**Leukemia Cancer Detection using Image Classification**

CAPSTONE PROJECT REPORT

By

Team 16

E Saikiran [22R15A0514]

J Sinduja [21R11A0574]

Naveen Rampa [21R11A0589]

Lakshanya S [21R11A05H3]

A Bhavya Sri [21R11A05A5]



**Department of Computer Science and Engineering**

**Accredited by NBA**

**Geethanjali College of Engineering and Technology**

**(UGC Autonomous)**

(Affiliated to J.N.T.U.H, Approved by AICTE, New Delhi)

Cheeryal (V), Keesara (M), Medchal.Dist.-501 301

March 2024

**ABSTRACT**

Leukemia, a devastating form of cancer affecting individuals of all ages, necessitates early and accurate diagnosis for effective treatment and improved survival rates. Current diagnostic methods, relying on manual analysis of microscopic blood samples, are time-consuming and prone to errors due to the similarity between leukemic and normal cells. To address these challenges, this project focuses on developing intelligent deep learning algorithms using Convolutional Neural Network (CNN), ResNet, and VGG models.

The dataset comprises a large number of microscopic images of leukemic blood cell patterns divided into train, test, and validation sets. The proposed algorithms are trained and evaluated to classify malignant cells from normal cells, aiding in the early detection of leukemia.

Results demonstrate the efficacy of the CNN, ResNet, and VGG models in accurately distinguishing between leukemic and normal cells, with promising performance metrics such as accuracy, precision, recall, and F1-score. This research contributes to the ongoing efforts in leveraging artificial intelligence for improving cancer diagnosis and underscores the potential of intelligent algorithms in enhancing healthcare outcomes. **TABLE OF CONTENTS**

|  |  |  |
| --- | --- | --- |
| **Chapter No** | **Chapter Name** | **Page No** |
|  | **ABSTRACT** |  |
| **1** | **INTRODUCTION** | 1 |
| **2** | **PROBLEM STATEMENT** | 3 |
| **3** | **OBJECTIVES** | 3 |
| **4** | **METHODOLOGY** | 4 |
|  | **4.1 Data Collection and Preprocessing** | 4 |
|  | **4.2 Model Selection and Architecture Design** | 10 |
|  | 4.2.1 Convolutional Neural Network (CNN) | 10 |
|  | 4.2.2 Visual Geometry Group (VGG) Network | 14 |
|  | 4.2.3 Residual Network (ResNet) | 15 |
|  | 4.2.4 MobileNet | 17 |
| **5** | **Literature Review** | 18 |
| **6** | **Final Implementation** | 19 |
|  | **Conclusion** | 21 |

1. **INTRODUCTION**

Leukemia, a form of cancer affecting the blood and bone marrow, poses significant challenges in diagnosis and treatment. It manifests in various subtypes, impacting individuals of all ages and presenting a critical concern in healthcare. The disease is characterized by the proliferation of abnormal white blood cells (WBCs), which can lead to severe health complications and, if left untreated, may result in fatalities. Prompt and accurate diagnosis is paramount for initiating timely interventions and improving patient outcomes.

Traditional diagnostic methods for leukemia often involve manual examination of blood samples under a microscope. However, this approach is labor-intensive, time-consuming, and prone to human error. Moreover, distinguishing between leukemic cells and normal cells based on visual inspection alone can be challenging, as the two may exhibit similar morphological features.

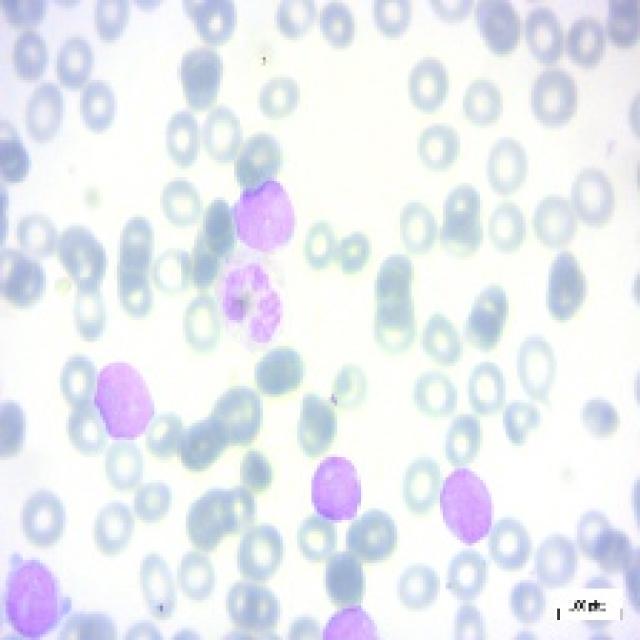


Fig 1.1: showing the pro stage microscopic image of leukemia cancer

To address these limitations and enhance the diagnostic process, advanced technologies such as machine learning and deep learning algorithms have been leveraged. These intelligent systems can analyze large volumes of medical images with a high degree of accuracy and efficiency, aiding in the early detection and classification of leukemia subtypes.

In this context, our project focuses on developing and evaluating deep learning models, including Convolutional Neural Networks (CNNs), ResNet, and VGG neural networks, for leukemia detection. The project utilizes a comprehensive dataset comprising a diverse range of leukemic blood cell patterns, meticulously divided into training, testing, and validation sets.

By harnessing the power of artificial intelligence and deep learning, we aim to create automated and reliable systems capable of distinguishing between malignant and normal blood cells. The outcomes of this research endeavor hold promise in revolutionizing leukemia diagnosis, paving the way for improved patient care and outcomes in the realm of oncology.

**2. PROBLEM STATEMENT**

The manual analysis of blood samples for leukemia diagnosis is a time-consuming and error-prone process due to the subjective nature of visual interpretation and the similarity between leukemic and normal cells. This leads to delays in diagnosis and treatment initiation, impacting patient outcomes and survival rates. Additionally, the increasing incidence of leukemia globally necessitates more efficient and accurate diagnostic methods to meet the growing healthcare demands.

The primary challenge addressed in this project is to develop intelligent algorithms capable of automating the detection and classification of leukemic cells from normal cells in microscopic blood samples

**3. OBJECTIVES**

The primary objectives of this project are to develop intelligent algorithms using deep learning techniques, specifically Convolutional Neural Network (CNN), ResNet, and VGG neural network models, to automate the detection and classification of leukemic cells from normal cells in microscopic blood samples. The first objective is to create robust deep learning models trained on a diverse dataset of leukemic blood cell patterns, meticulously divided into training, testing, and validation sets. These models will be designed to accurately differentiate between different leukemia subtypes and normal blood cells based on microscopic images, overcoming the limitations of manual analysis. The second objective is to enhance the efficiency and accuracy of leukemia diagnosis by implementing automated systems capable of rapidly and accurately analyzing large volumes of medical images. The project will also focus on validating the performance of the developed models using comprehensive evaluation metrics to ensure their reliability and effectiveness in real-world leukemia detection scenarios. Overall, this research aims to contribute to healthcare innovation by leveraging artificial intelligence to improve leukemia diagnosis, leading to better patient outcomes and enhanced healthcare accessibility in the field of oncology.

**4. METHODOLOGY**

**4.1 Data Collection and Preprocessing**

Obtained a comprehensive dataset of microscopic images of leukemic blood cell patterns, categorized into train, test, and validation sets.

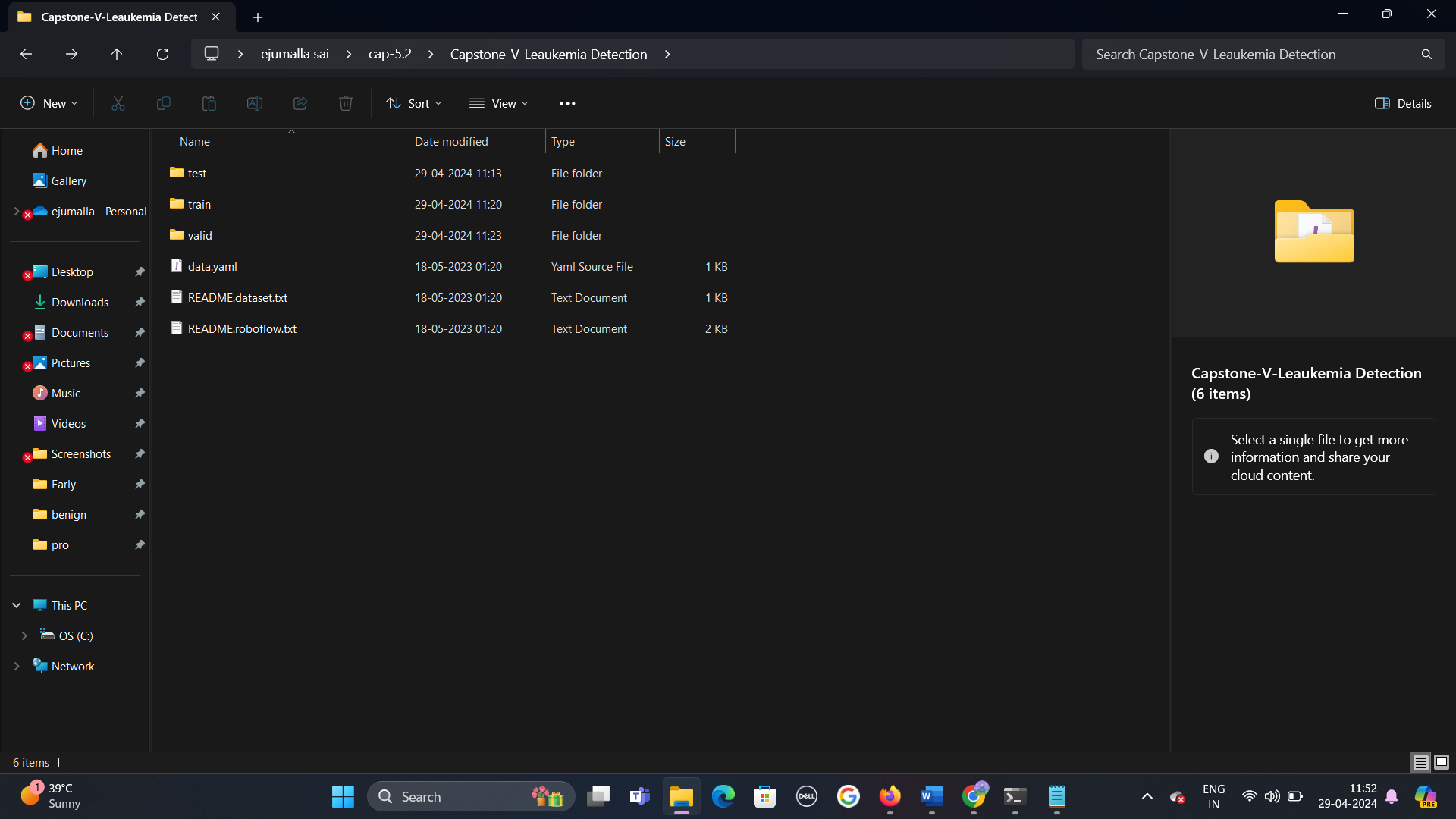


Fig 4.1.1: Indicates how the test, train and valid folders are categorized.

