PHOTO: DARCY PETERKA

E6156 - Topics in SW Engineering Management
Nestly: Cloud Rental Property Management System

Zhang, Youcheng(yz4589), Yao, Xirui(xy2571), Chen, Shiying(sc5299),
Lin, Runze(rl3376), Wang, Isa(yw3886)

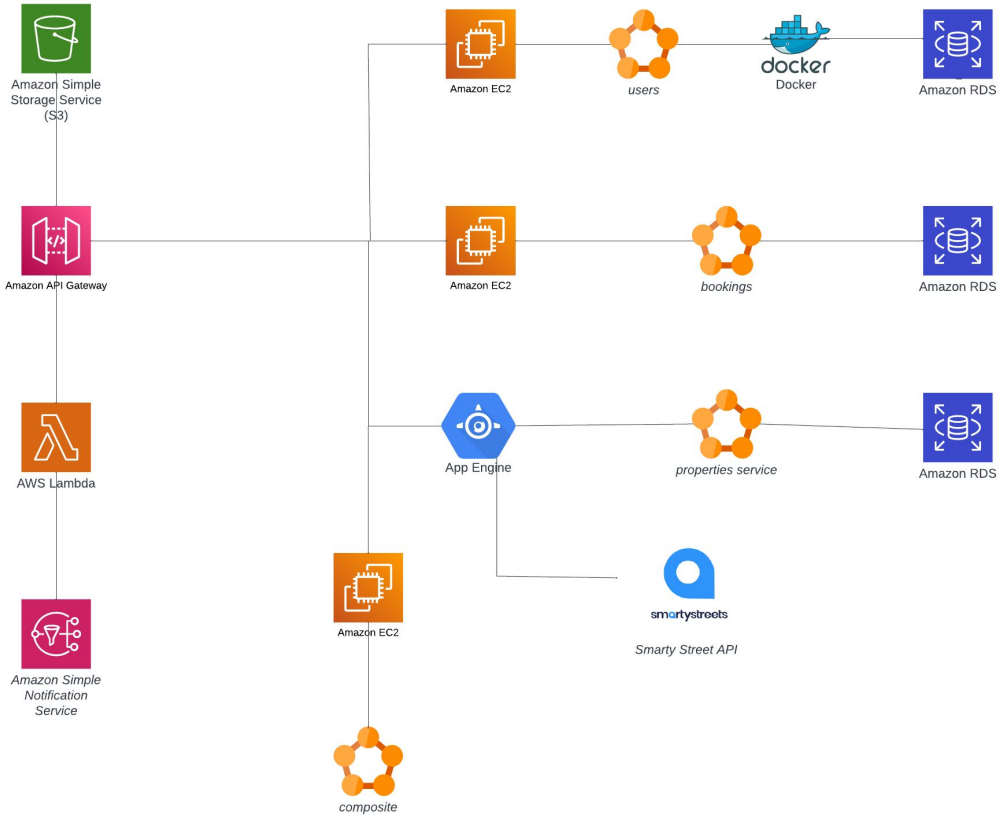
Content

1. Introduction

   Nestly: Cloud Rental Property Management

   - Facilitating seamless property listings and bookings for diverse users.
   - Ensuring smooth interactions between hosts, guests, and administrators.
   - Integrating multiple cloud services and maintaining architectural consistency across them.
2. 10 Required implementation demo
3. How our application works

