# Introduction

Project Title: Pattern Sense: Classifying Fabric Patterns using Deep Learning

Team Members:

* Deepika Rama Lakshmi V. Model Design & Training

* Jeevan Nissy P. Data Collection & Labeling

* Ramdevu Vijay Frontend Integration

* Jagadeswari P. Backend Development & Deployment

# Project Overview

Purpose:

To automate the process of identifying and categorizing fabric patterns using deep learning, reducing manual labor and increasing efficiency in industries like fashion, textiles, and interior design.

Features:

* Image upload and prediction interface

* Classification of fabric patterns (e.g., stripes, floral, polka dots, geometric)

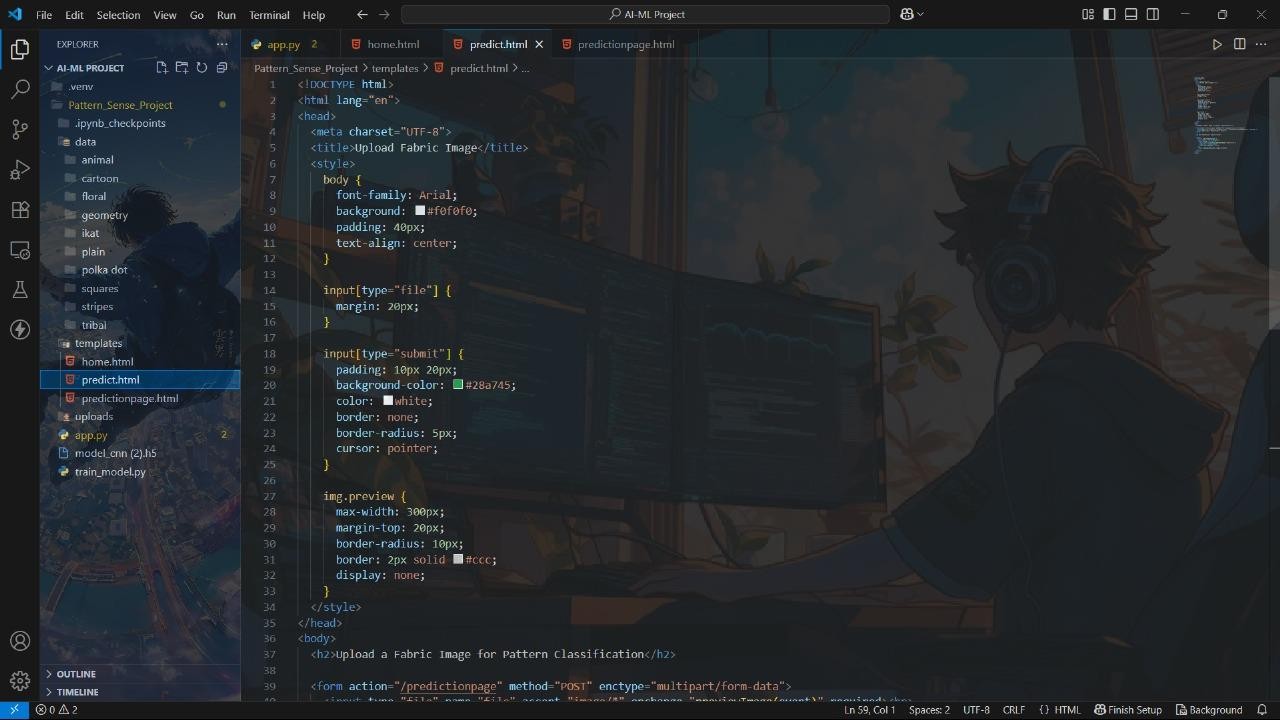
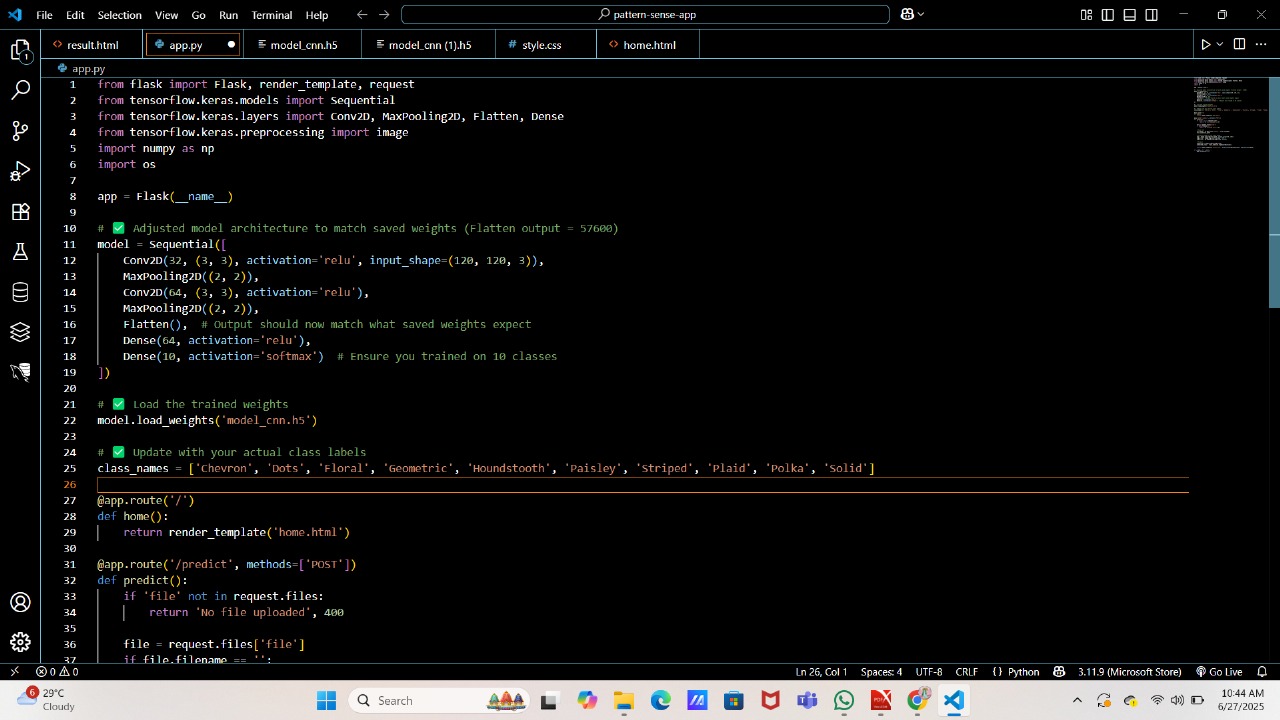
* REST API for integration

