Paperless-ngx installieren

Unter Portainer, auf Stacks und neuen Stack im Editor erstellen.

Wichtig: Unter PAPERLESS_SECRET_KEY: XXX eine Zeichenkette generieren lassen.

Docker Compose

[crayon-6767a1b9cad44996288214/]

Benutzer erstellen

über ssh verbinden. Anschließend in die Konsole vom Docker Container bewegen:
[crayon-6767a1b9cad49497475274/]
[crayon-6767a1b9cad4b932394900/]

Dokumente + Einstellungen exportieren

über ssh verbinden. In dem Pfad nach -z wird eine Datei erstellt
[crayon-6767a1b9cad4c624747662/]

SQL Server Datenbank Design -> Kroki ER Diagramm

https://kroki.io/examples.html#erd

[crayon-6767a1b9cb14a321017938/]

SQL Query um Tabellen und Spalten auszulesen inkl. PK und FK

[crayon-6767a1b9cb2d3526220256/]

Host oder Webhost mit Autofac

[crayon-6767a1b9cb453629360079/]

Entity Framework EF – Tabellen Models generieren lassen

Alt + T -> „NU“ eingeben
[crayon-6767a1b9cb646424030539/]

Windows Powershell Dienst einrichten

[crayon-6767a1b9cb7b6937150308/]

SQL Server Statistik IO und Time

[crayon-6767a1b9cb920707199552/]

Nuxt 3 mit Keycloak autorisieren

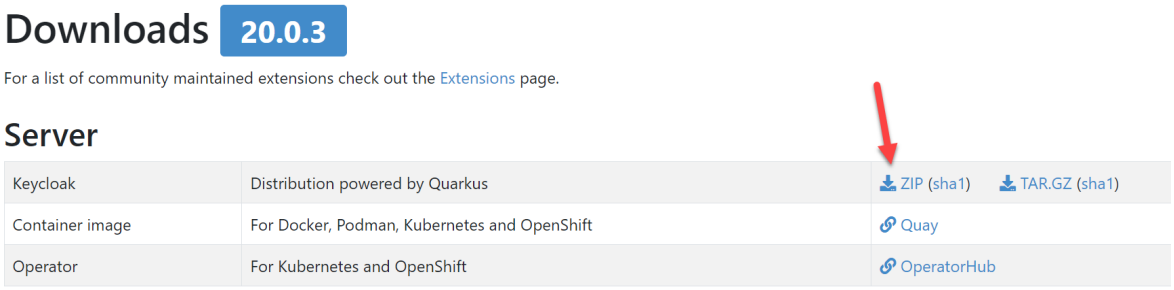
In diesem Beitrag will ich eine Schritt für Schritt Anleitung geben, wie man Nuxt 3 mit Keycloak autorisiert.

Die Thematik um Autorisierung und Authentifizierung bedarf einer soliden Kenntnis in der Thematik. Ich empfehle daher unbedingt sich die Zeit zu nehmen und das folgende Video anzuschauen und sicher zu gehen, dass alles verstanden wurde:

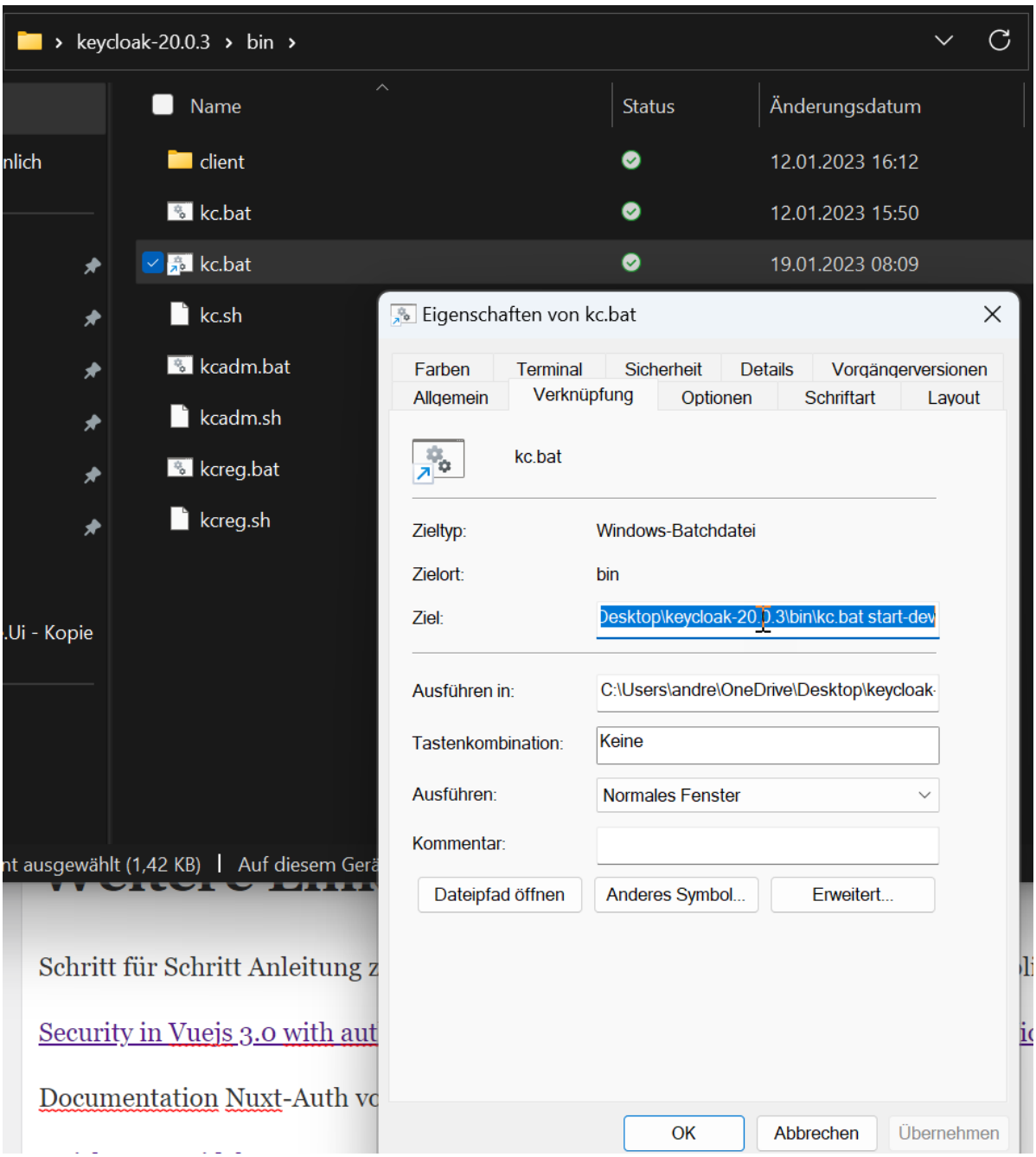
Zum Ende füge ich weitere Lohnenswerte Links hinzu.

