CPSC 2350

Group 2 Project

Langara College

Github Repository: https://github.com/kng72/cpsc2350-group2

Colin li
Siam Shafiq
Ho Chun Alvin Li
Ki Hin Ng
February 18, 2022

Colin Li
Siam Shafiq
Ho Chun Alvin Li
Ki Hin Ng

Contents Pages

Colin Li
Siam Shafiq
Ho Chun Alvin Li
Ki Hin Ng

Overview of Project

The goal of our project is to create a website that can be used by travellers that will find the weather of a city and the covid situation of the country that the city is located in. Our hope for this project is that this will help people decide on where they want to travel by looking at the weather in the city they are travelling to and also seeing if the covid situation is relatively safe or not by giving the user data on the covid cases and letting the user assess the risks of travelling there.

Our project will be done using an agile SDLC model and a mixture of Scrum and Kanban frameworks which will allow us to be more flexible with decision making so that development phases don't need to be set in stone such as in the waterfall model. The work breakdown structure will look more like the kanban style where we will have columns of todo, assigned, tested/checked, and finished. The items in the columns will change each week during the Scrum Sprint planning and will be looked over during the review session so that we can be more dynamic with our project and it will allow for the person working on their assigned task to ask for help from others to check, test, or work on features with them.

To develop our project the technology stack will include HTML, CSS, and JavaScript. We are able to pull data from the weather and covid APIs by using JavaScript. The weather API will be used to display the temperature, duration of the day, and cloud coverage of the city and the covid API will be used to get the active cases, recovered, and deaths to covid in the country where the city is located.

For our current schedule, we have set aside Wednesdays for review of our progress in our tasks as well as discuss problems that have arisen or could arise. We have also set aside Saturdays, after class, for Scrum sprint planning for the rest of the week. We have set harder deadlines for ourselves during our planning phase in order to get our ideas down so that we have more time to research and test them, allowing more time for revisions if it’s found that our initial idea won’t work.

Colin Li
Siam Shafiq
Ho Chun Alvin Li
Ki Hin Ng

SDLC Model Chosen: Agile (Scrum & Kanban framework)

Before we decided on a Software Development Life Cycle, we had to first assess each member's technical and theoretical knowledge. After a small survey on how much experience we have, we decided it was best to go with an iterative approach such as **Agile**. Agile allows for the safest approach to building something tangible given the skills that we have, and by compartmentalising features in our website, we are able to keep adding features to it whilst still having a working product.

This gives us the most leeway in assessing our progress throughout the term and adjusting accordingly. Because there are no constraints of the project's scope, if we find ourselves to be struggling in adding a particular feature, we can choose to omit that entirely without having to jeopardise having a working solution at all. Scrum in this situation would be our best bet due to the flexibility in allowing for week to week changes, decisions, and reviews from the group.

Other methodologies such as Waterfall, would require a strong foresight of what features we want and a concrete timeline, but because we are inexperienced and our scope is not well defined from the start. We will use a kanban style WSB to make a weekly to-do, in progress, checked/tested, and completed kanban board. This will allow us to discuss who does what part during weekly scrum sprint planning and review, giving us a more dynamic way of assigning jobs that are better suited for each of our skill sets.

Colin Li
Siam Shafiq
Ho Chun Alvin Li
Ki Hin Ng

User stories

Weather API:

1. As a travel photographer, I want to know about the weather and the duration of the day, so that I can have better planning when taking outdoor photos. Depending on the weather, I may need to bring different photography gear, such as additional lights for a cloudy day and a waterproof bag to protect my camera on a rainy day.
2. As an exchange student, I want to know about the weather and temperature of the country I am going to so that I know what clothes I should take beforehand. It is important for me to know about the weather before I start my trip since I will be staying there for over a month or more.

