



DevSpace Chat – Setup Guide

This guide will walk you through setting up and running the **DevSpace Chat** application on your local machine.



Prerequisites

Please ensure the following are installed **before** proceeding:

- [Docker](#)
- [Java 17](#)
- [Node.js & npm](#)



Start Keycloak & Postgres via Docker

1. Open your terminal.
2. Navigate to the project root directory:

```
cd devspace-chat-repo
```

3. Start the services using Docker:

```
docker-compose up
```

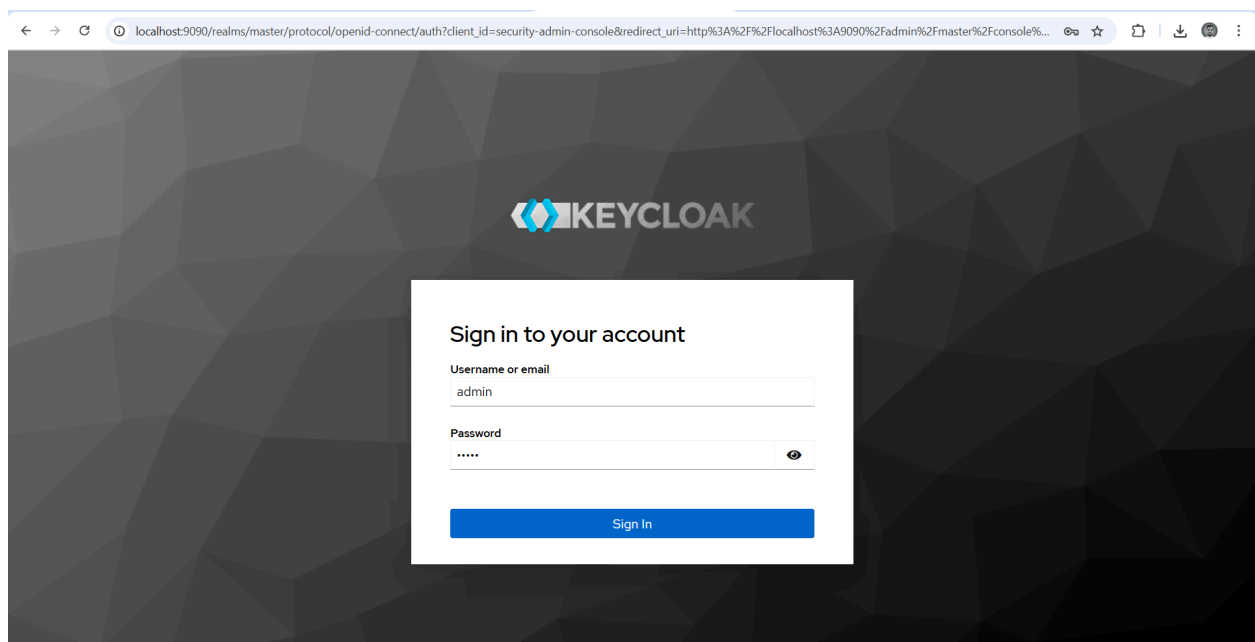
✓ **Note:** Ensure Docker Desktop is installed and running.

The screenshot shows the Docker Desktop interface. On the left is a sidebar with navigation options: Containers, Images, Volumes, Builds, Docker Hub, Docker Scout, and Extensions. The main area is titled 'Containers' and shows system usage: CPU at 0.31% / 800% and memory at 618.75MB / 7.54GB. A table of running containers is displayed, with three containers listed: 'devspace-chat-repo', 'postgres-sql-wac', and 'keycloak-wac'. The 'postgres-sql-wac' and 'keycloak-wac' rows are highlighted with a red box. Below the table, there are 'Walkthroughs' and a progress bar.

	Name	Container ID	Image	Port(s)	CPU (%)	Last started	Actions
<input type="checkbox"/>	devspace-chat-repo	-	-	-	0.44%	20 minutes ago	[Stop] [Refresh] [Delete]
<input type="checkbox"/>	postgres-sql-wac	cc73319a1b5b	postgres	5432:5432	0%	20 minutes ago	[Stop] [Refresh] [Delete]
<input type="checkbox"/>	keycloak-wac	1217af97f167	keycloak/keycloak:26.0.0	9090:8080	0.44%	20 minutes ago	[Stop] [Refresh] [Delete]

