



## **Programacion para el Internet**

Universidad Anáhuac Mayab.

Parcial 2

Ingeniería en Tecnologías de la Información y Negocios Digitales.

Profesor: Fabricio A. Suarez Dominguez.

Fecha de realización: 26 de Abril de 2025

Fecha de entrega: 28 de Abril de 2025

Estudiante:

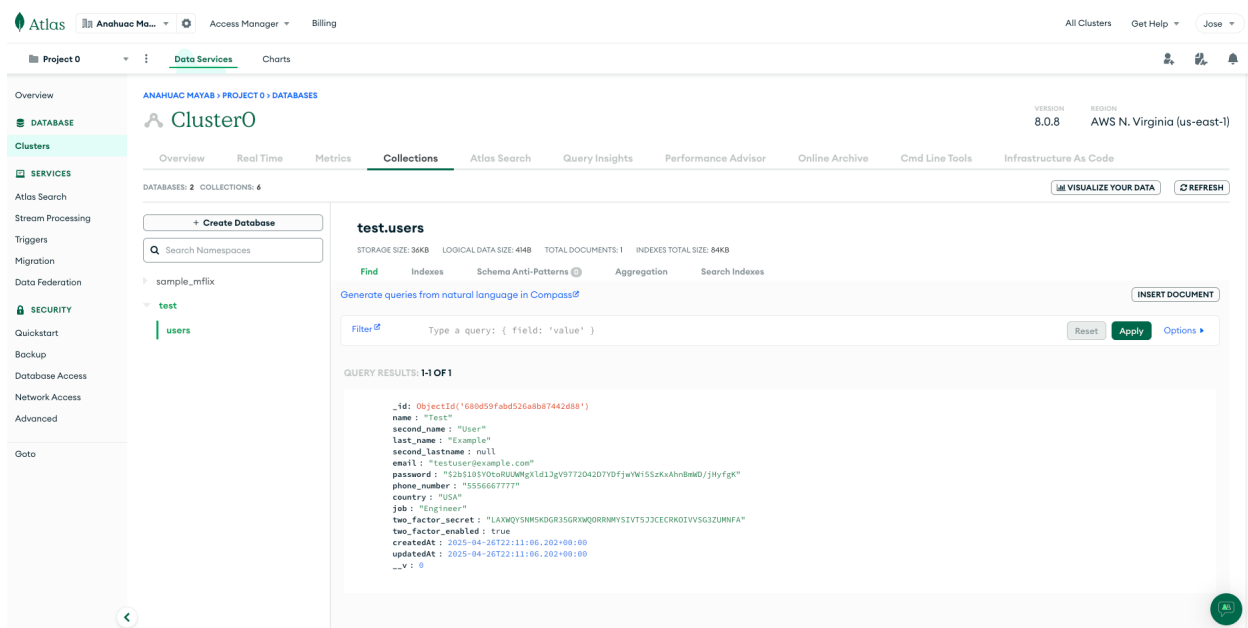
Jose Ricardo Pacheco Chanico

# Definicion del Objetivo

El objetivo de este proyecto, es crear un backend usando diferentes tecnologias como Node JS y MongoDB. Este backend tiene como idea implementarse mas adelante en un frontend como lo es el que se hizo para el primer parcial. En mi caso una website para informar y prevenir a los usuarios sobre los ataques de phishing, proporcionando ejemplo reales. Al iguales que consejos y videos para que los usuario aprendan sobre lo que es y como protegerse. El backend cuenta con la habilidad de registrar usuarios (les asigna un codigo qr para 2 factor authentication), autenticar usuarios cuando entran a su cuenta, actualizar usuarios y borrar usuarios. Al igual que contiene practicas de seguridad, como lo es el hashing, sal y pimienta, 2 factor authentication.

## Base de Datos MongoDB:

Se uso la instancia en la nube de Mongo Atlas. En esta instancia se creo un cluster en donde por el momento para las pruebas se uso la base de datos de test con la collection de users.