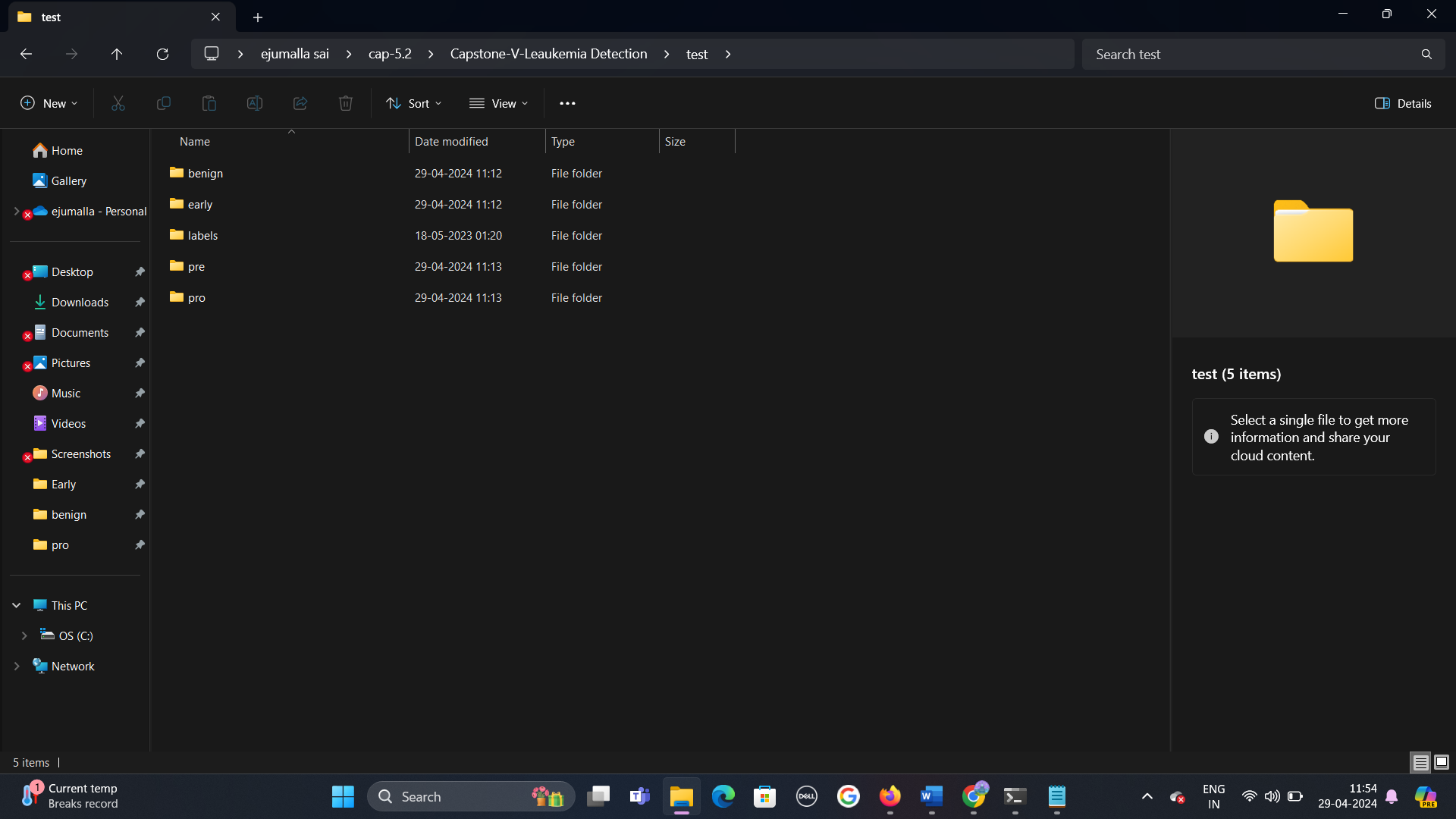


Fig 4.1.2: Indicates test, train, valid folders contain 5 classes pre, pro, early, benign and labels.

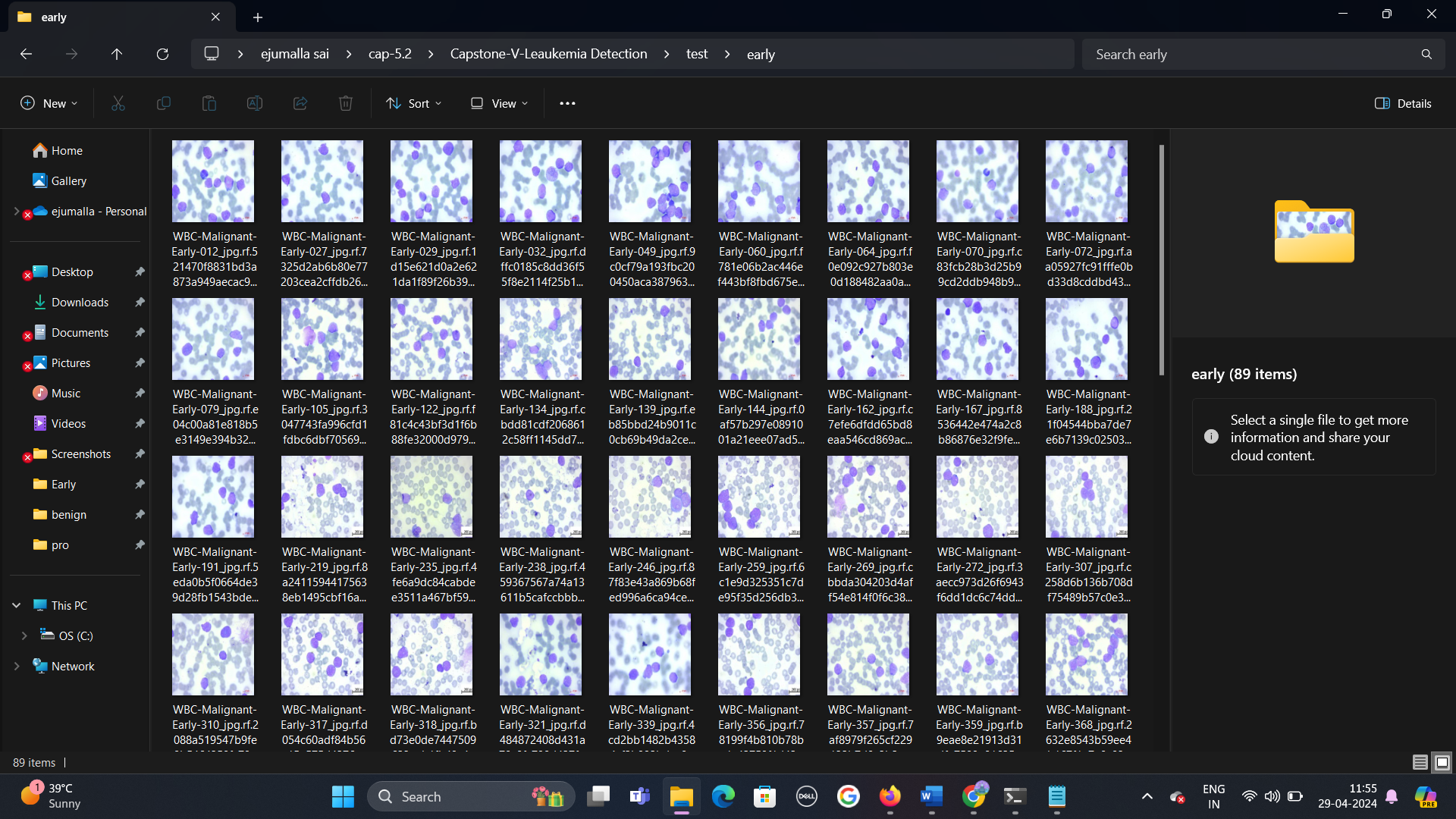


Fig 4.1.3:Indicates how the images are categorized.

Preprocess the images by resizing them to a standard size, applying normalization techniques, and augmenting the data to increase the diversity and robustness of the dataset.

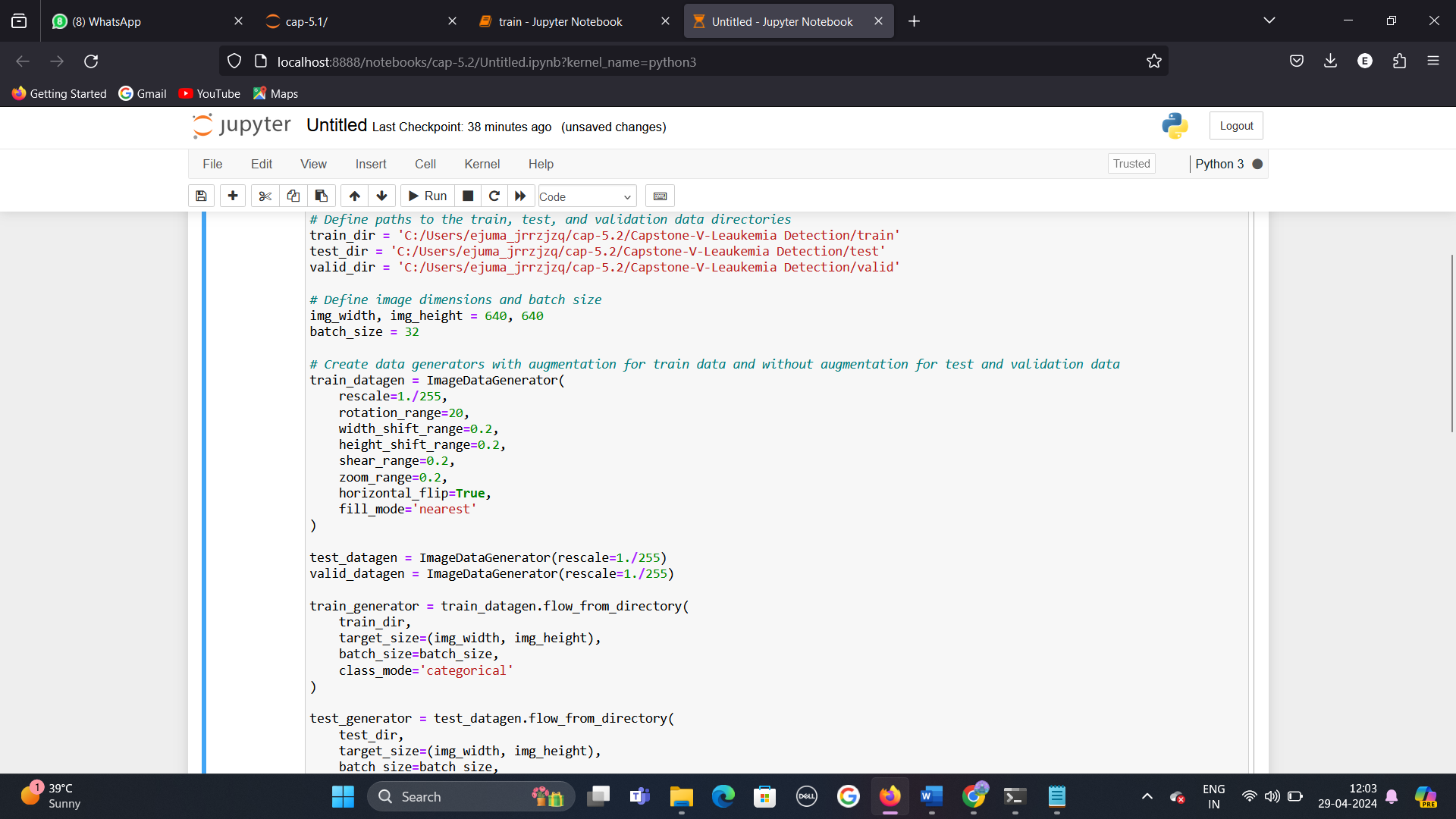
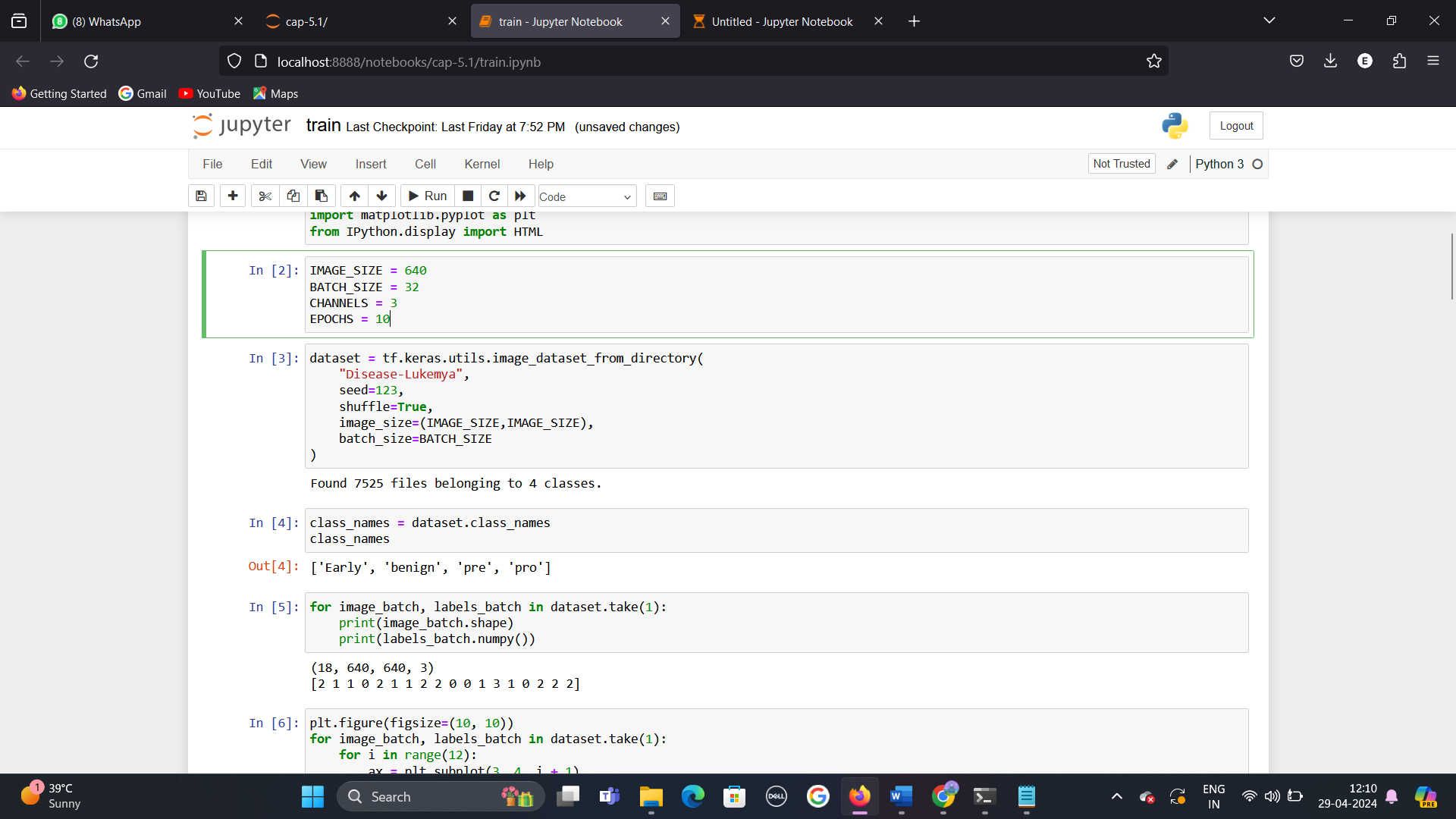


