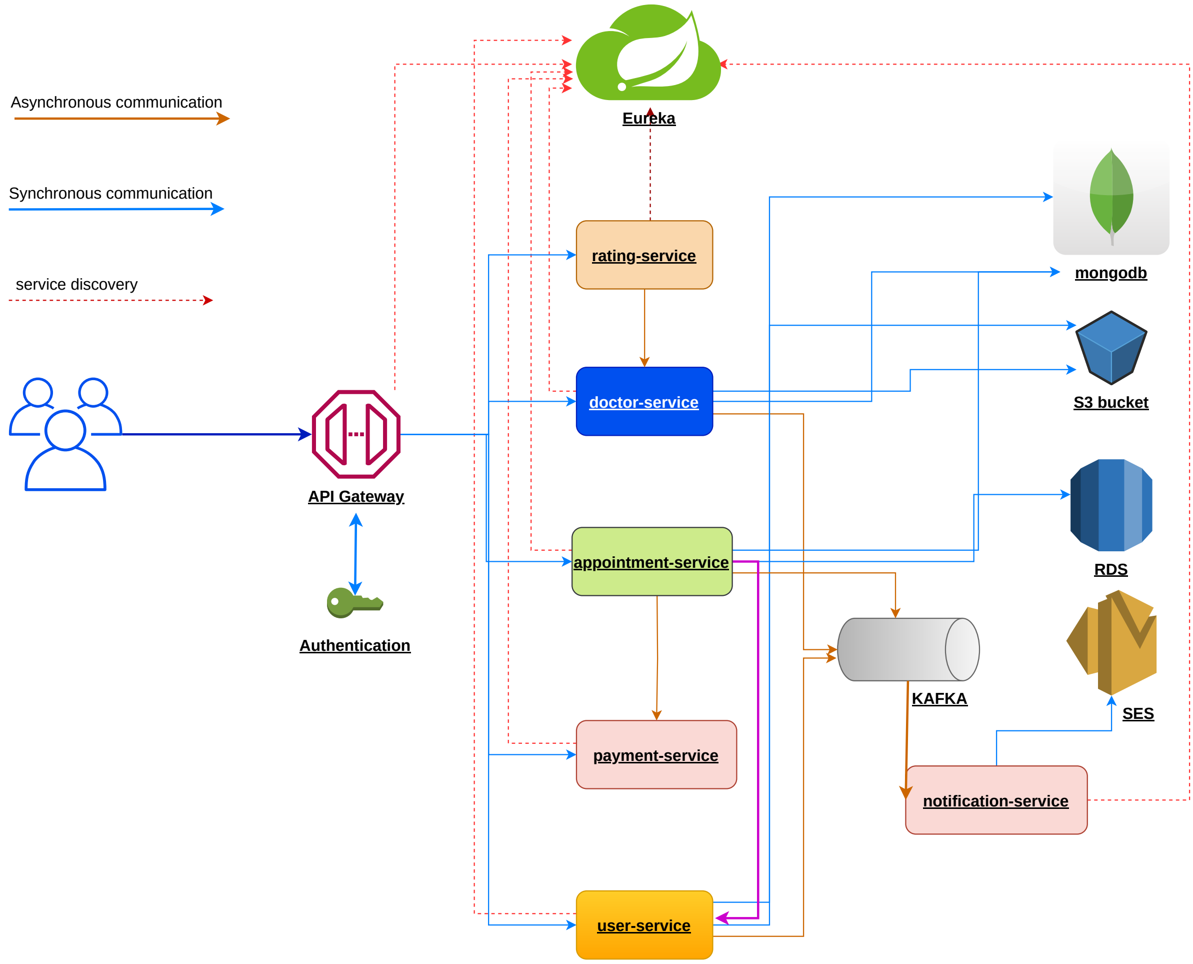
CodeLogic.pdf

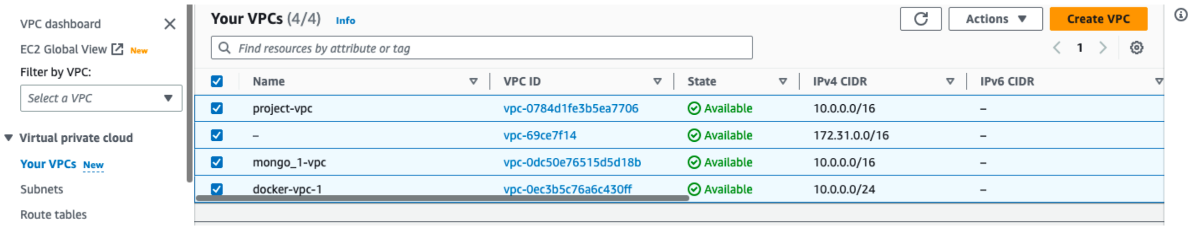
Architecture



Setup Required to run application

VPC setup:

- create an `vpc` first before creating the resources in aws so we can use same vpc for all the resources.

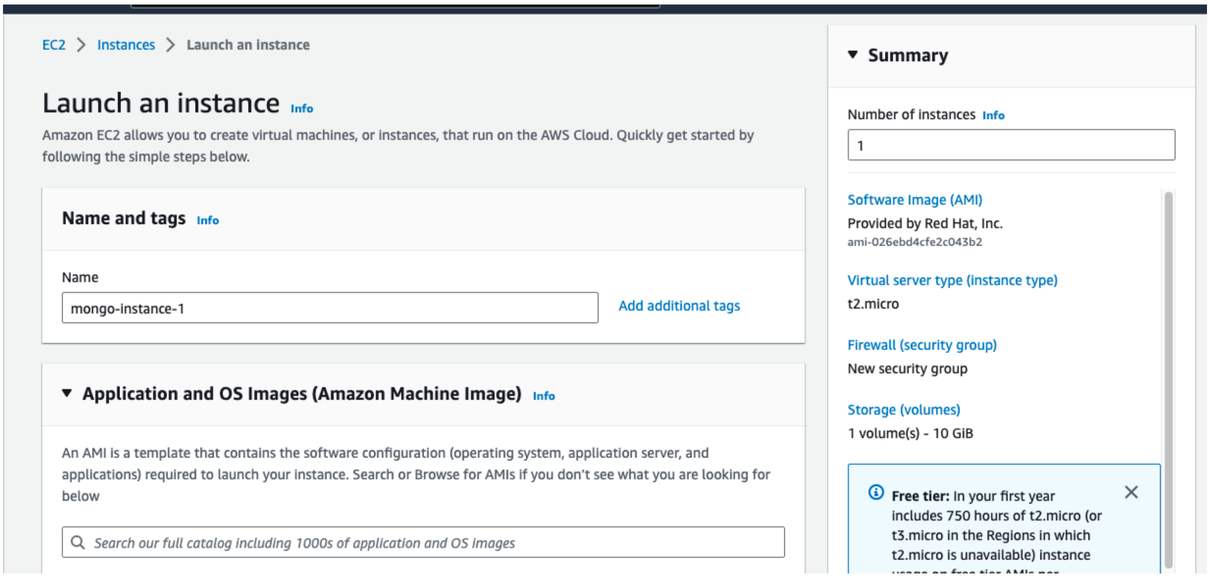


Your VPCs (4/4) Info

☑	Name	VPC ID	State	IPv4 CIDR	IPv6 CIDR
☑	project-vpc	vpc-0784d1fe3b5ea7706	Available	10.0.0.0/16	–
☑	–	vpc-69ce7f14	Available	172.31.0.0/16	–
☑	mongo_1-vpc	vpc-0dc50e76515d5d18b	Available	10.0.0.0/16	–
☑	docker-vpc-1	vpc-0ec3b5c76a6c430ff	Available	10.0.0.0/24	–

MongoDb setup:

- we create an `ec2` instance of `t2-micro` and install mongodb in that



Launch an instance

Amazon EC2 allows you to create virtual machines, or instances, that run on the AWS Cloud. Quickly get started by following the simple steps below.

Name and tags

Name: mongo-instance-1

Application and OS Images (Amazon Machine Image)

An AMI is a template that contains the software configuration (operating system, application server, and applications) required to launch your instance. Search or Browse for AMIs if you don't see what you are looking for below

Search our full catalog including 1000s of application and OS images

Summary

Number of instances: 1

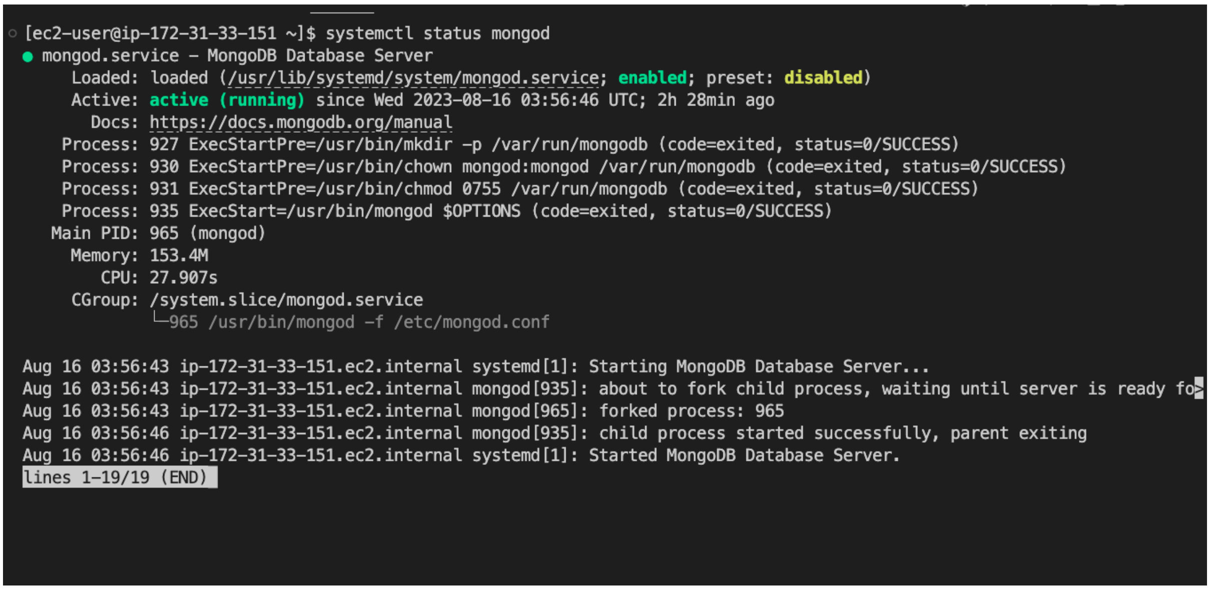
Software Image (AMI)
Provided by Red Hat, Inc.
ami-026ebd4cfe2c043b2

Virtual server type (instance type)
t2.micro

Firewall (security group)
New security group

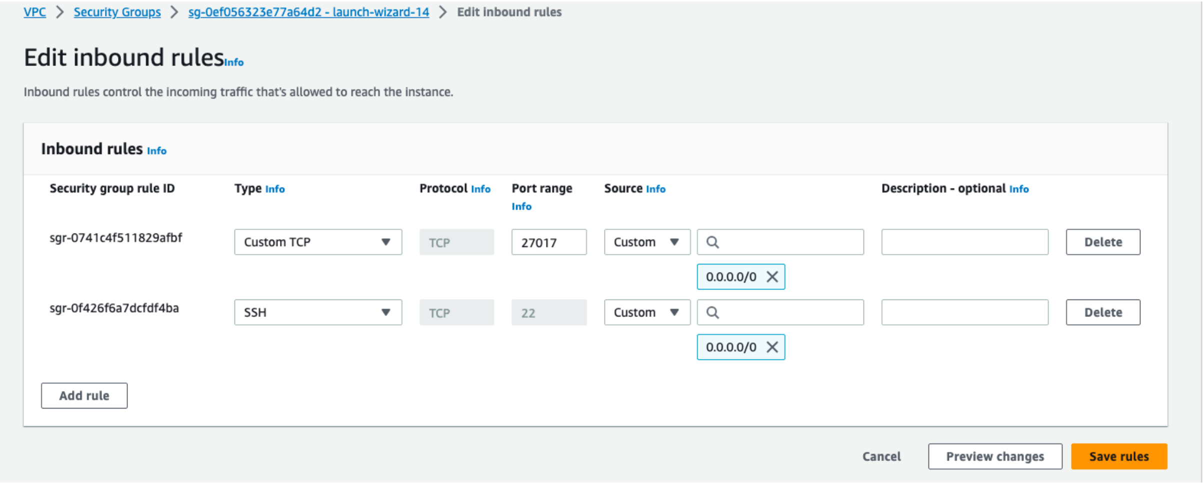
Storage (volumes)
1 volume(s) - 10 GiB

Free tier: In your first year includes 750 hours of t2.micro (or t3.micro in the Regions in which t2.micro is unavailable) instance



```
○ [ec2-user@ip-172-31-33-151 ~]$ systemctl status mongod
 ● mongod.service - MongoDB Database Server
      Loaded: loaded (/usr/lib/systemd/system/mongod.service; enabled; preset: disabled)
      Active: active (running) since Wed 2023-08-16 03:56:46 UTC; 2h 28min ago
        Docs: https://docs.mongodb.org/manual
     Process: 927 ExecStartPre=/usr/bin/mkdir -p /var/run/mongodb (code=exited, status=0/SUCCESS)
     Process: 930 ExecStartPre=/usr/bin/chown mongod:mongod /var/run/mongodb (code=exited, status=0/SUCCESS)
     Process: 931 ExecStartPre=/usr/bin/chmod 0755 /var/run/mongodb (code=exited, status=0/SUCCESS)
     Process: 935 ExecStart=/usr/bin/mongod $OPTIONS (code=exited, status=0/SUCCESS)
    Main PID: 965 (mongod)
      Memory: 153.4M
         CPU: 27.907s
      CGroup: /system.slice/mongod.service
              └─965 /usr/bin/mongod -f /etc/mongod.conf

 Aug 16 03:56:43 ip-172-31-33-151.ec2.internal systemd[1]: Starting MongoDB Database Server...
 Aug 16 03:56:43 ip-172-31-33-151.ec2.internal mongod[935]: about to fork child process, waiting until server is ready fo>
 Aug 16 03:56:43 ip-172-31-33-151.ec2.internal mongod[965]: forked process: 965
 Aug 16 03:56:46 ip-172-31-33-151.ec2.internal mongod[935]: child process started successfully, parent exiting
 Aug 16 03:56:46 ip-172-31-33-151.ec2.internal systemd[1]: Started MongoDB Database Server.
 lines 1-19/19 (END)
```

Change security group to accept external request for mongo port `27017`



VPC > Security Groups > sg-0ef056323e77a64d2 - launch-wizard-14 > Edit inbound rules

Edit inbound rules Info

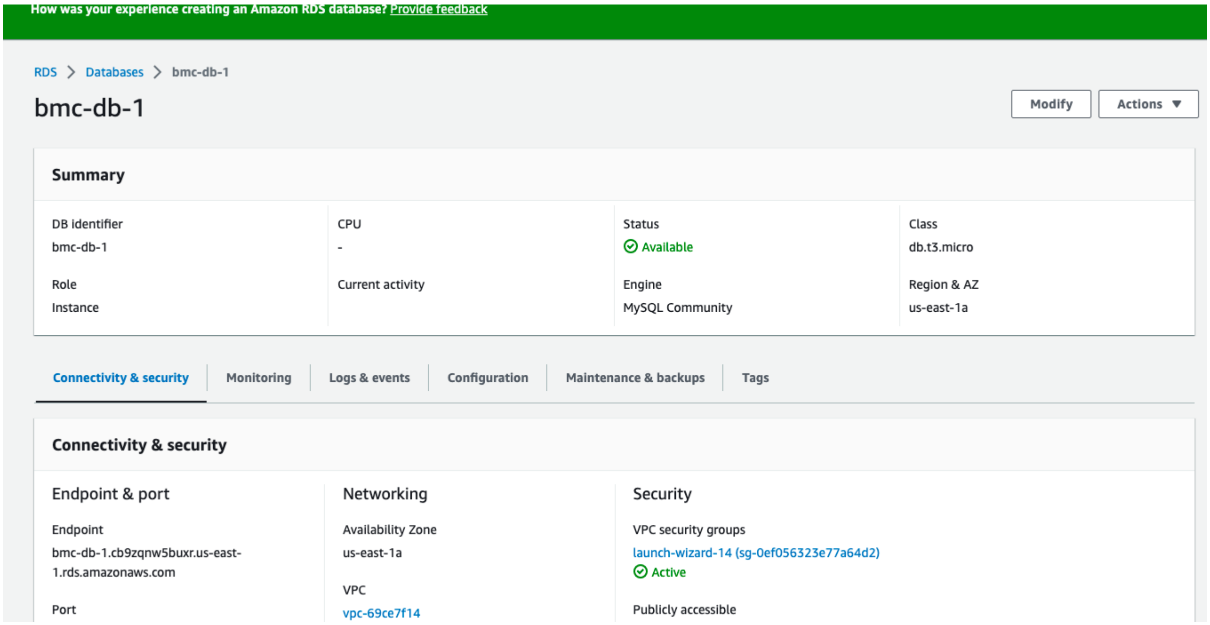
Inbound rules control the incoming traffic that's allowed to reach the instance.

Inbound rules Info

Security group rule ID	Type Info	Protocol Info	Port range Info	Source Info		Description - optional Info	
sgr-0741c4f511829afbf	Custom TCP	TCP	27017	Custom	0.0.0.0/0		Delete
sgr-0f426f6a7dcfdf4ba	SSH	TCP	22	Custom	0.0.0.0/0		Delete

RDS setup:

- we required relational database for running our application so set we setup `DynamoDb` `rds` instance



How was your experience creating an Amazon RDS database? Provide feedback

RDS > Databases > bmc-db-1

bmc-db-1

Summary

DB identifier bmc-db-1	CPU -	Status Available	Class db.t3.micro
Role Instance	Current activity	Engine MySQL Community	Region & AZ us-east-1a

Connectivity & security

Endpoint & port

Endpoint
bmc-db-1.cb9zqnw5buxr.us-east-1.rds.amazonaws.com

Port

Networking

Availability Zone
us-east-1a

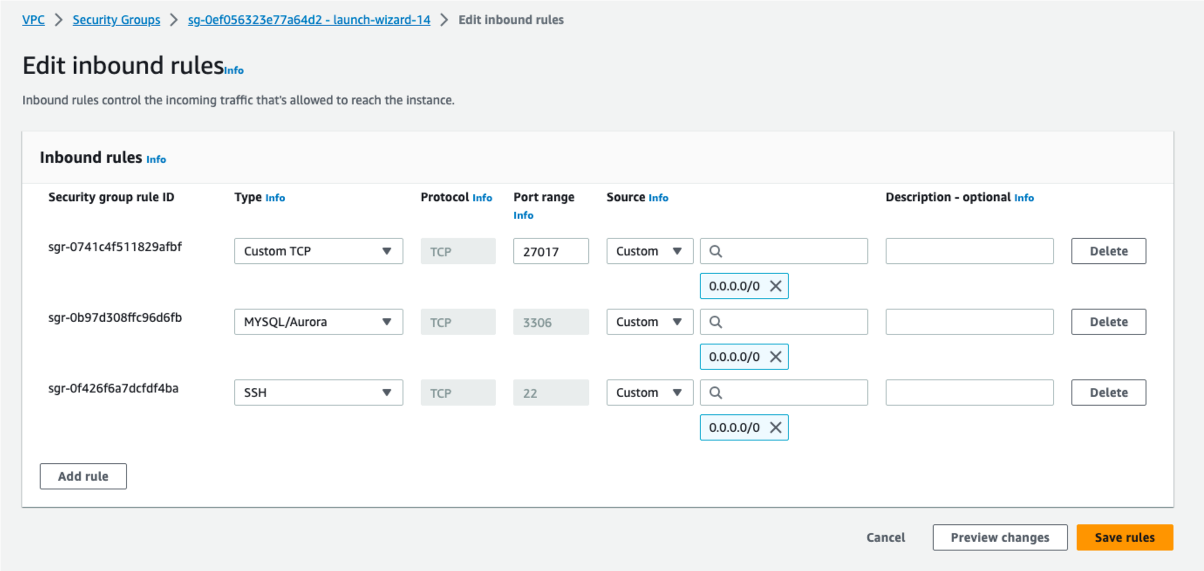
VPC
vpc-69ce7f14

Security

VPC security groups
launch-wizard-14 (sg-0ef056323e77a64d2)
Active

Publicly accessible

Change security group to accept external request for `rds` port `3306`



VPC > Security Groups > sg-0ef056323e77a64d2 - launch-wizard-14 > Edit inbound rules

Edit inbound rules Info

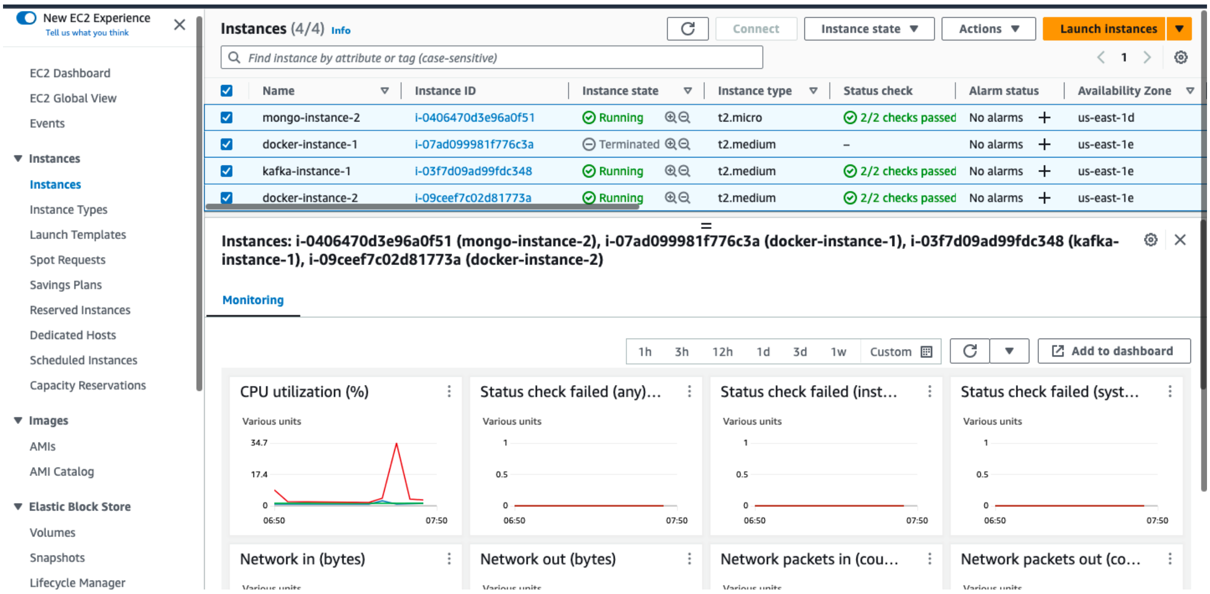
Inbound rules control the incoming traffic that's allowed to reach the instance.

Inbound rules Info

Security group rule ID	Type Info	Protocol Info	Port range Info	Source Info		Description - optional Info	
sgr-0741c4f511829afbf	Custom TCP	TCP	27017	Custom	0.0.0.0/0		Delete
sgr-0b97d308ffc96d6fb	MYSQL/Aurora	TCP	3306	Custom	0.0.0.0/0		Delete
sgr-0f426f6a7dcfdf4ba	SSH	TCP	22	Custom	0.0.0.0/0		Delete

Kafka Setup

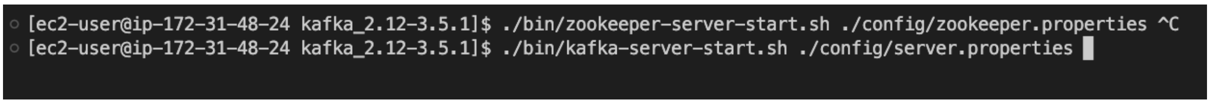
- we are using external kafka server for our application.
- so created one more `ec2` instance of `t2-medium` and installed kafka there



Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Availability Zone
mongo-instance-2	i-0406470d3e96a0f51	Running	t2.micro	2/2 checks passed	No alarms	us-east-1d
docker-instance-1	i-07ad099981f776c3a	Terminated	t2.medium	–	No alarms	us-east-1e
kafka-instance-1	i-03f7d09ad99fdc348	Running	t2.medium	2/2 checks passed	No alarms	us-east-1e
docker-instance-2	i-09ceef7c02d81773a	Running	t2.medium	2/2 checks passed	No alarms	us-east-1e

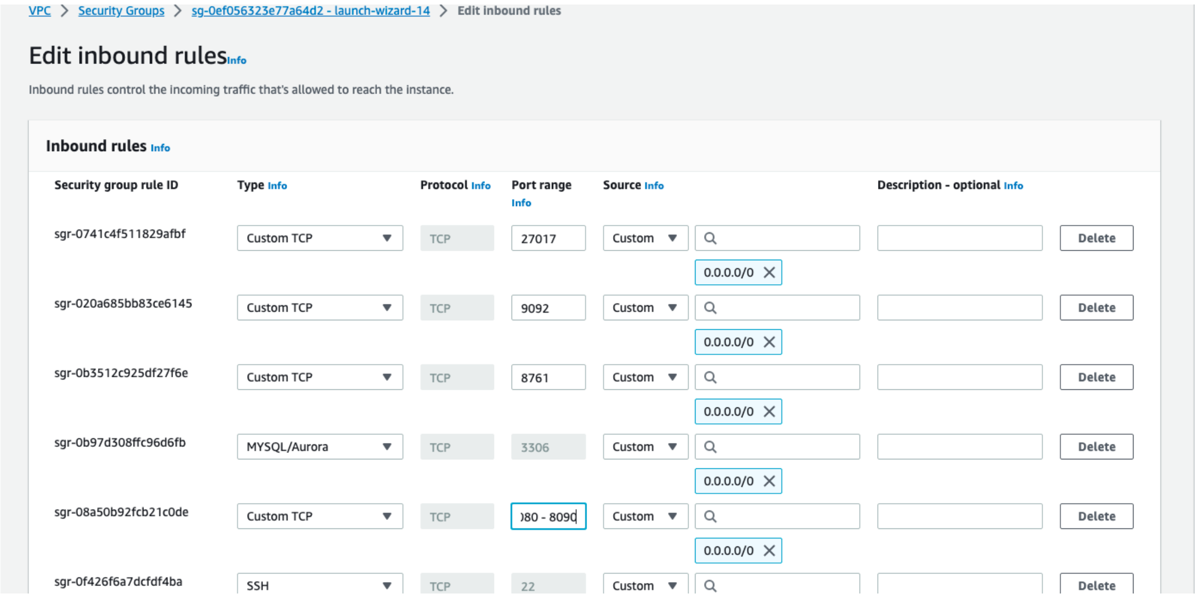
Instances: i-0406470d3e96a0f51 (mongo-instance-2), i-07ad099981f776c3a (docker-instance-1), i-03f7d09ad99fdc348 (kafka-instance-1), i-09ceef7c02d81773a (docker-instance-2)

start kafka from binaries



```
○ [ec2-user@ip-172-31-48-24 kafka_2.12-3.5.1]$ ./bin/zookeeper-server-start.sh ./config/zookeeper.properties ^C
○ [ec2-user@ip-172-31-48-24 kafka_2.12-3.5.1]$ ./bin/kafka-server-start.sh ./config/server.properties █
```

edit security group to access kafka from external open port `9092`



VPC > Security Groups > sg-0ef056323e77a64d2 - launch-wizard-14 > Edit inbound rules

Edit inbound rules Info

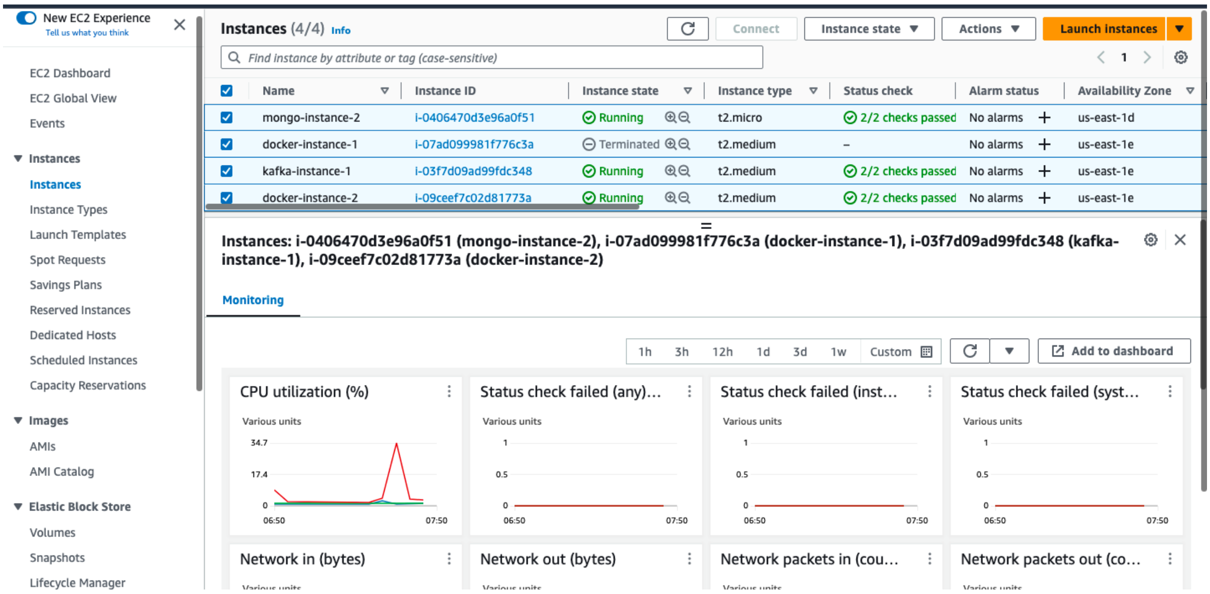
Inbound rules control the incoming traffic that's allowed to reach the instance.

