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.	

Category	Description	Reviewers Comment	Action taken by reviewed group
Build	Could you clone from Git and build using the README file?		
Legibility	Was the flow sane and were variable names and methods easy to follow? Does the code adhere to general guidelines and code style?		
Impleme ntation	is it shorter/easier/faster/cleaner/ safer to write functionally equivalent code? Do you see useful abstractions?		
Maintain ability	Are there unit tests? Should there be? Are the test covering interesting cases? Are they readable?		
Require ments	Does the code fulfill the requirements?		
Other	Are there other things that stand out that can be improved?		

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.