

BITS Assignment

Requirement Clarity

Authorization:

Specific instructions for securely handling API Key/Authorization token should be provided.

E.g. The key should not be logged while validating
How long the api key should be valid.

User Creation:

1. Need more business context about the Rating If current user status is rejected how the user's status will be changed in case of increased rating.
2. How password should be handled? While the endpoint requires a password, there is no mention of security practices related to password handling, such as hashing. This should be specified to ensure security best practices. Such as: This sensitive data should not be logged any internal logging system where while creating the user. This data should not be part of any responses.
3. How the application will behave if any extra attribute is passed in request body?
4. How the application will behave if request body does not contains any required fields?
5. In response body what is the significance of "status" and "error" as in case of success message this will be always "success" & "null" respectively.
6. Valid date range(based on allowed age group) for date of birth should be provided.
7. How the app will behave if a more than one user creating request is triggered concurrently with same email address?

Error Handling:

1. It's specified what the error response looks like but there should be defined error categories and error messages for common scenarios(e.g., missing required fields, invalid formats). Detailing these scenarios will help in both development and testing.
2. No error specifications are provided for 5XX series error. There should be custom **default** error handler.

Get User Details:

1. In response body what is the significance of "status" and "error" as in case of success message this will be always "success" & "null" respectively. It can be removed and validation can be done with response body data and status code.

Test Cases:

Authorization:

1. Request is trigger without valid api key should return 401 error for user Creation
2. Request is trigger without valid api key should return 401 error for getting user data

Request Response Data format validation:

User Creation:

1. Request body should only contains following format:

```
{
  "title": string,
  "firstName": string,
  "lastName": string,
  "dateOfBirth": date(yyyy-mm-dd),
  "email": string,
  "password": string,
  "rating": integer
}
```

Error should be thrown if any mandatory fields are missing.
Error should be thrown if any additional attributes are send as per request

2. Response Body should be in following format:

```
{
  "status": string,
  "data": {
    "userId": string,
    "status": string,
    "title": string,
    "firstName": string,
    "lastName": string,
    "dateOfBirth": date(yyyy-mm-dd),
    "email": string,
    "rating": integer
  },
  "error": null
}
```

Get user record:

1. Response Body should be in following format:

```
{
  "status": string,
  "data": {
    "userId": string,
    "status": string,
    "title": string,
    "firstName": string,
    "lastName": string,
    "dateOfBirth": date(yyyy-mm-dd),
    "email": string,
    "rating": integer
  },
  "error": null
}
```

Error:

1. Validate Error Response json:

```
{
  "errorType": string,
  "errorMessage": string
}
```

Input Field Validation:

title should be option

Verify that if **title** only accepts following values

1. Mr
2. Mrs
3. Miss
4. Ms
5. Mx

Verify error message in invalid **title** value

firstName field validation

1. Should be mandatory
2. Should accept character length between 2(inclusive) and 255(inclusive)

lastName field validation

1. Should be mandatory
2. Should accept character length between 2(inclusive) and 255(inclusive)

dateOfBirth field validation

1. Should be mandatory
2. Date format should be YYYY-MM-DD

email field validation

1. Should be unique, one account can be created using a single email id
2. Error message should be thrown same email id is used to create multiple user

password field validation

1. Should be mandatory

rating field validation

1. Should be mandatory
2. Accepted value should be between 0 to 10
3. Should not accept any value out side of the range

Rating Business Logic Validation

1. Account status should be **rejectd** if the rating is 0
2. Account status should be **new** if the rating is 1-4
3. Account status should be **active** if the rating is ≥ 5

Get User end point validation

1. User details should be return with valid user id provided as path parameter
2. Error message should be displayed if invalid user id provided as path parameter
3. Appropriate error message should be displayed if no user id is provided

Server response

1. All server response should contain appropriate HTTP Code
 - **Success:** 200
 - **Bad/Unfulfilled User request:** 4XX
 - **Server Side Error:** 5XX