MUSHROOM DISEASE PREDICTION

Mini Project Report

Submitted by

ZANHA JAMAL

Submitted in partial fulfillment of the requirements for the award of the degree of

Master of Computer Applications
Of
A P J Abdul Kalam Technological University

FEDERAL INSTITUTE OF SCIENCE AND TECHNOLOGY (FISAT)®
ANGAMALY-683577, ERNAKULAM(DIST)
FEBRUARY 2022

DECLARATION

I, **Zanha Jamal**, hereby declare that the report of this project work, submitted to the Department of Computer Applications, Federal Institute of Science and Technology (**FISAT**), Angamaly in partial fulfillment of the award of the degree of Master of Computer Application is an authentic record of my original work.

The report has not been submitted for the award of any degree of this university or any other university.

Date : 04-03-2022

Place: Angamaly

FEDERAL INSTITUTE OF SCIENCE AND TECHNOLOGY (FISAT)®

ANGAMALY, ERNAKULAM-683577

DEPARTMENT OF COMPUTER APPLICATIONS

CERTIFICATE

This is to certify that the project report titled **quot;MUSHROOM DISEASE PREDICTIONquot;** submitted by **Zanha Jamal** towards partial fulfillment of the requirements for the award of the degree of Master of Computer Applications is a record of bonafide work carried out by them during the year 2021.

Project Guide **Head of the Department**

Submitted for the viva-voice held on . . . . at . . . .

Examiner1 :

Examiner2 :

ACKNOWLEDGEMENT

I am extremely glad to present my mini project which i did as a part of our curriculum. I take this opportunity to express my sincere thanks to those who helped me in bringing out the report of my project.

We are deeply grateful to Dr. George Manoj George, Principal, FISAT, Angamaly and Dr. C. Sheela, Vice Principal, FISAT, Angamaly. Our sincere thanks to Dr.Deepa Mary Mathews, Head of the department of MCA, FISAT, who had been a source of inspiration. During the period of our project work, we have received generous help from Mrs. Sona Mary Louis, my project guide, which i like to put on record here with deep gratitude and great pleasure.

I express my heartfelt thanks to all the faculty members in our department for their constant encouragement and never ending support throughout the project.

Finally i am grateful to all our friends who gave us a lot of sugges- tions for the successful completion of this project.

ABSTRACT

Mushrooms have high benefits in the human body. However, not all mushrooms are edible. While some have medical properties to cure cancer, some other types of mushrooms may contain viruses that carry infectious diseases. This paper proposed the principles of data analysis in order to present the prototype of mushroom classification based on physical characteristics. We created a model of mushroom classification by using Machine Learning (ML) with the mushroom dataset, comprising a total of 800 samples from the physical data of 22 attributes and it divide into two class as a toxic and non-toxic. The investigators designed the experiment in which200 samples were randomly assigned to the mushroom population, consisting of 200 equally toxic and non- toxic mushrooms. For the quality, many ML were comparison such as Naïve Bayes Updateable, Naïve Bayes, SGD Text, LWLand KNearest Neighbor (k-NN). The results showed that K-NNgave the highest classification accuracy rate of 100

In this model we also predict the disease that is effected to the mushroom and also give a detailed description on the specefic disease and suggest the remedies.we are training the system with mainly 4 diseases Inky Cap,Dry Bubbles,Cobweb,Pink Mold.

Contents

Chapter 1

INTRODUCTION

Mushroom can be found extensively in a verity of natural environments and visual identification of mushroom species is well established.Some mushrooms are known because of their nutritional and therapeutical properties.Some species are known all over the world because of their toxicity that causes fatal accidents every year mainly due to misidentification

This paper focus to create a model for the classification of the mushroom which will be divided into toxic mushrooms and non toxic to the body that will be examined from the charecteristic of each mushroom species.that's or what kind of elements are and analyzed that is a mushroom that can be eaten or is it a toxic mushroom.

In this model we also predict the disease that is effected to the mushroom and also give a detailed description on the specefic disease and suggest the remedies.we are training the system with mainly 4 diseases Inky Cap,Dry Bubbles,Cobweb,Pink Mold.

Chapter 2

PROOF OF CONCEPT

Objectives

The main objective is to classify edible and non edible mushrooms and to predict the disease effected to the mushroom. It is needed to build a model that can differentiate between “toxic” mushrooms and “non toxic” mushrooms. we propose a solution to classify mushrooms and predict the diseasem using the machine learning ensemble approach,By using those properties, we train a combination of different machine learning algorithms using various machine learning methods that are not thoroughly explored in the current literature.

Here we Focused on a particular area for classifying. classifying based on thier properties such as cap lenghth,cap shape etc. and we predict the disease by collecting the data and then we will train the system with this features. it includes the comparison of various previous methodologies proposed using different datasets and with different characteristics and accomplishments. Time saving Accuarcy

Chapter 3

IMPLEMENTATION

In this project one of the main step is importing the data set.the dataset is took from Kaggle.com.classificationdataset consist of a total of 800 samples from the physical data of 22 attributes and it divide into two class as a toxic and non-toxic. The investigators designed the experiment in which200 samples were randomly assigned to the mushroom population, consisting of 200 equally toxic and non-toxic mushrooms.to predict the disease the dataset consit of 4 images of the disease cobweb,inky cap,pink mold,dry bubble After that the processes such as customization of model and training the model with the data are done.Once a model is trained I can get know the training and validation accuracy of the model and also calculate the training and validation loss etc.After building the models , I compare the models and choose the model from that which have the highest accuracy. That model will be fine tuned and saved. Design and develop the web user interface using the web framework Flask. The saved model will be integrated with the User Interface.

TOOLS OR PROGRAMMING LANGUAGE

- FRONT END :

  - Html
  - CSS
  - Bootstrap
- BACK END :
  - Python

MODULES

1. Data Preprocessing :

Data preprocessing is an iterative process for the transformation of the raw data into understandable and useable forms. Raw datasets are usually characterized by incompleteness, inconsistencies, lacking in behavior, and trends while containing errors. The preprocessing is essential to handle the missing values and address inconsistencies

2. Feature Extraction :

Feature extraction involves reducing the number of resources required to describe a large set of data. When performing analysis of complex data one of the major problems stems from the number of variables involved. Feature extraction is a general term for methods of constructing combinations of the variables to get around these problems while still describing the data with sufficient accuracy.

3. Training the Model :

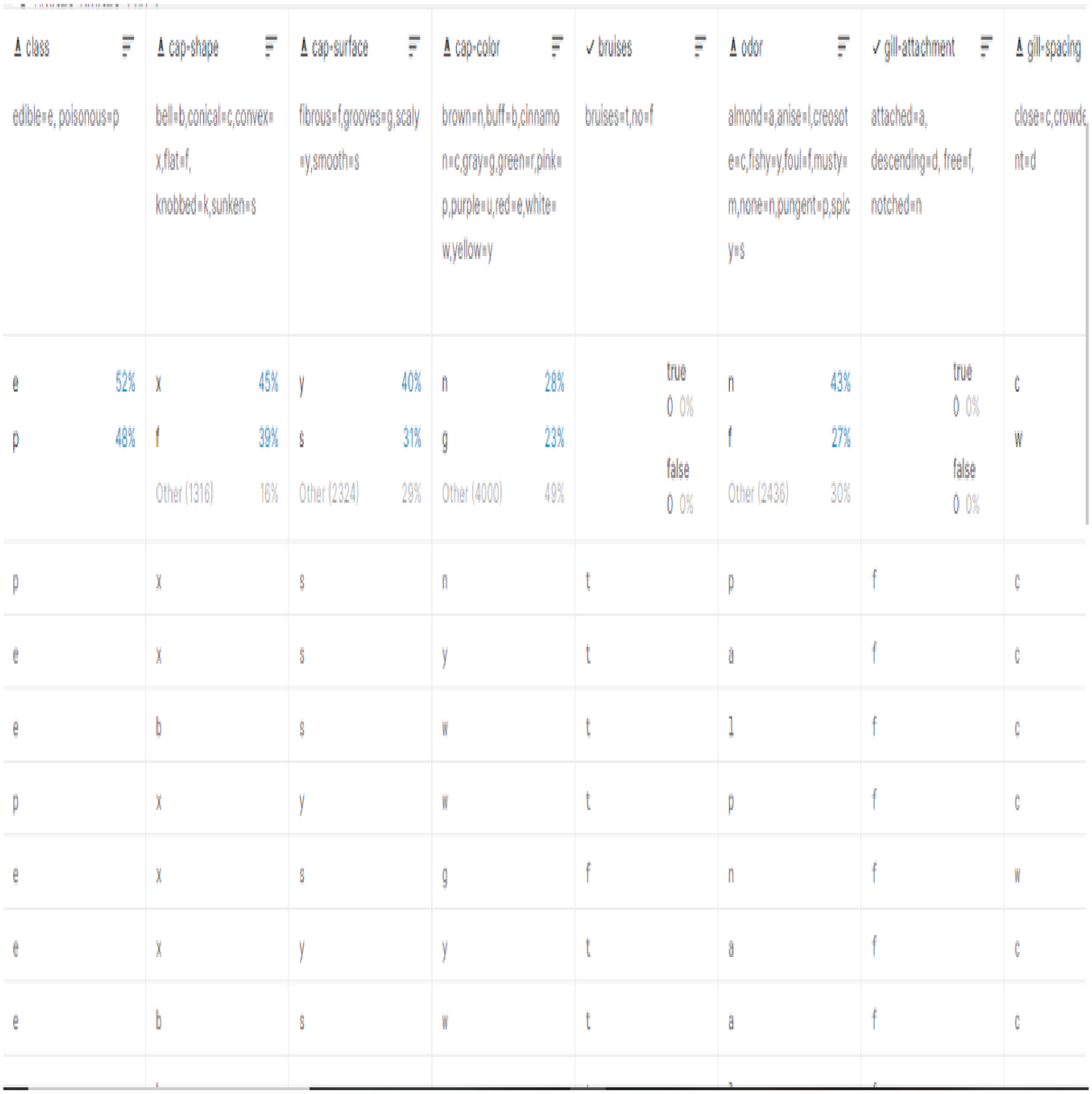
A training model is a dataset that is used to train an ML algorithm. It consists of the sample output data and the corresponding sets of input data that have an influence on the output. The training model is used to run the input data through the algorithm to correlate the processed output against the sample output. The result from this correlation is used to modify the model

