**Day-by-Day Plan to Complete the Project**

**✅ Day 1 – Setup & Dataset Preparation**

**Goal:** Set up your environment and prepare the dataset

**Tasks:**

* Create your project folder: bagging\_mlops\_project/
* Install necessary libraries:

bash

CopyEdit

pip install scikit-learn pandas mlflow dvc

* Create folder structure:

css

CopyEdit

bagging\_mlops\_project/

├── data/

├── models/

├── src/

* Download or generate dataset (e.g., Iris):

python

CopyEdit

from sklearn.datasets import load\_iris

import pandas as pd

df = load\_iris(as\_frame=True).frame

df.to\_csv("data/iris.csv", index=False)

**Deliverables:**

* data/iris.csv
* Folder structure ready

**✅ Day 2 – Model Building with Hyperparameter Tuning**

**Goal:** Train a Bagging model and optimize it

**Tasks:**

* Write src/train.py to:
  + Load dataset
  + Train BaggingClassifier
  + Use GridSearchCV for tuning
  + Evaluate with accuracy
  + Save model using pickle

**Sample Command:**

bash

CopyEdit

python src/train.py

**Deliverables:**

* src/train.py
* models/best\_model.pkl

**✅ Day 3 – Integrate MLflow (Experiment Tracking)**

**Goal:** Add MLflow logging for reproducibility and versioning

**Tasks:**

* Modify train.py to:
  + Log hyperparameters
  + Log accuracy
  + Log trained model
* Initialize MLflow tracking

bash

CopyEdit

mlflow ui

* Open: <http://localhost:5000>

**Deliverables:**

* Tracked runs in MLflow UI
* Logged model, metrics, params

**✅ Day 4 – Use DVC for Data & Model Versioning**

**Goal:** Track data and model versions using DVC

**Tasks:**

* Initialize DVC

bash

CopyEdit

dvc init

* Track dataset:

bash

CopyEdit

dvc add data/iris.csv

* Track model:

bash

CopyEdit

dvc add models/best\_model.pkl

* Add and commit:

bash

CopyEdit

git add data/iris.csv.dvc models/best\_model.pkl.dvc .dvc .gitignore

git commit -m "Track data and model with DVC"

**Deliverables:**

* .dvc files for dataset and model
* Git commits with DVC

**✅ Day 5 – Finalize, Document & (Optional) Push to GitHub**

**Goal:** Wrap up project and prepare for presentation or upload

**Tasks:**

* Create README.md:
  + Project purpose
  + Setup instructions
  + How to run
  + How to view MLflow UI and DVC files
* Create requirements.txt

bash

CopyEdit

pip freeze > requirements.txt

* (Optional) Push to GitHub:

bash

CopyEdit

git remote add origin <your\_repo\_url>

git push -u origin main

**Deliverables:**

* README.md
* GitHub repo (if applicable)
* Final project ready to showcase