

Table of Contents

1. Abbreviations.....	3
2. Overview.....	3
3. End User Guide.....	4
3.1 How to access the software.....	4
3.1.1 End User Prerequisites.....	4
3.1.2 Browser Compatibility.....	4
3.2 Limitations.....	5
3.3 Software Overview.....	6
3.3.1 View of Home Page.....	6
3.3.2 View of Upload Picture Page.....	8
3.3.3 View of Upload Video Page.....	9
3.3.4 View of Contact Us Page.....	10
3.4 User Interaction with User Interface.....	11
3.4.1 Real-time detection feature with Home Tab	11
3.4.2 Upload Picture feature with Upload Picture Tab	12
3.4.3 Upload Video feature with Upload Video Tab	15
3.4.4 Information with Contact Us Tab	17
3.5 Exiting the Software	18
3.6 Dealing with Special Situations.....	19
3.6.1 Model Not Loaded Initially.....	19
3.6.2 Unsupported File Format.....	20
3.6.3 No File Selected When Uploading.....	21
4. Technical Guide.....	22

1. Abbreviations

This section lists and explains abbreviations commonly used throughout the guide and in the software.

Abbreviation	Description
YOLO	You Only Look Once (A family of real-time object detection models)
UI	User Interface
JPEG/PNG	Common image file formats (Joint Photographic Experts Group / Portable Network Graphics)
MP4/MOV	Common video file formats (MPEG-4 / QuickTime format)
RM	Malaysian Ringgit (Currency used for bird nest pricing)
IP	Internet Protocol (Used for network communication)

This helps users and administrators quickly reference terms they may not be familiar with during the usage or setup of the software.

2. Overview

This software is designed to detect, classify, and estimate the value of edible bird nests in real time, based on their quality grades. The software uses object detection models (YOLOv8, YOLOv9, YOLOv10) to analyze video feeds and images, identify bird nests and classify them into Grade A, B, or C based on their quality.

The software consists of four main sections:

- Home: Real-time detection through a surveillance camera feed.
- Upload Picture: Detection and classification of edible bird nests from uploaded images.
- Upload Video: Detection and classification from uploaded videos.
- Contact Us: Provides details for contacting support or the development team.

The system can process both live video feeds and uploaded media, making it versatile for a range of use cases. Users can monitor bird nests in real time or analyze previously recorded media, while administrators can configure the software, customize models, and manage data through the backend.

3. End User Guide

3.1 How to access the software

This section describes how users can access and interact with the software from their devices.

3.1.1 End User Prerequisites

Before using the software, users need to ensure their systems meet the following requirements:

- Device: The software runs on any device that supports modern web browsers, such as laptops, desktops, or tablets.
- Operating System: No specific operating system is required, but the device should be capable of running one of the supported browsers.
- Network: A stable internet connection is necessary for real-time video processing or for uploading images and videos.
- Supported Models: The software comes with YOLOv8, YOLOv9, and YOLOv10 models for object detection. Users can also upload their own models by providing a file path.

3.1.2 Browser Compatibility

The software works on all major web browsers, including:

- Google Chrome: Version 80 or higher.
- Mozilla Firefox: Version 75 or higher.
- Safari: Version 13 or higher.
- Microsoft Edge: Version 85 or higher.

Users should ensure that JavaScript is enabled and that their browser is up-to-date for smooth operation. Browser caching should also be cleared regularly to prevent any loading issues during real-time detection or media uploads.

3.2 Limitations

While our software provides high-quality detection and classification capabilities, there are certain limitations to consider:

1. Model Accuracy: The accuracy of the detection depends on the quality of the pre-trained model used (YOLOv8, YOLOv9, YOLOv10). If a custom model is used, its accuracy depends on how well it has been trained on similar datasets.
2. Processing Time:
 - Real-time detection: When using a live video feed, processing occurs in real-time, but slight delays may occur depending on the network and computational load.
 - Uploaded media: For images and videos, larger file sizes or longer video durations will increase processing time. For example, longer videos may take more than 20 seconds to process, depending on their length and resolution.
3. Supported File Types:
 - Images: Only JPEG and PNG formats are supported for image uploads.
 - Videos: Only MP4 and MOV formats are supported for video uploads.
 - If unsupported file types are uploaded, the system will display an error message.
4. Custom Models: Users can upload their own detection models, but these models must be properly configured and formatted to work with the software. Incorrect model paths or improperly formatted models will result in errors during the detection process.
5. Device Performance: Detection accuracy and processing time may be affected by the performance of the user's device. More complex models or larger videos might require more processing power, resulting in slower performance on less powerful devices.

3.3 Software Overview

3.3.1 View of Home Page

The Home Page of the software is the main interface where real-time detection of edible bird nests takes place. It serves as the dashboard for monitoring and analyzing live video feeds, allowing users to detect and classify bird nests based on their quality.

The Home Page is designed for simplicity and functionality, offering an intuitive way for users to interact with the detection models, view real-time results, and manage data.

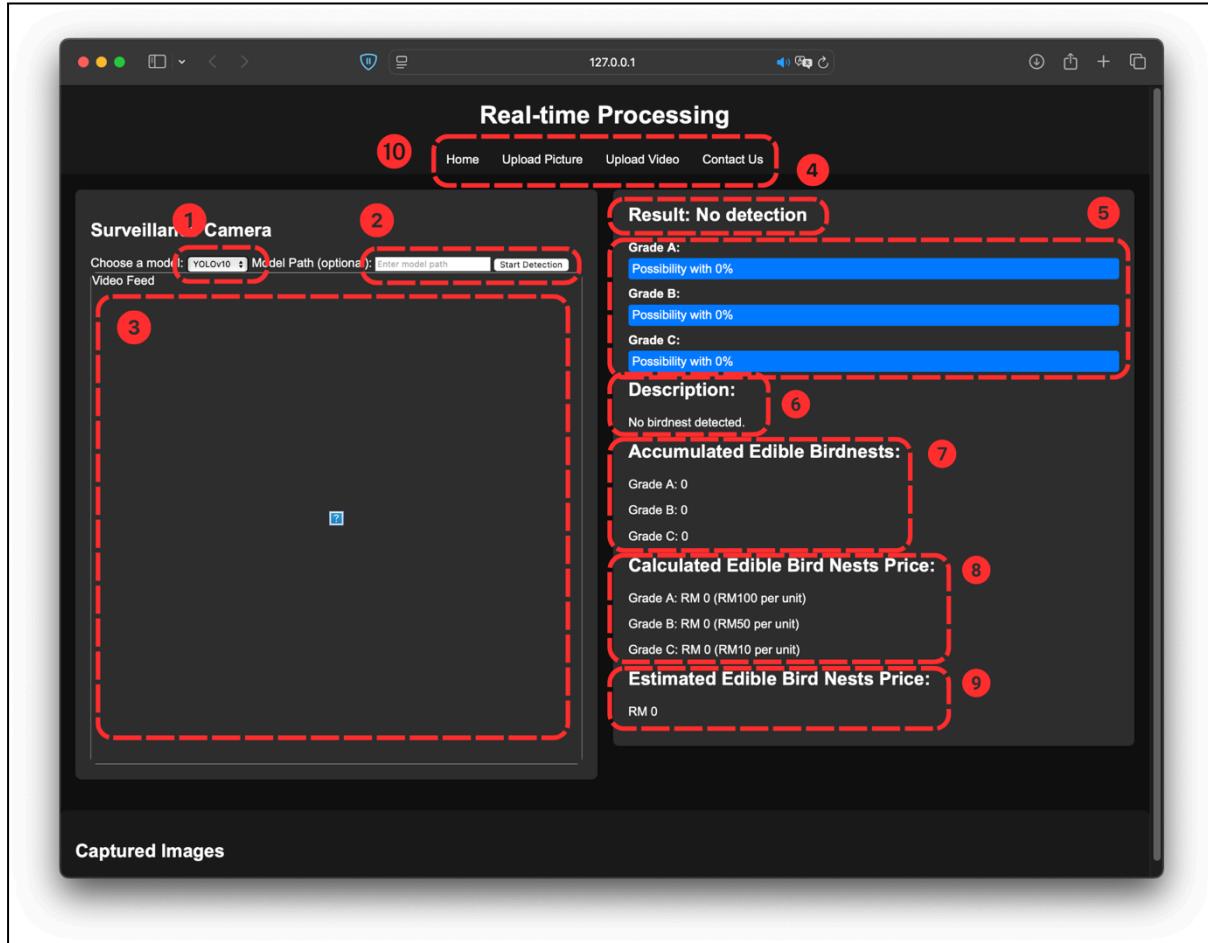


Figure 1. Home page user interface

Figure 1. Numbered sections	Description	Function
1	Model Selection Dropdown	Allows users to select a pre-trained model (e.g., YOLOv8, YOLOv9, YOLOv10).
2	Model Path Input	Users can input a custom model path for detection if they are not using pre-trained models.
3	Video Feed Display Area	Displays the live video feed for real-time detection.
4	Detection Status	Displays the detection result (e.g., "No detection" or "Grade A Birdnest").
5	Possibility Results	Shows the confidence percentages for each grade of detected bird nests (A, B, C).
6	Detection Description	Provides details about the detected grade of the bird nest or states "No birdnest detected."
7	Accumulated Edible Bird Nests Count	Shows the total number of bird nests detected for each grade (A, B, C).
8	Calculated Edible Bird Nests Price	Displays the calculated price of detected bird nests based on the count for each grade.
9	Estimated Edible Bird Nests Price	Shows the overall estimated price based on the accumulated total from the calculated price section.
10	Navigation Menu	Provides links to other sections of the application (Home, Upload Picture, Upload Video, Contact Us).

Table 1: Description and function for each element from Figure 1.

3.3.2 View of Upload Picture Page

The Upload Picture page allows users to upload an image for processing and detection of edible bird nests. This section provides a step-by-step guide on how this functionality works.

