Team Atmosphere

Alice Ni, Moody Rahman, Joseph Yusufov, David Wang

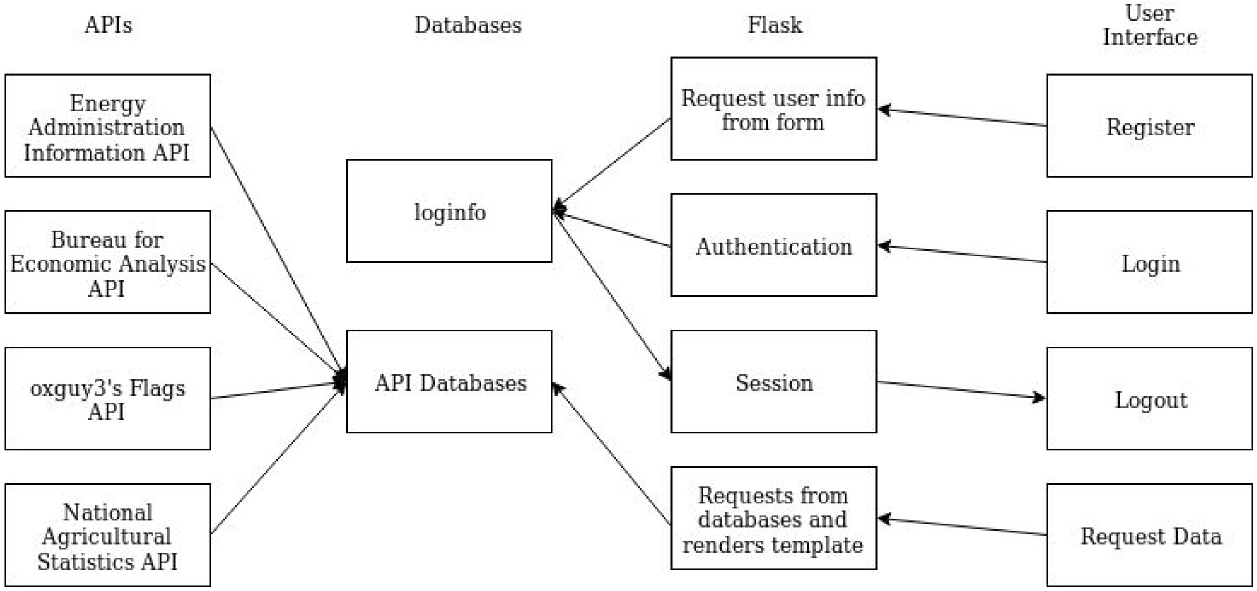
ROLES:

Alice - PM, Bootstrap
Moody - Database, python
Joseph - API, Bootstrap
David - Python, HTML

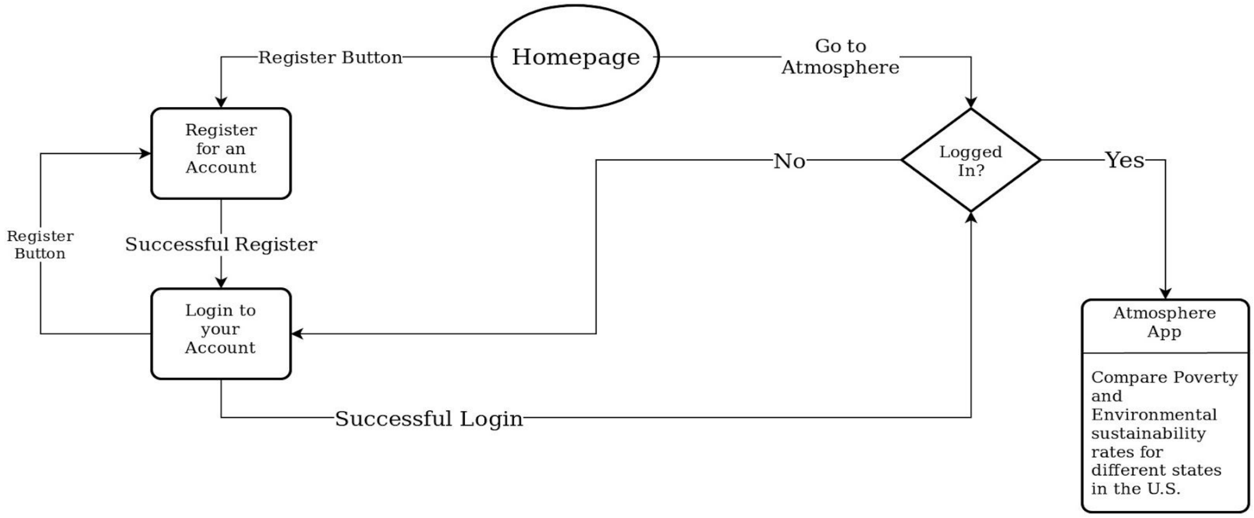
Functionality

- Interactive website that allows users to view and compare data from different states
- A few built-in data variables, stored as JSON files in a JSON file folder
    - Population per state
    - Income
    - Carbon emissions
- Users can choose the variables they want to compare, making a request to a Flask app that reads from the databases. If the data is not in the database, the app makes a request to the API that has the specified data.
- The data will be stored in JSON files relative to each API used. API information will be cached into the files as they are requested by users.
- Scattergrams and charts will be generated by the parameters specified by the user. Displays information in a table numerically as well.
- Implements bootstrap
    - Navbar
    - Charts
    - Tables
    - Flashed messages

