**TEAM BONELESS ICE**: Ayham Alnasser, Clement Chan, Kiran Vuksanaj, Tiffany Cao (PM)

SoftDev1 pd1

P01 -- Ocean Getaways

## PROGRAM OVERVIEW:

This website is essentially a helping guide to travelers going to other cities or countries globally. There is a single search form, allowing the users to input a city and country, which will be stored in a session. If the given city and country doesn't exist, is ambiguous, or misspelled, the website will show an error message and refresh. Using the IP Stack API, we can obtain the user's location and country, for which we can then find currency symbol using the MapQuest API. The same is done for the user's desired city and country. This information is used to compute the currency exchange rate, using the Currency Exchange API. The website will also show information regarding the weather of the city given using the Dark Sky API. Finally, using the Wikipedia (for images) and MapQuest APIs, users will be provided with images and a map of the city they gave, respectively. As an extra feature, there will also be a small section extracted from the Wikipedia API about the city given (taken from the Wikipedia page of that city).

## PROGRAM COMPONENTS AND APIs:

### BACK END + APIs USED:

Dark Sky (Clement):

- https://api.darksky.net/forecast/OUR_KEY/[Latitude],[Longitude]
- This API gives us the weather forecast for a certain location using coordinates
- Reports current weather as well as forecasts daily for the next week, hourly for the next two days, and minutely for the next hour (only precipitation data).
- The daily and hourly forecasts would be useful information for people to plan their trips.
- We will store weather forecast images in a weather database, so that each forecast will have a complementary image to go with it. For example, a forecast of a clear day will have an image of a clear sky next to it.

Air Visual (Clement): (SCRAPPED)

- http://api.airvisual.com/v2/nearest_station?lat=[LAT]&lon=[LONG]&key=[YOUR_API_KEY]
- http://api.airvisual.com/v2/city?city=[CITY]&state=[STATE]&country=[COUNTRY]&key=[YOUR_API_KEY]
- This reports the weather quality/pollution for a given city, or the data from the closest weather station to given coordinates
- The information will be used in addition with the weather data retrieved from the Dark Sky API

REST Countries (Kiran):
- https://restcountries.eu/rest/v2/alpha/US
- Gives information about countries, including their currency symbol, when the country code is entered
- This is necessary to connect the country outputted from the Geocoding API to the currency that should be checked from the Currency Exchange API
- No quotas are placed on requests, so it's not necessary to cache the information from this API

MapQuest Open Geocoding (Kiran):
- http://open.mapquestapi.com/geocoding/v1/address?key=towBT1Gfo92PG6GjBcJs7NoIswGUtsaH&location=New York City
- Gives us information about a queried location from the MapQuest database
- Useful fields: latitude/longitude, country, mapUrl
- Image from mapUrl, with modified zoom values, can be used to display a MapQuest map of the city (Clement)

Unsplash (Ayham): (USES OAUTH2, REPLACING W/ WIKIPEDIA)
- https://api.unsplash.com/search/photos?page=1&query=[CITY NAME]
- We use unsplash for the images that are associated with the searched location, Big Ben for London, Empire State Building for NYC etc.
- These photos are then immediately yoinked into the SQL DB where it is preserved so that we can minimize calls to this API
- The JSON output is fairly dense, but we can go straight to the largest image and save it in accordance to the length and width values given in the JSON

IP Location (Ayham): (SCRAPPED)
- http://ip-api.com/json/149.89.151.100?fields=status,message,lat,lon,currency,counry,query

- Using this API url, with no key necessary, information about the approximate location of an IP address can be obtained, alongside a wide variety of information about the location (including the currency code)
- When a user accesses the site, their IP address can be recorded and this API may be used to determine their current location, and as an extension what currency should be displayed on the website (see currency exchange API)
- **Note:** In development builds of Flask apps (i.e. everything we have used so far), the IP address returned is 127.0.0.1; because of this, location **cannot** be determined by IP address until sites are hosted.
- The symbol of the local currency can be retrieved from the JSON object by using data['currency']