Inbound rules Info

Security group rule ID	Type Info	Protocol Info	Port range Info	Source Info		Description - optional Info	
sgr-0741c4f511829afbf	Custom TCP	TCP	27017	Custom	0.0.0.0/0		Delete
sgr-020a685bb83ce6145	Custom TCP	TCP	9092	Custom	0.0.0.0/0		Delete
sgr-0b3512c925df27f6e	Custom TCP	TCP	8761	Custom	0.0.0.0/0		Delete
sgr-0b97d308ffc96d6fb	MYSQL/Aurora	TCP	3306	Custom	0.0.0.0/0		Delete
sgr-08a50b92fcb21c0de	Custom TCP	TCP	)80 - 8090	Custom	0.0.0.0/0		Delete
sgr-0f426f6a7dcfdf4ba	SSH	TCP	22	Custom			Delete

Docker setup

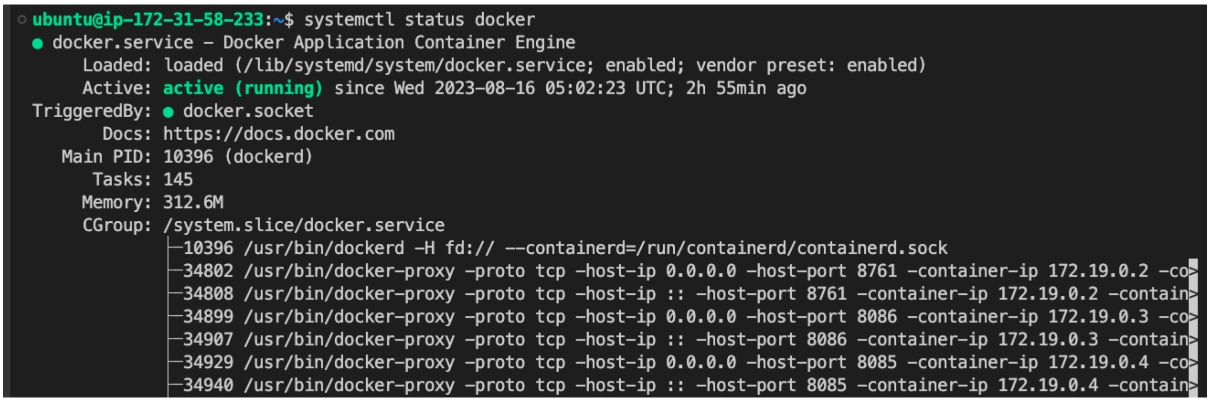
- to setup docker we created a one more instance with `t2-medium`
- then installed docker



Instances (4/4) Info

☑	Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Availability Zone
☑	mongo-instance-2	i-0406470d3e96a0f51	Running	t2.micro	2/2 checks passed	No alarms	us-east-1d
☑	docker-instance-1	i-07ad099981f776c3a	Terminated	t2.medium	–	No alarms	us-east-1e
☑	kafka-instance-1	i-03f7d09ad99fdc348	Running	t2.medium	2/2 checks passed	No alarms	us-east-1e
☑	docker-instance-2	i-09ceef7c02d81773a	Running	t2.medium	2/2 checks passed	No alarms	us-east-1e

Instances: i-0406470d3e96a0f51 (mongo-instance-2), i-07ad099981f776c3a (docker-instance-1), i-03f7d09ad99fdc348 (kafka-instance-1), i-09ceef7c02d81773a (docker-instance-2)

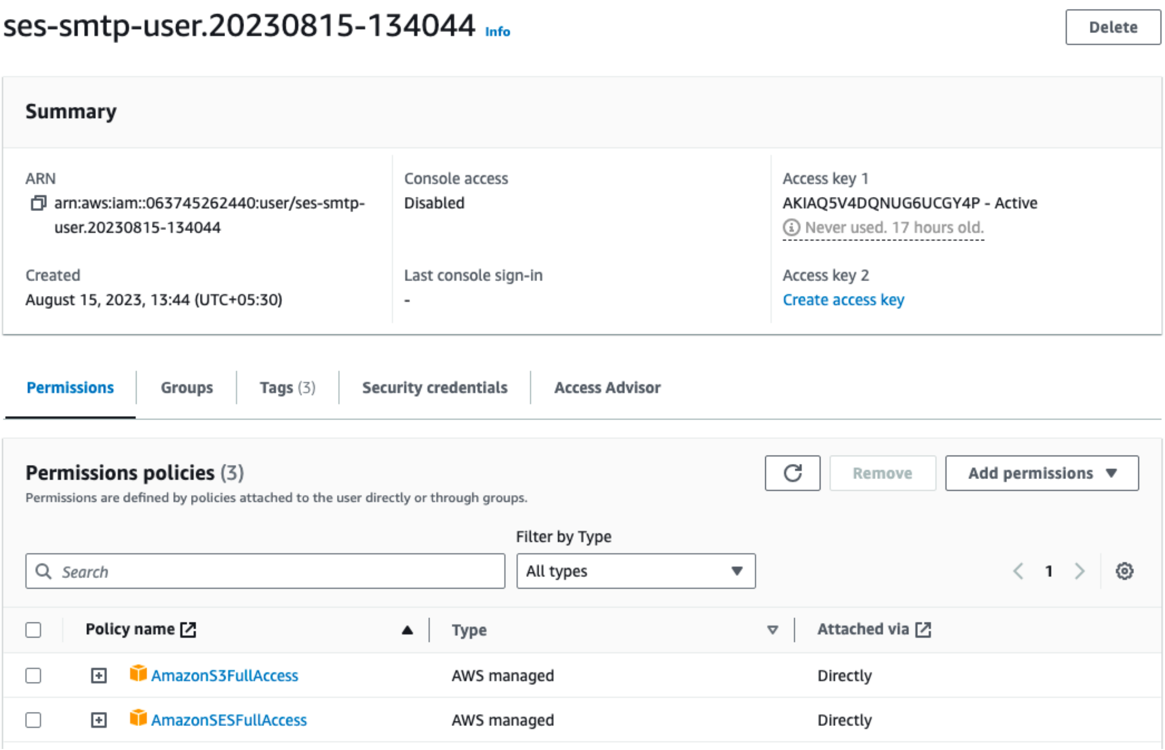


```
○ ubuntu@ip-172-31-58-233:~$ systemctl status docker
 ● docker.service - Docker Application Container Engine
      Loaded: loaded (/lib/systemd/system/docker.service; enabled; vendor preset: enabled)
      Active: active (running) since Wed 2023-08-16 05:02:23 UTC; 2h 55min ago
 TriggeredBy: ● docker.socket
        Docs: https://docs.docker.com
    Main PID: 10396 (dockerd)
       Tasks: 145
      Memory: 312.6M
      CGroup: /system.slice/docker.service
              ├─10396 /usr/bin/dockerd -H fd:// --containerd=/run/containerd/containerd.sock
              ├─34802 /usr/bin/docker-proxy -proto tcp -host-ip 0.0.0.0 -host-port 8761 -container-ip 172.19.0.2 -co>
              ├─34808 /usr/bin/docker-proxy -proto tcp -host-ip :: -host-port 8761 -container-ip 172.19.0.2 -contain>
              ├─34899 /usr/bin/docker-proxy -proto tcp -host-ip 0.0.0.0 -host-port 8086 -container-ip 172.19.0.3 -co>
              ├─34907 /usr/bin/docker-proxy -proto tcp -host-ip :: -host-port 8086 -container-ip 172.19.0.3 -contain>
              ├─34929 /usr/bin/docker-proxy -proto tcp -host-ip 0.0.0.0 -host-port 8085 -container-ip 172.19.0.4 -co>
              ├─34940 /usr/bin/docker-proxy -proto tcp -host-ip :: -host-port 8085 -container-ip 172.19.0.4 -contain>
```

- make sure `rds` , `mongodb` and `kafka` services must be accessible from `docker instance`

IAM user setup for SES and S3 access

- create an `IAM` user with full access to `SES` and `S3`
- create a `Access Key` for that user and save the `Access Key` and `Password`
- This is required to be passed to the application while running



ses-smtp-user.20230815-134044 Info

Delete

Summary

ARN
arn:aws:iam::063745262440:user/ses-smtp-user.20230815-134044

Console access
Disabled

Access key 1
AKIAQ5V4DQNUG6UCGY4P - Active
Never used. 17 hours old.

Created
August 15, 2023, 13:44 (UTC+05:30)

Last console sign-in
-

Access key 2
Create access key

Permissions | Groups | Tags (3) | Security credentials | Access Advisor

Permissions policies (3)

Permissions are defined by policies attached to the user directly or through groups.

Remove | Add permissions ▼

Search

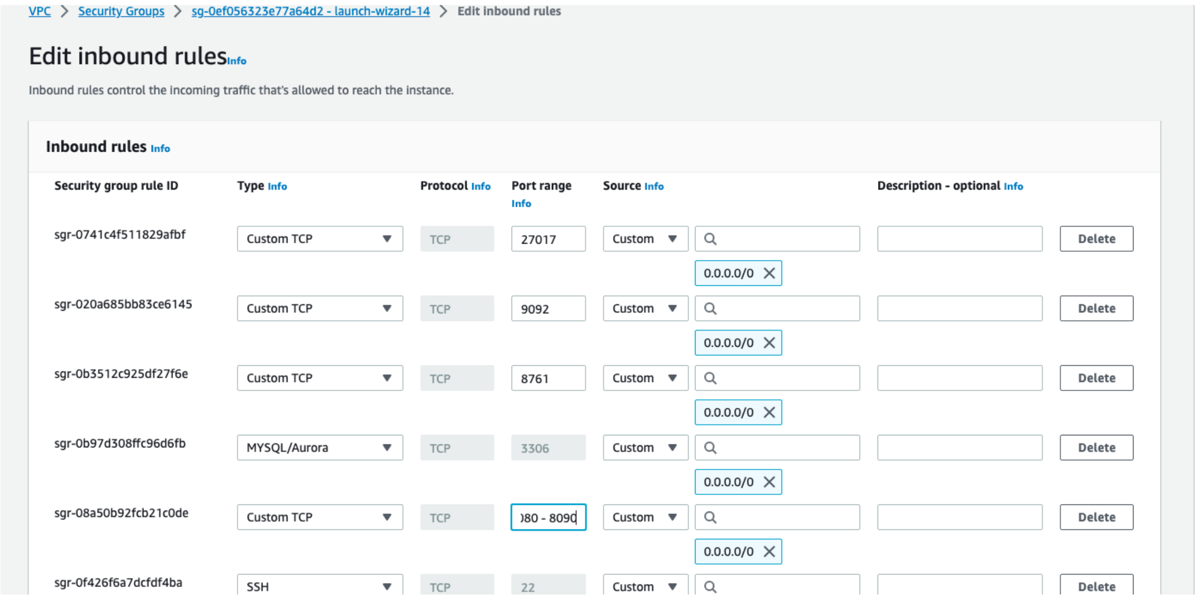
Filter by Type: All types

□	Policy name	Type	Attached via
□	AmazonS3FullAccess	AWS managed	Directly
□	AmazonSESFullAccess	AWS managed	Directly

open ports for external access to access the apis

→ ports `8761` for eureka service to run

→ ports `8080` to `8090` for applications to expose the services



VPC > Security Groups > sg-0ef056323e77a64d2 - launch-wizard-14 > Edit inbound rules

Edit inbound rules Info

Inbound rules control the incoming traffic that's allowed to reach the instance.

Inbound rules Info

Security group rule ID	Type Info	Protocol Info	Port range Info	Source Info		Description - optional Info	
sgr-0741c4f511829afbf	Custom TCP	TCP	27017	Custom	0.0.0.0/0		Delete
sgr-020a685bb83ce6145	Custom TCP	TCP	9092	Custom	0.0.0.0/0		Delete
sgr-0b3512c925df27f6e	Custom TCP	TCP	8761	Custom	0.0.0.0/0		Delete
sgr-0b97d308ffc96d6fb	MYSQL/Aurora	TCP	3306	Custom	0.0.0.0/0		Delete
sgr-08a50b92fcb21c0de	Custom TCP	TCP	)80 - 8090	Custom	0.0.0.0/0		Delete
sgr-0f426f6a7dcfdf4ba	SSH	TCP	22	Custom			Delete

Running an application

steps to run the application:

1. upload the `BookMyConsultation.zip` to the `docker instance`
2. unzip the file and navigate to the file
3. before running need to give parameter to the application to start

   Only `docker-compose.yml` file need to be changed

   1. `mongo host` and `port` changes

      → replace `MONGO_HOST` value with `mongo running instance id`

      → replace `MONGO_PORT` value with port where mongo is running in mongo host

      → replace `MONGO_DATABASE` with mongo db name

   2. `MYSQL RDS` changes.

      → replace `MYSQL_HOST` value with `rds public IP`

      → replace `MYSQL_HOST_PORT` value with mysql port set by you

      → replace `MYSQL_USER` value with admin user details set for rds

      → replace `MYSQL_PASSWORD` value with admin user password set for rds

      → replace `MySQL_DATABASE` value with database name

   3. `KAFKA` changes

      → replace `KAFKA_HOST` value with kafka instance ip where kafka can be accessed from.

      → replace `KAFKA_HOST_PORT` value with kafka port value set by you

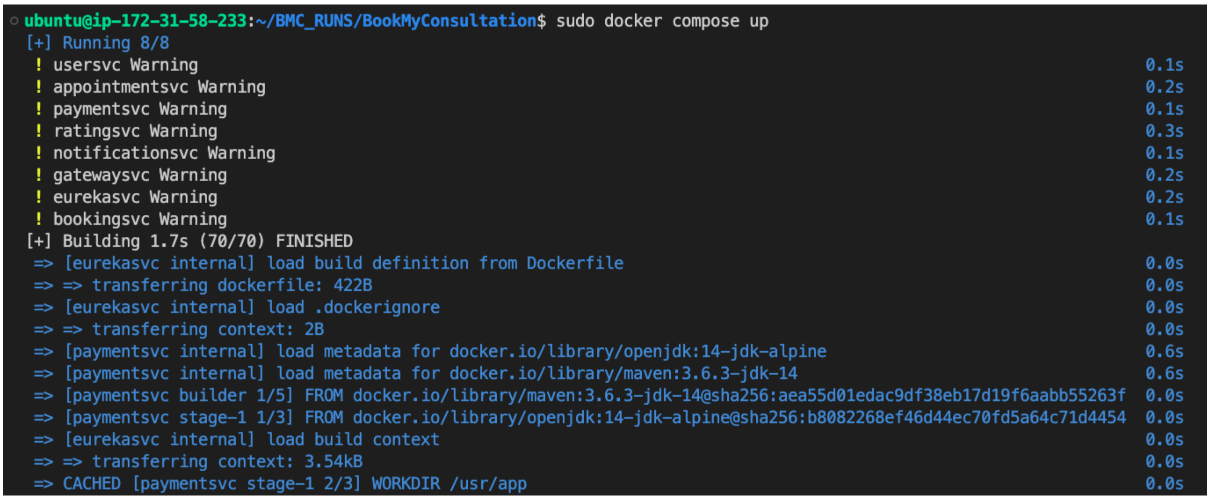
   4. `SES` and `S3` `access key` and `secret key`

      → replace values of `SES_ACCESSKEY` and `S3_ACCESS_KEY` with the token access token name generated

→ replace values of `SES_SECRETKEY` and `S3_SECRET_KEY` with the password generated for the access token

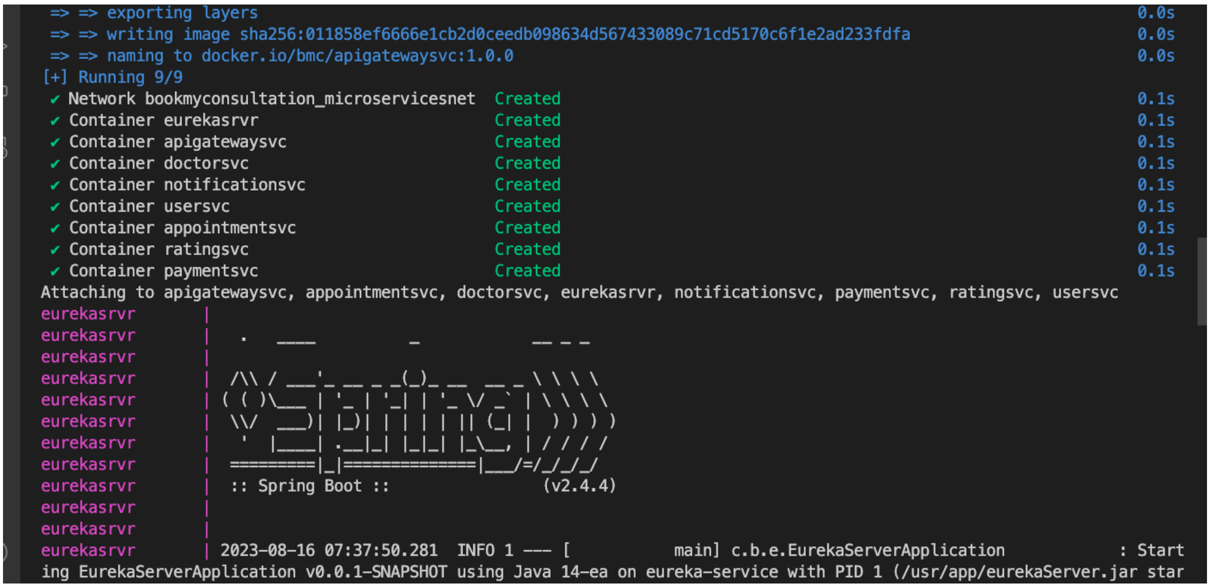
→ replace the `SES_FROM_EMAIL` with the email from which emails need to be sent this should be verified email.

4. After properly replacing all the values we are ready to run the application.
5. execute command `sudo docker compose up`
6. it will references the `Dockerfile` present in each services home folder and build the image one-by one and start the containers.



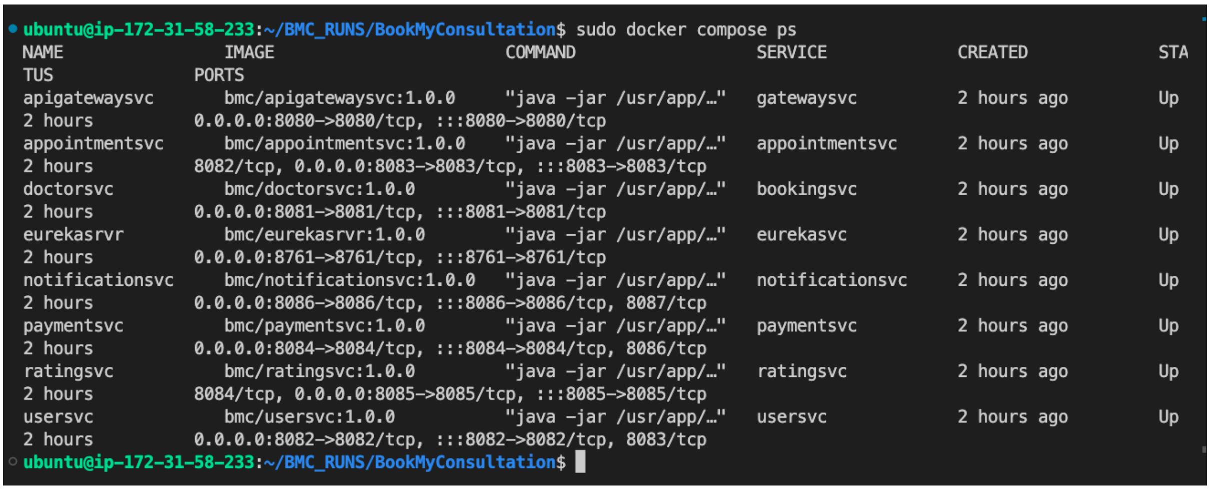
```
○ ubuntu@ip-172-31-58-233:~/BMC_RUNS/BookMyConsultation$ sudo docker compose up
  [+] Running 8/8
   ! usersvc Warning                                                                                                     0.1s
   ! appointmentsvc Warning                                                                                              0.2s
   ! paymentsvc Warning                                                                                                  0.1s
   ! ratingsvc Warning                                                                                                   0.3s
   ! notificationsvc Warning                                                                                             0.1s
   ! gatewaysvc Warning                                                                                                  0.2s
   ! eurekasvc Warning                                                                                                   0.2s
   ! bookingsvc Warning                                                                                                  0.1s
  [+] Building 1.7s (70/70) FINISHED
   => [eurekasvc internal] load build definition from Dockerfile                                                         0.0s
   => => transferring dockerfile: 422B                                                                                   0.0s
   => [eurekasvc internal] load .dockerignore                                                                            0.0s
   => => transferring context: 2B                                                                                        0.0s
   => [paymentsvc internal] load metadata for docker.io/library/openjdk:14-jdk-alpine                                    0.6s
   => [paymentsvc internal] load metadata for docker.io/library/maven:3.6.3-jdk-14                                       0.6s
   => [paymentsvc builder 1/5] FROM docker.io/library/maven:3.6.3-jdk-14@sha256:aea55d01edac9df38eb17d19f6aabb55263f     0.0s
   => [paymentsvc stage-1 1/3] FROM docker.io/library/openjdk:14-jdk-alpine@sha256:b8082268ef46d44ec70fd5a64c71d4454     0.0s
   => [eurekasvc internal] load build context                                                                            0.0s
   => => transferring context: 3.54kB                                                                                    0.0s
   => CACHED [paymentsvc stage-1 2/3] WORKDIR /usr/app                                                                   0.0s
```

you can see applications coming up one by one



```
 => => exporting layers                                                                               0.0s
 => => writing image sha256:011858ef6666e1cb2d0ceedb098634d567433089c71cd5170c6f1e2ad233fdfa            0.0s
 => => naming to docker.io/bmc/apigatewaysvc:1.0.0                                                    0.0s
[+] Running 9/9
 ✔ Network bookmyconsultation_microservicesnet  Created                                               0.1s
 ✔ Container eurekasrvr                         Created                                               0.1s
 ✔ Container apigatewaysvc                      Created                                               0.1s
 ✔ Container doctorsvc                          Created                                               0.1s
 ✔ Container notificationsvc                    Created                                               0.1s
 ✔ Container usersvc                            Created                                               0.1s
 ✔ Container appointmentsvc                     Created                                               0.1s
 ✔ Container ratingsvc                          Created                                               0.1s
 ✔ Container paymentsvc                         Created                                               0.1s
Attaching to apigatewaysvc, appointmentsvc, doctorsvc, eurekasrvr, notificationsvc, paymentsvc, ratingsvc, usersvc
eurekasrvr       |
eurekasrvr       |   .   ____          _            __ _ _
eurekasrvr       |
eurekasrvr       |  /\\ / ___'_ __ _ _(_)_ __  __ _ \ \ \ \
eurekasrvr       | ( ( )\___ | '_ | '_| | '_ \/ _` | \ \ \ \
eurekasrvr       |  \\/  ___)| |_)| | | | | || (_| |  ) ) ) )
eurekasrvr       |   '  |____| .__|_| |_|_| |_\__, | / / / /
eurekasrvr       |  =========|_|==============|___/=/_/_/_/
eurekasrvr       |  :: Spring Boot ::                (v2.4.4)
eurekasrvr       |
eurekasrvr       |
eurekasrvr       | 2023-08-16 07:37:50.281  INFO 1 --- [           main] c.b.e.EurekaServerApplication            : Start
ing EurekaServerApplication v0.0.1-SNAPSHOT using Java 14-ea on eureka-service with PID 1 (/usr/app/eurekaServer.jar star
```