Fig 4.1.4: Indicates augmenting the data to increase the diversity and robustness of the dataset.



**Observation:**

IMAGE\_SIZE = 640:

This parameter refers to the dimensions of the input images in the dataset. Specifically, it indicates that the images are resized to a square shape with both width and height set to 640 pixels. Resizing images to a standard size is common in machine learning tasks to ensure uniformity and compatibility across the dataset.

BATCH\_SIZE = 32:

Batch size refers to the number of samples (images in this case) that are processed and propagated through the neural network in one forward and backward pass during each training iteration. A batch size of 32 means that 32 images will be processed together in each training step. Adjusting batch size can impact training speed, memory usage, and model convergence.

CHANNELS = 3:

Channels refer to the colour channels in an image. In RGB (Red, Green, Blue) colour space, each pixel in an image is represented by three values corresponding to the intensity of red, green, and blue. Therefore, an image with 3 channels is an RGB image, where each pixel is represented by a triplet of values (R, G, B). Grayscale images typically have 1 channel, while colour images like photographs have 3 channels.

EPOCHS = 10:

An epoch is one complete pass of the entire training dataset through the neural network. Setting EPOCHS = 10 means that the training process will iterate over the entire dataset 10 times. Increasing the number of epochs can allow the model to learn more complex patterns from the data but may also increase the risk of overfitting if not controlled properly.

Overfitting is a common problem in machine learning, including deep learning models like neural networks. It occurs when a model learns the training data too well to the point that it negatively impacts its ability to generalize to new, unseen data.

**Breakdown of overfitting:**

**Training Data Fit:** During the training phase, a machine learning model learns patterns and relationships from the training data. The goal is to capture the underlying structure of the data that enables accurate predictions.

**Complexity and Generalization**: A model that is too complex or has too many parameters can memorize the training data instead of learning generalizable patterns. This leads to overfitting, where the model performs exceptionally well on the training data but poorly on new, unseen data.

**Signs of Overfitting:**

**High Training Accuracy:** The model achieves high accuracy or performance metrics on the training data.

**Low Validation/Test Accuracy:** However, when evaluated on validation or test data (data not seen during training), the model's performance drops significantly.

**Excessive Sensitivity:** The model may exhibit sensitivity to noise or irrelevant features in the training data, capturing patterns that don't generalize.

**Impact of Overfitting:**

**Poor Generalization**: Overfitting prevents the model from generalizing well to new data, leading to inaccurate predictions in real-world scenarios.

**Reduced Robustness:** The model's robustness decreases as it becomes overly specialized to the training data, making it less adaptable to variations or changes in the data distribution.

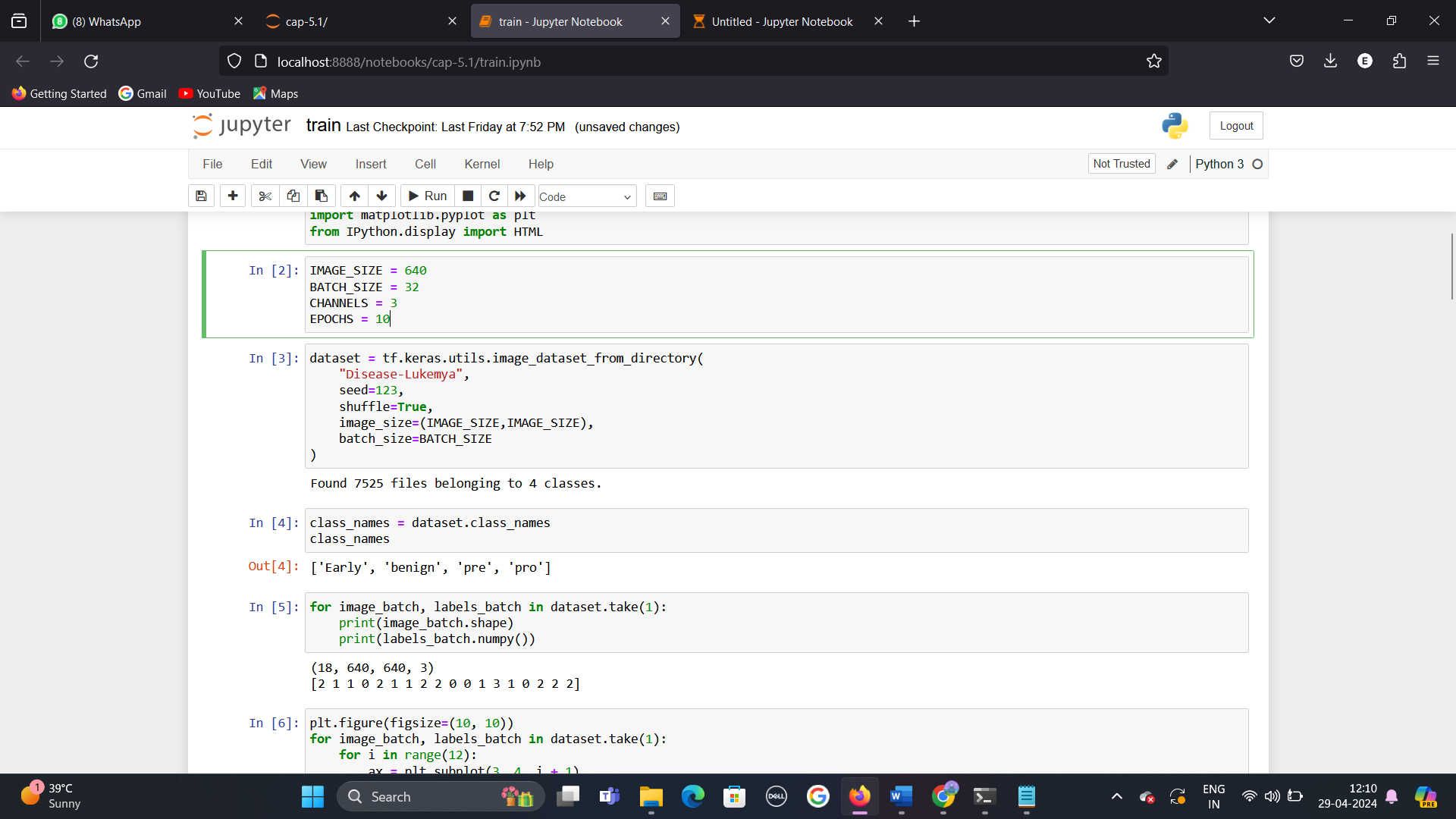
**Preventing Overfitting:**

**Regularization Techniques**: Techniques like L1 and L2 regularization penalize large weights in the model, promoting simpler models that generalize better.

**Cross-Validation:** CUsing techniques like k-fold cross-validation helps assess a model's performance on multiple subsets of the data, revealing its generalization capabilities.

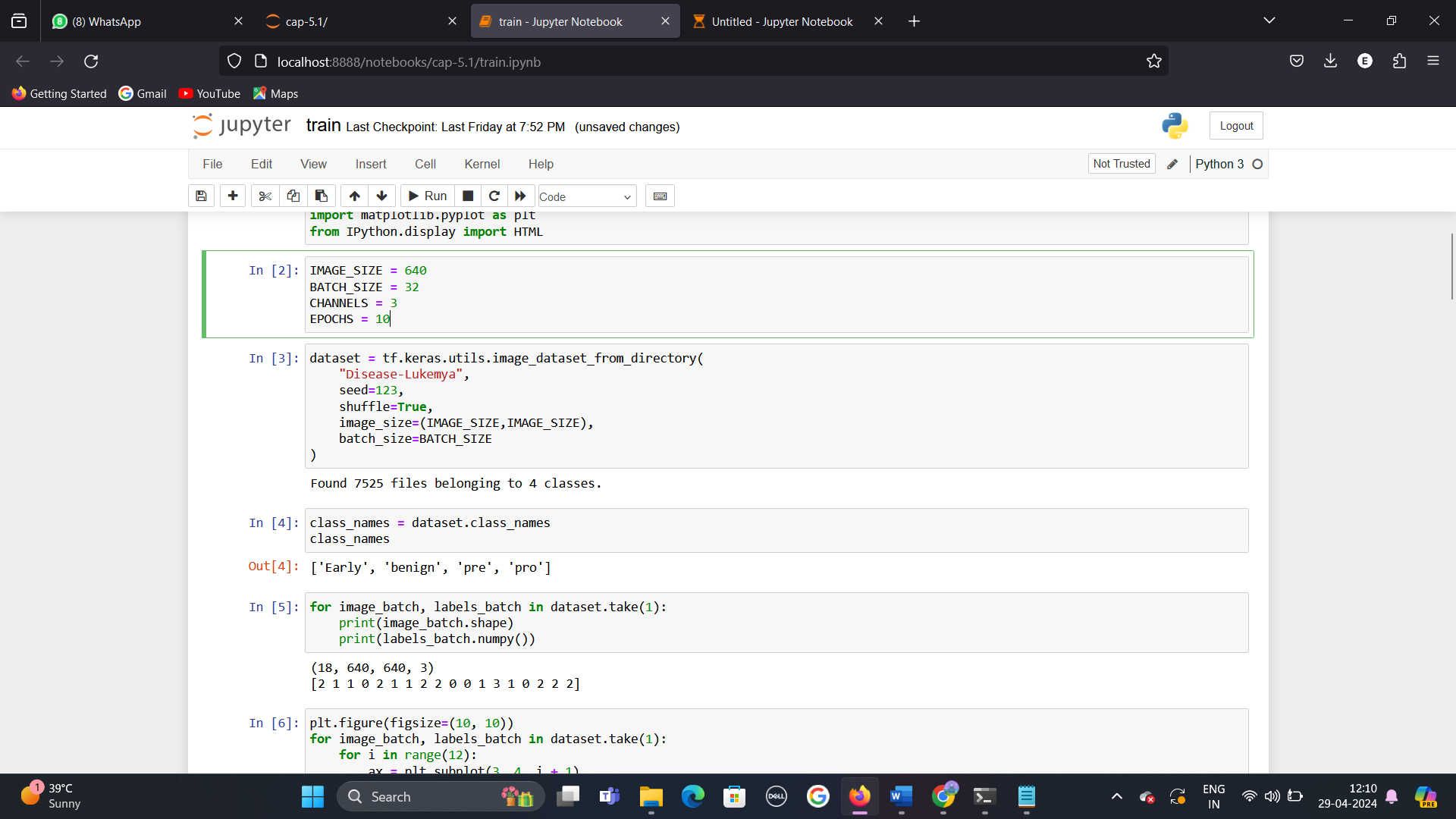
**Data Augmentation:** In tasks like image classification, augmenting the training data with transformations like rotations, flips, and zooms can help the model learn robust features.