IP Stack (Ayham):

- Same functionality of the former IP Location API, but outputs the countrycode instead of currency.
- Uses the ISP's IPv4 rather than browsers, so hosting it locally should be a non-issue
- The country code is output to the REST Countries API, where the currency code outputs

Wikipedia (Tiffany + Ayham):

- https://en.wikipedia.org/w/api.php?action=query&list=search&srsearch=CITY&utf8=&format=json
- https://en.wikipedia.org/w/api.php?action=query&prop=extracts&explaintext&exintro&titles=WIKIPAGE&format=json
- This API can be employed to get information about a given topic via its Wikipedia page.
- It can be used in our project to display a 'blurb' of information about a city once searched, coming from the first paragraph summarizing the Wikipedia article
- It should be noted that this extraction will likely require parsing of HTML, which is the format used to store the content of the Wikipedia page itself (even when requested in a JSON request).
- https://en.wikipedia.org/w/api.php?action=query&titles=CITY&prop=images&format=json&imlimit=3
- The Wikipedia API can also be used to obtain images (that are related to the city) from the Wikipedia database. We will limit the number to three images and display them on the information page.

Currency Exchange (Tiffany):

- Middle step: REST Countries API
    - This API gives the currency symbol of a given country
- https://api.exchangerate-api.com/v4/latest/USD
- Using the above API url, we can obtain the rates of currency exchange between USD and 50 other currencies in a ratio.
- The IP Stack API has a field that gives the local country based on the IP address. With the local currency and the desired currency (using REST Countries), we can obtain the exchange rate.
- The base currency, destination currency, and the exchange rate can be stored in a database, along with a timestamp. If a user asks for a base and destination that already exists, and the timestamp shows that the latest entry for that pair is less than 24 hours before, the exchange rate will be returned. If the entry is over 24 hours old, update the database by calling the API and getting the updated rate. The new rate is then returned to the user.
    - This is done to minimize the number of API calls we have to make to the Currency Exchange API.

Extra Features (All):

- A feature that allows the user to see the latest news from the country or city they searched for. This would theoretically be made possible with the GNews API, though there might be some extra work involved since the API apparently requires JavaScript. If we have time, we would implement this, and it would either be in a new HTML page or displayed along with the information page.
- In addition to the currency exchange information, we can also implement an in-page calculator that lets the user input an amount of money in their base currency (using a dropdown menu to choose their base), and converting that into its monetary equivalent in the currency of the location provided. ✓
- Inspirational travel quotes with They Said So Quotes API (inspiration from Team "Will Code For Food")

**FRONT END FRAMEWORK: *BOOTSTRAP***

- In order to more easily create a pleasant design and user experience for users, the front end framework *bootstrap* will be used.
- In order to take advantage of layout tools, each section of the page will be wrapped in a `container` div; the header and footer will be fluid, encompassing the full page, and the content will be a standard container, centering content and allowing margins
- example components to be used:
    - Navbar: navigate between pages of results

    - Alert: display, in varying colors, flashed messages
    - Jumbotron: introduce users to the site, display content such as weather results
- Bootstrap will be complemented with our own stylesheet defining additional details, especially the color scheme

**FRONT TO BACK END:**

- The landing HTML page will showcase instructions on how to use the website, as well as a search box to look for a city and/or country.
    - This search input is stored in a session.
    - Warning messages will be flashed if the given city and/or country does not exist, and the page will refresh.
- After a successful search, the next page will show the weather forecast. This will include information from precipitation rate, weather predictions for the next week, and more for the chosen destination, with images to accompany the data.
    - The weather information will be obtained with the Dark Sky API.
    - Images that accompany the weather forecast will be retrieved from the weather images dictionary, which provides url links for the images.
- The currency exchange rate page will display a ratio of the base currency to the desired currency. There will also be a form input for users to give an amount of money (in the base currency) and submit it, and the page will then return that money's equal value in the desired currency.
    - The currency used in the base country and desired country can be obtained using the REST Countries API.
    - The currency exchange rate will be obtained using the Currency Exchange API.
    - The conversion of the money is a quick and easy calculator function.
- The information page will feature a slideshow of images from the desired destination, with text boxes giving a quick summary of relevant information of that place. There will also be a map of the desired city.
    - All of the data on this page will be in a cached database to avoid the need to call the APIs multiple times (and go over the quota). The timestamp will also be stored to make sure that the information is up to date. The information should refresh once in a while (we'll decide on the time).
    - Before calling all the APIs described below to retrieve the data, the app will search the database for the desired city to check if the information already exists. If it does, we will use that information (given that it's up to date). Otherwise, we will call the APIs, retrieve the information, and then store it in the database.