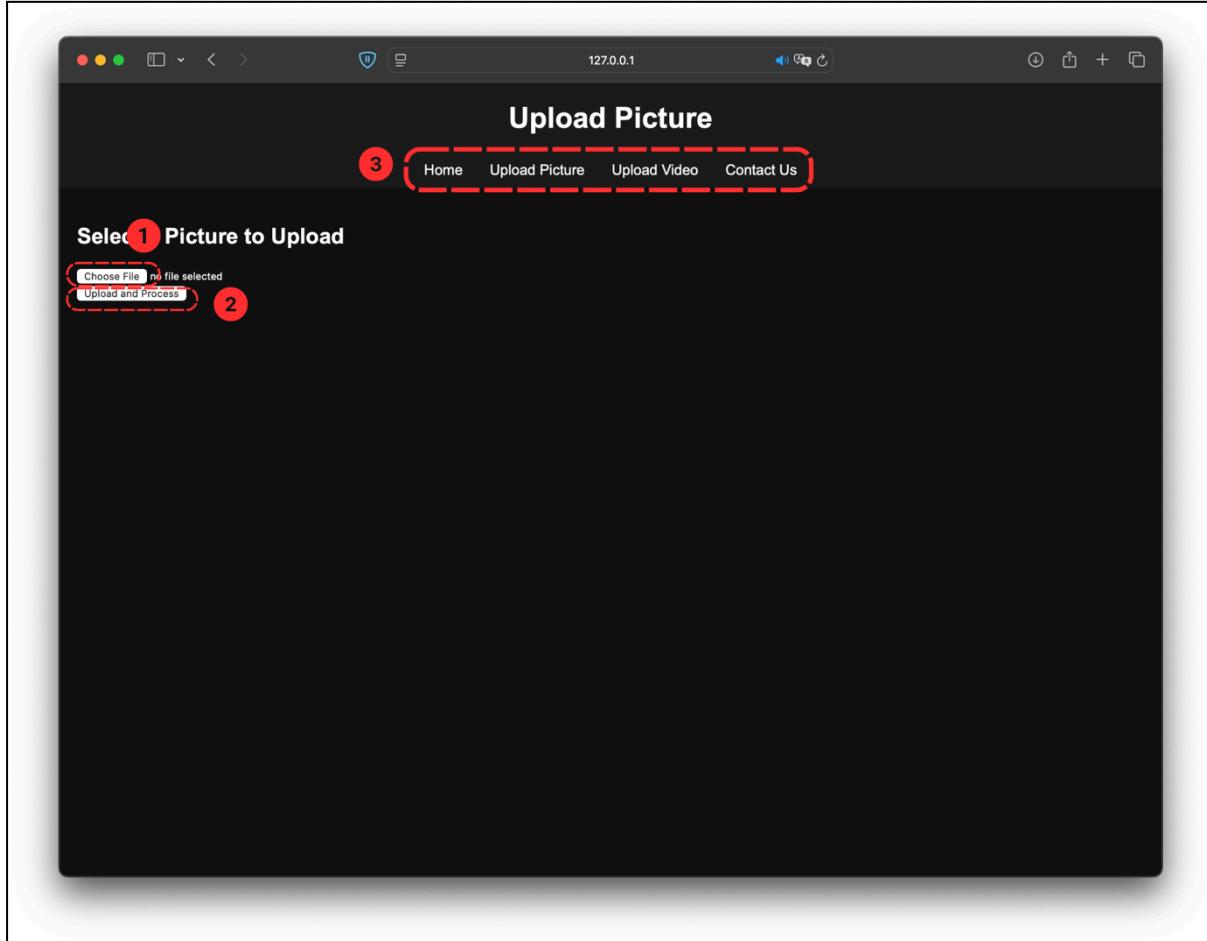


Figure 2. Upload picture page user interface

Figure 2 Numbered sections	Description	Function
1	Choose File Button	Allows the user to browse and select an image file from their local device to upload for processing.
2	Upload and Process Button	After selecting a file, the user clicks this button to initiate the bird nest detection process.
3	Navigation Menu	Provides access to other sections of the application, such as Home, Upload Picture, Upload Video, and Contact Us.

Table 2: Description and function for each element from Figure 2.

3.3.3 View of Upload Video Page

The Upload Video page works similarly to the Upload Picture page but is designed for video processing.

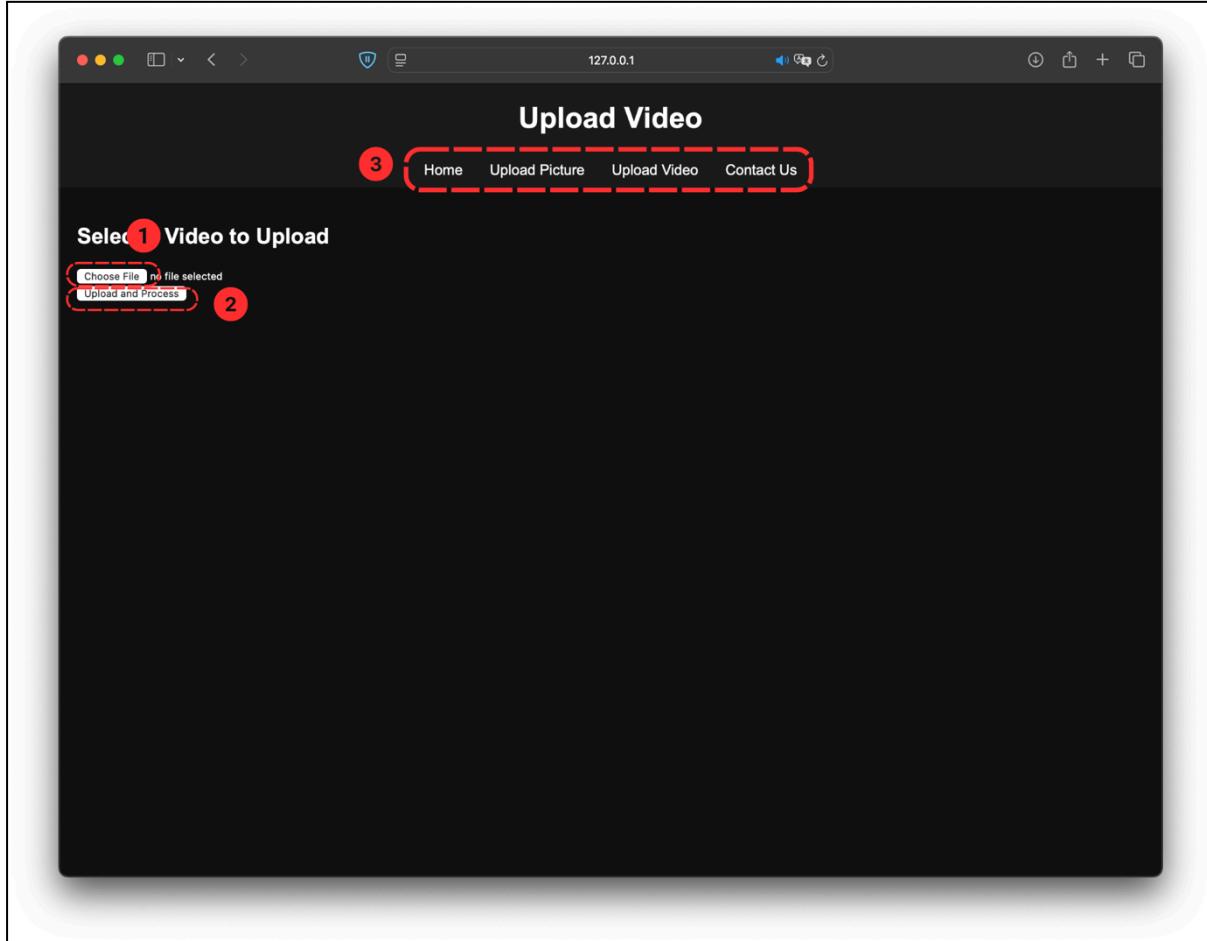


Figure 3. Upload video page user interface

Figure 3 Numbered sections	Description	Function
1	Choose File Button	Allows the user to browse and select a video file from their local device to upload for processing.
2	Upload and Process Button	After selecting a video, the user clicks this button to initiate bird nest detection in the video.
3	Navigation Menu	Provides access to other sections of the application, such as Home, Upload Picture, Upload Video, and Contact Us.

Table 3: Description and function for each element from Figure 3.

3.3.4 View of Contact Us Page

The Contact Us page provides essential contact information for support and queries related to the software.

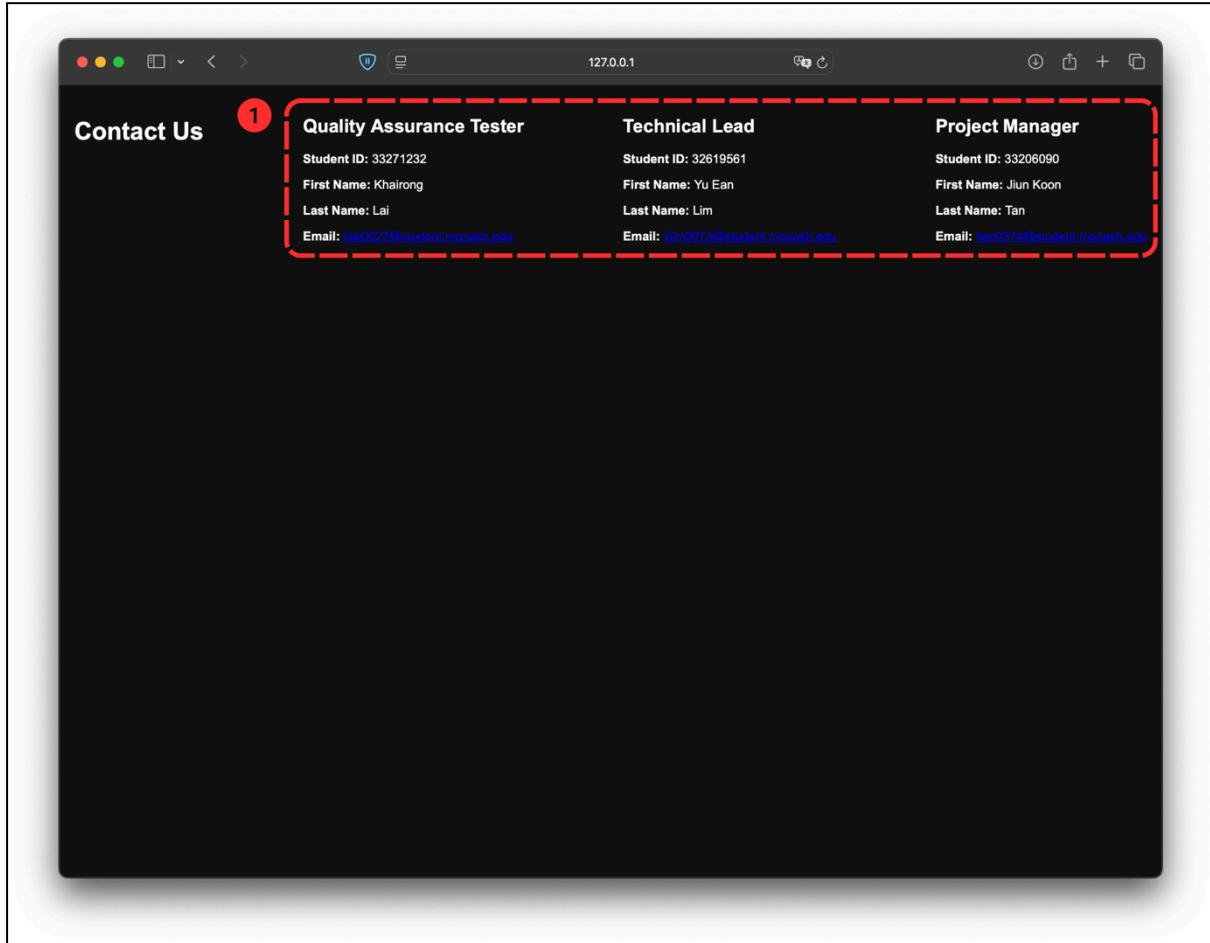


Figure 4. Contact us page user interface

The Contact Us page lists the key team members involved in the project, including their:

- Role (e.g., Quality Assurance Tester, Technical Lead, Project Manager).
- Name and Student ID (if relevant for student projects).
- Email addresses, allowing users to contact the appropriate team members for support or queries.

