and discuss on the project when in our meeting. We discuss about the outline, how our website is going to look like.	Our project is held in OSU server, and which this will limited our tools.	Plant to have some basic php for our website. We also using slack to talk about some new ideas.
3	We created some design from all of us, and we decided which design we going to use in our meeting.	Some of the part in OUS server we are not able to access, therefore this may run into some problem.	After we decided the design, then we plan to email our client for our design approved.
4	We discuss about how we are going to separate the task we going to implement.	We might be falling behind in implementation.	Plan we share what we wrote next meeting and improve what we got.
5	This week I implement our editor UI page. I did implement some sidebar with some UI.	There is some part of the UI I met and I pretty hard for me to use GitHub for merging and pushing.	We plan to meet and I need to ask teammate for helping get use to using GitHub.
6	All the web page are looking good, we share our code and discuss what we need to improve. We also have poster draft 1 and we were working good on it.	3D part is kind of left behind because it is very hard, and have to learn three.js. In addition, there is some functionality not working correct.	I plan to work on those functionality and try to make it working correctly.

7	We meet our client this week and demo what we have got so far.Client also give some information of what we need to do. We also discuss what we need to do next in our meeting.	The 3D side of the app is still a little buggy and it needs some work.	We tried to work on our website, and make all the functionality work correct and also start working on the requirement that client request.
8	We have design review and we demo in our design review. We also have to write feedback to other team.	3D part is a bit behind, and there is some functionality not working on 3d part.	We tried to work on it and integrate on 3D feature, and some other minor functionality for our website.
9	I finish the design review feedback to other team. On that design review demo, I did learn a lot of different kind of project, and how they implement.	We might be a bit behind Beta functionality, 3D part is a bit behind and the rest of our code is pretty well done.	We tried to our code is working correct and all the functionality is working.In addition, we tried to work on 3D part and to make sure some of the 3D functionality is working.
10	Our progress is slowing down because is end of term and a lot of other class work to do, also we have to prepare Beta demo.	We are lack of communication because we are busying of other works.	We tried to finish the requirement we made before Beta demo and also some functionality that we added.

4.2.5 Drake Evans

Week	Progress	Problems	Plans
1	Progress hasn't begun yet, we all decided to take a week to get settled into our classes and start next week.	No problems to report.	Plans for next week include starting our code, divvying up portions of work, etc.

2	Progress for the week is good, we have an initial design for the webpage, we have a many of the different parts of the project set up as issues on github to track progress better. We also have a simple php page set up for the website that we will begin to modify this week and start getting a product that looks more like what we are hoping to achieve.	No major problems to speak of at this point in time.	Plans for the coming week include getting webgl code base started, possibly getting a 3d model visualized on the webpage by the end of the week, reshaping the php page to look more like our design, as well as reshaping our design as we find ways to improve it. We may start setting up mySQL stuff, but not totally sure about that yet. Also setting up more concrete dates we want to have certain github issues completed by.
3	I had to miss the meeting on Wednesday for personal reasons, but I have begun work on the webgl code base for the project.	Problems are so far non existent, aside from some slight issue for me making as much time as I'd like to work on things, but the weekend is coming up, and I'll be able to work on it lots then.	Plans for the week include getting a working model display for the webpage, and at least make it moveable by mouse click and drag.
4	No progress report for this week		
5	Progress in week 5 is good, the website moves along pretty well, though we will need to hunker down this week to get the functionality we want for the alpha due at the end of the week. My progress is good, I have three.js working and a skeleton code for displaying a cube done, I just need to add interactivity and some aesthetics. After that, adding the hole placement functionality is next.	Problems aren't very present, other than making sure we get the alpha out on time.	Plans for the week include meeting with the team a few times to crank out the rest of the website for alpha. Adding user interactivity and hole placement must be usable by the end of the week.

6	Progress is good this week. Personally I've got 3d display working at a base level with user interaction. Other teammates have added webpages and login systems.	No major problems, just the continued time crunch with other class deadlines causing me to have less than desired time to work on the project.	Next step for me is to have editor options change the cube parameters. Following that, adding a hole placement feature. These things must be done before next Friday, as the client wants a demo then with these features implemented.
7	Progress this week is strong, a lot of progress from team members and myself. We also had a meeting with our client to demo our progress, and it went well. I have completed a proof of concept for hole placement and all the visuals that come with it. It's buggy, but just getting something to show our client was most important.	Problems aren't too big, just some merging issues. No real group conflicts.	Plans for the week include fleshing out the hole placement, adding in edge types, merging with master, and other features yet to be implemented.
8	No big progress this week, has the design review and other big projects to focus on.	No problems to speak of.	Plans for the week include integrating 3d modeling with the master branch, as well as getting edge type functionality.
9	Unfortunately no progress this week, I ended up being sick Saturday through well, today actually, with a fever and massive headache. Couldn't get any work done.	Problems are shown above, and no group problems.	Plans are to try and get all the functionality working but the end of the week.
10	Progress is good, a proper merge with the master branch has happened for the editor, a bug fix for the improper location on the hole placement had gone through, and work on saving the Model and edge type is in the works.	Problems for me include prioritization of all my assignments, but no group problems.	Plans include doing the group assignments due on Tuesday and finishing up functionality for at least a proof of concept for all base features.

5 POSTER

Our poster is the format of School of Electrical Engineering and Computer Science at Oregon State University and is included below.

COLLEGE OF ENGINEERING | Electrical Engineering and Computer Science | CS 35

Boxeur

Need to 3D print a case for a project but don't know how to use modeling software? Boxeur has you covered.

3D Case Design

- Our client Donald Heer noticed that many students don't have experience with or access to 3D computer-aided design applications to model the enclosures they need.
- This project will support many students who want to use the 3D printing and laser cutting tools in the OSU makerspaces, but don't have the skills required in 3D modeling software.
- We aimed to create an intuitive and easy to use experience, that can result in the exact case users need.
- Mistakes happen and sometimes revisions need to be made to projects, which is why we also included login functionality which lets users save projects to their account to edit later.

Project Advantage

- This product is undoubtedly very useful to OSU students who don't have experience.
- It is a web application so that users don't have to download or install it.
- It allows to save your project for future use to avoid lose your design.

Oregon State University

A simple editor

Using the three.js library we implemented an editor that has a live 3d visualization of the case being designed. We valued "what you see is what you get" design principles by having the model update as users input changes.

The four tabs on the left of the editor show different aspects of the case that can be edited. "Dimensions" lets users set the length, width, and height of the box. "Edge type" gives options for the types of edges that the editor supports, like interlocking finger shapes or a flat edge.