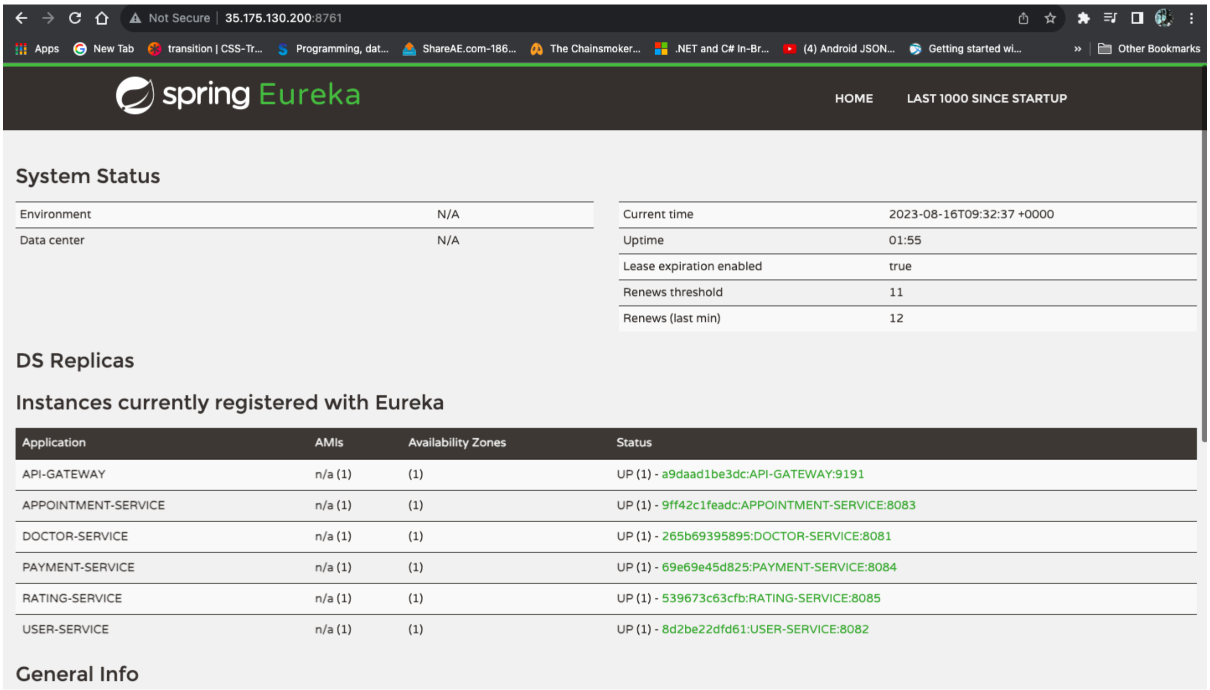
once all applications are up we can see the containers running as shown below



```
● ubuntu@ip-172-31-58-233:~/BMC_RUNS/BookMyConsultation$ sudo docker compose ps
 NAME                IMAGE                       COMMAND                  SERVICE             CREATED             STA
 TUS             PORTS
 apigatewaysvc       bmc/apigatewaysvc:1.0.0     "java -jar /usr/app/…"   gatewaysvc          2 hours ago         Up
 2 hours         0.0.0.0:8080->8080/tcp, :::8080->8080/tcp
 appointmentsvc      bmc/appointmentsvc:1.0.0    "java -jar /usr/app/…"   appointmentsvc      2 hours ago         Up
 2 hours         8082/tcp, 0.0.0.0:8083->8083/tcp, :::8083->8083/tcp
 doctorsvc           bmc/doctorsvc:1.0.0         "java -jar /usr/app/…"   bookingsvc          2 hours ago         Up
 2 hours         0.0.0.0:8081->8081/tcp, :::8081->8081/tcp
 eurekasrvr          bmc/eurekasrvr:1.0.0        "java -jar /usr/app/…"   eurekasvc           2 hours ago         Up
 2 hours         0.0.0.0:8761->8761/tcp, :::8761->8761/tcp
 notificationsvc     bmc/notificationsvc:1.0.0   "java -jar /usr/app/…"   notificationsvc     2 hours ago         Up
 2 hours         0.0.0.0:8086->8086/tcp, :::8086->8086/tcp, 8087/tcp
 paymentsvc          bmc/paymentsvc:1.0.0        "java -jar /usr/app/…"   paymentsvc          2 hours ago         Up
 2 hours         0.0.0.0:8084->8084/tcp, :::8084->8084/tcp, 8086/tcp
 ratingsvc           bmc/ratingsvc:1.0.0         "java -jar /usr/app/…"   ratingsvc           2 hours ago         Up
 2 hours         8084/tcp, 0.0.0.0:8085->8085/tcp, :::8085->8085/tcp
 usersvc             bmc/usersvc:1.0.0           "java -jar /usr/app/…"   usersvc             2 hours ago         Up
 2 hours         0.0.0.0:8082->8082/tcp, :::8082->8082/tcp, 8083/tcp
○ ubuntu@ip-172-31-58-233:~/BMC_RUNS/BookMyConsultation$ 
```

now we can see the services in `eureka service discovery`

visit eureka by following url `http://<docker-ec2-public-IP>:8761` replace `docker-ec2-public-IP` with proper ip address of where you are running docker compose



System Status

Environment	N/A
Data center	N/A

Current time	2023-08-16T09:32:37 +0000
Uptime	01:55
Lease expiration enabled	true
Renews threshold	11
Renews (last min)	12

DS Replicas

Instances currently registered with Eureka

Application	AMIs	Availability Zones	Status
API-GATEWAY	n/a (1)	(1)	UP (1) - a9daad1be3dc:API-GATEWAY:9191
APPOINTMENT-SERVICE	n/a (1)	(1)	UP (1) - 9ff42c1feadc:APPOINTMENT-SERVICE:8083
DOCTOR-SERVICE	n/a (1)	(1)	UP (1) - 265b69395895:DOCTOR-SERVICE:8081
PAYMENT-SERVICE	n/a (1)	(1)	UP (1) - 69e69e45d825:PAYMENT-SERVICE:8084
RATING-SERVICE	n/a (1)	(1)	UP (1) - 539673c63cfb:RATING-SERVICE:8085
USER-SERVICE	n/a (1)	(1)	UP (1) - 8d2be22dfd61:USER-SERVICE:8082

General Info

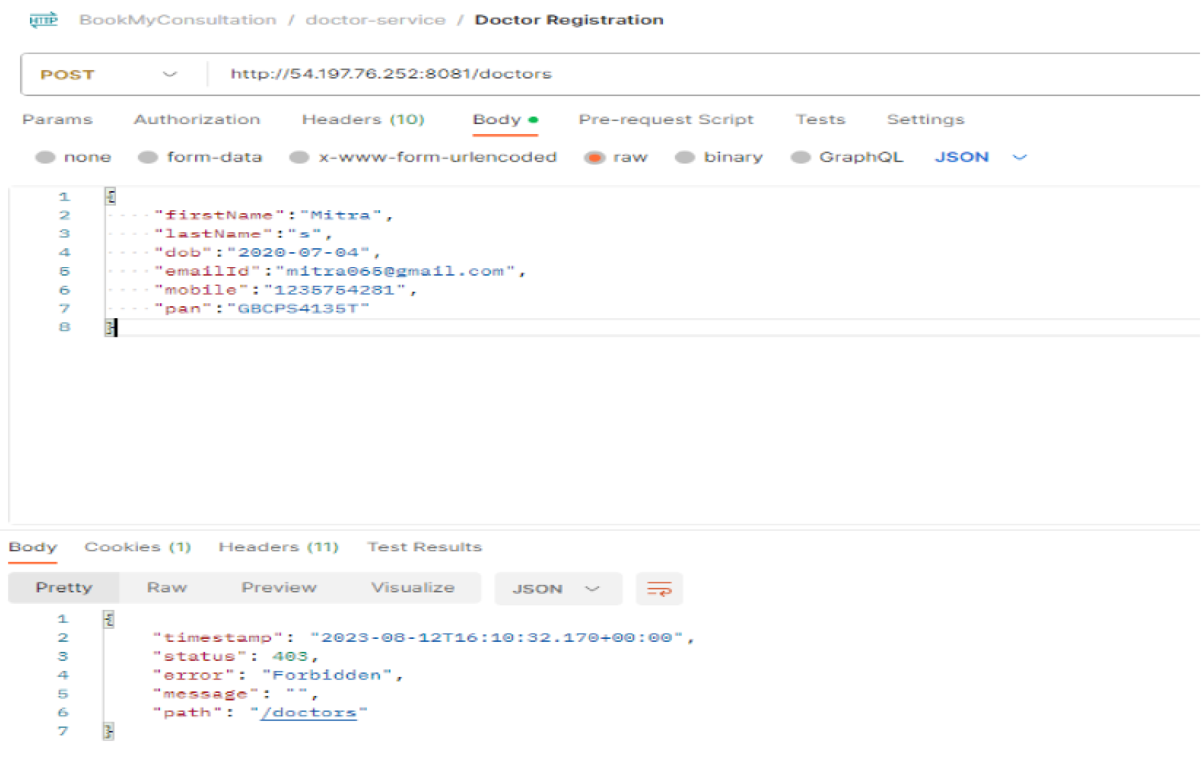
to bring down the services `sudo docker compose down`

API test screen shots

1. Doctor-Onboarding Service

Endpoint 1: URI: `/doctors`
This endpoint is responsible for collecting information about the doctor.

- Endpoint is forbidden, for roles other than User, Admin

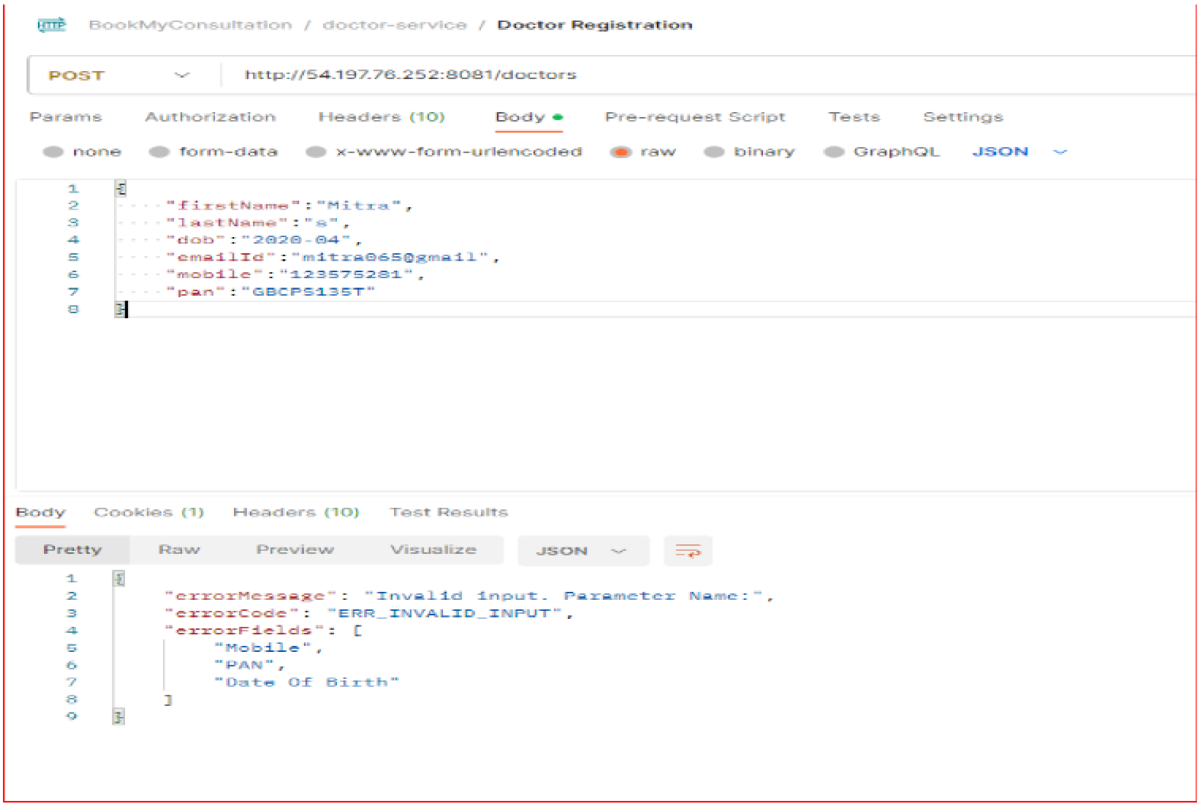


POST http://54.197.76.252:8081/doctors

```
{
    "firstName":"Mitra",
    "lastName":"s",
    "dob":"2020-07-04",
    "emailId":"mitra065@gmail.com",
    "mobile":"1235754281",
    "pan":"GBCPS4135T"
}
```

```
{
    "timestamp": "2023-08-12T16:10:32.170+00:00",
    "status": 403,
    "error": "Forbidden",
    "message": "",
    "path": "/doctors"
}
```

- Endpoint is forbidden, for roles other than User, Admin

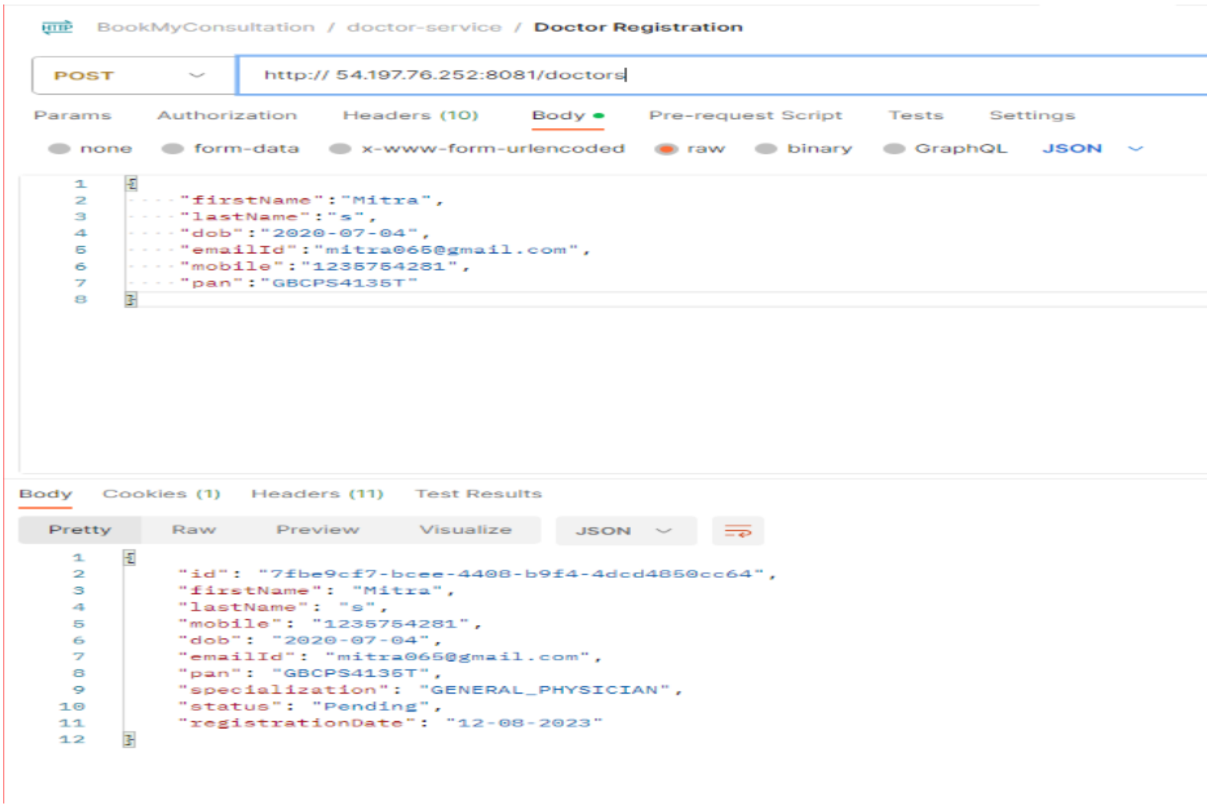


POST http://54.197.76.252:8081/doctors

```
{
    "firstName":"Mitra",
    "lastName":"s",
    "dob":"2020-04",
    "emailId":"mitra065@gmail",
    "mobile":"123575281",
    "pan":"GBCPS135T"
}
```

```
{
    "errorMessage": "Invalid input. Parameter Name:",
    "errorCode": "ERR_INVALID_INPUT",
    "errorFields": [
        "Mobile",
        "PAN",
        "Date Of Birth"
    ]
}
```

- Success Scenario

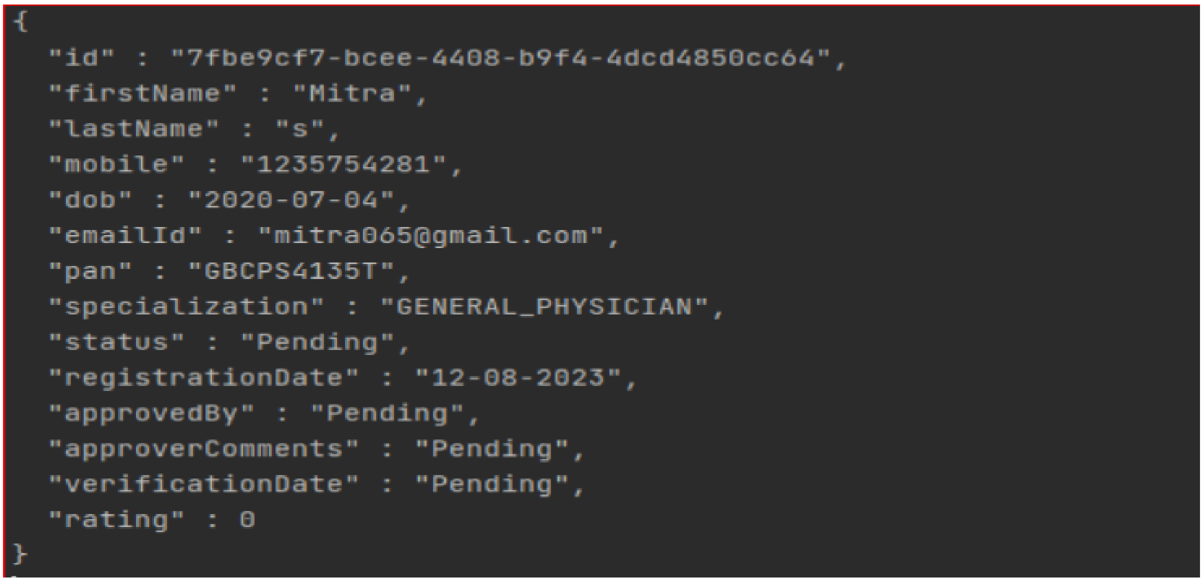


POST http:// 54.197.76.252:8081/doctors

```
{
    "firstName":"Mitra",
    "lastName":"s",
    "dob":"2020-07-04",
    "emailId":"mitra065@gmail.com",
    "mobile":"1235754281",
    "pan":"GBCPS4135T"
}
```

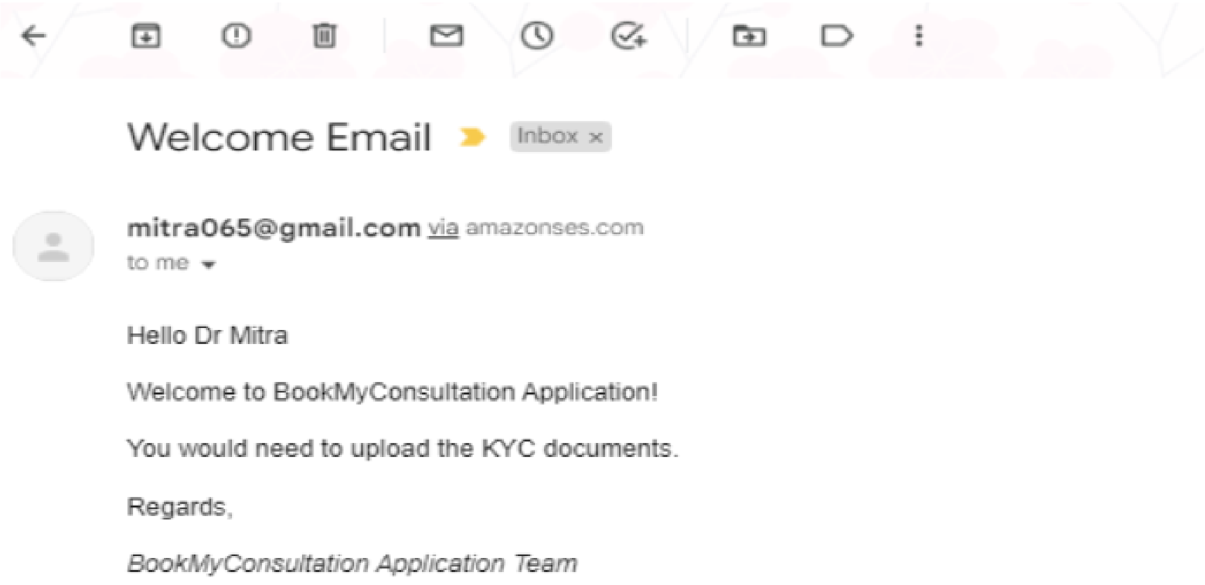
```
{
    "id": "7fbe9cf7-bcee-4408-b9f4-4dcd4850cc64",
    "firstName": "Mitra",
    "lastName": "s",
    "mobile": "1235754281",
    "dob": "2020-07-04",
    "emailId": "mitra065@gmail.com",
    "pan": "GBCPS4135T",
    "specialization": "GENERAL_PHYSICIAN",
    "status": "Pending",
    "registrationDate": "12-08-2023"
}
```

Kafka consumer logs



```
{
  "id" : "7fbe9cf7-bcee-4408-b9f4-4dcd4850cc64",
  "firstName" : "Mitra",
  "lastName" : "s",
  "mobile" : "1235754281",
  "dob" : "2020-07-04",
  "emailId" : "mitra065@gmail.com",
  "pan" : "GBCPS4135T",
  "specialization" : "GENERAL_PHYSICIAN",
  "status" : "Pending",
  "registrationDate" : "12-08-2023",
  "approvedBy" : "Pending",
  "approverComments" : "Pending",
  "verificationDate" : "Pending",
  "rating" : 0
}
```

- Verification mail Sent to Doctor



Welcome Email

mitra065@gmail.com via amazonses.com
to me

Hello Dr Mitra

Welcome to BookMyConsultation Application!

You would need to upload the KYC documents.

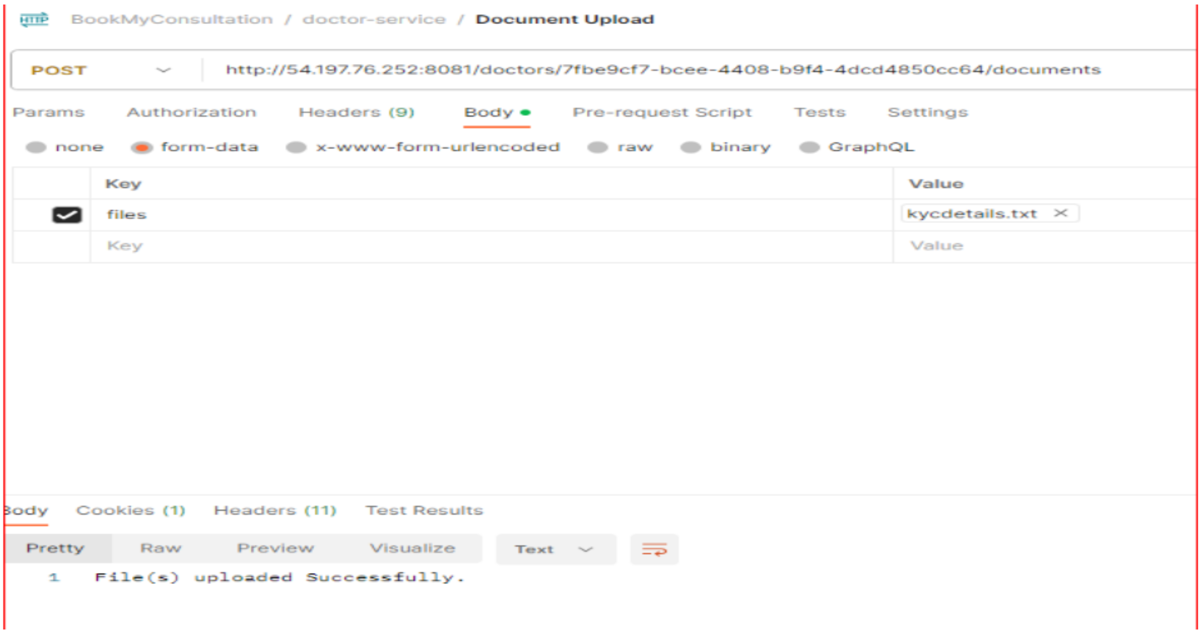
Regards,

BookMyConsultation Application Team

Endpoint 2: URI: `/doctors/{doctorId}/document`

This endpoint is responsible for uploading the documents to an S3 bucket by the doctor.

