

Beverage Booker	
Iteration Plan 4	Date: 24/08/20

Beverage Booker Iteration Plan 4

1. Key milestones

Milestone	Date
Iteration start	24/08/20
Added Size Selection for Drink Menu Options	27/08/20
Completed Automated Testing for Menu Options (Fill Cart)	27/08/20
Implemented Push Notifications	31/08/20
Implemented Inventory System	31/08/20
Tested Inventory System	01/09/20
Implemented Add & Edit Menu Item (Manager)	06/09/20
Iteration stop	06/09/20

2. High-level objectives

- Complete push notification mechanism and implement in the Staff/Customer app
Outcome: A push notification mechanism is implemented so that when an order is filled by the staff the customer will get a notification through the customer app saying their order is ready
- Complete automated integration testing for the queueing system
Outcome: Integration testing for the queueing system has been completed and the queueing branch merged to master.
- Implement checkboxes for the Fill Order use case that work with itemStatus variable in the database
Outcome: Checkboxes state on order items is saved using the itemStatus column of the 'cartitem' table, allowing staff to exit an order without losing their order progress
- Add size selection for drinks to the menu options in the AddToCartActivity and then conduct automated testing on the implementation
Outcome: The customer will be able to select sizes for their drinks. The menu options implementation will be fully tested and ready to merge to master.
- Plan and then implement inventory system within the server/database for the cafe
Outcome: Inventory table and system within the database/server implemented and working with purchases
- Conduct research into implementing the View Order use-case. Implement and then test the View Order use-case.
Outcome: View Order functionality is implemented and working within the Customer application. Use-case has passed Instrumented Unit and Integration testing.
- Implement regex/input validation to remove bad input for the Book Delivery address input page
Outcome: Regex/input validation is working and removes bad input for street number and street name fields
- Compile photos for each menu item
Outcome: A set of menu item photos that can be added to the database

Beverage Booker	
Iteration Plan 4	Date: 24/08/20

9. Implement and the Add & Edit Menu Item (Manager) use-cases
Outcome: The Manager can add and edit a menu item in the Staff App
10. Implement and test the View & Update Inventory (Manager) use-cases
Outcome: The Manager can view and update inventory in the Staff App and the use-cases have passed testing
11. Implement and test the delete single cart item (Empty Cart) use-case
Outcome: The user can delete single items from their cart in the customer app. The implementation has passed automated testing.
12. Implement and test the View Account (Edit Info) use-case
Outcome: The user can view their account details and update their registered email, password and mobile number. The implementation has passed automated testing.
13. Implement the Manage Staff functions (Add, Delete) along with staff level restrictions within the Staff App
Outcome: The manager can add new staff members and delete staff members once they are no longer employed. Staff members can only access certain features based on their staff level/assignment.
14. Write UATs for end of iteration testing
Outcome: UATs are ready for the end of iteration round of testing
15. Conduct end of iteration UAT testing on implemented use-cases
Outcome: UAT testing has been carried out on the implemented use-cases for this iteration.
16. Push the latest version of the local server up to the web server and update the database structure.
Outcome: Web server has the latest version of our local PHP server and is functioning without issues ready for the end of iteration UAT. Database tables are updated with any structural changes.

Beverage Booker	
Iteration Plan 4	Date: 24/08/20

3. Work Item assignments

Key:

Complete - Completed successfully this iteration.

Incomplete - Remains incomplete but not due to lack of time devoted by assigned team members and/or is postponed/rescheduled.

Incomplete - Remains incomplete and will require explanation and/or escalation.

