4/6/2019

UMS Documentation

Technical Doc & Guide

Abdul Mobeen
MOBEENDEV@GMAIIL.COM

UMS - Documentation

Contents

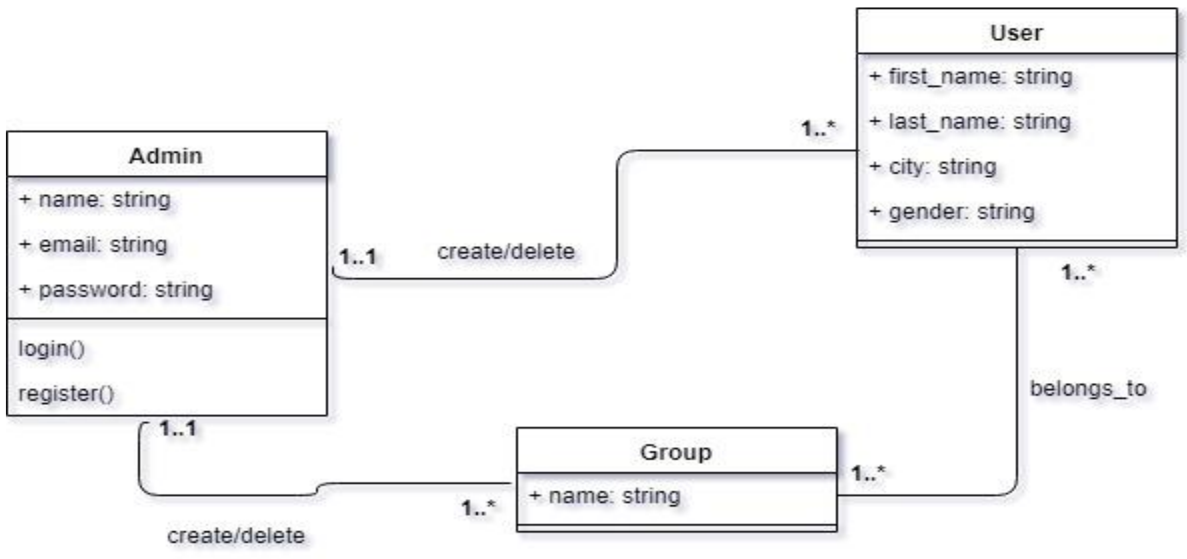
Introduction

The task is completed to develop the **User management system** with following storied.

- A person can register as Admin
- Admin can login into the system
- Admin can add users
    - a user has following attributes
        - first_name.
        - last_name
        - gender
        - city
- Admin can view the User
- Admin can delete users from groups.
    - When admin deletes a user, his/her joined group will be detached but user will not be removed from the DB.
- Amin can attach users to a group they aren't already part of.
- Admin can remove users from a group.
- Admin can create groups.
    - A group has 'name' attribute
- Admin can view the groups with all the user's in that group on detail screen
- Admin can delete groups when they no longer have any users/members

UML

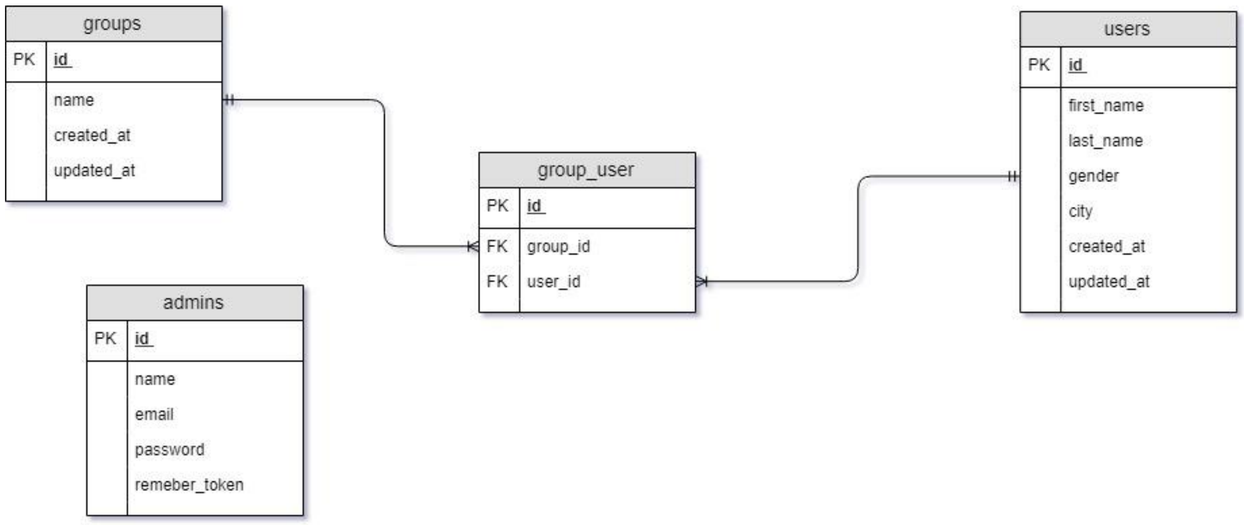
A domain model in problem solving and software engineering is a conceptual model of all the topics related to a specific problem. It describes various entities, their attributes, roles, and relationships, plus the constraints that govern the problem domain



ERD

As this web application uses **user** and **groups** data, so for this task I have used relational database, to store and retrieve the data, figure below shows tables and relationships among the tables which are used for this application.

I have assumed that there's **many to many** relationships between **users<->groups**. i.e. a user can be in multiple groups and a group can have many users.



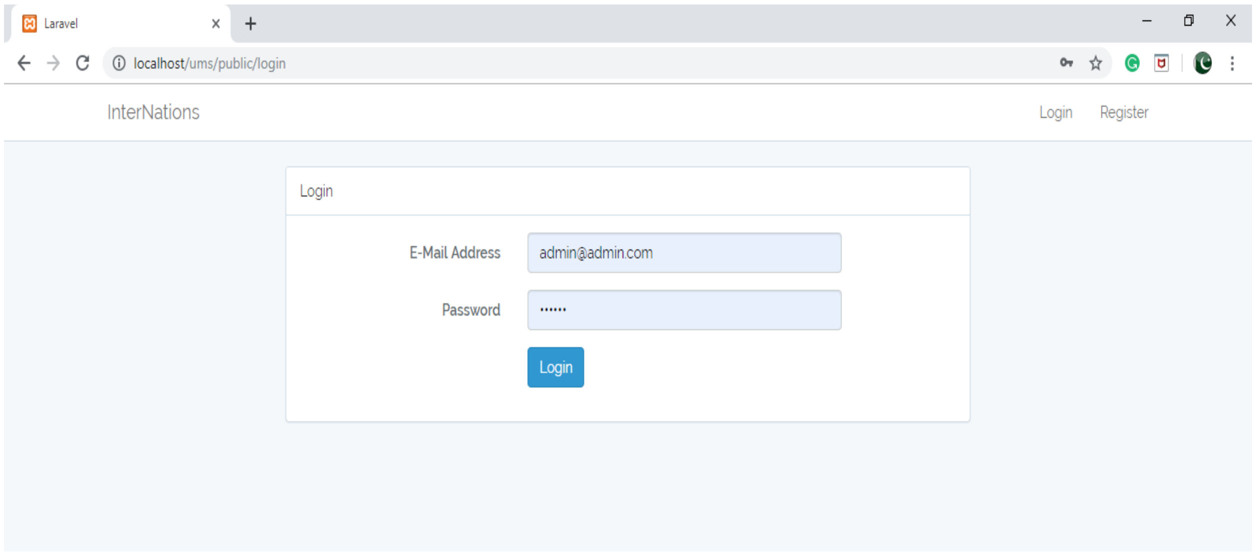
System Architecture

For this task, I have used three tier architecture i.e MVC or Model View Controller. I have also used the repository pattern where possible to create the abstraction for data access layer and to handle the domain objects (users, groups).

