Slope Search Project Report

Members:

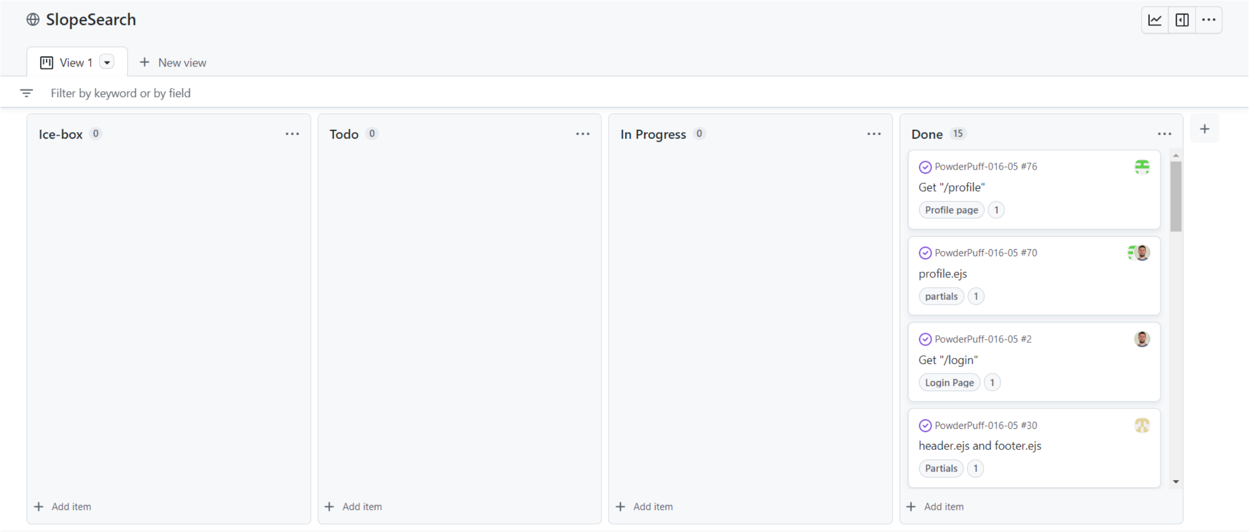
- Daniel Evarone
- Dylan Eilber
- Matt Procter
- Pankaj Behera
- Walter Virany

Description:

Our website, Slope Search, is a multipurpose informational website for ski resorts in Colorado. Slope Search lets users discover their new (or old) favorite ski resorts. If you are looking for a new location to check out in Colorado, you can explore all the resorts and determine which resort is right for you! When users first visit our website, they will land on the home page where users have the option to browse all the ski resorts in Colorado through the form of several cards which show an image of the resort, with the name of the resort and the location of the resort. They may then click on any specific resorts that might interest them to learn more information about that particular resort. Specific information about each resort includes an image, the location, elevation, annual snowfall, resort website, and the number of terrain parks. Then users can visit our about page which tells them about the vision of our website and the reason why we created it. After this the user may choose to create a new account through the register feature after which they can login to their account. Once logged in, the options in the menu bar change to Profile and Logout instead of Register and Login. They can then view their own profile and browse the website.

Project Tracker: https://github.com/users/wvirany/projects/3/views/1

The screenshot only shows a few of our tasks, but we had assigned a lot more tasks to each member of the group and the list of all the tasks may be viewed by using the link given above for the Project Tracker. That being said, we discussed and assigned many other smaller tasks to each other through our communication platform i.e. Discord.



SlopeSearch

View 1 | + New view

Filter by keyword or by field

Ice-box 0

Todo 0

In Progress 0

Done 15

PowderPuff-016-05 #76
Get "/profile"
Profile page 1

PowderPuff-016-05 #70
profile.ejs
partials 1

PowderPuff-016-05 #2
Get "/login"
Login Page 1

PowderPuff-016-05 #30
header.ejs and footer.ejs
Partials 1

Video: https://youtu.be/zVoLkqMd_M8

The above is a link to our video of the website which we uploaded to youtube, please use this link to view the video. Thanks!

Version Control: https://github.com/wvirany/PowderPuff-016-05

The above is a link to the version control i.e. our GitHub repository which contains all the required files except the video of the website as we were unable to push a video into our repository but you can access that video through the above youtube link that we have provided.

Contributions:

Daniel: I began with the implementation of the login page, for both the index.js and the login.ejs partial. After working on the bones of the pages, we had issues with Docker and I worked on a resolution. Once those pages were created I then deployed the session variable for the dynamic menu bar to toggle between login/register and profile/logout. I then created the profile and populated the data from the database to display for the user currently logged in. During the final week of the project I enhanced the visuals of all pages and password/email validation.

