

Vanier College
Software Development AEC
Web Services
420-941-VA sect. 01222

Final Report

Professor: Mohammed Ali Hasheminezhad

2023-11-07

Team Members

Belal Mohsen

Shunzi Yao

Ana Pechonkin

Table of Contents

Table of Contents.....	2
SkiSmart: The Ultimate Ski Resort Finder.....	3
Project Aim.....	3
Project Description.....	3
Functional Requirements.....	3
Currently Implementation (Current Features):.....	3
Upcoming Features (Planned):.....	4
Non-Functional Requirements.....	4
User Stories.....	4
Individual's Role and Responsibilities.....	6

SkiSmart: The Ultimate Ski Resort Finder

Project Aim

The main goal of the SkiSmart application is to revolutionize the skiing experience. We want to provide ski enthusiasts with a dynamic and user-focused platform. SkiSmart aims to streamline the process of locating ideal ski resorts based on weather conditions, as well as user date and distance preferences. By harnessing real-time data and user feedback, we hope that the SkiSmart app will aspire to empower skiers with personalized, reliable information, ensuring an exceptional and well-informed skiing adventure.

Project Description

SkiSmart is an application designed for ski enthusiasts looking for the perfect skiing destination. It offers users tailored resort recommendations based on location, date, and distance preferences.

Functional Requirements

Currently Implementation (Current Features):

Home Page:

- **Postal Code and Date-based Search:** Users can input their postal code, select their desired skiing date, and adjust their search radius for personalized searches. Distances to the ski resorts are accurately calculated using the Haversine formula, ensuring precise and reliable information about the proximity of ski resorts.
- **Search Functionality:** Upon entering their preferences, users can initiate a search that redirects them to the results page.

Results Page:

- **Resort Listings:** Displays all ski resorts within the selected distance from the inputted postal code, with the distances calculated using the Haversine formula for accuracy.
- **Top Resorts Highlight:** The top 3 best-rated resorts are featured prominently, making it easy for users to identify the most recommended options.
- **Filtering Options:** Users can filter the results by rating or distance for enhanced convenience and personalized results.

Resort Details Page:

- **Comprehensive Resort Information:** Provides in-depth details about the selected ski resort.
- **Weather Information:** Includes current weather information to help users plan their visit.

- **Interactive Map:** Features an interactive map for easy navigation to the resort.
- **User Reviews:** Showcases user reviews, offering insights from other skiing enthusiasts.

Upcoming Features (Planned):

User Authentication: Login and Sign-up functionalities for personalized experience.

User Profiles: Future updates will allow skiers to create profiles to save favourite resorts, access past searches, and customize app settings.

Algorithmic Resort Rating: We plan to implement a sophisticated algorithm to rate resorts based on weather conditions, snow quality, and user reviews.

Contact Us Page: A dedicated page for users to reach out for support or inquiries.

About Us Page: A page about the team who created the SkiSmart App.

Non-Functional Requirements

- **Performance:** The application should load quickly and respond promptly to user inputs.
- **Usability:** The application should be user-friendly and easy to navigate for people of all ages.
- **Reliability:** The application should consistently provide accurate information and remain operational.
- **Security:** User data should be protected with appropriate security measures.
- **Scalability:** The application should be able to accommodate an increasing number of users.
- **Maintainability:** The application should be easy to update and maintain over time.

User Stories

Postal Code, Date, and Distance-Based Search

As a user,

I want to enter my postal code, select a date, and set a maximum distance.

So that I can find ski resorts near me that are open on my desired date and within my preferred distance range.

Filtering Based on Rating and Distance

As a user,

I want to filter ski resorts by rating or distance.

So that I can easily identify the top-rated or closest resorts to visit.

Detailed Resort Information

As a user,

I want to view detailed information about each resort, including weather conditions, user reviews, and an interactive map.

So that I can make an informed decision about where to ski.

Responsive Design

As a user,

I want the website to be responsive and mobile-friendly.

So that I can plan my ski trips on-the-go with ease.

Sign-Up

As a new user,

I want to easily sign up for an account on SkiSmart.

So that I can save my preferences, favorite resorts, and access my search history for a personalized experience.

Login

As a returning user,

I want to be able to log in to my SkiSmart account.

So that I can quickly access my personalized settings, saved resorts, and plan my next ski trip.

Saving Favorite Resorts

As a returning user,

I want to save my favorite resorts in my profile.

So that I can quickly access them for future trips.

Profile Creation

As a frequent user,

I want to create a user profile.

So that I can personalize my experience and have a more tailored service.

Individual's Role and Responsibilities

In the spirit of transparency and ensuring equitable distribution of work, each team member has detailed their contributions to the project.

- **UI Interface (material design, activity flow)**

The team created the following pages originally:

- o Belal: Home Page, Detail Page
- o Chelsea: Sign-up Page, User Profile Page
- o Ana: Sign-in Page, Results Page

Then we all focused on making the Home Page, Results Page and Details Page look better.

We each did 1/3 of the work for the UI Interface.

- **API calls (JSON, Async, Threading)**

The team decided to use Google APIs(Places API, place details API, Google map API, and Google geocoding API) and Visual Crossing Weather API:

- o Belal and Ana focused on Google APIs
- o Chelsea focused on the Weather API

We helped each other with each of the APIs as well.

We each did 1/3 of the work for the API calls.

- **Hosting (on an AWS server:IP Address: <http://3.145.39.202/>)**

We all practiced the hosting on our own and then worked on it together during class time.

We each did 1/3 of the work for the Hosting on AWS.

- **CI/CD:** Belal
- **Unit Tests:** Chelsea
- **API Documentation:** Ana

This is a self-declaration, and we attest to the honesty and accuracy of the above distribution of tasks and contributions.