Project 5:

Design

Simplified Sckhaedgeuler (SS)



Snow College

*Generation of a Prototype*

Leedan Johnson, Alex Thayn, Jackson Porter

Table of Contents

[**1. System Concept Statement**](#_8jrhi7g83wro) **3**

[**2. What We Included**](#_8lxcxr7bhfqy) **4**

[**3. Process of Building**](#_1jnk6la5ldhe) **4**

[**4. The Prototype:**](#_nvgrq0mpisfi) **7**

[System User - Create a room reservation](#_d3rotp31nkos) 7

[Manager - Approve Request to Override Reservation](#_ejg2lmxcteoy) 10

[Manager - Review/Resolve Room Feedback](#_y1u8lcpwlyc1) 11

[System Admin - Override with a Paid Room Reservation](#_q6hqudjncei4) 13

[Pilot Testing Results:](#_k49kdpi5rwhh) 17

[Alex Pilot Test](#_2v5mqbcuvldd) 17

[Student Test](#_f1s1zso3qa0r) 17

[Building Manager Test](#_z32ifk5kcsdh) 18

[System Administrator Test](#_deu6bupdz5tr) 19

[Leedan Pilot Testing](#_dz9v3ed0acis) 20

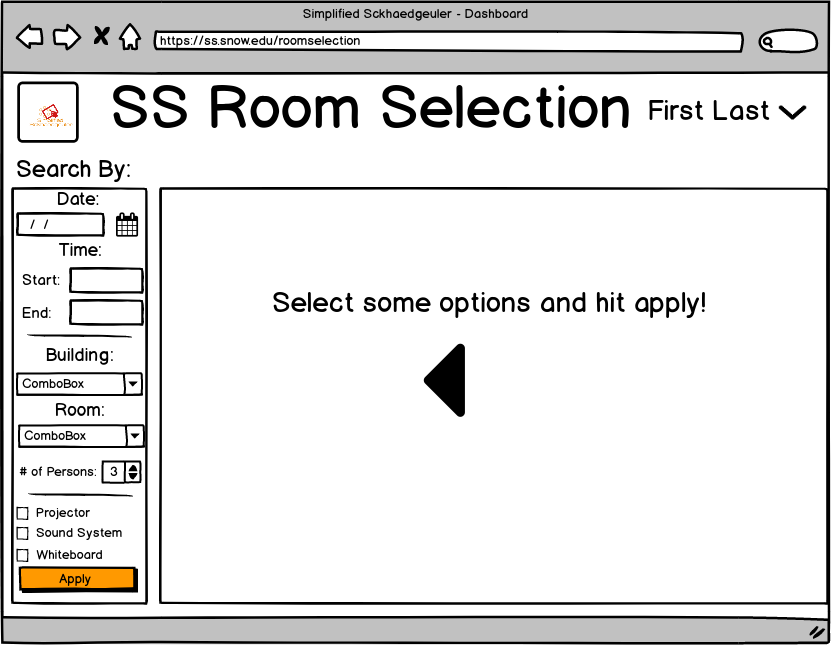
[Student User test](#_9nyypfxkhf40) 20

# 1. System Concept Statement

The *Simplified Sckhaedgeuler* provides an all-around solution to room scheduling for the Snow College campus. From students to building managers our software adapts to fit each user’s needs:

* Experience a consistent and easy to use web interface.
* Finding a room for last minute study sessions is no longer a problem.
* Searching for the “right” room has been simplified, users can filter search results to find the room of their dreams.
* Academics take priority! No more getting kicked out of your reservation.
* Managers will maintain control of each building and receive up-to-date feedback about room issues from users.
* View and report issues with a room instantly.
* Never forget a meeting again! Sign up for notifications to receive reminders about your upcoming reservations and events to which you’ve been invited.

Our software helps all overcome the challenges of traditional room management, allowing the process to be effortless and enjoyable.

