

My Date Is Here RESTful API Documentation

Description: This guide outlines the basic principles of using the MDIH RESTful API to manage user's details and some actions

This documentation is program language agnostic and our examples will use command-line CURL to send the requests to MDIH

1) Authentication/Login

If the user want to make any API call then he has to be a logged in user. This is call will accept username and password and validates that user. If user successfully validated then this call will return header OK with 200 status else return error message "HTTP Basic: Access denied."

URL: `curl -u username:password http://www.mydateishere.com/api/session.json -d /login=userlogin' -d 'password=userpassword' -X POST`

Parameters:

login : this is mandatroy

password: this is mandatory

HTTP method: POST

Response:

Success: {"message":"Requested user is authorized"}

Fail: {"error":"Requested user is not authorized"}

2) User registration

This call is to allow user to register with MDIH this is same as registering throug website

URL:

`curl -u username:password http://www.mydateishere.com/api/users -d user[first_name]=xxxx -d user[middle_name]=xxxx -d user[last_name]=xxxx -d user[email]=xxxx -d user[login]=xxxxx -d user[password]=xxxxx -d user[password_confirmation]=xxxx -d user[gender]=Male -d user[zip]=xxxx`

`-d user[orientation]=xxxx -d user[date_of_birth(1i)]=xxxx -d user[date_of_birth(2i)]=xxxx -d user[date_of_birth(3i)]=xxxx -d lookin_genders=Male -X POST`

Parameters:

user[first_name] : mandatory and maximum is 40 characters

user[middle_name] : optional

user[last_name] : mandatory and maximum is 40 characters

user[email] : mandatory, shoul be like email format, minimum is 6 and maximum is 100 characters

user[login] : mandatory, should be only letters, numbers, and .-_@ minimum is 6 and maximum is 100 characters

user[password] : mandatroy and should match with password_confirmation parameter

user[password_confirmation] : mandatroy and should match with password parameter

user[gender]: mandatroy and value is either Male or Female or Transgender

user[zip] : optional

user[orientation] : mandatroy and value is either Straight or Gay or Lesbian or Bisexual or Transgender

user[date_of_birth(1i)] : mandatory and value is the year

user[date_of_birth(2i)] : mandatory and value is month number

user[date_of_birth(3i)] : mandatory and value is day number

lookin_genders : optional and value is Male, Female, Bisexual, Transgender or all with coma with one white space(,) separated string example Male,Female

HTTP method: POST

Response:

Success: {"message":"Thanks for signing up! We're sending you an email with your activation link."}

Fail: Array with error messages

3) Forgot password

This is to request to get the link to reset user's password

URL:

curl -u username:password http://www.mydateishere.com/api/users/forgot_password.json -d email=example@gmail.com -X POST

Parameters:

email : mandatory and should exists user with that email

HTTP method: POST

Response

Success: {"message":"We sent you an email with instructions to reset your password."}

Fail: any one of following error

1) {"error":"Please enter your email to reset password."}

2) {"error":"No user exist with #{email}. Please enter your correct email."}

3) {"error":"You are just invited and not activated your account. Please signup to activate your account."}

4) Get User details

This call is to get requested user's details

URL:

curl -u username:password http://www.mydateishere.com/api/users/1.json -X GET

Parameters:

id : mandatory and should be an integer

HTTP method: GET

Response

Success: Users details and profile details of given user

Fail: {"error": "User not exist with this id"}

5) Create profile

This call is to create profile for given user

URL :

```
curl -u username:password http://www.mydateishere.com/api/users/#{user id whose profile need to  
create}/profiles.json -X POST -d profile[address_one]=xxxx -d profile[address_two]=xxxx -d profile[city]=xxxx -d  
profile[state]=xxxxx -d profile[country]=xxxxx -d profile[height]=xxxx -d height_in=Inches -d profile[weight]=xxxx -d  
weight_in=Kgs -d profile[body_type]=xxxx -d profile[marital_status]=xxxx -d profile[ethnicity]=xxxx -d  
profile[hair_color]=xxxx
```

