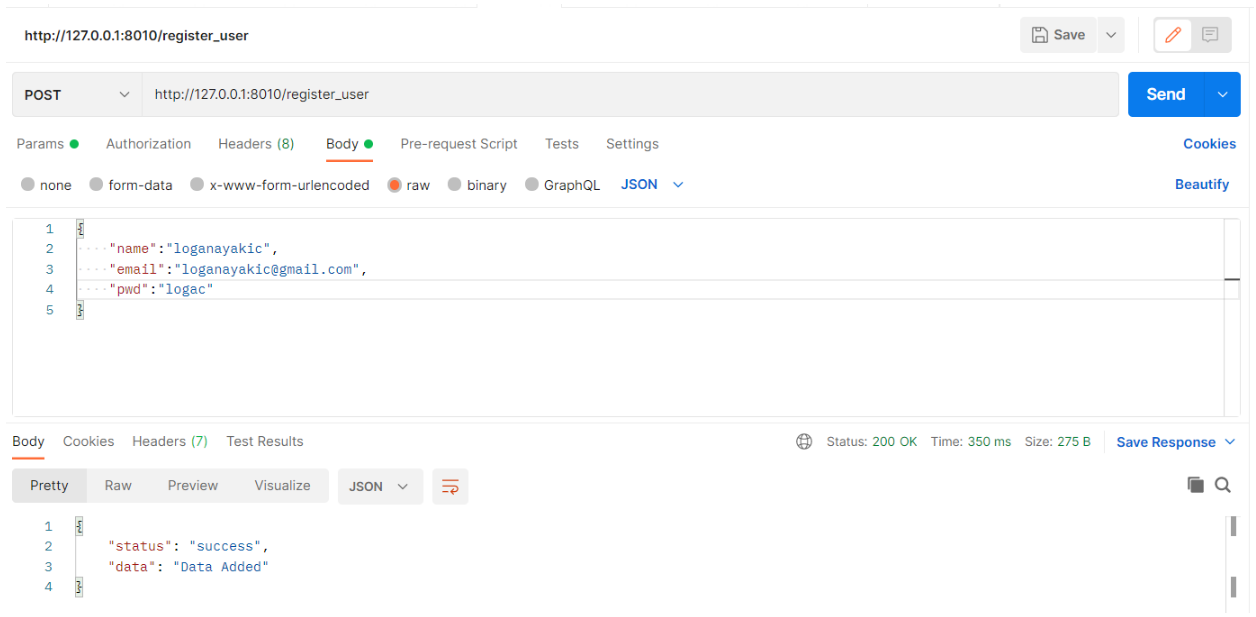
Unit testing

1.Register user (Create op.)

Successful registration of new user



http://127.0.0.1:8010/register_user

POST http://127.0.0.1:8010/register_user

Body: raw JSON

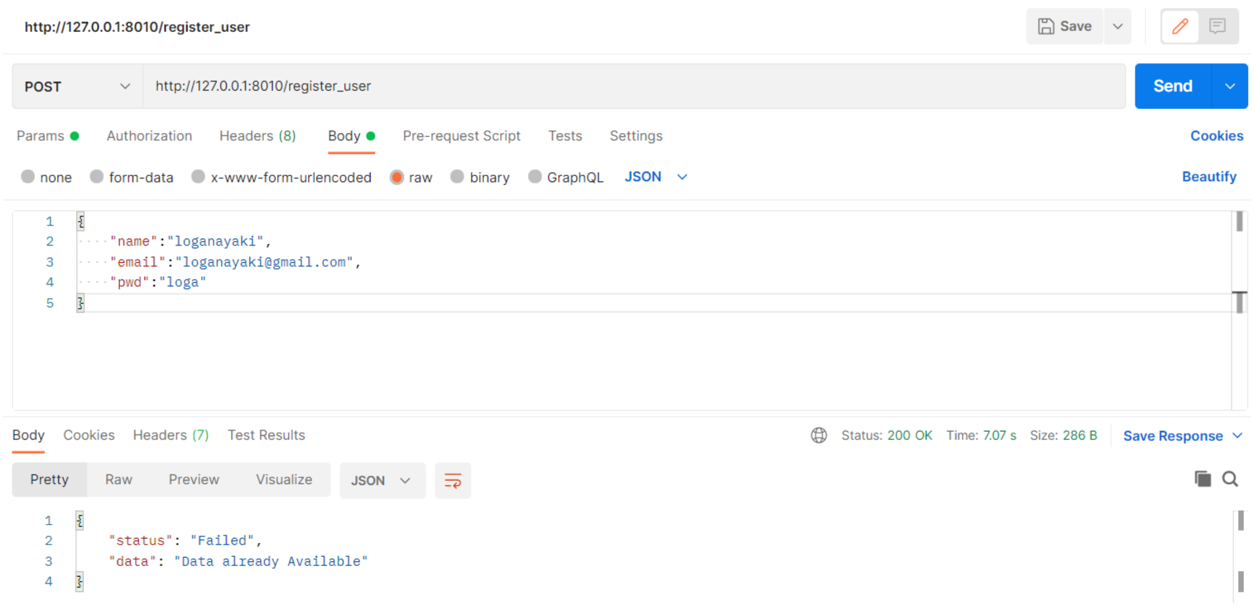
```
{
    "name":"loganayakic",
    "email":"loganayakic@gmail.com",
    "pwd":"logac"
}
```