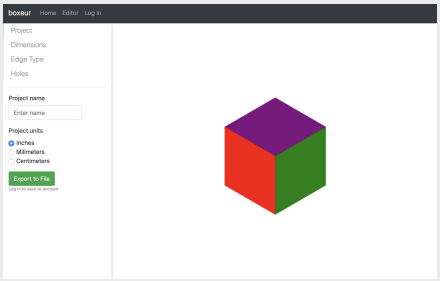


Figure 1: The editor page for Boxeur

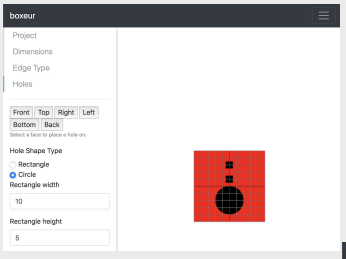


Figure 2: The hole cutting interface in the editor

Freedom in design

The "Holes" tab lets users cut different shapes and sizes of holes to allow for the openings needed for various projects. Users can select the face they would like to add holes to, then select the desired shape and size of the hole. Finally they can cut the holes by clicking directly on the case.

Save for later

When finished editing a project in the editor, users can save the model to the user's account if they log in with a google email. The library screen lets logged in users continue working on saved projects or delete them.

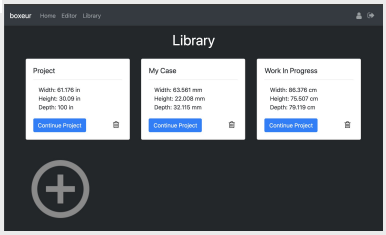
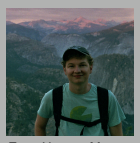
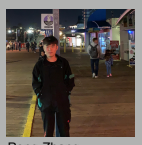


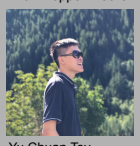
Figure 3: The library screen

About our team

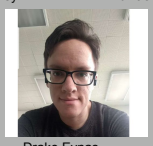
- We are computer science students from a variety of backgrounds who enjoy creating web tools for improving people's work efficiency.
- Don Heer is an EECS teacher at OSU who has lead the TekBots program to teach students about programming and robotics. He also works closely with the school's makerspaces.


Evan Hopper-Moore


Peng Zhang


Yu Chuan Tey


Yuxiao Huang


Drake Evnas

Donald Heer - ECE Professor at OSU
heer@eecs.oregonstate.edu

Evan Hopper-Moore - CS Junior
hopperme@oregonstate.edu

Yu Chuan Tey - CS Junior
teyy@oregonstate.edu

Drake Evans
evansdr@oregonstate.edu

Yuxiao Huang
huangyux@oregonstate.edu

Peng Zhang
zhangpen@oregonstate.edu

6 Project Documentation

The file structure of our project is a fairly standard setup for a simplistic PHP web application. All of the pages for the site are in the root directory and most of them include PHP files from the includes directory. The bulk of the html content is inside of the includes directory, while the supporting js and css is in their respective folders. The 3D simulation on the editor page is implemented in js/3dmodel.js and the supporting code for handling the editors inputs is in js/editor.js.

To run this project we used OSU's student servers by putting the project into the public_html folder. This will result in the website being hosted at http://web.engr.oregonstate.edu/ ONID_USERNAME/PROJECT_FOLDER_NAME/index.php. There is more information in the readme about troubleshooting and installation if needed. We also used the ONID MySQL server (specifically on Evan's ONID hopperme) as our database, which can be set up through the ONID login page.

We used Google OAuth for login functionality which requires that the exact redirect url for login is included in the whitelist on the admin page. For future development and final implementation a new Google API console account will need to be made for full control. Instructions on how to set up OAuth can be found on Google's Identity Platform (https://developers.google.com/identity/protocols/oauth2/web-server). We will keep the secret keys for OAuth and database login in the code, but for full control we recommend creating new accounts and replacing the keys.

7 Recommended Resources

The three.js developers have fairly extensive documentation that is all collected very neatly on the three.js docs site (https://threejs.org/docs/). This site also includes helpful examples and code snippets that helped us along the way. To implement exporting to .dxf, research on libraries like the npm package three-dxf (https://www.npmjs.com/package/three-dxf) could lead to a solution, but we concluded that we would have to implement our own way of arranging the dxf file type. To gain insight on how to format a dxf file, the official reference guide (https://images.autodesk.com/adsk/files/autocad_2012_pdf_dxf-reference_enu.pdf) could be a useful resource. Finally, for help with developing in PHP and integrating with MySQL, the W3Schools website (https://www.w3schools.com/php/) has an extensive tutorial with many helpful examples.

8 Conclusions and Reflections

8.1 Evan Hopper-Moore

Through this project I learned about 3D programming in Javascript using the library three.js. I also learned a lot more about setting up a website in PHP and connecting to MySQL database. Besides technical experience I also gained experience in working as a team, communicating over slack, and writing technical documents to describe project requirements and technical solutions. Through these experiences I learned how involved a full scale project can be, from writing documents to putting in the implementation work. I also learned that managing a project through GitHub issues and mileposts can be a great way to keep track of tasks that are left to do. GitHub was a very crucial tool for working in a team as well as slack for communication. If I could do it over again, I would have organized calls to meet up throughout the week to make sure the team was on the same page. I would have also started implementing more features earlier in the timeline of the project.

8.2 Peng Zhang

For the technical information, I have learned about using PHP to build a website and I have understood how to use JavaScript to create a 3D model. This allows me to understand the connection between PHP and JavaScript. In addition, I learned how to use the CSS to accurately adjust the website layout and the styling of the page. For the non-technical information, I learned how to plan a solution to our project, it is very important because it determines whether our progress can be completed on schedule. I also learned how to arrange time reasonably and strive to complete the task within the prescribed time. In addition, I learned how to work as a team, communicating problems and solving problems with each other has allowed us to learn from each other's strengths, this is a very useful experience for me. Project work is interesting because when we just got in touch with it, we don't know what it will look like in the future, so whenever we finish a part and summarize it, we always find some problems, such as unreasonable time assigned. Therefore, I learned self-assessment from the project work so that I won't make the same mistakes next time and improve work efficiency. For the project management, I learned to use GitHub to manage the project. As a team member, I was responsible for completing the tasks assigned by our leader, and I leaned how to assign tasks reasonably. I would to do a plan earlier and make the solutions according to the plan.

8.3 Yuxiao Huang

When I cooperated with other members for this project, I got a deeper understanding of Javascript and PHP. Before starting constructing the website, I have no idea about the relationship between PHP and a website. CSS is an important part of building a website and beautifying the details of the website. Since I learned CSS, I do not accept much experience on utilizing it but the project improved my ability of making the website aesthetic. About utilizing GitHub in a project is also significant to a team work, because the key point of a group work is to work efficiently and orderly. GitHub provides a beneficial environment for managing each part of our project. Every member could review the work process by checking the GitHub updated content. The layout design of a website should be the first thing to construct the whole frame and it saves our time when we are working on it. Except the improvement of my technical part, I learned some stuff from the non-technical information. Fortunately, I met several friendly team members and they helped me to solve the technical problems of website construction. For example, our team needs to utilize MySQL when we are creating a database. Although I have received MySQL information from other computer science course, I am still not familiar with using it out off that class. In my memory, my team members showed some helpful instances to me. At the middle phase of the project, we realized the team need a leader to guide other members to work efficiently. About the project management, I got plenty of methods from our leader Evan who arranged our tasks methodically. Actually, I was not good at working with other people in a team before joining the current group, but the current team members taught me how I communicate with the colleagues efficiently and how I manage the relationship between the work time and my life time. Work with other people is better than work alone. At least, we choose to deal with the issue with him quicker when he met a technical problem or non-technical problem. If I get a chance to do this project over again, I would choose to prepare more rudimentary PHP knowledge before starting constructing the website. And I still want the current team members indisputably. Work together with these team members is my pleasure.

