Test plan

General Information

Project name	Testing an online banking website
Created by	Paula Humeniuk
Preparation date	22/02/2024
Version	1.0
Status	Approved

Revision history

Version	Description	Author	Date	Approved by	
				Author	Date
1.0	Document created	Paula Humeniuk	15/02/2024	John Doe	22/02/2024

Table of contents

1. Test objectives 🏹

The purpose of creating this test plan is to thoroughly test the functionality of the website based on the provided specifications. The website address is **https://www.hsbc.co.uk/**

All recognized defects 🐞 will be corrected by the developers and passed on to testers for retests and regression testing ✅.

2. Scope 🧺

Test levels covered in the test plan:

- Unit Testing
- API Testing
- Integration Testing
- System testing

Types of tests covered in the test plan:

- Functional tests - the software will be validated against the provided specifications
- Integration tests - Api testing
- Automatic tests - implementation of automatic tests for key system functionalities according to the provided test cases
- UI Testing - focused on checking the appearance, functionality, and usability mainly of Graphical user interface (GUI)
- Compatibility Testing - 4 newest web browsers and devices

Types of tests excluded from the test plan:

- Security testing - lack of specialists in the field of pen testing within the team. This field of testing will be outsourced.

3. Test subject 🔬

The subject of tests is the HSBC UK website available at https://www.hsbc.co.uk/ along with all its functionalities.

4. Acceptance criteria 👌

1. Performance tests

- Server response time cannot be longer than 500 ms
- The software must withstand the load of 1,000 users at the same time

2. Functional tests

- The whole software product works according to the requirements, and no significant errors appear in the application. A software product must pass all the planned tests

3. Automatic tests

- All test cases have been automated
- Tests have been connected with CI/CD tool - Jenkins

---

5. Failure criteria ⛔

1. Performance tests

- Server response time is longer than 500 ms
- The software cannot handle the load of 1,000 users at the same time

2. Functional tests

- The whole software product won't work according to the requirements, and significant errors appear in the application. A software product won't pass all the planned tests

3. Automatic tests

- Test cases haven't been automated
- Tests haven't been connected with CI/CD tool - Jenkins

---

6. Entry criteria 🚪

- Test environment launched
- The test environment must be close to the production
- Access to IOS device

---

7. Exit Criteria 🤝

- All test cases have been executed
- All types and levels of test covered by this test plan have been performed
- All defects have been fixed

---

8. Functionalities to test 🔎

- Logging into the portal
- Updating user's details
- Making a payment / sending money
- Applying to increase, decrease or removing overdraft
- Set up, amend or cancel a standing order.
- Integration with the international transfers
- UI - validation of data fields, navigational elements, menu items and all other main pieces of GUI

---

9. Test environment 🎄

Tests of the online bank website will be carried out on a dedicated server https://www.test-hsbc.co.uk . The testing environment has parameters very close to production.

Specification:
Processor: 2GHz
RAM: 4GB
Disk: 240 GB
System: Linux Debian 17

Tests will be performed using the following physical devices:

1. Personal computer (PC)
- Processor I3 3,6 GHz
- RAM: 16GB

2. Iphone 13
- Processor: Apple A15 Bionic
- RAM: 4GB
- IOS version 17

---

10. Defect priorities 🐞

We distinguish 5 priorities of defects in our test plan:

- Malfunction
- Blocking error
- High priority
- Medium priority
- Low priority

---

11. Test site 🏠

Tests will be run in at the headquarters of the software development company.

---

12. Schedule & Estimation 🗓️

1. Static testing:
- Verification of specifications
- Verification of User Stories

⏱️Time: 10h

2. Functional testing:
- Verification of implemented functionalities in accordance with the requirements
- Test cases execution
- Reporting all defects

⏱️Time: 30h

3. Performance testing:
- Server response time verification
- Load test

⏱️Time: 14h

4. Automatic testing:

- Preparation of test structure
- Implementation of automatic tests based on provided test cases
- Connection of tests with Jenkins tool
- Connecting tests with Grafana tool

⏱️Time: 40h

---

13. Test deliverables/Reports 📊

- Test cases
- User stories
- Defect Report
- Automatic test scripts
- Performance Testing Metrics

---

14. Tools 🛠️

1. Issue tracking:
- Jira

2. Test cases:
- TestLink
- XRay

3. Automation:
- WebdriverIO
- SuperTest
- Allure
- Jenkins
- Grafana

4. Performance testing:
- Jmeter

5. Screenshots and screencast software:
- PicPick
- ShareX

---

15. List of browsers 🌐

- Chrome
- FireFox
- Edge
- Safari

16. Incident management 🚩

Once a defect is detected, it will reported in the defect management system - Jira. The defect will be given a priority and a person responsible for fixing it will be assigned to it.

When the developer corrects such an error, it will then be passed on to the tester for both, retesting and a regression testing.

If the defect is repaired incorrectly and bug still occurs, an issue will be re-opened and assigned to the developer for re-analysis.

17. Roles and responsibilities within Test Team 👩‍👩‍👦👨‍👦

- Project Manager - Managing the whole testing process and providing all the needed resources for the testing activities.
- QA Tech Lead - Managing the QA team from a technical perspective and analyzing the tasks and distributing them between team members. Additionally, creating test documentation, including test cases, test plans, etc. and supporting Project Manager
- QA Engineer - executing tests and logging all found errors into the bug tracking system.