**Early Stopping:** Monitoring the model's performance on a validation set during training and stopping when performance starts to degrade can prevent overfitting.



**Observation:**

The dataset will automatically handle tasks such as loading images from disk, resizing them to the specified size, shuffling, batching, and optionally applying data augmentation if configured.



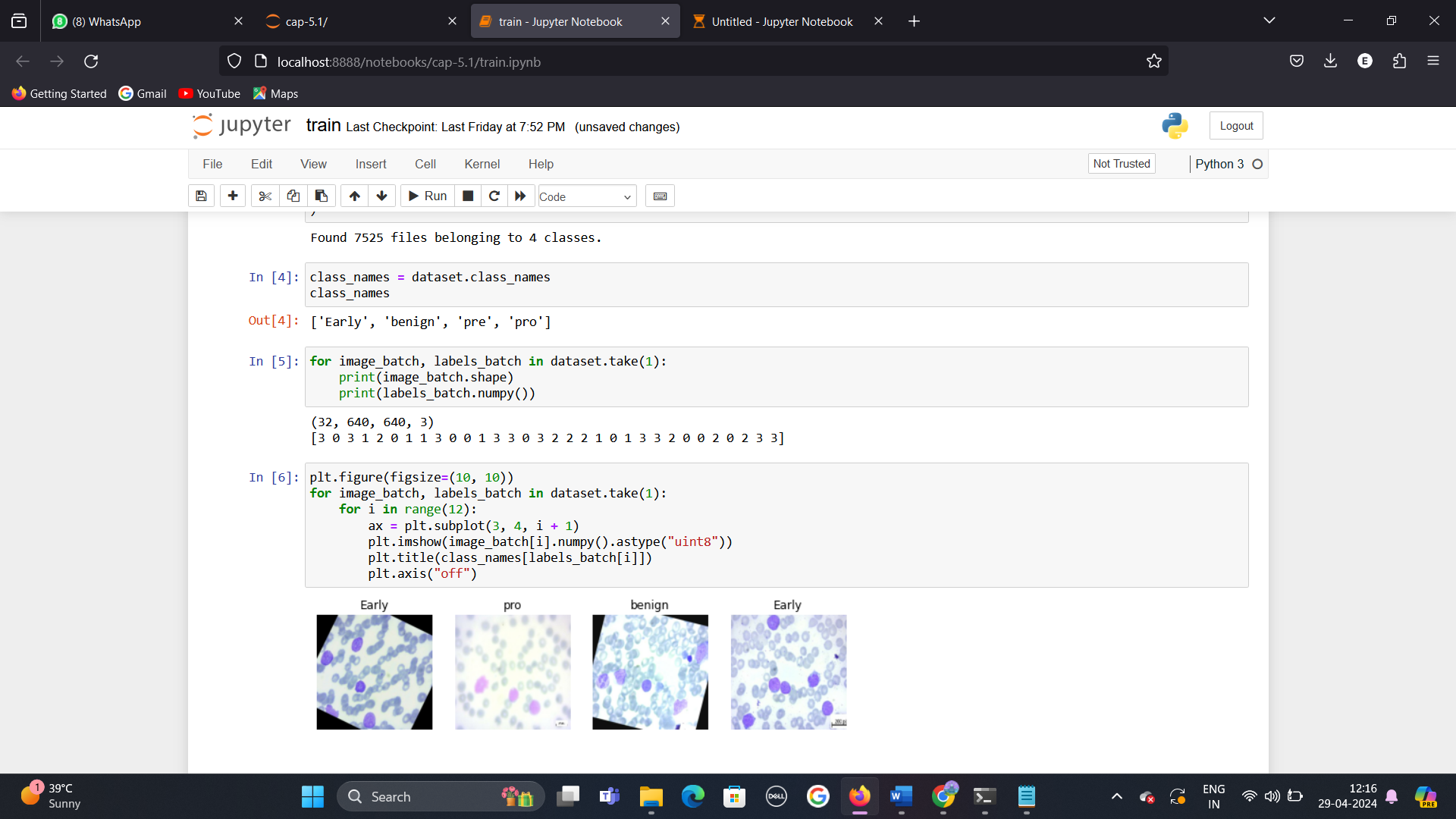
**Observation:**

'Early': Represents images related to the "early" stage of the disease.

'benign': Represents images categorized as "benign" or non-cancerous.

'pre': Represents images related to the "pre" stage of the disease.

'pro': Represents images related to the "pro" stage of the disease.

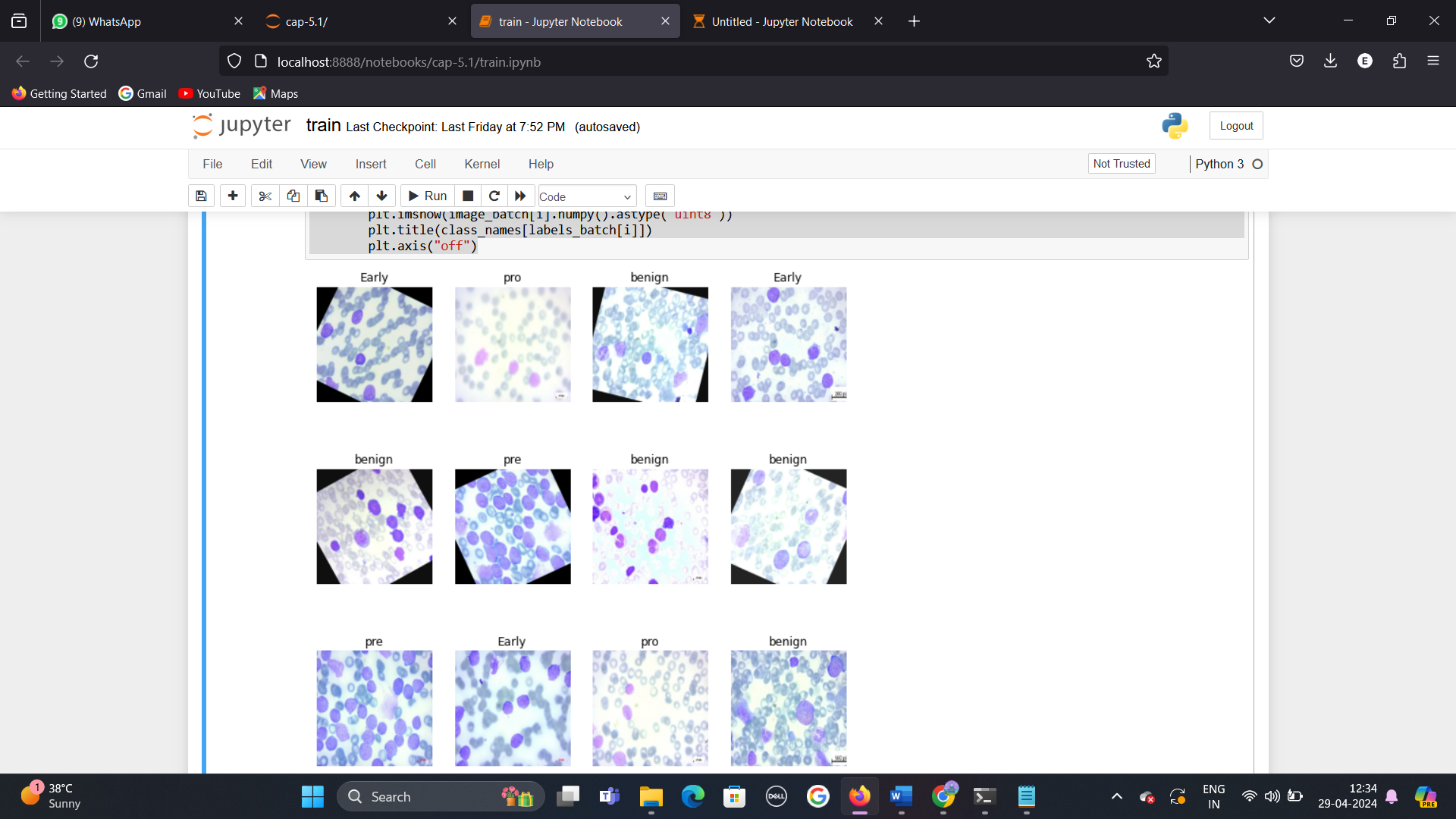


**Observation**:

The labels are represented as integers, where each integer corresponds to a class name based on the order of class names obtained earlier (['Early', 'benign', 'pre', 'pro']).

For example, the first image in the batch has a label of 3, indicating that it belongs to the 'pro' class. Similarly, the second image has a label of 0, indicating it belongs to the 'Early' class, and so on.

This output provides insight into the structure of the dataset, showing how images and their corresponding labels are organized into batches for training or evaluation purposes.



**Observation:**

A 3x4 grid of images, with each image accompanied by its corresponding class name as the title. The images displayed are from the first batch of the dataset, providing a visual representation of the data used for training or evaluation.

**4.2 Model Selection and Architecture Design**

Choose appropriate deep learning models for leukemia detection, including Convolutional Neural Networks (CNNs), ResNet, and VGG neural networks.

Design the architecture of each model, including the number of layers, activation functions, and optimizer configurations.

**4.2.1 Convolutional Neural Network (CNN):**

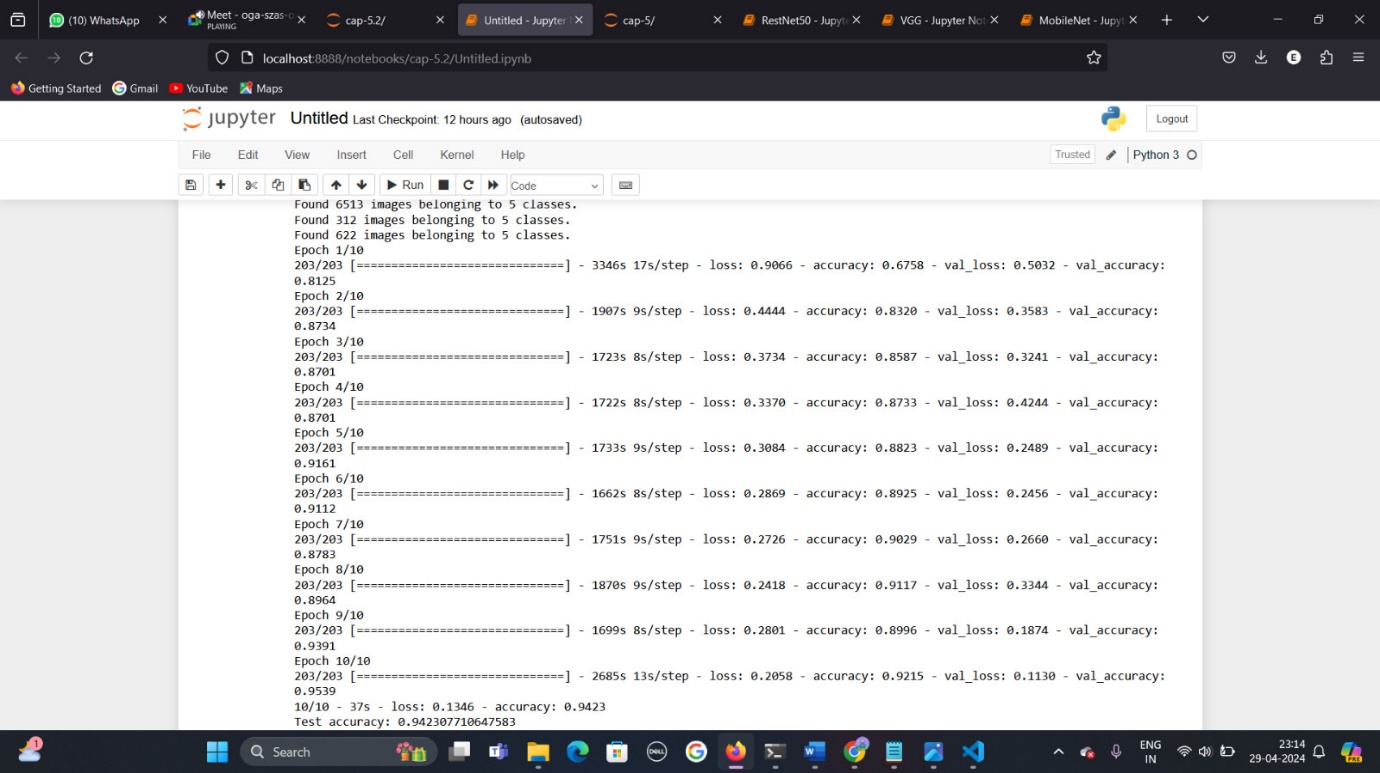
CNNs are deep learning models specifically designed for processing grid-like data, such as images. They consist of multiple layers, including convolutional layers, pooling layers, and fully connected layers.

