Assignment 4 - Part 2

Start Assignment

Due Friday by 11:59am **Points** 100 **Submitting** a website url

Assignment:

In this part of the assignment, we will add three main aspects.

1. We will secure the API with JWT tokens

- Update the Fast API to support JWT tokens
- Update the Streamlit frontend. Only authenticated users should be able to use the API

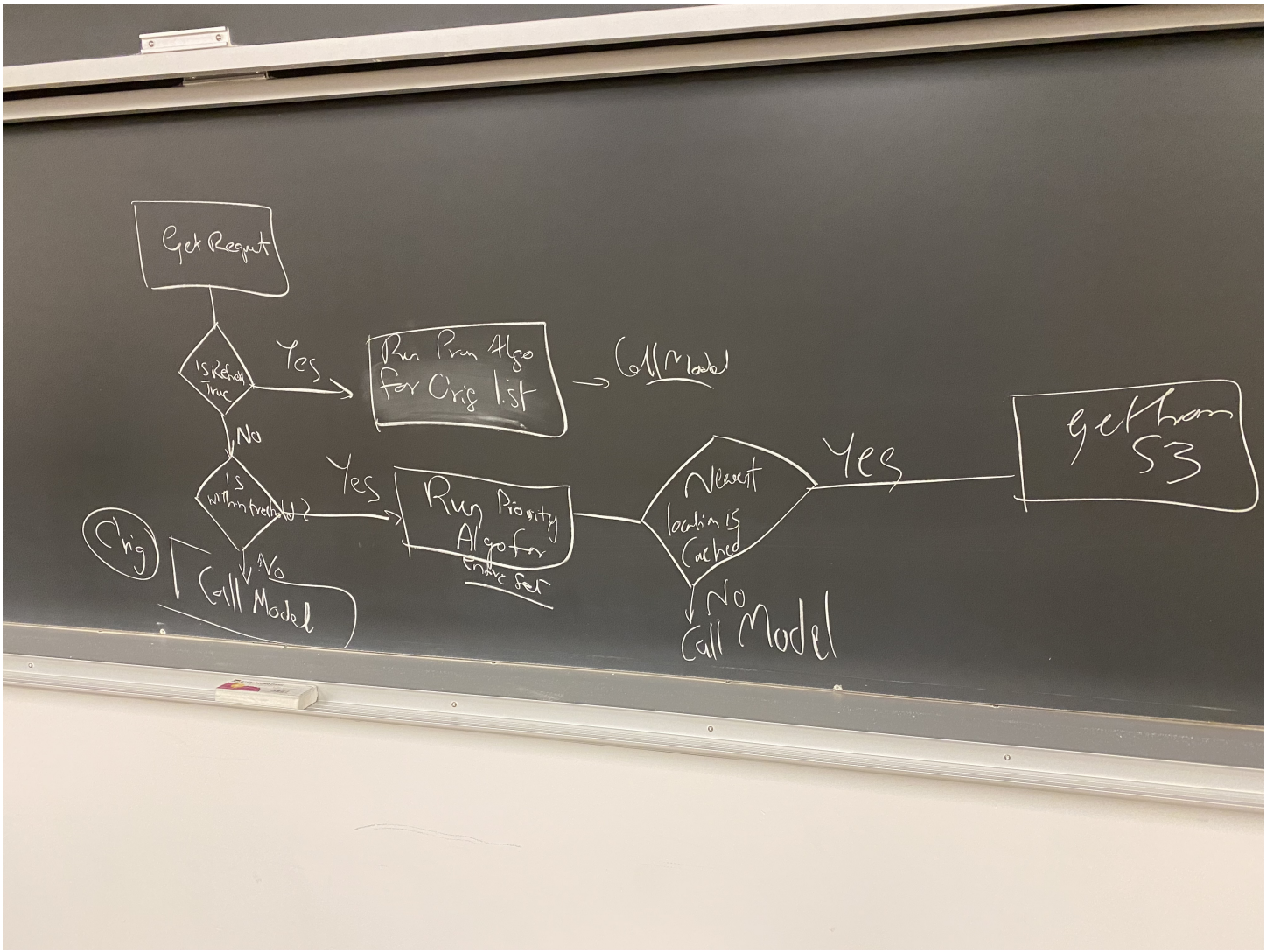
2. We will enable result caching using Airflow.

Airflow pipeline.

Create a list of 50 locations. For these locations, the Airflow pipeline will create the nowcast every hour and store it in a cloud location (You can use S3 or any other storage choice). Note that the airflow update <timestamp> needs to be stored so that the next part can access it.

3. Add additional parameters to the API. The Streamlit app will change to accommodate the new API parameters. The parameters are:
    1. Force refresh - To bypass the cache and get data from the model
    2. Threshold time - If <now> is within the the last update <timestamp> + <threshold time>, add the 50 locations to the 150 original location list when you search for the nearest location the user has entered. If the cached location is the closest, get data from the cache else, get data from the model.

The blackboard logic image is attached.



Other notes:

Use the architecture tools you have learnt in earlier assignments to draw the architecture for the entire system.

In addition to the development documentation, write a “User’s manual” and include 5-10 test cases to show how to use the application

IMPORTANT NOTES:

1. To address the concerns of GitHub snooping, the Readme.md should include the following attestation.

“WE ATTEST THAT WE HAVEN’T USED ANY OTHER STUDENTS’ WORK IN OUR ASSIGNMENT AND ABIDE BY THE POLICIES LISTED IN THE STUDENT HANDBOOK”

2. I have heard concerns from some student groups that the work was not evenly done by the team members. The readme.md should include a listing of individual contributions for each team member. In addition, you should list the %contribution to project adding up to 100%. For example : Satish 40%, Ramesh 50%, Sri 10%

Tutorials for preparation

1. Secure the API using jwt tokens

See following for tutorials and implement and add 5 more tests to test the secure api

- https://fastapi.tiangolo.com/tutorial/security/oauth2-jwt/

- https://testdriven.io/blog/fastapi-jwt-auth/
- **https://christophergs.com/tutorials/ultimate-fastapi-tutorial-pt-10-auth-jwt/ (https://christophergs.com/tutorials/ultimate-fastapi-tutorial-pt-10-auth-jwt/)**
- Video tutorial: **https://www.youtube.com/watch?v=0_seNFCtglk (https://www.youtube.com/watch?v=0_seNFCtglk)**

(https://www.youtube.com/watch?v=0_seNFCtglk)

2. Airflow pipelines

- https://airflow.apache.org/docs/apache-airflow/stable/tutorial.html
- ETL Pipeline: **https://airflow.apache.org/docs/apache-airflow/stable/tutorial_taskflow_api.html (https://airflow.apache.org/docs/apache-airflow/stable/tutorial_taskflow_api.html)**
- **https://www.youtube.com/watch?v=XD7euLOzKbs (https://www.youtube.com/watch?v=XD7euLOzKbs)**

(https://www.youtube.com/watch?v=XD7euLOzKbs)

Tutorial

https://www.youtube.com/watch?v=YWtfU0MQZ_4 (https://www.youtube.com/watch?v=YWtfU0MQZ_4)

(https://www.youtube.com/watch?v=YWtfU0MQZ_4)

https://github.com/karpenkovarya/airflow_for_beginners (https://github.com/karpenkovarya/airflow_for_beginners)