La Fresca Administration Manual

This document provides detailed instructions for setting up and running the La Fresca project locally using Docker. Follow the steps below to get a successful local deployment.

Prerequisites

Install Docker

You must have Docker and Docker Compose installed. Follow the instructions below based on your operating system or follow the official docker installation guide. Link

Windows

1. Download **Docker Desktop** from https://www.docker.com/products/docker-desktop.
2. Install Docker Desktop:
   - Run the installer and follow the on-screen instructions.
   - Ensure that **WSL 2** is enabled during installation.
3. Start Docker Desktop and confirm it is running.

macOS

1. Download **Docker Desktop for Mac** from https://www.docker.com/products/docker-desktop.
2. Install Docker Desktop:
   - Open the `.dmg` file and drag Docker to your Applications folder.
   - Open Docker from Applications and follow the setup instructions.
3. Verify installation:

```
docker --version
docker-compose --version
```

Ubuntu

1. Set up Docker's apt repository:

```
 # Add Docker's official GPG key:
 sudo apt-get update
 sudo apt-get install ca-certificates curl
 sudo install -m 0755 -d /etc/apt/keyrings
 sudo curl -fsSL https://download.docker.com/linux/ubuntu/gpg -o
/etc/apt/keyrings/docker.asc
 sudo chmod a+r /etc/apt/keyrings/docker.asc

 # Add the repository to Apt sources:
 echo \
```

```
  "deb [arch=$(dpkg --print-architecture) signed-
by=/etc/apt/keyrings/docker.asc] https://download.docker.com/linux/ubuntu \
  $(. /etc/os-release && echo "$VERSION_CODENAME") stable" | \
  sudo tee /etc/apt/sources.list.d/docker.list > /dev/null
  sudo apt-get update
```

2. Install the Docker packages.:

```
  sudo apt-get install docker-ce docker-ce-cli containerd.io docker-buildx-
plugin docker-compose-plugin
```

3. Add your user to the Docker group:

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

Log out and back in for this change to take effect.

Step 1: Clone the Repositories

1. Clone the **local deployment repository** with all the neccecsary docker compose files and change to that directory(All the docker instructions should be executed within this directory.):

```
git clone https://github.com/La-Fresca/la-fresca-local-deployment
cd la-fresca-local-deployment
```

2. Clone the **frontend** repository:

```
git clone https://github.com/La-Fresca/la-fresca-frontend
```

3. Clone the **backend** repository:

```
git clone https://github.com/La-Fresca/la-fresca-backend
```

Step 2: Configure Environment Files

`.env` in `la-fresca-local-deployment`

1. Copy the `sample.env` file to a `.env` file:

```
cp sample.env .env
```

2. sample `.env` will look as following. This will work below default settings:

```
ME_CONFIG_MONGODB_SERVER="user:1234@mongo"
ME_CONFIG_BASICAUTH_USERNAME="la-fresca"
ME_CONFIG_BASICAUTH_PASSWORD="1234"
ME_CONFIG_SITE_BASEURL=/mongo-express
MONGODB_USERNAME="user"
MONGODB_PASSWORD="1234"
LAFRESCA_DB="mongodb://la-fresca:1234@mongo/LaFresca_DB"
```

Explanation of `.env` Variables

- `ME_CONFIG_MONGODB_SERVER`: Connection string for MongoDB in the format `username:password@host`. This is used by Mongo Express.
- `ME_CONFIG_BASICAUTH_USERNAME`: The username for basic authentication to access Mongo Express.
- `ME_CONFIG_BASICAUTH_PASSWORD`: The password for basic authentication to access Mongo Express.
- `ME_CONFIG_SITE_BASEURL`: The base URL path for accessing Mongo Express. In this case, it will be available at `lafresca.com/mongo-express`.
- `MONGODB_USERNAME` and `MONGODB_PASSWORD`: Credentials for connecting to the MongoDB database.
- `LAFRESCA_DB`: The MongoDB connection string for the La Fresca database.

`.env` in `la-fresca-frontend`

1. Navigate to the `la-fresca-frontend` folder and copy the `sample.env` to a new `.env` file:

```
cp sample.env .env
```

2. Edit `.env` with the following:

```
VITE_API_URL="https://lafresca.com/api"
VITE_UPLOAD_URL="https://lafresca.com/upload"
COOKIE_PROTOCOL="https:"
```

Explanation of `.env` Variables

- `VITE_API_URL`: Base URL for the backend API that the frontend will interact with.
- `VITE_UPLOAD_URL`: URL for file uploads.
- `COOKIE_PROTOCOL`: Specifies the protocol (`http` or `https`) for managing cookies securely.

Step 3: Update the Hosts File

Windows

1. Open Notepad as Administrator.
2. Navigate to the hosts file:

```
C:\Windows\System32\drivers\etc\hosts
```

3. Add the following line:

```
127.0.0.1    lafresca.com
```

macOS / Linux

1. Open the terminal and edit the hosts file:

```
sudo nano /etc/hosts
```

2. Add the following line:

```
127.0.0.1    lafresca.com
```

3. Save and exit (`Ctrl+O`, `Enter`, `Ctrl+X`).

Step 4: Start the Application

1. Navigate to the `la-fresca-local-deployment` folder:

```
cd la-fresca-local-deployment
```

2. Start the Docker containers:

```
docker-compose up -d
```

3. Wait until all containers are up and running. You can check the status with:

```
docker ps
```

Step 5: Access the Application

1. Open a browser and navigate to:

   ```
   https://lafresca.com
   ```

   There will be some warnings appear in the browser due to the self signed certificates used by caddy webserver. Please look below in the Notes section for further information.

