**Content**

1. Overview
2. Development Dependencies
3. Work flow
4. Authentication and Authorization
5. DB design

6. Things to do

1. Overview

Due to COIVD-19 pandemic, “Challa-ahoy”, a dining restaurant needs a visitor registering system for anyone that visit their restaurant. Based on such requirements, the project has implemented a web application system that allows visitor to register their:

- First name and last name

- Email address

- Phone numbers

- Time of the visit

It also requires a page for administrator to view the visitor list on any given time and to generate visitor report.

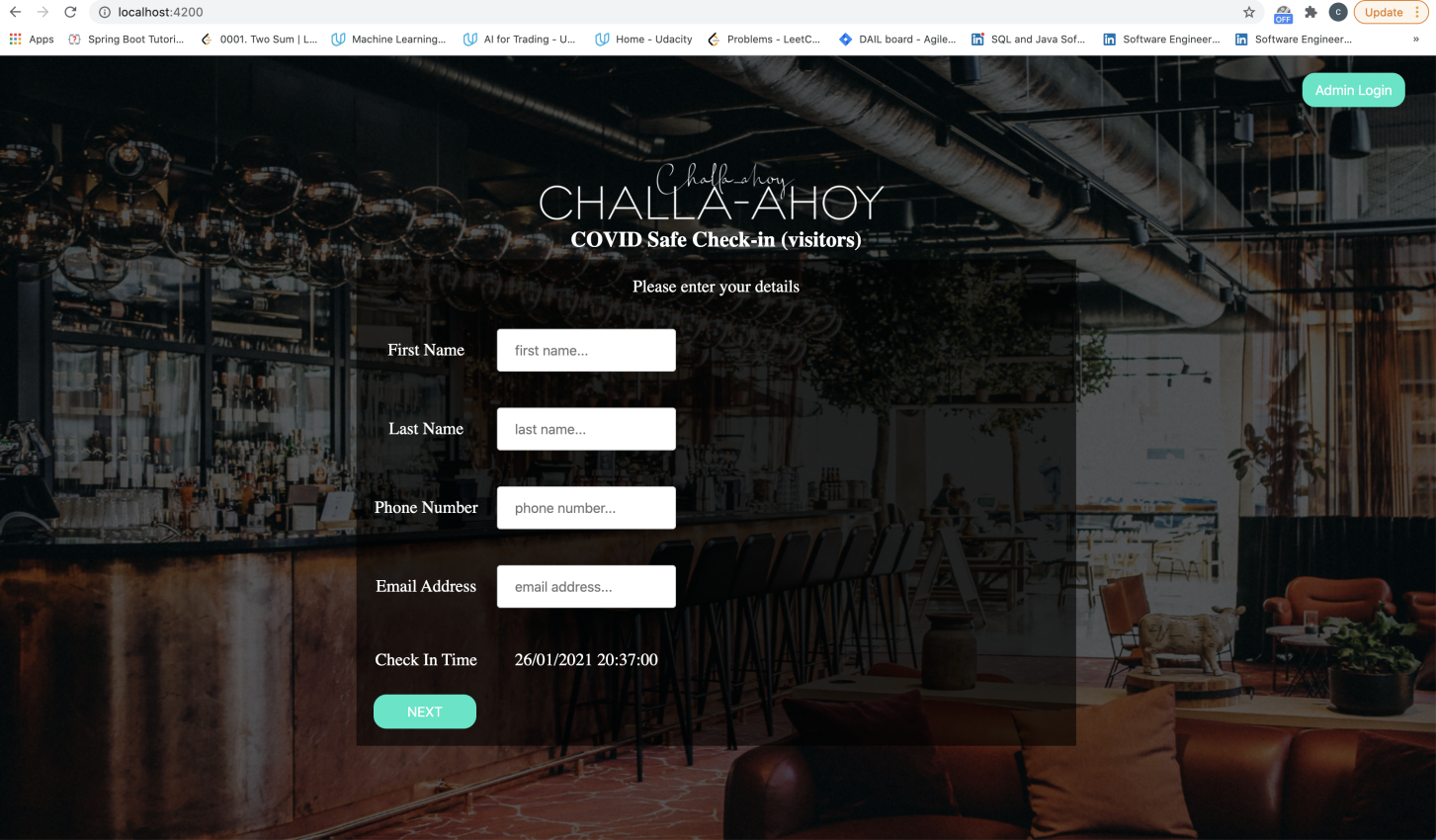
2. Development Dependencies

The project uses Angular (Authentication guard) as front-end, SprintBoot (JWT, JPA, Spring Security) as back-end MongoDB as database. Admin needs to login the portal before they can view the visitor data list and generate visitor report.

**Admin login username:*hello* and password:*1234567*.** Please contact tech support to create admin username and password in database as there is no signup function provided at the moment.

