eCR Now Setup Instructions – Linux

# **Prerequisites**

* **docker-desktop to be installed in your system**
* **ecr-now.zip shared by our team with env & other files.**

## System Requirements for adding docker-desktop in windows

* **Operating System:** A 64-bit version of one of the following:
* Ubuntu
* Debian
* Fedora
* CentOS
* **Hardware:**
* Minimum 4GB RAM.
* BIOS-level hardware virtualization support.

## **Installing Docker**

### **Installing Docker on Linux**

#### **For Ubuntu/Debian:**

1. **Update your package database:**

sudo apt update

1. **Install Docker:**

sudo apt install docker-ce

## **Verifying Configuration**

1. Run the following command to check Docker version

**docker --version**

## **Application Setup with Docker**

**Step: 1 Creating a docker network**

Create a docker network on the name of ecrnow\_network**,** use below command:

docker network create ecrnow\_network

**Step: 2 Run PostgreSQL Container**

Run the PostgreSQL container with the specified network and environment variables, use below command:

docker run -d --network ecrnow\_network --name postgres\_container --restart always -e POSTGRES\_PASSWORD=ecrnow@2024 -p 5432:5432 postgres:15

Note: Add password of your choice  
  
***Step: 3 Run PgAdmin Container***

1)Run PgAdmin Container:

docker run -d --network ecrnow\_network --name pgadmin-container -p 5050:80 -e PGADMIN\_DEFAULT\_EMAIL=user@domain.com -e PGADMIN\_DEFAULT\_PASSWORD=postgres -d dpage/pgadmin4

Remember to replace [user@domain.com](mailto:user@domain.com) and postgres with your actual email and password.

This command runs PgAdmin in a Docker container (dpage/pgadmin4) on port 5050. It sets the default login credentials ([user@domain.com](mailto:user@domain.com) and postgres).  
  
  
 2) Refer this link [Connecting PostgreSQL Container using PgAdmin 4](#_Connecting_to_Database)

**Step: 4 Create ecrnow Database**

**If the “ ecrnow” database is created using PgAdmin in Step 3, you can skip this step**.

Execute the following command to create the **ecrnow** database inside the PostgreSQL container

docker exec -it postgres\_container psql -U postgres -c "CREATE DATABASE ecrnow;"

**Step: 5 Setting up environment variables**

Unzip the ecr-now.zip place it in drive and go to env.list file inside ecr-now folder and adjust variables as shown below

LOGGING\_FILE\_LOC= /home/your\_username/ecr-now/ecrNow.log

ESRD\_FILE\_LOC= /home/your\_username/ ecr-now/eRSDv2\_specification\_bundle.json

SCHEMATRON\_FILE\_LOC=/home/your\_username/ /ecr-now/schematron/CDAR2\_IG\_PHCASERPT\_R2\_STU1.1\_SCHEMATRON.sch

SCHEMA\_FILE\_LOC= /home/your\_username/ ecr-now/schemas/CDA\_SDTC.xsd

KAR\_DIRECTORY= /home/your\_username/ ecr-now/kars

BSA\_OUTPUT\_DIRECTORY= /home/your\_username/ecr-now/bsa-output/

CUSTOM\_QUERY\_DIRECTORY= /home/your\_username/ecr-now/custom-queries

DATABASE\_URL=jdbc:postgresql://postgres\_container:5432/postgres

**Note:- Replace /home/your\_username with the actual path to the directories on your machine.**

### **Step: 6 Run eCR Now Backend**

Run the eCR Now backend container with the specified environment variables and volume mappings:

docker run -d --network ecrnow\_network --restart always --env-file /home/your\_username/ecr-now/env.list -e jdbc.username=postgres -e jdbc.password=ecrnow@2024 -e security.key=test123 --link postgres\_container:postgres -v /home/your\_username/ecr-now:/home/your\_username/ecr-now -p 8081:8081 --name ecr-now drajerhealth/ecr-now:ecr-now-3.1.4-postgres



