

Fullstack Project

Objectif:

This project facilitates profile management. With this application it is possible to create a user type (administrator, commercial ...), to create a user, to list the user type and the users.

Tools:

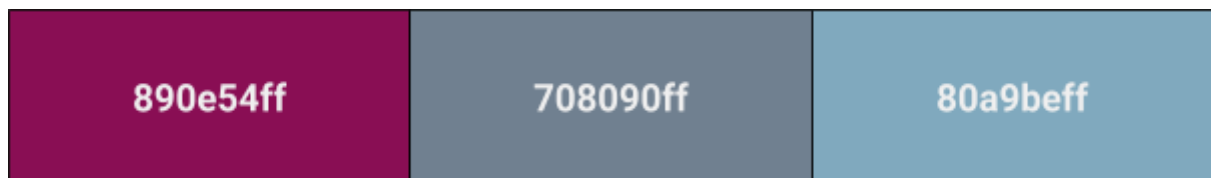
To realize this fullstack project you will use docker and docker compose, angular and spring boot.

Create a working environment with docker and docker-compose to carry out this fullstack project. You will use spring boot for the backend part and angular for the frontend part.

Graphic design:

The design of this project is free. You are not obliged to respect the design proposed in these specifications. However it is important to make a clean design.

If you want to be inspired by the proposed design, here are the color codes:



The project

this project consists of 5 pages:

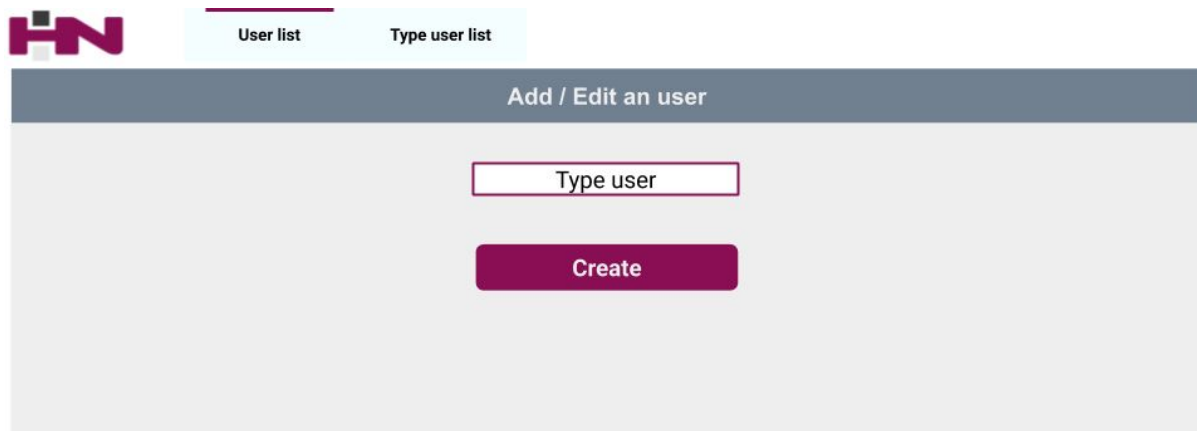
- Type user form
- Type user list
- User form
- User list
-

You can also create a home page according to your choice. But you can use one of the lists (type user or user) as home page

The menu :

the menu is made up of 2 links, one to go to the list of user types and another for the user list

Type user form :