Parameters:

profile[address_one] : optional and should be string

profile[address_two] : optional and should be string

profile[city] : optional and should be string

profile[state] : optional and should be string

profile[country] : optional and should be country name. Please check in edit profile page for options

profile[height] : optional and should be integer

height_in : value should be either Inches or Centimeters. Default is Centimeters

profile[weight] : optional and should be integer

weight_in : value should be either Kgs or Pounds. Default is Pounds

profile[body_type] : optional and any one of selected from options. Please check in edit profile page for options

profile[marital_status] : optional and any one of selected from options. Please check in edit profile page for options

profile[ethnicity] : optional and any one of selected from options. Please check in edit profile page for options

profile[hair_color] : optional and any one of selected from options. Please check in edit profile page for options

profile[eye_color] : optional and any one of selected from options. Please check in edit profile page for options

profile[skin_color] : optional and any one of selected from options. Please check in edit profile page for options

profile[num_of_children] : optional and any one of selected from options. Please check in edit profile page for options

profile[want_children] : optional and any one of selected from options. Please check in edit profile page for options

profile[pets] : optional and many of selected from options with sapared by coma(,) with one white space. Please check in edit profile page for options

profile[drink] : optional and any one of selected from options. Please check in edit profile page for options

profile[drugs] : optional and any one of selected from options. Please check in edit profile page for options

profile[income_level] : optional and any one of selected from options. Please check in edit profile page for options

profile[languages_spoken] : optional and many of selected from options with sapared by coma(,) with one white space. Please check in edit profile page for options

profile[religion] : optional and any one of selected from options. Please check in edit profile page for options

profile[profession] : optional and should be string

profile[education_level] : optional and any one of selected from options. Please check in edit profile page for options

profile[interests] : optional and many of selected from options with sapared by coma(,) with one white space. Please check in edit profile page for options

profile[description] : optional and should be string

profile[looking_for] : optional and any one of selected from options. Please check in edit profile page for options

profile[headline] : optional and should be string

profile[nationality] : optional and should be string

profile[favorite_music] : optional and many of selected from options with sapared by coma(,) with one white space. Please check in edit profile page for options

profile[favorite_tv_show] : optional and should be string

profile[favorite_books] : optional and should be string

profile[favorite_movies] : optional and should be string

profile[favorite_avtivity] : optional and many of selected from options with sapared by coma(,) with one white space. Please check in edit profile page for options

profile[favorite_type_of_cuisine] : optional and should be string

profile[what] : means what i am value is optional and should be string.

profile[role_model] : optional and should be string

profile[fetish] : optional and many of selected from options with sapared by coma(,) with one white space.
Please check in edit profile page for options

profile[sexual_activity] : optional and many of selected from options with sapared by coma(,) with one white space. Please check in edit profile page for options

profile[smoke] : optional and any one of selected from options. Please check in edit profile page for options

profile[zodiac_sign] : optional and any one of selected from options. Please check in edit profile page for options

profile[favorite_sports] : optional and many of selected from options with sapared by coma(,) with one white space. Please check in edit profile page for options

profile[favorite_clubs] : optional and many of selected from options with sapared by coma(,) with one white space. Please check in edit profile page for options

profile[favorite_resaurants] : optional and should be string

profile[ideal_places] : optional and should be string

profile[ideal_vacation] : optional and should be string

profile[favorite_authors] : optional and should be string

profile[favorite_actors] : optional and should be string

profile[best_feature] : optional and any one of selected from options. Please check in edit profile page for options

profile[political_view] : optional and any one of selected from options. Please check in edit profile page for options

profile[body_hair] : optional and any one of selected from options. Please check in edit profile page for options

profile[favorite_food] : optional and many of selected from options with sapared by coma(,) with one white space. Please check in edit profile page for options

HTTP method: POST

Response

:
Success: { "message": "Profile was created successfully" }

Fail: Array with error messages

6) Update profile

This is to update user's profile

URL :

```
curl -u username:password http://www.mydateishere.com/api/users/#{user id whose profile need to update}/profiles/#{profile id}.json -X PUT -d profile[address_one]=xxxx -d profile[address_two]=xxxx -d profile[city]=xxxx -d profile[state]=xxxxx -d profile[country]=xxxxx -d profile[height]=xxxx -d height_in=Inches -d profile[weight]=xxxx -d weight_in=Kgs -d profile[body_type]=xxxx -d profile[marital_status]=xxxx -d profile[ethnicity]=xxxx -d profile[hair_color]=xxxx
```