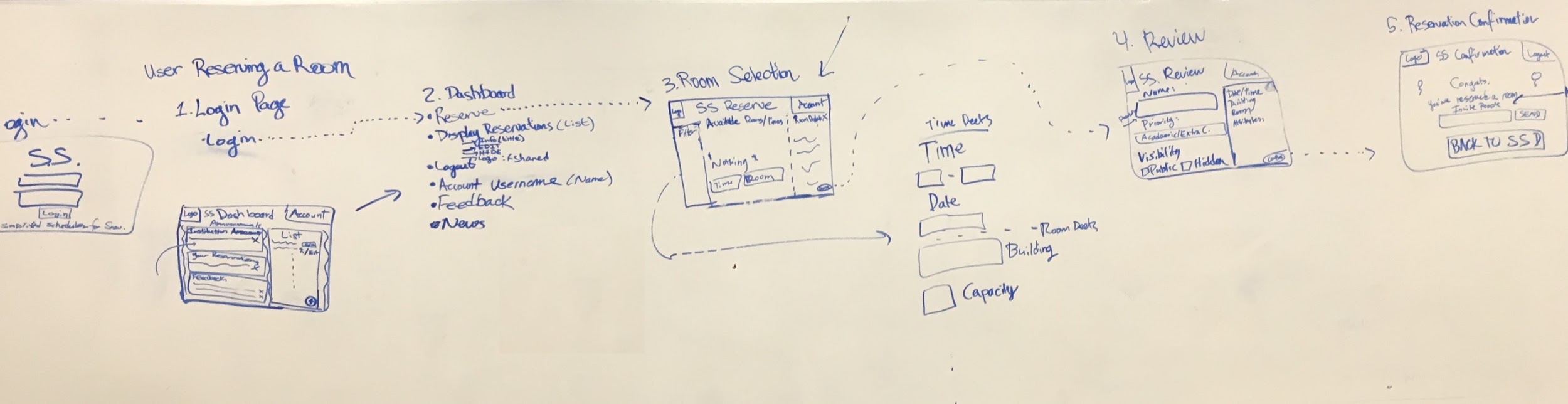


# 2. What We Included

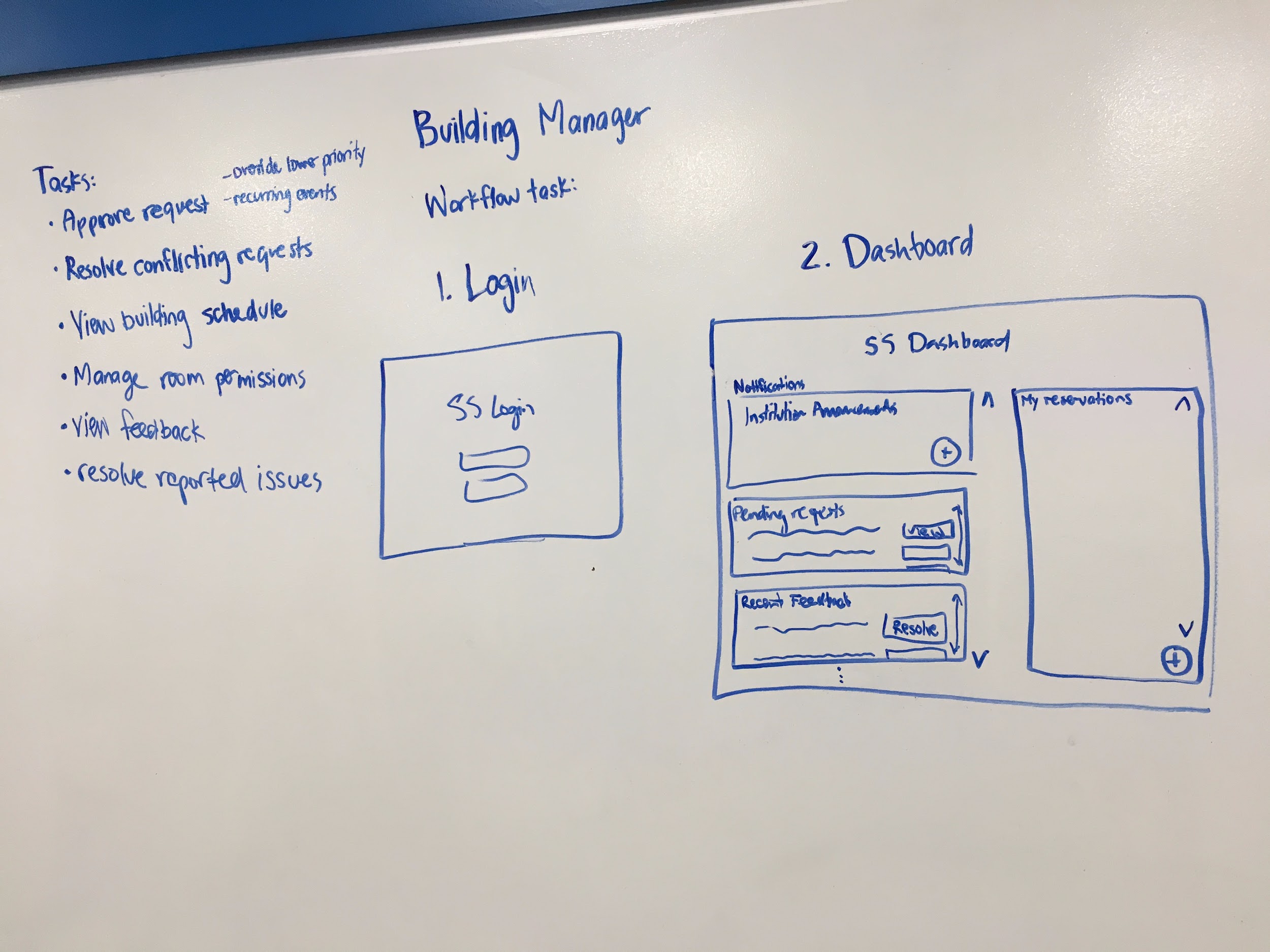
For this project, our group met many times and talked together about the different scenarios that we could have our users go through. We decided to include the following items (as seen in step 3) within our project as we felt that it was the most relevant and most appropriate to our project. The scenarios and tasks that we chose to wireframe were based off of needs and concerns of our interviewees and workshop participants in our earlier projects. This allows them to see that our product does indeed solve these issues they needed solved.

# 3. Process of Building

This project was different than any of the projects we have done before, and required the inclusion of all previous activities up to this point. This project is where all the ideas from previous project start to come together into a single idea. To start out this project, we took ourselves to a whiteboard and began to create a workflow for each persona. We began by drawing ideas for our wireframes on the board and iterated through many different versions before coming to a common consensus. We focused on one main task for each persona and drew a quick idea of what the wireframes would look like for each persona. We wanted to keep the interface as consistent as possible between the users, so we reused many of the same concepts and layouts throughout all the wireframes we created.



For the system user persona the task we decided to represent was the process of reserving a room. We drew the process on the whiteboard (shown above) and then took those ideas and used a wireframing software called balsamiq to turn those sketches into wireframes. Leedan also created an html mockup of the website for the student persona, to allows us to see what the system would look like in a web browser. [Link](http://www.ljdemosites.x10host.com/SS/)

For the Building Manager we went through a similar process, the task we decided to focus on for this persona was to resolve conflicting reservations and to view and resolve room feedback left by users. We created wireframes to represent these workflows with balsamiq software. When users leave feedback about issues they had while using the room, it automatically pops up in the building managers notification feed and they can view the feedback straight from the dashboard. Then they will either move the issue to pending or resolve it immediately depending on what the issue was.

For the System Administrator persona we chose to give this person the task of resolving a conflict that occurs between a newly created outside business reservation that would need to override an existing reservation. The admin is in charge of resolving conflicting room requests and we created the wireframes that allows the admin to go through this process. For an outside business to make a reservation they must do it with the system admin and when the admin creates a new reservation that conflicts with an existing one, the software will automatically detect this and allow them to decide which reservation to keep.

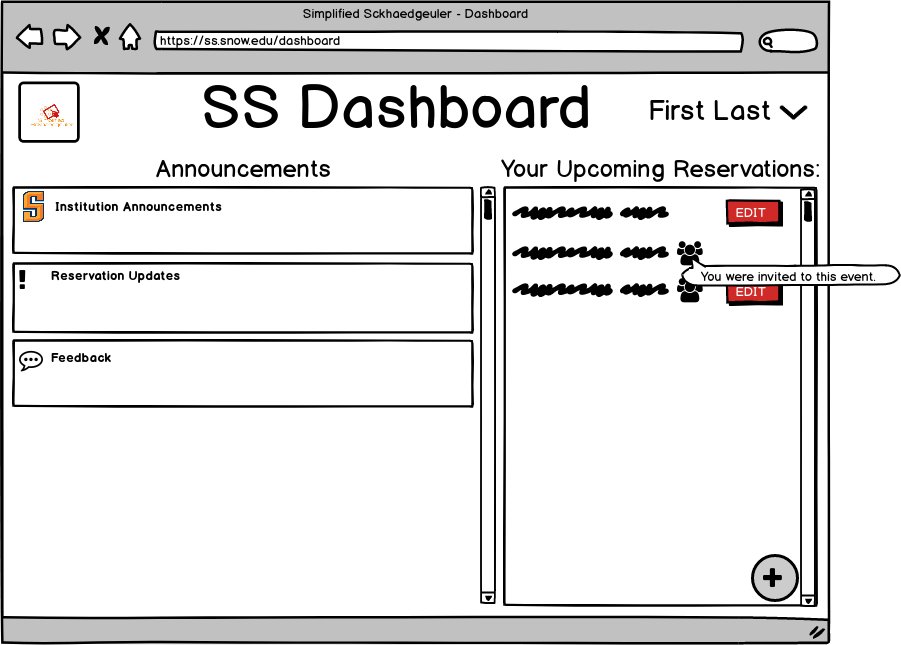
# 4. The Prototype:

Below we have included our wireframes and described the flow between each wireframe. There are 3 workflows included for each person: system user, building manager (2 tasks shown here), and system admin.

