

Mr Buggy 7 – API Testing TEST PLAN

1. Introduction

This document showcases a sample test plan used in the API application testing process, based on the Mr Buggy 7 app. The goal of Mr Buggy is to efficiently manage Change Requests for multiple Providers. The program contains simplified database for the requests and has a fully functional account system with different access levels. Following test plan will present basic informations about testing process, available environment and tools, responsible team, documentation and final products of the task.

2. Test approach

API testing will be based on the specification document provided by the client, describing existing resources and available HTTP methods. API of Mr Buggy 7 app consists of four branches: USER, UNIT, PROVIDER and CR. Due to the budget constrains (and showcase nature of this testing project), the team will perform scaled down testing, narrowed down to the USER branch. Resources and corresponding methods will be tested via test cases stored in one file (MB7-ATC). Test cases will be designed around three different approaches – positive, negative and destructive. For each resource the following attributes of the request's response will be verified: status code, payload, state and headers.

List of tested resources with corresponding valid methods:

| |
|---|
| HEAD /users – returns headers of the resource |
| GET /user – returns users list GET /user/ID – returns user data based on the provided ID number GET /user/profile – returns the data of logged-in user |
| POST /user/TYPE – creates new user of given TYPE (admin, employee, provider) |
| PUT /user/ID – modifies the account of user with provided ID number PUT /user/ID/block – blocks the account of user with provided ID number PUT /user/ID/active – unblocks the account of user with provided ID number PUT /user/profile – modifies the data of logged-in user |
| DELETE /user/ID – deletes the account of user with provided ID number |

3. Entry criteria

Testing may begin after criteria listed below are met:

- initial Test Plan document is accepted by the Test Director
- application is in stable build and deployed on the development environment
- environment is prepared (hardware and software listed in the "Environment" and "Tools" sections are installed/functional)
- issue tracker is accessible by the team (created repository with appropriate access levels granted to the team members)
- the team is ready to start (team members delegated to the task have a knowledge of the necessary tools and have received project documents with other testware assets)

4. Exit criteria

Testing may end after one of the criteria listed below is met:

- critical issue is blocking further testing (testing is postponed)
- designated test cases were completed by the tester

5. Issue tracking and management

Issues reported during the projects will be stored in the external web-based repository on GitHub (read more at "Tools"). Defects will be graded by the single metric called "Priority", which indicates the severity of the problem and its impact on the software, therefore an urgency to fix.

Priority will be graded accordingly:

- **Critical** – for issues affecting application stability; major security risks
- **High** – for issues affecting key functionalities; minor stability and security risks
- **Medium** – for issues affecting side functionalities; major content or validation defects
- **Low** – for cosmetic issues; lesser content or validation defects

6. Test products

The project will result in following products:

- Test plan (ID: MB7-ATP)
- Issue database (ID: MB7-A01 -> MB7-AXX)
- Test cases document (ID: MB7-ATC)
- Test report (ID: MB7-ATR)
- exported Postman Collections (ID: MB7-API-1, MB7-API-2, MB7-API-3)

7. Environment

Tests will be conducted in the following environment:

Lenovo ThinkPad T440p

- Intel(R) Core(TM) i7-4712MQ CPU @ 2.30 GHz
- Intel(R) HD Graphics 4600
- 8.00 GB RAM
- 1920x1080 Display
- Windows 10 Pro 64-bit, version 21H2, build 19044.1826

8. Tools

List of tools and repositories used during tests:

- Postman v9.29.0 (main tool for creating and managing API requests)
- ShareX 14.1 (video capture)
- Lightshot 5.5.0.4 (screenshots and image editing)
- Paint (image editing)
- LibreOffice 7.2.7.2 (documentation, issues, exporting to PDF)
- Git with GitHub integration (issue repository, project documents repository)

9. Team

List of people responsible for the project:

- Adam Nowak – Test Lead
- Jan Kowalski – Tester

10. Project documents

Additional documents provided by the client:

- MrBuggy-Functional-Specification-v1.pdf – product specification
- MrBuggy-API-Postman-v1.pdf – product's API specification