Titrate Installation Guide

Table of Contents

Document Control

Revision History

Version	Date	Updated By	Description
V1.0	11-02-2024	Nikhil	Ubuntu & Python Installation
V1.1	08-03-2024	Ramesh	Database API & UI Setup

Document Approval

Date	Name	Role	Status
	Ramachandra Damera	Manager	Approved

Objective

This document provides a step-by-step guide to install and configure the application on a client environment. It covers the backend API and frontend UI setup, including database configuration using PostgreSQL and environment setup for both components.

➢ Prerequisites

- Git installed on the machine
- A 64-bit version of Windows 11
- VS Code
- Node.js and npm
- Ubuntu above 22.0
- Python 3.10+ and pip in Ubuntu
- PostgreSQL 13+ installed and running
- Access to Git repositories for API and UI
- An active internet connection

➢ Database Setup

Step 1: Connect to Your Server

- Launch **pgAdmin** and expand the **Servers** panel.
- Right-click your server (e.g., *PostgreSQL 14* or *localhost*) and ensure you're logged in as a role with **CREATE** permission (usually postgres).

Step 2: Open "Create Database" Dialog

- Under your server node, right-click **Databases** → select **Create** → **Database.**

Step 3: General Tab – Define Basics

- **Database**: Type your desired name (e.g., my_first_database).
- **Owner**: Choose the role (default: postgres).
- **Check the Port**
- **Give the host name, password and user name.**

Note: Remember the credentials given here.

Step 4: Save & Finish

- Click **Save** to create the database. The new entry will appear under the **Databases** node.

➢ Get the pull from GIT for API and UI respectively.

https://github.com/Brase-Technologies-Pty-Ltd/Mitte-Analytics.git

➢ Configuring and Running the API

- Open the file manager and navigate to ***Titrate\api***
- Open Command Prompt from the navigated path.
- Open .env file and configure the data base credentials as noted while creating a database.

```
# Environment
NODE_ENV=development

# Database Configuration
PORT=4000
DB_USERNAME=postgres
DB_PASSWORD=postgres
DB_NAME=imprestStock_db
DB_HOST=localhost

# Real SMTP (production)
SMTP_HOST=mail.kloners.in
SMTP_PORT=465
SMTP_SECURE=true
SMTP_FROM=stock_isms@kloners.in
SMTP_USER=stock_isms@kloners.in
SMTP_PASS=Rameshaa@16

# Dev SMTP (Papercut - runs on localhost)
DEV_SMTP_HOST=13.126.10.72
DEV_SMTP_PORT=1025
DEV_SMTP_SECURE=false
DEV_SMTP_FROM=local@devmail.test

# Test customer email
CUSTOMER_EMAIL=ayyalarameshnaidu@gmail.com

# Default Admin Credentials
DEFAULT_ADMIN_EMAIL=admin@admin.com
DEFAULT_ADMIN_PASSWORD = Admin@123

# Purchase Order Folder
PO_FOLDER=
"D:\Projects\Projects\brasetech\source\api\purchase_orders"
```

- Give the path for PO_FOLDER where you wish to save the file.
- Open the terminal and Run the command “npm install”.
- Run the command “***npm start***” in the command prompt. // command to start the API.

Note: The Titrate application will be running on the default port localhost: 4000

On running the API, default user will be created and added to the database. You can use the same to login and add new users and any other data respectively to the application.

DEFAULT_ADMIN_EMAIL=admin@admin.com

DEFAULT_ADMIN_PASSWORD = Admin@123

For confirmation you will be displayed with the message with port and IP address.

➢ Setting Up and Running the Web Server

- Go to file manager and navigate to ***Titrate\api_ui***
- Open command prompt from the navigated path
- Open .env file and configure URL and App name and save.

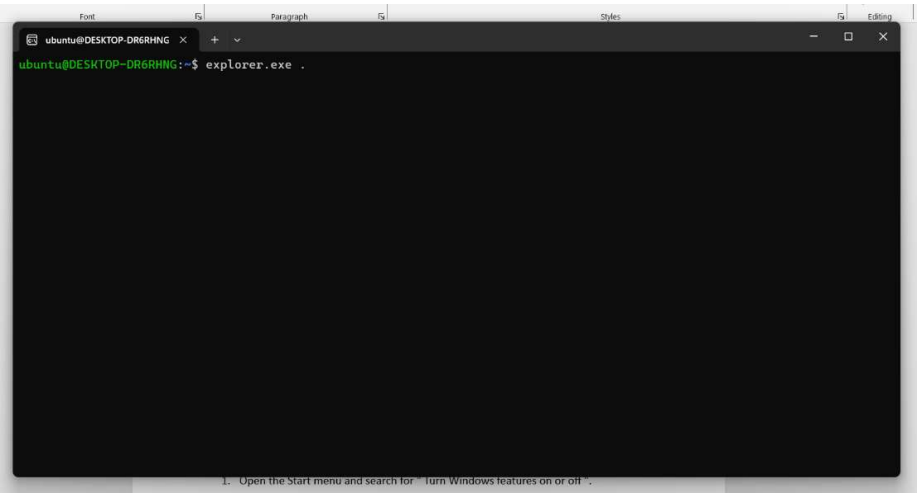
- Run the command “npm install”.
- Run the command “***npm run dev***” in the command prompt // command to start the web