Note: -

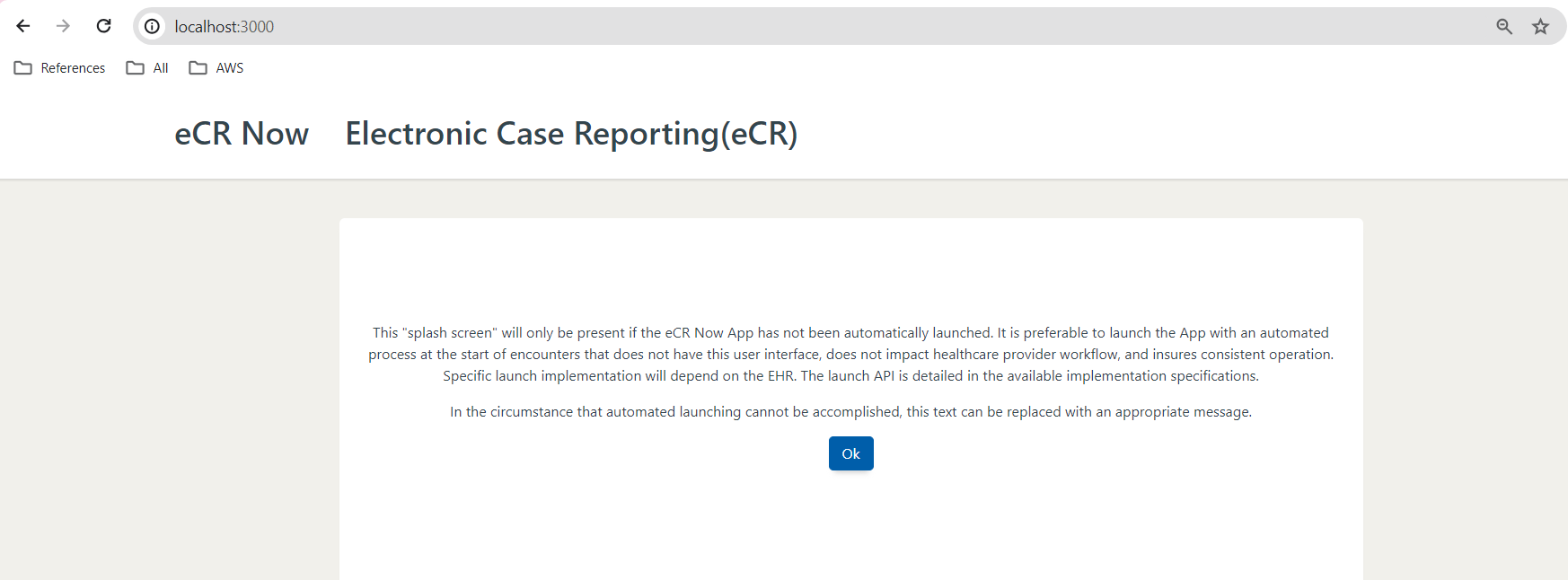
* **Replace /home/your\_username with the actual paths to the directories on your machine. Ensure these paths match the paths specified in your env.list.**
* **Add the PostgreSQL password used when creating the PostgreSQL container.**

