**Project Setup**

* Create a GCP account
* After that, create a bucket under the project (Make sure to unclick the “Enforce public access prevention on this bucket” option. Otherwise, we cannot access this bucket folder)
* Then upload the csv file to that newly create bucket.
* Next step is to build the virtual environment
  + Before making any codes, we must create a virtual environment first
  + For that, we must create a folder in our local machine
  + Start the terminal of this folder and run below codes
    - python -m venv venv
    - source venv/bin/activate
    - Make below sub folders and python script
      * templates
      * src (Inside this folder, create a python file called “\_\_init\_\_.py”)
      * config (Inside this folder, create a python file called “\_\_init\_\_.py”)
      * utils (Inside this folder, create a python file called “\_\_init\_\_.py”)
      * pipeline (Inside this folder, create a python file called “\_\_init\_\_.py”)
      * artifacts
      * static
      * setup.py (Project management code will be here). Add below code to this setup.py file

A computer screen shot of a program code

AI-generated content may be incorrect.

* Run this code now: pip install -e .
* Then come to the “src” folder. Under that folder, create a python file called “logger.py” and paste below code

A computer screen shot of a program

AI-generated content may be incorrect.

* Then come to the “src” folder again. Under that folder, create a python file called “custom\_exception.py” and paste below code

A computer screen shot of a program

AI-generated content may be incorrect.

**Data Ingestion**

* In this stage, we are getting data from google bit bucket to vs code.
* First, we must create a service account. Before making the service account we must install google cloud CLI to our local machine. [Link](https://cloud.google.com/sdk/docs/install)
* After installing it, add “scikit-learn” and “google-cloud-storage” libraries into requirements file. Then run “pip install -e .” code
* Then navigate to GCP 🡪 IAM Admin 🡪 Service Accounts 🡪 create new service account 🡪 give a name to the service account 🡪 under role, select “storage admin” 🡪 under role, select “storage object viewer” 🡪 Done
* Go to buckets 🡪 select the bucket name 🡪 click three dots 🡪 Edit access 🡪 Add principal 🡪 Under new principal tab, add the service account 🡪 again give the same roles (storage admin and storage object viewer) and click save
* Again navigate to the service account 🡪 Now we have to create a key 🡪 Click three dots 🡪 manage keys 🡪 Add key 🡪 create new key 🡪 JSON 🡪 Create
* Come to VS code agin 🡪 type below code in the terminal

export GOOGLE\_APPLICATION\_CREDENTIALS=/Users/thilina/Downloads/mlops-thilina-d275217160f5.json

* Create a file called “config.yaml” under config folder and paste below code

A screenshot of a computer

AI-generated content may be incorrect.

* Create a file called “paths\_config.py” under config folder and paste below code

A computer screen shot of a code

AI-generated content may be incorrect.

* Create a file called “common\_functions.py” under utils folder and paste below code. This python file will use to run yaml files

A screen shot of a computer program

AI-generated content may be incorrect.

* Create a file called “data\_ingestion.py” under src folder and paste below code.

A screenshot of a computer program

AI-generated content may be incorrect.

A computer screen with text and images

AI-generated content may be incorrect.

A computer screen with text on it

AI-generated content may be incorrect.

**Data Training**

**Save Processed Datasets:**

* Update the “paths\_config.py” file

A screen shot of a computer code

AI-generated content may be incorrect.

* Update the “config.yaml” file
* Create the “data\_preprocessing.py” file under src folder

**Experiment & Model Tracking**

* Add new lines to “config\_path.py” file

A screen shot of a computer

AI-generated content may be incorrect.

* Create a new file called “model\_params.py” under config folder

A screenshot of a computer program

AI-generated content may be incorrect.

* Create a new file called “model\_training.py” under src folder (it consist with mlflow as well)
* To run each src folder python file, we can use below code

cd /Users/thilina/Downloads/MLOps\ Projects/Project\_01

python -m src.model\_training

* If we want to see those results in mlflow dashboard, we can run “mlflow ui” code

**Data versioning and Code versioning**

* Create a new folder called “pipeline” and inside that make a file called “training\_pipeline.py”

A screen shot of a computer program

AI-generated content may be incorrect.

* Run

cd /Users/thilina/Downloads/MLOps\ Projects/Project\_01

python -m pipeline.training\_pipeline

* Make a GitHub repo and push the code, after creating “.gitignore” file (it will fill with files and folders that do not need to push to GitHub repo)

**User App Building**

**CI/CD Deployment**