## System User - Create a room reservationLogin Page.png

This is the Login page. It is the first thing that the user will see when attempting to use the Simplified Sckhaedeuler. This will be the same for all users of the system. Users will enter their credentials, and be taken to the system dashboard, which will vary by type of user. (Administrator, Manager, or Student)

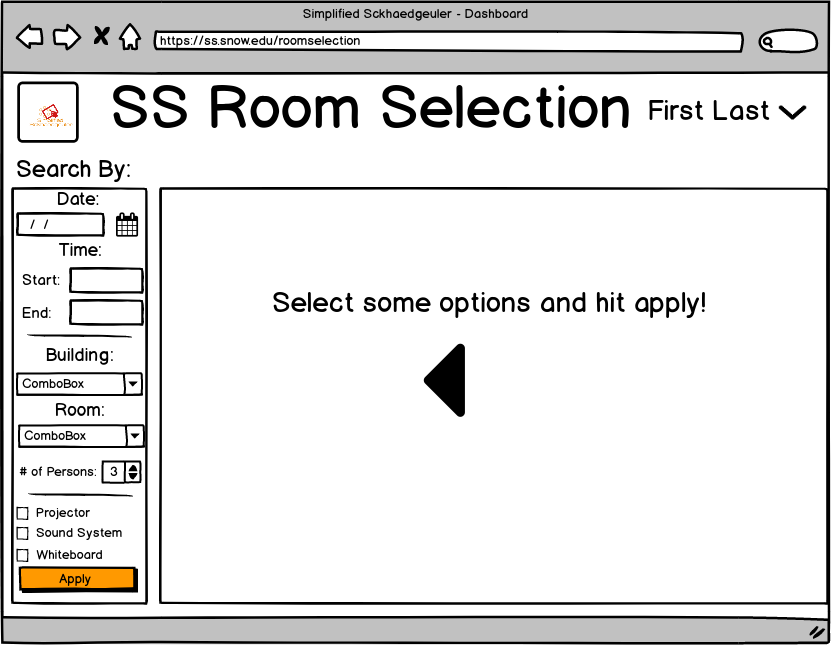
Functionality is also included for resetting/recovering a lost username/password.



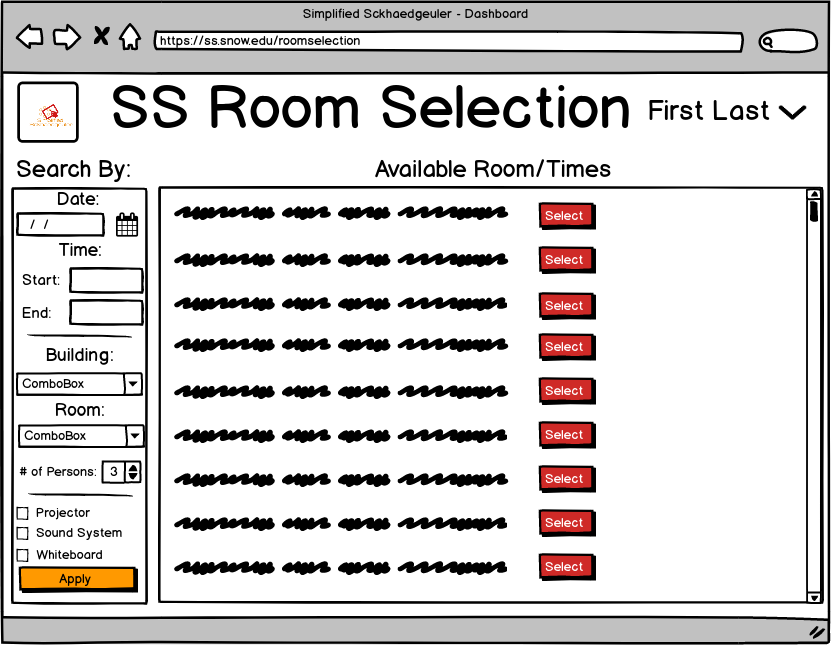
This is the student dashboard. On the left students can see notifications from the organization, updates about their reservations, and be prompted to give feedback regarding previous reservations.

On the right students can see any upcoming reservations, both those they have made, and those they have been invited to.

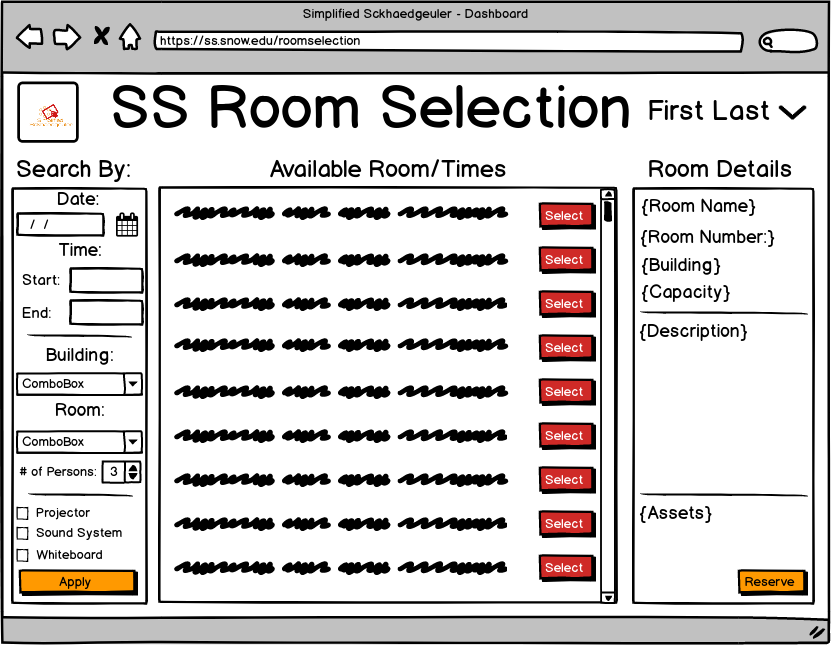
By clicking the ‘+’ icon, they can add a room reservation.

In the upper right corner, they can click on and manage profile info.

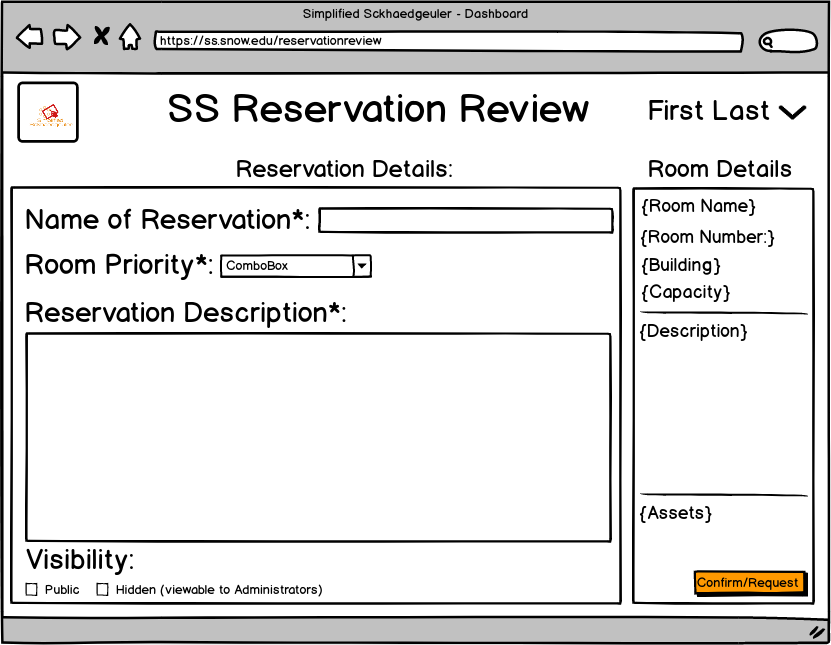
This page is accessed by the student user upon clicking the ‘+’ icon from the dashboard. By filling in the various fields, on the leftmost pane, the user can search for available rooms. Based on the fields entered, the results pane will show different results.