**Step: 7 Run eCR Now UI**

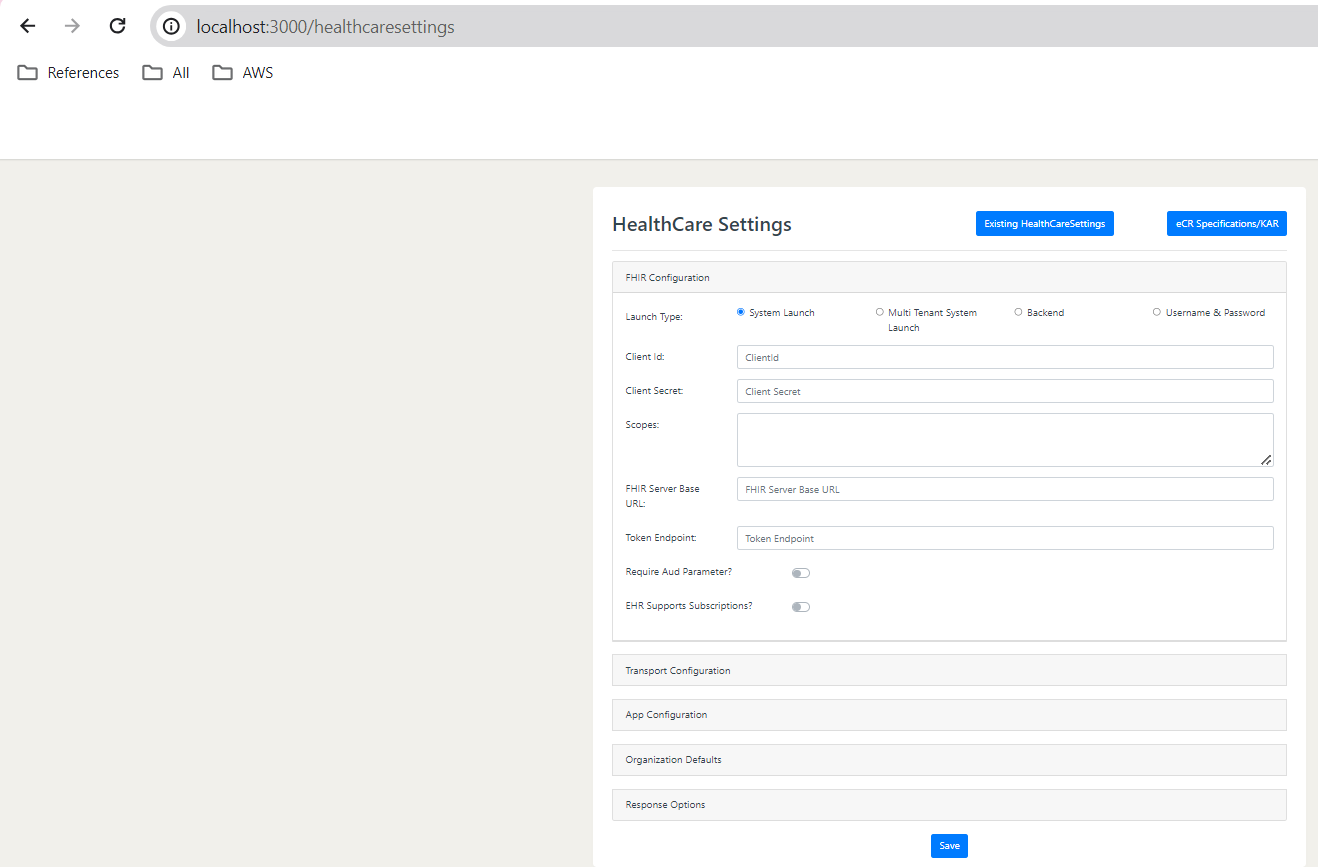
Run the eCR Now UI container with the specified environment variable:

docker run -d --network ecrnow\_network --restart always -e REACT\_APP\_ECR\_BASE\_URL=http://localhost:8081 --name ecrNow-ui -p 3000:3000 drajerhealth/ecr-now:ecr-now-ui-3.1.4

