**Internship Report**

**Introduction**

This internship has been one of the most practical learning experiences for me, offered by Nullclass. I got to work on six different tasks that combined machine learning, computer vision, and logic-building. Each project was different in its own way and helped me strengthen my problem solving approach. More than just accuracy, I realized the importance of designing systems that actually work in real-world situations.

**Background**

Before starting this internship, I had some knowledge of machine learning and Python but had not applied it to such diverse problems. Most of my past learning was theoretical or based on simple datasets. Through these tasks, I finally experienced what it feels like to connect models with real use cases, like predicting age from voice or detecting cars in traffic.

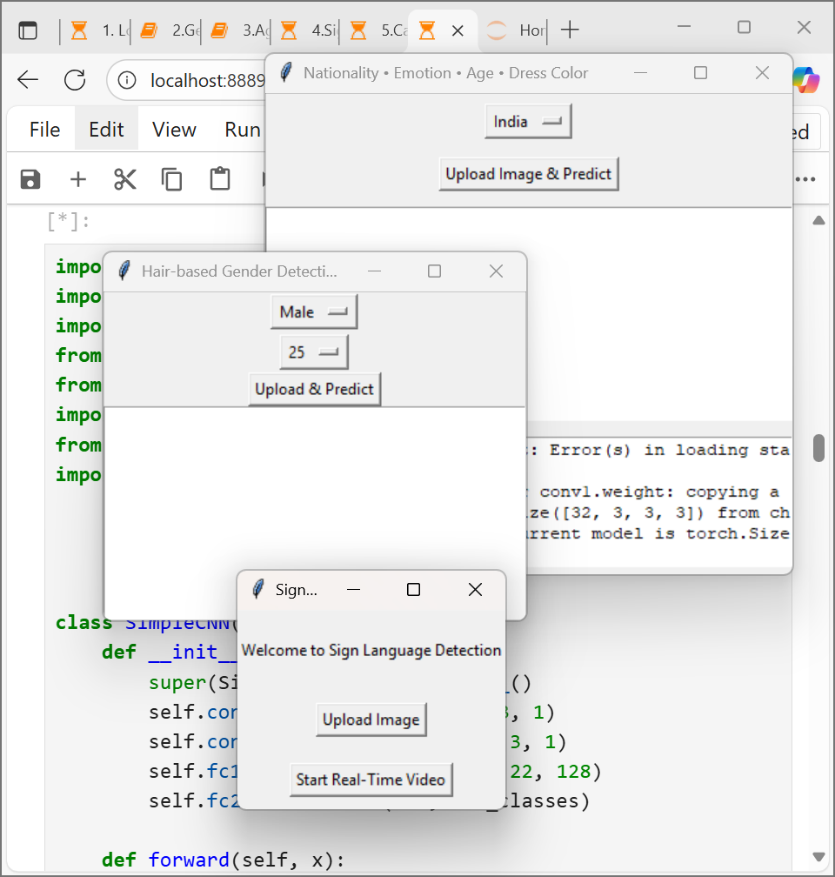
**Learning Objectives**

* To build practical machine learning models from scratch.
* To integrate logic-based conditions with predictions (for example, hair and age logic).
* To work with real-time data (video, audio, and images).
* To create simple GUIs so the models are more interactive.
* To evaluate models using metrics like confusion matrix and precision-recall.
* To manage outputs (like saving results in Excel) for usability.

**Activities and Tasks**

**Task 1: Long Hair Identification**

I created a gender detection system that treated hair length as the deciding factor for people aged 20–30. If the person was younger or older, the gender was predicted normally. I built a GUI where users could upload images and see results. This task pushed me to think beyond standard classification.

