Shanoir NG – User Managment

Software Design Description

Table des matières

[1 Microservice context 2](#_Toc472408750)

[1.1 Microservice presentation 2](#_Toc472408751)

[1.2 Main functionalities 2](#_Toc472408752)

[1.3 Application users 2](#_Toc472408753)

[2 Functional architecture 2](#_Toc472408754)

[2.1 User 2](#_Toc472408755)

[2.1.1 First name 2](#_Toc472408756)

[2.1.2 Last name 2](#_Toc472408757)

[2.1.3 Username – used also as a login (UNIQUE) 2](#_Toc472408758)

[2.1.4 E-mail – used also as a login (UNIQUE) 3](#_Toc472408759)

[2.1.5 Password 3](#_Toc472408760)

[2.1.6 Creation date 3](#_Toc472408761)

[2.1.7 Expiration date 3](#_Toc472408762)

[2.1.8 Role 3](#_Toc472408763)

[2.1.9 PACS Access (canAccessToDicomAssociation) 3](#_Toc472408764)

[2.1.10 Medical 3](#_Toc472408765)

[2.1.11 Team 4](#_Toc472408766)

[2.1.12 Is first expiration notification sent 4](#_Toc472408767)

[2.1.13 Is first expiration notification sent 4](#_Toc472408768)

[2.2 User account request 4](#_Toc472408769)

[2.2.1 Institution 4](#_Toc472408770)

[2.2.2 Service 4](#_Toc472408771)

[2.2.3 Function 4](#_Toc472408772)

[2.2.4 Study 4](#_Toc472408773)

[2.2.5 Contact 4](#_Toc472408774)

[2.2.6 Work 4](#_Toc472408775)

[2.3 Functionalities 4](#_Toc472408776)

[2.3.1 View user 4](#_Toc472408777)

[2.3.2 Create user 4](#_Toc472408778)

[2.3.3 Manage user account requests 5](#_Toc472408779)

[2.3.4 Manage expiration requests 5](#_Toc472408780)

[2.3.5 Delete user 5](#_Toc472408781)

[2.3.6 Update user 6](#_Toc472408782)

[3 Software/technical Architecture 6](#_Toc472408783)

[3.1 Interfaces 6](#_Toc472408784)

[3.1.1 Communication with other microservices 6](#_Toc472408785)

[3.1.2 Communication with Shanoir 6](#_Toc472408786)

# Microservice context

## Microservice presentation

This microservice is a part of the Shanoir-NG application and is responsible for user management and their requests.

## Main functionalities

Shanoir NG main functionalities are:

* Managing accounts (create, delete, update)
* Managing user’s role (user/guest/expert/admin)
* Managing user’s settings (i.e. is medical)
* Managing teams
* Validating users’ requests
* Sending global e-mails

## Application users

Target population is:

* Shanoir administrators
* Shanoir users of any level (could only update their own preferences)

# Functional architecture

## User

### First name

May contain letters, numbers, whitespaces, special characters, accents. Minimum 2. Maximum: 50. Mandatory

### Last name

May contain letters, numbers, whitespaces, special characters, accents. Maximum length: 50. Mandatory

### Username – used also as a login (UNIQUE)

May contain letters, numbers, no whitespaces, accents, special characters are allowed.

Must be unique. Minimum: 4, maximum:20

Automatically created from the first character of the first name + last name (unless already in use).

### E-mail – used also as a login (UNIQUE)

Must be unique.

Mandatory

### Password

Password must contain at least 1 letter, 1 number and 1 special character (%,$,#,@) and be at least of 8 characters long. Mandatory

### Creation date

Creation date of the account – visible but not modifiable in UI. Facultative and default to today’s date

### Expiration date

Expiration date of the account (the account becomes inactive).

Once inactive, user can send a request to extend this date (administrator accepts this with few clicks in **Manage user account requests** menu). Facultative but default to today +1 year

### Role

We define four roles of user account. One user can have only one role.

Role **Medical** disappears in new Shanoir. All users being Medical can be changed to User. Mandatory

#### Guest

A guest can only see the data. He can't create any data.

There are two possible configurations:

* Challenge – where Guests can download data
* Neuroinfo – where Guests cannot download data

#### User

This is the basic role in Shanoir. Set this role to the users that are not intended to be responsible for any research study and who don't have a strong knowledge of the Shanoir features. This type of user can download and import data if they are allowed to by the responsible of the research study of the data.

#### Expert

The expert role aims to be set to the users that might create research studies and study cards, and define the quality checks. The users with this role may also delete data.