Status: 200 OK Time: 350 ms Size: 275 B

```
{
    "status": "success",
    "data": "Data Added"
}
```

2. Trying to register already existing user based on email id.

o/p: Data already available



http://127.0.0.1:8010/register_user

POST http://127.0.0.1:8010/register_user

Body: raw JSON

```
{
    "name":"loganayaki",
    "email":"loganayaki@gmail.com",
    "pwd":"loga"
}
```

Status: 200 OK Time: 7.07 s Size: 286 B

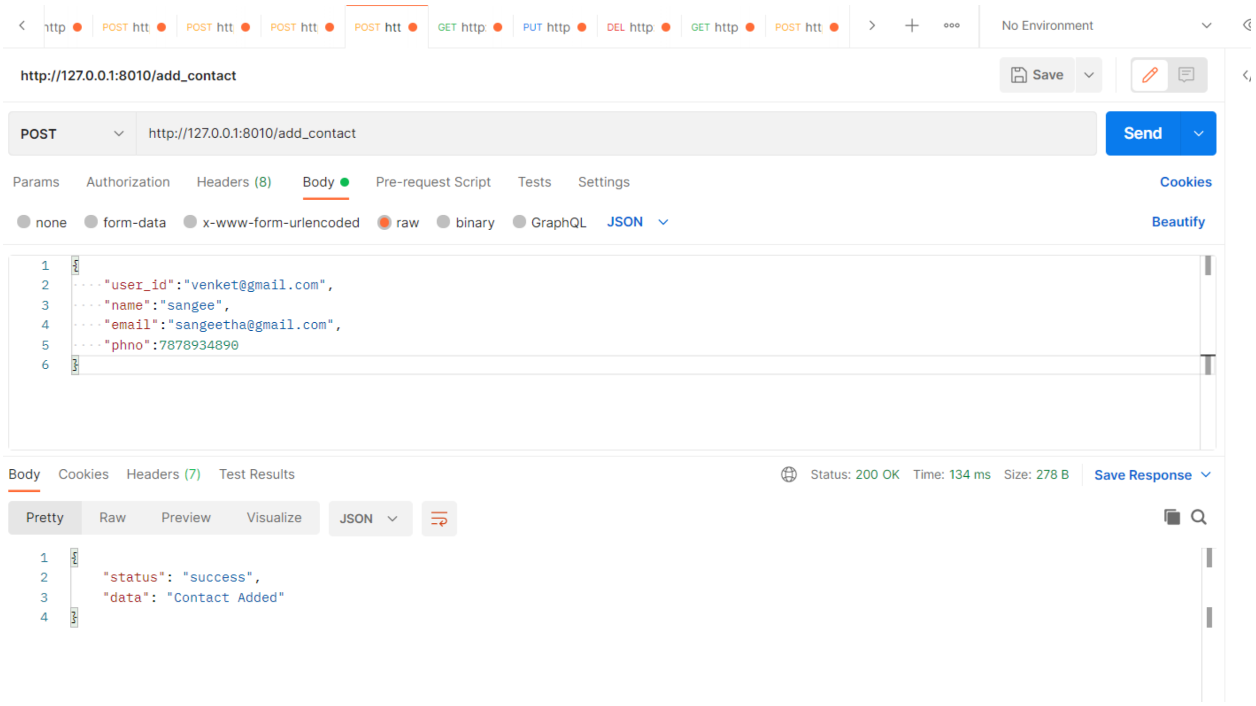
```
{
    "status": "Failed",
    "data": "Data already Available"
}
```