4. Evaluation :

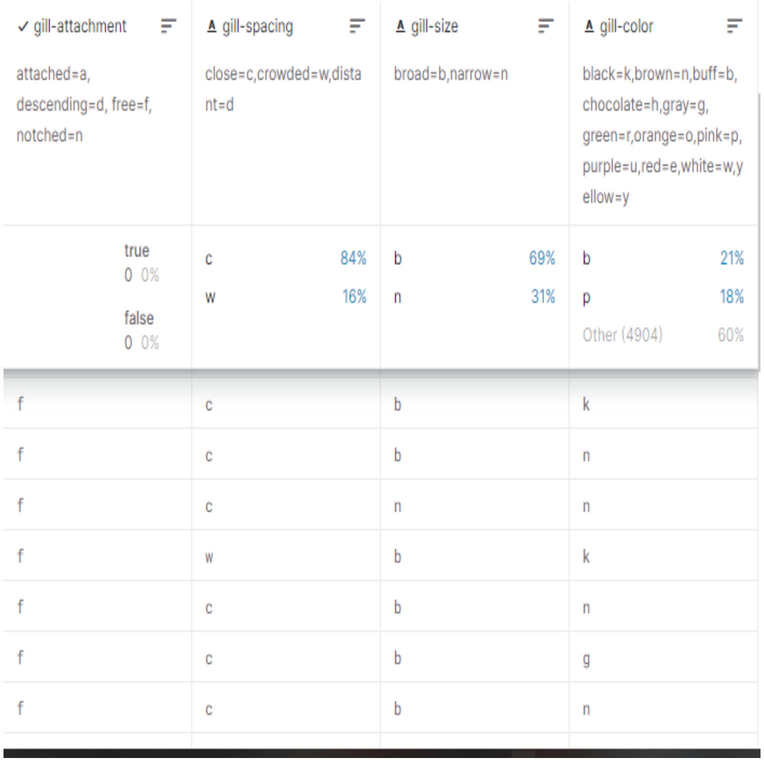
Model evaluation techniques in machine learning are helping us to find a better model among all other models in machine learning. It is simply the selection of machine learning models or measuring the performance of machine learning models.

DATASET

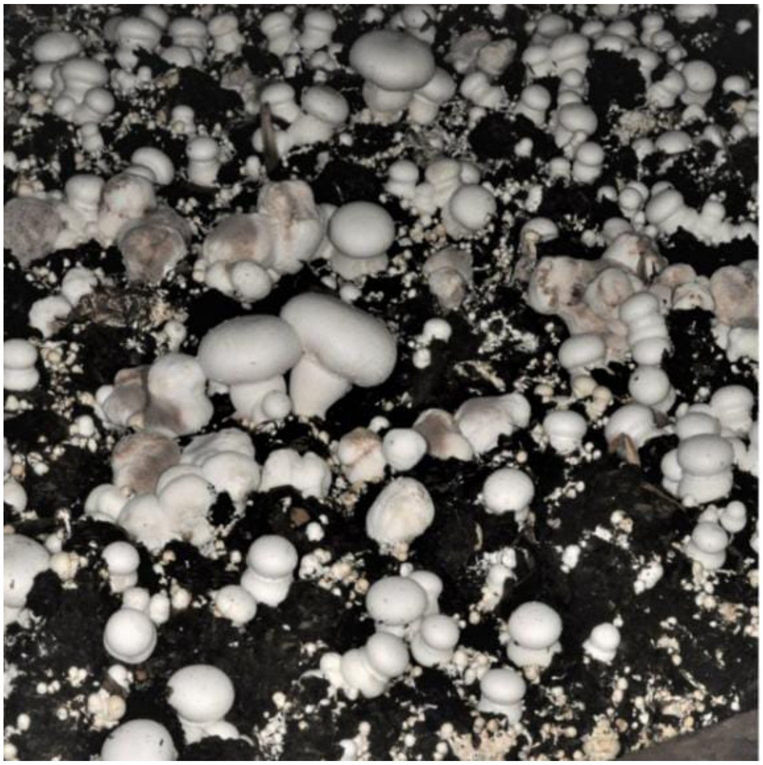
The data set from Kaggle for building the model.dataset consist of a total of 800 samples from the physical data of 22 attributes and it divide into two class as a toxic and non-toxic. The investigators designed the experiment in which200 samples were randomly assigned to the mushroom population, consisting of 200 equally toxic and non- toxic mushrooms.to predict the disease the dataset consit of 4 images of the disease cobweb,inky cap,pink mold,dry bubble



class	cap-shape	cap-surface	cap-color	bruises	odor	gill-attachment	gill-spacing
edible=e, poisonous=p	bell=b,conical=c,convex=x,flat=f, knobbed=k,sunken=s	fibrous=f,grooves=g,scaly=y,smooth=s	brown=n,buff=b,cinnamon=c,gray=g,green=r,pink=p,purple=u,red=e,white=w,yellow=y	bruises=t,no=f	almond=a,anise=l,creosote=c,fishy=y,foul=f,musty=m,none=n,pungent=p,spicy=s	attached=a, descending=d, free=f, notched=n	close=c,crowded=nt=d
e 52% p 48%	x 45% f 39% Other (1316) 16%	y 40% s 31% Other (2324) 29%	n 28% g 23% Other (4000) 49%	true 0 0% false 0 0%	n 43% f 27% Other (2436) 30%	true 0 0% false 0 0%	c w
p	x	s	n	t	p	f	c
e	x	s	y	t	a	f	c
e	b	s	w	t	l	f	c
p	x	y	w	t	p	f	c
e	x	s	g	f	n	f	w
e	x	y	y	t	a	f	c
e	b	s	w	t	a	f	c



✓ gill-attachment	A gill-spacing	A gill-size	A gill-color
attached=a, descending=d, free=f, notched=n	close=c,crowded=w,distant=d	broad=b,narrow=n	black=k,brown=n,buff=b, chocolate=h,gray=g, green=r,orange=o,pink=p, purple=u,red=e,white=w,yellow=y
true 0 0% false 0 0%	c 84% w 16%	b 69% n 31%	b 21% p 18% Other (4904) 60%
f	c	b	k
f	c	b	n
f	c	n	n
f	w	b	k
f	c	b	n
f	c	b	g
f	c	b	n



- **ALGORITHM TO BE USED**

- **1. KNearest Neighbour:**

K-Nearest Neighbour is one of the simplest Machine Learning algorithms based on Supervised Learning technique.K-NN algorithm assumes the similarity between the new case/-data and available cases and put the new case into the category that is most similar to the available categories. K-NN algorithm stores all the available data and classifies a new data point based on the similarity. This means when new data appears then it can be easily classified into a well suite category by using K- NN algorithm. K-NN algorithm can be used for Regression as well as for Classification but mostly it is used for the Classification problems. K-NN is a non-parametric algorithm, which means it does not make any assumption on underlying data. It is also called a lazy learner algorithm because it does not learn from the training set immediately instead it stores the dataset and at the time of classification, it performs an action on the dataset. KNN algorithm at the training phase just stores the dataset and when it gets new data, then it classifies that data into a category that is much similar to the new data

- **2. Convolutional Neural Network:**

A Convolutional Neural Network (ConvNet/CNN) is a Deep Learning algorithm which can take in an input image, assign importance (learnable weights and biases) to various aspects/objects in the image and be able to differentiate one from the other. The pre-processing required in a ConvNet is much lower as compared to other classification algorithms. While in primitive methods filters are hand-engineered, with enough training, ConvNets have the ability to learn these filters/characteristics.