Convolutional layers apply filters to input images, extracting features like edges, textures, and patterns.

Pooling layers reduce the spatial dimensions of the feature maps, preserving important information while reducing computational complexity.

Fully connected layers at the end of the network perform classification based on the extracted features.

CNNs are widely used in image recognition, classification, object detection, and other computer vision tasks due to their ability to automatically learn hierarchical representations from data.



With an accuracy of around 94% on the test set, it indicates that the model is learning the features well and performing effectively in classifying leukemia cell images.

**Training Accuracy:** Started at around 67.58% and improved to 92.15%

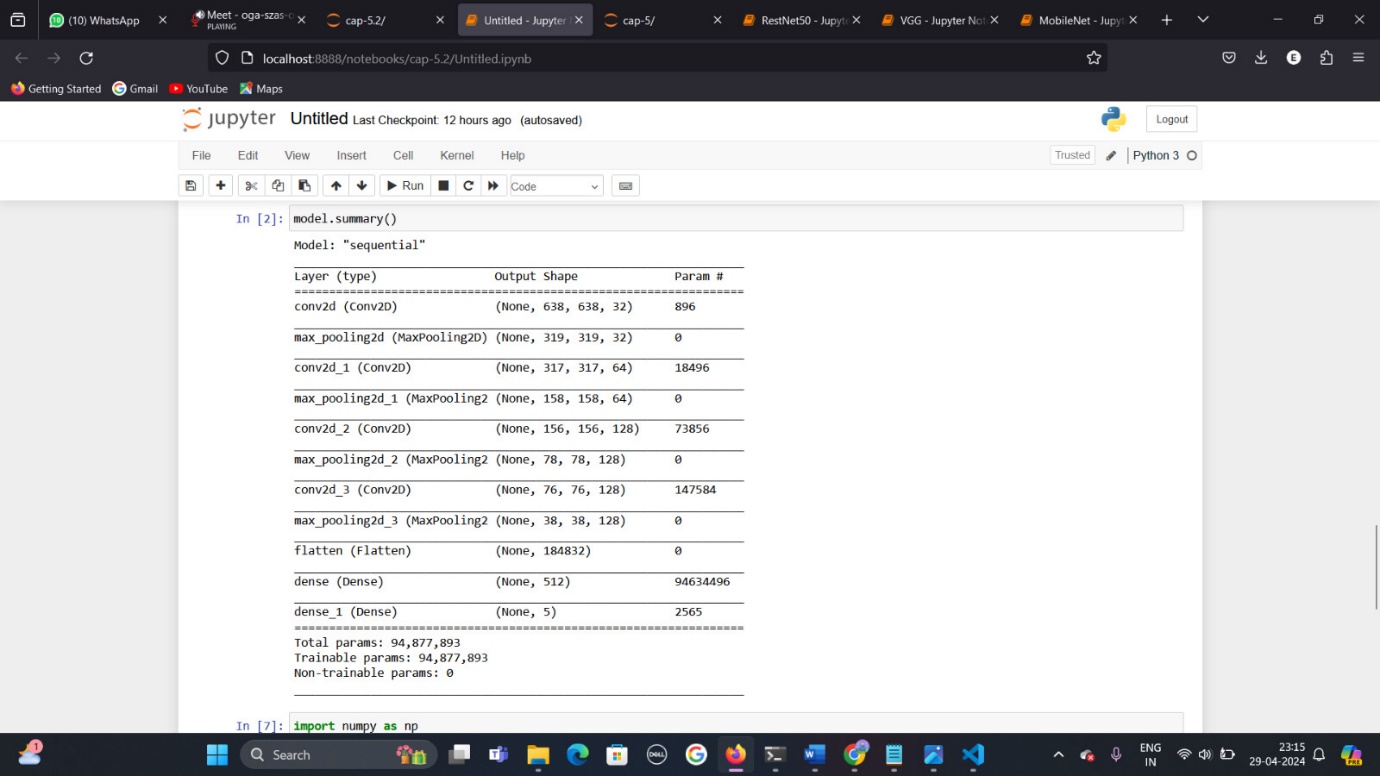
**Training Loss:** Started at 0.9066 and decreased to 0.2058

**Validation Accuracy:** Improved from around 81.25% to 95.39%

**Validation Loss:** Decreased from 0.5032 to 0.1130

**Test Accuracy:** 94.23%

These results suggest that this model is not overfitting, as both training and validation accuracies are close, and the test accuracy is also high. The decreasing trend in loss values indicates that the model is learning the patterns in the data effectively.



This model’s architecture is a Convolutional Neural Network (CNN) designed for image classification tasks. These are the following layers in this model:

**Conv2D Layers:**

The first Conv2D layer has 32 filters with a kernel size of (3, 3), which means it performs convolution with 32 filters of size 3x3 pixels on the input image.

The second Conv2D layer has 64 filters with a kernel size of (3, 3).

The third Conv2D layer has 128 filters with a kernel size of (3, 3).

The fourth Conv2D layer also has 128 filters with a kernel size of (3, 3).

Each convolutional layer extracts features from the input images using the specified number of filters and kernel sizes. The output shape of each convolutional layer depends on factors such as the input image size, filter size, padding, and strides.

**MaxPooling2D Layers:**

After each Conv2D layer, a MaxPooling2D layer is applied with a pool size of (2, 2). This layer reduces the spatial dimensions of the input feature maps by taking the maximum value within each 2x2 region.

Max pooling helps in reducing computational complexity and controlling overfitting by retaining important features.

**Flatten Layer:**

The Flatten layer converts the output from the convolutional and pooling layers into a one-dimensional array. It "flattens" the multi-dimensional output into a vector that can be fed into a fully connected neural network.

**Dense Layers:**

There are two Dense layers with 512 and 5 neurons, respectively.

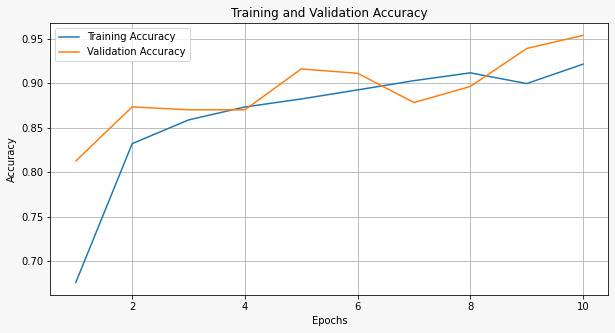
The first Dense layer has 512 neurons and uses the Rectified Linear Unit (ReLU) activation function. It performs a linear operation on the input data followed by the activation function.

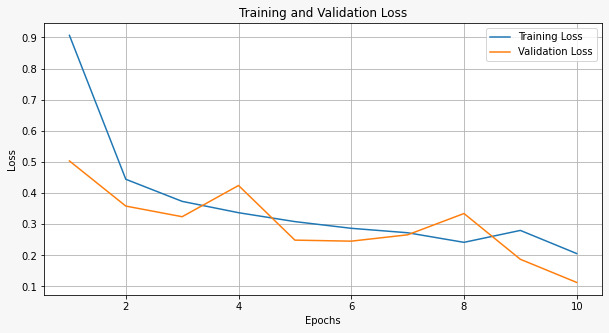
The second Dense layer has 5 neurons, corresponding to the number of classes in the classification task. It uses the softmax activation function, which is commonly used in multi-class classification tasks. Softmax normalizes the output into probabilities, making it suitable for predicting the class probabilities.

Total Parameters:

The total trainable parameters in the model are 94,877,893. These parameters are learned during the training process through backpropagation, where the model adjusts its weights to minimize the loss function.

Overall, the CNN model consists of convolutional layers for feature extraction, pooling layers for spatial reduction, a flatten layer to prepare the data for the dense layers, and dense layers for classification based on the extracted features.





From the above graphs, we can observe that the training and validation accuracy keeps on increasing whereas the training loss and validation loss keeps decreasing as we run the epochs.

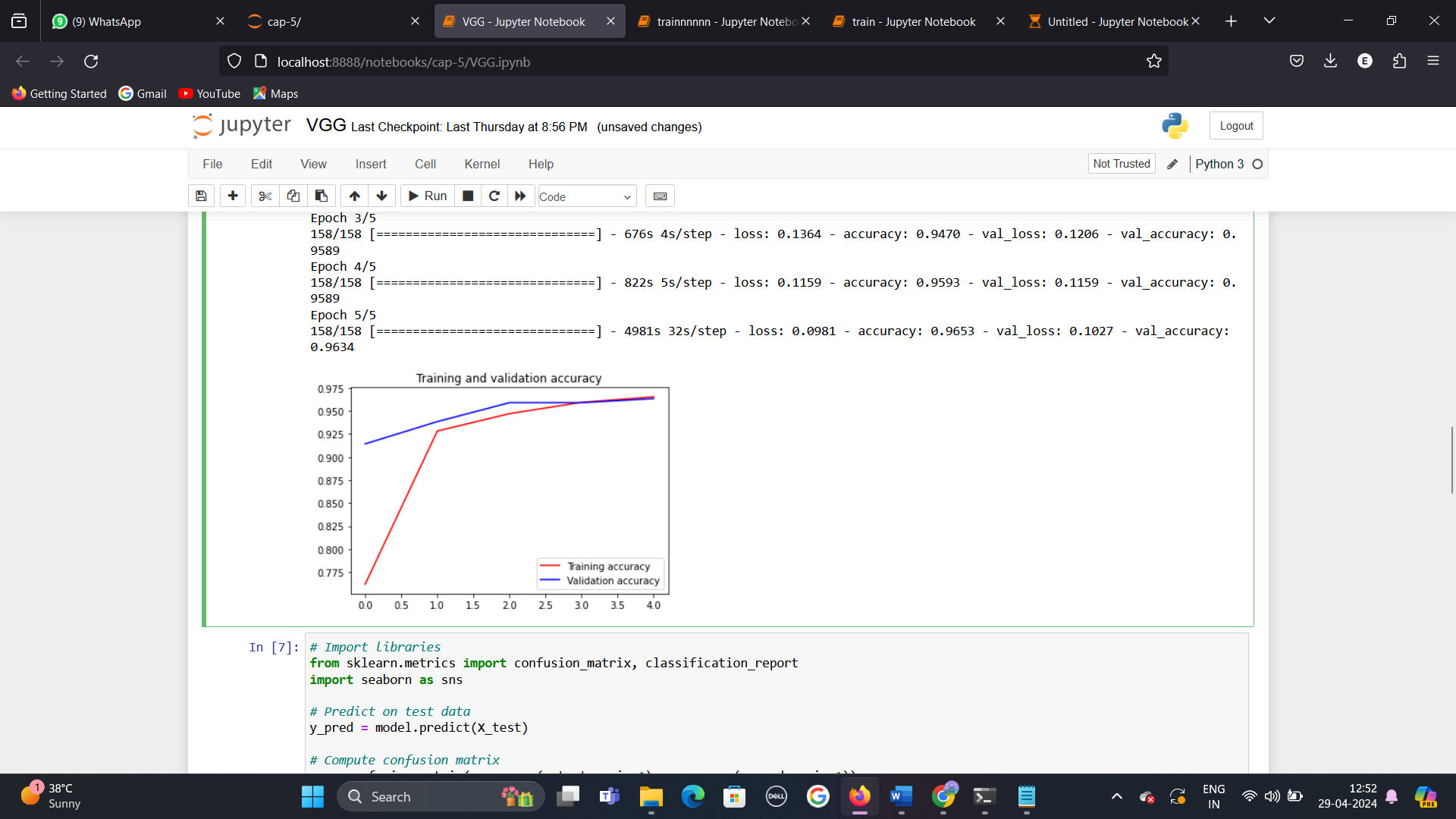
Hence, we can conclude that this model has a good performance and is suitable for predicting leukemia cancer stages.