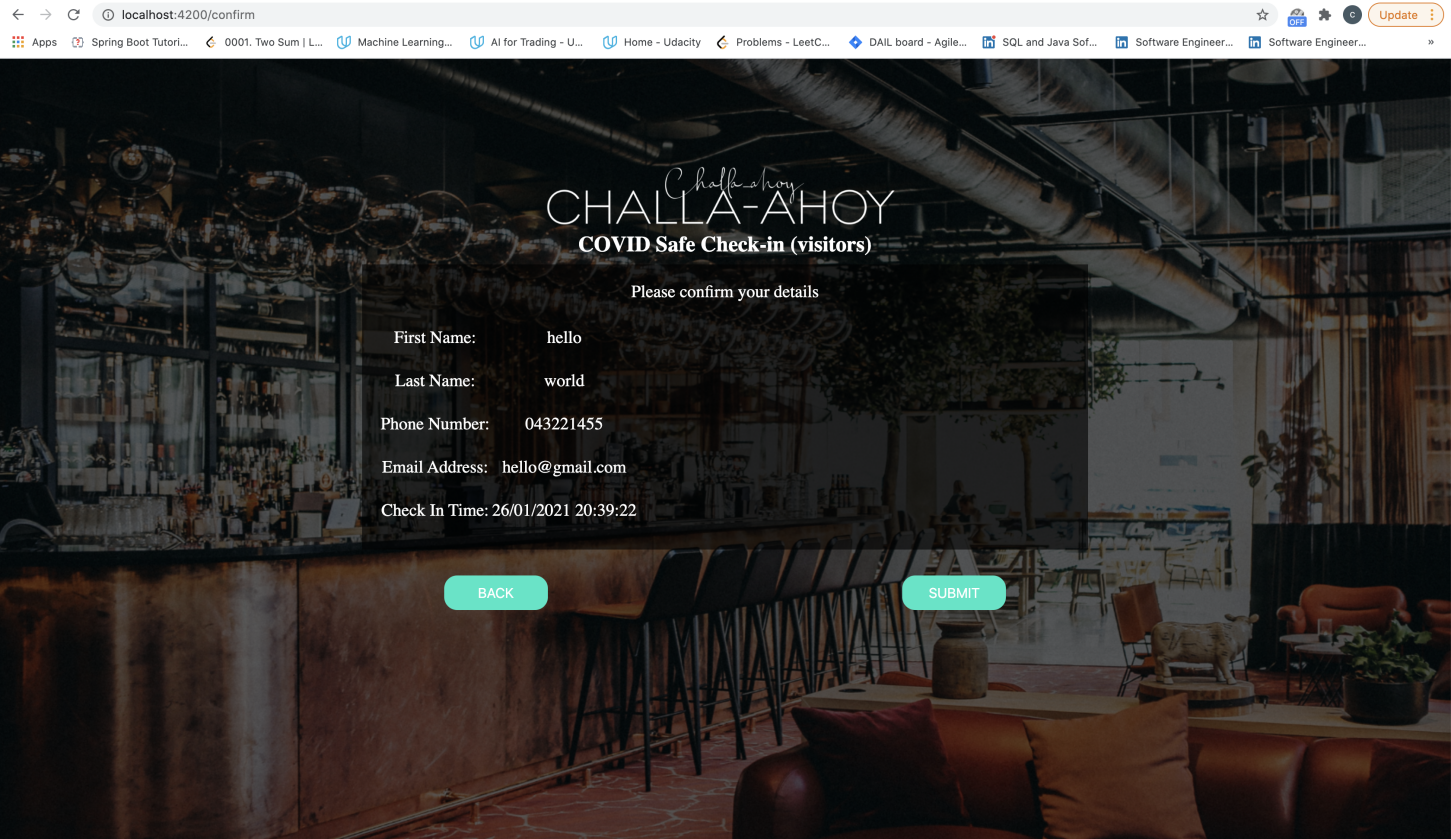
1. Work flow

3.1.Visitor Register



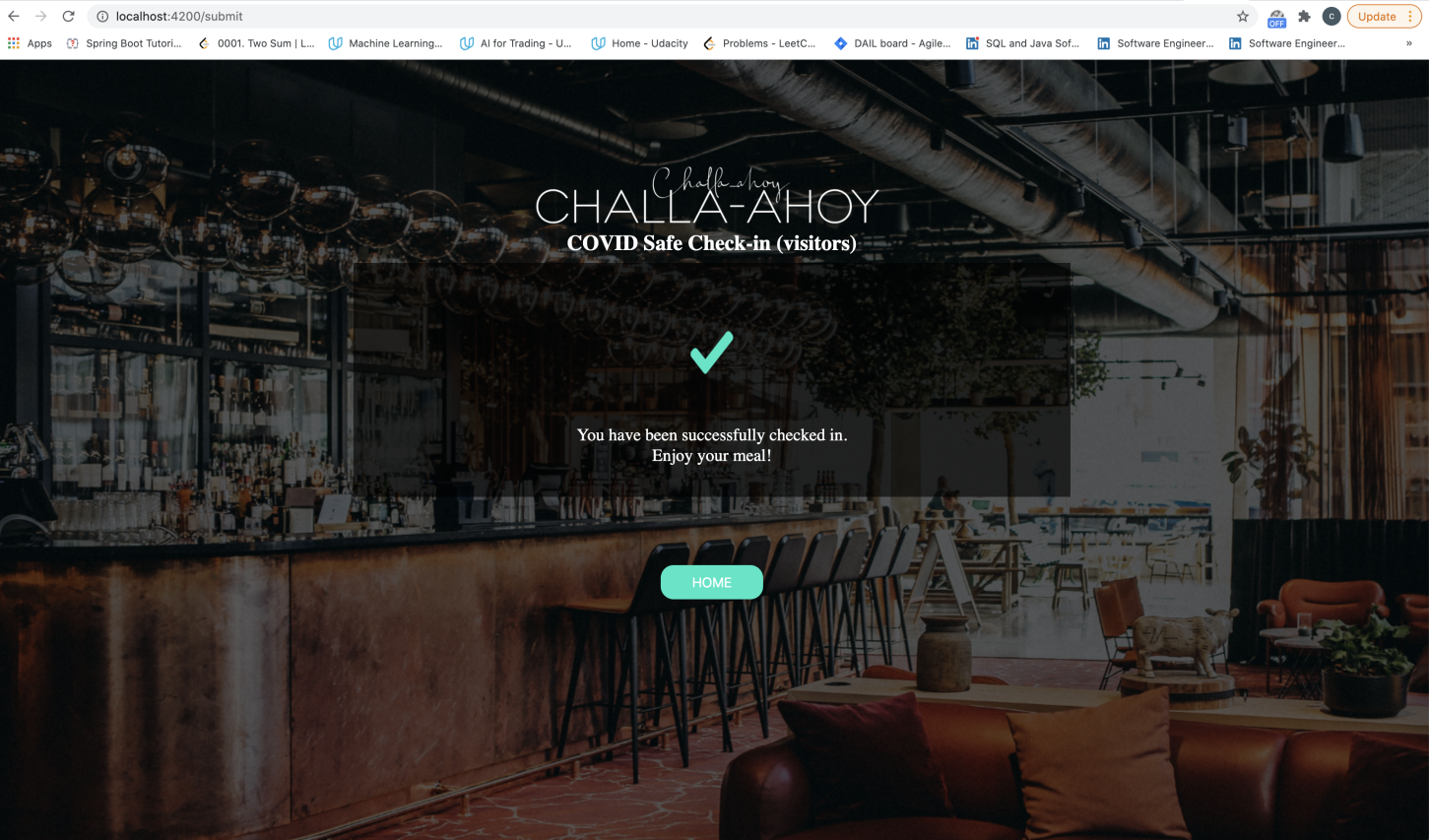
3.2 Visitor Register Confirm

After visitor fills in the register form in register page (3.1), clicking “NEXT” button will forward to this confirm page.