The architecture of a ConvNet is analogous to that of the connectivity pattern of Neurons in the Human Brain and was inspired by the organization of the Visual Cortex. Individual neurons respond
to stimuli only in a restricted region of the visual field known as the Receptive Field. A collection of such fields overlap to cover the entire visual area.

Pooling layer is responsible for reducing the spatial size of the Convolved Feature. This is to decrease the computational power required to process the data through dimensionality reduction. Furthermore, it is useful for extracting dominant features which are rotational and positional invariant, thus maintaining the process of effectively training of the model.

There are two types of Pooling: Max Pooling and Average Pooling. Max Pooling returns the maximum value from the portion of the image covered by the Kernel. On the other hand, Average Pooling returns the average of all the values from the portion of the image covered by the Kernel.

Max Pooling also performs as a Noise Suppressant. It discards the noisy activations altogether and also performs denoising along with dimensionality reduction. On the other hand, Average Pooling simply performs dimensionality reduction as a noise suppressing mechanism. Hence, we can say that Max Pooling performs a lot better than Average Pooling.

The Convolutional Layer and the Pooling Layer, together form the i-th layer of a Convolutional Neural Network. Depending on the complexities in the images, the number of such layers may be increased for capturing low-levels details even further, but at the cost of more computational power.

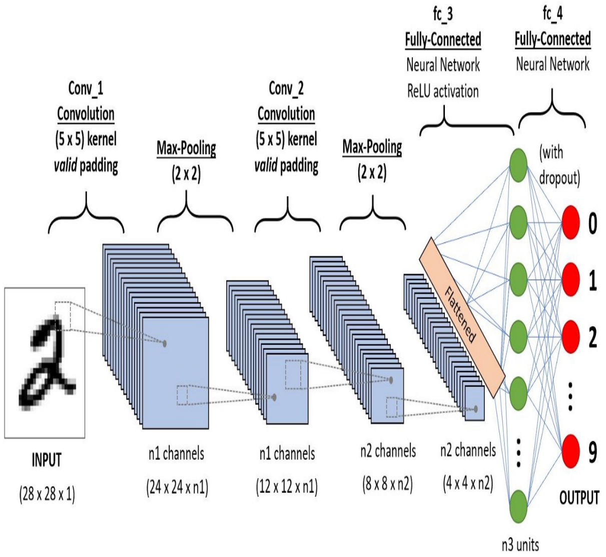


Figure 3.1: A CNN sequence to classify handwritten digits.

Chapter 4

RESULT ANALYSIS

Accuracy is often the most used metric representing the percentage of correctly predicted observations, either true or false. To calculate the accuracy of a model performance, the following equation can be used: In most cases, high accuracy value represents a good model, but considering the fact that we are training a classification model in our case, an article that was predicted as true while it was actually false (false positive) can have negative consequences; similarly, if an article was predicted as false while it contained factual data, this can create trust issues.
Accuarcy = TP+TN/TP+TN+FP+FN

Confusion Matrix

A Confusion matrix is an N x N matrix used for evaluating the performance of a classification model, where N is the number of target classes. The matrix compares the actual target values with those predicted by the machine learning model. This gives us a holistic view of how well our classification model is performing and what kinds of errors it is making.

For a binary classification problem, we would have a 2 x 2 matrix as shown below with 4 values:

1. TP = True Positives

2. FP = False Positives

3. TN = True Negatives

4. FN = False Negatives

Chapter 5

CONCLUSION AND FUTURE SCOPE

- **Conclusion**

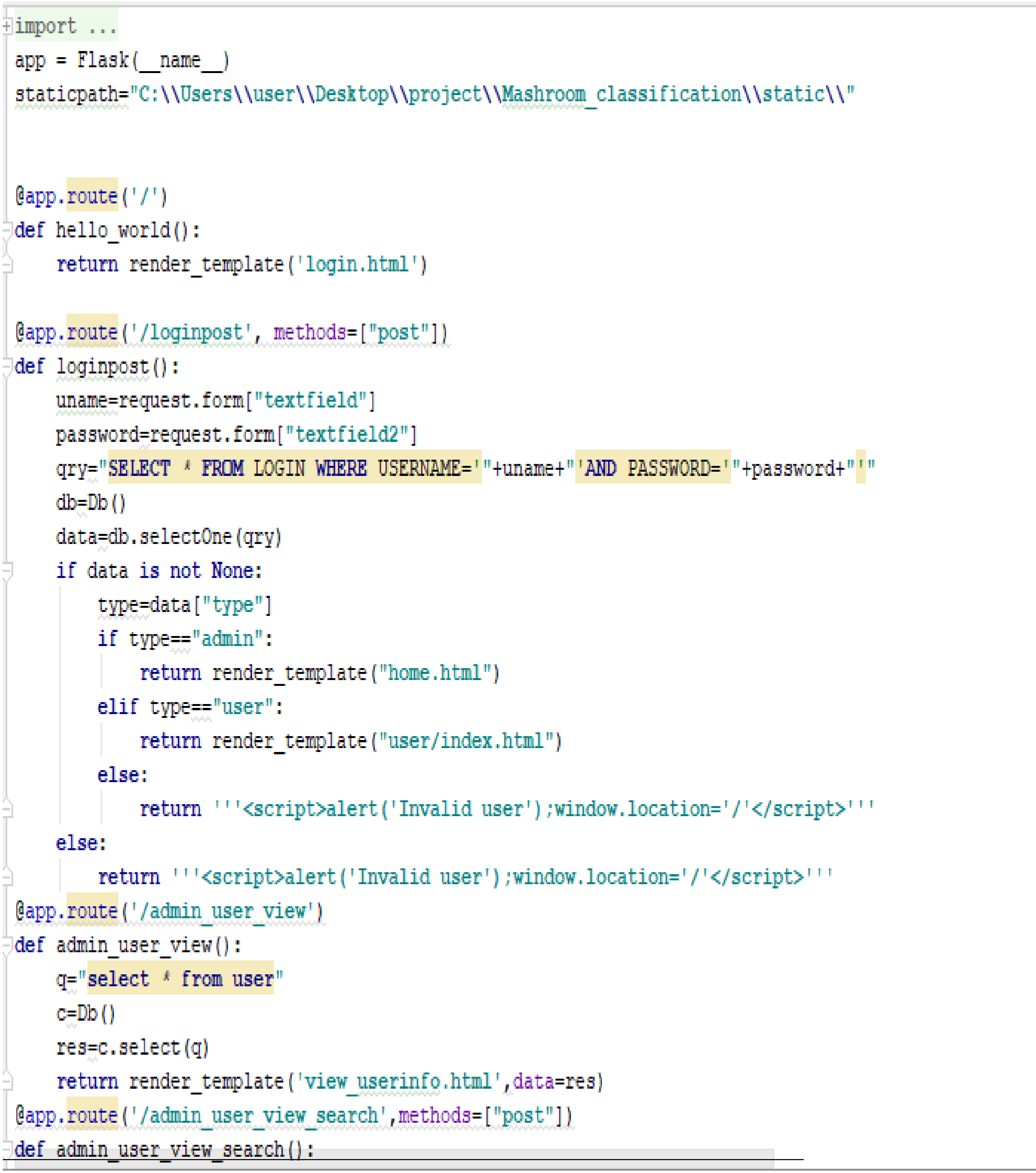
Proposed a mushroom disease prediction is to predict the disease effected to the mushroom and give a detail description about the disease to the user and the remedies to those disease using the Machine learning tools. This system can be also used to classify the mushroom based on certain properties of the mushroom.and help in reducing human intervention. From the result, when tested against the trash dataset, i got an accuracy of 83.48 percentage. The prediction process will be faster and intelligent using our system without or reducing human involvement.

