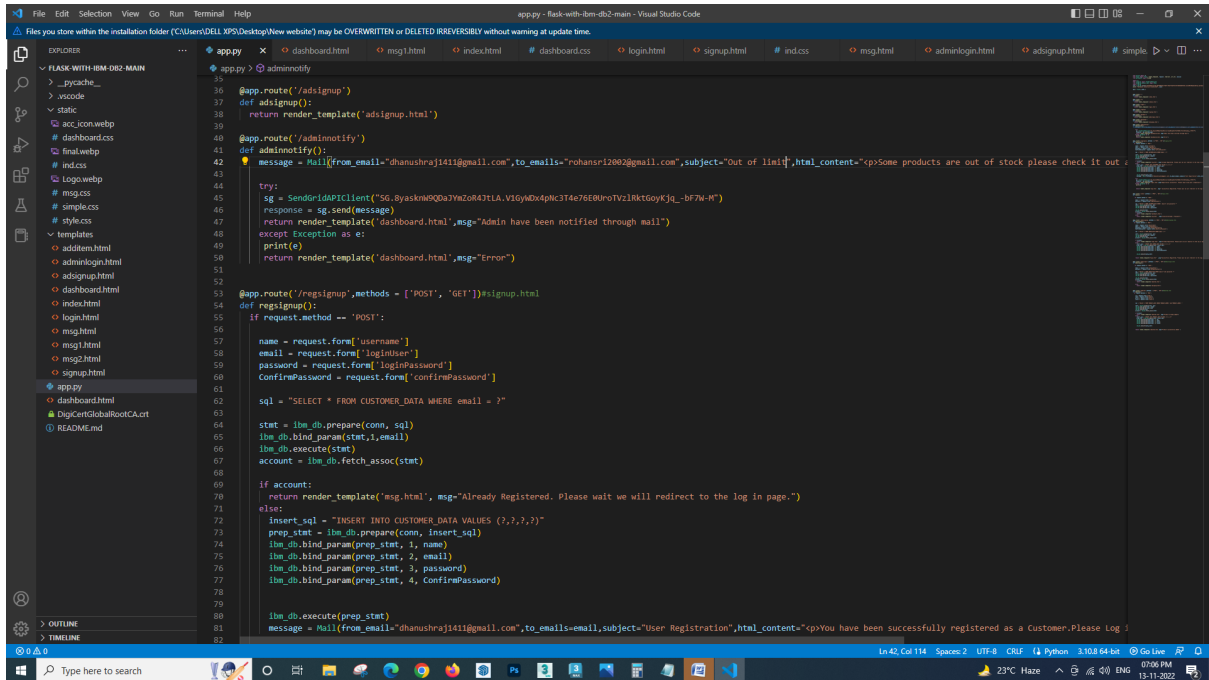


# INTEGRATING SENDGRID SERVICE

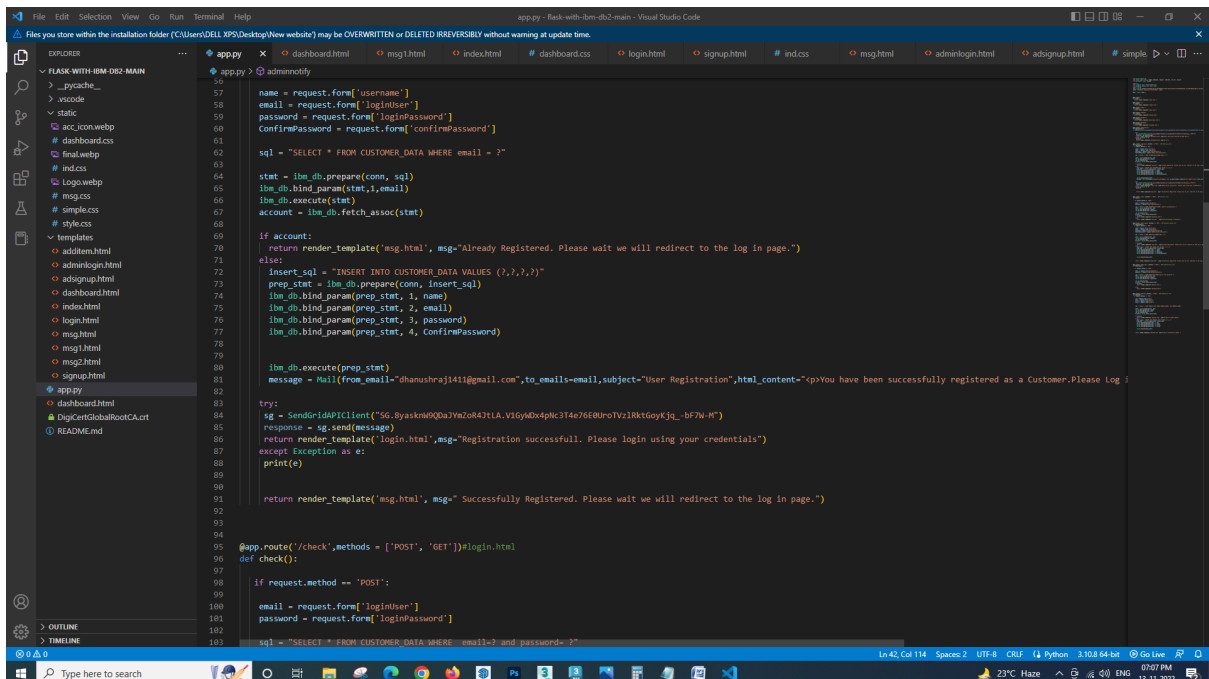
Team Id : PNT2022TMID27048

Project Name : Inventory Management For Retailers

Source Code :