Overall Structure

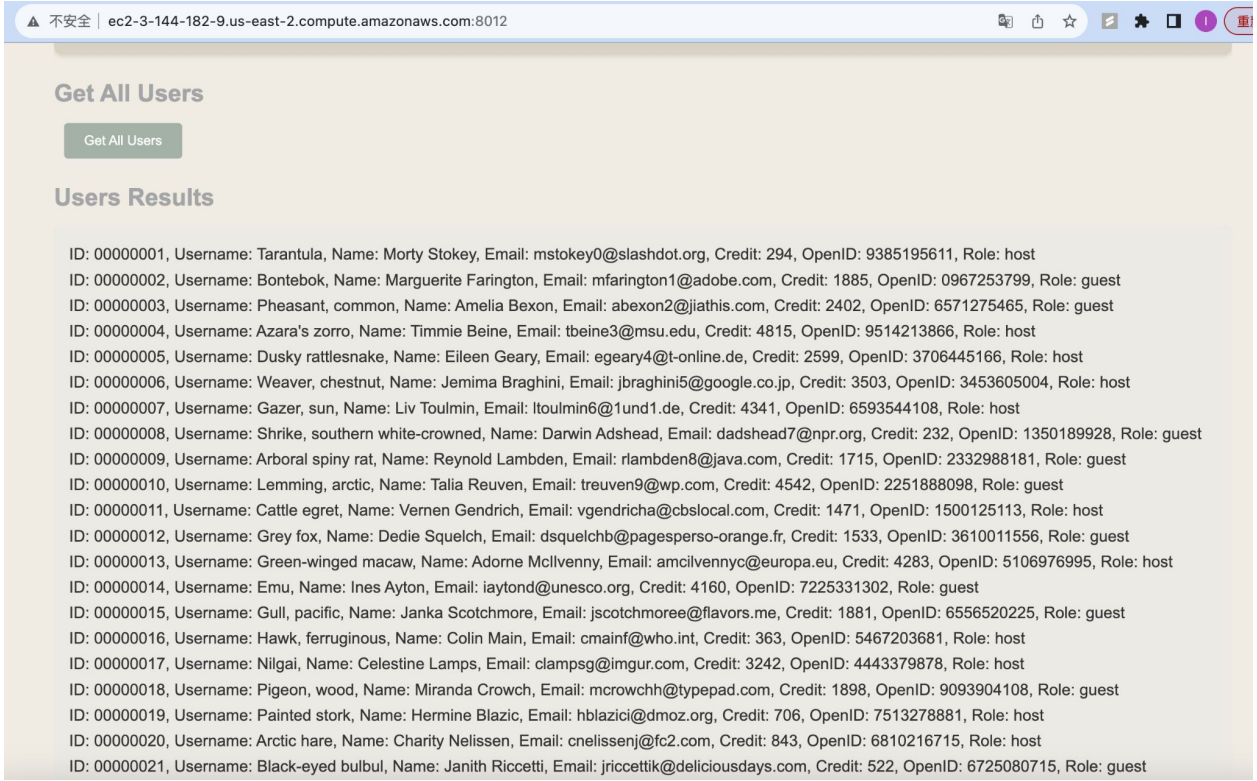


Enhance Microservices Implementation

Enhanced Microservices

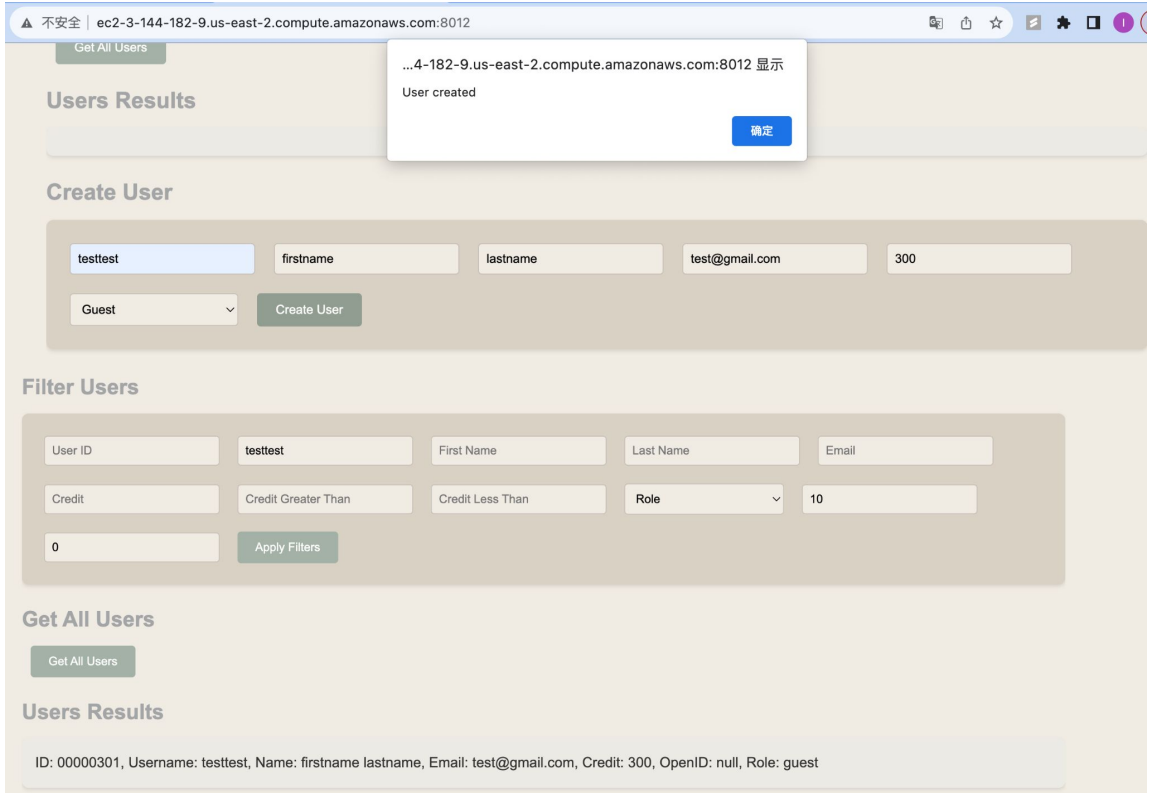
finished on all 3 microservices, using Users as example

GET:
(getting all here)



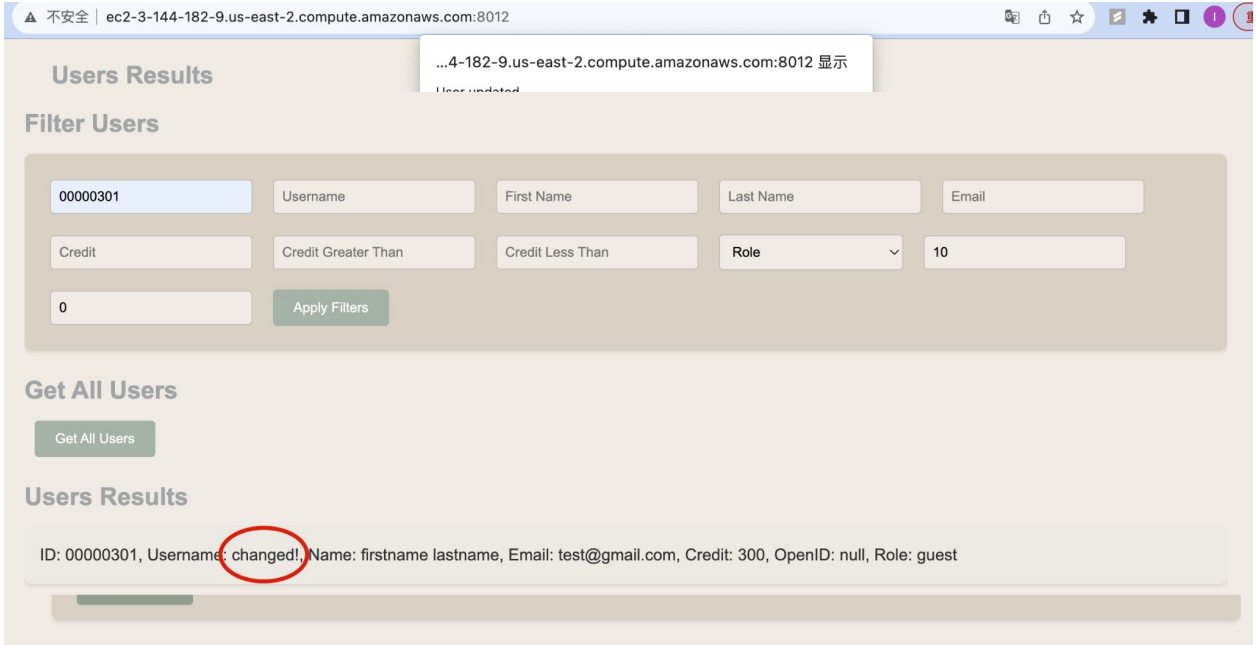
Enhance Microservices

PUT: (added a test user ‘testtest’ and the GET on username=testtest)