- **Future Scope**

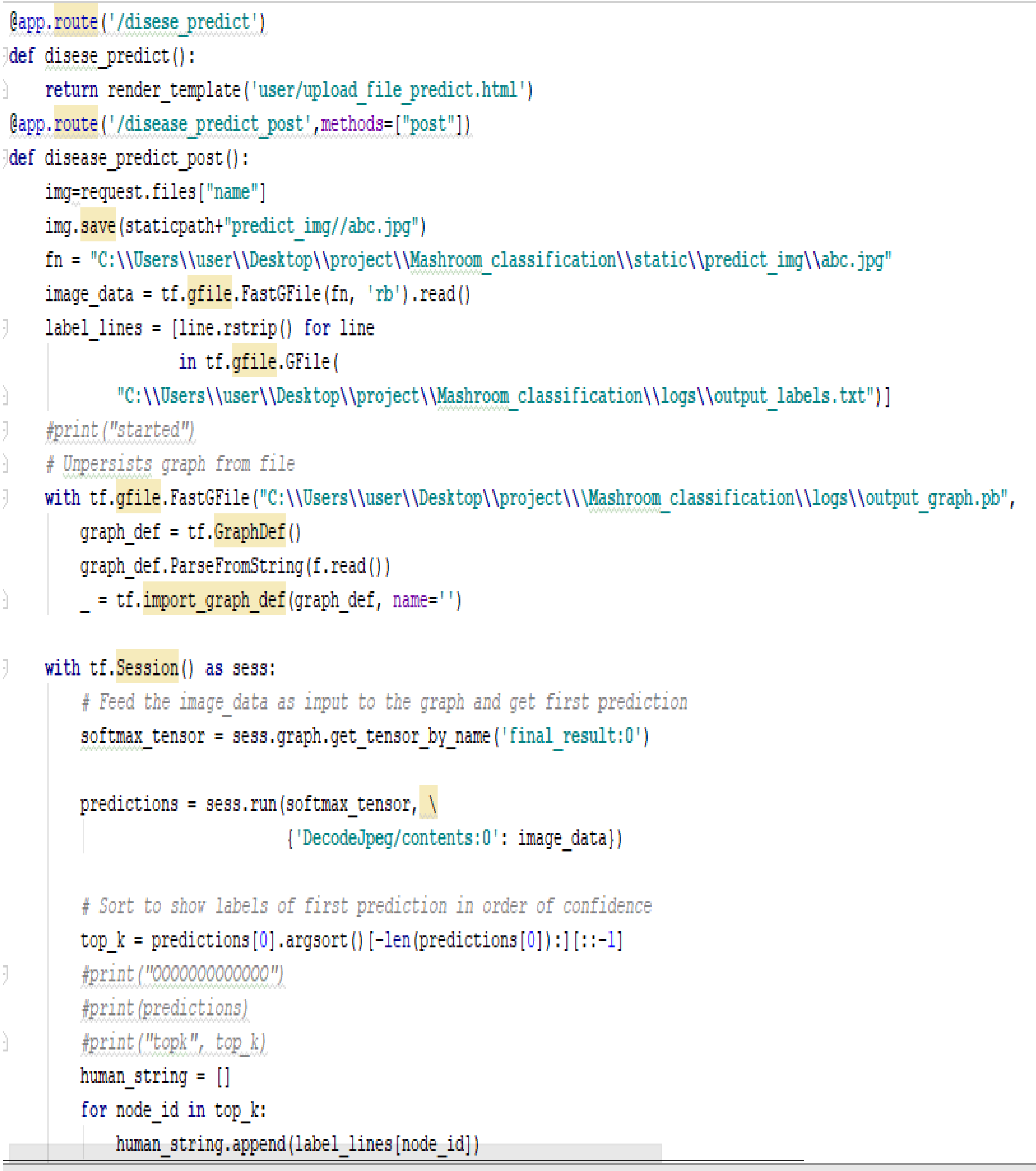
This project can be further enhanced to provide greater flexibility and performance with certain modification whenever necessary. If more image is added to the dataset, the system accuracy can be improved In the future, I will tend to improve our system to be able to categories more disease item.

