HI NUTS!

User Guide

ABSTRACT
This guide documents various steps to installation and usage of Hi Nuts application
Team
Zhou Zhibiao (A0269368J)
Sureshkumar Sujatha (A0269371W)
Tan Jun Ming (A0269372U)
Lim Zhi Jing (A0269638J)
Pattern Recognition System (Part- Time).
July 2023

Summary

This guide provides steps for installation and usage of the app. Mac operating system is used for the preparation of the guide. Other operating system uses the same steps unless otherwise stated.

Pre-requisites

Operating System

1. Linux
2. Windows
3. Mac

Required Programs

1. Anaconda (we have tested our applica;on based on conda installa;on of python. Any standalone installa;on or other type of installa;on may require self troubleshoo;ng if found any errors)
2. Python

Install python libraries

Please use admin account for this application before installation and usage of the application. Example:

1. For windows, run the terminal as administrator
2. For linux and mac: use sudo su

```
sudo su
```

Please run following steps in the terminal

```
pip install Flask
pip install glob
pip install torch
pip install torchvision
pip install shutils
pip install Werkzeug
pip install pandas
pip install opencv-python
pip install numpy
pip install matplotlib
pip install ultralytics
pip install pycopy-xml.etree.ElementTree
pip install Pillow
pip install pathlib
pip install numpy --upgrade
```

Hi Nuts App

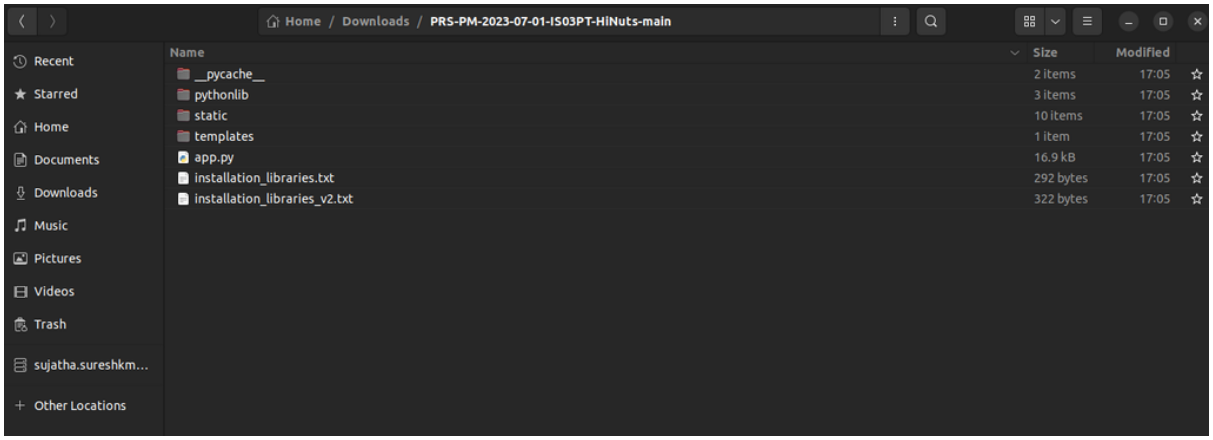
Download App

Click here to download the Hi Nuts App source.

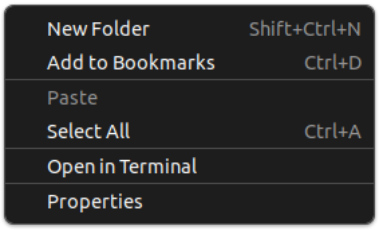
Download the source from the above mentioned source code link. Unzip the file “PRS-PM-2023-07-01-IS03PT-HiNuts-main.zip”

Start App Hi Nuts

1. From Finder(Mac), Files(Linux), File Explorer (Windows): Go into the unzipped folder. Folder contents looks below



2. Right click and choose “Open in Terminal”



2. Please use admin account for this applica;on before installa;on and usage of the applica;on. Example:
    1. For windows, run the terminal as administrator
    2. For linux and mac: use sudo su
3. For Mac/Linux systems run the following commands:

```
export FLASK_APP=app
export FLASK_DEBUG=1
flask run
```

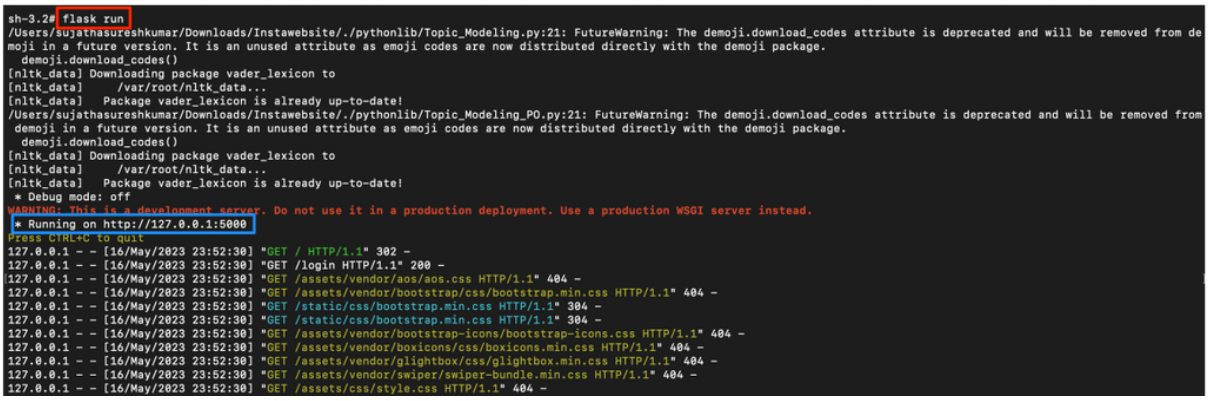
5. For Windows run the following commands:

```
set FLASK_APP=app
```

```
set FLASK_DEBUG=1
flask run
```

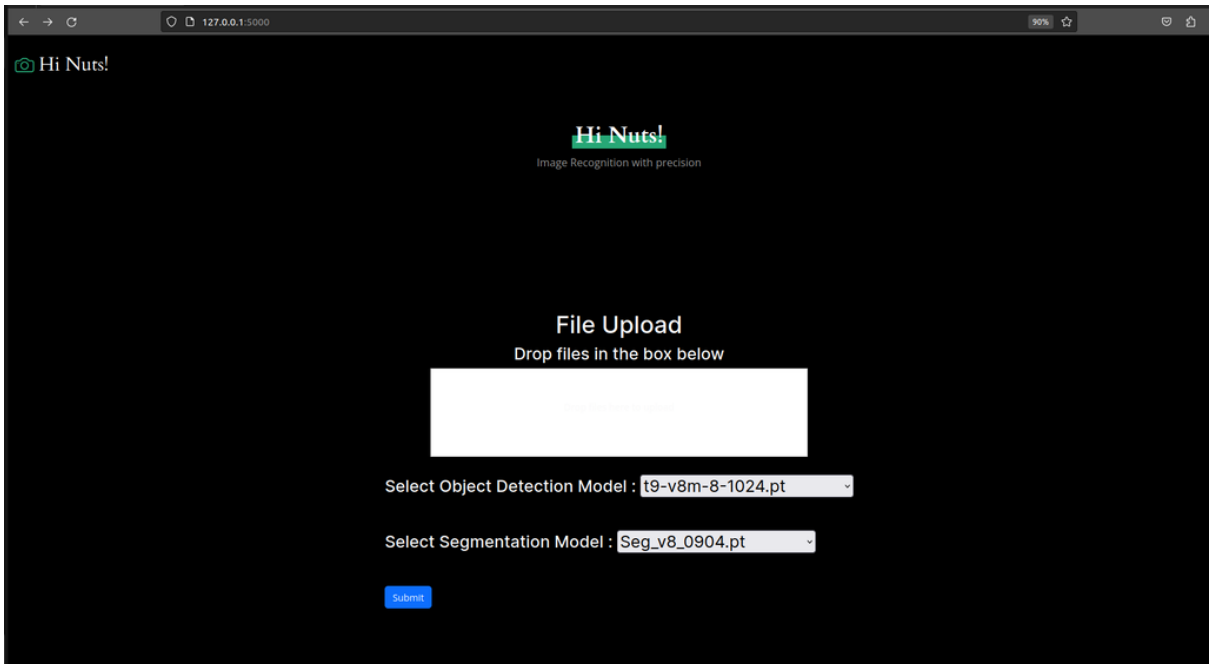
6. Identify the URL that is set by flask:

By default: http://127.0.0.1:5000/



7. Go to any web browser (Chrome/Safari/Firefox) and paste/type the URL given in flask. E.g.

Eureka!! Now you can access the Hi Nuts app.



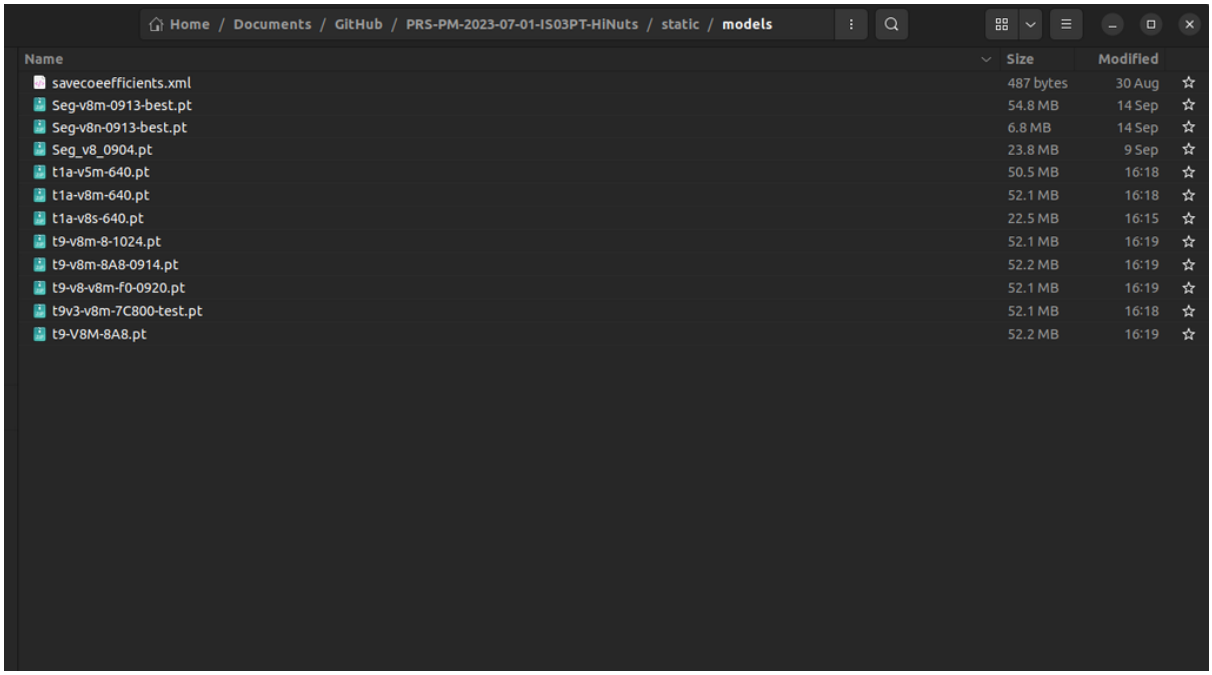
Features

Introduction Folder structure

1. Model files are stored in this folder location (**static/models**).

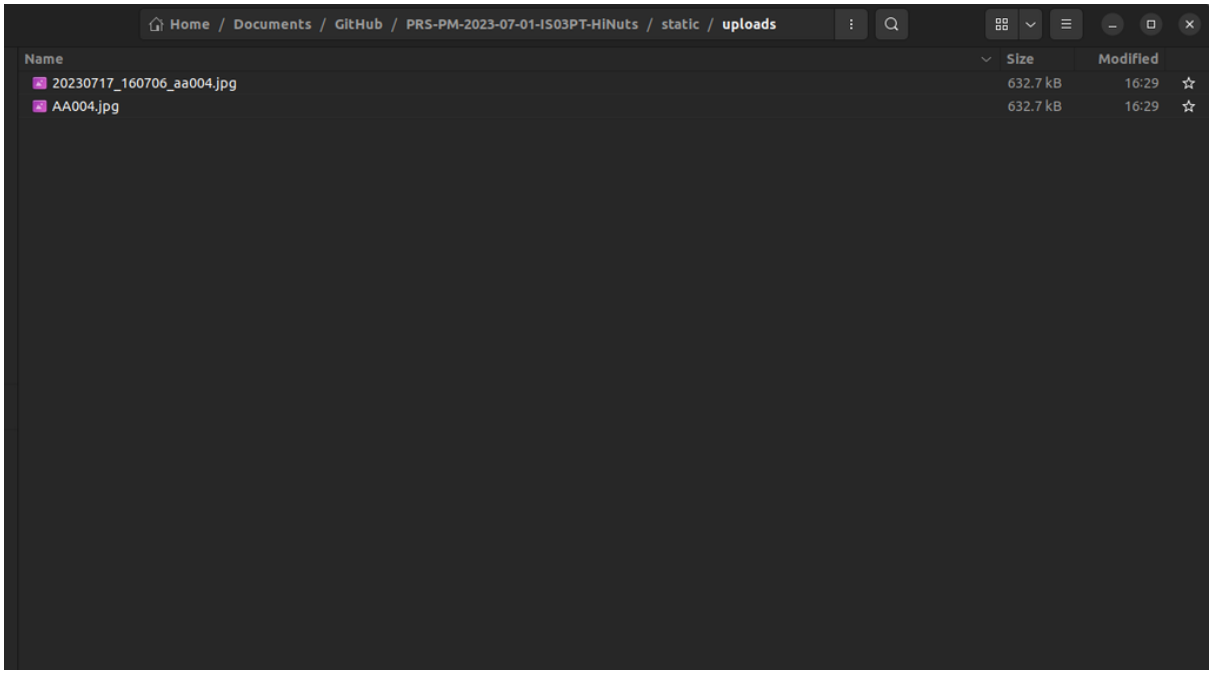
- **Usage:** model files will be provided by the data scientists shall be stored in this folder location.