Note: The Titrate application will be running on the default port localhost: 80

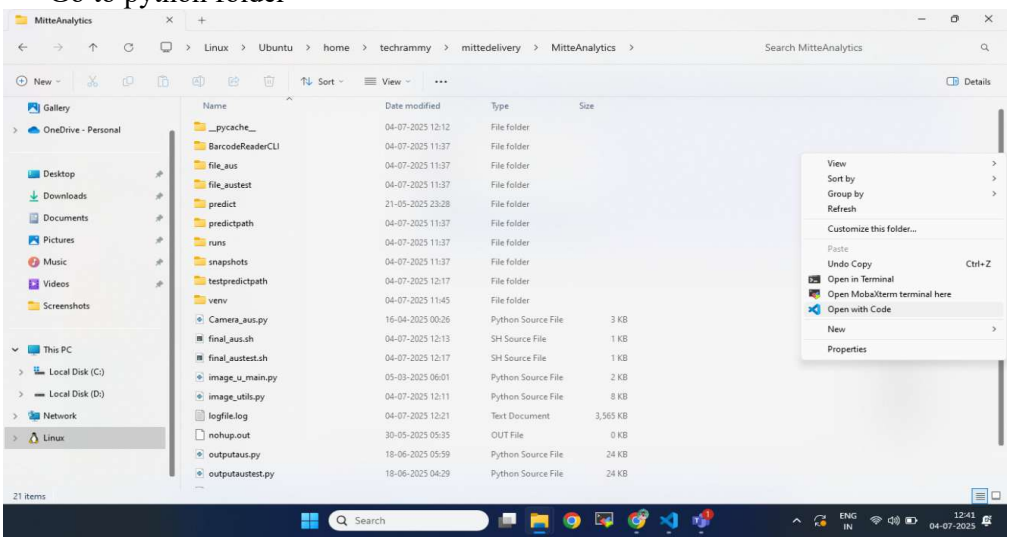
➢ Run Python Server

Steps to run Python Code:

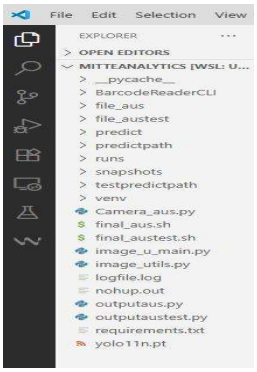
- Open Ubuntu and type “explorer.exe .”



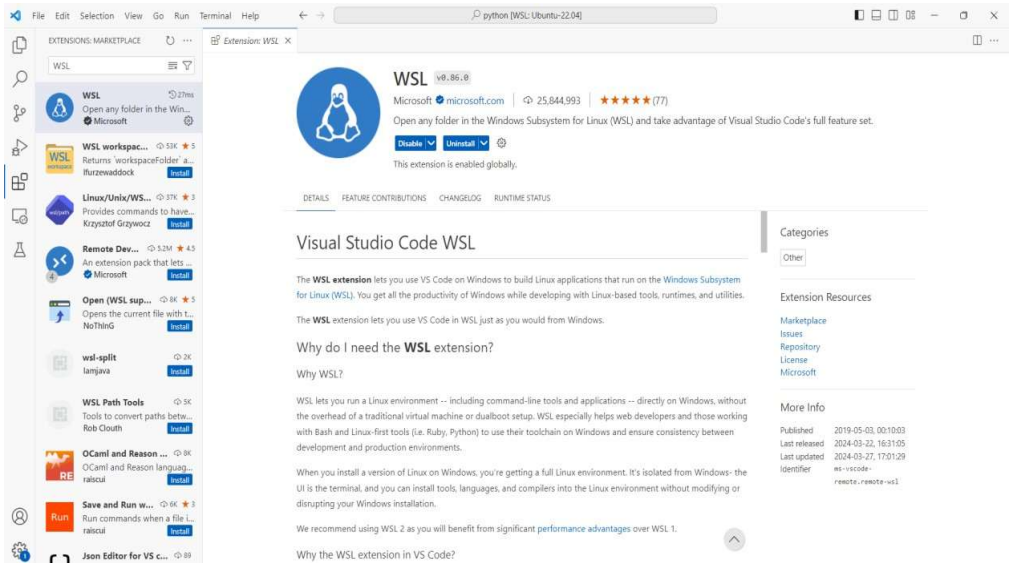
- Pull the code from git repo
- Go to python folder

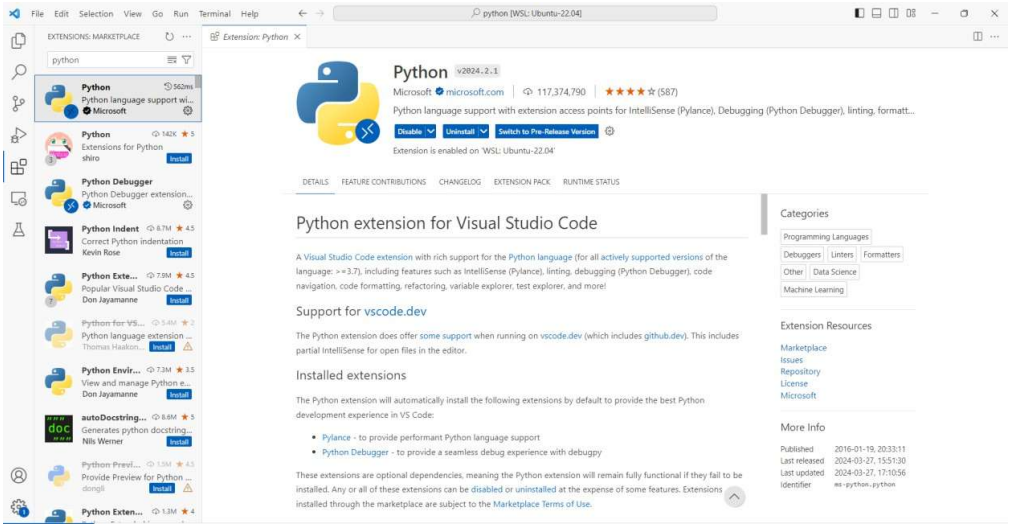


- Open the folder in VS python code you will be able to see the files on the left side.

