

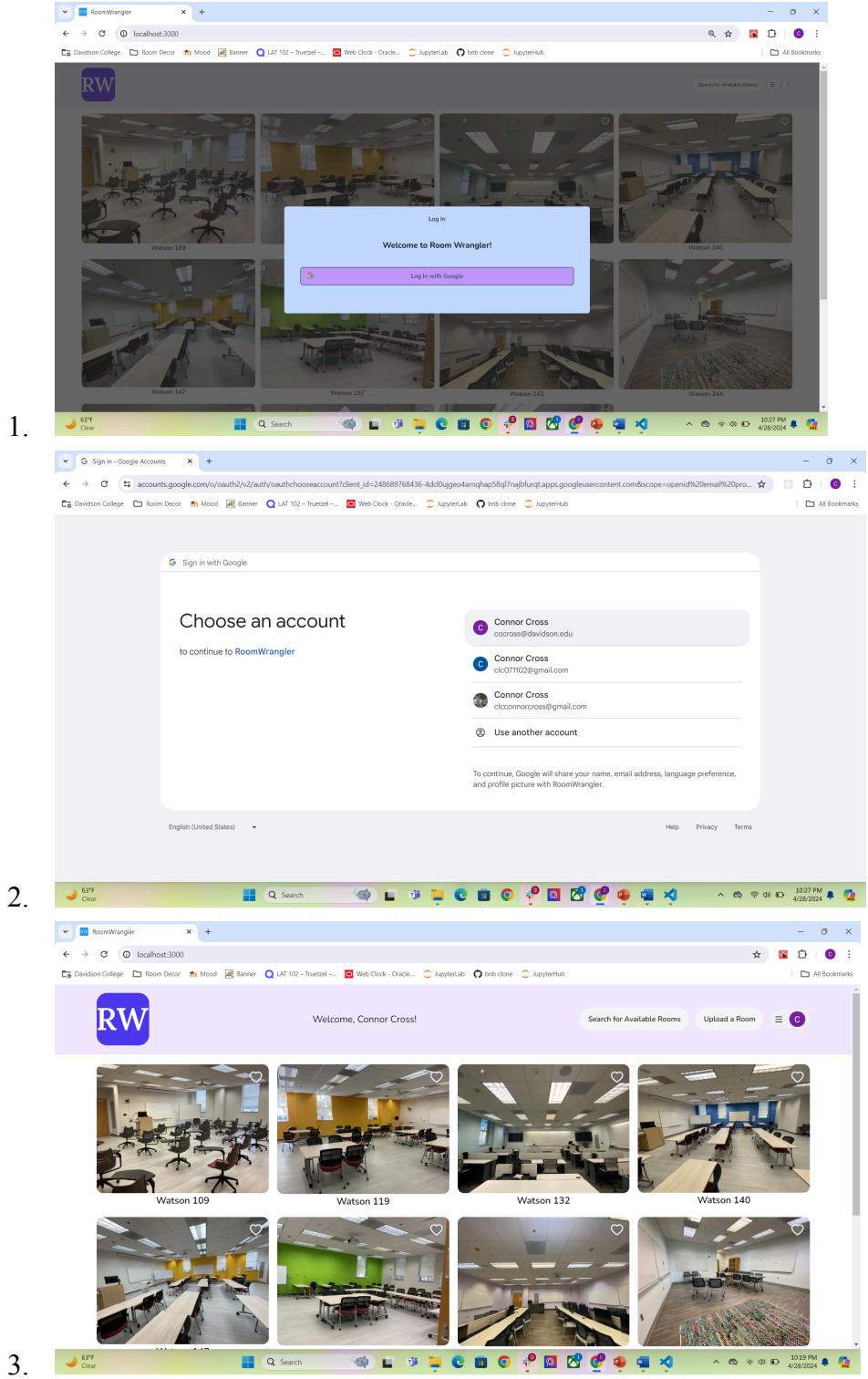
## **7. Current Status**

### *a. Summarize the current implementation status of your system.*

We have successfully implemented all targeted features in the current system. Our system includes a functional login page that utilizes the Google OAuth2 API for user authentication. If a user is not already registered, the system automatically creates a new user entry in our database. Our interface features a navbar that allows users to return to the home page by clicking an icon and also greets them with a personalized welcome message. The navbar provides options to search for rooms based on criteria such as time availability, room number, and the presence of projectors, whiteboards, and computers. For users with admin permissions, there is an added functionality to upload room details through a modal accessible via the navbar. The navbar also includes a user menu for navigating to various pages on our site. The home page displays all rooms from the database in their own cards, with options to mark rooms as favorites. The favorites page, accessible through the navbar, showcases all favorited rooms for a given user. Additionally, there is an upcoming reservations page that lists all future reservations linked to the user, providing options to cancel reservations as needed. Clicking on any room redirects the user to a detailed page about that room, where they can make reservations and view the room's schedule. Users with admin permissions, as well as elevated students and