**Artificial Intelligence**

**LAB PROJECT**

**SUBMISSION**

**Submitted to:**

Ms. Chandani Kumari

**Submitted by:**

Akshita, Harshit Sharma,Vineet Jain

**Problem statement:**

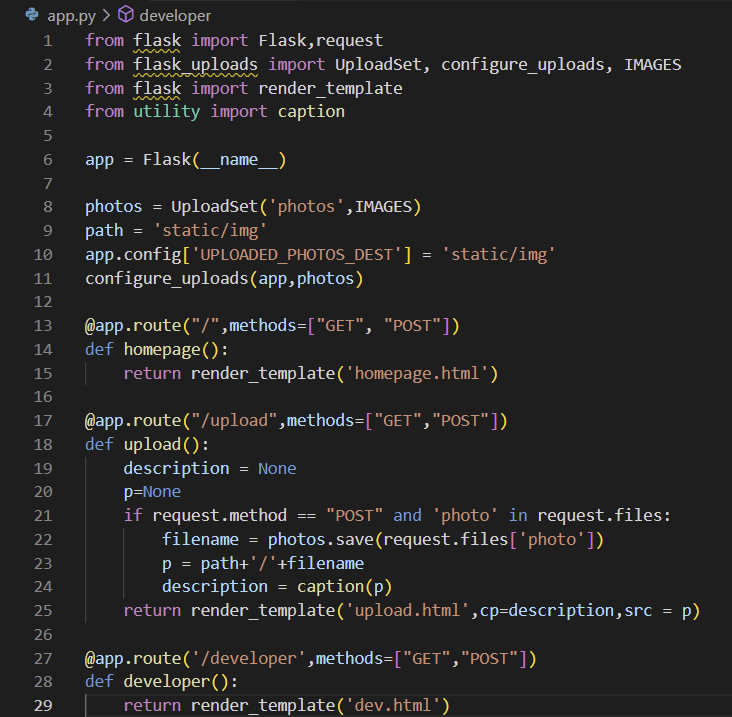
To create caption from Image.

**Description of problem:**

An image to caption generator is a type of machine learning model that is designed to automatically generate a textual description of an image. The system is trained on a large dataset of images and corresponding captions to learn patterns and relationships between the visual features of an image and the language used to describe it.When given a new image as input, the image to caption generator uses its learned knowledge to generate a natural language description of the image. This can be used for a variety of applications, such as automatically generating captions for social media posts or aiding visually impaired individuals in understanding the content of images.The performance of image to caption generators is typically evaluated based on the accuracy and relevance of the generated captions compared to human-written captions. Recent advancements in deep learning and natural language processing have led to significant improvements in the quality of image to caption generators, making them increasingly useful for a wide range of applications.

