**Sprint Planning:** GotoGro-MRM

**Team Details**

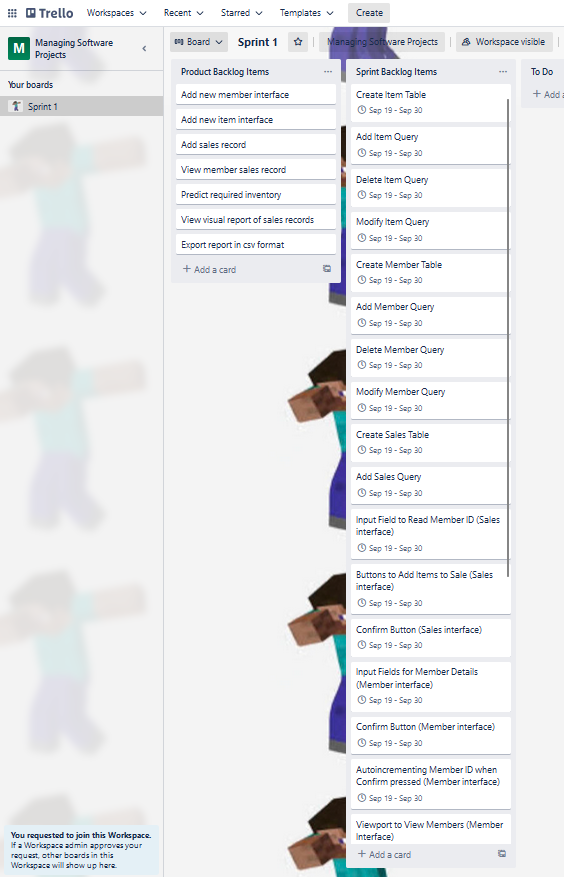
|  |  |
| --- | --- |
| **Team Name:** | MSP 14 |
| **Tutorial:** | Tue 2:30 ATC325 |
| **Tutor:** | Dr Kaberi Naznin |

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

**Sprint Management**

Some bullshit here

**Day 0 Trello Task Board**



The Trello board captures the list of tasks discussed and justified in task 08P. They were given bounded dates for completion which will be used by the burndown chart. Currently they all have the same completion date, however they are listed in order of importance. In future team meetings we may opt to put stricter dates especially on the earlier items.

Chart, line chart

Description automatically generated**Day 0 Burndown Chart**

Of the 27 items created for this sprint the ideal burndown chart was created by Trello. It appears that the sprint is designed with a small break in the middle, this gives the team a chance to get ahead if we continue at a steady pace.

**Day 0 Github Repo**

Taking the overarching items from the backlog and breaking them down into achievable tasks gave the team a very solid idea of exactly what must be done. The following tables summarise the breakdown of each item.

**Table 4. Inventory Table Item Breakdown**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Num** | **Level** | **Product** | **Item** | **Description** | **Prerequisites** | **Team Member** | **Est. Time** |
| 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 |
| 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 |
| 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 |
| 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 |

**Table 5. Member Table Item Breakdown**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Num** | **Level** | **Product** | **Item** | **Description** | **Prerequisites** | **Team Member** | **Est. Time** |
| 5 | Critical | Member Table | Create Member Table | Member table with data validation - NOT NULL, etc - which records member details | - | Dylan | 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 |
| 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 |
| 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 |

**Table 6. Sales Record Table Item Breakdown**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Num** | **Level** | **Product** | **Item** | **Description** | **Prerequisites** | **Team Member** | **Est. Time** |
| 10 | Critical | Sales Record Table | Create Sales Table | Sales table with references to both the items and members table | - | Dylan | 0.5 |
| 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 decrease by the amount bought | 1, 10 | Dylan | 0.3 |

**Table 7. Add Sales Record UI Item Breakdown**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Num** | **Level** | **Product** | **Item** | **Description** | **Prerequisites** | **Team Member** | **Est. Time** |
| 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 |
| 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 |
| 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 |

**Table 8. Add/Mod New Member UI Item Breakdown**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Num** | **Level** | **Product** | **Item** | **Description** | **Prerequisites** | **Team Member** | **Est. Time** |
| 15 | Major | Add/Mod New Member UI | Input Fields for All Member Details | Text checking on input fields to minimise the chance of invalid data being entered | 5-6 | Rabya | 4 |
| 16 | Major | Add/Mod New Member UI | Confirm Button | Button to confirm the member details, checks the inputs then sends it to the member table | 15 | Nic | 2 |
| 17 | Critical | Add/Mod New Member UI | Autoincrementing Member ID | When the confirm button is pressed the member ID is automatically generated and added to the database | 16 | Rabya | 2 |
| 18 | Major | Add/Mod 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 |
| 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 |
| 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 |

**Table 9. Add/Mod New Item UI Item Breakdown**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Num** | **Level** | **Product** | **Item** | **Description** | **Prerequisites** | **Team Member** | **Est. Time** |
| 21 | Major | Add/Mod New Item UI | Input Field to Add Item | Text checking on input fields to minimise the chance of invalid data being entered | 1-2 | Dylan | 4 |
| 22 | Major | Add/Mod New Item UI | Confirm Button | Pushes item record to the item table | 22 | Thomas | 2 |
| 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 |
| 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 |
| 25 | Major | Add/Mod New Item 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 | 4, 21-24 | Nic | 4 |

Time estimates were made using previous experiences as best the team could manage. In total, this sprint is estimated to take 43.2 hours out of a scheduled 48 hours. This leaves a little extra room for complications or simply underestimating the timeline.

Further to that, sprint 1 is packed much more densely than sprint 2 intentionally so that if something ends up unfinished then it can be resolved in the next sprint. This is much better than the alternative of unfinished work at the end of sprint 2 with no more time left to do anything about it.

**Member Comments**

**Table 10. Member Comments**

|  |  |
| --- | --- |
| **Name** | **Description** |
| Dylan | The sprint planning phase was very successful, the team is confident that the goals set are achievable within the week. |
| Simon | I feel that we utilised our sprint planning phase effectively to delegate goals to members in a manner that plays to our strengths as a team. |
| Rabya |  |
| Cody |  |
| Thomas |  |
| Nic | The sprint planning phase broke down all tasks into smaller sub-tasks and effectively delegated the tasks per group member. The sprint is now fully planned out and will ensure that it will be more easily completable within the timeframe. |