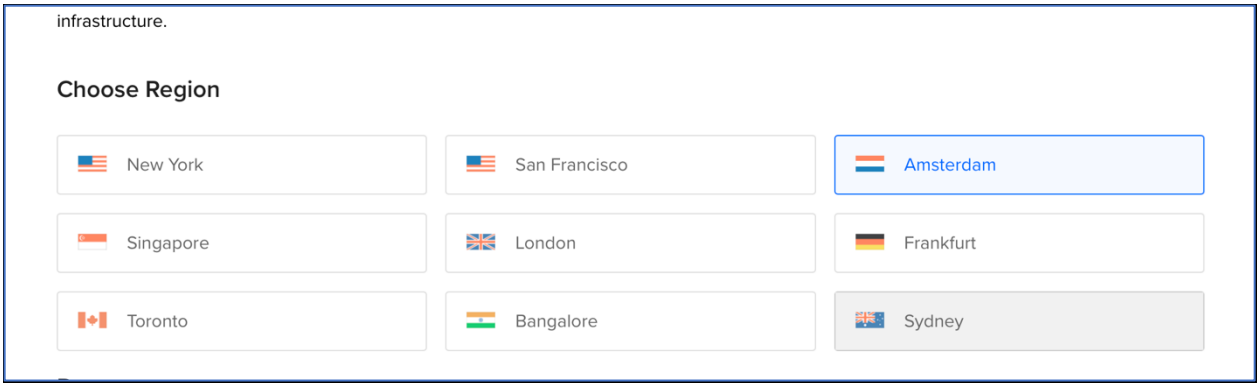
C4G BLIS V3.8 DEPLOYMENT GUIDE

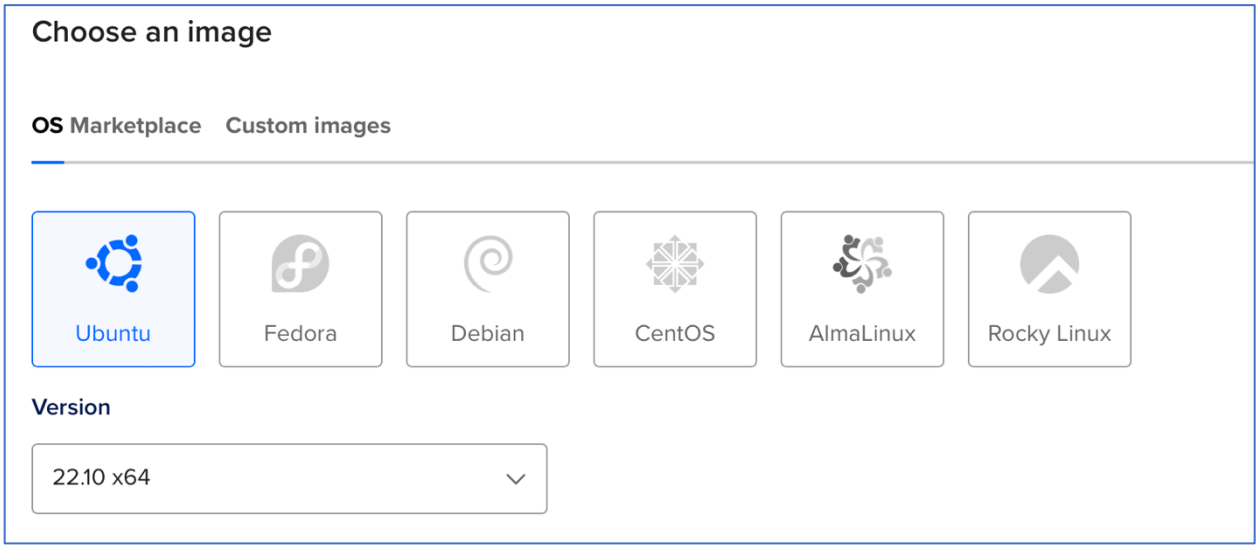
This document details the steps for deployment of C4G BLIS V3.8 in the digital Ocean account.

Assumption: The user has a digital ocean account with a project already set up. A domain is registered for the application as well.

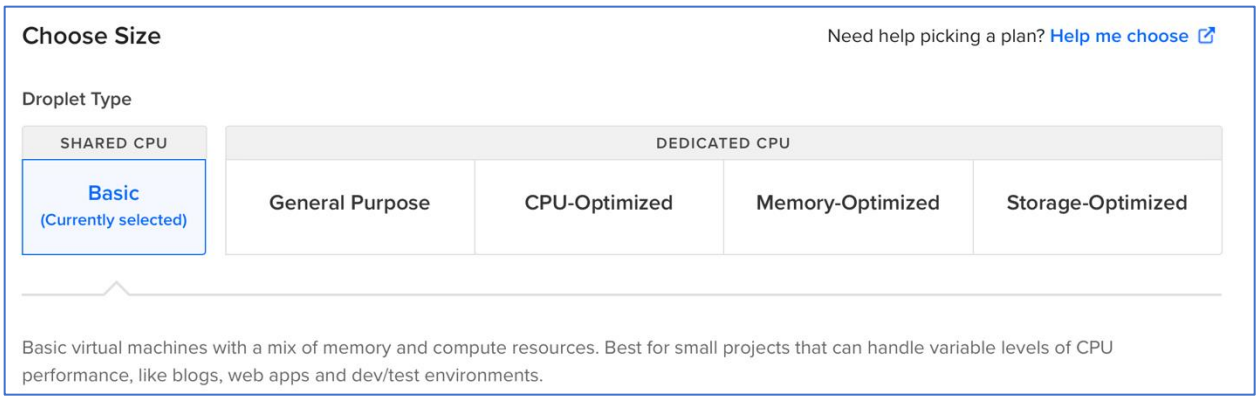
STEP 1 : Chose a region (preferably closer to the country location)



