

Project Proposal: 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

Quality Management

Quality management on a software product relies on creating quantifiable test benchmarks and ensuring the software performs according or exceeding the criteria. Each item should be developed considering the user/client requirements.

Table 1. Definition of Done

Condition	Description
Functional Suitability	Functional suitability concerns the physical abilities of the software and if they satisfy the requirements. For this project the program must be able to perform the functionality explained in Task 02P, specifically: <ul style="list-style-type: none">- Users must be able to add, search for and retrieve details of active members by using a member ID lookup graphical interface.- Users must be able to generate reports in csv format summarising sales data over user-defined timeframes.- No more than 2 minor (still allows program to complete task) issues discovered over testing time of one week.- Able to run on computers with minimum system specs of 1GB of RAM and intel i3 or equivalent.
Performance Efficiency	Performance efficiency is a measure of how much time it takes for the program to perform key functions. No benchmark was specified by the client, but we can make a reasonable assumption to arrive at the following parameters: <ul style="list-style-type: none">- It should take no more than 2 seconds for the program to add a record to the database once the button is pressed.- It should take no more than 2 seconds to display the result when searching for a member by their ID.- It should take no more than 1 second to add an item to the sales tab of a given member, this is essential so that the sales provider can continue scanning items in a timely manner.- It should take no more than 5 seconds to compile and publish a daily report in csv format.- It should take no more than 5 minutes to compile and publish a monthly report in csv format.
Compatibility	Compatibility refers to the program's ability to be run on machines with different architecture and operating systems: <ul style="list-style-type: none">- No specific platform has been specified by the client. The program should

	<p>therefore be functional on the 3 most common operating systems: Windows, Linux and Mac.</p> <ul style="list-style-type: none"> - Cannot noticeably slow down other programs running on the target machine.
Usability	<p>Useability concerns the ease of use of the program considering the average training of the user. Though hard to quantify, the following parameters will be used to satisfy this condition:</p> <ul style="list-style-type: none"> - An operator familiar with the company should be capable of using the interface with <5% mistakes in 2 days or less. - An operator new to the company should be capable of using the interface with <5% mistakes in a week or less. - Users should not be able to enter any incorrectly formatted member or item ID's. - Interface should be similar to currently existing POS systems. - Users should be able to use the interface to the same degree as other aspects of their job, ie. if someone with a given disability is capable of working at GotoGro, then they should be able to use the program interface.
Reliability	<p>Reliability refers to the program's ability to perform its function correctly and accurately:</p> <ul style="list-style-type: none"> - To consider the software successful it should have an internal error (not caused by user error) rate of no higher than 0.5%. - Worded differently, only 1 error is permissible in every 200 item or member entries. - All database data should be preserved in the case of an outage or failure.
Portability	<p>Portability refers to the ability of the software to be used in multiple environments and still perform as expected:</p> <ul style="list-style-type: none"> - System must completely replace paper-based ordering system.

Team Discussion

Many values were consistent across the team, especially in relation to functional stability and performance. The definitions mostly contested related to defining useability, reliability, and portability. Tables 2-6 summarise the discussion and ultimate justifications that led to selecting the conditions above.

Table 2. Justification for Functional Suitability

	Detail
Chosen Definition	<p>Users must be able to add, search for and retrieve details of active members by using a member ID lookup graphical interface.</p> <p>Users must be able to generate reports in csv format summarising sales data over user-defined timeframes.</p> <p>No more than 2 minor (still allows program to complete task) issues discovered over testing time of one week.</p> <p>Able to run on computers with minimum system specs of 1GB of RAM and intel i3 or equivalent.</p>
Alternate Definition	None.
Deciding Factors	Not applicable. The team had already agreed on functional scope meaning there was no alternate opinions to be chosen between.
Justification	No justification needed; these are the functional components specified by the client.
Conformation to ISO Standards	Our requirements fulfill the "functional completeness" subheading, as meeting them satisfies the complete design brief provided by the client. Functional correctness is not explicitly addressed however we discussed

	<p>using a testing window (of a week) during which no more than 2 minor issues (cosmetic or database related) can be discovered, else the program would be considered incorrectly functioning.</p> <p>Specifying the RAM required on the operating machine obeys functional appropriateness as POS systems are likely to be lightweight and potentially old, therefore the program must run as intended on systems with the specified specifications.</p>
--	---

Table 3. Justification for Performance Efficiency

	Detail
Chosen Definition	<p>It should take no more than 2 seconds for the program to add a record to the database once the button is pressed.</p> <p>It should take no more than 2 seconds to display the result when searching for a member by their ID.</p> <p>It should take no more than 1 second to add an item to the sales tab of a given member, this is essential so that the sales provider can continue scanning items in a timely manner.</p> <p>It should take no more than 5 seconds to compile and publish a daily report in csv format.</p> <p>It should take no more than 5 minutes to compile and publish a monthly report in csv format.</p>
Alternate Definition	<p>Various small changes between 1-3 seconds for all record-based additions and manipulations.</p> <p>Generating a daily report should take no more than 5 minutes to compile.</p>
Deciding Factors	<p>There was a discrepancy between the expected time it would take for a report to be generated. The 5-minute suggestion was made under the assumption that reports would only be generated at standardised off-peak times like end of the shift or end of the month making time efficiency not a priority.</p> <p>Contrastingly, it was discussed that the amount of data needed to generate even a daily report should be able to be done in under 5 seconds in csv format.</p>
Justification	<p>Taking a standard daily estimate for supermarket sales as roughly \$70,000, and the fact that the average shopper in Australia spends \$160 on groceries per week (across all trips, but considered as a single trip here), we estimate that GotoGro serves approximately 440 members a day. Generating a csv file with 440 lines is easily achievable in under 5 seconds, even if each item the customer bought was itemised (assume 30 items per customer on average, yielding 13,125 entries).</p> <p>Additionally, it was discussed that a shift supervisor might want to quickly make a report halfway through the day to gauge their sales performance, in this case it is ideal to return this report as quickly as possible to not interrupt service.</p>
Conformation to ISO Standards	<p>All parameters come under the ISO subheading “time behaviour”, as they all define time specific benchmarks which the software must meet.</p> <p>Resource utilisation was not considered in depth given the overall spec limitation of the kind of devices that would be used (POS machines).</p> <p>Capacity is difficult to quantify given the client did not provided details of the number of customers currently in their database, however yielding a search result in under 2 seconds provides a soft benchmark on capacity, as meeting this requirement depends on the current database size.</p>

