**Final Project: Model Deployment**

**Name: Email:**

Statement:

For this assignment's preparation, the author(s) did not use any generative AI tools.

For this assignment's preparation, the author(s) have utilized [Generative AI Tool Name], a language model created by [Generative AI Tool Provider]. Within this assignment, the [Generative AI Tool Name] was used for purposes such as [e.g., brainstorming, grammatical correction, writing paraphrasing, citation, specific sections of the assignment].”

**Objective**: Train a machine learning model and deploy it as a web service using cloud platforms or local deployment tools. Understand the complete process of transforming a data science model into a consumable API endpoint.

**Tasks**:

1. **Dataset Selection & Model Training**:
   * Choose a dataset (suggested datasets provided below) or use the model from a previous assignment.
   * Train a machine learning model on this dataset.
   * Serialize and save the trained model for deployment.
2. **Deployment Preparation**:
   * Choose a deployment tool or platform (e.g., Flask, FastAPI for local deployment; AWS Lambda, Google Cloud Run, or Heroku for cloud deployment).
   * Write a wrapper function or API endpoint that takes in input data, processes it, and returns predictions from the trained model.
3. **Deployment**:
   * If using a local tool, set up the web service and ensure it can accept requests and return predictions.
   * If using a cloud platform, follow the platform's guidelines to deploy your model and create an accessible endpoint.
4. **Testing**:
   * Once deployed, test the model endpoint using sample data. Ensure it's responsive and returns correct predictions.
   * Consider using tools like Postman to send requests to your API.
5. **Documentation**:
   * Write a brief documentation on how to use your API: the endpoint URL, the format of input data required, the expected output, etc.
   * Reflect on challenges faced during the deployment process and any lessons learned.

**Suggested Datasets & Tasks**:

1. **Tabular Data**:
   * [UCI Wine Quality Dataset](https://archive.ics.uci.edu/ml/datasets/wine+quality): Predict the quality of wine based on physicochemical tests.
     1. <https://archive.ics.uci.edu/dataset/186/wine+quality>
2. **Image Data**:
   * [MNIST](http://yann.lecun.com/exdb/mnist/): Handwritten digit classification. Users could send an image of a handwritten digit to the API, and it returns the predicted digit.
     1. <http://yann.lecun.com/exdb/mnist/>
3. **Text Data**:
   * [IMDb Reviews](https://www.kaggle.com/lakshmi25npathi/imdb-dataset-of-50k-movie-reviews): Sentiment analysis on movie reviews. Users send a movie review text, and the API returns a sentiment (positive/negative).
     1. <https://www.kaggle.com/datasets/lakshmi25npathi/imdb-dataset-of-50k-movie-reviews>

**Assessment Criteria**:

1. **Model Quality** (20%): The accuracy and robustness of the trained model.
2. **Deployment Implementation** (20%): Proper setup of the deployment environment and functioning of the API.
3. **Testing** (10%): Successful testing of the deployed model with various inputs.
4. **Documentation Quality** (40%): Clarity, comprehensiveness, and correctness of the API documentation.
5. **Reflection & Challenges** (10%): Insightfulness of reflections on the deployment process and problem-solving for challenges faced.