This page shows the results after the user has made various selections in the filter pane. The user may now select from the list of results to view room details, or add filters and click apply once more.



This page shows the results of selecting one of the rooms resulting from the filtered search. Details are now shown on the details pane, and the user has the option to reserve the selected room.

This page results from the user selecting “Reserve”: from the previous page. The user will now enter a name for the event/reservation, select the type of event (thus determining its priority) and optionally a short description of the reservation/event. 

The user will also be given the opportunity to mark the event details as private, and they will no longer be visible to other users.

The reservation can then be finalized. (Submitted or requested as appropriate)

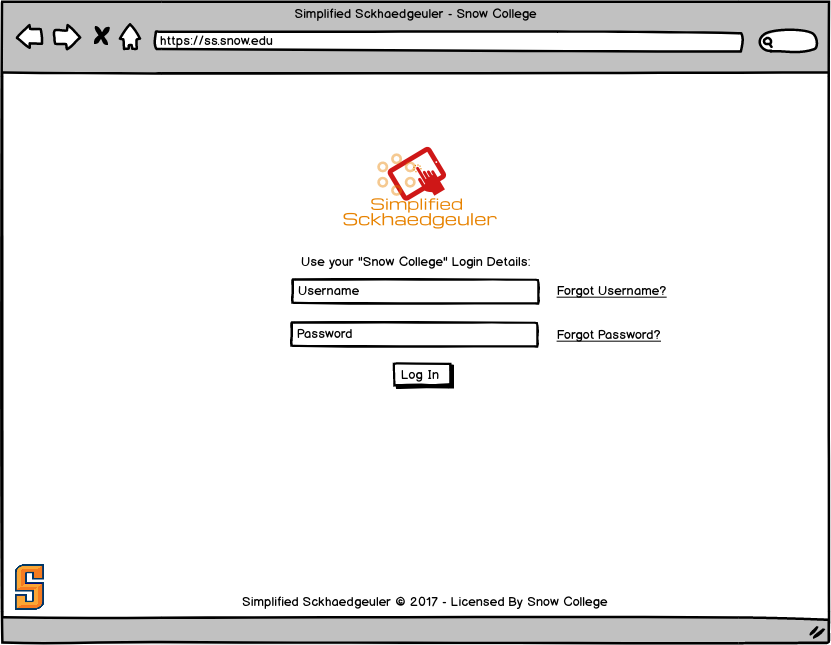


After having selected confirm or request on the previous page, the user is taken to this page. Here the user is prompted to check their email to view the room confirmation, they can also enter the email addresses of individuals they would like to invite to their reservation. They also have the option to go directly back to the dashboard, or logout.



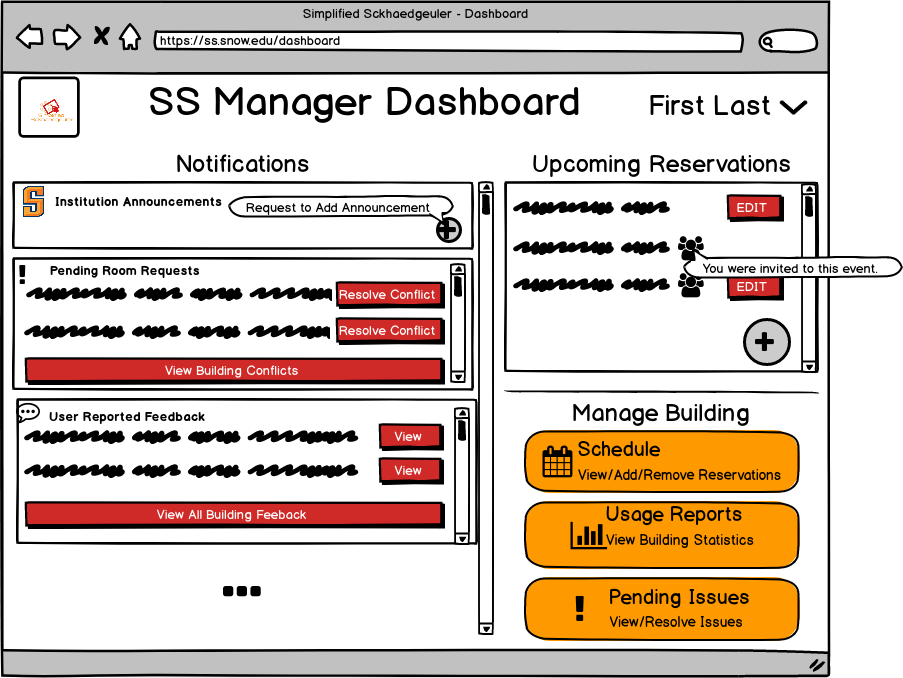
If the user has chosen to invite others to the reservation, they are taken to this page after entering their emails. The user can now choose to return to the dashboard, or logout.

## Manager - Approve Request to Override Reservation

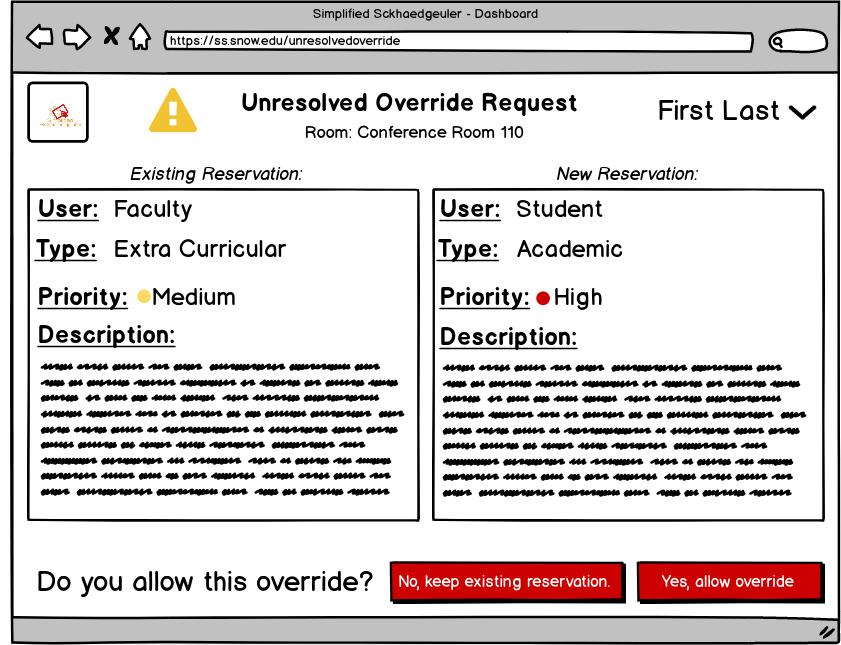


This is the Login page. It is the first thing that the user will see when attempting to use the Simplified Sckhaedeuler. This will be the same for all users of the system. Users will enter their credentials, and be taken to the system dashboard, which will vary by type of user (Administrator, Manager, or Student).