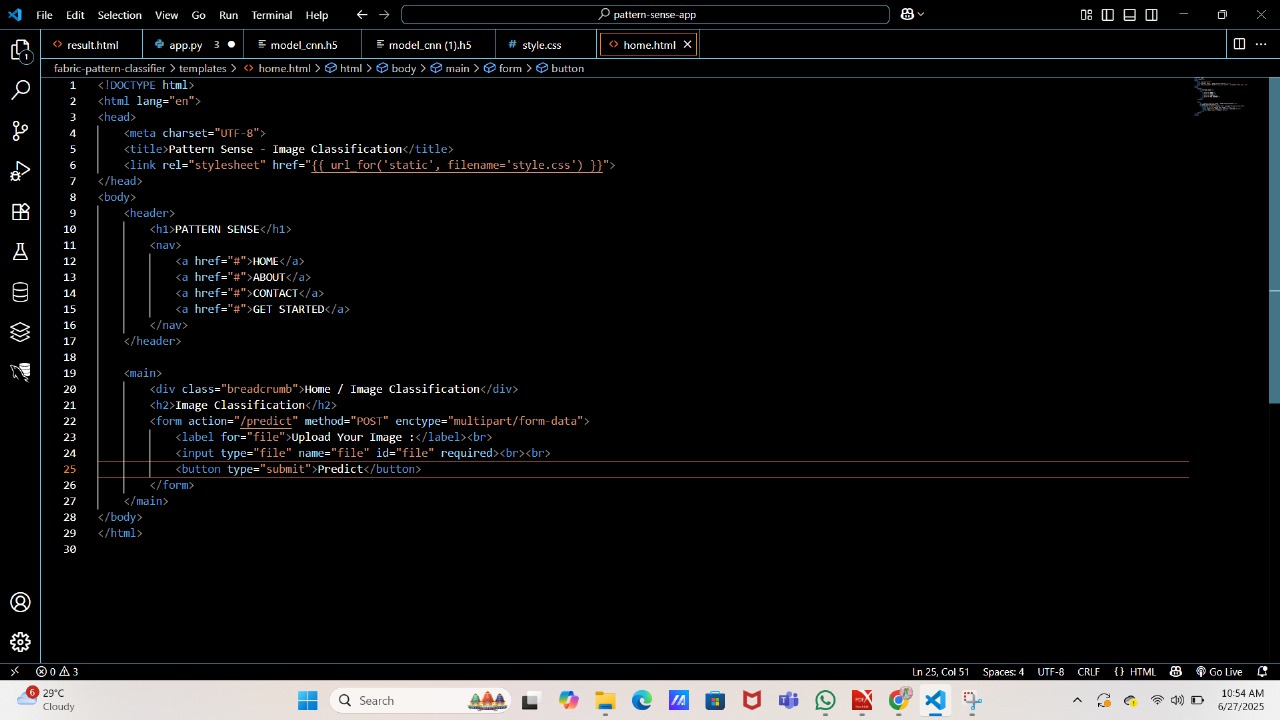
* Optional anomaly detection for pattern defects