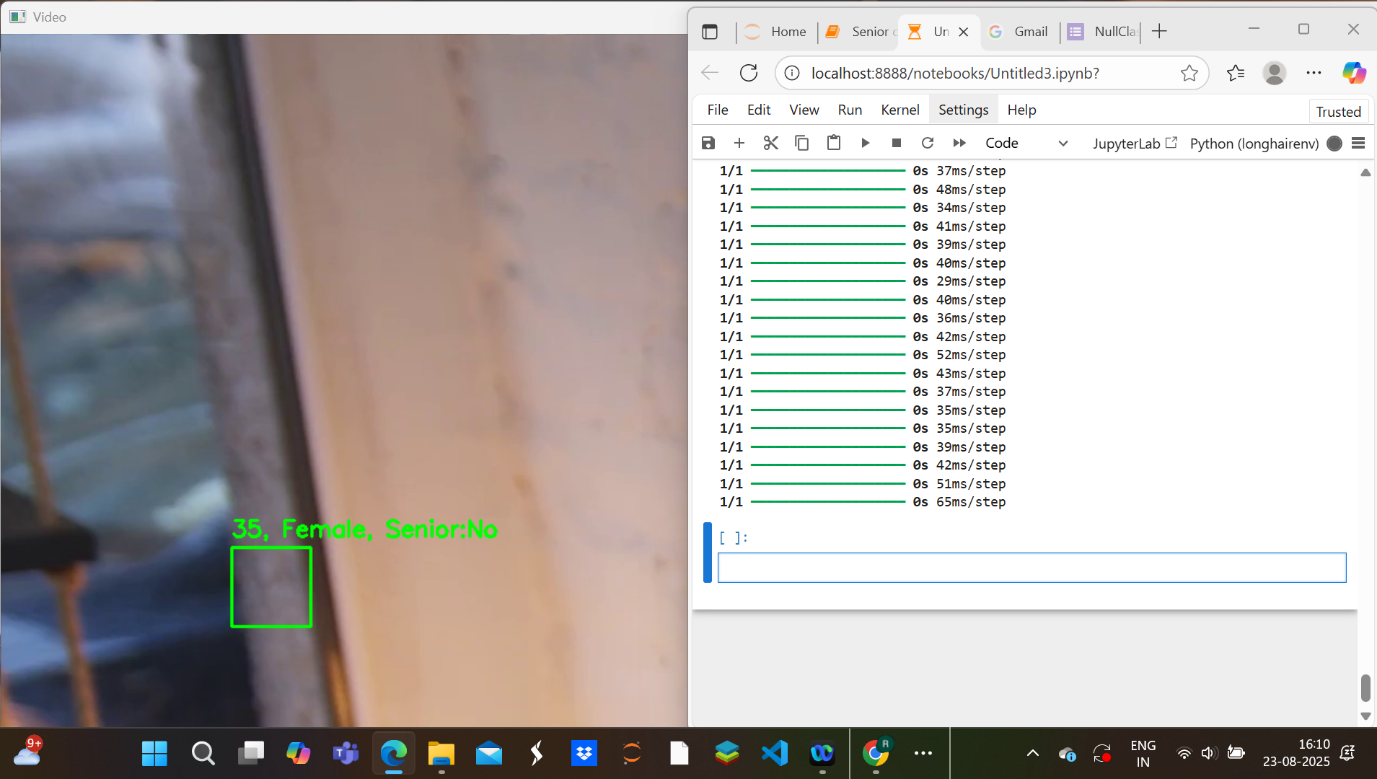


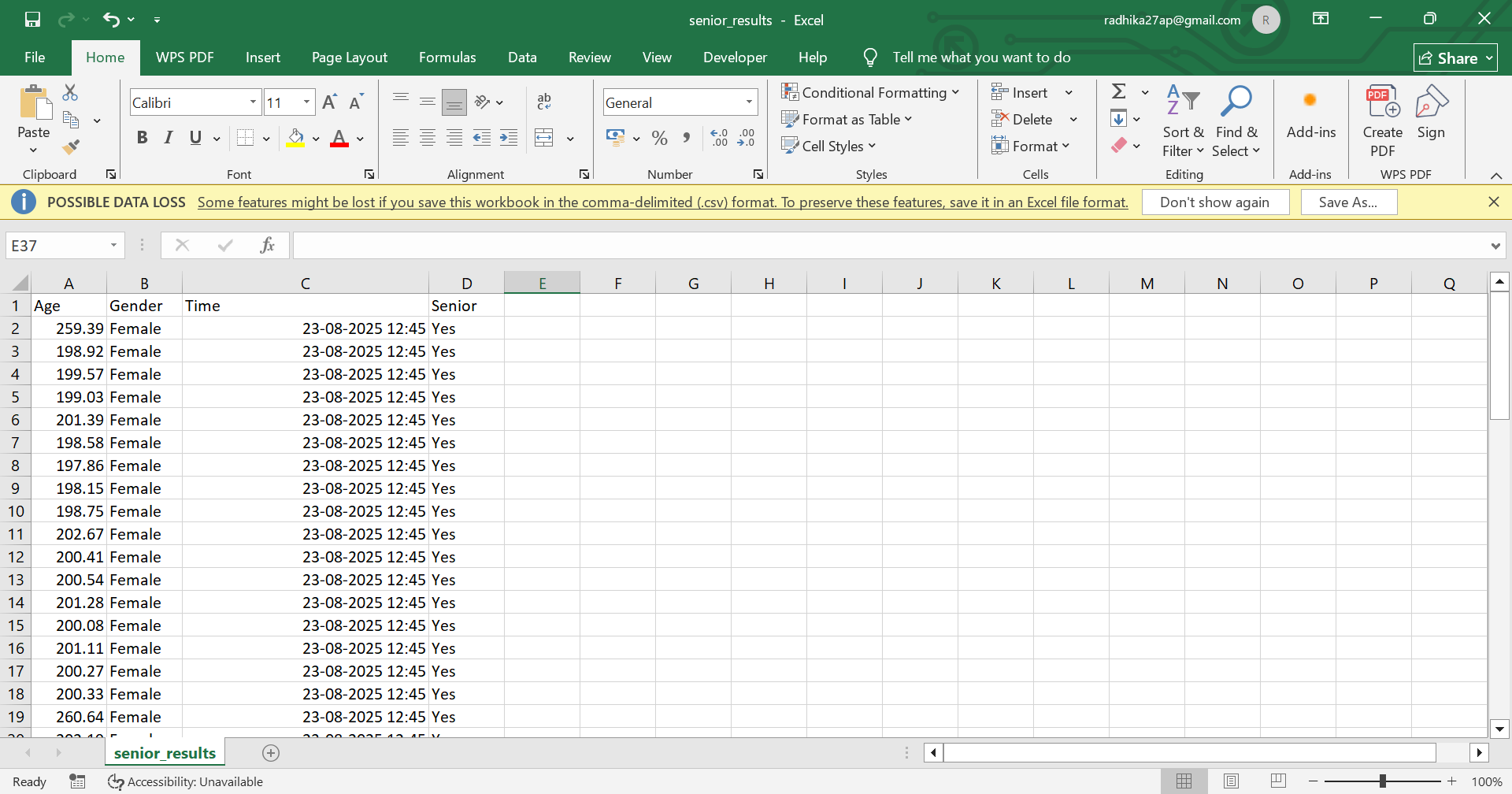


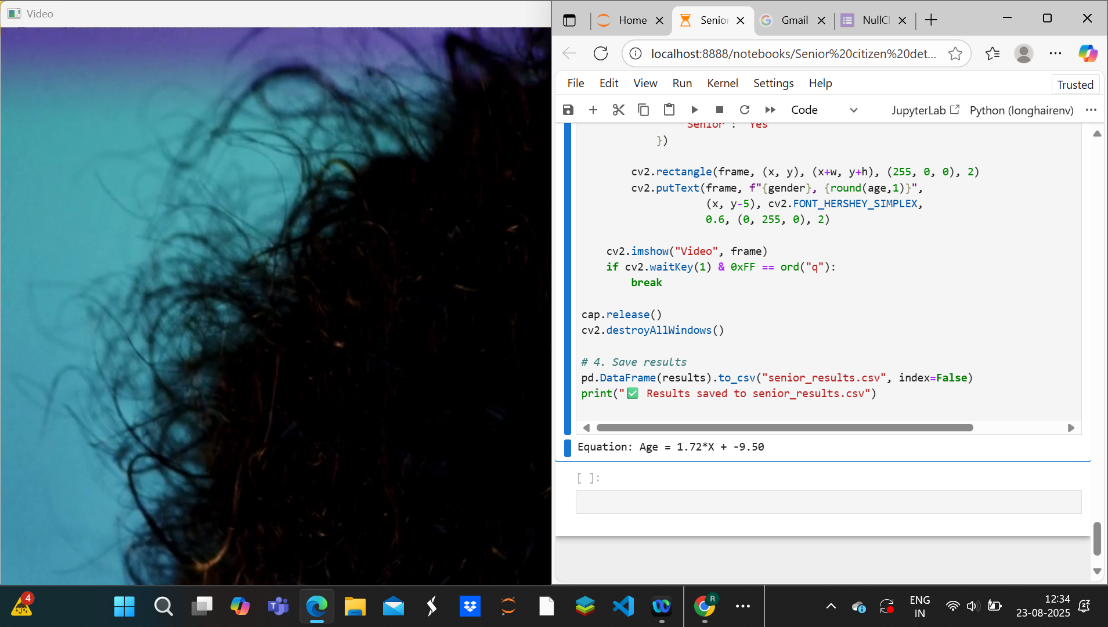
**Task 2: Senior Citizen Identification**

This task involved detecting multiple people from video/webcam feed and predicting their age and gender. Anyone above 60 was marked as a senior citizen, and I also saved their details with date and time in an Excel file. It was my first time combining real-time detection with external storage.



