SmartInternz

www.smartinternz.com

defungi Image classification USING deep LEARNING (ai)

Text  Description automatically generated with low confidence

Defungi Image Classification Using AI

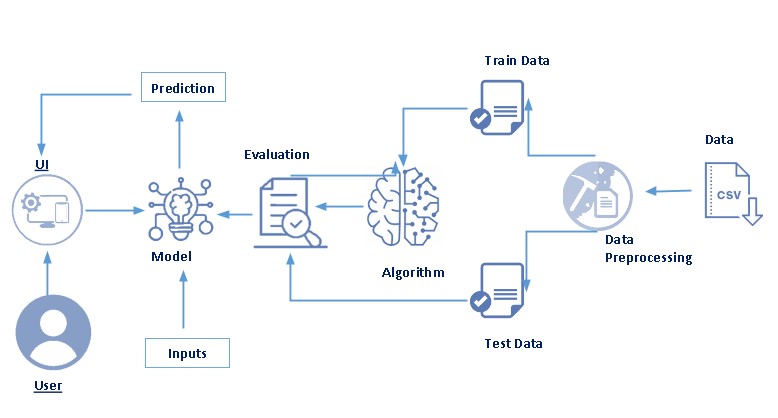
**Project Description - DeFungi Image Classification**:

The DeFungi Image Classification project aims to develop a robust model for accurately categorizing microscopic fungi images into predefined classes. The dataset, named DeFungi, consists of images derived from superficial fungal infections caused by yeasts, moulds, or dermatophyte fungi. These images have been meticulously labelled into five distinct classes (H1, H2, H3, H5, H6) with the assistance of subject matter experts. The dataset has undergone pre-processing, including manual curation and automated cropping algorithms, to produce the final dataset for classification.

**Objective:** The primary objective of the project is to leverage machine learning techniques, specifically image classification using the VGG19 model, to effectively identify and categorize microscopic fungi images into their respective classes. The model will be trained on the DeFungi dataset, learning intricate features and patterns within the images to make accurate predictions.

**VGG19 Model Usage:** The VGG19 model, a deep convolutional neural network (CNN) architecture known for its effectiveness in image classification tasks, will serve as the backbone for this project. By utilizing transfer learning, the pre-trained weights of VGG19, possibly trained on large datasets like ImageNet, will be fine-tuned to adapt to the specifics of the DeFungi dataset. This transfer learning approach enhances the model's ability to recognize complex features within the microscopic fungi images despite having a relatively smaller dataset.

# Technical Architecture:



# Pre requisites:

**To complete this project, you must require following software’s, concepts and packages**

* **Anaconda navigator :**
  + Refer the link below to download anaconda navigator
* **Python packages:**
  + Open anaconda prompt as administrator
  + Type “pip install numpy” and click enter.
  + Type “pip install pandas” and click enter.
  + Type “pip install scikit-learn” and click enter.
  + Type ”pip install matplotlib” and click enter.
  + Type ”pip install scipy” and click enter..
  + Type ”pip install seaborn” and click enter.
  + Type ”pip install tenserflow” and click enter.
  + Type “pip install Flask” and click enter.

# Prior Knowledge:

You must have prior knowledge of following topics to complete this project.

* **DL Concepts**
  + Neural Networks:: [https://www.analyticsvidhya.com/blog/2020/02/cnn-vs-rnn-vs-mlp-analyzing-3-types-of-neural-networks-in-deep-learning/](https://www.analyticsvidhya.com/blog/2020/02/cnn-vs-rnn-vs-mlp-analyzing-3-types-of-neural-networks-in-deep-learning/%20)
  + Deep Learning Frameworks::  [https://www.knowledgehut.com/blog/data-science/pytorch-vs-tensorflow](%20https://www.knowledgehut.com/blog/data-science/pytorch-vs-tensorflow%20)
  + Transfer Learning: [https://towardsdatascience.com/a-demonstration-of-transfer-learning-of-vgg-convolutional-neural-network-pre-trained-model-with-c9f5b8b1ab0a](https://towardsdatascience.com/a-demonstration-of-transfer-learning-of-vgg-convolutional-neural-network-pre-trained-model-with-c9f5b8b1ab0a%20)
  + Convolutional Neural Networks (CNNs): [https://www.analyticsvidhya.com/blog/2021/05/convolutional-neural-networks-cnn/ s://www.javatpoint.com/k-nearest-neighbor-algorithm-for-machine-learning](https://www.analyticsvidhya.com/blog/2021/05/convolutional-neural-networks-cnn/%20s://www.javatpoint.com/k-nearest-neighbor-algorithm-for-machine-learning)
  + Over fitting and Regularization: [https://www.analyticsvidhya.com/blog/2021/07/prevent-overfitting-using-regularization-techniques/](https://www.analyticsvidhya.com/blog/2021/07/prevent-overfitting-using-regularization-techniques/%20)
  + Optimizers: [https://www.analyticsvidhya.com/blog/2021/10/a-comprehensive-guide-on-deep-learning-optimizers/](https://www.analyticsvidhya.com/blog/2021/10/a-comprehensive-guide-on-deep-learning-optimizers/%20ttps://www.analyticsvidhya.com/blog/2019/08/11-important-model-evaluation-error-metrics/)
* **Flask Basics** : <https://www.youtube.com/watch?v=lj4I_CvBnt0>