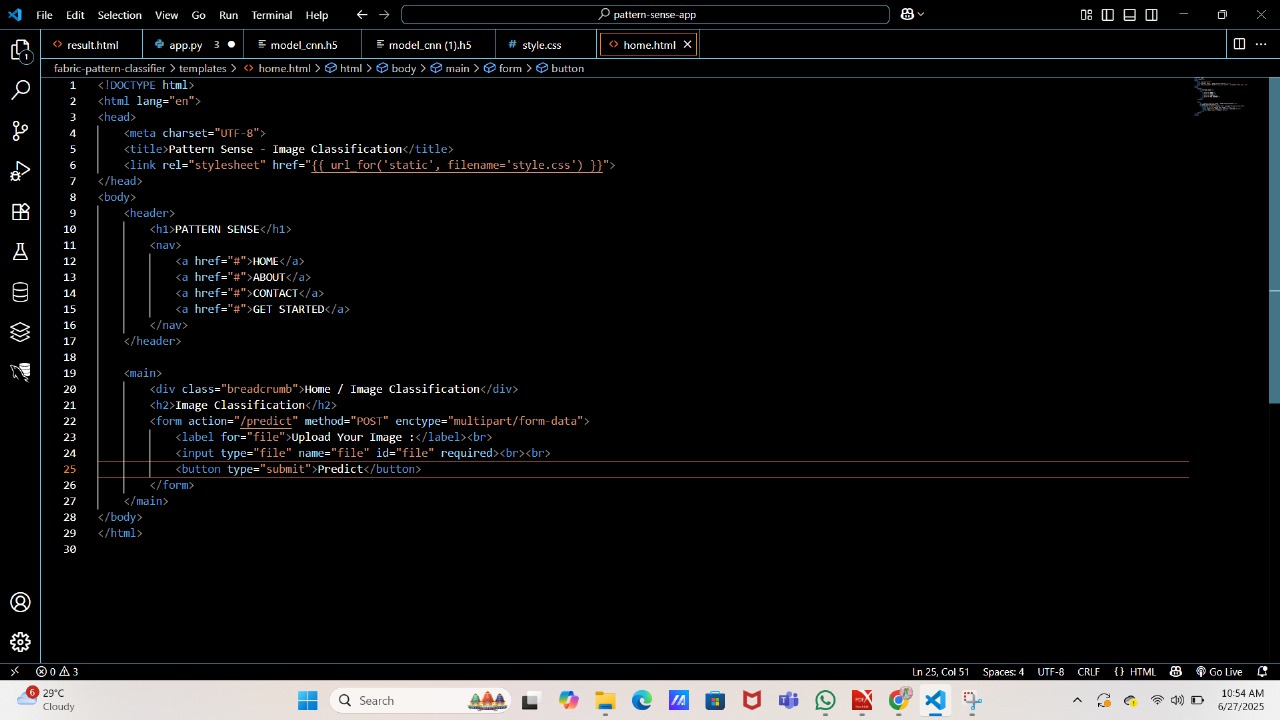
# Architecture

**Frontend:**

* Simple React/HTML form for uploading images and displaying predictions







**Backend:**

* Flask application exposing prediction API

* Uses trained CNN model (e.g., VGG16/ResNet-based)

Database (Optional):

* SQLite or MongoDB for storing user uploads and results (if extended to a full-stack app)

# Setup Instructions

Prerequisites:

* Python 3.8+

* Flask

* TensorFlow / Keras

* NumPy, OpenCV, Pillow

Installation:

git clone https://github.com/your-repo/pattern-sense cd pattern-sense

pip install -r requirements.txt

# Folder Structure

pattern-sense/ app.py model/ fabric\_classifier.h5 static/

uploads/ templates/ index.html

utils/

preprocessing.py requirements.txt

# Running the Application

Backend:

python app.py

Frontend:

- Open browser at [http://127.0.0.1:5000](http://127.0.0.1:5000/)

# API Documentation

Endpoint: POST /predict

Request:

- file (form-data): image file of fabric

Response:

{

"prediction": "Floral"

}

# Authentication

No authentication currently used (can be added via JWT for user-specific sessions in full-stack deployment).

