

Online real estate agency

Introduction:



This project is inspired from <https://www.muse.place/>

The goal is to have a 3D interactive World to collaborate in context of visiting an accommodation with an agent.

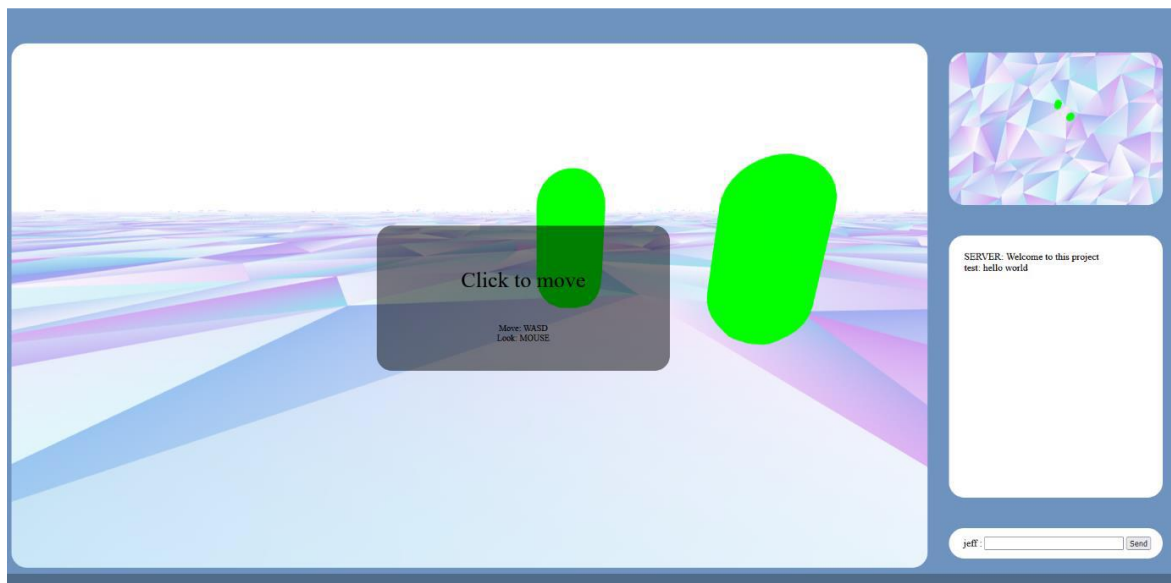
To do that we need:

- A 3D world space to move around
- Clients/server connection to communicate between Users and to edit data
- A Database for consistency and avoid delay
- Some listener to interact with web page
- Accounts to manage connection

To deals with those sparts, we will use some libraries/API/addons like:

- Node.js (server/app management)
- Three.js (3D world)
- Express (application structure)
- EJS (html managing)
- Socket.io (web socket, communication clients/server)
- JQuery (call ajax with database)

When we get there, we want to build a server/app with real time interactions and a fancy User Interface to avoid mistake and miss understanding.



Features:

We are looking for features to implement which are:

- Shared views
- Accounts
- An 3D editing menu shared between users
- Moving/Adding objects

Shared views:

To get shared views we have to save user proprieties like camera angle, position. Then we switch them with those of the actual user or we create a state to switch views without losing the old data. I choose to use a state to switch, because in this way the user still can move in front of others with others' view (it makes something like a 2nd person camera). And we need a way to switch views (here I use buttons which are in same color of user character).

Accounts:

I didn't focus on this feature because it was not mainly relative to the class, but it's easily possible to make it better with more security by adding password and encrypting data. Actually, you just need your username to connect and account have to be added by the developer to the database (Json).

3D menu:

For this feature, I didn't manage to have enough time to work on it. But I will continue to work on this project after the class and will try to make it implemented.

To make it, in three.js side I have the possibility to know a "raycaster" which is the idea of a direction line projected by the camera. With this option I need to generate an object design as the menu and to cast it when asked or detected.