1. Keycloak installieren

1. Keycloak herunterladen: downloads – Keycloak



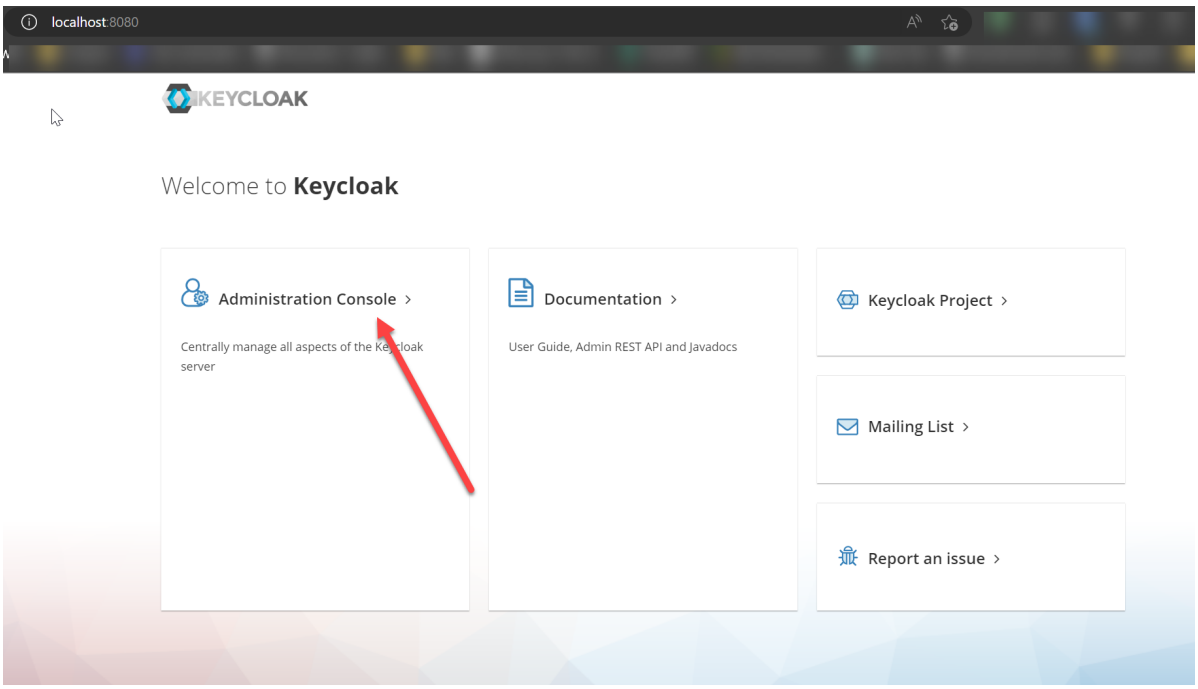
2. Zip-Datei entpacken und in das **bin** Verzeichnis wechseln. Dort eine Verknüpfung vom **kc.bat** erstellen, dann auf Eigenschaften und beim Ziel **start-dev** anhängen



