Wi-Fi Network Scanner

Introduction

Wi-Fi networks are everywhere, but users often struggle to connect to the strongest and most secure one. This project aims to solve this problem by building a **real-time Wi-Fi network scanner** that lists all nearby wireless networks with detailed info such as **signal strength, channel number, and security type**.

Unlike traditional CLI-based scanners, this tool provides a **user-friendly GUI**, supports **CSV export**, sends **notifications on new network detection**, and **auto-refreshes every 5 minutes** — making it practical for everyday use and showcasing real-world development skills.

Objective

- To scan and display all nearby Wi-Fi networks.
- To allow users to identify the **strongest and safest** networks.
- To notify the user when new networks are detected.
- To offer data export functionality in .csv format.
- To enhance recruiter appeal through professional GUI and automation features.

Tools & Technologies Used

Tool/Library	Purpose
Python 3.x	Core programming language
pywifi	Wi-Fi network scanning & interface handling
tkinter	GUI design
plyer	Desktop notifications
csv module	Exporting scanned data to file

Features Implemented

Wi-Fi Scanning Logic

Scans all nearby networks, captures details like:

- SSID (Network name)
- Signal strength (dBm)
- Security type (Open, WPA, WPA2)
- Channel number (converted from frequency)

Real-Time GUI (Tkinter)

- Sleek **dark mode** interface
- Dynamic table with network details
- Buttons for manual refresh and export

Auto Refresh (Every 5 minutes)

Automatically refreshes network list using Tkinter's after() function.

Desktop Notification

Detects **new SSIDs** compared to previous scans and sends a **system notification** using plyer.

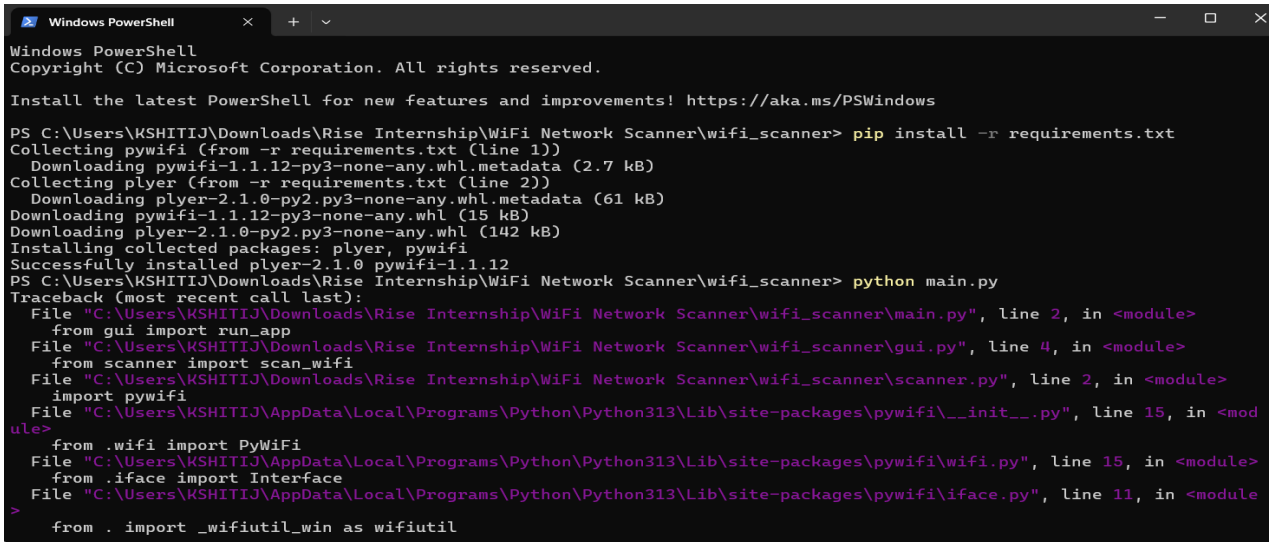
Export to CSV

Enables users to save scan results into a structured CSV file.