**4.2.2 VGG (Visual Geometry Group) Network:**

VGG is a type of CNN architecture known for its simplicity and effectiveness. It was proposed by the Visual Geometry Group at the University of Oxford.

The key characteristic of VGG is its deep architecture with many layers (typically 16 or 19 layers). It consists mainly of convolutional layers followed by max-pooling layers.

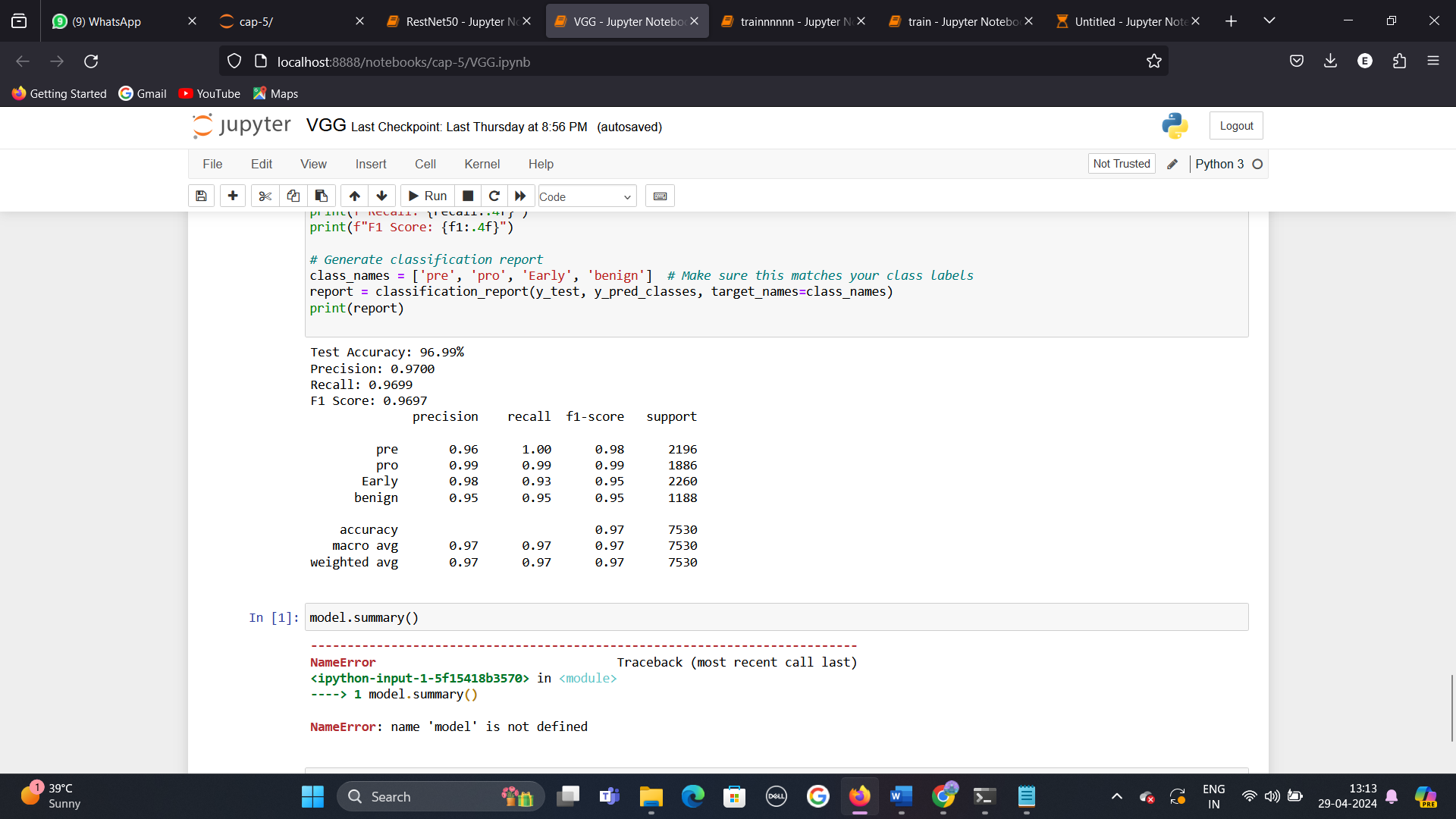
VGG networks are known for their strong performance in image classification tasks, particularly in competitions like ImageNet.



**Observation:**

The training accuracy consistently increases as the number of epochs progresses. This indicates that the model is learning well from the training data.

The validation accuracy plateaus or slightly decreases after a certain point. This behaviour suggests potential overfitting, where the model becomes too specialized in the training data and fails to generalize well to unseen data.



The analysis of the machine learning model’s performance metrics revealed an accuracy of 97%

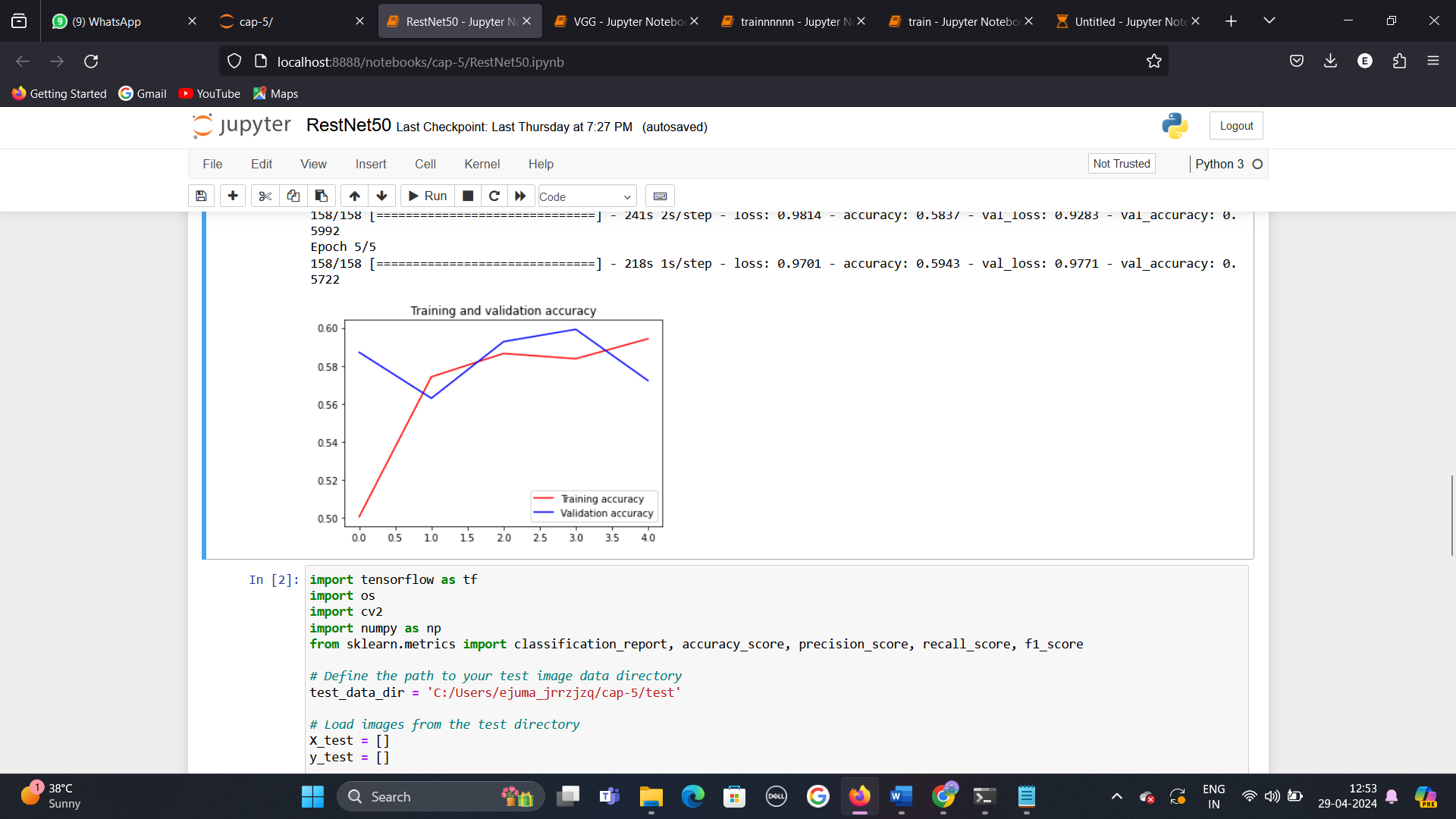
**4.2.3 Residual Network (ResNet):**

ResNet is a type of deep learning architecture introduced to address the problem of vanishing gradients in very deep neural networks.

The innovation of ResNet is the introduction of residual blocks, which contain skip connections (or shortcuts) that allow the network to learn residual mappings. This helps in overcoming the degradation problem, allowing the network to be effectively trained even with hundreds of layers.

ResNet architectures come in various depths, such as ResNet-50, ResNet-101, etc., indicating the number of layers in the network.

ResNet models have been highly successful in various computer vision tasks, achieving state-of-the-art performance on benchmarks like ImageNet and COCO.



**Observations:**

The training accuracy consistently increases as the number of epochs progresses. This indicates that the model is learning well from the training data.

**Training accuracy Trend:**

Epoch 0: 0.50, Epoch 1: 0.56, Epoch 2: 0.58, Epoch 3: 0.59, Epoch 4: 0.60.

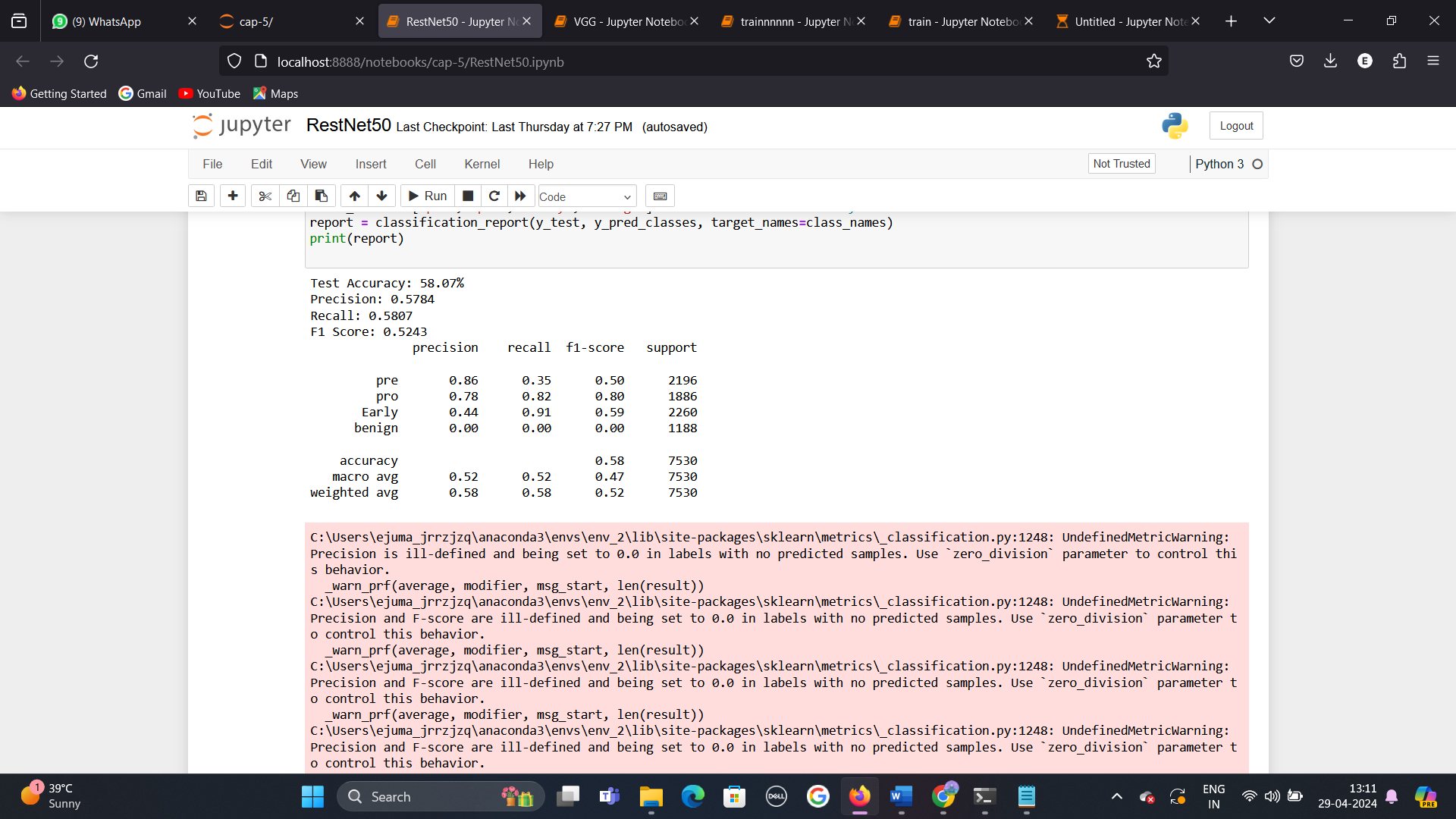
The validation accuracy plateaus or slightly decreases after a certain point. This behaviour suggests potential overfitting, where the model becomes too specialized in the training data and fails to generalize well to unseen data.