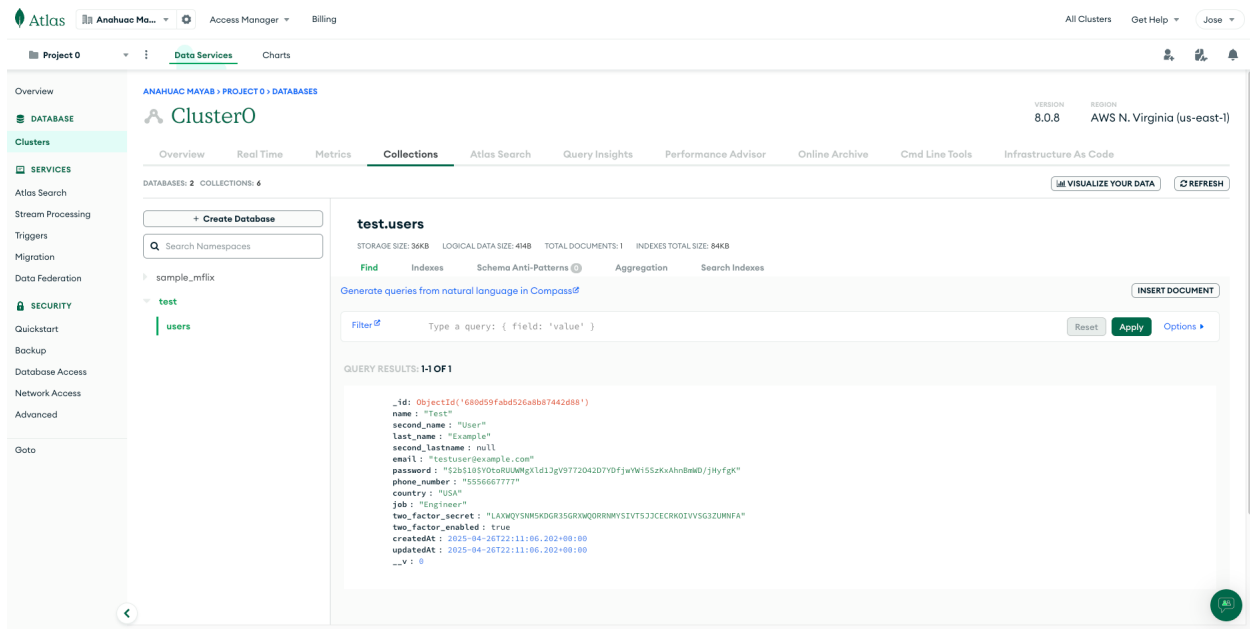
## Create User:

The screenshot displays the Burp Suite REST client interface. At the top, there's a navigation bar with "Overview", "POST https://phishing-backe...", and other tabs. Below it, the URL bar shows "https://phishing-backend-gp0j.onrender.com/api/auth/signup". The method dropdown is set to "POST". To the right are "Save" and "Share" buttons. Below the URL bar, there are tabs for "Params", "Authorization", "Headers (9)", "Body", "Scripts", and "Settings". Under the "Body" tab, there are radio buttons for different formats: "none", "form-data", "x-www-form-urlencoded", "raw" (selected), "binary", "GraphQL", "JSON", and "Beautify". The main area shows a raw JSON request body:

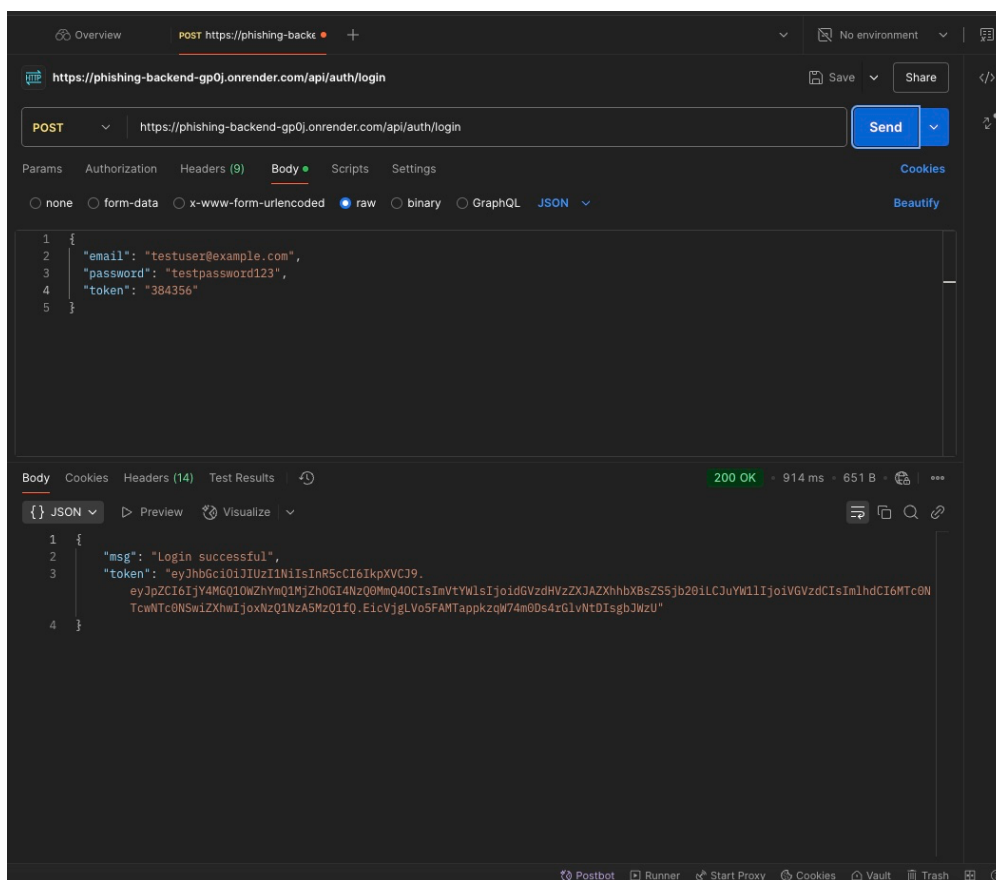
```
{
  "name": "Test",
  "second_name": "User",
  "last_name": "Example",
  "second_lastname": null,
  "email": "testuser@example.com",
  "password": "testpassword123",
  "phone_number": "5556667777",
  "country": "USA",
  "job": "Engineer"
}
```