Next you can see the screen for above stories.

Web Application Screens

Login Screen:



Laravel

localhost/ums/public/login

InterNations

Login Register

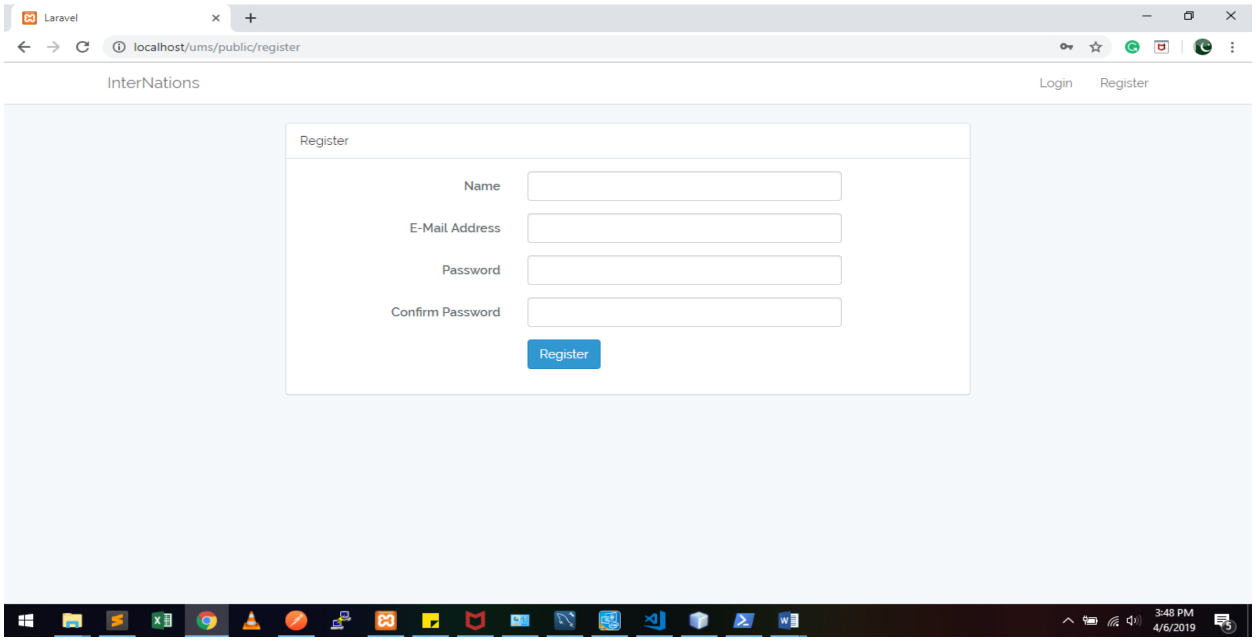
Login

E-Mail Address: admin@admin.com

Password: ••••••

Login

Register Screen



Laravel

localhost/ums/public/register

InterNations

Login Register

Register

Name

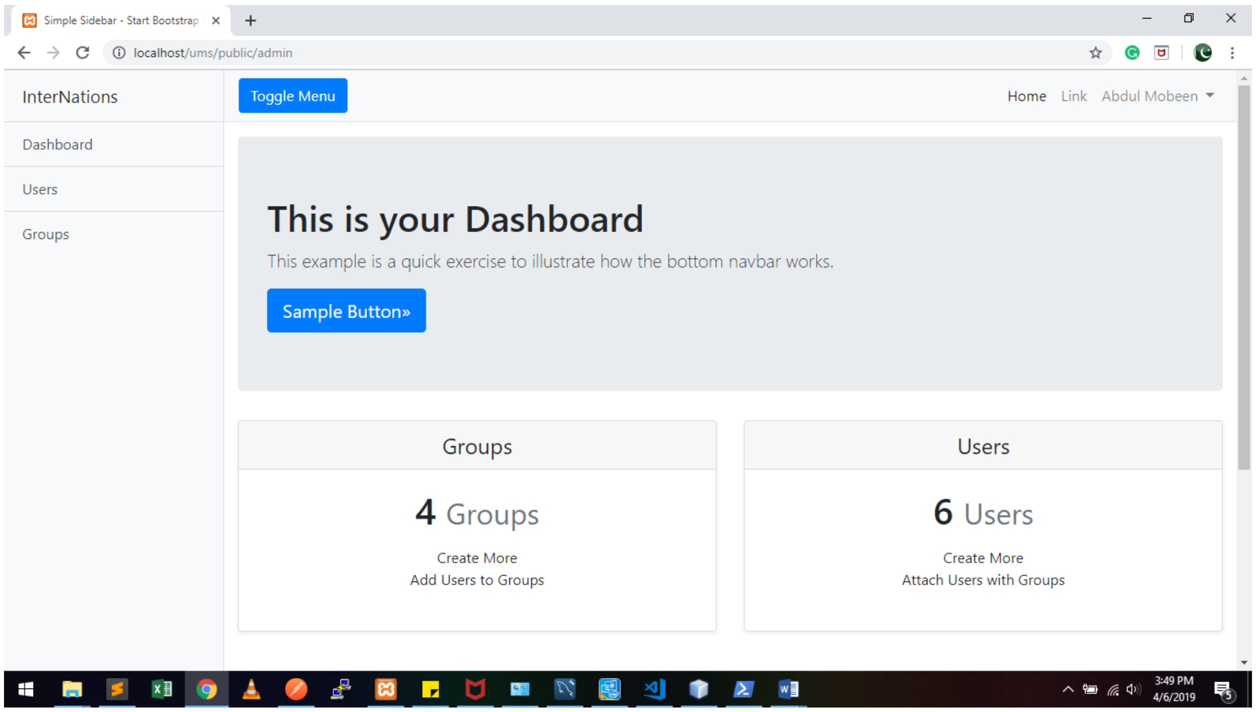
E-Mail Address

Password

Confirm Password

Register

Dashboard



Simple Sidebar - Start Bootstrap

localhost/ums/public/admin

InterNations

Toggle Menu

Home Link Abdul Mobeen

Dashboard

Users

Groups

This is your Dashboard

This example is a quick exercise to illustrate how the bottom navbar works.