8.4 Yu Chuan Tey

Through this project, I learn a lot from my team, including coding, how to cooperate with teammate. For the coding part, I did learn a lot like php, three.js, and some html. Before this project, I have no clue about writing a web page is able to

use .php instate of .html. On the other hand, I did learn a lot of technical writing skills, to intro to our project by using technical terms. In addition, we did plan what language we are going to use, how we are going to implement, after that we only implement coding after we have a clear image of what we are going to do. For non-technical information, I did learn about time management and communication skills. We manage our time wisely in order to finish the task. For communication skills, I did remember start of the project we lack of communication and cause some problem, and at the end of the project, we did a lot better than before. Therefore I think this is one of the big improvement from our team! I learned about project management by using GitHub. GitHub is really useful and able to cooperate everyone code into one. In addition, slack is a really good communication tool for all of us. If I could do it all over again, one thing I would change is communication skills, we should communicate frequently and this will improve our teamwork and might done our project better.

8.5 Drake Evans

This project definitely presented a very unique challenge and experience. It was the first experience I had working in a tech team on a sizeable project. On the coding side, I got more experience with OpenGL style code, utilizing the third party library Three.js in order to produce graphical models for the user to visualize the product. In terms of non-technical things, I learned much about working with a team, and the importance of properly assessing how to break up a project, and assigning levels of importance to different parts of a project. That's certainly something we as a team could've done better, in order to have better completed the project. In terms of project management, we definitely didn't assign resources properly. For working with a team, I have learned much. Its very important to keep constant communication about project elements going. Its importance to ask for help when needed, especially when an element of a project proves to be more work than initially anticipated. If I could've done this all over, I certainly would've done things differently. For one, I would've made sure we all better researched each individual aspect of the project, and assessed its importance and time requirements better. The 3D modeling aspect of this project definitely had more work than I expected, and not assigning two people to work on it definitely hurt finishing the project. Also not properly researching exporting to .DXF files ahead of time caused that element to be unfinished. But all in all, at the end of this project, I am satisfied with the experience I got from it. It was certainly a lot of fumbling that will teach me to better research for these projects and to seek help when needed.