**Go to browser and hit http://localhost:3000**



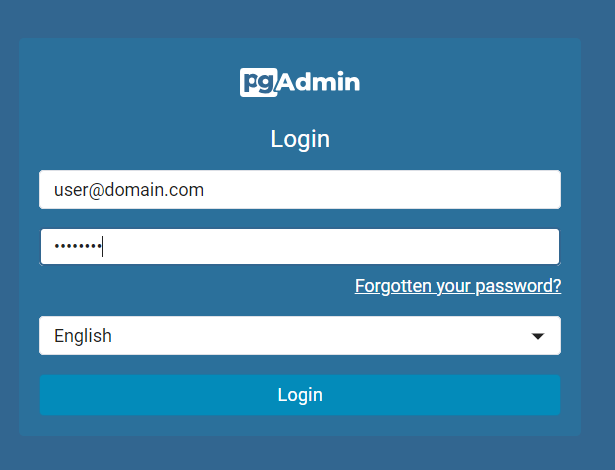
<http://localhost:3000/healthcaresettings>



# Connecting to Database Container using pgAdmin 4

## Log in to pgAdmin 4

Once the container is successfully running (if you encounter any issues, it’s a good idea to check the Docker Desktop app to ensure the container is running), you can access pgAdmin by navigating to localhost:5050 in a web browser of your choice.

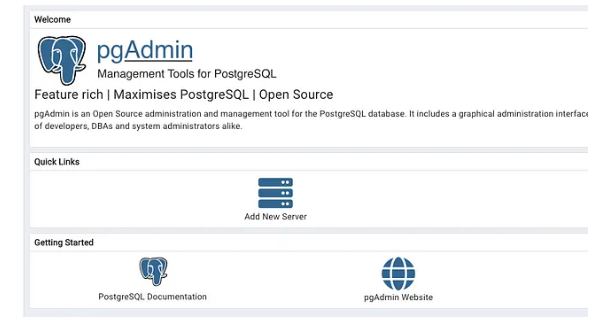
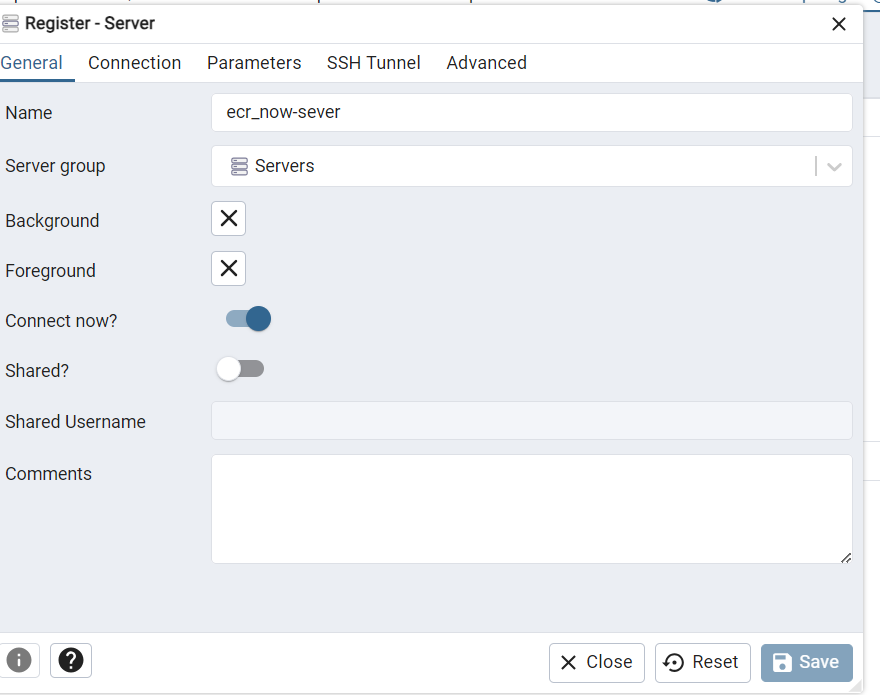
  
  
  
You will then see a login prompt; you will be able to log in with the e-mail address and password that you specified previously when running the pg Admin Docker container

, in our case “user@domain.com” and “postgres”.

Note :-The username and password should be the same as those used when running the pg Admin Docker container.