Chapter 6

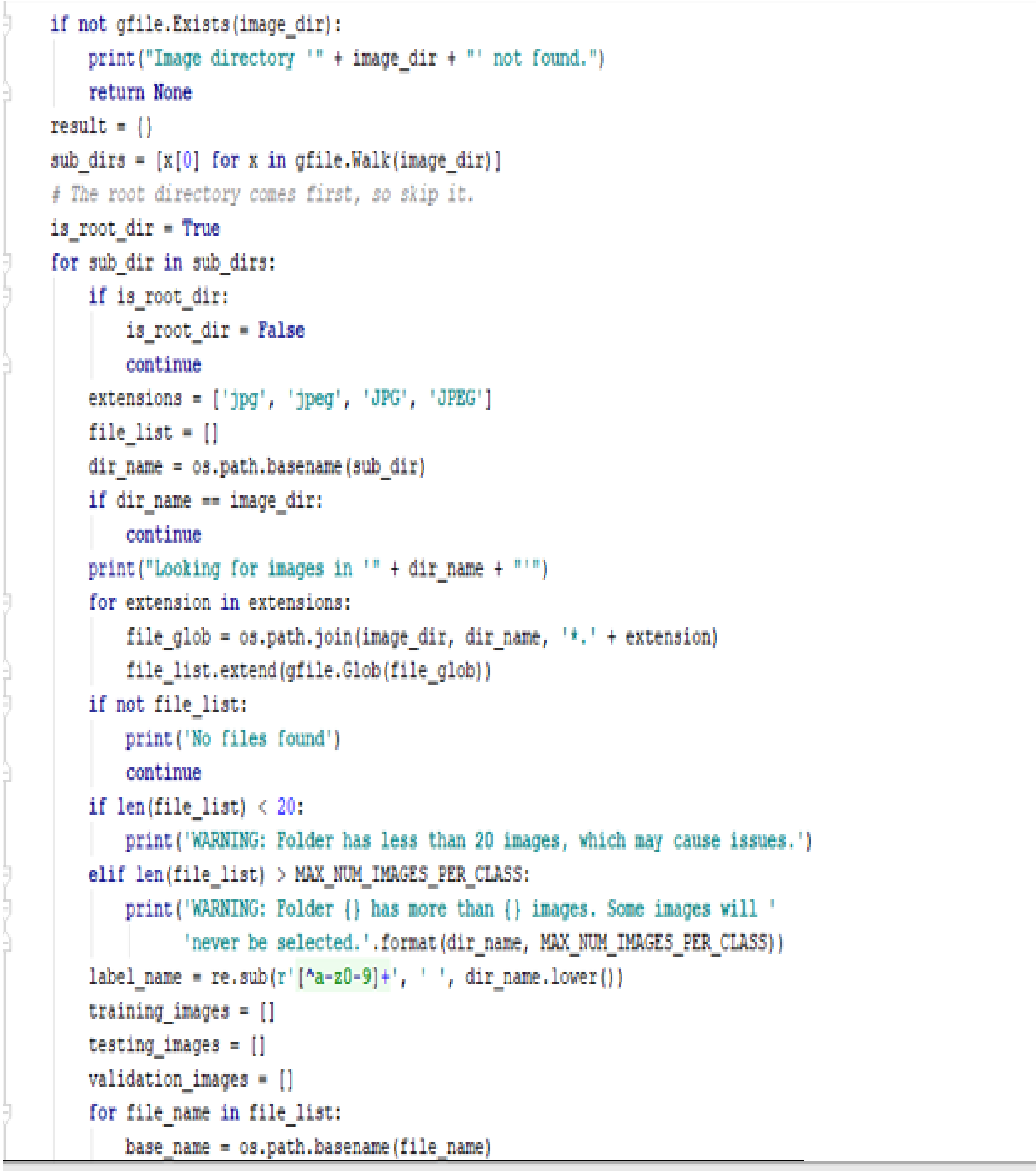
APPENDIX



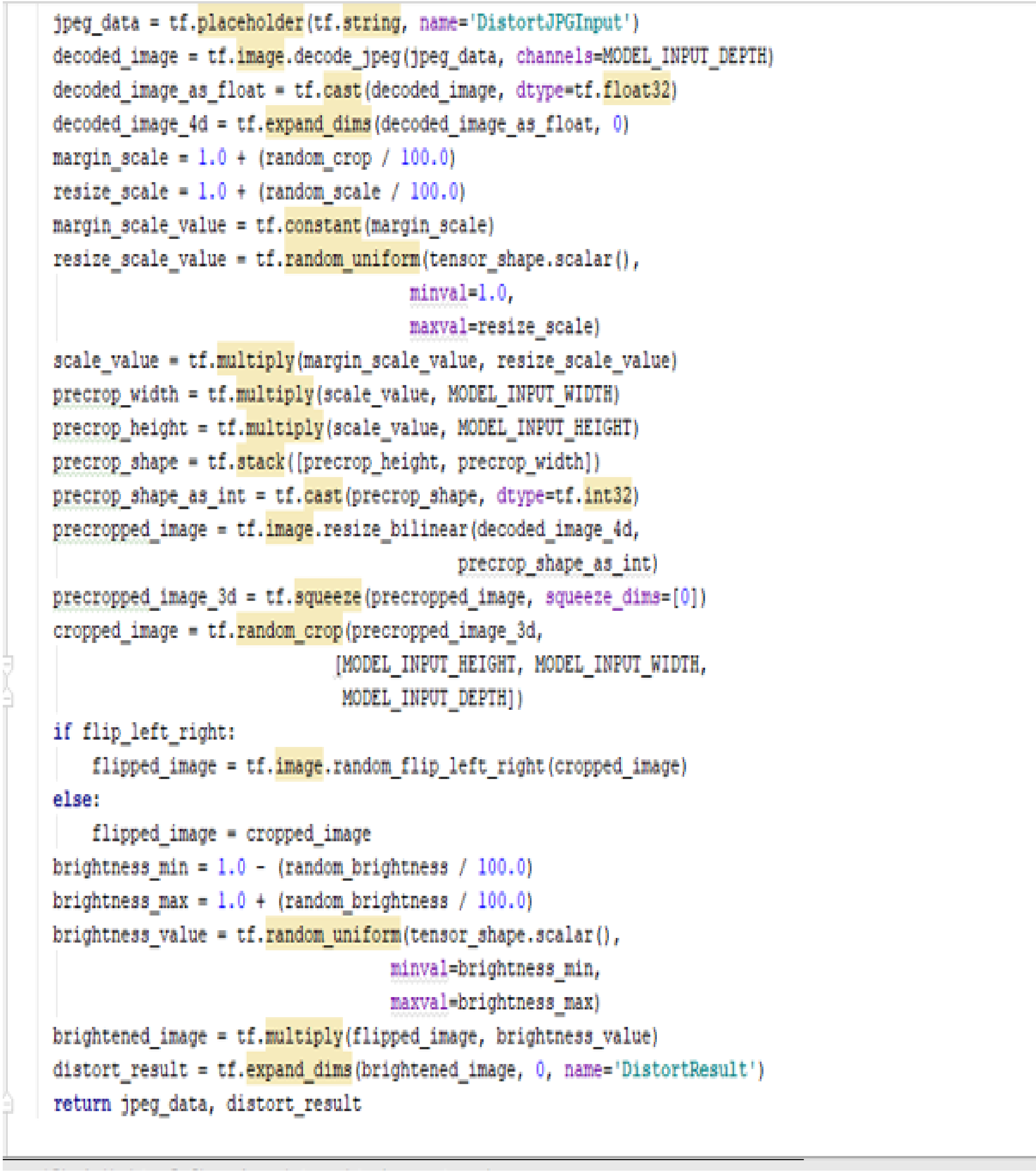
```
import ...
app = Flask(__name__)
staticpath="C:\\Users\\user\\Desktop\\project\\Mashroom_classification\\static\\"


@app.route('/')
def hello_world():
    return render_template('login.html')

@app.route('/loginpost', methods=["post"])
def loginpost():
    uname=request.form["textfield"]
    password=request.form["textfield2"]
    qry="SELECT * FROM LOGIN WHERE USERNAME='"+uname+"'AND PASSWORD='"+password+"'"
    db=Db()
    data=db.selectOne(qry)
    if data is not None:
        type=data["type"]
        if type=="admin":
            return render_template("home.html")
        elif type=="user":
            return render_template("user/index.html")
        else:
            return '''<script>alert('Invalid user');window.location='/'</script>'''
    else:
        return '''<script>alert('Invalid user');window.location='/'</script>'''
@app.route('/admin_user_view')
def admin_user_view():
    q="select * from user"
    c=Db()
    res=c.select(q)
    return render_template('view_userinfo.html',data=res)
@app.route('/admin_user_view_search',methods=["post"])
def admin_user_view_search():
```



```
@app.route('/disese_predict')
def disese_predict():
    return render_template('user/upload_file_predict.html')
@app.route('/disease_predict_post',methods=["post"])
def disease_predict_post():
    img=request.files["name"]
    img.save(staticpath+"predict_img//abc.jpg")
    fn = "C:\\Users\\user\\Desktop\\project\\Mashroom_classification\\static\\predict_img\\abc.jpg"
    image_data = tf.gfile.FastGFile(fn, 'rb').read()
    label_lines = [line.rstrip() for line
                   in tf.gfile.GFile(
            "C:\\Users\\user\\Desktop\\project\\Mashroom_classification\\logs\\output_labels.txt")]
    #print("started")
    # Unpersists graph from file
    with tf.gfile.FastGFile("C:\\Users\\user\\Desktop\\project\\\Mashroom_classification\\logs\\output_graph.pb",
        graph_def = tf.GraphDef()
        graph_def.ParseFromString(f.read())
        _ = tf.import_graph_def(graph_def, name='')

    with tf.Session() as sess:
        # Feed the image_data as input to the graph and get first prediction
        softmax_tensor = sess.graph.get_tensor_by_name('final_result:0')

        predictions = sess.run(softmax_tensor, \
                               {'DecodeJpeg/contents:0': image_data})

        # Sort to show labels of first prediction in order of confidence
        top_k = predictions[0].argsort()[-len(predictions[0]):][::-1]
        #print("0000000000000")
        #print(predictions)
        #print("topk", top_k)
        human_string = []
        for node_id in top_k:
            human_string.append(label_lines[node_id])
```



```
if not gfile.Exists(image_dir):
    print("Image directory '" + image_dir + "' not found.")
    return None
result = {}
sub_dirs = [x[0] for x in gfile.Walk(image_dir)]
# The root directory comes first, so skip it.
is_root_dir = True
for sub_dir in sub_dirs:
    if is_root_dir:
        is_root_dir = False
        continue
    extensions = ['jpg', 'jpeg', 'JPG', 'JPEG']
    file_list = []
    dir_name = os.path.basename(sub_dir)
    if dir_name == image_dir:
        continue
    print("Looking for images in '" + dir_name + "'")
    for extension in extensions:
        file_glob = os.path.join(image_dir, dir_name, '*.' + extension)
        file_list.extend(gfile.Glob(file_glob))
    if not file_list:
        print('No files found')
        continue
    if len(file_list) < 20:
        print('WARNING: Folder has less than 20 images, which may cause issues.')
    elif len(file_list) > MAX_NUM_IMAGES_PER_CLASS:
        print('WARNING: Folder {} has more than {} images. Some images will '
              'never be selected.'.format(dir_name, MAX_NUM_IMAGES_PER_CLASS))
    label_name = re.sub(r'[^a-z0-9]+', ' ', dir_name.lower())
    training_images = []
    testing_images = []
    validation_images = []
    for file_name in file_list:
        base_name = os.path.basename(file_name)
```



```
jpeg_data = tf.placeholder(tf.string, name='DistortJPGInput')
decoded_image = tf.image.decode_jpeg(jpeg_data, channels=MODEL_INPUT_DEPTH)
decoded_image_as_float = tf.cast(decoded_image, dtype=tf.float32)
decoded_image_4d = tf.expand_dims(decoded_image_as_float, 0)
margin_scale = 1.0 + (random_crop / 100.0)
resize_scale = 1.0 + (random_scale / 100.0)
margin_scale_value = tf.constant(margin_scale)
resize_scale_value = tf.random_uniform(tensor_shape.scalar(),
                                       minval=1.0,
                                       maxval=resize_scale)
scale_value = tf.multiply(margin_scale_value, resize_scale_value)
precrop_width = tf.multiply(scale_value, MODEL_INPUT_WIDTH)
precrop_height = tf.multiply(scale_value, MODEL_INPUT_HEIGHT)
precrop_shape = tf.stack([precrop_height, precrop_width])
precrop_shape_as_int = tf.cast(precrop_shape, dtype=tf.int32)
precropped_image = tf.image.resize_bilinear(decoded_image_4d,
                                            precrop_shape_as_int)
precropped_image_3d = tf.squeeze(precropped_image, squeeze_dims=[0])
cropped_image = tf.random_crop(precropped_image_3d,
                               [MODEL_INPUT_HEIGHT, MODEL_INPUT_WIDTH,
                                MODEL_INPUT_DEPTH])
if flip_left_right:
    flipped_image = tf.image.random_flip_left_right(cropped_image)
else:
    flipped_image = cropped_image
brightness_min = 1.0 - (random_brightness / 100.0)
brightness_max = 1.0 + (random_brightness / 100.0)
brightness_value = tf.random_uniform(tensor_shape.scalar(),
                                     minval=brightness_min,
                                     maxval=brightness_max)
brightened_image = tf.multiply(flipped_image, brightness_value)
distort_result = tf.expand_dims(brightened_image, 0, name='DistortResult')
return jpeg_data, distort_result
```

