

Test Plan

Contents

Introduction.....	2
Test Items (Functions)	2
Software Risk Issues	3
Features to be tested	3
Features not to be tested.....	4
Testing Approach.....	4
Entry Criteria:	5
Exit Criteria:	5
Test Deliverables	5
Environmental Needs	5

Introduction

This test plan is aimed to describe the test process of the 'Swagger Petstore' project according the MAIN TASK of the "Dev Challenge 12" Championship (QA direction). This document let us receive the understanding of the testing scope, the testing approaches applied to, the test environment and the test deliverables.

Test Items (Functions)

Following test items are planned to be tested:

Requirement Type	User's flow	Main flows	Test Item
Functional	Create new pet	Positive flows	Create new pet
			Create new pet with several added tags
			Create new pet with several added URLs
		Negative flows	Create new pet with non-predefined tags
			Create new pet with the same added tags
			Create new pet with the same added URLs
		Fields validation	All fields save valid data
			All fields don't save invalid data
		Form fields validation	Create new pet by filling all required and non-required fields
			Create new pet by filling only required fields
			Create new pet by filling not all required fields
			Create new pet by filling only non-required fields
		Tags sorting	Tags are sorted after saving the entered data during creating new pet
	Update pet	Positive flows	Update all fields for existing pet
			Update only required fields for existing pet
			Update only non-required fields for existing pet
			Update empty fields for "Tag"/"Category" fields
			Update "Status" field
			Update the quantity of added tags
			Update the quantity of added URLs
		Negative flows	Update non-existing pet
			Update existing pet with empty required fields
			Update existing pet with added several tags by filling the empty tags
			Update existing pet with added several URLs by filling the empty URLs
			Update status for pet with non-predefined status
	Delete pet	Positive flows	Delete existing pet
		Negative flow	Delete non-existing flow
	Search	Positive flows	Find pets with available status
			Find pets with pending status

Requirement Type	User's flow	Main flows	Test Item
			Find pets with sold status
			Find all pet with available or pending status
			Find all pets with sold or pending status
		Negative testing	Find all pet without predefined statuses
	Search	SQL injection	Find pet by ID with adding ' symbol
Security	Create new pet	XSS injection	Find pet by ID with adding -1 symbols
			Create new pet with XSS injections in text fields

Software Risk Issues

The 'Swagger Petstore' project is the client-server web-application. The server part of the application is planned to be tested.

The critical areas of testing may be following issues:

- Ability to use and understand a new tool;
- Misunderstanding the main requirements due to lack of knowledge for further planned delivered features;
- Global server problems with access for testing due to arising quantity of QAs due to absence of communication among QA team;
- Rising quantity of corrupted data in DB due to absence of documentation and regulation regarding using the format of test data and test users;
- Poorly documented models that will be used for testing;
- Government regulations and rules

Features to be tested

Following features are planned to be tested:

- New pet creation;
- Existing pet updating;
- Pet deleting;
- Search for pets

All of the following features have next risks not to be tested or delivered in time:

Priority of risk	Risk	Feature	Preventive Actions
High	Providing the poorly security testing due to lack of knowledge	New pet creation	1. Find more practical information regarding security testing in Internet. 2. Find more information about usage of XSS injection. 3. Find more information about the security of web-application.

Priority of risk	Risk	Feature	Preventive Actions
Medium	Using not proper tools for testing due to lack of experience with Postman working	Existing pet updating	1. Find more detailed information regarding difference between POST and PUT queries
Medium	Providing the poorly security testing due to lack of knowledge	Search for pet	1. Find more practical information regarding security testing in Internet. 2. Find more information regarding using SQL queries in the testing process.
Low	Providing poorly test coverage for main functionality due to absence of documentation related to other components for searching	Search for pet	1. Read the provided documentation more deeply in order to find all the models and links. 2. Investigate the application using the syntax of post/get queries

Features not to be tested

Following features are not to be tested during this release cycle:

Features	Reason
Create new user	This features weren't listed in the MAIN TASK. So they will be tested in the next release or they have been already tested in the previous release.
Create the list of users	
Login	
Logout	
Search for existing user	
Update user	
Delete user	
Upload image for pet	This feature isn't planned to be tested due to lack of knowledge and experience of QA.
Create new purchase	This features are listed in the BONUS TASK. And they will be tested in case of time availability.
Update new purchase	
Delete new purchase	
Search for purchase	

Testing Approach

Following testing types will be used during testing the listed features:

- Exploratory testing and error guessing– will be used for requirement clarification and investigation;
- Functional testing – will be used according to existing documentation (described models and queries)

The server part of application will be tested with using of API that were provided by organizers. All the designed verifications and tests will be provided manually.

Entry Criteria:

- The test environment must be ready and accessible;
- The needed requirements are provided;
- All testing executive artefact are ready

Exit Criteria:

- All designed verifications and tests are run;
- All found defect are investigated and reported;

Test Deliverables

After finishing the testing of the listed features following test deliverables will be provided:

- Test plan document;
- Tools and their outputs (repository with all created queries);
- Defect reports;
- Summary report with corrective actions;
- Presentation of the results

Environmental Needs

Following environment will be used during testing:

- OS Windows 10 Pro N, version 1709, build 16299.309
- Postman v.6.0.10