

# Assignment 02: Building a Custom CNN for Animal Image Classification

---

## Objective:

Design, train, and evaluate a **custom Convolutional Neural Network (CNN)** from scratch to classify animal images.

---

## Tasks:

### 1. Data Preprocessing:

- Load the **animal image dataset**.
  - Perform the following preprocessing steps:
    - Resize all images to **128x128**.
    - Normalize pixel values to the range `[0, 1]`.
  - Split the dataset into **train, validation, and test** sets:
    - 70% for training
    - 15% for validation
    - 15% for testing
- 

### 2. Build a Custom CNN:

- Design a CNN with the following components:
    - **Input Layer:** Accepts `128x128x3` images.
    - **Convolutional Layers:**
      - Use at least **3 convolutional layers** with **ReLU activation**.
      - Include **MaxPooling** after each convolution.
    - **Fully Connected Layers:**
      - Add **1 hidden dense layer** with 128 neurons and ReLU activation.
      - Add an **output layer** with softmax activation for classification.
    - Use **Dropout** to reduce overfitting.
- 

### 3. Model Training:

- Compile the model using:
  - **Loss:** `categorical_crossentropy`
  - **Optimizer:** `Adam`
  - **Metrics:** `accuracy`

- Train the model for **15 epochs** using the training set.
  - Use the validation set to monitor overfitting.
- 

#### 4. Model Evaluation:

- Evaluate the trained CNN on the **test set**.
  - Report the following metrics:
    - Accuracy
    - Precision, Recall, and F1-Score (classification report)
- 

#### 5. Visualizations:

- Plot the **training and validation accuracy/loss** over epochs.
  - Display the **Confusion Matrix** for test set predictions.
- 

#### 6. Bonus Task (Optional):

- Compare the performance of your custom CNN with a **pre-trained model** like **MobileNetV2** or **ResNet50**.
- 

### Submission Requirements:

- Submit a **Jupyter Notebook** containing:
  - Well-documented code for each step.
  - Output metrics and visualizations.
  - A short conclusion discussing the model's performance.