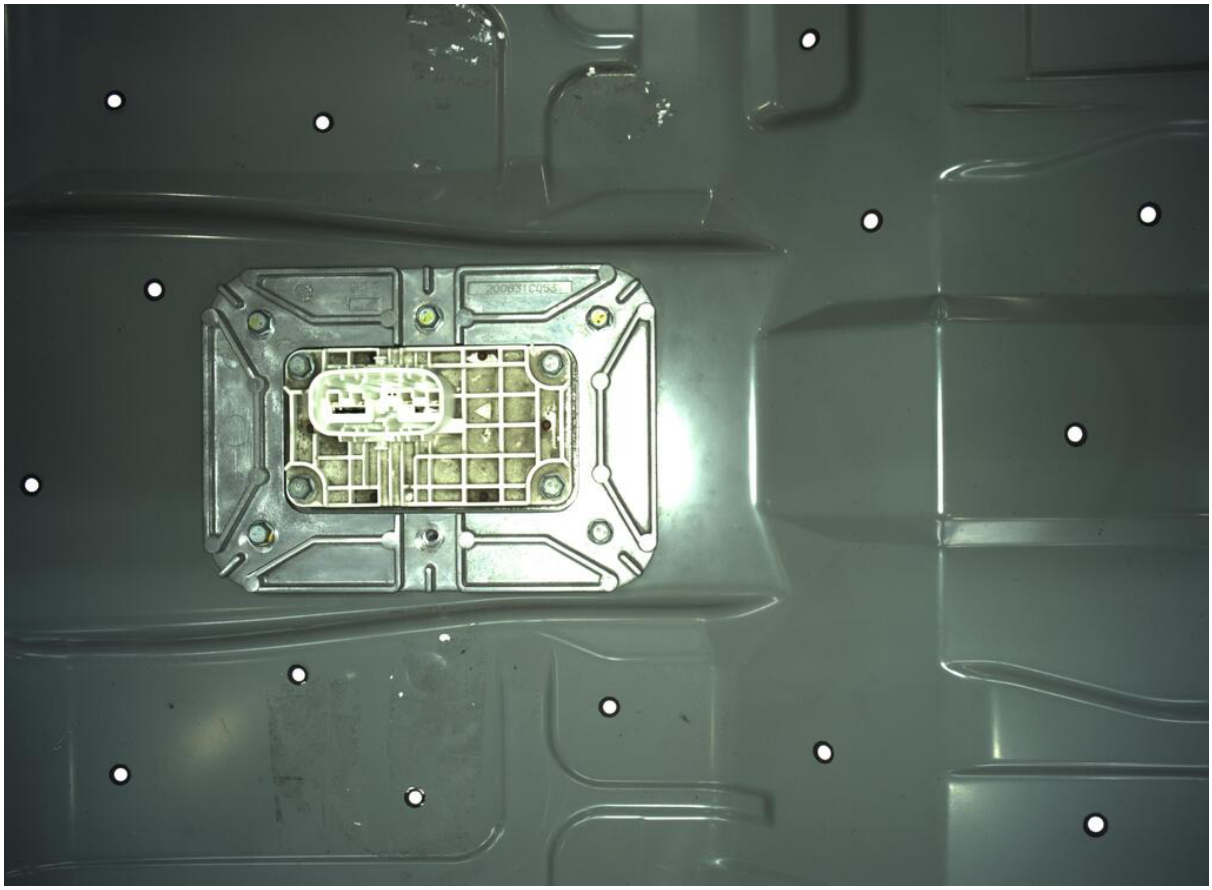
Model file type	**Format**	**Remarks**
Camera Calibration	XML	File name should be “savecoeefficients.xml“
Object Detection	pt	File name should start with “t”
Segmentation and Edge detection	pt	File name should start with “Seg“



2. Upload Images

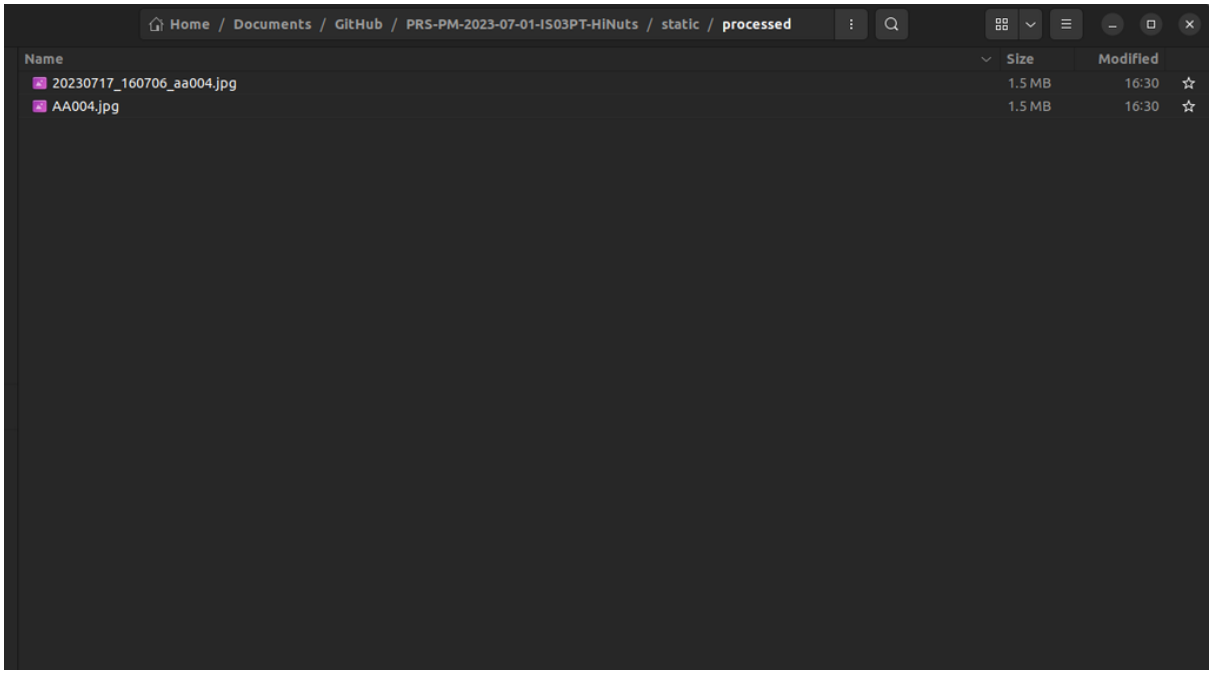
- **Usage:** When a user uploads a images it is stored in this location.
- **Folder location** : (static/uploads).