3. Adding contact for registered user

User id must be registered user id

i/p: Name, email and phone no.

o/p:Contact Added



http://127.0.0.1:8010/add_contact

POST http://127.0.0.1:8010/add_contact

```
{
    "user_id":"venket@gmail.com",
    "name":"sangee",
    "email":"sangeetha@gmail.com",
    "phno":7878934890
}
```

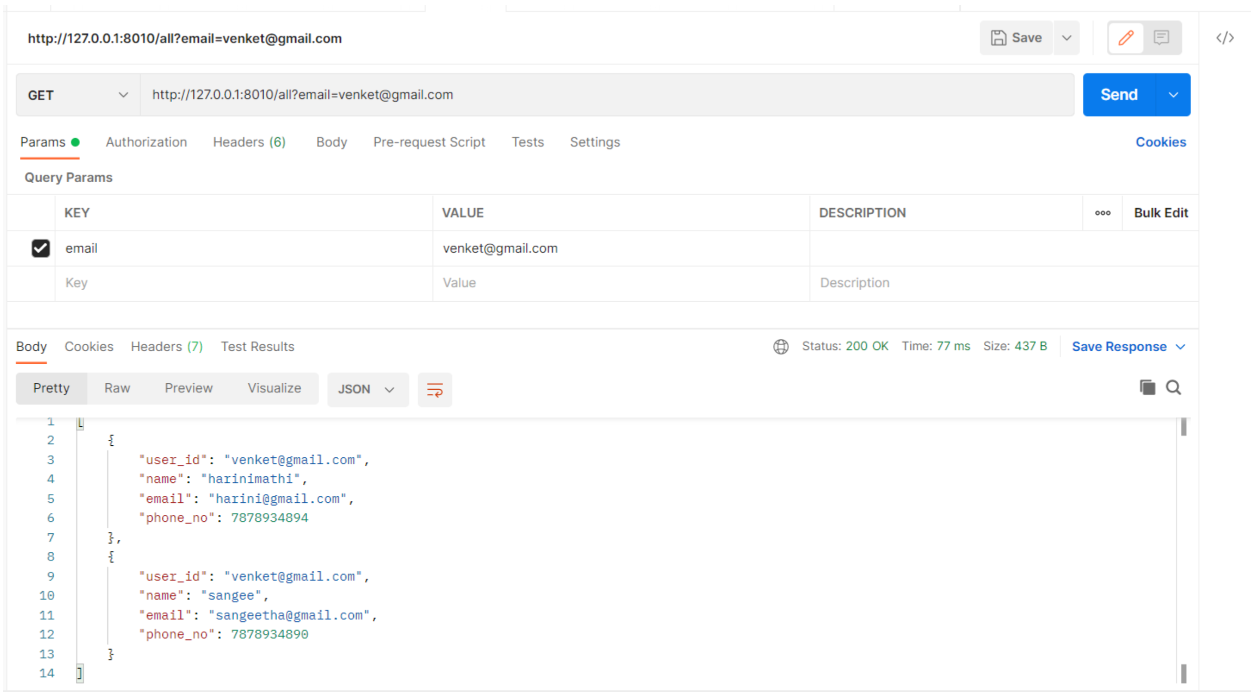
Status: 200 OK Time: 134 ms Size: 278 B

```
{
    "status": "success",
    "data": "Contact Added"
}
```

4.Fetch all contact of a particular user. (Read op.)

