**QAbify Challenge 2018**

The chosen language for this challenge is Javascript; in the git repository you can find two separated solutions, one using Botkit, a library for bot integration with Slack and another one using javascript and Jasmine.

To run the code in the SlackBot.js solution it’s mandatory to install ‘botkit’ running npm install ‘botkit’ –dev (<https://github.com/howdyai/botkit/blob/master/docs/readme.md#botkit---building-blocks-for-building-bots>). You will have to install some other packages too, but they are all included in package.json, so, running npm install should be enough. The bot connects to Slack and hires the nearest taxi to any user who talks to the bot saying “taxi”. I tried to get user location with a lot of packages but I couldn’t, so the data is mocked to test if it works.

The other solution has two files: botClass.js with the logic of the bot and bot.spec.js with the testing code to run in SpecRunner.html; the testing suite is set to run with mocked data and the real API calls are disabled. You can run botClass.js getting data from the API but I wasn’t able to manage asynchrony in the test suite. To test this solution I’ve chosen to install Jasmine standalone (<https://github.com/jasmine/jasmine#installation>).

The logic used in both solutions is the same: the user calls the bot by texting some phrase with the word “taxi” on it; then, the bot gets user location and searches for any available taxi. The one with the shorter distance to the user’s location is chosen and its state is changed to “hired”. I have set a maximun distance value to compare locations, so if the nearest taxi is 100km away from the user’s location, the bot informs that there’s no available taxi at that moment. False locations have been set to test the code and verify that the hired taxi is correct.

I had some issues with the API trying to hire a taxi since it randomly responded with errors. Also, the endpoint GET /taxis/<city>/<car\_name> only takes into account the car name, ignoring the city value.