OpenBazaar Redevelopment Test Plan

The Fair Traders
Daniel Mandel - mandeldr
Shandelle Murray - murras25
Connor Sheehan - sheehacg

December 7, 2015

Abstract

This document outlines the test plan for the Openbazaar redevelopment project.

Contents

Revision History

Revision Number	Revision Date	Description of Change	Author
1	October 20th, 2015	Created Test Plan Document	Daniel Mandel
2	December 7th, 2015	Remove messaging testing. Add encryption and GUID testing plans.	Connor Sheehan

Table 1: Table to capture the history of the document

Test Plan Identifier

Version 1.

References

IEEE Test Plan outline is used.

Introduction

This is the test plan for the OpenBazaar desktop application. It will include only the most relevent portions of the project that are related to the process of application initialization, creating contracts and the browsing of products and stores. The main intention of this plan is to simulate a user interacting with the system in real time in order to ensure the robustness and reliability of the application.

Test Items

The following is a list of the most significant portions of the application to be tested:

1. Overall ease of use for sample users including the process of installation.

2. Correctness of messages being sent and received between mock server and application front-end

3. Security of Ricardian Contracts in order to prevent unauthorized changes.

4. Functionality of a user related to viewing a market's items, creating a Ricardian contract, and purchasing items.

5. The application initialization procedures.

6. Data security within the application.

Software Risk Issues

With an online decentralized marketplace, there comes significant software risks that are out of the developer's control. Some of these risks may include the following:

- The selling of illegal goods or services.

- IP spoofing and false identification/location of a user, marketplace, or notary.

- The price of a bitcoin.

Features to be Tested

The following is a list of functionality aspects to be focused on:

- A buyer must have the ability to view markets and market items correctly from market to market.

- A seller must have the ability to post items for sale with the correct product details.

- The notary must have the ability to offer notary/mediation services.

- The application must initialize a user with unique globally unique identifier (GUID).

- The application must encrypt user data to ensure anonymity and confidentiality.

Features not to be Tested

The following areas will not be focused on during the testing of the application:

- The appearence or general look and feel of the application

- The network or ability to search for peers on the network

- The accessibility requirements of the project

Approach

The relevant components of this test plan relate to the GUI and the controllers in the application. User input into text fields, drop down panels and other graphical boxes will need to be checked for validity and security. On a commerce related application, injections and other malicious methods are viciously pursued by people with bad intentions. Unit testing of functions which convert input into machine readable format will ensure a safer trading environment for users. Since the creation of a reliable and secure peer-to-peer network is outside the scope of this project, we will implement the principle of hardware hiding within our application. Our GUI will then use an interface that could be applied to any network. This API will be unit tested to ensure it is valid and secure to use. On this platform, Ricardian Contracts are used to create both machine and human readable agreements about transactions between parties. The functions used to turn user input into a Ricardian contract will be unit tested, verifying that the ID created is global and unique, signatures are valid, and content does not contain any injections, malicious code or otherwise undesired data. Each contract is also accompanied by a hash to provide authenticity, which can be tested for accuracy.

In order to test ease of use, a sample user, such as a person that does not specialize in software, will be timed performing various tasks such as creating a contract, buying an item, and installing the software. An acceptable time limit in order to satisfy the ease of use requirement will be determined at a later date.

Item Pass/Fail Criteria

There will be a variety of criterion that all unit tests will have to pass in order to satisfy the functional and nonfunctional requirements.

Suspension Requirements and Resumption Requirements

This is the resumption/suspension requirements.

Test Deliverables

- Test Plan Revision 0

- Test Report Revision 0

- Test Plan Revision 1

- Test Report Revision 1

Remaining Test Tasks

Following the completion of this test plan, there will still be a significant amount of testing for the project. The networking component of the project will need thorough tests to ensure secure communication between nodes. The network will be tested to ensure that two nodes can exchange messages and find each other through other networked nodes. Messages between nodes will need to be checked to ensure nobody is attempting to spoof the identity of another user.

Environmental Needs

The following are needed in order to support the testing:

- PyUnit Framework

- Python IDE such as PyCharm

- A desktop computer with OS X or Linux installed with the bash terminal

Summary

OpenBazaar will exist only if the main users have the ability to do what they need, and the application's technical details are correctly implemented. Buyers need to have the ability to post items for sale and the information that goes along with it. Sellers need to be able to view all of the items in a particular marketplace as well as each one individually. The application must meet confidentiality requirements, and uniquely identify users without requiring personal information. This test plan provides a framework for how the functionality will be tested for each significant role in the OpenBazaar.