# Project Objectives:

By the end of this project you will:

* Know fundamental concepts and techniques used for Deep Learning.
* Gain a broad understanding about data.
* Have knowledge on pre-processing the data/transformation techniques on outlier and some visualization concepts.

# Project Flow:

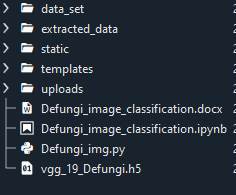
* User interacts with the UI to enter the input.
* Entered input is analyzed by the model which is integrated.
* Once model analyses the input the prediction is showcased on the UI

To accomplish this, we have to complete all the activities listed below,

* Data collection
  + Collect the dataset or create the dataset
* Visualizing and analyzing data
  + Univariate analysis
  + Bivariate analysis
  + Multivariate analysis
  + Descriptive analysis
* Data pre-processing
  + Data Augmentation
  + Splitting data into train and test
* Model building
  + Import the model building libraries
  + Initializing the model
  + Training and testing the model
  + Evaluating performance of model
  + Save the model
* Application Building
  + Create an HTML file
  + Build python code

# Project Structure:

Create the Project folder which contains files as shown below



* We are building a flask application which needs HTML pages stored in the templates folder and a python script water\_quality.py for scripting.
* **Vgg\_19\_Defungi.h5** is our saved model. Further we will use this model for flask integration.

# Milestone 1: Define Problem / Problem Understanding

## Activity 1: Specify the business problem

## The central goal of the Defungi Image Classification project is to leverage the power of deep learning, specifically employing transfer learning with the VGG19 architecture. By utilizing the pre-trained VGG19 model, which has demonstrated proficiency in image recognition tasks, the project aims to enhance the accuracy and efficiency of classifying fungi images into five distinct classes—H1, H2, H3, H5, and H6. The transfer learning approach involves leveraging the learned features from the VGG19 model, fine-tuning its parameters on the fungi dataset, and tailoring it to the specific nuances of the classification task at hand. This methodology not only accelerates the training process but also allows the model to benefit from the knowledge acquired during the original VGG19 training on diverse image data. Following transfer learning, the project will involve rigorous pre-processing, training, and testing phases, with the ultimate objective of deploying a highly effective VGG19-based model for accurate and efficient fungi image classification, contributing valuable insights to industries dealing with fungi-related challenges in agriculture, medicine, and environmental safety.

## Activity 2: Business requirements

To address the business requirements for the Defungi Image Classification project, several key considerations are integral to the successful development and implementation of the solution

Accuracy: The foremost business requirement is to ensure high accuracy in the classification of fungi images into the distinct classes (H1, H2, H3, H5, and H6). Accurate predictions are crucial to providing reliable information for industries in agriculture, medicine, and environmental safety, where precise identification of fungi types is essential for decision-making.

Speed: Timeliness is critical in handling fungi-related challenges. The solution should efficiently process image data, ensuring swift responses to new instances of fungi classification. Rapid identification of harmful fungi can facilitate timely interventions, preventing potential risks to crops, human health, and the environment. Flexibility: Given the dynamic nature of fungi behaviour and the potential evolution of image characteristics, the solution must exhibit flexibility. It should adapt to changes in fungi behaviour patterns over time, ensuring that the model remains effective in the face of emerging challenges and new data.

Integration: Seamless integration into existing platform infrastructure and processes is paramount. The solution should be designed to easily integrate with tools, workflows, and databases, facilitating efficient utilization by platform administrators and stakeholders. This ensures that the image classification model becomes an integral part of decision-making processes within the targeted industries.