i/p: user id

o/p: list of contacts



http://127.0.0.1:8010/all?email=venket@gmail.com

GET http://127.0.0.1:8010/all?email=venket@gmail.com

	KEY	VALUE	DESCRIPTION
✓	email	venket@gmail.com	
	Key	Value	Description

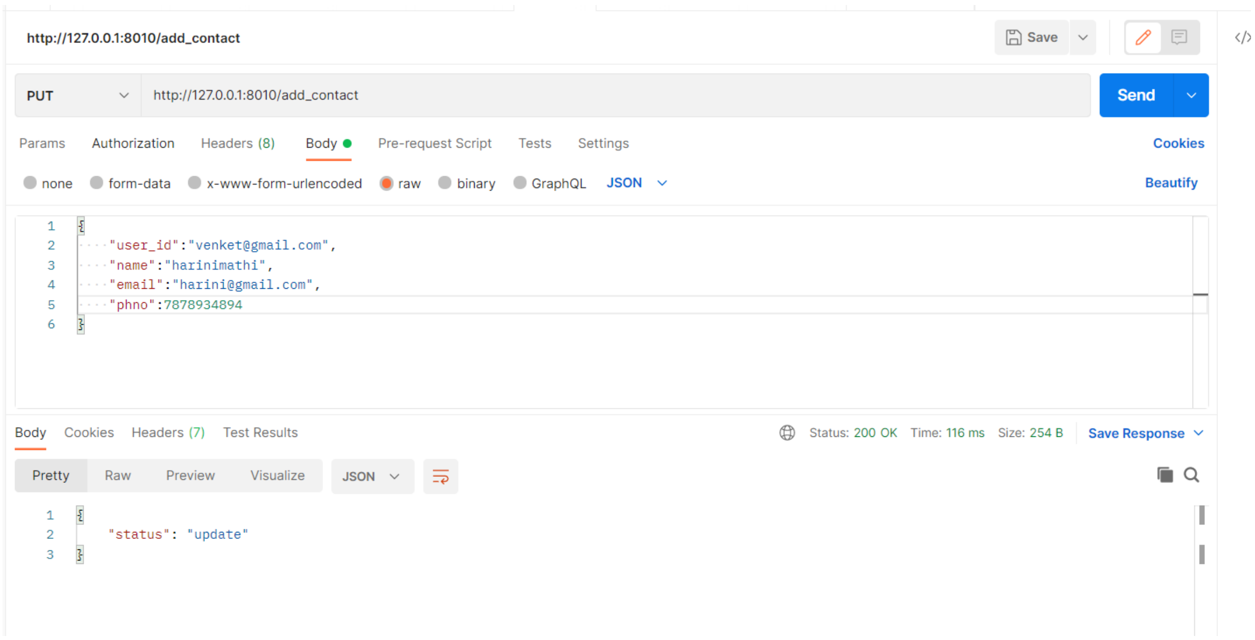
Status: 200 OK Time: 77 ms Size: 437 B

```
[
    {
        "user_id": "venket@gmail.com",
        "name": "harinimathi",
        "email": "harini@gmail.com",
        "phone_no": 7878934894
    },
    {
        "user_id": "venket@gmail.com",
        "name": "sangee",
        "email": "sangeetha@gmail.com",
        "phone_no": 7878934890
    }
]
```

5. Update a contact's phone no./name (Update op.)

i/p: user id, contact's email id, new phone no./name

o/p: Successful updation



PUT http://127.0.0.1:8010/add_contact

```
{
    "user_id":"venket@gmail.com",
    "name":"harinimathi",
    "email":"harini@gmail.com",
    "phno":7878934894
}
```