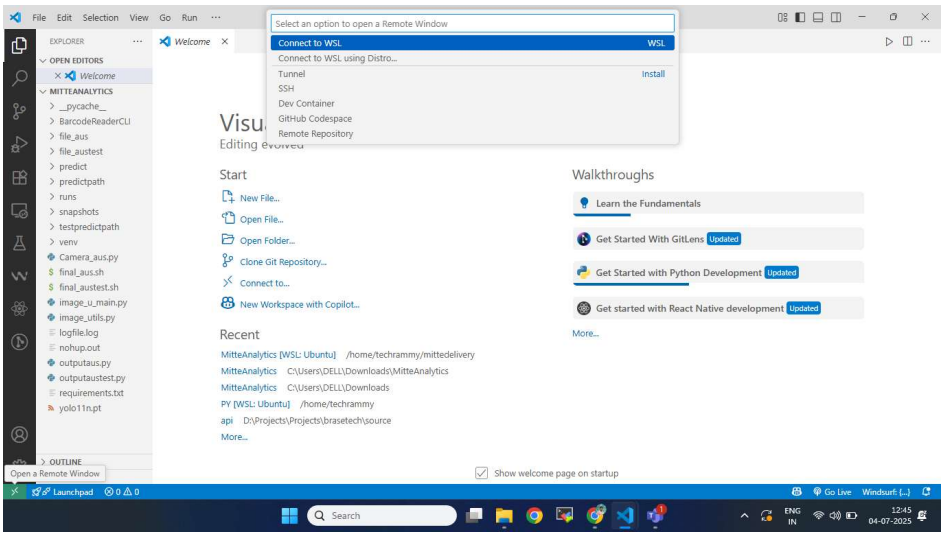


- Now open extensions using command “Ctrl+Shift+x” and download WSL, python

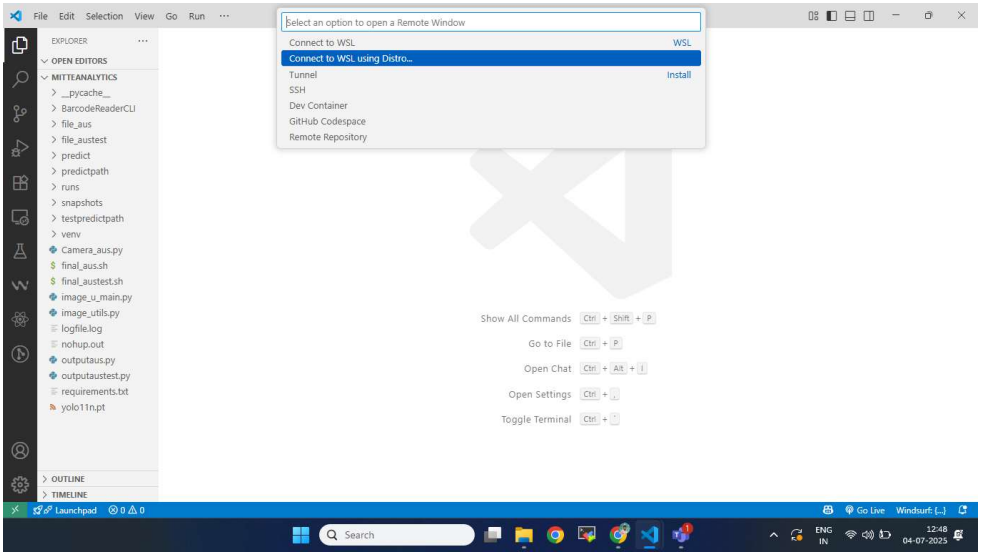


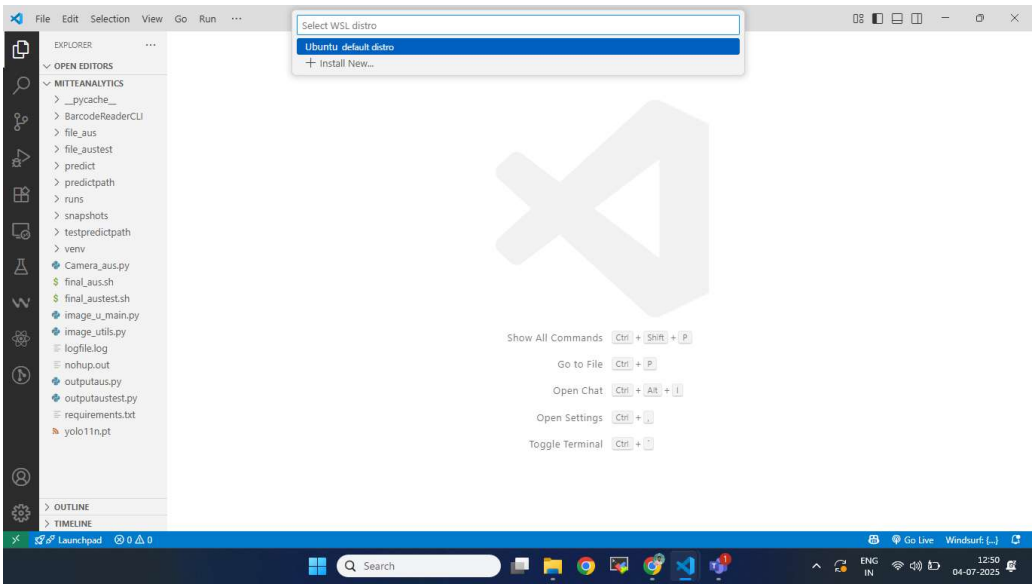


- Click Green button at the left bottom