COVID-19 API:

1. As an old couple, we do want to travel but we have concerns about current public health, because we are not young anymore, and once we get infected, we might have serious health problems. We want to check the status of COVID-19, including the number of active cases in the city we want to travel to so that we can be well-prepared for the pandemic. We can estimate how effectively and how seriously a country is dealing with the virus so we can have more peace of mind while travelling.
2. As a young travel lover, I would like to check the amount of active COVID-19 cases in the destination that I want to go to because if the pandemic in the place is severe, then the government might place a lot of restrictions like closing theatres or museums, I might not be able to enjoy all of the spots in that city.

Colin Li
Siam Shafiq
Ho Chun Alvin Li
Ki Hin Ng

Outline of our technology stack and why we chose it

Our tech stack considers all aspects of our development process. From the design phase, the development phase, and down to the languages and APIs that will be used to develop this application.

This project will be based on using HTML, JavaScript, and CSS for development. We will build a website that gives an easy glance into the situation of the country at the given moment in regards to weather and COVID.

This website will use JavaScript to pull data from a weather API and a COVID-19 API for the selected city, then it will give the weather of that city and the current covid cases in the country, and the information is displayed on one of the webpages.

We chose these languages because all of our team members are familiar with using these technology stacks, as learnt from previous Langara's courses such as CPSC1030, CPSC1045, CPSC2030, etc.

List of chosen APIs

1. Weather - https://openweathermap.org/

   The OpenWeather API, based on the information on the site proved to be the API of choice for us. It allows for JSON responses, paired with a simple API Key endpoint as opposed to having any complicated authentication procedures. The JSON data has exactly what we needed for this website as well, which are temperature, weather description, sunrise-sunset time amongst other things. The API also includes data for every city in the world, which was the strongest reason to pick this API as we are looking to cover a global scope.

2. COVID-19 - https://covid-19.dataflowkit.com/

   This COVID-19 API is interesting the way it works. It scrapes data from websites that have COVID data to get its data, therefore it has a fully fleshed out JSON response that includes the COVID data for all countries in the world. That is particularly important to us as most COVID APIs focus on one country only, which would mean 200 different APIs to get the data of every country. Therefore this API replaces 200 of them, which makes this an easy choice for us. The data includes infections, recoveries, deaths, etc.

Colin Li
Siam Shafiq
Ho Chun Alvin Li
Ki Hin Ng

Planned features per each API
(in detail - 3 features per each API, 6 features minimum)

Weather API features:

1. Temperature
   The website will display the current temperature at the searched location, similar to a weather website. This data will also include other data such as humidity, feels like temperature, the general forecast of the area, etc.

2. Sunrise/Sunset time (Duration of Day)
   By using the sunrise and sunset time, we can calculate how long the days are at the search location. This may be especially useful to tourists who want to escape the long nights higher up in the Northern Hemisphere. Some tourists may want this information to gauge the level of safety in an area as more daylight may mean more light for safety or other reasons such as filming opportunities, etc.

3. Cloud coverage
   The JSON response of this weather API talks about the cloud cover of the searched location. Using that, we can display graphics to represent that cloud cover in an animated format. This information may be especially useful for astronomy reasons, but also as a general understanding of how sunny it may be during the daytime.

Covid19 API features:

1. Active Cases
   This data is pulled directly from the JSON response and using this we can ascertain how strong the COVID situation is in the desired destination at the current time.

2. Recovered
   By using the recovered data, users can come to understand how good the medical services are in the area, as more recovery rates mean a higher standard of medical services, and COVID policies in place.

3. Deaths
   The number of fatalities from COVID gives the user an idea of the vaccination situation in the location. A high % of fatalities could mean that a strong vaccination program is not in place, which accounts for the higher death counts.

Work breakdown structure (WBS)

We will be using a mixture of Scrum and Kanban frameworks to assign work for each Sprint week with Sprint planning on Saturdays and reviews on Wednesdays.