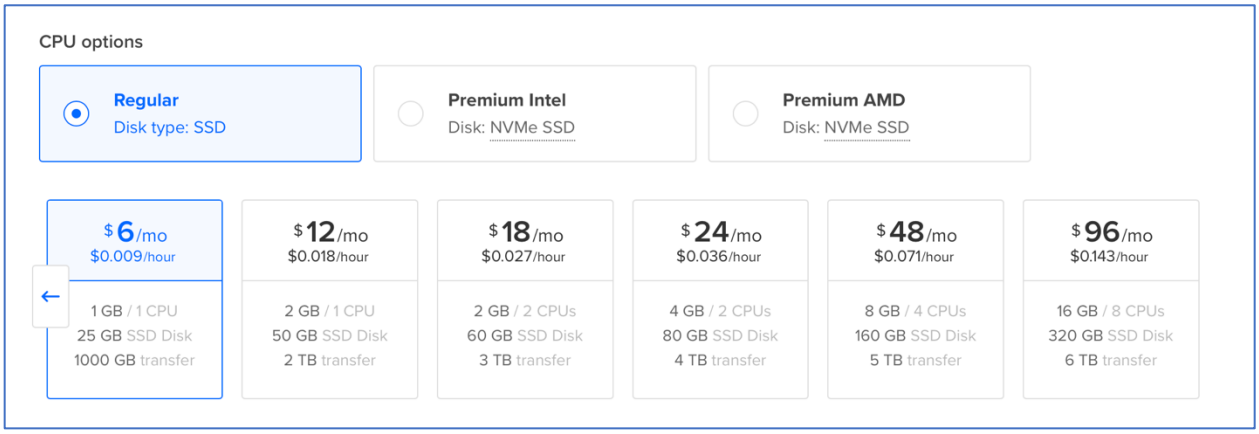
STEP 2 : Choose an Image (Select Ubuntu)



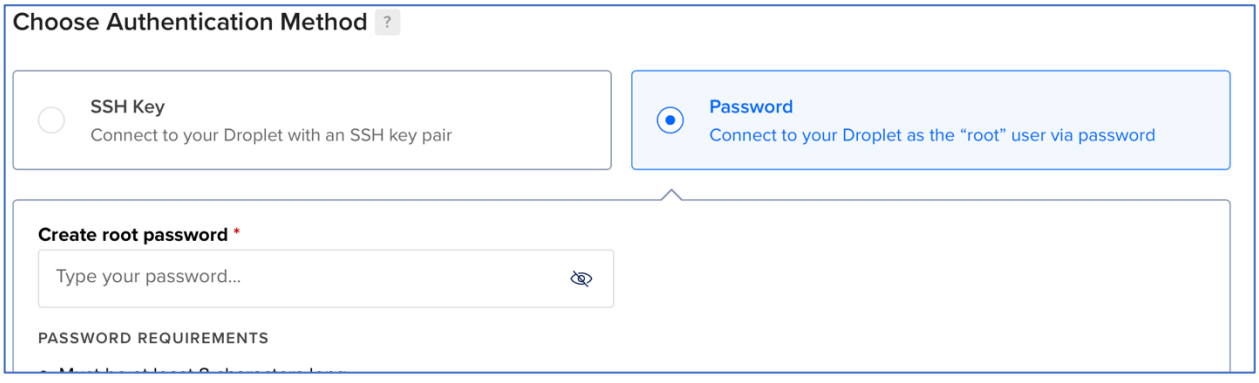
STEP 3: Choose droplet type (Select Basic)