Appendices

9 Essential Code Listings

The following code listings cover the core features of the project. The first is 3dmodel.js which handles the representation of the 3D model on the editor page. Next we cover library.php which connects to our database and fetches the saved projects to display them on the library page.

```
1 var camera, scene, renderer;
2 var material, outlinemat;
3 var controls;
4 var grid;
5 var raycaster, holeGeometry, holeMaterial, holeMesh;
```

```
6  var subtract;
7
8  var geometryList = [];
9  var meshList = [];
10 var outlineList = [];
11 var edgeType = 0;
12 var _face = 0;
13 var faceColors = [];
14
15 var holesList = []; //Hole list for saving the box
16
17 var lastWidth = 50, lastDepth = 50, lastHeight = 50; // !!!!!!  USE THESE FOR SAVING THE BOX SIZE
       !!!!!!!!!!!//
18 var boxWidth = 50, boxHeight = 50, boxDepth = 50;
19
20 var lastHoleType = "rect";
21 var holewidth = 5, holeheight = 5;
22
23 var thickness = 5;
24
25 var canvas = document.getElementById("model_canvas").getContext("webgl");
26 var canvasDims = document.getElementById("model_canvas").getBoundingClientRect();;
27 var width = canvasDims.width;
28 var height = canvasDims.height;
29 var aspect = width/height;
30
31 var mouse = new THREE.Vector2();
32
33 var removeMouseListener = false;
34 var removeHoleClickListener = false;
35
36 var saveError = false;
37
38 init();
39
40 setListeners();
41
42 animate();
43
44 //Set up variables, scene and renderer elements.
45 function init() {
46   scene = new THREE.Scene();
47   scene.background = new THREE.Color(0xffffff);
48
49   camera = new THREE.OrthographicCamera( 0.3 * width / - 2, 0.3 * width / 2, 0.3 * height / 2, 0.3 *
       height / - 2, 1, 2000 );
50
51   camera.position.x = 51;
52   camera.position.y = 51;
53   camera.position.z = 51;
```

```
54
55    scene.add(camera);
56
57    material = new THREE.MeshBasicMaterial({ color: 0xffffff, vertexColors: THREE.FaceColors });
58
59    flatEdgeModel();
60
61    /*outlinemat = new THREE.MeshBasicMaterial({color: 0x000000, side: THREE.BackSide});
62    //Setup outlines for visibility
63    for(var i=0; i<6; i++){
64      outlineList[i] = new THREE.Mesh(geometryList[0][i], outlinemat);
65      outlineList[i].scale.multiplyScalar(1.01);
66    }*/
67
68
69    /***** All "outlineList" related code is a NYI feature. Breaks the model view if uncommented. *****/
70
71    faceColors[0] = 0xff0000; //Front
72    faceColors[1] = 0x008000; //Right
73    faceColors[2] = 0x0000ff; //Back
74    faceColors[3] = 0xffff00; //Left
75    faceColors[4] = 0x800080; //Top
76    faceColors[5] = 0xff5733; //Bottom
77
78    for(var i=0; i<6; i++){
79      scene.add(meshList[0][i]);
80      //scene.add(outlineList[i]);
81    }
82
83    //holeGeometry = new THREE.BoxGeometry(5, 5, 15);
84    holeMaterial = new THREE.MeshBasicMaterial({color: 0xffffff});
85    holeGeometry = new THREE.BoxGeometry(5, thickness*2, 5);
86    holeMesh = new THREE.Mesh(holeGeometry, holeMaterial);
87    scene.add(holeMesh);
88
89    raycaster = new THREE.Raycaster();
90    raycaster.params.Line.threshold = 2;
91
92    renderer = new THREE.WebGLRenderer( { antialias: true, canvas: model_canvas} );
93    renderer.setSize(width, height, false);
94
95    controls = new THREE.OrbitControls(camera, document.getElementById("model_canvas"));
96  }
97
98  // checks to see if the session variables are set to load a saved project
99  function checkLoad() {
100   // if the session variables are set
101   if (sessionStorage.load == "true") {
102     // get project vars from session storage
103     load_name = sessionStorage.name;
```

```
    load_unit = sessionStorage.unit;
    load_width = parseFloat(sessionStorage.width);
    load_height = parseFloat(sessionStorage.height);
    load_depth = parseFloat(sessionStorage.depth);
    load_edgeType = parseInt(sessionStorage.edgeType);
    load_holes = JSON.parse(sessionStorage.holes);
    // clear storage
    sessionStorage.clear();

    // set project values in input boxes
    window.projectName = load_name;
    window.unit = load_unit;
    $("#name-input").val(load_name);
    $("#height-value").val(load_height);
    $("#width-value").val(load_width);
    $("#depth-value").val(load_depth);
    if (load_unit == "mm") {
      document.getElementById('milimeters').checked = true;
    }
    if (load_unit == "cm") {
      document.getElementById('centimeters').checked = true;
    }

    $("#project-options form").trigger("input");

    $("#slider-height").slider('value',load_height);
    $("#slider-width").slider('value',load_width);
    $("#slider-depth").slider('value',load_depth);

    loadDimensions(load_height, load_width, load_depth);
    loadHoles(load_holes);
  }
}

//Animation loop
function animate() {

  onWindowResize();

  controls.update();
  requestAnimationFrame( animate );

  render();

}

//Function to resize canvas when window changes size.
function onWindowResize(){

  canvasDims = document.getElementById("model_canvas").getBoundingClientRect();
```

```
  width = canvasDims.width;
  height = canvasDims.height;

  aspect = width / height;

  if (canvas.width !== width || canvas.height !== height) {
    renderer.setSize(width, height, false);
    camera.aspect = aspect;
  }

    camera.updateProjectionMatrix();

}

//Renderer function
function render(){

  if(scene.getObjectByName('grid') != null){
    raycaster.setFromCamera( mouse, camera );

    var intersects = raycaster.intersectObject( grid );

    if ( intersects.length > 0 ) {

      var fixed = intersects[0].point;

      holeMesh.position.copy( fixed );
      holeMesh.visible = true;

    } else {

      holeMesh.visible = false;

    }
  } else {

    holeMesh.visible = false;

  }

  renderer.render( scene, camera );
}

//
//
//
//
//Basic 3d code ends here, the following functions are listener handlers and helper functions for code
    clarity
//
```

```
203 //
204 //
205 //
206
207
208 //Function to change edge types
209 // *** NYI ***
210 function edgeTypeHandler(event){
211
212   switch(event.target.id){
213     case "flat":
214       break;
215     case "finger":
216       break;
217     case "t-slot":
218       break;
219   }
220 }
221
222 //Function for changing the hole shape
223 function holeType(event){
224
225   scene.remove(holeMesh);
226
227   holeMesh.quaternion.set(0, 0, 0, 0);
228
229   switch(event.target.id){
230     case "rect":
231       holeGeometry = new THREE.BoxGeometry(holewidth, thickness*2, holeheight);
232       holeMesh = new THREE.Mesh(holeGeometry, holeMaterial);
233       scene.add(holeMesh);
234       lastHoleType = "rect";
235       break;
236     case "triangle":
237       break;
238     case "circle":
239       holeGeometry = new THREE.CylinderGeometry(holewidth, holewidth, thickness*2, 30);
240       holeMesh = new THREE.Mesh(holeGeometry, holeMaterial);
241       scene.add(holeMesh);
242       lastHoleType = "circle";
243       break;
244     default:
245       break;
246   }
247
248   switch(_face){
249     case 0:
250       holeMesh.rotateX(Math.PI/2);
251       break;
252     case 1:
```

```
      holeMesh.rotateZ(Math.PI/2);
      break;
    case 2:
      holeMesh.rotateX(Math.PI/2);
      break;
    case 3:
      holeMesh.rotateZ(Math.PI/2);
      break;
    case 4:
    case 5:     //Top and bottom faces dont need any rotations
      break;
  }

}

//Change box geometry based on form values when a slider is being input or if a value is entered into the form.
//Really need to think of a more elegant way to do each face other than a switch.
function updateDimensions(event){


  //Only 2 sides need to be translated depending on what measurement is being changed
  //console.log(event.target.id, boxDepth, boxHeight, boxWidth);
  switch(event.target.id){
    case "slider-width":
    case "width-value":
      meshList[edgeType][1].geometry.translate((boxWidth-lastWidth)/2, 0, 0);
      meshList[edgeType][3].geometry.translate((-(boxWidth-lastWidth))/2, 0, 0);
      /*outlineList[1].geometry.translate((boxWidth-lastWidth)/2, 0, 0);
      outlineList[3].geometry.translate((-(boxWidth-lastWidth))/2, 0, 0);*/
      break;
    case "slider-height":
    case "height-value":
      meshList[edgeType][4].geometry.translate(0, (boxHeight-lastHeight)/2, 0);
      meshList[edgeType][5].geometry.translate(0, (-(boxHeight-lastHeight))/2, 0);
      /*outlineList[4].geometry.translate(0, (boxHeight-lastHeight)/2, 0);
      outlineList[5].geometry.translate(0, (-(boxHeight-lastHeight))/2, 0);*/
      break;
    case "slider-depth":
    case "depth-value":
      meshList[edgeType][0].geometry.translate(0, 0, (boxDepth-lastDepth)/2);
      meshList[edgeType][2].geometry.translate(0, 0, (-(boxDepth-lastDepth))/2);
      /*outlineList[0].geometry.translate(0, 0, (boxDepth-lastDepth)/2);
      outlineList[2].geometry.translate(0, 0, (-(boxDepth-lastDepth))/2);*/
      break;
  }

  //We want two measurements of each box face to scale, and one to translate, in order to preserve eventual material thickness property
  for(var i=0; i<6; i++){
```

```
    switch(i){
      case 0: //Front
        meshList[edgeType][i].geometry.scale(boxWidth/lastWidth, boxHeight/lastHeight, 1);
        //outlineList[i].geometry.scale(boxWidth/lastWidth, boxHeight/lastHeight, 1);
        break;
