

Installation Guide with AWS (Advanced)

Step by step journey using AWS EC2 Ubuntu 24.04

For this guide you need basic knowledge with Linux, SSH, Bash

- [Getting Started](#)
- [Register AWS Account](#)
- [Register Domain Name](#)
- [Create Virtual Machine](#)
- [Install Finmars Platform](#)

Getting Started

Hello!

Welcome to Finmars, below you will find guidance of how to Install Everything

We also recommend you to obtain [Finmars ID](#)

Here is summary of what we will achieve during this tutorial

To complete this guide you will need **Mac** or **Linux**, it will require to use **Terminal**

- Register AWS account - [Link](#)
 - Register Domain Name - [Link](#)
 - Create Virtual Machine (VM) - [Link](#)
 - Install Finmars Platform - [Link](#)
 - Get to know our [User Quick Start](#) guide for next steps
-

If you have some troubles during Installation - reach for any support channel:

- Create a [Github Issue](#)
- Join our [Discord](#) Server
- Contact us at support@finmars.com

Register AWS Account

Here is a simple, step-by-step guide to make an AWS account:

1. **Open your web browser**
(for example: Chrome, Firefox, or Edge).
2. **Go to the AWS website**
Type aws.amazon.com in the address bar and press Enter.
3. **Start the sign-up**
Click the **“Create an AWS Account”** button at the top right.
4. **Enter your email and name**
 - In the first box, type your email address.
 - In the box below, type a name for your account (you can use your name or any name you like).
5. **Choose a password**
 - Make a password you can remember.
 - Re-type it to confirm.
6. **Fill in your contact details**
 - Select **“Professional”** or **“Personal”** account.
 - Type your full name, phone number, and address.
 - Click **“Continue”**.
7. **Add payment information**
 - AWS needs a credit or debit card to verify you.
 - Type your card number, month/year, and name on card.
 - Click **“Verify and Add”**.
8. **Confirm your phone number**
 - Choose your country code and type your phone number.
 - Click **“Send SMS”**.
 - You’ll get a text with a code. Type that code into the box.
9. **Pick a support plan**
 - You can choose the **Free Tier** plan (it has no monthly cost).
 - Click **“Continue”**.
10. **Finish and sign in**
 - After a few minutes, AWS will finish setting up.
 - Click **“Sign In to the Console”**.
 - Enter your email and password again.

You now have an AWS account! ☑☑

You can log into the AWS Console and start using services.

Go to next step: [Register Domain Name](#)

Register Domain Name

You can register a domain right inside AWS using Route 53. Here's how, in simple steps:

- 1. Sign in to AWS Console**
Go to <https://console.aws.amazon.com> and log in.
- 2. Open Route 53**
In the search bar at top, type "Route 53" and click the Route 53 service.
- 3. Go to Registered Domains**
In the left menu, click "**Registered domains.**"
- 4. Start a new registration**
Click the "**Register domain**" button.
- 5. Search your name**
 - In the box, type the name you want (for example,).
 - Click "**Check**".
- 6. Choose an available name**
 - If it's free, click "**Add to cart.**"
 - If not, try a different name or ending (like .
- 7. View your cart**
Click "**Review**" or go to the cart icon.
- 8. Enter contact details**
 - AWS needs your name, address, email, and phone.
 - If you want privacy, check "**Enable privacy protection.**"
 - Click "**Continue.**"
- 9. Verify and purchase**
 - Review the price and years (1 year, 2 years, etc.).
 - Click "**Complete order.**"
- 10. Wait for confirmation**
AWS will send you an email when your domain is ready. It usually takes a few minutes.
- 11. Create a Hosted Zone** (to use your domain)
 - Back in Route 53, click "**Hosted zones.**"
 - Click "**Create hosted zone.**"
 - Type your new domain name and click "**Create.**"
- 12. Point your domain to AWS**
 - In your hosted zone, copy the "Name servers" listed.
 - If AWS registered your domain, this is set automatically.
 - If you used another registrar, paste these name servers into their DNS settings.