#### Administrator

An administrator has the same rights as a medical doctor-user and can manage the users of Shanoir.

### PACS Access (canAccessToDicomAssociation)

Allows user to access PACS: Boolean. Facultative and default to No

### Medical

If user is medical: Boolean.

**Not used in the Shanoir-NG – used only for migrated data**

### Team

Team is a group of users that share the datasets belonging to at least one of them, i.e. if one user is the owner of the dataset, all users in the team can see it.

Team is facultative.

One user can belong to only one team.

Only admin can add/delete/modify teams.

### Is first expiration notification sent

This Boolean value is used to manage the user’s account expiration, default “0”.

Sent 30 days before the expiration of the account.

### Is first expiration notification sent

This Boolean value is used to manage the user’s account expiration, default null.

Sent 7 days before the expiration of the account.

## User account request

Created by person who doesn’t have any account (no authentication required). New account request creates an inactive user (account\_request\_demand = 1) and an account request in account\_request\_info linked to this user.

This user must be validated by admin and cannot log in till that time.

User form to fill in (users table):

* First Name - obligatory
* Last Name – obligatory
* Email – obligatory
* Expiration date – facultative
* Is medical – obligatory but default to No
* Can import from PACS – obligatory but default to No

Fields for account\_request\_info:

### Institution

User’s institution – obligatory (ex. Inria Rennes)

### Service

User’s service – obligatory (ex. Visages)

### Function

User’s function – obligatory (ex. PhD Student)

### Study

Study the user wants to have access to (ex. EMISEP). obligatory

### Contact

User’s contact in Shanoir – used in case of questions (ex. Anthony Hopkins). obligatory

### Work

“What would you like to do in Shanoir” -> to verify the suitable role for the user (ex. download and upload data, manage study). obligatory

After sending the request, a confirmation appears on the screen saying that “An email has been sent to the administrator who will validate your request. You will receive an email once your account activated.”

User is redirected to the main page.

## Functionalities

### View user

* Must be authenticated to see user’s data

### Create user

Creating a new user is possible by a:

* Person requesting to have an account. In this case the creation process has 2 steps:
  + Filling in the required fields (first name, last name, e-mail, password, institute, service, function, study, contact, comment and team for shanoir-challenges)
  + Accepting the user request by the administrator and filling in the other fields (role, pacs import, team). The account is now active.
* Administrator directly in the UI using the function “Create new user” and filling in all the fields manually, including password

### Manage user account requests

Available for administrators only.

Inactive users (with flag account\_request\_demand =1) whose account must yet be activated appear on the top of the list of users. After clicking on Edit user button, two actions are possible:

#### Accept user

To do that, the administrator must first choose a role and then click on Accept button. account\_request\_demand passes to 0.

#### Deny user

By clicking on Deny button. The user and his account request info object is deleted from the database.

### Manage expiration requests

Once a notification is sent (first or second) the user has the possibility to ask for an extension of his account.

#### Requesting new expiration date by user

* The button *Request an extension* appears on the main page.
* Once clicked, a new window appears that allows to choose *new expiration date* (by default it is the same date) and update the account\_request\_info object (*institution, service, function, study, contact, work*)
* Button *Request an extension* validates the form and sends the information to the administrator
* Confirmation is shown: *An email has been sent to the administrator who will validate your account extension request. You will receive an email once your account extended.*
* This button doesn’t disappear. Next request will only update the same request (in Shanoir old there are two) until the request in treated.

#### Accepting the request by administrator

* Only an administrator can accept or reject the extension of the account.
* On the users list it is possible to filter the users and show only extension requests.
* By clicking on the Edit button it is possible to accept (button *Accept*) or deny the request (button *Deny extension*) as well as modifying other fields (role, expiration date etc).
* In both cases, an email is sent to inform the user that the extension has been granted or rejected with a new (=accept) or old (=deny) expiration date.

### Delete user

This action is reserved for the administrators. In the Administration panel he chooses the user to delete and confirms his choice.

This option appears next to the Edit button and requires confirmation (pop-up).

User is deleted from the database along with his account request info object.

### Update user

Fields updatable by user (for all the users except for Guests) are:

* first name
* last name
* e-mail
* password

Guest accounts are not allowed to edit any field.

Administrator can update all the fields on the Edit page.

# Software/technical Architecture

## Interfaces

### Communication with other microservices

User management microservices uses web services and message queues.

Bla bla bla

### Communication with Shanoir

Every time a request is sent from Shanoir, it arrives via MQ to Shanoir-NG.