Title: Create project MLflow

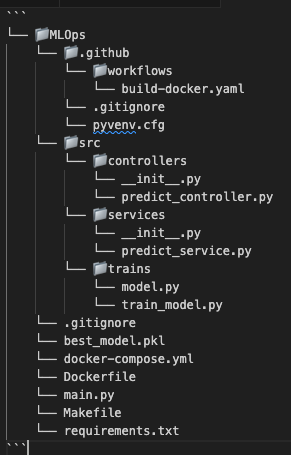
Email: [Khánh Nguyễn Đặng Quốc](mailto:khanh23mse43013@fsb.edu.vn)

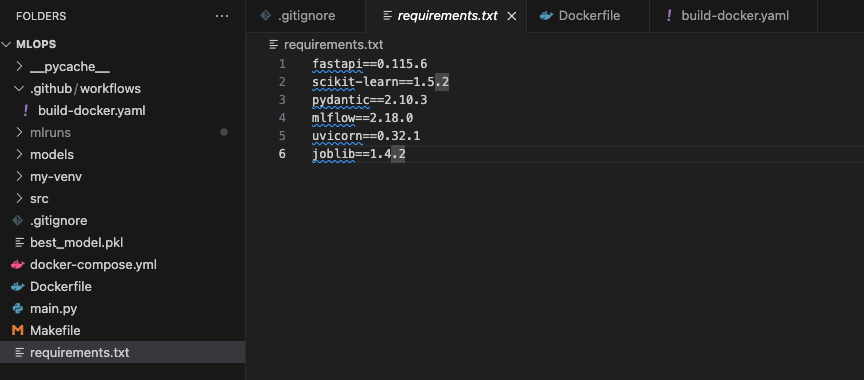
Phone: 0386686177 (Use it if have issue)

Link github: https://github.com/khanh23mse43013/mlflow-fsb

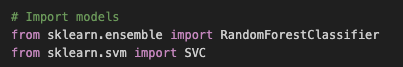
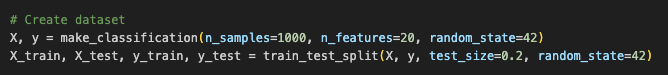
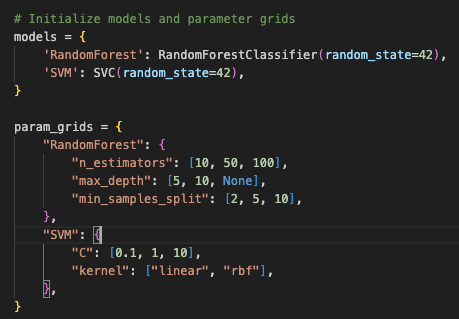
Description: Describe the steps I took in the project and the results after implementation