Now your domain is registered and ready in AWS! ☐☐

You can add records (A, CNAME, MX) in your hosted zone to make your finmars work. See it in "

Create Virtual Machine" Page

Create Virtual Machine

Sure! Let's make your EC2 and name it **finmars-platform-vm**. Follow these steps:

1. **Sign in to AWS**
 - Open your browser and go to console.aws.amazon.com.
 - Enter your AWS email and password.
2. **Open EC2**
 - At the top, click the search box and type **EC2**.
 - Click **EC2** under "Services."
3. **Launch a new instance**
 - Click the blue **Launch instances** button.
4. **Name your instance**
 - In the **Name tag** box, type **finmars-platform-vm**.
5. **Choose AMI (Ubuntu 24.04)**
 - Scroll or search for **Ubuntu Server 24.04 LTS**.
 - Pick 64-bit (x86) (should be default)
 - Click **Select**.
6. **Select instance type (2 vCPU, 8 GiB RAM)**
 - Find and click **t3.large** (it has 2 vCPU and 8 GiB).
7. **Create or select key pair**
 - Choose **Create a new key pair**.
 - Name it (e.g. **finmars-platform-vm-key**).
 - Click **Create Key Pair** and save the `.pem` file safely. - **Do not Lose this file, if you lose it, you will not able to connect to your VM again**
8. **Configure instance details**
 - Under **Subnet**, pick one (any is fine).
 - Subnet - Default is ok (or do accordingly to your network configuration)
 - Turn **Auto-assign Public IP** to **Enable**. (If already enabled - OK)
 - Turn on checkbox **Allow SSH traffic from**
 - Turn on checkbox **Allow HTTPS traffic from the internet**
 - Turn on checkbox **Allow HTTP traffic from the internet**
 - Leave the rest as default.
 - Click **Next: Add Storage**.
9. **Add storage (256 GiB)**
 - Change the size from **8** to **256** in the root volume row.
 - Keep the volume type as **gp3** or **gp2**.
10. **Review and launch**
 - Check all your settings.
 - Click **Launch Instance**.
11. **Wait for your VM**
 - Click **View Instances**.

- Wait until its status is **running** and checks pass.

12. Connect to your VM

- Select the instance named **finmars-platform-vm**.
- Click **Connect**.
- Follow the instructions, for example:

```
# Move your Download key file to secure folder (~/.Downloads not recommended)

# if this first connect
chmod 400 "finmars-platform-vm.pem"
ssh -i ./finmars-platform-vm.pem ubuntu@<Public-IP>

# if this first connect
# Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
# yes
```

Your EC2 named **finmars-platform-vm** is ready! ☐☐

Now you need to assign your Public IP of your freshly created VM to subdomain of your domain.

1. Sign in to AWS

Go to console.aws.amazon.com and log in.

2. Open Route 53

In the top search bar, type **Route 53**, then click the service.

3. Go to Hosted Zones

In the left menu, click **"Hosted zones."**

4. Select your domain

Find and click the zone named your_domain.tld (for example, .

5. Create the first record

- Click **"Create record."**
- In **Record name**, type (so full name is). - It is Record for Actual Finmars Platform
- For **Record type**, choose **A - IPv4 address**.
- In **Value**, type your EC2 public IP (for example,). You can find it in EC2 details
- Leave **TTL** as default (300).
- Click **"Create records."**

6. Create the second record

- Click **"Create record"** again.
- In **Record name**, type (so full name is). - It is Record for Single-Sign-On (SSO) Finmars

- For **Record type**, choose **A - IPv4 address**.
- In **Value**, type the same EC2 public IP.
- Click **“Create records.”**

7. Wait a few minutes

DNS needs a little time to spread out. After about 5 minutes, both

- `finmars.example.com`
- `finmars-auth.example.com`

will go to your VM’s public IP.

That’s it! Now both sub-domains point to your **finmars-platform-vm** server.