of the screen to open a remote window

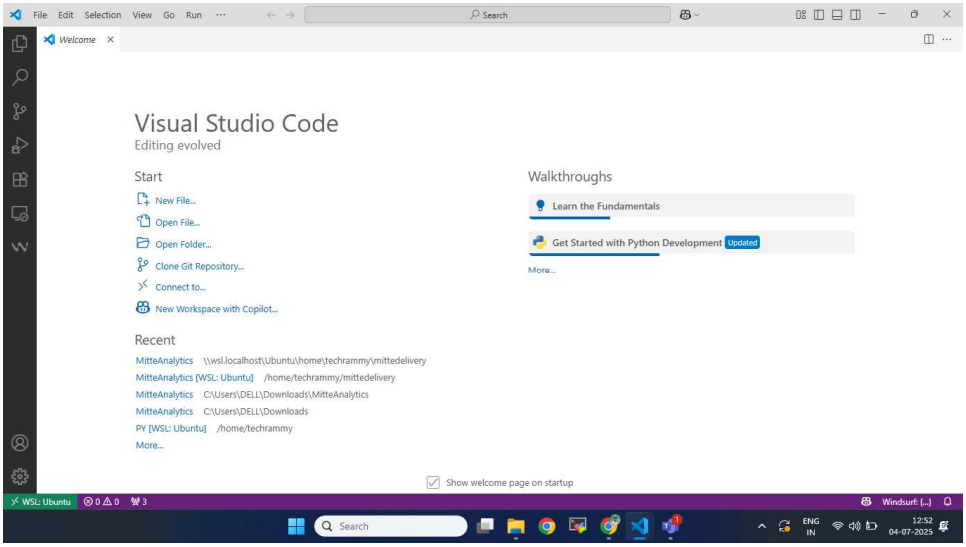


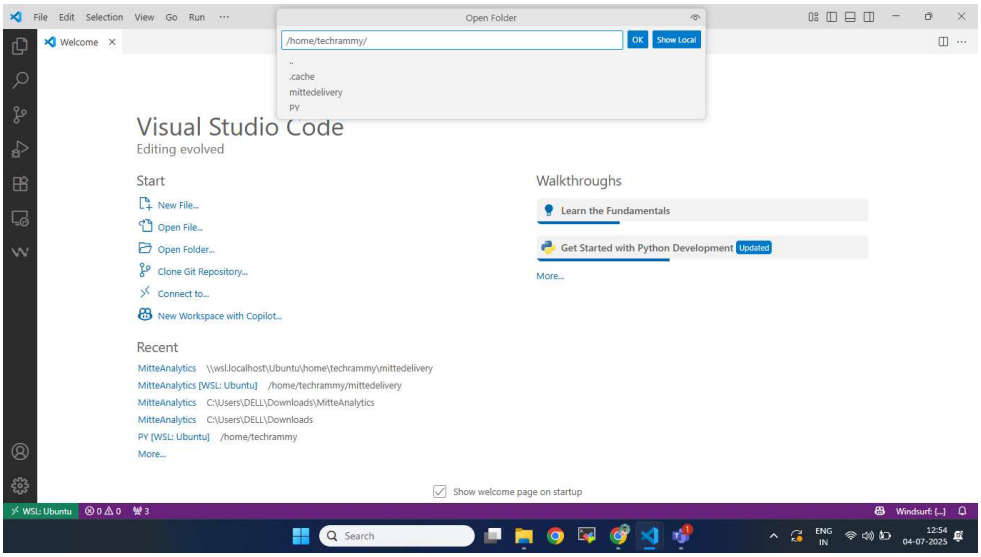
- A search filter will be opened at the top of screen. Select “Connect to WSL Distro…..” and then select “ubuntu”