Parameters:

All the parameters mentioned in the create profile API call

HTTP method: PUT

Response

Success: { "message": "Profile was created successfully" }

Fail: Array with error messages

7) Update profile photo

This is used to change user's profile image

URL: curl -u username:password http://www.mydateishere.com/api/users/#{user id whose profile need to update}/profiles/update_photo.json -X PUT -d profile[avatar]=#{ upload photo to assign as user's profile photo }

Parameters:

profile[avatar] : image file which is either jpeg, jpg, png, gif, bmp, x-png, pjpeg

HTTP method: POST

Response

Success: { "message": "Profile photo was updated successfully" }

Fail: Array with error messages

8) Get user's matched list

This is used to get the matched users list of a user

URL: curl -u username:password http://www.mydateishere.com/api/users/my_matches.json

HTTP method: GET

Response

Success: Array of matched users list

9) Add user to matched users list

Add a user to logged in user's matched list

URL: `curl -u username:password http://www.mydateishere.com/api/users/4/add_to_my_match.json -d 'match_percent=23' -X POST`

Parameters:

`match_percent`: integer like 5/10/15/20/25/30...

HTTP method: POST

Response

Success: `{"message": "Successfully added user to your matched list"}`

Fail: `{"error": "Adding a user to your matched list failed. Please try again"}`

10) Remove user from matched users list

Remove a user from logged in user's matched list

URL: `curl -u username:password http://www.mydateishere.com/api/users/#{matcher id}/delete_from_my_match.json -X DELETE`

HTTP method: DELETE

Response

Success: `{"message": "User successfully removed from your matched list"}`

Fail: `{"error": "Error while removing user from your match list. Please try again"}`

11) Send message to users

This is used to send a message to a user

URL: curl -u username:password http://www.mydateishere.com/api/messages.json -d 'message[subject]=sgdsgdsg' -d 'message[body]=sgdsgdsg' -d 'to=4' -X POST

Parameters:

message[subject] : subject of the message

message[body] : body of the message

to: user ids with separated by coma(,) Ex: 2,3,4,5

HTTP method: POST

Response

Success: {"message":"Message was successfully sent."}

Fail:

1) {"error":"You do not have permissions to send message to some users from selection"}

If any of recepents blocked you then above error will through

2) {"error":"Subject and Body can't blank"}

If user did not passed message parameters or parameters with blank string then above error will through

3) {"error":"Invalid users selected"}

If to parameter is blank or contain invalid user ids then above message

12) Get list of product/gifts to send

URL: curl -u username:password http://www.mydateishere.com/api/gifts.json -X GET

HTTP method: GET

Response

Success: Array of gifts

13) Get list of received/sent gifts of a user

URL: curl -u username:password http://www.mydateishere.com/api/gifts/my_gifts.json -d 'type=#{received | sent}' -X GET

Parameters:

type : gifts type and the value is either received or sent. Default value is received

HTTP method: GET

Response

Success: Array of recived or sent gifts

14) Send gift to a friend

URL: curl -u username:password http://www.mydateishere.com/api/gifts/#{product/gift id}/send_to_friend.json -d 'message=sent from api call' -d 'user_id=4' -X POST

Parameters:

id : is the id of any gifts returned in the 13th API call

message : to send message along with gift and value is text message(should be less than 254 charecters)

user_id : should be integer and is a user id of receiver

HTTP method: POST

Response

Success: {"message":"Your virtual gift has been sent!"}

Fail:

1) {"error": "Virtual Gift not was sent. Please send again "}

If any error happend while saving details then above error will be returned

2) {"error": "Invalid user id or product id"}

If invalid user id or product/gift id passed then above error will be returned

15) delete sent or received gift

URL: curl -u username:password http://www.mydateishere.com/api/gifts/12.json -X DELETE

Parameters:

id : Id of the sent or received gift which is one of the record id of 14th API call

HTTP method: DELETE

Response

Success: {"message": "Virtual Gift deleted successfully"}

Fail:

1) {"error": "Permission denied"}

If user does not have permissions(which id is not of his received/sent gift id) to delete the gift id he passed