The following Work Items will be addressed in this iteration:

Work Item ID	Name or key words of description	Outcome	State	Assigned to (name)	Estimated Hours	Hours worked	Estimate of hours remaining
1.0	Complete push notification mechanism and implement in the Staff/Customer app	Emily implemented the push notifications working off the 'completedorders' table. Benn then adjusted this to work off the orderStatus variable in the 'orders' table, due to a bug on the backend trigger.	Complete	Emily/Benn	16	20.25	0
1.1	Implement push notification that is triggered by the 'Complete Order' button	Notification methods are now triggered via the confirm order button	Complete	Emily	2	1.5	0
1.2	Timer method	Timer method now queries the database to check if order status is finished	Complete	Emily	2	3	0
1.3	API methods: check for order	Check order being in completedorders table. Checks for !null	Complete	Emily	2	3.5	0
1.4	API method: change status to indicate notification sent	Changes the ordersStatus to indicate the notifications have been completed	Complete	Emily	2	4	0
1.5	Continue research into push notifications and review Emily's implementation	Emily walked me through the push notifications implementation via the chat. Working together we found that it was one line of code in the setStatusNotify method that needed adjusting. Committed the change and tested the notifications via two emulators, one for the customer and the other for staff. Notification showed up on the Customer App upon the staff member clicking 'Complete Order' in the Staff App.	Complete	Benn	2	2	0

Beverage Booker	
Iteration Plan 4	Date: 24/08/20

1.6.1 1.6.2	Adjustment to notification trigger	Found the notifications did have a bug and weren't triggering at times, or were continually being triggered as the orderStatus value was not being updated. Found the best solution was to change the trigger to orderStatus = 0 in the 'orders' table. Instead of moving the order to another table this value will just change from a 1 for active to a 0 for complete order, which will trigger the notification.	Complete	Benn	4	4	0
1.7.1 1.7.2 1.7.3 1.7.4	Use cartID to get corresponding order placed for notification	getCartID method in PlaceOrderActivity returns the active cart. cartID is then passed to OrderConfirmationActivity once an order is placed. cartID is then used along with userID by the NotificationOutput class to obtain the exact order it should be listening to for a change to the orderStatus variable.	Complete	Benn	2	2.25	0
2.0	Complete automated integration testing for the queueing system	Automated Integration testing for the queueing system completed. Passed testing.	Complete	Jacob	2	2.5	0
2.1	Complete automated integration testing for the queueing system	Testing conducted for functionality of the queueing system also fixed up previous tests where entering the staff ID was 0001 now 1001.	Complete	Jacob	2	2.5	0
3.0	Implement checkboxes for the Fill Order use case that work with itemStatus variable in the database	Fill Order use-case fully implemented. Backend trigger was adjusted to meet up with notifications due to a bug. Testing will carry over and be completed next iteration.	Incomplete	Jake/ Benn/ Jacob	16	18.5	4
3.1.1 3.1.2 3.1.3	Implement checked itemStatus for order items	Helped Jake via voice chat to implement the checkbox for itemStatus	Complete	Benn	2	2	0

Beverage Booker	
Iteration Plan 4	Date: 24/08/20

3.2.1	Implement locking mechanism for the Complete Order button in Fill Order use case.	Even with Jacob trying to help out we couldn't figure out an idea that worked and each solution had different issues and became a near impossible fix.	Retired	Jake	4	13	0
3.2.2	Implement locking mechanism for the Complete Order button in Fill Order use case.	Attempted to help Jake with the checkboxes and locking/unlocking the button but didn't really come up with any solution to the problem.	Retired	Jacob	4	3	0
3.2.3 3.2.4	Switched implementation of locking mechanism to a confirmation screen.	Made a confirmation box that is meant to replace the lock by preventing the user from continuing through complete order unless they are sure that the order is complete.	Complete	Jake	2	0.5	0
3.3.1 3.3.2 3.3.3	Fix bug in backend trigger and meet up with push notification adjustments	Took some troubleshooting, but managed to get the 'complete order' button click/confirmation to trigger the update to orderStatus in the orders table. Still unsure why Jake's original trigger wasn't always working as it looked sound. Was only completing the update approximately 50% of the time.	Complete	Benn	2	3	0
3.4	Test	Will carry over to the next iteration.	Incomplete	Jake	4	0	4
4.0	Add size selection for drinks to the menu options in the AddToCartActivity and then conduct automated testing on the implementation	Size selection for drink menu items has been implemented in the Customer App. The full menu options implementation has passed unit and integration testing.	Complete	Benn	5	4	0
4.1.1 4.1.2 4.1.3	Implement size selection within the AddToCartActivity	Size selection for drinks is now implemented within the AddToCartActivity. Small is set as default. Medium adds \$0.50 to item price and Large adds \$1.00	Complete	Benn	2	1.75	0
4.2	Unit test the menu options implementation	Instrumented Unit Test for the menu options (Fill Cart) completed and passed.	Complete	Benn	1	0.75	0