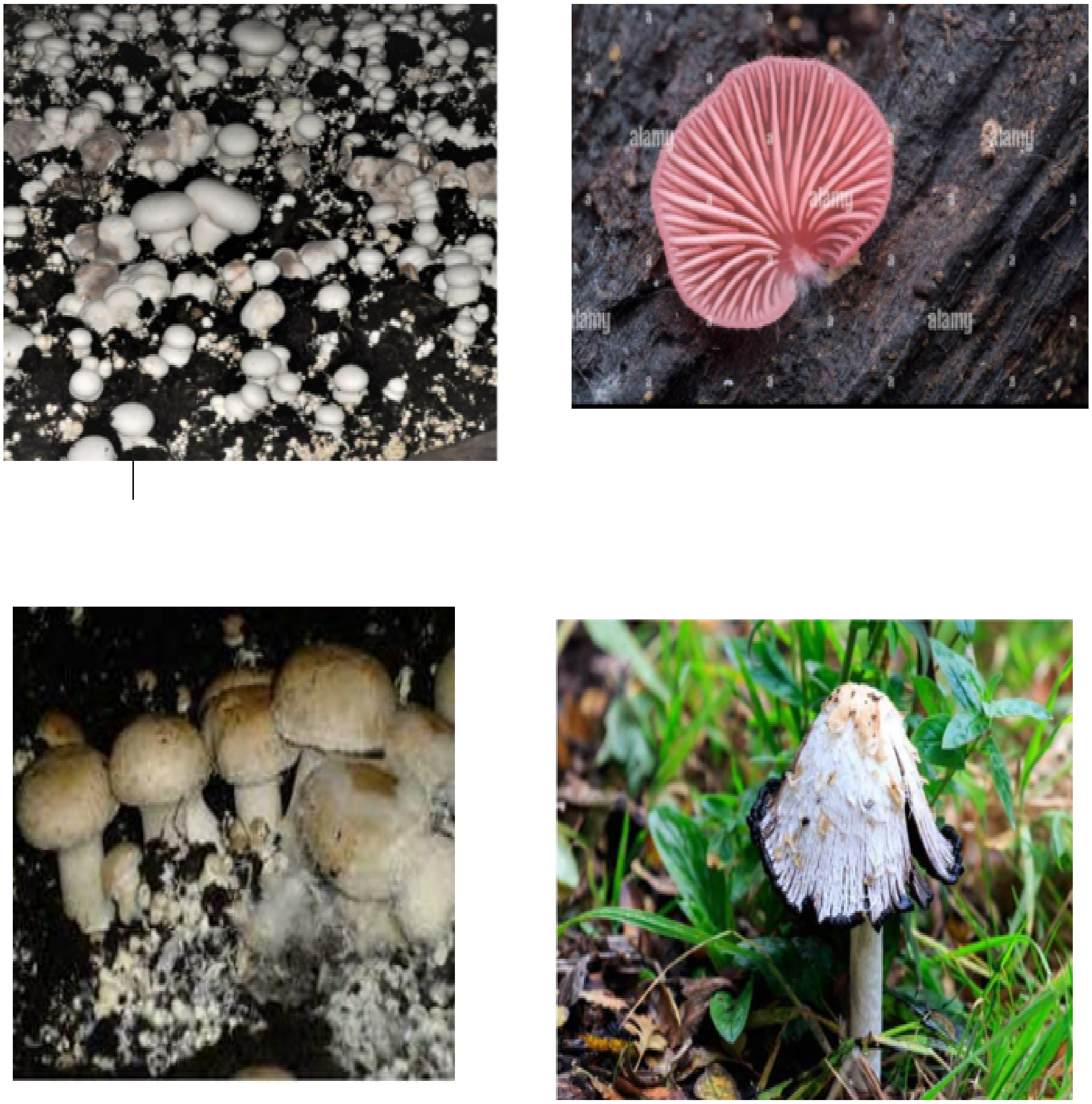


Figure 6.1: Sample Dataset.

