

# APIs for CRUD operations

*Created by Ashish Bansode  
bansodeashish24@gmail.com*

## **Database Schema:**

```
CREATE DATABASE social-networking-app;  
USE social-networking-app;
```

## **users table:**

```
CREATE TABLE users (id INT AUTO_INCREMENT PRIMARY KEY,  
                      username VARCHAR(255),  
                      password VARCHAR(255),  
                      is_logged_in BOOLEAN DEFAULT 0,  
                      createdAt DATETIME DEFAULT CURRENT_TIMESTAMP,  
                      updatedAt DATETIME DEFAULT CURRENT_TIMESTAMP);
```

## **API Description and usage:**

### **1. Register User**

**API name:** http://IPAddress:PORT/register\_user

**Example:** http://192.168.1.3:1337/register\_user

**Method:** POST

Find and create (if not found)

#### **Request Parameters:**

```
{  
    "username" : "maitreyee",  
    "password" : "password"  
}
```

#### **Response:**

```
{
  status : 0,
  message : "User registered successfully"
}
```

**OR**

```
{
  status : -1,
  error : "Username already exists"
}
```

**OR**

```
{
  status : -1,
  error : "There was an error while registering the user"
}
```

## **2. Login User**

**API name:** http://IPAddress:PORT/login\_user

**Example:** http://192.168.1.3:1337/login\_user

**Method:** PUT

Find and update

**Request:**

```
{
  "username" : "maitreyee",
  "password" : "password"
}
```

**Response:**

```
{
    status : 0,
    message : "User logged in successfully"
}
```

**OR**

```
{
    status : -1,
    error : "Error while logging in, please check your username and password."
}
```

**OR**

```
{
    status : -1,
    error : "There was an error while logging in"
}
```

### **3. Update User Password**

**API name:** http://IPAddress:PORT/update\_password

**Example:** http://192.168.1.3:1337/update\_password

**Method:** PUT

Find and update

**Request:**

```
{
    "username" : "maitreyee",
    "old_password" : "password",
    "new_password" : "passWORD123"
}
```

**Response:**

```
{  
    status : 0,  
    message : "Password changed successfully"  
}
```

**OR**

```
{  
    status : -1,  
    error : "Please check your username and old password."  
}
```

**OR**

```
{  
    status : -1,  
    error : "There was an error while updating the password"  
}
```

#### **4. Delete User**

**API name:** http://IPAddress:PORT/delete\_user

**Example:** http://192.168.1.3:1337/delete\_user

**Method:** DELETE

Find and delete

**Request:**

```
{
  "username" : "maitreyee",
  "password" : "passWORD123"
}
```

**Response:**

```
{
  status : 0,
  message : "User deleted successfully"
}
```

**OR**

```
{
  status : -1,
  error : "There was an error while deleting the user"
}
```

**5. Get Users**

**API name:** http://IPAddress:PORT/get\_users

**Example:** http://192.168.1.3:1337/get\_users

**Method:** GET

This API returns a list of all registered users.

**Request:** (no request parameters required)

**Response:**

```
{
  "status": 0,
  "users": [
    {
      "createdAt": "2019-03-17T08:15:16.000Z",
      "updatedAt": "2019-03-17T08:15:16.000Z",
      "id": 3,
      "username": "rohit",
      "password": "password",
      "is_logged_in": false
    },
    {
      "createdAt": "2019-03-21T19:34:03.000Z",
      "updatedAt": "2019-03-21T19:34:03.000Z",
      "id": 6,
      "username": "maitreyee",
      "password": "password",
      "is_logged_in": false
    }
  ]
}
```

**OR**

```
{
  status : -1,
  error : "There was an error while fetching the users"
}
```

**Useful links and commands to fix commonly occurring issues:**

**Link to fix WAMP installation issue**

<https://www.microsoft.com/en-us/download/details.aspx?id=30679>

**Command to Restart MySQL service:**

```
sudo service mysql restart
```

```
sudo service mysqld restart
```

**Command to find port number on which MySQL is running**

```
mysql -u root
```

```
mysql> SHOW GLOBAL VARIABLES LIKE 'PORT';
```