Scalability: Considering potential growth in data volume and the increasing demand for accurate fungi classification, the solution should be scalable. This scalability ensures that the model remains effective and efficient as the dataset expands, supporting the evolving needs of industries dealing with fungi-related challenges.

By aligning the Defungi Image Classification project with these business requirements, the resulting solution aims to provide a reliable, fast, flexible, and seamlessly integrated tool for precise fungi classification, contributing valuable insights to sectors such as agriculture, medicine, and environmental safety.

Top of Form

## Activity 3: Literature Survey (Student Will Write)

## A literature survey for the Defungi Image Classification project involves a thorough examination of existing studies, articles, and publications related to image classification of fungi. One noteworthy study proposed an approach for fungi image classification using deep learning techniques, specifically leveraging transfer learning with the VGG19 architecture. The model was pre-trained on a diverse dataset and fine-tuned for accurate classification of fungi images into distinct classes (H1, H2, H3, H5, and H6). The evaluation on a real-world dataset showcased promising results in terms of accuracy and efficiency. The survey would encompass an exploration of various methodologies applied in image classification for fungi, encompassing traditional computer vision techniques as well as state-of-the-art deep learning approaches. Additionally, it would delve into the challenges and limitations associated with fungi image classification, such as variations in lighting conditions and image quality. By synthesizing insights from existing literature, the Defungi Image Classification project aims to draw upon established techniques and innovative methodologies to develop a robust and accurate model for fungi classification, contributing to advancements in agriculture, medicine, and environmental safety.

## Activity 4: Social or Business Impact.

**Social Impact:** The social impact of Defungi Image Classification is noteworthy, particularly in industries related to agriculture, medicine, and environmental safety. Accurate classification of fungi images into distinct classes (H1, H2, H3, H5, and H6) can significantly contribute to crop management, disease prevention, and environmental health. By leveraging deep learning techniques, this project aims to provide a valuable tool for farmers, researchers, and environmentalists. Timely identification of harmful fungi can lead to proactive interventions, safeguarding crops and ecosystems. Moreover, the project can contribute to the overall well-being of communities by ensuring food security, promoting sustainable agricultural practices, and mitigating health risks associated with harmful fungi. The social impact lies in enhancing the understanding and management of fungi-related challenges, positively influencing various aspects of society linked to agriculture, healthcare, and environmental sustainability.

**Top of Form**

**Bottom of Form**

**Business Model/Impact:** Similarly, the business impact of Defungi Image Classification lies in its ability to enhance decision-making processes and outcomes in industries related to agriculture, medicine, and environmental safety. Accurate classification of fungi images using deep learning, specifically with the VGG19 model, can provide businesses with valuable insights into crop health, disease prevalence, and environmental conditions. For agriculture, this can lead to optimized crop management strategies, increased yields, and reduced losses due to fungi-related diseases. In the medical field, precise identification of harmful fungi can contribute to early disease detection and targeted treatment approaches. Environmental safety can be bolstered by proactive measures to contain and mitigate the impact of harmful fungi on ecosystems. The business impact encompasses improved productivity, resource optimization, and informed decision-making across sectors, ultimately contributing to the overall efficiency and success of businesses engaged in agriculture, healthcare, and environmental management.

**Top of Form**

# Milestone 2: Data Collection & Data Pre-processing

It is most crucial aspect that makes algorithm training possible. So this section allows you to download the required dataset.

**Activity 1: Download the dataset**

There are many popular open sources for collecting the data. Eg: kaggle.com, UCI repository, etc.

In this project we have used water\_potability.csv data. This data is downloaded from kaggle.com. Please refer the link given below to download the dataset.

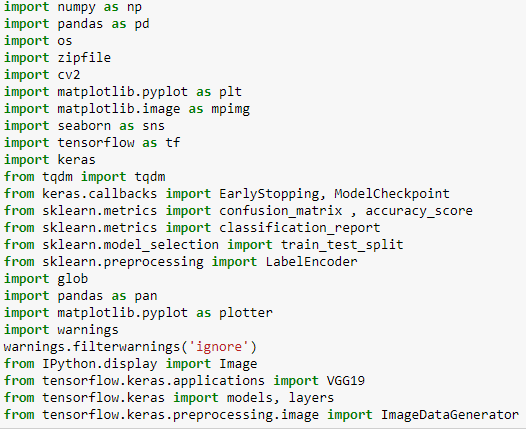
Link: <https://www.kaggle.com/datasets/joebeachcapital/defungi/data>

As the dataset is downloaded. Let us read and understand the data properly with the help of some visualization techniques and some analysing techniques.