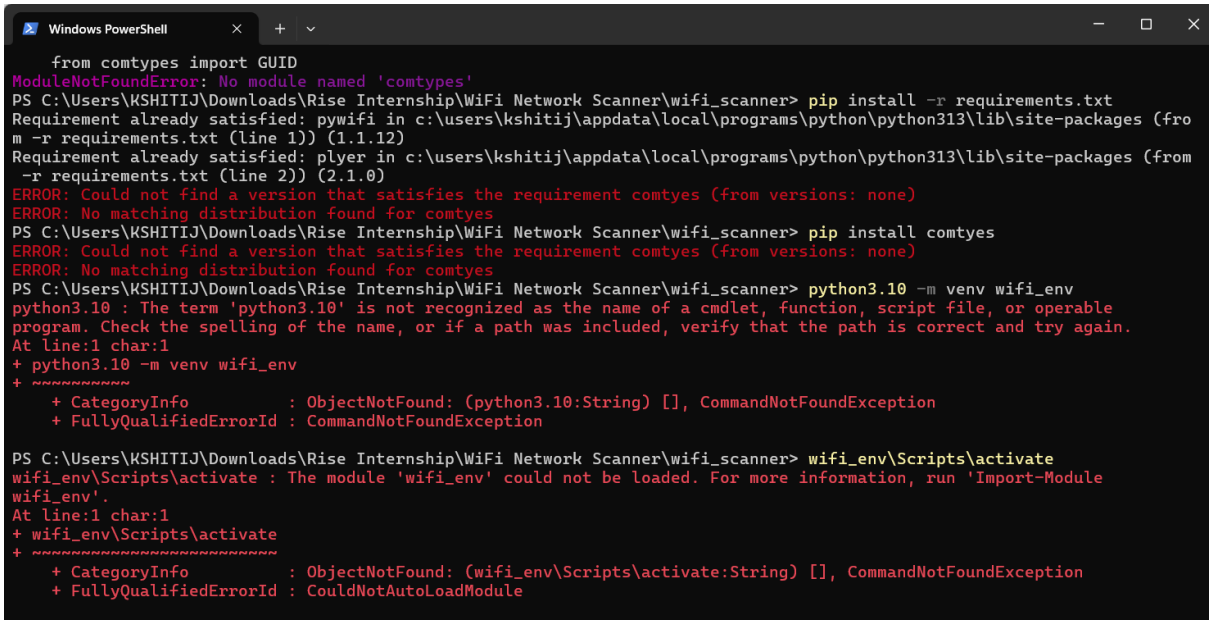
Challenges Faced

- **Python 3.13 incompatibility** with comtypes and pywifi
- Resolved by switching to **Python 3.10**, which is compatible with all required libraries.
- Mapping Wi-Fi **frequencies to channel numbers** accurately.
- Cross-platform considerations (Windows/Linux/Mac)

