

# Deployment - Sprint 2

---

- **Update to UI for Weather Observations from Sprint 1**

- PastWeatherInquiry.py
  - User needs to hit F5 to begin running the program.
  - User needs to enter the option of Hourly or Daily depending on the timeframe of weather observations the user is requesting
    - If hourly
      - User needs to enter the date and time or range of dates and times for observational forecast data to be inquired. User must abide by the date format provided (Jan 01 2021 2 PM - Jan 01 2022 1PM and Jan 01 2021 for range of dates and single date respectively). It should be noted that the earliest date to be inquired is Jun 12 2017.
    - If daily
      - User needs to enter the date or range of dates for observational forecast data to be inquired. User must abide by the date format provided (Jan 01 2021 - Jan 01 2022 and Jan 01 2021 for range of dates and single date respectively). It should be noted that the earliest date to be inquired is Jun 12 2017.
  - If a user inputs a date that does not abide by the format, the user will be notified and advised to hit enter to try again.
  - User will need to hit “enter” on keyboard to reinquire another date or date range or type “exit” to end program.

- **Program Files for success of website**

# Deployment - Sprint 2

---

- WeatherEyesMain.py would ideally run 24/7 to collect data on an hourly basis.
- WeatherEyesMain.py
  - For collecting and storing hourly and daily forecast data from three weather providers on an hourly basis while the program is running.
  - For collecting and storing hourly weather observations from National Weather Service and daily weather observations from National Weather Service
  - User needs to hit F5 to begin running the program.
  - If National Weather Service scraping raises an exception, user will be prompted to run exclusively the National Weather Service class.
- Files needed in shared folder for successful running:
  - WeatherEyesMain.py
  - PastWeaterInquiry.py
  - NationalWeatherServ.py
  - AccuWeather\_WebScraper.py
  - WeatherComPdoc.py
  - Convert.py
  - HistoricScraper.py
  - HistoricHourlyScraper.py
- Installations in cmd line for successful running:
  - Pymongo
    - pip install pymongo
  - URL Lib 3
    - pip install urllib3
  - Requests
    - pip install requests
  - Pytest
    - pip install pytest
  - Coverage
    - pip install coverage
  - Beautiful Soup
    - <https://www.crummy.com/software/BeautifulSoup/bs4/doc/#installing-beautiful-soup>

- **WeatherEyes Website**

# Deployment - Sprint 2

---

- Website server is via Repl
  - Invitation would be via August Delbert
- Before running, one must install dns python in the shell
  - pip install dnspython
- Run the program and once console has started the server, use link to display page:
  - <https://WeatherEyes-1.delberta.repl.co>
- User must register if new to website and then log in to be directed to homepage
  - If first time logging in, user must update preferences
- User can use the side navigation bar to access the following pages:
  - Home
    - Preferred output will be displayed. If preferences have not been set, user will be prompted to set preferences.
  - Preferences
    - Space to set and update preferences
  - Current Forecast
    - User will be prompted for forecast output with options of hourly or daily
  - Prior Forecast
    - User will be prompted with options for a specific request
  - Observational Inquiry
    - User will be prompted with options for a specific request
  - Register
    - To create a new account
  - Login
    - To log in if not actively logged in
  - Logout
    - To log out