Milestone 4 Description

> *Deadline Tuesday 17/12 @ 11:59PM*

Objectives:

1. Create an ETL pipeline using Airflow
2. Creating a dashboard for the output data

NOTE: You can use the docker compose file and the dockerfile uploaded in lab 9, but make sure to expose the port for the dashbaord from the dockerfile, as well as include all the libraries you need in the requirements file.

Datasets

In this milestone, you will be using the original dataset used in milestones 1 and 2.

NOTE: This milestone **does not** include Kafka from milestone 2, so when using the functions from m2 do not include kafka functions.

Requirements

Part 1: Copying your functions

In this part you should copy all your functions from mielstone 2, bundle them into 3 functions in demonstrated in the lab

- `extract_clean`: this function should extract the uncleaned csv file from the data folder, then call all the cleaning functions (or `clean()` function if only one) and then save the intermediate csv file as `fintech_clean.csv/parquet`
- `transform`: this function should extract the cleaned csv file and perform all transformations applied (eg. Log transformations, normalisation, standardization, etc.), then save the transformed file to `fintech_transformed.csv/parquet`
- `load_to_db`: this function reads the transformed csv file and load it to the postgres database that we added to the `docker-compose.yaml` file in the lab, which was called `pgdatabase` container.

NOTE, in this milestone you can include the `functions.py` file in the dag folder for simplicity and import the functions in the dag file using `from functions import ...`, but in real life, it is a good practice to include your functions in an external directory.

Part 2: Creating the dashboard code

In this section you will be first trying out how to create a dashboard using the tools we mentioned in the lab using your cleaned dataset from milestone 1, then after all is working fine, you will add it to the dag which will be covered in the next section in details.

The code for the dashboard part should all be in a python file called `fintech_dahsboard.py`, with a big function called `create_dashboard(...)` that will then be called from within the dag file as well. You can also place this file in the dags folder alongside the functions file.

You are allowed to use either `plotly dash` or `streamlit` which were covered in the lab.

IMPORTANT: Your dashboard should include a title of your choice and a subtitle or small paragraph underneath it with your name and id. Also for each question, include the question as a title above its graph.

NOTE: the hinted graphs are only to help you, but you are free to use any other chart/graph type for each question. **Create a dashbaord to answer the following questions.**

1. What is the distribution of loan amounts across different grades? (use letter grades or encoded grades (1-7) not grades from the uncleaned file)
    - **Hint**: Use a box plot or violin plot to show the spread of loan amounts for each grade.
2. How does the loan amount relate to annual income across states ? **(Interactive)**
    - **Hint**: Use a scatter plot with loan amount and annual income (original values).
    - Add color-coding based on loan status (e.g., fully paid, default).
    - Have a dropdown with all states to filter **either** each state (unique value of states) or `all` which shows all states (*Hint*: you need to check if the value is 'all' don't apply a filter), make `all` the default and first option.
3. What is the trend of loan issuance over the months (number of loans per month), filtered by year? **(Interactive)**
    - **Hint**: Use a line graph showing the count or total loan amount issued per month.
    - Filter using a dropdown with the years available.
4. Which states have the highest average loan amount?
    - **Hint**: Use a bar chart to display average loan amounts for each state, or an interactive choropleth map to enhance the visualization (optional), showing states shaded by their average loan amounts.
5. What is the percentage distribution of loan grades in the dataset?
    - **Hint**: Use a histogram or kde plot.

> ***Note**: If a question is annotated Interactive you have to make it interactive regardless of the graph you choose to make.*

Part 3: Creating the DAG

Using the code template we had in the lab, create a data pipeline using Airflow called `fintech_dag.py`, with the following tasks

```
extract_clean --> transform  --> load_to_db --> run_dashboard
```

Make sure to save the intermediate datasets after each task, since airflow does not allow you to pass dataframes between tasks.

Part 4: Submission Video Recording

After you are done with the pipeline, with all tasks running sucessfully and no errors occur, you are asked to record a quick video showing the following

- Airflow ui DAGs page with your dag present

- Run the DAG from the airflow dashboard, showing all is running correctly
- Open the dashboard you created from your browser and showcase your dashbaord, by showcase I mean for each part, recite the question, then talk about the chart you chose, while showing the filters (if there are ones) you made.

Your video is expected to be from 5-10 minutes, but if it is less or more (not much) there is no problem.

BONUS: Exceptional Dashboard UI

Love building dashboards? Here’s your chance to shine! Blow us away with your creativity and skills, it’s your moment to stand out and leave your mark!

> *Note that: This bonus allows you to earn up 5% extra on this milestone. Which is only 0.25% of the whole course, so if you are not interested in building cool dashboards with exceptional UI, or don't have time, just don't bother to do it.*

Deliverables

All the files mentioned below should reside in a folder called milestone 4 on you root drive folder created previously in milestone 1.

1. dags folder including
    - DAG file you created (`fintech_dag.py`)
    - functions file (`functions.py`)
    - dashboard file (`fintech_dashbaord.py`)
2. Video recording having the name `fintech_dahsboard_showcase_{your_id}`

Submission guidelines

Upload all the deliverables in your google drive milestone folder.

Best of luck.

.