

SPRINT-4

Date	16 November 2022
Team ID	PNT2022TMID08681
Project Name	Smart fashion Recommender application

Sendmail.py

```
Import smtplib
```

```
Import sendgrid
```

```
Import os
```

```
From sendgrid.helpers.mail import Mail, Email, To, Content
```

```
SUBJECT = "Interview Call"
```

```
S = smtplib.SMTP('smtp.gmail.com', 587)
```

```
Def sendmail(TEXT,email):
```

```
    Print("sorry we cant process your candidature") S
```

```
    = smtplib.SMTP('smtp.gmail.com', 587)
```

```
    s.starttls()
```

```
    s.login(il.pradeepthi@gmail.com, "oms@1Ram") message
```

```
    = 'Subject: {}\n\n{}'.format(SUBJECT, TEXT)
```

```
    s.sendmail(il.pradeepthi@gmail.com, email, message) s.quit()
```

```
def sendgridmail(user,TEXT):
```

```
    sg =
```

```
    sendgrid.SendGridAPIClient('SG.nouVVZMwQTSYtih73r1TxQ.3H0kajWkEYpo0RV1iarxSVKbqvtj  
y
```

```
Z_nhPbKi3zeZnc') from_email = Email(sandeep@thesmartbridge.com) # Change to your
```

```
    verified sender to_email = To(user) # Change to your recipient
```

```
subject = "Sending with SendGrid is Fun" content
= Content("text/plain",TEXT)
mail = Mail(from_email, to_email, subject, content)

# Get a JSON-ready representation of the Mail object
Mail_json = mail.get()

# Send an HTTP POST request to /mail/send
Response = sg.client.mail.send.post(request_body=mail_json)
Print(response.status_code)
Print(response.headers)
```

Deployment.yaml

```
apiVersion: apps/v1
kind: Deployment
metadata: name:
jobportal
spec:
  replicas: 1
  selector:
    matchLabels:
      app: flasknode
  template:
    metadata:
      labels:
        app: flasknode
```

spec:

containers:

- name: flasknode image:
icr.io/helloworld1/newhelloworld
imagePullPolicy: Always ports:
 - containerPort: 5000

Service.yaml

apiVersion: v1

kind: Service

metadata:

name: flask-node-deployment

spec:

ports:

- port: 5000 targetPort: 5000 selector:
app: flasknode