*What is the main goal of the feature?*

The main goal of milestone 8 was to add the ability to create new user (Client, Artist, Admin). Add ability to show a list of users, add ability to show or hide fake user in system.

*What are the feature inputs?*

* http://<server>/administration/users/list.
* Acted database.
* Only users with ADMIN role can access that feature

*What are the feature outputs?*

1. add new migration where add new field “fake” to User model

- **app/DoctrineMigrations/Version20160725182702.php**

1. change “fake” to TRUE if User.id < 300 in migration file

- **app/DoctrineMigrations/Version20160725182702.php**

1. Add new parameter for show|hide fake users (**fake\_users**)  
   **-app/config/parameters.yml**
2. Add new parameter for resend configure time of expire confirmation token (**confirmation\_period\_resend**)

**-app/config/parameters.yml**

1. Add new parameter for configure time of expire confirmation token (**confirmation\_period**)

**-app/config/parameters.yml**

1. Add ability for resend confirmation token for users

-s**rc/ActedLegalDocsBundle/Cotroller/AdminController.php::resendConfirmationTokenAction()**

1. Add list of user to admin part and ability for manage all user in system

-s**rc/ActedLegalDocsBundle/Cotroller/AdminController.php::userListAction()**

-s**rc/ActedLegalDocsBundle/Cotroller/AdminController.php::createNewUserAction()**

-s**rc/ActedLegalDocsBundle/Resources/view/Security/create\_user\_confirmation.txt.twig**

1. Add **console** script for create, edit, delete users with ADMIN role

-s**rc/ActedLegalDocsBundle/Command/CreateUserCommand.php**

1. Add remove user ability

-s**rc/ActedLegalDocsBundle/Cotroller/AdminController.php::deleteUserAction()**

1. Add ability for activate or deactivate user

-s**rc/ActedLegalDocsBundle/Cotroller/AdminController.php::changeStatusAction()**

1. Add filters to user list page

-s**rc/ActedLegalDocsBundle/Repository/UserRepository.php::getUsersList()**

*What are the main business rules implemented?*

Add ability for shoving and manage users in system. Create,edit,delete new admin. This is restricted to users with ADMIN role.

A console application is required for Creating new users with ADMIN role

*Define a quick “How it works” schema*

We can login with ADMIN role and change user data - After that we should see changes in user profile.

Admin can create, edit, delete performance, change status of performance and other abilities for any artist in system.

We can also hide fake users created initially with Fixtures so that the do not apprear in search page, spotlight, recommended results.

This is done with a parameter and is enabling/disabling Fake users in the entire application.

*Which bundle handles the feature?*

not new bundle has added

*What are the main classes?*

**src/ActedLegalDocsBundle/Cotroller/AdminController.php** - work with admin dashboard

*What are the services?*

not

*What are the Twig templates?*

1. src/Acted/LegalDocsBundle/Resources/views/Admin/usersList.html.twig
2. src/Acted/LegalDocsBundle/Resources/views/Admin/createUserAdmin.html.twig

*What are the add-on bundle you installed?*

not

*What are the commands to run to install the feature?*

**php app/console do:mi:mi (executes migration of the Database)**

**php app/console assets:install (install assets not part of the Mockup, essentially for the Admin page)**

**php app/console ca:cl --env=prod**

*What are the commands to run to test the feature?*

We do not have tests for this feature.

*How to test the feature?*

For creating and managing users with ADMIN role, the following command is to be used:

EXAMPLE:

**Creation** of new user with ADMIN rolephp app/console create:admin [example@acted.com](mailto:example@acted.com) **create** Xs123456

**Editing** an existing user and changing his email address

*php app/console create:admin* [*example22@acted.com*](mailto:example@acted.com)***edit*** *Xs123457* [*example@acted.com*](mailto:example@acted.com)

in this example we set:  
 New email = [*example22@acted.com*](mailto:example@acted.com)  
 password = *Xs123457  
for user with use email* [*example@acted.com*](mailto:example@acted.com)

Deleting a user with ADMIN role

php app/console create:admin [example@acted.com](mailto:example@acted.com) **delete**

*~~php app/console create:admin~~* ***~~email~~****~~(required)~~* ***~~flag~~****~~(required)~~* ***~~password~~****~~(optional)~~* ***~~search\_email~~****~~(optional)~~*

php app/console create:admin **PARAM1**(required) **PARAM2**(required) **PARAM3**(optional) **PARAM4**(optional)

**PARAM1**: the **Email** address of the user with ADMIN role.

This is new user Email, or the current Email of user if PARAM4 is used

**PARAM2**: the **Function** to be executed

Can be create, edit or delete (please mention if this is case sensitive or not)

**PARAM3**: the user **Password**

This is required for create and edit

This is not required for delete

**PARAM4**: the existing user **Email**

This is required for edit only, if the email address is to be changed.

This is not required for create and delete