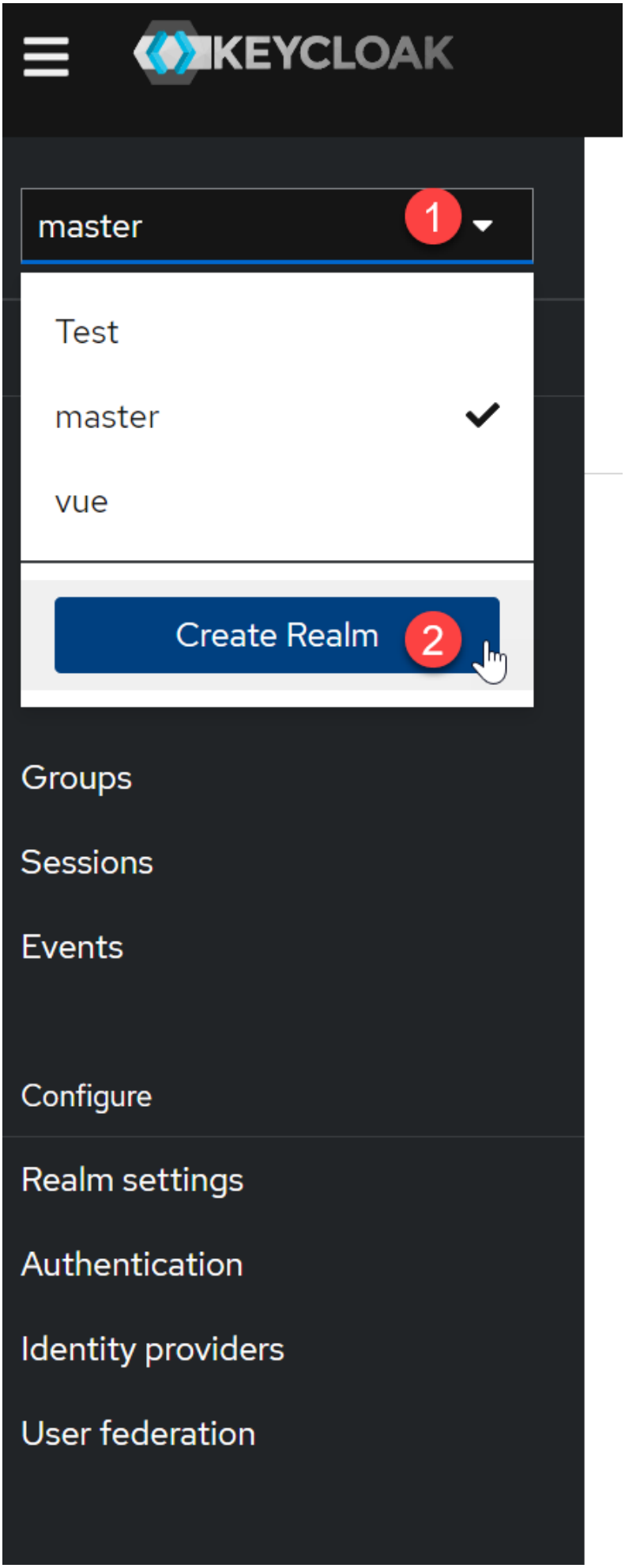
3. Verknüpfung starten und keycloak ist unter der Adresse http://localhost:8080 erreichbar. Dort einen neuen Admin User anlegen

