

Hackathon mock 1 1487

Title: E-Waste Management System with Recycling Center Suggestion

Objective:

The project aims to create a simple and effective system for managing e-waste items, tracking their recycling status, and suggesting appropriate recycling centers. The system will help users log new e-waste items, mark items as recycled, generate detailed reports on recycling status, and provide recommendations for recycling centers based on the item type.

Key Features:

1. Item Logging:
 - Users can log new e-waste items into one of three categories: Small Electronics, Large Appliances, and Batteries.
 - Each item is recorded with a name and its recycling status (initially set to "Not Dropped Off").
2. Recycling Status Tracking:
 - Users can update the status of items to "Dropped Off," moving them from the "Not Dropped Off" list to the "Recycled" list.
 - Recycled items are tracked separately from items that are yet to be recycled.
3. Reporting:
 - Generates detailed reports showing:
 - The number of items in each category that are yet to be recycled.
 - The number of items that have already been recycled.
 - A table-like structure listing each item by name, with counts of how many are yet to be recycled and how many are recycled.
4. Recycling Center Suggestions:
 - The system queries a local API to suggest recycling centers based on the type of e-waste item.
 - Provides users with a list of suggested centers where they can dispose of their e-waste.

Technologies Used:

- **Python:** For the core functionality and data management.
- **Flask:** To serve the recycling center API.

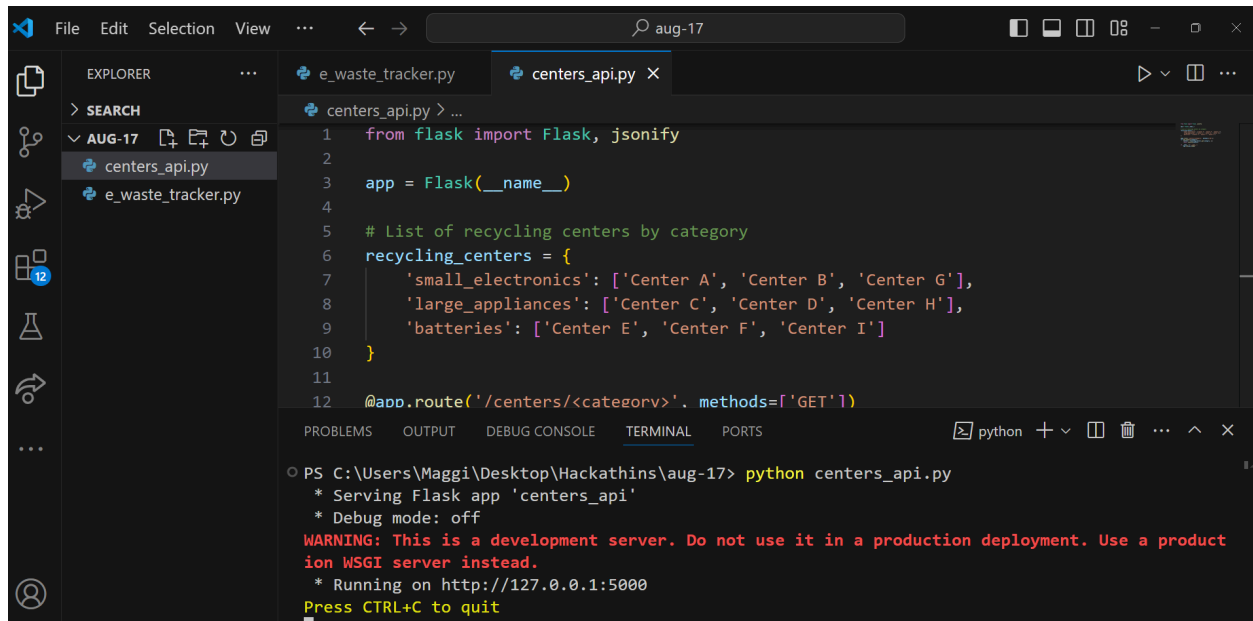
Implementation Steps:

1. Develop core Python functionality for logging items, tracking recycling status, and generating reports.
2. Create a Flask API to provide recycling center suggestions.

This system is designed to be user-friendly and efficient, helping to streamline e-waste management and promote environmentally responsible disposal practices.