3.3 Visitor Info Submit

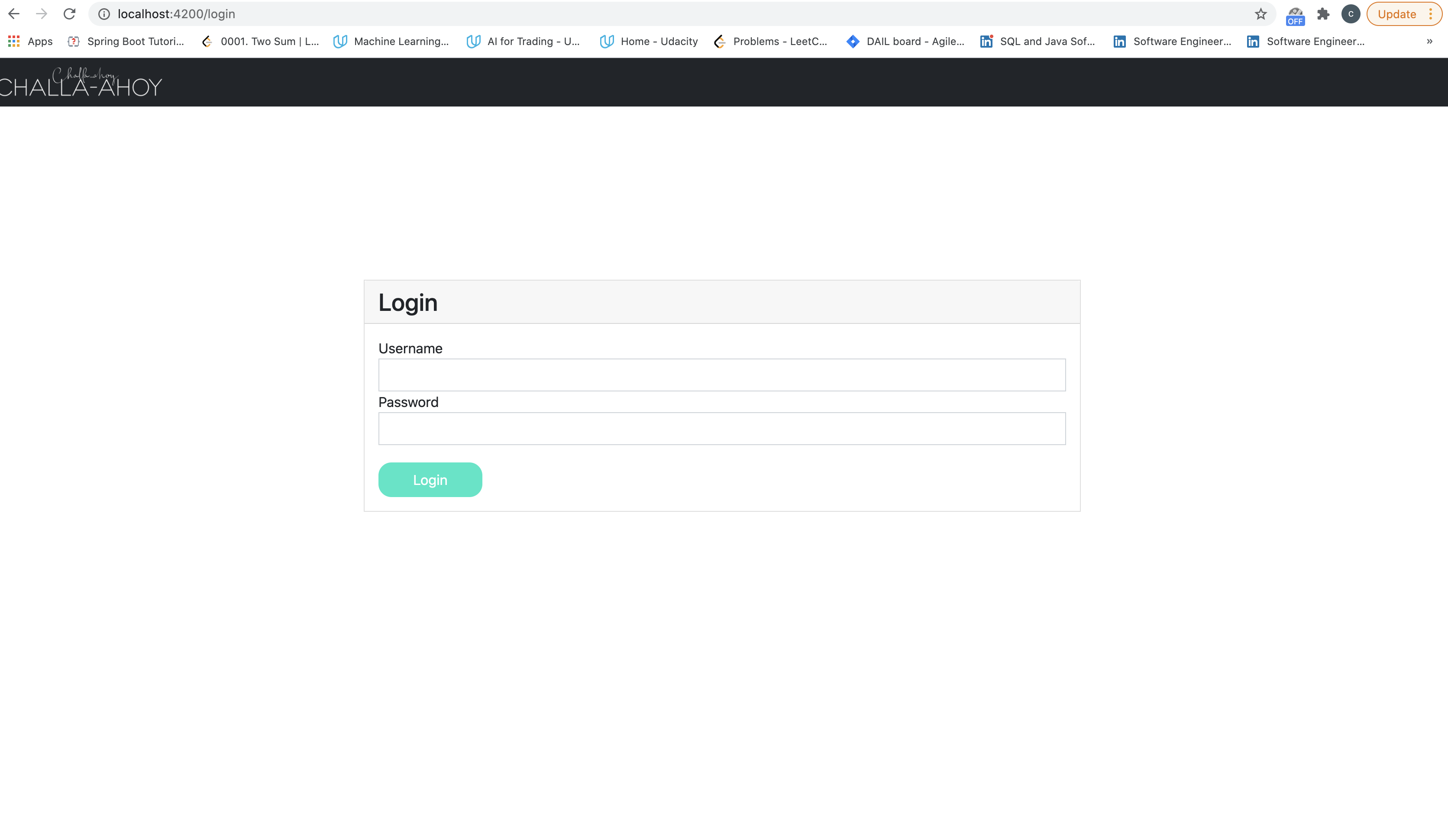
After clicking “SUBMIT” button in confirm page (3.2), it forwards to submit page.



3.4 Admin Login

