**System Test Plan Template**

## 1. Test Plan Objective

This plan is to help the guide the direction of the final product. The test plan will provide documentation on how the testing will be made and at what time in the project

## 2. Product History

In previous versions, there was only a server or database that only had direct communication with emails. This email was just to show there was direct communication for cryptocurrency acceptance. Then the website was added in order had items for sale. However, the pictures only appear in MAC’s operating systems and has problems in Windows’ operating systems.

## 3. Features to be tested

List all features to be tested. Organize the list in the way that makes most sense- user features, or by level:

The first feature would be how easy the website would be accessible to every day customers.

The second would be the visual appearance of the website.

The third feature would be how easily the customer would be able to buy their preferred jeans

The fourth feature would be to see if the buyer has enough funds to complete the transaction.

## 4. Features not to be tested

What won’t be tested is how fast the emails are received. If multiple items are able to be added to shopping cart.

## 5. Environmental requirements

The only hardware is form the computer itself. We are using PyCharm as our IDE and Amazon AWS for the database

## 6. Control Procedure

Bugs are going to be discussed during the weekly meetings. QA will hand over a document to the developers of the bugs found and how to fix them. Bug will include any visual aspects of the website such images loading right and if information of the client is accessible.

The QA is going to test to according the client’s wishes in a black box environment.

## 7. Risks

Risks are going to include scheduling. Due to the nature of a short time given for the project, failing to meet sprints will be crucial. PM will have to be on top of things making sure developers do not fall behind.

The Risk matrix is given.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | Negligible | Marginal | Critical | Catastrophic |
| Certain |  | R7 |  |  |
| Likely |  | R6 | R3 | R2 |
| Possible |  |  | R5,R9 | R1, R8 |
| Unlikely |  |  |  |  |
| Rare |  |  |  | R4 |

R1: Database can’t store customer information

R2: Problem with communication between webpage and database

R3: Can’t process payments

R4: Communication error with developing team and business analyst

R5: Communication error with business analyst and client

R6: Developers/QA fail to document bug issues

R7: Arguments among team members about implementation

R8: Missed deadline

R9: PM is not checking up on team members to see progress

## 8. Test Strategy

There is going to be three sprints. Each sprint has their own criteria to meet.

### 8.1 Sprint 1

There will be no testing in this sprint. This sprint is all about research, ideas of implementation and setting meeting dates.

### 8.2 Sprint 2

This sprint will have two prototypes to be tested. The first is just a simple website that will have a few items. The purpose of this prototype is understand python code interfacing with a server.

The second prototype will be with a more elegant website. This prototype will be related to the finish product. In this prototype heavy testing will be made to see how will the program reacts to the clients needs.

### 8.3 Sprint 3

Sprint 3 has the finished product. However, testing will still occur in order to find any bugs in the program.