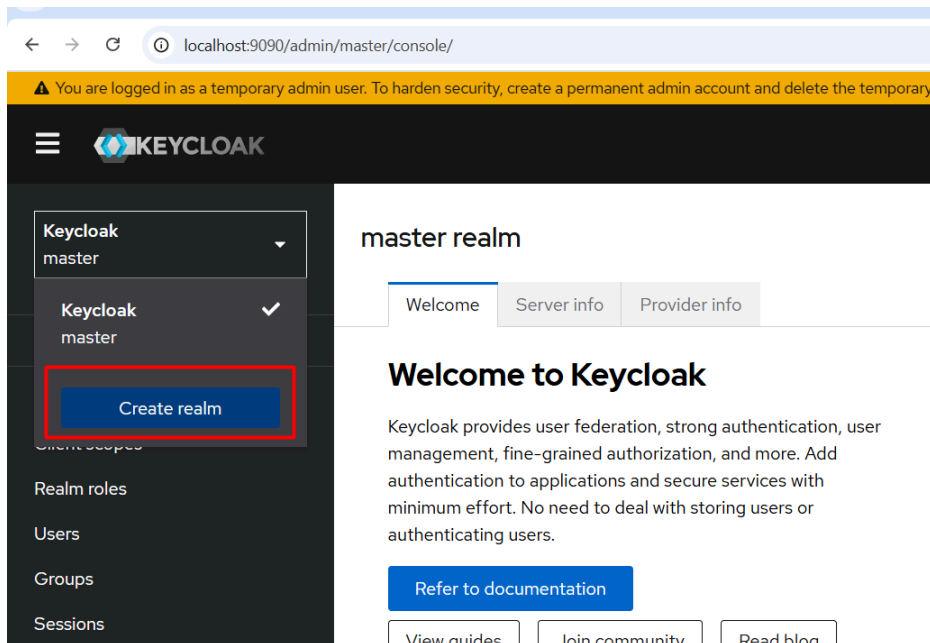
Keycloak Setup

1. Open your browser and navigate to:
<http://localhost:9090/admin/>
2. Login with credentials:
 - **Username:** admin
 - **Password:** admin

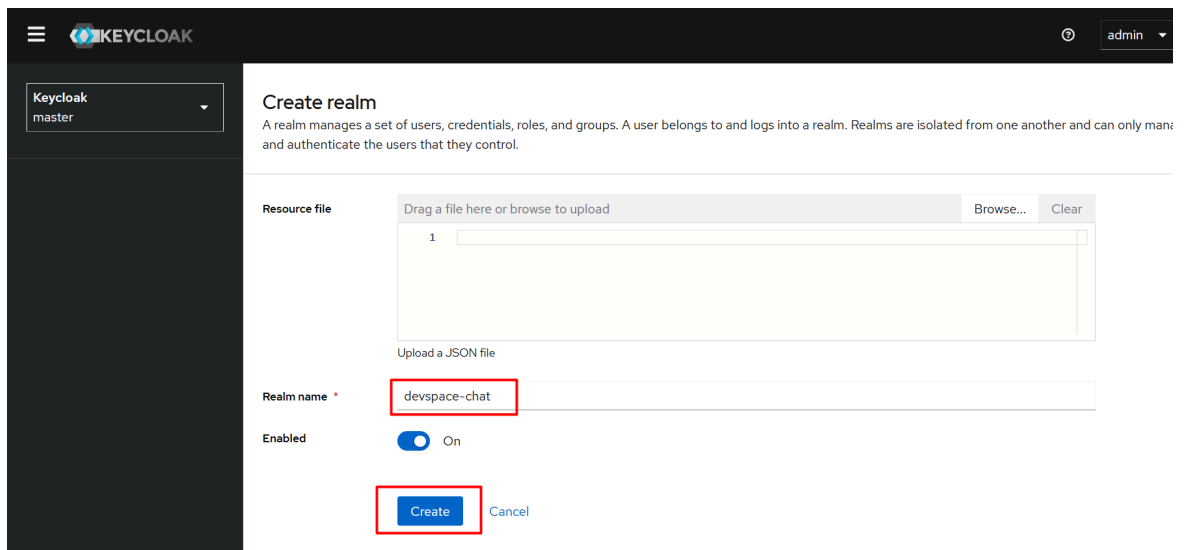


1. Create Realm

- Click the **top-left dropdown**.