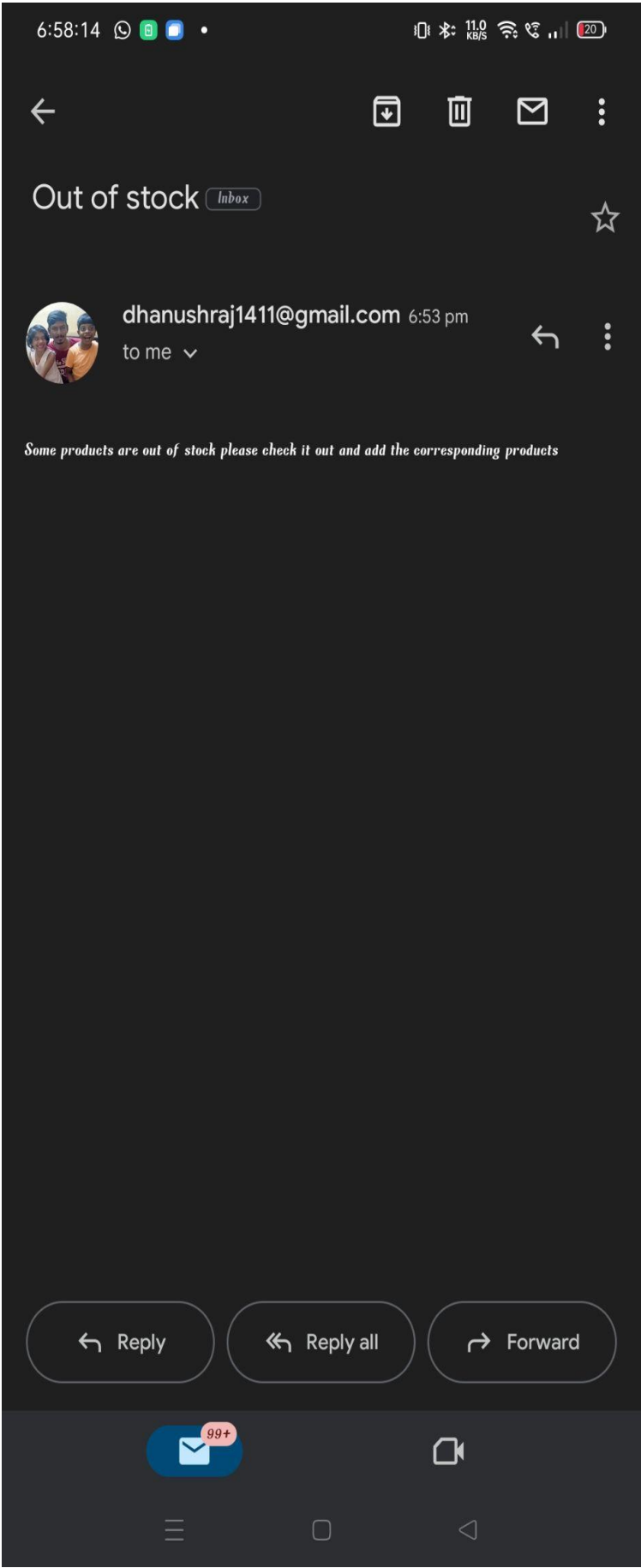


```
15 # app.py > adminnotify
16
17 @app.route('/adminnotify')
18 def adminnotify():
19     message = Mail(from_email="dhanushraj1411@gmail.com", to_emails="rohanisri2002@gmail.com", subject="Out of limit", html_content="<p>Some products are out of stock please check it out :</p>")
20     try:
21         sg = SendGridAPIClient("SG.8yaskm9QGa7YmZoR43t1LA.V16yNdxpK3I4e76I8UroTVz1RhtGoyKJq_bf7N-M")
22         response = sg.send(message)
23         return render_template("dashboard.html", msg="Admin have been notified through mail")
24     except Exception as e:
25         print(e)
26         return render_template("dashboard.html", msg="Error")
27
28 @app.route('/regsignup', methods = ['POST', 'GET'])#signup.html
29 def regsignup():
30     if request.method == 'POST':
31         name = request.form['username']
32         email = request.form['loginUser']
33         password = request.form['loginPassword']
34         ConfirmPassword = request.form['confirmPassword']
35
36         sql = "SELECT * FROM CUSTOMER_DATA WHERE email = ?"
37         stat = ibm_db.prepare(conn, sql)
38         ibm_db.bind_param(stat,1,email)
39         ibm_db.execute(stat)
40         account = ibm_db.fetch_assoc(stat)
41
42         if account:
43             return render_template("msg.html", msg="Already Registered. Please wait we will redirect to the log in page.")
44         else:
45             insert_sql = "INSERT INTO CUSTOMER_DATA VALUES (?, ?, ?, ?)"
46             prep_stat = ibm_db.prepare(conn, insert_sql)
47             ibm_db.bind_param(prep_stat, 1, name)
48             ibm_db.bind_param(prep_stat, 2, email)
49             ibm_db.bind_param(prep_stat, 3, password)
50             ibm_db.bind_param(prep_stat, 4, ConfirmPassword)
51
52             ibm_db.execute(prep_stat)
53             message = Mail(from_email="dhanushraj1411@gmail.com", to_emails=email, subject="User Registration", html_content="<p>You have been successfully registered as a Customer.Please Log i")
54             try:
55                 sg = SendGridAPIClient("SG.8yaskm9QGa7YmZoR43t1LA.V16yNdxpK3I4e76I8UroTVz1RhtGoyKJq_bf7N-M")
56                 response = sg.send(message)
57                 return render_template("login.html", msg="Registration successful. Please login using your credentials")
58             except Exception as e:
59                 print(e)
60                 return render_template("msg.html", msg="Successfully Registered. Please wait we will redirect to the log in page.")
61
62 @app.route('/check', methods = ['POST', 'GET'])#login.html
63 def check():
64     if request.method == 'POST':
65         email = request.form['loginUser']
66         password = request.form['loginPassword']
67         sql = "SELECT * FROM CUSTOMER_DATA WHERE email=? and password= ?"
```



```
15 # app.py > adminnotify
16
17 @app.route('/adminnotify')
18 def adminnotify():