Functionality is also included for resetting/recovering a lost username/password.

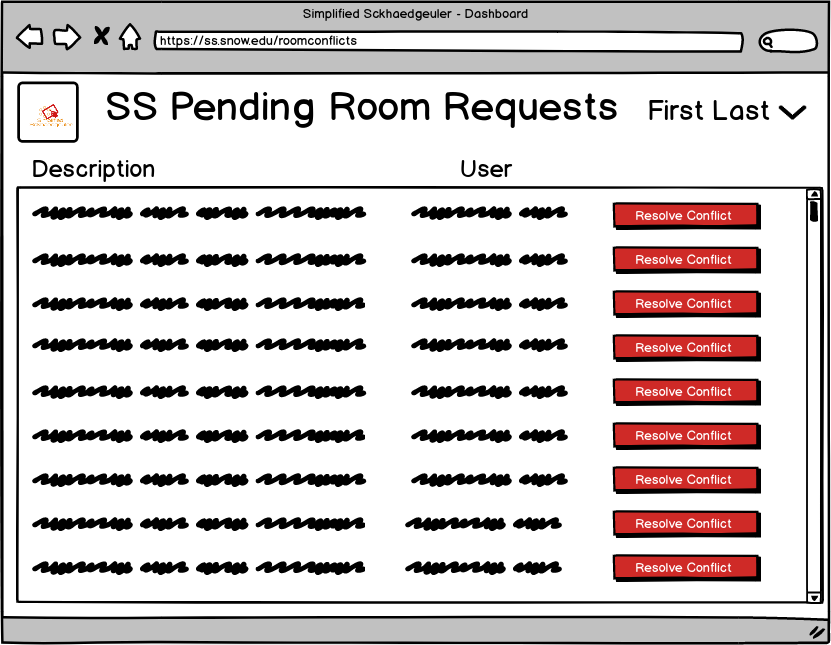


This is the Manager dashboard page. On the left notifications are shown, including institution announcements and a button to make a request for an announcement to be made (needs system admin approval). Requests made by users and awaiting approval are also shown, as well as feedback given by users about rooms.

On the right the manager can see upcoming reservations, and see links to various manager activities, including scheduling, reports and pending issues.  


From the dashboard, if the manager has clicked to resolve a specific conflict in pending room requests, they are taken directly to the conflict resolution page for that specific conflict.

The conflict resolution page displays the conflicting reservations, and allows the manager to select whether the existing reservation should be kept, or overridden by a new reservation.



If the manager selects the generic “View Building Conflicts” from the dashboard, the manager is taken to a page showing all conflicts for the building, and by selecting one of the conflicts, is then taken to the conflict resolution page (see last wireframe)



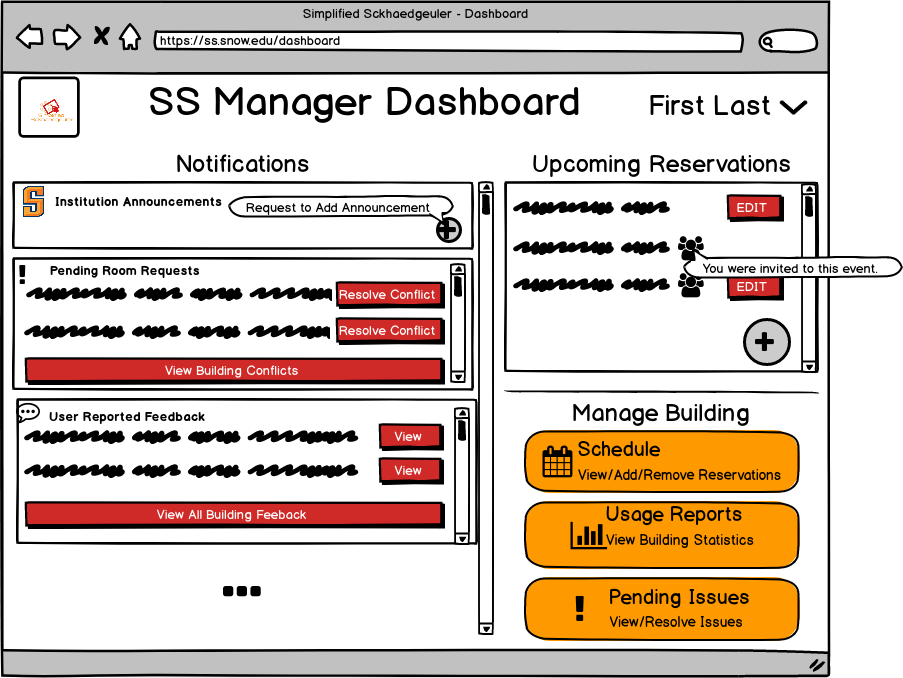
This page is displayed after a manager had resolved a conflict. By selecting “Resolve Others?” the manager is taken to the page showing all conflicts in the building. The manager can also choose to return to the dashboard, or logout.

## 

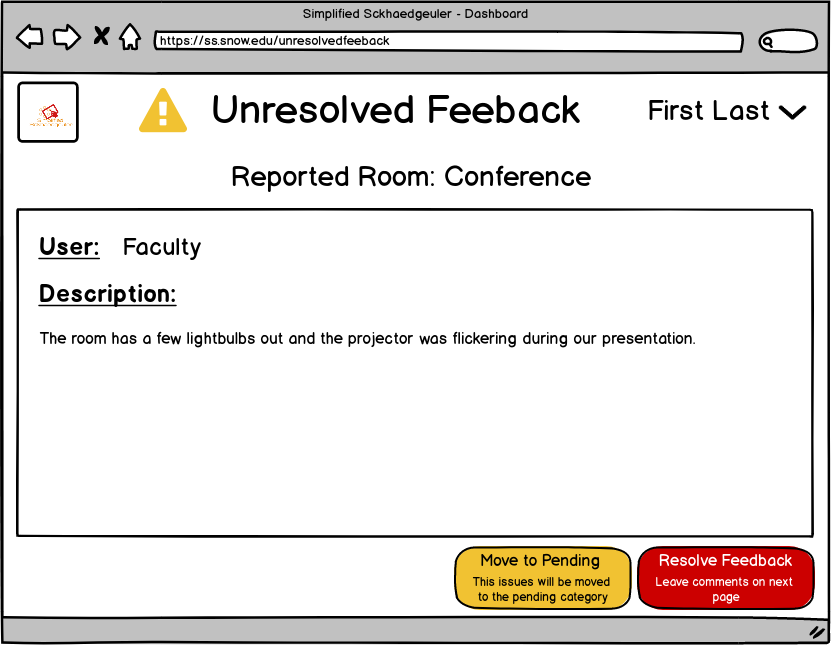
## Manager - Review/Resolve Room Feedback1.png

This is the Login page. It is the first thing that the user will see when attempting to use the Simplified Sckhaedeuler. This will be the same for all users of the system. Users will enter their credentials, and be taken to the system dashboard, which will vary by type of user. (Administrator, Manager, or Student)

Functionality is also included for resetting/recovering a lost username/password.

