

# Design Inspection

Author: Whole development team

Defect #	Description	Severity	How it was corrected
1	User schema did not have attribute that were needed for machine learning to work	2	2 arrays were added to the user schema for machine learning purposes
2	Recipe Viewer always returns a no recipes found error message everytime the page was opened regardless of if there were recipes or not in the database but still displays all recipes found	3	Needs to be fixed by making sure that the results from the backend are properly checked to see if there is data returned or not
3	Navbar would overlap content from other modules and other pages	2	Fixed by properly flushing the bottom of the navbar to ensure that it would no longer overlap other modules
4	There was no easy way to identify a recipe making it hard to refer to a particular recipe	1	Added an "id" property to every recipe
5	When signing in to the site, it would route the user to the profile page, whether the api call returned good or bad	1	Instead of having the button be a link, the redirect was handled programmatically if the api call returned.

# Code Inspection

Defect #	Description	Severity	How it was corrected
1	When searching recipes backend would load entire recipe database in search and then search that	2	Assigned ID to each recipe and searched by ID making the process quicker.
2	Not using secret key in cookies used by web application making it a security vulnerability.	2	We added and used a secret key for every cookie.
3	When adding ingredients in the create a recipe page, the remove button for each ingredient was causing a stair step effect instead of being aligned on the same column on the right due to the way the css interacted between levels	3	Structure of the css div tags was changed in order to make sure that the weird interaction stopped occurring.
4	All buttons were 50% of their regular size because a css property defined was overriding the width of all other buttons	2	CSS property that was overriding button size was removed.
5	User's name from Google authentication was not being received properly.	2	Looking at the returned user object for the correct entries.
6	There was no way to	1	Updated the user's

	update authentication tokens associated with a user.		authentication token every time they sign in and store it in the db.
7	There was no way to properly see if a user was logged in or not	2	A cookie system was added to the default route to see if the user was logged in.

## Unit Testing Report

This system relies on the following modules:

- Database module
- Backend module
- Authentication module

### Database Module

The Database module is the module responsible for storing user and recipe data such as username, ingredients, instructions etc. The database module was tested manually.

Product	Bazaar Web Application		
Module	Database Module		
Date			
Author	Development Team		
Defect #	Description	Severity	How it was corrected
1	Through manual testing we found that there was no way to store authentication token	1	Added an entry to user schema to track authentication token

### Backend Module

The Backend module is the module responsible for directing the user throughout the web application as well as modifying data and communication between the frontend and the database. The backend module was tested by the supertest and mocha frameworks.

Product	Bazaar Web Application		
Module	Backend Module		
Date			
Author	Development Team		
Defect #	Description	Severity	How it was corrected
1	Through manual testing we discovered no way to log out because there was no logout functionality.	2	Added logout function within the backend module.

### Authentication Module

The Authentication module is the module responsible for allowing users to create accounts and login to Bazaar. The authentication module was tested partly through the mocha framework and partly through manual testing.

Product	Bazaar Web Application		
Module	Authentication Module		
Date			
Author	Development Team		
Defect #	Description	Severity	How it was corrected
1	Our current way of authenticating users had no way of communicating successful authentication to the front end due to lack of built in functionality in react	1	Fixed by having the front end handle authentication and send the back end a validation token

### Account Module

The account module is the module responsible for the account page of the application where users can view their settings and change their username. This module was tested via manual testing and enzyme.

Product	Bazaar Web Application		
Module	Account Module		
Author	Development Team		
Defect #	Description	Severity	How it was corrected
1	No changes in code since sprint 1	N/A	N/A

### Navigation Module

The navigation module contains the navigation bar, which contains the search bar, and the sign in/sign out buttons. This module was tested by manual testing and enzyme

Product	Bazaar Web Application		
Module	Navigation Module		
Author	Development team		
Defect #	Description	Severity	How it was corrected
1	After signing in, buttons on the nav bar are aligned vertical, which can overlap UI elements.	2	Align buttons horizontally (not yet implemented).

### Sign In/Sign Up Module

The Sign In/ Sign Up Module contains the pages for signing into the app, as well as a page to create an account for Bazaar. These modules were created by manual testing and enzyme

Product	Bazaar Web Application		
Module	Sign In/Sign Up Module		

Author	Development team		
Defect #	Description	Severity	How it was corrected
1	Discovered an error where you could login without entering anything into either field	1	Added null checks.
2	Through testing it was discovered that the google API was not configured correctly.	1	Updated code in the front-end side to handle the google API (not yet implemented).

### Recipe Viewer Module

The Recipe Viewer module aggregates other UI components in order to display data. These modules was tested by manual testing.

Product	Bazaar Web Application		
Module	Recipe Viewer Module		
Author	Development team		
Defect #	Description	Severity	How it was corrected
1	Each individual recipe was not being displayed properly	2	The formatting for each recipe that was being pulled in was fixed to display correctly

### Shopping List Module

The Shopping List module display ingredients that a user adds to the shopping list feature. The module was tested through manual testing.

Product	Bazaar Web Application		
Module	Recipe Viewer Module		
Author	Development team		
Defect #	Description	Severity	How it was corrected

1	When adding multiple ingredients to the form, the remove step button was displaying more to the right of the last one instead of aligning properly		Changed so that the remove button for each ingredient was aligned properly.
2	New empty ingredients could be added to the shopping list		Added a check so that null strings could not be added to the list

### Create Recipe Module

The Create Recipe modules allows users to create new recipes and add them to the database. This module was by a mix of manual testing and enzyme.

Product	Bazaar Web Application		
Module	Recipe Viewer Module		
Author	Development team		
Defect #	Description	Severity	How it was corrected
1	When adding multiple steps to the form, the remove step button was displaying 50px to the right of the last one instead of aligning properly	3	Changed so that the remove button for each step was aligned properly.
2	New steps could be made that were empty and didn't contain any text	3	Before a new step could be added, there was a check to see if there was any text in the previous step
3	New ingredients could be made that were empty and didn't contain any text	3	Before a new ingredient could be added, there was a check to see if there was any text in the previous step