1) {"error": "Invalid id passed"}

If id is invalid then above error message will be returned

16) Get simple search results

URL:

```
curl -u username:password http://www.mydateishere.com/api/searches/simple_results.json -d 'user_name=abcdef' -X GET
```

OR

```
curl -u username:password http://www.mydateishere.com/api/searches/simple_results.json -X GET -d 'i_am_a=#{Male or Female}' -d 'looking_for=#{Male or Female}' -d 'zip_code=123456' -d 'distance=5' -d 'has_picture=on' -d 'has_contacted=on' -d 'online_only=on' -d 'new_members=on'
```

Parameters:

For simple search there are two types to get search results. One is search based on user name and another one is search based on filters

user_name : This is to search users based on user name

i_am_a : Is a string and values should be Male or Female. If no value passed then default is Male

looking_for : Is a string and values should be Male or Female. If no value passed then default is Female

zip_code : Integer type and should be a US zip code only

distance : Integer and required to find out users with given distance from given zip code

has_picture : If this parameter present and value 'on' passed, then only matched users who are having profile picture are returned

has_contacted : If this parameter present and value 'on' passed, then users whom the logged in user has already contacted are removed from results

online_only : If this is parameter present and value 'on' passed, then returned matched users and who are online

new_members : If this is parameter present and value 'on' passed, then returned matched users and who are activated in last 14 days

HTTP method: GET

Response

Success: Array of matched users

17) Get advanced search results

URL: `curl -u username:password http://www.mydateishere.com/api/searches/advance_results.json -X GET`

Parameters:

`i_am_a` : Is a string and values should be Male or Female. If no value passed then default is Male

`looking_for` : Is a string and values should be Male or Female. If no value passed then default is Female

`zip_code` : Integer type and should be a US zip code only

`distance` : Integer and required to find out users with given distance from given zip code

`has_picture` : If this parameter present and value 'on' passed, then only matched users who are having profile picture are returned

`has_contacted` : If this parameter present and value 'on' passed, then users whom the logged in user has already contacted are removed from results

`online_only` : If this is parameter present and value 'on' passed, then returned matched users and who are online

`new_members` : If this is parameter present and value 'on' passed, then returned matched users and who are activated in last 14 days

Weight search criteria

`weight_any` : If this parameter is present and value is passed as 1 then no weight criteria

If above parameter is not passed then following parameters are required for weight criteria

`weight_from`: Integer value

`weight_to` : Integer value

`weight_in` : Is to check for units and value is either Pounds or Kgs

Height search criteria

`height_any` : If this parameter is present and value is passed as 1 then no height criteria

If above parameter is not passed then following parameters are required for height criteria

`height_in` : Is to check for units and value is either Inches or Centimeters

If `height_in` value is Inches then following parameters are required

`height_feet_from`: integer value and should be between 2 to 6

height_inches_from: integer value and should be between 1 to 11

height_feet_to: integer value and should be between 2 to 6

height_inches_to: integer value and should be between 1 to 11

If height_in value is Centimeters then following parameters are required

height_cm_from : integer value

height_cm_to : Integer value

ethnicity : Any values with coma(,) separated from "Will tell you later, Asian, Black, Caucasian/White, Hispanic/Latino, Middle Eastern, Mixed Race, Pacific Islander, Other"

skin_color : Any values with coma(,) separated from "Will tell you later, Black, Brown, White, Tan, Olive, Other"

eye_color : Any values with coma(,) separated from "Black, Brown, Blue, Hazel, Green, Gray, Other"

hair_color : Any values with coma(,) separated from "Black, Blond, Brown, Red, Gray, White, Bald, Mixed Color, Shaved, Salt n pepper, Other"

personality : Any values with coma(,) separated from "Will tell you later, Slim, Average, Athletic, Muscular, Bodybuilder, Large, Stocky, Swimmer, Runner, Other"

body_hair : Any values with coma(,) separated from "Will tell you later, None, , Some, Hairy, Shaved"

best_feature : Any values with coma(,) separated from "Arms, Belly button, Butt, Calves, Chest, Eyes, Feet, Hands, Hair, Legs, Lips, Neck, You will find out later"