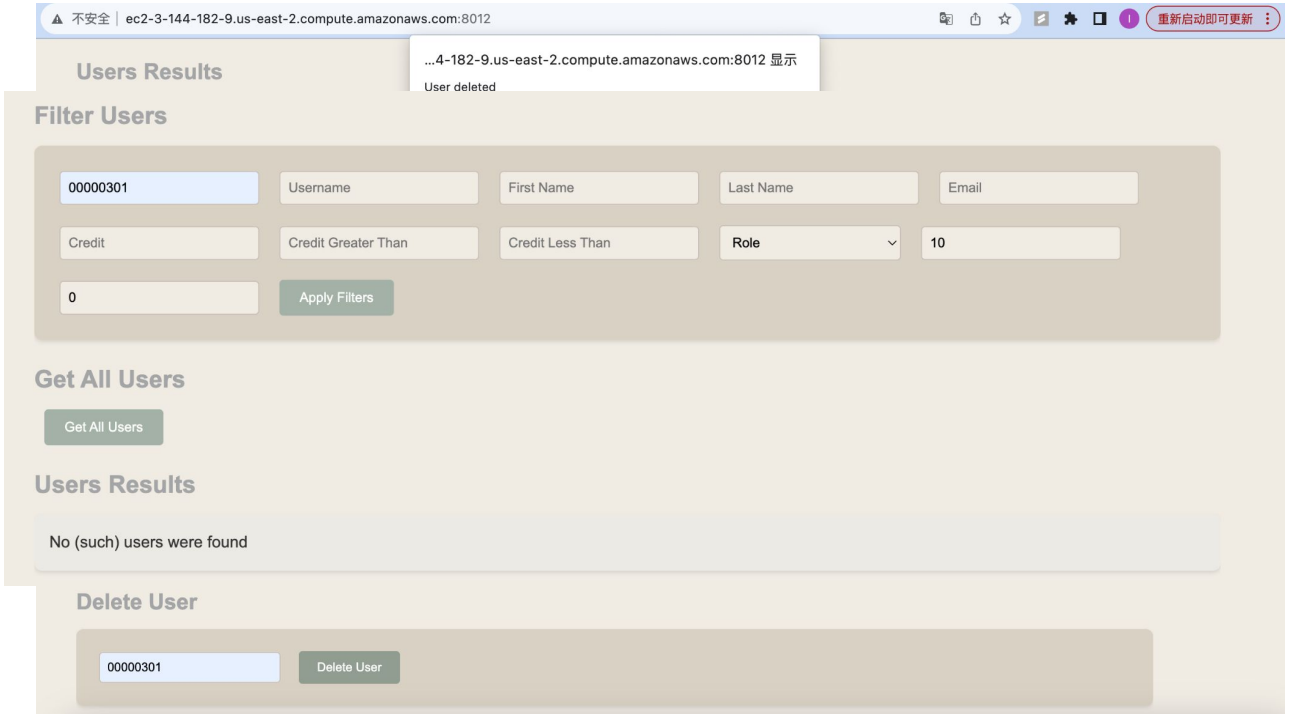
Enhance Microservices

POST: (updated the added user using user id=00000301, username to ‘changed!’, and the GET on user id=00000301, showing username changed to ‘changed!’)



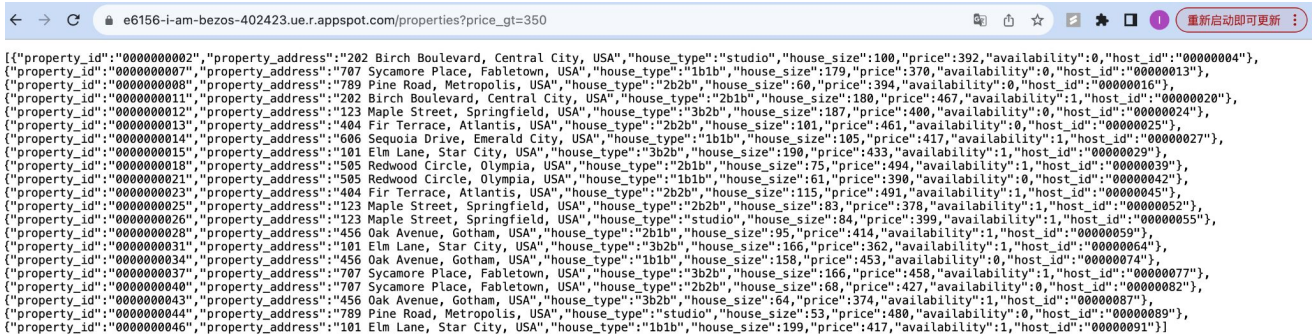
Enhance Microservices

DELETE (DELETE user id=00000301, and GET user id=00000301 returning “No (such) users were found”)