Steps to follow to get it running:

Get the flask server running on your computer

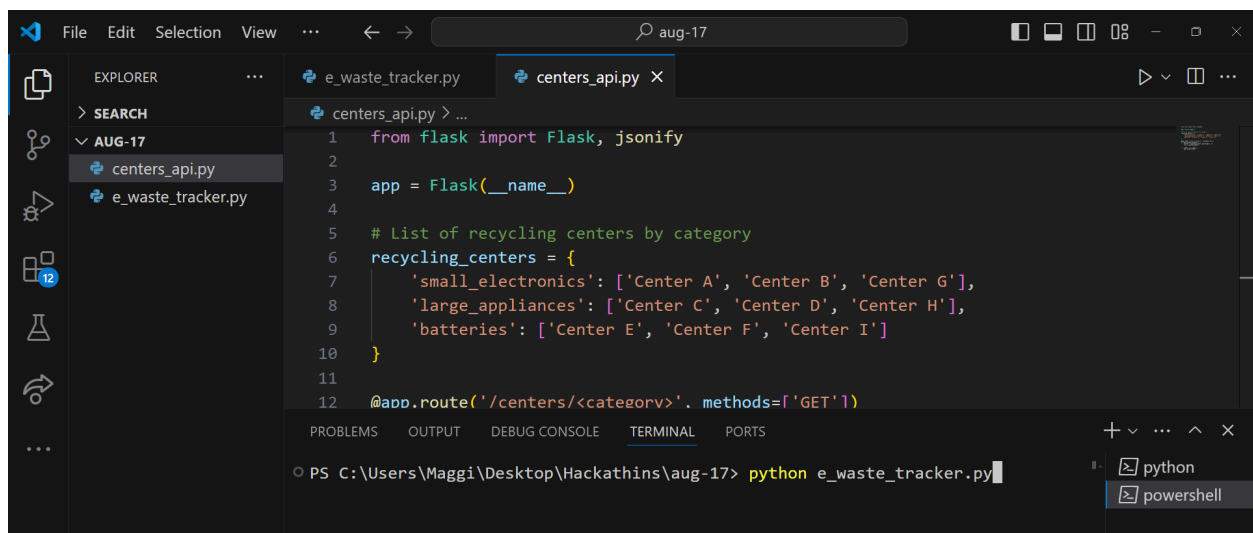


The screenshot shows the Visual Studio Code interface with the 'centers_api.py' file open. The code defines a Flask application with a list of recycling centers. The terminal at the bottom shows the command 'python centers_api.py' being executed, resulting in the server starting on http://127.0.0.1:5000.

```
1 from flask import Flask, jsonify
2
3 app = Flask(__name__)
4
5 # List of recycling centers by category
6 recycling_centers = {
7     'small_electronics': ['Center A', 'Center B', 'Center G'],
8     'large_appliances': ['Center C', 'Center D', 'Center H'],
9     'batteries': ['Center E', 'Center F', 'Center I']
10 }
11
12 @app.route('/centers/<category>'. methods=['GET'])
```

```
PS C:\Users\Maggi\Desktop\Hackathins\aug-17> python centers_api.py
* Serving Flask app 'centers_api'
* Debug mode: off
WARNING: This is a development server. Do not use it in a production deployment. Use a product
ion WSGI server instead.
* Running on http://127.0.0.1:5000
Press CTRL+C to quit
```

After the flask server is up and running



The screenshot shows the Visual Studio Code interface with the 'e_waste_tracker.py' file open. The terminal at the bottom shows the command 'python e_waste_tracker.py' being executed, resulting in the server starting on http://127.0.0.1:5000.

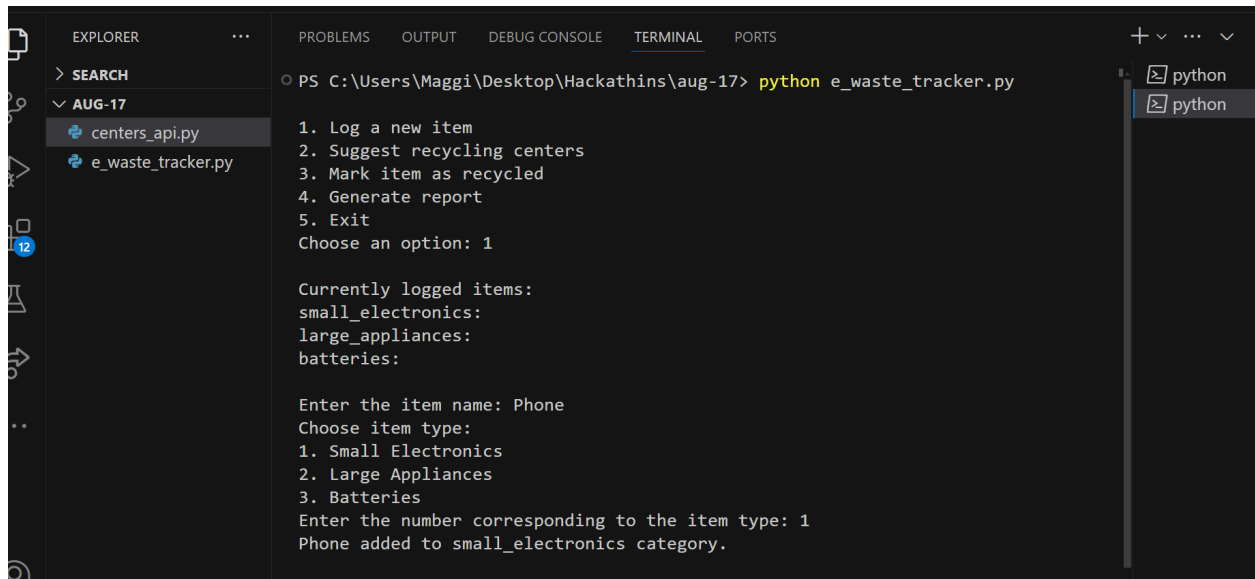
```
1 from flask import Flask, jsonify
2
3 app = Flask(__name__)
4
5 # List of recycling centers by category
6 recycling_centers = {
7     'small_electronics': ['Center A', 'Center B', 'Center G'],
8     'large_appliances': ['Center C', 'Center D', 'Center H'],
9     'batteries': ['Center E', 'Center F', 'Center I']
10 }
11
12 @app.route('/centers/<category>'. methods=['GET'])
```

```
PS C:\Users\Maggi\Desktop\Hackathins\aug-17> python e_waste_tracker.py
```