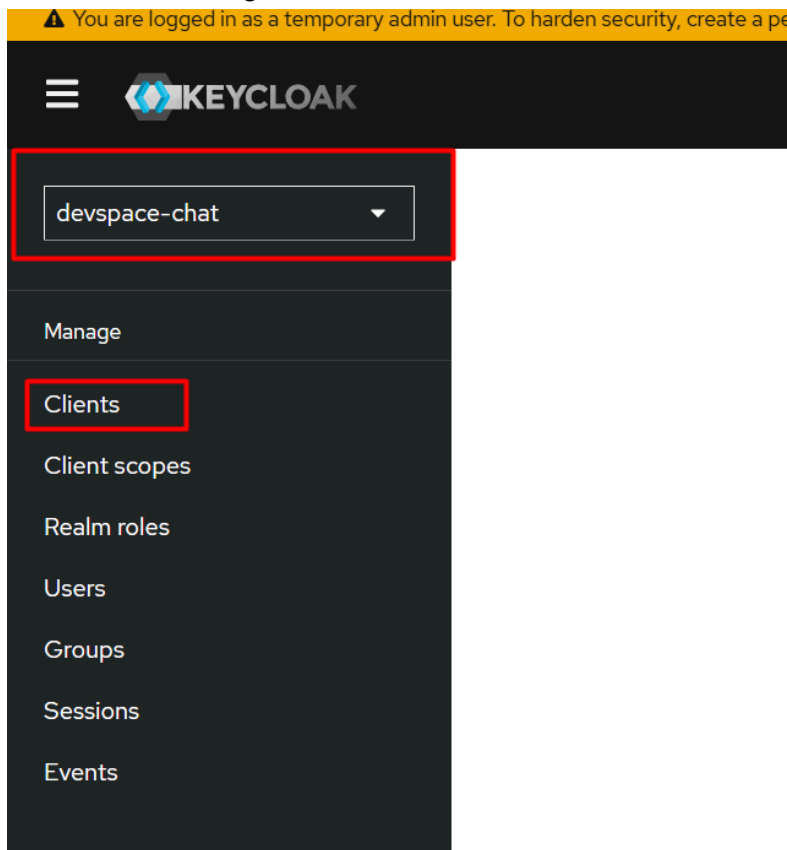


- Select **"Create Realm"** from the bottom of the list.
- Name your realm: **devspace-chat**
- Click **Create**.



2. Create Client

- In the left sidebar, go to **Clients**.



- Click **Create client**.
- Set
 - **Client Type:** OpenID

- **Client ID:** devspace-chat-app

Clients > Create client

Create client

Clients are applications and services that can request authentication of a user.

- 1 General settings
- 2 Capability config
- 3 Login settings

Client type ⓘ OpenID Connect

Client ID * ⓘ devspace-chat-app

Name ⓘ

Description ⓘ

Always display in UI ⓘ ☒ On

Back Next Cancel

- Click **Next** twice (no changes on the second screen).
- Enter the following:
 - **Root URL:** http://localhost:4200
 - **Home URL:** http://localhost:4200
 - **Valid Redirect URIs:** http://localhost:4200/*
 - **Web Origins:** *

- Click **Save**.

Clients are applications and services that can request authentication of a user.