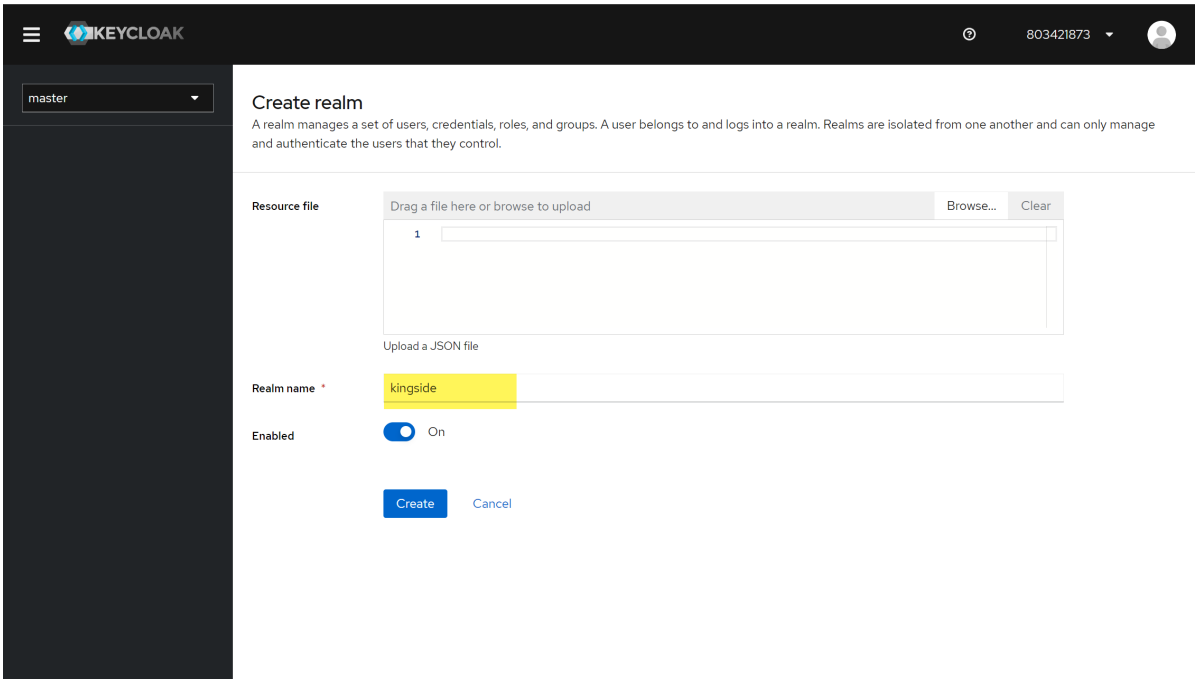
2. Keycloak einrichten

1. Keycloak Administarion console aufrufen über http://localhost:8080

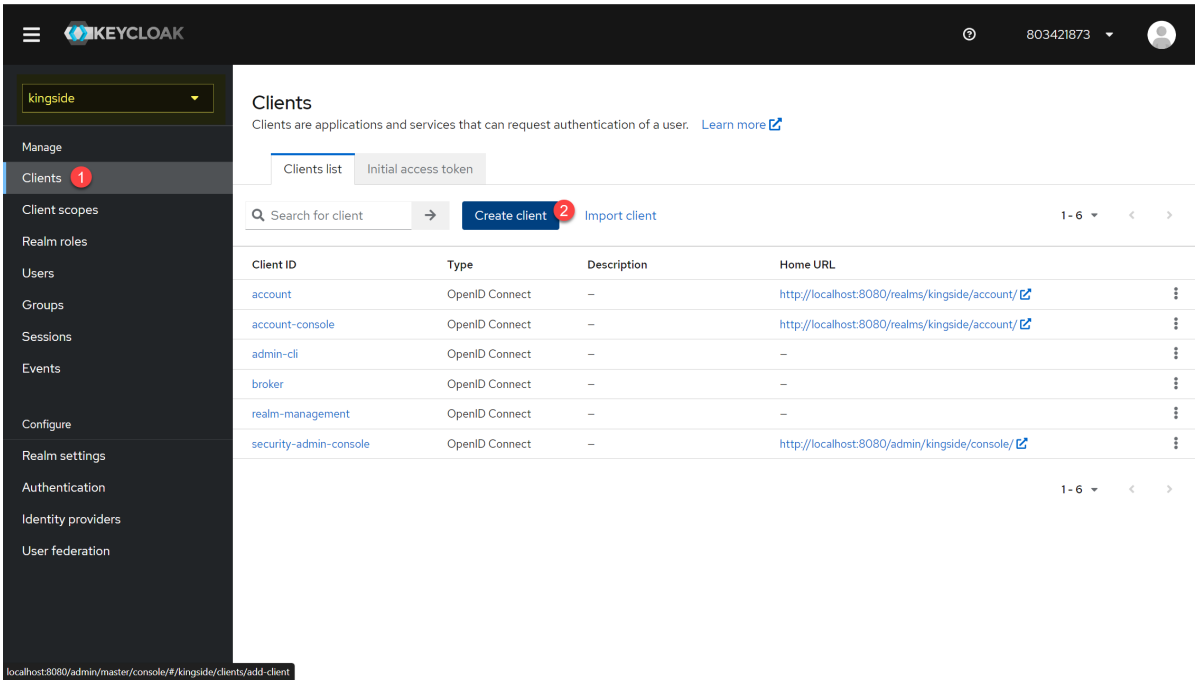


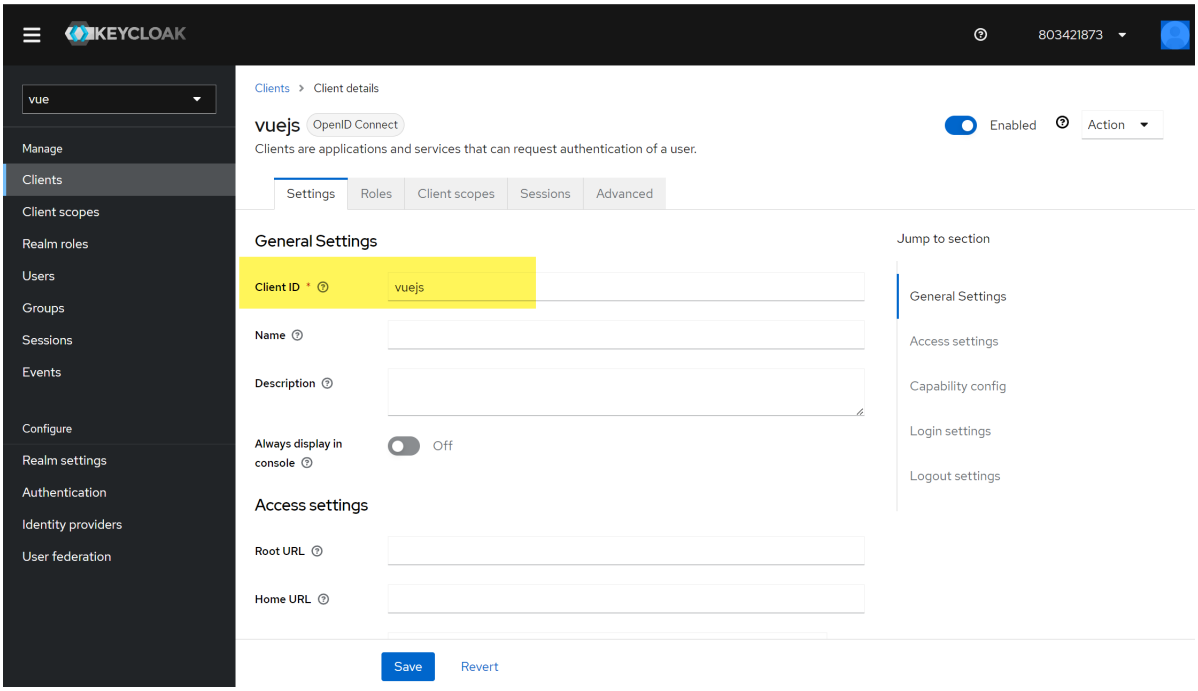
2. Neuen realm erstellen und nur **Realm name** eintragen. Wichtig! Keine Sonderzeichen benutzen