    - Images will be obtained using the Unsplash API, while information on the city will be taken from the Wikipedia API.
    - The map of the desired city can be created with the MapQuest Geocoding API, with the help of the IP Location API.
- Navigation bar will provide quick and easy access to a destination's weather information, currency exchange information, photos and brief summary/description/information about the destination, and an option to go back to the search or landing page. If the last option is chosen, the session will end, and a new session will be started.

**DATABASES + DICTIONARIES**:

Weather Images Dictionary:
(**Weather, url**)

| **Weather** | **url** |
|---|---|
| clear-day | https://image.flaticon.com/icons/svg/136/136723.svg |
| clear-night | https://image.flaticon.com/icons/svg/414/414840.svg |
| cloudy | https://image.flaticon.com/icons/svg/414/414825.svg |
| partly-cloudy-day | https://image.flaticon.com/icons/svg/1146/1146808.svg |
| partly-cloudy-night | https://image.flaticon.com/icons/svg/414/414831.svg |
| fog | https://image.flaticon.com/icons/svg/2076/2076827.svg |
| rain | https://image.flaticon.com/icons/svg/1691/1691521.svg |
| sleet | https://image.flaticon.com/icons/svg/2204/2204342.svg |
| wind | https://image.flaticon.com/icons/svg/2201/2201153.svg |

| | |
|---|---|
| thunderstorm | https://image.flaticon.com/icons/svg/2201/2201130.svg |
| snow | https://image.flaticon.com/icons/svg/1116/1116724.svg |
| hail | https://image.flaticon.com/icons/svg/1779/1779856.svg |
| tornado | https://image.flaticon.com/icons/svg/2206/2206194.svg |

Cached Place Info Database:
(**countrycode TEXT, city TEXT PRIMARY KEY, currency TEXT, latitude REAL, longitude REAL, info TEXT, images TEXT, last_cached TIMESTAMP**)

| **Country** | **City** | **Currency** | **Lat** | **Lon** | **info** | **Images** | **last_cached** |
|---|---|---|---|---|---|---|---|
| U.S. | New York | USD | 40.730646 | -73.986614 | The **City of New York**, usually referred to as either **New York City** (**NYC**) or simply **New York** (**NY**), is the most populous city in the United States... | https://assets.simpleviewinc.com/simpleview/image/upload/c_fill,h_474,q_75,w_640/v1/clients/newyorkstate/5232359e_e163_475c_abe3_0f20af112a8c_ae020bfc-a771-4564-87b7-479fbe55735d.jpg | 2019-11-17 07:25 |
| Britain | London | GBP | 51 | 0.13 | **London** is the capital and largest city of England and the United Kingdom. Standing on the River Thames in the | https://cdn.londonandpartners.com/visit/general-london/areas/river/76709-640x360-houses-of-parliament-and-london-eye-on-thames-from-above-640.jpg | 2019-11-17 23:05 |

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| | | | | | south-east of England, at the head of its 50-mile (80 km) estuary leading to the North Sea, London has been a major settlement... | | |
| France | Paris | EUR | 49 | 2.35 | **Paris** is the capital and most populous city of France, with an area of 105 square kilometres (41 square miles) and an official estimated population of 2,140,526 residents as of 1 January 2019... | https://static.independent.co.uk/s3fs-public/thumbnails/image/2019/08/07/08/paris.jpg?w968 | 2019-11-17 23:06 |