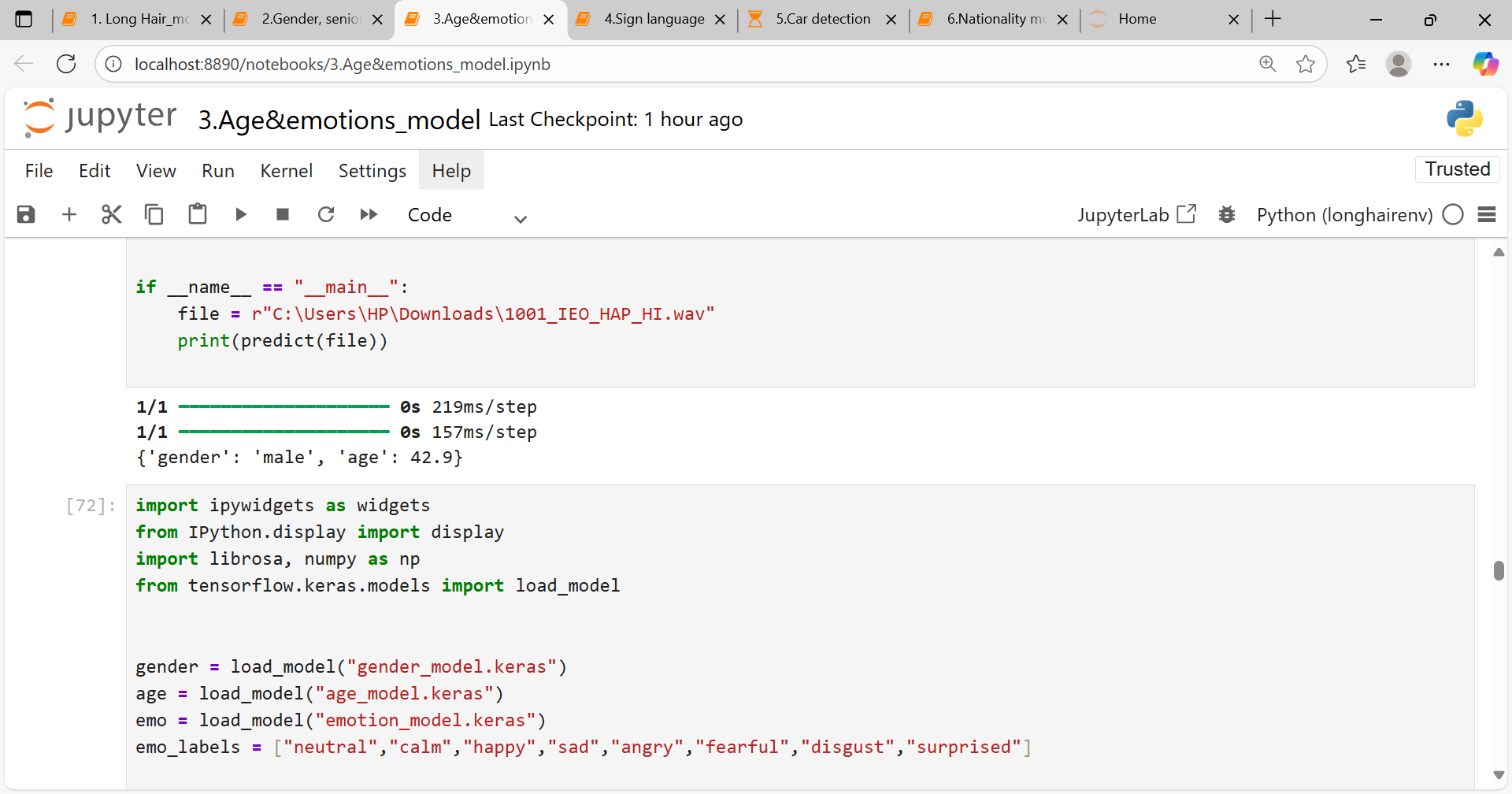


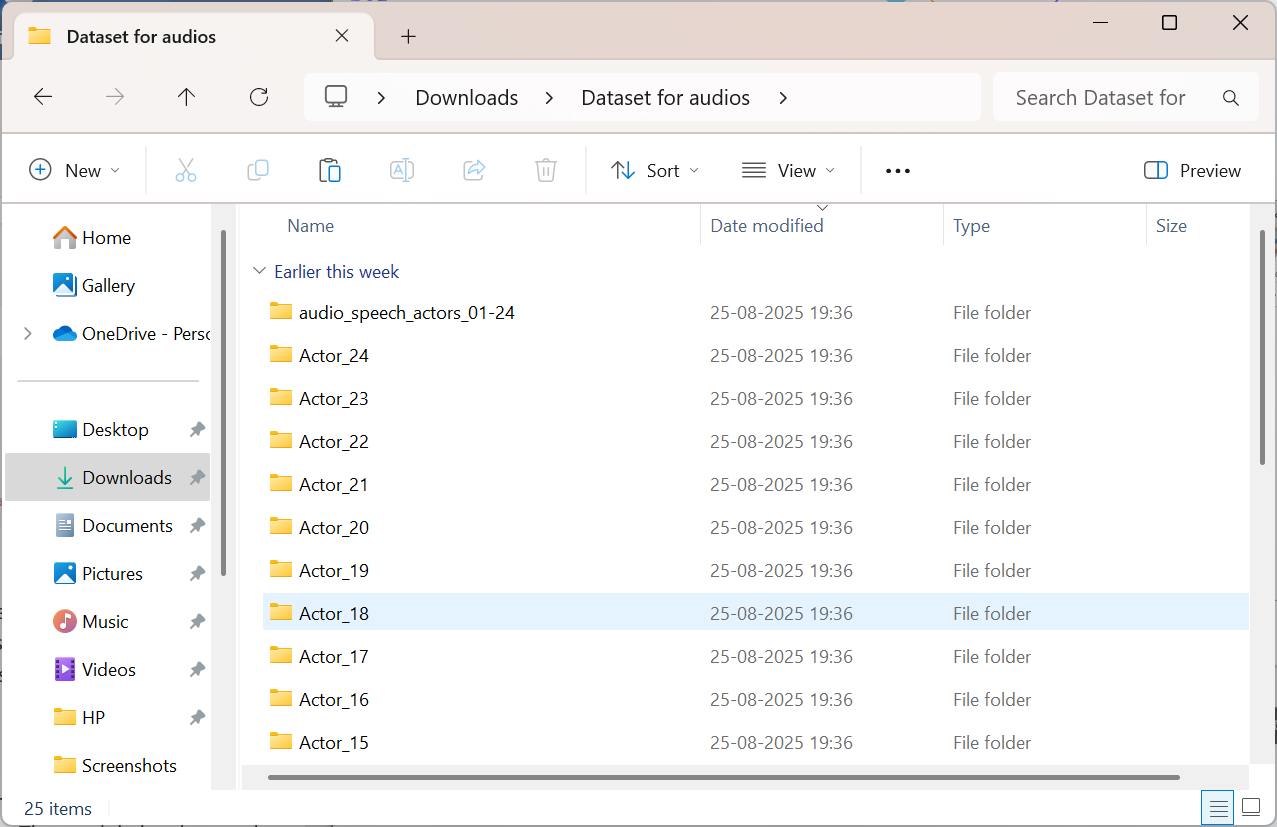




**Task 3: Age and Emotion Detection through Voice**

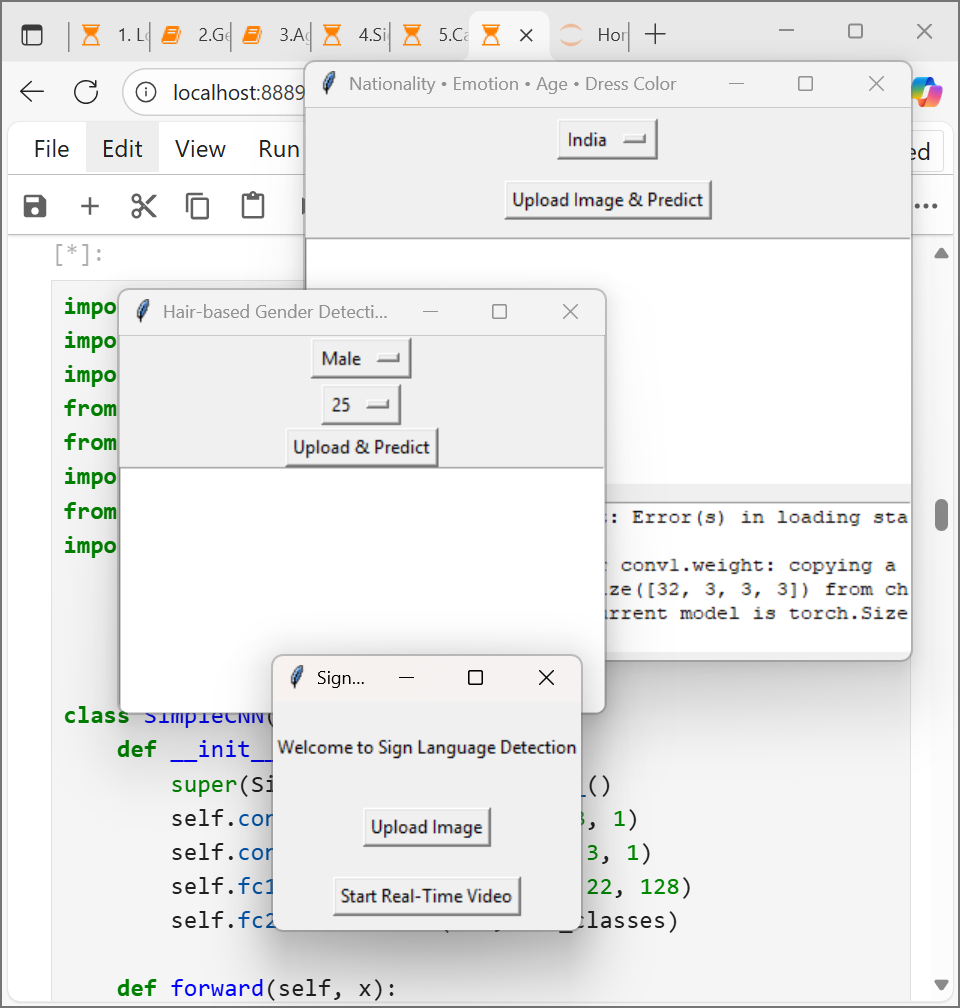
Here, I worked on detecting age and emotions using voice samples. The model only accepted male voices and rejected female voices with a clear message. If the detected age was above 60, the system also classified emotions. I created a small GUI to upload and process audio files.