professors, can edit room details and set up recurring reservations for a semester. Exclusive to admins, there is a permissions page where they can manage user access levels by searching through all registered users and adjusting their permissions to either expand or restrict their access to the system.

### *b. Screenshots*

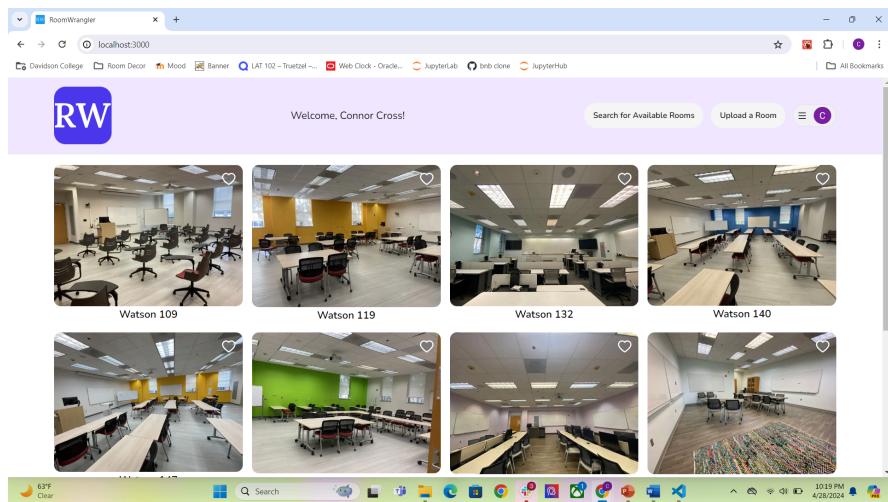
- i. *Add the screenshot(s) of the working system. Add the screenshots in the sequence of actions. This means that your screenshots will show you the flow of how the user interacts with the system. Think of it as how you would walk through the live demonstration during the presentation. Add the following details for each screenshot: 1. What are you trying to show in the screenshot? 2. Which part in the system description diagram does the part belong to? 3. What are the input and output of the system? 4. Why is this part of the system important?*
- ii. Login with Google API



- This sequence of screenshots displays the process of a user logging into our website. Initially, the user is forced to log in with an unclosable modal which prompts them to do so. It is located in the login modal section of our system description diagram. The input is handled through Google

OAuth, and after the user logs in we get their email and names from the OAuth API. The login modal is critical so we can create and store various users along with their reservations, favorite rooms, and permissions levels.

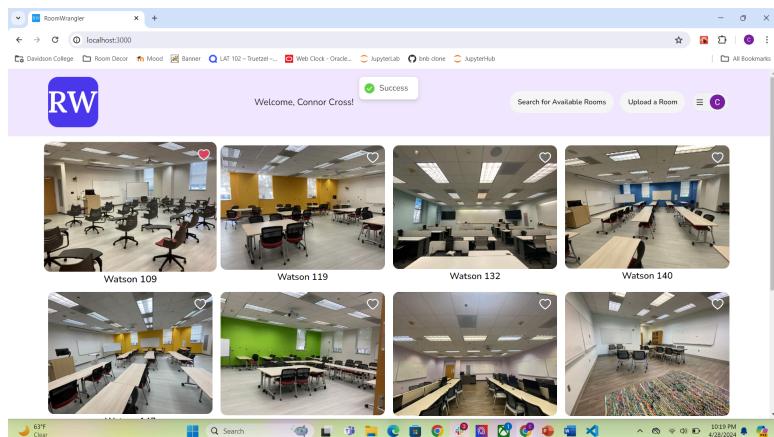
### iii. Home Page



1.