Step 1: Setup struct of folder

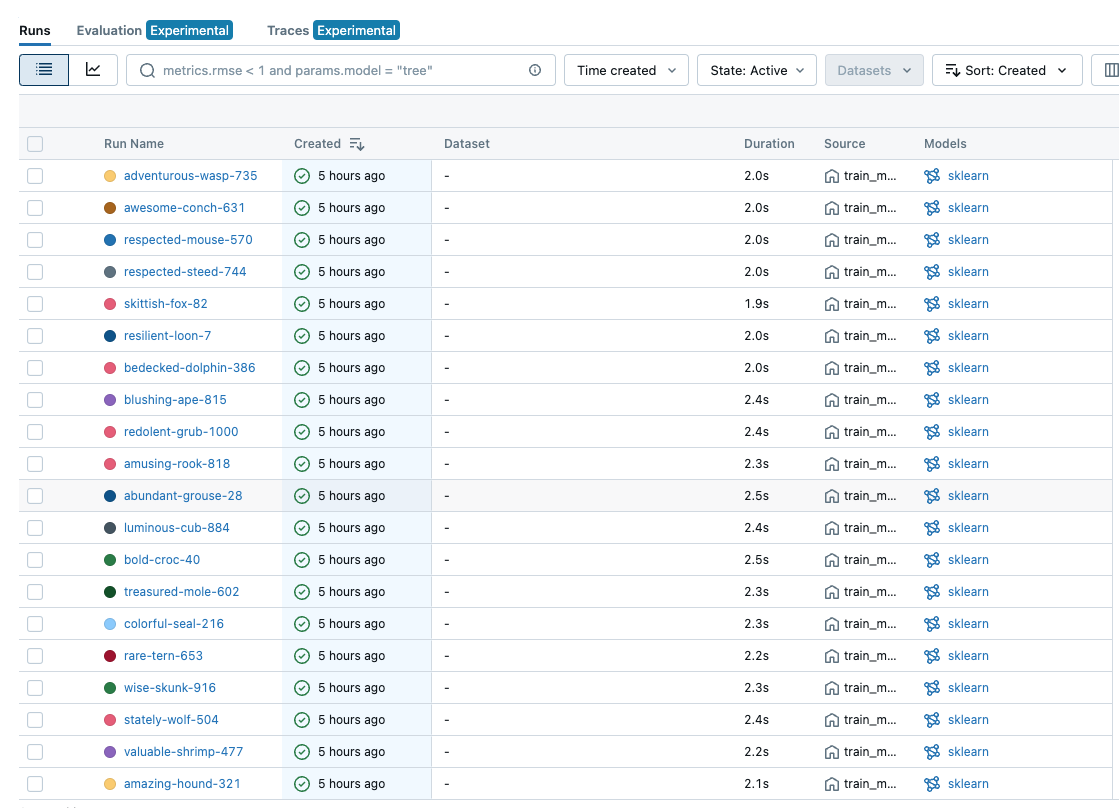
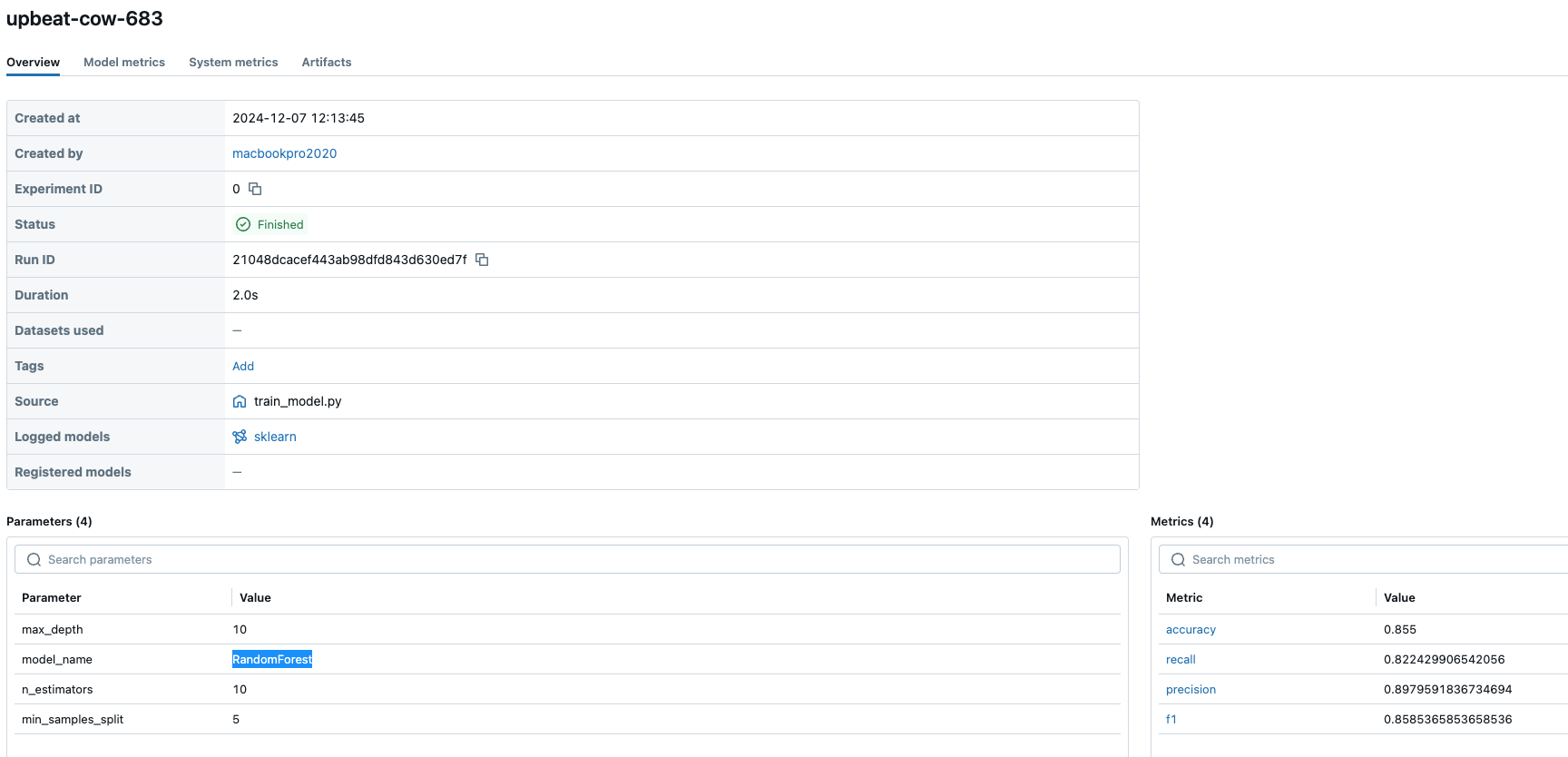


