Configuration Manual

Mitigating Cyber Risks in Next-Generation Aircraft: Securing Flight Control Systems from Remote Cyber Attacks

Msc Cybersecurity

Rahul Nalwale

Student ID: x22194339

School of Computing

National College of Ireland

Supervisor: Michael Pantridge

National College of Ireland

MSc Project Submission Sheet

School of Computing

Student Name:	Rahul Nalwale		
Student ID:	22194339		
Programme:	MSc Cybersecurity	**Year:**	2023/24
Module:	Practicum		
Lecturer:	Michael Pantridge		
Submission Due Date:	12/08/2024		
Project Title:	Mitigating Cyber Risks in Next-Generation Aircraft: Securing Flight Control Systems from Remote Cyber Attacks		
Word Count:	1050	**Page Count:**	14

I hereby certify that the information contained in this (my submission) is information pertaining to research I conducted for this project. All information other than my own contribution will be fully referenced and listed in the relevant bibliography section at the rear of the project.
ALL internet material must be referenced in the bibliography section. Students are required to use the Referencing Standard specified in the report template. To use other author's written or electronic work is illegal (plagiarism) and may result in disciplinary action.

Signature: Rrnalwale

Date: 12/08/24

PLEASE READ THE FOLLOWING INSTRUCTIONS AND CHECKLIST

Attach a completed copy of this sheet to each project (including multiple copies)	□
Attach a Moodle submission receipt of the online project submission, to each project (including multiple copies).	□
You must ensure that you retain a HARD COPY of the project, both for your own reference and in case a project is lost or mislaid. It is not sufficient to keep a copy on computer.	□

Assignments that are submitted to the Programme Coordinator Office must be placed into the assignment box located outside the office.

Office Use Only	
Signature:	
Date:	
Penalty Applied (if applicable):	

Configuration Manual

Forename Surname

Student ID:

1. Introduction

Purpose:

To provide step-by-step instructions for setting up and running the project.

Overview of the Project:

Project name: [Your Project Name].

Technologies used: React, Node.js with Express, FastAPI, Python, shadcn/ui for UI.

Project focus: The proposed project is concerned with the design of a viable system that relies on machine learning algorithms to counter cybersecurity threats in novel-generation aircraft. It combines the frontend featuring React with the shadcn/ui library, Node. js with Express for managing the backend part, and FastAPI built-in Python for the integration of the AI anomaly detection function that provides real-time protection and data analysis.

Scope: The range involves the architecture of a sophisticated stack in the sphere of aviation cybersecurity with frontend, backend, and machine learning components. It entails the creation of development environments, deployment of the threat detection models, and development processes. The manual helps one through installation, use and diagnosis to guarantee adequate protection for the next-generation aircraft.

2. System Requirements

Hardware Requirements:

- Processor: Intel Core i7 or equivalent.
- RAM: 16GB.

- Storage: 500GB SSD.

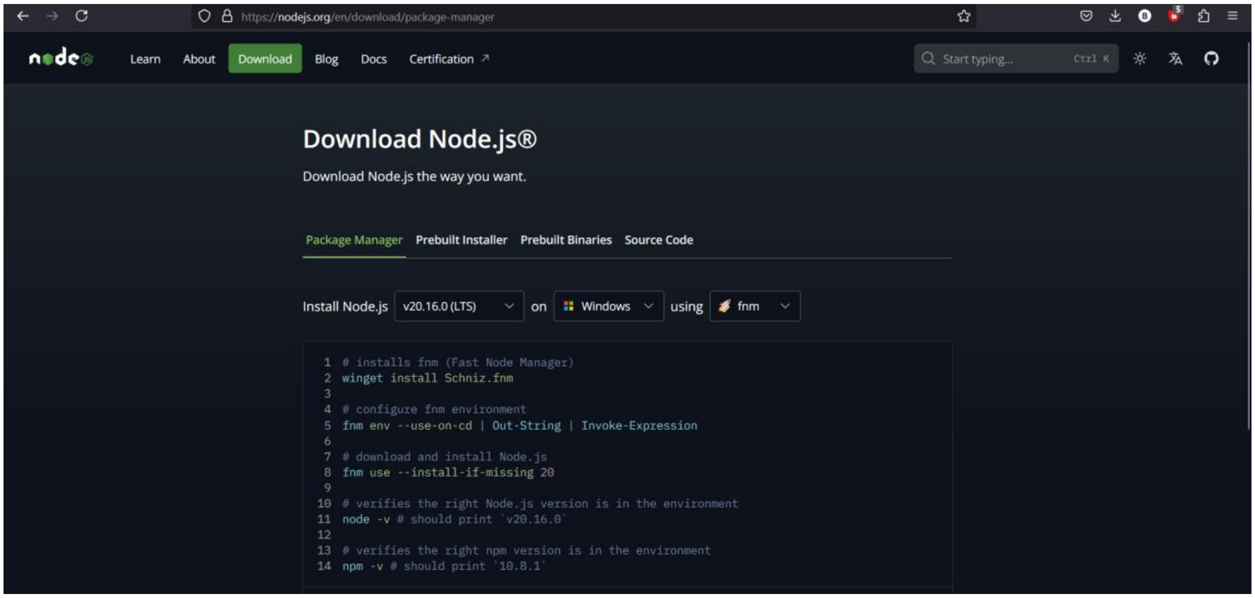
Software Requirements:

- Operating System: Windows 10 / macOS Catalina.
- IDE: Visual Studio Code or PyCharm.
- Node.js with Express for the backend.
- FastAPI for Python-based AI functionalities.
- React with shadcn/ui for the front end.

3. Environment Setup

Development Tools:

Install Node.js:



Download Node.js®

Download Node.js the way you want.

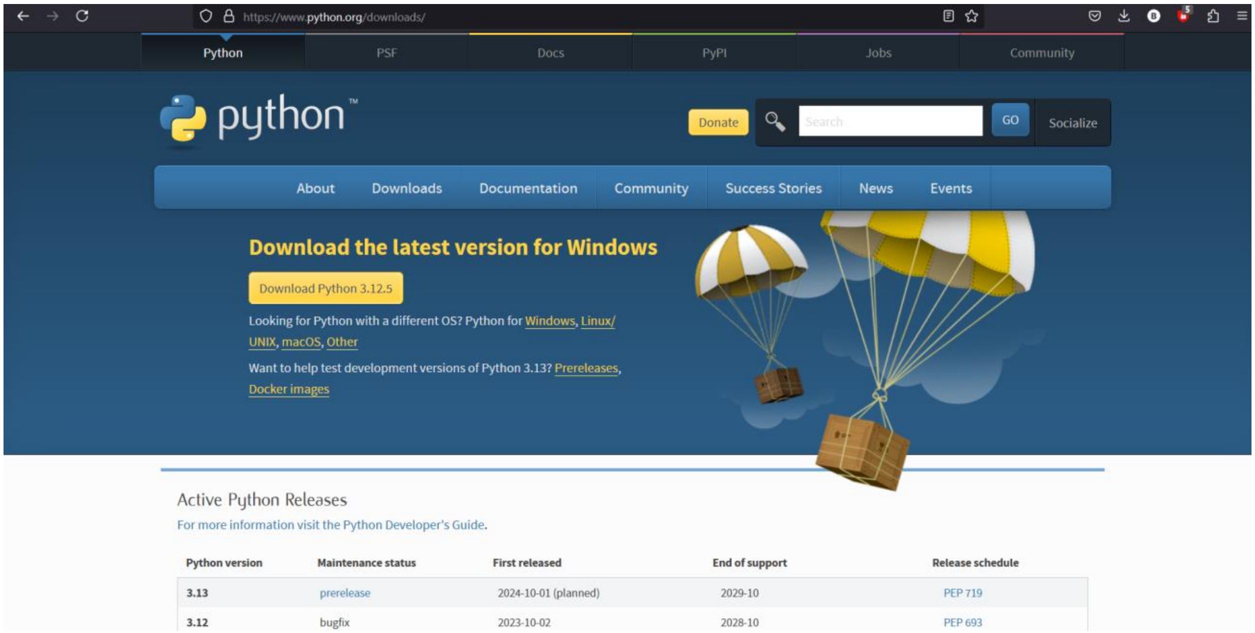
Install Node.js v20.16.0 (LTS) on Windows using fnm

```
# installs fnm (Fast Node Manager)
winget install Schniz.fnm

# configure fnm environment
fnm env --use-on-cd | Out-String | Invoke-Expression

# download and install Node.js
fnm use --install-if-missing 20

# verifies the right Node.js version is in the environment
node -v # should print `v20.16.0`

# verifies the right npm version is in the environment
npm -v # should print `10.8.1`
```