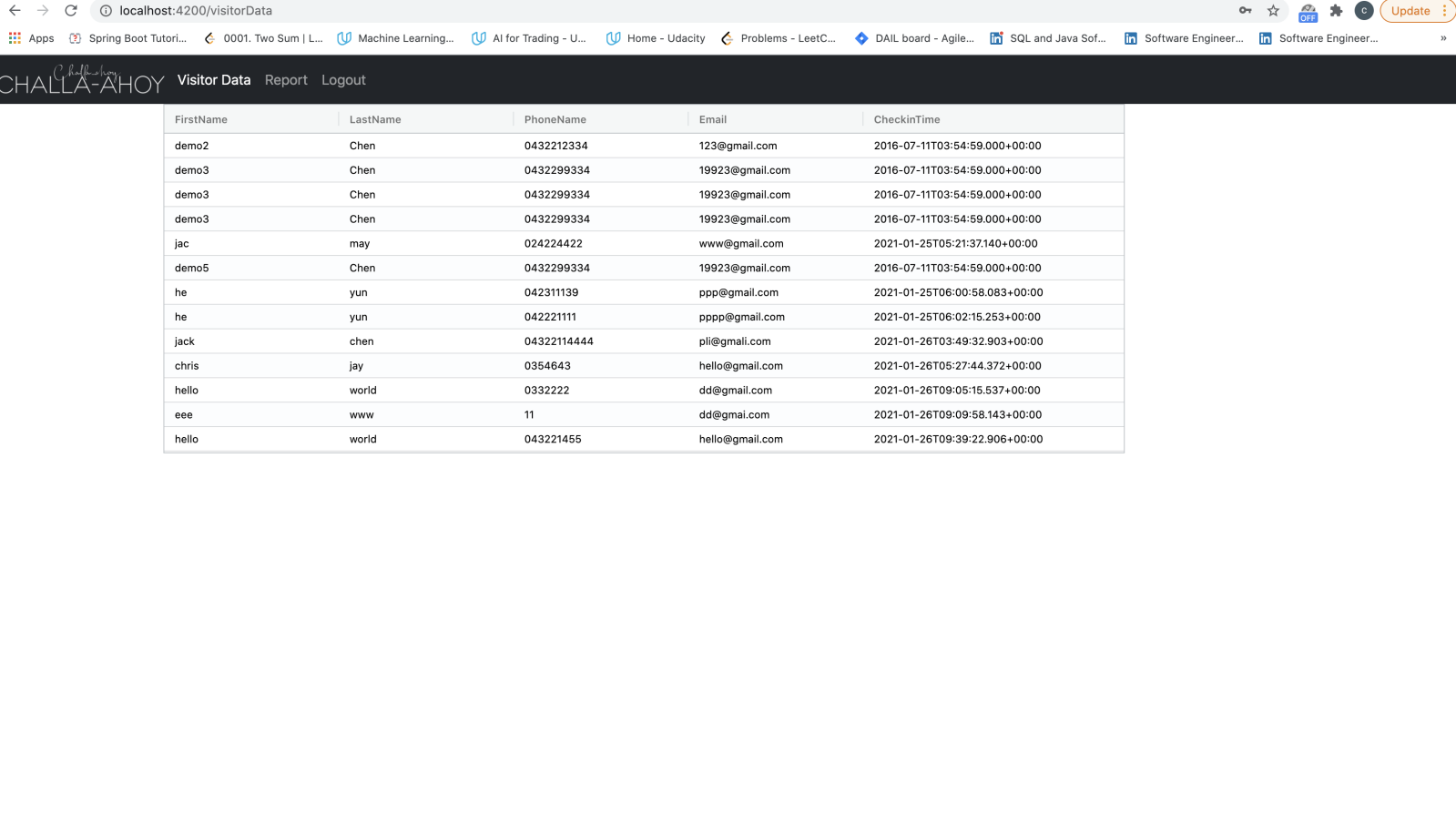
After clicking “Admin Login” button located at right-up corner in visitor register page (3.1), it forwards to admin login page.