Task order	TODO	Assigned/ Working on	Tested/ Checked	Completed	Expected hours	Actual hours
	Milestone #1					
1.0	Initial Planning					
1.1	Tech Planning					
1.1.1	APIs /API Features	Siam Shafiq	Colin Li	✅	1	2
1.1.2	Tech Stack	Ki Hin Ng	Colin Li	✅	1	1
1.1.3	SDLC	Siam Shafiq	Colin Li/ Siam Shafiq	✅	2	2
1.1.4	Wireframe	Siam Shafiq	Colin Li	✅	2	2
1.1.5	Data Flow Diagram	Alvin Li	Colin Li	✅	2	2
1.1.6	Complete Tech Planning			✅	8	9
1.2	Communications/ Time Management			✅		
1.2.1	Project Schedule	Colin Li	Colin Li	✅	1	1
1.2.2	WSB	Colin Li, Ho Chun Alvin Li	Colin Li	✅	2	3
1.2.3	Complete Communications & Time Allocation Planning			✅	3	4
1.3	Goal Overview			✅		
1.3.1	User Stories	Ki Hin Ng, Alvin Li	Colin Li	✅	1	1

1.3.2	Overview	Colin Li	Colin Li	✅	1	1
1.3.3	Complete Overview			✅	2	2
1.4	Complete Initial Planning			✅	13	15
2.0	Presentation #1					
2.1	PowerPoint	Ki Hin Ng	Team	✅	1	1
2.2	Concept Explanations					
2.2.1	Overview of Project	Colin Li	Team	✅	0.5	0.5
2.2.2	SDLC Model	Ki Hin Ng	Team	✅	0.5	0.5
2.2.3	Application Features	Siam Shafiq	Team	✅	0.5	0.5
2.2.4	Complete Concept Explanations			✅	2.5	2.5
2.3	Tech Choice Explanations			✅		
2.3.1	Wireframes	Siam Shafiq	Team	✅	0.5	0.5
2.3.2	Data Flow Diagram	Alvin Li	Team	✅	0.5	0.5
2.3.4	Complete Tech Choice Explanation			✅	1	1
2.4	Complete Presentation #1			✅	3.5	3.5
	Milestone #2					
3.0	Development of HTML, CSS, JS Boilerplate					

3.1	Base Page Development					
3.1.1	HTML Boilerplate				2	
3.1.2	CSS				2	
3.1.3	JS				3	
3.1.4	Complete Base Page Development				7	
3.2	Complete Base Website Interface				7	
4.0	Functionality					
4.1	Weather Functionality					
4.1.1	Update Weather Temperature Data				3	
4.1.2	Update Daylight Duration Data				2	
4.1.3	Update Cloud Coverage Data & Animation				2	
4.1.4	Finalise weather API features				7	
4.2	Covid Tracker Functionality					
4.2.1	Show Active Cases				3	
4.2.2	Show Recovery Rate				2	
4.2.3	Show Death Rate				2	
4.2.4	Update the Risk Bar				2	
4.2.5	Finalise Covid API Features				9	

4.3	Finalise Functionality				16	
5.0	Testing					
5.1	HTML Testing					
5.1.1	Validate the Code				0.5	
5.1.2	Search Bar				0.5	
5.1.3	Complete HTML Testing				1	
5.2	CSS Testing					
5.2.1	Validate the Code				1	
5.2.2	Responsiveness				2	
5.2.3	Color Scheme				0.5	
5.2.4	Layout				1	
5.2.5	Complete CSS Testing				4.5	
5.3	JS Testing					
5.3.1	Validate the Code				5	
5.3.2	Test APIs with Different Queries				5	
5.3.3	Caching API data				5	
5.3.4	Test errors				6	
5.3.5	Complete JS Testing				21	
5.4	Complete Testing				26.5	
6.0	Deployment Preparations					

6.1	Secure API Keys in .env files				3	
6.2	Update Github Repository				1	
6.2.1	Merge branches				2	
6.3	Run the Website				2	
6.4	Complete Deployment Preparations				8	