Beverage Booker	
Iteration Plan 4	Date: 24/08/20

4.3	Integration test the menu options implementation	Instrumented Integration Test between the menu options (Fill Cart) and Place Order/Checkout use-case completed and passed.	Complete	Benn	2	1.5	0
5.0	Plan and then implement inventory system within the server/database for the cafe	Inventory System has been implemented within the server and the Customer App. Implementations were tested (Fill Cart & Empty Cart use-cases) and passed.	Complete	Benn	17	17.25	0
5.1.1 5.1.2	Implement a method for displaying menu items as 'Sold Out' within the Customer App	Added 'itemStock' column to the 'items' table in the database. the getItems method that retrieves the menu items now passes down the stock value for each menu item. If a menu item type = 'food' & itemStock <=5 then the recycler adapter disables the 'add to cart' button and displays red 'SOLD OUT' text for that item in the menu. Marking items as sold out when 5 or less are in stock is to stop over ordering.	Complete	Benn	1	0.75	0
5.2.1 5.2.2	Update Fill Cart use-case to include inventory system	Updated addToCart method on server to include decrementing stock level by quantity for that item. Also checks that enough of the item is in stock before adding the item to cart. Updated the Customer App to show the number of items in stock for food items in Browse Menu.	Complete	Benn	2	2	0

Beverage Booker	
Iteration Plan 4	Date: 24/08/20

5.3.1 5.3.2	Update Customer App to show current stock levels on the AddToCartActivity	Implemented getMenuItem method on the server that allows for retrieval of values for a single menu item (including stock value). Implemented stock level to display on screen in the AddToCartActivity below quantity button. Quantity upper limit is set so it can't go higher than the stock level at the time the activity opens. If the item becomes sold out, "SOLD OUT" displays where stock level would be and the 'Add to Cart' button locks.	Complete	Benn	2	2	0
5.4.1 5.4.2 5.4.3	Add quantity back to itemStock level in 'items' table when the user empties their cart	Had some difficulty with the PHP for this, but in the end got it working. Had to do some research on arrays and looping with PHP. Updated the emptyCart method and added getCartItemIDsAndQuantities method on the server. The latter returns itemIDs and associated quantities. The empty cart method then iterates over this array, adding these quantities back to the stock levels for each menu item.	Complete	Benn	3	4.5	0
5.5.1 5.5.2 5.5.3	Update 'deleteCartItem' method to add quantity of item back into inventory	Updated the deleteCartItem method on the server so now the quantity for that item is added back into the inventory. Updated the getCartItems method on the server so it now retrieves itemID and itemSize as part of the cart array. Updated the deleteCartItem method in the Customer App so it now passes itemID, itemQuantity, and itemSize of the clicked item back up to the server for deletion from cart.	Complete	Benn	2	2.5	0

Beverage Booker	
Iteration Plan 4	Date: 24/08/20

5.6	Test inventory system (Fill Cart) in the Customer App	Updated the Fill Cart automated test to now check that a sold out item cannot be added to cart. Also checks that the “SOLD OUT” text is displayed for that menu item.	Complete	Benn	2	1.5	0
5.7	Test inventory system (Empty Cart) in the Customer App	Updated the Empty Cart automated test to work with our new menu layout and inventory system. Test successfully adds an item to cart, empties the cart, and then verifies the item is no longer in the customer cart.	Complete	Benn	1	0.75	0
5.8.1 5.8.2	Test Fill Cart & Empty Cart for the local server	Updated the Fill Cart & Empty Cart Postman test collections and then ran them against the local server. All tests passed.	Complete	Benn	1	1.5	0
5.9.1	Fix bug in addToCart method on the server	I found a bug in my code for the addToCart method where it wasn't properly checking itemType before proceeding with adding the item to cart. Fixed the issue and also tidied up the addToCart method by pulling repeated code out into their own methods.	Complete	Benn	2	1.5	0
5.9.2	Test the addToCart method (Fill Cart use-case) on the local server	Ran the Postman automated tests for the Fill Cart use-case against the local server. The updated implementation passed all 21 tests.	Complete	Benn	1	0.25	0
6.0	Conduct research into implementing the View Order use-case. Implement and then test the View Order use-case.	Got as far as the timer/progress bar working off a button click and saving/running in the background using Shared Preferences. The timer is also being set by an estimated order time, but I am midway through implementing the calculations on the server.	Incomplete	Benn	21	17.5	3

