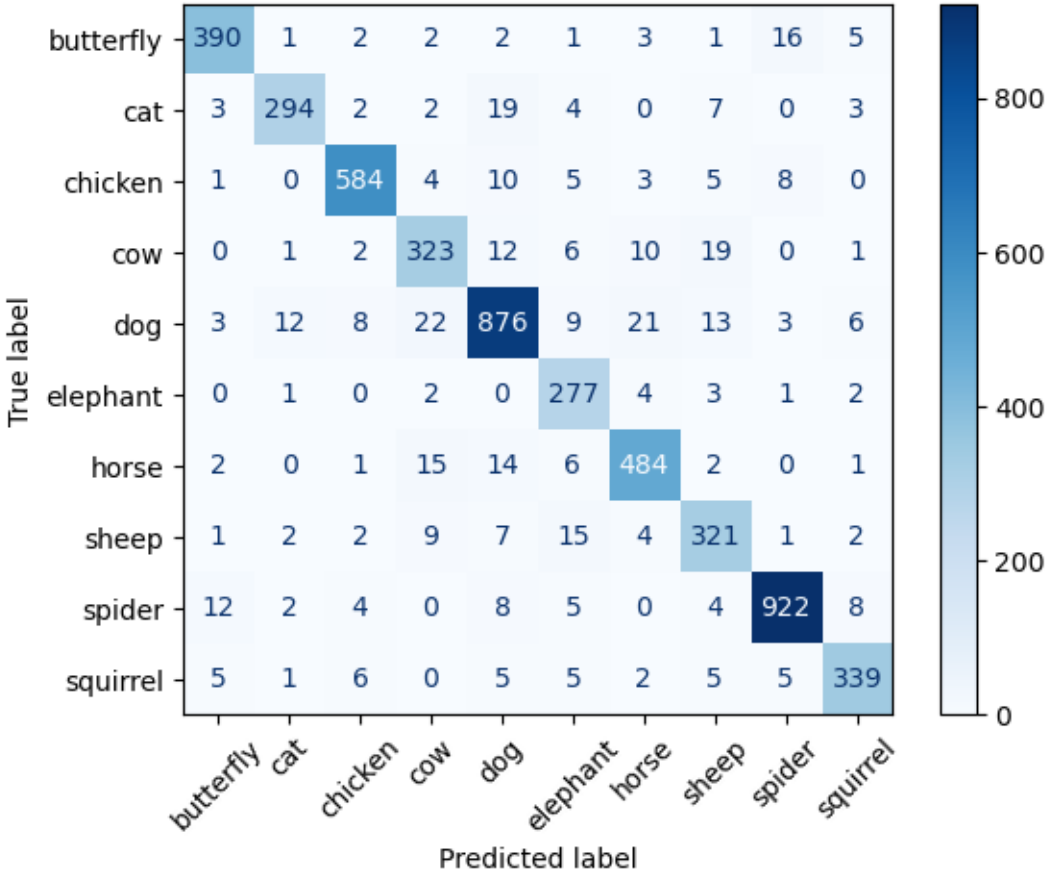
Best Results of the Animals-10 Dataset

Results

Model Evaluation for the validation set

```
82/82 [==============================] - 8s 92ms/step - loss: 0.2825 - accuracy: 0.9178
Validation Loss: 0.2825167179107666
Validation Accuracy: 0.9177637696266174
```

