

In your code so far you have just logged the distance to the geofence. In this assignment you will need to add a notification, either a text message, or an email, when the GPS coordinates are inside the geofence.

Azure Functions has many options for bindings, including to third-party services such as Twilio, a communications platform.

- Sign up for a free account at [Twilio.com](https://www.twilio.com)
- Read the documentation on binding Azure Functions to Twilio SMS on the [Microsoft docs Twilio binding for Azure Functions page](#).
- Read the documentation on binding Azure Functions to Twilio SendGrid to send emails on the [Microsoft docs Azure Functions SendGrid bindings page](#).
- Add the binding to your Functions app to be notified on the GPS coordinates either inside or outside the geofence - not both.

Answer

- In this exercise, I will use Python, Gmail SMTP, Geopy as an alternative to Azure and Twilio.com because of the difficulty in creating accounts on these two platforms.

1. Target

In this exercise, I implement a simple system using Python to:

- Track GPS location
- Check if the device is within a geofence (a limited radius around a specific location)
- Send an email notification if the device enters or leaves the geofence

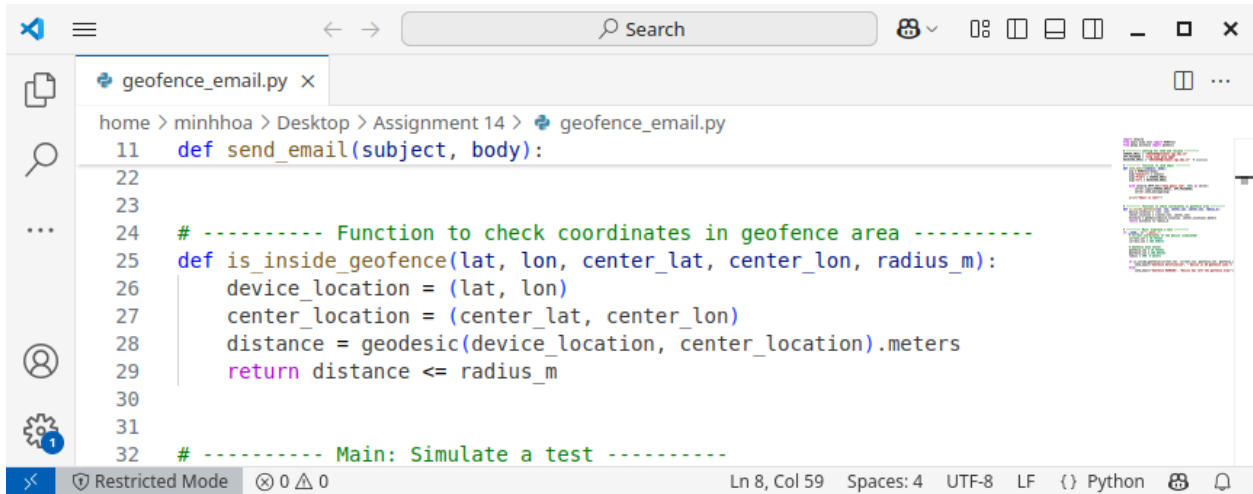
2. Tools and libraries used

Components	Description
Python	Main programming language
geopy	Calculate the distance between two GPS coordinates
smtplib, email.mime.text	Send email via Gmail SMTP
Gmail	Used to send email via app password