Beverage Booker	
Iteration Plan 4	Date: 24/08/20

6.1	Research how to implement a timer to display estimated time until the order is ready	I watched a Youtube tutorial series on how to implement a countdown timer within Android. I also read over the official documentation. I am hoping to set the timer based on an estimated order time that will be a total of average prep times for each menu item in an order.	Complete	Benn	1	1.75	0
6.2	Implement basic Countdown Timer in the Customer App	Have implemented a basic countdown timer on the order confirmation screen. At the moment the timer is hardcoded to start at 10 mins when the activity begins. The countdown displays minutes and seconds and updates in 1 second intervals on screen.	Complete	Benn	2	1.25	0
6.3	Adjust timer so it works using Shared Preferences	The timer now uses Shared Preferences, which allows for the user to exit or completely close the app. The timer runs by comparing its end time to that of the System clock, so it can essentially run when the app is not open.	Complete	Benn	2	1.25	0
6.4	Reset timer and update message on screen if push notification is triggered	If a push notification is triggered before the estimated order time reaches zero, then the Notification class calls a method that resets the timer. The text is also programmed to change to an "order ready" message on screen if the timer reaches zero/the notification is triggered.	Complete	Benn	1	1	0
6.5	Implement Progress Bar that works in unison with the Countdown Timer	Had difficulty with this but I now have a circular progress bar that fills as the estimated order time counts down. Progress is set using the timer value. The progress is also saved using Shared Preferences like the countdown timer, and because it is set by this timer that uses the System time it "continues to run" so to speak if the app is closed or	Complete	Benn	4	3.75	0

Beverage Booker	
Iteration Plan 4	Date: 24/08/20

		the user moves to a different screen.					
6.6.1 6.6.2 6.6.3 6.6.4	Implement progress bar and timer to start automatically	Had a lot of difficulty with this. The timer/progress bar works perfectly if i use a button click to start it. Getting it to start at the beginning of the activity results in a bug. The timer starts, but does not save if the app is closed or exited. Instead it pauses and I could not find a workaround. A lot of what i was trying involved checking the database for the order/completion, but this seemed to create buggy behaviour with the timer. I think for now it may have to start with a button click from the user.	Complete	Benn	4	4.5	0
6.7.1 6.7.2	Use 'orderTime' from the orders table	Implemented getOrderTime method on the server which retrieves the estimated fill time for an order. Also updated the placeOrder method to use orderTime. In the customer app i made adjustments so the timer is now set from the order time pulled from the server. At the moment it is just running off a dummy value.	Complete	Benn	2	2.25	0
6.8	Implement methods for calculating cartTime and orderTime	Implemented getItemTime which retrieves the fill time for a single menu item. Updated insertDrinkInActiveCart so for drinks it is now getting the itemTime for that menu item and then adding it to the cumulative total - cartTime. The idea is when an order is placed, cartTime will then become orderTime as the estimated time to fill that order and this will set the timer. Also implemented the getCartTime method to return the current total.	Complete	Benn	2	1.75	0