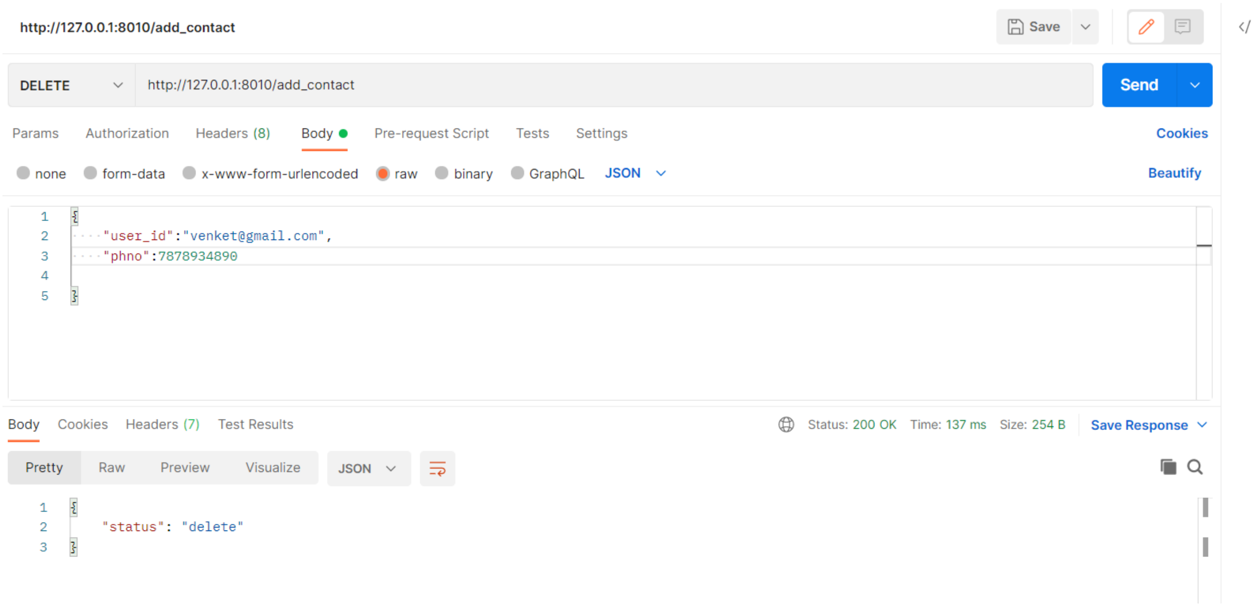
Status: 200 OK Time: 116 ms Size: 254 B

```
{
    "status": "update"
}
```

6.Delete contact (Delete op.)

i/p: user id, phone no.

o/p: Successfully deleted contact



DELETE http://127.0.0.1:8010/add_contact

```
{
    "user_id":"venket@gmail.com",
    "phno":7878934890

}
```

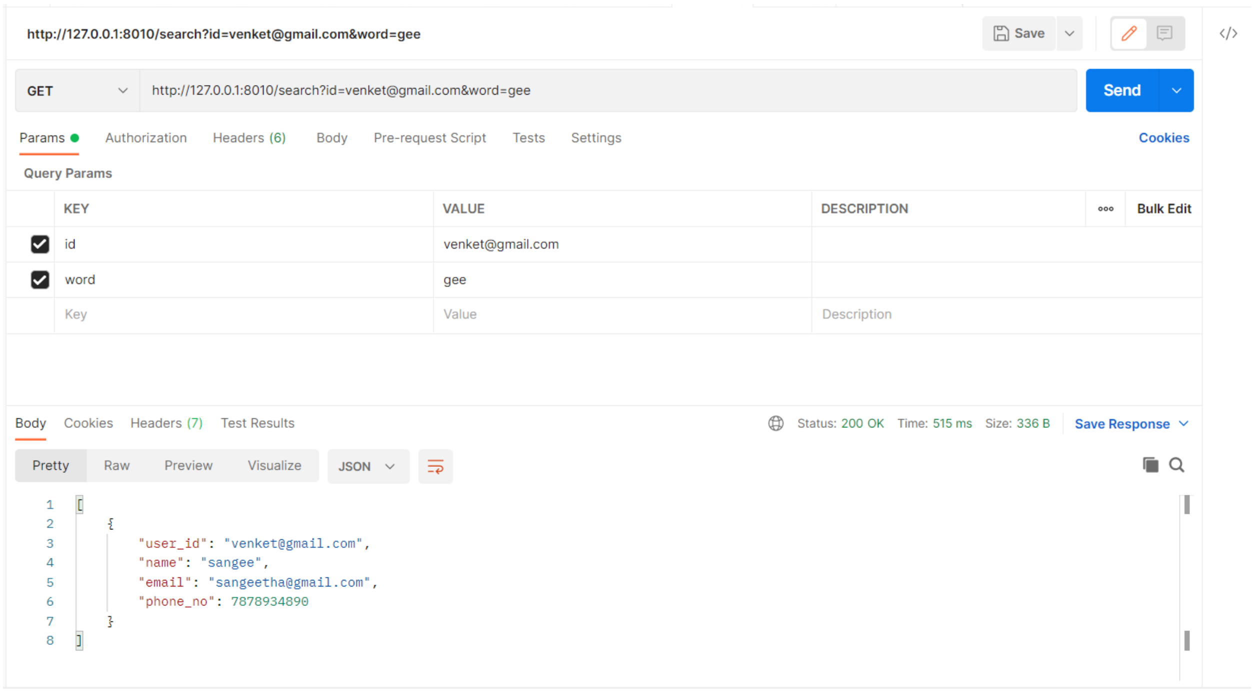
Status: 200 OK Time: 137 ms Size: 254 B

```
{
    "status": "delete"
}
```