Sample Button»

Groups

4 Groups

Create More
Add Users to Groups

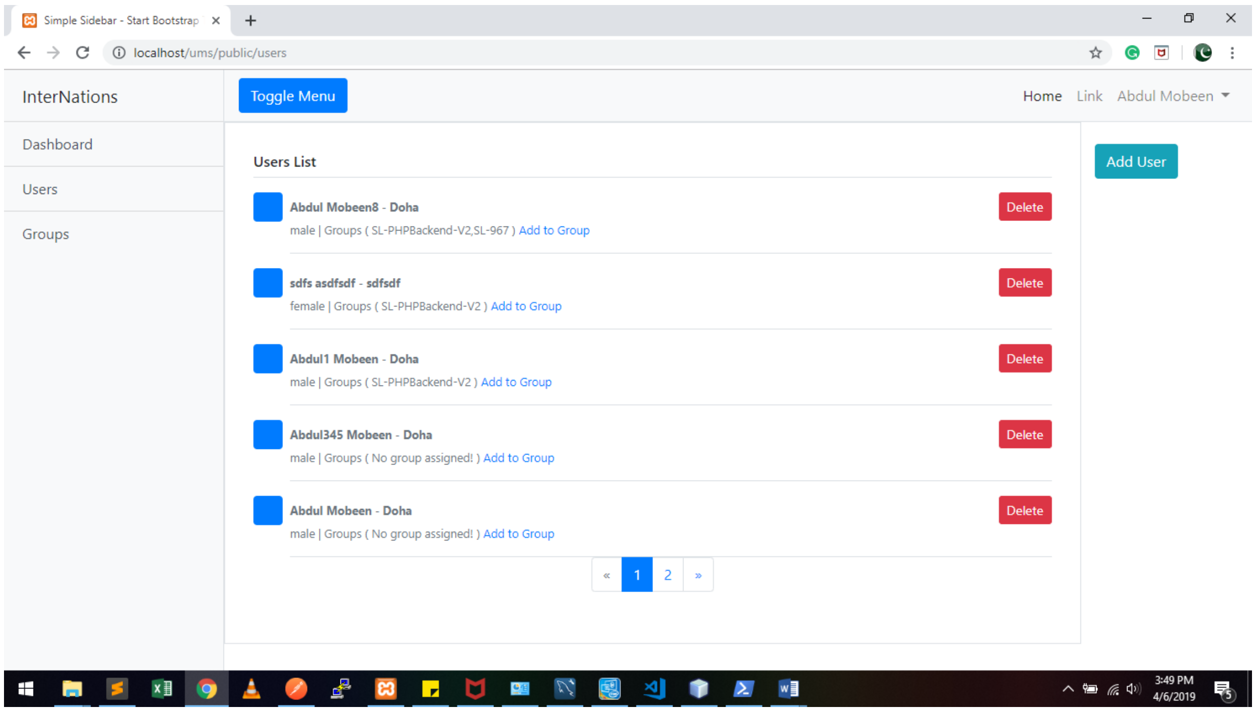
Users

6 Users

Create More
Attach Users with Groups

3:49 PM
4/6/2019

User Listing



Simple Sidebar - Start Bootstrap

localhost/ums/public/users

InterNations

Toggle Menu

Home Link Abdul Mobeen

Dashboard

Users

Groups

Add User

Users List

Abdul Mobeen8 - Doha
male | Groups (SL-PHPBackend-V2,SL-967) Add to Group
Delete

sdfs asdfsdf - sdfsdf
female | Groups (SL-PHPBackend-V2) Add to Group
Delete

Abdul1 Mobeen - Doha
male | Groups (SL-PHPBackend-V2) Add to Group
Delete

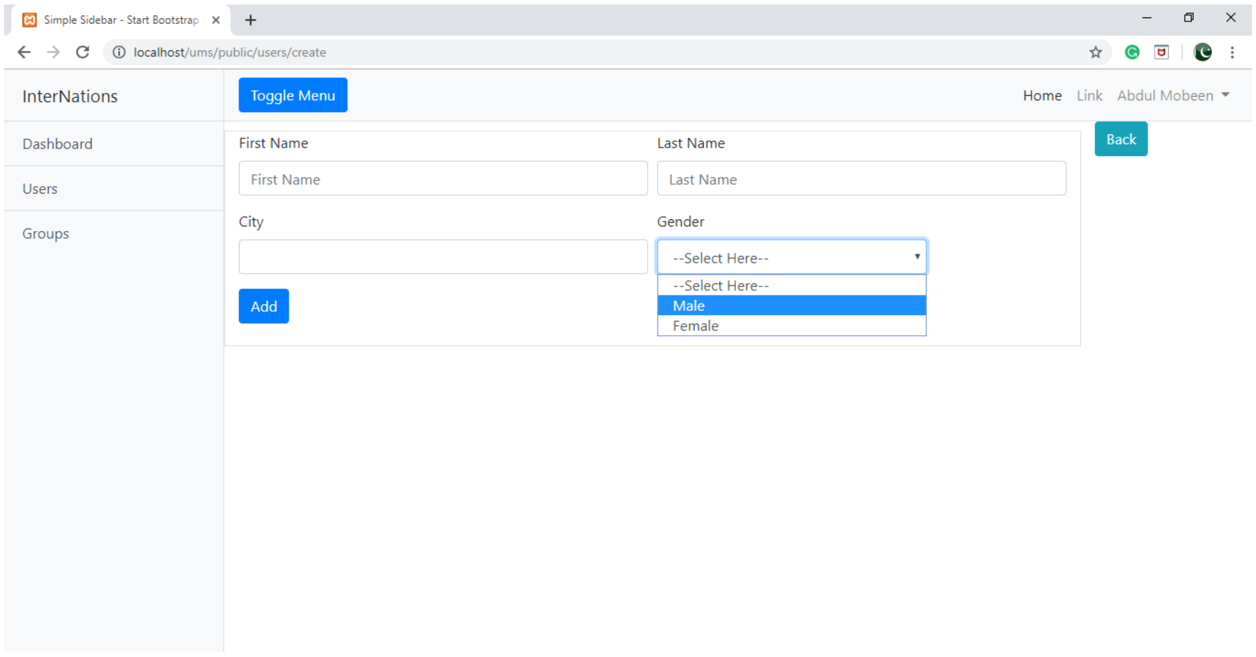
Abdul345 Mobeen - Doha
male | Groups (No group assigned!) Add to Group
Delete

Abdul Mobeen - Doha
male | Groups (No group assigned!) Add to Group
Delete

« 1 2 »

3:49 PM
4/6/2019

Add New User



Simple Sidebar - Start Bootstrap

localhost/ums/public/users/create

InterNations

Toggle Menu

Home Link Abdul Mobeen

Dashboard

Users

Groups

Back

First Name: First Name

Last Name: Last Name

City:

Gender: --Select Here--

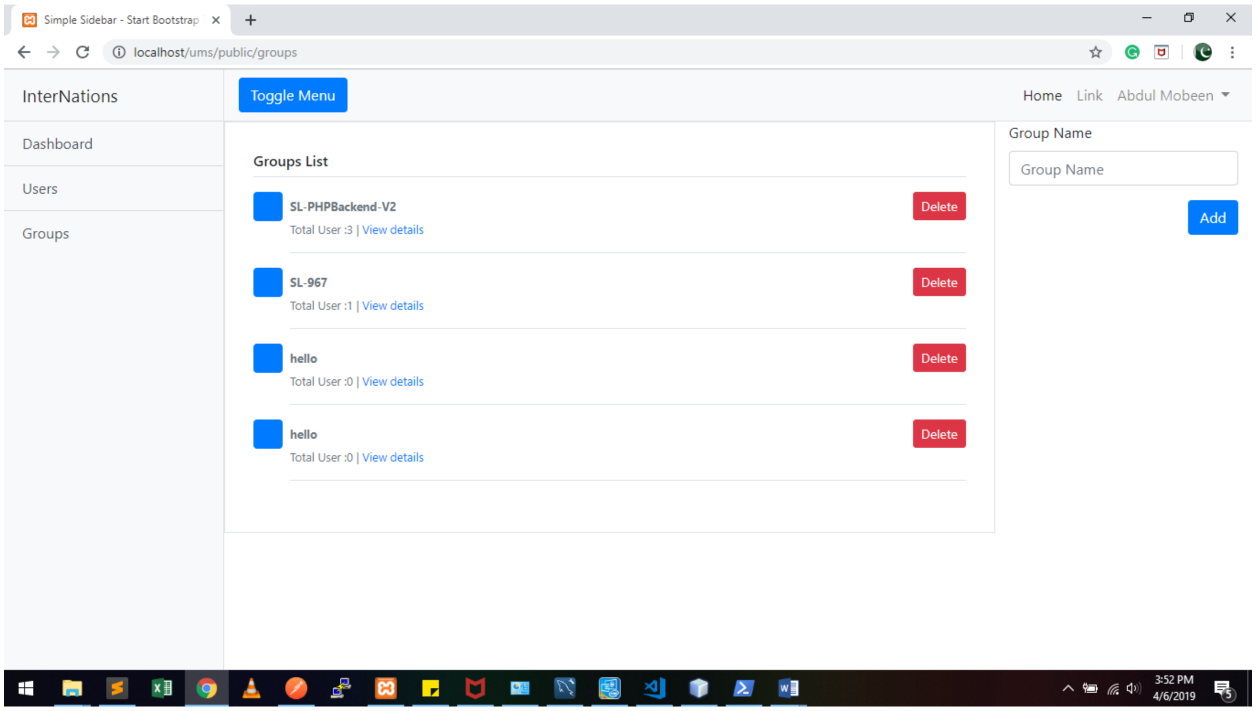
--Select Here--

Male

Female

Add

Groups Listing and Add Group



Simple Sidebar - Start Bootstrap

localhost/ums/public/groups

InterNations

Toggle Menu

Home Link Abdul Mobeen

Dashboard

Users

Groups

Groups List

SL-PHPBackend-V2
Total User :3 | View details
Delete

SL-967
Total User :1 | View details
Delete

hello
Total User :0 | View details
Delete

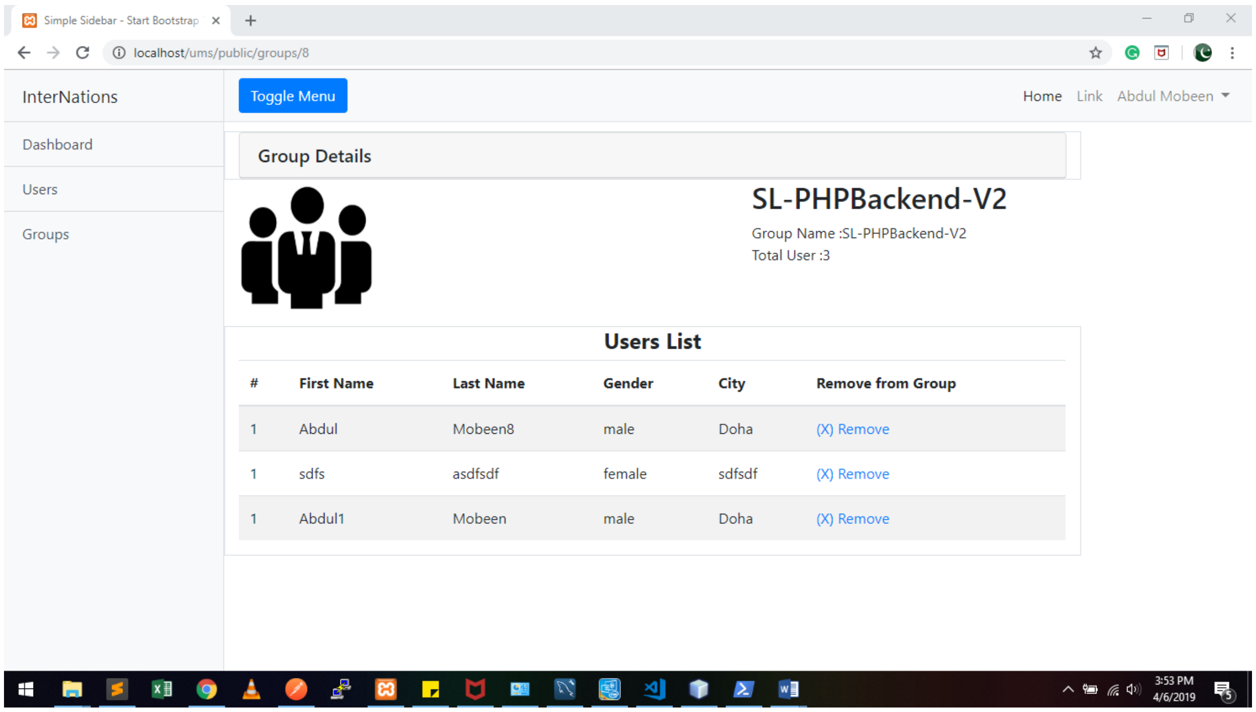
hello
Total User :0 | View details
Delete

Group Name: Group Name

Add

3:52 PM 4/6/2019

View Group Detail



Group Details

SL-PHPBackend-V2

Group Name :SL-PHPBackend-V2
Total User :3

Users List

#	First Name	Last Name	Gender	City	Remove from Group
1	Abdul	Mobeen8	male	Doha	(X) Remove
1	sdfs	asdfsdf	female	sdfsdf	(X) Remove
1	Abdul1	Mobeen	male	Doha	(X) Remove

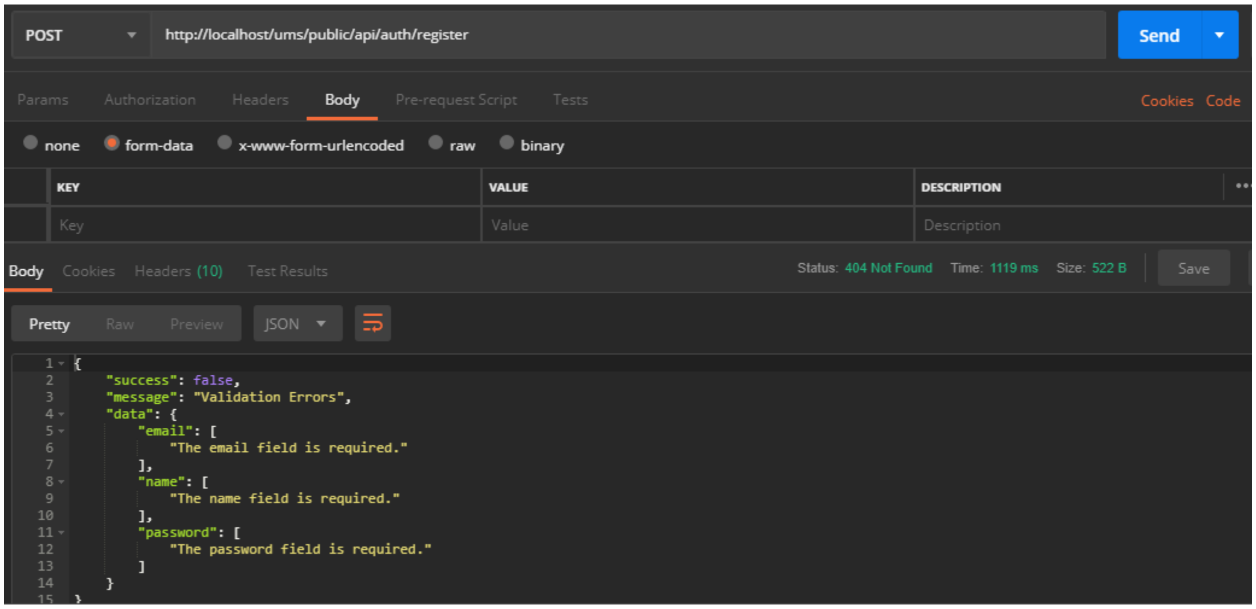
APIs

I have only created following APIs just for the demonstration

- Admin Login
- Admin Registration
- Get all users with their joined groups
- Get all groups with all the user’s in that group