Beverage Booker	
Iteration Plan 4	Date: 24/08/20

6.9	Adjust remaining addToCart methods to update cartTime as items are added to cart	Did not get to this job this iteration. Will carry over and be completed as a priority.	Incomplete	Benn	3	0	3
7.0	Implement regex/input validation to remove bad input for the Book Delivery address input page		Incomplete	Emily	2	1.5	1
7.1	Implement the regex/input validation for Book Delivery address input	Partially implemented	Incomplete	Emily	2	1.5	1
8.0	Compile photos for each menu item		Incomplete	Emily	5	0	5
8.1	Find test images for the menu	Not required this iteration	Postponed	Emily	1	0	1
8.2	Research how to add them to the database	The server will be used to hold the images. They will not be stored locally. Just a small resolution so as to not overload the server.	Incomplete	Emily	3	0	3
8.3	Check for local storage possibility	Carries over to next iteration.	Incomplete		1	0	1
9.0	Implement and test the Add & Edit Menu Item (Manager) use-cases	Use-cases done testing incomplete but will be done next iteration.	Incomplete	Jacob	12	13	2
9.1	Research and plan a method to implement Add and Edit menu Item use-cases	I plan on having a new screen with all the fields of an item, the add and modify buttons will lead to the same page however the modify button will utilise the intent to send the data of the item through to the page and will alter the fields depending on their values. Because these are booleans I will most likely use checkboxes just like Benn's implementation on the customer app.	Complete	Jacob	3	3.5	0
9.2.1 9.2.2	Implement Add Menu Item use-case	On the menu screen it has an add button once clicked it asks for a type then goes to the item form which has all fields associated with the	Complete	Jacob	4	6.5	0

Beverage Booker	
Iteration Plan 4	Date: 24/08/20

		item that can be filled out and added to the database.					
9.3	Test Add Menu Item use-case	Carry over into the next iteration.	Incomplete	Jacob	1	0	1
9.4.1 9.4.2	Implement Edit Menu Item use-case	Modify takes the information of the item being modified straight to the item form which then fills out the fields with said information.	Complete	Jacob	3	3	0
9.5	Test Edit Menu item use-case	Carry over into the next iteration.	Incomplete	Jacob	1	0	1
10.0	Implement and test the View & Update Inventory (Manager) use-cases	Did not get to this job this iteration. Will carry over and be completed next iteration	Incomplete	Benn	10	0	10
10.1	Implement the View Inventory use-case	Will be completed next iteration.	Incomplete	Benn	4	0	4
10.2	Implement Update Inventory use-case	Will be completed next iteration.	Incomplete	Benn	4	0	4
10.3	Test the View & Update Inventory implementation	Will be completed next iteration.	Incomplete	Benn	2	0	2
11.0	Implement and test the delete single cart item (Empty Cart) use-case	Use-case mostly done just need to be tested.	Incomplete	Jacob/ Benn	5	4.25	0
11.1.1 11.1.2	Implement basic functional delete single cart item	So far it all works, however it is a general one and doesn't account for different types of the same order e.g. heated muffin and muffin will both get deleted even though they are different items.	Complete	Jacob	3	2.5	0
11.2.1 11.2.2	Expand upon general implementation adding the rest of the item components to the database calls	Code is expanded just need to fix delete button so it isn't overlapping comment in the xml file	Complete	Jacob	2	1	0
11.3	Integration test for Delete single cart item	Carry over into the next iteration.	Incomplete	Jacob	1	0	1
11.4	Test Delete Single Cart Item against the local server	Tested the deleteCartItem method using Postman against the local server as part of the Empty Cart use-case. Passed testing.	Complete	Benn	1	0.75	0