2. To view the database, go to:

   ```
   https://lafresca.com/mongo-express
   ```

   Use the following credentials for basic authentication:

   - **Username**: `la-fresca`
   - **Password**: `1234`

3. Default credentials to log in as Super Admin:

   - **Email**: `admin@admin.com`
   - **Password**: `admin@123`

Notes

- To stop the Docker containers, run:

  ```
  docker-compose down
  ```

- If you encounter issues, check the logs with:

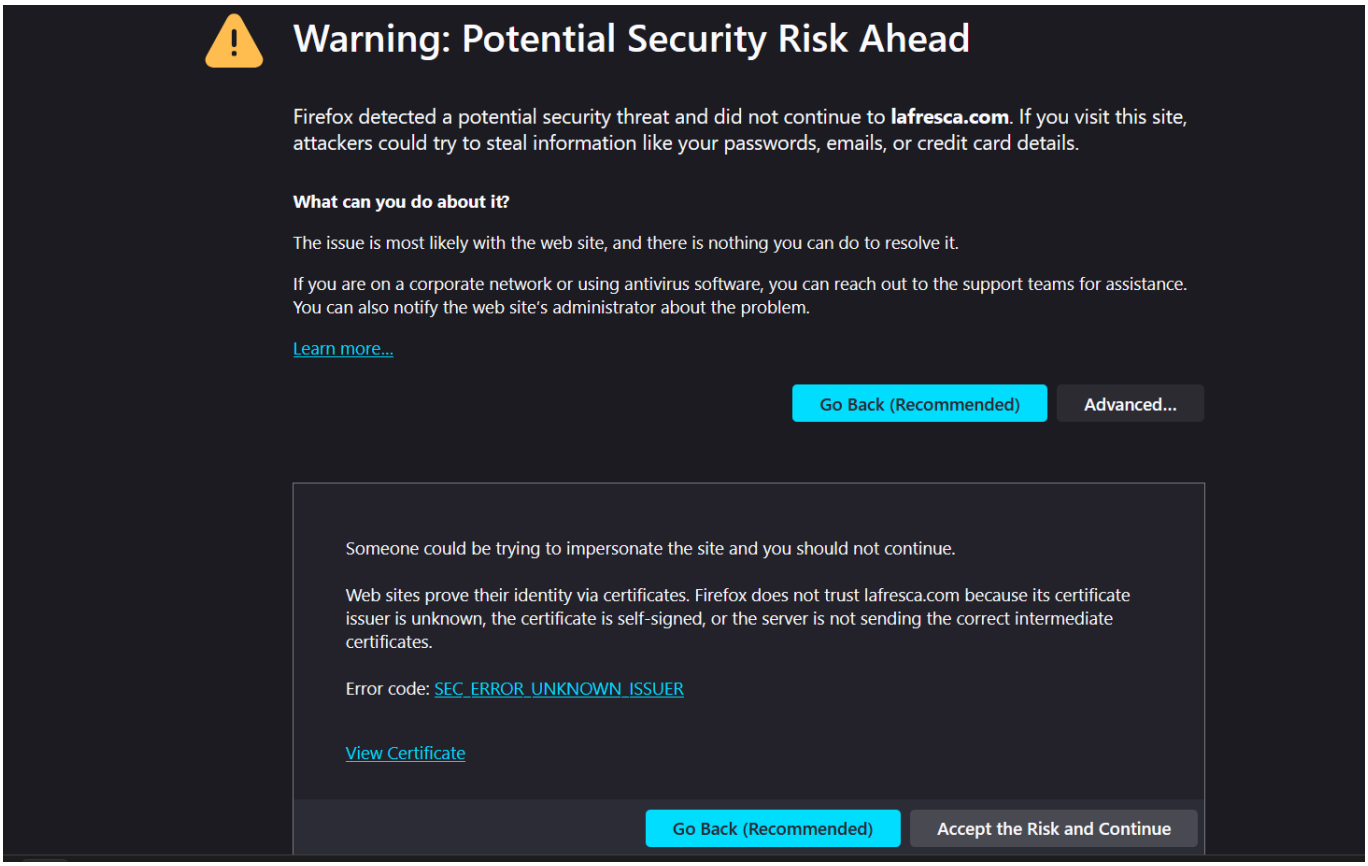
  ```
  docker-compose logs -f
  ```

Ignore Self-Signed Certificate Warning

When accessing the application for the first time, your browser may display a security warning due to the self-signed certificate. Follow these steps to proceed:

In Firefox:

1. Click on **Advanced**.
2. Click **Accept the Risk and Continue**.



In Chrome:

1. Click on **Advanced**.
2. Select **Proceed to unsafe**.