- a. Here is a view of the home page. It belongs on the home page part of our system description diagram. It can change its output based on input given in the search model along with the user favoriting various rooms, which causes it to update whether or not a room is favorited in the database. Its output (the room cards) are all pulled from the database, and thus it will remember whether or not a room is favored. The home page is critical to our success, as it's the first thing the user sees. We want a nice, intuitive layout so that the user has a positive experience on our platform.

### iv. Favoriting

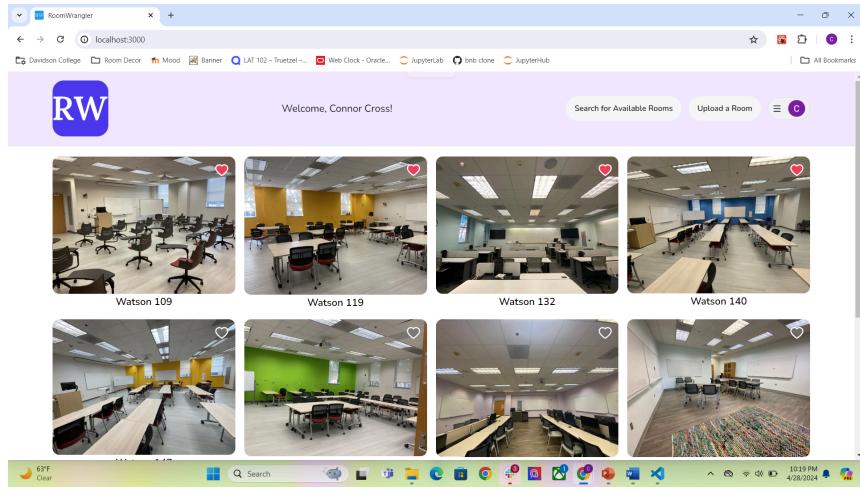


1.

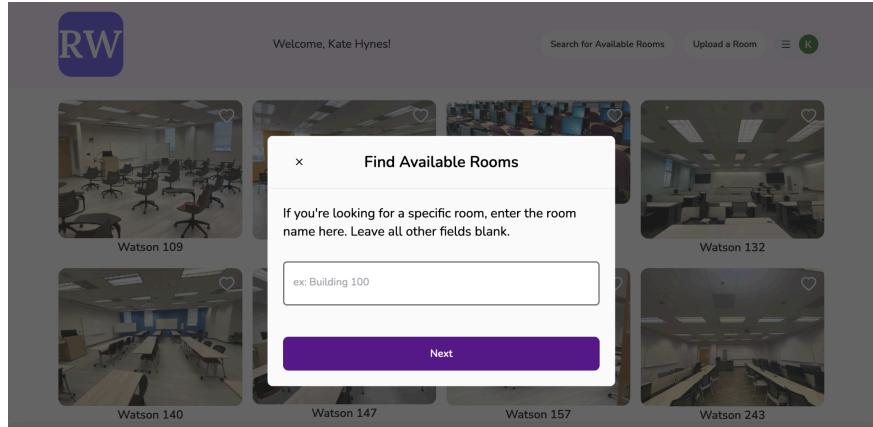
- 2.
- 
- 3.
- 
- 4.
- a. This sequence of pictures shows how to favorite a room on our website and how to access the “Favorite Rooms” page. It can be found in the “Favorites Page” part of the system description diagram. The input comes from the little heart button in the corner of each room card, which, once clicked will turn red, indicating that the user has favorited this room. We can see in picture *a* what it looks like for one

room to be favorited, and *b* shows multiple. *C* shows how to access the favorites page. Once the user has input his or her favorite rooms on the home page, he or she can see them displayed on the “Favorite Rooms” page. We thought this would likely be important so that people could have a way of easily accessing rooms where they may have recurring reservations.