This page acts as a convenient directory for users seeking technical assistance, project inquiries, or feedback submission.

3.4 User Interaction with User Interface

This section outlines how users interact with the software's user interface across its various tabs. Each tab represents a unique functionality, enabling users to perform tasks such as real-time bird nest detection, uploading and processing images or videos, and accessing contact information. Each interaction is designed to be user-friendly, allowing seamless transitions between tasks with minimal effort.

3.4.1 Real-time detection feature with Home Tab

The Home Tab is the central hub for real-time detection of edible bird nests. Users interact with this tab primarily to monitor live video feeds and detect bird nests based on selected models. Here are the steps:

Step 1: Pick a model and start detection

Pick a model from the drop-down menu or provide a custom model path and start detection.

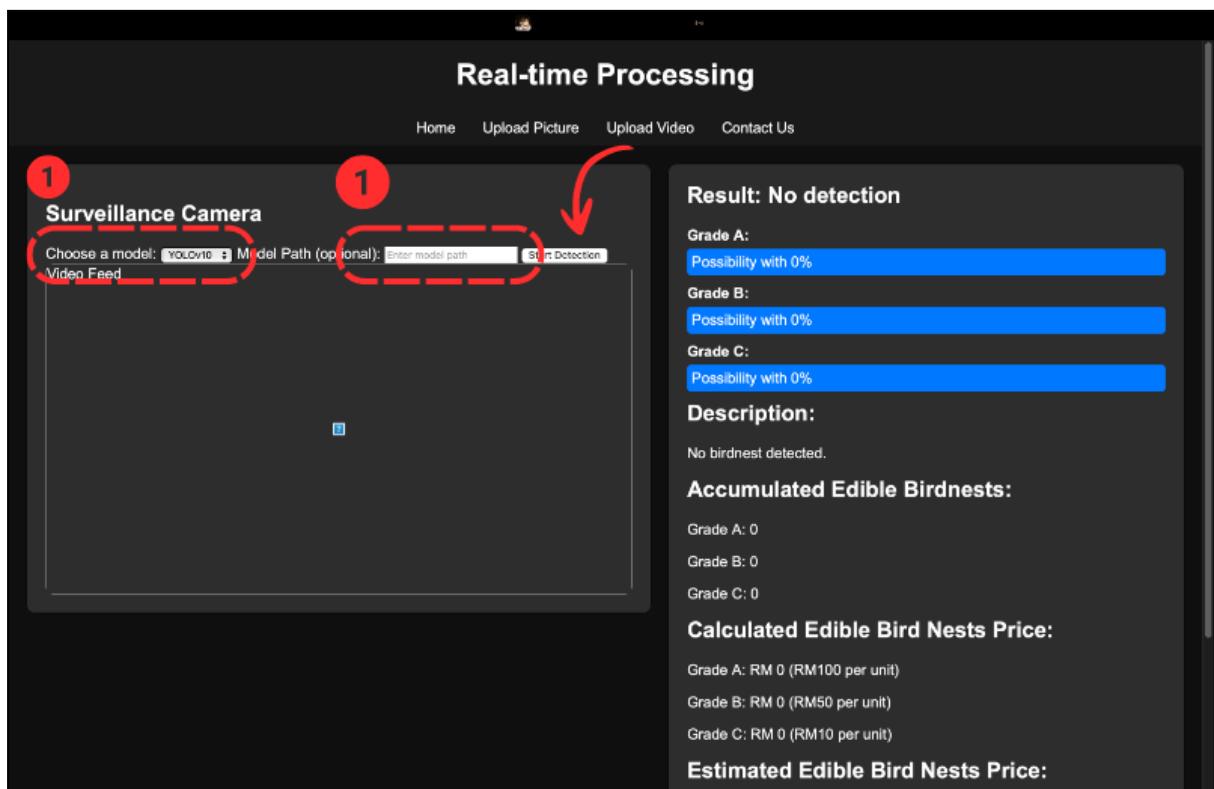


Figure 5. Home page user interface for selecting model

Step 2: Prepare your laptop or phone as a camera

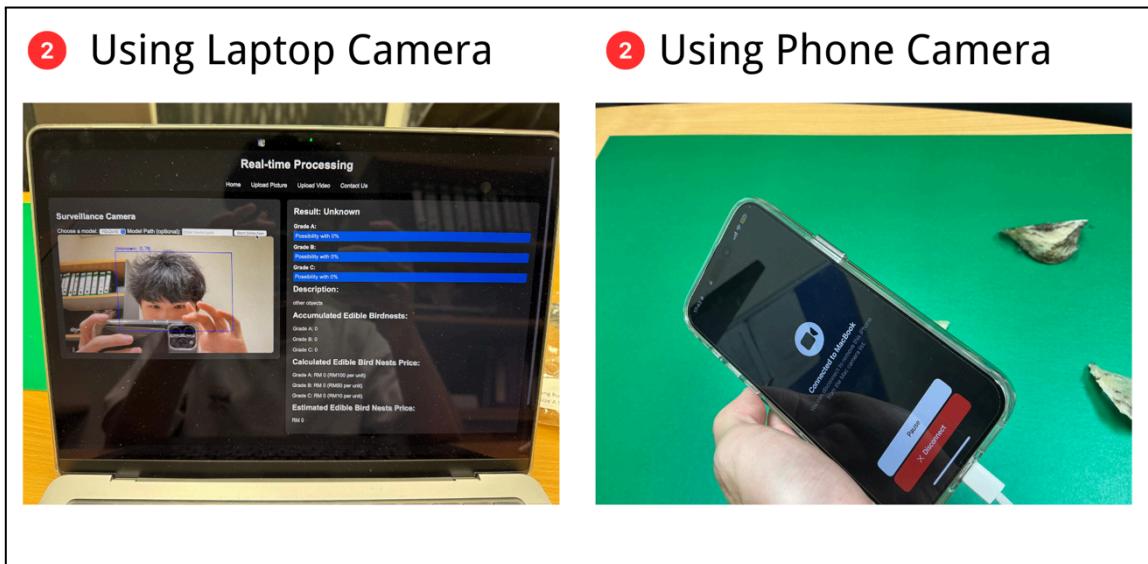


Figure 6. Both methods for detection

Step 3: View and Interpret the Detection Results

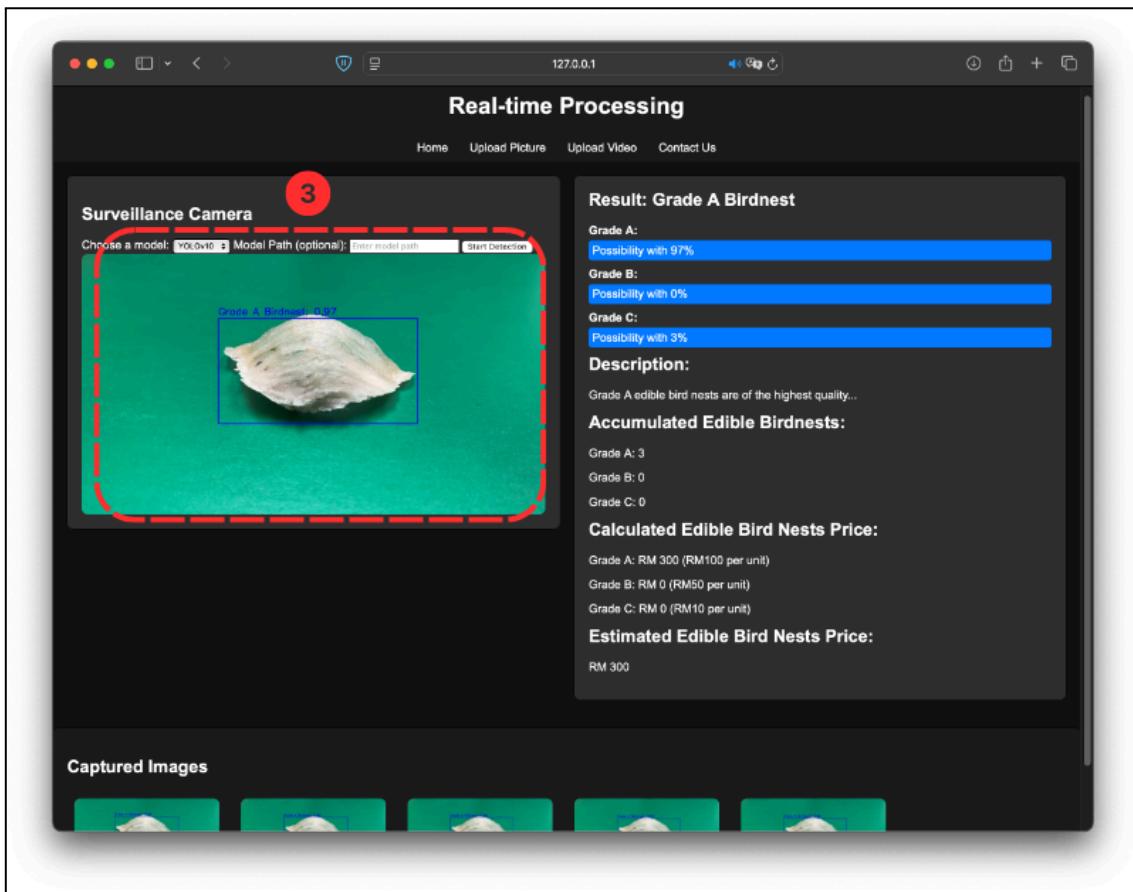


Figure 7. Home page user interface during real-time detection

Once the detection process begins, you will be able to view real-time results on the Home Tab dashboard as well as the top 5 recently captured edible bird nest images

3.4.2 Upload Picture feature with Upload Picture Tab

The Upload Picture Tab allows users to upload images for processing and bird nest detection.