- Success Scenario – Document Uploaded Successfully



BookMyConsultation / doctor-service / Document Upload

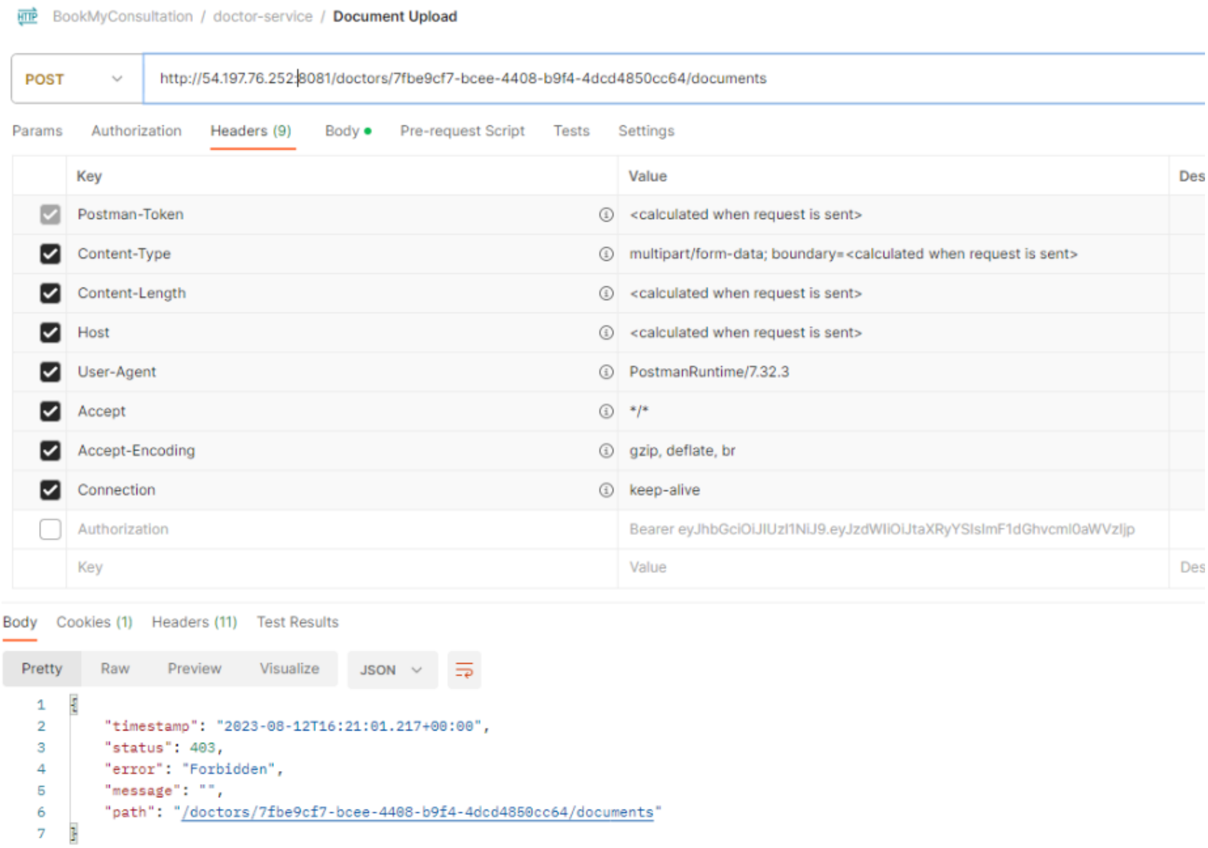
POST http://54.197.76.252:8081/doctors/7fbe9cf7-bcee-4408-b9f4-4dcd4850cc64/documents

Body: form-data

	Key	Value
☑	files	kycdetails.txt
	Key	Value

```
File(s) uploaded Successfully.
```

- Endpoint Forbidden with authorisation header



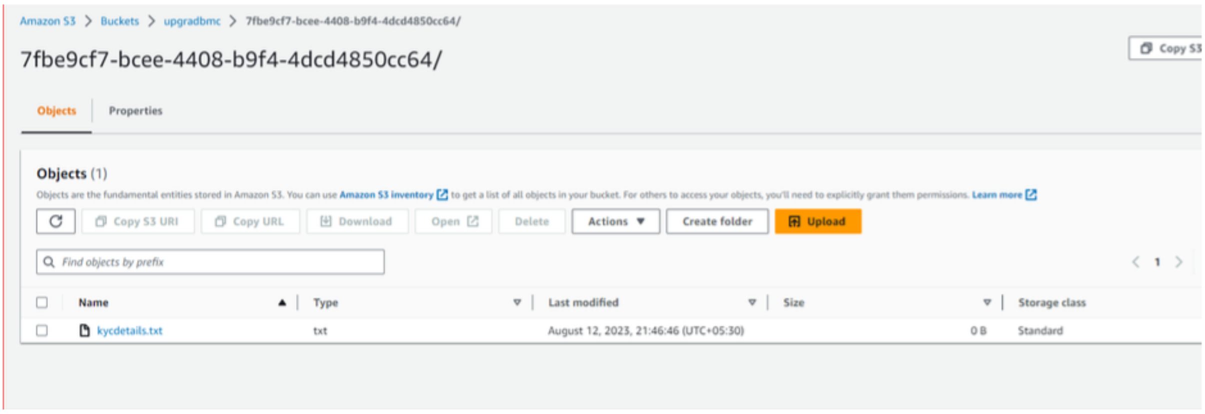
BookMyConsultation / doctor-service / **Document Upload**

POST http://54.197.76.252:8081/doctors/7fbe9cf7-bcee-4408-b9f4-4dcd4850cc64/documents

	Key	Value	Des
☑	Postman-Token	<calculated when request is sent>	
☑	Content-Type	multipart/form-data; boundary=<calculated when request is sent>	
☑	Content-Length	<calculated when request is sent>	
☑	Host	<calculated when request is sent>	
☑	User-Agent	PostmanRuntime/7.32.3	
☑	Accept	*/*	
☑	Accept-Encoding	gzip, deflate, br	
☑	Connection	keep-alive	
□	Authorization	Bearer eyJhbGciOiJIUzI1NiJ9.eyJzdWIiOiJtaXRyYSIsImF1dGhvcml0aWVzIjp	
	Key	Value	Des

```
{
    "timestamp": "2023-08-12T16:21:01.217+00:00",
    "status": 403,
    "error": "Forbidden",
    "message": "",
    "path": "/doctors/7fbe9cf7-bcee-4408-b9f4-4dcd4850cc64/documents"
}
```

- Files Uploaded in S3 bucket



Amazon S3 > Buckets > upgradbmc > 7fbe9cf7-bcee-4408-b9f4-4dcd4850cc64/

7fbe9cf7-bcee-4408-b9f4-4dcd4850cc64/

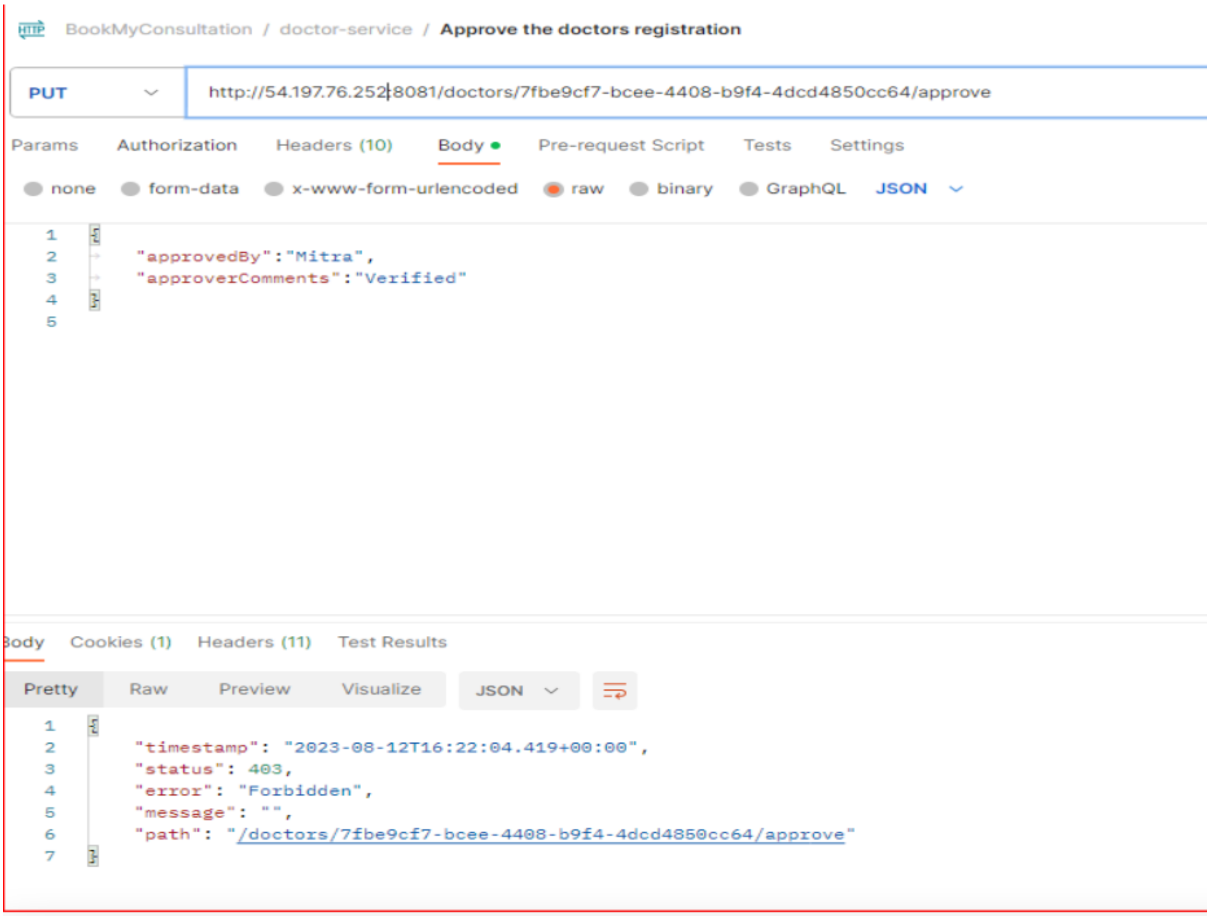
Objects (1)

Objects are the fundamental entities stored in Amazon S3. You can use **Amazon S3 inventory** to get a list of all objects in your bucket. For others to access your objects, you'll need to explicitly grant them permissions. **Learn more**

□	Name	Type	Last modified	Size	Storage class
□	kycdetails.txt	txt	August 12, 2023, 21:46:46 (UTC+05:30)	0 B	Standard

Endpoint 3: URI: `/doctors/{doctorId}/approve`

Endpoint Forbidden for unauthorised users

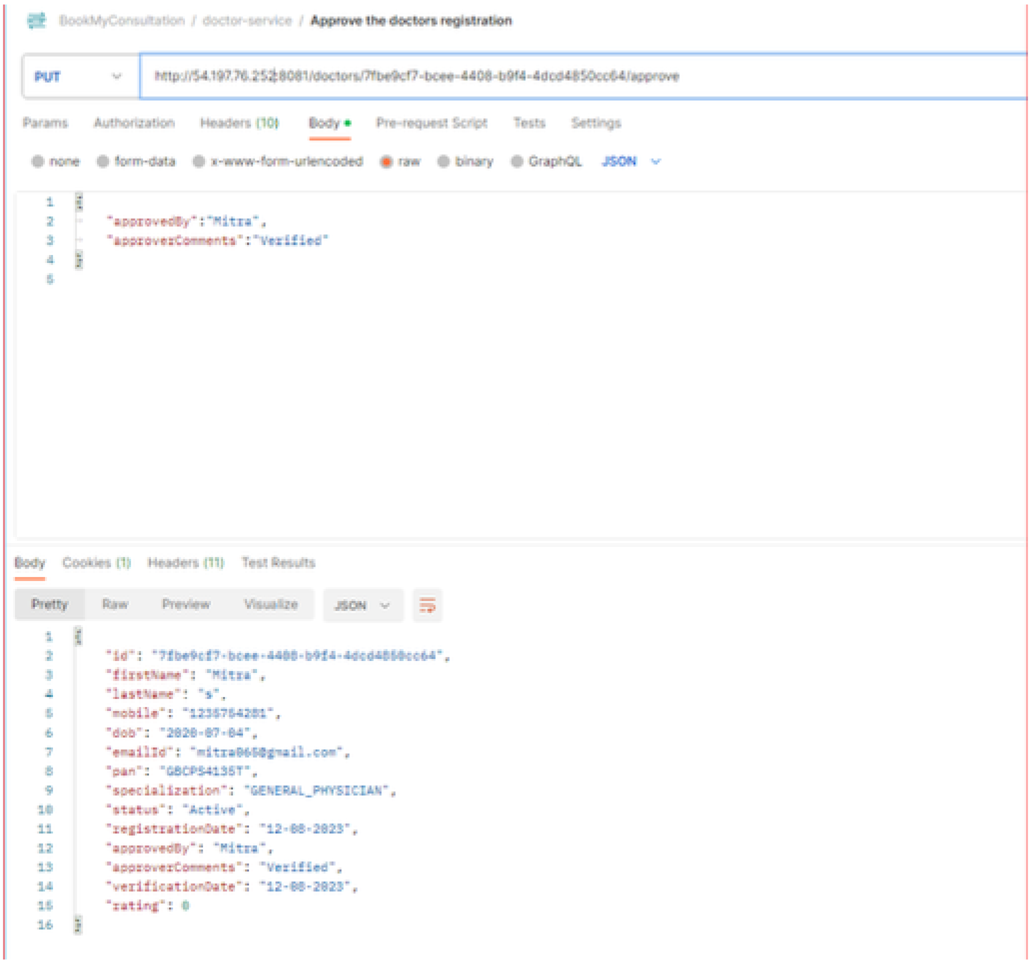


PUT http://54.197.76.252:8081/doctors/7fbe9cf7-bcee-4408-b9f4-4dcd4850cc64/approve

```
{
    "approvedBy":"Mitra",
    "approverComments":"Verified"
}
```

```
{
    "timestamp": "2023-08-12T16:22:04.419+00:00",
    "status": 403,
    "error": "Forbidden",
    "message": "",
    "path": "/doctors/7fbe9cf7-bcee-4408-b9f4-4dcd4850cc64/approve"
}
```

- Successful Scenario – Approve Request

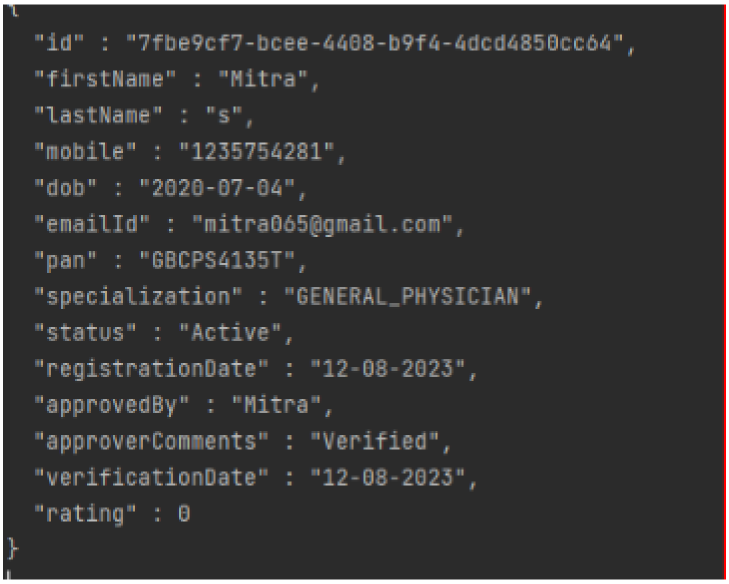


PUT http://54.197.76.252:8081/doctors/7fbe9cf7-bcee-4408-b9f4-4dcd4850cc64/approve

```
{
    "approvedBy":"Mitra",
    "approverComments":"Verified"
}
```

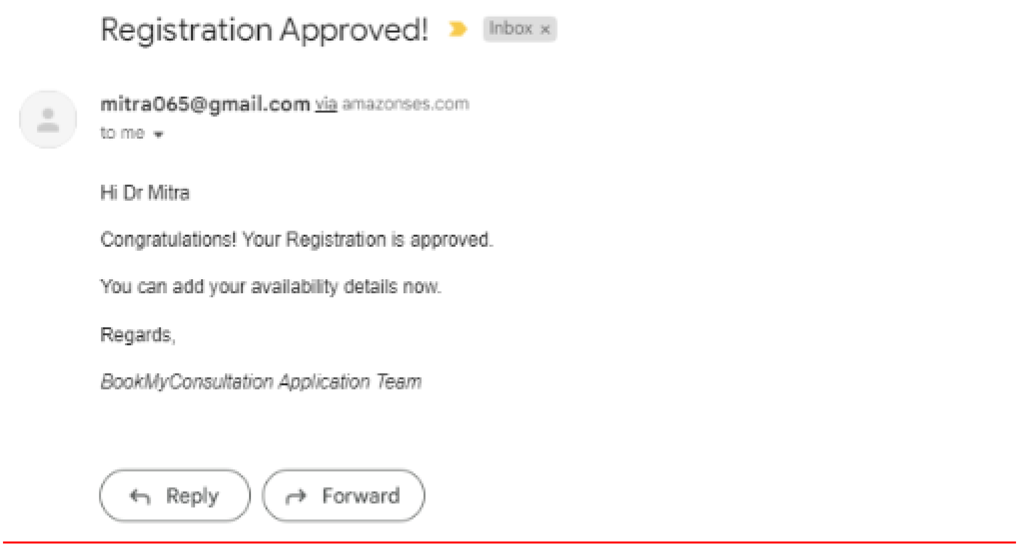
```
{
    "id": "7fbe9cf7-bcee-4408-b9f4-4dcd4850cc64",
    "firstName": "Mitra",
    "lastName": "s",
    "mobile": "1235754281",
    "dob": "2020-07-04",
    "emailId": "mitra065@gmail.com",
    "pan": "GBCPS4135T",
    "specialization": "GENERAL_PHYSICIAN",
    "status": "Active",
    "registrationDate": "12-08-2023",
    "approvedBy": "Mitra",
    "approverComments": "Verified",
    "verificationDate": "12-08-2023",
    "rating": 0
}
```

- Kafka Consumer Logs



```
{
  "id" : "7fbe9cf7-bcee-4408-b9f4-4dcd4850cc64",
  "firstName" : "Mitra",
  "lastName" : "s",
  "mobile" : "1235754281",
  "dob" : "2020-07-04",
  "emailId" : "mitra065@gmail.com",
  "pan" : "GBCPS4135T",
  "specialization" : "GENERAL_PHYSICIAN",
  "status" : "Active",
  "registrationDate" : "12-08-2023",
  "approvedBy" : "Mitra",
  "approverComments" : "Verified",
  "verificationDate" : "12-08-2023",
  "rating" : 0
}
```

- Registration Approved Mail Screenshot



Registration Approved! Inbox ×

mitra065@gmail.com via amazonses.com
to me

Hi Dr Mitra

Congratulations! Your Registration is approved.

You can add your availability details now.

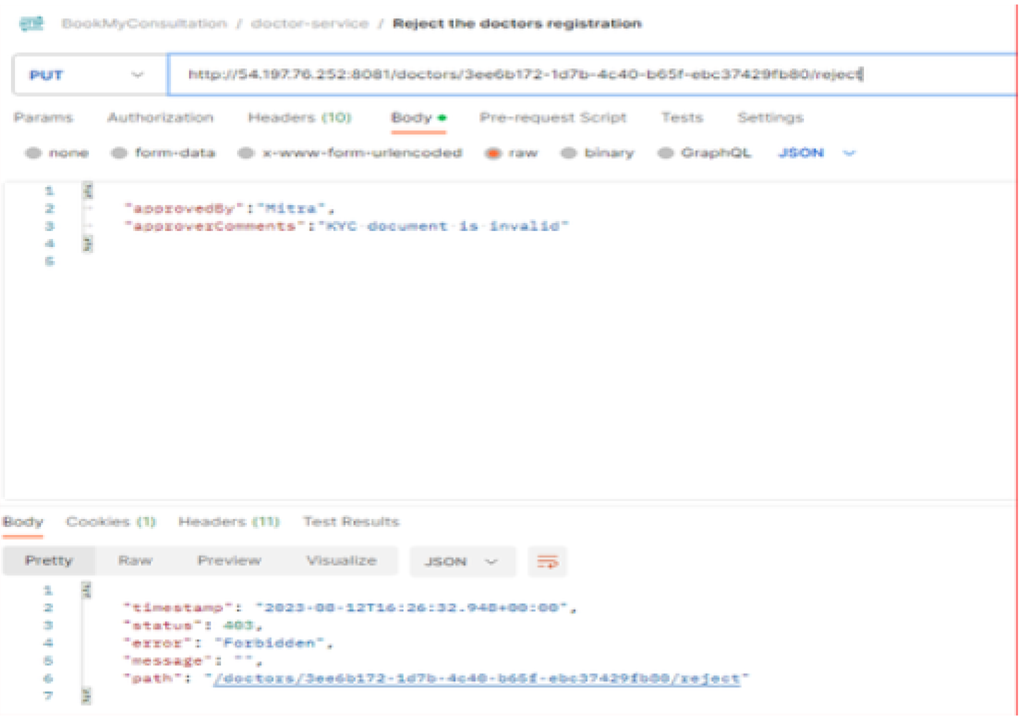
Regards,

BookMyConsultation Application Team

Endpoint 4: URI: `/doctors/{doctorId}/reject`

This endpoint is responsible for rejecting the registration of the doctor

- Endpoint Forbidden for unauthorised users

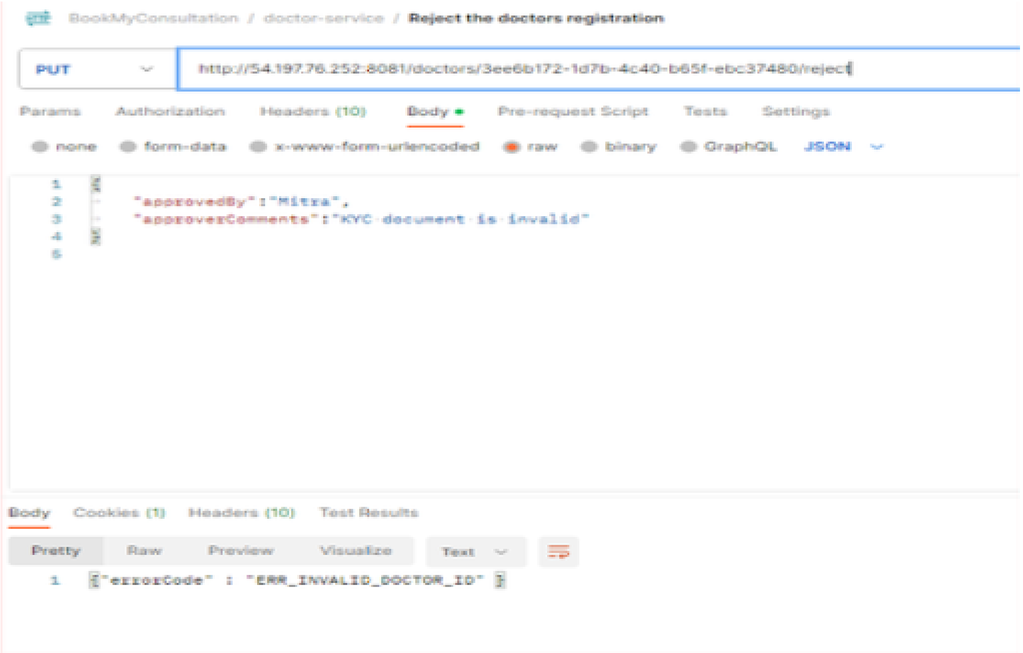


PUT http://54.197.76.252:8081/doctors/3ee6b172-1d7b-4c40-b65f-ebc37429fb80/reject

```
{
    "approvedBy":"Mitra",
    "approverComments":"KYC document is invalid"
}
```

```
{
    "timestamp": "2023-08-12T16:26:32.948+00:00",
    "status": 403,
    "error": "Forbidden",
    "message": "",
    "path": "/doctors/3ee6b172-1d7b-4c40-b65f-ebc37429fb80/reject"
}
```

- Validation Error when Invalid Doctor Id is passed



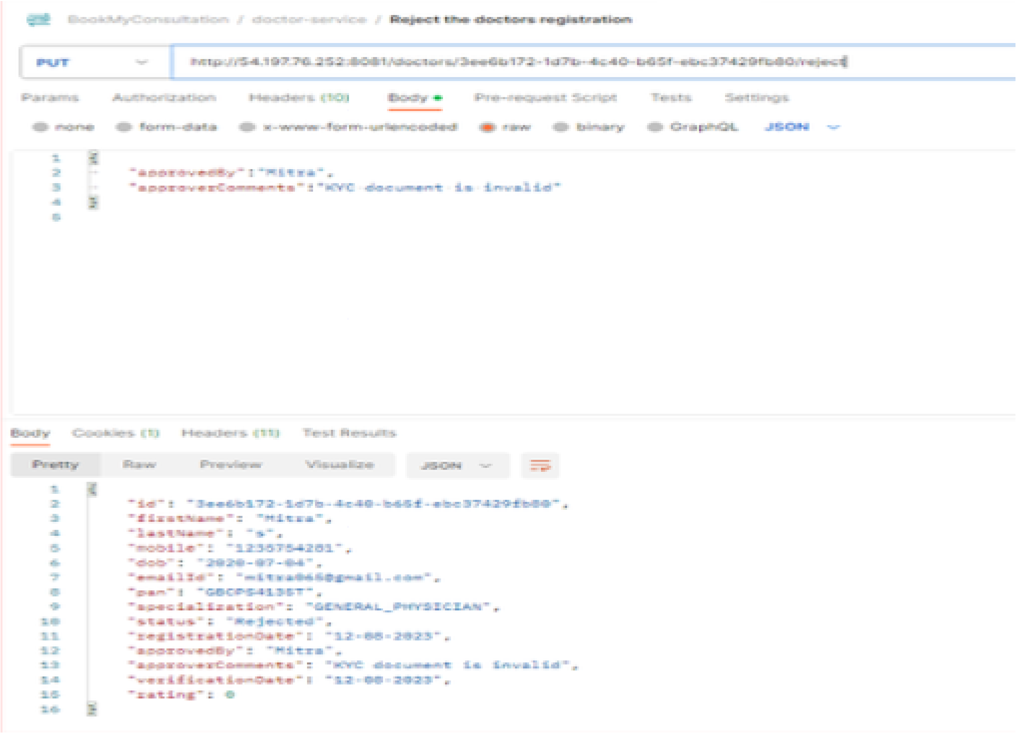
BookMyConsultation / doctor-service / Reject the doctors registration

PUT http://54.197.76.252:8081/doctors/3ee6b172-1d7b-4c40-b65f-ebc37480/reject

```
{
    "approvedBy":"Mitra",
    "approverComments":"KYC document is invalid"
}
```

```
{"errorCode" : "ERR_INVALID_DOCTOR_ID" }
```

- Success Scenario - Rejection



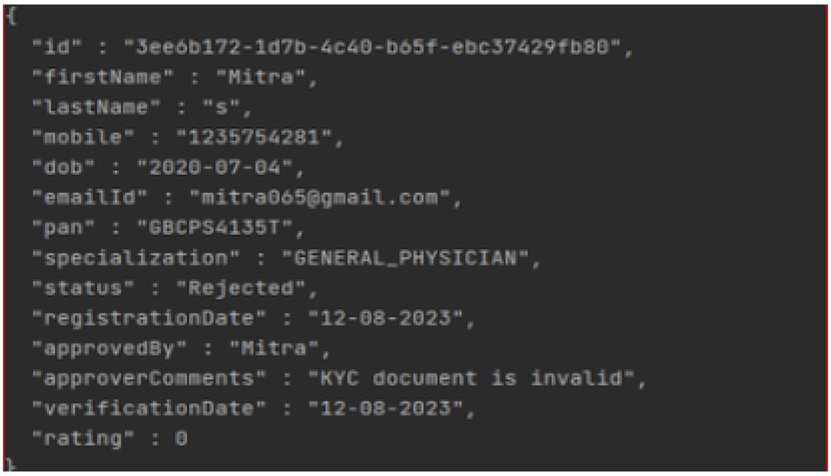
BookMyConsultation / doctor-service / Reject the doctors registration

PUT http://54.197.76.252:8081/doctors/3ee6b172-1d7b-4c40-b65f-ebc37429fb80/reject

```
{
    "approvedBy":"Mitra",
    "approverComments":"KYC document is invalid"
}
```

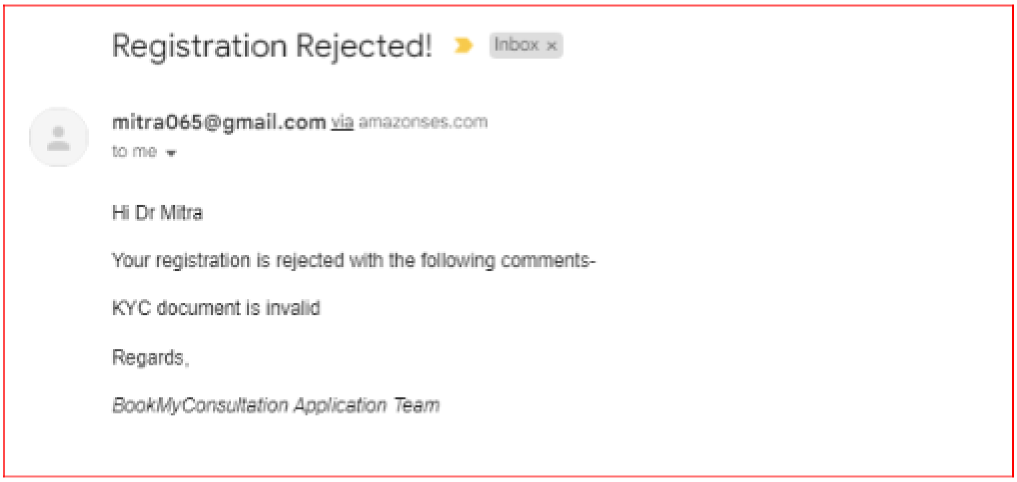
```
{
    "id": "3ee6b172-1d7b-4c40-b65f-ebc37429fb80",
    "firstName": "Mitra",
    "lastName": "s",
    "mobile": "1235754201",
    "dob": "2020-07-04",
    "emailId": "mitra065@gmail.com",
    "pan": "GBCPS4135T",
    "specialization": "GENERAL_PHYSICIAN",
    "status": "Rejected",
    "registrationDate": "12-08-2023",
    "approvedBy": "Mitra",
    "approverComments": "KYC document is invalid",
    "verificationDate": "12-08-2023",
    "rating": 0
}
```

- Kafka Consumer Logs



```
{
  "id" : "3ee6b172-1d7b-4c40-b65f-ebc37429fb80",
  "firstName" : "Mitra",
  "lastName" : "s",
  "mobile" : "1235754281",
  "dob" : "2020-07-04",
  "emailId" : "mitra065@gmail.com",
  "pan" : "GBCPS4135T",
  "specialization" : "GENERAL_PHYSICIAN",
  "status" : "Rejected",
  "registrationDate" : "12-08-2023",
  "approvedBy" : "Mitra",
  "approverComments" : "KYC document is invalid",
  "verificationDate" : "12-08-2023",
  "rating" : 0
}
```

- Registration Rejection Mail Received Screenshot



Registration Rejected!

mitra065@gmail.com via amazonses.com

to me

Hi Dr Mitra

Your registration is rejected with the following comments-

KYC document is invalid

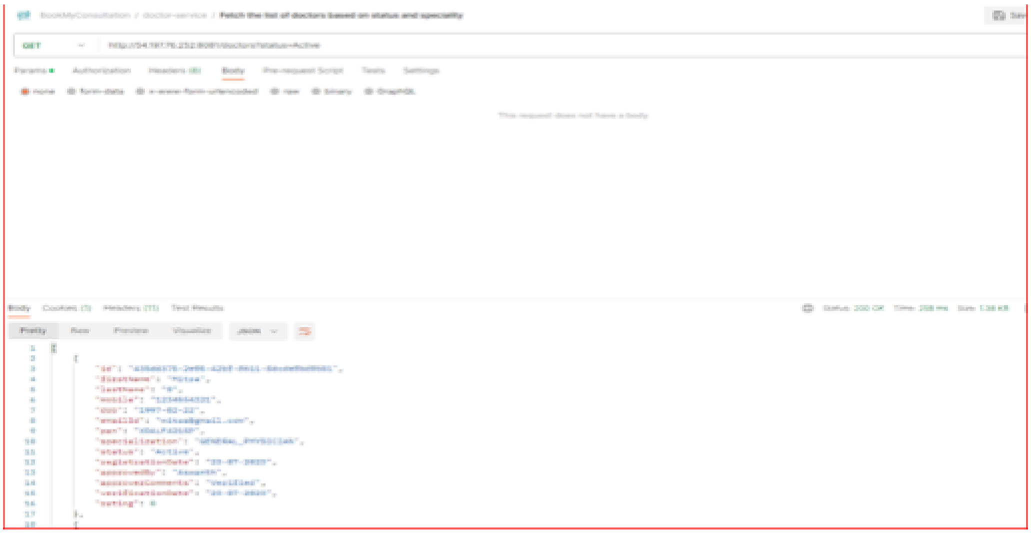
Regards,

BookMyConsultation Application Team

Endpoint 5: URI: `/doctors?<Required Status & Speciality Filter Condition>`

This endpoint is responsible for returning the list of 20 doctors sorted by ratings.

