**Test plan**

**100carrots**

**1) Introduction**

**100carrots** – site about jewelry where you can earn money by answering questions.

After registration on the site, a section with a balance, section with questions and a jewelry store is available. You can buy something if there is money on the balance sheet. And for this you need to answer the right questions. For an incorrect answer, the balance is minus.

**2)** **Scope of work**

Test registration, entry and exit, answers on questions and buy of jewelry on the site.

What do you need to have?

Any browser. For example, Google Chrome.

Have Microsoft Word and Excel to work with.

Use Telegram to communicate with the team.

Use Jira to deal with bugs.

Use Сhrome DevTools in Google Chrome browser.

Using Bandicam and Lightshot to record videos and screenshots.

The mobile version of the site will be tested.

Do test adaptation to translation from a browser

**3) Quality and acceptance criteria**

All the necessary documentation has been collected: test cases and bug reports.

At the time of completion of testing, the product should not have bugs with the critical and major statuses.

All bug reports are recorded in the bug tracker Jira.

**4) Critical success factors**

If there are no delays from the development team, then everything will successfully reach the end. If the test documentation is ready and checked, then there should be no problems. Access to all programs and systems must also be provided, in my case it is Jira. Discussion of all processes of work with PM and BA.

**5) Risk management**

The Internet may stop working. Then you can switch to mobile Internet. A VPN can come in handy if the internet is unstable. Team members may get sick. Need to think about a replace and employeed. Or that have people on team who can close several positions. Also, team can have people from different countries, and these are different time zones. Windows may fail on a laptop, then you need to quickly reinstall Windows, know how to do it and have a flash drive and a disk image at hand.

**6)Resources (Key project resources, Test team, Test hardware, Test tools)**

Yana, Dmitry, Ilya - team. Programs: Jira, any browser, or declared by the customer (you can work with several browsers), Minimum system requirements: OS - Windows 10. iron core i-3, RAM 4 GB.

**6.4 Test tools**

Use the Jira bug tracking system. Put bugs there. Excel, running a test case in it. Use Microsoft word to write a test report.

**7) Test documentation**

Test plan: Responsible - Ilya, compiled once before testing, storage location - Google drive. Or telegram chat.

Test cases: Responsible - Dmitry, compiling before testing, storage location - Google drive. Or telegram chat.

Yana writes bug reports using Jira. All team members have access to the Jira.

Test report: Responsible - Ilya, compiled once before testing, storage location - Google drive. Or telegram chat.

**8) Test strategy**

**8.1 Entry criteria**

First of all, the project documentation must be prepared. Removed unnecessary and incompatible ideas with the project (at the stage of compiling the documentation, such moments should be found).

Technique must be prepared. Required operating system.

**8.2 Test methods**

Manual testing

**8.3 Test types**

Increase the efficiency of detailing - System testing. Code access verification is a black box method. By degree of development: danger. The principle of working with the application: positive and negative. Non-functional testing (by object) - security check, GUI Non-functional: usability. On code start - static.

**8.4 Test levels**

Smoke Testing + DV

**8.5 Bug and documentation tracking**

Track down bugs with test cases. Write them down in excel. And enter them into the Jira bug tracking system.

**8.5.1 Bug severity definition**

For severity and priority, will use gradations: critical, major, average, minor, enhancement - severity, blocker, critical, major, minor – priority.

**9) Testing schedule**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Activities** | **Date start** | **Date end** | **appointment** | **location** | **content of the work** |
| **Test plan** | **26.05.22** | **27.05.22** | **Detailed description of the work plan** | **Minsk** | **Introduction,** **Scope of work,** **Quality and acceptance criteria, Critical success factors, Risk management,** **Resources, Test documentation,** **Test strategy, Testing schedule** |
| **Test cases** | **26.05.22** | **28.05.22** | **List of modules and their functionality to be tested** | **Minsk** | **module check** |
| **Bug report** | **28.05.22** | **30.05.22** | **Control bugs** | **Szcecin** | **Identify bugs, describe them and bring them into the system** |
| **Test report** | **30.05.22** | **31.05.22** | **test results** | **Minsk** | **described what was tested. The launch of the sprint and its duration, what bugs were found and what needs to be fixed in the first place.** |