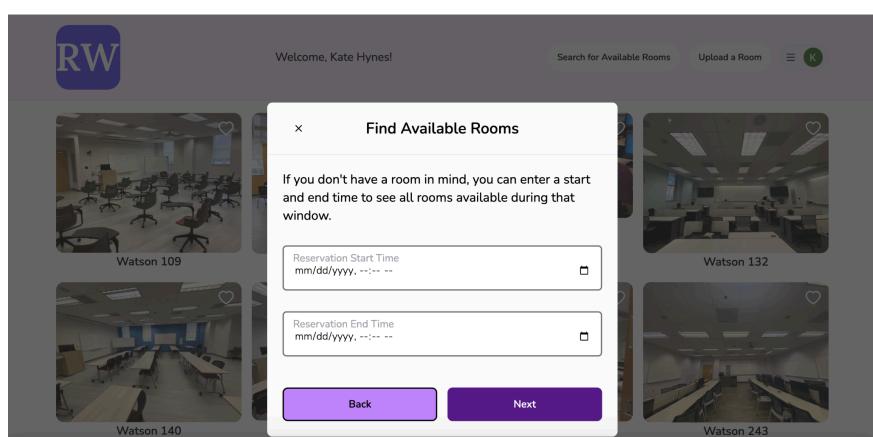
## v. Search



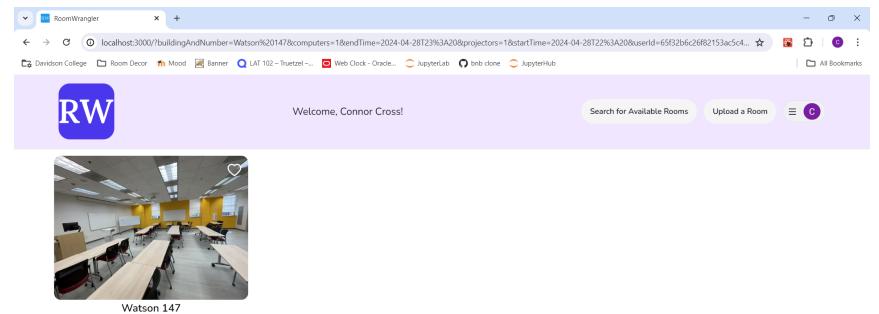
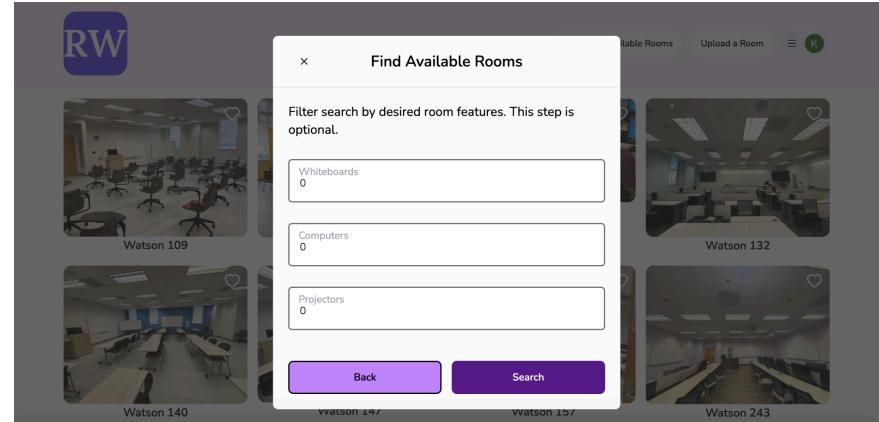
1. A screenshot of the RoomWrangler application interface. The top navigation bar shows 'RoomWrangler' and the URL 'localhost:3000'. Below the navigation is a purple header with the letters 'RW'. The main content area displays a grid of eight room thumbnails, each with a small red heart icon in the top right corner. The rooms are labeled Watson 109, Watson 119, Watson 132, Watson 140, Watson 141, Watson 142, Watson 157, and Watson 243. At the bottom of the screen is a Windows taskbar with various pinned icons and the date/time '4/28/2024 10:19 PM'.