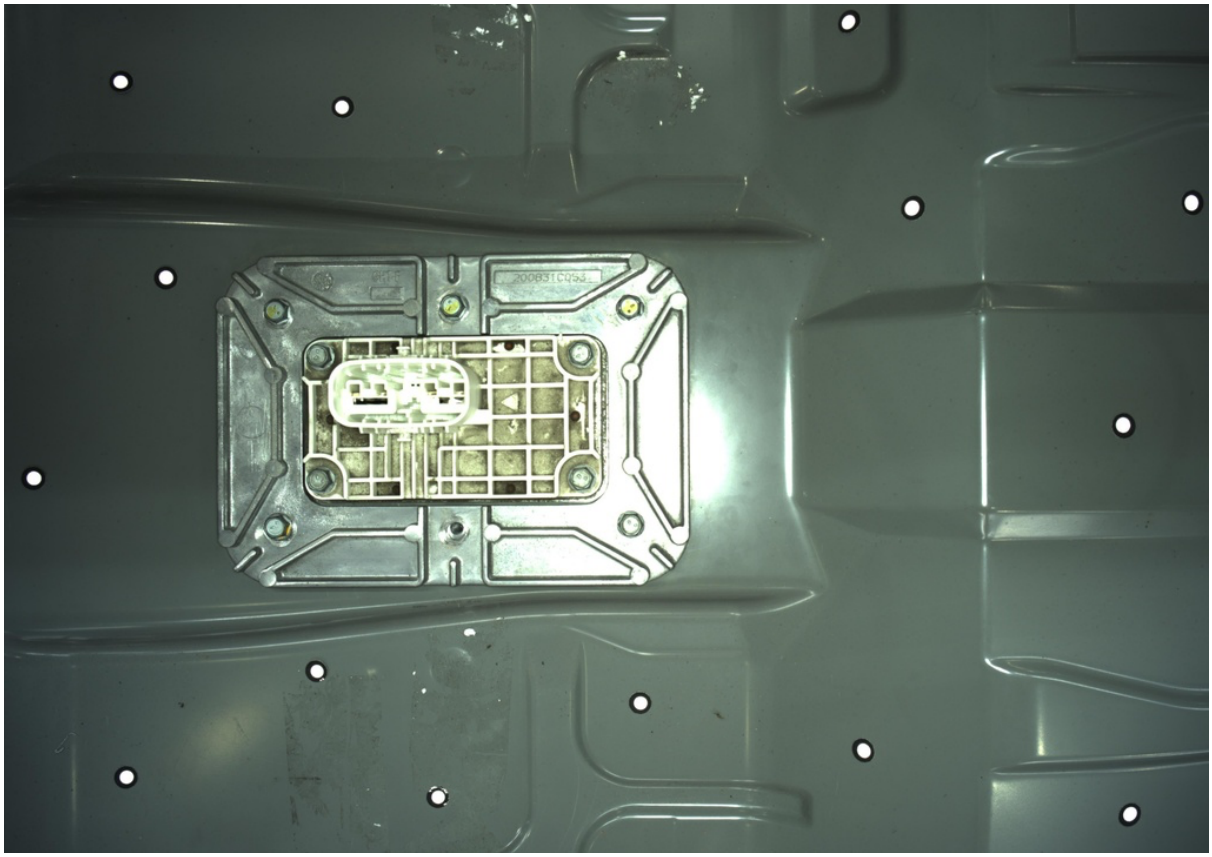




3. Processed Images

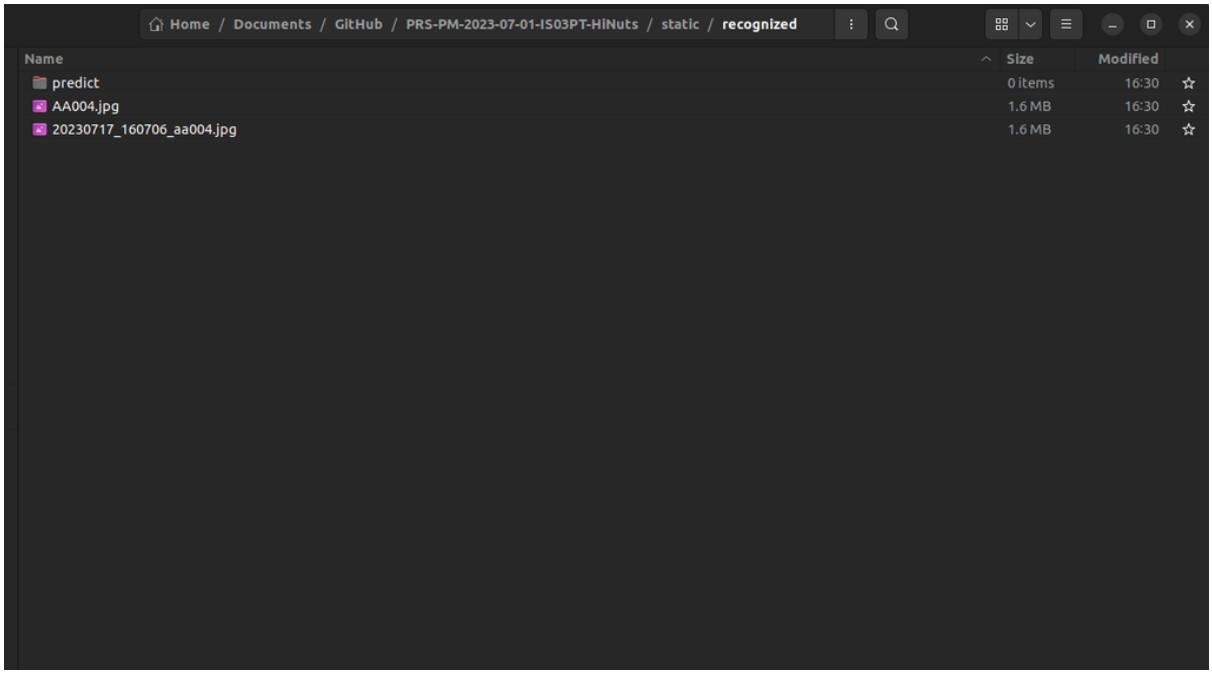
- **Usage**: Images are stored here after camera calibration
- **Folder location** : (static/processed).

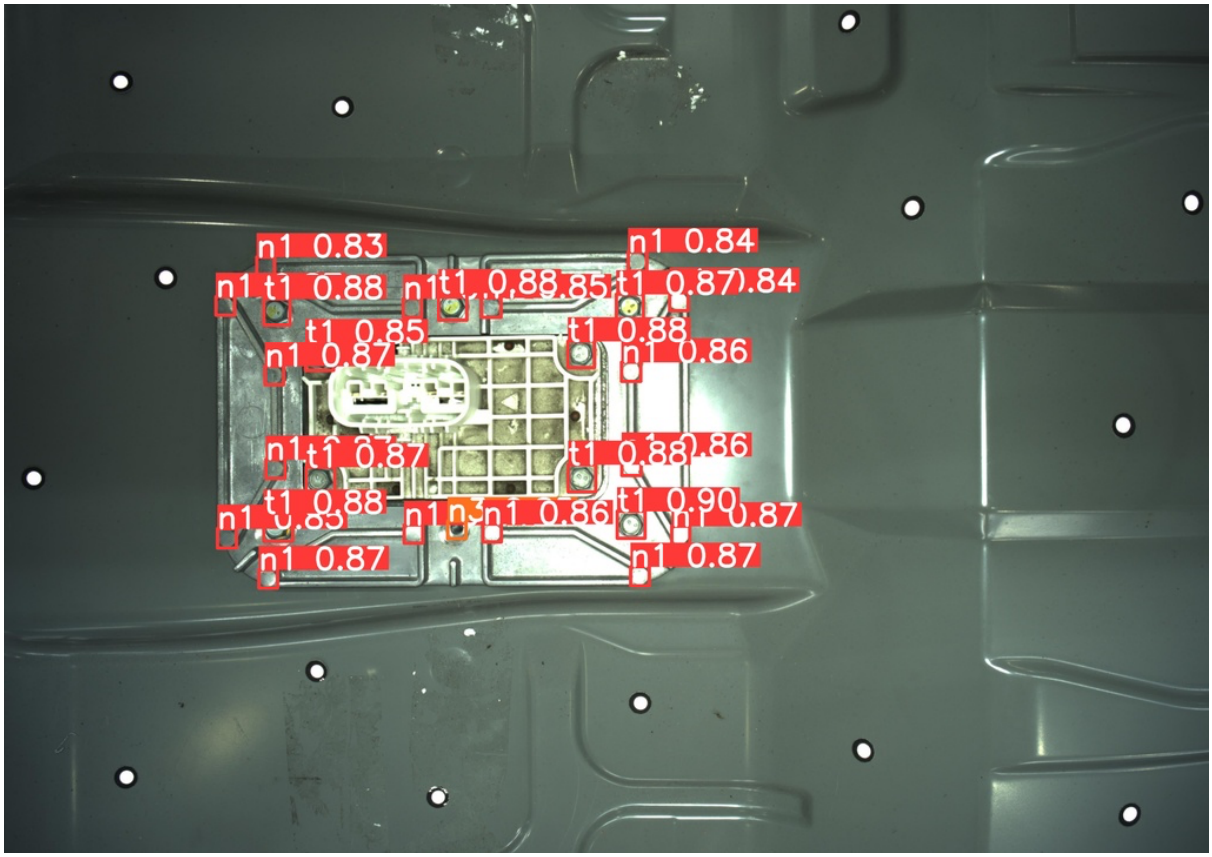




4. Recognized Images

- **Usage**: Images are stored here after object detection
- **Folder location** : (static/recognized).