APIs snapshots

Admin Registration API



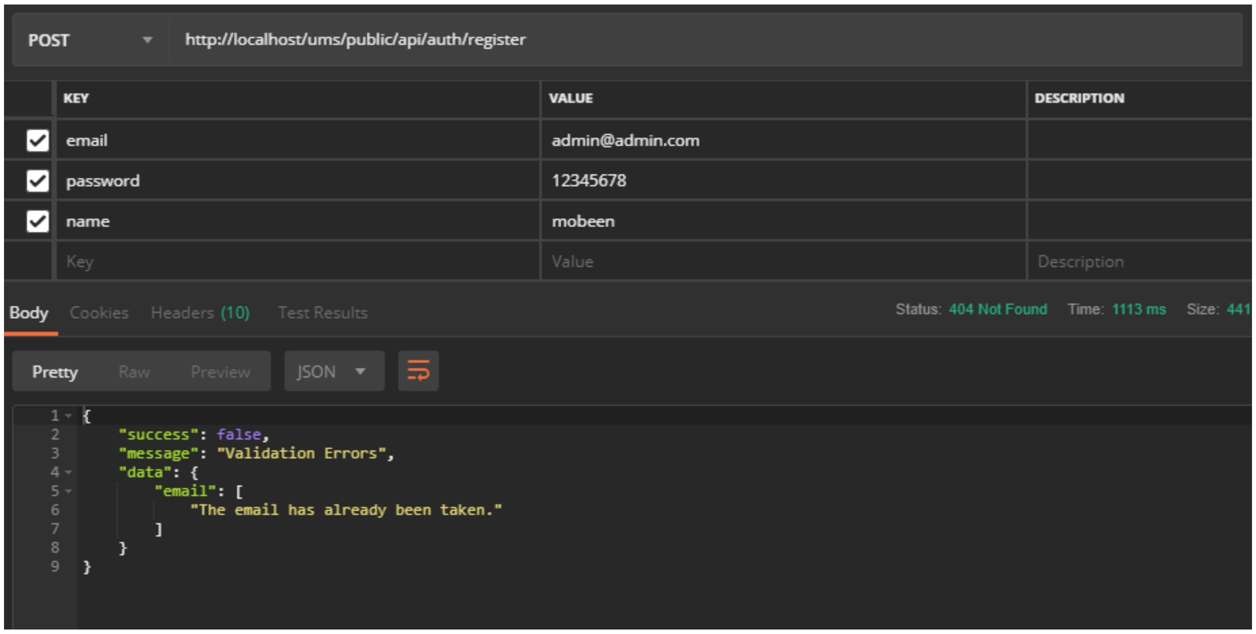
POST http://localhost/ums/public/api/auth/register

Body: form-data

KEY	VALUE	DESCRIPTION
Key	Value	Description

Status: 404 Not Found Time: 1119 ms Size: 522 B

```
{
    "success": false,
    "message": "Validation Errors",
    "data": {
        "email": [
            "The email field is required."
        ],
        "name": [
            "The name field is required."
        ],
        "password": [
            "The password field is required."
        ]
    }
}
```

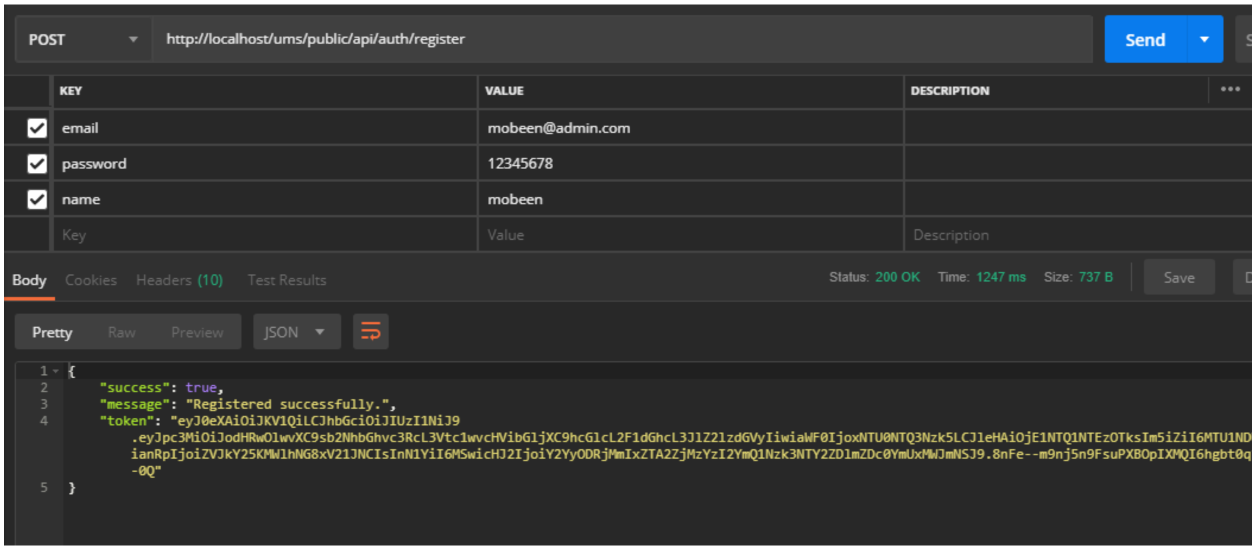


POST http://localhost/ums/public/api/auth/register

	KEY	VALUE	DESCRIPTION
✓	email	admin@admin.com	
✓	password	12345678	
✓	name	mobeen	
	Key	Value	Description

Status: 404 Not Found Time: 1113 ms Size: 441

```
{
    "success": false,
    "message": "Validation Errors",
    "data": {
        "email": [
            "The email has already been taken."
        ]
    }
}
```