Step 2:Install necessary libraries

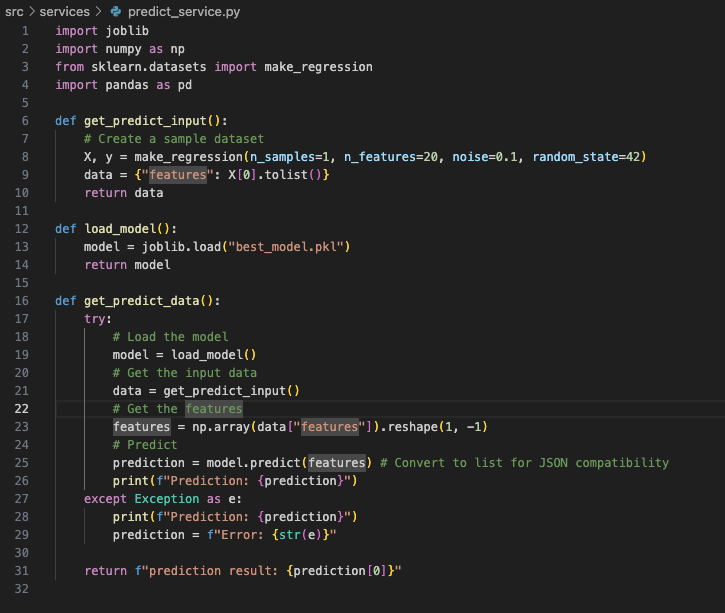
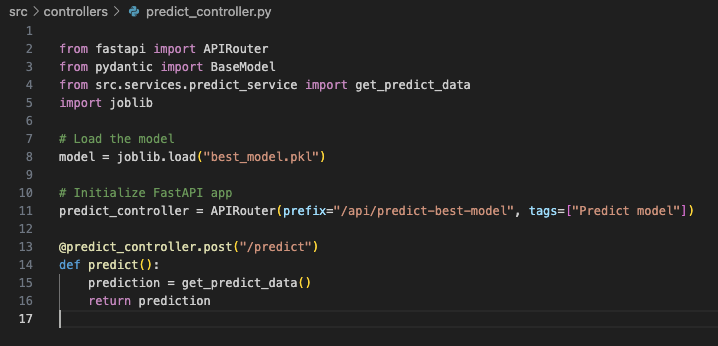
Step 3: Build model follow:

* Import necessary libraries
* Import model from sklearn
* Create dataset for training model 
* Initialize model and parameter grids(use for hyperparameter tuning)
* Setup MLflow tracking(locally)
* Training model includes: Configure model with parameters, Train the model, Evaluate the model, Log parameters and metrics to MLflow, Log the model to MLflow.
* Compare and get the best model for training(I compare accuracy between all model at Initialize model and get highest accuracy model )
* Save the best model