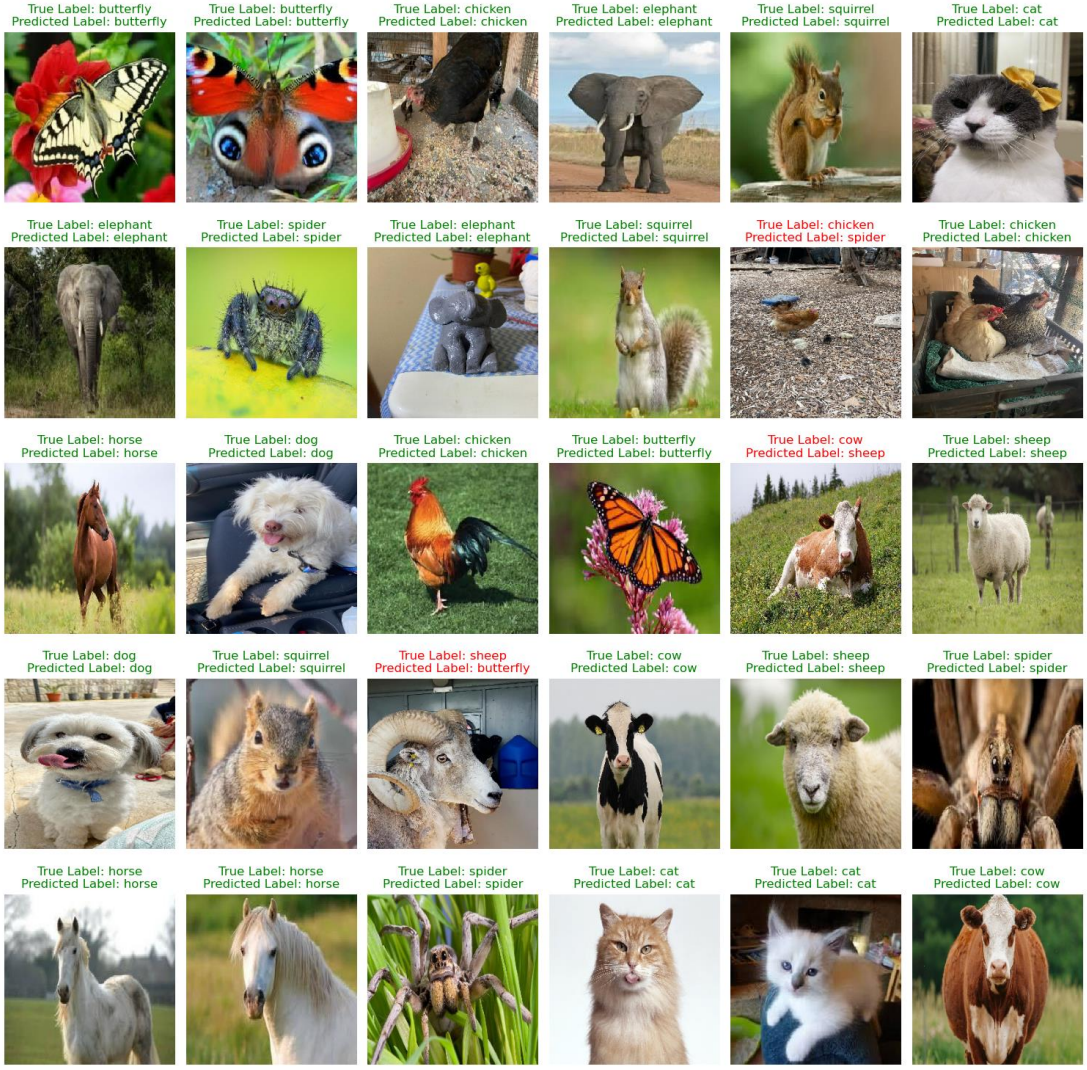
Validation set Confusion Matrix



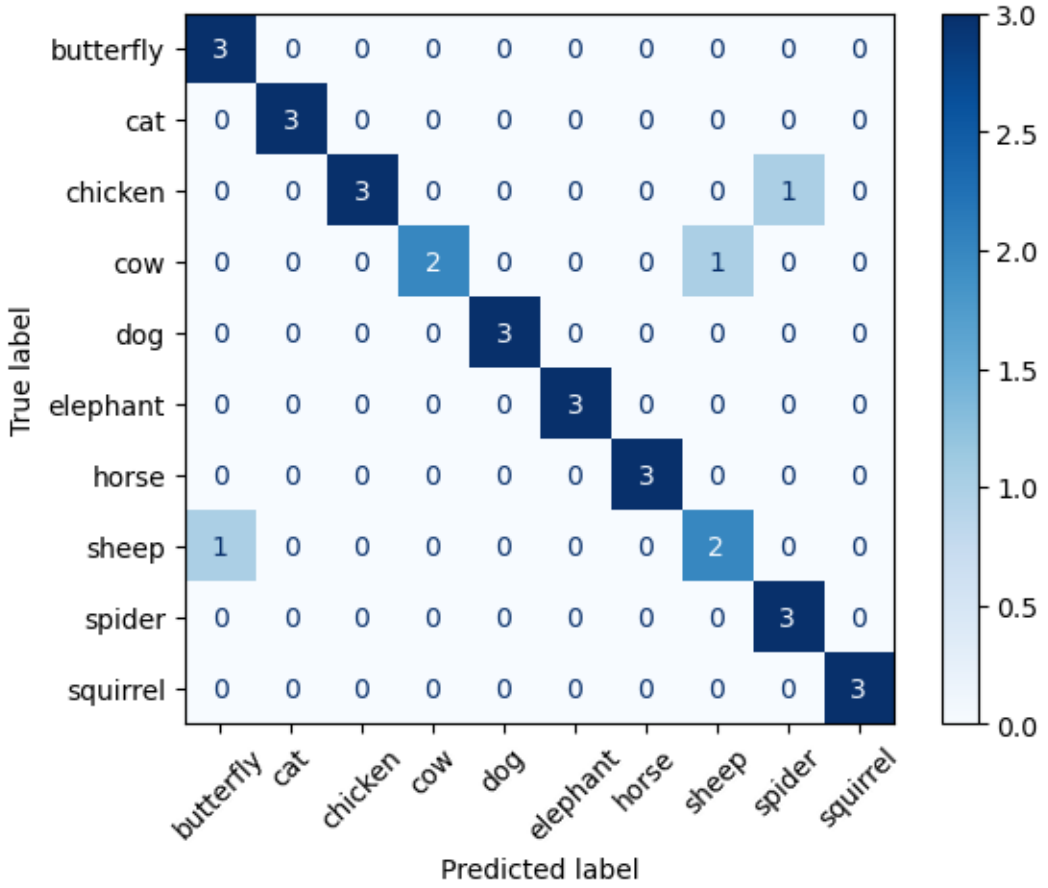
Validation Set Classification Report

```
82/82 [==============================] - 7s 89ms/step
Classification Report:
======================
              precision    recall  f1-score   support

   butterfly       0.94      0.92      0.93       423
         cat       0.94      0.88      0.91       334
     chicken       0.96      0.94      0.95       620
         cow       0.85      0.86      0.86       374
         dog       0.92      0.90      0.91       973
    elephant       0.83      0.96      0.89       290
       horse       0.91      0.92      0.92       525
       sheep       0.84      0.88      0.86       364
      spider       0.96      0.96      0.96       965
    squirrel       0.92      0.91      0.92       373

    accuracy                           0.92      5241
   macro avg       0.91      0.91      0.91      5241
weighted avg       0.92      0.92      0.92      5241
```