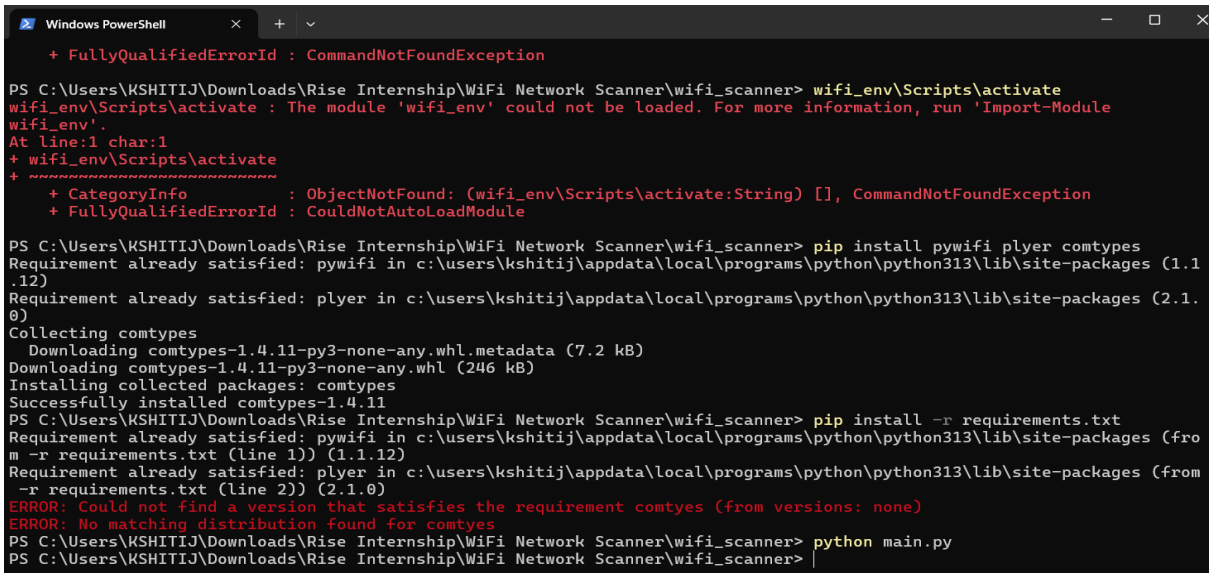
Screenshots



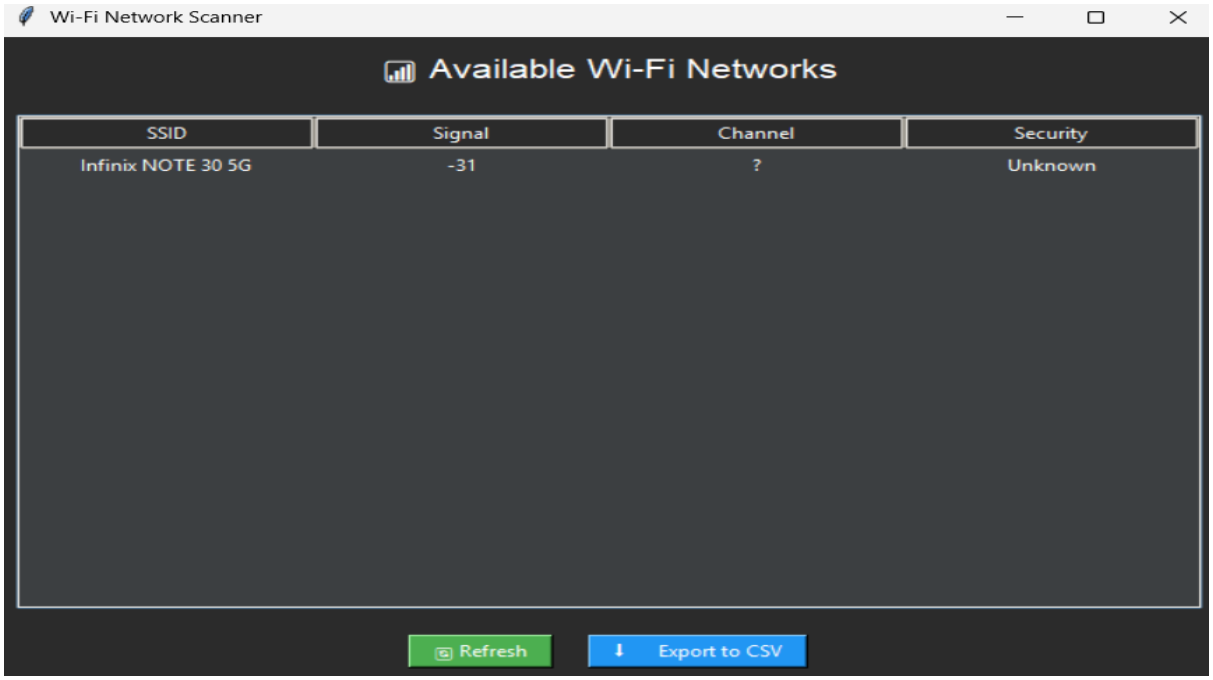
```
Windows PowerShell
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

Install the latest PowerShell for new features and improvements! https://aka.ms/PSWindows

PS C:\Users\KSHITIJ\Downloads\Rise Internship\WiFi Network Scanner\wifi_scanner> pip install -r requirements.txt
Collecting pywifi (from -r requirements.txt (line 1))
  Downloading pywifi-1.1.12-py3-none-any.whl.metadata (2.7 kB)
Collecting plyer (from -r requirements.txt (line 2))
  Downloading plyer-2.1.0-py2.py3-none-any.whl.metadata (61 kB)
Downloading pywifi-1.1.12-py3-none-any.whl (15 kB)
Downloading plyer-2.1.0-py2.py3-none-any.whl (142 kB)
Installing collected packages: plyer, pywifi
Successfully installed plyer-2.1.0 pywifi-1.1.12
PS C:\Users\KSHITIJ\Downloads\Rise Internship\WiFi Network Scanner\wifi_scanner> python main.py
Traceback (most recent call last):
  File "C:\Users\KSHITIJ\Downloads\Rise Internship\WiFi Network Scanner\wifi_scanner\main.py", line 2, in <module>
    from gui import run_app
  File "C:\Users\KSHITIJ\Downloads\Rise Internship\WiFi Network Scanner\wifi_scanner\gui.py", line 4, in <module>
    from scanner import scan_wifi
  File "C:\Users\KSHITIJ\Downloads\Rise Internship\WiFi Network Scanner\wifi_scanner\scanner.py", line 2, in <module>
    import pywifi
  File "C:\Users\KSHITIJ\AppData\Local\Programs\Python\Python313\Lib\site-packages\pywifi\__init__.py", line 15, in <mod
ule>
    from .wifi import PyWiFi
  File "C:\Users\KSHITIJ\AppData\Local\Programs\Python\Python313\Lib\site-packages\pywifi\wifi.py", line 15, in <module>
    from .iface import Interface
  File "C:\Users\KSHITIJ\AppData\Local\Programs\Python\Python313\Lib\site-packages\pywifi\iface.py", line 11, in <module
>
    from . import _wifiutil_win as wifiutil
```