Component Map


APIs
Databases
Flask
User Interface
Energy Administration Information API
Bureau for Economic Analysis API
oxguy3's Flags API
National Agricultural Statistics API
loginfo
API Databases
Request user info from form
Authentication
Session
Requests from databases and renders template
Register
Login
Logout
Request Data

Site Map



Atmosphere Site Map

Alice Ni, Moududur Rahman,
David Wang, Joseph Yusufov

Database Layout

loginfo

username TEXT	password TEXT
“jimbob”	“cooljoe23”
“hamlet”	“macbeth”

username: displayed name for each account, entered by the user
password: entered by the user

<username>

favorites TEXT
“CA”
“NY”

Table name: There exists one favorites table for each user that wishes to track favorite states, and this table is named the user’s username
Favorites: Alpha codes for states, a new row for each favorite state in a user’s table..

Front End

- base.html
  - Base template for all the pages
- index.html
  - Extends base.html
  - User **must** login or register for an account before viewing the site
  - Instructions on how to navigate the site
  - Buttons to register/login
- login.html
  - Extends base.html
  - Form for submitting an existing username or password
- register.html
  - Extends base.html
  - Form for creating an account
- welcome.html
  - Extends base.html
  - Home page that displays real-time data for the U.S. as a country (total U.S. population, carbon emissions, etc.)

    - “Stats by State” button- Option for user to select a single state and view all the available statistics for that state via a radio form
    - “Analysis by Indicators” button - Option for user to select and compare two variables via a form
    - Submit button brings the user to another page that displays all the requested data
- lookup.html
    - Page that displays the specified state(s) and data in a table
    - Generates a scattergram showing relation between selected data
    - If user wants to request specific data, they can enter it into a form
        - If data does exist in APIs, add data to the database
- analysis.html
    - Extends base.html
    - Allows users to choose independent and dependent variables
    - Generates a scattergram and graph that displays the requested variables and the 50 states are the points on the scattergram
    - Generates a table that displays the information numerically

Back End

- app.py
    - Login system
    - Registration system
    - Routes
        - “/”
            - Renders “index.html” if user not logged in
            - Renders “welcome.html” if user logged in
        - “/login”
            - Renders “login.html”
            - Redirect to “/home”
        - “/register”
            - Renders “register.html”
            - Redirect to “/”
        - “/welcome”
            - If user is not logged in, redirect to “/”
            - Renders “welcome.html”
            - Displays user specific info
        - “/lookup”
            - Renders “lookup.html”
            - Displays user’s recently viewed comparisons
        - “/analysis”
            - Renders “analysis.html” using data from cache file
            - Processes User-inputted arguments for the graph
        - “/logout”

            - Removes user from Sessions
            - Redirects to “/”
        - “/auth”
            - Will never be displayed to user
            - Checks if user is in session
- cache.py
    - Enters all relevant data points into a JSON cache every time that the server is started, and alerts the user when caching is complete.

Functions

- login()
    - @param: username
    - @param: password
- register()
    - @param: username
    - @param: password
    - Cannot register a username already in use
- welcome()
    - @param: username
- login()
    - Renders index.html
- auth()
    - @param: username
    - Checks if user is in session
- lookup()
    - @param: username
    - Makes requests to the API depending on the request the user makes
- analysis()
    - @param: username
    - Generates a graph and a table based on what the user requested through the data form. Takes in the independent and dependent variables
- logout()
    - Logs out the user

APIs

- U.S. National Agricultural Statistics API
- U.S. Bureau for Economic Analysis API
- U.S. Energy Information Administration API
- oxguy3's "flags" API
- CanvasJS Graphing API

Important Links:

National Agricultural Statistics API - https://quickstats.nass.usda.gov/api

Key: 79900EE9-743F-3CBA-AD8A-26063F956065

https://quickstats.nass.usda.gov/api/get_param_values/?key=79900EE9-743F-3CBA-AD8A-26063F956065&param=sector_desc

Example of GET request of all the corn produced by Virginia since 2012

http://quickstats.nass.usda.gov/api/get_counts/?key=79900EE9-743F-3CBA-AD8A-26063F956065&commodity_desc=CORN&year__GE=2012&state_alpha=VA

Returns: {"count":13048}

Global Climate API --

https://datahelpdesk.worldbank.org/knowledgebase/articles/902061-climate-data-api

Working request:

http://climatedataapi.worldbank.org/climateweb/rest/v1/country/mavg/tas/1980/1999/FRA

General format:

http://climatedataapi.worldbank.org/climateweb/rest/v1/country/**type/var/start/end/ISO3[.ext]**

Census Bureau:

https://api.census.gov/data/2018/acs/acs1?get=NAME,group(B01001)&for=us:1&key=07626e3b3578edd0e55ba15cb38770a85aedd31d

https://www.census.gov/data/developers/data-sets/acs-1year.html

Bureau for Economic Analysis:

https://apps.bea.gov/API/signup/index.cfm

Graphs: https://canvasjs.com/docs/charts/basics-of-creating-html5-chart/

Energy Information Administration:

Single-stat graph for a region over time:
https://www.eia.gov/opendata/embed.php?type=chart&series_id=EMISS.CO2-TOTV-TT-TO-AL.A