Enhance Microservices

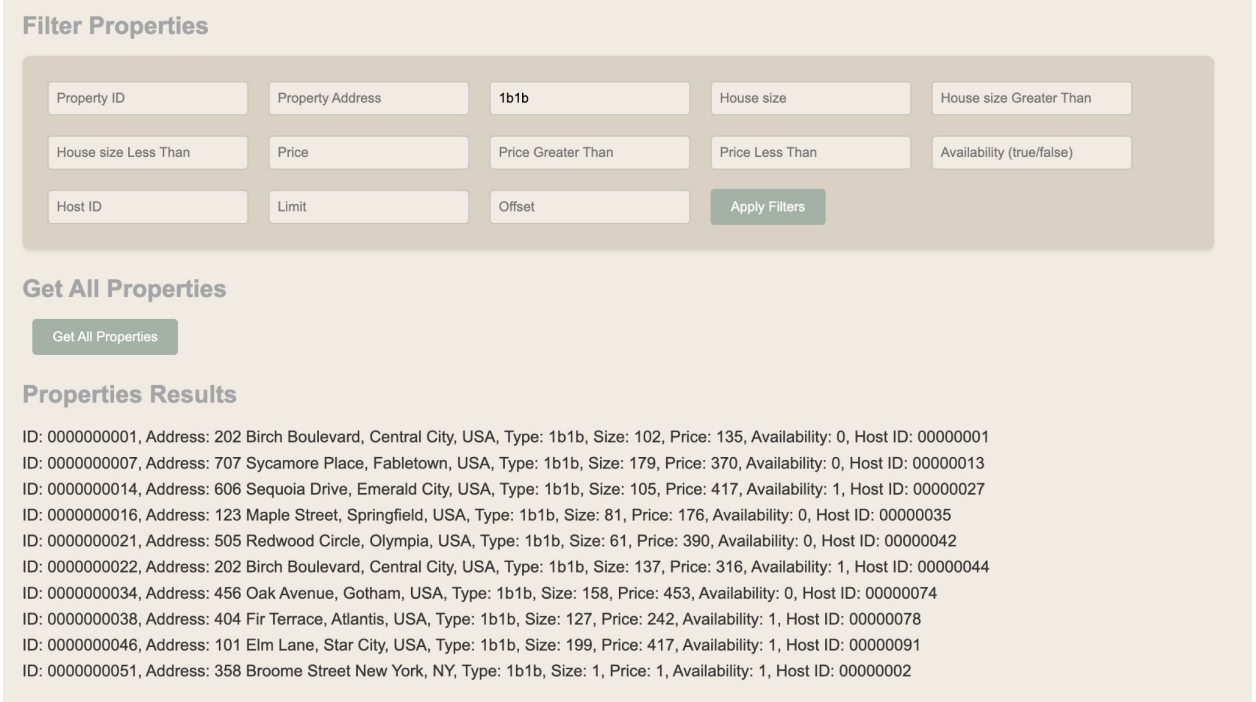
Querying (realized on Properties) (E.g. price_gt=350, return properties with price greater than 350)




```
[{"property_id":"0000000002","property_address":"202 Birch Boulevard, Central City, USA","house_type":"studio","house_size":100,"price":392,"availability":0,"host_id":"00000004"},
{"property_id":"0000000007","property_address":"707 Sycamore Place, Fabletown, USA","house_type":"1b1b","house_size":179,"price":370,"availability":0,"host_id":"00000013"},
{"property_id":"0000000008","property_address":"789 Pine Road, Metropolis, USA","house_type":"2b2b","house_size":60,"price":394,"availability":0,"host_id":"00000016"},
{"property_id":"0000000011","property_address":"202 Birch Boulevard, Central City, USA","house_type":"2b1b","house_size":180,"price":467,"availability":1,"host_id":"00000020"},
{"property_id":"0000000012","property_address":"123 Maple Street, Springfield, USA","house_type":"3b2b","house_size":187,"price":400,"availability":0,"host_id":"00000024"},
{"property_id":"0000000013","property_address":"404 Fir Terrace, Atlantis, USA","house_type":"2b2b","house_size":101,"price":461,"availability":0,"host_id":"00000025"},
{"property_id":"0000000014","property_address":"606 Sequoia Drive, Emerald City, USA","house_type":"1b1b","house_size":105,"price":417,"availability":1,"host_id":"00000027"},
{"property_id":"0000000015","property_address":"101 Elm Lane, Star City, USA","house_type":"3b2b","house_size":190,"price":433,"availability":1,"host_id":"00000029"},
{"property_id":"0000000018","property_address":"505 Redwood Circle, Olympia, USA","house_type":"2b1b","house_size":75,"price":494,"availability":1,"host_id":"00000039"},
{"property_id":"0000000021","property_address":"505 Redwood Circle, Olympia, USA","house_type":"1b1b","house_size":61,"price":390,"availability":0,"host_id":"00000042"},
{"property_id":"0000000023","property_address":"404 Fir Terrace, Atlantis, USA","house_type":"2b2b","house_size":115,"price":491,"availability":1,"host_id":"00000045"},
{"property_id":"0000000025","property_address":"123 Maple Street, Springfield, USA","house_type":"2b2b","house_size":83,"price":378,"availability":1,"host_id":"00000052"},
{"property_id":"0000000026","property_address":"123 Maple Street, Springfield, USA","house_type":"studio","house_size":84,"price":399,"availability":1,"host_id":"00000055"},
{"property_id":"0000000028","property_address":"456 Oak Avenue, Gotham, USA","house_type":"2b1b","house_size":95,"price":414,"availability":1,"host_id":"00000059"},
{"property_id":"0000000031","property_address":"101 Elm Lane, Star City, USA","house_type":"3b2b","house_size":166,"price":362,"availability":1,"host_id":"00000064"},
{"property_id":"0000000034","property_address":"456 Oak Avenue, Gotham, USA","house_type":"1b1b","house_size":158,"price":453,"availability":0,"host_id":"00000074"},
{"property_id":"0000000037","property_address":"707 Sycamore Place, Fabletown, USA","house_type":"3b2b","house_size":166,"price":458,"availability":1,"host_id":"00000077"},
{"property_id":"0000000040","property_address":"707 Sycamore Place, Fabletown, USA","house_type":"2b2b","house_size":68,"price":427,"availability":0,"host_id":"00000082"},
{"property_id":"0000000043","property_address":"456 Oak Avenue, Gotham, USA","house_type":"3b2b","house_size":64,"price":374,"availability":1,"host_id":"00000087"},
{"property_id":"0000000044","property_address":"789 Pine Road, Metropolis, USA","house_type":"studio","house_size":53,"price":480,"availability":0,"host_id":"00000089"},
{"property_id":"0000000046","property_address":"101 Elm Lane, Star City, USA","house_type":"1b1b","house_size":199,"price":417,"availability":1,"host_id":"00000091"}]
```

Enhance Microservices

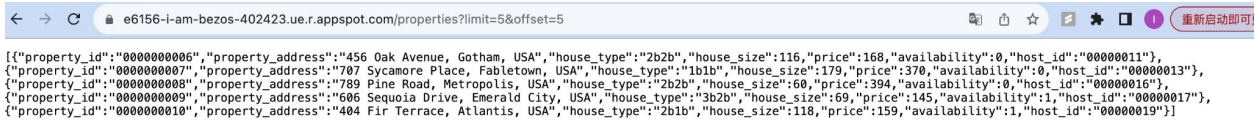
Querying (realized on Properties) (E.g. with front end, house_type = 1b1b)



Enhance Microservices

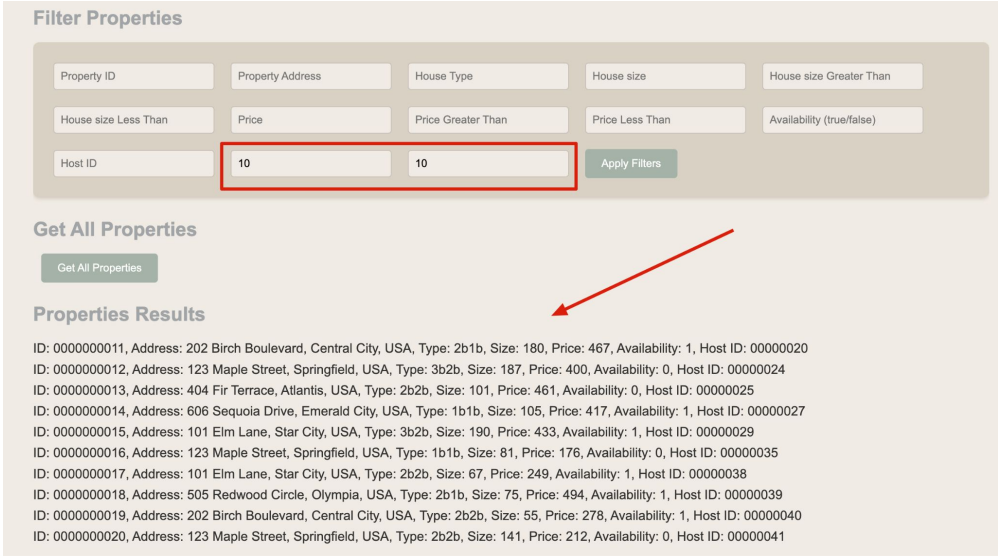
finished on all 3 microservices, using Users as example

(E.g. Limit = 5 & Offset = 5 -> we got IDs from 6 to 10, instead of from 1 to 50)