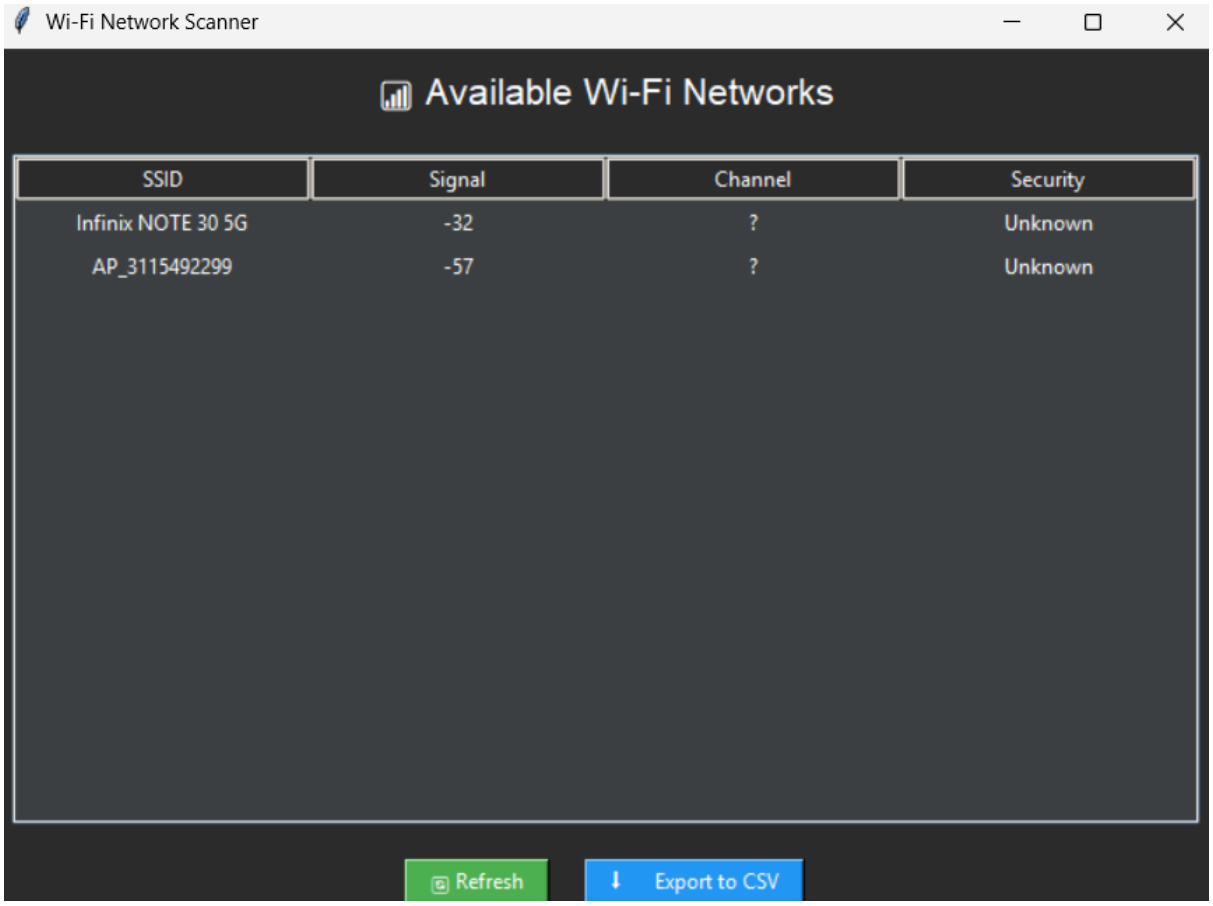


```
Windows PowerShell
    from comtypes import GUID
ModuleNotFoundError: No module named 'comtypes'
PS C:\Users\KSHITIJ\Downloads\Rise Internship\WiFi Network Scanner\wifi_scanner> pip install -r requirements.txt
Requirement already satisfied: pywifi in c:\users\kshitij\appdata\local\programs\python\python313\lib\site-packages (fro
m -r requirements.txt (line 1)) (1.1.12)
Requirement already satisfied: plyer in c:\users\kshitij\appdata\local\programs\python\python313\lib\site-packages (from
 -r requirements.txt (line 2)) (2.1.0)
ERROR: Could not find a version that satisfies the requirement comtyes (from versions: none)
ERROR: No matching distribution found for comtyes
PS C:\Users\KSHITIJ\Downloads\Rise Internship\WiFi Network Scanner\wifi_scanner> pip install comtyes
ERROR: Could not find a version that satisfies the requirement comtyes (from versions: none)
ERROR: No matching distribution found for comtyes
PS C:\Users\KSHITIJ\Downloads\Rise Internship\WiFi Network Scanner\wifi_scanner> python3.10 -m venv wifi_env
python3.10 : The term 'python3.10' is not recognized as the name of a cmdlet, function, script file, or operable
program. Check the spelling of the name, or if a path was included, verify that the path is correct and try again.
At line:1 char:1
+ python3.10 -m venv wifi_env
+ ~~~~~~~~~~
    + CategoryInfo          : ObjectNotFound: (python3.10:String) [], CommandNotFoundException
    + FullyQualifiedErrorId : CommandNotFoundException

