Sprint Review: GotoGro-MRM

Team Details

Team Name:	MSP 14
Tutorial:	Tue 2:30 ATC325
Tutor:	Dr Kaberi Naznin

Members:	
Dylan Jarvis	102093138
Rabya Tayal	103144215
Simon Tran	103602807
Thomas Babicka	103059885
Cody Cronin-Sporys	103610020
Nicholas Dyt	101624265

Stakeholder Feedback

Demonstration to stakeholder was conducted. All items were able to be demonstrated, Table 1 summarises the feedback on the design at the end of Sprint 1.

Table 1. Feedback

Item	Justification
General UI	- The UI is not aesthetically appealing but is functional. Consider changing colour schemes and adjusting button positioning.
Inventory UI	- The clear button is too close to the input fields and feels like the natural button to click after completing entry. Consider moving this somewhere safer to avoid user frustration. - Edit Qty button should be renamed to Edit Item (cosmetic fix). - Delete button needs to have a popup which covers the screen to confirm whether the user wants to delete the record. - Consider adding a “category” column to the item table, mainly for sorting and reporting purposes.
Member UI	- Remove credit card details from the member table as these are unnecessary. - Add an email column instead. - Add input checking to make sure emails are in valid format (something@somthing.som) - Delete button needs to have a popup which covers the screen to confirm whether the user wants to delete the record.
POS UI	- Confirm sale button not working as intended.

These items are minor fixes but have been added to the overall production document to make sure they are attended to in sprint 2.

Progress Justification

As demonstrated, all items were able to be demonstrated. According to the breakdown designed in an earlier task, the overall work for this sprint is shown in Figure 1:

Nu	Lev	Product	Item	Description	Prerequis	Team	Est	Actual
1	Critical	Inventory Table	Create Item Table	Item table with data validation - NOT NULL, etc - which records item details for company inventory	-	Dylan	0.3	0.1
2	Major	Inventory Table	Create Queries to Add Item	Simple query to add item, will be triggered by visual fields on the UI	1	Dylan	0.3	0.1
3	Minor	Inventory Table	Create Queries to Delete Item	Simple query to delete item, will be triggered by visual fields on the UI	1	Dylan	0.3	0.1
4	Minor	Inventory Table	Create Queries to Modify Item	Simple query to modify item, will be triggered by visual fields on the UI	1-2	Dylan	0.3	0.1
5	Critical	Member Table	Create Member Table	Member table with data validation - NOT NULL, etc - which records member details	-	Dylan	0.3	0.3
6	Critical	Member Table	Create Queries to Add Member	Simple query to add member, will be triggered by visual fields on the UI	5	Dylan	0.3	0.1
7	Minor	Member Table	Create Queries to Delete Member	Simple query to delete member, will be triggered by visual fields on the UI	5	Dylan	0.3	0.2
8	Minor	Member Table	Create Queries to Modify Member	Simple query to modify member, will be triggered by visual fields on the UI	5-6	Dylan	0.3	0.2
10	Critical	Sales Record Table	Create Sales Table	Sales table with references to both the items and members table	-	Dylan	0.5	0.6
11	Critical	Sales Record Table	Create Query to Add Sale	Simple Query to add sale, will be triggered by UI interface basically simulating a POS machine. When sale is added inventory of the item should decreas by the amount bought	1, 10	Dylan	0.3	0.2
12	Critical	Add Sales Record UI	Input Field to Add Member ID	Member ID must be added before any items such that each item can be associated with the correct member in the sales table	10-11	Cody	4	3.3
13	Major	Add Sales Record UI	Buttons to Add Different Items to Sale	Buttons simulate the effect of a barcode being scanned or similar, adding	10-12	Cody	4	3.6
14	Critical	Add Sales Record UI	Button to Confirm Sale	Pushes all sales records to the sales table, trigger the decrementing inventory for given items	11-12	Nic	2	1.1
15	Major	New Member UI	Input Fields for All Member Details	Text checking on input fields to minimise the chance of invlid data being entered	5-6	Rabya	4	3.4
16	Major	New Member UI	Confirm Button	Button to confirm the member details, checks the inputs then sends it to the member table	15	Nic	2	3
17	Critical	New Member UI	Autoincrementing Member ID	is automatically generated and added to the database	16	Rabya	2	1.2
18	Major	New Member UI	Viewport to View Members	Snapshot of the member table, needs to be able to be filtered by search interface	5	Rabya	2	3.2
19	Major	Add/Mod New Member UI	Search Input Field	By typing member ID in and confirming, the viewport will display the member searched for (or nothing if no results found)	18	Simon	2	2.1
20	Major	Add/Mod New Member UI	Modify a Member Record	Selecting the searched member result will populate the text fields with saved data. Writing over these with new information and confirming will save over the old record with the new information	8, 15-19	Simon	4	4.2
21	Major	Add/Mod New Item UI	Input Field to Add Item	Text checking on input fields to minimise the chance of invlid data being entered	1-2	Dylan	4	3
22	Major	Add/Mod New Item UI	Confirm Button	Pushes item record to the item table	22	Thomas	2	3
23	Major	Add/Mod New Item UI	Viewport to View Item	Snapshot of the item table, needs to be able to be filtered by search interface	1	Thomas	2	2.4
24	Major	Add/Mod New Item UI	Search Input Field	By typing item ID in and confirming, the viewport will display the item searched for (or nothing if no results found)	23	Thomas	2	1.8
25	Major	Add/Mod New Item UI	Modify an item Record	Selecting the searched member result will populate the text fields with saved data. Writing over these with new information and confirming will save over the old record with the new information	4, 21-24	Nic	4	3.1
						Total	43.2	40.4

Importantly, the total time was 43.2 hours. Due to the extensive planning time, these estimates were fairly accurate with the general trend being a slightly overbudgeted time. The overall hours reported from the team for Sprint 1 came in at 40.4 hours of work, within 3 hours of the predicted time.

That said, the time for each item was not perfectly estimated. In general, the SQL and database components were completed faster than the budgeted time. As where the input fields of the UI design for each page. Contrastingly, the time spent on actually coding the confirm buttons to work including error checking and passing information to the database was underestimated.

The reason for this was mainly due to the fact that the code was expected to be written into the input field objects themselves, but since all of the code was run on the confirm button,

this ended up being where the bulk of the time was spent. In future the team needs to be more aware of exactly which items involve coding and have the times adjusted accordingly as coding takes up the bulk of the time in any given task.

Organisational Feedback

On the whole the itemisation of each task was specific and atomic enough that the group had a clear idea of what they needed to do to tick that item off. Where an item included UI design, the team had already agreed on some rough outlines which were good enough to start work without wasting time considering design elements.

On the topic of design, at a code level the system seems very robust with easy to access and maintain tables in the database. From a design perspective there is obvious room for improvement, but this has been scheduled for the end of sprint 2. Ultimately, the functionality is top priority over the aesthetics.