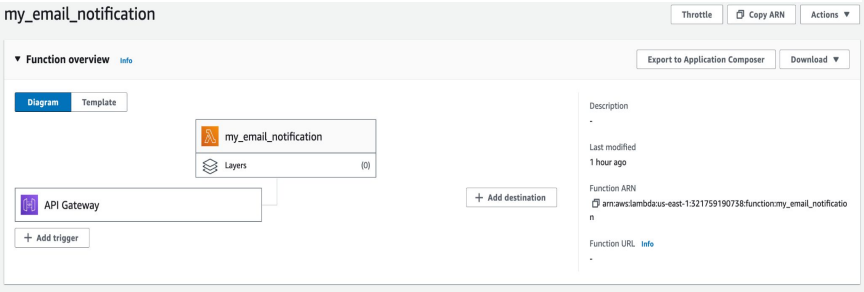
(E.g. with front end, Limit = 10 & Offset = 10 -> we got IDs from 11 to 20, instead of from 1 to 50)

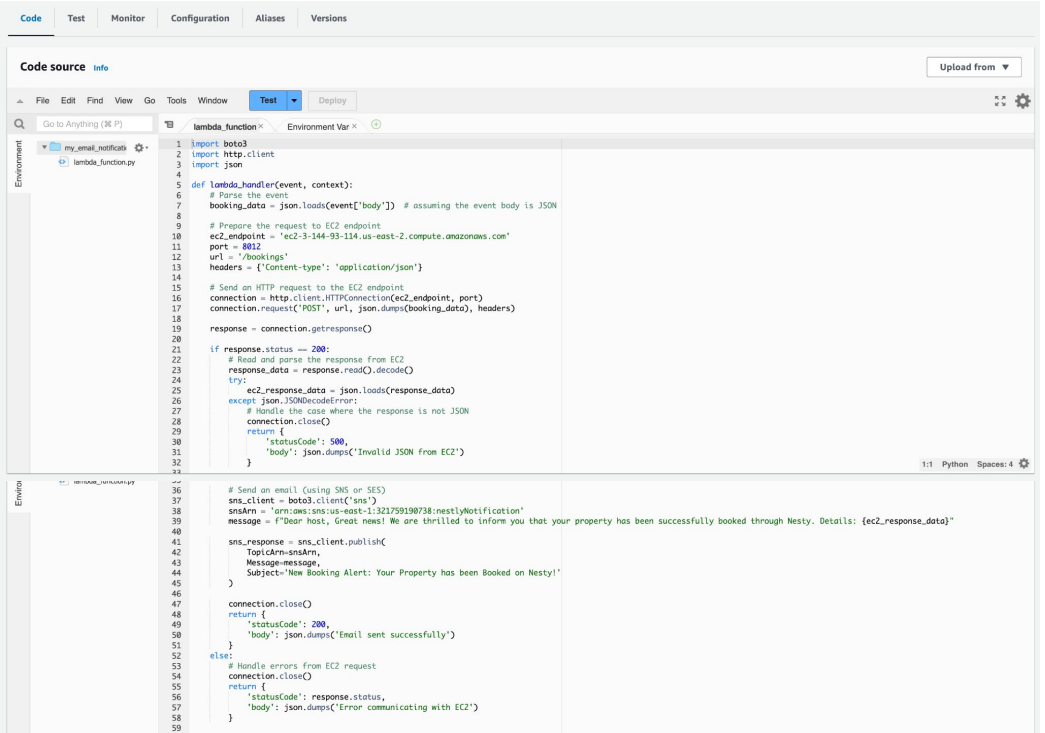
Pagination & offset (realized on Properties)



Events, Notifications, Pub/Sub

Events, Notifications, Pub/Sub





Composition/Aggregators

Composition/Aggregators

Sync: (designated order always)

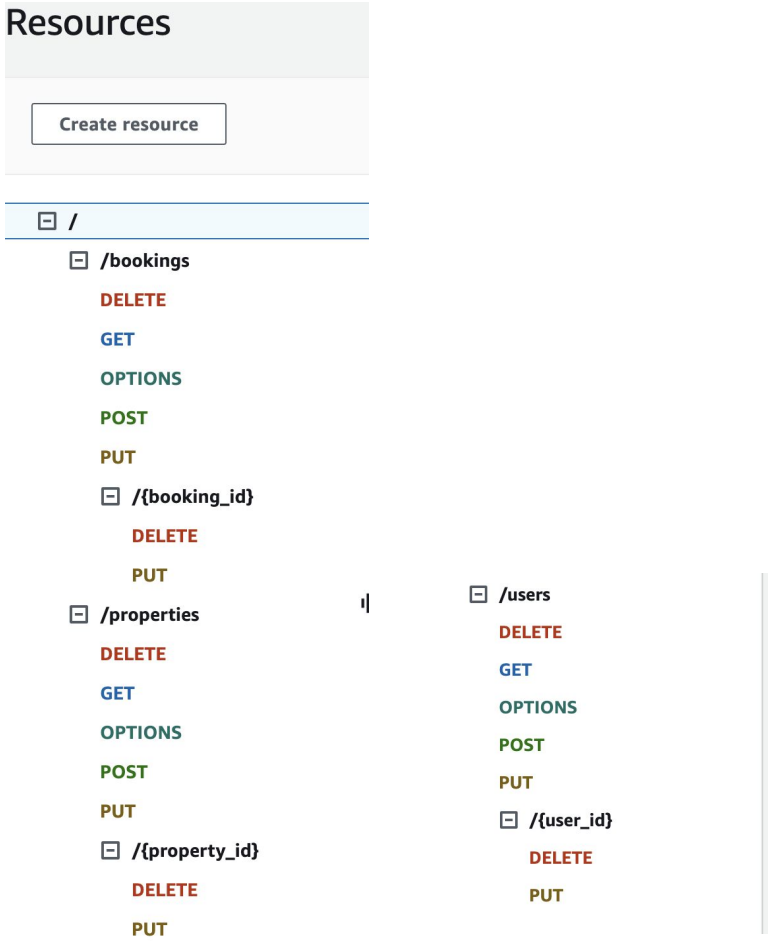
```
INFO:     Started server process [14318]
INFO:     Waiting for application startup.
INFO:     Application startup complete.
INFO:     Uvicorn running on http://0.0.0.0:8012 (Press CTRL+C to quit)
INFO:     127.0.0.1:52368 - "GET /docs HTTP/1.1" 200 OK
INFO:     127.0.0.1:52368 - "GET /openapi.json HTTP/1.1" 200 OK
users service has returned a response (in sync)
properties service has returned a response (in sync)
bookings service has returned a response (in sync)
INFO:     127.0.0.1:52369 - "GET /sync-aggregator HTTP/1.1" 200 OK
users service has returned a response (in sync)
properties service has returned a response (in sync)
bookings service has returned a response (in sync)
```