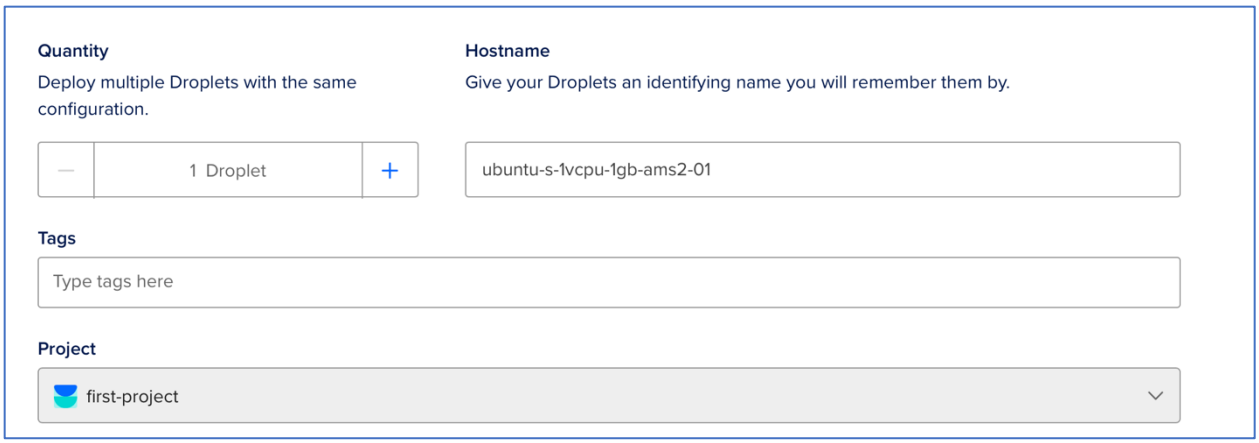
STEP 4: Choose CPU option



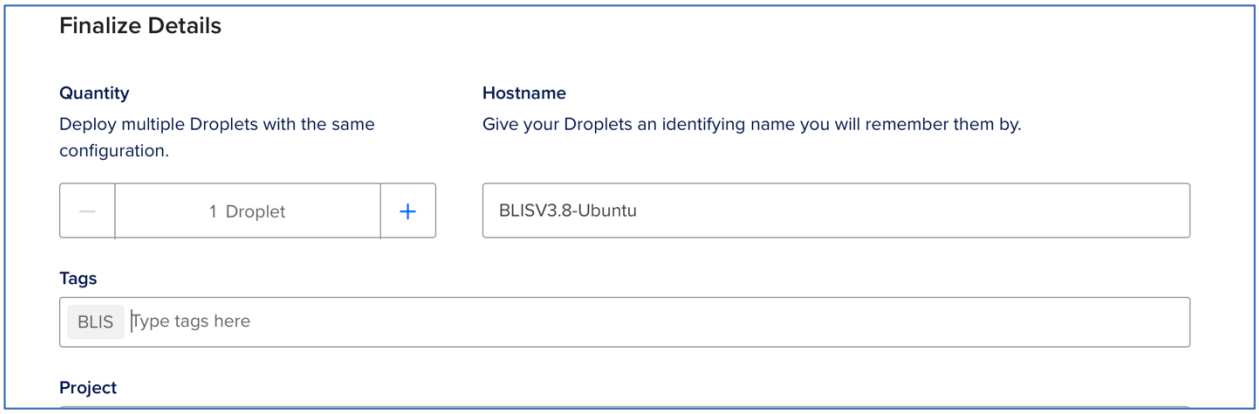
STEP 5: Choose the Method to access the console



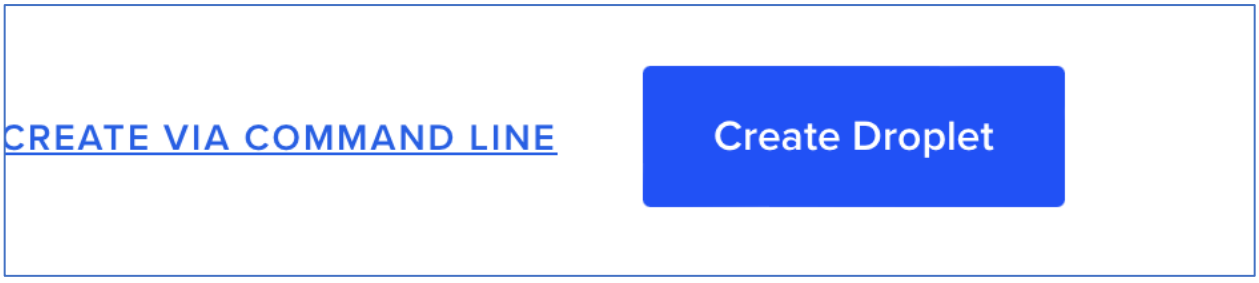
STEP 6: Review the finalize details



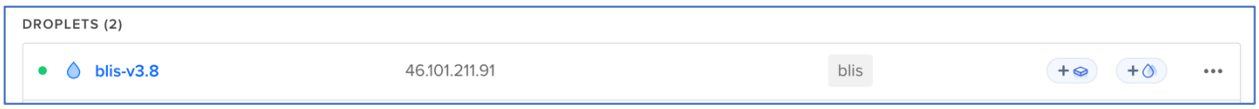
STEP 7: Good practice to add tag as shown below



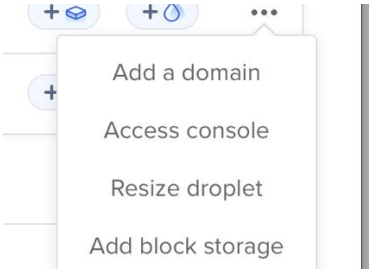
STEP 8: Create Droplet by pressing the blue button at the end