Screenshots in console for each functionality

Logging

–Small electronics



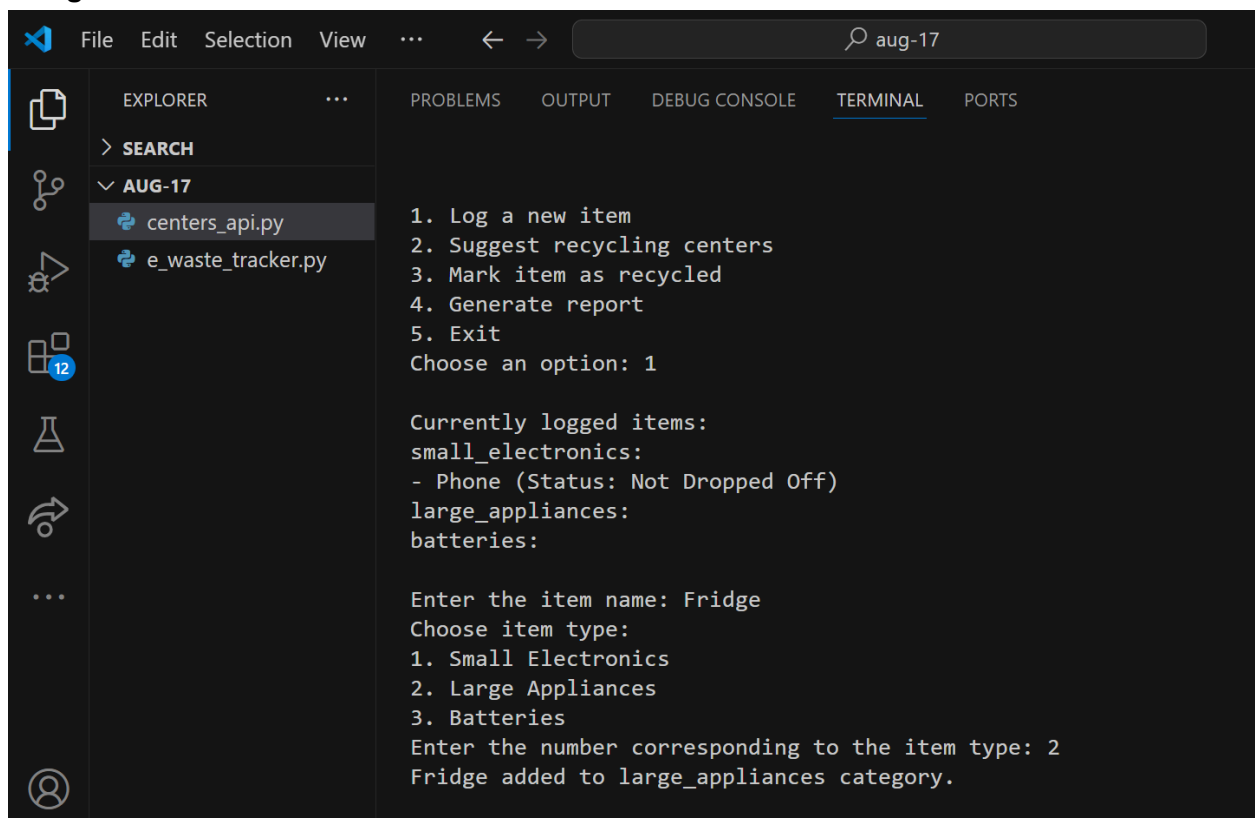
```
PS C:\Users\Maggi\Desktop\Hackathins\aug-17> python e_waste_tracker.py

1. Log a new item
2. Suggest recycling centers
3. Mark item as recycled
4. Generate report
5. Exit
Choose an option: 1

Currently logged items:
small_electronics:
large_appliances:
batteries:

Enter the item name: Phone
Choose item type:
1. Small Electronics
2. Large Appliances
3. Batteries
Enter the number corresponding to the item type: 1
Phone added to small_electronics category.
```