Beverage Booker	
Iteration Plan 4	Date: 24/08/20

12.0	Implement and test the View Account (Edit Info) use-case	Did not get to this job this iteration. Will carry over and be completed next iteration	Incomplete	Unassigned	6	0	6
12.1	Implement View Account use-case	Will be completed next iteration.	Incomplete	Unassigned	4	0	4
12.2	Test the View Account implementation	Will be completed next iteration.	Incomplete	Unassigned	2	0	2
13.0	Implement the Manage Staff functions (Add, Delete) along with staff level restrictions within the Staff App	Did not get to this job this iteration. Will carry over and be completed next iteration	Incomplete	Unassigned	6	0	6
13.1	Implement Add Staff feature	Will be completed next iteration.	Incomplete	Unassigned	3	0	3
13.2	Implement Delete Staff feature	Will be completed next iteration.	Incomplete	Unassigned	3	0	3
14.0	Write UATs for end of iteration testing	UATs written and checked before committing to repository.	Complete	Benn	4	4.25	0
14.1	Write UATs for end of iteration testing	UATs written for the inventory system (Fill Cart/Empty Cart), the drink size options (Fill Cart/Place Order) and order completion/push notifications (Fill Order).	Complete	Benn	3	3.25	0
14.2	Check UAT document/results and commit to repository	Checked over the UATs for any errors before committing to the repository.	Complete	Benn	1	1	0
15.0	Conduct end of iteration UAT testing on implemented use-cases	All team members have completed the end of iteration UATs.		Multiple	4	4.75	0
15.1	Conduct end of iteration UAT testing on implemented use-cases	UATs completed for the inventory system (Fill Cart/Empty Cart), the drink size options (Fill Cart/Place Order) and order completion/push notifications (Fill Order).	Complete	Benn	1	1.25	0
15.2	Conduct end of iteration UAT testing on implemented use-cases	UATs completed for the completed tasks of this iteration	Complete	Emily	1	1	0

Beverage Booker	
Iteration Plan 4	Date: 24/08/20

15.3	Conduct end of iteration UAT testing on implemented use-cases	UAT testing was carried out successfully.	Complete	Jacob	1	1	0
15.4	Conduct end of iteration UAT testing on implemented use-cases	UAT testing completed for the iterations use cases.	Complete	Jake	1	1.5	0
16.0	Push the latest version of the local server up to the web server and update the database structure	Latest version of our PHP server has been pushed to the web server and is fully operational. Database is updated and the app is working for UATs.	Complete	Benn	1	0.75	0
16.1	Push the latest version of the local server up to our hosted web server (beveragebooker.com)	Merged master into the currently open development branch for latest changes. Then zipped the BeverageApi repo and uploaded it to the server.	Complete	Benn	0.5	0.5	0
16.2	Updated the database structure	Added latest changes to 'orders', 'cart' and 'items' tables in the database. Also quickly ran through some of the app functions to check it was working.	Complete	Benn	0.5	0.25	0

Beverage Booker	
Iteration Plan 4	Date: 24/08/20

4. Issues

Issue	Status	Notes
No issues to report this iteration.		

5. Assessment

Assessment target	To complete all assigned jobs by the end of the iteration.
Assessment date	07/09/20
Participants	Benn Curby, Emily Carter, Jake Durnford and Jacob Kennedy
Project status	Orange/Green - Most jobs completed or partially completed due to time constraints.

6. Assessment against objectives:

Key:

Complete - Completed successfully this iteration.

Incomplete - Remains incomplete but not due to lack of time devoted by assigned team members and/or is postponed/rescheduled.

Incomplete - Remains incomplete and will require explanation and/or escalation.

Objective	Status
Complete push notification mechanism and implement in the Staff/Customer app	Complete - Emily implemented the push notifications working off the 'completedorders' table. Benn then adjusted this to work off the orderStatus variable in the 'orders' table, due to a bug on the backend trigger.
Complete automated integration testing for the queueing system	Complete - Jacob completed automated integration testing on the queueing system for orders in the staff app. The implementation passed testing.
Implement checkboxes for the Fill Order use case that work with itemStatus variable in the database	Complete - Jake successfully implemented the Fill Order use-case. Benn adjusted the Backend trigger due to a bug and to also meet up with adjustments made to push notifications. Incomplete - Testing will carry over and be completed next iteration as a priority.
Add size selection for drinks to the menu options in the AddToCartActivity and then conduct automated testing on the implementation	Complete - Benn has implemented size selection for drink menu items in the Customer App. The full menu options implementation has passed unit and integration testing.