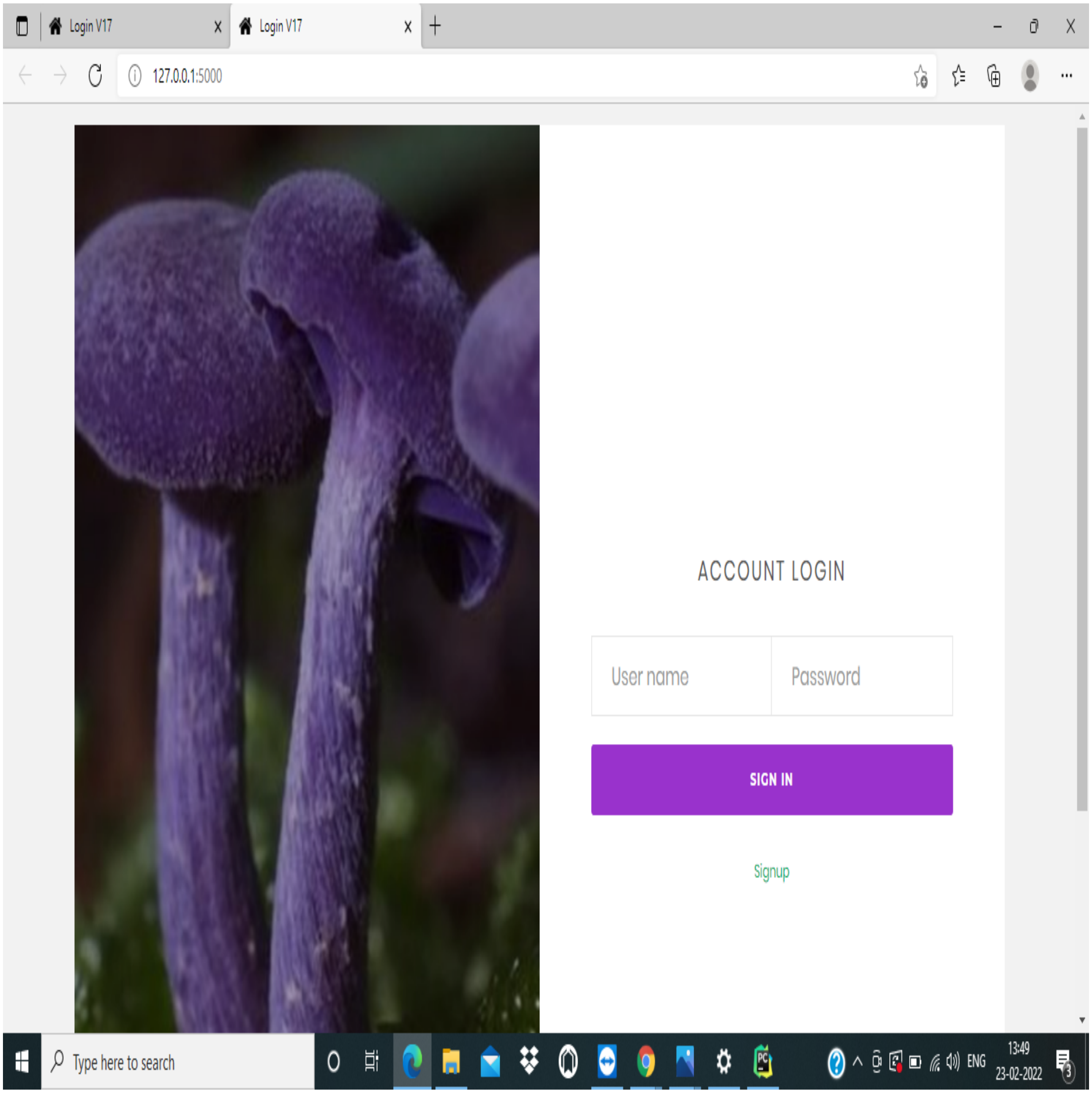
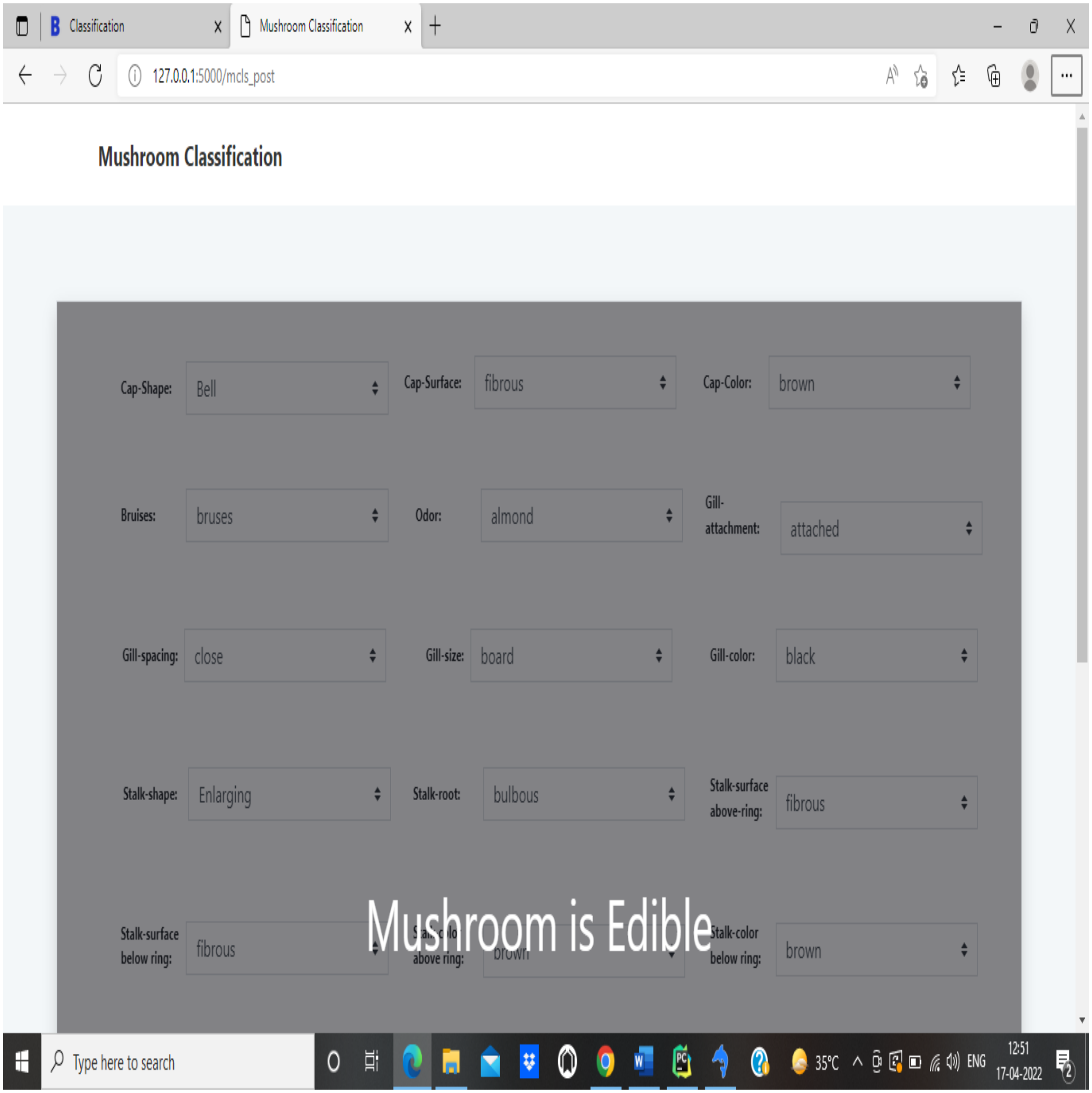


Figure 6.2: UI Design



Mushroom Classification

Cap-Shape: Bell
Cap-Surface: fibrous
Cap-Color: brown

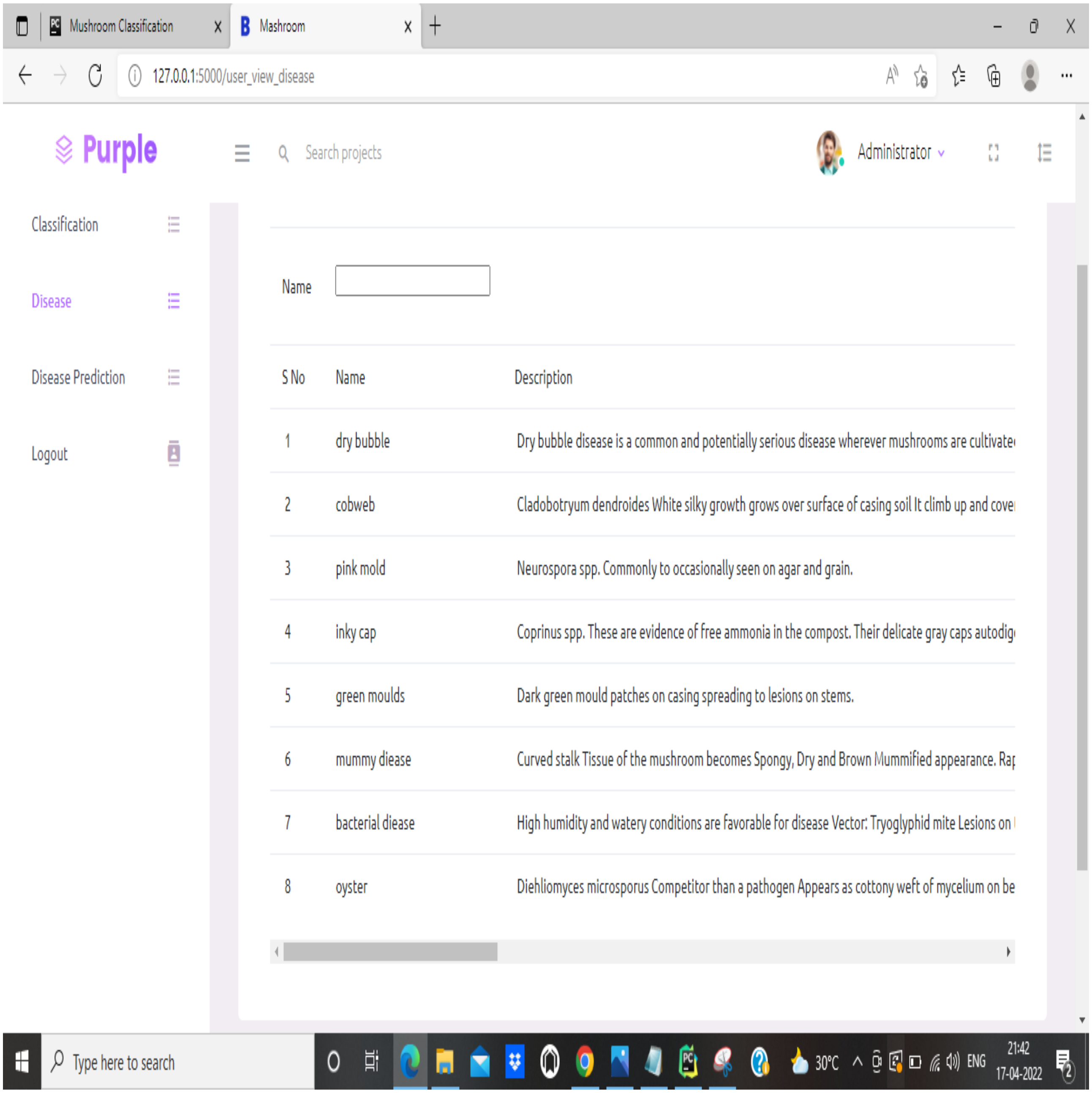
Bruises: bruses
Odor: almond
Gill-attachment: attached