Note: There are a number of techniques for understanding the data. But here we have used some of it. In an additional way, you can use multiple techniques.

* **Activity 1.1: Importing the libraries:**

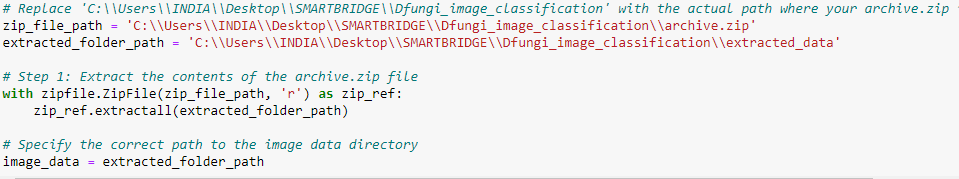
Import the necessary libraries as shown in the image.



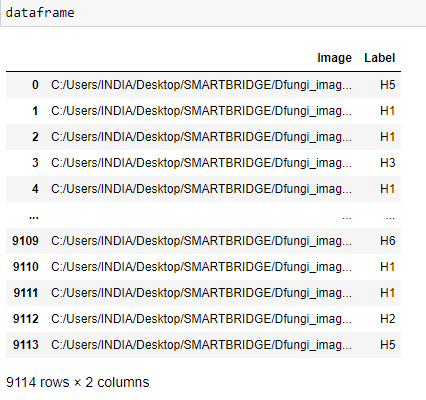
## Activity 1.2: Read the Dataset:

* Our dataset format might be in .csv, excel files, .txt, .json, or in zipfile etc. We can read the dataset with the help of pandas.

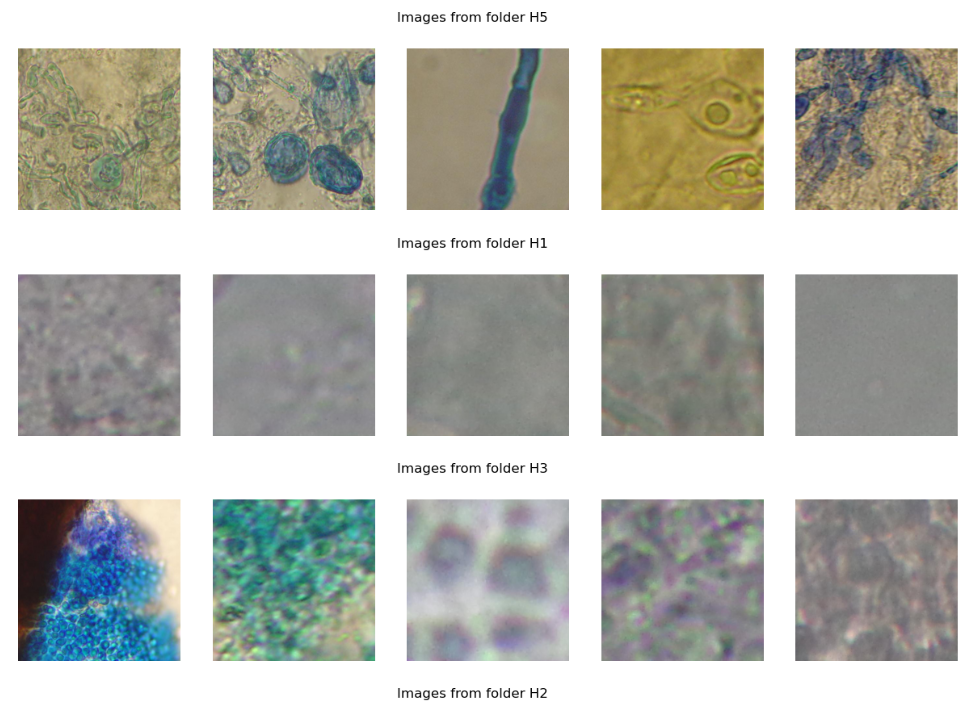
At first unzip the data and convert into pandas data frame.