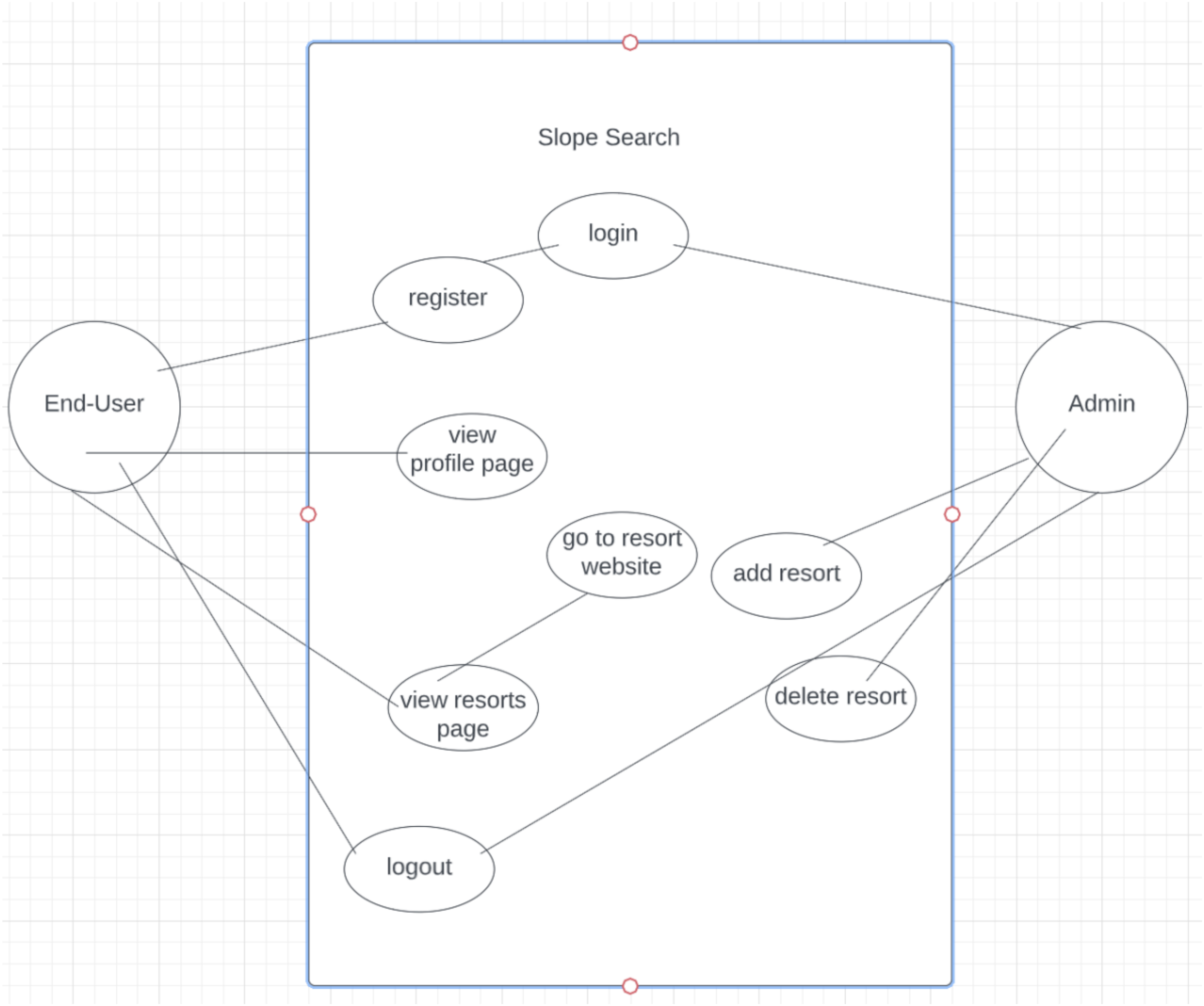
Dylan: For the project I was assigned to work on the database and work a bit on the review feature. I created the file with all of the tables in the database and I also contributed to populating the database as well. For the review feature I created the modal.html, and modal.js files for a working review modal. This modal was removed from the final as we were not able to get it connected and working with the database before the due date.

Matt: For the project I worked on the database tables with Dylan to create the data storage for the project. I looked for an API for the ski resorts and ski runs before we decided to manually input all the resort data. I input all data for the ski resorts in Colorado and selection of runs from the main resorts, before the runs feature was scrapped. I also cleaned up the formatting on the profile page to make it more legible for the users.

Pankaj: My contributions to the project were mainly the register feature and making sure the information gets stored in our local database. I worked on the API's for register and profile pages and also helped out on other pages and partials. I did the initial front-end messages and overall look of the website after which Daniel improved and stylized the front-end. I then worked on the proper deployment of the website and fixing some other errors that we got during execution. I also worked on the index.js page and all the other common files to get better functionality of the website.

Walter: My contributions were primarily to the home and resorts page, as well as helping to debug issues on many other portions of the project. I created many of the ejs pages and partials to get the project running. I wrote the APIs to populate the home and resort pages with data from the database. From there, I conceptualized and created the user interfaces for the home and resorts page, which were then further developed by other group members.

Use Case Diagram:



Test Results:

Daniel ran 2 test cases with 2 different users, both not currently enrolled in this class. The users were given the site with no instructions to see their natural course of actions. Both users were able to easily navigate through the website, exploring each page available and eventually arriving at either the login or registration page. Once they reached the registration page they were asked to register with an invalid email address so that the email validation could be tested. In addition, the password validation was successful, prompting the users to create a password that matched our requirements. Once the registration was successful they were redirected to the login page where they were able to login with the credentials they registered with.

Our test cases included:

- Password has invalid credentials
- Email has invalid credentials
- All registration fields have valid input
- Login functionality with valid credentials
- “Check it out!” button redirects to a page with selected resort details

The users naturally fulfilled some test cases simply by using the website, and the remaining test cases were verbally requested and successfully checked. All use cases tested worked as intended.

Deployment:

We were able to deploy our project on the cloud using the information given on the section of deployment. We have stored all the pictures and the features we had at that point on our website in the MilestoneSubmissions/ProjectDeployment_Cloud.pdf. In this pdf, we can clearly see that the website was launched on the cloud as the initial url is: ‘csci3308.int.colorado.edu:3000\home’.

Even though we were able to run the project on the cloud, most of the class was having issues with deploying their project on the cloud and hence we decided to launch our project on localhost itself, which worked perfectly fine.

The way we deployed our project is using docker through localhost, we just had to change our directory to our Project and then change our directory to the src folder after which we ran the common docker-compose up which runs the docker-compose.yaml and our website gets deployed on localhost:3000. To restart the project we can use docker-compose down –v, which will shut down the volumes and hence even our database will get restarted after which we can again deploy our project using docker-compose up.