At the bottom, there's a status bar indicating "201 Created", "2.14 s", and "2.21 KB". Below this, there are tabs for "Body", "Cookies", "Headers (14)", "Test Results", and icons for "Preview", "Visualize", and other tools. The "Body" tab is active, showing a JSON response:

```
{
  "msg": "User registered successfully",
  "qr": "data:image/png;base64,iVBORw0KGgoAAAANSUUhEgAAAMQAAADECAyAAAApo5rAAAAak1EQVR4AewaftIAAAj8SURBV03BQYolyZlAQdUg739lnWiWiqcgveyuytIvYha63/97DWOH7WwsIdwt4WGsdD2ut42GdtTystY6HtdbxsnY6HTZax8Na63hYaxOPa63jYa11Pkyljh8+pP13UVuqk8kbFpDJVTCPtaTyRsWk8kbFpDJV3KMFZPKtcwk8jdVfOJhrXU8rLWoh7XW8cOXVxyTyshqU8WklMlv8omJSomSlRuVqJWSmSpuKiaVm4qbik9UfJPKNz2stY6HtdbxsnY6fvhlKm9UVFHxCZWp4kZlqpqqJpWp4kbldZwp4jep3FSofJGxw96WGsdD2ut42GdtfzwH6fyomyomld9UMancVLxRMam8UTgp/C95WGsdD2ut42GdtfzwH6cxqUwVnyqTYl0xVUwbq6hMFw9UTCpTxY3KJ1Smiv8lD2ut42GdtTystY4fflnFP0nlEyo3FW+o3Kh8QmWmqCpuVN5QmSreqPg3eVhzHQ9rrreNhrXX88GUqf5PKVDGpTBWTylQxqUwVvk8pUmalMFZPKVDGpvExqUwVvk8pUmalMFZPKGY/Zg9rrreNhrXU8rLWohZ5U8w9W8U0qNyrvLYh8ptU3qi4gfveVhzHQ9rrreNhrXX88GVqJWSumYVN6ouFF5Q2WqmFR+k8pUCaNyu3FTMaLMFZPKVHGjmlXCqEWvK8pNsce1lrHwlrrreFhzHT98q0ITK1PFJ1RuKiaVqWJSomSsq+SWWquFGZKj6hMLVMkm+o3Ki8oFi3Payljoe1lvGwi1rsD/4ilW+qmFSmklllpqhUflPFJ1j1+fUfnKM9UTC03FZPKVPND2ut42GdtTystY4ffpnKV0GpTBXfpDJVTcrfVDGp3KhMFZPKVHGjmlV8QmWmqCpuV64q3gj4TQ9rrreNhrXU8rLWOH/4ylRuVmd4p35aZiUpkvknljYo3VD6h8kbFpDJVVKEYVUwnq6j4xMna63hYaxOPa63D/uAfipDJV3KjVCeWqn6iyVG64g/kkqb1RMMm9UTCpTxRsqNxwTYlTXiYe1lvGwi1joellqH/cEHVKasKWqeENlqpUbloomlalUimpmFRuKT50mSreUlmmpFSmiknbq64Ubmpp+MTDWUt4WGsdD2ut44cPVUqnypTxaQyVUwU8WkMQm8UTGp3FRMKjcqb618m1V8QmVSmsqmiknmx7WmsfDmwUt4NGsdP3xiZaaYVD6h8omSekWNijcqvgliUvmEyhsVno3FW9U3Kj8poe1lvGwi1joellrHD1+m8omKb1KZKT5QuamYVG4q3pWpY1K5qbhR+YTKVPEJlanIRmwq+E0Pa63jYa11Pkyljh8+VPEJlU9U3FRMKm9UTCpVnXUTCqfUjkb1RuKmd4q3pU3VG4q/qahTdbxsnY6HTZah/3BB1RuKiaVNYomlx9Sxd+kc1PxhpUman8TRU3KLPfDjVfOJhrXU8rLWOH7XW8cOHkiaVm4pJZaaYVG4qPgeYXYTyk3FtcUbK1PFVDGpTBWTyk3FjcpUman8mzyStY6HtdbxsnY
```



**Login User:**



# Update User

The screenshot displays a REST client interface with a PUT request to `https://phishing-backend-gp0j.onrender.com/api/auth/update/680d59fabd526a8b87442d88`. The request body is a JSON object with `job` and `country` fields. The response is a 200 OK status with a JSON body containing a success message and detailed user information.

**Request:**

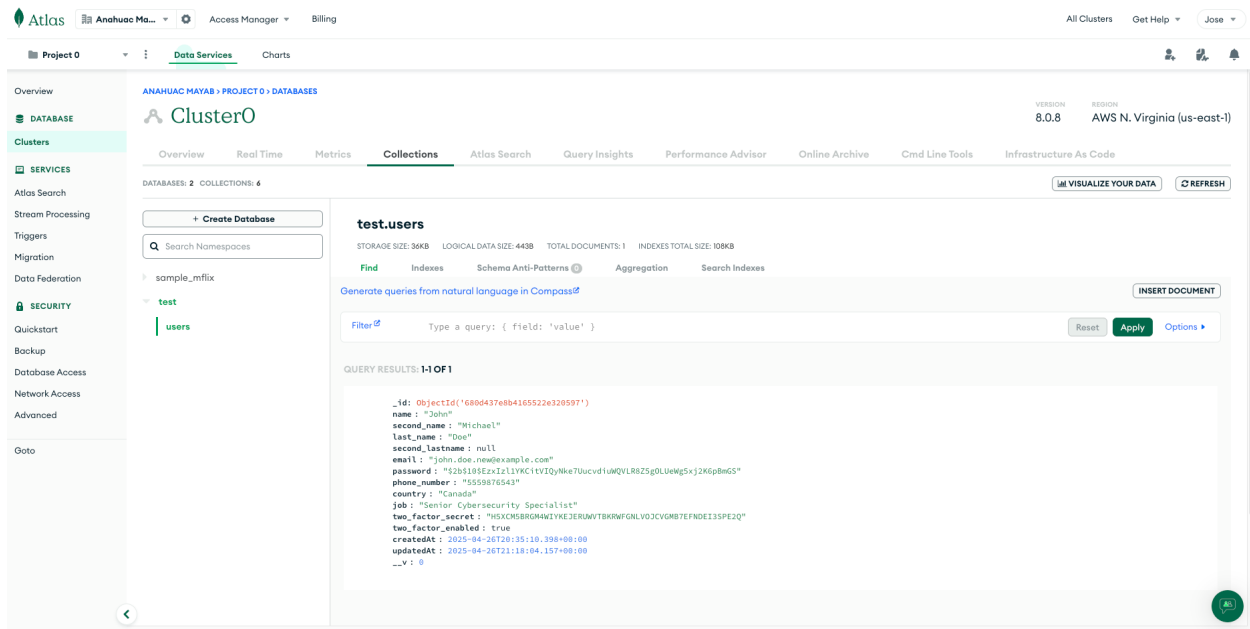
```
PUT https://phishing-backend-gp0j.onrender.com/api/auth/update/680d59fabd526a8b87442d88
```

