

# Air Quality Dashboard - Project Setup Guide

## Air Quality Dashboard – Project Setup Guide

### 1. Install Python

- Download from: <https://www.python.org/downloads/>
- Check "Add Python to PATH" during installation.

### 2. Create and Activate Virtual Environment

```
cd D:\mote\iot_project
```

```
python -m venv venv
```

```
venv\Scripts\activate
```

### 3. Install Django & MySQL Dependencies

```
pip install django==4.2
```

```
pip install mysqlclient
```

(If mysqlclient fails: pip install pymysql and set pymysql.install\_as\_MySQLdb() in settings.py)

### 4. Project Structure

```
iot_project/
```

```
├─ venv/
```

```
├─ airmonitor/
```

```
│   └─ settings.py
```

```
└─ dashboard/
```

```
| |─ models.py
| |─ views.py
| |─ templates/dashboard/index.html
| |─ static/dashboard/style.css
└─ manage.py
```

## 5. Configure Database in settings.py

```
DATABASES = {
    'default': {
        'ENGINE': 'django.db.backends.mysql',
        'NAME': 'airmonitor_db',
        'USER': 'root',
        'PASSWORD': '',
        'HOST': 'localhost',
        'PORT': '3306',
    }
}
```

## 6. Create Models in dashboard/models.py

```
class Reading(models.Model):
    temperature = models.FloatField()
    humidity = models.FloatField()
    gas = models.FloatField()
    pm25 = models.FloatField()
    pm10 = models.FloatField()
    timestamp = models.DateTimeField(auto_now_add=True)
```

## 7. Run Migrations

```
python manage.py makemigrations
```

```
python manage.py migrate
```

## 8. Register App and Setup URLs

Add 'dashboard' in INSTALLED\_APPS

Map view in airmonitor/urls.py

## 9. Static Files Setup

```
STATIC_URL = '/static/'
```

```
STATICFILES_DIRS = [ BASE_DIR / "static" ]
```

## 10. Run the Server

```
python manage.py runserver
```

Visit: <http://127.0.0.1:8000>

## 11. Features

- Live sensor reading
- PM2.5 & PM10 Chart
- Historical reading table

## 12. Add Test Data

```
python manage.py shell
```

```
from dashboard.models import Reading
```

```
Reading.objects.create(...)
```

exit()

Done!

## Django views.py Code

```
# views.py
```

```
from django.shortcuts import render
from .models import Reading
import json # needed for JS-safe lists
```

```
def index(request):
    readings = Reading.objects.order_by('-timestamp')[:50][::-1] # last 50, oldest first

    # Prepare data for Chart.js
    timestamps = [r.timestamp.strftime("%H:%M") for r in readings]
    pm25_values = [r.pm25 for r in readings]
    pm10_values = [r.pm10 for r in readings]

    latest = readings[-1] if readings else None

    return render(request, 'dashboard/index.html', {
        'reading': latest,
        'timestamps': json.dumps(timestamps),
        'pm25_values': json.dumps(pm25_values),
        'pm10_values': json.dumps(pm10_values),
        'history_readings': readings,
    })
```