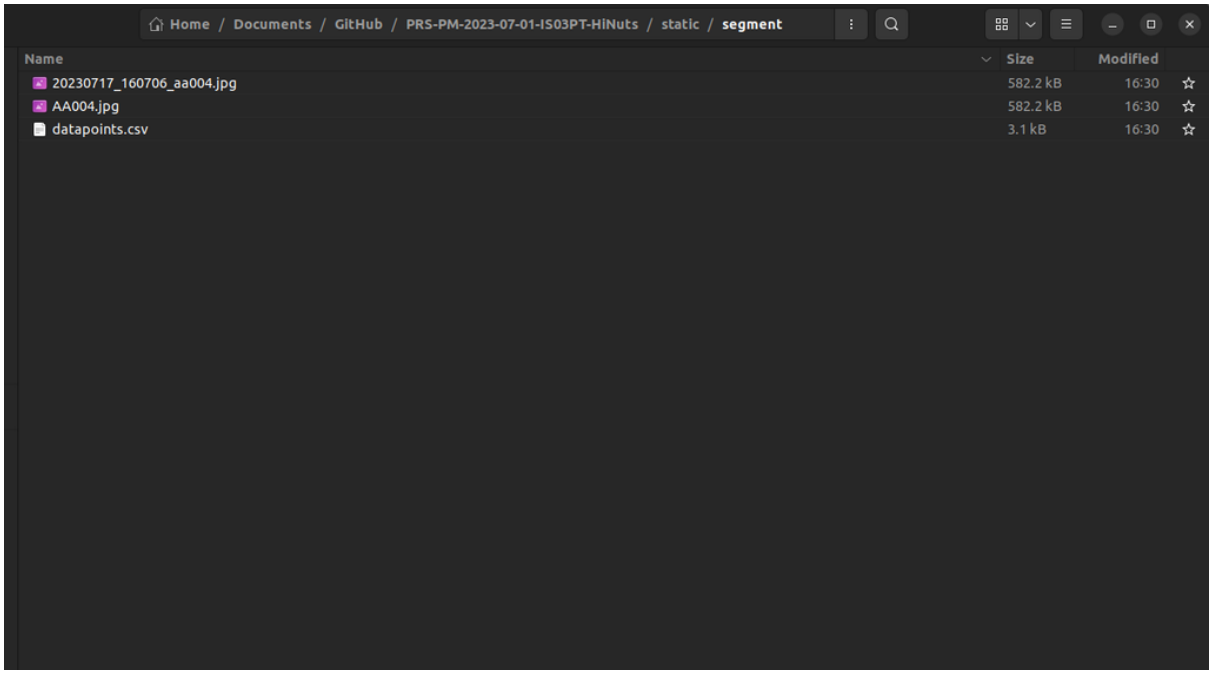


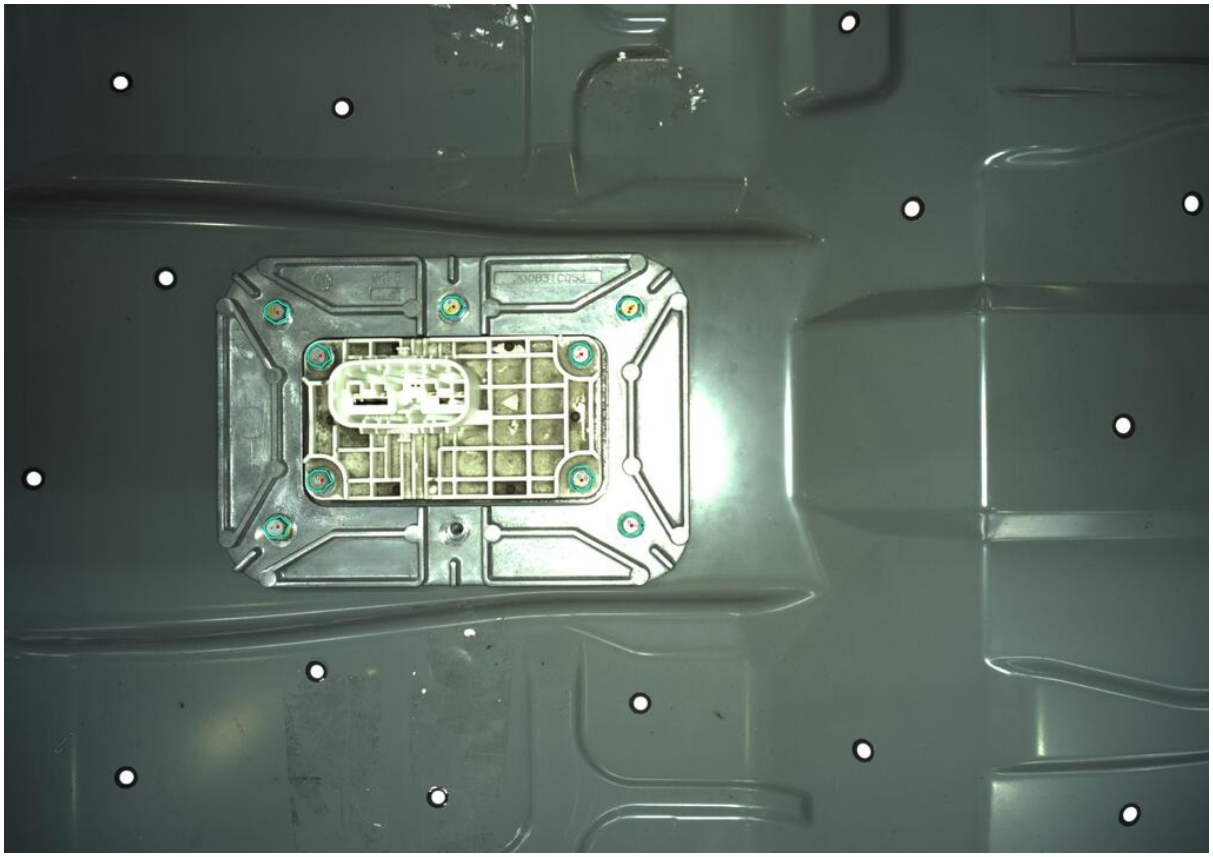
5. Segmentation Images

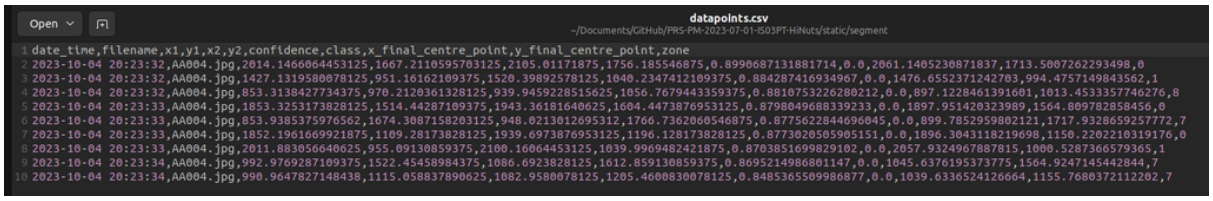
Usage: Images are stored here after segmentation and edge detection. A datapoints.csv is the final

output with the centre coordinates that is being shown in the UI are stored here.