- With Status Filter


BookMyConsultation / doctor-service / Fetch the list of doctors based on status and speciality

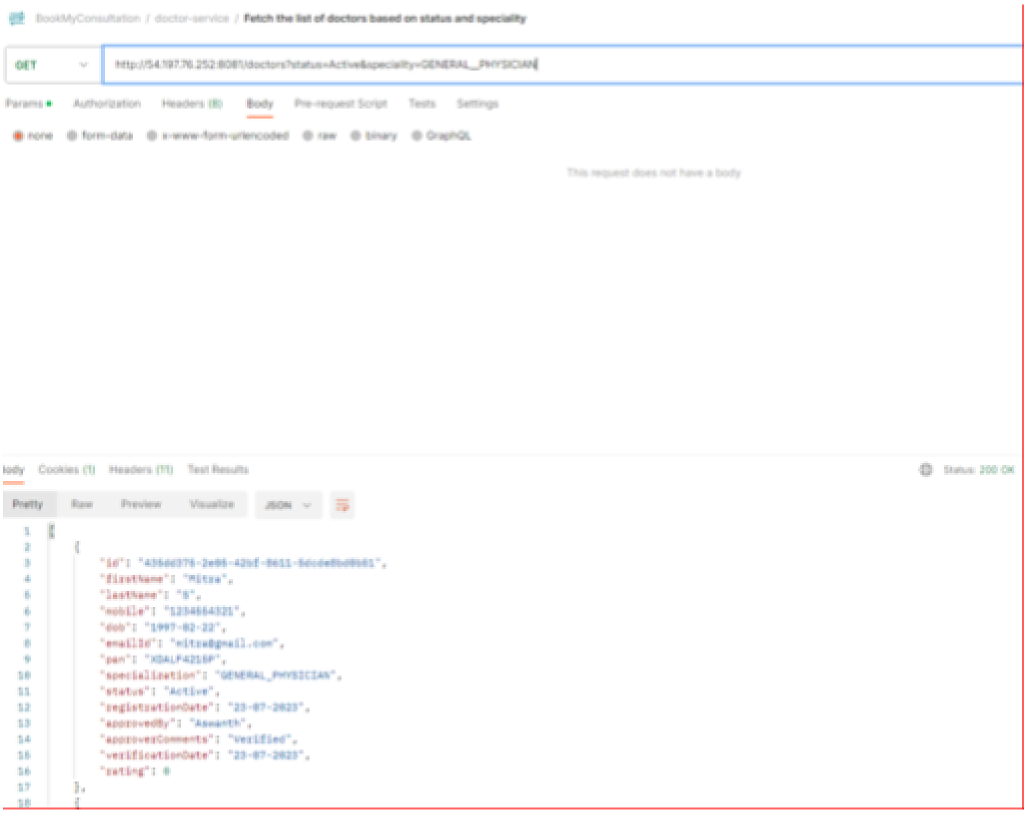
GET http://54.197.76.252:8081/doctors?status=Active

This request does not have a body

Status: 200 OK Time: 258 ms Size: 1.38 KB

```
[
    {
        "id": "435dd376-2e86-42bf-8611-5dcde8bd8b61",
        "firstName": "Mitra",
        "lastName": "S",
        "mobile": "1234554321",
        "dob": "1997-02-22",
        "emailId": "mitra@gmail.com",
        "pan": "XDALF4216P",
        "specialization": "GENERAL_PHYSICIAN",
        "status": "Active",
        "registrationDate": "23-07-2023",
        "approvedBy": "Aswanth",
        "approverComments": "Verified",
        "verificationDate": "23-07-2023",
        "rating": 0
    },
    {
```

- With Status & Specialty Filter


BookMyConsultation / doctor-service / Fetch the list of doctors based on status and speciality

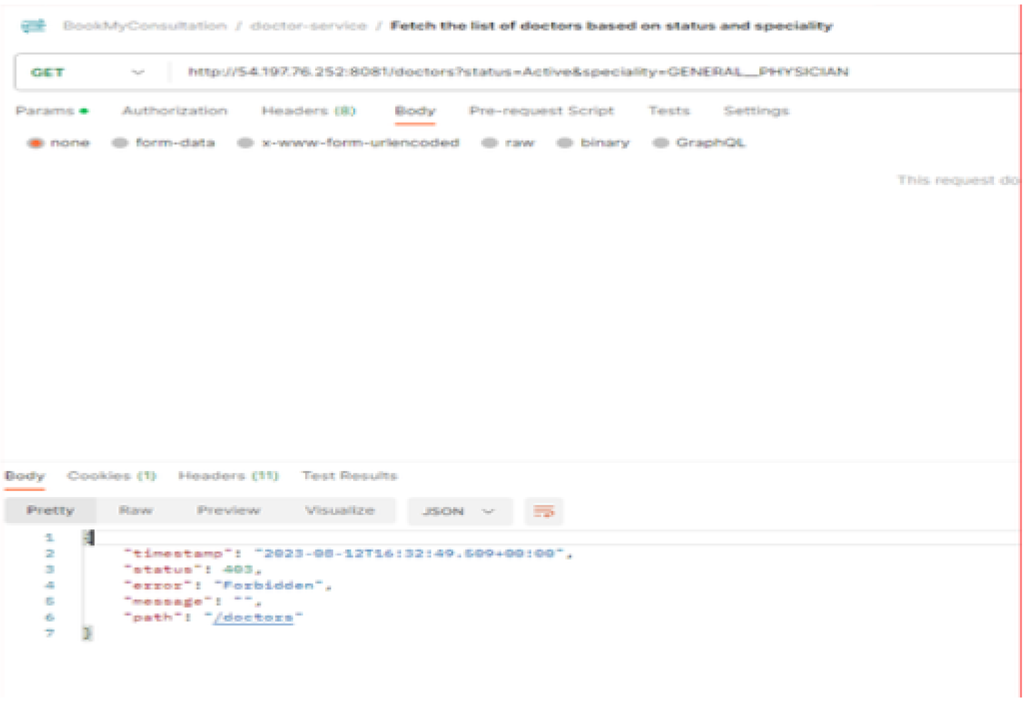
GET http://54.197.76.252:8081/doctors?status=Active&speciality=GENERAL_PHYSICIAN

This request does not have a body

Status: 200 OK

```
[
    {
        "id": "435dd376-2e86-42bf-8611-5dcde8bd8b61",
        "firstName": "Mitra",
        "lastName": "S",
        "mobile": "1234554321",
        "dob": "1997-02-22",
        "emailId": "mitra@gmail.com",
        "pan": "XDALF4216P",
        "specialization": "GENERAL_PHYSICIAN",
        "status": "Active",
        "registrationDate": "23-07-2023",
        "approvedBy": "Aswanth",
        "approverComments": "Verified",
        "verificationDate": "23-07-2023",
        "rating": 0
    },
    {
```

- Forbidden Error for Unauthorised error



BookMyConsultation / doctor-service / Fetch the list of doctors based on status and speciality

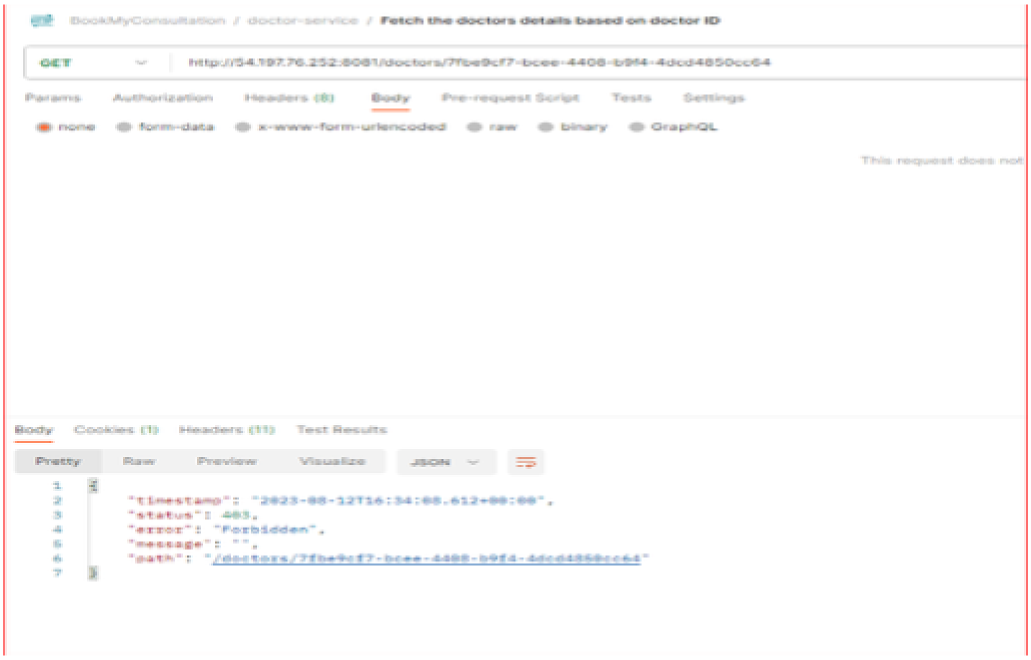
GET http://54.197.76.252:8081/doctors?status=Active&speciality=GENERAL__PHYSICIAN

```
{
    "timestamp": "2023-08-12T16:32:49.509+00:00",
    "status": 403,
    "error": "Forbidden",
    "message": "",
    "path": "/doctors"
}
```

Endpoint 6: URI: `/doctors/{doctorId}`

This endpoint is responsible for returning the details of the doctor based on the doctor ID.

- Forbidden Error for Unauthorised error

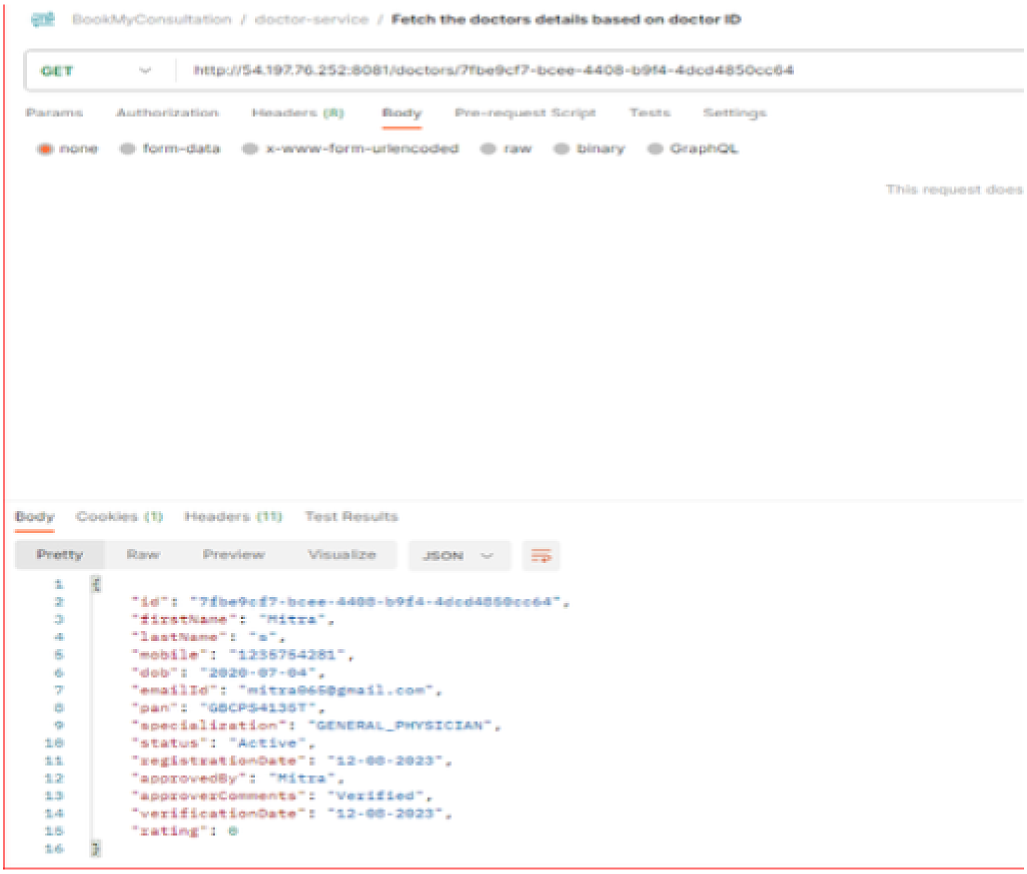


BookMyConsultation / doctor-service / Fetch the doctors details based on doctor ID

GET http://54.197.76.252:8081/doctors/7fbe9cf7-bcee-4408-b9f4-4dcd4850cc64

```
{
    "timestamp": "2023-08-12T16:34:08.612+00:00",
    "status": 403,
    "error": "Forbidden",
    "message": "",
    "path": "/doctors/7fbe9cf7-bcee-4408-b9f4-4dcd4850cc64"
}
```

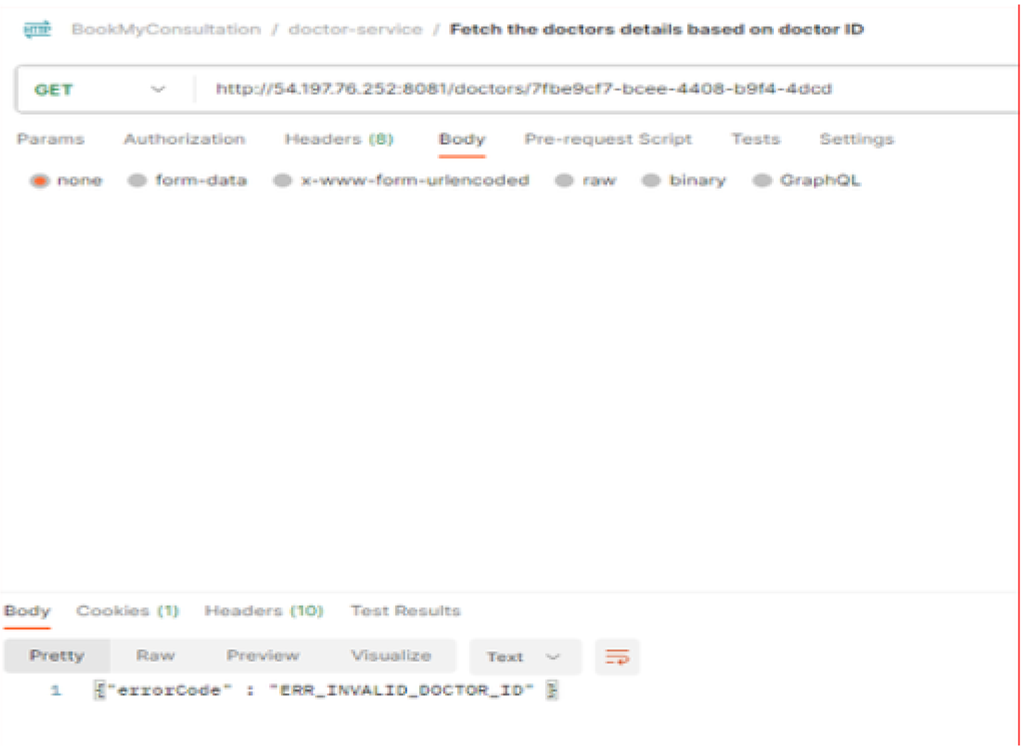
- Success Scenario



http://54.197.76.252:8081/doctors/7fbe9cf7-bcee-4408-b9f4-4dcd4850cc64

```
{
    "id": "7fbe9cf7-bcee-4408-b9f4-4dcd4850cc64",
    "firstName": "Mitra",
    "lastName": "s",
    "mobile": "1235764281",
    "dob": "2020-07-04",
    "emailId": "mitra065@gmail.com",
    "pan": "GBCPS4135T",
    "specialization": "GENERAL_PHYSICIAN",
    "status": "Active",
    "registrationDate": "12-08-2023",
    "approvedBy": "Mitra",
    "approverComments": "Verified",
    "verificationDate": "12-08-2023",
    "rating": 0
}
```

- Validation Error – When Invalid Doctor Id passed



http://54.197.76.252:8081/doctors/7fbe9cf7-bcee-4408-b9f4-4dcd

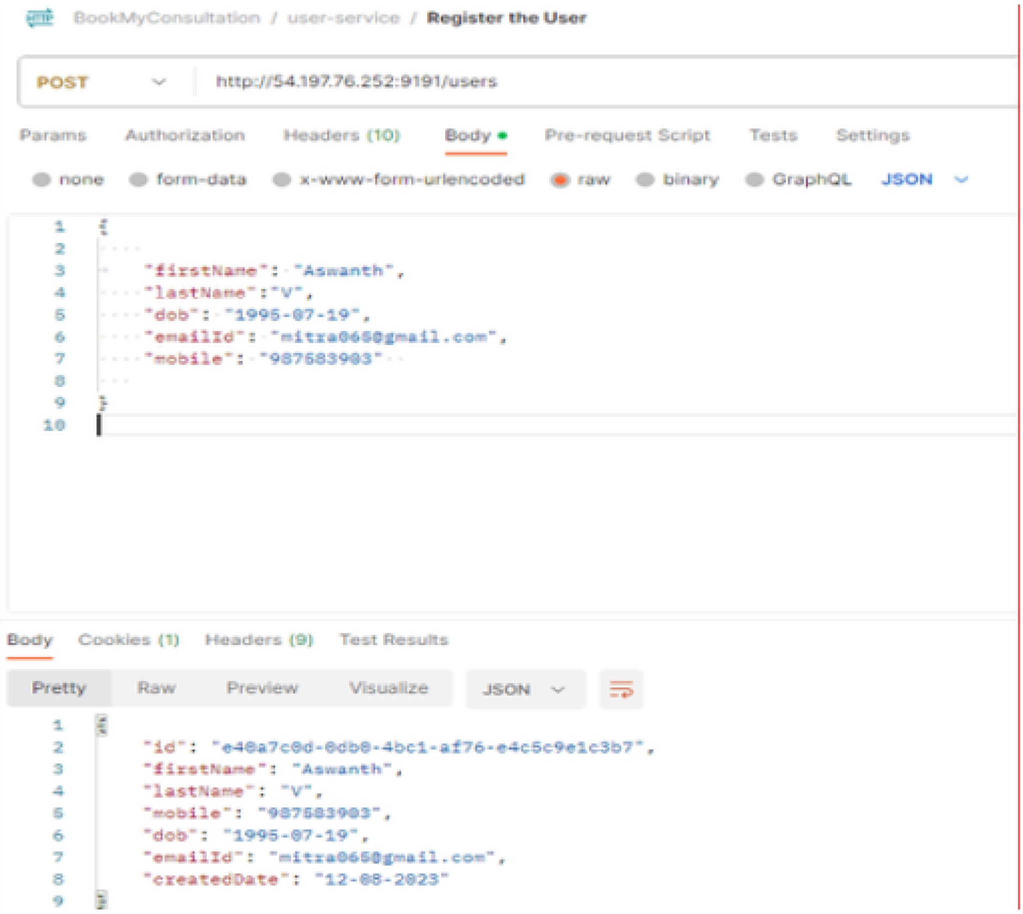
```
{"errorCode" : "ERR_INVALID_DOCTOR_ID" }
```

User -Onboarding Service

Endpoint 1: URI: `/users`

This endpoint is responsible for collecting information about the user.

- Successful Scenario

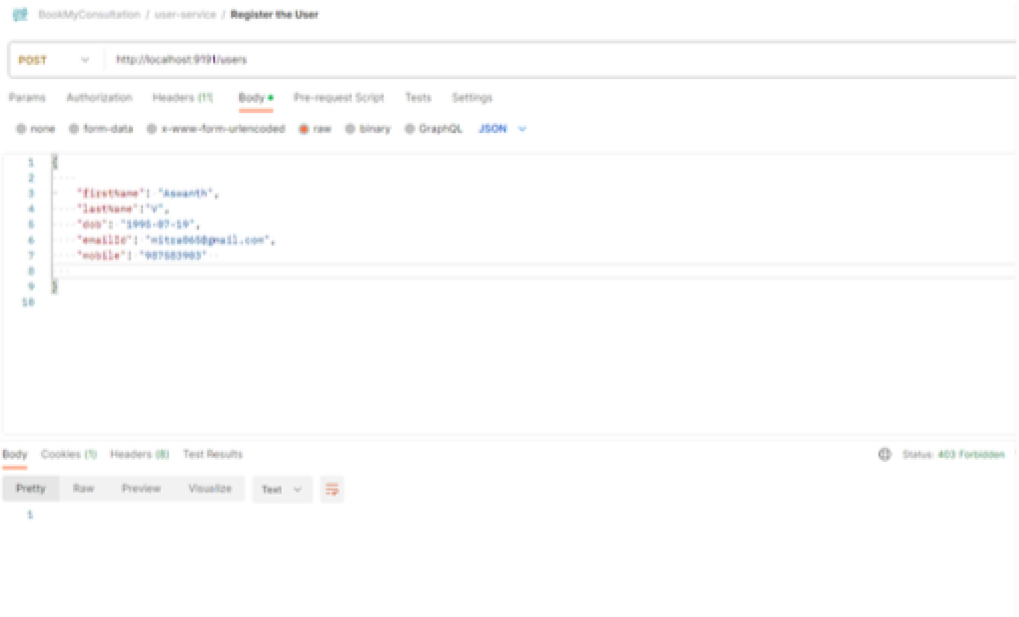


http://54.197.76.252:9191/users

```
{

    "firstName": "Aswanth",
    "lastName":"V",
    "dob": "1995-07-19",
    "emailId": "mitra065@gmail.com",
    "mobile": "987583903"

}

```

```
{
    "id": "e40a7c0d-0db0-4bc1-af76-e4c5c9e1c3b7",
    "firstName": "Aswanth",
    "lastName": "V",
    "mobile": "987583903",
    "dob": "1995-07-19",
    "emailId": "mitra065@gmail.com",
    "createdDate": "12-08-2023"
}
```

- Forbidden Error for unauthorised error

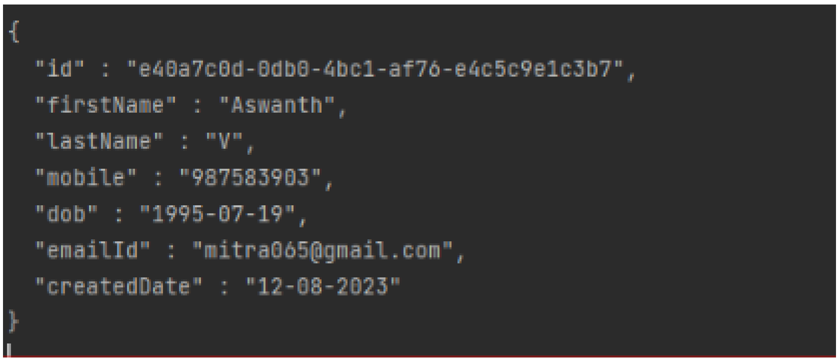


http://localhost:9191/users

```
{

    "firstName": "Aswanth",
    "lastName":"V",
    "dob": "1995-07-19",
    "emailId": "mitra065@gmail.com",
    "mobile": "987583903"

}

```

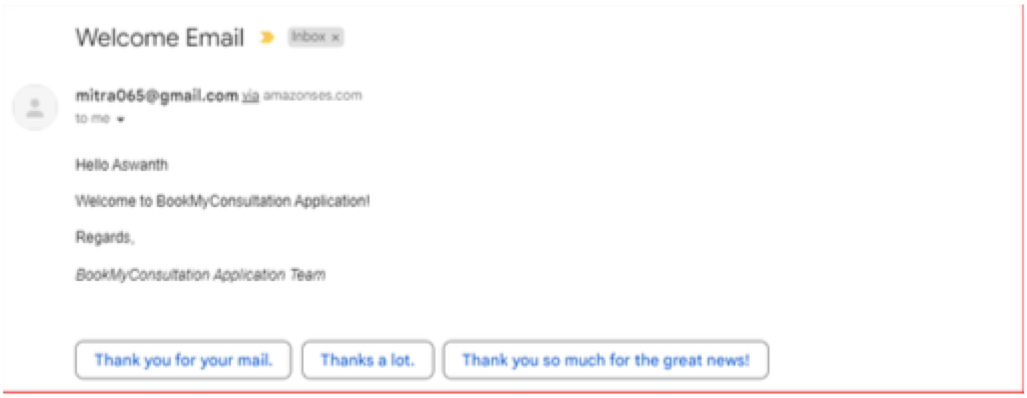
Status: 403 Forbidden

- Kafka Consumer logs



```
{
  "id" : "e40a7c0d-0db0-4bc1-af76-e4c5c9e1c3b7",
  "firstName" : "Aswanth",
  "lastName" : "V",
  "mobile" : "987583903",
  "dob" : "1995-07-19",
  "emailId" : "mitra065@gmail.com",
  "createdDate" : "12-08-2023"
}
```

- Verification mail received after user registration



Welcome Email

mitra065@gmail.com via amazonses.com
to me

Hello Aswanth

Welcome to BookMyConsultation Application!