Gill-spacing: close
Gill-size: board
Gill-color: black

Stalk-shape: Enlarging
Stalk-root: bulbous
Stalk-surface above-ring: fibrous

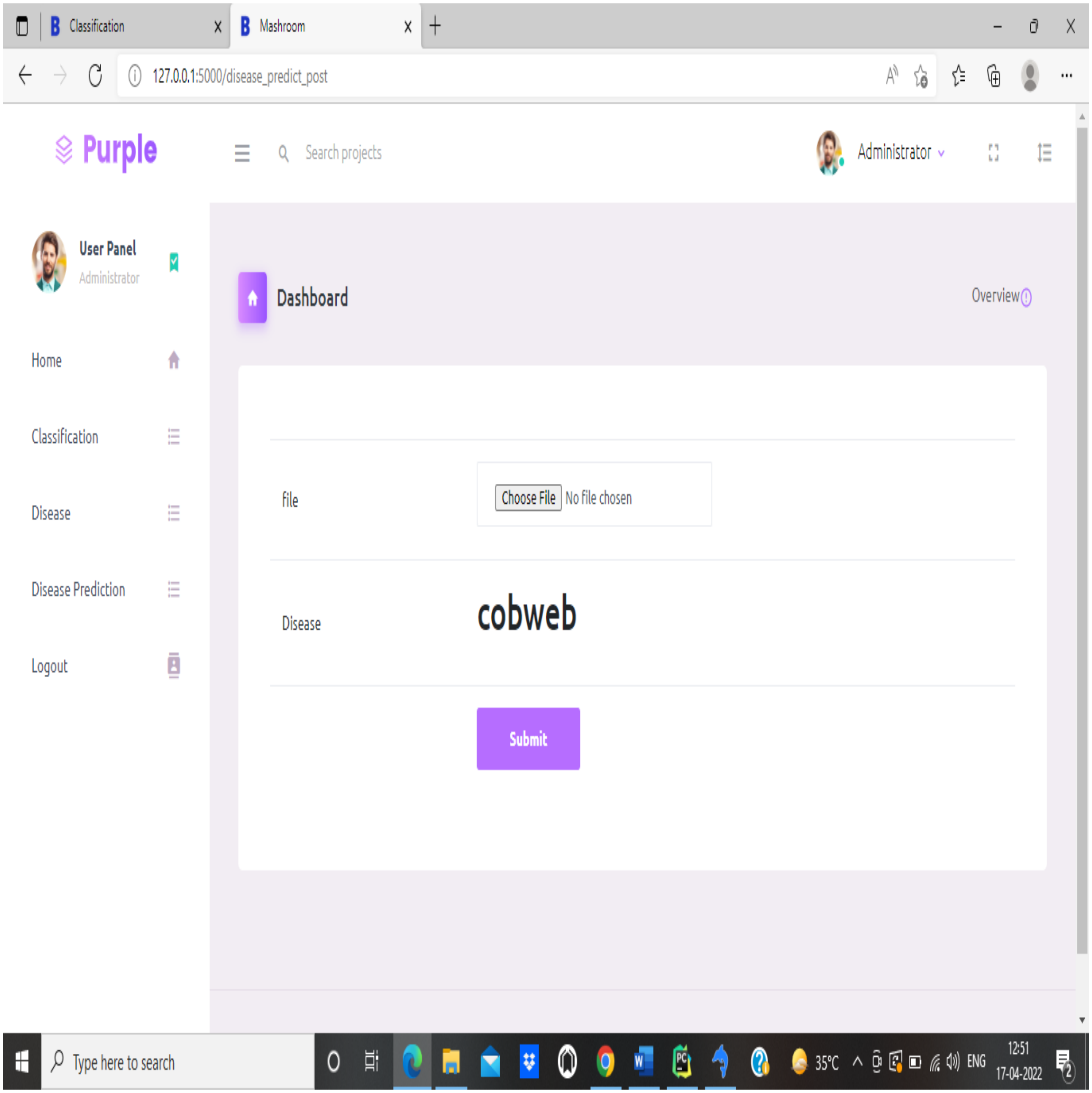
Stalk-surface below ring: fibrous
Stalk-color above ring: brown
Stalk-color below ring: brown

Mushroom is Edible



Name □

S No	Name	Description
1	dry bubble	Dry bubble disease is a common and potentially serious disease wherever mushrooms are cultivate
2	cobweb	Cladobotryum dendroides White silky growth grows over surface of casing soil It climb up and cove
3	pink mold	Neurospora spp. Commonly to occasionally seen on agar and grain.
4	inky cap	Coprinus spp. These are evidence of free ammonia in the compost. Their delicate gray caps autodig
5	green moulds	Dark green mould patches on casing spreading to lesions on stems.
6	mummy diease	Curved stalk Tissue of the mushroom becomes Spongy, Dry and Brown Mummified appearance. Ra
7	bacterial diease	High humidity and watery conditions are favorable for disease Vector: Tryoglyphid mite Lesions on
8	oyster	Diehliomyces microsporus Competitor than a pathogen Appears as cottony weft of mycelium on be



File: No file chosen

Disease: cobweb

Chapter 7

REFERENCES

[1] K. Mallikarjuns, N. John Sushma, G. Narasimha, K. VenkateswaraRao,L.Manojan
of palladium nanoparticles by using broth of ediblemushroomextract,”InternationalConferenceonNanoscience,Engineering and Technology (ICONSET 2011), Chennai, 2011, pp.612-615,doi:10.1109/ICONSET.20

[2] Fangkun Zhu, Li Qu, Wang Xuejing and Fan Wenxiu, ”Heavy metalsinsomeediblemushroomsconsumedinXinxiang,China,”2011Internation
Conference on New Technology of Agricultural, Zibo,2011,pp. 1034-1036,doi:10.1109/ICAE.2011.5943965.

[3]Z. Zhang,H. PanandL. Fan, ”AntioxidantPropertiesofDifferent Solvents Extracts from Three Edible Mushrooms,” 2009 3rdInternationalConferenceonBioinformaticsandBiomedicalEngineering, Beijing, 2009, pp. 1-4, doi:10.1109/ICBBE.2009.5162375.

[4] N. Chumuang, P. Pramkeaw and A. Farooq, ”Electrical Impedance OfBreast’s Tissue Classification By Using Bootstrap Aggregating,” 201915thInternationalConferenceonSignal-

ImageTechnologyInternet-Based Systems (SITIS), Sorrento, Italy, 2019, pp. 551-556,doi:10.1109/SITIS.2019.00093.

[5]J.Vieira,R.P.DuarteandH.C.Neto,”kNN-STUFF:kNNSTreamingUnitforFpgas,”inI
170877,2019,doi:10.1109/ACCESS.2019.2955864.

[6] H.Chen,L.Fu,P. Dong,X. Zhang andX.Lu,”Acute Toxicity Evaluation and Compositional Analysis of a Polysaccharide from theMedicinalMushroomInonotusobliquus,”20093rdInternationalConference on Bioinformatics and Biomedical Engineering, Beijing,2009,pp. 1-4,doi:10.1109/ICBBE.2009.5162275.

[7] A. Masoudian and K. A. Mcisaac, ”Application of Support VectorMachinetoDetectMicrobialSpoilageofMushrooms,”2013International Conference on Computer and Robot Vision, Regina, SK,2013,pp. 281-287,doi: 10.1109/CRV.2013.10.

[8] T. Pinthong, W. Yimyam, N. Chumuang and M. Ketcham, ”LicensePlate Tracking Based on Template Matching Technique,” 2018 18thInternationalSymposiumonCommunicationsandInformationTechno
303,doi:10.1109/ISCIT.2018.8587992.

[9]C. Liou et al., ”Design and Fabrication of Micro-LED Display withSapphireMicro-ReflectorArray,”2020IEEE33rdInternationalConferenceonM
1156,doi:10.1109/MEMS46641.2020.9056340.

[10]F. Zhao and Q. Tang, ”A KNN Learning Algorithm for Collusion-ResistantSpectrumAuctioninSmallCellNetworks,”inIEEEAccess,vol.6,pp.
45803,2018,doi:10.1109/ACCESS.2018.2861840.

[11]K.Thirunavukkarasu,A.S.Singh,P.RaiandS.Gupta,”ClassificationofIRISDatasetus
Communication and Automation (ICCCA), Greater Noida,India,2018,pp.1-4,doi: 10.1109/CCAA.2018.8777643.

[12] P.Tamrakar,S.S.Roy,B.SatapathyandS.P.S.Ibrahim,”Integrationof lazy learning associative classification with kNN algorithm,” 2019InternationalConferenceonVisionTowardsEmergingTrendsin-Communication and Networking (ViTECoN), Vellore, India, 2019, pp.1-4,doi:10.1109/ViTECoN.2019.8899415.