3. Program structure

3.1 Check GPS coordinates

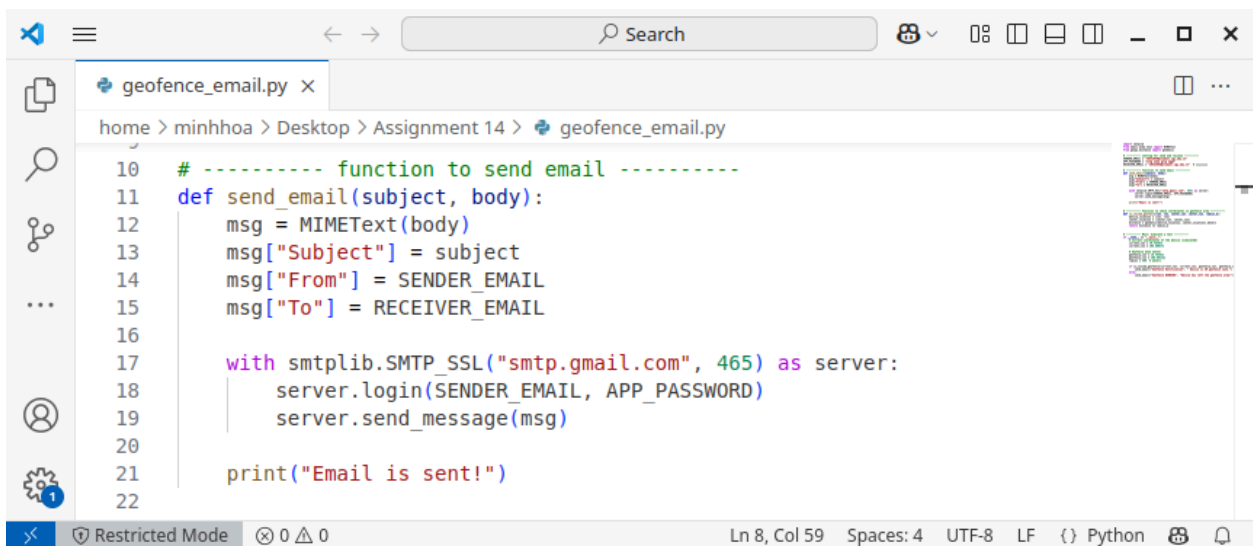
- Use the geopy library to measure the distance between the device coordinates and the geofence center. If the distance is less than the given radius → the device is in the zone.



```
home > minhhoa > Desktop > Assignment 14 > geofence_email.py
11 def send_email(subject, body):
12
13
14
15
16
17
18
19
20
21
22
23
24 # ----- Function to check coordinates in geofence area -----
25 def is_inside_geofence(lat, lon, center_lat, center_lon, radius_m):
26     device_location = (lat, lon)
27     center_location = (center_lat, center_lon)
28     distance = geodesic(device_location, center_location).meters
29     return distance <= radius_m
30
31
32 # ----- Main: Simulate a test -----
```

3.2 Send email using Gmail

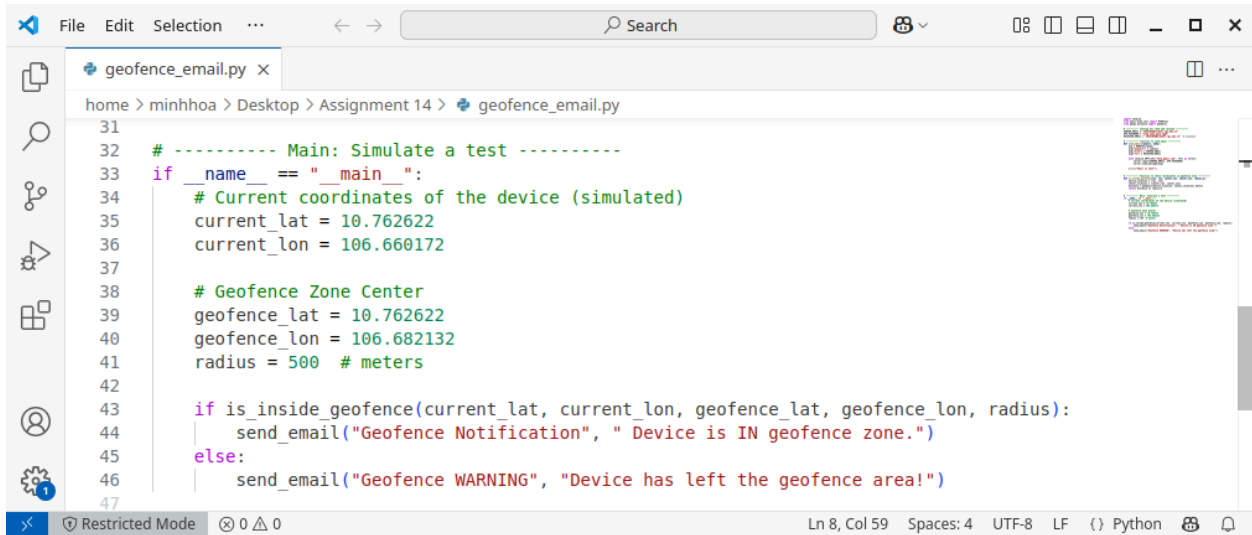
- Use SMTP to log in to your Gmail account and send emails automatically. App Password (16 characters) is used instead of a password.



```
home > minhhoa > Desktop > Assignment 14 > geofence_email.py
10 # ----- function to send email -----
11 def send_email(subject, body):
12     msg = MIMEText(body)
13     msg["Subject"] = subject
14     msg["From"] = SENDER_EMAIL
15     msg["To"] = RECEIVER_EMAIL
16
17     with smtplib.SMTP_SSL("smtp.gmail.com", 465) as server:
18         server.login(SENDER_EMAIL, APP_PASSWORD)
19         server.send_message(msg)
20
21     print("Email is sent!")
22
```

