Team 16 Sprint 1 Testing for ¼ Cup App

Design Inspection

Product	¼ Cup Design Inspection
Date	2/11/18
Author	Team 16

Defect #	Description	Severity	How corrected
1	Validation messages that alert the user of an incorrect or incomplete user input, the message is being displayed as an Alert UI Object. We do not want this message to be displayed in such a cumbersome way to the user.	3	We have decided to display this message in a Toast UI Object to minimize the time on screen and user interaction to dismiss the error message.
2	When the user turns the phone, it will reload the activity and upon reloading the activity it loses all of the data that was being displayed before the turning of the device.	1	We moved all of the variables that were being used and accessed from the MainActivity to the view model controller
3	Upon trying to run the code on an earlier OS it became apparent that there was an error with the way that the onClick handler for enter needs to be handled differently.	3	Limiting the OS to a the version we selected upon creating the project. Also there will be a emulator that will be kept to this OS version for future testing. To fix this issue, we added an or statement that checked for the null enter (older OS) or timer enter (newer OS).
4	When we click on the add item button the keyboard opens up. If you select another view the keyboard remains open. If you turn the phone it consumes the entire screen.	2	Our possible fixes would be to find a way to disable the keyboard any time the view changes or a menu selection is selected. [Not yet implemented]

Code Inspection

Product	¼ Cup Code Inspection
Date	2/11/18
Author	Team 16

Defect #	Description	Severity	How corrected
1	When trying to create a new list and list item, the user can input no string for the name and the list will still be created, just with a blank name.	1	Adding a class that does data validation, including checking the string input from the user for a non empty string.
2	Data validation class threw an error when trying to call it from the main activity. This crashes the app anytime the code would be run.	1	Upon looking into the code, it was noticed that the global variable calling the validation class was not initialized.
3	Firebase depends on Google Play Services, so the app crashed on devices that did not have Google Play Services installed.	2	Added method to check if Play Services were available and prompt user to enable/install them if they are not.

Unit Testing

Product	¼ Cup ShoppingList Unit Testing
Module	ShoppingListRepository
Date	2/11/18
Author	Team 16

The Shopping List Repository is the interface in which the application is communicating with the FireBase FireStore for functionality related to the shopping list. This functionality includes, Creating/Deleting a ShoppingList object, and adding Items to a ShoppingList object.

Defect #	Description	Severity	How corrected
1	When trying to create a new list, the user	1	Adding a class that does

	can input no string for the name and the list will still be created, just with a blank name.		data validation, including checking the string input from the user for a non empty string.
2	When trying to create a new list item, the user can input no string for the name and the list item will still be created, just with a blank name.	1	Adding a class that does data validation, including checking the string input from the user for a non empty string.
3	When adding an item to a list through the ShoppingListRepository, it retrieves a new instance of FirebaseFirestore, rather than using the listsCollection member variable.	2	The method was modified to use the listsCollection member variable.
4	When creating a new ShoppingList via the ShoppingListRepository, it retrieves a new instance of FirebaseFirestore, rather than using the listsCollection member variable.	2	The method was modified to use the listsCollection member variable.

Product	¼ Cup ShoppingList Unit Testing
Module	ShoppingList ListView Model
Date	2/11/18
Author	Team 16

The ListViewModel holds references to LiveData supplied the repository classes and maintains those references while Activities undergo configuration changes so that the data outlives the Activity.

Defect #	Description	Severity	How corrected
1	When trying to rotate the screen with all Data being stored on the Main UI activity, The data would not be retained and redisplayed because the UI activity would be destroyed and re-created every time the screen was turned.	3	Our possible fixes would be to find a way to disable the keyboard any time the view changes or a menu selection is selected. [Not yet implemented]
2	When the app is rotated and the	1	We moved the variables kept

	screen is left the state of the application was not changing. We found via unit tests that certain variables were not persisting through state changes.		in the view activity over to the view model and were able to persist these values.

Product	¼ Cup ShoppingList Unit Testing
Module	Validation
Date	2/11/18
Author	Team 16

The data validation module holds the class that is used to do data validation. This is a module that was used to make sure that user input was correct.

Defect #	Description	Severity	How corrected
1	When calling the data validation, there was an invalid string set to the validator. Null value.	1	Needed to check if it is null before checking the length.
2	Invalid way of comparing strings to see if they are empty lead to an if never hitting.	1	Needed to change the way that empty was checked.

Product	¼ Cup ShoppingList Unit Testing
Module	WelcomeMessage
Date	2/11/18
Author	Team 16

The welcome message module is the opening activity that pops up when the application is started. It allows the user to login and create their account either through google or by directly adding a email password validation. From there it opens up the main activities.

Defect #	Description	Severity	How corrected
1	When we logged in and then closed out of the application the login state was not persisting. Users would have to re login every time.	1	We saved the state of the user via firebase and stored the data locally to the phone so the values persisted.
2	Originally the user state we did not think needed to be persisted at this point. But on inspection of our user stories we realized that user states needed to be kept for sprint one implementation.	1	We added firebase authentication and began persisting user id throughout the application.