You can verify it by run following command in Terminal (On Mac or Linux)

```
dig finmars.example.com
dig finmars-auth.example.com
```

Output should be like:

```
; <<>> DiG 9.10.6 <<>> finmars.example.com
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 39082
;; flags: qr rd ra; QUERY: 1, ANSWER: 1, AUTHORITY: 0, ADDITIONAL: 1

;; OPT PSEUDOSECTION:
; EDNS: version: 0, flags:; udp: 4096
;; QUESTION SECTION:
;finmars-platform-vm.finmars.com. IN A

;; ANSWER SECTION:
finmars.example.com. 300 IN A 203.0.113.25

;; Query time: 12 msec
;; SERVER: 192.168.178.1#53(192.168.178.1)
;; WHEN: Wed Jun 11 20:10:02 CEST 2025
;; MSG SIZE rcvd: 76
```

Now go to next step: [Install Finmars Platform](#)

Install Finmars Platform

Here is a Full Guide to install Finmars on your Ubuntu VM:

Complete time of full installation is less than 30 minutes, by the end of it you will get a Finmars Platform Installed on your Linux Server

1. Connect to your VM

Example:

```
# if this first connect
chmod 400 finmars-platform-vm.pem

ssh -i ./finmars-platform-vm.pem ubuntu@<Public-IP>

# if this first connect
# Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
# yes
```

2. Install Docker

```
sudo apt update
sudo apt install -y ca-certificates curl gnupg lsb-release ntp

wget -qO- https://get.docker.com/ | sh

sudo usermod -aG docker $USER
newgrp docker
```

To check if Docker installed:

```
docker version
```

Should output:

```
Client: Docker Engine - Community
Version:      28.2.2
...
```

3. Install Make

```
sudo apt install -y make git
```

4. Create the finmars folder

```
sudo mkdir -p /opt/finmars  
sudo chown $USER:$USER /opt/finmars
```

5. Clone the Finmars Platform Community Edition repository from Github

See <https://github.com/finmars-platform/finmars-community-edition> repository

```
cd /opt/finmars  
git clone https://github.com/finmars-platform/finmars-community-edition.git .
```

6. Configure env

```
make generate-env
```

7. (Optional) Check created env `cat .env`

```
REALM_CODE=realm00000  
BASE_API_URL=space00000  
SECRET_KEY=2e849ee2  
JWT_SECRET_KEY=f0d51adba17320c742fb1f046122ce1a8e22ca679bf7c7df28aa873ed5ca3d7a  
ENCRYPTION_KEY=a594c6607a48629884753fe0a6b5a907c6d8be8d63e4c274c67be040c276b1c4  
  
DOMAIN_NAME=ap-finmars.finmars.com  
CSRF_COOKIE_DOMAIN=ap-finmars.finmars.com  
CSRF_TRUSTED_ORIGINS=https://ap-finmars.finmars.com  
  
PROD_APP_HOST=https://ap-finmars.finmars.com  
APP_HOST=https://ap-finmars.finmars.com  
PROD_API_HOST=https://ap-finmars.finmars.com  
API_HOST=https://ap-finmars.finmars.com  
  
KEYCLOAK_REALM=finmars  
KEYCLOAK_SERVER_URL=https://ap-finmars-auth.finmars.com  
KEYCLOAK_URL=https://ap-finmars-auth.finmars.com  
PROD_KEYCLOAK_URL=https://ap-finmars-auth.finmars.com  
  
NUXT_APP_BASE_URL=/  
  
DB_HOST=db  
DB_PORT=5432  
DB_NAME=finmars_dev
```

DB_USER=finmars_dev
DB_PASSWORD=e21717b7ba0d4287dcdc292edc3c2164

KC_DB_URL_HOST=db_keycloak
KC_DB_PORT=5432
KC_DB_USERNAME=keycloak
KC_DB_PASSWORD=5de8131f84d79b68ba47d25a922dae92
KC_DB_URL_DATABASE=keycloak

