**System and Unit Test Report**

**Product name** : AMAZINO

**Team name** : Brokers

**Date**: 07/23/2019

**System Test scenarios**

* **Sprint 1**

User story 1: As a user, I need to login/logout to track my purchases and listings.

User story 2: As a developer, I need a database for viewing the whole log of user actions and managing the website traffic.

User story 3: As a seller, I need to register my stuff with some information such as price/simple description.

Scenario:

1. Go to AMAZINO website (<https://amazino-3b363.firebaseapp.com/>); select sign up; type

* Display name = <Joe Smith>
* Email = <Joe.Smith@gmail.com>
* Password = <JoeSmithIsCool!123>
* Confirm Password = <JoeSmithIsCool!123>
* Press Signup Button
* The page should be redirected to the market page, and can see ‘Success!’ message

1. On the navbar, press the Create Listing Icon; type

* Title = <New Unused Spatula>
* Description = <Great for flipping pancakes>
* Image <Spatula.jpeg>



* Price = <20>
* Category = <Cooking>
* Bet Period = <30 days>
* Press Create Listing Button
* The user should be returned to the Market Listing page, and can see the new item in the listing.
* The developer should be able to see the new item in the firebase console database.
* **Sprint 2**

User Story 1: As a buyer, I need to be able to bet for the product with my own point.

User Story 2: As a buyer, I need to charge points / be able to pay money for stuff.

User Story 3: As a dev, I need to have a fair drawing algorithm for deciding the winner.

User story 4: As a seller, I need to set the due date to get back my registered item.

User story 5: As a buyer, I need a set due date to get my points back if the item is not raffled.

User story 6: As a dev, I need a logical database schema for managing the website functions.

Scenario:

1. User should be able to charge points.
   * Click the profile icon on the navbar
   * The menu appears on the left side, and click the ‘profile’ menu
   * The user can see his own balance in the page (ex. 20)
   * Enter numeric input (ex. 50) into the input box in the page
   * Click ADD BALANCE button next to it
   * The user can see his own balance is updated (ex. 20 -> 70)
2. User bets on a listing on the market page.

* Click the item that the user wants to bet in the Market page
* Drag the progress bar as much as the user wants (up to 50% of the whole price) to bet on the item
* Press “Bet” button on the bottom
* If the user has enough balance for the betting payment,
  + If not, the user cannot bet and can see an error message ‘not enough balance’ on the site.
* <User Balance = User Balance - Amount Bet> (the user can see his updated balance in the profile page)
* And the user can see the changed progress bar, and ‘refund’ button next to the ‘bet’ button after the betting.
  + The developer should be able to see the updated ‘bets’ in the firebase console database with the correct user-item and the correct price.
* Click the ‘refund button’
  + Then the betting is canceled and the user’s balance becomes the previous balance

1. When betting is complete, the winner is chosen by a fair algorithm.

* When bets reach 100% of price, seller can see the ‘raffle’ button on item detail page
* Click the Raffle button, and the raffling starts in the server
* Item is removed from the market page
* The seller is notified and gets the price
* <Seller Balance = Seller Balance + Price of Item>
* The winner is notified and gets the item (winner and losers can see the notifies in ‘Notifications’ page in the menu

1. If betting didn’t reach 100% by the due date, all the betting is refunded and the item is deleted automatically.

* User Balance = User Balance + Amount Bet
* Listing is removed from the database
* **Sprint 3**

User Story 1: As a buyer, I need to search for the products I want.

User Story 2: As a user, I need a well designed and intuitive user interface.

User Story 3: As a seller, I need to start raffling phase when the item’s state is ‘readyToRaffle’

Scenario:

1. In the Market Listing page:

* Click the Search icon next in the navbar
* Type any keyword to search from the market
* The market listing is updated, and the user can see the listings that contains the keyword in their name

1. In the Market Listing page:

* Check the category box icon above the listings
* The market listing is updated, and the user can see the listings that are in the checked categories.

1. In the Market Listing page:

* Check the status of the item via the displayed tags
* Bet on any item tagged “Waiting for bets” if account has large enough balance, and your previous bets on that item do not exceed 50% of the price.

1. From the listing page of any listing the current user has posted:

* Once the betting reaches 100%, the seller can click the “raffle” button to choose a winner.
* A success message is shown, including the name of the winner.
* The item listing page is updated with the contact information of the winner, only visible to the sellers.
* The winner and seller receive notifications.

1. From any listing page as a user

* After the seller raffle item, if theuser wins they will be notified.
* Winning user can see the seller’s contact info from the listing page.

**Unit tests**

The reports all about our testings are saved in our github directory,

<https://github.com/diamond264/AMAZINO/tree/master/Testing>