Folder location : (static/segment).

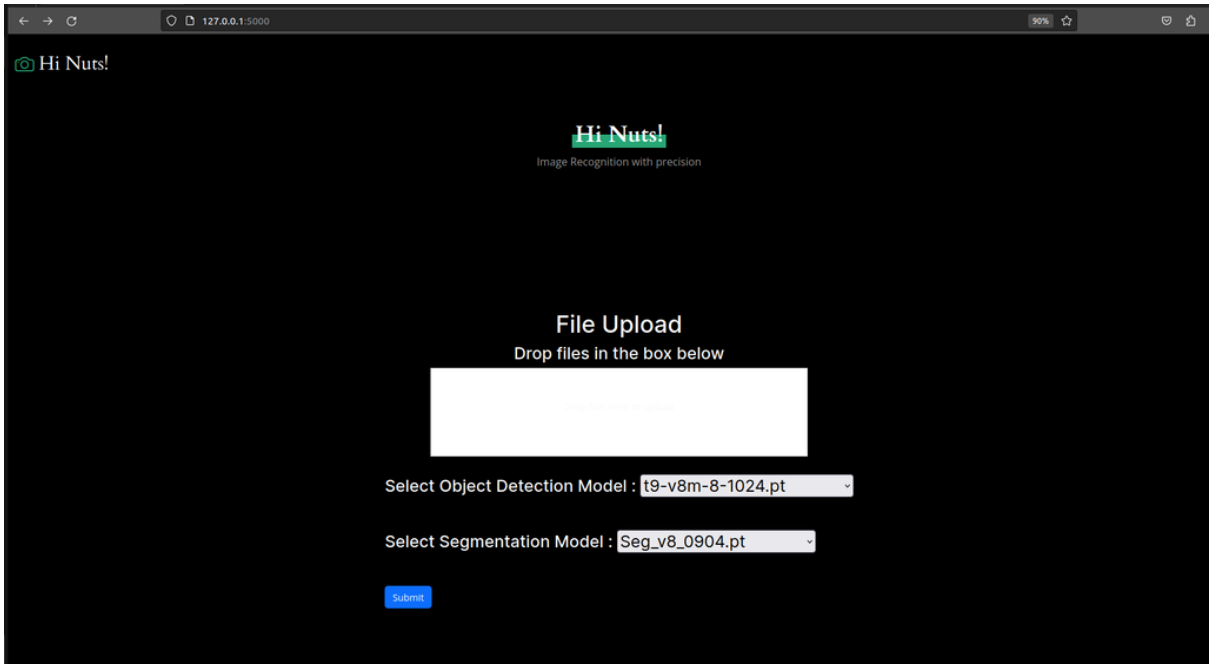




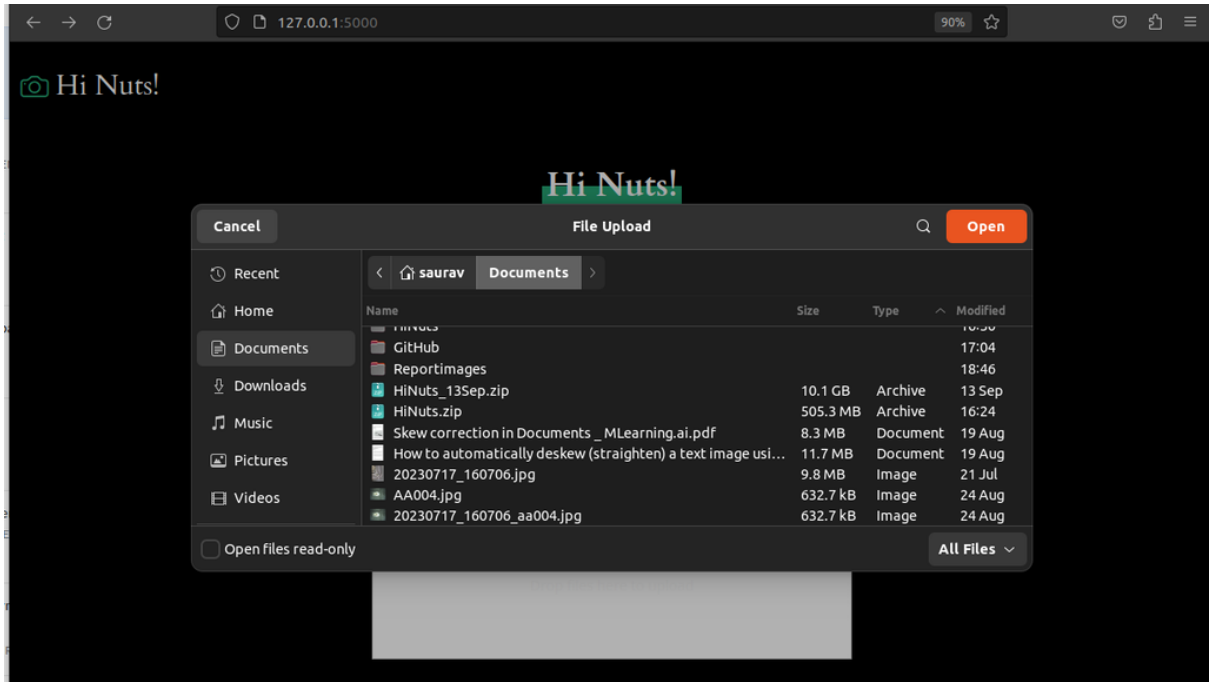


App Usage

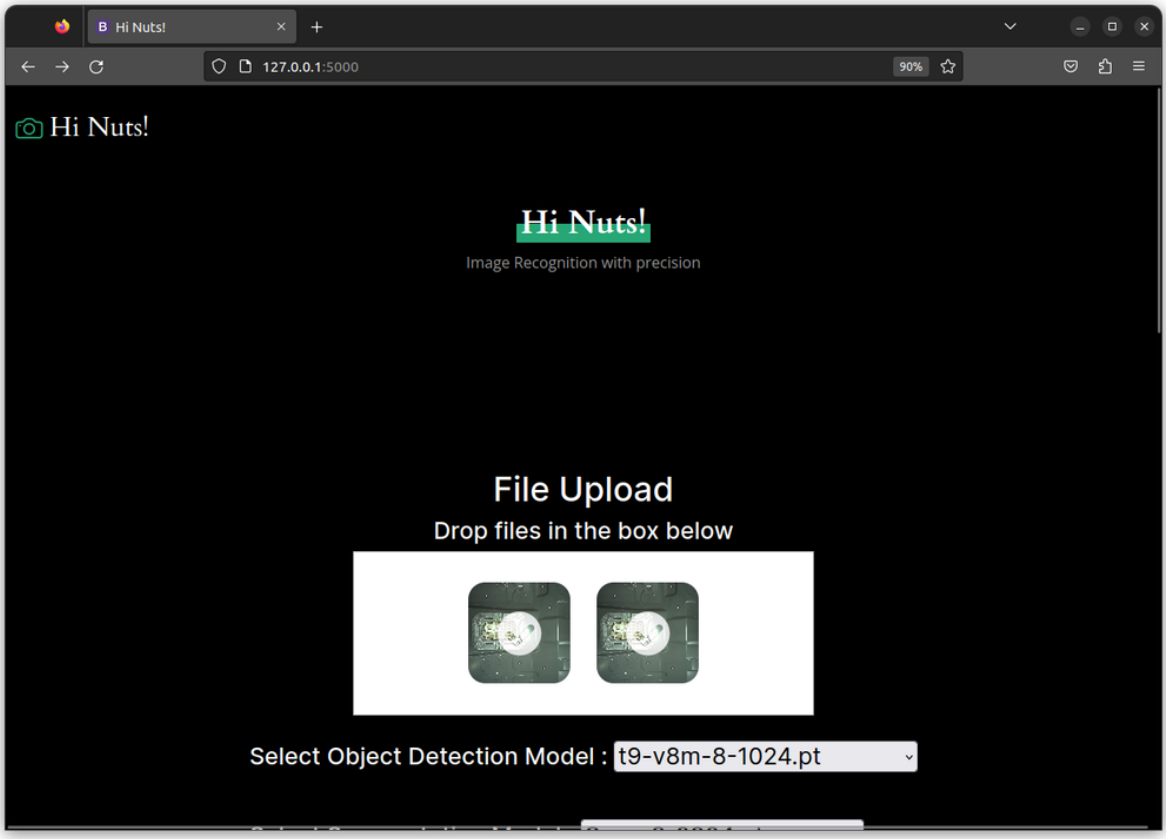
1. Go to the main page: By default: http://127.0.0.1:5000/