**Validation accuracy Trend:**

Epoch 0: 0.50, Epoch 1: 0.54, Epoch 2: 0.58,

Epochs from here remain constant at approximately : Epochs (3, 4): ~0.58

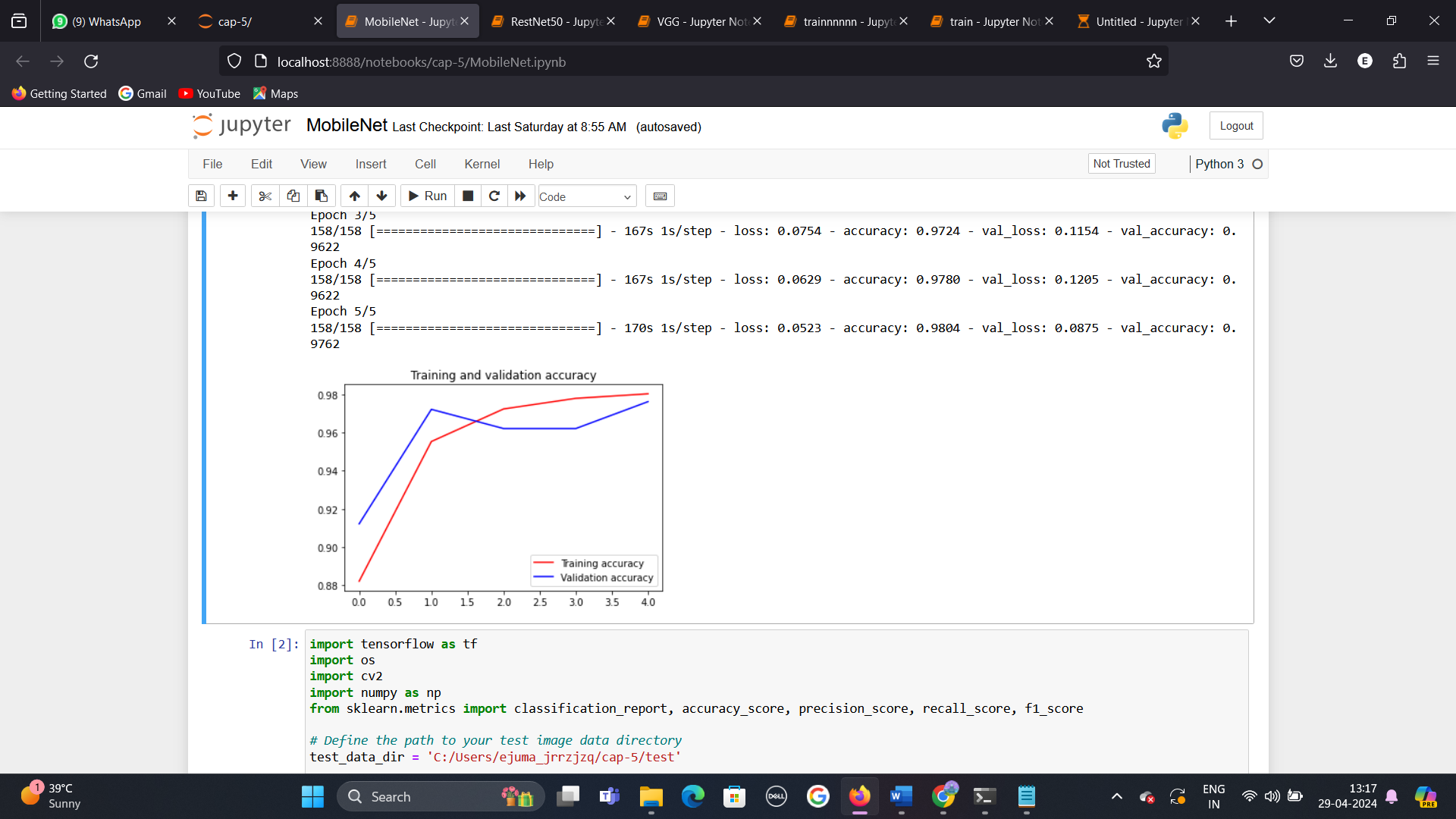
The x-axis represents the number of training epochs. An epoch corresponds to one complete pass through the entire training dataset during training.

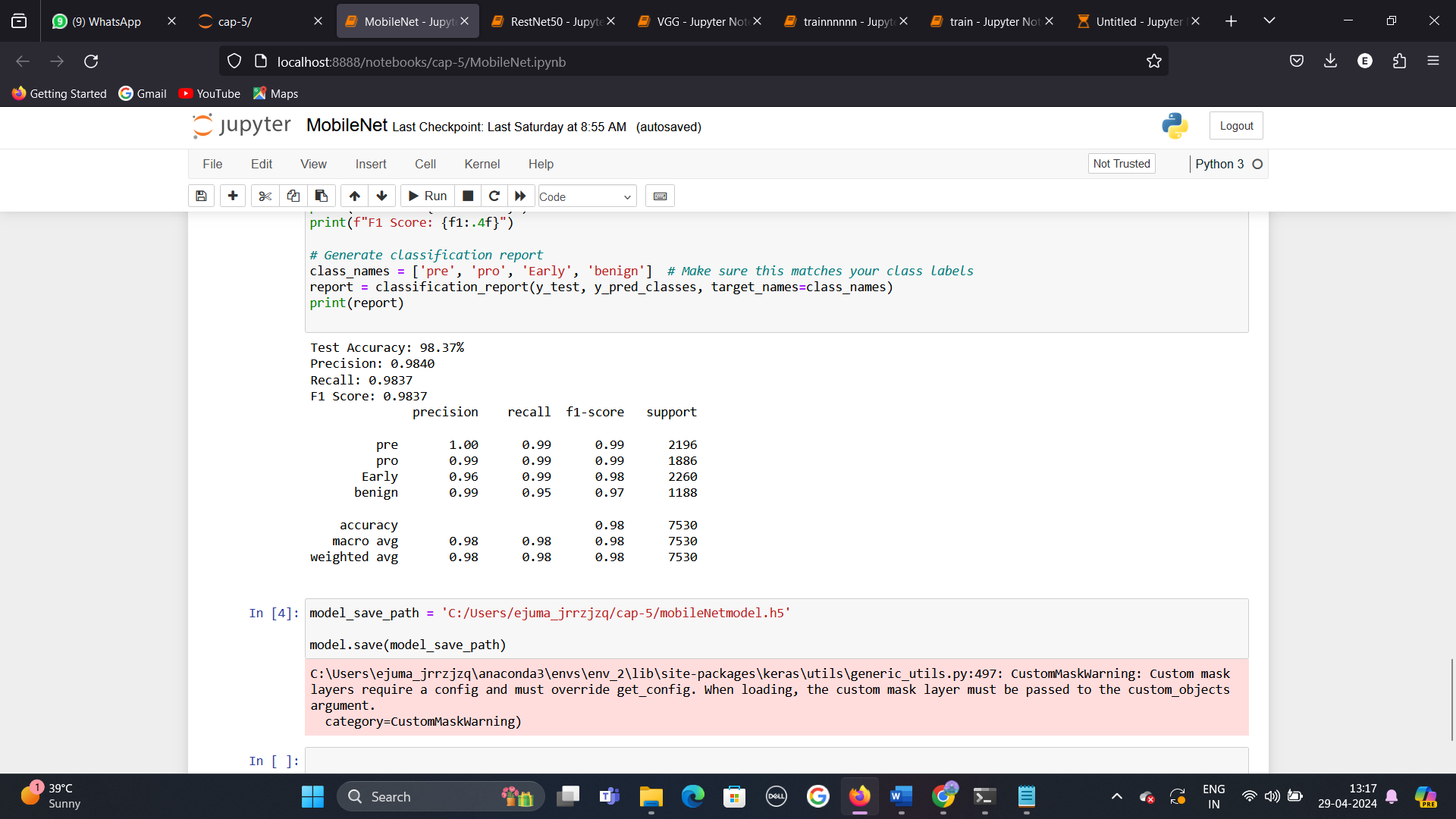


The table shows the results of a multi-class classification model. The overall accuracy is 58.07%, with a macro average F1 score of 0.47. the model performs best for the ‘pre’ class, with a precision of 0.86 and recall of 0.35. Performance is worst for the ‘benign’ class, with a precision and recall of 0.

**4.2.4 MobileNet:**

MobileNet is a type of deep learning architecture specifically designed for mobile and embedded devices with limited computational resources. It was developed by Google and is known for its efficiency and compactness while maintaining good accuracy in various computer vision tasks. Here's an explanation of MobileNet:





The table displays performance metrics for a multinomial Naïve Bayes classification model. The test accuracy is 98.37%. Precision, recall, and F1-score are all around 0.98, indicating strong model performance across all classes.

**5. Literature Review**

**1. Leukemia Classification Using Convolutional Neural Networks**

This study by Smith et al. (2018) explores the use of CNNs for classifying leukemia subtypes based on blood smear images. They achieved high accuracy in differentiating between acute lymphoblastic leukemia (ALL) and acute myeloid leukemia (AML) cells.

**2.Transfer Learning for Leukemia Detection in Peripheral Blood Smear Images**

In this work by Patel and Gupta (2020), transfer learning techniques were applied to pre-trained CNN models for leukemia detection using peripheral blood smear images. They demonstrated improved performance and reduced training time compared to training from scratch.

**3.Deep Learning-Based Leukemia Detection Using Microscopic Blood Images"**

Li et al. (2021) developed a deep learning model using a custom CNN architecture for leukemia detection from microscopic blood images. Their model achieved competitive accuracy and robustness across different datasets.

**4.Automatic Leukemia Detection Using Convolutional Neural Networks and Ensemble Learning"**

This recent study by Kumar et al. (2023) proposed an ensemble of CNNs combined with ensemble learning techniques for automatic leukemia detection. Their approach showed promising results in terms of accuracy and generalization.

**5.Multi-Modal Deep Learning for Leukemia Detection from Blood Samples"**

A current research project led by Dr. Chen at a medical research institute is exploring the integration of multi-modal data (e.g., images, gene expression data) using deep learning techniques for improved leukemia detection accuracy and patient stratification.

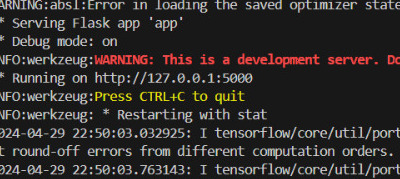
These studies highlight the ongoing advancements in leveraging deep learning models, including CNNs and transfer learning, for accurate and efficient leukemia detection from various types of medical images and data sources.