Here are the steps:

Step 1: Click the "Choose File" Button

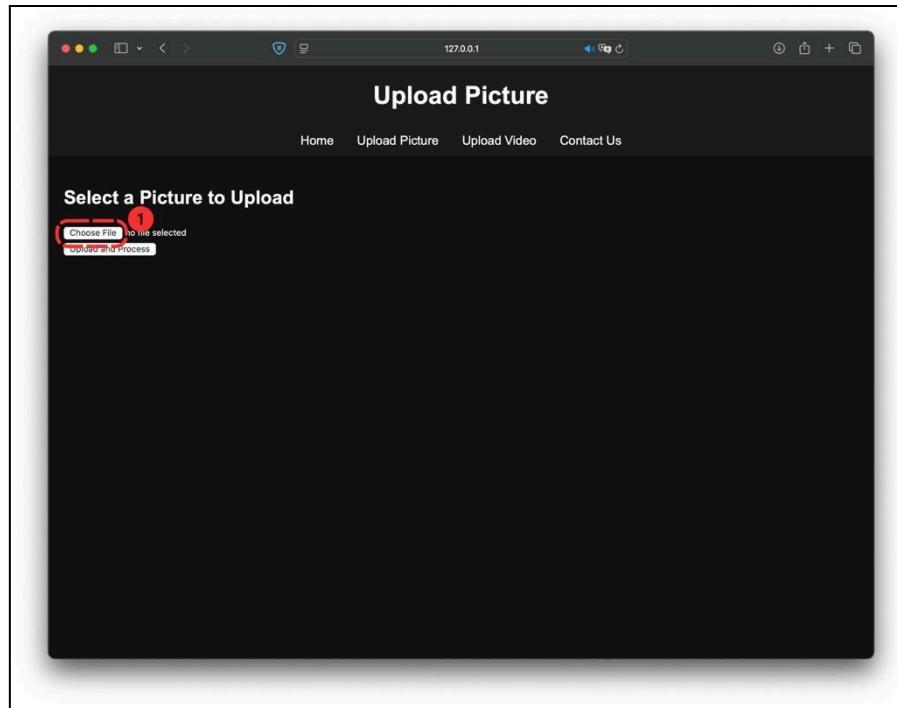


Figure 8. Upload picture page user interface

Step 2: Select a Supported Image as Input

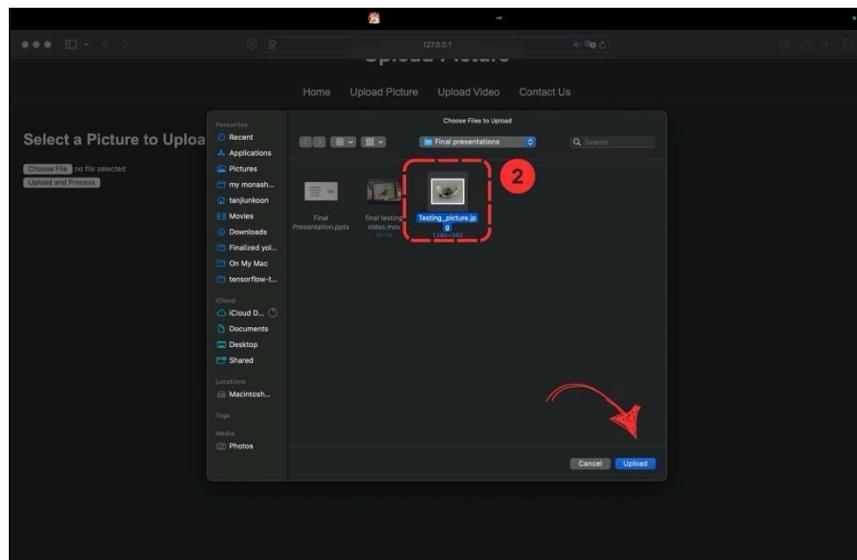


Figure 9. Select image user interface

Step 3: Press the "Upload and Process" Button

Once you see the selected file icon appear next to the Choose File button, the file is ready for upload.

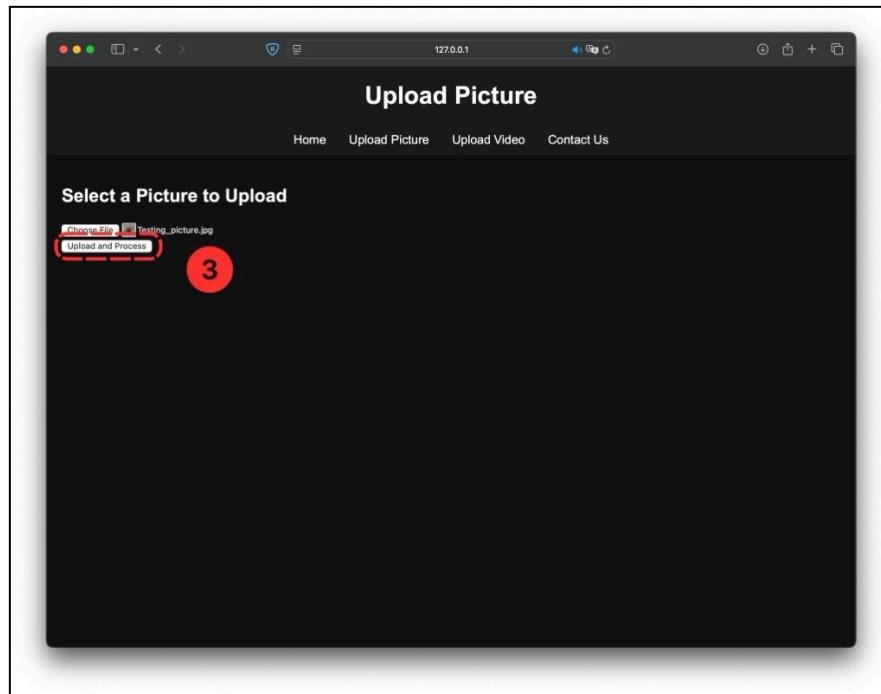


Figure 10. User interface for upload and process the picture

Step 4: View the Result from the Website

After the file has been processed, the results will be displayed directly on the website

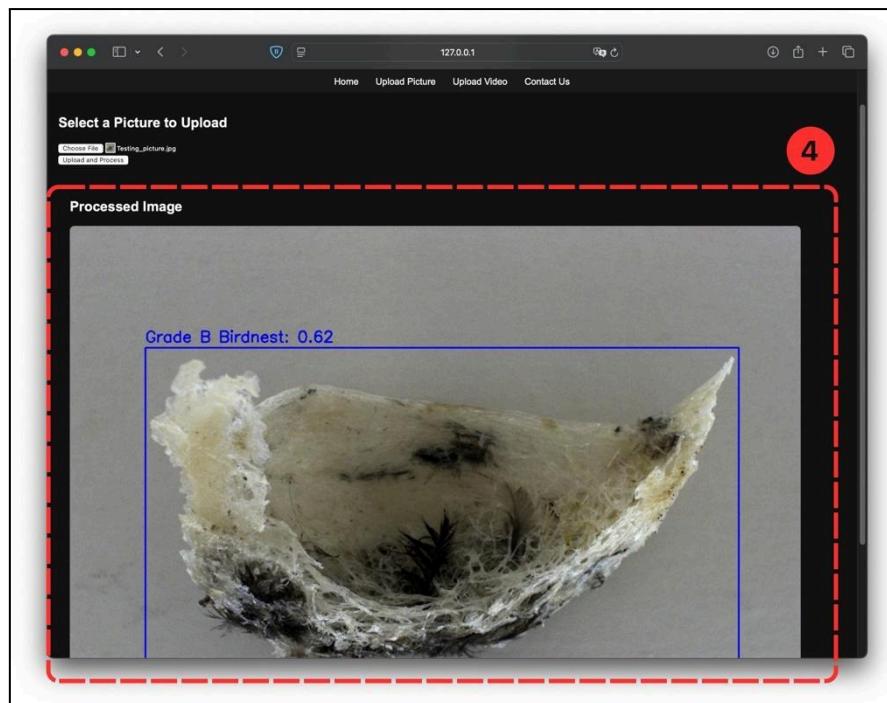


Figure 11. Upload picture user interface showing result

3.4.3 Upload Video feature with Upload Video Tab

The Upload Video Tab allows users to upload videos for bird nest detection. The interaction workflow is similar to the Upload Picture Tab but optimized for video content.

Here are the steps:

Step 1: Click the "Choose File" Button

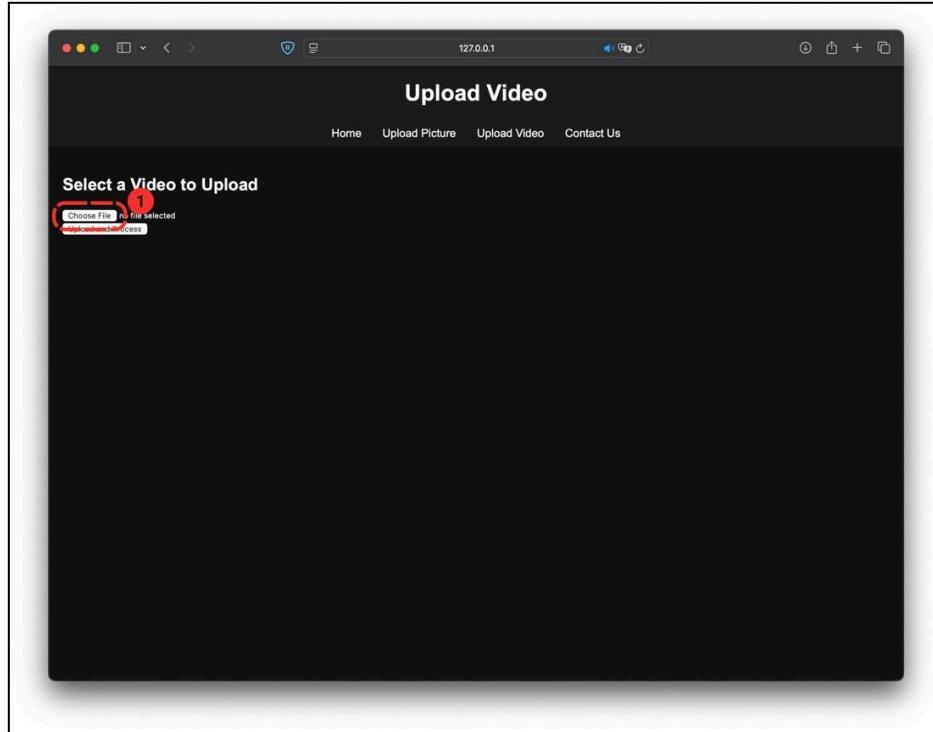


Figure 12. Upload video page user interface

Step 2: Select a Supported Video as Input

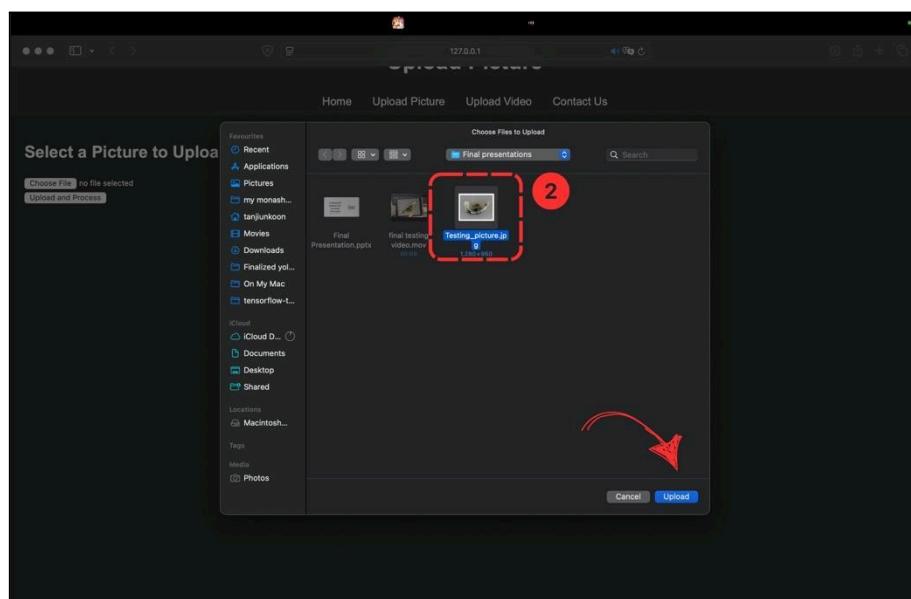


Figure 13. Select video user interface

Step 3: Press the "Upload and Process" Button

Once you see the selected file icon appear next to the Choose File button, the file is ready for upload.