-Large electronics



```
1. Log a new item
2. Suggest recycling centers
3. Mark item as recycled
4. Generate report
5. Exit
Choose an option: 1

Currently logged items:
small_electronics:
- Phone (Status: Not Dropped Off)
large_appliances:
batteries:

Enter the item name: Fridge
Choose item type:
1. Small Electronics
2. Large Appliances
3. Batteries
Enter the number corresponding to the item type: 2
Fridge added to large_appliances category.
```

-batteries

```
1. Log a new item
2. Suggest recycling centers
3. Mark item as recycled
4. Generate report
5. Exit
Choose an option: 1

Currently logged items:
small_electronics:
- Phone (Status: Not Dropped Off)
large_appliances:
- Fridge (Status: Not Dropped Off)
batteries:
```

```
Enter the item name: lithium
Choose item type:
1. Small Electronics
2. Large Appliances
3. Batteries
Enter the number corresponding to the item type: 3
lithium added to batteries category.
```

Now i will add some more in the same manner and ask for the report
After asking for the report

```
1. Log a new item
2. Suggest recycling centers
3. Mark item as recycled
4. Generate report
5. Exit
Choose an option: 4

Report:

Small_electronics:
Item Name      Count (Not Recycled) Count (Recycled)
-----
Phone          1                  0
Tab            1                  0

Large_appliances:
Item Name      Count (Not Recycled) Count (Recycled)
-----
Fridge         1                  0
AC             1                  0

Batteries:
Item Name      Count (Not Recycled) Count (Recycled)
-----
lithium        1                  0
Duracell       1                  0

1. Log a new item
```

Now i going to ask for suggestion for recycle centers

```
1. Log a new item
2. Suggest recycling centers
3. Mark item as recycled
4. Generate report
5. Exit
Choose an option: 2
Choose item type:
1. Small Electronics
2. Large Appliances
3. Batteries
Enter the number corresponding to the item type: 1
Suggested recycling centers for small_electronics: ['Center A', 'Center B', 'Center G']

1. Log a new item
2. Suggest recycling centers
3. Mark item as recycled
4. Generate report
5. Exit
Choose an option: 2
Choose item type:
1. Small Electronics
2. Large Appliances
3. Batteries
Enter the number corresponding to the item type: 2
Suggested recycling centers for large_appliances: ['Center C', 'Center D', 'Center H']

1. Log a new item
2. Suggest recycling centers
3. Mark item as recycled
4. Generate report
5. Exit
Choose an option: 2
Choose item type:
1. Small Electronics
2. Large Appliances
3. Batteries
Enter the number corresponding to the item type: 3
Suggested recycling centers for batteries: ['Center E', 'Center F', 'Center I']

1. Log a new item
2. Suggest recycling centers
3. Mark item as recycled
4. Generate report
5. Exit
Choose an option: █
```

Here it will call the **centers_api.py**

Now lets update some status for not recycled to recycled and then ask for report

```

Choose an option: 3
Choose item type:
1. Small Electronics
2. Large Appliances
3. Batteries
Enter the number corresponding to the item type: 1
Enter the item name: Phone
Phone has been marked as Dropped Off.

1. Log a new item
2. Suggest recycling centers
3. Mark item as recycled
4. Generate report
5. Exit
Choose an option: 3
Choose item type:
1. Small Electronics
2. Large Appliances
3. Batteries
Enter the number corresponding to the item type: 3
Enter the item name: Duracell
Duracell has been marked as Dropped Off.

1. Log a new item
2. Suggest recycling centers
3. Mark item as recycled
4. Generate report
5. Exit
Choose an option: 4

Report:

Small_electronics:
Item Name          Count (Not Recycled) Count (Recycled)
-----
Tab                1                  0
Phone              0                  1

Large_appliances:
Item Name          Count (Not Recycled) Count (Recycled)
-----
Fridge             1                  0
AC                 1                  0

Batteries:

```

```

Batteries:
Item Name          Count (Not Recycled) Count (Recycled)
-----
lithium            1                  0
Duracell           0                  1

1. Log a new item
2. Suggest recycling centers
3. Mark item as recycled
4. Generate report
5. Exit
Choose an option: █

```

Last is exit functionality

1. Log a new item
2. Suggest recycling centers
3. Mark item as recycled
4. Generate report
5. Exit

Choose an option: 5

PS C:\Users\Maggi\Desktop\Hackathins\aug-17>