Note: username:hello password:1234567



3.5 Admin Visitor Data

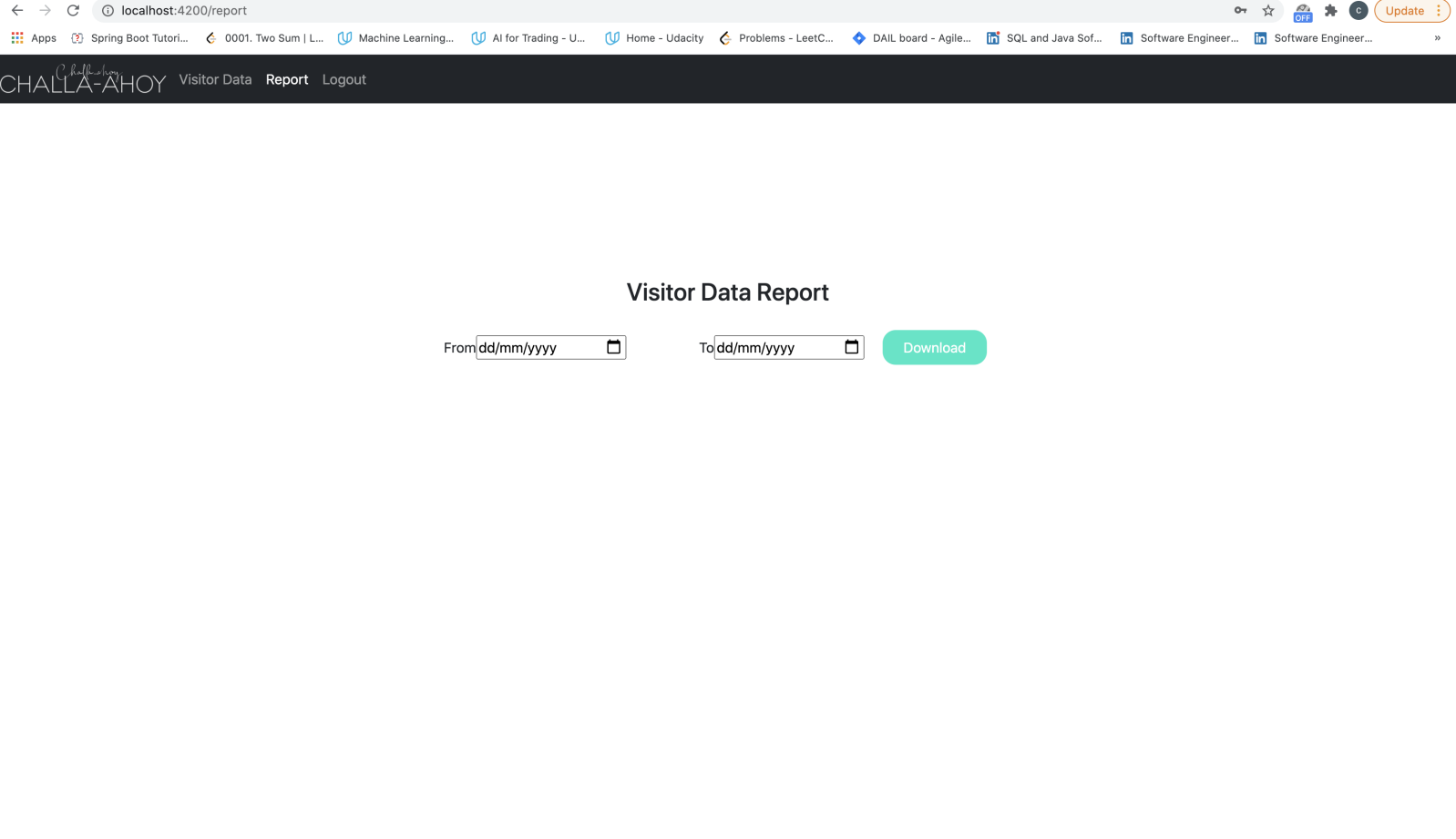
Visitor data page will present after successfully login.