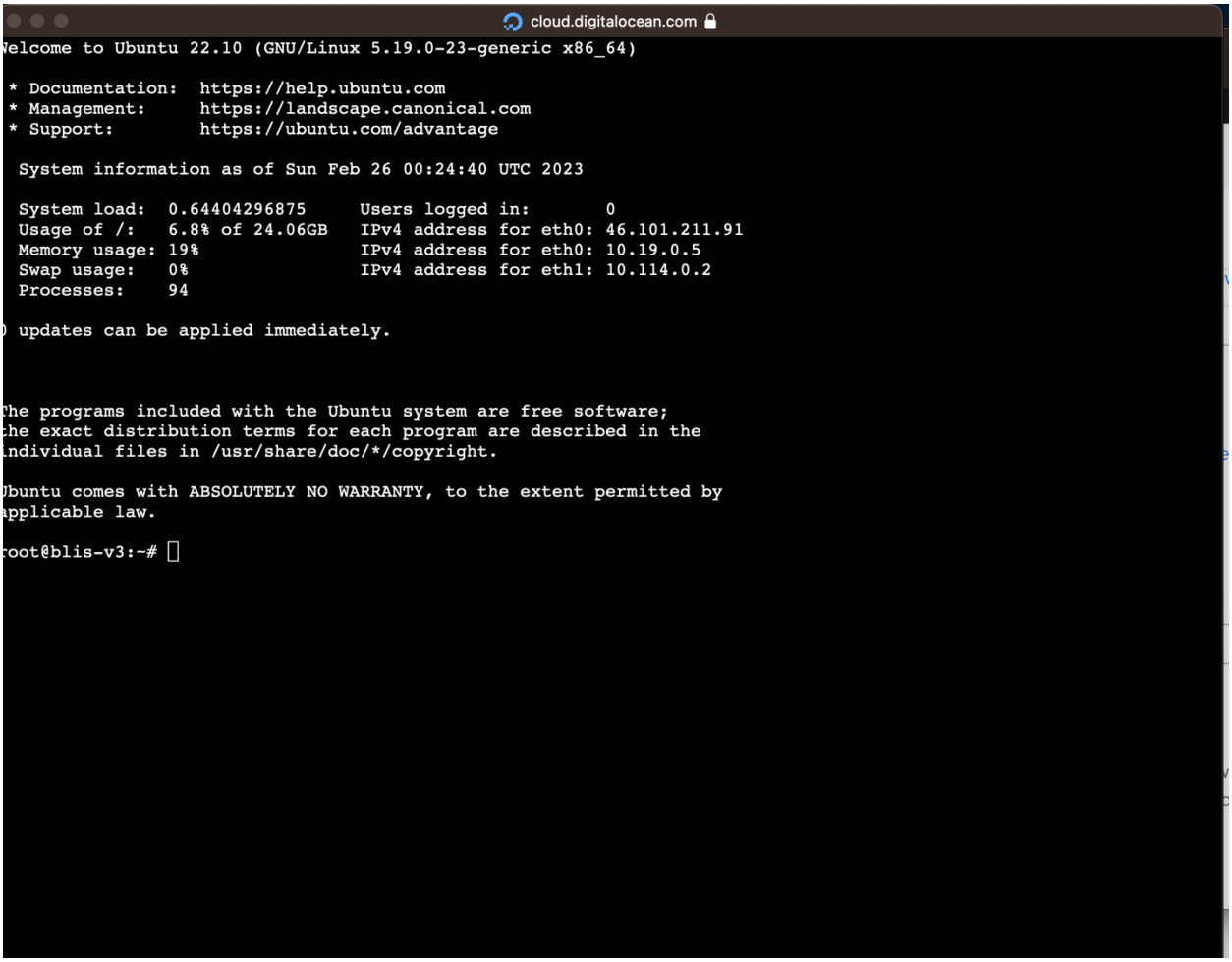
STEP 8: Droplet creation might take a few minutes. You should be able to see the provisoned droplet in the home page under Droplets like shown below



STEP 9: Review the menu by clicking the dots at the end of the droplet. Chose Access Console and login with root and the password you selected. Or use the SSH method to login



STEP 10: Console view



STEP 11: next steps are to set up the Linux machine. Run the following command:

```
curl https://raw.githubusercontent.com/C4G/BLIS/master/docker/bootstrap.sh |
bash
```

STEP 12: Run the commands for update and installing docker compose. **Please check if Docker compose is already set up. If it is set up you can skip this step.**

```
sudo curl -L
"https://github.com/docker/compose/releases/download/1.29.2/docker-compose-
$(uname -s)-$(uname -m)" -o /usr/local/bin/docker-compose
```

```
root@blis-v3:~# sudo curl -L "https://github.com/docker/compose/releases/download/1.29.2/docker-c
s)-$(uname -m)" -o /usr/local/bin/docker-compose
  % Total    % Received % Xferd  Average Speed   Time    Time     Time  Current
                                 Dload  Upload   Total   Spent    Left  Speed
  0     0    0     0    0     0      0      0 --:--:-- --:--:-- --:--:--     0
100 12.1M  100 12.1M    0     0  47.8M      0 --:--:-- --:--:-- --:--:-- 47.8M
root@blis-v3:~#
```

STEP 13: Provide permission to run docker compose.

```
chmod +x /usr/local/bin/docker-compose
```

STEP 14: Now we are ready to install BLIS solution. Run command below to pull down the git release version