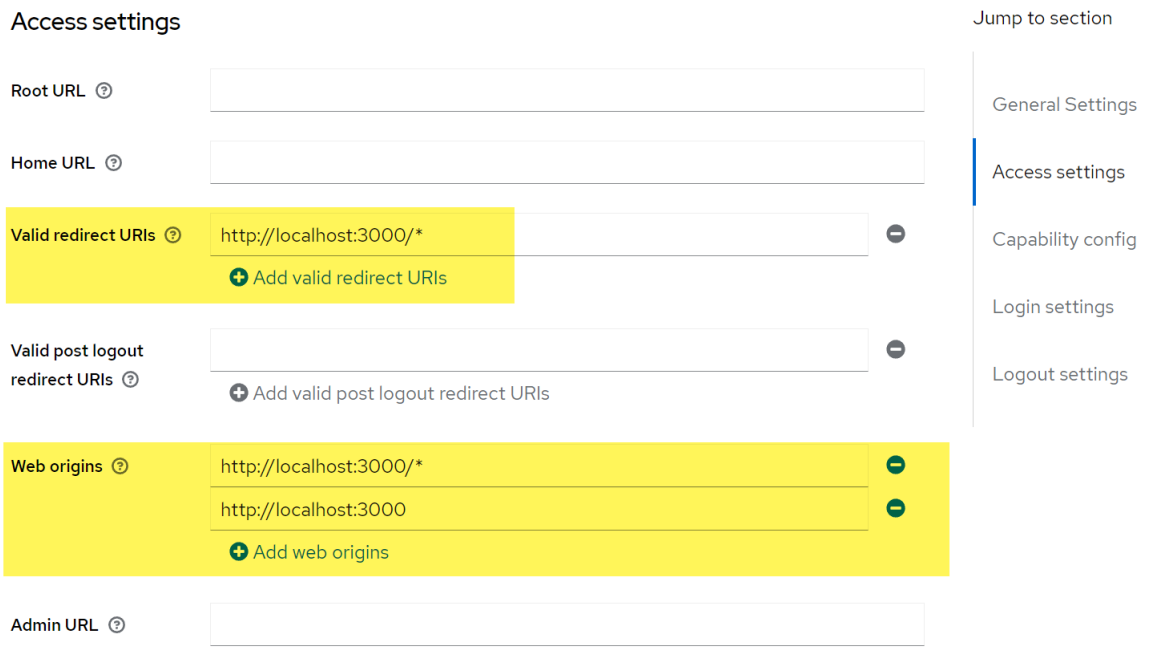

KEYCLOAK
master
1
Test
master
vue
Create Realm
2
Groups
Sessions
Events
Configure
Realm settings
Authentication
Identity providers
User federation



3. Jetzt in dem angelegten realm einen neuen Client erstellen




KEYCLOAK
803421873
vue
Manage
Clients
Client scopes
Realm roles
Users
Groups
Sessions
Events
Configure
Realm settings
Authentication
Identity providers
User federation
Clients > Client details
vuejs
OpenID Connect
Clients are applications and services that can request authentication of a user.
Enabled
Action
Settings
Roles
Client scopes
Sessions
Advanced
General Settings
Client ID *
vuejs
Name
Description
Always display in console
Off
Access settings
Root URL
Home URL
Save
Revert
Jump to section
General Settings
Access settings
Capability config
Login settings
Logout settings


Access settings
Root URL
Home URL
Valid redirect URIs
http://localhost:3000/*
Add valid redirect URIs
Valid post logout redirect URIs
Add valid post logout redirect URIs
Web origins
http://localhost:3000/*
http://localhost:3000
Add web origins
Admin URL
Jump to section
General Settings
Access settings
Capability config
Login settings
Logout settings

Capability config

Client authentication: Off

Authorization: Off

Authentication flow:
- [x] Standard flow
- [ ] Direct access grants
- [ ] Implicit flow
- [ ] Service accounts roles
- [ ] OAuth 2.0 Device Authorization Grant
- [ ] OIDC CIBA Grant

Login settings

Login theme: Choose...