RABBITMQ_HOST=rabbitmq

REDIS_HOST=redis

USE_FILESYSTEM_STORAGE=True

SERVER_TYPE=local
DEBUG=False
USE_DEBUGGER=False
DJANGO_LOG_LEVEL=INFO
PROFILER=False
ENABLE_DEV_DOCUMENTATION=False

EDITION_TYPE=community

ADMIN_USERNAME=test
ADMIN_PASSWORD=test
REDIRECT_PATH="/realm00000/space00000/a/#!/dashboard"

MAIN_DOMAIN_NAME=ap-finmars.finmars.com
AUTH_DOMAIN_NAME=ap-finmars-auth.finmars.com

8. Release certs

```
make init-cert
```

You should see something like this as successful result:

```
[+] Running 2/2
✓ Network finmars_default      Created
✓ Container finmars-certbot-1   Created
Attaching to certbot-1
certbot-1 | Saving debug log to /var/log/letsencrypt/letsencrypt.log
certbot-1 | Account registered.
certbot-1 | Requesting a certificate for finmars-platform-vm.finmars.com and finmars-platform-vm-auth.finmars.com
certbot-1 |
certbot-1 | Successfully received certificate.
certbot-1 | Certificate is saved at: /etc/letsencrypt/live/finmars-platform-vm.finmars.com/fullchain.pem
certbot-1 | Key is saved at: /etc/letsencrypt/live/finmars-platform-vm.finmars.com/privkey.pem
certbot-1 | This certificate expires on 2025-09-09.
certbot-1 | These files will be updated when the certificate renews.
certbot-1 | NEXT STEPS:
certbot-1 | - The certificate will need to be renewed before it expires. Certbot can automatically renew the certificate in the
certbot-1 |   ound, but you may need to take steps to enable that functionality. See https://certbot.org/renewal-setup for instructions.
certbot-1 |
certbot-1 | -----
certbot-1 | If you like Certbot, please consider supporting our work by:
certbot-1 |   * Donating to ISRG / Let's Encrypt:  https://letsencrypt.org/donate
certbot-1 |   * Donating to EFF:                  https://eff.org/donate-le
certbot-1 | -----
```

9. Init keycloak

```
make init-keycloak
```

You should see something like this as successful result:

```
[+] Running 1/1
✓ Container finmars-db_keycloak-1 Running
✓ Container finmars-keycloak-1 Started
⏸ Waiting for Keycloak to be ready...
Waiting for Keycloak to be ready...
Keycloak is ready!
✓ Configuring admin credentials...
Logging into http://localhost:8080 as user admin of realm master
+ Creating user admin...
Created new user with id '7d11f9ff-a9e2-4ec7-b141-b57665a9303d'
+ Setting password for user admin...
[+] Running 4/4
✓ Container finmars-keycloak-1 Removed
✓ Container finmars-certbot-1 Removed
✓ Container finmars-db_keycloak-1 Removed
✓ Network finmars_default Removed
✓ Done!
ubuntu@ip-172-31-26-65: /opt/finmars$
```

10. Run database migrations

```
make migrate
```

You should see something like this as successful result:

```
Applying workflow.0016_alter_schedule_options_alter_space_options_and_more... OK
Applying workflow.0017_workflow_finished_at... OK
Applying workflow.0018_workflow_parent... OK
Applying workflow.0019_alter_task_status... OK
Applying workflow.0020_delete_repeated_workflow_templates... OK
Applying workflow.0021_alter_workflowtemplate_options_and_more... OK
Applying workflow.0022_add_export_backend_historical_records_crontab_task... OK
[local] [realm00000] [workflow] [INFO] [2025-06-11 18:52:41,221] [MainProcess] [workflow] [apps:46] - Bootstrapping Workflow Application
[local] [realm00000] [workflow] [INFO] [2025-06-11 18:52:41,221] [MainProcess] [workflow] [apps:50] - bootstrap: Current search path: space00000
[local] [realm00000] [workflow] [INFO] [2025-06-11 18:52:41,225] [MainProcess] [workflow] [apps:79] - bootstrap:creating_new_space: space00000
[local] [realm00000] [workflow] [INFO] [2025-06-11 18:52:41,228] [MainProcess] [workflow] [apps:90] - Finmars bot created User matching query does not exist.
[+] Running 3/3
✓ Container finmars-redis-1 Removed
✓ Container finmars-db-1 Removed
✓ Network finmars_default Removed
✓ Done!
ubuntu@ip-172-31-26-65: /opt/finmars$
```