2. A screenshot of the RoomWrangler application interface. The top navigation bar shows 'RoomWrangler' and the URL 'localhost:3000'. Below the navigation is a purple header with the letters 'RW'. The main content area displays a grid of room thumbnails. A modal dialog box titled 'Find Available Rooms' is open in the center. It contains instructions: 'If you're looking for a specific room, enter the room name here. Leave all other fields blank.' Below this is a text input field with the placeholder 'ex: Building 100'. At the bottom of the dialog is a purple 'Next' button. The background grid of rooms includes Watson 109, Watson 140, Watson 141, Watson 142, Watson 157, Watson 132, Watson 243, and Watson 147.

3. A screenshot of the RoomWrangler application interface. The top navigation bar shows 'RoomWrangler' and the URL 'localhost:3000'. Below the navigation is a purple header with the letters 'RW'. The main content area displays a grid of room thumbnails. A modal dialog box titled 'Find Available Rooms' is open in the center. It contains instructions: 'If you don't have a room in mind, you can enter a start and end time to see all rooms available during that window.' Below this are two input fields: 'Reservation Start Time' (placeholder 'mm/dd/yyyy, --:-- --') and 'Reservation End Time' (placeholder 'mm/dd/yyyy, --:-- --'). At the bottom of the dialog are two buttons: a purple 'Back' button on the left and a purple 'Next' button on the right. The background grid of rooms includes Watson 109, Watson 140, Watson 141, Watson 142, Watson 157, Watson 132, Watson 243, and Watson 147.



5.

- a. These pictures show the process of searching for a room. First, the user begins from the home page, pressing the “Search Available Rooms” button. This opens the search modal, which is labeled as “Search Rooms Modal” in our system description diagram. From there, as shown on pictures, *b,c*, and *d*, the user has the option to search for a specific room, a specific time, and specific properties of the room such as number of whiteboards, computers, or projectors. If the user searches for a specific room, then they can bypass the time window. However, if the user does not have a specific room in mind, then we require that a specific time be entered. From this input, we generate the output, which is a list of rooms matching the search criteria. We thought this would be important to facilitate the user’s experience in finding a room which best suits his or her needs and availability.

vi. Upload Room

1.

**Upload a Room**

Select the building and enter the room number.

Wall  
 Chambers  
 Watson  
 Library

Room Number  
100

Next

**Upload a Room**

Add additional information about the room.