---

```
git clone https://github.com/C4G/BLIS.git
```

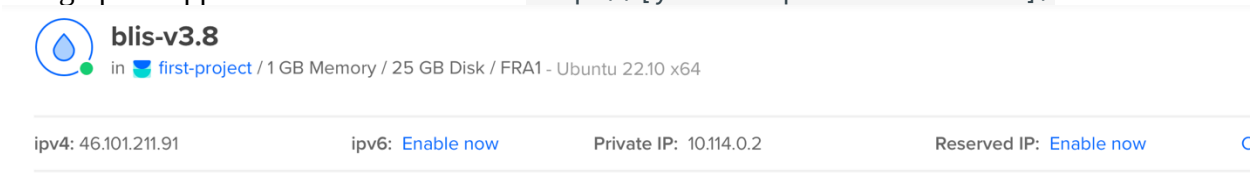
```
root@blis-v3:~# git clone https://github.com/C4G/BLIS.git
Cloning into 'BLIS'...
remote: Enumerating objects: 12731, done.
remote: Counting objects: 100% (840/840), done.
remote: Compressing objects: 100% (429/429), done.
remote: Total 12731 (delta 411), reused 757 (delta 385), pack-reused 11891
Receiving objects: 100% (12731/12731), 116.42 MiB | 17.58 MiB/s, done.
Resolving deltas: 100% (5865/5865), done.
root@blis-v3:~#
```

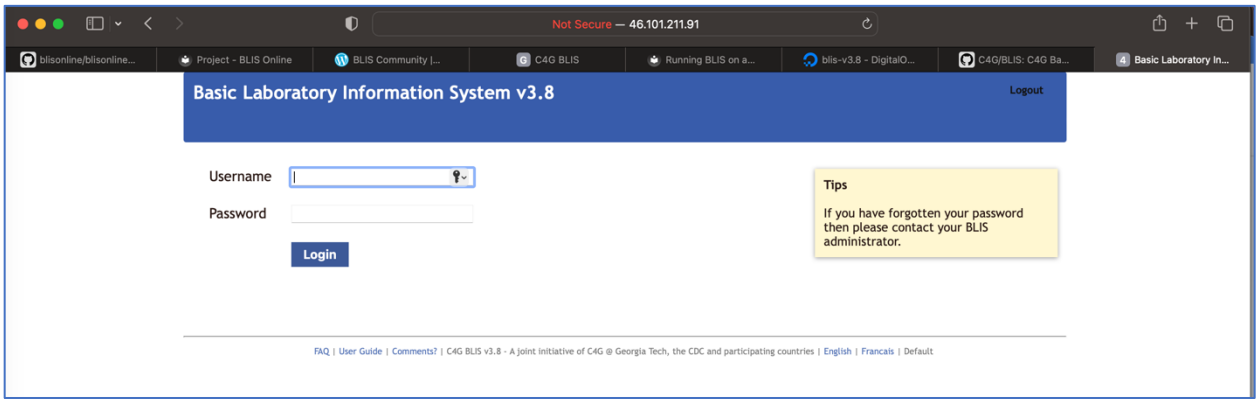
STEP 15: change directory to docker folder and run docker compose to bring the containers up

```
cd BLIS/docker
docker-compose up -d
```

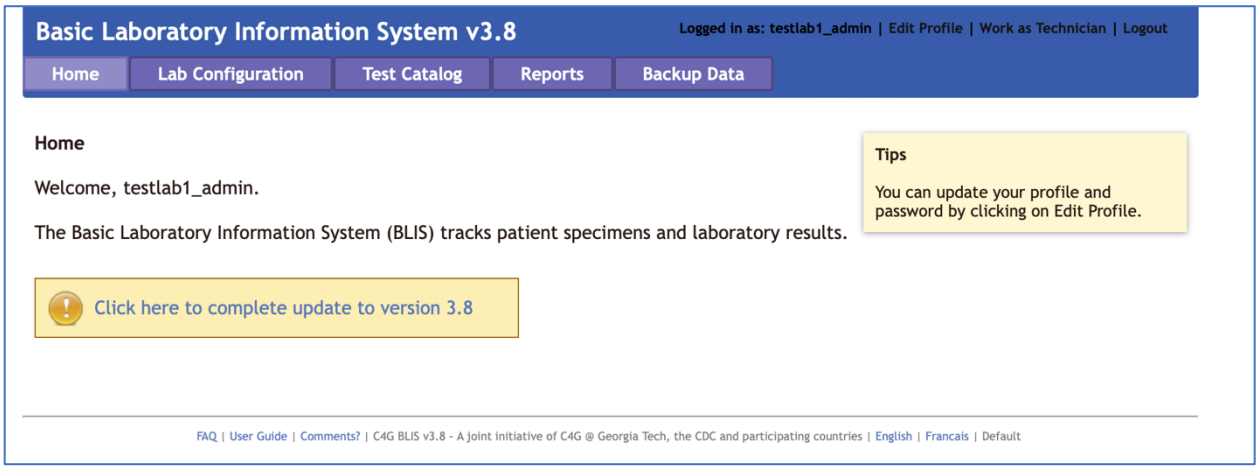
```
root@blis-v3:~/BLIS/docker# docker-compose up -d
Creating network "docker_default" with the default driver
Creating volume "docker_blis-data" with default driver
Creating volume "docker_blis-files" with default driver
Creating volume "docker_blis-apache2" with default driver
Creating volume "docker_blis-letsencrypt" with default driver
Pulling db (mysql:5.7)...
5.7: Pulling from library/mysql
e048d0a38742: Pull complete
c7847c8a41cb: Pull complete
351a550f260d: Pull complete
8ce196d9d34f: Pull complete
17febb6f2030: Pull complete
d4e426841fb4: Pull complete
fda41038b9f8: Pull complete
f47aac56b41b: Pull complete
a4a90c369737: Pull complete
97091252395b: Pull complete
84fac29d61e9: Pull complete
Digest: sha256:8cf035b14977b26f4a47d98e85949a7dd35e641f88fc24aa4b466b36beecf9d6
Status: Downloaded newer image for mysql:5.7
Pulling app (ghcr.io/c4g/blis:latest)...
latest: Pulling from c4g/blis
bdec8b63443a: Pull complete
26372d544197: Pull complete
5bfd1a80ce62: Pull complete
357f2a7fa360: Pull complete
b29324bfa7b7: Pull complete
5ba5441dca37: Pull complete
0b1fc1415be8: Pull complete
c1d491f3b904: Pull complete
bd15de8d8797: Pull complete
b664430e5920: Pull complete
de287c25c4d9: Pull complete
02646b0fa301: Pull complete
2815e62f249a: Pull complete
Digest: sha256:145d94fe78d0498b2ca1e5edff3c1411932b547e01272333b8479c991a0e2281
Status: Downloaded newer image for ghcr.io/c4g/blis:latest
Creating docker_db_1 ... done
Creating docker_app_1 ... done
root@blis-v3:~/BLIS/docker#
```