Async: different orders

```
INFO:     127.0.0.1:52408 - "GET /async-aggregator HTTP/1.1" 200 OK
users service has returned a response (in async)
properties service has returned a response (in async)
bookings service has returned a response (in async)
INFO:     127.0.0.1:52415 - "GET /async-aggregator HTTP/1.1" 200 OK
bookings service has returned a response (in async)
properties service has returned a response (in async)
users service has returned a response (in async)
INFO:     127.0.0.1:52420 - "GET /async-aggregator HTTP/1.1" 200 OK
bookings service has returned a response (in async)
properties service has returned a response (in async)
users service has returned a response (in async)
INFO:     127.0.0.1:52424 - "GET /async-aggregator HTTP/1.1" 200 OK
users service has returned a response (in async)
bookings service has returned a response (in async)
properties service has returned a response (in async)
INFO:     127.0.0.1:52428 - "GET /async-aggregator HTTP/1.1" 200 OK
bookings service has returned a response (in async)
users service has returned a response (in async)
properties service has returned a response (in async)
INFO:     127.0.0.1:52436 - "GET /async-aggregator HTTP/1.1" 200 OK
bookings service has returned a response (in async)
users service has returned a response (in async)
properties service has returned a response (in async)
INFO:     127.0.0.1:52440 - "GET /async-aggregator HTTP/1.1" 200 OK
users service has returned a response (in async)
bookings service has returned a response (in async)
properties service has returned a response (in async)
INFO:     127.0.0.1:52444 - "GET /async-aggregator HTTP/1.1" 200 OK
bookings service has returned a response (in async)
users service has returned a response (in async)
properties service has returned a response (in async)
INFO:     127.0.0.1:52449 - "GET /async-aggregator HTTP/1.1" 200 OK
properties service has returned a response (in async)
bookings service has returned a response (in async)
users service has returned a response (in async)
INFO:     127.0.0.1:52453 - "GET /async-aggregator HTTP/1.1" 200 OK
```

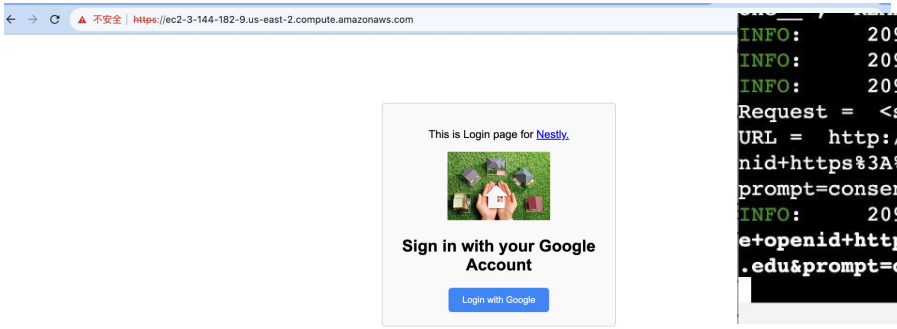
API Gateway

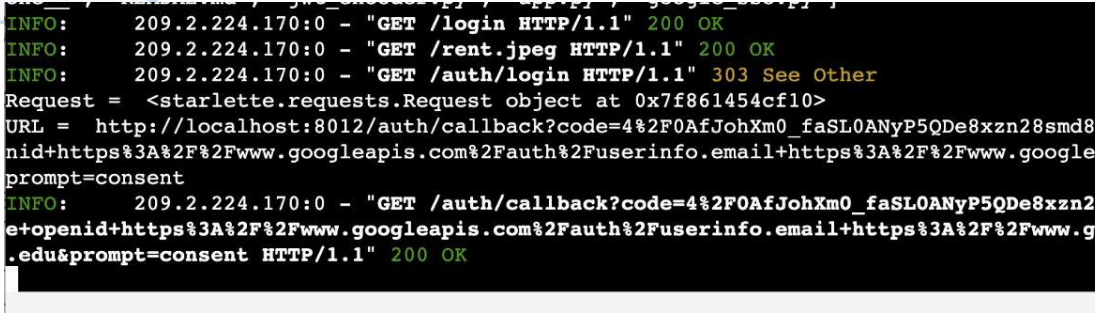
API Gateway

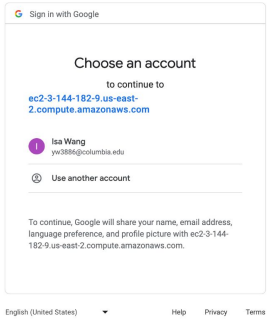

Resources
Create resource
/
/bookings
DELETE
GET
OPTIONS
POST
PUT
/{booking_id}
DELETE
PUT
/properties
DELETE
GET
OPTIONS
POST
PUT
/{property_id}
DELETE
PUT
/users
DELETE
GET
OPTIONS
POST
PUT
/{user_id}
DELETE
PUT

Single Sign-On (SSO)

Single Sign-On (SSO)