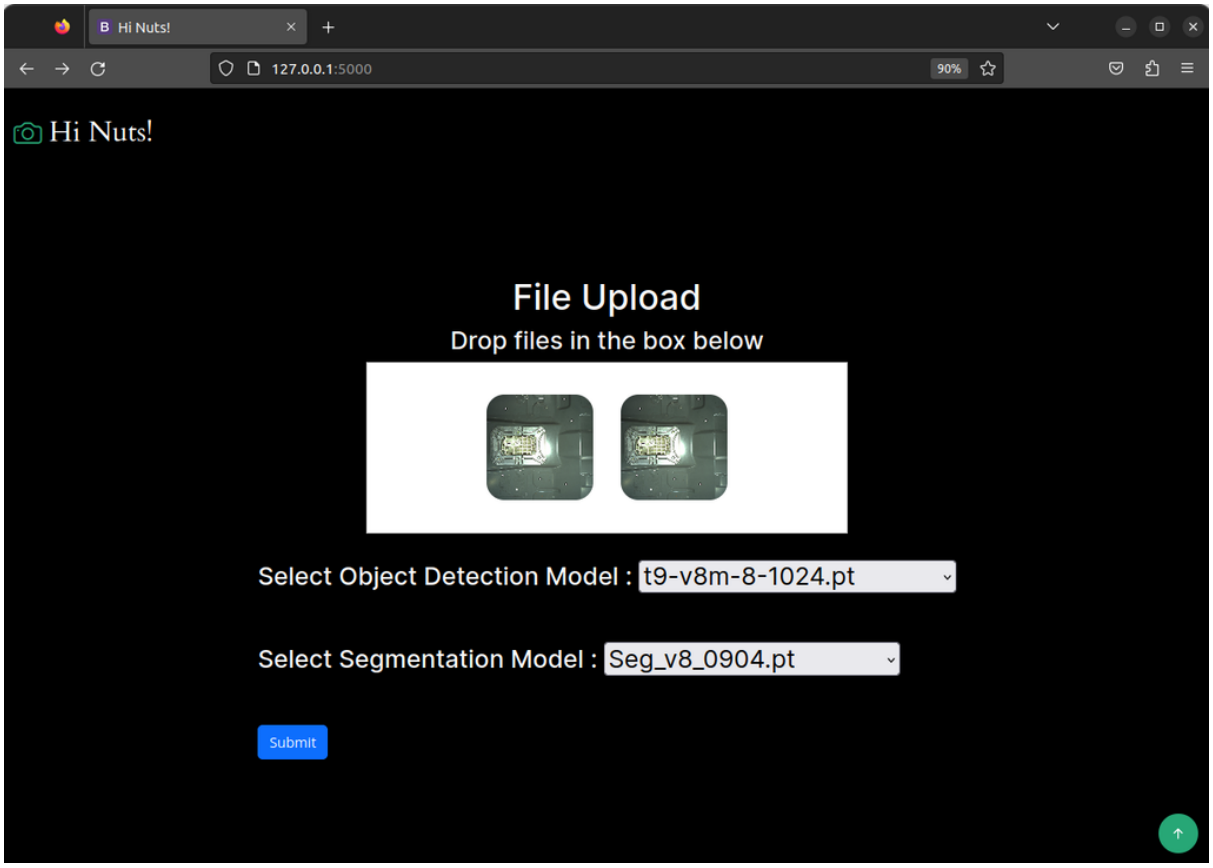
2. Click the white box and file explorer option will open up to choose the images.



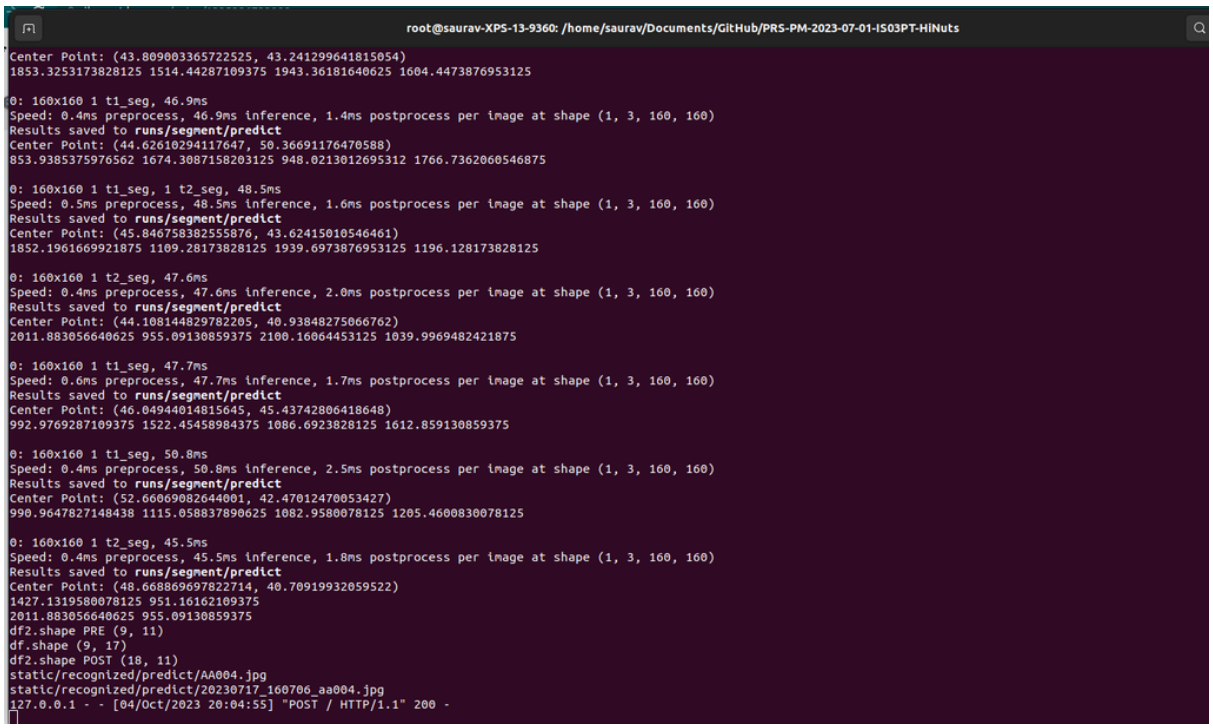
3. When the images are uploaded sucessfully and the tick mark shown as the images are ready for processing.



