Shanoir NG

Access and rights management

Table of contents

1 Introduction 1

2 Roles 1

3 User management 1

3.1 Users microservice 2

3.2 Keycloak 2

4 Rights management 2

5 Access management 3

6 ROLE\_ADMIN 3

# Introduction

This document defines the overall behaviour of Shanoir, concerning access and rights management. This includes which user roles exist, which role can see which content and do what in Shanoir.

# Roles

4 roles exist:

* ROLE\_ADMIN: system administrators
* ROLE\_EXPERT
* ROLE\_USER
* ROLE\_GUEST?

# User management

In Shanoir-NG users are managed in two systems: Keycloak and the users microservice. They interact together to manage the users of Shanoir-NG. In case of a migration the database of Shanoir-Old can be involved as well to get access to the current list of users within Shanoir-Old, users table.

1. **Keycloak:** open access and identity management tool (own docker container)  
   Keycloak manages the user, the login on using OpenID Connect and the user credentials to authenticate. Keycloak is as well aware of the user roles.
2. **Users microservice:** Shanoir-NG specific microservice (own docker container)  
   Contains tables: ***roles*** (for mapping), ***user\_account\_request\_infos*** and ***users***.  
   For *users* see below listing of attributes:

* id
* account\_request\_demand
* account\_request\_info\_id
* can\_access\_to\_dicom\_association
* creation\_date
* **email**
* expiration\_date
* extension\_date
* extension\_motivation
* extension\_request\_demand
* keycloak\_id
* first\_name
* first\_expiration\_notification\_sent
* second\_expiration\_notification\_sent
* last\_name
* last\_login
* team\_name
* **username**
* **role\_id**

## Users microservice

Receives an account request, generates a potential username and saves user at first in users table. Generates a potential password and creates user with password in keycloak.

User should be added to users table, but only in case of account request acceptance add into keycloak.

Rather than specifying a new password manually a better security practice is to use the PUT /admin/realms/{realm}/users/{id}/execute-actions-email admin call with "UPDATE\_PASSWORD" as the required action. This causes Keycloak to send an email to the user that gives a link for the user to set a new password.

KeycloakClient -> KeycloakShanoirUtil

PasswordUtils

Sends emails to the users. EmailService

## Keycloak

Handles forgot password. Therefore he needs to have smtp server configured to work.

How do new users come into keycloak?

Users in keycloak have an attribute userId, that links back to users ms, used in the token.

# Rights management

Todo: how to implement rel\_study\_user?

Today the following rights exist for a user in relation with each study: REF\_STUDY\_USER\_TYPE

* IS\_RESPONSIBLE\_FOR\_THE\_RESEARCH\_STUDY
* CAN\_SEE\_DOWNLOAD\_IMPORT\_DATASETS\_AND\_MODIFY\_THE\_STUDY\_PARAMETERS
* CAN\_SEE\_DOWNLOAD\_AND\_IMPORT\_DATASETS
* CANNOT\_SEE\_OR\_DOWNLOAD\_DATASETS
* CAN\_SEE\_AND\_DOWNLOAD\_DATASETS

# Access management

1. The "admin" user will have to be deleted. In Shanoir-NG we will implement a 100% tracing of user actions as this can become highly important for CNIL and other responsibilities. Therefore we need to be able to identify the human person behind each action done in Shanoir. Users (== 1 physical person) with "admin" role (ROLE\_ADMIN) will replace the "admin" user. This enables us to know precisely who did what in the system, that we can do audit tracing later.
2. Today each user (physical person) can only have ONE role.
3. Access to the keycloak admin interface MUST be blocked from outside access

# ROLE\_ADMIN

* Administrators, users with the ROLE\_ADMIN, can access to all contents in Shanoir
* They see all studies in Shanoir, to have an overview of the system
* Admin roles should be given access only from specific network addresses