Download from Node.js website.

Run npm install to set up the environment.

Install Python:



Download the latest version for Windows

Looking for Python with a different OS? Python for Windows, Linux/UNIX, macOS, Other

Want to help test development versions of Python 3.13? Prereleases, Docker images

Active Python Releases

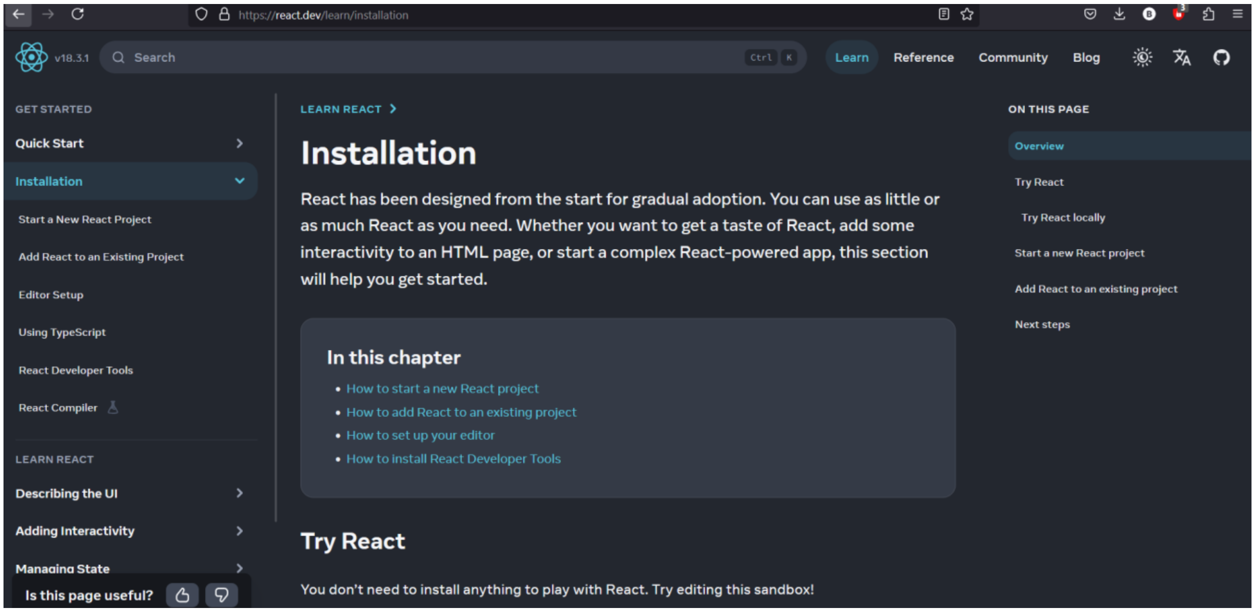
For more information visit the Python Developer's Guide.

Python version	Maintenance status	First released	End of support	Release schedule
3.13	prerelease	2024-10-01 (planned)	2029-10	PEP 719
3.12	bugfix	2023-10-02	2028-10	PEP 693

Download Python 3.8 or later.

Use pip install fastapi to set up FastAPI.

Install React:



Installation

React has been designed from the start for gradual adoption. You can use as little or as much React as you need. Whether you want to get a taste of React, add some interactivity to an HTML page, or start a complex React-powered app, this section will help you get started.

In this chapter

- How to start a new React project
- How to add React to an existing project
- How to set up your editor
- How to install React Developer Tools

Try React

You don't need to install anything to play with React. Try editing this sandbox!

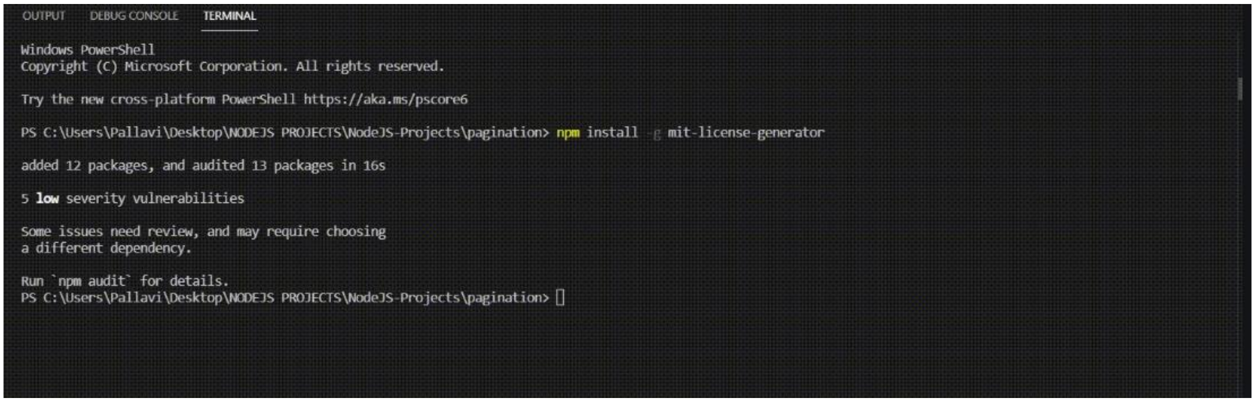
Use npx create-react-app to initialize a React project.

Install shadcn/ui components as needed for UI development.