4. Choose the object detection model and segmentation model that is available from the dropdown and click submit.

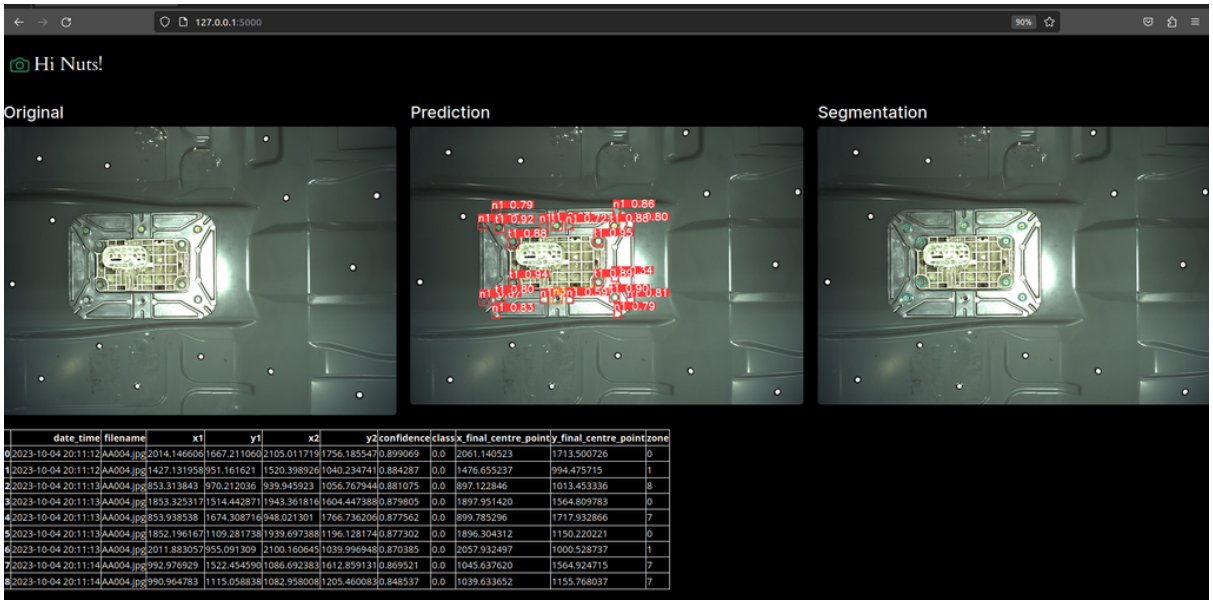


5. You will be able to monitor the process from the terminal. wait for the output “127.0.0.1 - - [04/Oct/2023 20:04:55] "POST / HTTP/1.1" 200 -“. this shows the whole process is complete.



6. UI automatically refreshes to show the output. **You may need to scroll down for the output.**

Output Type	Output Definition	Remarks
Original	This is the image that is uploaded by the user	
Prediction	This is the image after image calibration and object detection	Output after Model 1 & 2
Segmentation	This is the image after Segmentation and image detection	Output after Model 3



	date_time	filename	x1	y1	x2	y2	confidence	class	x_final_centre_point	y_final_centre_point	zone
0	2023-10-04 20:11:12	AA004.jpg	2014.146606	1667.211060	2105.011719	1756.185547	0.899069	0.0	2061.140523	1713.500726	0
1	2023-10-04 20:11:12	AA004.jpg	1427.131958	951.161621	1520.398926	1040.234741	0.884287	0.0	1476.655237	994.475715	1
2	2023-10-04 20:11:13	AA004.jpg	853.313843	970.212036	939.945923	1056.767944	0.881075	0.0	897.122846	1013.453336	8
3	2023-10-04 20:11:13	AA004.jpg	1853.325317	1514.442871	1943.361816	1604.447388	0.879805	0.0	1897.951420	1564.809783	0
4	2023-10-04 20:11:13	AA004.jpg	853.938538	1674.308716	948.021301	1766.736206	0.877562	0.0	899.785296	1717.932866	7
5	2023-10-04 20:11:13	AA004.jpg	1852.196167	1109.281738	1939.697388	1196.128174	0.877302	0.0	1896.304312	1150.220221	0
6	2023-10-04 20:11:13	AA004.jpg	2011.883057	955.091309	2100.160645	1039.996948	0.870385	0.0	2057.932497	1000.528737	1
7	2023-10-04 20:11:14	AA004.jpg	992.976929	1522.454590	1086.692383	1612.859131	0.869521	0.0	1045.637620	1564.924715	7
8	2023-10-04 20:11:14	AA004.jpg	990.964783	1115.058838	1082.958008	1205.460083	0.848537	0.0	1039.633652	1155.768037	7

6. Data Output in table format. The rows reflects number of objects detected. during the process of the detection there can be both relevant nuts and noises. we discard the noises and capture only relevant nuts.

Output Name	Output Description
Index	number of objects detected
date_time	date time when the processing is done
x1	detected object left x boundary
y1	detected object left y boundary
x2	detected object right x boundary
y2	detected object right y boundary
confidence	confidence of the object detection
class	class identified
x_final_centre_point	x coordinate centre point
y_final_centre_point	y coordinate centre point
zone	zone identified that the robot goes in sequence.

	date_time	filename	x1	y1	x2	y2	confidence	class	x_final_centre_point	y_final_centre_point	zone
0	2023-10-04 20:11:12	AA004.jpg	2014.146606	1667.211060	2105.011719	1756.185547	0.899069	0.0	2061.140523	1713.500726	0
1	2023-10-04 20:11:12	AA004.jpg	1427.131958	951.161621	1520.398926	1040.234741	0.884287	0.0	1476.655237	994.475715	1
2	2023-10-04 20:11:13	AA004.jpg	853.313843	970.212036	939.945923	1056.767944	0.881075	0.0	897.122846	1013.453336	8
3	2023-10-04 20:11:13	AA004.jpg	1853.325317	1514.442871	1943.361816	1604.447388	0.879805	0.0	1897.951420	1564.809783	0
4	2023-10-04 20:11:13	AA004.jpg	853.938538	1674.308716	948.021301	1766.736206	0.877562	0.0	899.785296	1717.932866	7
5	2023-10-04 20:11:13	AA004.jpg	1852.196167	1109.281738	1939.697388	1196.128174	0.877302	0.0	1896.304312	1150.220221	0
6	2023-10-04 20:11:13	AA004.jpg	2011.883057	955.091309	2100.160645	1039.996948	0.870385	0.0	2057.932497	1000.528737	1
7	2023-10-04 20:11:14	AA004.jpg	992.976929	1522.454590	1086.692383	1612.859131	0.869521	0.0	1045.637620	1564.924715	7
8	2023-10-04 20:11:14	AA004.jpg	990.964783	1115.058838	1082.958008	1205.460083	0.848537	0.0	1039.633652	1155.768037	7