## Connect to Database Container/ Add Server

In the next step, we are going to connect to the database container. For this, you need to click on Add New Server:

  
  
And enter the relevant information to connect to our database, in the Name field we can choose an Alias to refer to our database in Pg Admin:  


Before we proceed, it’s important to obtain the IP address of the “postgres” container. To find the IP address, you can execute the following command in your terminal (Linux/macOS) or PowerShell (Windows):

docker inspect -f '{{.NetworkSettings.Networks.ecrnow\_network.IPAddress}}' postgres\_container

In our case, it returned the IP address 172.17.0.3 (this may or may not be true for you). Now we enter all the connection information:

Host name/address: 172.17.0.3 (might be different for you)   
 or   
Host name/address: postgres\_container (name of Postgres container)

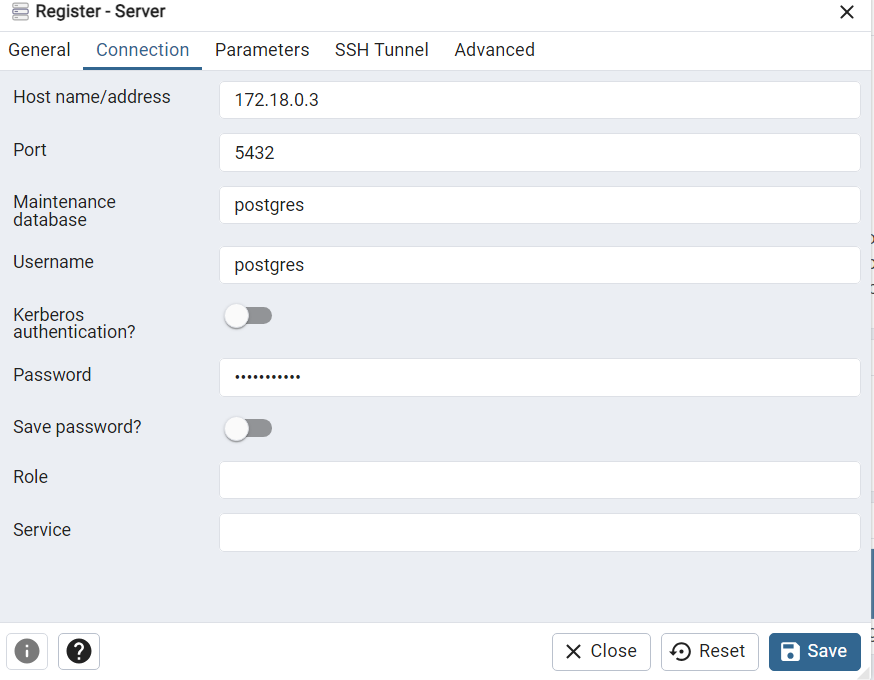
Port: 5432 (port of postgres container)

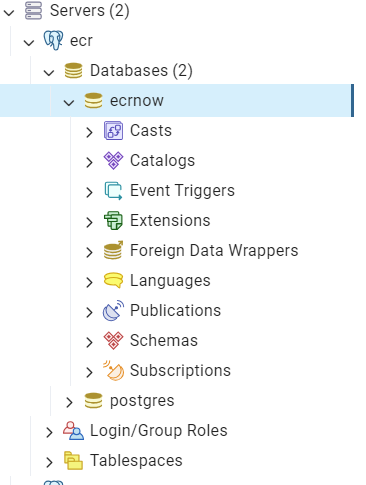
Maintenance database: postgres

Username: postgres (or another name if you changed it)

Password: ecrnow@2024 (or whatever password you selected for Postgres container)

Optional: Set save password to true



Click Save and you will be able to select your database server from the menu on the left side:  
  
  
  
  
  
Note:-Create the ecrnow database if database does not exist

Congratulations, now your environment should be up and running. For questions about pgAdmin 4 refer to the documentation:  
[PgAdmin Docs](https://www.pgadmin.org/docs/pgadmin4/development/index.html?source=post_page-----4a8d81048aea--------------------------------)