Dependencies:

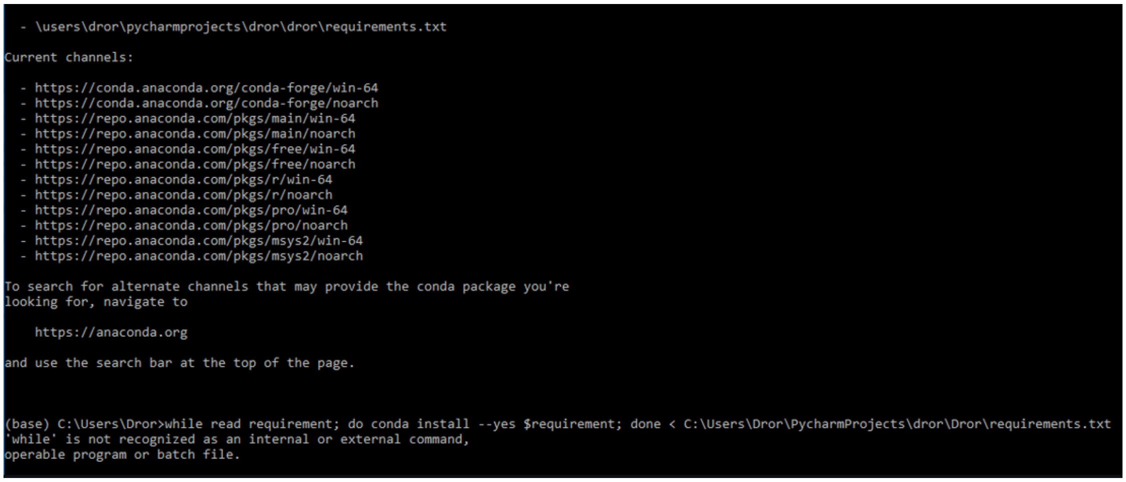
Install project dependencies:

- Use npm install for Node.js dependencies.



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

Try the new cross-platform PowerShell https://aka.ms/pscore6

PS C:\Users\Pallavi\Desktop\NODEJS PROJECTS\NodeJS-Projects\pagination> npm install -g mit-license-generator

added 12 packages, and audited 13 packages in 16s

5 low severity vulnerabilities

Some issues need review, and may require choosing
a different dependency.

Run `npm audit` for details.
PS C:\Users\Pallavi\Desktop\NODEJS PROJECTS\NodeJS-Projects\pagination> []
```

- Use pip install -r requirements.txt for Python dependencies.



```
  - \users\dror\pycharmprojects\dror\dror\requirements.txt

Current channels:

  - https://conda.anaconda.org/conda-forge/win-64
  - https://conda.anaconda.org/conda-forge/noarch
  - https://repo.anaconda.com/pkgs/main/win-64
  - https://repo.anaconda.com/pkgs/main/noarch
  - https://repo.anaconda.com/pkgs/free/win-64
  - https://repo.anaconda.com/pkgs/free/noarch
  - https://repo.anaconda.com/pkgs/r/win-64
  - https://repo.anaconda.com/pkgs/r/noarch
  - https://repo.anaconda.com/pkgs/pro/win-64
  - https://repo.anaconda.com/pkgs/pro/noarch
  - https://repo.anaconda.com/pkgs/msys2/win-64
  - https://repo.anaconda.com/pkgs/msys2/noarch

To search for alternate channels that may provide the conda package you're
looking for, navigate to

    https://anaconda.org

and use the search bar at the top of the page.



(base) C:\Users\Dror>while read requirement; do conda install --yes $requirement; done < C:\Users\Dror\PycharmProjects\dror\Dror\requirements.txt
'while' is not recognized as an internal or external command,
operable program or batch file.
```

Configuration Files:

Example settings.py for Python backend:

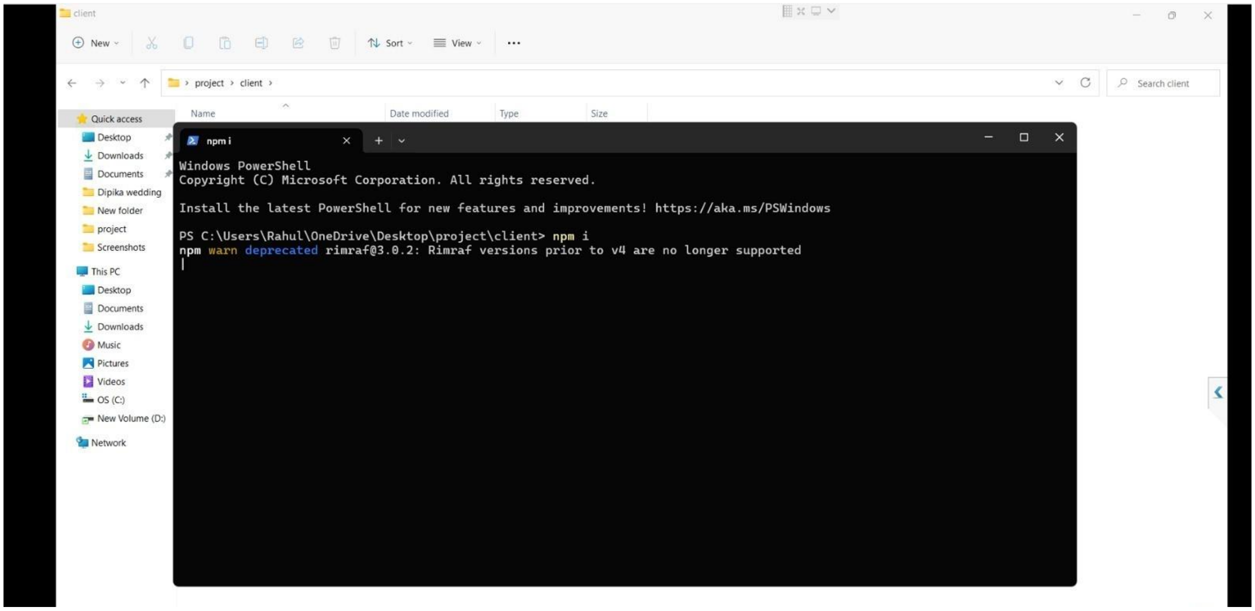
Configure AI models and API endpoints.

Example config.js for Node.js backend:

4. Installation Instructions

Step-by-Step Installation:

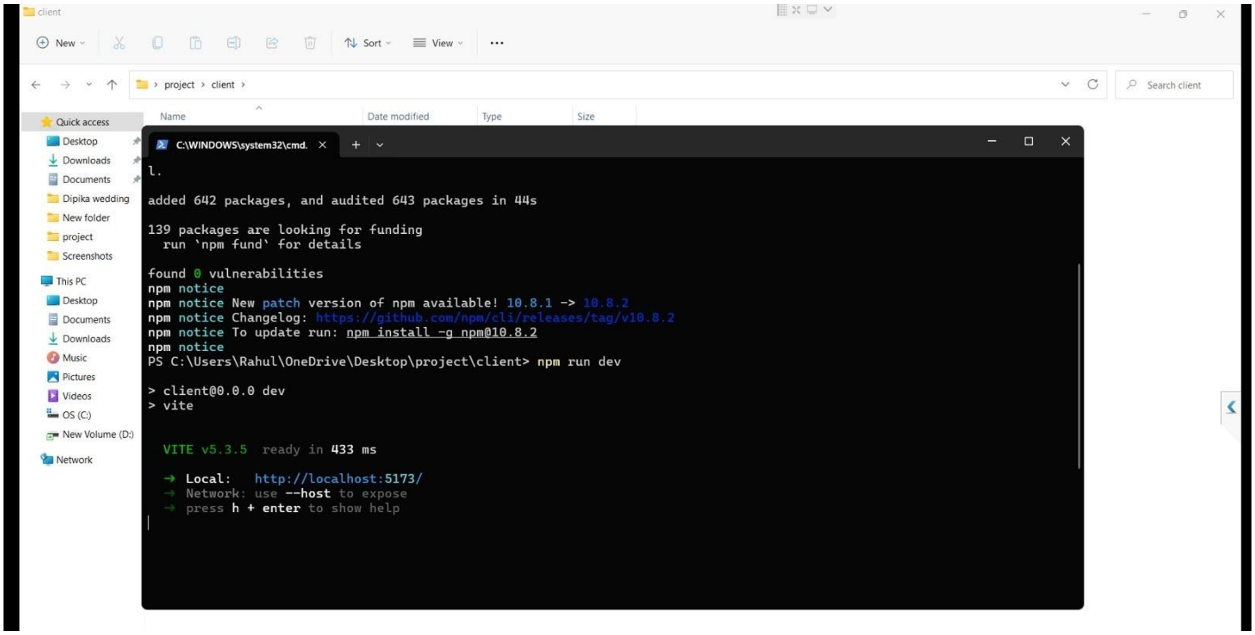
Download the project files:



```
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\Rahul\OneDrive\Desktop\project\client> npm i
npm warn deprecated rimraf@3.0.2: Rimraf versions prior to v4 are no longer supported
|
```

- Obtain the project files from the provided source (e.g., USB, shared drive, email attachment).
- Extract the project files:
- Unzip the downloaded project archive to a desired location on your machine.
- Navigate to the project directory:
- Use the terminal or command prompt to navigate to the project directory where the main application files are located.

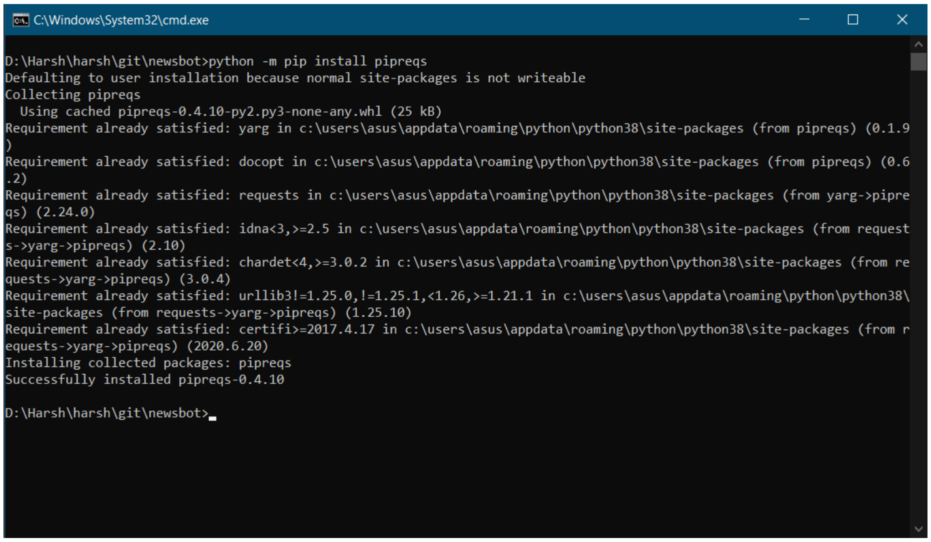
Install Node.js dependencies:



```
l.

added 642 packages, and audited 643 packages in 44s

139 packages are looking for funding
  run `npm fund` for details

found 0 vulnerabilities
npm notice
npm notice New patch version of npm available! 10.8.1 -> 10.8.2
npm notice Changelog: https://github.com/npm/cli/releases/tag/v10.8.2
npm notice To update run: npm install -g npm@10.8.2
npm notice
PS C:\Users\Rahul\OneDrive\Desktop\project\client> npm run dev

> client@0.0.0 dev
> vite


  VITE v5.3.5  ready in 433 ms

  ➜  Local:   http://localhost:5173/
  ➜  Network: use --host to expose
  ➜  press h + enter to show help
```

Run `npm install` in the Node.js backend directory to install all necessary packages.

Install Python dependencies:

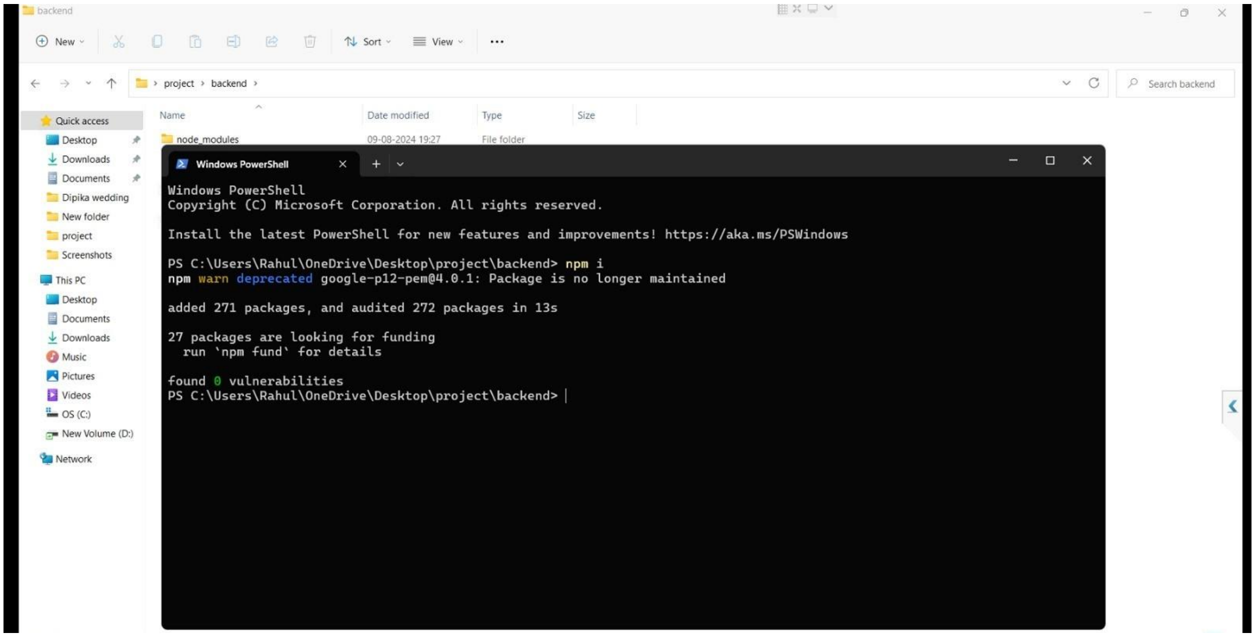


```
D:\Harsh\harsh\git\newsbot>python -m pip install pipreqs
Defaulting to user installation because normal site-packages is not writeable
Collecting pipreqs
  Using cached pipreqs-0.4.10-py2.py3-none-any.whl (25 kB)
Requirement already satisfied: yarg in c:\users\asus\appdata\roaming\python\python38\site-packages (from pipreqs) (0.1.9
)
Requirement already satisfied: docopt in c:\users\asus\appdata\roaming\python\python38\site-packages (from pipreqs) (0.6
.2)
Requirement already satisfied: requests in c:\users\asus\appdata\roaming\python\python38\site-packages (from yarg->pipre
qs) (2.24.0)
Requirement already satisfied: idna<3,>=2.5 in c:\users\asus\appdata\roaming\python\python38\site-packages (from request
s->yarg->pipreqs) (2.10)
Requirement already satisfied: chardet<4,>=3.0.2 in c:\users\asus\appdata\roaming\python\python38\site-packages (from re
quests->yarg->pipreqs) (3.0.4)
Requirement already satisfied: urllib3!=1.25.0,!=1.25.1,<1.26,>=1.21.1 in c:\users\asus\appdata\roaming\python\python38\
site-packages (from requests->yarg->pipreqs) (1.25.10)
Requirement already satisfied: certifi>=2017.4.17 in c:\users\asus\appdata\roaming\python\python38\site-packages (from r
equests->yarg->pipreqs) (2020.6.20)
Installing collected packages: pipreqs
Successfully installed pipreqs-0.4.10