Consent required: Off

Display client on screen: Off

Logout settings

Front channel logout: On

Front-channel logout URL:

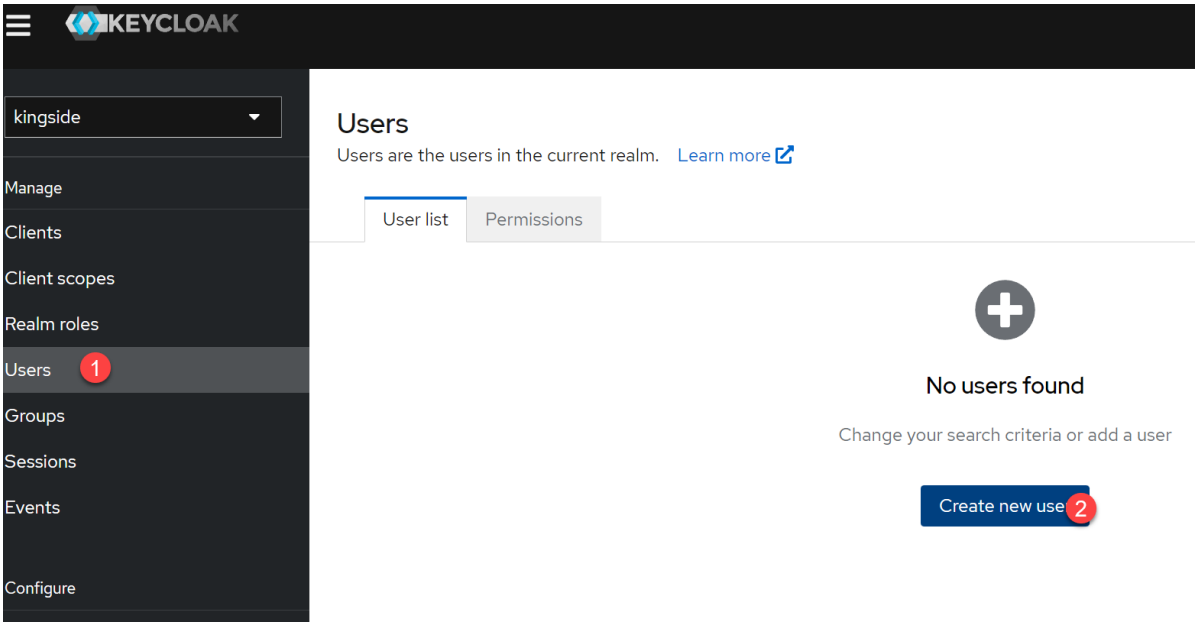
Backchannel logout URL:

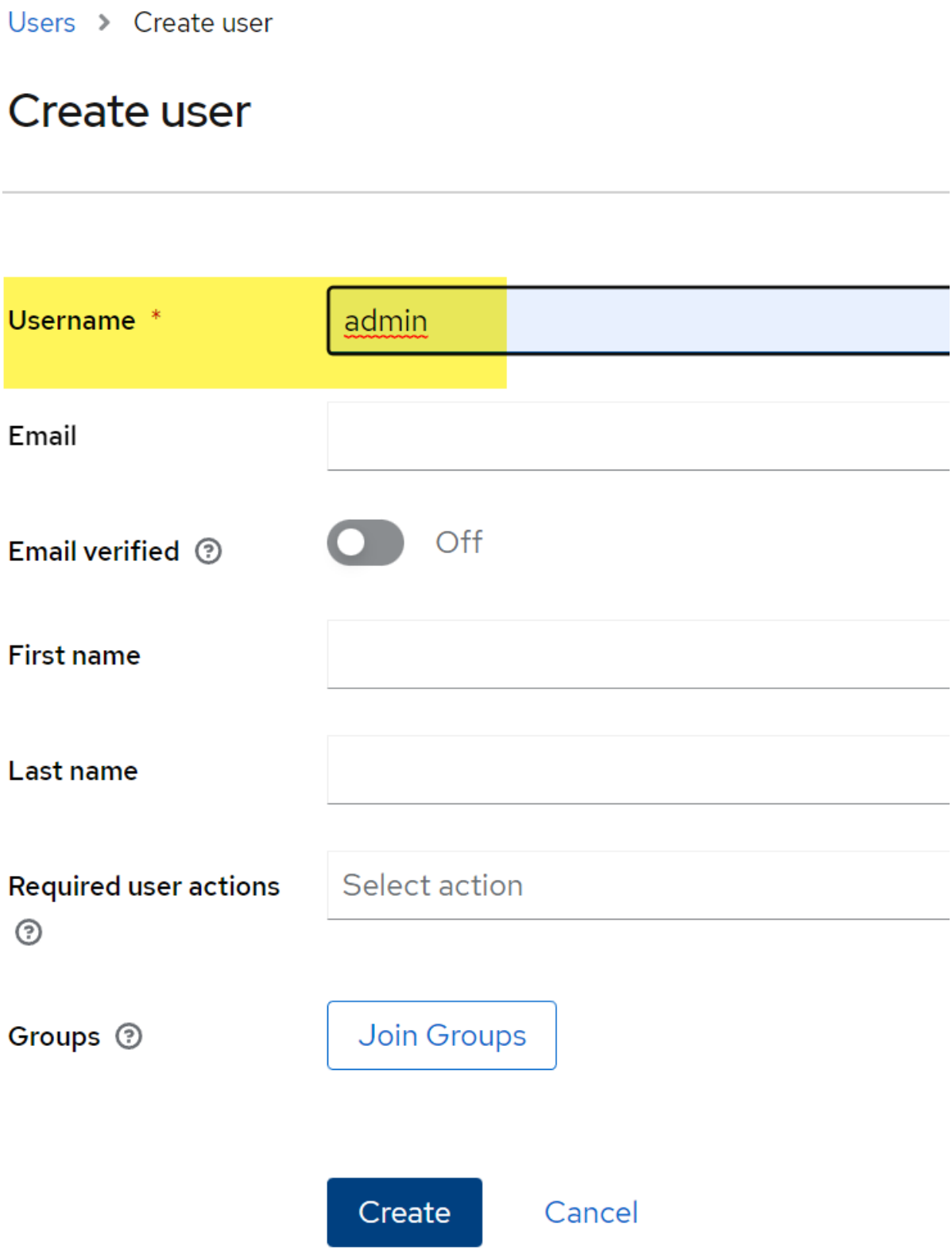
Backchannel logout session required: On

Backchannel logout revoke offline sessions: Off

Save Revert

4. User erstellen. Der **Username** reicht





5. Zu dem User wechseln und folgende Sachen um **Credentials** ergänzen

Users > User details

admin

Details | Attributes | Credentials | Role mappir

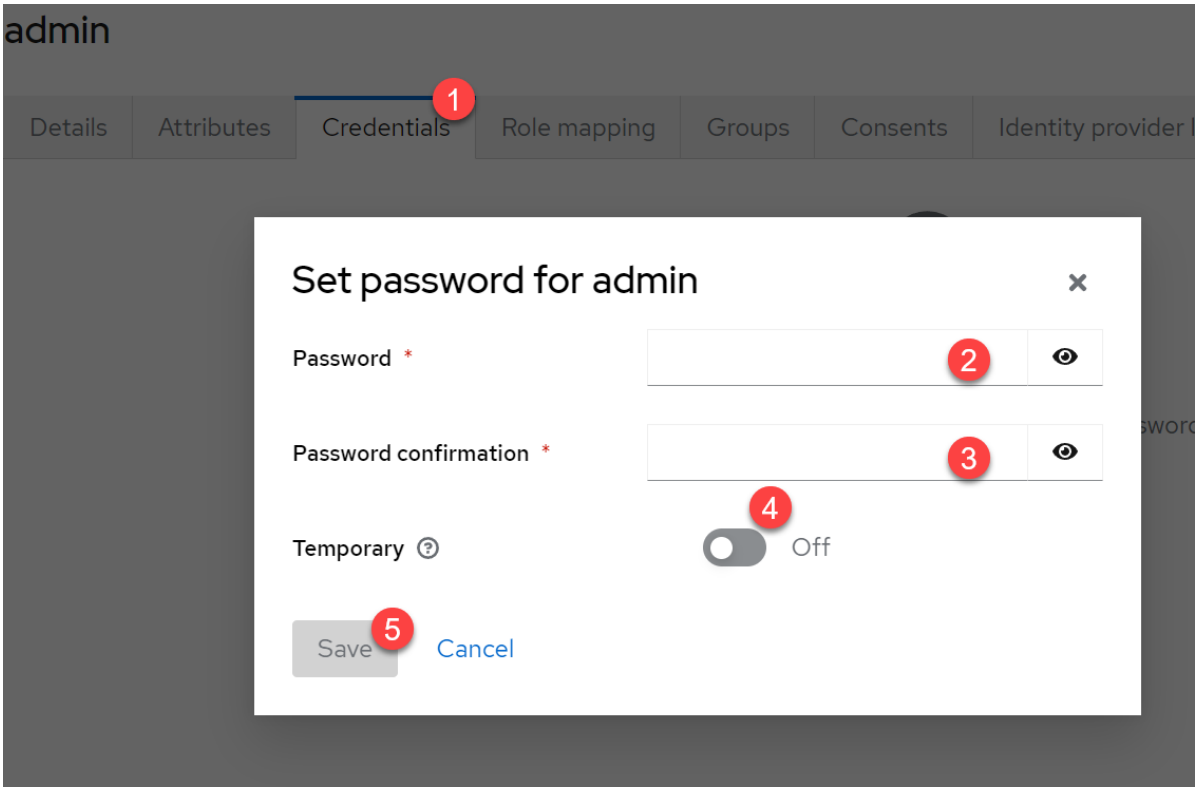
ID *: f6c242e6-cab5-4323-8d76-96

Created at *: 1/30/2023, 10:50:24 AM

Username *: admin

Email:

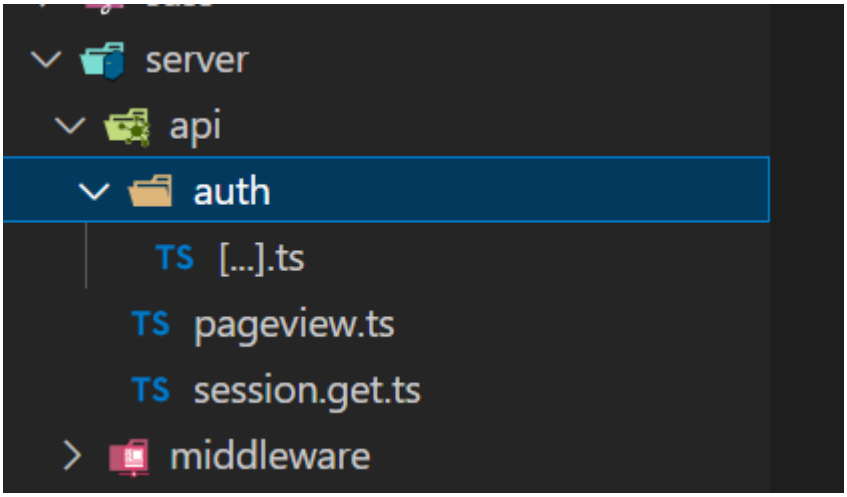
Email verified: On



3. nuxt 3 App aufsetzen

Installiertes nodeJs und pnpm sind Voraussetzung

1. Visual studio Code starten und das prime-vue starter repository clonen: https://github.com/sfxcode/nuxt3-primevue-starter.git
2. Verzeichnis auswählen wo die repo hin soll und mit **öffnen** bestätigen
3. Wir wechseln zum Terminal und installieren sidebase nuxt-auth. Zur Zeit des Schreibens existiert auch eine 0.4.0 alpha Version. Diese Version verursacht aber Fehler mit Keycloak.:
   [crayon-6767a1b9cba6a695010788/]
4. nun in der nuxt.config.ts die modules ergänzen und nuxt mitteilen, dass Autorisierung auf jeder Seite gelten soll:
   [crayon-6767a1b9cba6d754649660/]
5. Jetzt muss unter server 2 Dateien erstellt werden:



1. **server\auth\[…].ts**
   [crayon-6767a1b9cba6f575743499/]
   und 2. **server\api\session.get.ts**
   [crayon-6767a1b9cba70339928501/]
6. Eine weitere Datei muss angelegt werden für die Umgebungsvariablen: **.env**
   [crayon-6767a1b9cba72187066999/]
   Lasst euch ein Secret irgendwo generieren. Dieser String ist nur für Demozwecke!!!

   Wichtig ist, dass die lokale Adresse mit 127.0.0.1 angegeben wird. localhost oder 0.0.0.0 kann nicht aufgelöst werden. In den Url ist der real name enthalten.

Weitere Links

Schritt für Schritt Anleitung zur lokalen Einrichtung von Keycloak und einer Vue Applikation:

Security in Vuejs 3.0 with authentication and authorization by KeyCloak Part 1 | by Nicolas Barlatier | Dev Genius

Documentation Nuxt-Auth von Sidebase:

Quick Start · sidebase

Nuxt-Auth wurde von Next-Auth portiert. Fehlende Dokumentation kann hier nachgelesen werden:

Introduction | NextAuth.js (next-auth.js.org)

Ich nutze das prime-vue starter Paket, welches nuxt 3 inkludiert hat. Ich kann das Paket jeden nur empfehlen!

sfxcode/nuxt3-primevue-starter: Build your VUE.js App with Nuxt3 . First Class PrimeVUE support. Formkit Validation included. (github.com)

Für OAuth und OpenId Connectt gibt es Debugger. Der Author ist Nathon aus dem YouTube Video weiter oben.

OAuth 2.0 debugger (oauthdebugger.com)

OpenID Connect debugger (oidcdebugger.com)

Keycloak Dokumentation:

Documentation – Keycloak

Nuxt3 mit Vuetify 3 – Teil 3: Entwickeln in Vue / Typescript

Typescript muss in VS Code installiert werden. -g steht für globally (Alle Projekte)
[crayon-6767a1b9cbd1c308830424/]
Typescript Version prüfen
[crayon-6767a1b9cbd29263050817/]

Enums
[crayon-6767a1b9cbd2b125997360/]
Objekte werden mit reactive<type>({…}) reactiviert und primitive Datentypen mit ref<string>(…)
[crayon-6767a1b9cbd2c172063814/]
Mixins werden in eine Vue Instanz hinzugefügt. So als würde man 2 Vue Instanzen verschmelzen. Composables sind dem ähnlich. In Mixins wird alles importiert, bei Composables muss jede Variable, Methode, etc. einzeln importiert werden.

Guter Artikel dazu:

What is a Vue.js Composable? – Vue.js Tutorials (vueschool.io)

Nuxt3 mit Vuetify 3 – Teil 2: Layout

Wenn man durch eine App navigiert, dann ist der Seitenaufbau überall identisch. Das Menü, die Kopfzeile und Fusszeile sind auf jeder Seite identisch. Lediglich der Inhalt ändert sich.

In Nuxt 3 werden Layouts im „layouts“-Ordner erstellt und anschließend in die app.vue hinzugefügt. Einzelne Seiten werden im „pages“-Ordner erstellt.

Nuxt 3 routet die pages anhand des Namens. Nur die page „index.vue“ wird zum root Knoten geroutet.

Also erstellen wir ein Verzeichnis „layouts“ und ein Verzeichnis „pages“.

In das „layouts“ Verzeichnis, eine Datei erstellen „default.vue“.

und folgenden Inhalt kopieren:
[crayon-6767a1b9cbfcd644952182/]
Dort, wo <slot /> steht, kommt der Inhalt der einzelnen Pages hinein.

Jetzt in das Verzeichnis „pages“

die Datei index.vue mit folgenden Inhalt
[crayon-6767a1b9cbfd4376239877/]
und die Datei „pageOne.vue“ mit folgenden Inhalt erstellen
[crayon-6767a1b9cbfd8108931745/]
In der app.vue müssen wir nun sagen, dass wir nicht mehr die Nuxt Welcome page sehen wollen, sondern unser Layout:
[crayon-6767a1b9cbfdb354719715/]