

<u>ID</u>	<u>Method Name</u>	<u>HTTP Method</u>	<u>Parameters</u>	<u>Returns</u>	<u>Explanation</u>
1	/users/	GET	-	returns all users: users[]	Using GET because a resource from server is needed and there isnt any wish to change the state of the server
2	/users/id/:id	GET	user id	Returns user	Using GET because a resource from server is needed and there isnt any wish to change the state of the server
3	/users/username/:username	GET	username	Returns user	Using GET because a resource from server is needed and there isnt any wish to change the state of the server
4	/users/categories/:id	GET	user_id	returns user's categories: userCategories[]	Using GET because a resource from server is needed and there isnt any wish to change the state of the server
5	/users/add	POST	firstName, lastName, city, country, username, password, email, q_restore_password[], a_restore_password[], categories[]	True: string: msg_success, False: msg_failure	using POST because we want to create another user in DB with all necessary details
6	/users/login/authenticate/	POST	Username, Password	true: Token False: failed_msg	Login cannot pass in GET True= success False=fail
7	/users/favoritesPoints/userId/:id	GET	user_id	True: Returns all favorites points of user: favoreitsPoints[], False: Returns failure_msg	Using GET because a resource from server is needed and there isnt any wish to change the state of the server
8	/users/favoritesPoints/add	POST	user_id, order_num, point_id	True: string: msg_success, False: msg_failure	using POST because we want to create another point of interest in the users favorites, in DB
9	/users/favoritesPoints/remove/userId/:id/pointId/:id	DELETE	pointId,userId	True: string: msg_success, False: msg_failure	using DELETE because we want to delete the requested point from the favorites points of interests list of the user.
10	/users/favoritesPoints/update/	PUT	pointId, orderNum, userId	True: string: msg_success, False: msg_failure	using PUT because we want to update the order of the favorites points of interests of the user
11	/users/favoritesPoints/2Latest/userId/:id	GET	userId	returns 2 latest favorite points : latest2SavedPoints []	Using GET because a resource from server is needed and there isnt any wish to change the state of the server
12	/users/qaRestorePassword/:id	GET	id	Returns all restored password questions with user_id: restoredPassword Questions[]	Using GET because a resource from server is needed and there isnt any wish to change the state of the server
13	/users/qaRestorePassword/check	POST	id, question, answer	True: username, password. False: failure_msg	using POST because if the answer is correct then the password is sent to the user and we want to send it in the body of the request for security reasons
14	/pointsOfInterests/	GET	-	True: Returns all points: PointsOfInterests[] false: error_msg	Using GET because a resource from server is needed and there isnt any wish to change the state of the server

15	/pointsOfInterests/id/:id	GET	pointId	Returns spcific point or error_msg	Using GET because a resource from server is needed and there isnt any wish to change the state of the server
16	/pointsOfInterests/popularsInCategory/:category	GET	category	all popular points from the category or error_msg	Using GET because a resource from server is needed and there isnt any wish to change the state of the server
17	/pointsOfInterests/category/:category	GET	category	Returns all points with the specified category: categoryPoints[] or failed_msg	Using GET because a resource from server is needed and there isnt any wish to change the state of the server
18	/pointsOfInterests/addView	PUT	point_id	string: success_msg or error_msg	using PUT because we want to update the value of views of the point.
19	/pointsOfInterests/name/:name	GET	point_name	specific point or error_msg	Using GET because a resource from server is needed and there isnt any wish to change the state of the server
20	/pointsOfInterests/populars/	GET	-	True: points with rating>=80% False: err_msg	Using GET because a resource from server is needed and there isnt any wish to change the state of the server
21	/pointsOfInterests/addPoint	POST	name, category, rating, description, picture	string: success_msg or error_msg	using POST because we want to create a new point of interest
22	/reviews/userId/:id	GET	user_id	Returns all reviews of specefic user: reviews[]	Using GET because a resource from server is needed and there isnt any wish to change the state of the server
23	/reviews/add/rate	POST	userId,pointId, rate	True: string: msg_success, False: msg_failure	using POST because we want to create another review on a point of interest in DB
24	/reviews/add/reviewMsg	POST	userId,pointId, reviewMsg, reviewDate	True: string: msg_success, False: msg_failure	using POST because we want to create another review on a point of interest in DB
25	/reviews/pointId/:id	GET	point_id	Returns all reviews of specefic point: reviews[]	Using GET because a resource from server is needed and there isnt any wish to change the state of the server
26	/reviews/2Latest/pointId/:id	GET	point_id	Returns 2 latest reviews of point: latest2Reviews []	Using GET because a resource from server is needed and there isnt any wish to change the state of the server
27	/categories/	GET	-	Returns all categories: categories[]	Using GET because there is no information to deliver to the server or any wish to change any information.
28	/categories/delete/	DELETE	-	True: string: msg_success, False: msg_failure	using DELETE because we want to delete all categories from the DB
29	/categories/delete/:category	DELETE	category	True: string: msg_success, False: msg_failure	using DELETE because we want to delete the requested category
30	/categories/init	POST	-	True: string: msg_success	using POST because we want to create all categories in the DB from the xml

31	/countries/	GET	-	Returns all countries: countries []	Using GET because there is no information to deliver to the server or any wish to change any information.
32	/countries/delete/	DELETE	-	True: string: msg_success, False: msg_failure	using DELETE because we want to delete all countries from the DB
33	/countries/delete/:country	DELETE	country	True: string: msg_success, False: msg_failure	using DELETE because we want to delete the requested country
34	/countries/init	POST	-	True: string: msg_success	using POST because we want to create all countries in the DB from the xml