

# Proof of Concept Plan

The Fair Traders

Daniel Mandel - mandeldr

Shandelle Murray - murras25

Connor Sheehan - sheehacg

October 2, 2015

## Abstract

This document contains a detailed plan for the implementation of a Proof of Concept plan for the OpenBazaar redevelopment project.

## 1 Introduction

Building a decentralized e-commerce network is an exciting endeavour with many fundamentally important components. The project has a unique set of functional requirements and many equally essential non-functional requirements that must be met. From the technical perspective, we must build a distributed computer system (network) that can communicate between nodes without the use of a centralized server. We must also build a graphical front-end to display this information to the user in a visually appealing way. From an economic perspective, we must build a system to ensure that trades are well defined and that each side understands the contractual nature of the exchange. Each of these approaches to the project has different risks and considerations.

## 2 Plan

Our plan is to implement both the server and client components of this project. We plan to implement a new front end; however, we will mostly use code that is already available in order to implement the server portion of the project. Essentially, we will create a daemon to listen on a computer port, which will access data from other network nodes as well as provide information to the graphical front end application. Using the graphical interface, we will allow users to create online stores, post product listings and prices, and initiate trade deals with other users on the network.

## **3 Risks**

### **3.1 Will a part of the implementation be difficult?**

The most difficult technical component of the project to develop in the given time would be creating an entire peer-to-peer network. Since the idea of Open-Bazaar is to be decentralized, its main requirement is that there is no main server hosting all of these peers. This would be very complex to create and maybe even beyond the scope of the project. After research we have discovered libraries in Python and Java that would allow us to use their framework in creating our peer-to-peer network. If this proves too difficult to implement, the entire network could be simulated using hardware hiding principles.

### **3.2 Will testing be difficult?**

Testing will be difficult for an implementation of a real peer-to-peer network. If we were to use a mock network and the MVC framework to develop the front end, testing would be much easier and could be done using something like PyUnit or JUnit.

### **3.3 Is a required library difficult to install?**

We have not discovered any necessary libraries thus far that are difficult to install or use on our machine.

### **3.4 Will portability be a concern?**

Depending on which language we choose to develop this, either Python or Java, the user will need to have those languages installed. That being said, we can bundle our software with the installer of those languages as needed.

### **3.5 Will the project size/scope be feasible?**

Implementing the entire project from scratch is not feasible within the scope of this project. Reimplementing the server side as well as the front end would not be feasible within the given time constraint. Instead, we will focus on creating a new front end that is much more easy to operate for the average user.