3.6 Admin Report

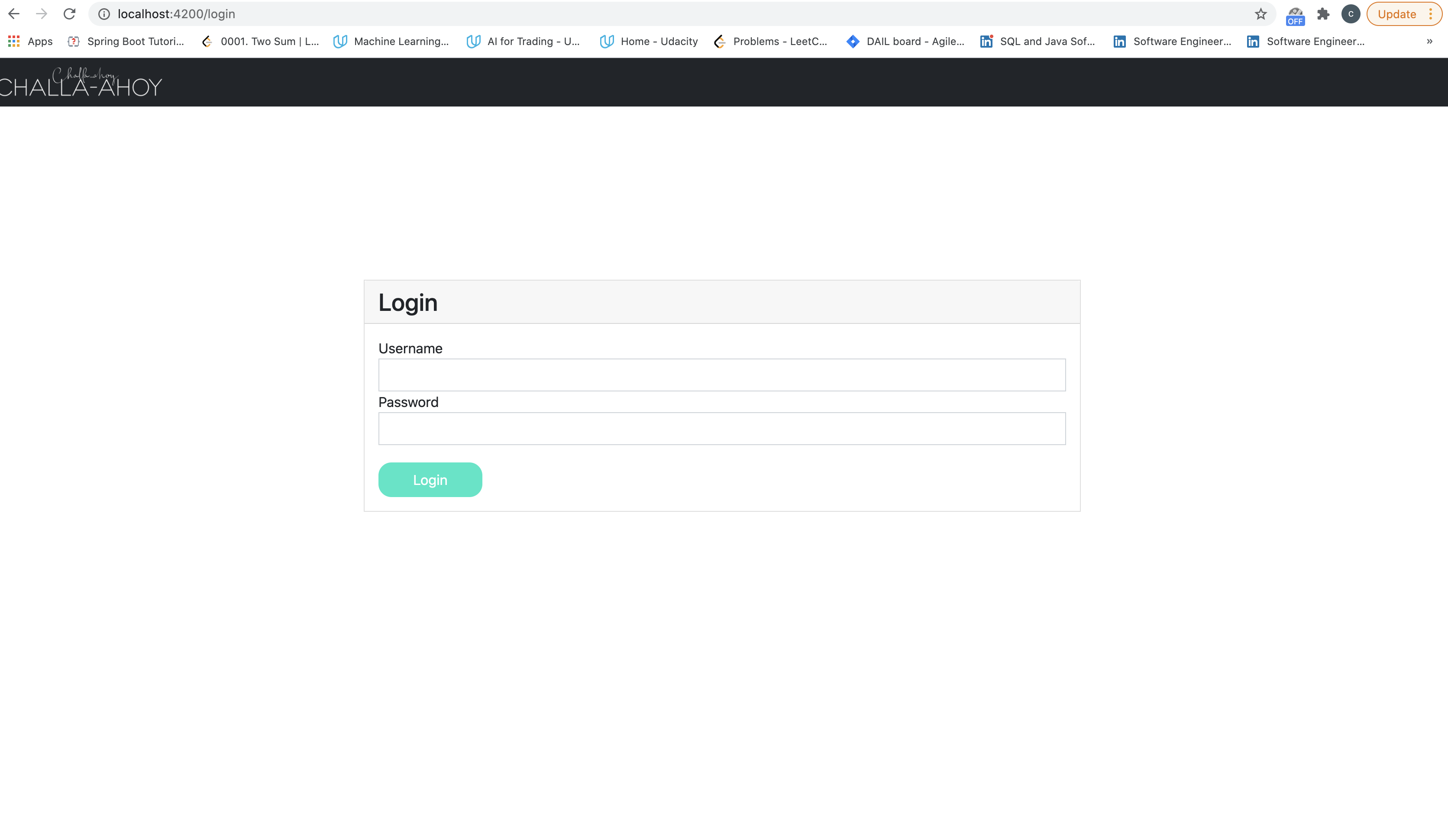
Note:Date period is not working at the moment which needs to be further developed.