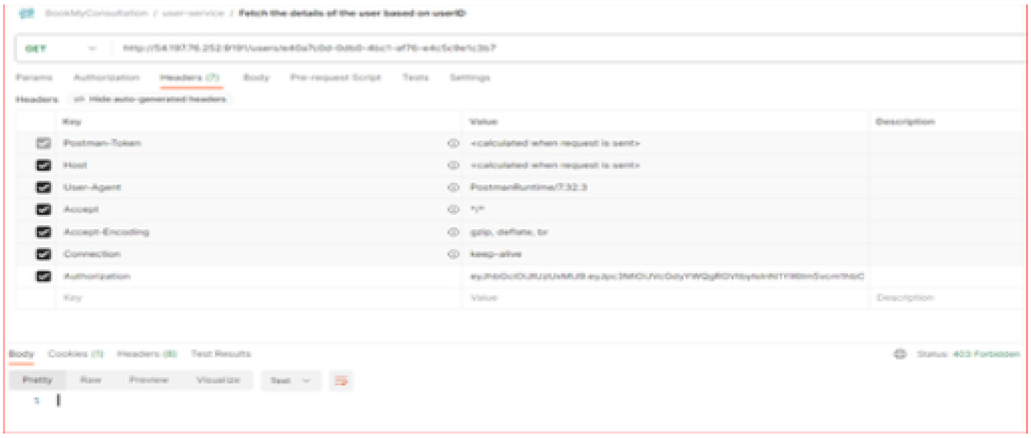
Regards,

BookMyConsultation Application Team

Endpoint 2: URI: `/users/{userID}`

This endpoint is responsible for collecting information about the user.

- Forbidden Error for unauthorised error



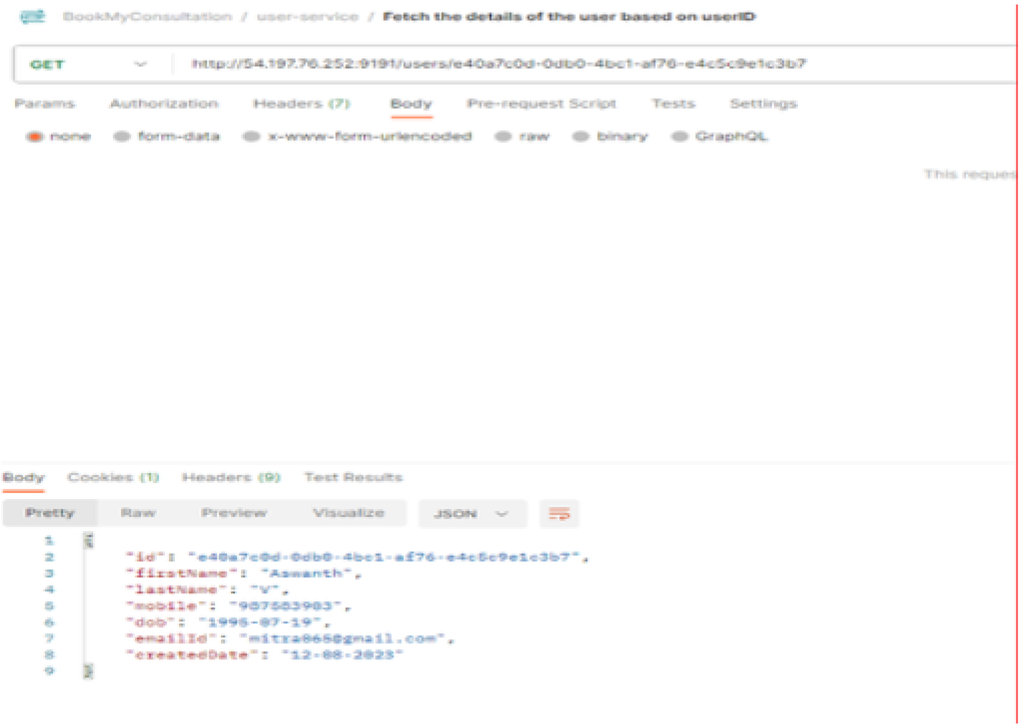
BookMyConsultation / user-service / **Fetch the details of the user based on userID**

GET http://54.197.76.252:8191/users/e40a7c0d-0db0-4bc1-af76-e4c5c9e1c3b7

	Key	Value	Description
☑	Postman-Token	<calculated when request is sent>	
☑	Host	<calculated when request is sent>	
☑	User-Agent	PostmanRuntime/7.32.3	
☑	Accept	*/*	
☑	Accept-Encoding	gzip, deflate, br	
☑	Connection	keep-alive	
☑	Authorization	eyJhbGciOiJIUzUxMiJ9.eyJ[illegible]	
	Key	Value	Description

Status: 403 Forbidden

- Successful Scenario

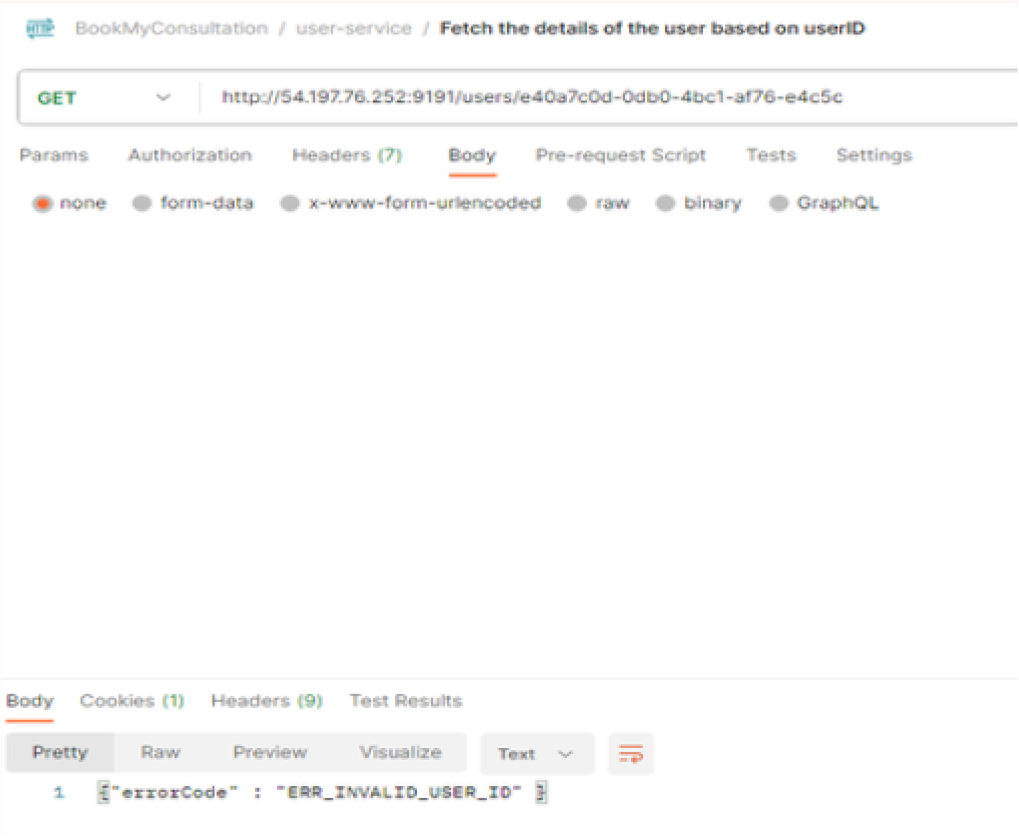


BookMyConsultation / user-service / **Fetch the details of the user based on userID**

GET http://54.197.76.252:9191/users/e40a7c0d-0db0-4bc1-af76-e4c5c9e1c3b7

```
{
    "id": "e40a7c0d-0db0-4bc1-af76-e4c5c9e1c3b7",
    "firstName": "Aswanth",
    "lastName": "V",
    "mobile": "9875853983",
    "dob": "1996-07-19",
    "emailId": "mitra865@gmail.com",
    "createdDate": "12-08-2023"
}
```

- Validation Error - Invalid User Id Passed


BookMyConsultation / user-service / Fetch the details of the user based on userID

GET http://54.197.76.252:9191/users/e40a7c0d-0db0-4bc1-af76-e4c5c

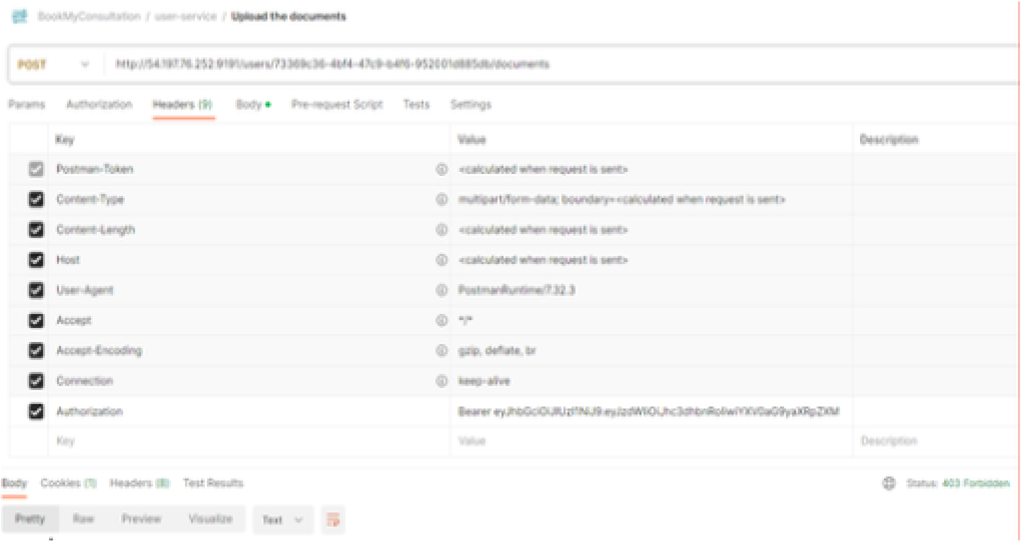
Params
Authorization
Headers (7)
Body
Pre-request Script
Tests
Settings
● none ○ form-data ○ x-www-form-urlencoded ○ raw ○ binary ○ GraphQL
Body
Cookies (1)
Headers (9)
Test Results
Pretty
Raw
Preview
Visualize
Text

```
{"errorCode" : "ERR_INVALID_USER_ID" }
```

Endpoint 3: URI: `/users/{id}/documents.`

A POST request will be sent to this endpoint to upload the files.

- Forbidden Error for unauthorised error

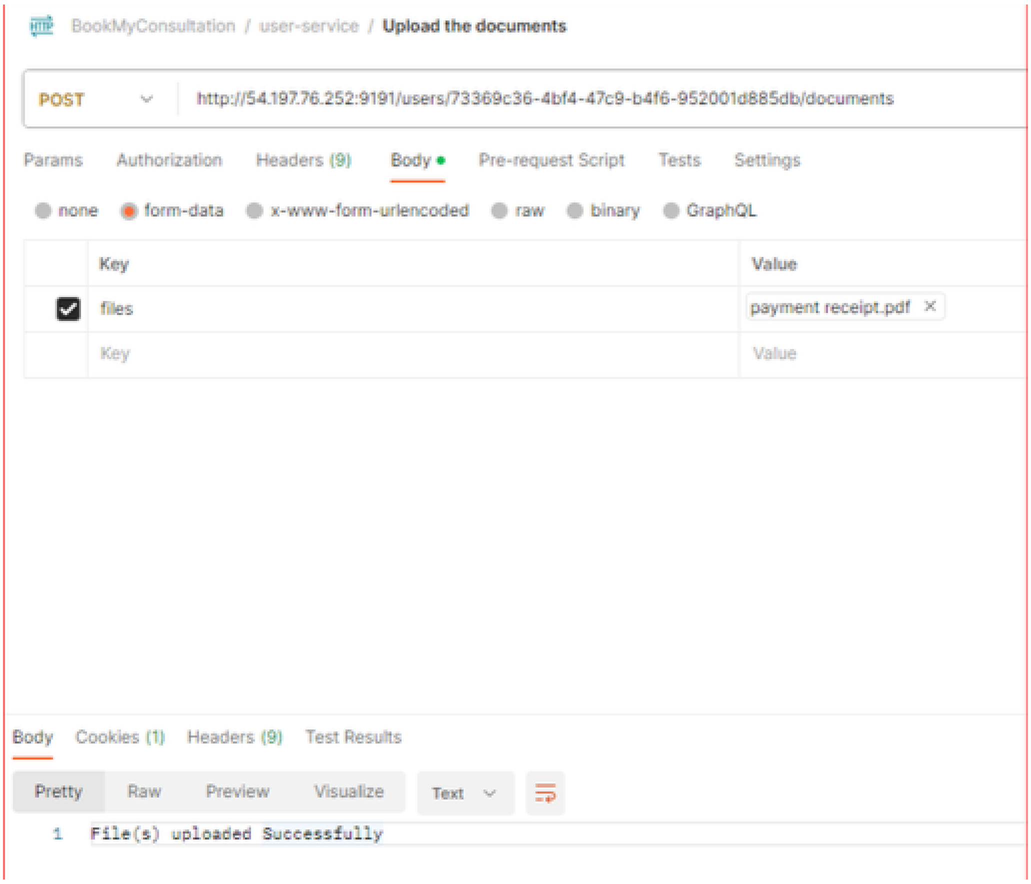


POST http://54.197.76.252:9191/users/73369c36-4bf4-47c9-b4f6-952001d885db/documents

	Key	Value	Description
☑	Postman-Token	<calculated when request is sent>	
☑	Content-Type	multipart/form-data; boundary=<calculated when request is sent>	
☑	Content-Length	<calculated when request is sent>	
☑	Host	<calculated when request is sent>	
☑	User-Agent	PostmanRuntime/7.32.3	
☑	Accept	*/*	
☑	Accept-Encoding	gzip, deflate, br	
☑	Connection	keep-alive	
☑	Authorization	Bearer eyJhbGciOiJIUzI1NiJ9.eyJzdWIiOiJhc3dhbnRoIiwiYXV0aG9yaXRpZXM	
	Key	Value	Description

Status: 403 Forbidden

- File Uploaded Successfully



BookMyConsultation / user-service / Upload the documents

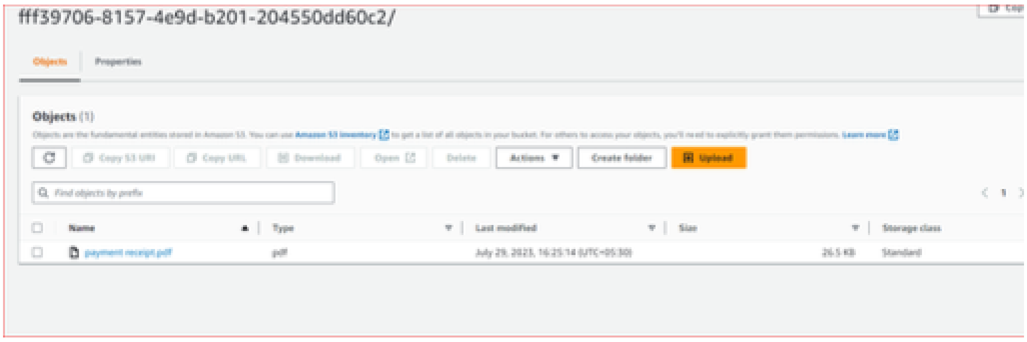
POST http://54.197.76.252:9191/users/73369c36-4bf4-47c9-b4f6-952001d885db/documents

○ none ● form-data ○ x-www-form-urlencoded ○ raw ○ binary ○ GraphQL

	Key	Value
☑	files	payment receipt.pdf
	Key	Value

```
File(s) uploaded Successfully
```

- File Uploaded in S3 bucket



fff39706-8157-4e9d-b201-204550dd60c2/

Objects (1)

Objects are the fundamental entities stored in Amazon S3. You can use Amazon S3 inventory to get a list of all objects in your bucket. For others to access your objects, you'll need to explicitly grant them permissions. Learn more

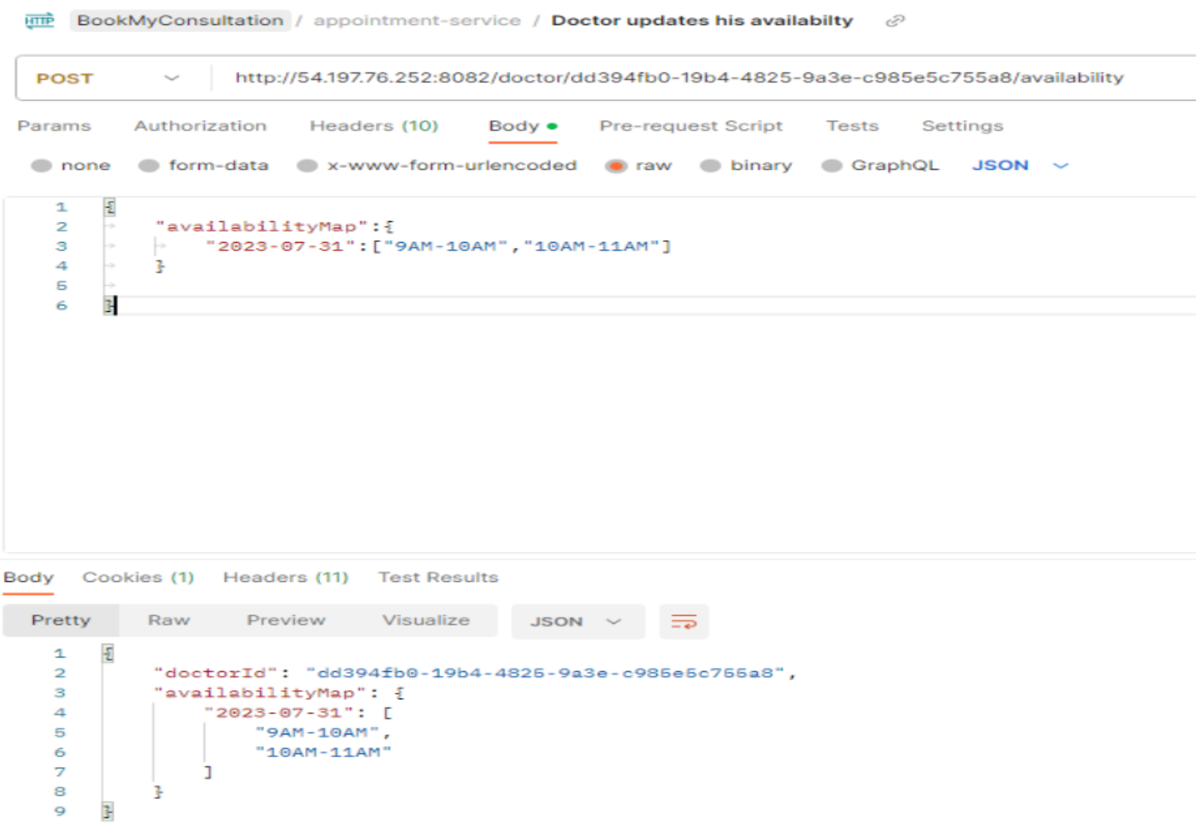
□	Name	Type	Last modified	Size	Storage class
□	payment receipt.pdf	pdf	July 29, 2023, 16:25:14 (UTC+05:30)	26.5 KB	Standard

Appointment Service

Endpoint 1: POST URI: `/doctor/{doctorId}/availability`

This endpoint is responsible for updating the availability of the doctors.

- Successful Scenario



BookMyConsultation / appointment-service / Doctor updates his availabilty

POST http://54.197.76.252:8082/doctor/dd394fb0-19b4-4825-9a3e-c985e5c755a8/availability

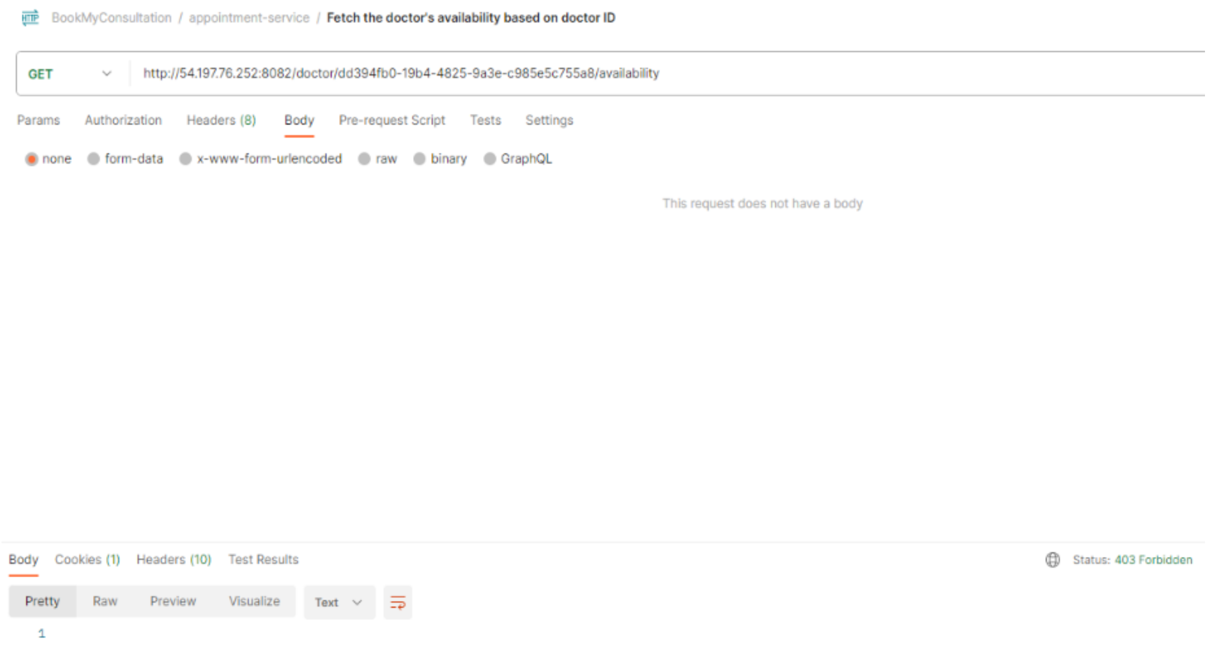
```
{
    "availabilityMap":{
        "2023-07-31":["9AM-10AM","10AM-11AM"]
    }

}
```

```
{
    "doctorId": "dd394fb0-19b4-4825-9a3e-c985e5c755a8",
    "availabilityMap": {
        "2023-07-31": [
            "9AM-10AM",
            "10AM-11AM"
        ]
    }
}
```

Endpoint 2: GET URI: `/doctor/{doctorId}/availability`

This endpoint is responsible for returning the availability of the doctors.

- Forbidden Error for Unauthorised Users



BookMyConsultation / appointment-service / **Fetch the doctor's availability based on doctor ID**

GET http://54.197.76.252:8082/doctor/dd394fb0-19b4-4825-9a3e-c985e5c755a8/availability

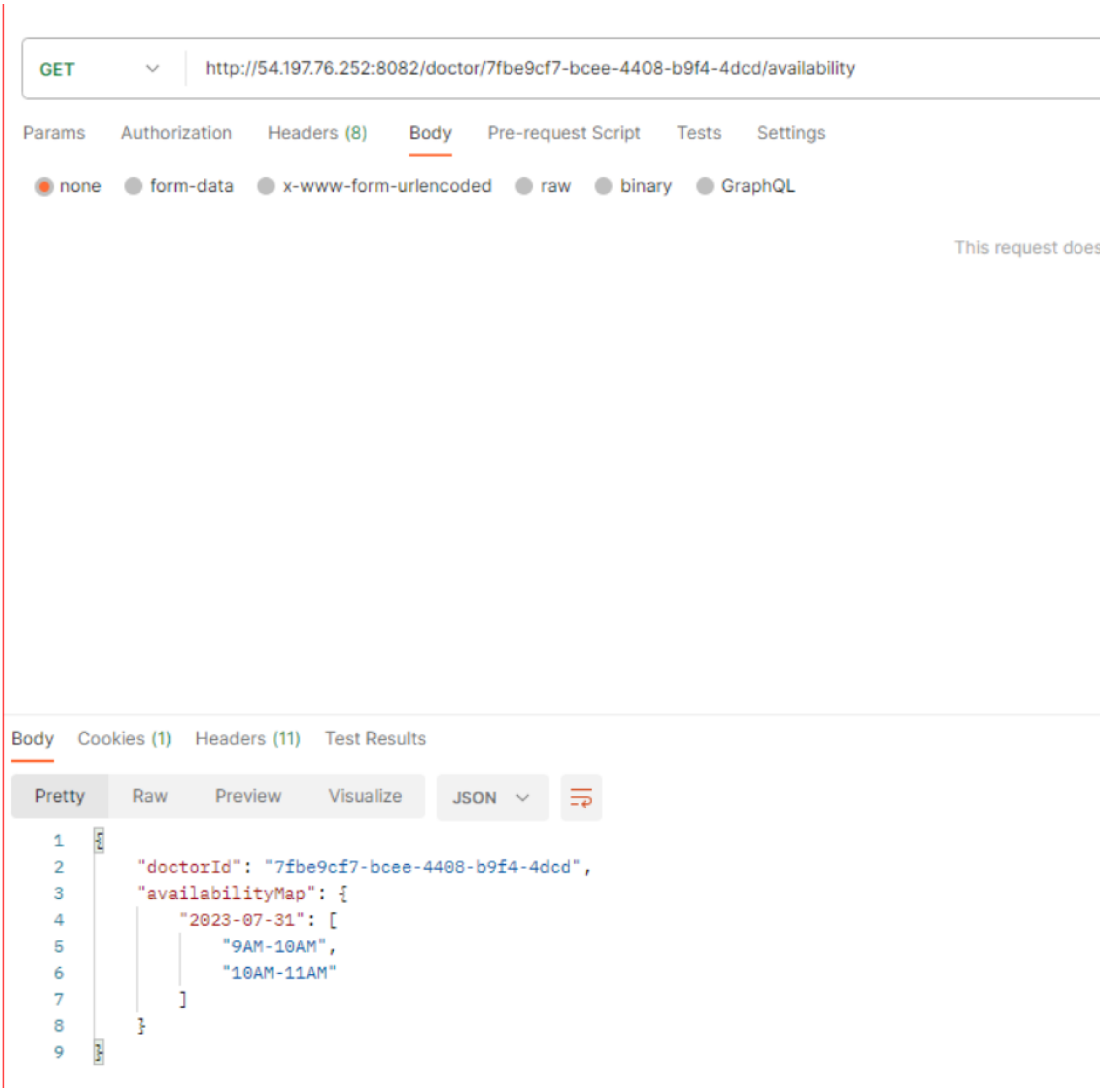
● none ○ form-data ○ x-www-form-urlencoded ○ raw ○ binary ○ GraphQL

This request does not have a body

Status: 403 Forbidden

```
1
```

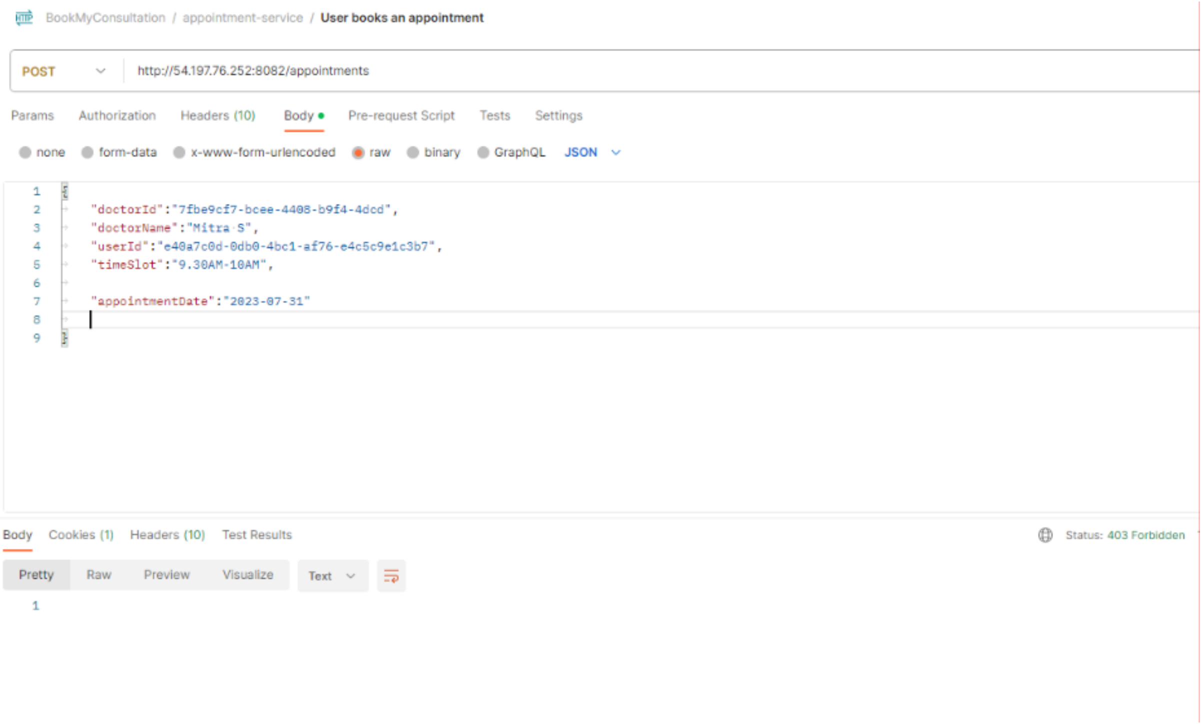
- Successful Scenario



```
{
    "doctorId": "7fbe9cf7-bcee-4408-b9f4-4dcd",
    "availabilityMap": {
        "2023-07-31": [
            "9AM-10AM",
            "10AM-11AM"
        ]
    }
}
```

Endpoint 3: URI: `/appointments`
This endpoint is responsible for booking an appointment.