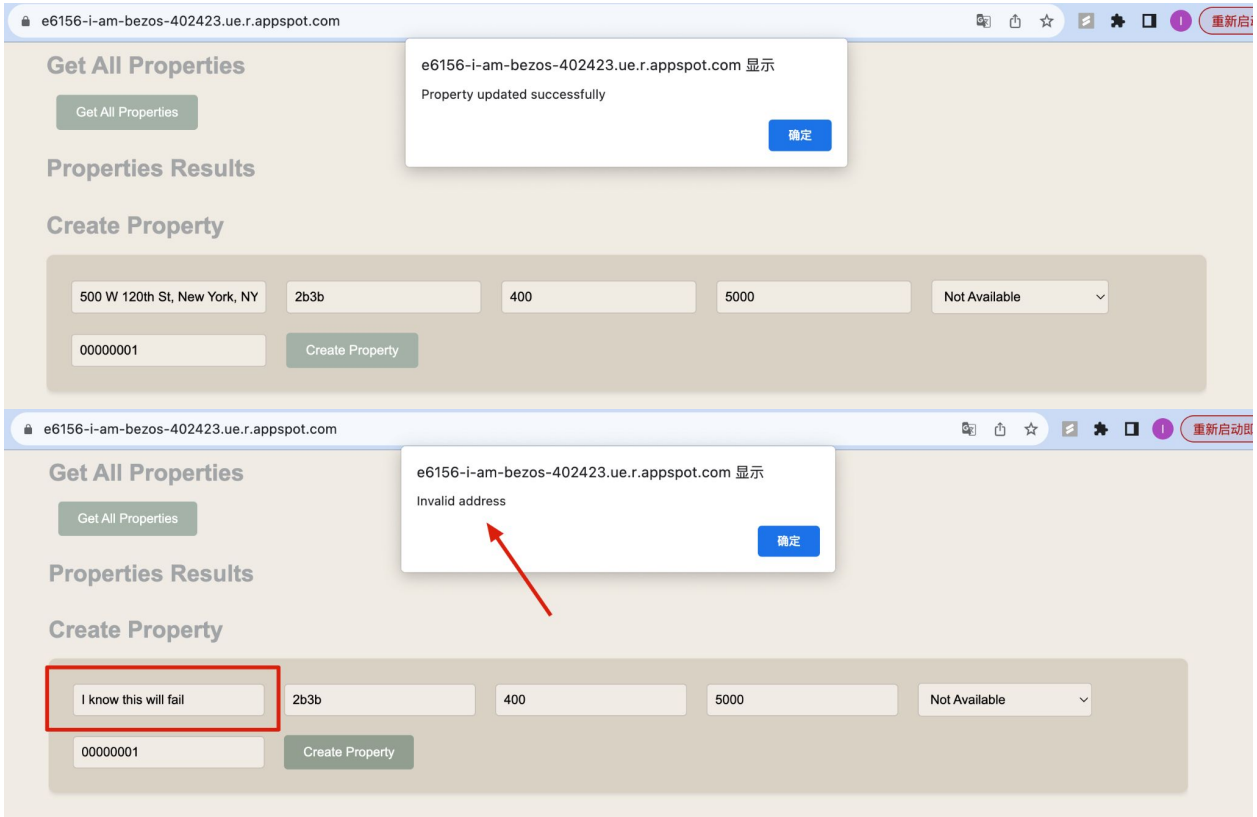






External API

External API



Middleware

Middleware

aws | Services | Search [Option+S]

```
From github.com:Runze-Lin/Micoservice-Bookings-and-Review
   82f7800..917fac4  main       -> origin/main
Updating 82f7800..917fac4
Fast-forward
 app.py                          | 2 +-
 static/user_booking_management.js | 5 +++--
 2 files changed, 4 insertions(+), 3 deletions(-)
[ec2-user@ip-172-31-10-119 Micoservice-Bookings-and-Review]$ python3 app.py
Database connected successfully!
INFO:     Started server process [905790]
INFO:     Waiting for application startup.
INFO:     Application startup complete.
INFO:     Uvicorn running on http://0.0.0.0:8012 (Press CTRL+C to quit)
Request: GET http://ec2-3-144-93-114.us-east-2.compute.amazonaws.com:8012/
Response: Status 307
INFO:     209.2.224.170:51657 - "GET / HTTP/1.1" 307 Temporary Redirect
Request: GET http://ec2-3-144-93-114.us-east-2.compute.amazonaws.com:8012/clientType
Response: Status 307
INFO:     209.2.224.170:51745 - "GET /clientType HTTP/1.1" 307 Temporary Redirect
Request: GET http://ec2-3-144-93-114.us-east-2.compute.amazonaws.com:8012/host
Response: Status 307
INFO:     209.2.224.170:51745 - "GET /host HTTP/1.1" 307 Temporary Redirect
Request: GET http://ec2-3-144-93-114.us-east-2.compute.amazonaws.com:8012/booking_management_host/00000001
Response: Status 307
INFO:     209.2.224.170:51745 - "GET /booking_management_host/00000001 HTTP/1.1" 307 Temporary Redirect
Request: GET http://ec2-3-144-93-114.us-east-2.compute.amazonaws.com:8012/bookings?host_id=00000001
Response: Status 200
INFO:     209.2.224.170:51767 - "GET /bookings?host_id=00000001 HTTP/1.1" 200 OK
```

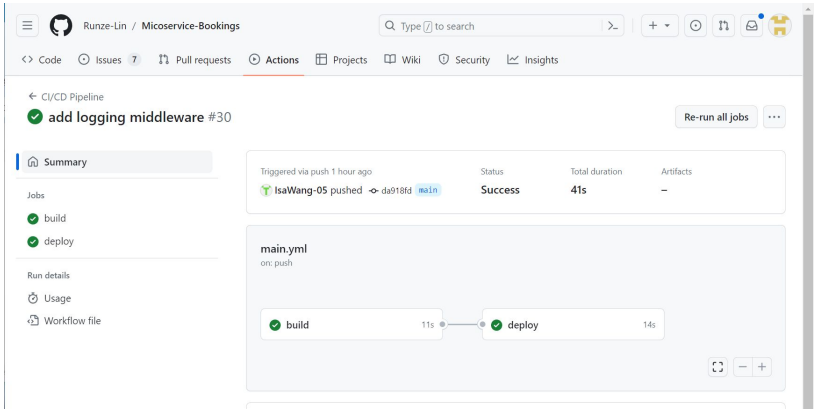
CI/CD

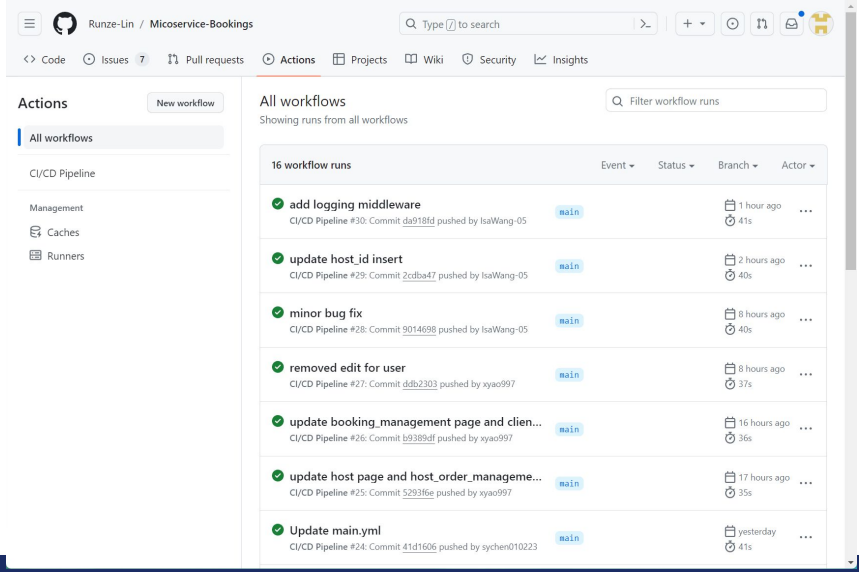
CI/CD

Jobs:

1. **Build Job:**
- **Environment:** It runs on the latest Ubuntu runner provided by GitHub Actions.
- **Steps:**
  - Checkout: actions/checkout@v2.
  - Python Setup: Sets up Python 3.8
  - Install Dependencies: Upgrades pip and installs pytest along with the packages specified in requirements.txt

2. **Deploy Job:**
- **Dependency:** It needs the build job to complete first.
- **Condition:** It only runs when a push event happens on the main branch.
- **Environment:** It also runs on the latest Ubuntu runner.
- **Steps:**
  - Checkout: Checks out the code.
  - Install Dependencies
  - SSH Key Setup
  - SSH Keyscan
  - Deploy