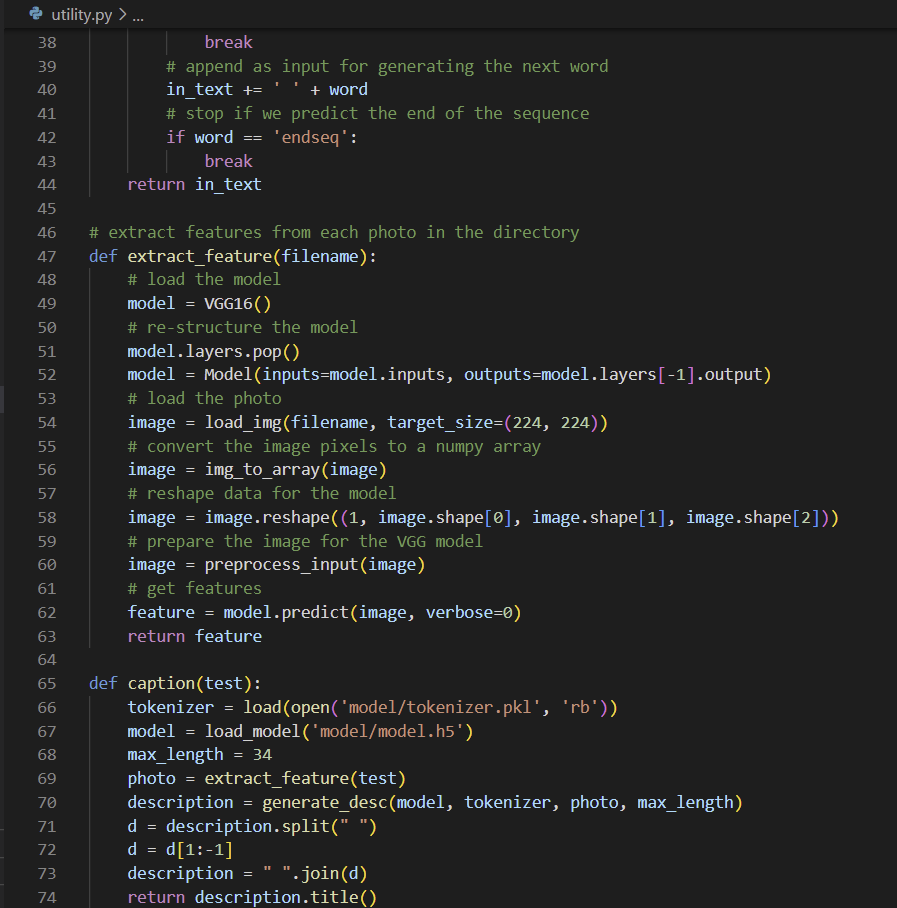
**Code and Explanation**

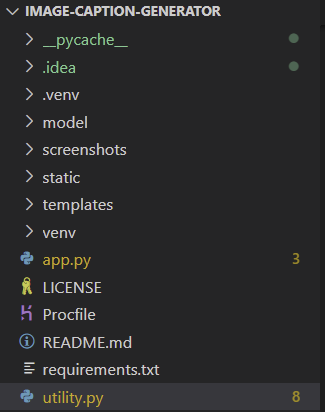
1. **app.py**



1. **utility.py**







IMPORTING USED MODULES

Brief description of the imported modules: -

1. **flask :-** Flask is a web application framework for Python. It is designed to be simple and lightweight, allowing developers to build web applications quickly and easily. Flask is also highly customizable, allowing developers to add their own functionality to the framework.
2. **flask\_upload :-** Flask-Uploads is an extension for Flask that provides flexible and easy-to-use upload handling capabilities. It allows developers to add file upload functionality to their Flask web applications with minimal code.
3. **numpy:-** NumPy is a Python module for scientific computing. It provides support for large, multi-dimensional arrays and matrices, as well as a large library of mathematical functions to operate on these arrays. NumPy is widely used in data analysis, scientific computing, and machine learning applications.
4. **keras :-** keras is a high-level neural network API written in Python, designed to enable fast experimentation with deep neural networks. It is built on top of lower-level deep learning frameworks such as TensorFlow and Theano, and provides a user-friendly interface for building, training, and deploying deep learning models.
5. **keras.preprocessing.sequence :-** The keras.preprocessing.sequence module provides utilities for working with sequence data, such as text and time-series data, in keras. It provides functions for converting text and sequences into numerical representations that can be fed into a neural network.