19     message = Mail(from_email="dhanushraj1411@gmail.com", to_emails="rohanisri2002@gmail.com", subject="Out of limit", html_content="<p>Some products are out of stock please check it out :</p>")
20     try:
21         sg = SendGridAPIClient("SG.8yaskm9QGa7YmZoR43t1LA.V16yNdxpK3I4e76I8UroTVz1RhtGoyKJq_bf7N-M")
22         response = sg.send(message)
23         return render_template("dashboard.html", msg="Admin have been notified through mail")
24     except Exception as e:
25         print(e)
26         return render_template("dashboard.html", msg="Error")
27
28 @app.route('/regsignup', methods = ['POST', 'GET'])#signup.html
29 def regsignup():
30     if request.method == 'POST':
31         name = request.form['username']
32         email = request.form['loginUser']
33         password = request.form['loginPassword']
34         ConfirmPassword = request.form['confirmPassword']
35
36         sql = "SELECT * FROM CUSTOMER_DATA WHERE email = ?"
37         stat = ibm_db.prepare(conn, sql)
38         ibm_db.bind_param(stat,1,email)
39         ibm_db.execute(stat)
40         account = ibm_db.fetch_assoc(stat)
41
42         if account:
43             return render_template("msg.html", msg="Already Registered. Please wait we will redirect to the log in page.")
44         else:
45             insert_sql = "INSERT INTO CUSTOMER_DATA VALUES (?, ?, ?, ?)"
46             prep_stat = ibm_db.prepare(conn, insert_sql)
47             ibm_db.bind_param(prep_stat, 1, name)
48             ibm_db.bind_param(prep_stat, 2, email)
49             ibm_db.bind_param(prep_stat, 3, password)
50             ibm_db.bind_param(prep_stat, 4, ConfirmPassword)
51
52             ibm_db.execute(prep_stat)
53             message = Mail(from_email="dhanushraj1411@gmail.com", to_emails=email, subject="User Registration", html_content="<p>You have been successfully registered as a Customer.Please Log i")
54             try:
55                 sg = SendGridAPIClient("SG.8yaskm9QGa7YmZoR43t1LA.V16yNdxpK3I4e76I8UroTVz1RhtGoyKJq_bf7N-M")
56                 response = sg.send(message)
57                 return render_template("login.html", msg="Registration successful. Please login using your credentials")
58             except Exception as e:
59                 print(e)
60                 return render_template("msg.html", msg="Successfully Registered. Please wait we will redirect to the log in page.")
61
62 @app.route('/check', methods = ['POST', 'GET'])#login.html
63 def check():
64     if request.method == 'POST':
65         email = request.form['loginUser']
66         password = request.form['loginPassword']
67         sql = "SELECT * FROM CUSTOMER_DATA WHERE email=? and password= ?"
```

Received Mail :





## User Registration

Inbox



dhanushraj1411@g... 6:59 PM  
to me ▾



You have been successfully registered as a Customer. Please Log in to see our Products.

↩ Reply

↩↩ Reply all

➦ Forward