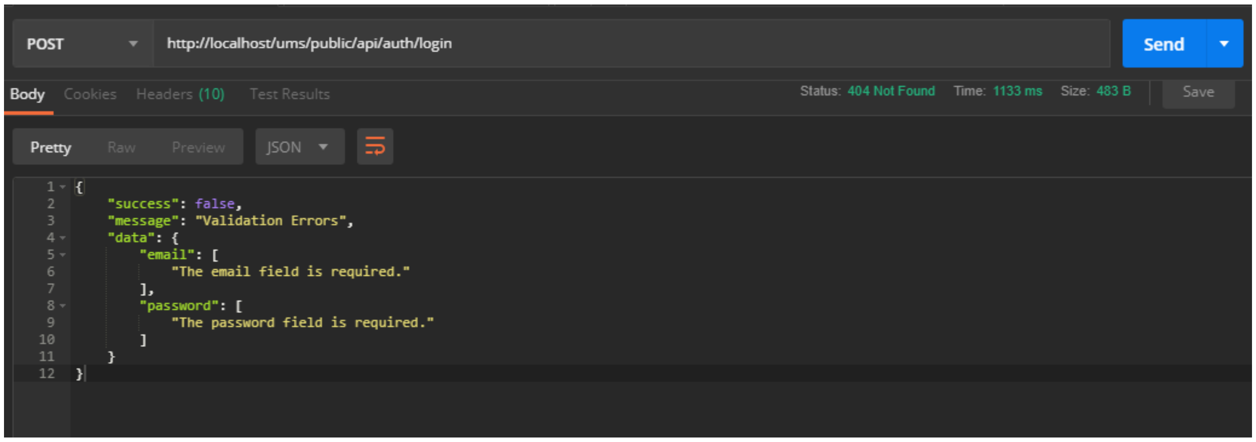

POST http://localhost/ums/public/api/auth/register

KEY	VALUE	DESCRIPTION
email	mobeen@admin.com	
password	12345678	
name	mobeen	

Status: 200 OK Time: 1247 ms Size: 737 B

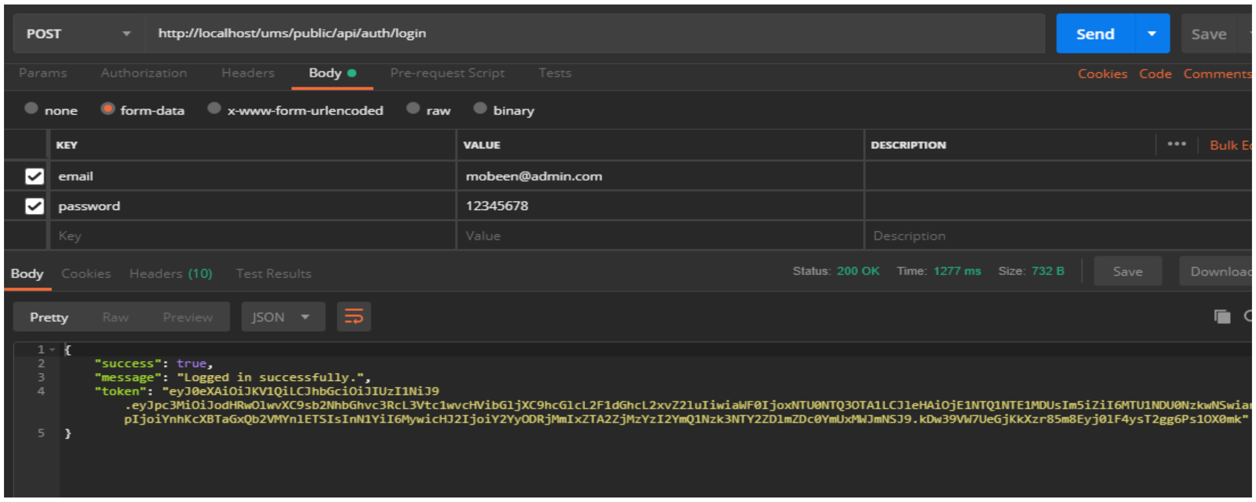
```
{
    "success": true,
    "message": "Registered successfully.",
    "token": "eyJ0eXAiOiJKV1QiLCJhbGciOiJIUzI1NiJ9
        .eyJpc3MiOiJodHRwOlwvXC9sb2NhbGhvc3RcL3Vtc1wvcHVibGljXC9hcGlcL2F1dGhcL3JlZ2lzdGVyIiwiaWF0IjoxNTU0NTQ3Nzk5LCJleHAiOjE1NTQ1NTEzOTksIm5iZiI6MTU1ND
        ianRpIjoiZVJkY25KMWlhNG8xV21JNCIsInN1YiI6MSwicHJ2IjoiY2YyODRjMmIxZTA2ZjMzYzI2YmQ1Nzk3NTY2ZDlmZDc0YmUxMWJmNSJ9.8nFe--m9nj5n9FsuPXBOpIXMQI6hgbt0q
        -0Q"
}
```

Admin Login API


POST http://localhost/ums/public/api/auth/login

Status: 404 Not Found Time: 1133 ms Size: 483 B

```
{
    "success": false,
    "message": "Validation Errors",
    "data": {
        "email": [
            "The email field is required."
        ],
        "password": [
            "The password field is required."
        ]
    }
}
```