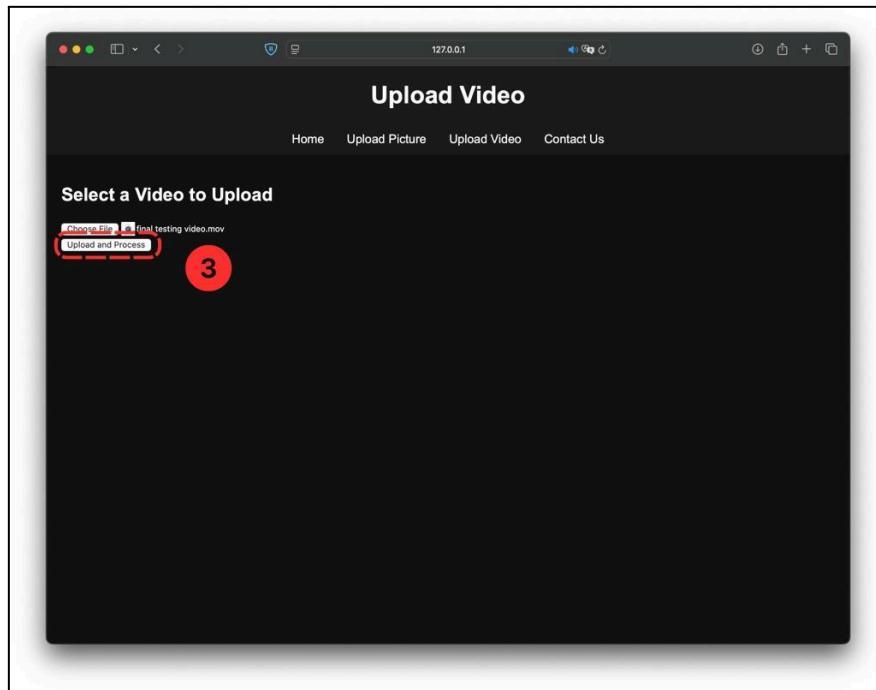


Figure 14. User interface for upload and process the video

Step 4: View the Result from the Website

The processed video will display detection results overlaid on the video frames, showing the bounding boxes and grade classification for each detected bird nest.

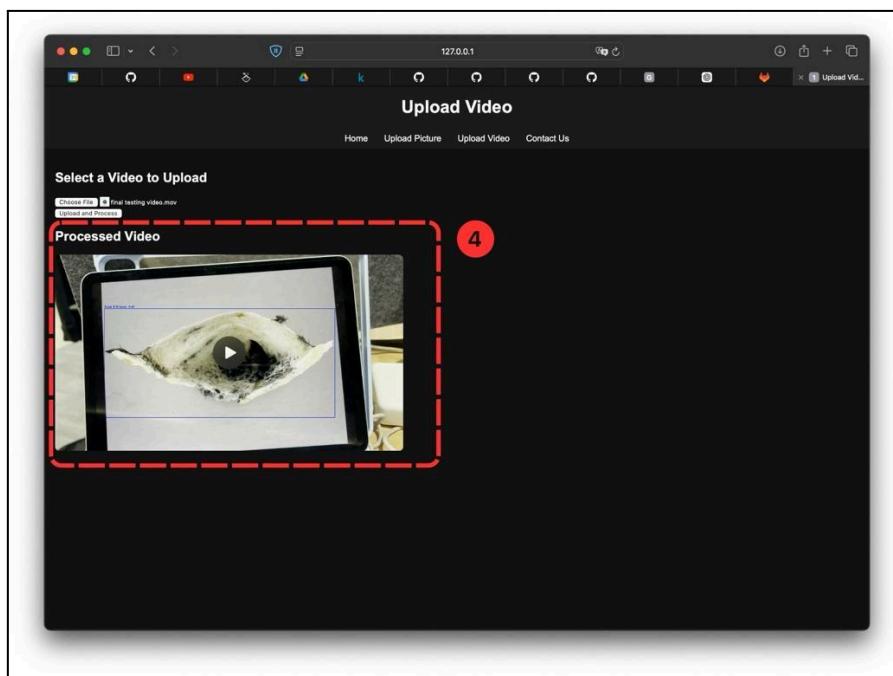


Figure 15. Upload video user interface showing result

3.4.4 Information with Contact Us Tab

The Contact Us Tab provides users with essential information for reaching out to the development team for support or inquiries.

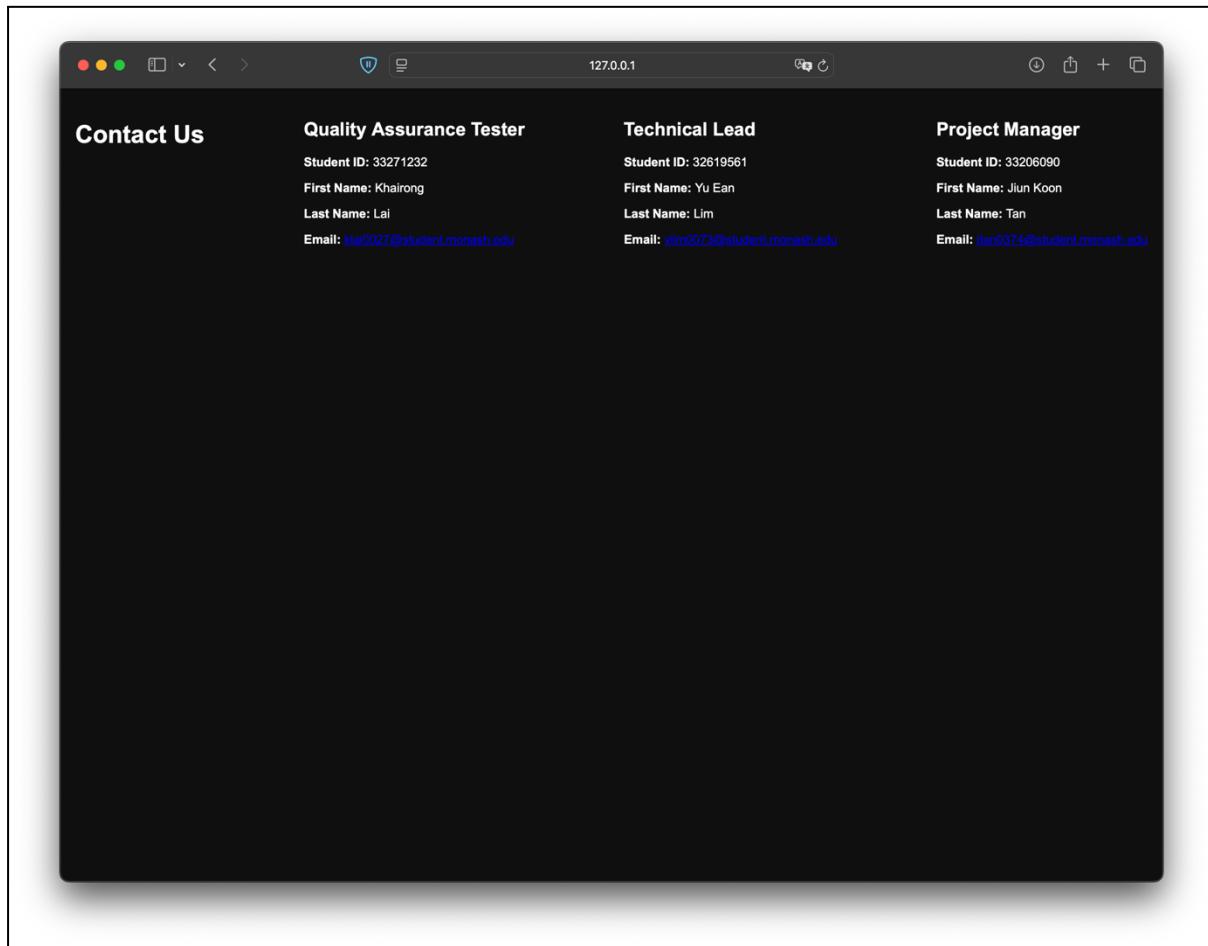


Figure 16. Contact us user interface

3.5 Exiting the Software

Exiting the software is straightforward and requires no special procedure:

- Close the Tab: If you're using the software in a browser tab, you can exit by simply closing the tab.
- Close the Entire Browser: Alternatively, you can exit the software by closing the entire browser. Be aware that closing the browser will end all sessions and close any other tabs that may be open.