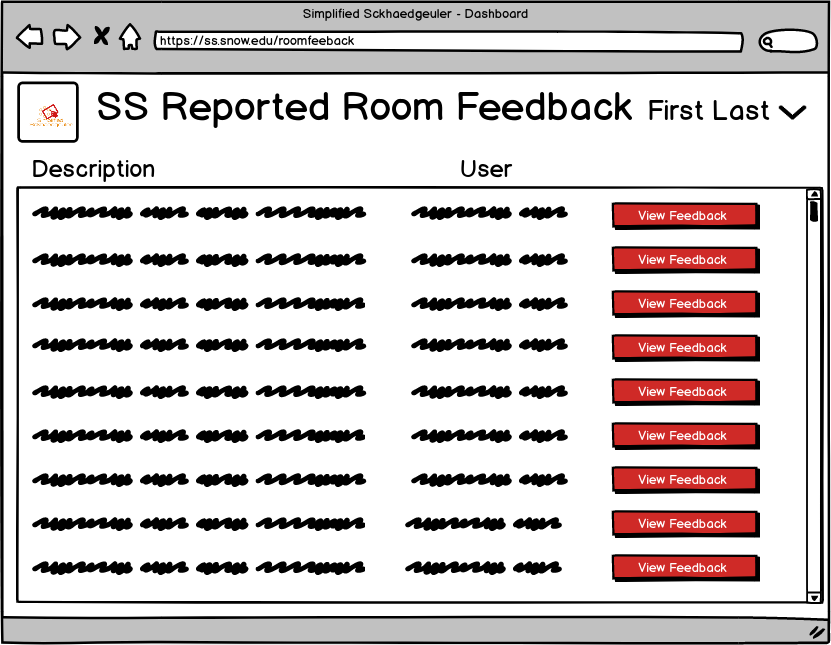


This is the Manager dashboard page. On the left notifications are shown, including institution announcements and a button to make a request for an announcement to be made. Requests made by users and awaiting approval are also shown, as well as feedback given by users about rooms.  
On the right the manager can see upcoming reservations, and see links to various manager activities, including scheduling, reports and pending issues.

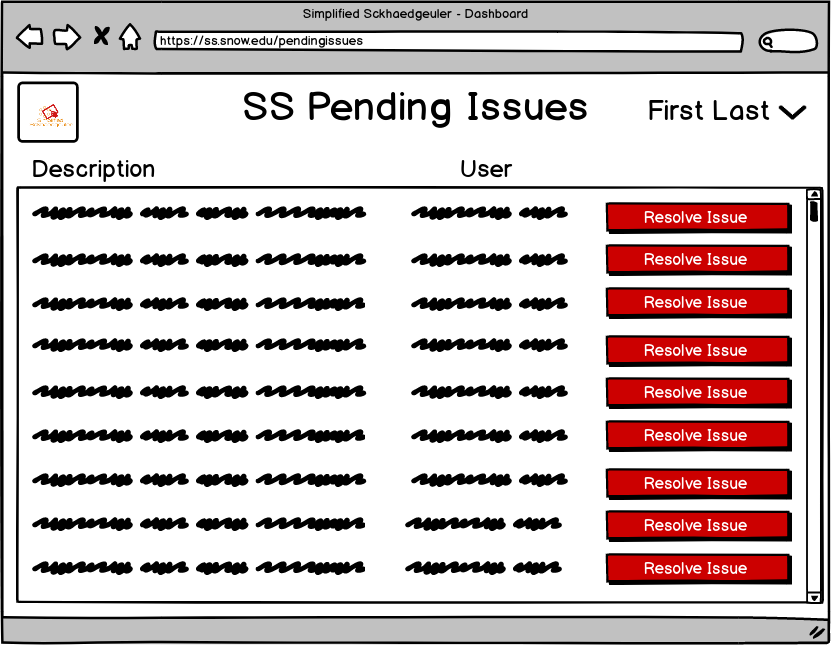


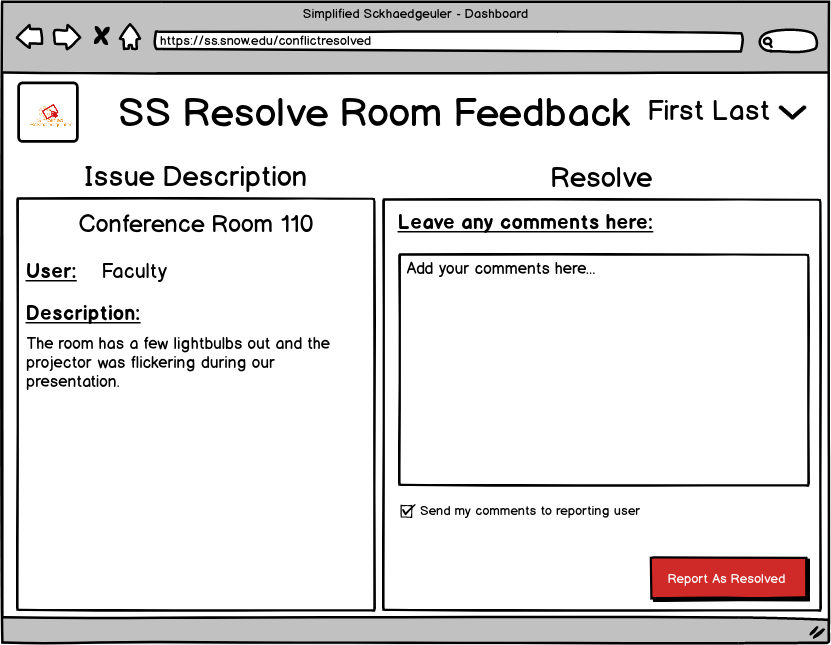
From the dashboard, if the manager has selected ‘view’ from the user reported feedback section, the manager will be taken directly to the unresolved feedback page.

From the unresolved feedback page, the manager can see all details regarding the report, and choose to either move the issue to pending(viewed but not yet resolved), or mark the issue as resolved.

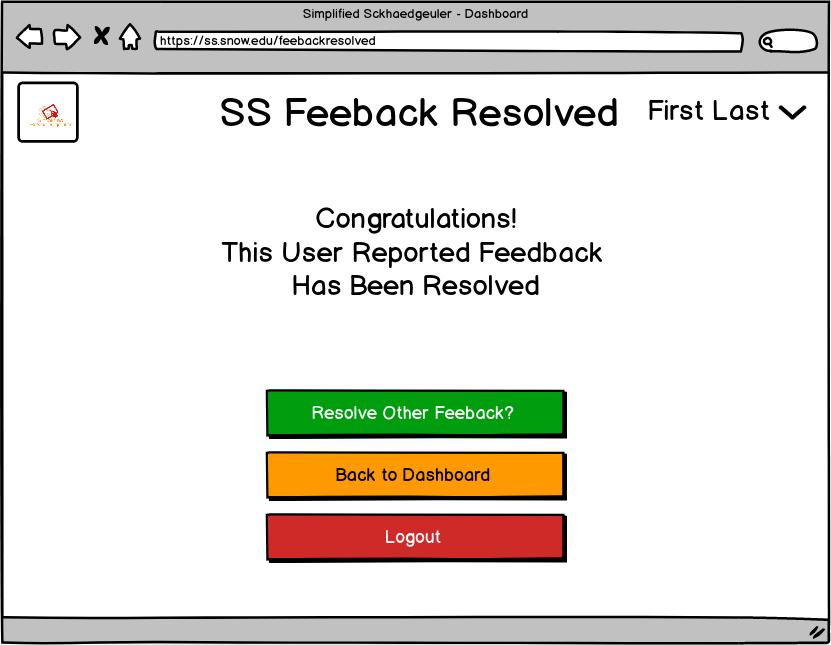


If the manager selects the “View all building feedback” link on the dashboard, they will be taken to a page showing all feedback. The manager can then select a specific feedback report, and then be taken to the unresolved feedback page (see last wireframe).