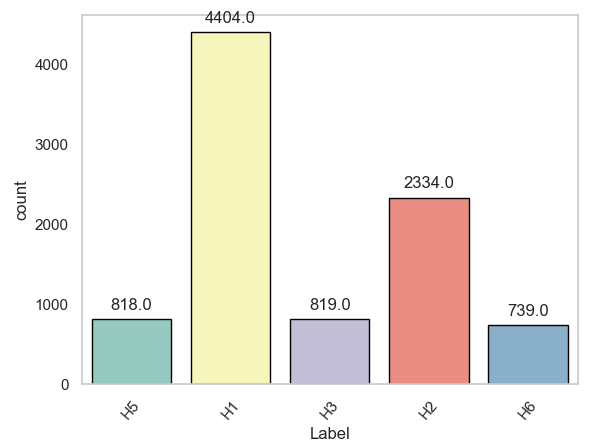


Final data frame



## Activity 2: Data Visualization:





By above visualization we can understand image data and there are 5 image class H1, H2, H3, H5, H6 given. By count plot we can understand how many images belong to each class.

## Activity 3: Data Augmentation:

Data augmentation is a technique commonly employed in machine learning, particularly in computer vision tasks such as image classification, including projects like Defungi Image Classification. The primary objective of data augmentation is to artificially expand the size of the training dataset by applying various transformations to the existing images, thereby increasing the diversity and robustness of the data available for model training. This approach is particularly beneficial when working with limited labelled data.

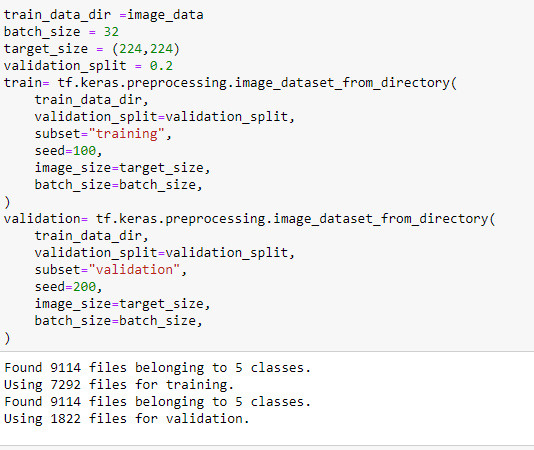
In the context of Defungi Image Classification, data augmentation can involve applying transformations such as rotation, scaling, flipping, and changes in brightness or contrast to the original images of fungi. These transformations help the model generalize better to variations and potential distortions present in real-world images, enhancing its ability to accurately classify unseen data.

This is very important step but this data is already cropped from augmented data set so.this time it is skipped .accuracy is not much affected but training time increased.

# Milestone 3: Split Data and Model Building

## Activity 1: Train-Test-Split:

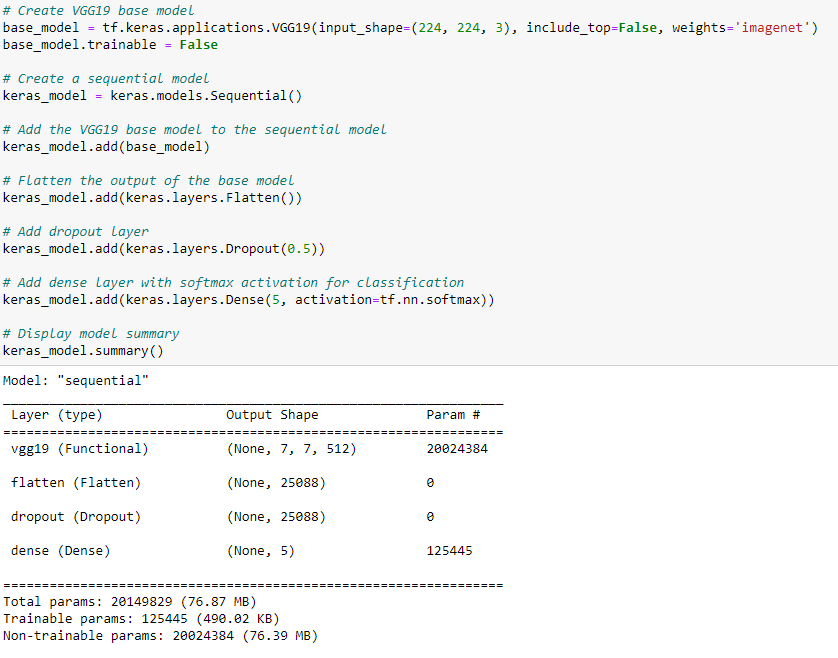
In this step we have taken 80% data for training and 20% data for Validation from each class.