Clicking “Download” button will download all visitor data into one .json file.



3.7 Admin Logout

After clicking logout button in header bar, it forwards to admin login page (3.4) and clears out JWT token at the same time.



1. Authentication and Authorization

There two roles in the project: a) admin b)visitor

Only admin role can get access to view visitor data list and generate reports while visitor role does not have such authorization

After admin login to the system, a JWT token is given to the client and the client can add this JWT token into http header to request the access to visitor data list and report module.

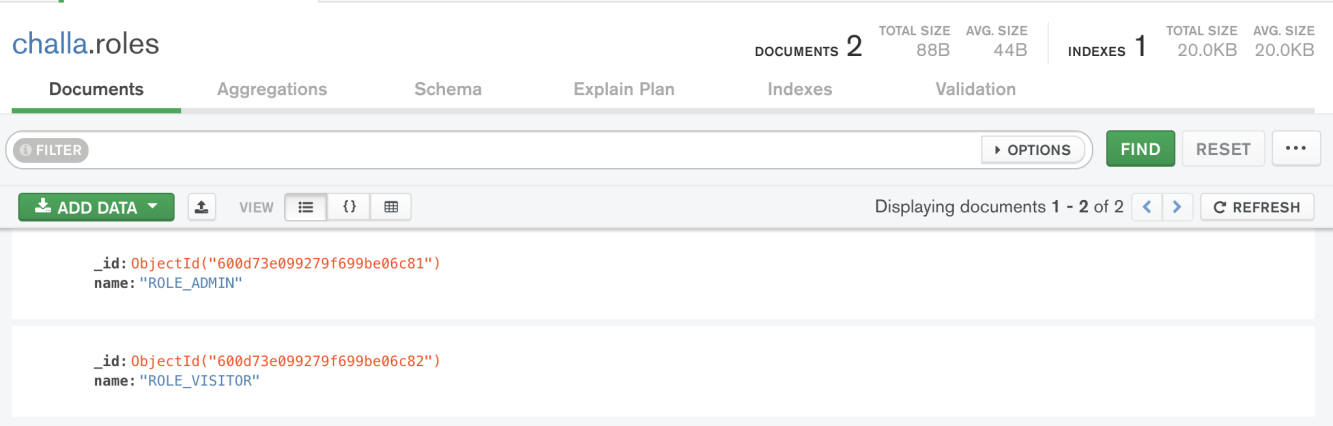
Any http request to visitor data list and report module without valid JWT token will be redirected to admin login page.

1. DB design

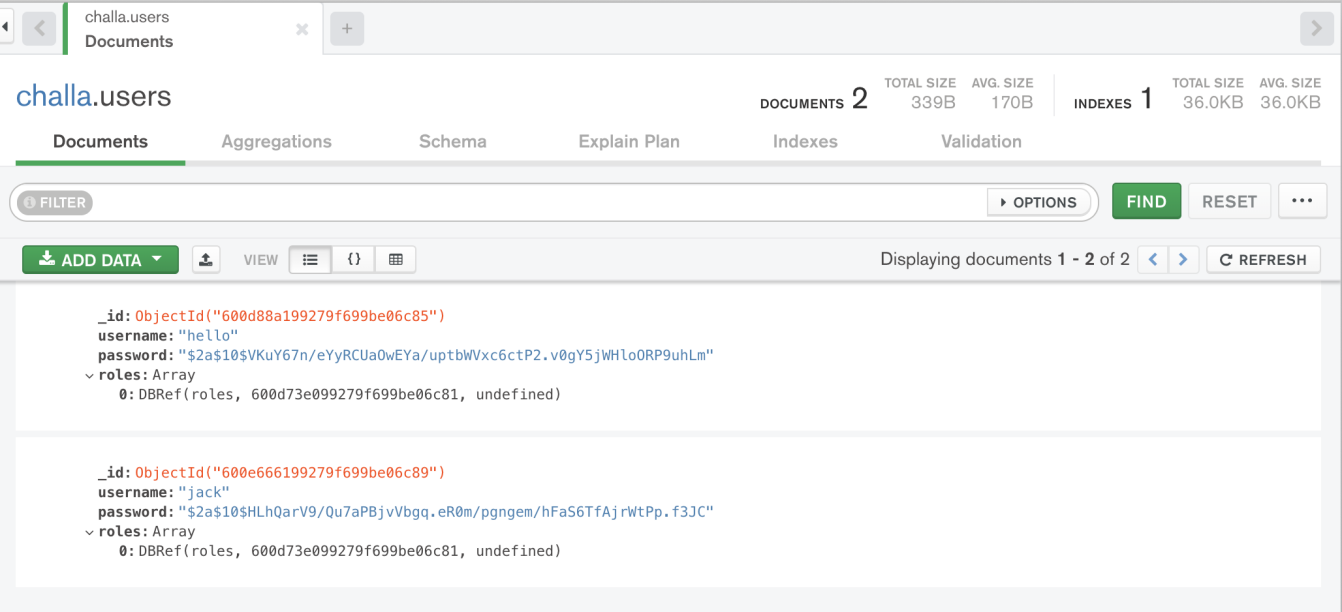
DB name is Challa. There are three collections: roles, users, visitors