STEP 16: BLIS should be up. Identify the IP address from the console like the one below Bring up the application on the browser. `http://[your droplet IP address]/`

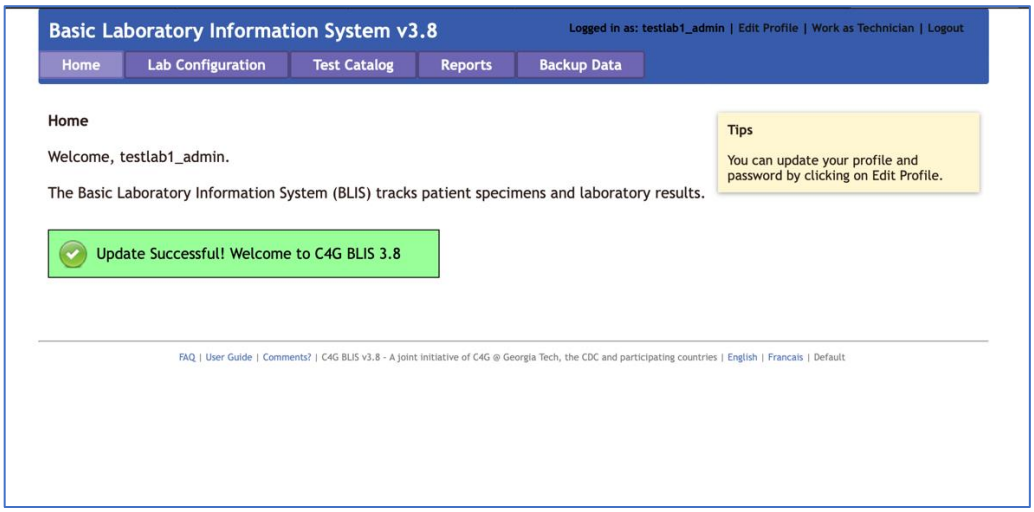




STEP 17: Please login using the credentials: testlab1_admin/admin123 (as admin) or testlab1_tech1, testlab1_tech2/tech123 (as lab technician).



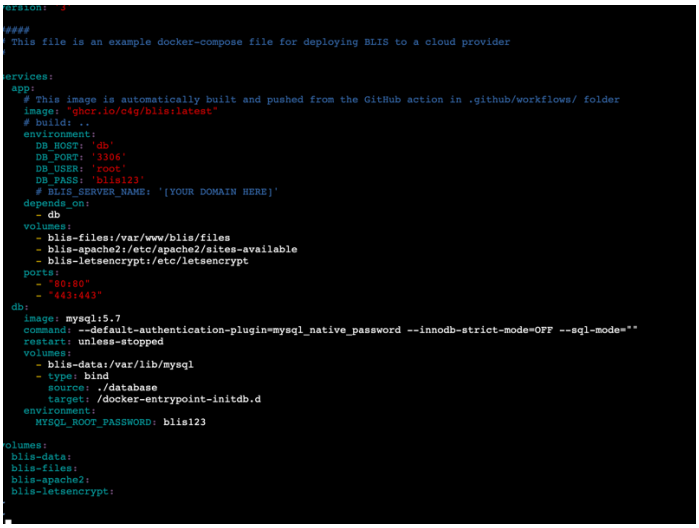
STEP 18: Once logged in, you should see the ‘click here to complete update to version 3.8” button on the screen. Please click to update to the latest version.



