SNOW MAPS

UMass Lowell

GUI Programming II Spring 2016

February 3, 2016

Jason Downing, Brendan Bousquet, Huy Huynh, JT Shepple

Table of Contents

Project Goal

Snow Maps will dynamically display ski conditions and trail statuses at multiple ski resorts on easy to view maps, viewable on all internet connected devices for easy access wherever the user happens to be.

Description of Features

The core idea beyond Snow Maps is that users will have access to multiple mountains on the same webpage, in an easy to view map format. Rarely do Ski Resorts display their trail conditions in such a way. Typically Ski Resorts dump their data to a table or a div on a webpage and force skiers and snowboarders to hunt through and figure out which trails they like and which trails they dislike.

The main homepage of Snow Maps will display a list of available mountains to view. Currently this is implemented as a simple table, which shows a 3x2 row of mountain logos / mountain names. In the future we plan to add a menu bar, with a drop down menu to easily navigate to other mountains and view different options.

The homepage will look different depending which platform the user is viewing from. On laptops or desktops the user will see more content at once, such as more available mountains on the main page. We will condense this information on tablets, and further condense on smart phones by focusing more on the menu bar to display information. This will make it easier to navigate on different devices, especially smart phones which will be our target as most users will want to check trail status and conditions on the go. We will however have some features focused on desktop users, such as the ability to view more than one mountain at a time, side by side or even in a 2 by 2 grid.

Each mountain will have its own HTML document with its own map in the form of an image with highlighting to indicate which trails and lifts are open. This will allow users to easily view each mountain's conditions in several easy to view images, accessible from any internet connected device.

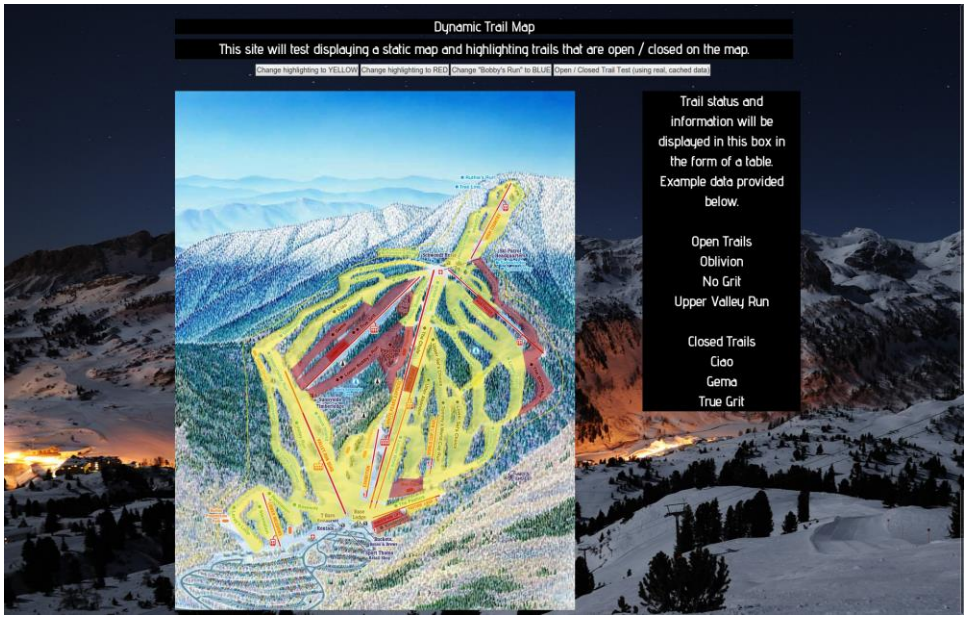
There will be additional features added as time permits. One concept is to display the trail condition data in the form of a table next to the trail map, so that users can both quickly see an overview and look for individual trails based on several parameters such as difficultly.

Another concept would be to add several buttons to toggle which trails are open. Currently this feature is present in the alpha version - you can toggle the map to display all the trails as open, and all the trails as closed. It should therefore be possible to add buttons to toggle all trails displayed, only trails which are open and only trails which are closed. This feature could allow users to quickly see only open trails, or only closed trails, which could allow them to quickly determine if their favorite trails are open or closed.

Example Screenshots

Desktop view:

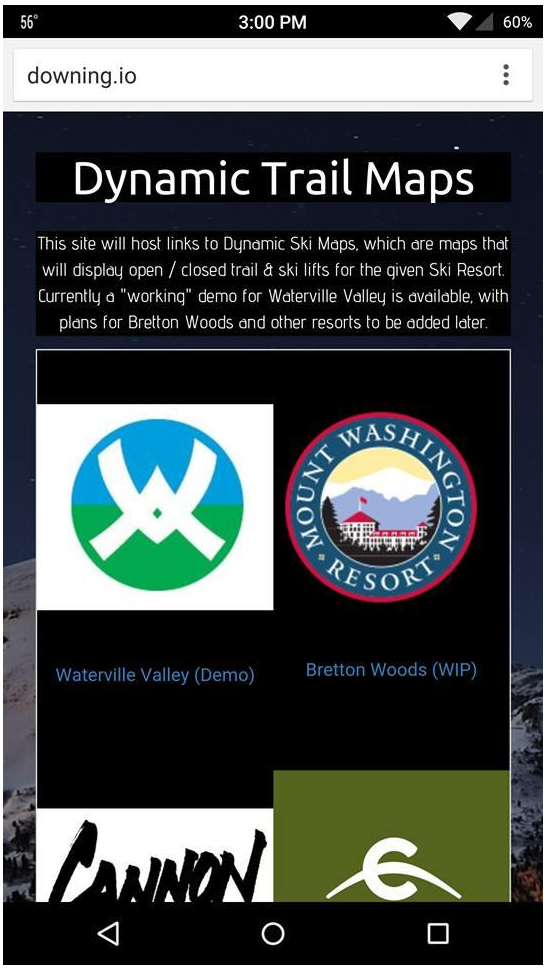
Starting screen with clickable logos of the ski resorts supported by the site.