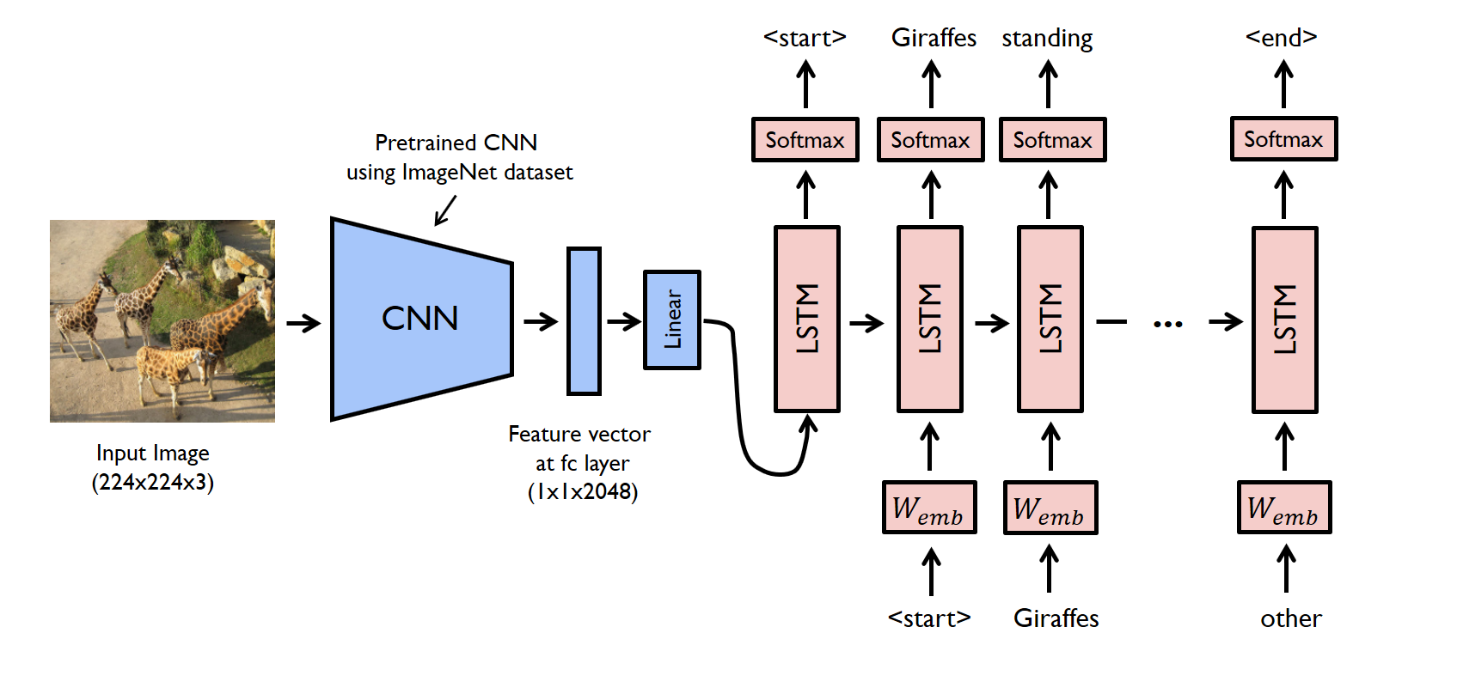
An image caption generator is a type of artificial intelligence (AI) system that uses computer vision and natural language processing (NLP) techniques to automatically generate a textual description or caption for an image. The goal of an image caption generator is to enable machines to understand the content of an image and describe it in a way that is similar to how humans would describe it.

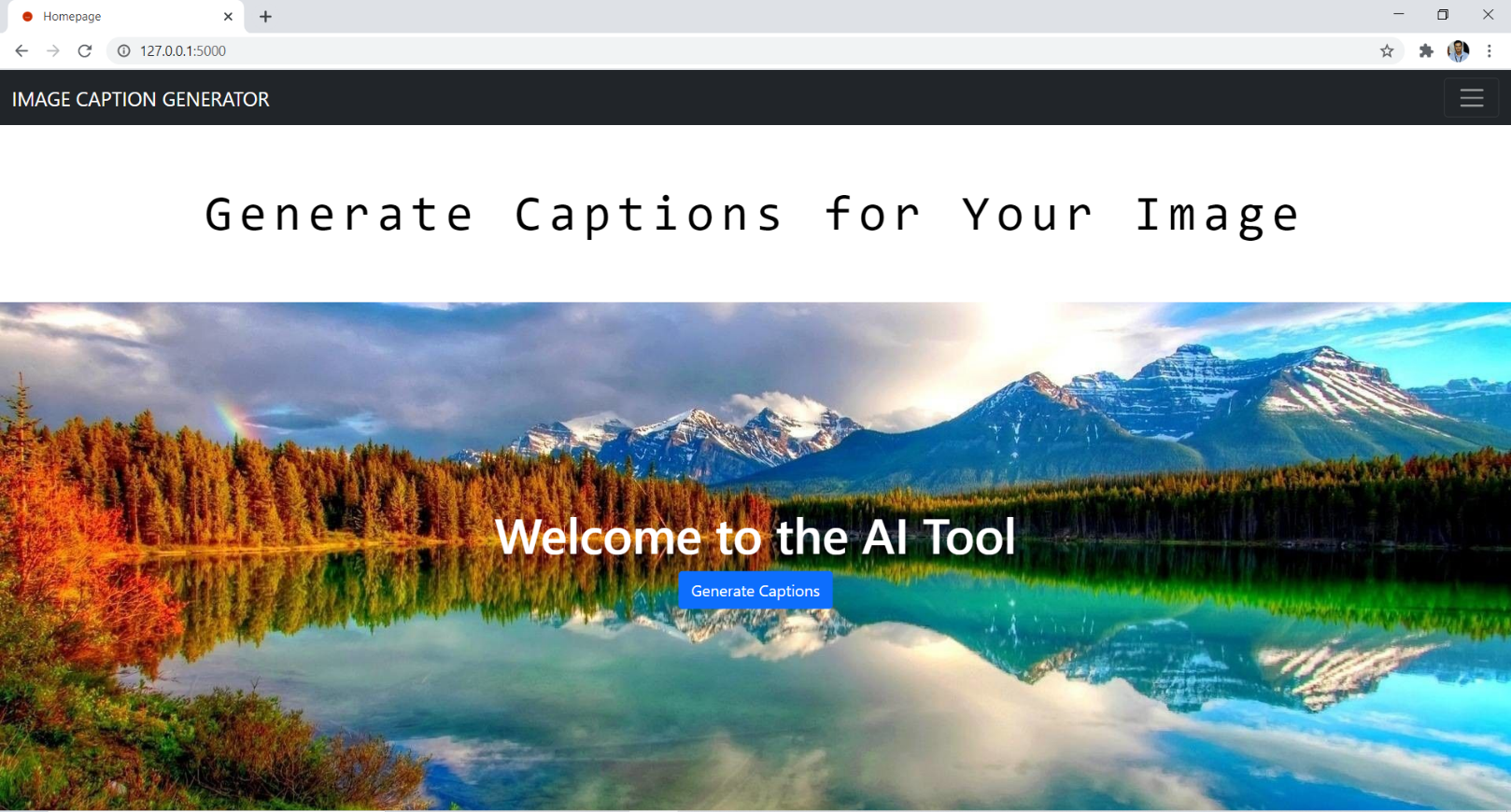
To achieve this, an image caption generator first needs to process the image and extract meaningful features from it, such as objects, colors, shapes, and textures. This is done using computer vision techniques, which involve analyzing the pixels in the image and identifying patterns and structures within it.

