Ayna ML Assignment

Welcome, prospective ML Intern! This assignment is designed to evaluate your understanding of deep learning concepts and your ability to implement and deploy machine learning models.

# Problem Statement

Your task is to train a UNet model from scratch to generate an image of a colored polygon. The model should take two inputs:

1. An image of a polygon (e.g., triangle, square, octagon).
2. The name of a color (e.g., "blue", "red", "yellow").

The model's output should be an image of the input polygon filled with the specified color.

# Dataset

The dataset required for this assignment can be accessed via the following link: [dataset.zip](https://drive.google.com/open?id=1QXLgo3ZfQPorGwhYVmZUEWO_sU3i1pHM)

The dataset contains the following structure:

* dataset/training/inputs/: Contains images of various polygons.
* dataset/training/outputs/: Contains images of colored polygons, corresponding to the inputs.
* dataset/training/data.json: A JSON file mapping input polygons and colors to their corresponding output images for training.
* dataset/validation/inputs/: Contains images of various polygons for validation.
* dataset/validation/outputs/: Contains images of colored polygons for validation.
* dataset/validation/data.json: A JSON file mapping input polygons and colors to their corresponding output images for validation.

You may also enhance the dataset—for example with paired input/output augmentations (rotate/scale) or by generating synthetic polygons.

# Requirements

1. **Model Implementation:**
   * Implement a UNet model from scratch. Prefer to use PyTorch as your deep learning framework.
   * Ensure your model is capable of taking both the polygon image and the color name as input.
2. **Experiment Tracking:**
   * Track your training runs using [wandb](https://wandb.ai/site/) (Weights & Biases).
   * Share your [wandb](https://wandb.ai/site/) project with us after the completion of the assignment.
3. **Inference and Testing:**
   * Create a Jupyter Notebook (.ipynb) to demonstrate the inference and testing of your trained model.
   * The notebook should clearly show how to load the model, provide example inputs (polygon image and color name), and visualize the generated output image.
4. **Report & Insights:**

Share a concise report (≈1–2 pages or a README.md) covering

* + Hyperparameters: what you tried, rationale, and final settings.
  + Architecture: UNet design/conditioning choices and any ablations.
  + Training dynamics: loss/metric curves, qualitative output trends, typical failure modes, and fixes attempted.
  + Key learnings

# Resources

1. **Conditional UNet reference (Diffusers):**
   * You can refer to the Hugging Face Diffusers [UNet2DConditionModel](https://huggingface.co/docs/diffusers/main/api/models/unet2d-cond) API and build on top of it if you prefer a conditioned UNet implementation.
2. **Free GPU options (T4/P100):**

You can train this assignment on **T4/P100 GPUs** available on:

* + [Google Colab](https://colab.research.google.com/) (free tier)
  + [Kaggle Notebooks](https://www.kaggle.com/code) (free tier)

# Deliverables

Please submit the following:

1. Your code for the UNet model implementation and training script.
2. The Jupyter Notebook (.ipynb) for inference and testing.
3. A link to your [wandb](https://wandb.ai/site/) project for tracking.
4. Insights report.

# Time to Complete

You have **3 days** to complete this assignment from the date of receipt. Fill this [form](https://docs.google.com/forms/d/e/1FAIpQLScQcX6v-MlsUsR5vaRVtyOCgjHgqQtnCXRh_uXQG-T5FTszXQ/viewform?usp=sharing&ouid=112888232990585572987) to submit the assignment.

# Support

If you have any questions or encounter issues, please contact [vatsa@getayna.com](mailto:vatsa@getayna.com).

We look forward to reviewing your submission!