Step 4: Show MLFlow UI and tracking model:

* List model for training on MLflow  
  Detail of a model on MLflow

Step 5: Build FastAPI

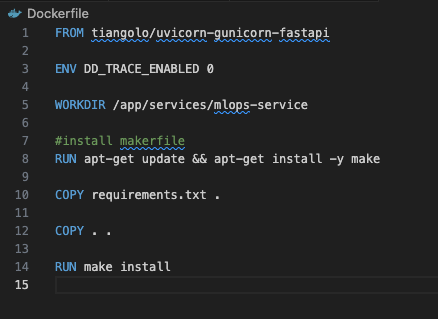
* I created a prediction service that reads the best model, makes predictions, and returns the predicted data
* I created an API endpoint in the predict controller file.

I was config swagger document.

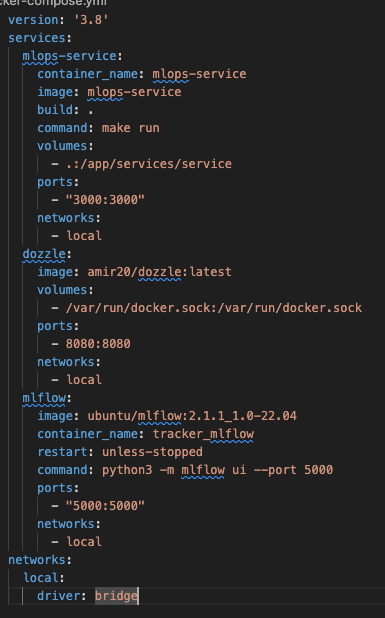


* It will description about my api predict at http://localhost:3000/docs

Step 6: Build dockerfile



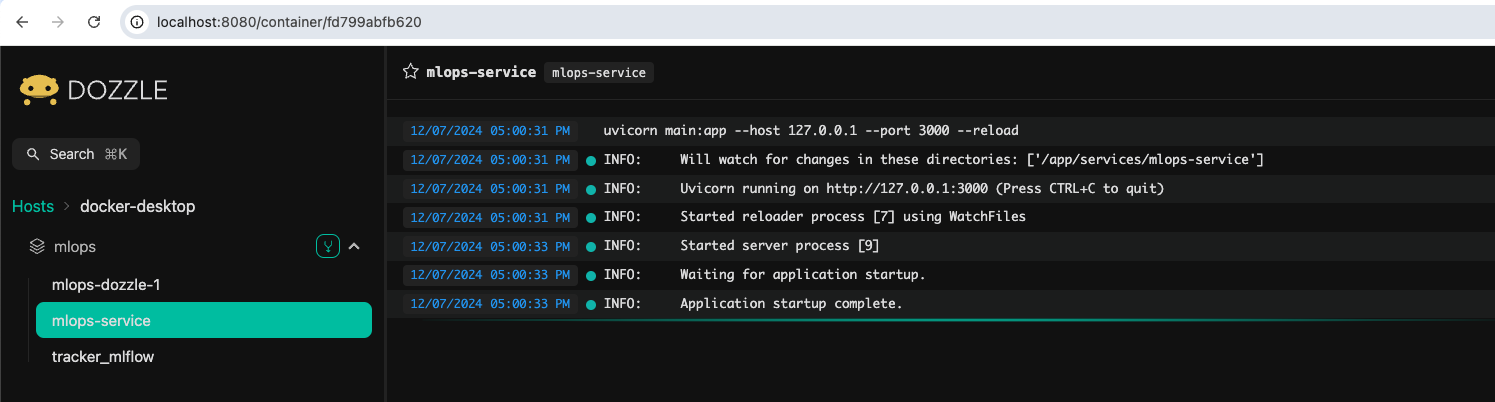
Step 7: Build docker-compose



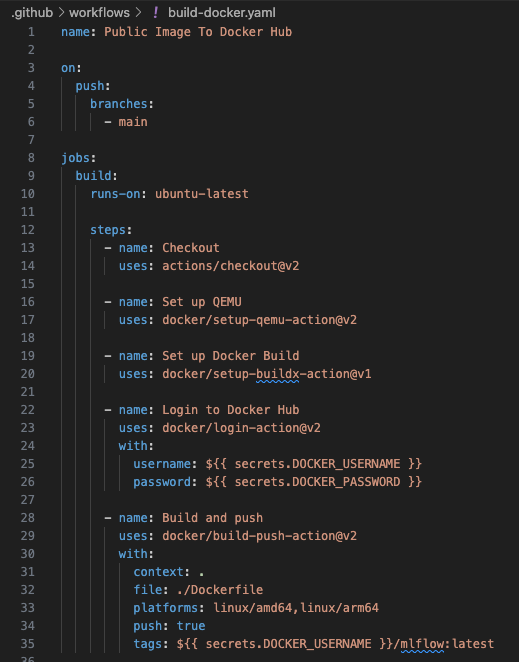
In my docker-compose file, I have created three services: mlops-service, mlflow, and dozzle (used for tracking logs via a UI).

If we want to deploy this on a server with CI/CD, I believe this file is necessary.