Floor  
8

Whiteboards  
7

Computers  
6

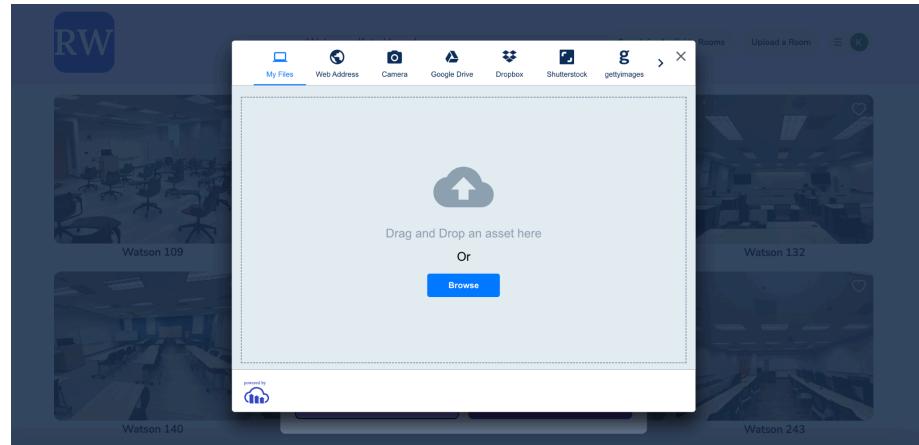
Projectors  
5

Capacity  
4

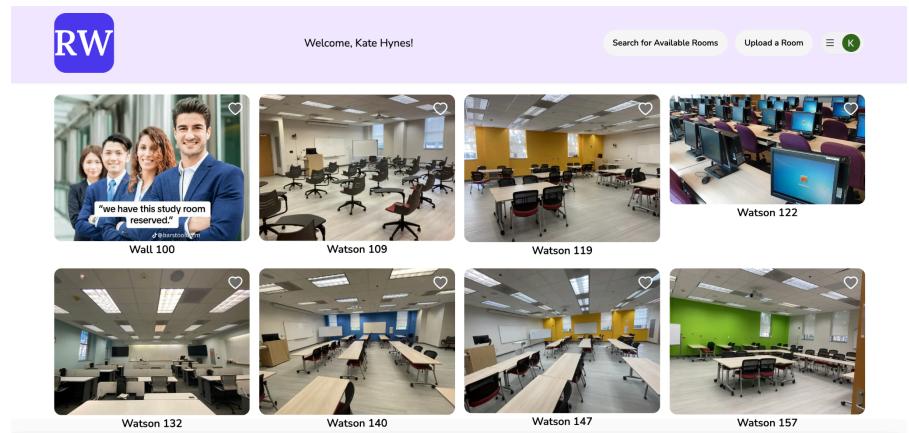
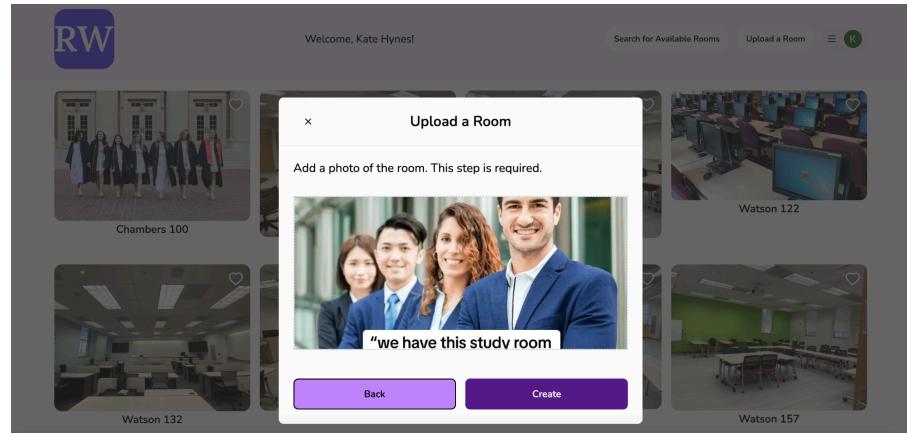
Back Next

Chambers 100  
Watson 132  
Watson 122  
Watson 157

Watson 157



3.

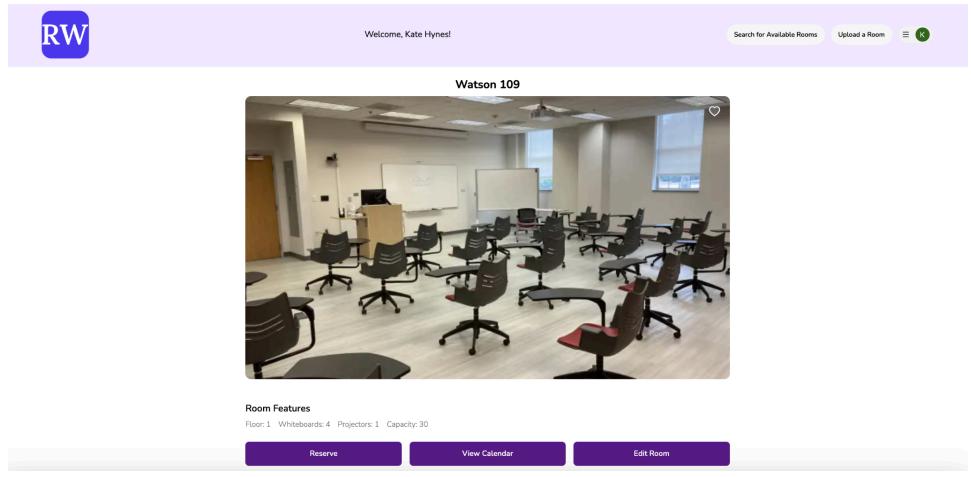


4.