The map will be highlighted to display trail information based on user input.

Mobile view (tablets / smartphones):

The site will condense on tablets to a row of 2 by 3, making it easier to view.



View in Android 5.1 Lollipop

Components

The UI portion of the site will be written in HTML, CSS and JavaScript. **Bootstrap** will be used to make the site responsive and work on smartphones, tablets and laptop/desktop computer screens. **jQuery** will also be used, as it is an easy to use JavaScript library for programming client side interactions. Several jQuery plugins will be used as well, such as the **jQuery Maphilight** plugin which allows dynamically highlighting an image based on an HTML image map. These image maps can be easily created using **GIMP**, a free image editor similar to Adobe Photoshop. The backend will be written using **Python**, and **BeautifulSoup** is a Python library which will be used for collecting the trail condition data by scraping several ski resort webpages. **Django** and **SQLite** will be used for the backend data processing, as Python will generate a JSON file for each mountain that can be easily stored in a database created using Django. Django has been chosen as most of the web scraping will be written in Python, and Django is an easy to use web framework for creating web pages quickly. It is also widely used and well documented online.

User Description

Snow Maps is built to support most internet connected devices. The requirements to use the application are ability to navigate a simple web interface and some knowledge of how to interpret a ski resort trail map.

Snow Maps will be convenient to users because it will aggregate information that would otherwise require the use of several different websites. The combination of snow conditions, trail maps, and trail/lift status all in one web application make Snow Maps a resource for novice and season skiers alike.

The common user is likely to be a regular skier who has knowledge of specific trails on a given mountain and has a desire to plan out their day on the mountain based on the trail status and snow condition information from Snow Maps.

Another common user is someone who is unfamiliar with a certain mountain and would like to visualize the path they want to take down the mountain. They can also see if trails are closed and find ski lifts more easily.

Issues

The main issue we ran into was how to get the trail and lift conditions data off of various ski resort websites. Instead of manually viewing each individual website and checking what trails are open and the prices, we would make it easier on the user to decide which mountains they want to look at and have all the information they need on one page. The most efficient way we can solve this problem is by using Python and scraping the data from their websites into useable JSON data and then displaying it nicely with bootstrap.

Project Schedule

Date	Task	Assignee
1/27/16	Create GitHub repository for the project and add the three other group members to it.	Jason
2/3/16	Create OneDrive group for sharing and creating the Project Proposal	Jason
2/3/16	Work on the Project Proposal and edit	Jason, Brendan, Huy, JT
2/10/16	Alpha Version - have a working client side version of two or three ski resorts to display the main idea of the project.	Jason, Brendan, Huy, JT
2/17/16	Begin working on the backend, and testing on a server.	Jason, Brendan, Huy, JT
3/8/16	Finalize alpha version	Jason, Brendan, Huy, JT
4/5/16	Finalize beta version	Jason, Brendan, Huy, JT
4/18/16	Prepare for class presentations	Jason, Brendan, Huy, JT
4/27/16	Prepare final submission	Jason, Brendan, Huy, JT

Criteria for Acceptability

This application should allow users the ability to:

- Smoothly navigate across home page and the various mountain map pages.
- Highlight open and closed trails.
- Highlight open and closed ski lifts.
- Highlight specific trails for each mountain.
- Show prices pulled from each mountain website.
- Select mountains to view on the same page.
- Automatically update the trail and snow conditions.

References

About SQLite. Web. 4 Feb. 2016. <http://www.sqlite.org/>.

"Django." *The Web Framework for Perfectionists with Deadlines*. Django, n.d. Web. 04 Feb. 2016. <https://www.djangoproject.com/>.

Kimball, Spencer, and Peter Mattis. "GIMP." *GIMP Full Atom*. The Gimp Team, n.d. Web. 04 Feb. 2016. <https://www.gimp.org/>.

Lynch, David. "JQuery Maphilight Documentation." *JQuery Maphilight Documentation*. N.p., n.d. Web. 04 Feb. 2016. <http://davidlynch.org/projects/maphilight/docs/>.

Richardson, Leonard. "Beautiful Soup." *Beautiful Soup*. Crummy, 29 Sept. 2015. Web. 04 Feb. 2016. <http://www.crummy.com/software/BeautifulSoup/>.