      case 1: //Right
        meshList[edgeType][i].geometry.scale(1, boxHeight/lastHeight, boxDepth/lastDepth);
        //outlineList[i].geometry.scale(1, boxHeight/lastHeight, boxDepth/lastDepth);
        break;
      case 2: //Back
        meshList[edgeType][i].geometry.scale(boxWidth/lastWidth, boxHeight/lastHeight, 1);
        //outlineList[i].geometry.scale(boxWidth/lastWidth, boxHeight/lastHeight, 1);
        break;
      case 3: //Left
        meshList[edgeType][i].geometry.scale(1, boxHeight/lastHeight, boxDepth/lastDepth);
        //outlineList[i].geometry.scale(1, boxHeight/lastHeight, boxDepth/lastDepth);
        break;
      case 4: //Top
        meshList[edgeType][i].geometry.scale(boxWidth/lastWidth, 1, boxDepth/lastDepth);
        //outlineList[i].geometry.scale(boxWidth/lastWidth, 1, boxDepth/lastDepth);
        break;
      case 5: //Bottom
        meshList[edgeType][i].geometry.scale(boxWidth/lastWidth, 1, boxDepth/lastDepth);
        //outlineList[i].geometry.scale(boxWidth/lastWidth, 1, boxDepth/lastDepth);
        break;
    }
    meshList[edgeType][i].geometry.verticesNeedUpdate = true;
    //outlineList[i].geometry.verticesNeedUpdate = true;
  }

  lastWidth = boxWidth;
  lastHeight = boxHeight;
  lastDepth = boxDepth;

}

// Load dimensions and resize box
function loadDimensions(h, w, d){
  boxHeight = h;
  boxWidth = w;
  boxDepth = d;

  //resize box to new dimensions
  geometryList[edgeType][1].translate((boxWidth-lastWidth)/2, 0, 0);
  geometryList[edgeType][3].translate((-(boxWidth-lastWidth))/2, 0, 0);

  geometryList[edgeType][4].translate(0, (boxHeight-lastHeight)/2, 0);
  geometryList[edgeType][5].translate(0, (-(boxHeight-lastHeight))/2, 0);

  geometryList[edgeType][0].translate(0, 0, (boxDepth-lastDepth)/2);
```

```
351   geometryList[edgeType][2].translate(0, 0, (-(boxDepth-lastDepth))/2);
352
353   //We want two measurements of each box face to scale, and one to translate, in order to preserve
          eventual material thickness property
354   for(var i=0; i<6; i++){
355     switch(i){
356       case 0: //Front
357         geometryList[edgeType][i].scale(boxWidth/lastWidth, boxHeight/lastHeight, 1);
358         break;
359       case 1: //Right
360         geometryList[edgeType][i].scale(1, boxHeight/lastHeight, boxDepth/lastDepth);
361         break;
362       case 2: //Back
363         geometryList[edgeType][i].scale(boxWidth/lastWidth, boxHeight/lastHeight, 1);
364         break;
365       case 3: //Left
366         geometryList[edgeType][i].scale(1, boxHeight/lastHeight, boxDepth/lastDepth);
367         break;
368       case 4: //Top
369         geometryList[edgeType][i].scale(boxWidth/lastWidth, 1, boxDepth/lastDepth);
370         break;
371       case 5: //Bottom
372         geometryList[edgeType][i].scale(boxWidth/lastWidth, 1, boxDepth/lastDepth);
373         break;
374     }
375     geometryList[edgeType][i].verticesNeedUpdate = true;
376   }
377
378   lastWidth = boxWidth;
379   lastHeight = boxHeight;
380   lastDepth = boxDepth;
381 }
382
383 // cuts holes from a saved list of holes
384 function loadHoles(holes) {
385   holesList = holes;
386   holesList.forEach(function(hole) {
387     scene.remove(holeMesh);
388     holeMesh.quaternion.set(0, 0, 0, 0);
389
390     switch(hole['type']){
391       case "rect":
392         holeGeometry = new THREE.BoxGeometry(hole['width'], thickness*2, hole['height']);
393         holeMesh = new THREE.Mesh(holeGeometry, holeMaterial);
394         scene.add(holeMesh);
395         break;
396       case "triangle":
397         break;
398       case "circle":
399         holeGeometry = new THREE.CylinderGeometry(hole['width'], hole['width'], thickness*2, 30);
```

```
        holeMesh = new THREE.Mesh(holeGeometry, holeMaterial);
        scene.add(holeMesh);
        break;
      default:
        break;
    }

    switch(hole['face']){
      case 0:
        holeMesh.rotateX(Math.PI/2);
        break;
      case 1:
        holeMesh.rotateZ(Math.PI/2);
        break;
      case 2:
        holeMesh.rotateX(Math.PI/2);
        break;
      case 3:
        holeMesh.rotateZ(Math.PI/2);
        break;
      case 4:
      case 5:     //Top and bottom faces dont need any rotations
        break;
    }
    pos = new THREE.Vector3(hole['x'], hole['y'], hole['z']);
    holeMesh.position.copy( pos );


    var newmat = new THREE.MeshBasicMaterial({ color: faceColors[hole['face']], vertexColors: THREE.
        FaceColors });
    subtract = threecsg.subtract(meshList[edgeType][hole['face']], holeMesh, newmat);
    scene.remove(meshList[edgeType][hole['face']]);
    //scene.remove(outlineList[_face]);

    /*outlineList[_face] = new THREE.Mesh(subtract, outlinemat);
    outlineList[_face].scale.multiplyScalar(1.5);*/

    scene.add(subtract);
    //scene.add(outlineList[_face]);
    meshList[edgeType][hole['face']] = subtract;
  });
}

//Function to change camera angle, call grid placement, and set up listener for hole placement helper
function holePlacement(event, x, y, z){

  camera.position.x = x; camera.position.y = y; camera.position.z = z;
  gridPlacer(event.target.id);

  document.getElementById("model_canvas").addEventListener('click', helper, false);
```

```
  //Though this helped prevent duplicate listeners, this caused a bug for not registering hole placement
       clicks on every other face button click
  /*if(removeHoleClickListener == true){
    document.getElementById("model_canvas").removeEventListener('click', helper, false);
    removeHoleClickListener = false;
  } else {
    document.getElementById("model_canvas").addEventListener('click', helper, false);
    removeHoleClickListener = true;
  }*/

}

//function to swap grid placement depending on which face button was clicked
function gridPlacer(face){

  if(scene.getObjectByName('grid') != null){
    scene.remove(grid);
  }

  holeMesh.quaternion.set(0, 0, 0, 0);

  switch(face){
    case "front":
      grid = new THREE.GridHelper(lastWidth, 10);
      grid.translateZ(lastDepth/2);
      grid.rotateX(Math.PI/2);
      _face = 0;
      holeMesh.rotateX(Math.PI/2);
      break;
    case "back":
      grid = new THREE.GridHelper(lastWidth, 10);
      grid.translateZ(-lastDepth/2);
      grid.rotateX(Math.PI/2);
      _face = 2;
      holeMesh.rotateX(Math.PI/2);
      break;
    case "top":
      grid = new THREE.GridHelper(lastHeight, 10);
      grid.translateY(lastHeight/2);
      _face = 4;
      break;
    case "bottom":
      grid = new THREE.GridHelper(lastHeight, 10);
      grid.translateY(-lastHeight/2);
      _face = 5;
      break;
    case "right":
      grid = new THREE.GridHelper(lastDepth, 10);
      grid.translateX(lastWidth/2);
      grid.rotateZ(Math.PI/2);
```

```
498       _face = 1;
499       holeMesh.rotateZ(Math.PI/2);
500       break;
501     case "left":
502       grid = new THREE.GridHelper(lastDepth, 10);
503       grid.translateX(-lastWidth/2);
504       grid.rotateZ(Math.PI/2);
505       _face = 3;
506       holeMesh.rotateZ(Math.PI/2);
507       break;
508   }
509
510   grid.name = "grid";
511   scene.add(grid);
512
513   if(removeMouseListener == true){
514     document.getElementById("model_canvas").removeEventListener('mousemove', onCanvasMouseMove, false);
515     removeMouseListener = false;
516   } else {
517     document.getElementById("model_canvas").addEventListener('mousemove', onCanvasMouseMove, false);
518     removeMouseListener = true;
519   }
520 }
521
522 //Helper function to place a hole.
523 function helper(){
524
525   if(scene.getObjectByName('grid') != null){
526     raycaster.setFromCamera( mouse, camera );
527
528     var intersects = raycaster.intersectObject( grid );
529
530     if ( intersects.length > 0 ) {
531
532       var intpoint = intersects[0].point;
533       console.log(intpoint);
534
535       /***** These lines will move the hole to a desired position, just change intpoint to a (new THREE.
              Vector3(x, y, z)) with the desired coordinates *****/
536       /*holeMesh.translateX(intpoint.x);
537       holeMesh.translateY(intpoint.y);
538       holeMesh.translateZ(intpoint.z);*/
539
540       var newmat = new THREE.MeshBasicMaterial({ color: faceColors[_face], vertexColors: THREE.
              FaceColors });
541       subtract = threecsg.subtract(meshList[edgeType][_face], holeMesh, newmat);
542       scene.remove(meshList[edgeType][_face]);
543       //scene.remove(outlineList[_face]);
544
545       /*outlineList[_face] = new THREE.Mesh(subtract, outlinemat);
```

```
      outlineList[_face].scale.multiplyScalar(1.5);*/

      scene.add(subtract);
      //scene.add(outlineList[_face]);
      meshList[edgeType][_face] = subtract;

      /*  SAVE HOLE OBJECTS HERE, THIS IS WHERE HOLE PLACEMENT OCCURS */
      if (lastHoleType == "rect"){
        holesList.push({type: lastHoleType, x:intpoint.x, y:intpoint.y, z:intpoint.z, face:_face, width:
            holewidth, height:holeheight})
      }
      if (lastHoleType == "circle"){
        holesList.push({type: lastHoleType, x:intpoint.x, y:intpoint.y, z:intpoint.z, face:_face, width:
            holewidth})
      }
    }
  }
}

//Function tracking mousemovement when in hole placement mode
function onCanvasMouseMove(event){

  event.preventDefault();