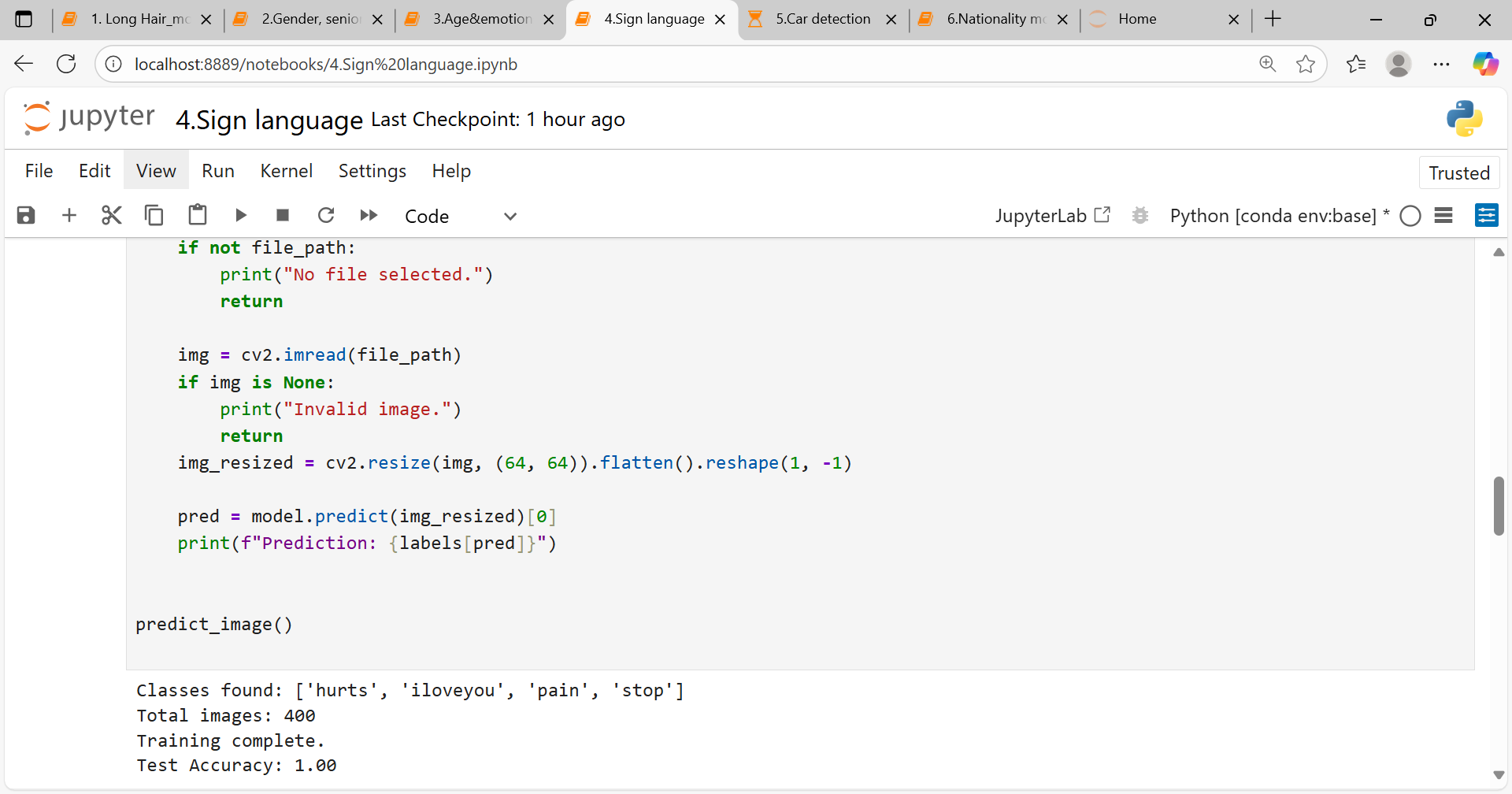




**Task 4: Sign Language Detection**

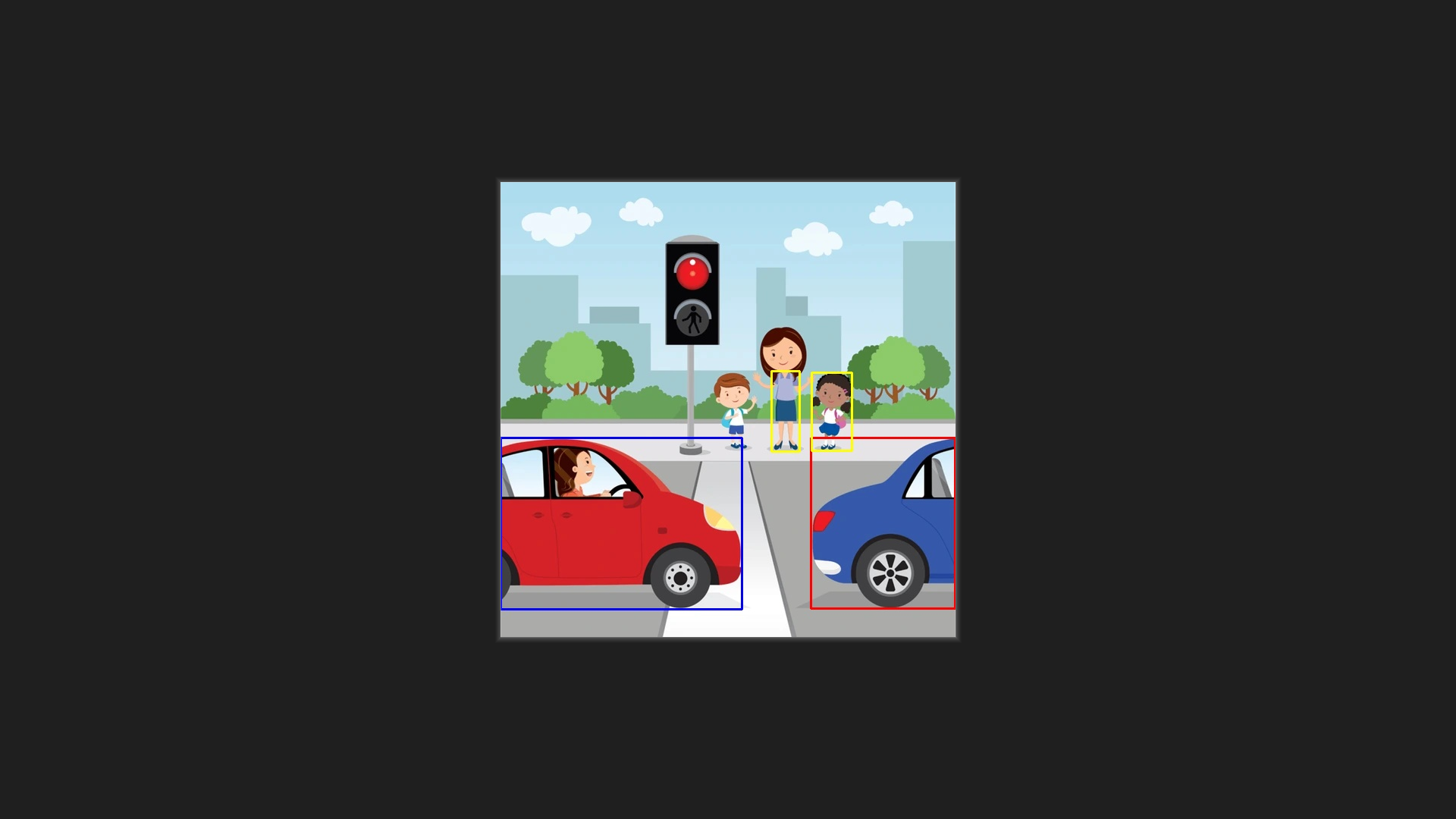
I trained a model to predict basic sign language gestures. The system worked both with uploaded images and live video. It was also restricted to operate only during 6 PM–10 PM. This gave me practice in linking time conditions with model predictions and providing both input and output views in the GUI.

