Compulsory Laboratory 4: Designing a Login and Registration System using Redis

- This lab will be evaluated and averaged for calculating your final score for the lab part.
- This lab corresponds to 25% of the final score of the labs.
- This lab has to be submitted before Tuesday**, 3 of May at 23:00.** And no assignments will be accepted after Sunday, **8 of May at 23:00.**
- Submissions after that day will be penalized with 10% per day.

The assignment consists of creating an online application using Node, Express, Bycript and Redis. The application has to allow users to register and use a password. Once they are registered they can log in to the system and perform store in the system their personal data.

The system by implementing Redis has to remember the session so the users do not have to log again into the system. Since it is not safe to store the passwords of the users in plain text, it would be best to store the passwords hashed using Bycript. Therefore, the hashed password, rather than the password is what is stored in the system. The password has to have 10 symbols and it is required to have at least one number, one Upper letter and one symbol between [$,%,£,&,@]. You can validate this using regex. The name of the user can only be letters and numbers. You can also use regex for this purpose.

When the users register he/she will have access to a screen with the following fields: Date of Birth, City of Birth, Email, Address, Gender, Hobbies, Civil state, Profession, Salary per year, upload a picture, and favourite sport. **If you prefer to allow the user to fill those values after registering with the user and password, that option is also fine.** (All values should have a default value). Once the user logs in to the system, it will access a section where those values will be displayed. In that new panel, the user can either modify those values or log out of the system.

In order to do so, we need to use the following technologies:

- MongoDB to store the users.
- Node to create the server where the website is stored.
- Redis to manage the sessions with the users.
- Bootstrap and JQuery for the front-end
- Bycript for creating a hash of the passwords
- Use Regex to validate that the user is only letters and/or numbers.

Please, create a document (Preferably in PowerPoint) of your application with screenshots of your application for all the steps the user has to go through. And two additional diagrams: First, the structure of your application (MongoDB, database, Redis, Bycript, Regex, JQuery, Bootstrap). And second, a diagram of the steps between the interaction of the client (the user) and the server (the platform).

The assignment will be evaluated as follows:

- Development of the server with Node.js and Bootstrap (2 pts).
- Implementation of the sessions with Redis (2 pts).

- Implementation of the password using Bycript (2 pts).
- Allow the user to modify the values (1 pts)
- Quality of the two diagrams explaining the process with screenshots of the steps (2 pts).
- Specification with screenshots of which parts have been accomplished and which have not (1 pts).

References:

- Login and Registration System Using Node MongoDB and Redis
- A quick way for hashing passwords using Bcrypt with Nodejs.
- Previous post Hashing Passwords with Node.js and bcrypt
- Bcrypt-cache
- Redis Caching in Node.js
- Authentication and Authorization using Redis | by Pankaj Panigrahi
- https://www.youtube.com/watch?v=9S-mphgE5fA