STEP 19: Update to https for the server URL by adding the certificate.
For this step, there should be a **domain registered.** Use the following link to attach the domain to the dropliet IP address. https://docs.digitalocean.com/products/networking/dns/quickstart/

After your domain is pointing to your BLIS host IP address

You will need to add the BLIS_SERVER_NAME to the **docker-compose.yml** configuration: Open the file in the cd BLIS/docker file and update the variable after removing the comment.



STEP 20: Then, (re)start BLIS:

# if BLIS is running
$ docker-compose down

# bring the database container up first and daemonize it
$ docker-compose up -d db

# bring the app container up alone, syncronously, so we can see the output
$ docker-compose up app

Make sure there are no errors in the output. The container will attempt to read the value of BLIS_SERVER_NAME and set the appropriate ServerName directive in the Apache2 web server configuration and a message will say that it is successful.

Assuming it is successful, you can quit with Ctrl-C and restart as a background process (docker-compose up -d app).

STEP 21:

In a separate terminal window, while BLIS is running, run the script:

docker-compose exec app get-https-cert.sh

This will verify the environment configuration seems correct and execute the certificate tool for you! Answer the questions about the domain to the best of your knowledge.

Once the domain is verified and the certificate installed, you can visit your BLIS instance with an https:// URL and hopefully it just works!

This completes the installation. Thank you!!!

SMOKE TEST
(LOGIN TO THE APP AND RUN THE FOLLOWING TEST FOR VALIDATION)

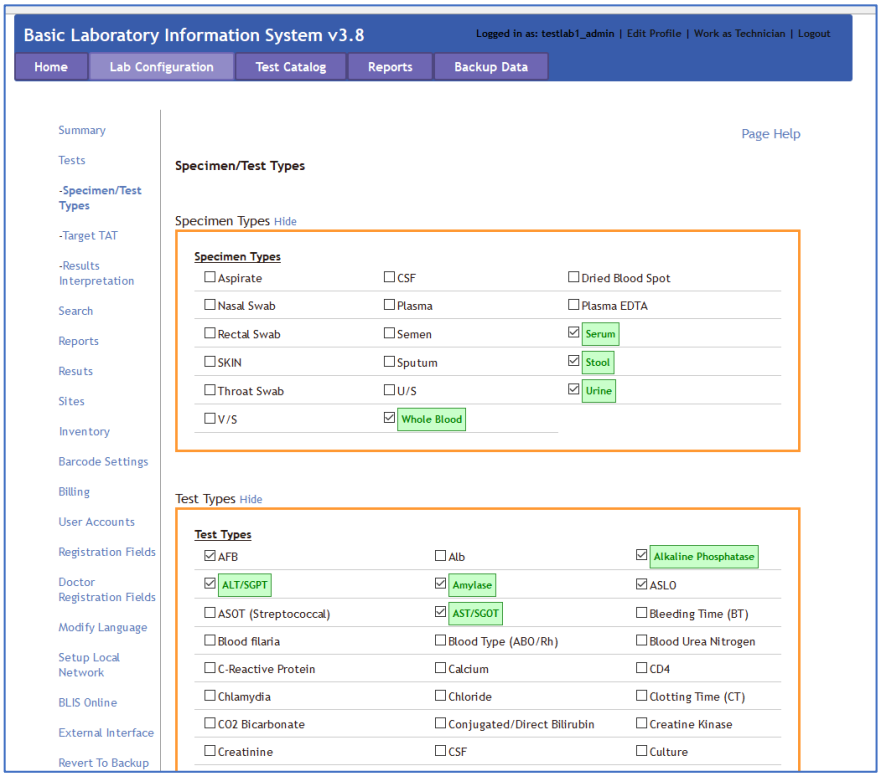
Workflow Test

- Workflow1: [Register Specimen] [admin login]
  - In Manager Mode:
    - Update specimen and test visibility in lab configuration
    - Add new specimen/test types
  - In Technician Mode:
    - Register a new patient
    - Register a new specimen type
    - Switch to the Results tab
    - Navigate to Single Specimen Results
    - search the patient's name
    - enter the results
    - Navigate to Verify Results
    - Verify the results entered
  - In Manager Mode:
    - Switch to Reports tab
    - Navigate to Patient Report
    - Search for the patient
    - Generate the report
    - Navigate to Daily Log