- Forbidden Error for Unauthorised Users



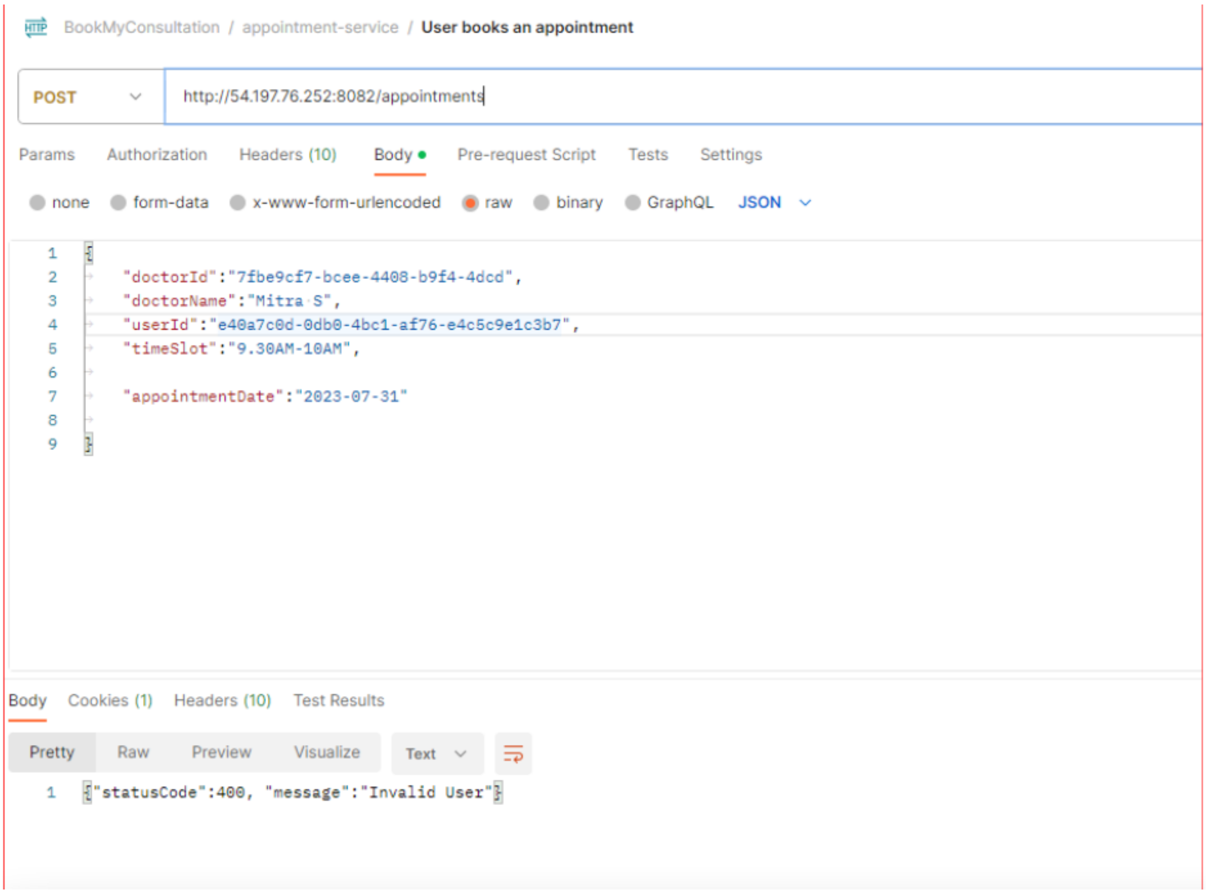
POST http://54.197.76.252:8082/appointments

Body: raw JSON

```
{
    "doctorId":"7fbe9cf7-bcee-4408-b9f4-4dcd",
    "doctorName":"Mitra S",
    "userId":"e40a7c0d-0db0-4bc1-af76-e4c5c9e1c3b7",
    "timeSlot":"9.30AM-10AM",

    "appointmentDate":"2023-07-31"

}
```

Status: 403 Forbidden

- Validation Error When Invalid User Id is Passed



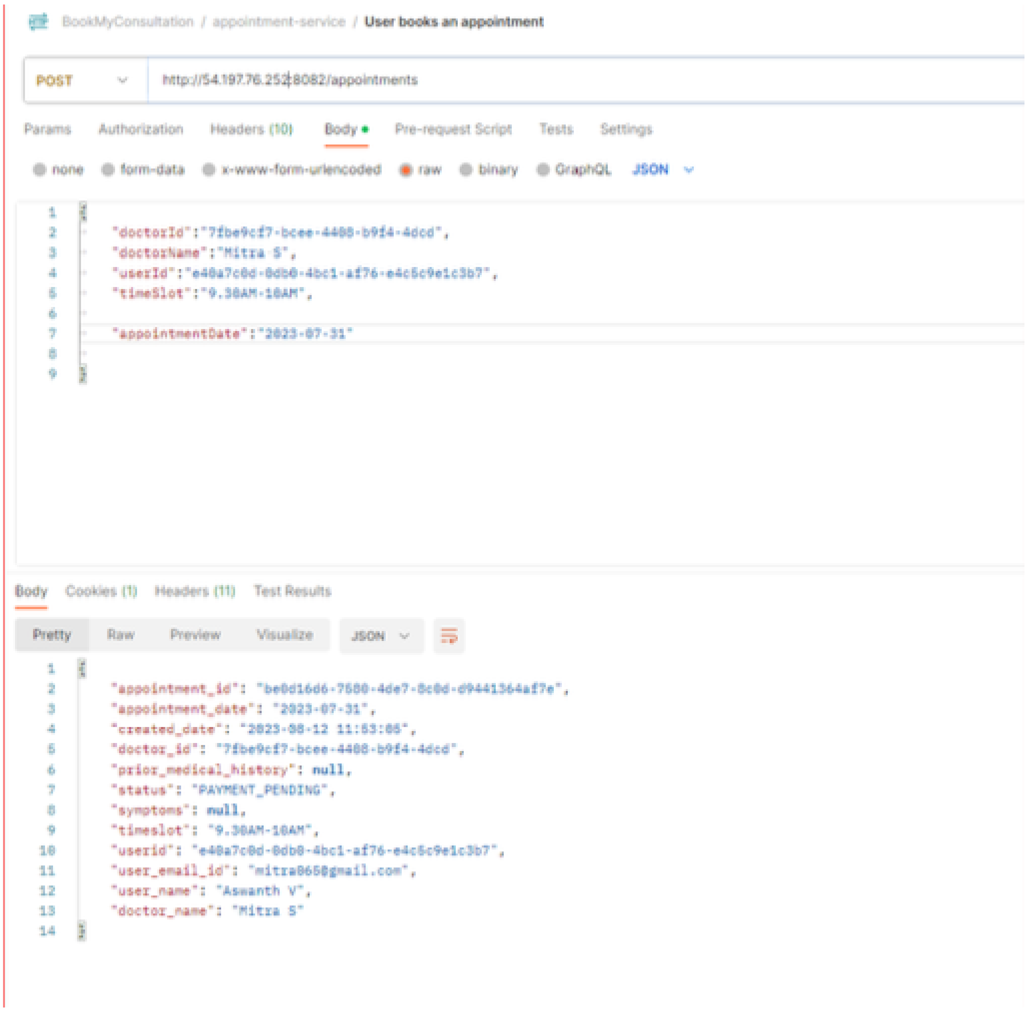
POST http://54.197.76.252:8082/appointments

Body: raw JSON

```
{
    "doctorId":"7fbe9cf7-bcee-4408-b9f4-4dcd",
    "doctorName":"Mitra S",
    "userId":"e40a7c0d-0db0-4bc1-af76-e4c5c9e1c3b7",
    "timeSlot":"9.30AM-10AM",

    "appointmentDate":"2023-07-31"

}
```

```
{"statusCode":400, "message":"Invalid User"}
```

- Successful Scenario



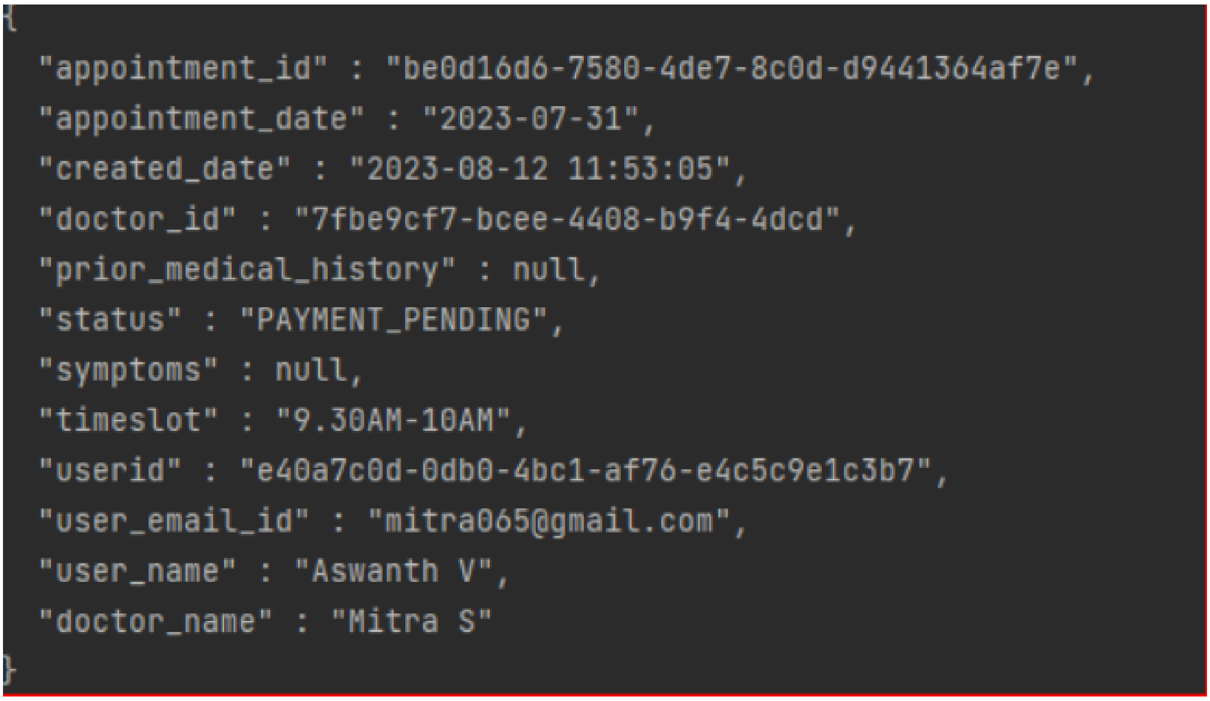
BookMyConsultation / appointment-service / User books an appointment

POST http://54.197.76.252:8082/appointments

```
{
    "doctorId":"7fbe9cf7-bcee-4408-b9f4-4dcd",
    "doctorName":"Mitra S",
    "userId":"e40a7c0d-0db0-4bc1-af76-e4c5c9e1c3b7",
    "timeSlot":"9.30AM-10AM",

    "appointmentDate":"2023-07-31"

}
```

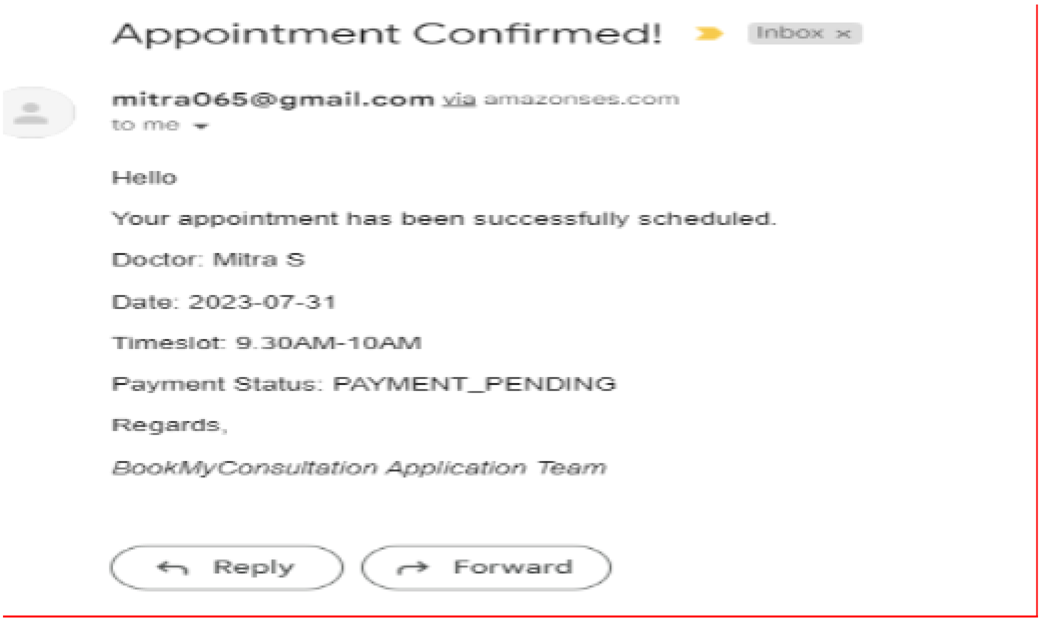
```
{
    "appointment_id": "be0d16d6-7580-4de7-8c0d-d9441364af7e",
    "appointment_date": "2023-07-31",
    "created_date": "2023-08-12 11:53:05",
    "doctor_id": "7fbe9cf7-bcee-4408-b9f4-4dcd",
    "prior_medical_history": null,
    "status": "PAYMENT_PENDING",
    "symptoms": null,
    "timeslot": "9.30AM-10AM",
    "userid": "e40a7c0d-0db0-4bc1-af76-e4c5c9e1c3b7",
    "user_email_id": "mitra065@gmail.com",
    "user_name": "Aswanth V",
    "doctor_name": "Mitra S"
}
```

- Kafka Consumer logs



```
{
  "appointment_id" : "be0d16d6-7580-4de7-8c0d-d9441364af7e",
  "appointment_date" : "2023-07-31",
  "created_date" : "2023-08-12 11:53:05",
  "doctor_id" : "7fbe9cf7-bcee-4408-b9f4-4dcd",
  "prior_medical_history" : null,
  "status" : "PAYMENT_PENDING",
  "symptoms" : null,
  "timeslot" : "9.30AM-10AM",
  "userid" : "e40a7c0d-0db0-4bc1-af76-e4c5c9e1c3b7",
  "user_email_id" : "mitra065@gmail.com",
  "user_name" : "Aswanth V",
  "doctor_name" : "Mitra S"
}
```

- Appointment Confirmation Mail



Appointment Confirmed!

mitra065@gmail.com via amazonses.com
to me

Hello

Your appointment has been successfully scheduled.

Doctor: Mitra S

Date: 2023-07-31

Timeslot: 9.30AM-10AM

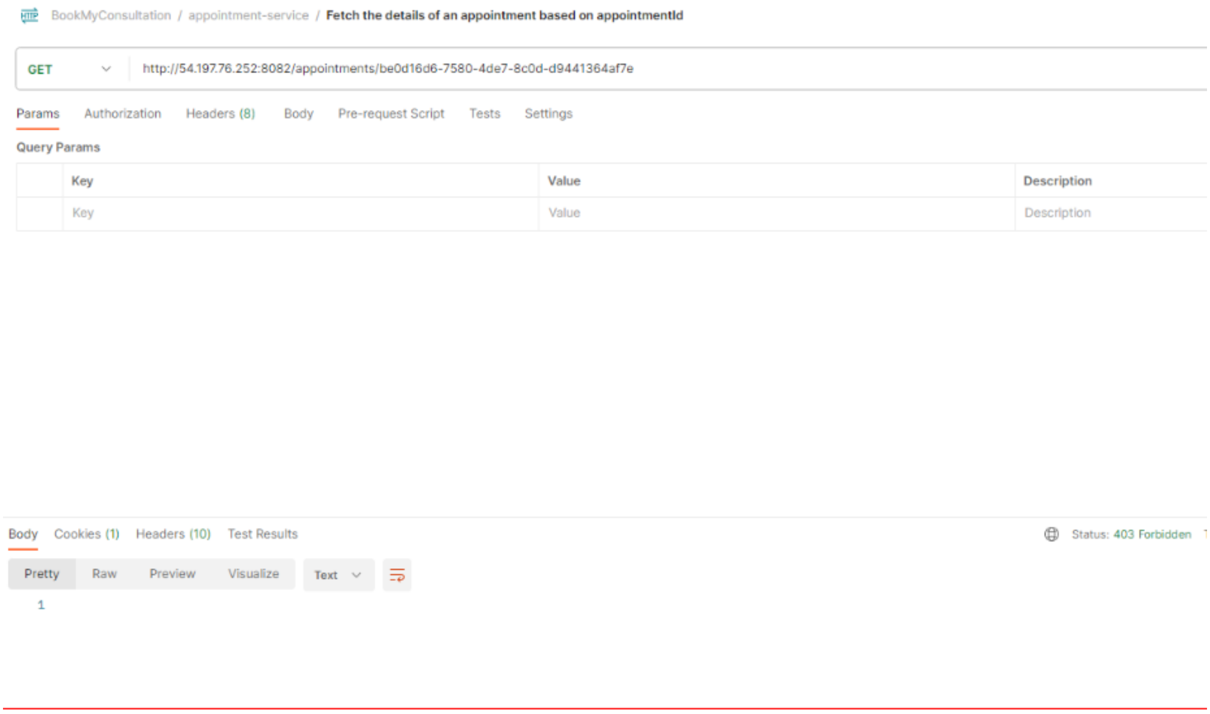
Payment Status: PAYMENT_PENDING

Regards,

BookMyConsultation Application Team

Endpoint 4: URI: `/appointments/{appointmentId}`
This endpoint is responsible for retrieving the details of an appointment.

- Forbidden Error for Unauthorised Users



BookMyConsultation / appointment-service / **Fetch the details of an appointment based on appointmentId**

GET http://54.197.76.252:8082/appointments/be0d16d6-7580-4de7-8c0d-d9441364af7e

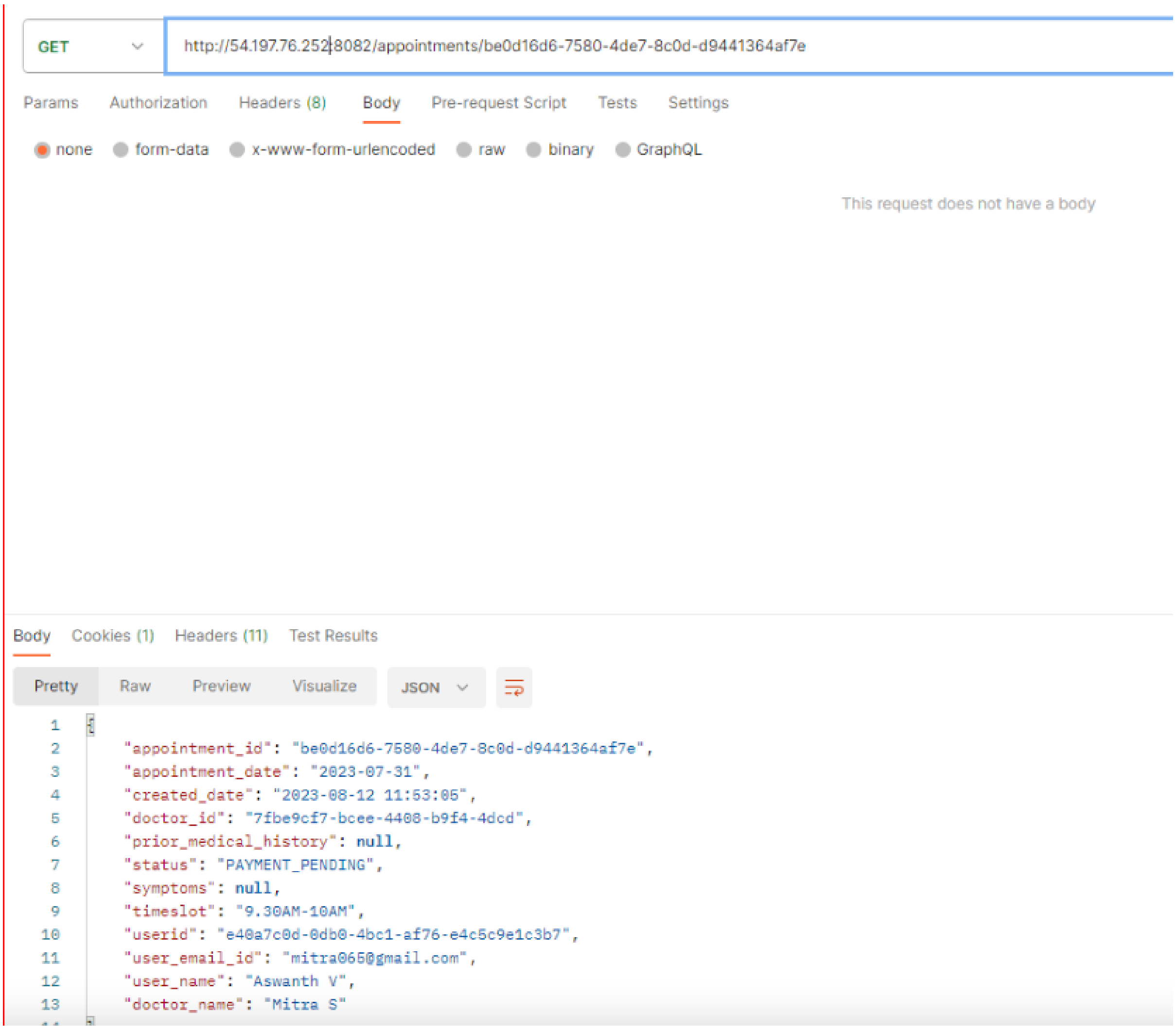
Query Params

	Key	Value	Description
	Key	Value	Description

Status: 403 Forbidden

```
1
```

- Successful Scenario



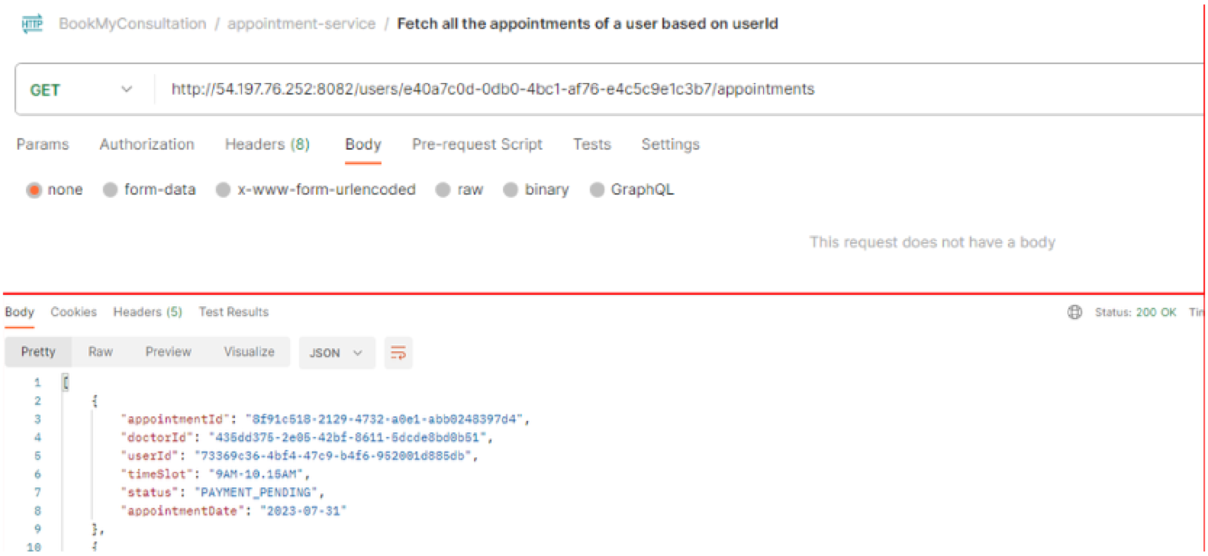
```
http://54.197.76.252:8082/appointments/be0d16d6-7580-4de7-8c0d-d9441364af7e
```

```
{
    "appointment_id": "be0d16d6-7580-4de7-8c0d-d9441364af7e",
    "appointment_date": "2023-07-31",
    "created_date": "2023-08-12 11:53:05",
    "doctor_id": "7fbe9cf7-bcee-4408-b9f4-4dcd",
    "prior_medical_history": null,
    "status": "PAYMENT_PENDING",
    "symptoms": null,
    "timeslot": "9.30AM-10AM",
    "userid": "e40a7c0d-0db0-4bc1-af76-e4c5c9e1c3b7",
    "user_email_id": "mitra065@gmail.com",
    "user_name": "Aswanth V",
    "doctor_name": "Mitra S"
```

Endpoint 5: URI: `/users/{userId}/appointments`

This endpoint is responsible for retrieving the details of all the appointments corresponding to a userId.