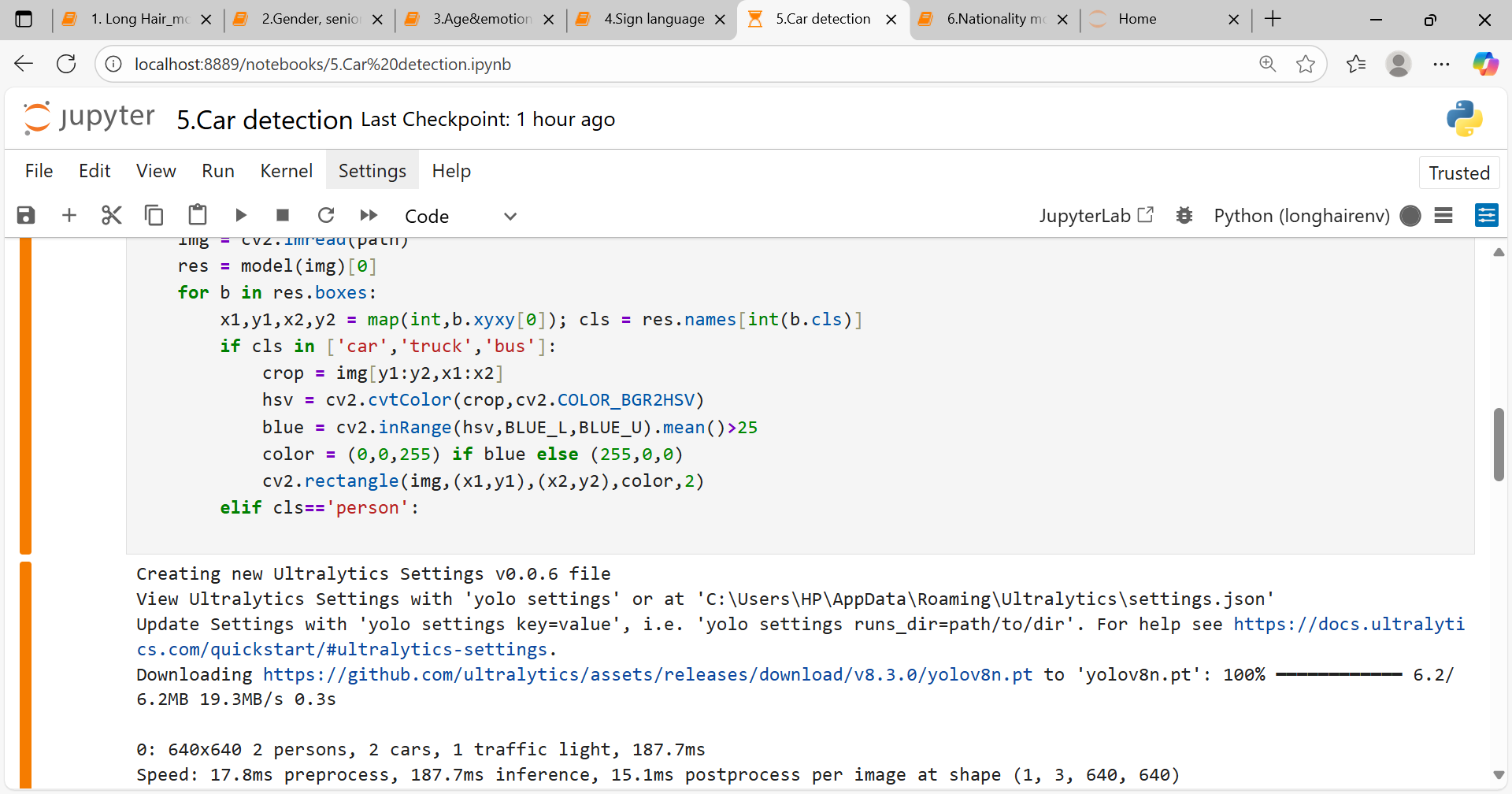




**Task 5: Car Colour Detection Model**

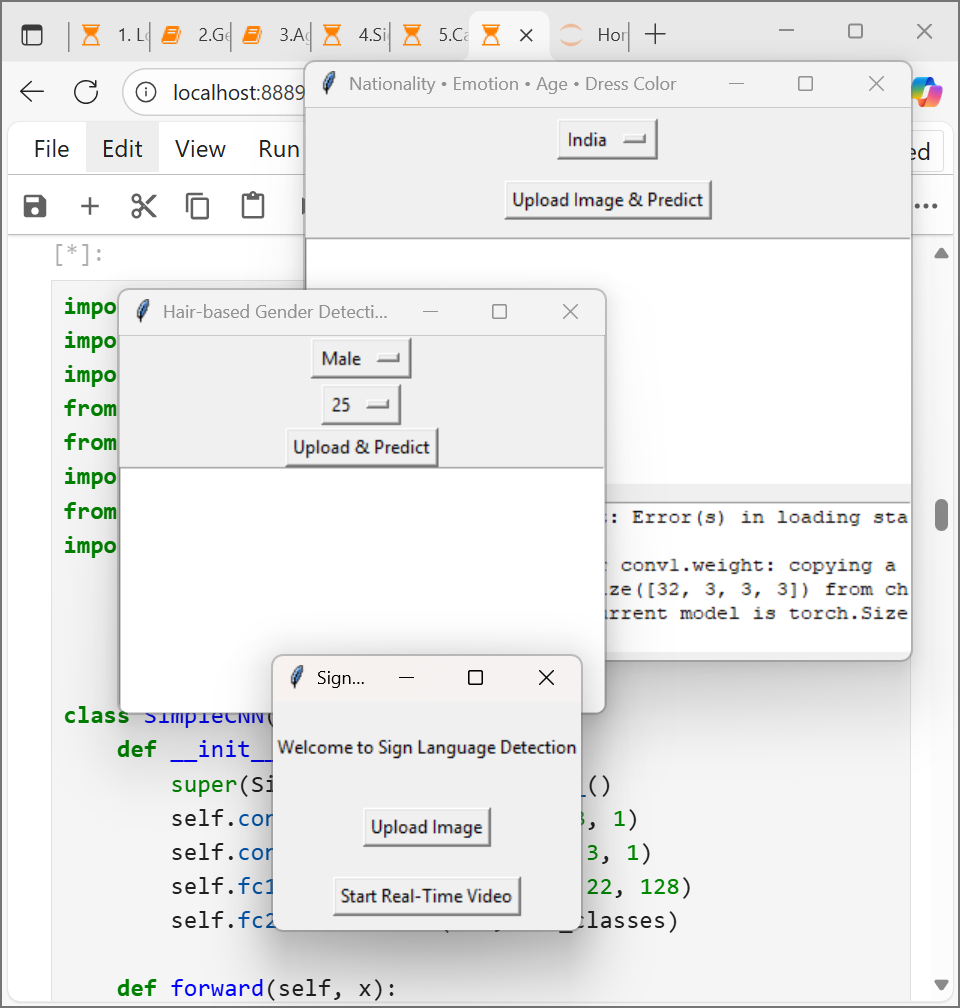
I used YOLO to detect cars, buses, and trucks in traffic. Blue cars were marked with red rectangles, while other cars got blue rectangles. The model also detected people at the signal and displayed their count. I added a GUI for image uploads and results, and a confusion matrix for evaluation.





**Task 6: Nationality Detection Model**

The nationality model predicted different features depending on the person’s nationality. Indians got predictions for age, dress color, and emotion, US nationals for age and emotion, and Africans for dress color and emotion. The GUI displayed both the input (uploaded image) and the output clearly.