Infrastructure As Code

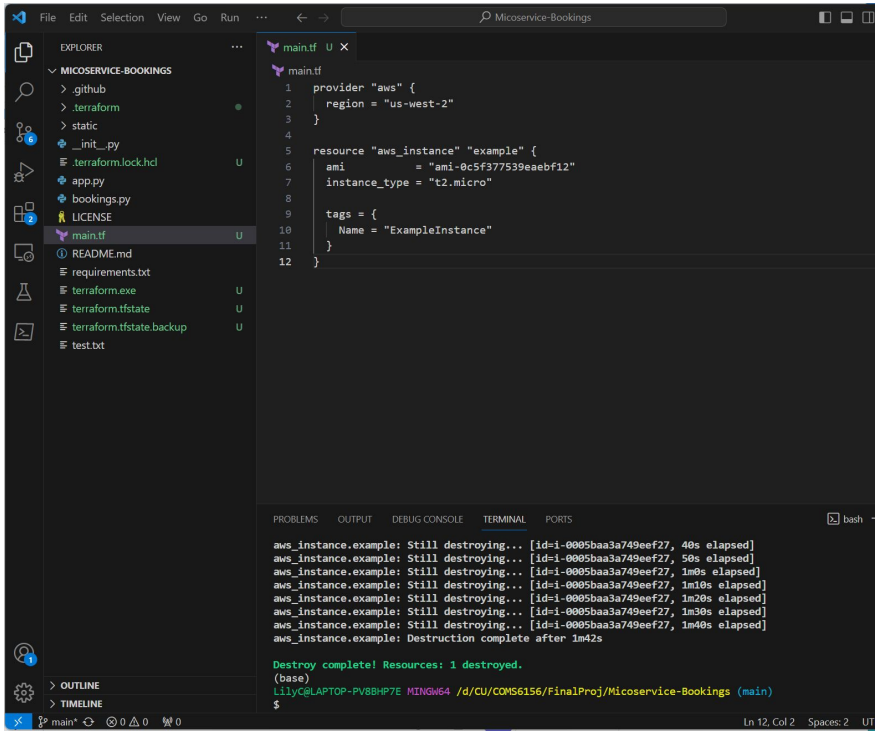
Infrastructure As Code

Use Terraform to create and initialize an EC2 instance in the terminal (./terraform apply), then destroy it in the terminal (./terraform destroy).

Create:

```
Plan: 1 to add, 0 to change, 0 to destroy.

Do you want to perform these actions?
  Terraform will perform the actions described above.
  Only 'yes' will be accepted to approve.

  Enter a value: yes

aws_instance.example: Creating...
aws_instance.example: Still creating... [10s elapsed]
aws_instance.example: Still creating... [20s elapsed]
aws_instance.example: Still creating... [30s elapsed]
aws_instance.example: Creation complete after 33s [id=i-05d80bde8aa335c24]

Apply complete! Resources: 1 added, 0 changed, 0 destroyed.
(base)
LilyC@LAPTOP-PV8BHP7E MINGW64 /d/CU/COMS6156/FinalProj/Micoservice-Bookings (
```

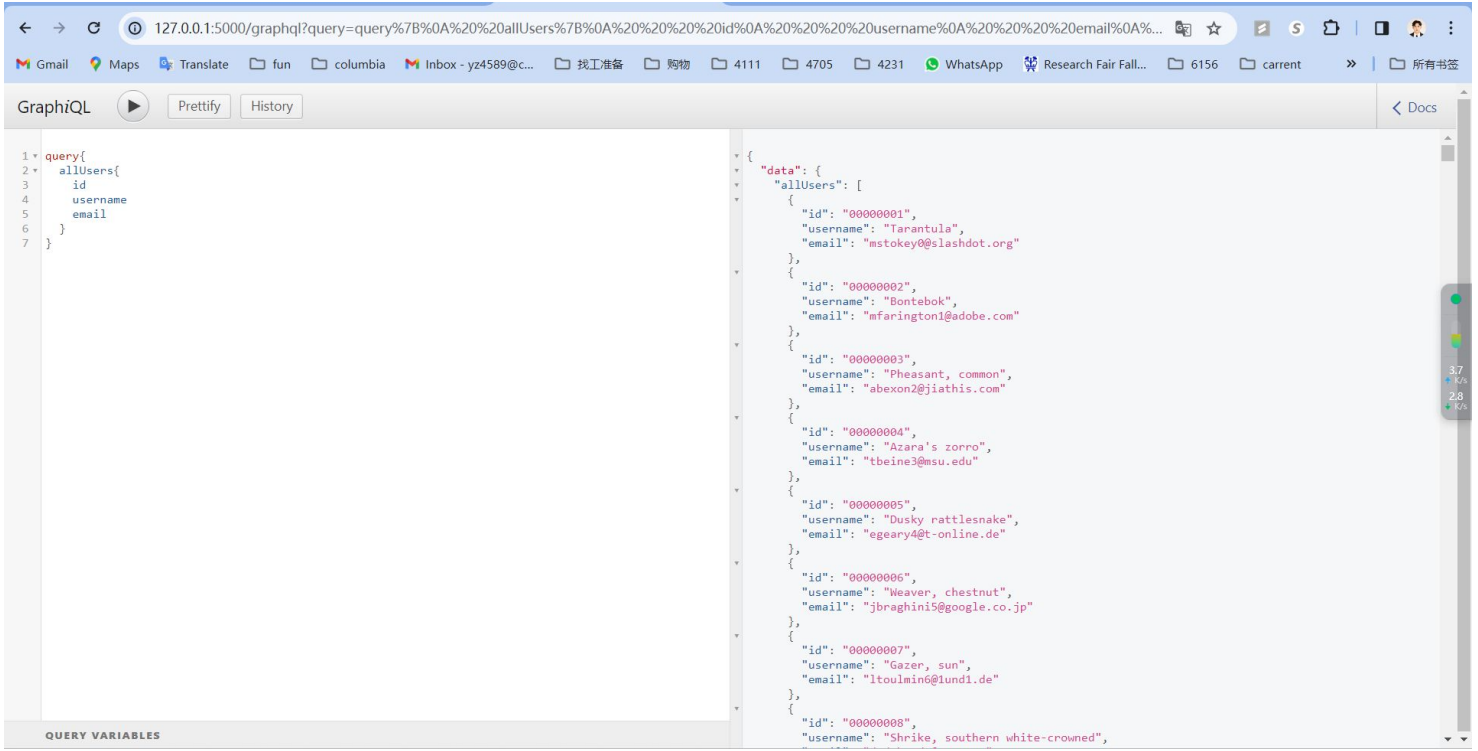
Destroy:



GraphQL

GraphQL

GraphQL to view the users table
Query the id, username, email.



Thank you for your listening!