Moving/Adding objects:

As same as the 3D menu, I didn't have enough time to work on it. Three.js have an addons that can be use to move object according to a "gizmo" with the cursor. Or it's possible to just use a translation. To do so it need to have a state when editing or moving around, then a transparent previsualization as a fast forward when moving before confirmation. (It is also mandatory to add those changes to the database to keep constancy).

Adding object would require a list of already existing models or a pop-up/interface to import files.

Issues:

The first issue that I encounter was to make the communication between node/express with three.js. It was news thing for me because before I only used classic html with some JavaScript and a lot of PHP. The problem was that when you reload the page you also reload the three.js script which load the scene and cause lags/delay.

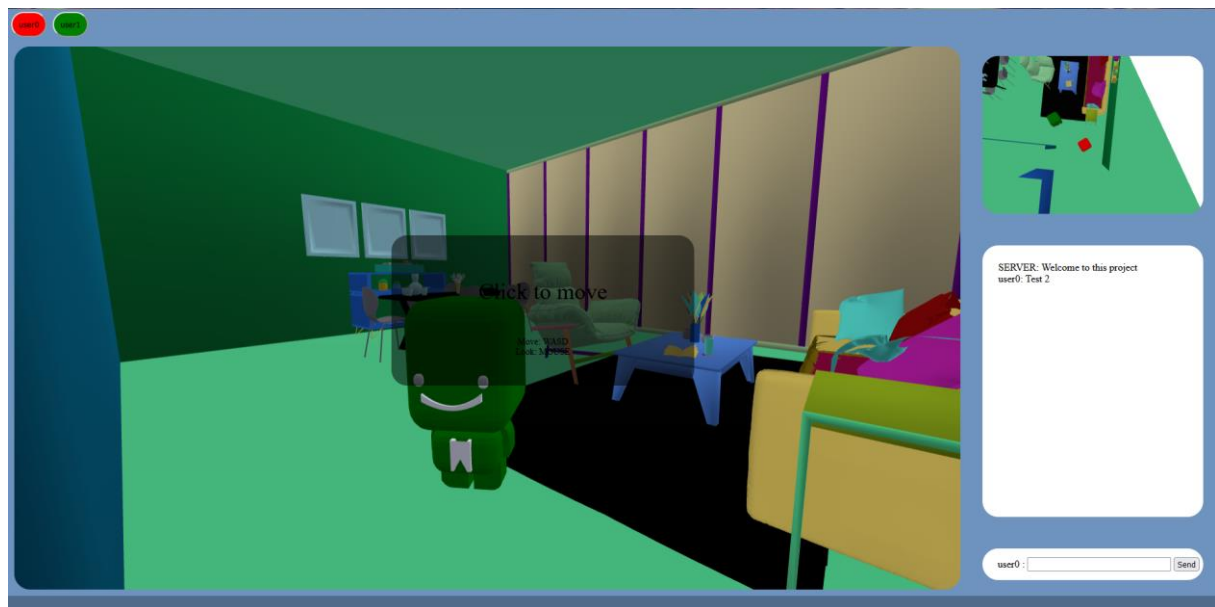
To make it working I discovered and used socket.io which let me modify data without reloading the whole page. But it also adds me some troubles with the database, because I had to put functions in the good order to avoid overwrite by multiple users and/or having corrupted data.

One big issue was to keep it in real-time with consistency. So, I created a script to update data for the user, the database and to emit for others. It uses a collaboration decentralized with socket.io.

Beside the code, finding models, textures or object for free was a pain and I couldn't make my own models for the time given. So, some models are not really fancy because of that.

And finally, I didn't have time to implement and learn all features in time by my own. So, I focused and the shared view, the real-time and consistency as I think it's the most important relatively to the class.

Conclusion:



That is the result I get from my work. I'm proud of what I accomplished according to time, but I will work more on this project because I really loved it.

The web site is working for multiple users in real-time with consistency. It is not really fancy as I would at first but I make shared view and database working well between users.

As a future outlook, in the future I would finish to implement others features and to put it online with encrypted data to make user data safe.