  mouse.x = ( ( event.clientX - canvasDims.left ) / ( canvasDims.right - canvasDims.left ) ) * 2 - 1;
  mouse.y = - ( ( event.clientY - canvasDims.top ) / ( canvasDims.bottom - canvasDims.top ) ) * 2 + 1;

  //console.log(mouse);
}

//Function to set up listeners. Keeps code a little cleaner near the top.

function setListeners(){
  //Set listeners for the dimension options
  document.getElementById("slider-width").addEventListener('input', updateDimensions, false);
  document.getElementById("slider-height").addEventListener('input', updateDimensions, false);
  document.getElementById("slider-depth").addEventListener('input', updateDimensions,false);
  document.getElementById("width-value").addEventListener('input', updateDimensions, false);
  document.getElementById("height-value").addEventListener('input', updateDimensions, false);
  document.getElementById("depth-value").addEventListener('input', updateDimensions,false);

  //Set listeners for edge types
  document.getElementById("flat").addEventListener('click', edgeTypeHandler, false);
  document.getElementById("fingers").addEventListener('click', edgeTypeHandler, false);
  document.getElementById("t-slot").addEventListener('click', edgeTypeHandler, false);

  //Set listeners for hole options
  document.getElementById("rect").addEventListener('click', holeType, false);
  document.getElementById("circle").addEventListener('click', holeType, false);
  document.getElementById("hole-width").addEventListener('input', function(){
```

```
    holewidth = document.getElementById("hole-width").value;
  }, false);
  document.getElementById("hole-height").addEventListener('input', function(){
    holeheight = document.getElementById("hole-height").value;
  }, false);

  //Set listeners for what side to look at during hole placement
  document.getElementById("front").addEventListener('click', function(e){holePlacement(e, 0, 0, 51)},
      false);
  document.getElementById("back").addEventListener('click', function(e){holePlacement(e, 0, 0, -51)},
      false);
  document.getElementById("top").addEventListener('click', function(e){holePlacement(e, 0, 51, 0)},
      false);
  document.getElementById("bottom").addEventListener('click', function(e){holePlacement(e, 0, -51, 0)},
      false);
  document.getElementById("right").addEventListener('click', function(e){holePlacement(e, 51, 0, 0)},
      false);
  document.getElementById("left").addEventListener('click', function(e){holePlacement(e, -51, 0, 0)},
      false);
}

// When saving the data for the entire object, youll need the size of the object itself, edge type (not
    currently implemented)
// and a list/array of holes, using the hole class below.

//  CODE FOR HOLES OBJECT  //
//Constructor
class Hole {
  constructor(x, y, z, type, face, width, height) {
    this.x = x;
    this.y = y;
    this.z = z;

    this.type = type;
    this.face = face;

    this.width = width;
    this.height = height;
  }
}

// Saving to users profile
try {
  document.getElementById("account-save").addEventListener('click', function() {
    // console.log(window.unit);
    // console.log(window.projectName);
    // console.log(lastWidth);
    // console.log(lastDepth);
    // console.log(lastHeight);
    // console.log(JSON.stringify(holesList));
```

```
    // console.log(edgeType);

    // make sure theres a project name
    if (window.projectName == ""){
      $("#save-error").removeClass("d-none");
    } else {
      if ($("#save-error").attr('class') == ""){
        $("#save-error").addClass("d-none");
      }
      // make POST request to backend
      $.post("saveproject.php", {
        name: window.projectName,
        unit: window.unit,
        height: lastHeight,
        width: lastWidth,
        depth: lastDepth,
        edgeType: edgeType,
        holes: JSON.stringify(holesList),
      }, function(data,status){
        console.log(status);
        $("#save-success").removeClass("d-none");
            setTimeout(function(){ $("#save-success").addClass("d-none"); }, 3000);
      });
    }

  });
} catch(e) {

}

// Export to file
document.getElementById("export").addEventListener('click', function() {
  console.log("export");
});
```

```
<?php include("includes/config.php");?>
<!DOCTYPE html>
<html>

<head>
  <?php
    $PAGE_TITLE = "Library";
    include("includes/head-contents.php");
  ?>
</head>
<style>
  body{
    background-color: #23272A;
    overflow: auto;
  }
</style>
```

```
17
18 <body>
19   <?php
20     include("includes/library-nav.php");
21
22     // load projects from the user into the array $projects
23     if (isset($_SESSION['access_token']) && $_SESSION['access_token']) {
24       $projects = [];
25
26       $servername = "oniddb.cws.oregonstate.edu";
27       $username = "hopperme-db";
28       $password = "lEl05pJffs7OtcfV";
29       $dbname = "hopperme-db";
30
31       // Create connection
32       $conn = new mysqli($servername, $username, $password, $dbname);
33       // Check connection
34       if ($conn->connect_error) {
35         die("Connection failed: " . $conn->connect_error);
36       }
37
38       // Get user from database
39       $email = $_SESSION["account"]["email"];
40         $result = $conn->query("SELECT * FROM users WHERE email='$email'");
41         if ($result->num_rows != 0) {
42             $row = mysqli_fetch_assoc($result);
43             $user =  $row['id'];
44
45         if (isset($_POST['del'])) {
46           if($_POST['del'] == 1) {
47             $proj_id = $_POST['id'];
48             $sql = "DELETE FROM projects WHERE id='$proj_id' AND user='$user' ";
49
50             if ($conn->query($sql) === TRUE) {
51               echo "Record deleted successfully";
52             } else {
53               echo "Error deleting record: " . $conn->error;
54             }
55           }
56         } else {
57           // Get all the projects that the user owns and append to projects array
58           $sql = 'SELECT * FROM projects WHERE user ="'. $user. '"';
59           $result = $conn->query($sql);
60           if ($result->num_rows > 0) {
61               while($row = $result->fetch_assoc()) {
62                   $response['id'] = $row["id"];
63                   $response['name'] = $row["name"];
64                   $response['width'] = $row["width"];
65                   $response['height'] = $row["height"];
66                   $response['depth'] = $row["depth"];
```

```
                $response['unit'] = $row["unit"];
                $response['edgeType'] = $row["edgeType"];
                $response['holes'] = $row["holes"];
            array_push($projects, $response);
            }
          }
        }
        }
      $conn->close();
    }
  ?>
  <?php if (!isset($_SESSION['access_token']) || !$_SESSION['access_token']): ?>
    <script>window.location = 'index.php';</script>
  <?php endif;?>

<div class="container">
    <h1 id="library-title">Library</h1>
  <div class="row" id="library-row">
    <?php foreach ($projects as $index=>$project): ?>
          <div class="col-md-6 col-lg-4 card-col">
              <div class="card" style="width: 18rem;">
                <div class="card-body">
                  <h5 class="card-title" style="font-weight: bold;"><?= $project['name']?></h5>
          <hr class="card-hr">
                  <p class="card-text">
            <ul class="specs-list">
              <li>Width: <?= $project['width']?> <?= $project['unit']?></li>
              <li>Height: <?= $project['height']?> <?= $project['unit']?></li>
              <li>Depth: <?= $project['depth']?> <?= $project['unit']?></li>
            </ul>
          </p>
          <a class="btn btn-primary cont-btn" id="cont-<?= $index?>">Continue Project</a>
                  <button class="btnDelete btn-delete" id="del-<?= $index?>">
            <i class="fa fa-trash-o fa-lg"></i>
                  </button>
                </div>
              </div>
          </div>
      <script type="text/javascript">
        document.getElementById("cont-<?= $index?>").addEventListener('click', function() {
          sessionStorage.load = 'true';
          sessionStorage.name = "<?= $project['name']?>";
          sessionStorage.width = <?= $project['width']?>;
          sessionStorage.height = <?= $project['height']?>;
          sessionStorage.depth = <?= $project['depth']?>;
          sessionStorage.unit = "<?= $project['unit']?>";
          sessionStorage.edgeType = <?= $project['edgeType']?>;
          sessionStorage.holes = '<?= $project['holes']?>';
          window.location.href = "editor.php";
        });
```

```
117
118         document.getElementById("del-<?= $index?>").addEventListener('click', function() {
119           $.ajax({
120               url: 'library.php',
121               type: "POST",
122               data: { del: 1, id: <?= $project['id']?> }
123           }).done(function( msg ) {
124               window.location = 'library.php';
125           });
126         });
127       </script>
128     <?php endforeach; ?>
129     <div class="col-md-6 col-lg-4" style="padding-top:40px;">
130       <a href="editor.php">
131         <image class="add" href="../css/editor.css" src="img/add.png" align="middle">
132       </a>
133     </div>
134   </div>
135
136 </div>
137
138
139 <?php include("includes/footer.php");?>
140
141 </body>
142
143 </html>
```

