**Chaktim Wong - 50280143**

**chaktimw@buffalo.edu**

**Group Number 75**

**Framework: Flask**

**Database: Mongodb**

The app consists of a Dockerfile, docker-compose, app.py, and html/css files. The dockerfile and docker-compose file creates the python container and the mongodb database. App.py handles the connections to the server and the database connection.

The app can be deployed in the command line with the following command(docker should be running in the background):

**“docker-compose up”**

\* Make sure the command line directory is set to the location of the project files.

After docker finishes initializing, the site will be able to be accessed on localhost:8000. When the user connects to the site, a get request retrieves the html and css files from the server.

On the homepage, there will be a form containing two fields where the user can type in their name and a comment. After inputting data into this form and hitting send, a post request will be sent to the server with the json data. From there, the server will process it, and then with the help of MongoEngine, the data is saved onto the Mongodb database.

Meanwhile, the client who sent the post request is redirected back to the homepage where a get request once again retrieves the homepage. However, this time, the server notices that there is data in the database and retrieves it. This data is added to the html file (using a template replace function in flask) and sent to the client. The client will once again see the homepage with the addition of their comment. Since the database will keep any data stored after a server restart, the client will be able to see their comments even after restarting their connection.