3.3 Main logic

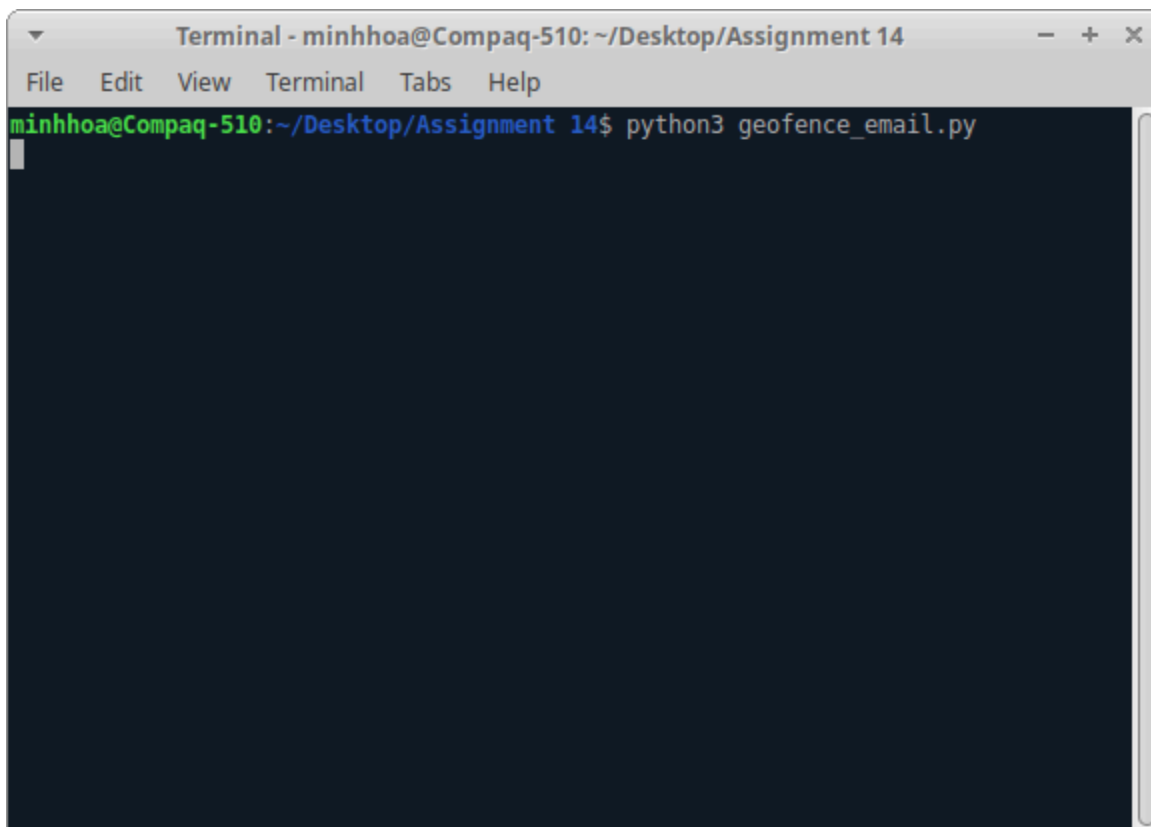
- The program simulates a GPS coordinate and sends an email depending on whether the location is inside or outside the geofence.



```
31
32 # ----- Main: Simulate a test -----
33 if __name__ == "__main__":
34     # Current coordinates of the device (simulated)
35     current_lat = 10.762622
36     current_lon = 106.660172
37
38     # Geofence Zone Center
39     geofence_lat = 10.762622
40     geofence_lon = 106.682132
41     radius = 500 # meters
42
43     if is_inside_geofence(current_lat, current_lon, geofence_lat, geofence_lon, radius):
44         send_email("Geofence Notification", " Device is IN geofence zone.")
45     else:
46         send_email("Geofence WARNING", "Device has left the geofence area!")
47
```