Cached Currency Database:
(**base TEXT, destination TEXT, rate REAL, timestamp BLOB**)

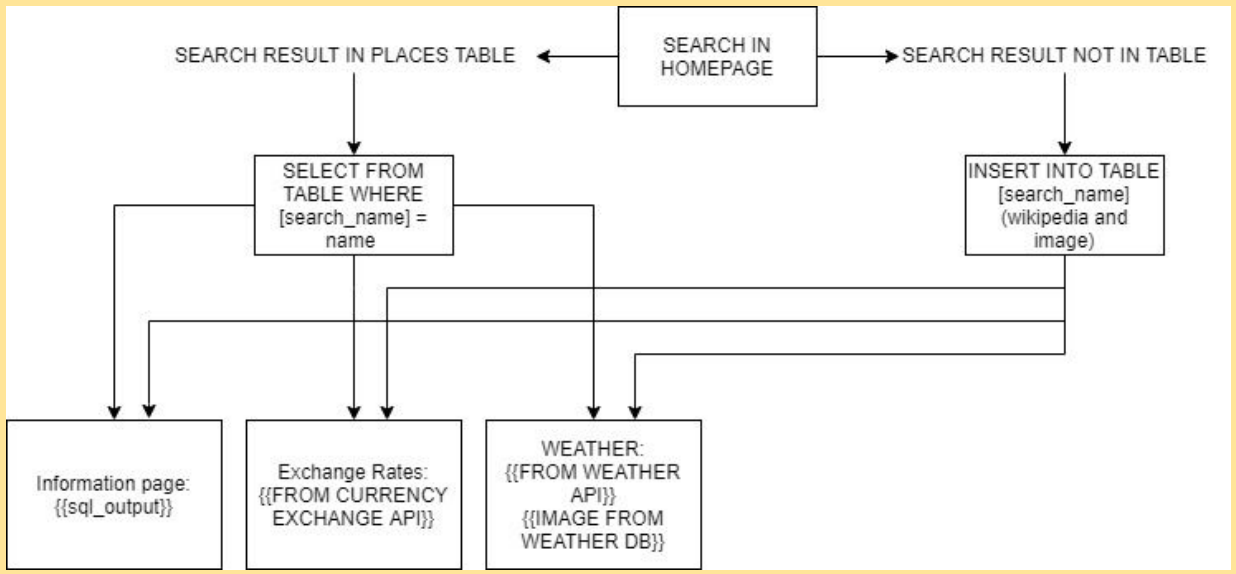
| **Base Currency** | **Desired Currency** | **Rate** | **Timestamp** |
|---|---|---|---|
| USD | CNY | 7.037717 | 2019-12-03 |
| CAD | USD | 0.752267 | 2019-12-03 |
| DOP | CNY | 0.136626 | 2019-12-03 |

Cached Map Database:

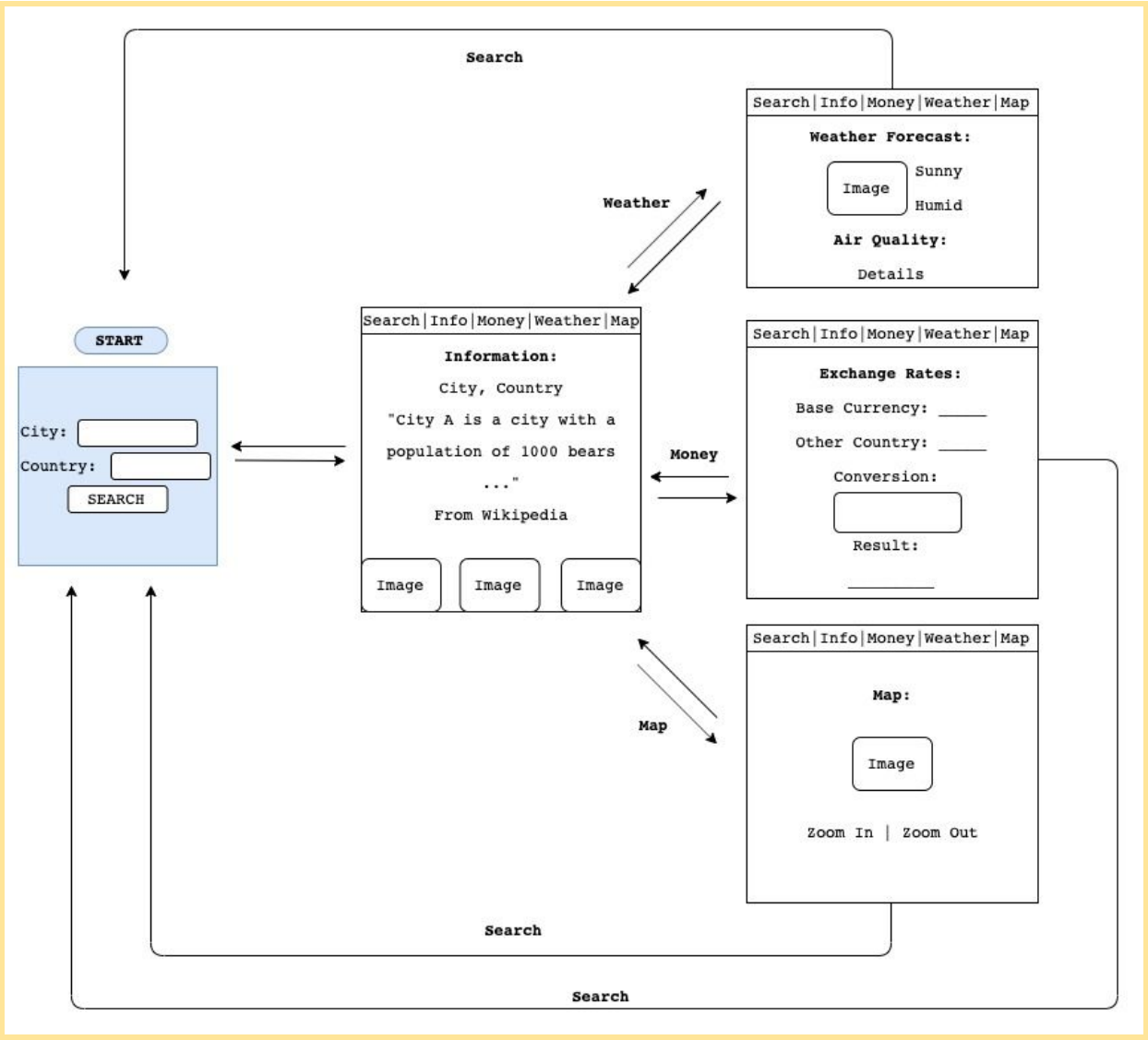
(**latitude REAL, longitude REAL, zoom INTEGER, last_cached TEXT, path TEXT**)

| Lat | Lon | Zoom | Last_Cached | Path |
|---|---|---|---|---|
| 40.730646 | -73.986614 | 11 | 2019-12-03 | static/maps/lat40.730646lon-73.986614zoom11.jpg |
| 27.180171 | 113.483764 | 9 | 2019-12-03 | static/maps/lat27.180171lon113.483764zoom9.jpg |
| 40.650104 | -73.949582 | 12 | 2019-12-03 | static/maps/lat40.650104lon-73.949582zoom12.jpg |

## COMPONENT MAP:



## SITE MAP:


Search
Search | Info | Money | Weather | Map
Weather Forecast:
Image
Sunny
Humid
Air Quality:
Details
Weather
START
City:
Country:
SEARCH
Search | Info | Money | Weather | Map
Information:
City, Country
"City A is a city with a population of 1000 bears ..."
From Wikipedia
Image
Image
Image
Money
Search | Info | Money | Weather | Map
Exchange Rates:
Base Currency: _____
Other Country: _____
Conversion:
Result:
_____
Map
Search | Info | Money | Weather | Map
Map:
Image
Zoom In | Zoom Out
Search
Search