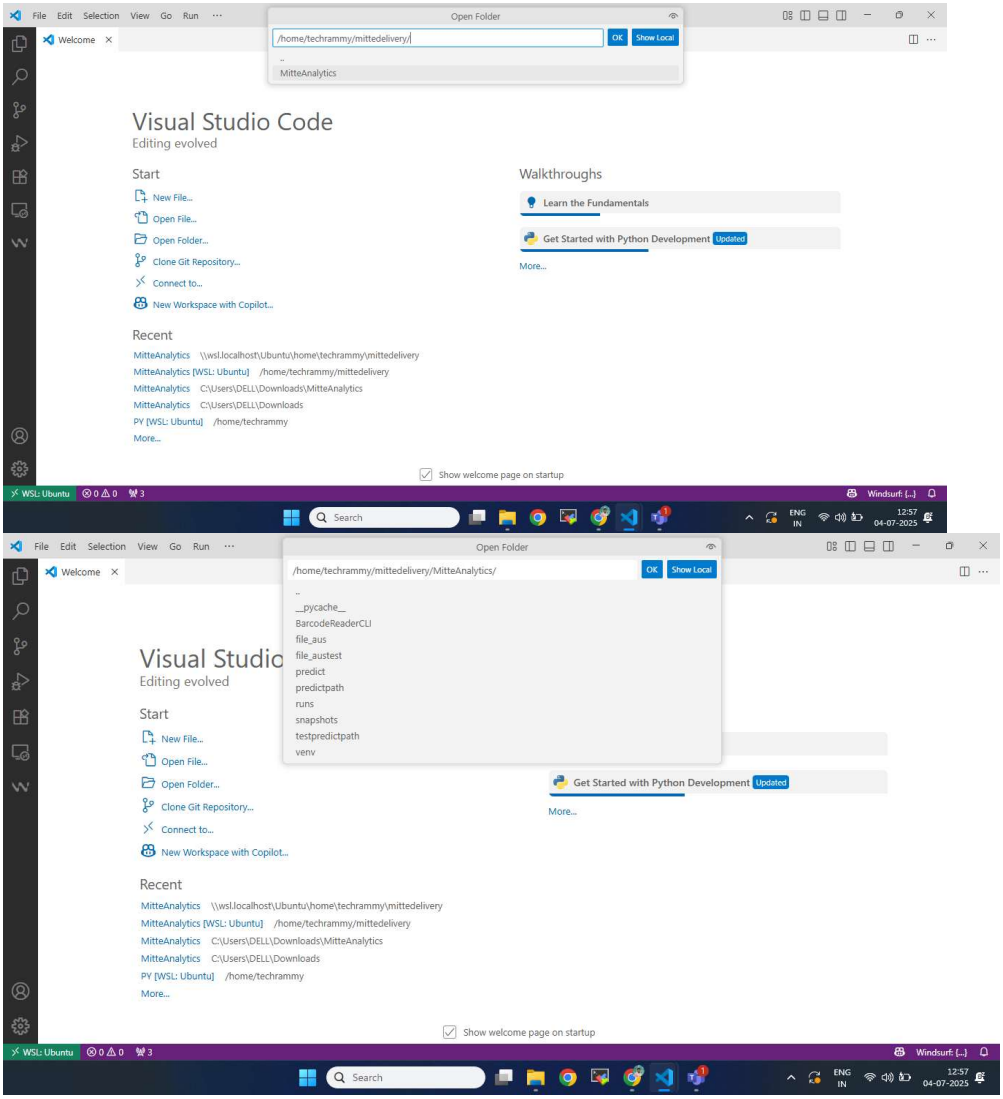


- It may take some time to set up VS Code in Ubuntu. Once completed, you will notice 'WSL: Ubuntu' displayed in the bottom-left corner of the window.

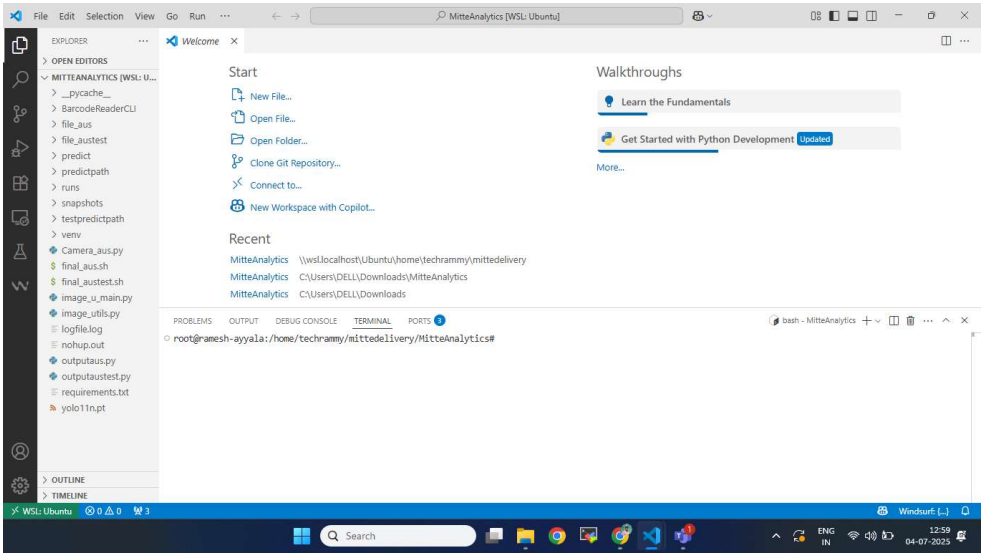




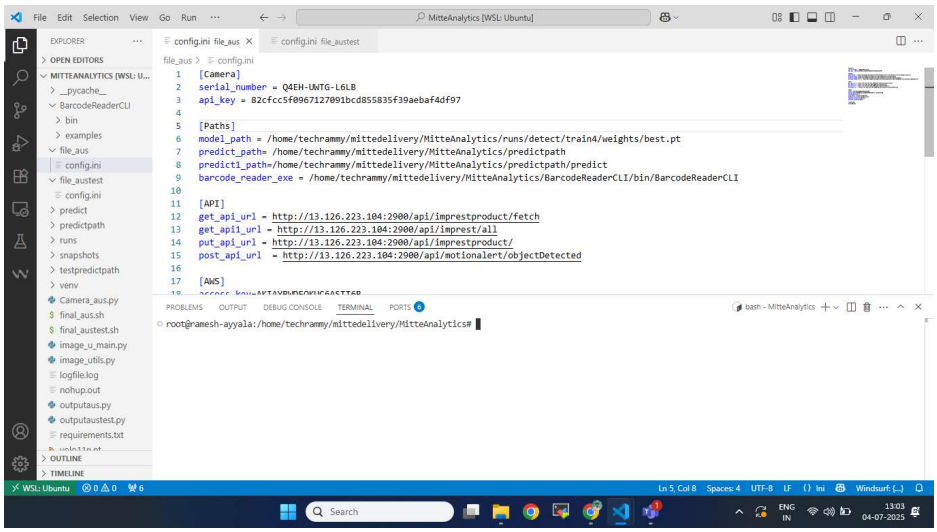
- Open and Select the folder and click “OK”



- Make sure that the all files are visible now on your left side navigation bar.