languages : Any values with coma(,) separated from options. Please check the website for language options

marital_status: Any values with coma(,) separated from "Will tell you later, Single, Married, In a relationship, Divorced, Widowed, Swinger, Engaged, Just living together, Other"

has_kids : Any values with coma(,) separated from "Will tell you later, 0, 1, 2, 3, 4, 5, More"

want_kids : Any values with coma(,) separated from "Yes, No, Haven't decided, Will tell you later"

religion : Any values with coma(,) separated from "Will tell you later, Catholic, Buddhist, Hindu, Lutheran, Atheist, Muslim, Spiritual, Jewish, Other"

fetish : Any values with coma(,) separated from "Will tell you later, Casual, Formal, Alternative, Drag, Leather, Military, Jock, Trendy, Punk, Skin, Rubber, Lingerie, Others"

sexual_activity : Any values with coma(,) separated from "Will tell you later, Casual, Formal, Wild, Kinky, Dominator, Top, Bottom, Leather, SM, Bondage, Cougar, Others, Vanilla"

smoke : Any values with coma(,) separated from "Will tell you later, Yes, No, Occasionally, Never"

drink : Any values with coma(,) separated from "Will tell you later, Yes, No, Occasionally, Never"

drugs : Any values with coma(,) separated from "Will tell you later, Yes, No, Occasionally, Never"

zodiac_sign : Any values with coma(,) separated from "Aries, Taurus, Gemini, Cancer, Leo, Virgo, Libra, Scorpio, Sagittarius, Capricorn, Aquarius, Pisces"

political_view : Any values with coma(,) separated from "Will tell you later, Liberal, Very liberal, Conservative, Very conservative, Moderate, No opinion, Other"

education_level : Any values with coma(,) separated from "Will tell you later, High school, College, Master, PHD, Apprenticeship, Other"

profession : should be any string

income_level : Any values with coma(,) separated from "15k-30k, 30k-45k, 45k-60k, 60k-75k, 75k-100k, Over 100k, Will tell you later"

favorite_food : Any values with coma(,) separated from "American, Caribbean, Chinese, French, Fusion,

Indian, Italian, Japanese, Korean, Malaysian, Mediterranean, Mexican, Middle Eastern, Russian, Spanish, Thai, Vegetarian, Vietnamese, Greek, Sea food, Other"

favorite_music : Any values with coma(,) separated from "Alternative Blues, Classic, Rock n Roll, Classical Dance/Electronica Hard Rock & Metal, Middle Eastern (Persian/Arabic), Religious, Modern Rock n Roll Pop/Top 40, New Age Rap/Hip Hop Reggae, Latin, Punk Acoustic, Gospel, Other"

favorite_movies : should be any string

favorite_actors : should be any string

favorite_tv_show : should be any string

favorite_books : should be any string

favorite_authors : should be any string

role_model : should be any string

favorite_type_of_cuisine : should be any string

favorite_sports : Any values with coma(,) separated from "Archery, Athletics, Badminton, Body Building, Baseball, Basketball, Boxing, Canoe/Kayak, Cricket, Cycling, Disc Sports, Diving, Equestrian, Fencing, Football (American), Golf, Football (soccer), Hockey, Gymnastics, Handball, Horse Racing, Ice Hockey, Martial Arts, Motorsports, Netball, Rowing, Rugby, Sailing, Shooting, Snow Boarding, Skiing, Cross Country, Softball, Surfing, Swimming, Table Tennis, Tennis, Tenpin Bowling, Triathlon, Volleyball, Waterpolo, Weightlifting, Wrestling, Other"

favorite_clubs : Any values with coma(,) separated from "Sports bar, Irish bar, Jazz bar, Disco, Lounge, Night clubs, Coffee shop, Wine bar, Hookah bar, Strip club, Other"

favorite_activity : Any values with coma(,) separated from "Walking in nature, Reading, Sport, Gym, Spa, Movies, Eating, Wine testing, Traveling, discovering new places, Flea markets, Sleeping, Scuba diving"

ideal_places : should be any string

ideal_vacation : should be any string

interests : Any values with coma(,) separated from "Art, Cars, Collectible, Music, Dancing, Meditation, Outdoor, Religion, Political, Others"