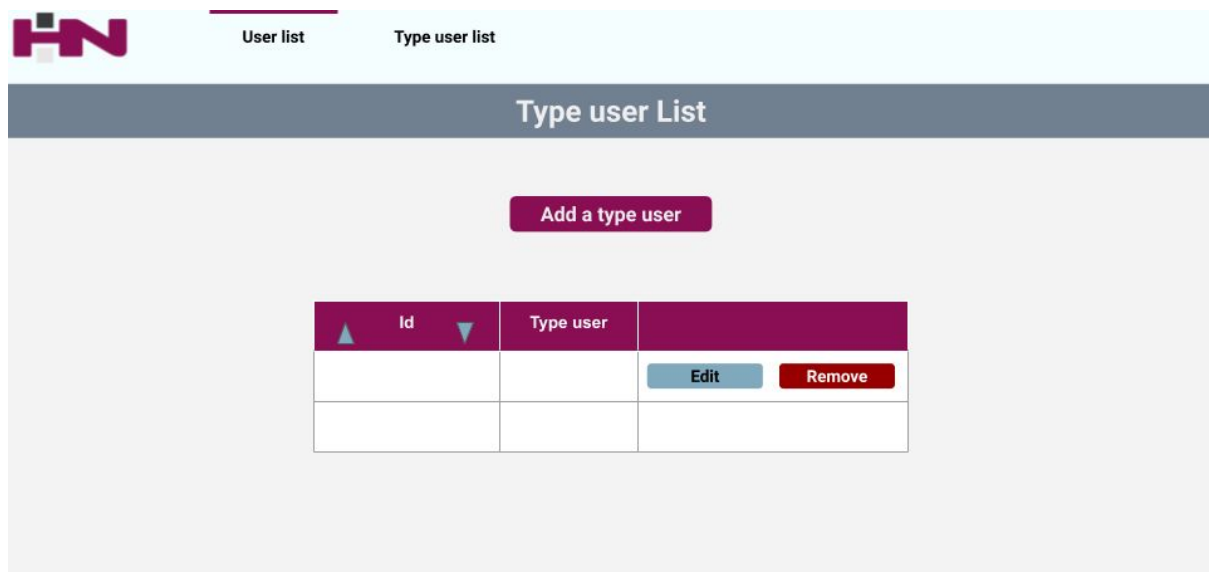


The screenshot shows a web interface with a header bar containing the 'HN' logo and two tabs: 'User list' and 'Type user list'. Below the tabs is a dark blue header with the text 'Add / Edit an user'. The main content area is light gray and contains a text input field labeled 'Type user' and a red 'Create' button below it.

This page contains a form with a single field (text type). It allows the addition of a user type (Administrator, commercial ...) in the database.

The form for editing a user type is the same, however the field will be pre-filled with the type of user to modify.

Type user list :



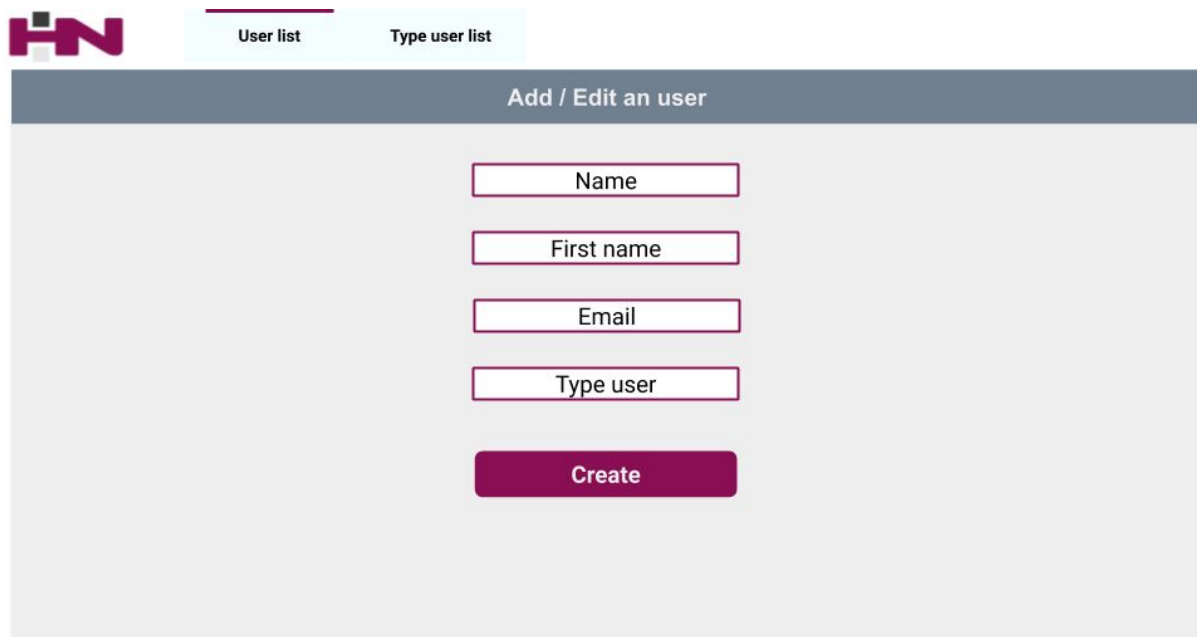
The screenshot shows a web interface with a header bar containing the 'HN' logo and two tabs: 'User list' and 'Type user list'. Below the tabs is a dark blue header with the text 'Type user List'. The main content area is light gray and contains a red 'Add a type user' button. Below the button is a table with three columns: 'Id', 'Type user', and an action column. The 'Id' column has a small upward arrow on the left and a downward arrow on the right. The 'Type user' column is empty. The action column contains two buttons: 'Edit' (light blue) and 'Remove' (red).