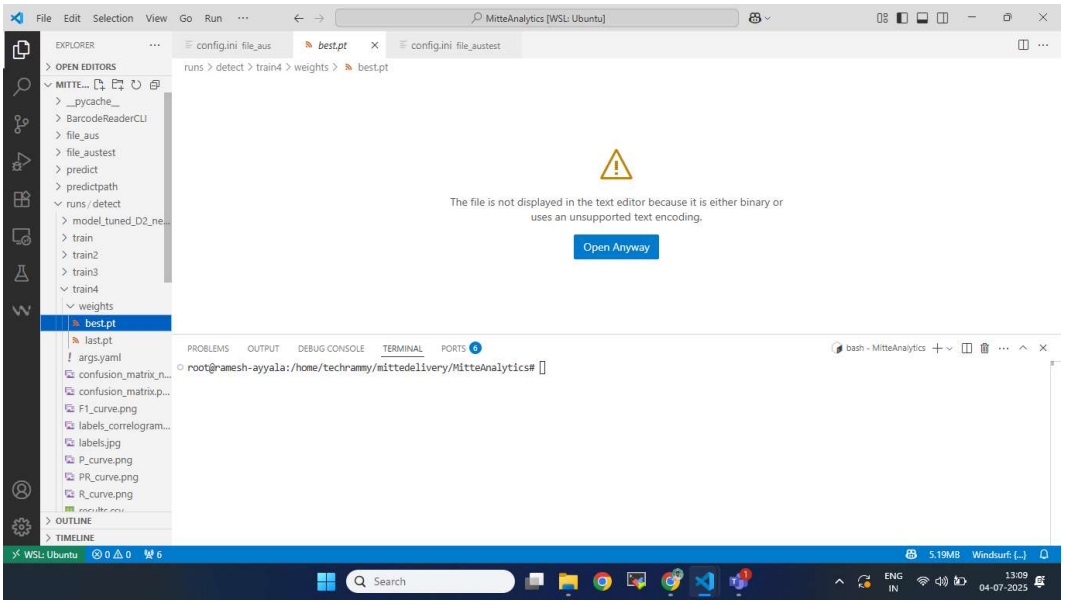


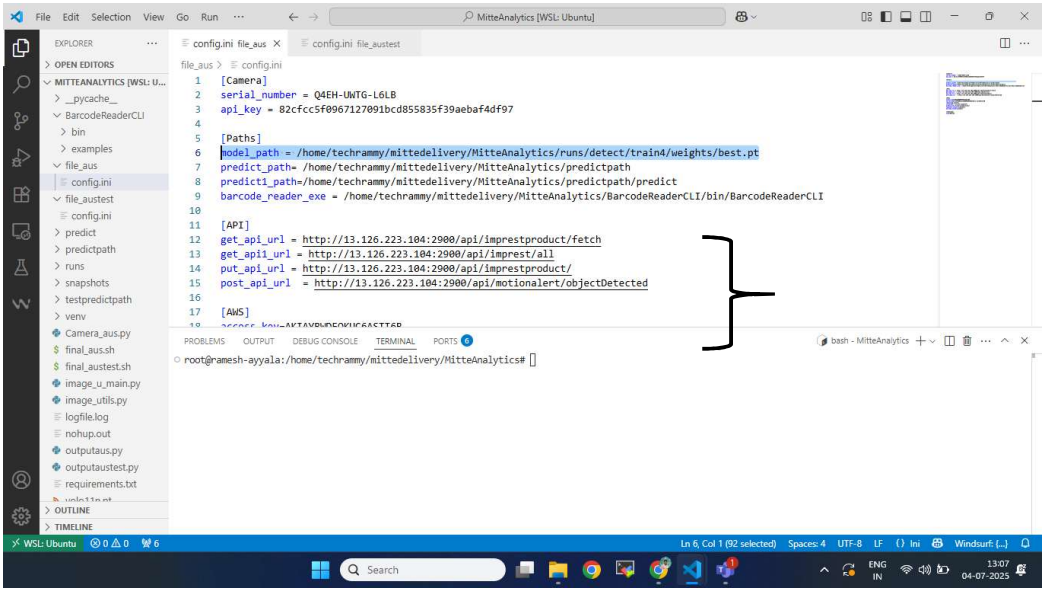
- Open config.ini which is present in files named “file_aus” and “file_austest”



- Change the Serial number and API key that matches with the camera running (If any).

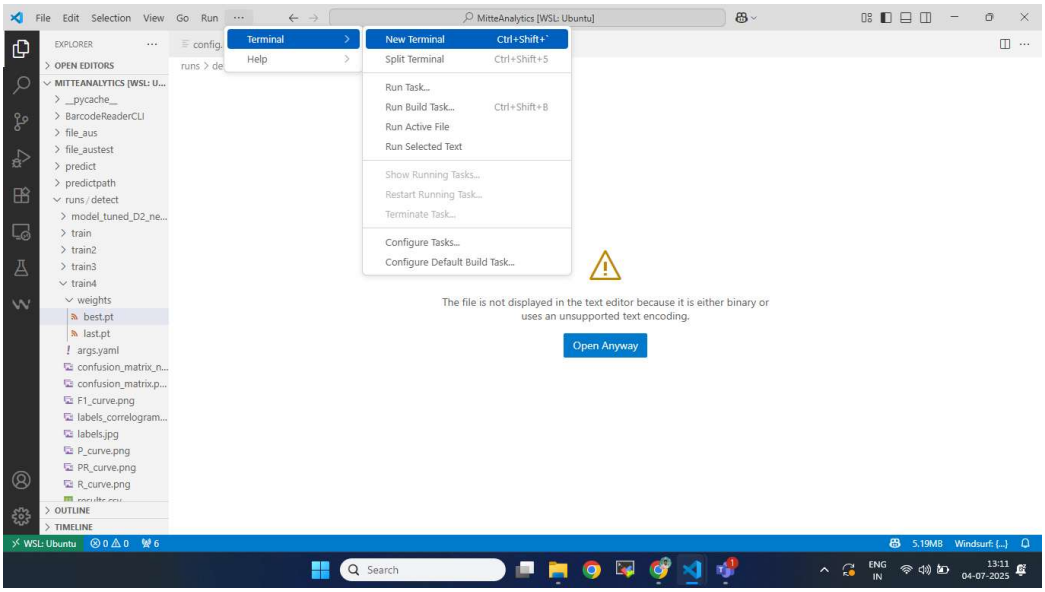
- Change the paths of [Paths] in both the config files of file_aus and file_austest folders. For model_path copy and paste the path of best.pt from weights->weights->best.pt. And follow the same for all paths in [Paths].