- Successful Scenario



BookMyConsultation / appointment-service / **Fetch all the appointments of a user based on userId**

GET http://54.197.76.252:8082/users/e40a7c0d-0db0-4bc1-af76-e4c5c9e1c3b7/appointments

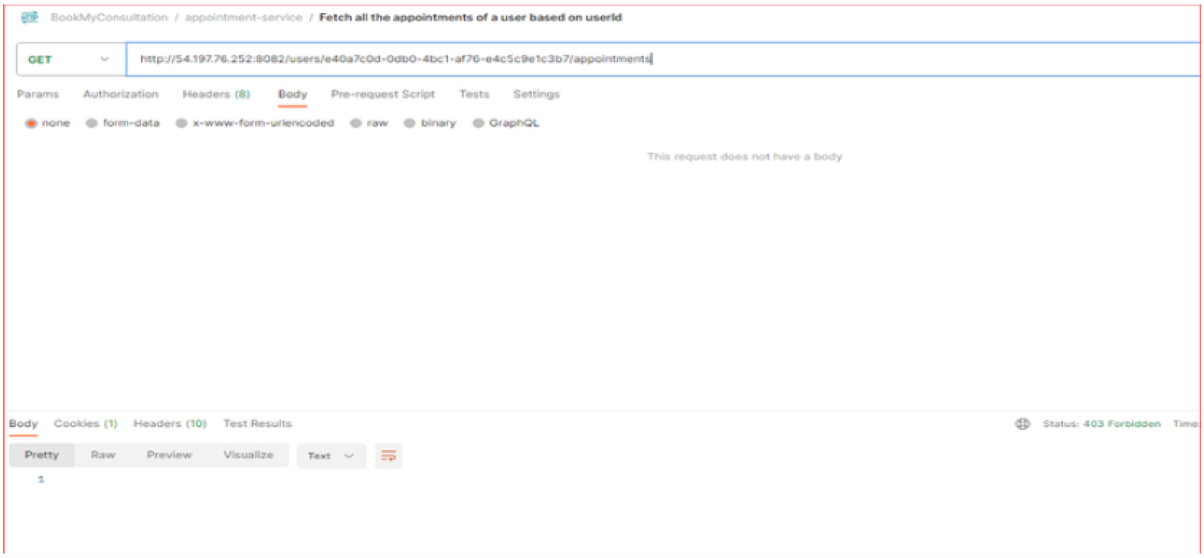
● none ○ form-data ○ x-www-form-urlencoded ○ raw ○ binary ○ GraphQL

This request does not have a body

Status: 200 OK

```
[
    {
        "appointmentId": "8f91c518-2129-4732-a0e1-abb0248397d4",
        "doctorId": "435dd375-2e05-42bf-8611-5dcde8bd0b51",
        "userId": "73369c36-4bf4-47c9-b4f6-952001d885db",
        "timeSlot": "9AM-10.15AM",
        "status": "PAYMENT_PENDING",
        "appointmentDate": "2023-07-31"
    },
    {
```

- Forbidden Error for Unauthorised Users



BookMyConsultation / appointment-service / **Fetch all the appointments of a user based on userId**

GET http://54.197.76.252:8082/users/e40a7c0d-0db0-4bc1-af76-e4c5c9e1c3b7/appointments

● none ○ form-data ○ x-www-form-urlencoded ○ raw ○ binary ○ GraphQL

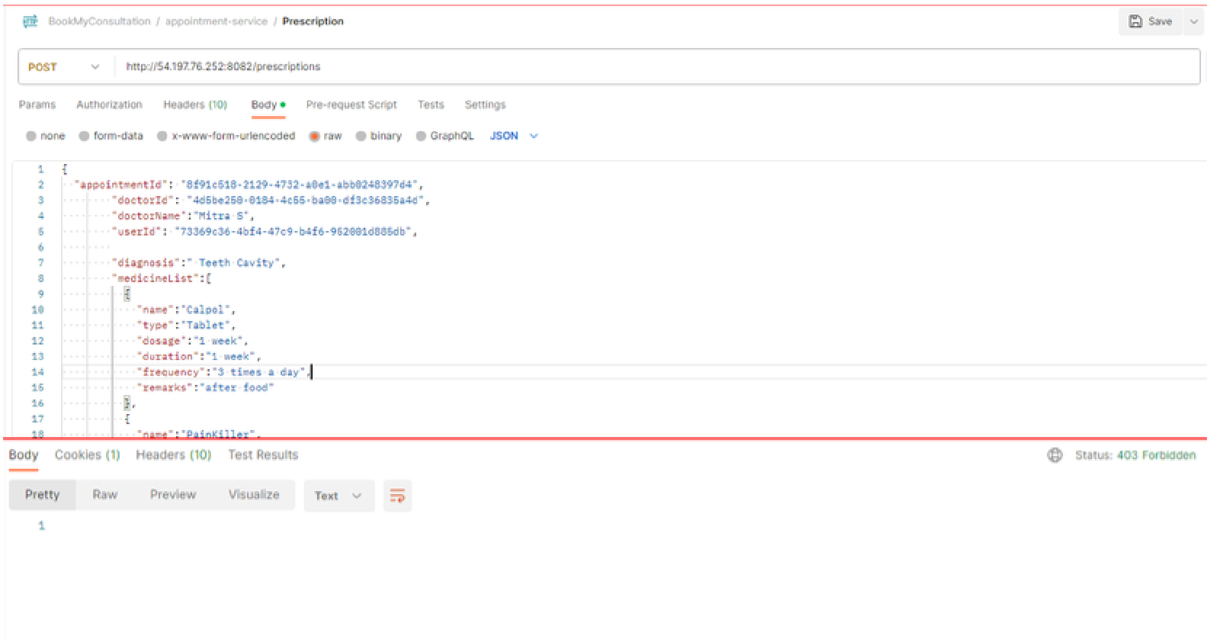
This request does not have a body

Status: 403 Forbidden

Endpoint 6: URI: `/prescriptions`

This endpoint is responsible for sending the prescriptions for the appointment.

- Forbidden Error for Unauthorised Users

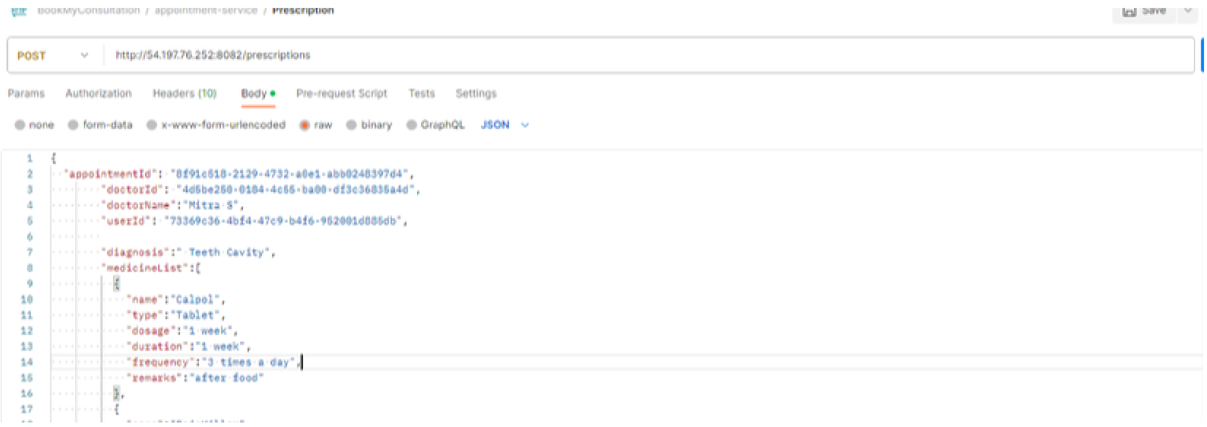


http://54.197.76.252:8082/prescriptions

```
{
  "appointmentId": "8f91c618-2129-4732-a0e1-abb0248397d4",
        "doctorId": "4d5be250-0184-4c55-ba00-df3c36835a4d",
        "doctorName":"Mitra S",
        "userId": "73369c36-4bf4-47c9-b4f6-952081d885db",

        "diagnosis":" Teeth Cavity",
        "medicineList":[
            {
              "name":"Calpol",
              "type":"Tablet",
              "dosage":"1 week",
              "duration":"1 week",
              "frequency":"3 times a day",
              "remarks":"after food"
            },
            {
              "name":"PainKiller",
```

Status: 403 Forbidden

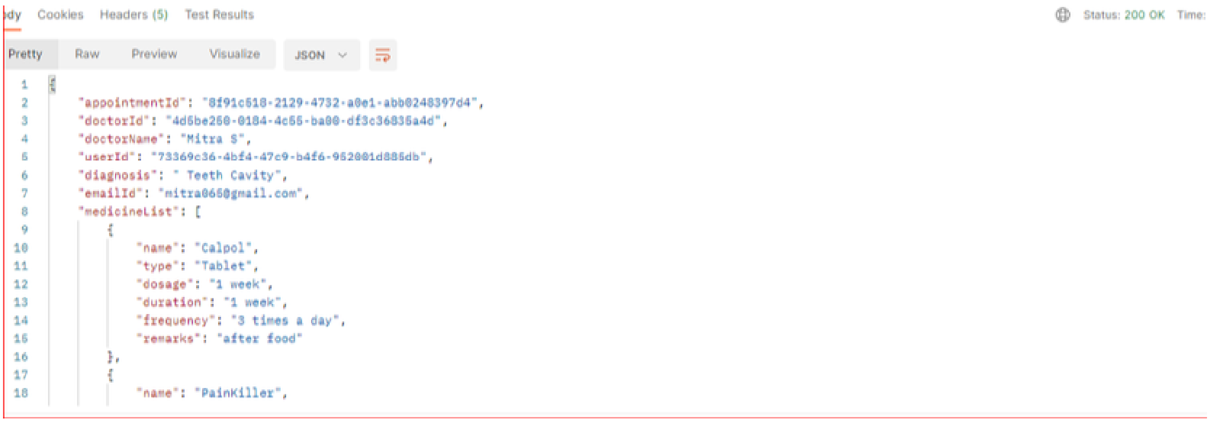
- Successful Scenario



BookMyConsultation / appointment-service / Prescription
Save
POST
Params
Authorization
Headers (10)
Body
Pre-request Script
Tests
Settings
○ none ○ form-data ○ x-www-form-urlencoded ● raw ○ binary ○ GraphQL JSON

http://54.197.76.252:8082/prescriptions

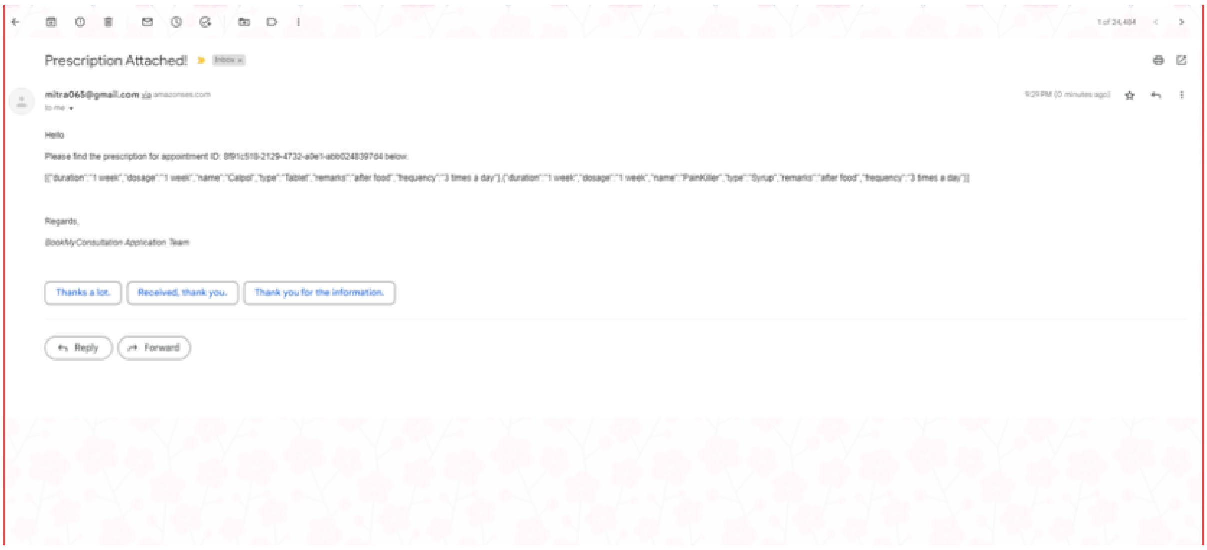
```
{
  "appointmentId": "8f91c618-2129-4732-a0e1-abb0248397d4",
        "doctorId": "4d5be250-0184-4c55-ba00-df3c36835a4d",
        "doctorName":"Mitra S",
        "userId": "73369c36-4bf4-47c9-b4f6-952081d885db",

        "diagnosis":" Teeth Cavity",
        "medicineList":[
            {
              "name":"Calpol",
              "type":"Tablet",
              "dosage":"1 week",
              "duration":"1 week",
              "frequency":"3 times a day",
              "remarks":"after food"
            },
            {
```



Status: 200 OK

```
{
    "appointmentId": "8f91c618-2129-4732-a0e1-abb0248397d4",
    "doctorId": "4d5be250-0184-4c55-ba00-df3c36835a4d",
    "doctorName": "Mitra S",
    "userId": "73369c36-4bf4-47c9-b4f6-952081d885db",
    "diagnosis": " Teeth Cavity",
    "emailId": "mitra065@gmail.com",
    "medicineList": [
        {
            "name": "Calpol",
            "type": "Tablet",
            "dosage": "1 week",
            "duration": "1 week",
            "frequency": "3 times a day",
            "remarks": "after food"
        },
        {
            "name": "PainKiller",
```

- Prescription Mail Received



Prescription Attached!

mitra065@gmail.com via amazonses.com

to me

Hello

Please find the prescription for appointment ID: 8f91c518-2129-4732-a0e1-abb0248397d4 below.

[{"duration":"1 week","dosage":"1 week","name":"Calpol","type":"Tablet","remarks":"after food","frequency":"3 times a day"},{"duration":"1 week","dosage":"1 week","name":"PainKiller","type":"Syrup","remarks":"after food","frequency":"3 times a day"}]

Regards,

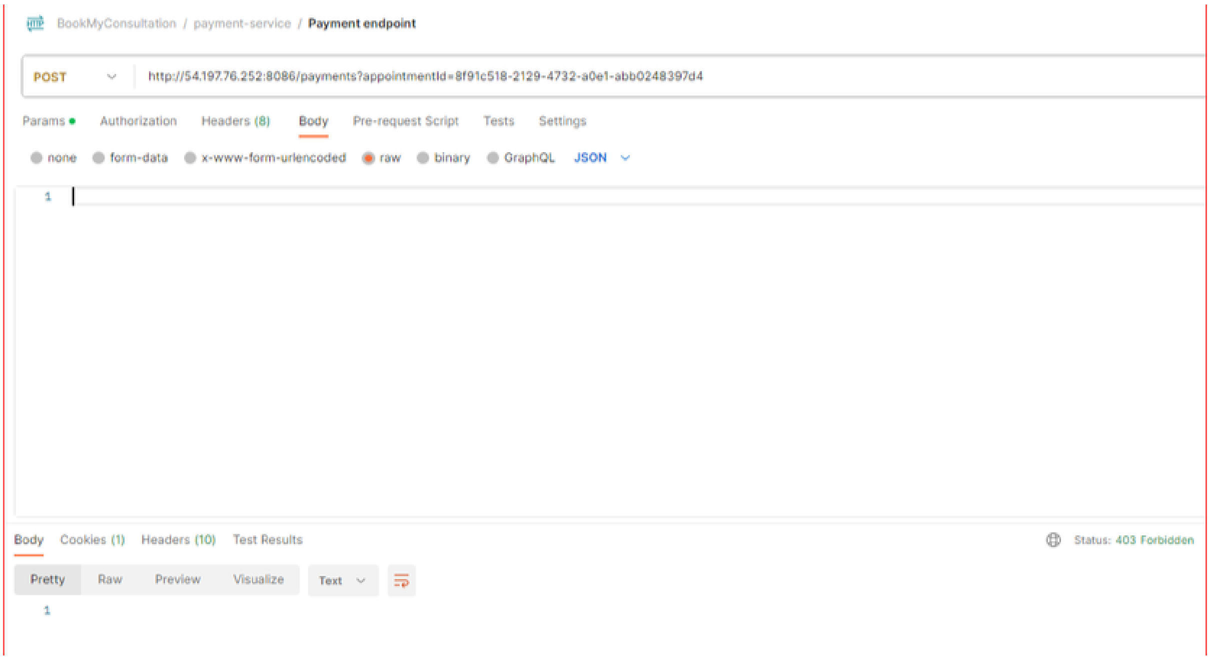
BookMyConsultation Application Team

Payment Service

Endpoint 1: URI: `/payments?appointmentId=<the appointmentId for which you want to make apayment>`

This endpoint is responsible for making payments.

- Forbidden Error for Unauthorised Users

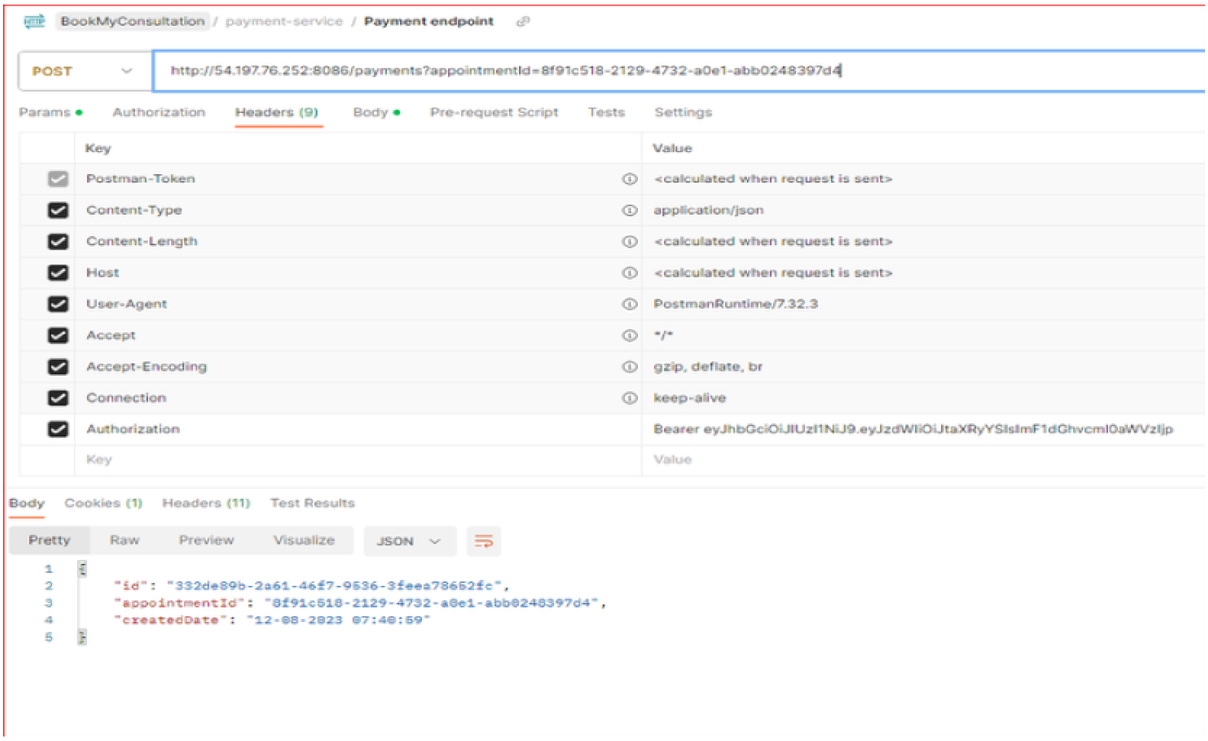


BookMyConsultation / payment-service / **Payment endpoint**

POST http://54.197.76.252:8086/payments?appointmentId=8f91c518-2129-4732-a0e1-abb0248397d4

Status: 403 Forbidden

- Successful Scenario



BookMyConsultation / payment-service / **Payment endpoint**

POST http://54.197.76.252:8086/payments?appointmentId=8f91c518-2129-4732-a0e1-abb0248397d4

	Key	Value
☑	Postman-Token	<calculated when request is sent>
☑	Content-Type	application/json
☑	Content-Length	<calculated when request is sent>
☑	Host	<calculated when request is sent>
☑	User-Agent	PostmanRuntime/7.32.3
☑	Accept	*/*
☑	Accept-Encoding	gzip, deflate, br
☑	Connection	keep-alive
☑	Authorization	Bearer eyJhbGciOiJIUzI1NiJ9.eyJzdWIiOiJtaXRyYSIsImF1dGhvcml0aWVzIjp
	Key	Value

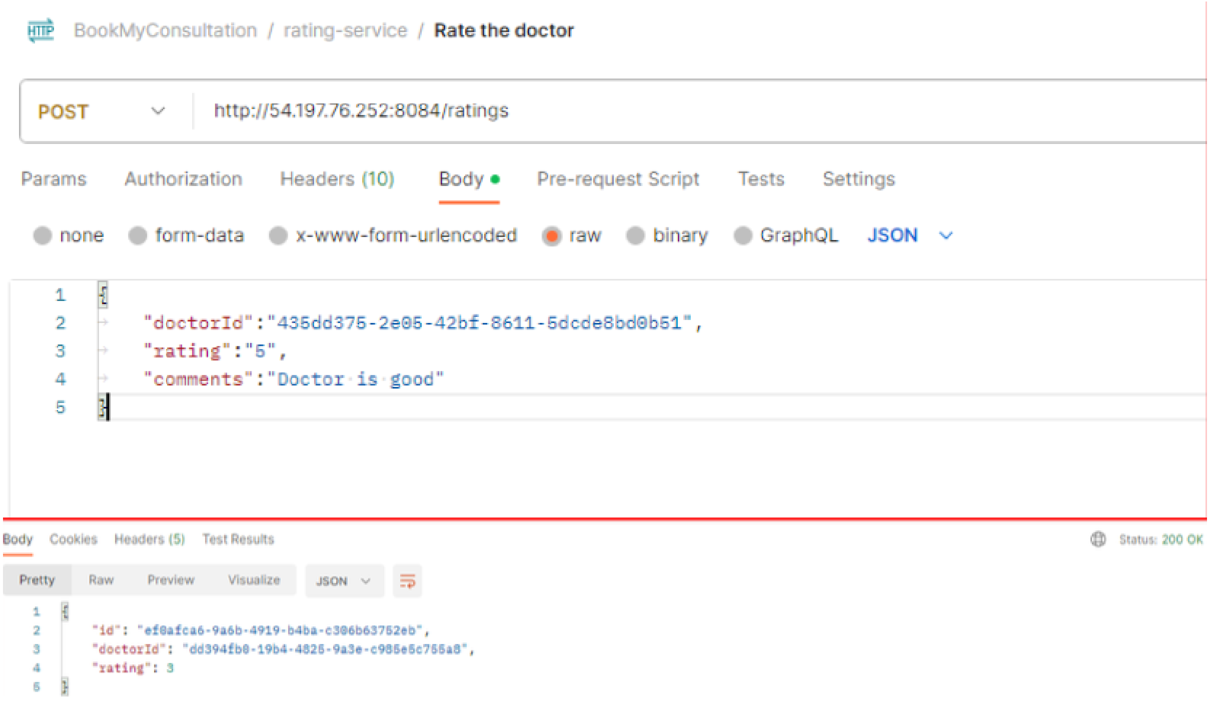
```
{
    "id": "332de89b-2a61-46f7-9536-3feea78652fc",
    "appointmentId": "8f91c518-2129-4732-a0e1-abb0248397d4",
    "createdDate": "12-08-2023 07:40:59"
}
```

Rating Service

Endpoint 1: URI: `/ratings`

This endpoint is used by the users to submit the ratings of their experience with the doctor with whom
they had an appointment.

- Successful Scenario



POST http://54.197.76.252:8084/ratings

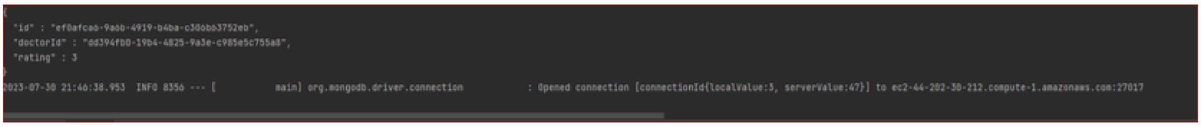
```
{
    "doctorId":"435dd375-2e05-42bf-8611-5dcde8bd0b51",
    "rating":"5",
    "comments":"Doctor is good"
}
```

Status: 200 OK

Body: Pretty, JSON

```
{
    "id": "ef0afca6-9a6b-4919-b4ba-c306b63752eb",
    "doctorId": "dd394fb8-19b4-4825-9a3e-c985e5c755a8",
    "rating": 3
}
```

- Kafka Logs Screenshot



```
{
  "id" : "ef0afca6-9a6b-4919-b4ba-c306b63752eb",
  "doctorId" : "dd394fb8-19b4-4825-9a3e-c985e5c755a8",
  "rating" : 3
}
2023-07-30 21:46:38.953  INFO 8356 --- [           main] org.mongodb.driver.connection            : Opened connection [connectionId{localValue:3, serverValue:47}] to ec2-44-202-30-212.compute-1.amazonaws.com:27017
```

Notification Service

The notification service uses Kafka to receive notifications sent from each service. For particular services, we receive confirmations to send emails using Amazon SES. The functionality has been implemented as shown in each of the required APIs.

The ***CreateDoctor*** and ***CreateUser*** API sends verification email using SES to the respective email ID to confirm that the email exists.

The ***approveDoctor***, ***rejectDoctor***, ***setAppointment*** and ***setPrescription*** APIs uses SES to send custom messages to the respective email ID of the Doctor or the User.

Other services send kafka notifications to print by themselves, as shown in each API