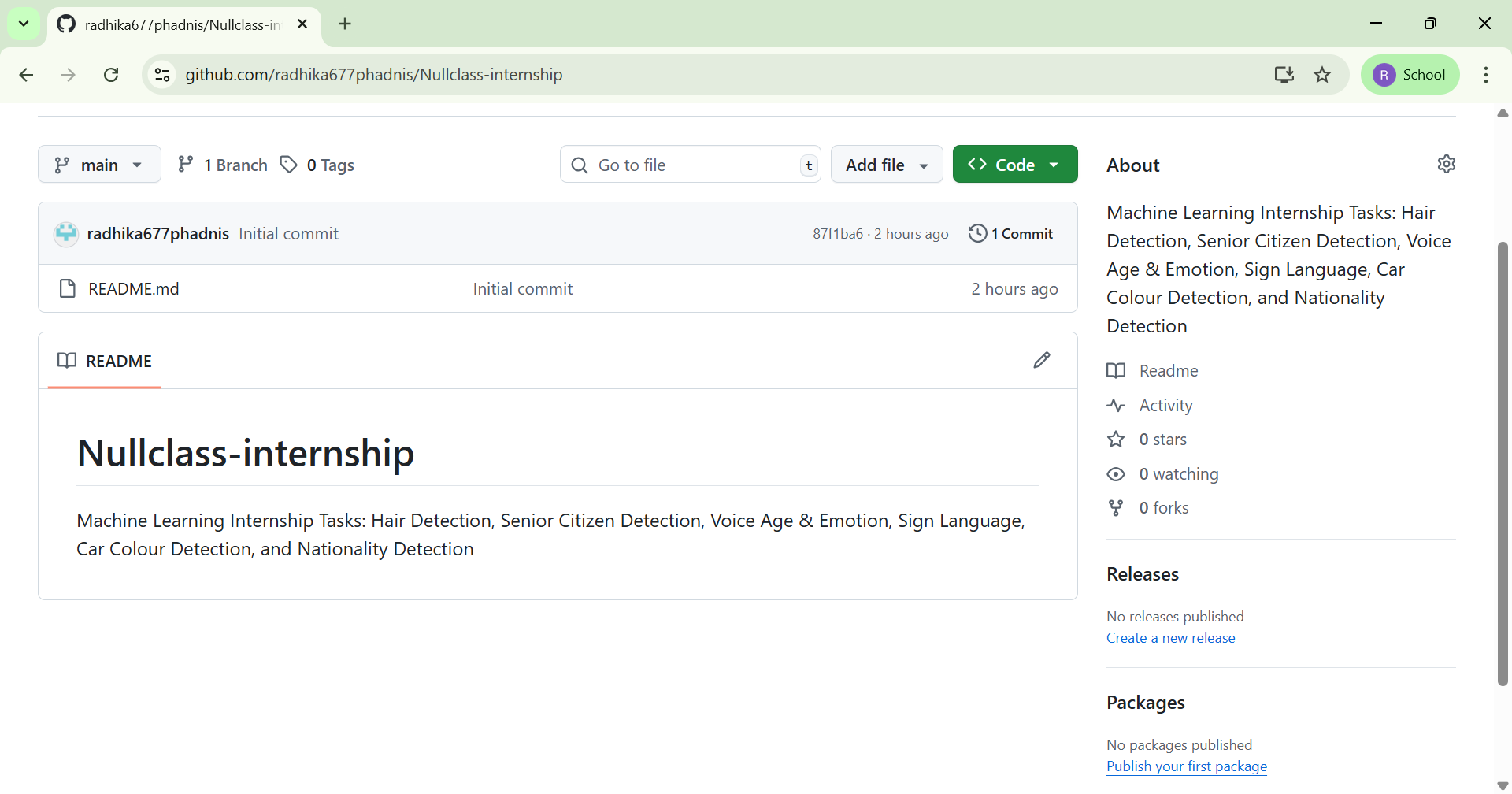
**Skills and Competencies**

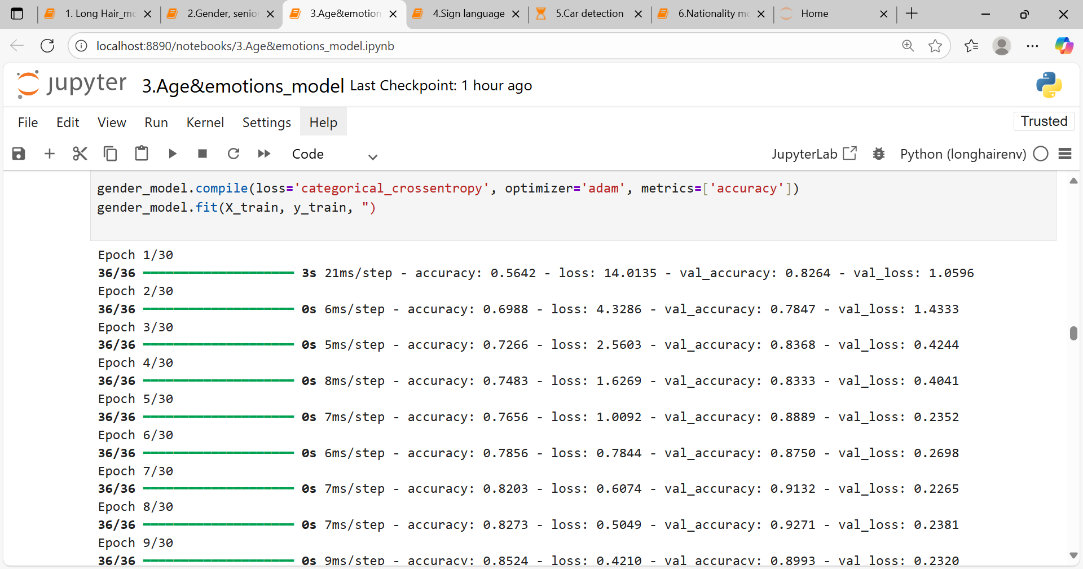
* Improved my Python coding skills and debugging ability.
* Learned to work with deep learning models, especially YOLO and CNNs.
* Gained confidence in GUI development (Tkinter).
* Practiced handling different types of data — image, video, and audio.
* Strengthened logic-building by adding conditions beyond model predictions.

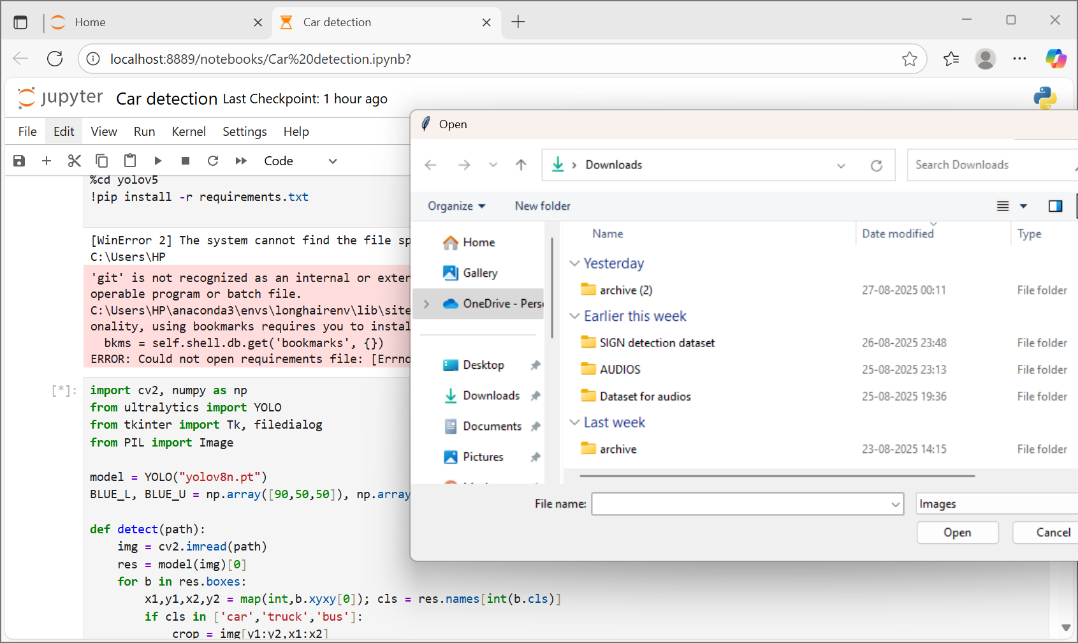
**Feedback and Evidence**

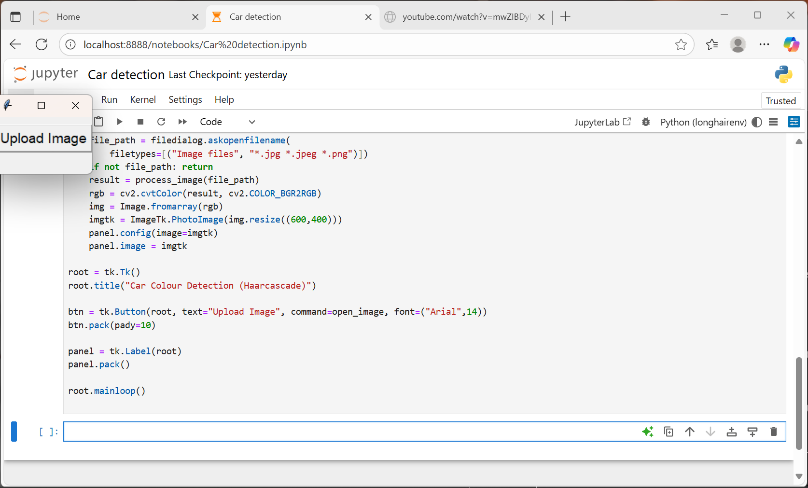
I tested each task with different inputs (images, voices, videos). The results were not always perfect, but I was able to see clear improvements after debugging. Saving outputs in Excel, showing predictions in GUI, and checking confusion matrices all provided evidence of progress.