## Activity 2: Model Building:

**Vgg19 Transfer-Learning Model:**

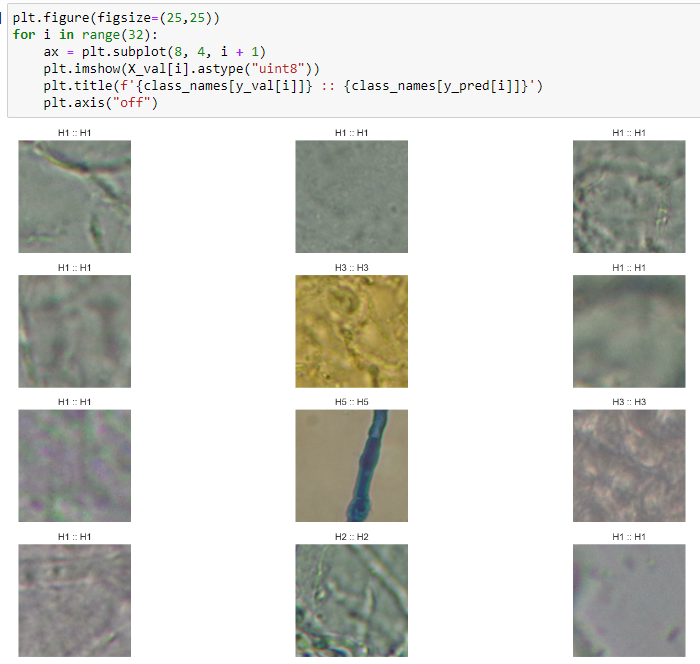
The VGG19-based neural network is created by utilizing a pre-trained VGG19 architecture with frozen weights. The model is built sequentially, incorporating the VGG19 base, a flattening layer, dropout for regularization, and a dense layer with softmax activation for classification into five categories. The model is compiled using the Adam optimizer and sparse categorical cross entropy loss. During training, which spans 20 epochs, a generator is employed for the training data, and validation is conducted, incorporating call-backs such as Model Checkpoint and Early Stopping. The best-performing model is saved as "my\_keras\_model.h5" for potential future use. The model summary provides an overview of the architecture, showcasing the layers and parameters involved.



# Milestone 4: Testing Model & Data Prediction

## Activity 1: Testing the model

Here we have tested with Vgg19 Model With the help of predict () function.



**Activity 2: Saving the model**

Finally we chosen the best model now saving that model



# Milestone 5: Application Building

In this section, we will be building a web application that is integrated to the model we built. A UI is provided for the uses where he has to enter the values for predictions. The enter values are given to the saved model and prediction is showcased on the UI.