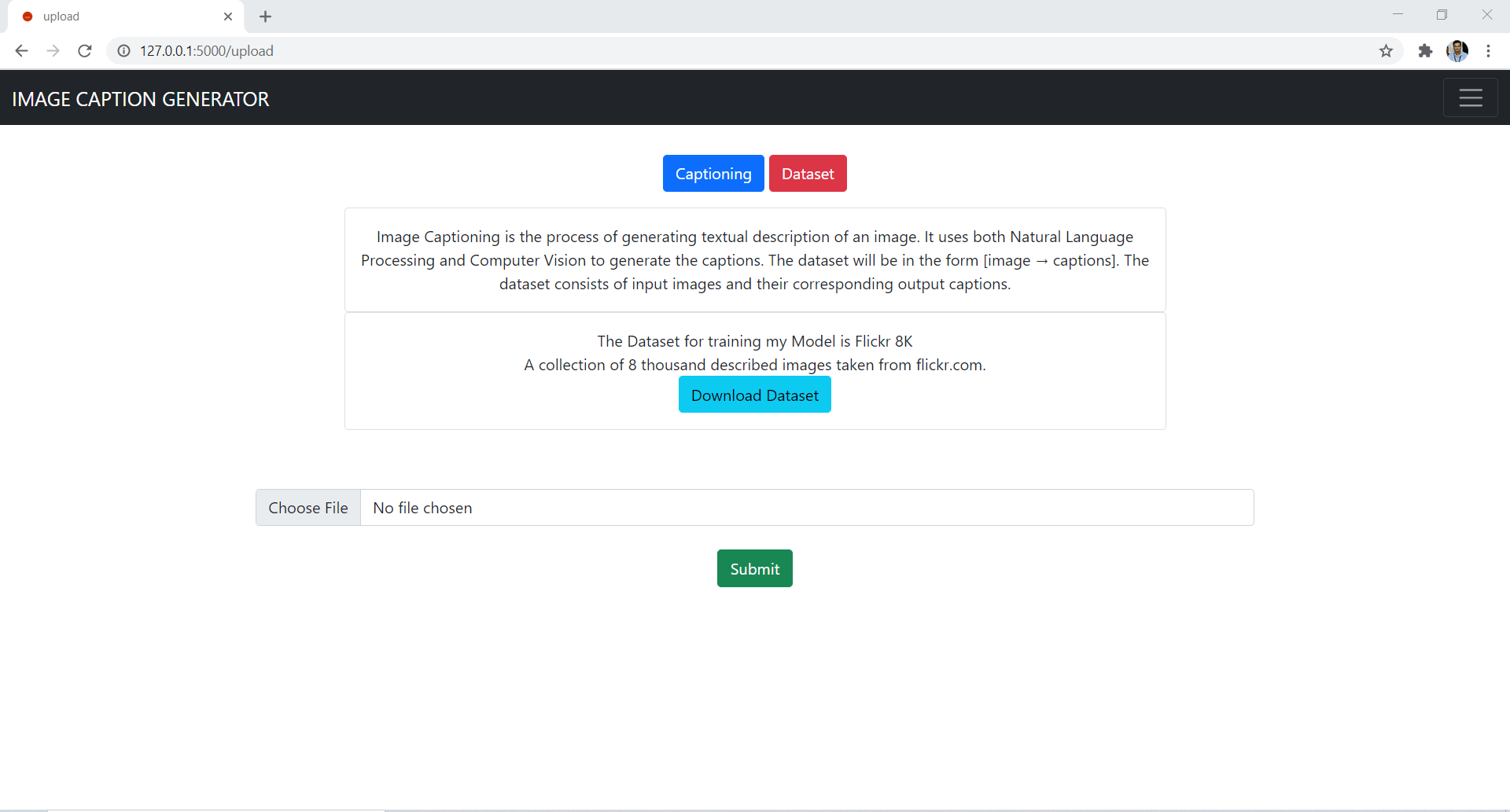
Once the image has been processed and its features have been extracted, an image caption generator uses a language model to generate a textual description of the image based on these features. The language model is trained on large amounts of textual data and is capable of generating human-like language.The output of an image caption generator is a textual description or caption that describes the content of the image in a way that is understandable to humans. For example, given an image of a dog playing with a frisbee in a park, an image caption generator might generate a caption like "A golden retriever running in a green field with a red frisbee in its mouth".Image caption generators have many practical applications, such as aiding visually impaired people, improving image search algorithms, and enhancing the user experience in social media platforms. They are also used in fields such as robotics, autonomous driving, and video surveillance to enable machines to better understand their environment.