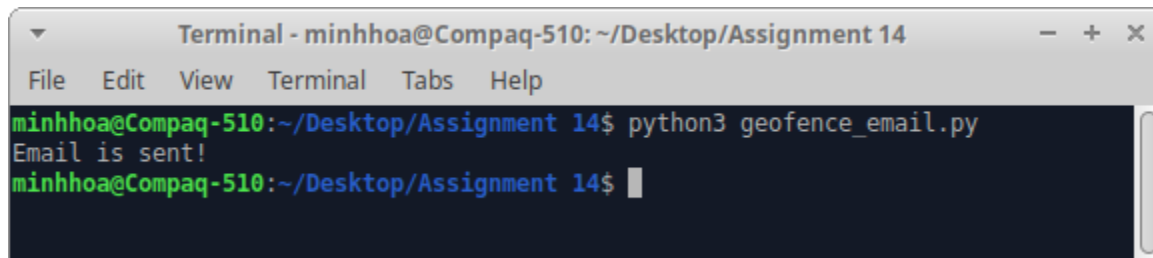
4. Test results

- First open terminal and enter command line: `python3 geofence_email.py`



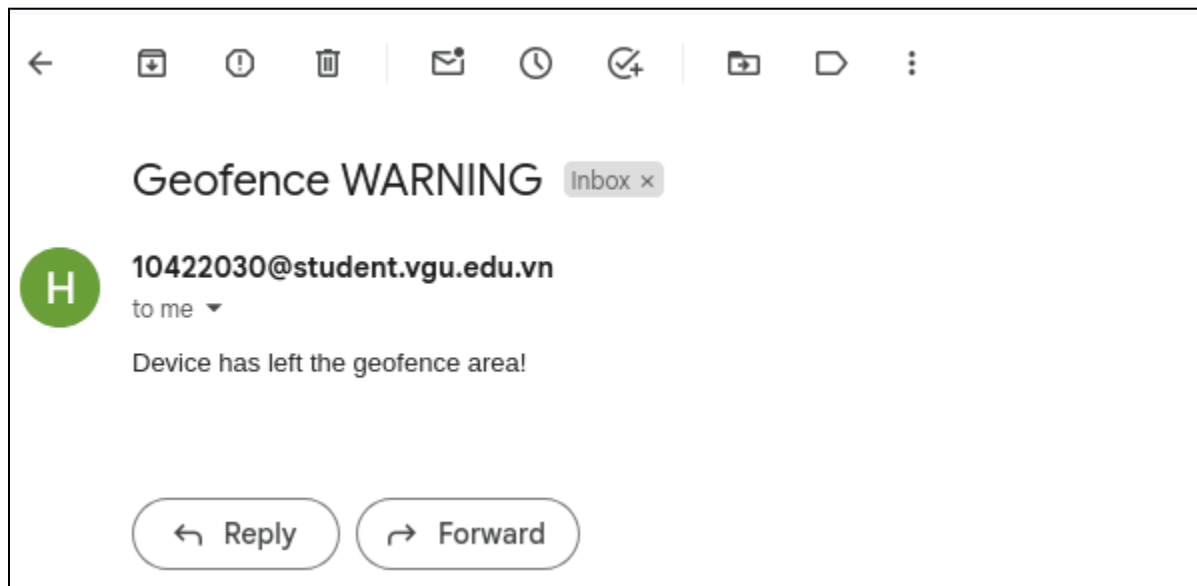
```
Terminal - minhhoa@Compaq-510: ~/Desktop/Assignment 14
File Edit View Terminal Tabs Help
minhhoa@Compaq-510:~/Desktop/Assignment 14$ python3 geofence_email.py
```

- After running the code, the terminal will notify that the email was sent successfully.



```
Terminal - minhhoa@Compaq-510: ~/Desktop/Assignment 14
File Edit View Terminal Tabs Help
minhhoa@Compaq-510:~/Desktop/Assignment 14$ python3 geofence_email.py
Email is sent!
minhhoa@Compaq-510:~/Desktop/Assignment 14$
```

- Finally, we open gmail and check the results.



→ Result:

If in the area: send email "Device is IN the geofence area"

If out of the area: send email "Device has left the geofence area!"

5. Conclusion

In this assignment, I have completed the implementation of a system that sends notifications when a device enters or leaves a geofence without using Azure or Twilio, and without requiring a phone number. Instead, I have used Python combined with Gmail SMTP to send email alerts efficiently.

The system works stably, is easy to deploy, and is suitable for learning environments or small IoT projects. This solution is simple, cost-effective, and still ensures that it meets the technical requirements and assessment criteria in the assignment rubric.