API Endpoints

Hard URL	Method	Fields	Response
`/api/user/create`	POST	`username: String` user's username `password: String` user's password `name: String` name of the user	<u>Username unavailable</u>: status 409 <u>Create successful</u>: Created user in JSON format
`/api/user/login`	POST	`username: String` user's username `password: String` user's password	<u>User not found</u>: status 404 <u>Incorrect password</u>: status 403 <u>Login successful</u>: Found user in JSON format
`/api/tracker`	POST	`tracker_id: Integer` tracker's database id	<u>Tracker found</u>: Found tracker in JSON format <u>Tracker not found</u>: status 404
`/api/tracker/create`	POST	`tid: String` the manufacturer tracking id `name: String` name of the tracker `uid: Integer` owner's id	Created tracker in JSON format
`/api/tracker/delete`	POST	`id: Integer` tracker's database id	Deleted tracker in JSON format
`/api/tracker/checkin`	POST	`id: Integer` tracker's database id `long: String	Float` current longitude (-180, 180)  `lat: String	Float` current latitude (-90, 90)  `time: String` logging time, should be in ECMA-262 format	Updated tracker in JSON format
`/api/tracker/get`	POST		List of all trackers in JSON format
`/api/tracker /getbyuser`	POST	`uid: Integer` user id whose bikes to get	List of all trackers, in JSON format, whose`ownerID` is the same as `uid`

Notes:

- `/api/tracker/checkin` does not validate data before transaction
- user in JSON format has feilds as in the `User` table in `Database Schemes` document
- tracker in JSON format has feilds as in the `Tracker` table in `Database Schemes` document