Note: Here, backend code is getting integrated with frontend using [API] section from config.ini file as in above image

- To open terminal select the …-> Terminal->New Terminal from menu bar



NOTE: There are 2 ways to run the code,

a) Create a venv (virtual environment) & run initially for installing all required packages and then run the code

- Create a Virtual Environment

To create a virtual environment named newvenv, run the following command: "python3 -m venv newvenv"

Activate the Virtual Environment

After creation, activate the environment using: "source newvenv/bin/activate"

Install Required Packages

Install dependencies from the requirements.txt file using: "pip3 install -r requirements.txt"

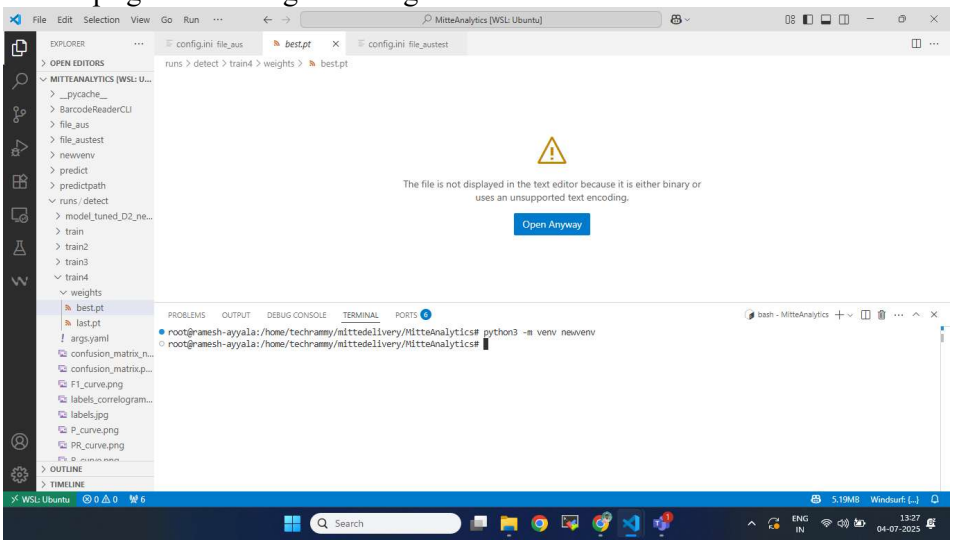
Note: If the above command fails or throws an error, try specifying the full path: "pip3 install -r path/to/requirements.txt"

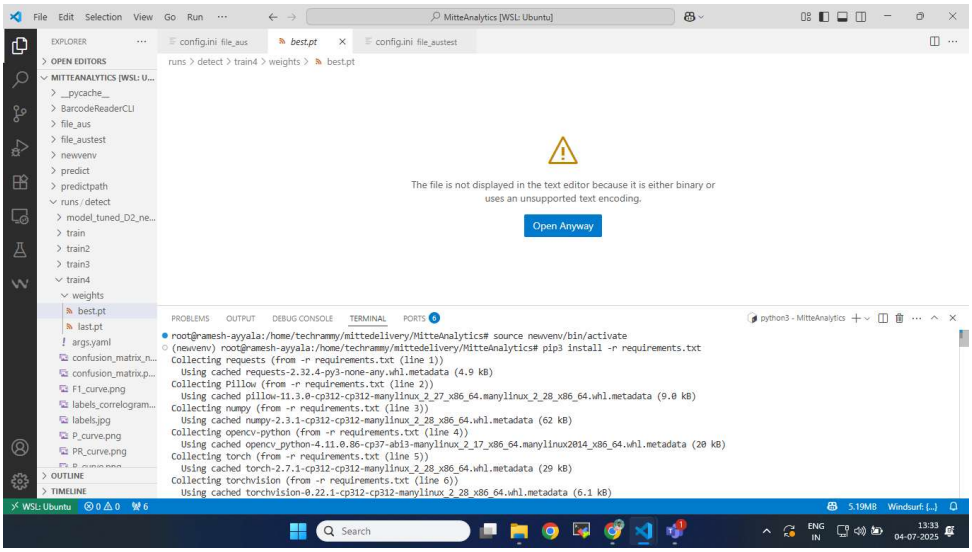
Install System Dependencies (if needed)

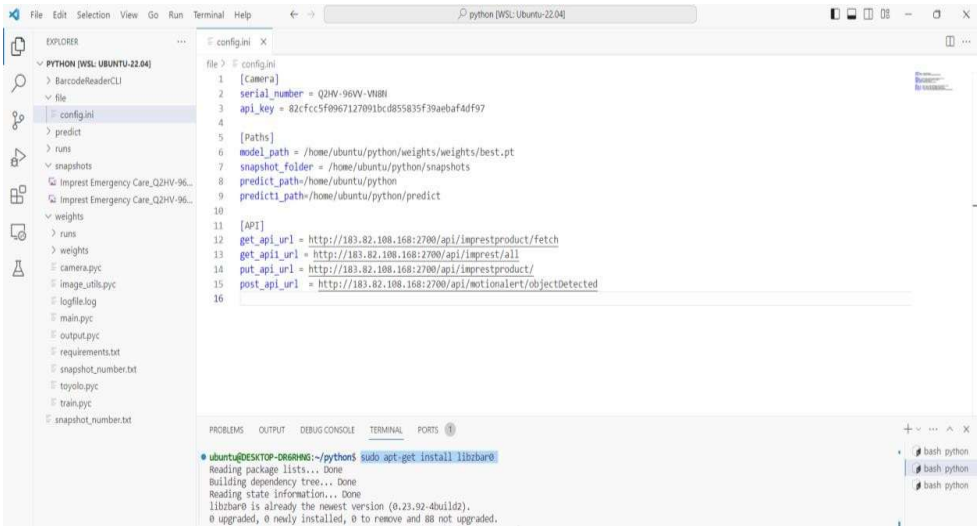
If errors occur while installing Python packages, run the following system-level installations:

"sudo apt-get install libzbar0"

