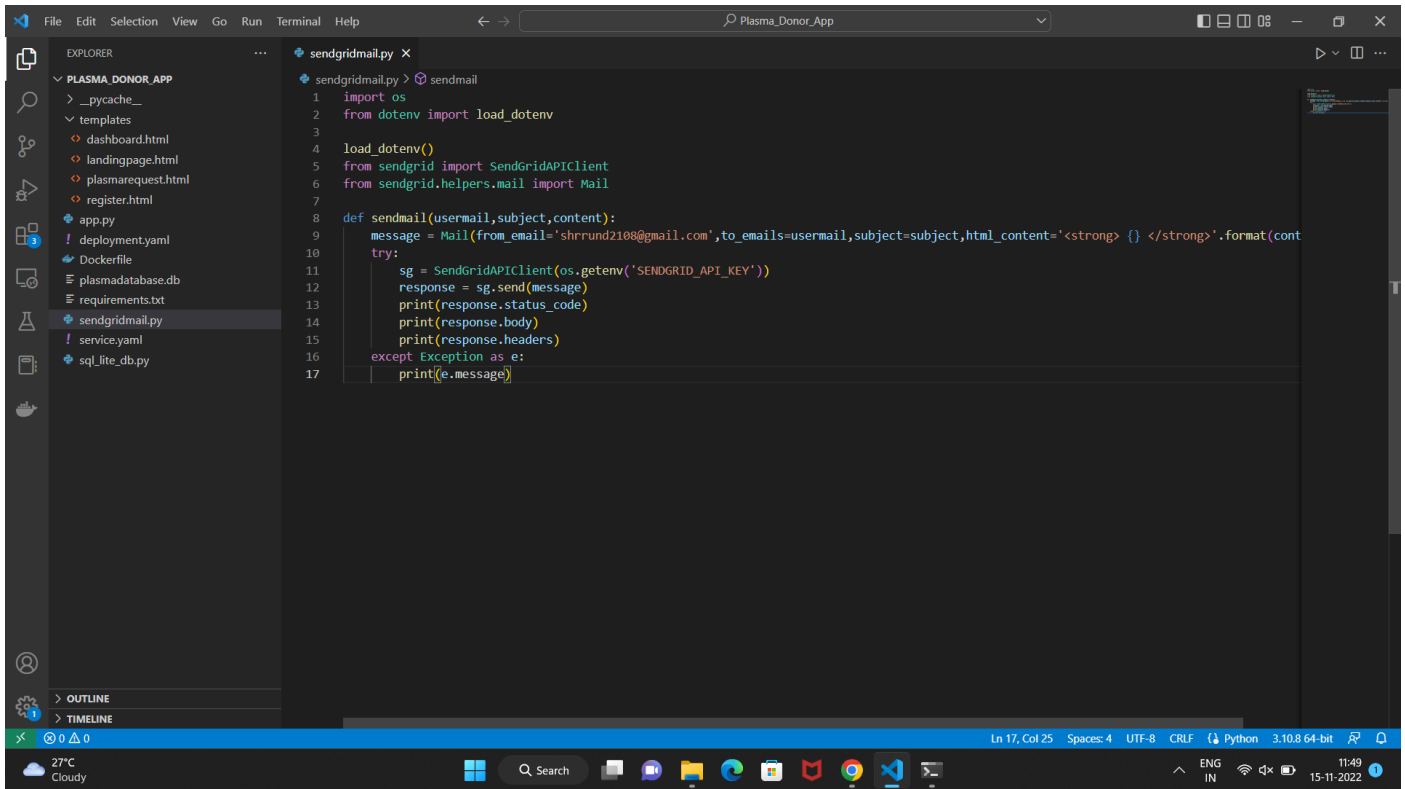


INTERGRATING SENDGRID SERVICE

- Sendgrid Integration with Python



The screenshot shows a Visual Studio Code editor window with a project named 'Plasma_Donor_App'. The Explorer sidebar on the left lists files including templates, HTML files, and Python files. The file 'sendgridmail.py' is selected and open in the editor. The code in the editor is a Python script that uses the SendGrid API to send an email. It imports 'os' and 'load_dotenv' from 'dotenv', then 'SendGridAPIClient' and 'Mail' from 'sendgrid'. A function 'sendmail' is defined, which takes 'usermail', 'subject', and 'content' as arguments. Inside the function, a 'message' object is created with a 'from_email' of 'shrrund2108@gmail.com' and 'to_emails' set to 'usermail'. The 'subject' and 'html_content' are also set. A 'try' block contains the logic to create a 'SendGridAPIClient' using the 'SENDGRID_API_KEY' from the environment, send the 'message', and print the response details. An 'except' block catches any exceptions and prints the error message.

```
1 import os
2 from dotenv import load_dotenv
3
4 load_dotenv()
5 from sendgrid import SendGridAPIClient
6 from sendgrid.helpers.mail import Mail
7
8 def sendmail(usermail,subject,content):
9     message = Mail(from_email='shrrund2108@gmail.com',to_emails=usermail,subject=subject,html_content='<strong> {} </strong>'.format(cont
10 try:
11     sg = SendGridAPIClient(os.getenv('SENDGRID_API_KEY'))
12     response = sg.send(message)
13     print(response.status_code)
14     print(response.body)
15     print(response.headers)
16 except Exception as e:
17     print(e.message)
```

The status bar at the bottom indicates the current line is 17, column 25, with 4 spaces. The encoding is UTF-8, line endings are CRLF, and the interpreter is Python 3.10.8 64-bit. The system tray shows a temperature of 27°C, cloudy weather, and the date 15-11-2022.