Beverage Booker	
Iteration Plan 4	Date: 24/08/20

Plan and then implement inventory system within the server/database for the cafe	Complete - Benn has implemented the inventory system within the server and Customer App. addToCart, emptyCart and deleteCartItem methods have all been updated on the server so they either increment or decrement stock quantities in the inventory. In the Customer App stock quantities are pulled down with menu items, and if an item is out of stock it will show "SOLD OUT" on the item along with locking the 'Add to Cart' button. The Fill Cart and Empty Cart use-case tests were updated for the Customer App, along with the Postman tests for the server. Both sets of use-case tests passed.
Conduct research into implementing the View Order use-case. Implement and then test the View Order use-case.	Incomplete - Benn implemented the timer/progress bar working off a button click and saving/running in the background using Shared Preferences. The timer is also being set by an estimated order time, but he is midway through implementing the calculations on the server. This will be completed next iteration along with testing for the implementation.
Implement regex/input validation to remove bad input for the Book Delivery address input page	Incomplete - Emily has partially implemented the input validation for Book Delivery address input using Google place autocomplete. This is not fully implemented, and will carry over to the next iteration.
Compile photos for each menu item	Incomplete - Emily looked at how would be best to display images for menu items - either in the database and read down with the menu items, or locally within the app. Low resolution images pulled down from the database looks like the most straightforward approach. This job is low priority compared to the other implementations so will likely be completed after prioritised functionality.
Implement and test the Add & Edit Menu Item (Manager) use-cases	Complete - Jacob successfully implemented the Add & Edit Menu Item use-cases for the backend (Manager) this iteration. Incomplete - Testing for both Add & Edit Menu Item implementations will carry over into the next iteration and be completed as a priority.
Implement and test the View & Update Inventory (Manager) use-cases	Incomplete - Benn did not get to this task this iteration. It will be implemented next iteration after completing the View Order use-case and testing.
Implement and test the delete single cart item (Empty Cart) use-case	Complete - Jacob successfully implemented the delete single cart item (Empty Cart) use-case. Benn made adjustments to his implementation to add the quantity back to the stock inventory upon deletion.

Beverage Booker	
Iteration Plan 4	Date: 24/08/20

	Incomplete - Automated testing was not carried out but will carry over to the next iteration and be completed as a priority.
Implement and test the View Account (Edit Info) use-case	Incomplete - Nobody was assigned to this job as we have been working off a “ticketed” system where if all previous work was finished then the next available team member would take the job. Will carry over and be completed next iteration.
Implement the Manage Staff functions (Add, Delete) along with staff level restrictions within the Staff App	Incomplete - Nobody was assigned to this job as we have been working off a “ticketed” system where if all previous work was finished then the next available team member would take the job. Will carry over and be completed next iteration.
Write UATs for end of iteration testing	Complete - Benn wrote the end of iteration UATs for the inventory system (Fill Cart/Empty Cart), the drink size options (Fill Cart/Place Order) and order completion/push notifications (Fill Order).
Conduct end of iteration UAT testing on implemented use-cases	Complete - All team members successfully completed the end of iteration UATs for the inventory system (Fill Cart/Empty Cart), the drink size options (Fill Cart/Place Order) and order completion/push notifications (Fill Order).
Push the latest version of the local server up to the web server and update the database structure.	Complete - Benn successfully updated the web server and database with the latest changes. Was a noticed bug with one of the inventory functions but this was fixed and then the updated version pushed to the server for the UATs.

Beverage Booker	
Iteration Plan 4	Date: 24/08/20

7. Deviations / Ongoing Tasks:

Deviations from the iteration plan and tasks that carry over into the next iteration.

Key:

Ongoing - Remains incomplete and is carrying over into the next iteration.

Ongoing - Remains incomplete and is carrying over into the next iteration - but will be escalated and assessed/completed as a priority.

Postponed - Remains incomplete and was not started this iteration. Task will carry over to the next iteration.

Postponed - Remains incomplete and was not started this iteration. Task will carry over to the next iteration and be completed as a priority/will be escalated.

Reassigned - Job has been reassigned to a new team member.

Reassigned - Job has been reassigned to a new team member/members and has been prioritised/escalated.