7. Search contact based on email id and name

i/p: user id, keyword

o/p: list of contacts having keyword in name or email.



http://127.0.0.1:8010/search?id=venket@gmail.com&word=gee

GET http://127.0.0.1:8010/search?id=venket@gmail.com&word=gee

Query Params

KEY	VALUE	DESCRIPTION
id	venket@gmail.com	
word	gee	

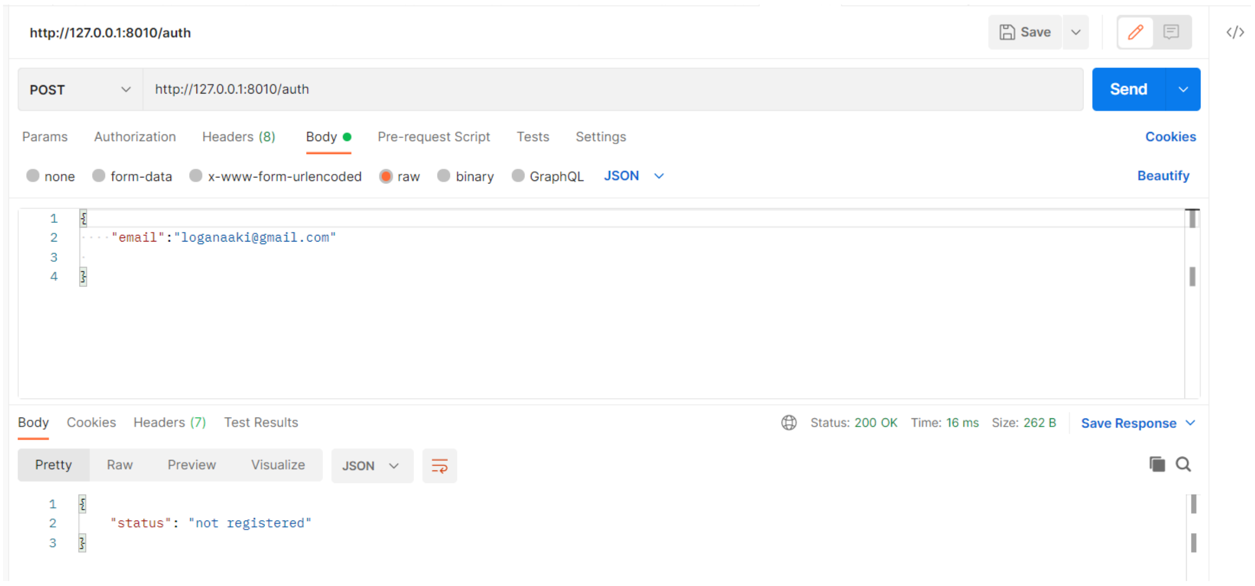
Status: 200 OK Time: 515 ms Size: 336 B

```
[
    {
        "user_id": "venket@gmail.com",
        "name": "sangee",
        "email": "sangeetha@gmail.com",
        "phone_no": 7878934890
    }
]
```

8. Check for authentication

i/p: email

o/p: Registered or Not Registered



http://127.0.0.1:8010/auth

POST http://127.0.0.1:8010/auth

```
{
    "email":"loganaaki@gmail.com"

}
```

Status: 200 OK Time: 16 ms Size: 262 B

```
{
    "status": "not registered"
}
```