

Vrajesh Mehta vam345@nyu.edu

Steps to run the program

1. npm install
2. npm app.js

DATABASE engine : mongoDB .

Database name : stitcodingChallenge

Table name : users

Api Design:

All the request type are JSON and response type are JSON expect the getEvents. getEvents will return Array of JSON

NOTE: After login put token in authorization tab in postman

Use Type = bearer token, put token and click on Preview request.

Now you can call setPreferences and getEvents. Otherwise, you will get the error invalid token

1. /register:

a. Method: post

b. Body: {

```
"username": "vrajesh978",    ← required
"password": "Vrajesh@978",  ← required
"firstName": "Vrajesh",     ← required
"lastName": "Mehta",       ← required
"Preferences": [
  {
    "category": "Music",
    "genreID": "R&B"
  },
  {
    "category": "Film",
    "genreID": "Comedy"
  }
]
```

c. Response:

- i. **on Success** : { message: "User is added to database" }
- ii. **error** :
 1. if username is already taken :
{ "message": "Username \"vrajesh978\" is already taken" }
 2. If any one of the required field is empty:
{

Vrajesh Mehta vam345@nyu.edu

```
"message": "User validation failed: preferences: Path  
`preferences` is required., lastName: Path `lastName` is required.,  
firstName: Path `firstName` is required., hash: Path `hash` is  
required., username: Path `username` is required."  
}
```

2. /login:

a. **Method:** post

b. **Body :**

```
{  
  "username": "vrajesh978",  
  "password": "Vrajesh@978"  
}
```

c. **Response:**

i. **OnSuccess:** json response → user's details and **token** for the user so that user can **make other api calls** using that token.

ii. **On Error:**

1. If username or password is wrong:

```
{  
  "message": "Username or password is incorrect"  
}
```

3. /getEvents:

a. **Method:** GET

b. **Requires token to call this method**

c. **Body:** no body needed since user has set the preferences while registering to our system or user can set the preferences by calling the setPreferences/{id}.

d. **Response:**

i. **onSuccess:** Array of the JSON response from the API

<https://yv1x0ke9cl.execute-api.us-east-1.amazonaws.com/prod/events>

- If the user has more than one preferences, it will fetch the events from the above API endpoint and return the array of the JSON response.
- Array[0] contains the first preference data, array[1] contains second preference data and so on.

ii. **onError:** Json response from the API

If token is missing :

```
{  
  "message": "invalid token"  
}
```

4. /setPreferences/{id}:

a. Method: PUT

b. Body:

```
{  
  "Preferences":[  
    {"category":"Music","genreID":"R&B"},  
    {"category":"Film","genreID":"Comedy"}  
  ]  
}
```

c. Response (json):

i. onSuccess:

```
{  
  "message": "user has setPreferences"  
}
```

ii. onError:

If token is missing :

```
{  
  "message": "invalid token"  
}
```