11. Start all services

```
make up
```

You should see something like this and other logs as successful result:

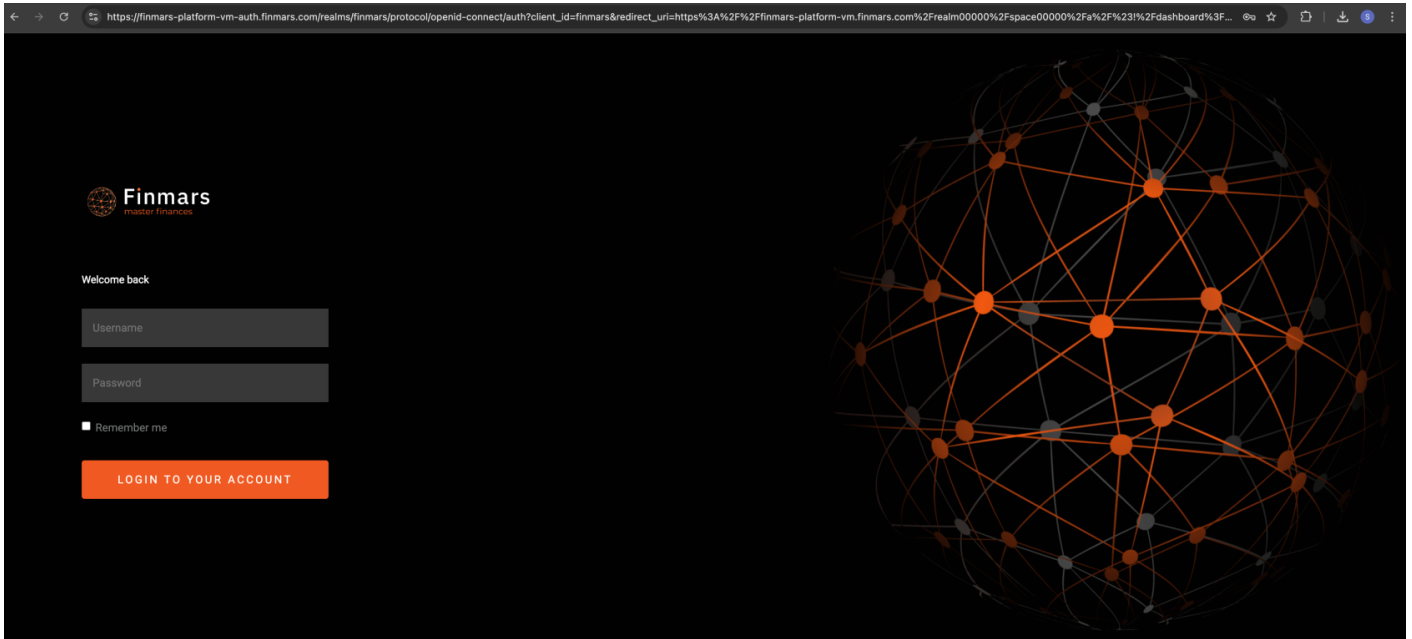
To Verify that is everything running

And you should see something like:

It means all the Docker Containers are running.