Id	Type user	
		Edit Remove

This page simply lists all the types of users of the database. A delete button allows you to delete a user type. the Edit button allows you to modify a user type.

This table can be sorted with the different parameters (id or type). The arrows are an example to sort the table.

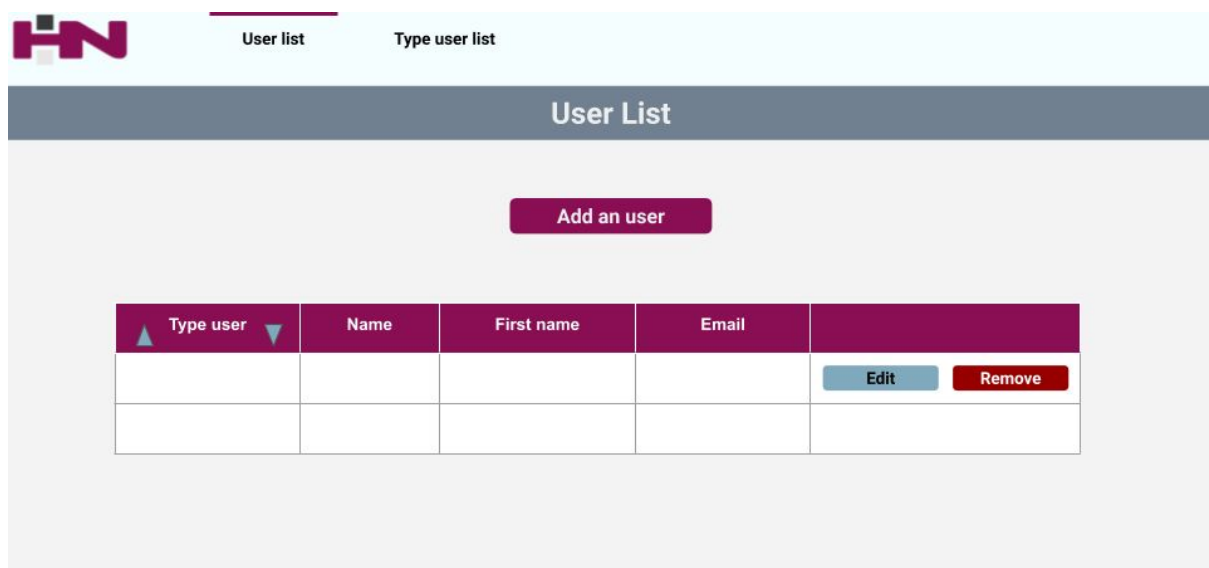
User Form :



The User Form interface features a header with the 'HN' logo and two tabs: 'User list' (active) and 'Type user list'. Below the tabs is a dark grey bar with the text 'Add / Edit an user'. The main form area contains four text input fields stacked vertically, labeled 'Name', 'First name', 'Email', and 'Type user'. At the bottom of the form is a red 'Create' button.

This page contains a form with a text field for the user's first and last name. A user list allows you to select a type of user contained in the database. The form for editing a user is the same, however the field are pre-filled.

User list :



The User list interface features a header with the 'HN' logo and two tabs: 'User list' (active) and 'Type user list'. Below the tabs is a dark grey bar with the text 'User List'. In the center of the page is a red 'Add an user' button. Below this is a table with five columns: 'Type user', 'Name', 'First name', 'Email', and an empty column. The 'Type user' column has a dropdown arrow. The 'Email' column has 'Edit' and 'Remove' buttons. The table has two empty rows below the header.

Type user	Name	First name	Email	
				Edit Remove

This page lists the users of the database. a delete button allows you to delete a user from the database. The edit button allows you to modify a user. To access a profile of a user, you can create a link in this table by clicking on the name for example. You also have the freedom to create a button to the profile. This table can be sorted with the different parameters (type user, name...)

At the spring boot level, two objects will have to be created: user and user type. Your database will thus be made up of two tables, with a foreign key linking the user and the type of user. It will be necessary to create the functions to add, retrieve, edit and delete a user and a user type.

Create a folder containing the front (angular) and the back (spring boot) with a docker compose file.