# Deploy to Production

## Prerequisites:

These components / applications must be installed before proceeding into deployment steps

- **Git**
  To install git
  `sudo yum update -y && sudo yum install git -y`

- **Docker**
  To install docker follow the steps on https://docs.docker.com/engine/install/rhel/

  After running command
  `sudo dnf config-manager --add-repo https://download.docker.com/linux/rhel/docker-ce.repo`

  open file `/etc/yum.repos.d/docker-ce.repo` and change the number pointed by red arrow in the image below with the RHEL version

```
[docker-ce-stable]
name=Docker CE Stable - $basearch
baseurl=https://download.docker.com/linux/rhel/9/$basearch/stable
enabled=1
gpgcheck=1
gpgkey=https://download.docker.com/linux/rhel/gpg
```

  Then continue the installation steps

## Assumptions:

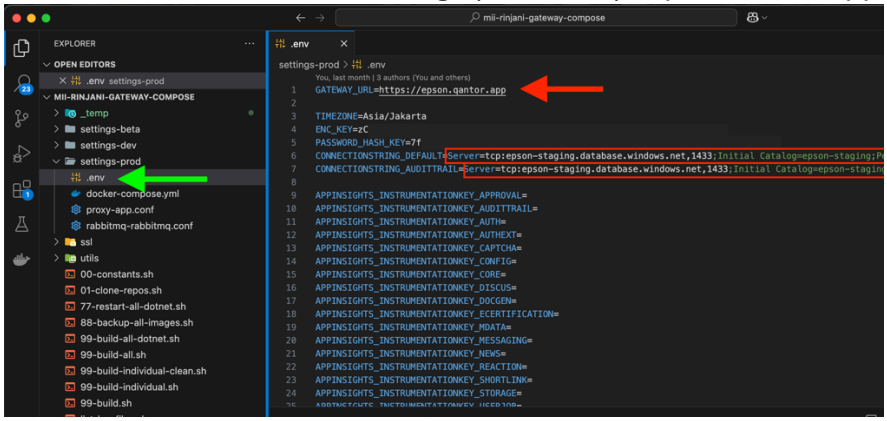
- The user for SSH is `app`
- The app will be installed on `/apps/rinjani` directory
- The git username is `epson-id-admin`

## 1. Adjust database connection string

1.1. Clone "compose" repository to your computer: `https://epson-id-admin@bitbucket.org/epson-id/epson-gateway-compose.git`

1.2. Open `settings-prod/.env` file (green arrow).
Revise `GATEWAY_URL` (red arrow): This is the URL that the user will use to open the application.

Revise the two connection strings (red boxes) to point to the application database.




1.3. Commit the changes and push the commit back to Bitbucket server

# 2. Import database structures

2.1. Open the `sql` folder and execute all SQL scripts ordered by file name.

# 3. Setup Deployment Script

3.1. Login to the application server via SSH.

3.2. Create deployment directory.

```
sudo mkdir -p /apps && \
sudo chown app:app /apps && \
mkdir -p /apps/rinjani && \
mkdir -p /apps/rinjani/src && \
cd /apps/rinjani/src
```

3.3. Clone compose repository. Make sure that current directory is `/apps/rinjani/src`.

```
git clone https://epson-id-admin@bitbucket.org/epson-id/epson-gateway-compose.git compose
```

3.4. Go to “compose” directory

```
cd /apps/rinjani/src/compose
```

3.5. Start the services by running this command

```
./99-build.sh
```

```
app@epson-staging:/apps/rinjani/src/compose$ ./99-build.sh
Env to deploy
1) BETA
2) DEV
3) PROD
4) quit
Please enter your choice: 3

Git server: bitbucket.org
Git username: epson-id-admin
Git password for epson-id-admin:
```

Type “3” then press “enter”
Type `epson-id-admin` for **Git username.**
Type the **Git password**

3.6. Verify that all services was started


3.7. Setup RabbitMQ

`docker exec -ti rabbitmq /home/script/init-user-vhost.sh`

```
app@epson-staging:/apps/rinjani/src/compose$ docker exec -ti rabbitmq /home/script/init-user-vhost.sh
Adding user "rinjani" ...
Done. Don't forget to grant the user permissions to some virtual hosts! See 'rabbitmqctl help set_permissions' to learn more.
Adding vhost "rinjani" ...
Setting permissions for user "rinjani" in vhost "rinjani" ...
Setting permissions for user "admin" in vhost "rinjani" ...
app@epson-staging:/apps/rinjani/src/compose$
```

Login to RabbitMQ Dashboard to validate that RabbitMQ is running: `http://<server IP address>:8080`. Note: do not use “https” and do not use server’s domain name

3.8. Restart all .NET continers

`./77-restart-all-dotnet.sh`

3.9. Verify Hangfire

Login to `http://<server IP address>:9090` to verify that Hangfire is running

3.10. Login to application's CMS: http://<server IP address>/admin.
Note: do not use "https" and do not use server's domain name