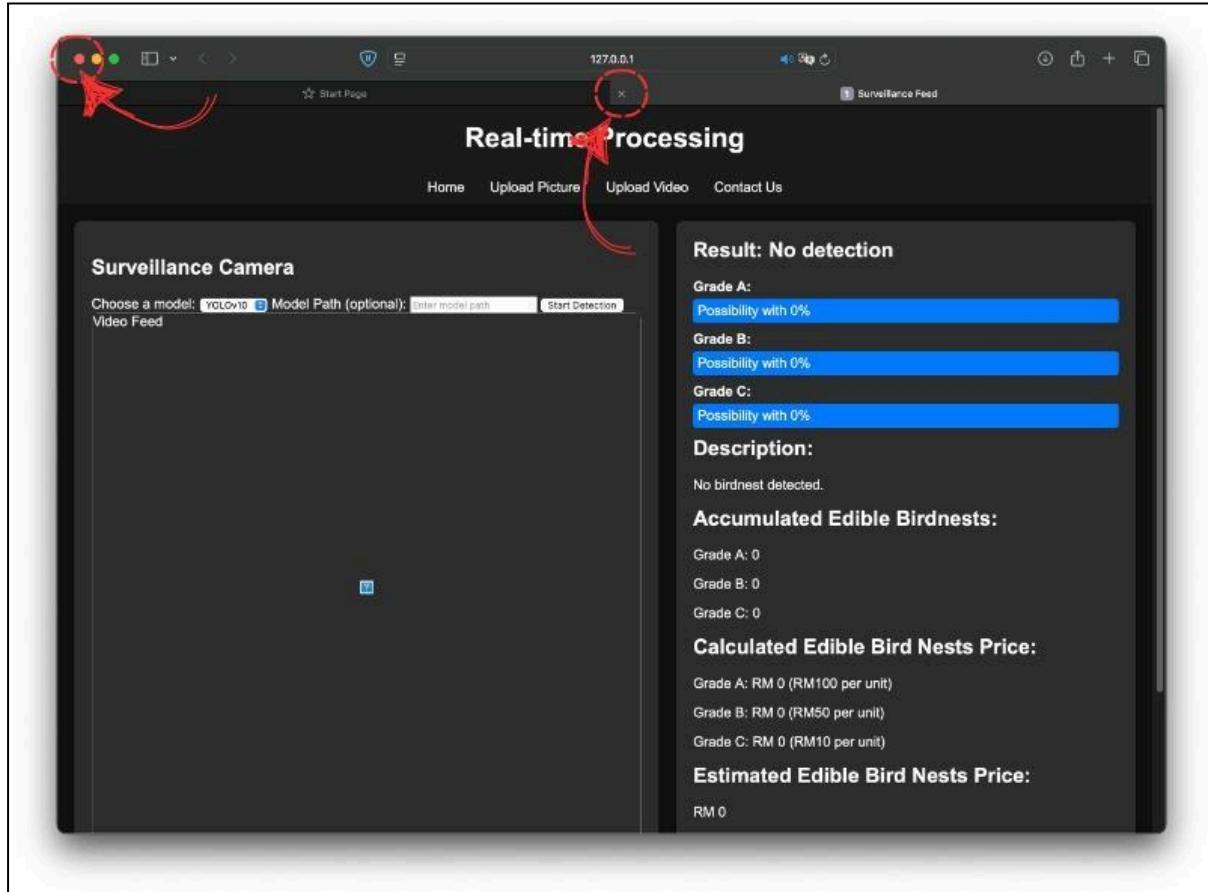


Figure 17. Exit user interface

3.6 Dealing with Special Situations

At times, users may encounter issues while using the software. Here's how to resolve some common problems:

3.6.1 Model Not Loaded Initially

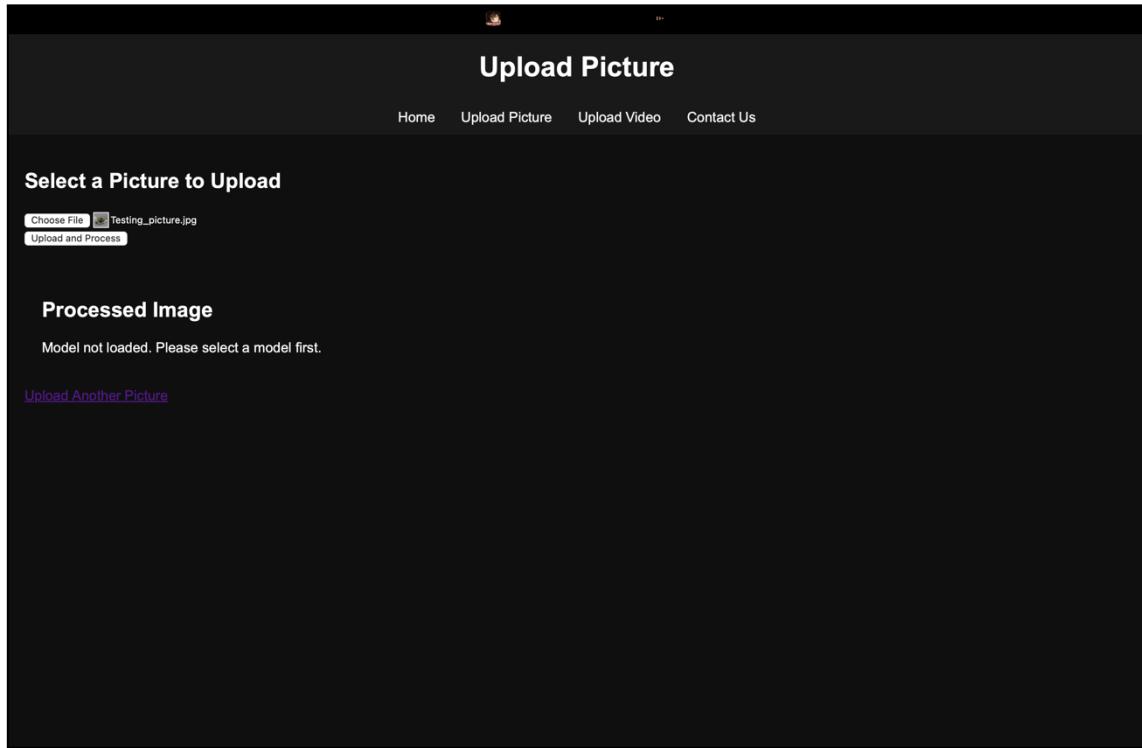


Figure 18. Model not loaded initially situation user interface

If you encounter the error “Model not loaded. Please select a model first” after uploading a file, this issue can be resolved by:

- Step 1: Navigate back to the Home Page.
- Step 2: Select a model from the dropdown or provide the custom model path.
- Step 3: Press the Start Detection button to load the model.
- Once the model is loaded, you can return to the Upload Picture or Upload Video tab and proceed with uploading and processing your media.

3.6.2 Unsupported File Format

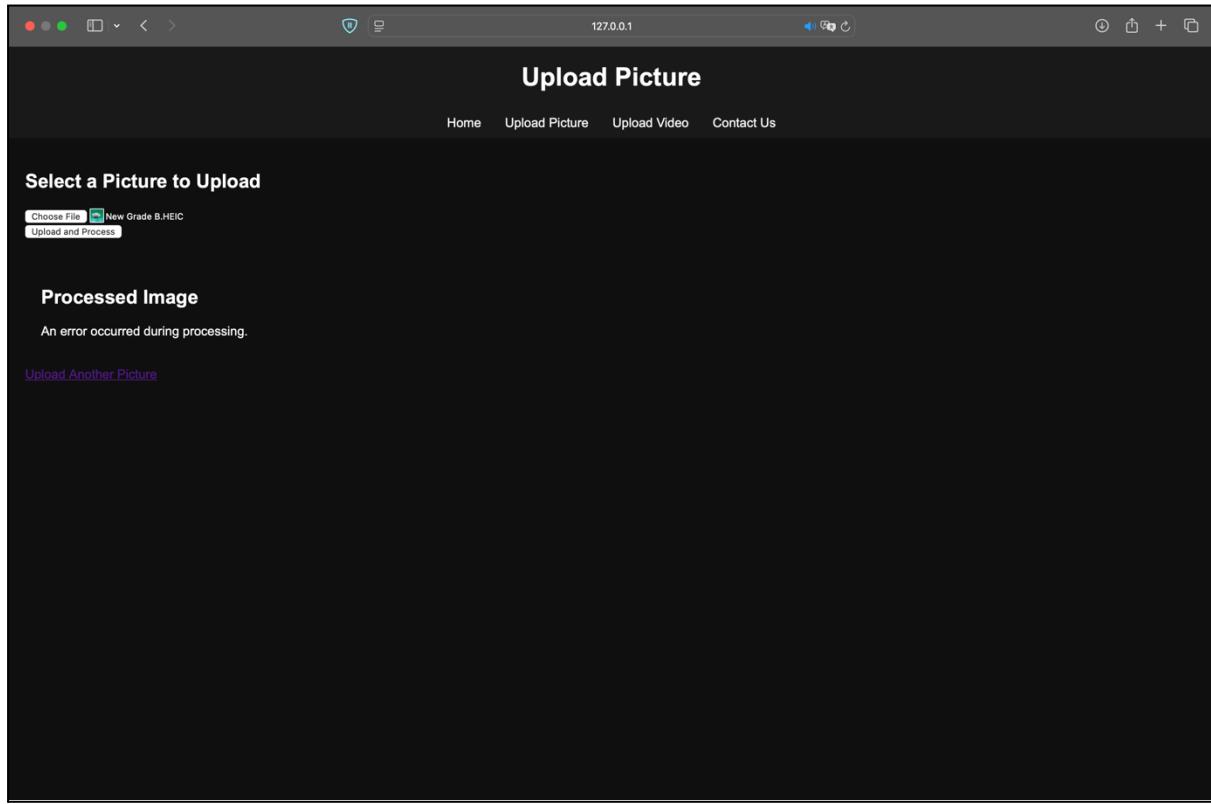


Figure 19. Unsupported file format situation user interface

If you upload a file with an unsupported format (such as HEIC for images), you'll see an error message like "An error occurred during processing." To resolve this:

Ensure that the file you're uploading is in a supported format:

- Images: Use JPEG or PNG formats.
- Videos: Use MP4 or MOV formats.

Once you have a valid file format, re-upload the file by selecting it again and clicking Upload and Process.