1 General settings

2 Capability config

3 Login settings

Root URL ⓘ

http://localhost:4200

Home URL ⓘ

http://localhost:4200

Valid redirect URIs ⓘ

http://localhost:4200/*

+ Add valid redirect URIs

Valid post logout redirect URIs ⓘ

+ Add valid post logout redirect URIs

Web origins ⓘ

*

+ Add web origins

Back

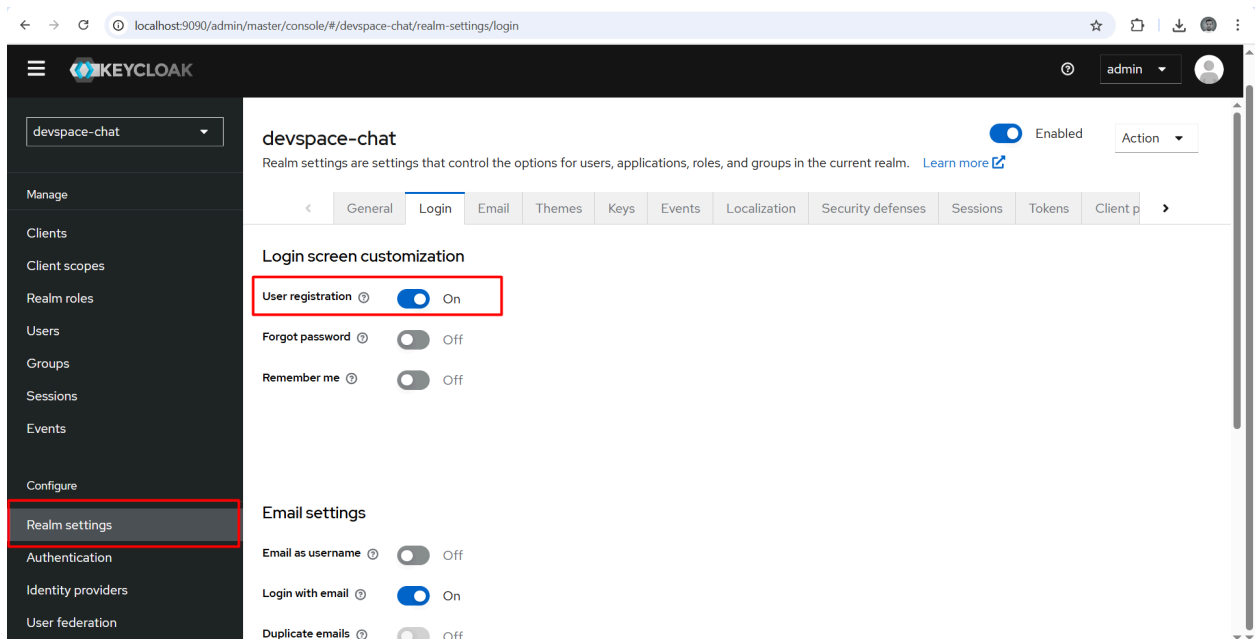
Save

Cancel

3. Enable User Registration

- Go to **Realm Settings** (left sidebar).

- Under the **Login** tab, toggle **User registration** to enabled.



✓ Keycloak is now configured and ready!

Backend Setup

1. Navigate to the backend directory:

```
cd devspace-chat-repo/devspace-chat-backend
```

2. Start the Spring Boot backend:

```
./mvnw.cmd spring-boot:run
```

Frontend Setup

1. Navigate to the frontend directory:

```
cd devspace-chat-repo/devspace-chat
```

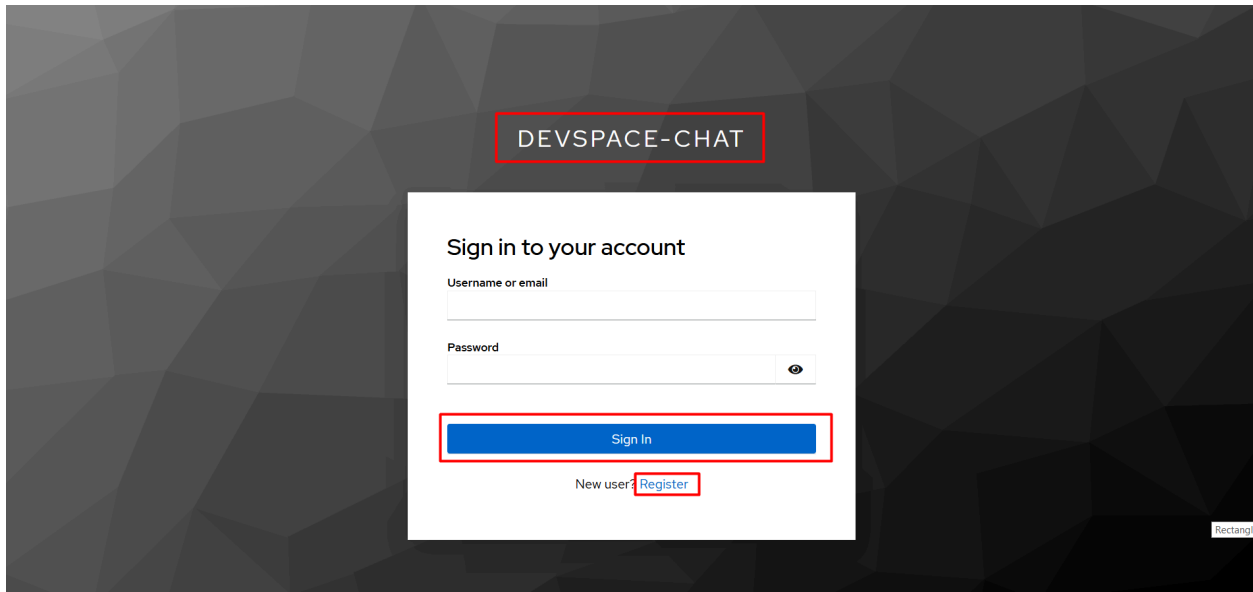
2. Install the dependencies:

```
npm install
```

3. Start the Angular application:

```
ng serve
```

4. Open your browser and visit: <http://localhost:4200/>



You're All Set!

The DevSpace Chat application should now be fully functional on your machine.