**Phase-2 Submission Template**

**Student Name:** [**A.Kiruthiga**]

**Register Number:** [620323104049]

**Institution:** [Bharathiyar Institute of Engineering for Women]

**Department:** [B.E(CSE)]

**Date of Submission:** [Insert Date]

**Github Repository Link:** [Update the project source code to your Github Repository]

### **1. Problem Statement**

1. Problem Statement

* **Real-World Problem:**

The project aims to recognize handwritten digits using deep learning techniques. This is crucial for applications such as automated postal sorting, bank check processing, and digit recognition in various forms of data entry.

* **Type of Problem:**

This is a classification problem where the goal is to classify images of handwritten digits (0-9).

* **Importance of Solving the Problem:**

Solving this problem enhances the efficiency of data entry processes, reduces human error, and improves the speed of digit recognition in various applications, making it highly relevant in the fields of finance, logistics, and education.

### **2. Project Objectives**

### **Technical Objectives:**

Develop a deep learning model capable of accurately classifying handwritten digits.

Optimize the model for high accuracy and low inference time.

* **Model Goals:**

Aim for an accuracy of over 95% on the test dataset, ensuring the model is interpretable and applicable in real-world scenarios.

* **Evolution of Goals:**

The goal has evolved to include not only accuracy but also the model's ability to generalize well on unseen data after initial data exploration.

### **3. Flowchart of the Project Workflow**

* **Visual Representation:**

[Insert a flowchart that visually represents the workflow from data collection, preprocessing, model training, evaluation, to deployment.]

Data collection(MINST dataset)

aatase dddadataset)

Data Processing

(Normalize, reshape, augment images)

Exploratoy Data Analysis

(Visualize distributions and samples)

Feature Engineering

((Augmentation, dimensionality reduction (optional))

Moodel Building

(CNN & Feedforward Neural Network)

Model Evaluation

Accuracy, confusion matrix

### 

Vizualization Of Results

Performance plots & interpretation

I

Deployment

tIntegrate model into application

### **4. Data Description**

* **Dataset Overview:**

The dataset used is the MNIST dataset, which consists of images of handwritten digits.

**Dataset Name and Origin:** MNIST dataset, available from Yann LeCun's website.

* **Types of Data**

Structured data in the form of images (28x28 pixel grayscale images).

* **Dataset Size:**

Contains 60,000 training images and 10,000 testing images.

* **Dataset Nature:**

Static dataset.

* **Target Variable:**

The target variable is the digit represented in each image (0-9).

### **5. Data Preprocessing**

* **Data Cleaning Steps:**

Normalize pixel values to a range of 0 to 1.

Reshape images to fit the input requirements of the neural network.

Augment the dataset with techniques such as rotation and shifting to improve model robustness.

* **Documentation:**

Document and explain each transformation step clearly in code and markdown.

### **6. Exploratory Data Analysis (EDA)**

* **Univariate Analysis:**

Analyze the distribution of digit classes using count plots to ensure balanced classes.

* **Bivariate/Multivariate Analysis:**

Visualize sample images of each digit class to understand the data better.

* **Insights Summary:**

Highlight that the dataset is well-balanced and that certain digits may have more variability in their representation.

### 

### **7. Feature Engineering**

* **Enhancements:**

Create additional features through image augmentation.

Use techniques like PCA for dimensionality reduction if necessary.

* **Justification:**

Justify the use of augmentation to improve model generalization and robustness.

### **8. Model Building**

* **Model Selection:**

Implement Convolutional Neural Networks (CNNs) and compare with a simple feedforward neural network.

* **Justification for Models:**

CNNs are selected due to their effectiveness in image classification tasks.

* **Data Splitting:**

Split the dataset into training, validation, and testing sets.

* **Model Training and Evaluation:**

Train models and evaluate initial performance using metrics such as accuracy and confusion matrix.

### **9. Visualization of Results & Model Insights**

* **Model Behavior Explanation:**

Use confusion matrices and accuracy plots to explain model performance.

* **Visual Comparisons:**

Include visual comparisons of model performance on the validation set.

* **Feature Interpretation:**

Interpret the model's predictions and highlight any misclassifications.

### **10. Tools and Technologies Used**

* **Programming Language:**

Python.

* **IDE/Notebook:**

Jupyter Notebook.

* **Libraries:**

TensorFlow, Keras, NumPy, Matplotlib, Seaborn.

* **Visualization Tools:**

Matplotlib for plotting results.

* **11. Team Members and Contributions**
* **List names and responsibilities**.
* **Clearly mention who worked on:**

**Data cleaning**: R.Madhumitha

**EDA**: P.Kowsalya

**Feature engineering**: A.Kiruthiga

**Model development**: R.Jayasri

**Documentation and reporting**: A.Kiruthiga ,R.Madhumitha , P.Kowsalya , R.Jayasri