3.6.3 No File Selected When Uploading

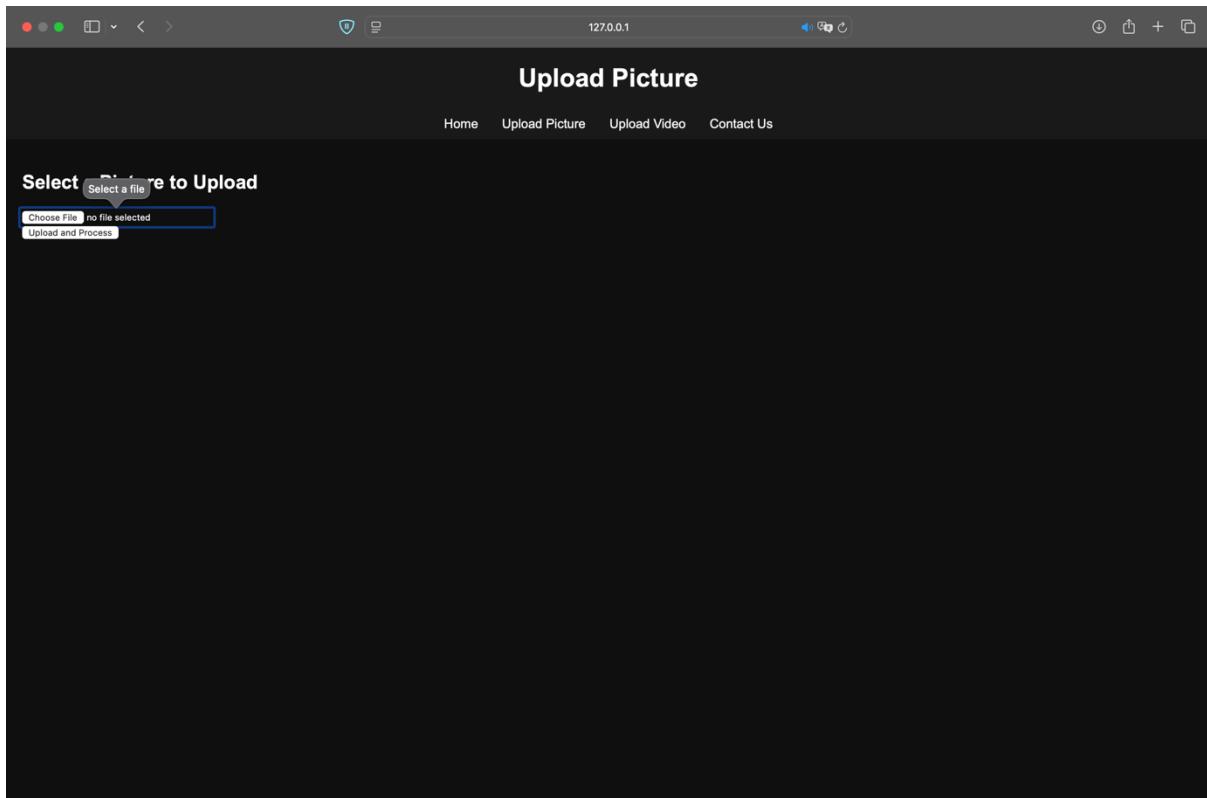


Figure 20. No file selected situation user interface

If you attempt to press the Upload and Process button without selecting a file, the system will not perform the detection process. To resolve this issue:

- Step 1: Make sure to first click the Choose File button.
- Step 2: Select the correct file from your device.
- Step 3: Once the file icon appears next to the Choose File button, click Upload and Process again.
- This will trigger the processing and display the detection results as expected.

4. Technical Guide

Step 1: Download and Install Visual Studio Code

The first step in setting up the software environment is downloading and installing Visual Studio Code (VS Code), a powerful and lightweight code editor. Follow these steps:

1. Go to the official Visual Studio Code website: <https://code.visualstudio.com/>.
2. Download the installer for your operating system (Windows, macOS, or Linux).
3. Follow the installation instructions provided on the website. Once installed, open Visual Studio Code on your computer.

Step 2: Open the Project Folder in Visual Studio Code

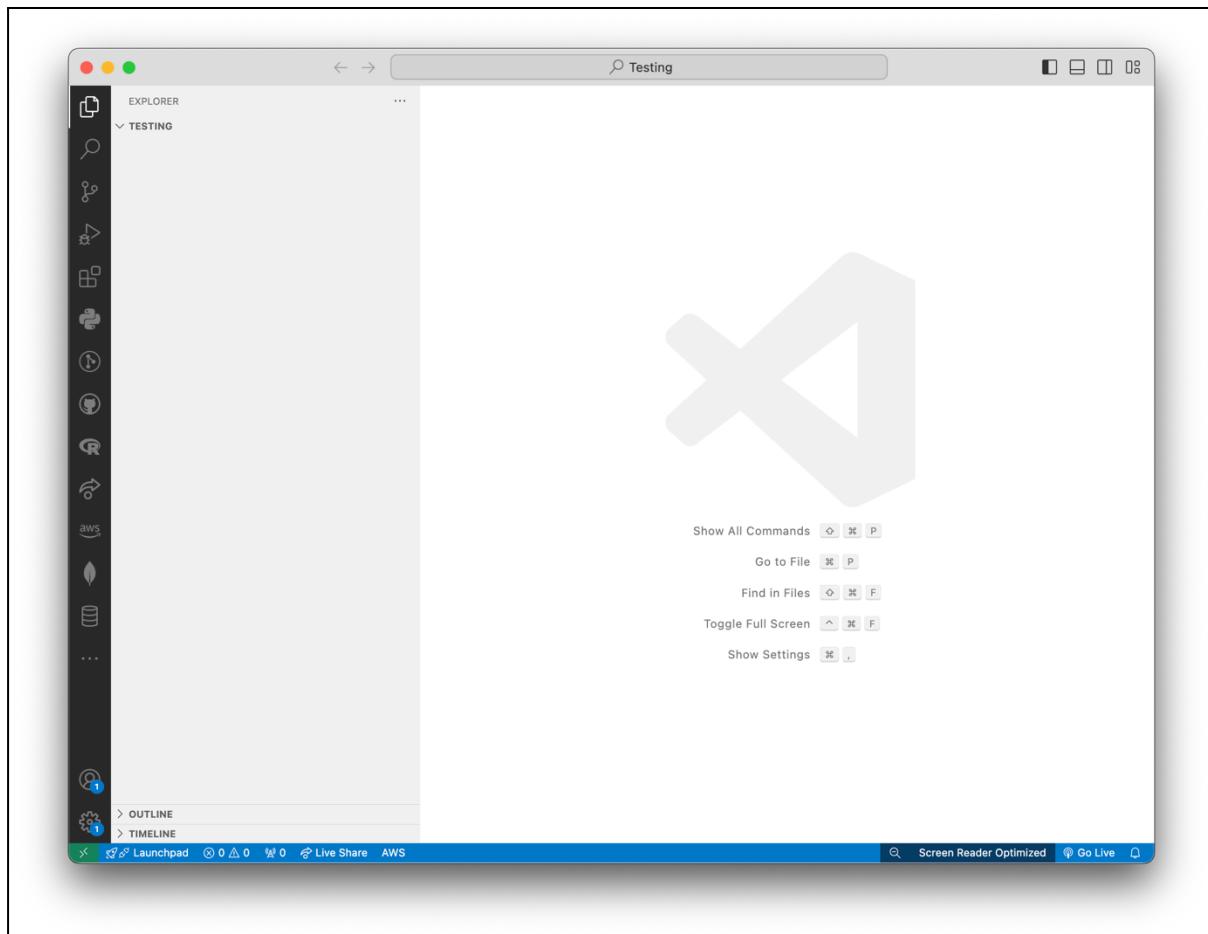


Figure 21. Visual Studio Code user interface

To work with the project code, you will need to open the project folder in Visual Studio Code:

1. After launching VS Code, go to the File menu and click Open Folder.
2. Navigate to the directory where you want to store the project files, or create a new folder for the project.
3. Select the folder and click Open. This will open the folder in the Visual Studio Code environment.

Step 3: Cloning the Project Code

To get a copy of the project code on your machine, you need to clone it from the GitHub repository. Follow these steps:

1. Open the terminal in Visual Studio Code by navigating to the Terminal menu and selecting New Terminal.
2. In the terminal, run the following command to clone the project repository:

```
git clone  
https://github.com/TanJiunKoon/2024\_Final\_Year\_Project\_MDS04.git
```
3. After the repository is cloned, change the directory to the project folder by typing:

```
cd 2024_Final_Year_Project_MDS04
```
4. Next, you'll need to clone the YOLOv10 repository, which is required for the object detection models:

```
git clone https://github.com/THU-MIG/yolov10.git
```

This process will download all the necessary files from both repositories into the folder on your local machine.

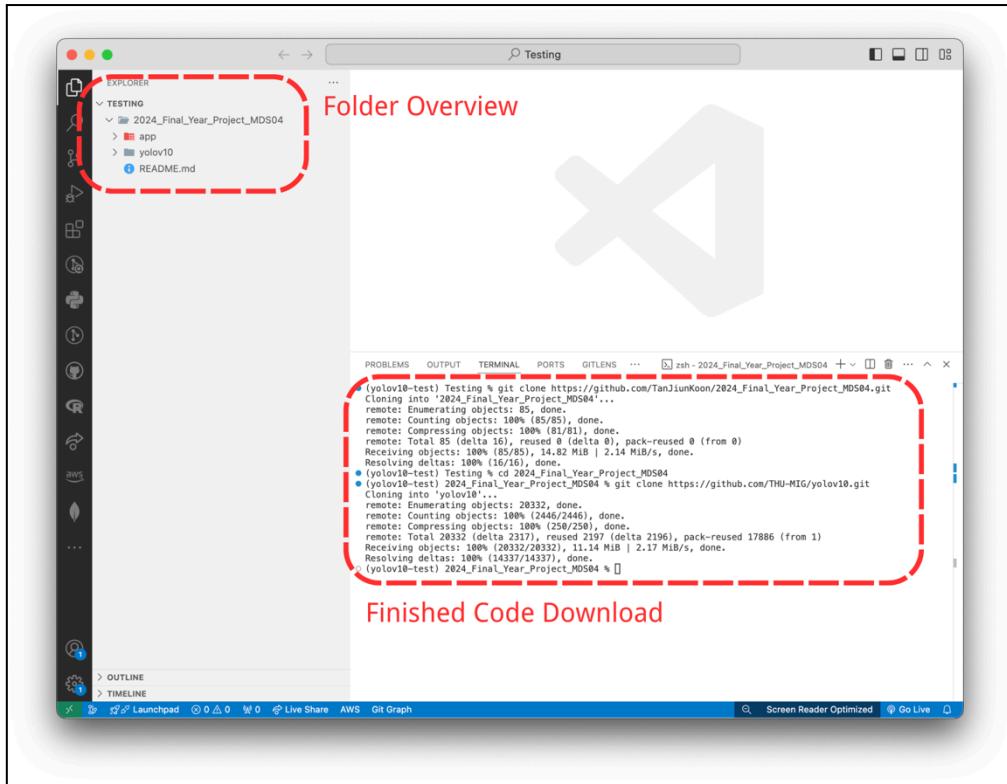


Figure 22. Visual Studio Code user interface after finished code cloning

Step 4: Creating the Conda Environment

To ensure that all dependencies are managed correctly, you'll need to create a separate Python environment using conda. This environment will isolate the software's dependencies from other projects on your machine. Follow these steps:

1. In the terminal, create a new environment for YOLOv10 by running:

```
conda create -n yolov10 python=3.9
```

2. Activate the newly created environment:

```
conda activate yolov10
```

You now have a clean environment where you can install the required libraries and dependencies for the project.

