

## API

ID	URI	HTTP Method	Parameters	Returns	Explanation
1	/USERS/Login	POST	{"Username", "Password"}	Token	Login cannot pass in GET (there's password involved which makes it not safe)
2	/USERS/Register	POST	{"USER_NAME", "FirstName", "LastName", "City", "Country", "Email", "PASSWORD", "QuestionAnswers"[], "favoriteCategories"[] }		<p>The user will choose at least two favorite categories. The user will answer all questions. Register cannot pass in GET (not safe)</p> <p>True=succeed False = failed</p>
3	/USERS/getRandomQuestion	GET	USER_NAME	{"QUESTION_ID"}	Using GET because there is no information to deliver to the server or any wish to change any information.
4	/USERS/RetrievePassword	POST	{"Username", "QUESTION", "ANSWER"}	{"PASSWORD"}	<p>User details stored in the DB and only the server can GET them.</p> <p>Using GET because there is no information to deliver to the server or any wish to change any information.</p>
5	/USERS/getAllQuestions	GET		QUESTIONS[] [{"QUESTION"}]	No need to be safe, no update in DB.
6	/POI/getThreePopularRandomPoints	GET		Point[] [{"POI_ID", "NAME", "IMAGE", "NOV", "NOR", "DESCRIPTION", "RANK", "CATEGORY"}]	Returns three points of interest. The three Points of interest are random points from all Points which their popularity is above specific rank.

					Using GET because there is no information to deliver to the server or any wish to change any information.
7	/POI/getTwoMostPopularPoints	GET	Token	Point[] [{"POI_ID", "NAME", "IMAGE", "NOV", "NOR", "DESCRIPTION", "RANK", "CATEGORY"}]	Returns two points of interest. The two Points of interest are most popular and related to user's favorite fields. Using GET because there is no information to deliver to the server or any wish to change any information.
8	/POI/ getTwoLastSavedPoints	GET	Token	Point[] [{"POI_ID", "NAME", "IMAGE", "NOV", "NOR", "DESCRIPTION", "RANK", "CATEGORY"}]	Returns two points of interest. The two Points of interest are two last points from list of user saved points. If list is empty return NULL. Using GET because there is no information to deliver to the server or any wish to change any information.
9	/POI/getTwoLastReviewsOnPoint	GET	POI_ID	REVIEWS[] [{"REVIEW_ID", "POI_ID", "REVIEW", "DATE"}]	No need to be safe, no update in DB.
10	/POI/getAllPoints	GET		Point[] [{"POI_ID", "NAME", "IMAGE", "NOV", "NOR", "DESCRIPTION", "RANK", "CATEGORY"}]	Returns list of all points of interest. The points sorted by category. Using GET because there is no information to deliver to the server or any wish to change any information.
11	/POI/saveFavoritePoint	POST	Token, {"POI_ID"}		Using POST because we need

					to deliver the Point_ID to the server for adding it to the DB of favorite Points of the specific user.
12	/POI/removeFavoritePoint	DELETE	Token, {"POI_ID"}		Using DELETE because we need to deliver the Point_ID to the server for deleting it from the DB of favorite Points. True = succeed False = failed
13	/POI/getAllSavedPoints	GET	Token	Point[] [{"POI_ID", "NAME", "IMAGE", "NOV", "NOR", "DESCRIPTION", "RANK", "CATEGORY"}]	Returns list of points of interest. The Points of interest are all points from list of user saved points. If list is empty return NULL. Using GET because there is no information to deliver to the server or any wish to change any information.
14	/POI/saveFavoritePointsToServer	POST	POIS_Array[] [{"POI_ID"}], Token		Using POST because we need to update the DB
15	/POI/reviewPoint	POST	{"POI_ID", "REVIEW", "rank" }		Using POST because we need to update the DB by adding the review to the list of reviews of an existing point.
16	/POI/Category	GET	CATEGORY	Point[] [{"POI_ID", "NAME", "IMAGE", "NOV", "NOR", "DESCRIPTION", "RANK", "CATEGORY"}]	Using get because no need to be safe, no updates on DB. Return all points of interest from the given Category.
17	/POI/setNewOrderForSavedPoints	POST	Point[] [{"POI_ID", "POSITION"}]		Using POST because we need to update the DB