Body (JSON):

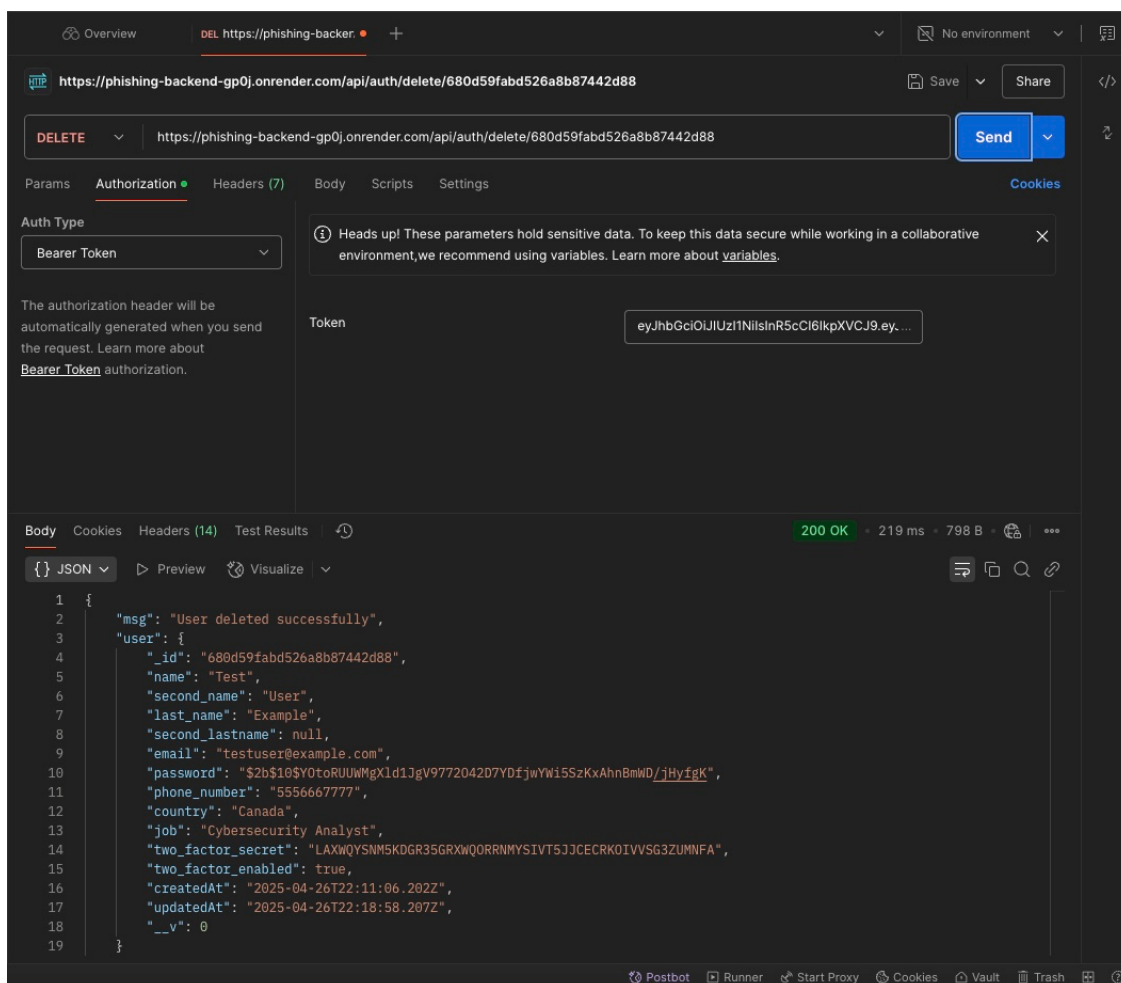
```
1 {
2   "job": "Cybersecurity Analyst",
3   "country": "Canada"
4 }
```