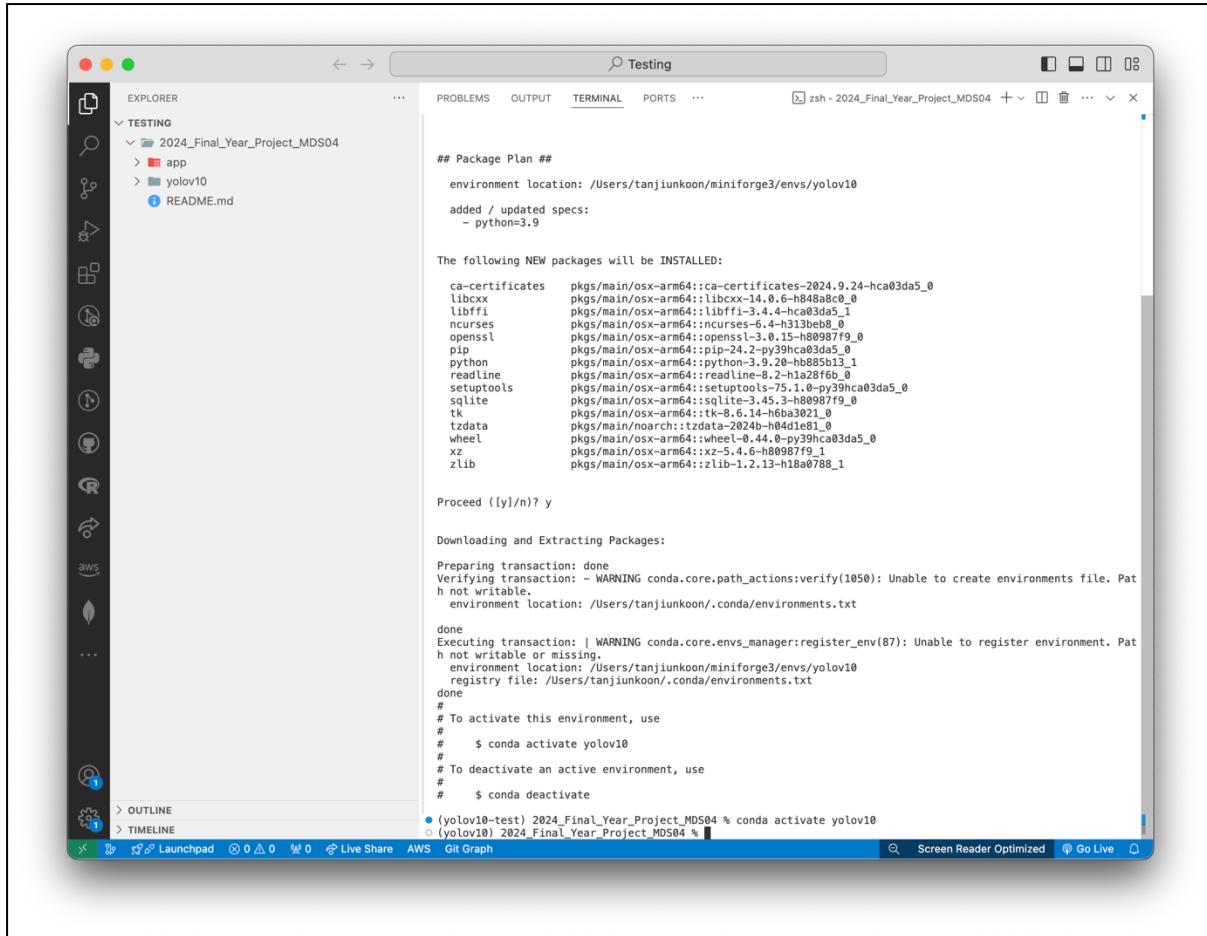


Figure 23. Visual Studio Code user interface after setting up environment

Step 5: Downloading Required Libraries and Dependencies

Once the environment is set up, you need to install all the required Python libraries and dependencies. Follow these steps:

1. Navigate to the `app` folder in the project directory:

```
cd app
```

2. Install all the necessary dependencies listed in the `requirements.txt` file:

```
pip install -r requirements.txt
```

3. After installing the dependencies for the main project, go back to the parent directory:

```
cd ..
```

4. Navigate to the YOLOv10 directory:

```
cd yolov10
```

5. Install the YOLOv10 package in editable mode:

```
pip install -e .
```

6. Once the YOLOv10 package is installed, return to the project root directory:

```
cd ..
```

At this stage, all the necessary libraries and models have been installed, and you are ready to run the software.

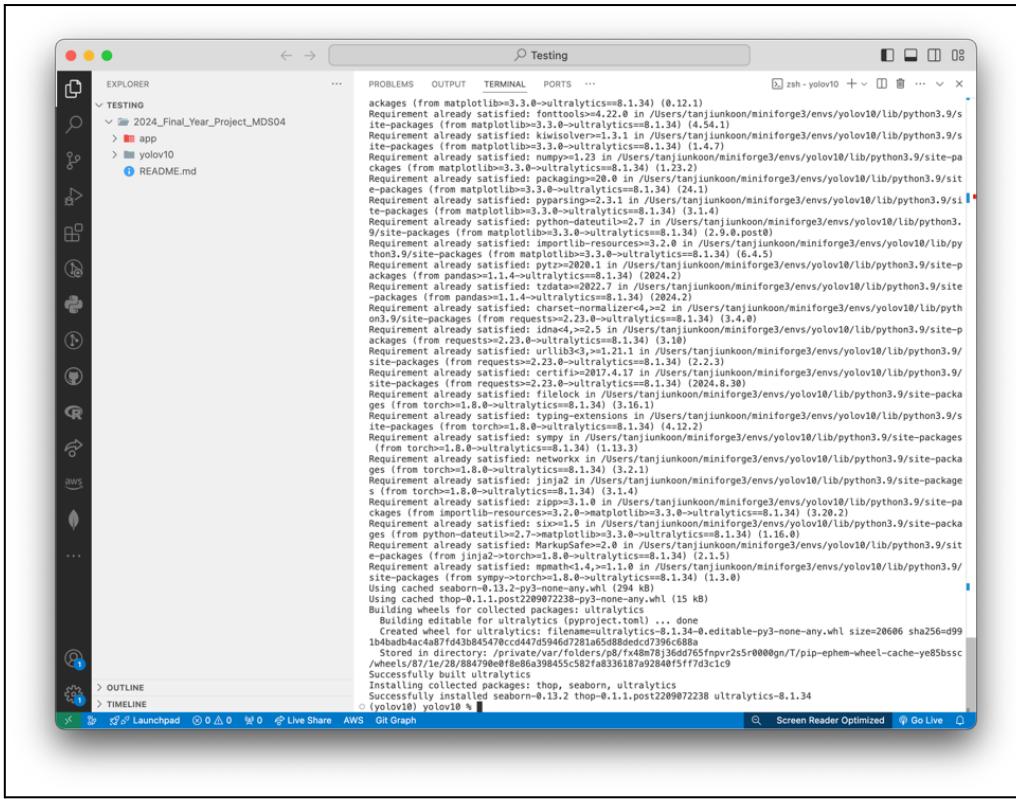


Fig. 24. Visual Studio Code user interface after downloading required libraries and dependencies

Step 6. Starting the Software

To run the software, follow these steps:

1. First, ensure that your Conda environment is activated:

```
conda activate yolov10
```

2. Start the server by running the following command:

```
uvicorn app.main:app --reload
```

This command will start a local development server, and you should see the message indicating that the server is running. You can now access the software in your browser at <http://127.0.0.1:8000>.

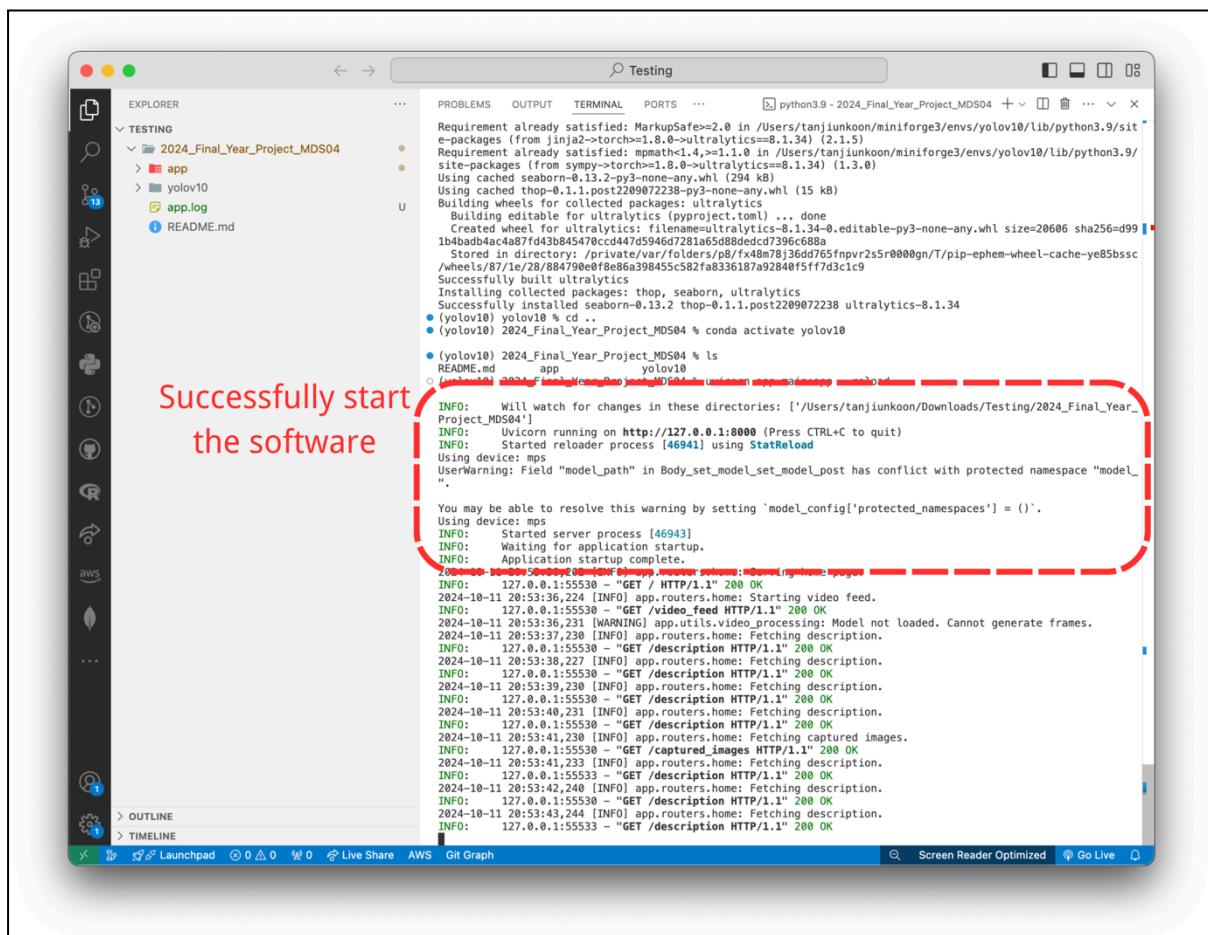


Figure 25. Visual Studio Code user interface after successfully launch the software