From the dashboard by clicking “Pending Issues” or by clicking “Move to Pending Issues” on the Unresolved Feedback page. It will display all pending issues. 



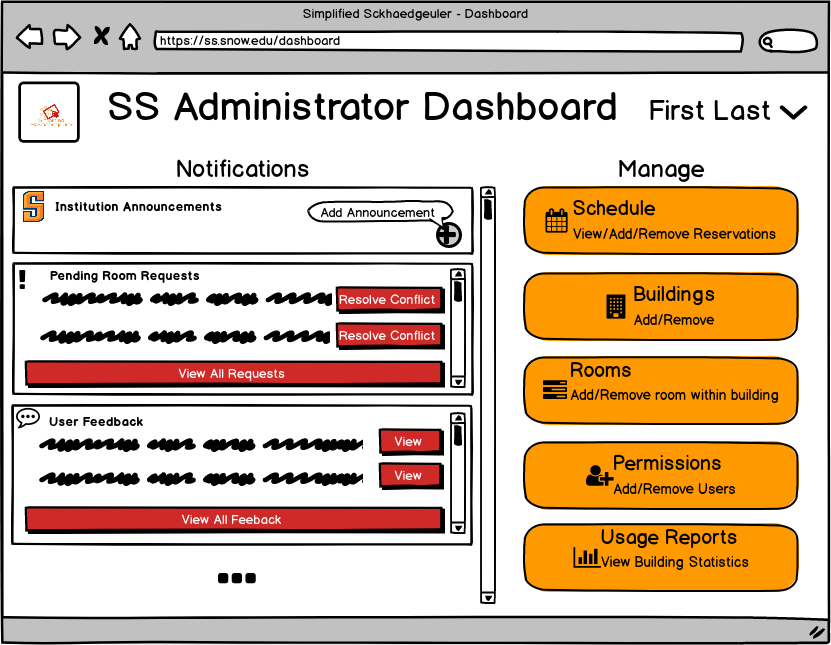
From the Unresolved Feedback Page or the Pending Issues pages, the manager can select the issue, view all information about the issue, write any comments about the issue and/or it’s resolution (optionally sending these comments to the reporter of the problem) and finally mark the issue as resolved.



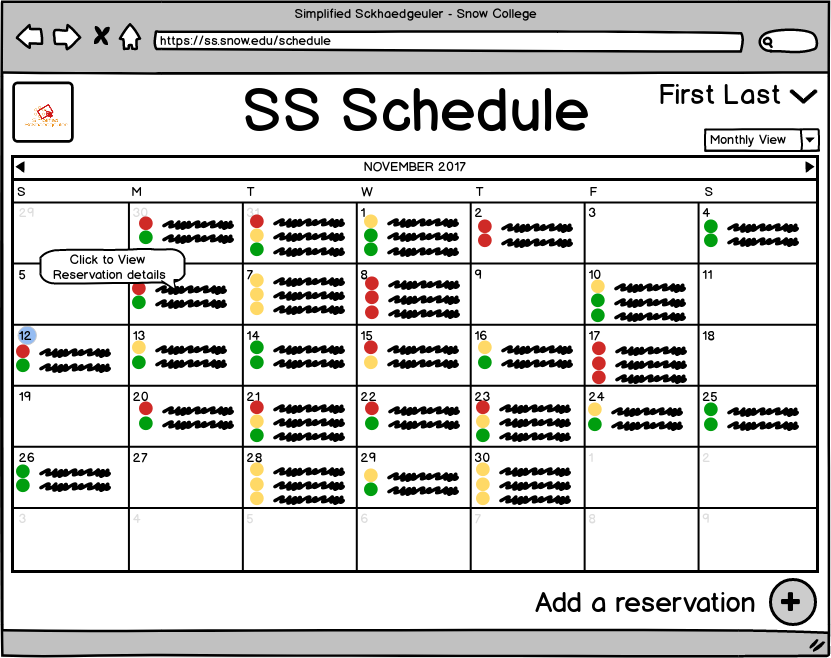
The manager is then taken to the feedback resolved page, where they are presented with the options to resolve other feedback, return to the dashboard, or logout.

## System Admin - Override with a Paid Room Reservation1.png

This is the Login page. It is the first thing that the user will see when attempting to use the Simplified Sckhaedeuler. This will be the same for all users of the system. Users will enter their credentials, and be taken to the system dashboard, which will vary by type of user. (Administrator, Manager, or Student)  
Functionality is also included for resetting/recovering a lost username/password.

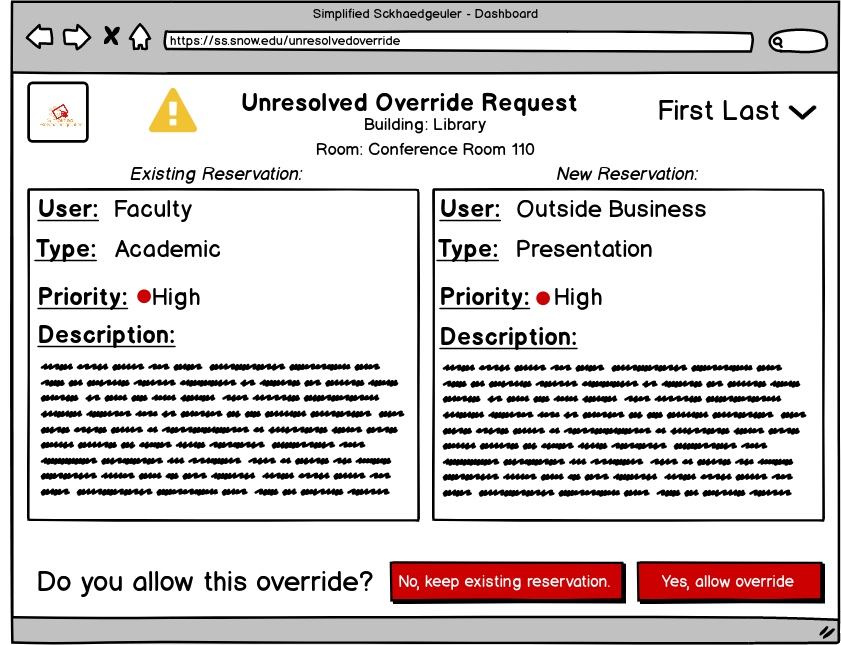


This is the System Admin dashboard. From here the admin can see and add institution announcements, resolve room requests and feedback like managers can, and select from admin tasks such as managing the global schedule, manage buildings, and rooms, manager permissions, and see usage reports.