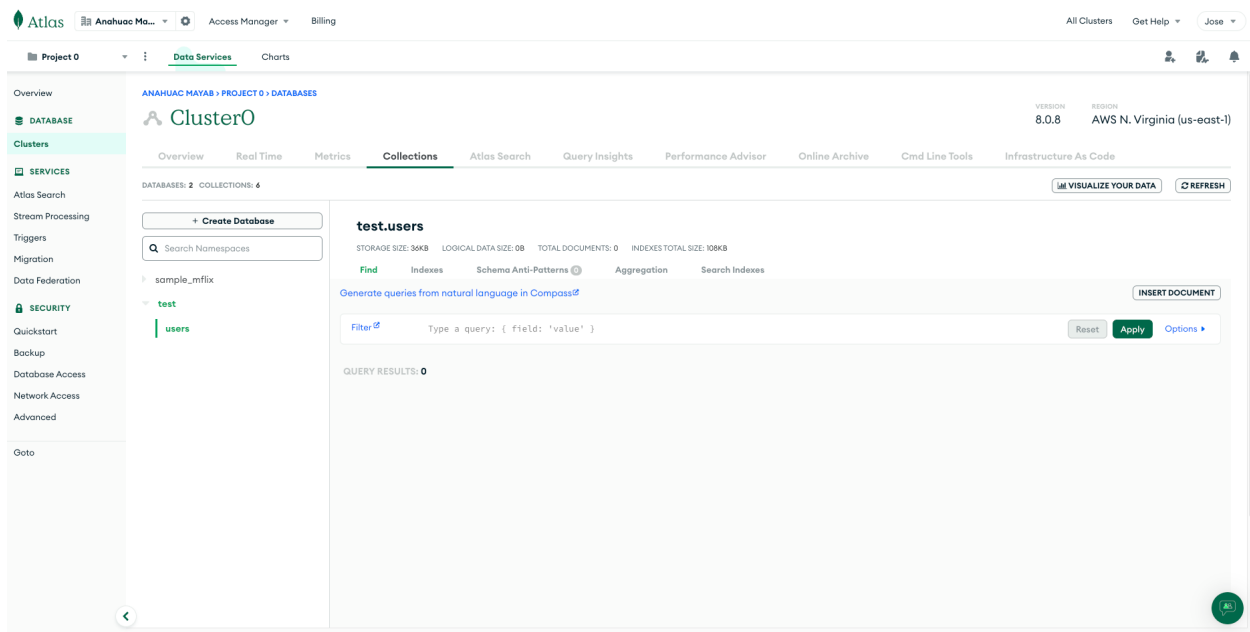
**Response:**

```
1 {
2   "msg": "User updated successfully",
3   "user": {
4     "_id": "680d59fabd526a8b87442d88",
5     "name": "Test",
6     "second_name": "User",
7     "last_name": "Example",
8     "second_lastname": null,
9     "email": "testuser@example.com",
10    "password": "$2b$10$Y0toRUUWMgXld1JgV977204207YDfjwYWi5SzkXAhnBmWD/jHyfgK",
11    "phone_number": "5556667777",
12    "country": "Canada",
13    "job": "Cybersecurity Analyst",
14    "two_factor_secret": "LAXWQYSNM5KDGR35GRXWQORRMYSIVT5JJCECRK0IVVSG3ZUMNFA",
15    "two_factor_enabled": true,
16    "createdAt": "2025-04-26T22:11:06.202Z",
17    "updatedAt": "2025-04-26T22:18:58.207Z",
18    "__v": 0
19  }
20 }
```