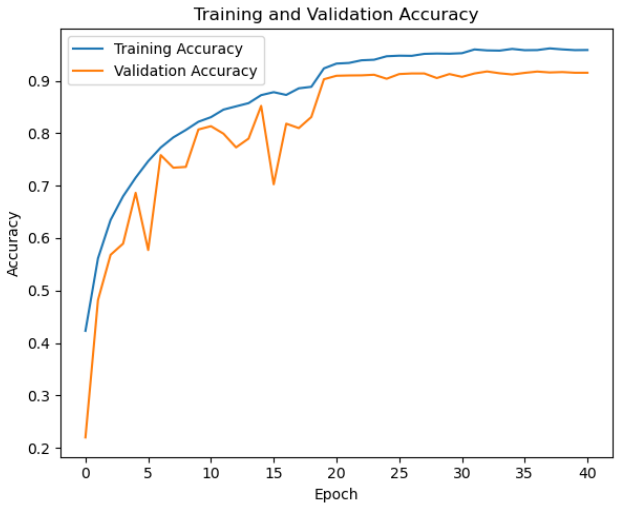
Random 30 pictures displayed from the Validation Set



Test set Confusion Matrix



Validation set accuracy and loss



Best Model Architecture

```
model = Sequential([
        Conv2D(32, (3, 3), activation='relu', input_shape=(target_size, target_size, 3)),
        BatchNormalization(),
        MaxPooling2D((2, 2)),
        Conv2D(64, (3, 3), activation='relu'),
        BatchNormalization(),
        MaxPooling2D((2, 2)),
        Conv2D(128, (3, 3), activation='relu'),
        BatchNormalization(),
        MaxPooling2D((2, 2)),
        Conv2D(256, (3, 3), activation='relu'),
        BatchNormalization(),
        MaxPooling2D((2, 2)),
        Conv2D(512, (3, 3), activation='relu'),
        BatchNormalization(),
        MaxPooling2D((2, 2)),
        Conv2D(1024, (3, 3), activation='relu'),
        BatchNormalization(),
        MaxPooling2D((2, 2)),
        Flatten(),
        Dense(512, activation='relu'),
        BatchNormalization(),
        Dense(10, activation='softmax')
])
```

Used Model Callbacks

```python
# Compile the model
model.compile(optimizer='adam', loss='categorical_crossentropy', metrics=['accuracy'])

# Define callbacks:
early_stopping = EarlyStopping(monitor='val_accuracy', patience=8, restore_best_weights=True)
model_checkpoint = ModelCheckpoint('best_model2.h5', monitor='val_accuracy', save_best_only=True)
reduce_lr = ReduceLROnPlateau(monitor='val_accuracy', factor=0.1, patience=4, verbose=1, mode='auto', min_delta=0.0001, cooldown=0, min_lr=0)

print("Model initialized and Callbacks defined.")
```