- These screenshots display the process of uploading a room in RoomWrangler. It can be found in the “Upload Room” section in the system description diagram. The user begins by clicking the “Upload Room” button on the navbar (only available to admins), and is then prompted to input all the information for the room. We used the Cloudinary API to allow for uploading images to our website. This input then gets sent to the database, and is output onto the homepage

once it has been sent to the database, as we can see in picture f. We thought this would be important because it makes it easier for institutions to add new rooms.

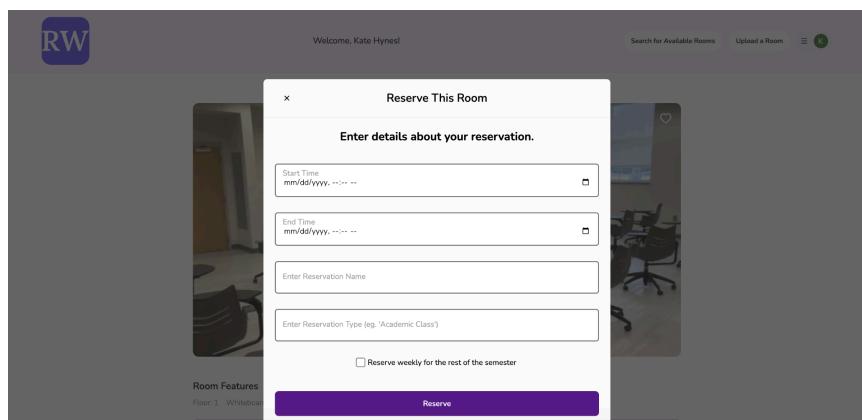
#### vii. Room Page



1.

- a. By clicking on any of the rooms on the home page or favorites page, the user can access the room page. It can be found in our system description diagram in the “Room Client” box. From the user’s input of selecting a room on the homepage, the database is queried to produce the output depicted in the picture above for a given room. From here, the user can reserve the room and view the calendar to see when it may be available. Admins also have the ability to edit a listing on this page. We thought this page would be important because we wanted to have some space with more details on the room where the user could perform more actions related to that specific room.

#### viii. Reserve Room

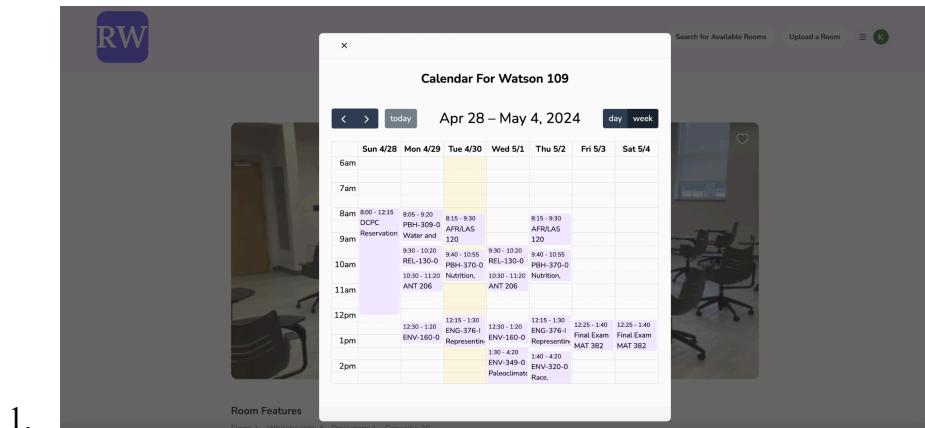


1.