10 Code Review Responses

For the code review we received feedback from other groups and took into account the points they made about our project. First we'll include the summary of the points made in the code review and how we responded to them, then we'll include all of the direct feedback from the other group members.

Summary of Code Review Feedback

1. **Build**
    a. Most reviews stated that the project was easy to get up and running but could have included more troubleshooting information in the readme. There were some issues with OAuth login functionality.
    b. We included more information on how to troubleshoot issues with installation and permission issues on the OSU servers. Updates were pushed to fix the permission errors in new projects for OAuth functionality along with instructions on how to contact us to get your server approved for OAuth redirects.
2. **Legibility**
    a. Overall most reviewers found our code easy to read and well organized. We got compliments on our file structure and folder names as well as our functions and variable names. We did receive a couple notes about styling in the javascript, which needs work with indentation and adding more comments
    b. We increased the amount of comments in the code making it easier to read, as well as running an automatic linter which fixes style issues in the code. While adding more features made our files longer and harder to understand, we kept our functions small, legible and well commented to make sure the code is still readable.
3. **Implementation**
    a. A few reviews remarked that our functions were a bit bloated in some places (especially our 3D code in js) but otherwise the reviews were positive. Our clean file structure and function abstraction resulted in a concise implementation.
    b. In response to the review about bloated functions in our javascript we made sure to split up the functions we could, and add more comments where functions couldn't be abstracted further. Most of the problems with messy code were in the js, as php helped keep the implementation clean and easy to read.
4. **Maintainability**
    a. All reviews were positive, but some mentioned the fact that there aren't unit tests and that they might not be needed.
    b. Since this is a web based project, unit tests aren't really the most applicable way to keep maintainability. The only problems we are worried about with maintainability are issues related to the database, which is hosted on the OSU ONID MySQL server. This allows us to access a visual

admin page where we can easily see the database and update the data as needed.

5. **Requirements**
    a. The main features lacking from our project at the time of the code review are mostly related to the editor UI such as hole placement as well as saving and loading projects. Most reviewers mentioned that some work was left to be done on the UI but the project looked close to finished.
    b. Before the code freeze we were able to complete hole placement and fix a lot of the bugs that were present at the code review. We were also able to finish implementing saving, loading, and deleting projects and making the projects viewable from the library screen.
6. **Other**
    a. Some reviews mentioned adding a tutorial page on the website, as well as an about page with information about the project and the team. Another reviewer talked about including instructions to run the project on a local device instead of the OSU servers.
    b. To incorporate these ideas we added more information on the homepage explaining some information about the team and the project as well as instructions on how to use it. We also added some information in the Github readme with ideas on how to run on a local machine and on what the OSU servers use for php.