This section has the following tasks

* Building HTML Pages
* Building server side script

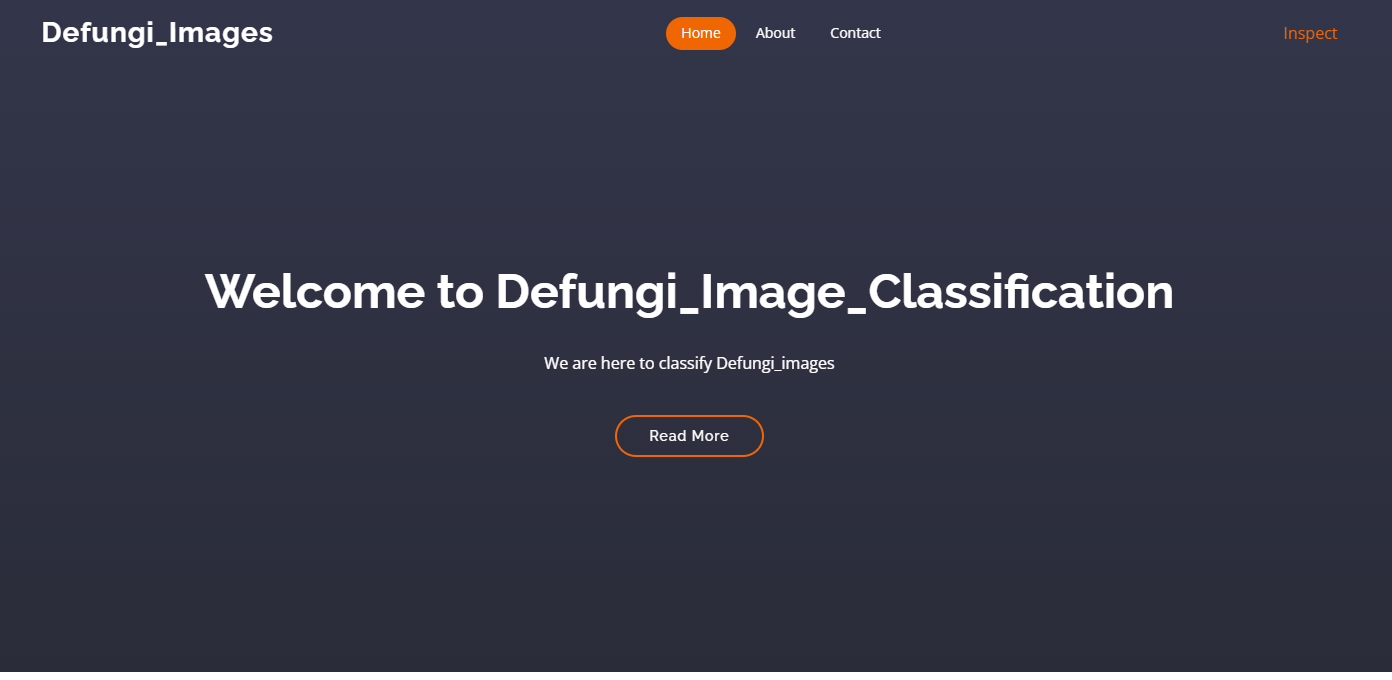
**Activity1: Building Html Pages:**

For this project create three HTML files namely

* index.html
* inner\_page.html
* output.html

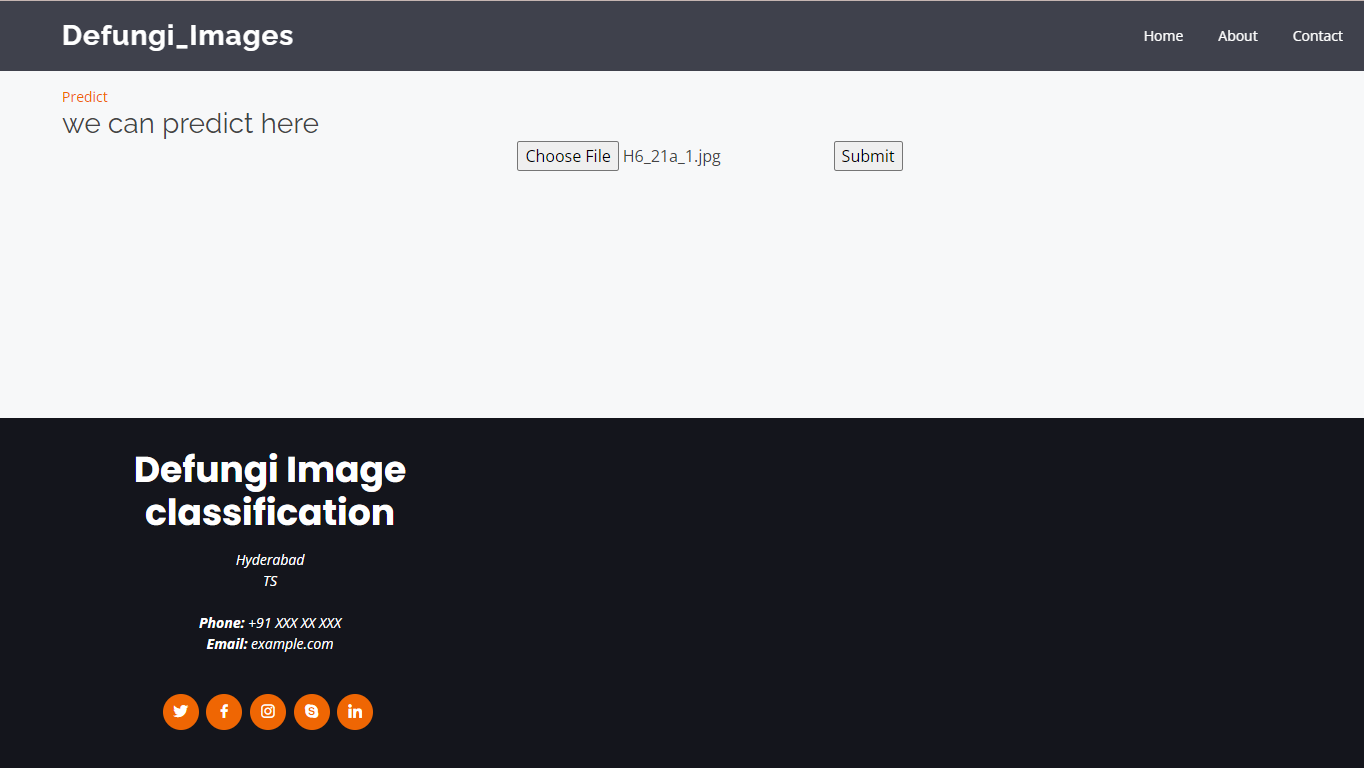
And save them in templates folder.