# User Interface

* Upload form with preview

* Display of predicted class

* Simple Bootstrap styling

# Testing

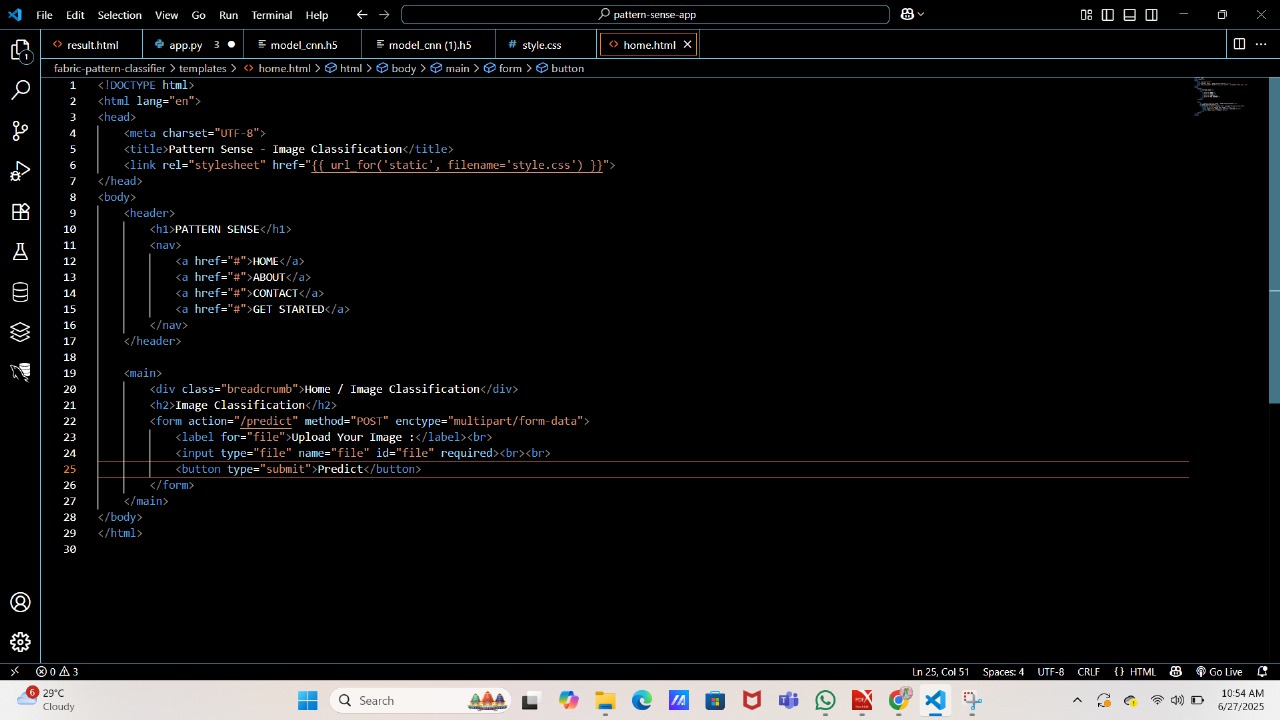
* Manual testing with a test dataset

* Accuracy metrics evaluated using validation data

* Future: add unit tests for API endpoints

# Screenshots or Demo

\_Include screenshots in the "static/screenshots" folder for rendering.\_



# Known Issues

* Model accuracy may vary with poorly lit or low-resolution images

* No user login system yet

# Future Enhancements

* Add user authentication and login

* Extend to mobile interface (React Native / Flutter)

* Train on larger and more diverse datasets

* Add pattern defect detection (for quality control use cases)

# Scenarios

Scenario 1: Fashion Industry

Fashion designers and manufacturers can use Pattern Sense to automatically classify patterns like stripes, floral prints, polka dots, and geometric designs, saving time and effort in manual categorization.

Scenario 2: Textile Quality Control

Textile manufacturers can leverage the tool to ensure quality control by analyzing pattern consistency and identifying defects automatically.

Scenario 3: Interior Design

Interior designers can use the system to match pattern styles with design themes quickly and efficiently, streamlining material selection for upholstery, curtains, and furniture.