**The function of an image caption generator can be broken down into the following steps:**

**1.** **Input:** The image caption generator takes an image as input.**2.** **Image Processing:** The image is processed using various computer vision techniques to extract meaningful features from the image, such as colors, shapes, objects, and textures.**3.** **Language Model**: A language model is then used to generate a textual description of the image based on the extracted features. The language model uses natural language processing (NLP) techniques to generate the text.**4.** **Output**: The final output of the image caption generator is a textual description or caption that describes the content of the image in a human-like language.Image caption generators can be used for a variety of applications, such as aiding visually impaired people, improving image search algorithms, and enhancing the user experience in social media platforms. They are also used in fields such as robotics, autonomous driving, and video surveillance to enable machines to better understand their environment.







**CNN :-**

CNN stands for Convolutional Neural Network, which is a type of artificial neural network used

primarily in image and video recognition tasks.A CNN consists of a series of interconnected layers, including convolutional layers, pooling

layers, and fully connected layers.Convolutional layers are the core building blocks of CNNs. They apply a set of filters to the input

image to extract important features, such as edges, corners, and patterns. Each filter is a small

matrix of weights that slides over the input image, performing a convolution operation that

produces a feature map.Pooling layers reduce the spatial dimensions of the feature maps generated by the convolutional

layers. They do this by down-sampling the feature maps, either by taking the maximum value

(max pooling) or the average value (average pooling) of a small region of the feature map.Finally, the fully connected layers take the output of the convolutional and pooling layers and use

it to classify the input image into one of several categories.Training a CNN involves adjusting the weights of the filters and layers to minimize the difference

between the predicted output and the true output. This is typically done using backpropagation, a technique that uses gradient descent to adjust the weights in a way that minimizes the error between

the predicted output and the true output.CNNs have proven to be very effective in image and video recognition tasks, and are used in a

wide range of applications, including self-driving cars, facial recognition, and medical image analysis.

**CNN in Image Caption Generator:-**

In an image to caption generator project using CNN, the CNN would be used as the image feature extractor.The CNN would take the input image and process it through a series of convolutional and pooling

layers to extract important features from the image. The output of the CNN would be a set of feature vectors that represent the important visual information in the image.These feature vectors would then be fed into a recurrent neural network (RNN) to generate the corresponding captions for the input images. The RNN would use a sequence-to-sequence model

to generate a sequence of words that form the caption for the input image.During the training process, the CNN and RNN would be trained together using a large dataset of

images and their corresponding captions. The weights of the CNN and RNN would be adjusted

using backpropagation and gradient descent to minimize the difference between the predicted

captions and the true captions.Once the model is trained, it can be used to generate captions for new images by feeding the image through the CNN to extract the feature vectors, and then feeding those feature vectors into the

RNN to generate the corresponding captions.This approach to image captioning using CNNs and RNNs has been shown to be very effective,

and is widely used in image captioning applications.