Story titile: User Registration

Story Description: As a User I should able to register myself by providing below inputs.

1. Name (text only).

2. email id (email format)

3. moblie number (only nubmers).

4. DOB (date - DD/MON/YYYY)

5. Gender (drop down or radio button).

6. Photo (file selection)

7. Password (Alpha numberic with 6-15 length)

8. Confirm password (Alpha numberic).

Message Scenarios:

|  |  |  |  |
| --- | --- | --- | --- |
| Conditions | Level | Message | Message code |
| Name is empty | Error | Name can’t be empty | MM0001 |
| Email id is empty | Error | Email id can't be empty | MM0011 |
| Invalid Email id | Error | Email id should have @\*\*\*\*.\*\* | MM0012 |
| Invalid DOB format | Error | DOB should be in DD/Mon/YYYY format | MM0013 |
| Password is empty | Error | Password can’t be empty | MM0014 |
| If the password is not as per required length | Error | password should be in between 6-15 character. | MM0015 |
| Confirm Password is empty | Error | Confirm Password can’t be empty | MM0016 |
| Mobile number is empty | Error | Mobile number can’t be empty | MM0017 |
| Invalid Mobile number | Error | Mobile number should be Exactly 10 digit. | MM0018 |
| After saved | Info | The user details saved successfully | MM0019 |

Error response:

For validation failure: HTTP status code 422

For System failure: HTTP Status code 503

Error response: {

“isError”:true,

“messages”: [{“messageCode”:””,

“message”:””,

“level”:”error”},

{“messageCode”:””,

“message”:””,

“level”:”error”},

{“messageCode”:””,

“message”:””,

“level”:”error”}]

}

Task List:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Assigned with | Story Point | Aprox. Start Date | End date |
| Crate DB table |  | 1 |  |  |
| Create REST resources |  | 5 | 11-Apr-2019 | 16-Apr-2019 |
| Create UI screen |  |  |  |  |

Task details:

⦁ Create below REST API resources –

⦁ API to save the user details - /user Http Method Post.

Payload: {

“name”:””,

“emailId”:””,

“mobileNumber”:””,

“dob”:””,

“gender”:””,

“password”:””,

“confirmPassword”:””

}

⦁ Get the user details - /user/{userId} Http Method – GET.

⦁ Update the user details - /user HttpMethod - PUT

Payload: {

“name”:””,

“emailId”:””,

“mobileNumber”:””,

“dob”:””,

“gender”:””,

“password”:””,

“confirmPassword”:””

}

⦁ Create the client UI screen -