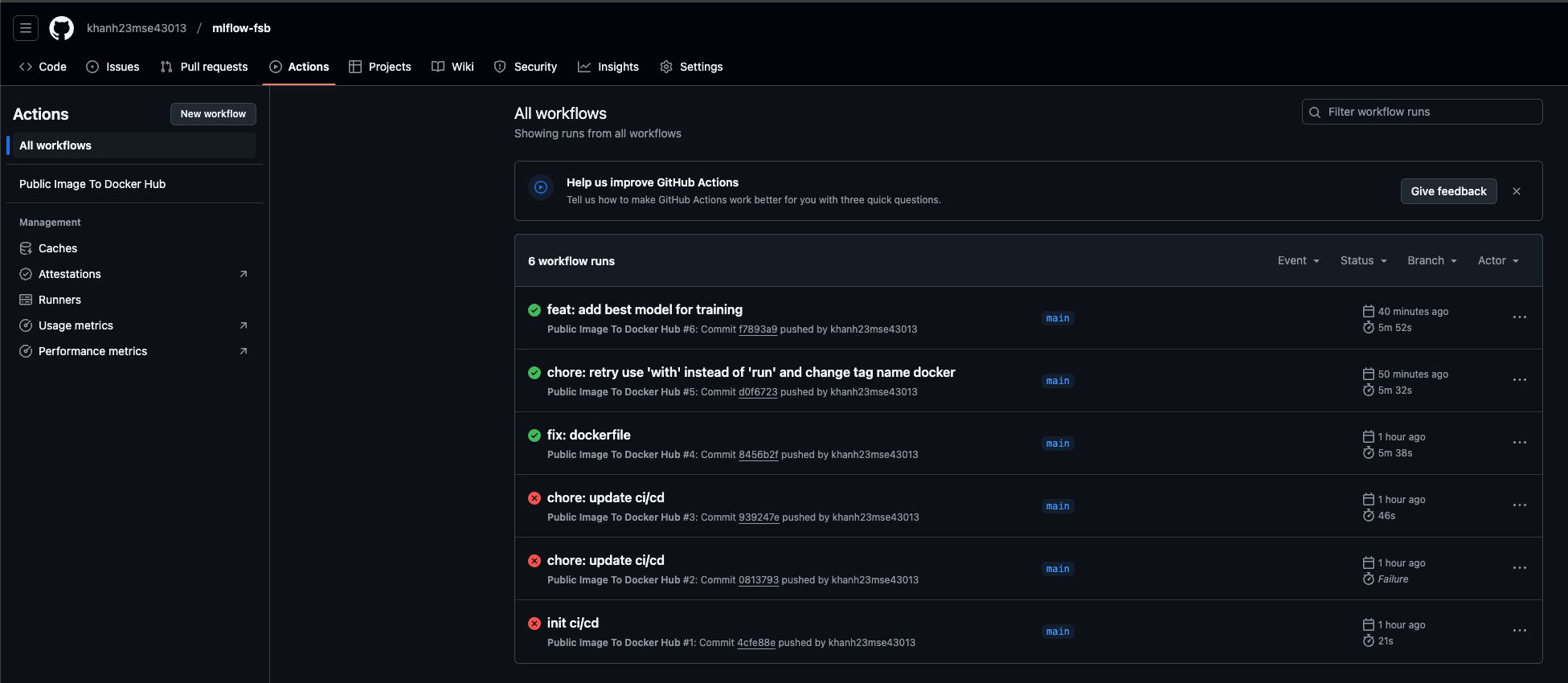
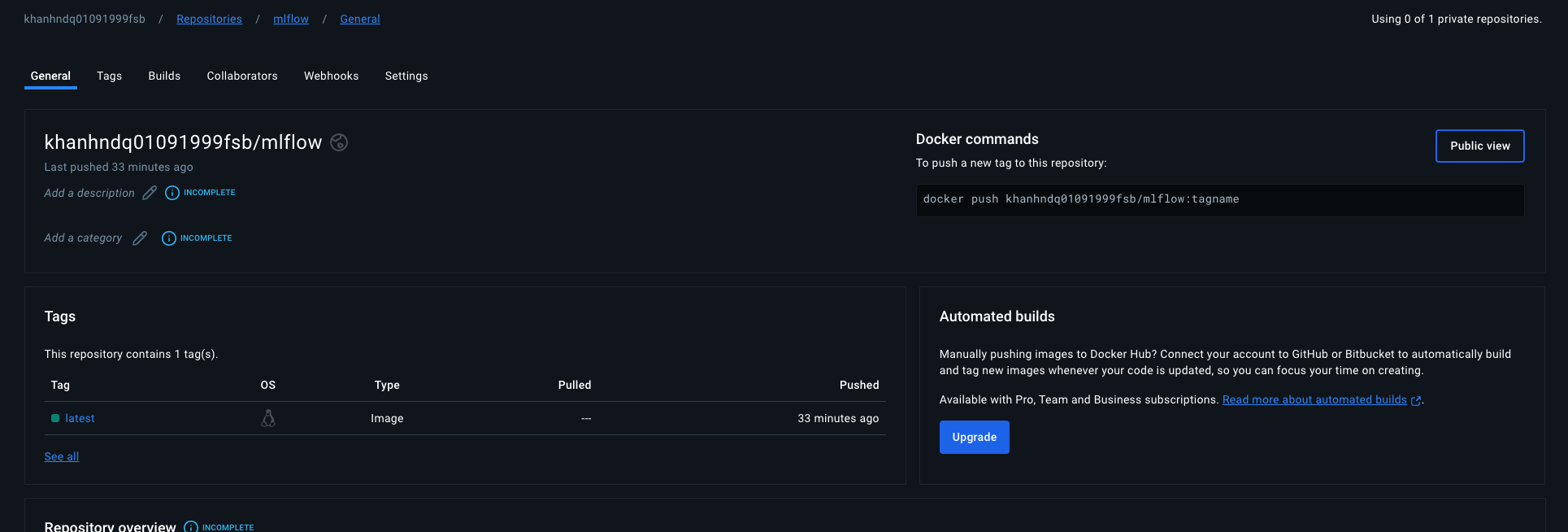
UI of dozzle below:



Step 8: I build a GitHub Action for this project. The project originally required using GitLab CI/CD, but GitLab CI/CD requires a Visa card, which I don’t have. Therefore, I used another approach and resolved it by using GitHub Actions instead.

I was build a build-docker.yaml for build CI/CD on github  
Before doing that, I created an access token on Docker Hub (necessary for pushing Docker images) and set it up in the GitHub secret environment.

When there is a new commit on the main branch:

It will trigger the CI/CD event and push the image to my Docker Hub:

Through this project, I have learned a lot, including how to use MLFlow. MLFlow is a very useful tool for visualizing changes in model training. In addition, I have gained more knowledge about FastAPI and how to use CI/CD (GitHub Actions), as well as Docker Hub. It is a pity that I cannot use GitLab CI/CD; however, I plan to strengthen my knowledge of GitLab CI/CD in the future.