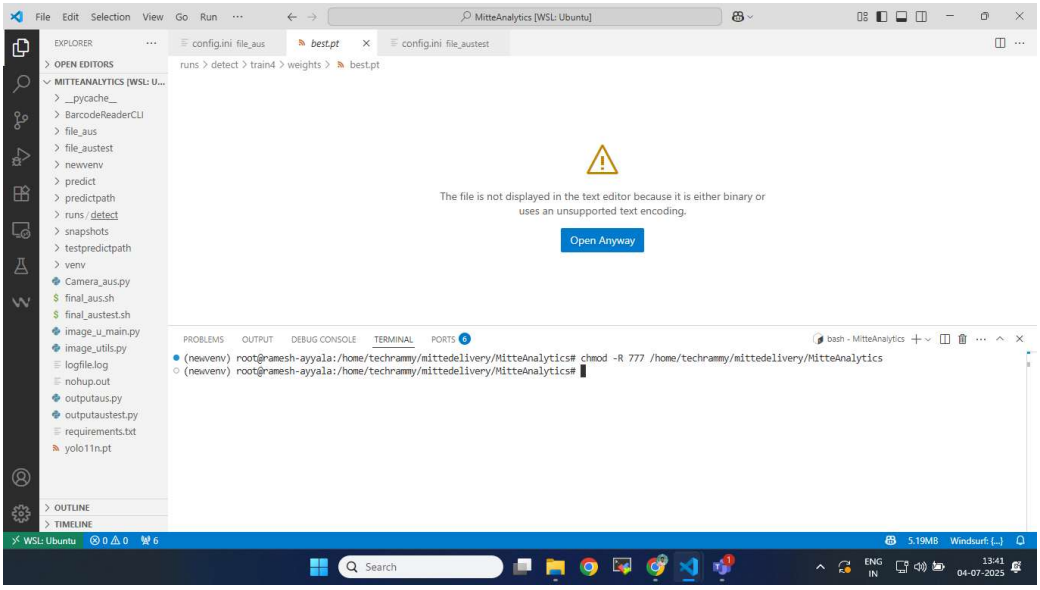
"sudo apt-get install libgl1-mesa-glx"



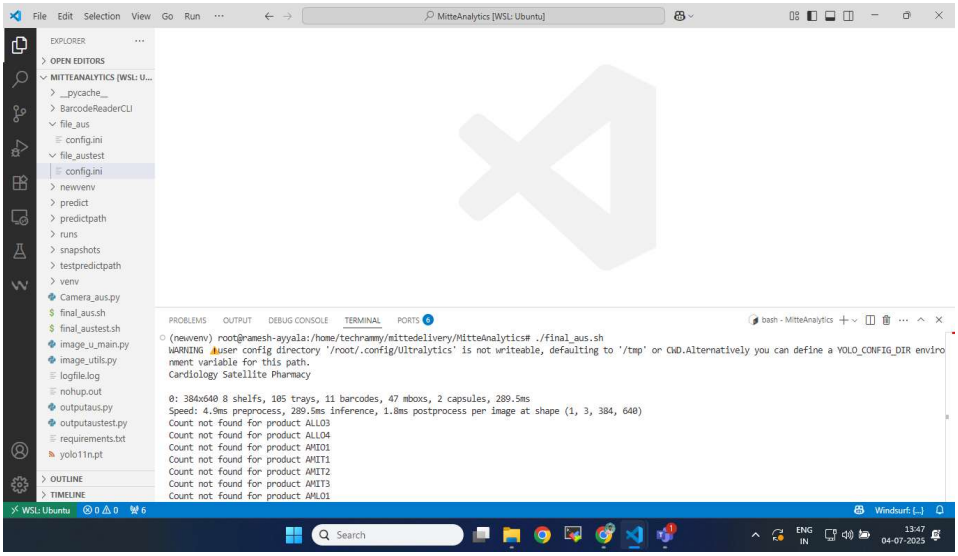


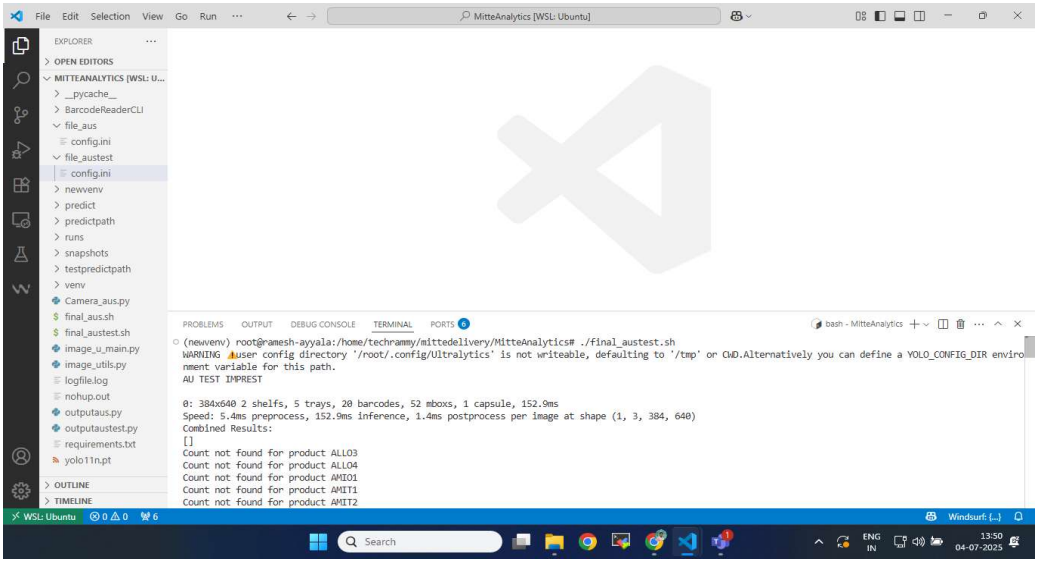


- Give permissions for the folder we have placed in by running command: “chmod –R 777 (path/to/folder)”



- Run the command in the terminal “./final_aus.sh” for running the camera of hospital imprest and run command in new terminal “./final_austest.sh” for running camera of AU test imprest.





This will get us the count and also will update the count in the UI Dashboard.