D:\Harsh\harsh\git\newsbot>
```

Run `pip install -r requirements.txt` in the Python backend directory to install required libraries.

Verifying Installation:

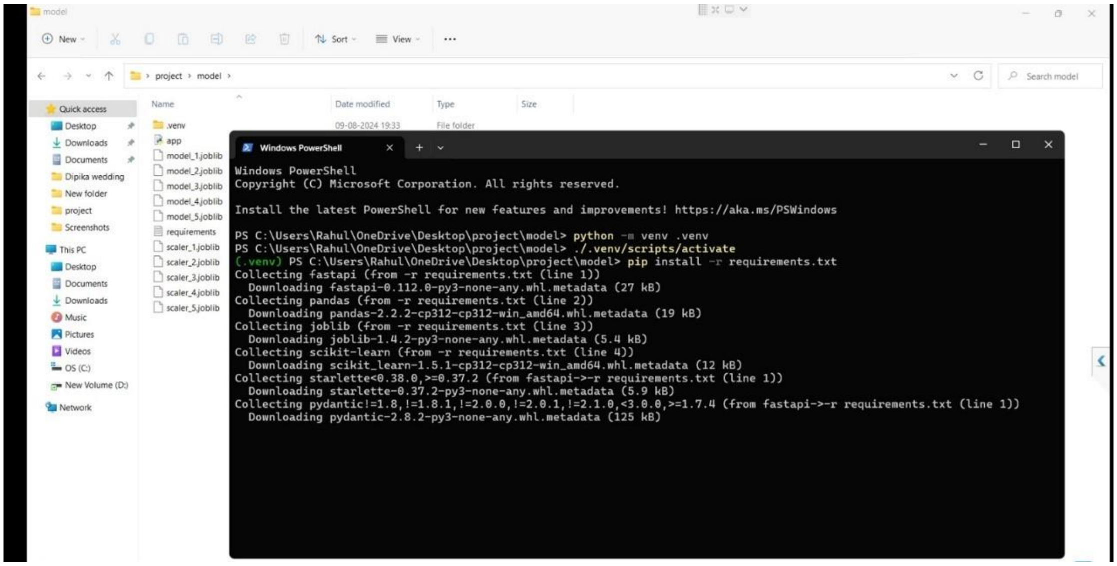
Check Node.js installation:



```
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\Rahul\OneDrive\Desktop\project\backend> npm i
npm warn deprecated google-p12-pem@4.0.1: Package is no longer maintained

added 271 packages, and audited 272 packages in 13s

27 packages are looking for funding
  run `npm fund` for details

found 0 vulnerabilities
PS C:\Users\Rahul\OneDrive\Desktop\project\backend> |
```

Run `node --version` to confirm that Node.js is correctly installed.

Verify Python setup:



```
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\Rahul\OneDrive\Desktop\project\model> python -m venv .venv
PS C:\Users\Rahul\OneDrive\Desktop\project\model> ./.venv/scripts/activate
(.venv) PS C:\Users\Rahul\OneDrive\Desktop\project\model> pip install -r requirements.txt
Collecting fastapi (from -r requirements.txt (line 1))
  Downloading fastapi-0.112.0-py3-none-any.whl.metadata (27 kB)
Collecting pandas (from -r requirements.txt (line 2))
  Downloading pandas-2.2.2-cp312-cp312-win_amd64.whl.metadata (19 kB)
Collecting joblib (from -r requirements.txt (line 3))
  Downloading joblib-1.4.2-py3-none-any.whl.metadata (5.4 kB)
Collecting scikit-learn (from -r requirements.txt (line 4))
  Downloading scikit_learn-1.5.1-cp312-cp312-win_amd64.whl.metadata (12 kB)
Collecting starlette<0.38.0,>=0.37.2 (from fastapi->-r requirements.txt (line 1))
  Downloading starlette-0.37.2-py3-none-any.whl.metadata (5.9 kB)
Collecting pydantic!=1.8,!=1.8.1,!=2.0.0,!=2.0.1,!=2.1.0,<3.0.0,>=1.7.4 (from fastapi->-r requirements.txt (line 1))
  Downloading pydantic-2.8.2-py3-none-any.whl.metadata (125 kB)
```

Run `python --version` and `pip list` to ensure Python and its dependencies are properly installed.

5. Running the Project

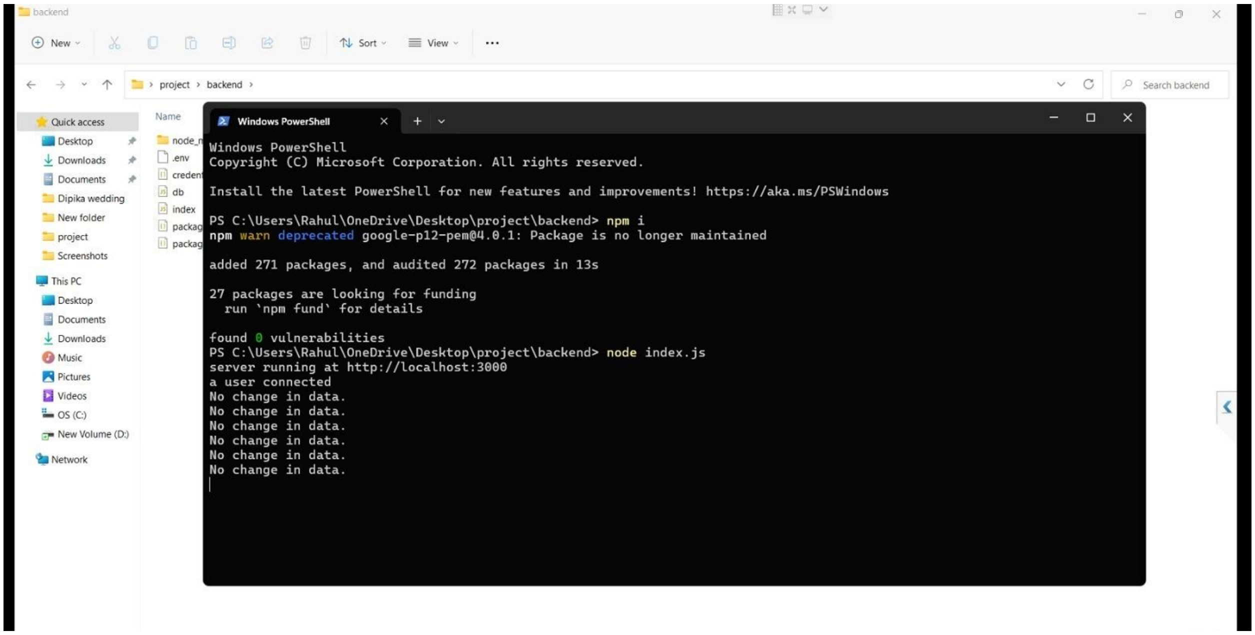
Initial Setup:

Database migration (if applicable):

For projects using a database, run the necessary migration commands (e.g., python manage.py migrate) to set up the database schema.

Running the Project:

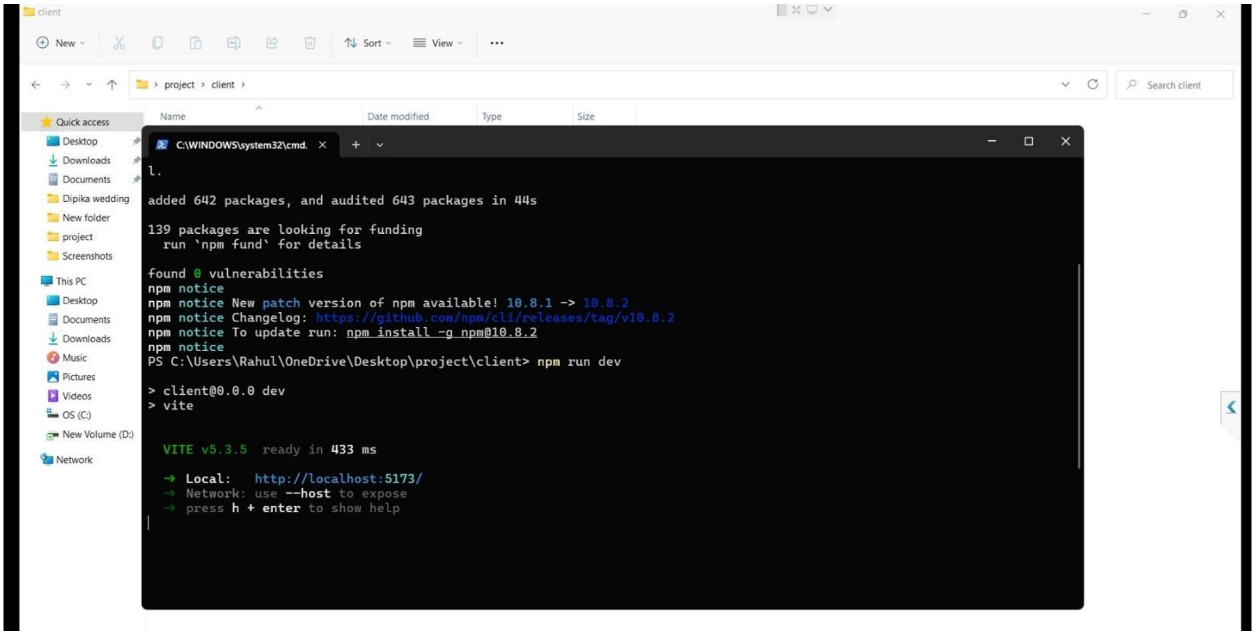
Start the Node.js backend:



```
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\Rahul\OneDrive\Desktop\project\backend> npm i
npm warn deprecated google-p12-pem@4.0.1: Package is no longer maintained

added 271 packages, and audited 272 packages in 13s

27 packages are looking for funding
  run `npm fund` for details

found 0 vulnerabilities
PS C:\Users\Rahul\OneDrive\Desktop\project\backend> node index.js
server running at http://localhost:3000
a user connected
No change in data.
No change in data.
No change in data.
No change in data.
No change in data.
No change in data.
```

In the terminal, navigate to the Node.js backend directory and run npm start to launch the backend server.

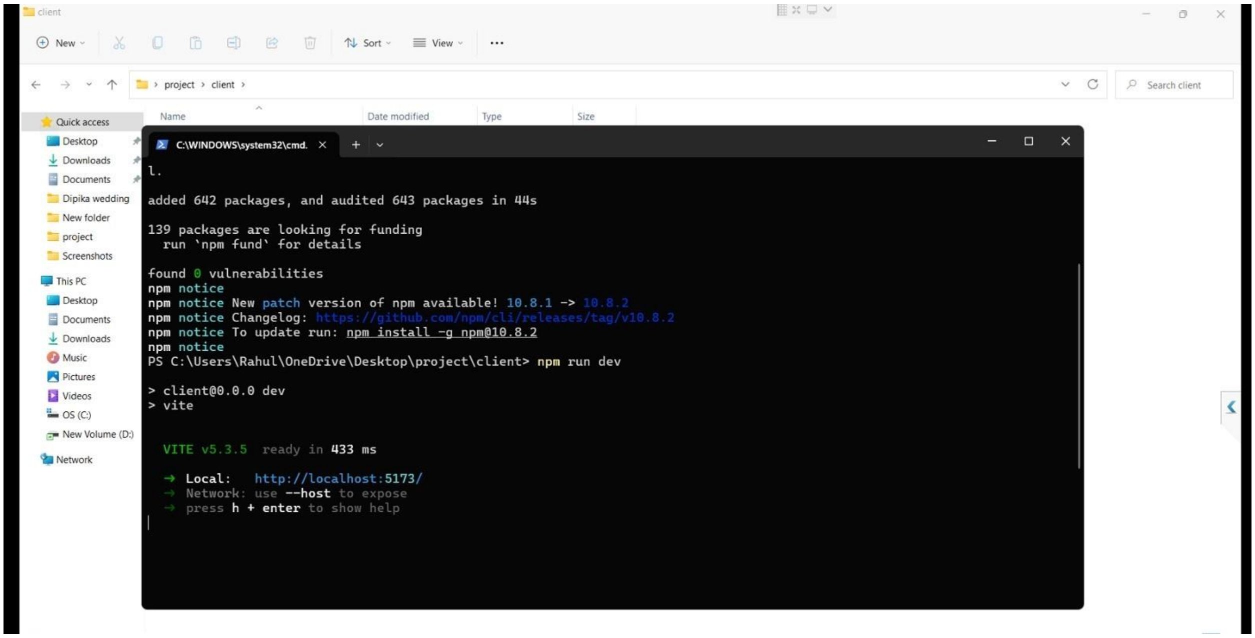
Start the Python backend:



```
l.

added 642 packages, and audited 643 packages in 44s

139 packages are looking for funding
  run `npm fund` for details

found 0 vulnerabilities
npm notice
npm notice New patch version of npm available! 10.8.1 -> 10.8.2
npm notice Changelog: https://github.com/npm/cli/releases/tag/v10.8.2
npm notice To update run: npm install -g npm@10.8.2
npm notice
PS C:\Users\Rahul\OneDrive\Desktop\project\client> npm run dev