Colin Li
Siam Shafiq
Ho Chun Alvin Li
Ki Hin Ng

Project schedule/timeline

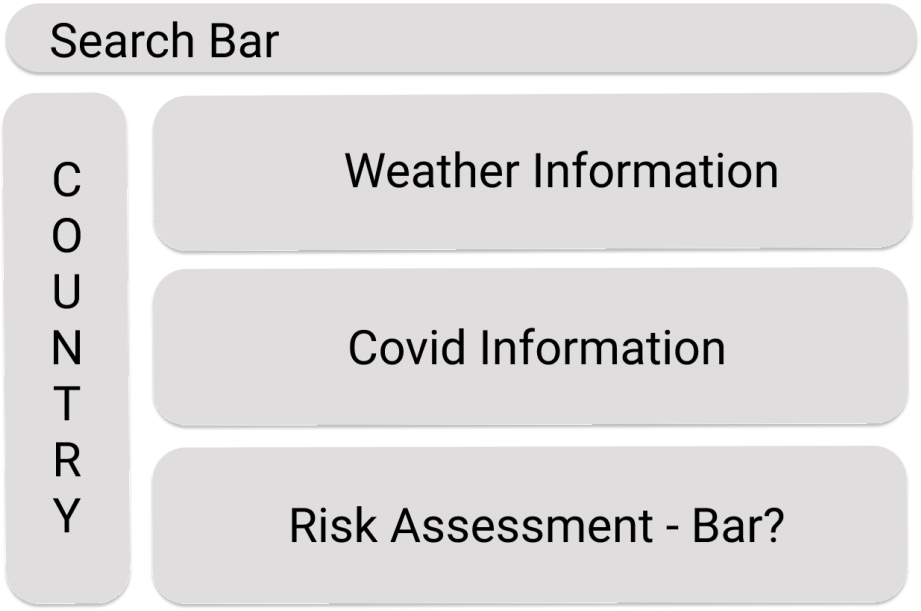
We will meet 2 times a week via Discord voice chat - every Wednesday at 10 pm we will have a review session on what we have completed and what we need help on. Saturday after class we will have a scrum sprint planning session for the upcoming week. We may have extra meetings 1 day before the deadline to review assignments and other work to make sure everyone is on the same page with tasks and goals.

Milestone #1 - Planning
Jan 29 - First meeting to have basic ideas
Feb 2 - Start working on the project report #1
Feb 4 - Working on check-in #1
Feb 5 - Check-in with instructor
Feb 12 - Complete report #1 (use remaining time to revise report if needed)
Start working on the presentation
Feb 18 - Finalize report and presentation ppt
Feb 19 - Report due date and presentation #1
First scrum sprint meeting to discuss how we will finish the GUI
Start working on the application
Feb 23 - First Scrum review session on our progress
Feb 26 - Finished GUI will be tested and reviewed by the group to make sure it's to everyone's liking and sprint planning for the API development stage will happen
Mar 2 - Review session of API implementation (hopefully 1 API is done at this point so we can test it)
Mar 5 - Completion of at least 1 API and its functions so that we can begin testing.
Scrum planning will begin for the second API.
Mar 9 - Review of second API development as well as continue to test 1st API if needed.
Mar 12 - Completion of the second API should happen around this time sprint planning for the planned merging of all code and functions as well as completion of the code.
Mar 16 - The last review before code is completed and begins final testing

Milestone #2 - Execution
Mar 19 - Check-in with instructor
Mar 23 - Finalise the code and ready for deployment
Mar 26 - Complete report #2
Start working on the presentation
Apr 1 - Final check for report and presentation #2
Apr 2 - Final report due date and presentation #2