Objective	Status
Implement checkboxes for the Fill Order use case that work with itemStatus variable in the database	Ongoing - Jake completed the Fill Order use-case implementation this iteration. The testing of the use-case will be completed next iteration as a priority.
Conduct research into implementing the View Order use-case. Implement and then test the View Order use-case.	Reassigned - During the mid-iteration team meeting Benn suggested he could take over the View Order use-case job while Jake was busy with another assignment. Jake agreed and the job was reassigned to Benn. Ongoing - Benn implemented the timer/progress bar working off a button click and saving/running in the background using Shared Preferences. The timer is also being set by an estimated order time, but he is midway through implementing the calculations on the server. This will be completed next iteration along with testing for the implementation.
Implement regex/input validation to remove bad input for the Book Delivery address input page	Ongoing - Emily has partially implemented the input validation for Book Delivery address input using Google place autocomplete. This is not fully implemented, and will carry over to the next iteration.
Compile photos for each menu item	Postponed/Ongoing - Emily looked at how would be best to display images for menu items - either in the database and read down with the menu items, or locally within the app. Low resolution images pulled down from the database looks like the most straightforward approach. This job is low priority compared to the other implementations so will likely be completed after prioritised functionality.
Implement and test the Add & Edit Menu Item (Manager) use-cases	Ongoing - Jacob completed the implementation of Add & Edit Menu Item, but the testing for both use-cases will carry over into the next iteration.

Beverage Booker	
Iteration Plan 4	Date: 24/08/20

Implement and test the View & Update Inventory (Manager) use-cases	Postponed - Benn did not get to this task this iteration. It will be implemented next iteration after completing the View Order use-case and testing.
Implement and test the delete single cart item (Empty Cart) use-case	Ongoing - Jacob completed the implementation for the delete single cart item (Empty Cart) use-case. Automated testing was not carried out but will carry over to the next iteration and be completed as a priority.
Implement and test the View Account (Edit Info) use-case	Postponed - Nobody was assigned to this job as we have been working off a “ticketed” system where if all previous work is finished then the next available team member will take the job. Not started this iteration, but will be completed next iteration.
Implement the Manage Staff functions (Add, Delete) along with staff level restrictions within the Staff App	Postponed - Nobody was assigned to this job as we have been working off a “ticketed” system where if all previous work is finished then the next available team member will take the job. Not started this iteration, but will be completed next iteration.

Beverage Booker	
Iteration Plan 4	Date: 24/08/20

8. Additions:

Tasks that were added mid iteration.

Key:

Complete - Completed successfully this iteration.

Ongoing - Remains incomplete and is carrying over into the next iteration.

Ongoing - Remains incomplete and is carrying over into the next iteration - but will be escalated and assessed/completed as a priority.

Objective	Status
Write UATs for end of iteration testing	Complete - Benn wrote the end of iteration UATs for the inventory system (Fill Cart/Empty Cart), the drink size options (Fill Cart/Place Order) and order completion/push notifications (Fill Order).
Conduct end of iteration UAT testing on implemented use-cases	Complete - All team members successfully completed the end of iteration UATs for the inventory system (Fill Cart/Empty Cart), the drink size options (Fill Cart/Place Order) and order completion/push notifications (Fill Order).
Push the latest version of the local server up to the web server and update the database structure.	Complete - Benn successfully updated the web server and database with the latest changes. Was a noticed bug with one of the inventory functions but this was fixed and then the updated version pushed to the server for the UATs.

Beverage Booker	
Iteration Plan 4	Date: 24/08/20

9. Risk Mitigation / Escalation

Tasks that require escalation or risk mitigation strategies.

Key:

Complete - Completed successfully this iteration.

Ongoing - Remains incomplete and will be addressed next iteration as a priority.

Risk	Status
Complete push notification mechanism and implement in the Staff/Customer app	Escalated and Completed - Benn and Emily worked together to finish the push notification implementation. Emily implemented the push notifications working off the 'completedorders' table. Benn then adjusted this to work off the 'orderStatus' variable in the 'orders' table, due to a bug on the backend trigger.