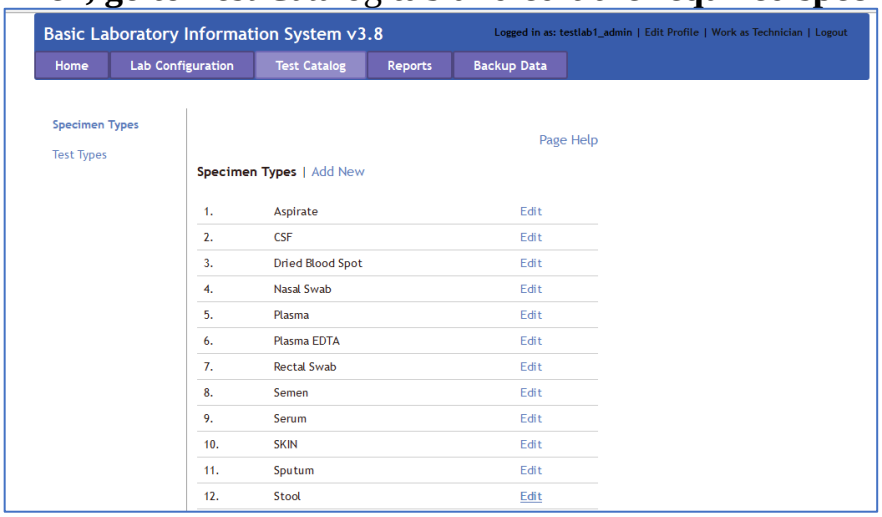
- Generate daily logs

Specimen Type Test

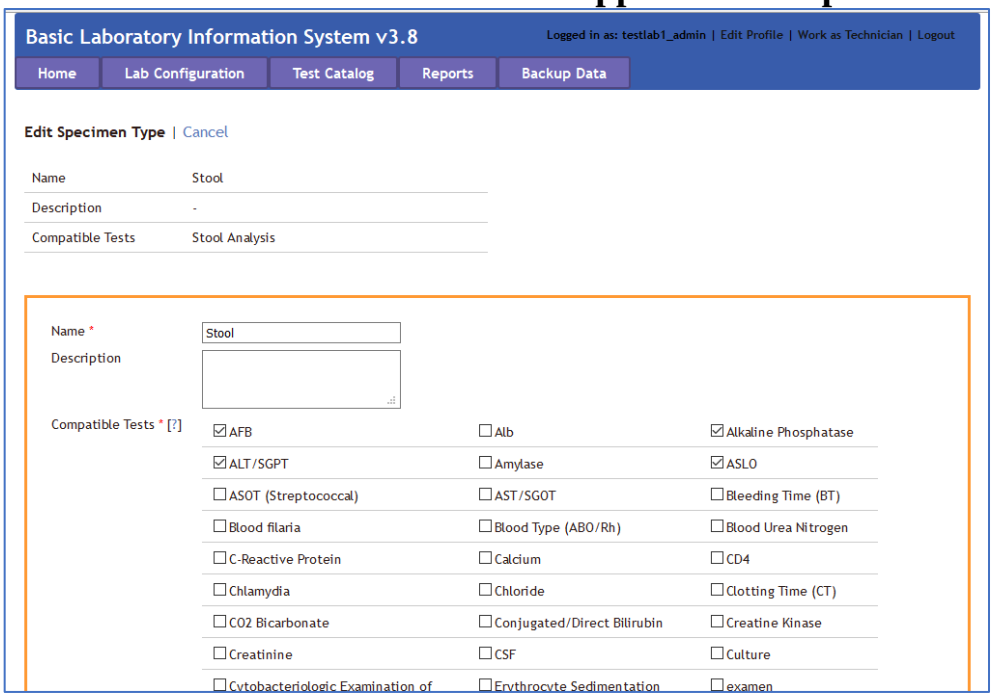
Please follow the steps below to enable tests for a specimen type:
First, go to the Lab Configuration page and enable the specimens and tests that should be visible:



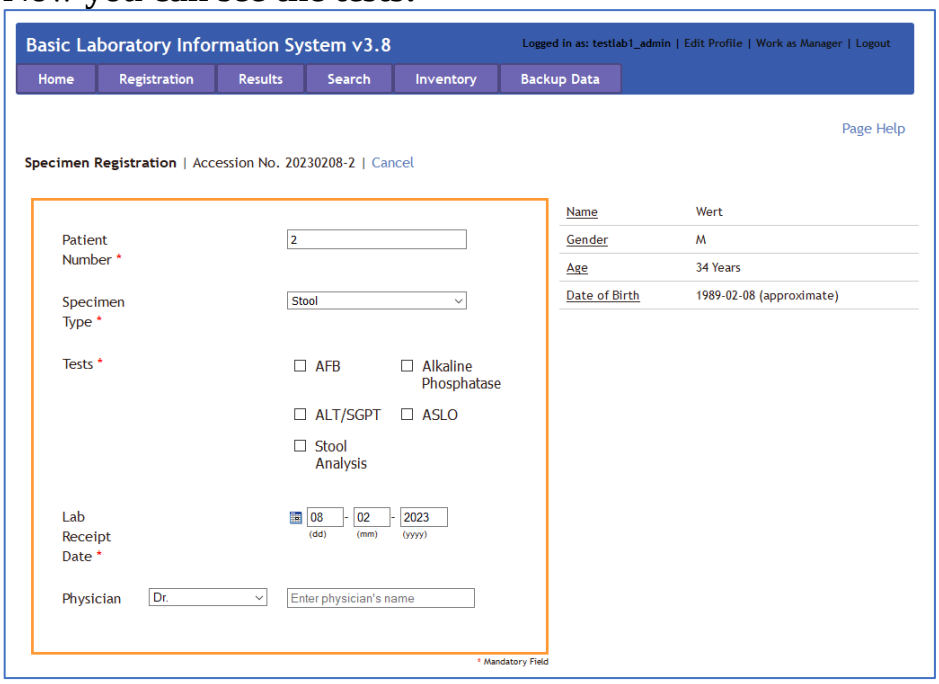
Then, go to Test Catalog tab and edit the required specimen:



Enable the tests that needed to be mapped for this specimen:



Now you can see the tests:



Smoke Test is concluded.