POST http://localhost/ums/public/api/auth/login

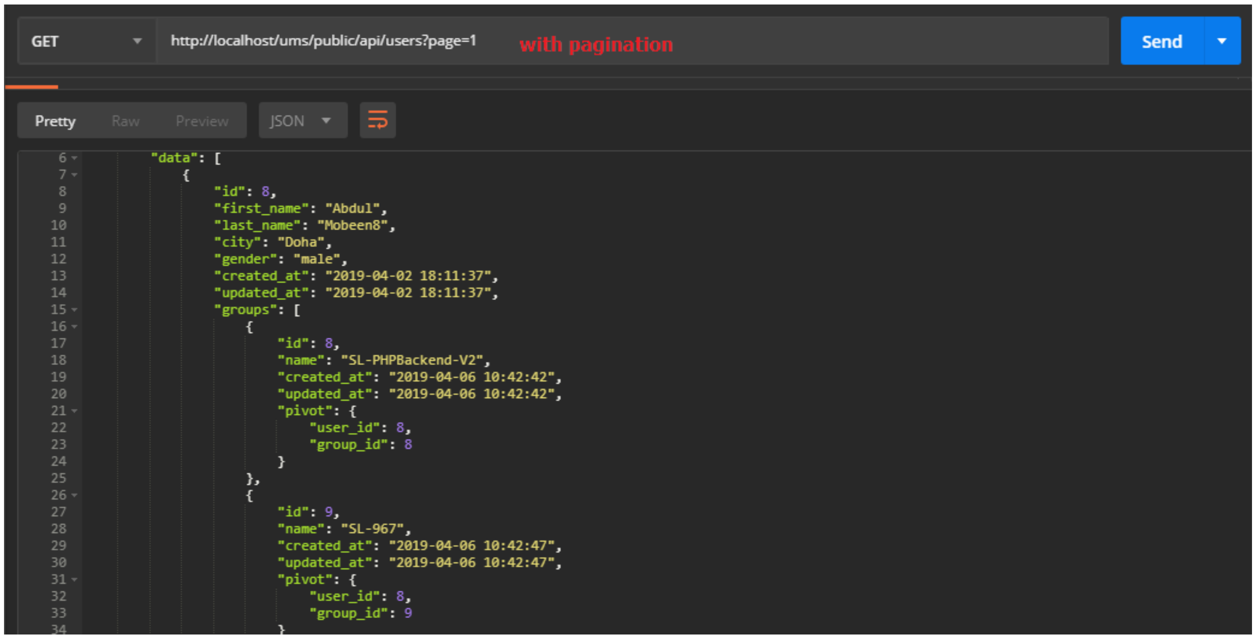
Body: form-data

KEY	VALUE	DESCRIPTION
email	mobeen@admin.com	
password	12345678	

Status: 200 OK Time: 1277 ms Size: 732 B

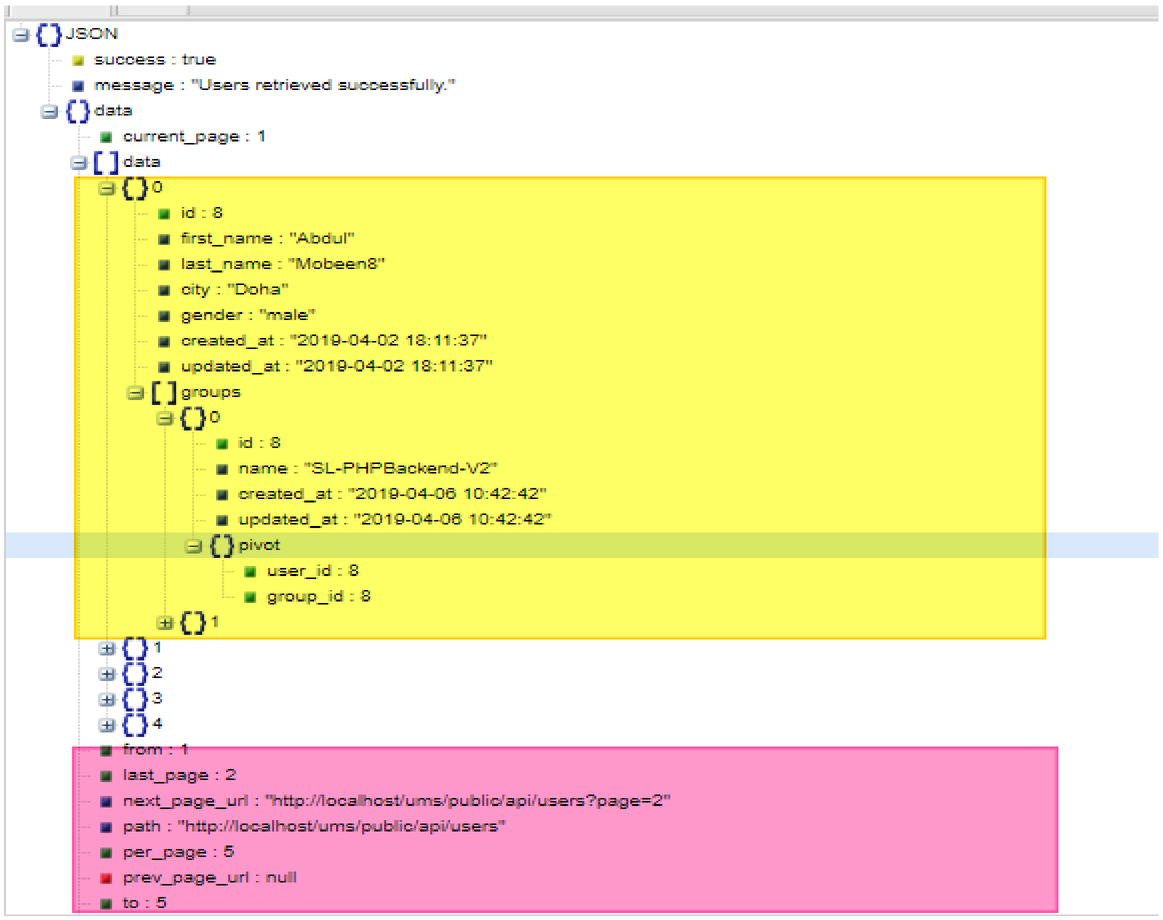
```
{
    "success": true,
    "message": "Logged in successfully.",
    "token": "eyJ0eXAiOiJKV1QiLCJhbGciOiJIUzI1NiJ9
        .eyJpc3MiOiJodHRwOlwvXC9sb2NhbGhvc3RcL3Vtc1wvcHVibGljXC9hcGlcL2F1dGhcL2xvZ2luIiwiaWF0IjoxNTU0NTQ3OTA1LCJleHAiOjE1NTQ1NTE1MDUsIm5iZiI6MTU1NDU0NzkwNSwia
        pIjoiYnhkcXBTaGxQb2VMYnlETSIsInN1YiI6MywicHJ2IjoiY2YyODRjMmIxZTA2ZjMzYzI2YmQ1Nzk3NTY2ZDlmZDc0YmUxMWJmNSJ9.kDw39VW7UeGjKkXzr85m8Eyj0lF4ysT2gg6Ps1OX0mk"
}
```

Get all users with their joined groups



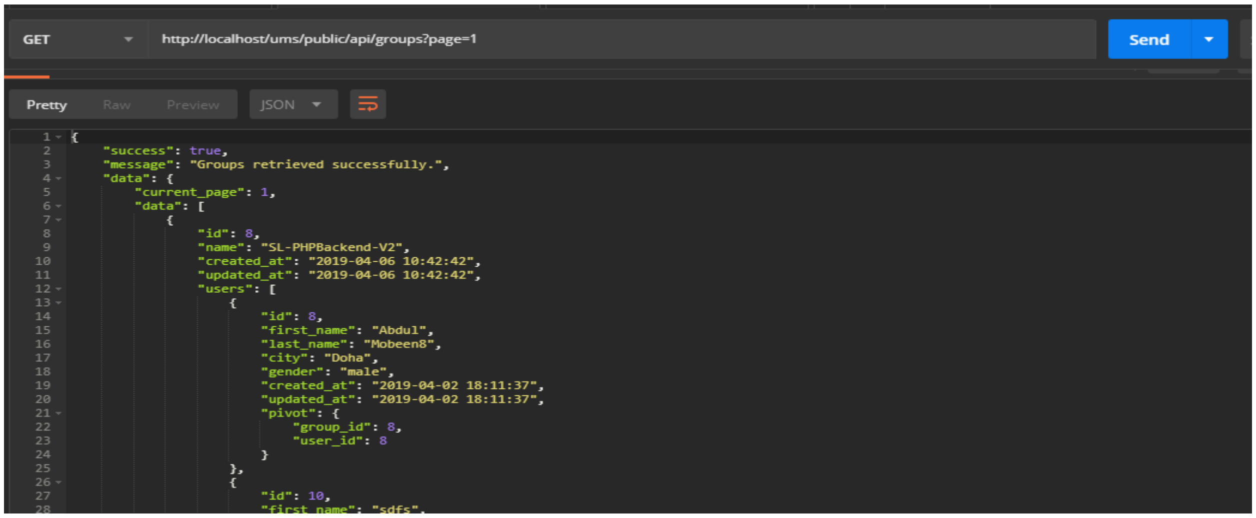
http://localhost/ums/public/api/users?page=1

```
        "data": [
            {
                "id": 8,
                "first_name": "Abdul",
                "last_name": "Mobeen8",
                "city": "Doha",
                "gender": "male",
                "created_at": "2019-04-02 18:11:37",
                "updated_at": "2019-04-02 18:11:37",
                "groups": [
                    {
                        "id": 8,
                        "name": "SL-PHPBackend-V2",
                        "created_at": "2019-04-06 10:42:42",
                        "updated_at": "2019-04-06 10:42:42",
                        "pivot": {
                            "user_id": 8,
                            "group_id": 8
                        }
                    },
                    {
                        "id": 9,
                        "name": "SL-967",
                        "created_at": "2019-04-06 10:42:47",
                        "updated_at": "2019-04-06 10:42:47",
                        "pivot": {
                            "user_id": 8,
                            "group_id": 9
                        }
```