PS C:\Users\KSHITIJ\Downloads\Rise Internship\WiFi Network Scanner\wifi_scanner> wifi_env\Scripts\activate
wifi_env\Scripts\activate : The module 'wifi_env' could not be loaded. For more information, run 'Import-Module
wifi_env'.
At line:1 char:1
+ wifi_env\Scripts\activate
+ ~~~~~~~~~~~~~~~~~~~~~~~~~
    + CategoryInfo          : ObjectNotFound: (wifi_env\Scripts\activate:String) [], CommandNotFoundException
    + FullyQualifiedErrorId : CouldNotAutoLoadModule
```



```
Windows PowerShell
    + FullyQualifiedErrorId : CommandNotFoundException

PS C:\Users\KSHITIJ\Downloads\Rise Internship\WiFi Network Scanner\wifi_scanner> wifi_env\Scripts\activate
wifi_env\Scripts\activate : The module 'wifi_env' could not be loaded. For more information, run 'Import-Module
wifi_env'.
At line:1 char:1
+ wifi_env\Scripts\activate
+ ~~~~~~~~~~~~~~~~~~~~~~~~~
    + CategoryInfo          : ObjectNotFound: (wifi_env\Scripts\activate:String) [], CommandNotFoundException
    + FullyQualifiedErrorId : CouldNotAutoLoadModule