> client@0.0.0 dev
> vite


  VITE v5.3.5  ready in 433 ms

  ➜  Local:   http://localhost:5173/
  ➜  Network: use --host to expose
  ➜  press h + enter to show help
```

Navigate to the Python backend directory and run uvicorn main: app --reload to start the FastAPI server.

Start the React frontend:

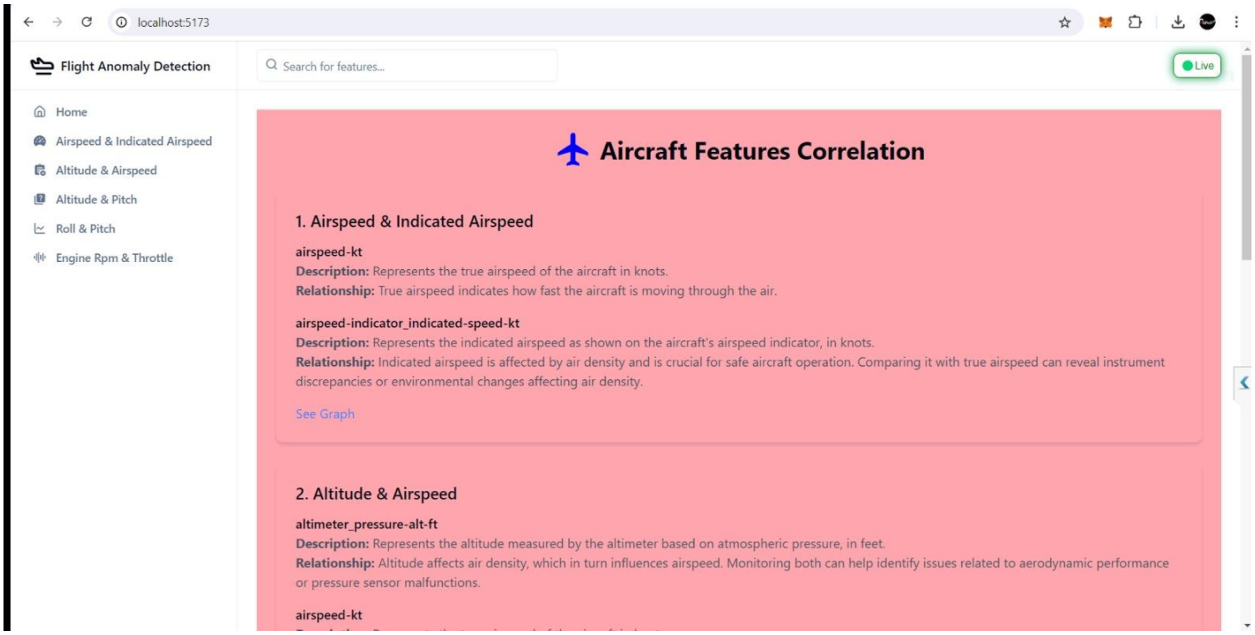


```
l.

added 642 packages, and audited 643 packages in 44s

139 packages are looking for funding
  run `npm fund` for details

found 0 vulnerabilities
npm notice
npm notice New patch version of npm available! 10.8.1 -> 10.8.2
npm notice Changelog: https://github.com/npm/cli/releases/tag/v10.8.2
npm notice To update run: npm install -g npm@10.8.2
npm notice
PS C:\Users\Rahul\OneDrive\Desktop\project\client> npm run dev

> client@0.0.0 dev
> vite


  VITE v5.3.5  ready in 433 ms

  ➜  Local:   http://localhost:5173/
  ➜  Network: use --host to expose
  ➜  press h + enter to show help
```

Navigate to the React frontend directory and run npm start to launch the frontend interface.

Access the application:

Open a web browser and navigate to http://localhost:3000 to view the running application.



Aircraft Features Correlation

1. Airspeed & Indicated Airspeed

airspeed-kt
Description: Represents the true airspeed of the aircraft in knots.
Relationship: True airspeed indicates how fast the aircraft is moving through the air.

airspeed-indicator_indicated-speed-kt
Description: Represents the indicated airspeed as shown on the aircraft's airspeed indicator, in knots.
Relationship: Indicated airspeed is affected by air density and is crucial for safe aircraft operation. Comparing it with true airspeed can reveal instrument discrepancies or environmental changes affecting air density.

See Graph

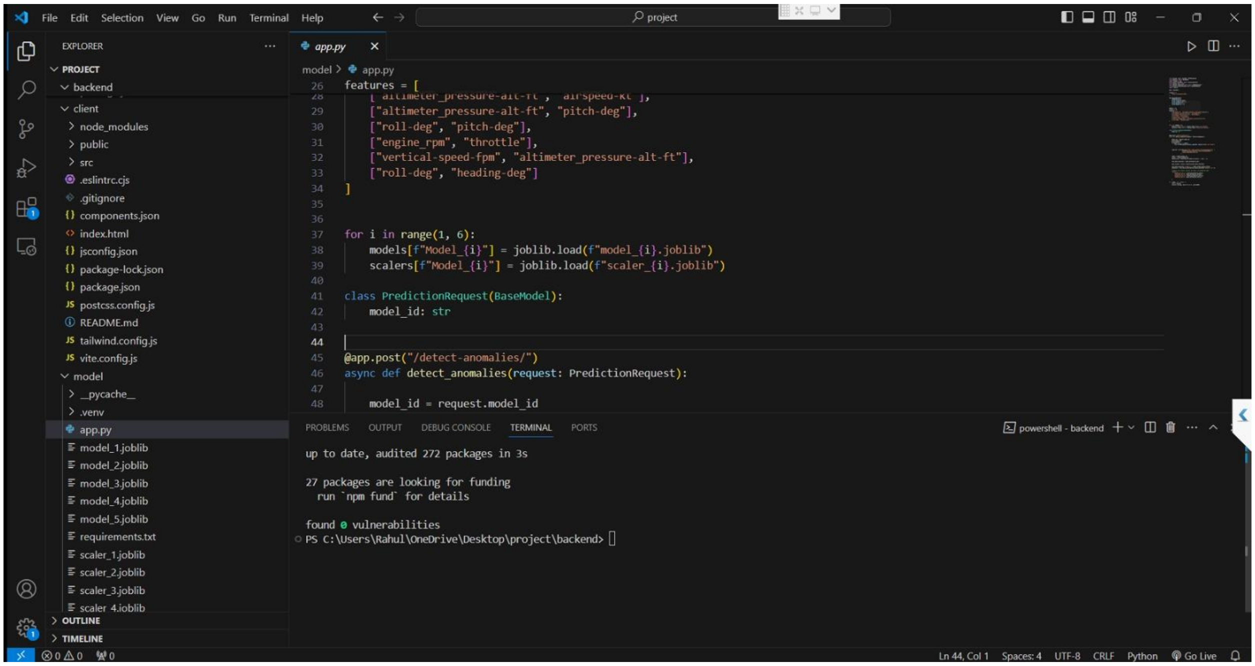
2. Altitude & Airspeed

altimeter_pressure-alt-ft
Description: Represents the altitude measured by the altimeter based on atmospheric pressure, in feet.
Relationship: Altitude affects air density, which in turn influences airspeed. Monitoring both can help identify issues related to aerodynamic performance or pressure sensor malfunctions.

airspeed-kt
Description: Represents the true airspeed of the aircraft in knots.

6. Project Structure

File Structure:



```python
features = [
    ["altimeter_pressure-alt-ft", "airspeed-kt"],
    ["altimeter_pressure-alt-ft", "pitch-deg"],
    ["roll-deg", "pitch-deg"],
    ["engine_rpm", "throttle"],
    ["vertical-speed-fpm", "altimeter_pressure-alt-ft"],
    ["roll-deg", "heading-deg"]
]


for i in range(1, 6):
    models[f"Model_{i}"] = joblib.load(f"model_{i}.joblib")
    scalers[f"Model_{i}"] = joblib.load(f"scaler_{i}.joblib")

class PredictionRequest(BaseModel):
    model_id: str


@app.post("/detect-anomalies/")
async def detect_anomalies(request: PredictionRequest):

    model_id = request.model_id
```

```
up to date, audited 272 packages in 3s

27 packages are looking for funding
  run `npm fund` for details

found 0 vulnerabilities
PS C:\Users\Rahul\OneDrive\Desktop\project\backend> 
```

`backend/`: Node.js backend with Express for handling authentication and API routing.

`frontend/`: React frontend using shadcn/ui for user interface components.

`API/`: Python backend with Fast API managing AI and machine learning models.

Code Explanation:

`server.js`: Main entry point for the Node.js backend, managing server setup and API routes.

`App.js`: Central React component handling the rendering and state management for the UI.

`main.py`: Entry point for FastAPI, defining routes and integrating machine learning models.

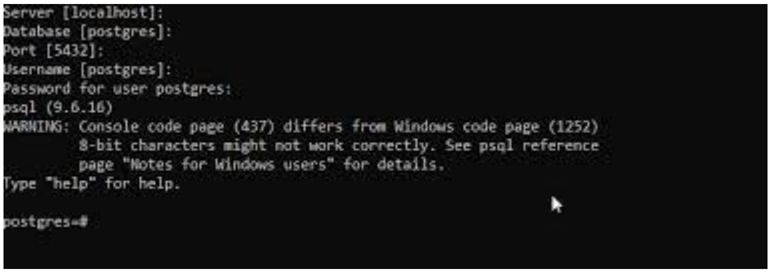
7. Troubleshooting

Common Issues:

Backend server not starting:

Ensure that all dependencies are installed correctly, and check that the correct ports are not in use.

API endpoints returning errors:



```
Server [localhost]:
Database [postgres]:
Port [5432]:
Username [postgres]:
Password for user postgres:
psql (9.6.16)
WARNING: Console code page (437) differs from Windows code page (1252)
         8-bit characters might not work correctly. See psql reference
         page "Notes for Windows users" for details.
Type "help" for help.

postgres=#
```

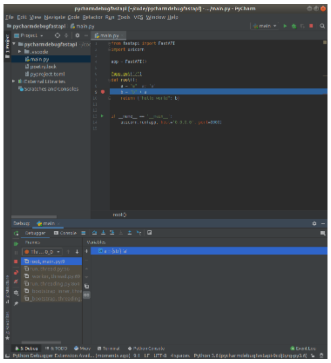
Verify the API routes in `server.js` and `main.py`, and ensure the backend services are running.

Logs and Debugging:

View Node.js logs:

Check the console output where the Node.js server is running for any errors or logs.

Debug Python backend:



pycharmdebugfastapi [~/code/pycharmdebugfastapi] - .../main.py - PyCharm

```
from fastapi import FastAPI
import uvicorn

app = FastAPI()

@app.get("/")
def root():
    a = "a" [illegible]
    b = "b" + a
    return {"hello world": b}


if __name__ == "__main__":
    uvicorn.run(app, host="0.0.0.0", port=8000)
```

a = {str} 'a'

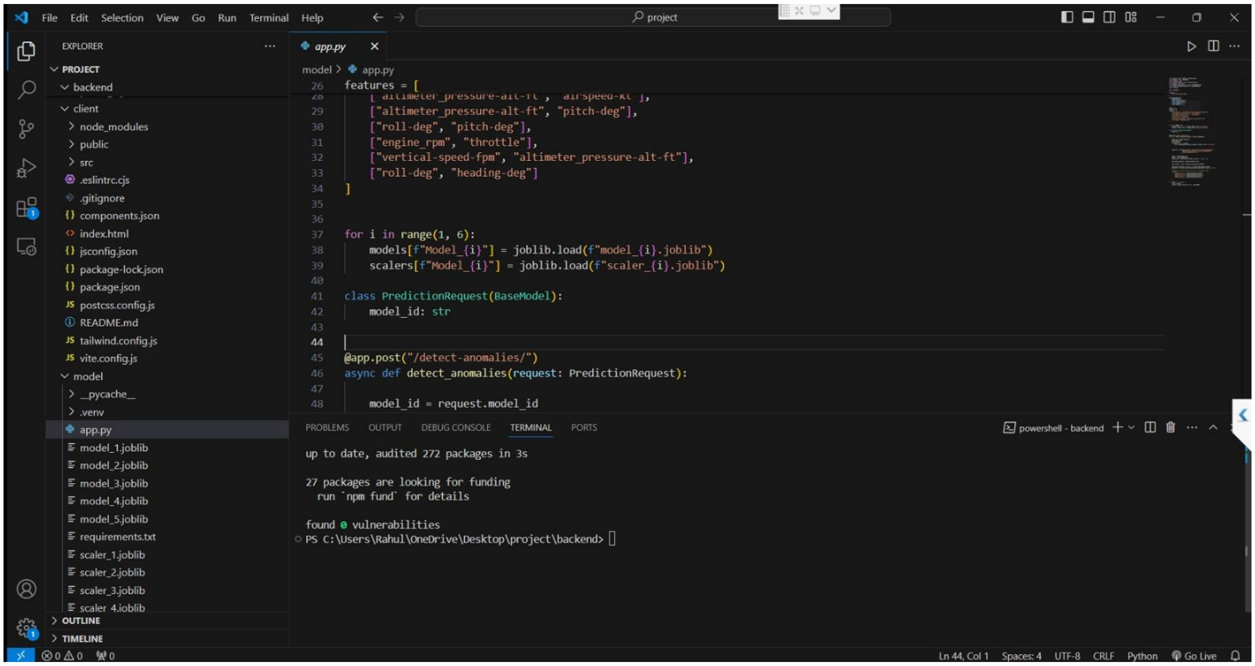
Use PyCharm or VS Code's built-in debugger to set breakpoints and inspect variables during FastAPI execution.

8. Maintenance

Updating the System:

Update dependencies:

Run `npm update` for Node.js dependencies



```python
features = [
    ["altimeter_pressure-alt-ft", "airspeed-kt"],
    ["altimeter_pressure-alt-ft", "pitch-deg"],
    ["roll-deg", "pitch-deg"],
    ["engine_rpm", "throttle"],
    ["vertical-speed-fpm", "altimeter_pressure-alt-ft"],
    ["roll-deg", "heading-deg"]
]


for i in range(1, 6):
    models[f"Model_{i}"] = joblib.load(f"model_{i}.joblib")
    scalers[f"Model_{i}"] = joblib.load(f"scaler_{i}.joblib")

class PredictionRequest(BaseModel):
    model_id: str


@app.post("/detect-anomalies/")
async def detect_anomalies(request: PredictionRequest):

    model_id = request.model_id
```

```
up to date, audited 272 packages in 3s

27 packages are looking for funding
  run `npm fund` for details

found 0 vulnerabilities
PS C:\Users\Rahul\OneDrive\Desktop\project\backend>
```

Apply database migrations (if applicable):

Run `python manage.py migrate` to apply any new database migrations.

Backup and Recovery:

Backup important files:

Regularly back up configuration files such as `config.js` for Node.js and `settings.py` for the Python backend.

Recovery process:

Restore files from the backup and re-run the necessary installation commands to reinstate the system.