- a. This screenshot shows the modal for reserving a room, found in the “Reserve Room” box of the system description

diagram. The user can select a time and input a reservation name and type, which gets posted to the database. Elevated students, professors, and admins can also make recurring weekly reservations. Its output is that other users then cannot reserve over this time, and it is also displayed on the calendar. This modal is, somewhat self-evidently, critical to our platform, as it's not a room reservation platform if you don't have the ability to reserve a room.

#### ix. Calendar View



1.

- a. This screenshot shows the calendar page, which is in the “Calendar” box in the system description diagram. It displays some weekly reservations, as mentioned in the number 7. There is no input for the calendar, and it’s generated from the reservations stored in our database. It also has various options so that the user can view one day at a time or a week at a time. The user can also navigate from one week to another using the arrows in the top left of the screen. We thought this would be really important, as it helps users make an informed decision on when they can rent a room.

#### x. Edit Room Information

1.

**Edit This Room**

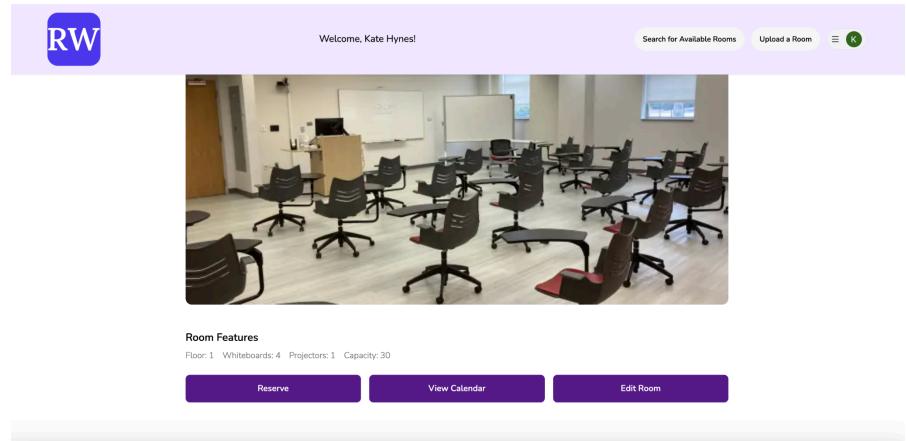
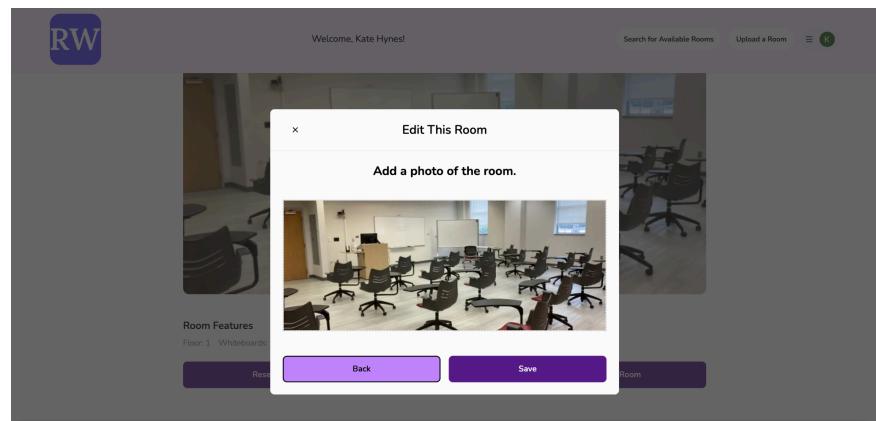
**Update Room Info**

Floor	1
Whiteboards	4
Computers	0
Projectors	1
Capacity	30

Room Features  
Floor: 1 Whiteboards

Reserve View Calendar Edit Room

Next



3.

- This sequence of screenshots shows the process of editing room information for rooms which have already been uploaded to the database. It can be found in the “Edit Room” box in the system description diagram. The inputs it takes are for changing the number of projectors, whiteboards, computers, capacity, and the image displayed for the room. It then sends these inputs to the database, and regenerates the output based on the updated room. This functionality is only available to admins. We thought it