PS C:\Users\KSHITIJ\Downloads\Rise Internship\WiFi Network Scanner\wifi_scanner> pip install pywifi plyer comtypes
Requirement already satisfied: pywifi in c:\users\kshitij\appdata\local\programs\python\python313\lib\site-packages (1.1
.12)
Requirement already satisfied: plyer in c:\users\kshitij\appdata\local\programs\python\python313\lib\site-packages (2.1.
0)
Collecting comtypes
  Downloading comtypes-1.4.11-py3-none-any.whl.metadata (7.2 kB)
Downloading comtypes-1.4.11-py3-none-any.whl (246 kB)
Installing collected packages: comtypes
Successfully installed comtypes-1.4.11
PS C:\Users\KSHITIJ\Downloads\Rise Internship\WiFi Network Scanner\wifi_scanner> pip install -r requirements.txt
Requirement already satisfied: pywifi in c:\users\kshitij\appdata\local\programs\python\python313\lib\site-packages (fro
m -r requirements.txt (line 1)) (1.1.12)
Requirement already satisfied: plyer in c:\users\kshitij\appdata\local\programs\python\python313\lib\site-packages (from
 -r requirements.txt (line 2)) (2.1.0)
ERROR: Could not find a version that satisfies the requirement comtyes (from versions: none)
ERROR: No matching distribution found for comtyes
PS C:\Users\KSHITIJ\Downloads\Rise Internship\WiFi Network Scanner\wifi_scanner> python main.py
PS C:\Users\KSHITIJ\Downloads\Rise Internship\WiFi Network Scanner\wifi_scanner>
```


Wi-Fi Network Scanner
Available Wi-Fi Networks
SSID
Signal
Channel
Security
Infinix NOTE 30 5G
-31
?
Unknown
Refresh
Export to CSV


Wi-Fi Network Scanner
Available Wi-Fi Networks
SSID
Signal
Channel
Security
Infinix NOTE 30 5G
-32
?
Unknown
AP_3115492299
-57
?
Unknown
Refresh
Export to CSV

Conclusion

This Wi-Fi Scanner project demonstrates practical **network-level programming**, **real-time system monitoring**, and **GUI development**. The integration of **notifications**, **data export**, and **auto-refreshing** logic simulates a **real-world application** suitable for both end users and IT professionals.

It showcases:

- Strong **Python development skills**
- Awareness of **networking fundamentals**
- Proficiency in **cross-platform desktop tools**
- Readiness to build and deploy **realistic software solutions**

Future Scope

- Add signal graph visualization (matplotlib or Plotly)
- Integrate with network config tools to auto-connect
- Create a standalone .exe using PyInstaller