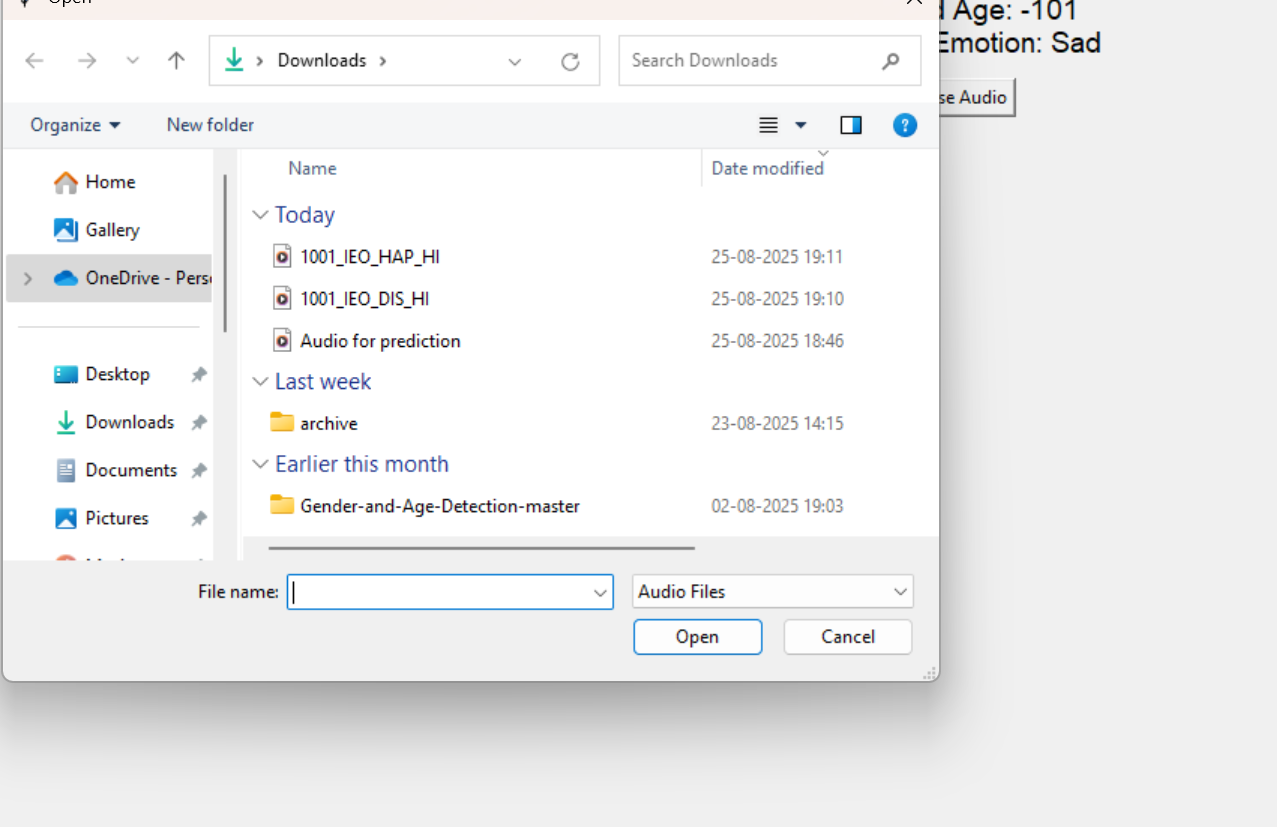


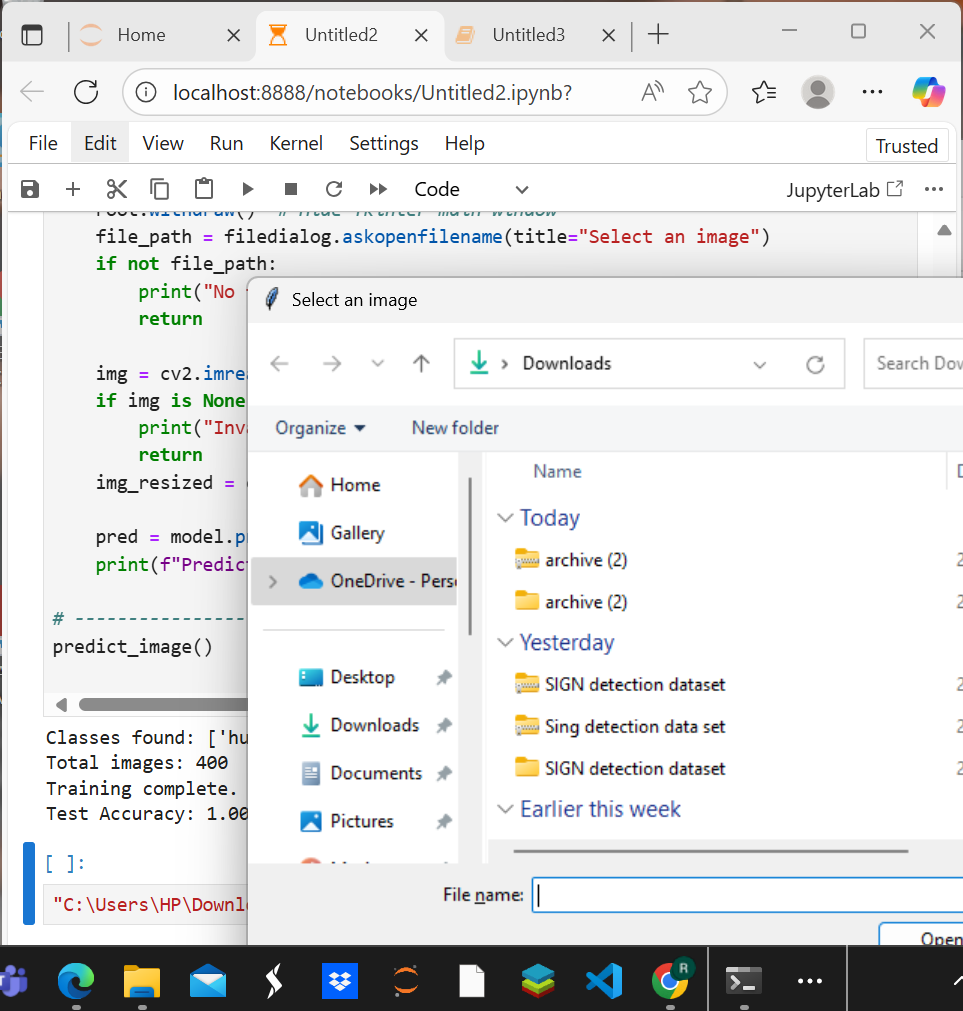












**Challenges and Solutions**

* At first, I struggled with handling real-time webcam data, but after trial and error, I managed to process it frame by frame.
* Integrating GUI with model predictions was tricky because of file handling, but breaking the code into functions helped.
* Accuracy wasn’t always high in the beginning, but adding more preprocessing and testing with sample data improved it.
* Managing large files was difficult, so I learned to upload them to Google Drive and link them in the README.

**Outcomes and Impact**

This internship gave me much more than just technical knowledge. I now feel confident in building small but complete ML projects on my own. I also learned how to present my work in an organized way using GitHub and documentation. Most importantly, I now approach problems with a balance of technical accuracy and practical usability.

**Conclusion**

Overall, this internship was a valuable experience. Each task challenged me differently, from logic design to GUI building and from real-time input to output storage. I made mistakes, but I also learned how to solve them myself. This experience has definitely improved my technical foundation, boosted my confidence, and given me a clearer idea of how to apply machine learning in real life.