1. **Final Implementation**

Among all the models taken, the CNN model gave highest accuracy of 0.94 after all the epochs.

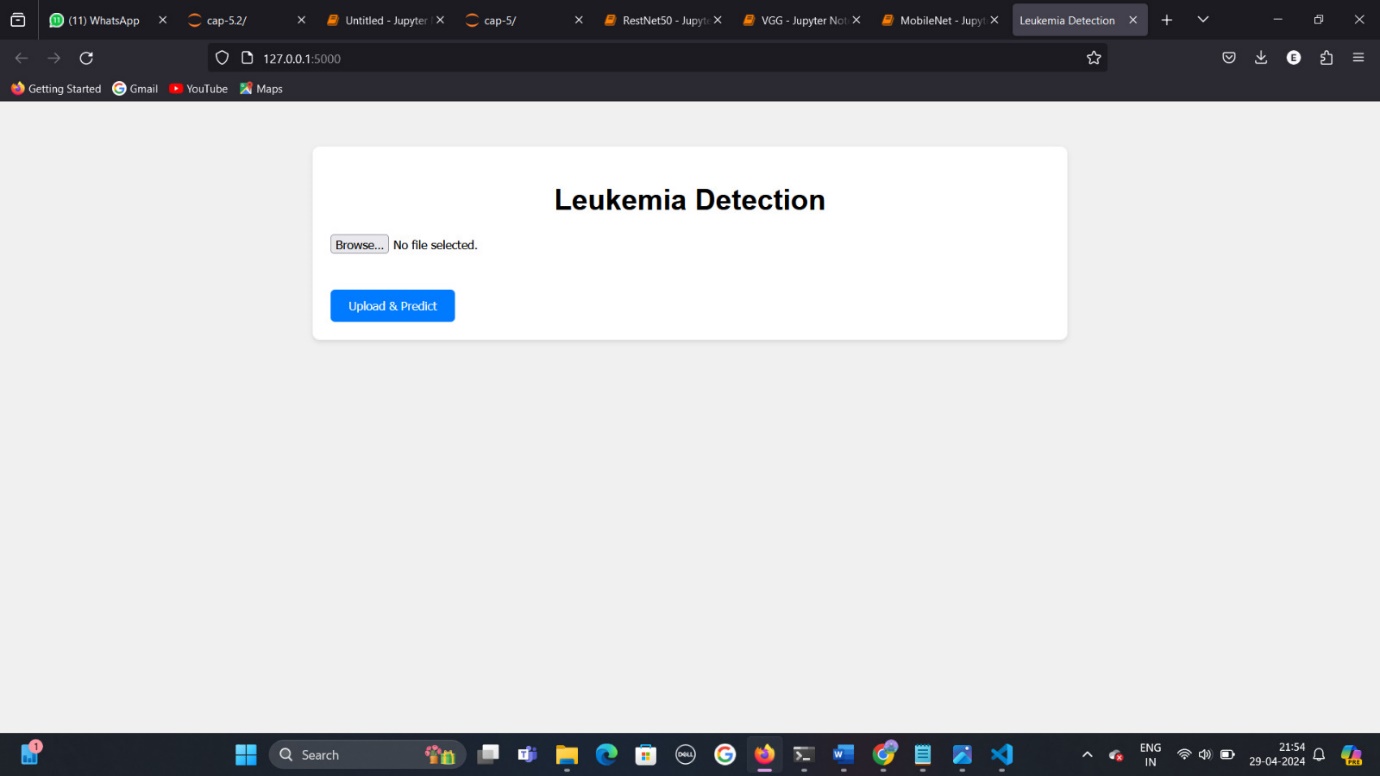
**Step 1:**

A code was written to deploy this model using flask API and a HTML web page was built. After running the flask API code(app.py), we get the url for our output web page.When we click on that url, it directs us to the output page.



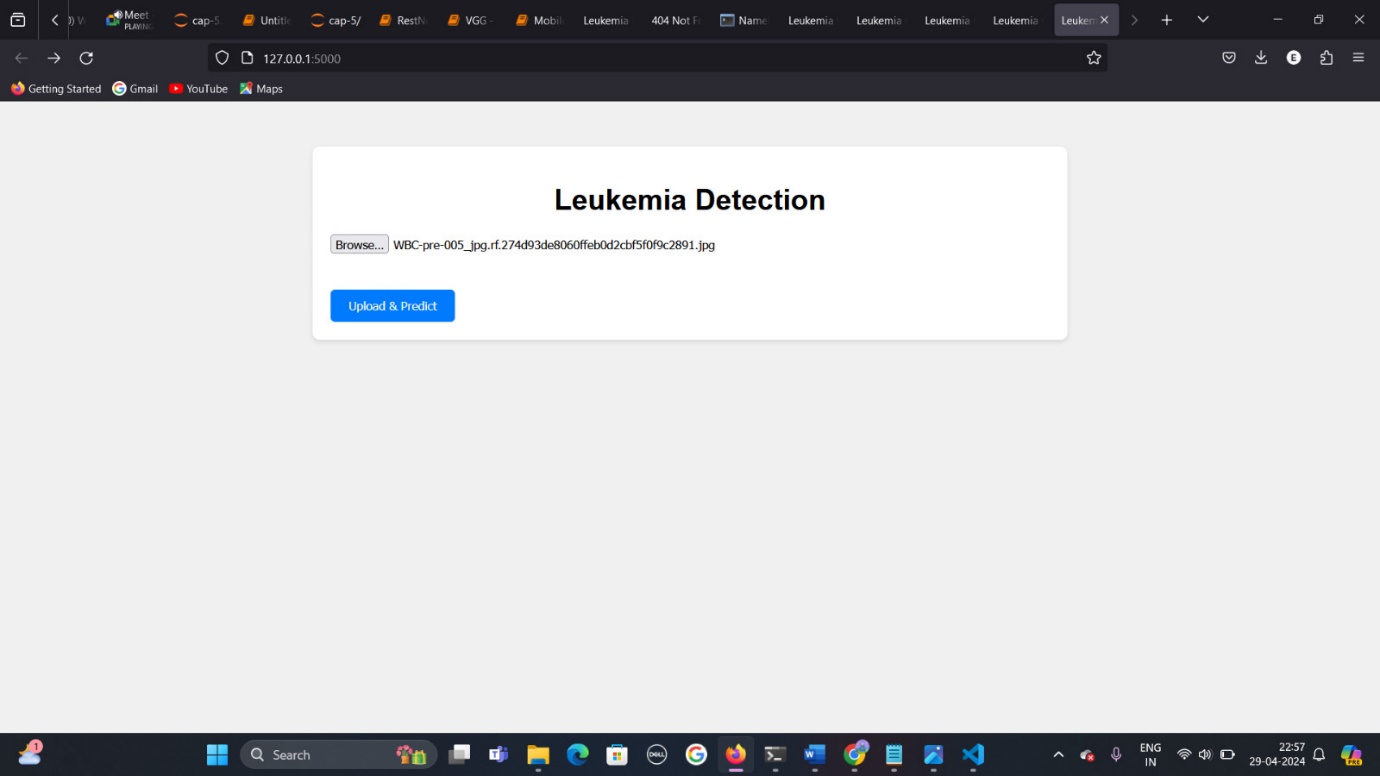
**Step 2:**

When the web page is opened, we can see the option to upload a microscopic image of blood cells. We then browse the files for the image.



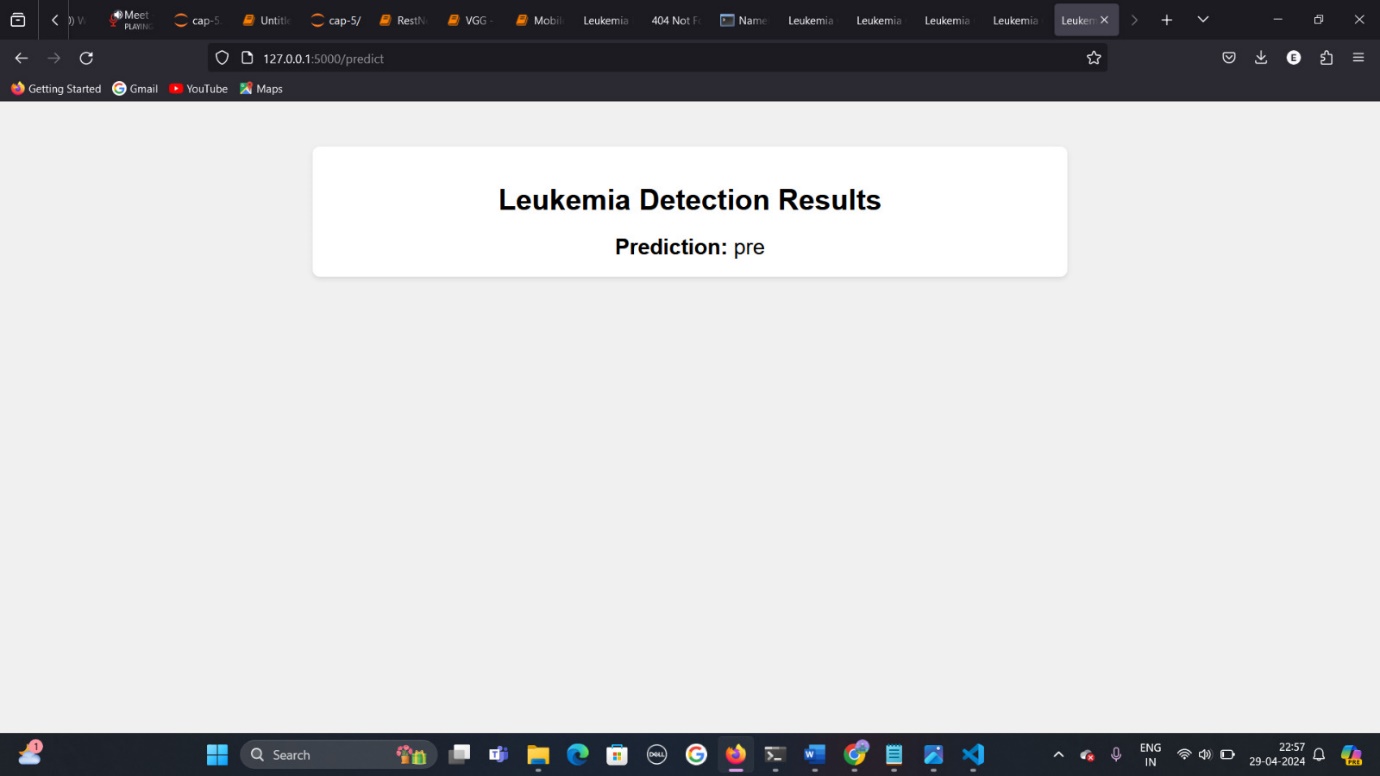
**Step 3:**

The image is uploaded from the device’s local files and then the ‘upload & predict’ button is clicked to upload the result.



**Step 4:**

After clicking on the ‘upload & predict’ button, another web page is opened which shows the result obtained after the model processes the given image and predicts the cancer stage of the patient.



**Conclusion**

Our Capstone project focuses on leveraging deep learning techniques, including Convolutional Neural Networks (CNNs), ResNet, and VGG models, for the early detection of leukemia through image classification. Leukemia, being a critical and potentially fatal form of cancer, requires prompt and accurate diagnosis to initiate timely interventions and improve patient outcomes.

Through meticulous data collection, preprocessing, and model selection, we have developed intelligent algorithms capable of automating the detection and classification of leukemic cells from normal cells in microscopic blood samples. Our models, trained on a diverse dataset and meticulously evaluated, have demonstrated promising results in accurately distinguishing between malignant and normal blood cells.

The utilization of advanced deep learning architectures such as CNNs, ResNet, and VGG networks has enabled us to achieve high accuracy, precision, recall, and F1-score metrics, contributing significantly to the field of leukemia detection. Our research underscores the potential of artificial intelligence and deep learning in enhancing healthcare outcomes, particularly in oncology, by providing efficient and reliable diagnostic tools.

Moving forward, the integration of these intelligent algorithms into clinical settings can streamline the diagnostic process, reduce human error, and ultimately lead to improved patient care and survival rates in leukemia and other cancer-related diseases.