**‘Surf Safari’ Senior Project Plan**

Kevin Moreland

**Problem and Motivation**

There exists a subset of adventure surfers interested in studying and traveling to remote, unmapped surf locations around the world. This is evident in many surf films, from the classic ‘Endless Summer’ by Bruce Brown in 1965 (which followed a group of surfers chasing Summer around the globe and discovering new surf spots, most notably Cape Saint Francis) to the more recent film ‘Cold Water Journal’ (where a group of surfers visit surf spots that were scouted out using Google Earth and weather data) by San Luis Obispo local Chris Burkard. More evidence of this subset of surfers is evident from the Google Earth contest in 2007 by Surfing Magazine which led to the discovery of the world-class surf spot ‘Skeleton Bay’ in South Africa. These surfers are in need of a program that combines satellite imagery, weather information, and other surf specific information into one intuitive interface.

Although websites such as Surfline and Magic Seaweed largely satisfy the casual surfer’s need for surf related knowledge when it comes to well established surf spots, they lack the flexibility to allow surfers to explore unmapped, remote locations around the world and asses their surf potential. Additionally, they tend to present information in a processed, consumer friendly manner that is not ideal for surfers who want to learn about surf forecasting by making their own predictions and making their own observations.

**Solution**

This project, ‘Surf Safari’, aims to satisfy the need in the surf community for a tool that can be used to discover unmapped surf locations and learn about surf forecasting. It will incorporate several features including topology data, wind data, satellite imagery, weather data, and buoy swell data into an explorable map interface on a website.

**Features**

I have divided my list of features into two groups, ‘basic features’ and ‘stretch goal features’. The ‘basic features’ are the bare minimum features I need to be able to consider this a complete project. The ‘stretch goal features’ could add a lot of great functionality to the project and will be implemented if I am able to complete my other work on time.

*Basic Features*

1. Live swell information (height, period, and direction) for any coastal point using the NOAA API
2. Interactive map with satellite imagery
3. Topology data overlayed on map
4. Display of wind, air temperature, and water temperature (if available) at locations picked on map
5. Tide information
6. Ability to calculate distance between points
7. Ability to save marked points on map
8. Ability for users to create accounts

*Stretch Goal Features*

1. Forecast of surf at specific coastal point
2. Displaying webcams located near the ocean using the Windy API for webcams
3. Ability to see satellite images of a specific location at a specified date using the NASA Earth API
4. Seafloor topology overlayed on map
5. Information about dangerous ocean life or hazards specific to locations (sharks, jellyfish, unclean water, etc)
6. Flight and/or drive time and cost estimates to travel to a discovered location from current location

**Assessment of Building Options for Project**

*Map API*

A major feature of this application will be an interactive satellite map. It is essential that I be able to overlay this map with weather and geographical data that I retrieve from other sources.

After reviewing several other options including Leaflet, TomTom, and Maxar I have decided that by far the most well-established map API that has all the functionality I need in terms of satellite imagery and customizability is MapBox GL JS.

*Front end*

My main choices for the frontend are React and Angular (because I am most familiar with these tools, and I will be able to develop the fastest with one of them). Both are widely used and well established, but I have decided to use React because of personal preferences (I prefer React’s one-way data binding and use of JSX and JavaScript instead of Angular’s two-way data binding and use of HTML templates and TypeScript).

*Back end*

I will use AWS Lambda for my backend so that I can have the simplicity of not worrying about managing a server and so that I can built my architecture as a microservice with several independent parts.

**Timeline for Quarter 1**

I decided to structure my timeline so that roughly every 10 days, I have a new deliverable due. In addition to the below deliverables, I will also be continuously reaching out to people experienced in surf forecasting and in discovering remote surf spots to see if new features are required or if certain features need adapting.

|  |  |
| --- | --- |
| **Goal** | **Deadline** |
| Complete plan for quarter 1 | 1/20 |
| Figma demo of project | 1/30 |
| A MapBox map is setup to toggle between satellite images and topology information | 2/10 |
| Finished all React components for frontend. All backend code is stubbed. | 2/20 |
| Implement ability to create accounts and save user information about marked locations on map | 3/1 |
| AWS Lambda setup in backend to retrieve swell data from NOAA, and return it to the website in a way it can interpret | 3/10 |

**Goals for Quarter 2**

Once all the frontend work is finished and some of the backend is complete in Quarter 1, I hope in Quarter 2 to:

1. Finish the backend code for retrieving weather conditions
2. Setup the backend code for registering/logging-in users
3. Setup a database in AWS to store user account information and data associated with accounts
4. Create a test suite for all my backend code to test edge cases and automatically detect errors before I release updates
5. Implement as many of the stretch-goal features as I can before the end of the quarter