Category	Description	Reviewers Comment	Action taken by reviewed group
Build	Could you clone from Git and build using the README file?	Yes. The instructions on the GitHub was helpful.	
Legibility	Was the flow sane and were variable names and methods easy to follow? Does the code adhere to general guidelines and code style?	The variable names and methods are easy to follow. And the code style is also good. The whole code gets divided into different blocks which makes it become easier to read.	
Implementation	is it shorter/easier/faster/cleaner/safer to write functionality equivalent code? Do you see useful abstractions?	It is good enough. I think most of the features are implemented, and the code is also looks clean & well organized.	
Maintainability	Are there unit tests? Should there be? Are the test covering interesting cases? Are they readable?	The speaker lead us went through some 3D design's functionalities such as putting hole on a cube and it works. I personally think the cursor size need to be consistent. The project can successfully scan the whole operating board and figure out where does the object located and what position can a cursor have valid scan.	
Requirements	Does the code fulfill the requirements?	I think they are almost there, just few functionalities need to be double checked such as putting holes on different surfaces, I noticed that the cursor has different sizes. And also the UI elements might need to be optimized in the future. But overall, it is a great project.	

Other	Are there other things that stand out that can be improved?	I think nothing needs to be improved except those few functionalities they talked about during the resentation.	

Code Review Feedback
Calvin Gagliano

1. **Group 35**
    a. **Build**
        i. Easy to build. There were a few issues with my public_html folder (I had to reset it to make it work). Then, there were issues with Google OAuth, but that's understandable since I'm not building a production build.
    b. **Legibility**
        i. It seems like there weren't many comments made in most of the code. Some lines of code showcased in the code review are not indented properly and do not follow the same standard as the rest of the code (for example, check indentation in editor.js). Overall, good quality and legible though.
    c. **Implementation**
        i. No, don't see many useful abstractions that they didn't use. Some of the functions seem a little bloated, but I don't know the libraries very well so it could've been the only way to do the things they are trying to do.
    d. **Maintainability**
        i. I didn't see any unit tests, but they should probably be put on hold until the last of the requirements are completed.
    e. **Requirements**
        i. There were some requirements that are not fulfilled completely, but they mentioned those in the presentation. The functionalities missing can most likely be implemented in the weeks before the code freeze.
    f. **Other**
        i. The permissions.sh didn't work for me which was unfortunate. Also, if you want non-OSU students to contribute, you may want to make a tutorial for installation/building without an OSU account.
2. **Team 37**
    a. **Build**
        i. Very easy to clone and build. Didn't have any problems with the web-based solution at least. README is clear and concise.
    b. **Legibility**
        i. I'm reviewing mainly the web application that was implemented with React. The files seem to lack comments in a lot of places, but overall it's clean code and easy to see where things are happening. Between files in the web application, there are different programming styles, but both are very readable so it's not that big of a deal
    c. **Implementation**
        i. I'm not aware of any better methods they could've used besides using Bootstrap for their CSS for the web application. I think there may be

libraries out there that port applications from iOS to Android and vice versa, but I'm not sure.

d. **Maintainability**
    i. I didn't see any unit tests. It might be useful to get a testing environment (or pre-production environment) for testing though since it's mainly a database application.
e. **Requirements**
    i. It looks like most of the requirements have been fulfilled which is great!
f. **Other**
    i. Make the build instructions more explicit on the web application README.

Category	Description	Reviewers Comment	Action taken by reviewed group
Build	Could you clone from Git and build using the README file?	Yes, the app is easy to install, and the instruction is clear.	
Legibility	Was the flow sane and were variable names and methods easy to follow? Does the code adhere to general guidelines and code style?	The variable names and methods are very easy to follow. The project seems to adhere to all the general guideline and styles.	
Implementation	is it shorter/easier/faster/cleaner/ safer to write functionally equivalent code? Do you see useful abstraction?	Using php for the project is a good choice, which simplify many of the login session implementation. The code overall looks clean, and the functionality of the 3D Canvas is fairly fast, and importantly working.	
Maintain ability	Are there unit tests? Should there be? Are the tests covering interesting cases? Are they readable?	There is no unit test to my experience, but the project doesn't seem to need any unit test. The code is readable and looks maintainable.	
Requirements	Does the code fulfill the requirements?	It looks like the login function needs some work, but overall really like the project. To my understanding, it looks like it fulfills the requirement.	
Other	Are there other things that stand out that can be improved?	Maybe you can add a "about me" section that explain the project is about and gives the credit to the developer. Also, maybe you could have a contact me section.	

1. **Team 35**
    a. **Build**
    Yes, the project is accessible on GitHub and the README gives enough instructions to set everything up.
    b. **Legibility**
    The variable names are making sense and easy to understand what it represents for. The code is in good file trees which can help easily find the related part.
    c. **Implementation**
    The implementation is good, functions are abstracted well, no redundant code, no spaghetti code, functions are all written to undertake one specific functionality.
    d. **Maintainability**
    Code is good to be maintained, directories name can easily guide through to find the part need to manipulate with.
    e. **Requirements**
    Most requirements are full filled.
    f. **Other**
    If there can have an instruction to solve the permission issue of running ./ permission.bash will be more helpful to set the project up.

Category	Description	Reviewers Comment	Action taken by reviewed group
Build	Could you clone from Git and build using the README file?	The README was really helpful and clear with the installation.	
Legibility	Was the flow sane and were variable names and methods easy to follow? Does the code adhere to general guidelines and code style?	Styling and code readability are on point and easy to follow. It seems that it followed the guideline and styling.	
Implementation	is it shorter/easier/faster/cleaner/safer to write functionally equivalent code? Do you see useful abstractions?	The code is clean and organized. I am truly amazed by how nicely done the 3D canvas is. Great use of WebGL.	
Maintainability	Are there unit tests? Should there be? Are the test covering interesting cases? Are they readable?	The presenter did demonstrate some of the features and functionalities of the project. I didn't see any unit test with the project. The code is readable and the cost of maintainability seems relatively low.	
Requirements	Does the code fulfill the requirements?	Some of the UI elements can be improved, like putting holes onto the 3D model during the demonstration was running into some issue. Overall, I think it seems to fulfill the requirements.	
Other	Are there other things that stand out that can be improved?	I think other than some of the UIs can be improved, there's nothing much for me. Since this is a web application, maybe you can include something like a "help" or some kind of feature that can provide more information about this application. Possibly some tutorials on some features that can be not so explicit to the users?	

Group 35

Build	Could you clone from git and build using the README?	Readme is kind of clear. Is it necessary that I run it on the school server, or can I host on my home computer?
Legibility	Was the flow sane and were variable names and methods easy to follow? Does the code adhere to general guidelines and code style?	Project looks good. The file structure is well-organized, variable names are reasonable, and functions are well-written. The opengl shaders, however, should probably be refactored to their own files instead of being literal strings, as that looks like it'd be a nightmare to fix.
Implementation	is it shorter/easier/faster/cleaner/ safer to write functionally equivalent code? Do you see useful abstractions?	I'm not too skilled at javascript, but it looks well-written to me. There are well-arranged files.
Requirements	Does the code fulfill the requirements?	Looks like it's nearing completion. The UI can be improved.
Maintainability	Are there unit tests? Should there be? Are the test covering interesting cases? Are they readable?	I couldn't find any tests, and the readme doesn't describe any. This is mostly a user-interface based app so tests might be more difficult, but I think a few might be useful.