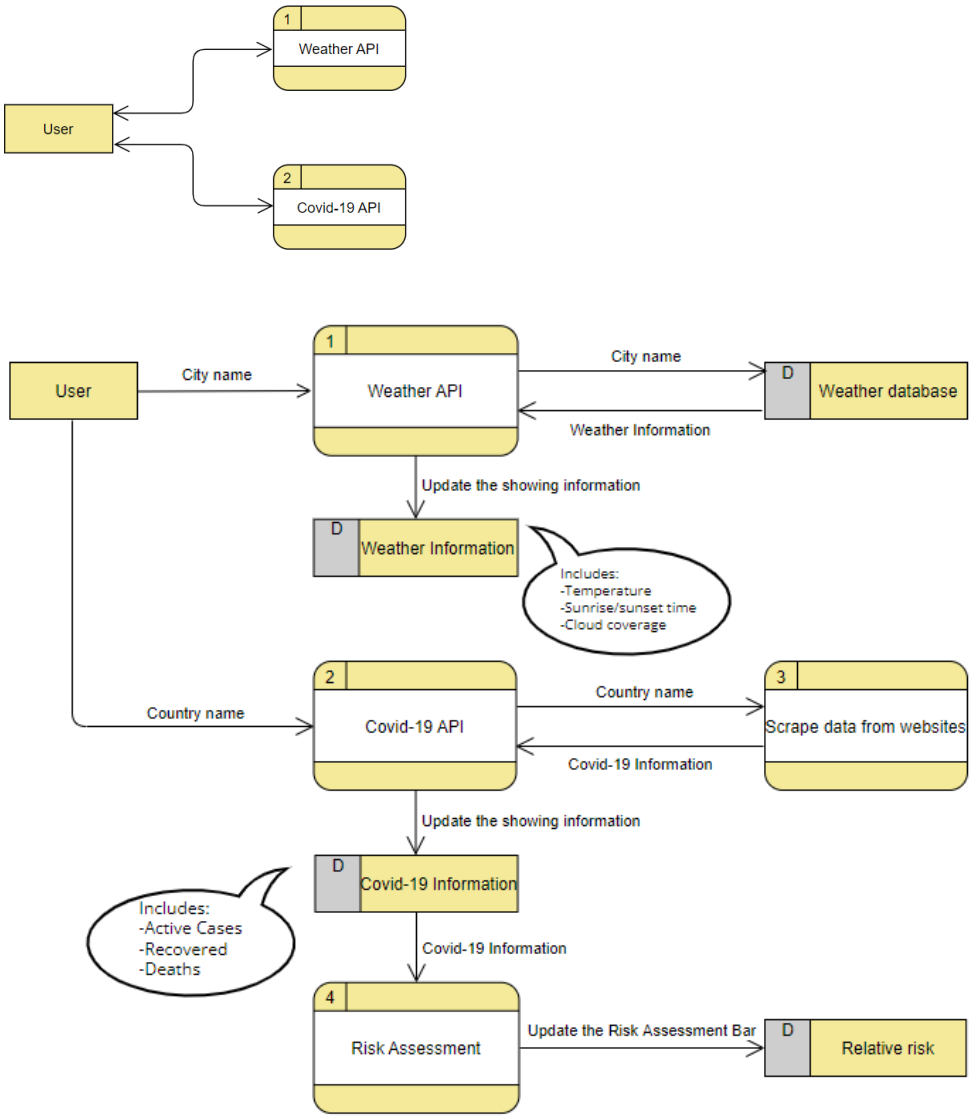
Colin Li
Siam Shafiq
Ho Chun Alvin Li
Ki Hin Ng

Wireframes and prototype elements for the application interface



This is a wireframe for the application interface of our website. The search bar is for the user to type the city they want to travel to. When the user presses enter, the below area will update and show all information about that city and corresponding country. This will include the weather of the city and the COVID-19 statistics across the country to where the city belongs. For the risk assessment, it is an indicator related to the COVID-19 active cases of that country. For example, if there are many active cases in that country currently, then there will be a nearly full bar with red colour, indicating the country is under pandemic and not safe for travelling.

Data flow diagrams (1 overview and 1 with details)



As the user inputs the city name on the search bar our website will retrieve it and use the weather API to search for the city name in the weather database. In return, the database will send back the weather information, including temperature, duration of the day, and cloud coverage of the city. Our website will then display them on the area under the search bar.

Our website will also convert the input city name into its corresponding country name, and the COVID-19 API will scrape data from websites that include COVID-19 information about the searching country. Once the data is collected, the integrated information will be displayed on the respective area of the interface. The information includes active COVID-19 cases of that country, amount of recovered cases and deaths related to COVID-19.