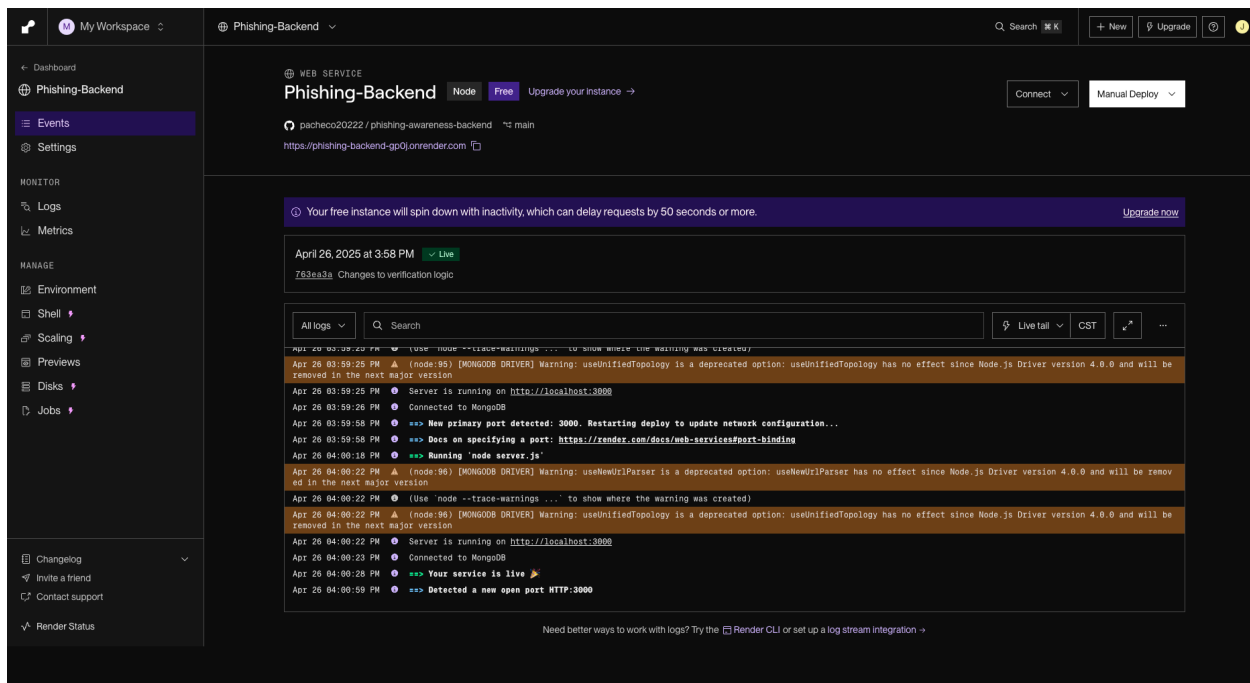
## Delete user:





## Deployment en render.com:

El backend fue deployed usando el free tier de la website de render.com. Las pruebas de postman se hicieron ya con el backend siendo deployed, al igual que cuando solo era local, las fotos si solo son de cuando el backend estaba deployed.



## Repositorio de Github y Conclusiones:

En conclusion este es un backend con un CRUD y APIs listas para el uso de creacion de usuario, login, actualizacion de usuarios, y borrar usuarios. Al igual que esta deployed en render y listo para ser aplicada a un frontend mas adelante.

Github: <https://github.com/pacheco20222/phishing-awareness-backend>