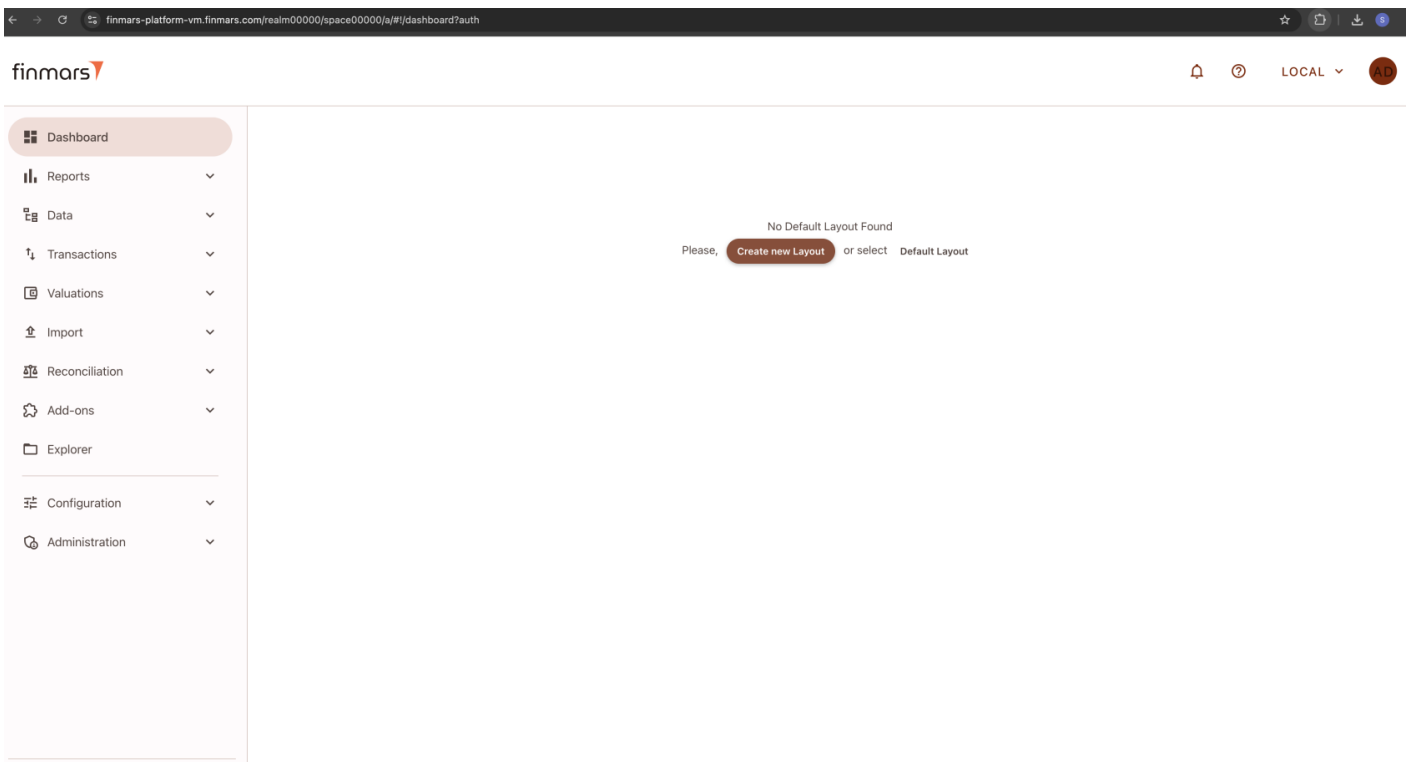
When all is done, you can Open Web Browser at **finmars.example.com** and you should see **Finmars Welcome Page**

After you press **Login** you will be redirected to **Finmars Single-Sign-On (SSO)** to **finmars-auth.example.com**



Login with your Credentials that you provided in 6) **Configure env** step. It should be yours `ADMIN_USERNAME` and `ADMIN_PASSWORD`

After Login you will be redirected to **Dashboard page** in **Finmars Platform**



Congratulations you finished your **Finmars Platform** Installation **Successfully!** Well Done!

Get to know our [User Quick Start](#) guide for next steps

If you have some troubles during Installation - reach for any support channel:

- Create a [Github Issue](#)
- Join our [Discord](#) Server
- Contact us at support@finmars.com