**

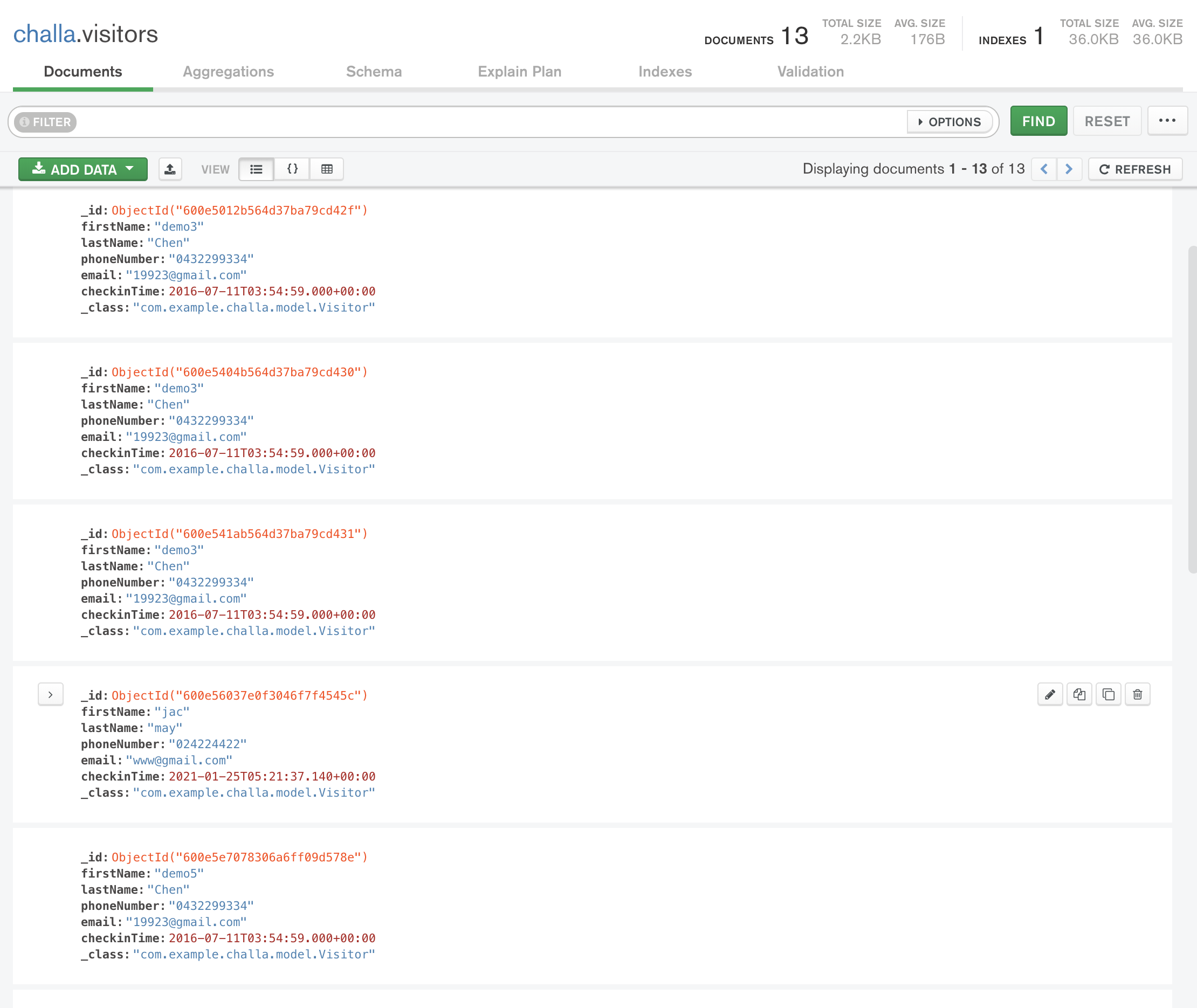
* Roles



* Users



* Visitors



6. Things to do

1. Visitor register, confirm and submit page needs to be UI responsive
2. Code validation for visitor register’s forms needs to be further improved
3. SpringBoot unit testing needs to be implemented
4. Both front-end and backend http error handling needs to be implemented
5. Code in Visitor register, confirm and submit page needs to be abstracted into common section
6. Code comments in SpringBoot needs to be added
7. Date period in admin report page needs to be further developed.