

FINAL TERM PROJECT (CC105)

Objective

Create a Django-based web application that:

- Trains and uses your own machine learning model.
- Integrates the trained model into the Django app.
- Provides a user-friendly prediction interface.
- Displays a dashboard with simple statistics and charts.
- Includes a basic user authentication system (login, register, logout).

This project allows you to showcase your full learning: from model training, Django app development, user management, and basic data visualization.

Requirements

1. Machine Learning Model

- Train your own machine learning model using a dataset of your choice.
- Save the trained model (e.g., using joblib, pickle, or TensorFlow SavedModel).
- Document the training process:
 - Dataset overview
 - Preprocessing steps
 - Model architecture/algorithm used
 - Training results (accuracy, loss, etc.)

2. Django App Setup

- Create a Django project and at least one app inside it.
- Pages required:
 - Home (Project intro)
 - Login/Register (Authentication pages)
 - Predict (Form input for predictions — only for logged-in users)
 - Dashboard (Statistics and charts — only for logged-in users)

3. Machine Learning Integration

- Load the trained ML model into Django (not during training).
- Accept user input through a form.
- Preprocess and pass the input to the model for prediction.
- Show prediction results nicely.

4. Simple Dashboard (Statistics and Visualization)

Display on your dashboard page:

- Dataset statistics (e.g., number of records, feature summary, target distribution).
- Model performance (e.g., accuracy score, confusion matrix, or loss value).
- At least 1–2 simple charts (e.g., bar chart, pie chart, line chart).

5. Authentication Feature (Simple User Management)

- Use Django's built-in authentication system (`django.contrib.auth``).
- Features required:
 - User Registration (sign up new users)
 - Login (authentication)

- Logout
- Restrict the Prediction and Dashboard pages to logged-in users only.
- Home page is accessible to everyone (even guests).

Deliverables

- Django project source code (GitHub link or ZIP file).
- Model training notebook or script (.ipynb or .py).
- Project documentation (Markdown or PDF) including:
 - Dataset description
 - ML model training steps
 - How authentication was added
 - Steps for integration
 - Challenges encountered
- 2-5 minutes screen recording demo of:
 - Logging in
 - Making a prediction
 - Viewing dashboard
 - Explaining model training and app integration

Deadline

Submission Date: May 10, 2025 (drive link)

Presentation Date: May 12,13,14, 2025

Evaluation Rubric

Criteria	Description	Points
Django App Structure	Proper project and app setup, working URLs	10
Machine Learning Model	Own-trained model, saved and documented	20
Model Integration	Correct loading, input preprocessing, prediction	20
Dashboard	Basic statistics and at least 1 simple visualization	15
Authentication	Working login, register, logout; page restrictions	15
Functionality	No major bugs, good user flow	10
Documentation & Code	Clear, well-commented, properly explained	5
Presentation Demo	Short, clear project walkthrough	5
Total		100