Test scenarios for thecocktaildb

Product:	thecocktaildb
Test area:	[Production]: https://www.thecocktaildb.com/api.php
Test Case Author:	Klyment Malakhov
Last updated date:	8 Jun 2019
List of assumptions:	1. We assume that there are access for DB, server, Developer team 2. There're only test case for API testing block: Method GET Search and Lookup, and that is enough for today 3. There's no restrictions for choosing tools for automation and manual testing 4. Main goal is agreed with stakeholders and it is growing of number user who pays for using APIs. Based on it, was put priority of tests

API testing

1. Testing API for key /1/ without Patreon supporters				Tools	Priority
		An open API helps to attract clients for its full use. It is important to consider that delay in response from service can push customers to pay for a more stable version. People will get tired of the unstable functionality and they may want to buy.		**Java, Rest assured**	**Normal**
1.1. Method GET Search					
	TC_11_001	POSITIVE - Test look up details functionality for cocktail	PASSED	Automated	Normal
	TC_11_002	POSITIVE - Test look up details functionality for ingredient	PASSED	Automated	Normal
	TC_11_003	NEGATIVE - Test look up details functionality for cocktail and ingredient	PASSED	Automated	Low
	TC_11_004	NEGATIVE - Test look up details functionality with empty id	PASSED	Automated	Low
	TC_11_005	NEGATIVE - Test look up details functionality with incorrect id	PASSED	Automated	Low
	TC_11_006	NEGATIVE - Test look up details functionality with nonexistent id	PASSED	Automated	Low
	TC_11_007	NEGATIVE - Test look up details functionality for trying to delete some data by id	PASSED	Automated	Low
	TC_11_008	NEGATIVE - Test look up details functionality for trying to put some data by id	PASSED	Automated	Low
1.2. Method GET Lookup					
	TC_12_001	POSITIVE - Test search functionality for cocktail	PASSED	Automated	Normal
	TC_12_002	POSITIVE - Test search functionality for ingredient	PASSED	Automated	Normal
	TC_12_003	NEGATIVE - Test search functionality for cocktail and ingredient	PASSED	Automated	Low
	TC_12_004	NEGATIVE - Test search functionality with empty id	PASSED	Automated	Low
	TC_12_005	NEGATIVE - Test search functionality with incorrect id	PASSED	Automated	Low
	TC_12_006	NEGATIVE - Test search functionality with nonexistent id	PASSED	Automated	Low
	TC_12_007	NEGATIVE - Test search functionality for trying to delete some data by id	PASSED	Automated	Low
	TC_12_008	NEGATIVE - Test search functionality for trying to put some data by id	PASSED	Automated	Low
1.3. Method GET Filter					
1.4. Method GET List					
2. Testing API for key /1/ with Patreon supporters				Tools	Priority
		This functionality is the main source of cash flow of this project. Therefore, we need to make sure that it is reliable and that there are no errors in its use.		**Java, Rest assured, Mocks are required - SoapUI**	**High**
2.1. Method GET Popular					
2.2. Method GET Filter - by multi-ingredient					
2.3. Method GET Recent					
3. Testing Image				Tools	Priority
		Displaying a picture is an important part of the functionality for API. The partners of this project are using APIs that reuse in other applications.		**Java, Rest assured, Server data validation**	**Normal**
3.1. Method GET Image medium size					
3.1. Method GET Image small size					
3.1. Method GET Image normal size					
4. Testing rights for the data uploading				Tools	Priority
		This functionality is available for the paid version. You also need to take into account that bars and bartenders may have rights to use it without payment.		**Java, Rest assured, Server data validation**	**Normal**
4.1. Testing image uploading					
4.2. Testing ingredient uploading					
4.1. Testing cocktail uploading					
4.1. Testing glasses uploading					

Security testing

1. Testing security standards	Tools	Priority
As consumers of the API are other applications (bars, bartenders, organization), you need to protect them from the developer errors. To do this, the application should be tested to find security holes.	**OWASP**	**Normal**
1.1. Security testing open APIs		
1.2. Security testing using key for Patreon supporters		
2. Testing security sensitive data	Tools	Priority
The payment system for clients must be secure to avoid the risk of theft of payment data. Also, it is needed to make sure that the unique key of the customer who bought the product cannot be stolen.	**Using standards OSSTMM or Sans. We will have to mock a DB, payment system.**	**High**
2.1. Security testing payment data (cards, bank account)		
2.2. Security testing of possibility of stolen user key		

Performance testing

1. Performance testing of open API	Tools	Priority
We need to be aware of capacity our open APIs. That gives us insurance that product can be attractive for new users	**Gatling or Jmeter**	**Normal**
1.1. Should cover all existing API		
2. Performance testing of API for Patreon	Tools	Priority
After buying the product, the API should be more stable! That is main benefit for client	**Gatling or Jmeter**	**Normal**
2.1. Should cover all existing API		

Stability testing

1. Stability testing of open API	Tools	Priority
The stability of working all service instances should be monitored. The notification of emergency case should be pushed to responsible peoples	**Grafana (server monitoring), Kibana (log monitoring)**	**Normal**
1.1. Increase number of instance and validate changes using performance tests		
1.2. Decrease number of instance and validate changes using performance tests		
1.3. Emulate high load test, stress testing		
1. Stability testing for Patreon APIs	Tools	Priority
Those part is more important, because here tests cover payments as well. The stability of working all service instances should be monitored. The notification of emergency case should be pushed to responsible peoples	**Grafana (server monitoring), Kibana (log monitoring)**	**High**
1.1. Increase number of instance and validate changes using performance tests		
1.2. Decrease number of instance and validate changes using performance tests		
1.3. Emulate high load test, stress testing		