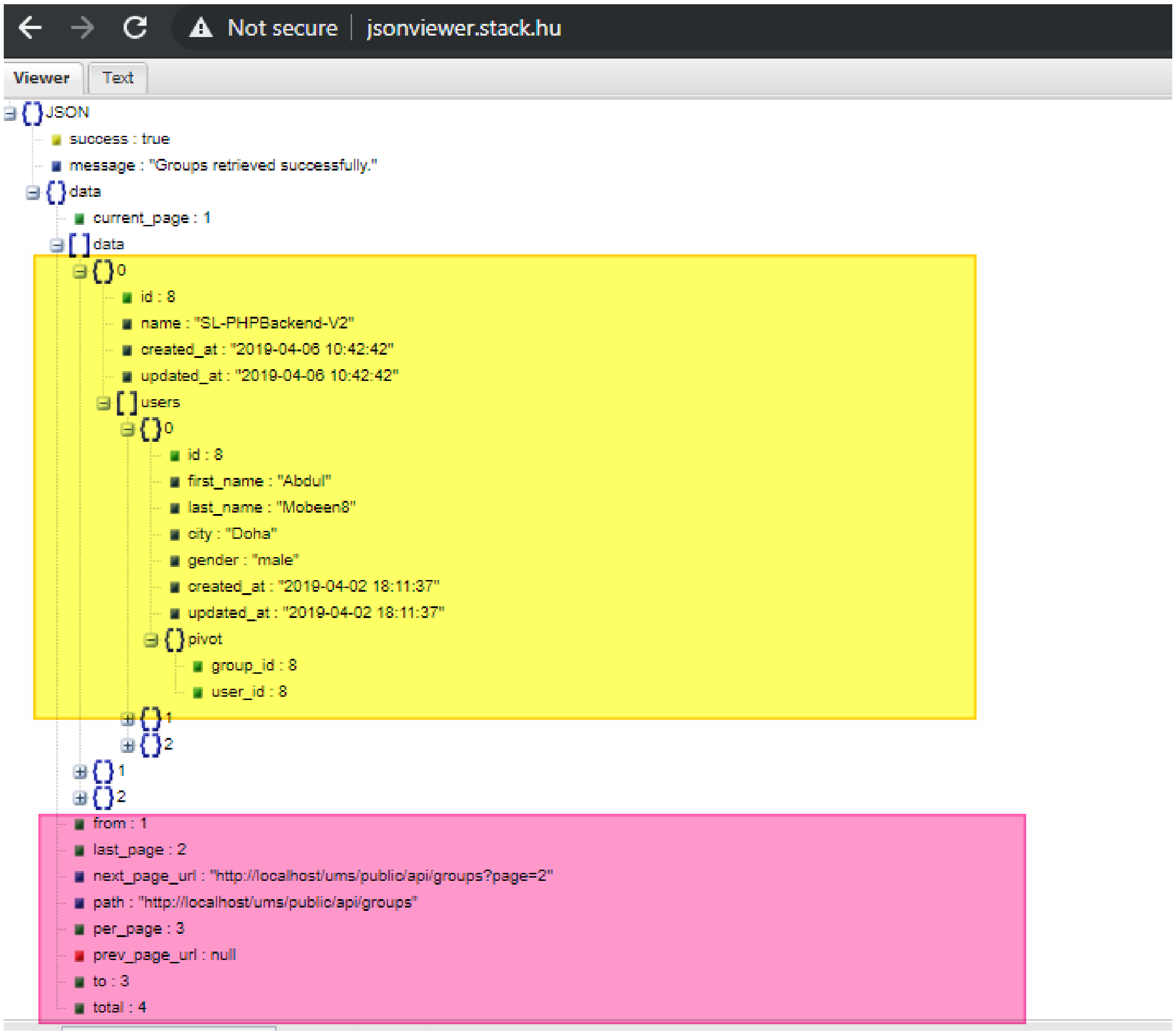
```
{} JSON
    success : true
    message : "Users retrieved successfully."
    {} data
        current_page : 1
        [] data
            {} 0
                id : 8
                first_name : "Abdul"
                last_name : "Mobeen8"
                city : "Doha"
                gender : "male"
                created_at : "2019-04-02 18:11:37"
                updated_at : "2019-04-02 18:11:37"
                [] groups
                    {} 0
                        id : 8
                        name : "SL-PHPBackend-V2"
                        created_at : "2019-04-06 10:42:42"
                        updated_at : "2019-04-06 10:42:42"
                        {} pivot
                            user_id : 8
                            group_id : 8
                    {} 1
            {} 1
            {} 2
            {} 3
            {} 4
        from : 1
        last_page : 2
        next_page_url : "http://localhost/ums/public/api/users?page=2"
        path : "http://localhost/ums/public/api/users"
        per_page : 5
        prev_page_url : null
        to : 5
```

Get all groups with all the user's in that group



GET http://localhost/ums/public/api/groups?page=1

```
{
    "success": true,
    "message": "Groups retrieved successfully.",
    "data": {
        "current_page": 1,
        "data": [
            {
                "id": 8,
                "name": "SL-PHPBackend-V2",
                "created_at": "2019-04-06 10:42:42",
                "updated_at": "2019-04-06 10:42:42",
                "users": [
                    {
                        "id": 8,
                        "first_name": "Abdul",
                        "last_name": "Mobeen8",
                        "city": "Doha",
                        "gender": "male",
                        "created_at": "2019-04-02 18:11:37",
                        "updated_at": "2019-04-02 18:11:37",
                        "pivot": {
                            "group_id": 8,
                            "user_id": 8
                        }
                    },
                    {
                        "id": 10,
                        "first_name": "sdfs",
```



Not secure | jsonviewer.stack.hu

```
{} JSON
   success : true
   message : "Groups retrieved successfully."
   {} data
      current_page : 1
      [] data
         {} 0
            id : 8
            name : "SL-PHPBackend-V2"
            created_at : "2019-04-06 10:42:42"
            updated_at : "2019-04-06 10:42:42"
            [] users
               {} 0
                  id : 8
                  first_name : "Abdul"
                  last_name : "Mobeen8"
                  city : "Doha"
                  gender : "male"
                  created_at : "2019-04-02 18:11:37"
                  updated_at : "2019-04-02 18:11:37"
                  {} pivot
                     group_id : 8
                     user_id : 8
               {} 1
               {} 2
         {} 1
         {} 2
      from : 1
      last_page : 2
      next_page_url : "http://localhost/ums/public/api/groups?page=2"
      path : "http://localhost/ums/public/api/groups"
      per_page : 3
      prev_page_url : null
      to : 3
      total : 4
```

Sample response objects for Users & Groups API are attached as well.

I have not applied the token check on the GET API calls for ease of testing the APIs.

System Implementation & Tools Technologies Uses

The system is developed using PHP, MySQL as main technologies and Apache is used as the web server to host this web application.

Other details are as follows

Laravel Framework, Bootstrap, Xdebug, Composer, XAMPP

Postman, Workbench, Chrome Browser

Setup Guide

Extract the Zip/Rar File.

Place the “ums” folder into XAMPP (htdocs) or WAMP (www) respective directory.

Open the Project in any code editor.

open the .env file and provide the DB details if yours are different e.g.

```
DB_CONNECTION=mysql

DB_HOST=127.0.0.1

DB_PORT=3306

DB_DATABASE=ums // this should remain same

DB_USERNAME=root

DB_PASSWORD=
```

Import the **ums_db.sql** file, it will create the database “ums” in your MySQL along with the required tables.

And hit this url: http://localhost/ums/public/

And you will see the login screen. As you are using it for the first time, just create an admin user using the Register Link and login.

If there is any issue in setting up the application, please let me know.

Thanks.