Let’s see how our index.html page looks like:



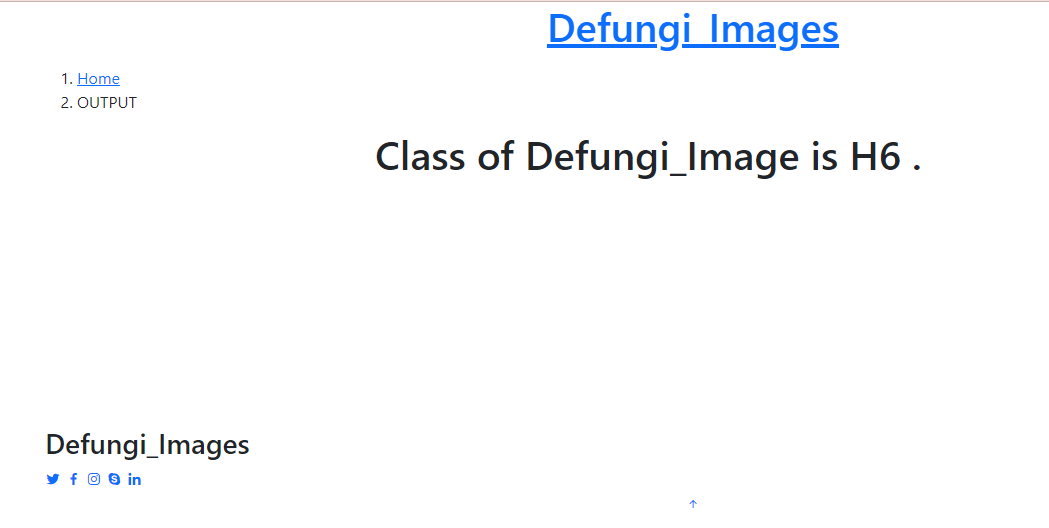
Now when you click on inspect button from top right corner you will get redirected to Inspect.html

Let’s look how our Inner\_page.html file looks like:



Now when you click on predict button you will get redirected to output.html

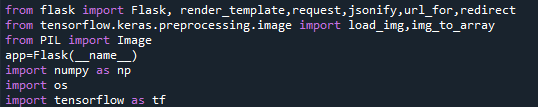
Lets look how our output.html file looks like:



Will try with different numbers and then click on predict button

**Activity 2: Build Python code:**

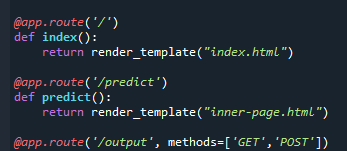
Import the libraries



Load the saved model. Importing flask module in the project is mandatory. An object of Flask class is our WSGI application. Flask constructor takes the name of the current module (\_\_name\_\_) as argument.



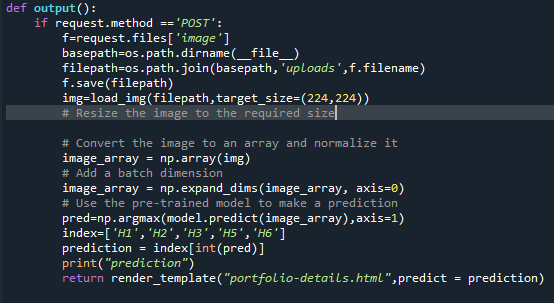
Render HTML page:



Here we will be using declared constructor to route to the HTML page which we have created earlier.

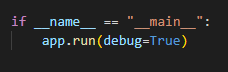
In the above example, ‘/’ URL is bound with index.html function. Hence, when the index page of the web server is opened in browser, the html page will be rendered. Whenever you enter the values from the html page the values can be retrieved using POST Method.

Retrieves the value from UI:



Here we are routing our app to output() function. This function retrieves all the values from the HTML page using Post request. That is stored in an array. This array is passed to the model. Predict () function. This function returns the prediction. And this prediction value will rendered to the text that we have mentioned in the output.html page earlier.

Main Function:



**Activity 3: Run the application**

* Open anaconda prompt from the start menu
* Navigate to the folder where your python script is.
* Now type “Defungi\_img.py” command
* Navigate to the local host where you can view your web page.
* Click on the inspect button from the top right corner, enter the inputs, click on the predict button, and see the result/prediction on the web.