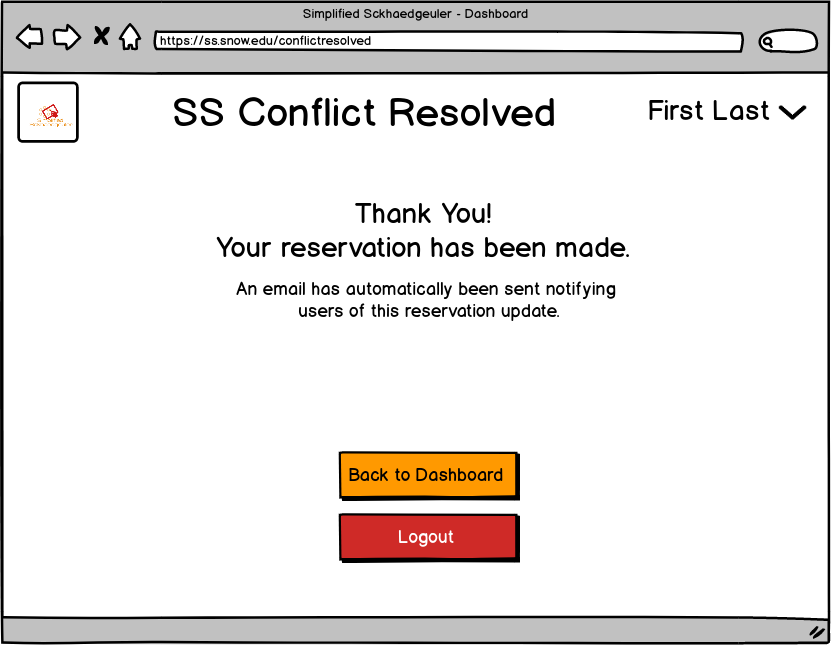


By selecting the “schedule option”, the admin is taken to the schedule page and can see a calendar with all current reservations (The view of this calendar can be changed using the dropdown). The admin can click on a reservation to see it’s details. The admin can also add a reservation.

**Note:** Between this last wireframe and the following, the process is *identical* to how a system user would reserve a room so we did not include all these extra wireframes here. Just imagine we had the system admin add a reservation for an outside business, just like the student added a reservation.



If the scheduled reservation is in conflict with an existing reservation, the admin is taken to the conflict resolution page. The admin can then review the information about both reservations, and choose whether or not to override the existing reservation.



The admin is then taken to the conflict resolved page, and choose whether to return to the dashboard, or logout.

5. Pilot Testing Specifications:

* 1. For our pilot test we created three different scenarios.
     1. For the first scenario, we created a scenario for a “System User.” System users are basic users that have basic functionality of the system (to make reservations). We asked the tester in this case to make a reservation of any type they wanted. A simple reservation confirmation/request is what we were looking for.
     2. For the second scenario, we have the Manager. A manager could be over a building, or a registrar. The manager has two different tasks that they could complete.
        1. First, the manager would process feedback given by those after reserving a room. The manager would have to decide to move the feedback to pending issues, or to mark the feedback as resolved depending on the severity of the feedback.
        2. In the second option, the tester would need to resolve a room conflict/override request. The tester needs to find the request to override a reservation and decide which to keep based on priority and if the request is legitimate.
     3. For the final scenario, we had the System Administrator. The System Administrator has all access to the system and resolves upper conflicts and requests made by managers and system users. The tester in this scenario will place a room reservation for a paid outside user of the institution over another appropriate reservation at the institution currently.

## Pilot Testing Results:

### Alex Pilot Test

#### Student Test

[Video Link](https://youtu.be/8jqyFBFHucU)

I (Alex) conducted a quick pilot test with of my 3 family members. For the first test I had my younger sister, Jayda, pretend she was a snow college student and go through the task of reserving a room. These were the directions she was given:

**Student Directions:**

Task description: As a snow college student you need to reserve room for your group study project.

1. Login to the scheduling website
2. Add a reservation
3. Select room options and apply search filter
4. Select a room
5. Reserve that room
6. Confirm reservation
7. Invite friends
8. Logout

Even with this quick pilot test I was able to gain some valuable insight into the user’s mind. She commented on different things that were confusing to her and I made changed to the prototype accordingly.   
**Possible changes to make:**

* She was confused how to create a reservation so I think we could change the prototype slightly to make it more clear how to add a reservation. All we have now is a + button to create a new reservation. We could add a description to this button and maybe we could make it bigger to draw the eye to it or change the color.
* She was also confused about about what the invite field was, I wasn’t sure if this was because she didn’t read the whole description or because it was just a confusing page. We will have to review and see if we can make this page easier to understand.

#### Building Manager Test

[Video Link](https://youtu.be/6VXkj43BcOE)

I had my mom, Alicia, pretend to be a building manager and go through the process of viewing feedback reported by users, and had her resolve that feedback. These are the directions she was given.

**Building Manager**

**Task description:** As the library building manager you are tasked with resolving issues reported about library rooms. Please follow these direction to resolve a reported room issue:

1. Login in to scheduling website
2. View all the user reported feedback for your entire building
3. View just one of the reported issues
4. Resolve the feedback
5. Report feedback as resolved
6. Logout

She provided some useful feedback and I used that feedback to make some change to the prototype.

**Possible changes to make:**

* She was confused about what the resolve conflict button would do and if it she would be able to leave her comments. One solution could be to change the text on that button to tell the user that they would be able to leave comments on the next page so they weren’t left wondering what that button would do and if that was the end of the conflict reservation process.
* She did mention that after using the software once or twice she felt like it would be more obvious to use, but we don’t necessarily want the learning curve to be to steep, so we need to figure out a way to mitigate that issue.

#### System Administrator Test

[Video Link](https://youtu.be/HpAv2I2ygyQ)

I had my dad, Trent, pretend to be the system admin and had him go through the process of resolving a conflicting room request. These are the instructions he was given:

**System Administrator**

**Task description:** As the Snow College system administrator one of your tasks is to resolve room schedules. Please follow these directions to view conflicting room requests and resolve the conflict.

1. Login to scheduling website
2. View all conflicting room requests
3. Select a conflict to resolve
4. Please resolve this conflict according to your discretion as the Snow College Administrator.
5. Logout

After his pilot test I made some initial changes and then we met as a group later and the prototype has since changed a lot.

**Possible changes to make:**

* After reviewing the workflow we wanted to pilot test with the group, we changed around the whole process. I was missing a few key tasks in my wireframe that we had to add. I felt like even though we changed the flow of resolving conflicts, I was still able to use the some of the comments he made to create a better wireframe
* The feedback I received from this test was still very useful and we used it while redesigning the workflow of this task.
* One of the biggest things I noticed was that for these users to figure out the initial workflow was confusing, and maybe we could resolve this issue by adding more descriptive buttons.

### Leedan Pilot Testing

#### Student User test

[Video Link](https://youtu.be/yJNSm1v9xWo)

The user is a previous Snow college student. She performed the requirements on a laptop, and the website did not perform as well as expected. Regardless, she was able to make the reservation successfully, although there were some points at which she struggled.

Instructions:

· Login

· Add a reservation

· Enter some given filters

· Select the first result

· Make the reservation

From the test, it was clear that not all of our users will know to use the plus symbol to add a reservation.

It’s also noticeable that we will need to improve upon the filter forms in implementation.

According to the comments made during pilot testing we were able to update our wireframes accordingly to create an easier flow for the users and make the process more obvious and hopefully we are ready for the evaluation stage.