

Assignment 7

GET USER Screenshot:

The screenshot shows a REST client interface with the following details:

- URL:** localhost:8080/rest/user/muktvk
- Method:** GET
- Authorization:** No Auth
- Status:** 200 OK
- Time:** 59 ms
- Body:** A JSON object representing a user profile.

```
1 [
2   {
3     "_id": {
4       "$oid": "56dcb850e4b0b0ec6774ceb5"
5     },
6     "first_name": "Vamsi Krishna",
7     "last_name": "Muktevi",
8     "username": "muktvk",
9     "password": "root"
10  }
11 ]
```

UMKC
CS5551

ADD USER request screenshot:

The screenshot shows the REST Client interface with the following details:

- Builder** tab is active.
- URL: `localhost:8080/rest/...`
- Method: **POST**
- Path: `localhost:8080/rest/user/addUser`
- Buttons: **Params**, **Send**, **Save**
- Tab bar: **Authorization**, **Headers (1)**, **Body** (selected), **Pre-request script**, **Tests**
- Form data: **form-data**, **x-www-form-urlencoded**, **raw** (selected), **binary**
- Content type: **JSON (application/json)**
- Body content (JSON):

```
1 {  
2   "firstName": "hello",  
3   "lastName": "test",  
4   "username": "testh",  
5   "password": "qwerty"  
6 }
```

ADD USER response screenshot:

The screenshot shows the REST Client interface displaying the response of the POST request:

- Builder** tab is active.
- URL: `localhost:8080/rest/...`
- Method: **POST**
- Path: `localhost:8080/rest/user/addUser`
- Buttons: **Params**, **Send**, **Save**
- Tab bar: **Body** (selected), **Cookies**, **Headers (4)**, **Tests (0/0)**
- Status: **200 OK**, Time: **1136 ms**
- Response format: **Pretty** (selected), **Raw**, **Preview**
- Response content (Pretty):

```
1 User is successfully added!!  
2 {"firstName": "hello", "lastName": "test", "username": "testh", "password": "qwerty"}
```

UMKC
CS5551

UPDATE USER request screenshot:

The screenshot shows a REST client interface with the following details:

- Builder** tab is active.
- URL: `localhost:8080/rest/user/updateUser`
- Method: **PUT**
- Body type: **JSON (application/json)**
- Body content:

```
1 {
2   "firstName": "Hello There",
3   "lastName": "test",
4   "username": "testh",
5   "password": "qwerty"
6 }
```
- Buttons: **Send**, **Params**, **Headers**, **Authorization**, **Tests**
- Environment: **No environment**

UPDATE USER response screenshot:

The screenshot shows the response of the PUT request in the REST client interface:

- Body** tab is active.
- URL: `localhost:8080/rest/user/updateUser`
- Method: **PUT**
- Status: **200 OK**
- Time: **75 ms**
- Response body (Pretty view):

```
1 User details successfully updated !!!
2 {"firstName": "Hello There", "lastName": "test", "username": "testh", "password": "qwerty"}
```
- Buttons: **Send**, **Params**, **Headers**, **Authorization**, **Tests**
- Environment: **No environment**

UMKC
CS5551

DELETE USER screenshot:

The screenshot displays a REST client interface with the following components:

- Header:** Includes tabs for 'Builder' and 'Runner', an 'Import' button, and a 'Sync Off' indicator. On the right, there are 'Sign in', 'Share', and 'Favorite' icons.
- Address Bar:** Shows the URL 'localhost:8080/rest...' and a dropdown menu set to 'No environment'.
- Request Configuration:**
 - Method:** 'DELETE' (selected from a dropdown).
 - URL:** 'localhost:8080/rest/user/remove/testh'.
 - Params:** A button to manage query parameters.
 - Send:** A blue button to execute the request.
 - Headers:** A tab showing 'Content-Type' set to 'application/json'.
 - Body:** A tab for the request body.
 - Pre-request script:** A tab for pre-request scripts.
 - Tests:** A tab for tests.
- Response:**
 - Status:** '200 OK'.
 - Time:** '183 ms'.
 - Body:** A tab showing the response body in 'Pretty' format:

```
{
  "User is successfully removed!!"
}
```