**DOCUMENTATION**

In this Handwritten prediciton projects used **MNIST** datset it used **FeedForward Neural Network(FNN)**

**Architecture of FNN**

* Input Layer: 10 neurons (one for each feature)
* Hidden Layer 1: 64 neurons, activation function: ReLU
* Hidden Layer 2: 32 neurons, activation function: ReLU
* Output Layer: 10 neurons (one for each class), activation function: Softmax

**Model of the FNN**

* Sequential model

**Steps in the FNN**

1. **Data Preparation**
2. **Model Initialization**
3. **Forward Propagation**
4. **Loss Calculation**
5. **Backpropagation**
6. **Weight Update**
7. **Model Training**
8. **Model Evaluation**
9. **Prediction**

**Steps in this model**

1. **Data Preparation**mnist = tf.keras.datasets.mnist  
     
   # Collecting dataset  
     
   (x\_train, y\_train), (x\_test, y\_test) = mnist.load\_data()  
     
   **Normalize the input data.**  
     
   x\_train = x\_train.reshape(-1, 28\*28).astype("float32") / 255.0  
   x\_test = x\_test.reshape(-1, 28\*28).astype("float32") / 255.0
2. **Model Initialization**model = Sequential()  
     
   # Input layer (implicitly added when specifying the input\_dim)  
   model.add(Dense(64, activation='relu', input\_dim=10))  
     
   # Hidden Layer 1  
   model.add(Dense(64, activation='relu'))  
     
   # Hidden Layer 2  
   model.add(Dense(32, activation='relu'))  
     
   #Output Layer  
   model.add(Dense(10, activation='softmax'))
3. **Forward Propagation**
4. **Loss Calculation**
5. **Backpropagation**
6. **Weight Update**
7. **Model Training**
8. **Model Evaluation**
9. **Prediction**