Table 4. Justification for Compatibility

	Detail
Chosen Definition	No specific platform has been specified by the client. The program should therefore be functional on the 3 most common operating systems: Windows, Linux and Mac. Cannot noticeably slow down other programs running on the target machine.
Alternate Definition	None.
Deciding Factors	Not applicable. The team agreed on platform compatibility.
Justification	It is industry standard to create programs with capability of running on at least the big three OS' being Windows, Mac and Linux. Similarly to resource capacity limitations, noticeable slowdown of other programs is a more qualitative metric but one of the most obvious to detect during testing.
Conformation to ISO Standards	ISO specifies co-existence as a subheading which is covered by our metric of ensuring no noticeable slowdown of other programs. The ISO specification of interoperability is not exclusively OS based, but also speaks to the interoperability of the program with other programs on the same device and alternative entry points such as mobile apps etc. Given our program is self-contained, and will be built to only talk to a specific single database, interoperability is a negligible consideration.

Table 5. Justification for Usability

	Detail
Chosen Definition	An operator familiar with the company should be capable of using the interface with <5% mistakes in 2 days or less. An operator new to the company should be capable of using the interface with <5% mistakes in a week or less. Users should not be able to enter any incorrectly formatted member or item ID's. Users should be able to use the interface to the same degree as other aspects of their job, ie. if someone with a given disability is capable of working at GotoGro, then they should be able to use the program interface. Interface should be similar to currently existing POS systems.
Alternate Definition	Operators should be able to use the interface with less than 5% error within a day.
Deciding Factors	Since members of the team have retail experience using similar systems we discussed and drew from those team members' experience to determine what timeframe was most reasonable to expect from a staff member.
Justification	Mistakes are damaging to a business; therefore, the priority is to ensure that the mistakes being made are minimised due to error checking. Obviously, user error is unavoidable (entering in the incorrect member ID), but this is where ease of access and learnability plays a key factor in helping team members to get comfortable with the interface as quickly as possible.
Conformation to ISO Standards	Appropriateness recognisability, and user interface aesthetics are both mentioned in ISO and are covered here under the line concerning designing the interface similar to currently existing systems.

	<p>Learnability and Operability are covered by the considerations speaking about the timeframe and error margin that workers should be expected to uphold using the system.</p> <p>Accessibility is directly addressed with our requirement that any current GotoGro worker should be able to use the interface.</p> <p>The final ISO requirement is user error protection. Given that items will be entered via barcode scan there is no need for error protection, the best error protection we can feasibly implement is ensuring correct formatting for member ID input.</p>
--	--

Table 6. Justification for Reliability

	Detail
Chosen Definition	<p>To consider the software successful it should have an internal error (not caused by user error) rate of no higher than 0.5%. Worded differently, only 1 error is permissible in every 200 item or member entries.</p> <p>All database data should be preserved in the case of an outage or failure. Only the current transaction will need to be redone.</p>
Alternate Definition	System should be able to be recovered and restored in 30 minutes or less.
Deciding Factors	Team discussion was conducted concerning the feasibility of restoring the system in a given timeframe. Though a good benchmark, it is unrealistic to hold the quality of the program on a timeframe that is largely out of our control. Even the best software can suffer an attack or malfunction that can take days to repair.
Justification	<p>Setting the benchmark for acceptable failure is somewhat arbitrary but based on the estimated number of customers served per day. Taking the number from above (440), our maximum permissible error criteria would allow for 2 issues in data/member entry over the course of a single day. Using team member retail experience, this was deemed acceptable because the rate of mistakes made due to operator error is already much higher and to the customer there is no distinction in their service experience.</p>
Conformation to ISO Standards	<p>ISO standards mention Maturity as the specification for how well the software performs under expected conditions, we have accounted for this with the minimum permissible error criteria.</p> <p>As for availability, since the system is small scale to start with (one shop only), the database access should not hinge on many factors other than the network setup in the store. It is not directly addressed in our conditions, but it is safe to assume that the program itself will not cause any foreseeable downtime.</p> <p>Fault tolerance is another standard that is hard to quantify, we do not benchmark it here as it is assumed that hardware faults will be covered by redundancy (multiple POS systems active).</p> <p>Recoverability is the final ISO standard, though we decided to exclude the timeframe constraint, we still address recoverability by enforcing the condition that no database entries should be lost in the case of system failure.</p>

Table 7. Justification for Portability

	Detail
Chosen Definition	System must completely replace paper-based ordering system.
Alternate Definition	Program should be able to be installed on new sites within a day.
Deciding Factors	The alternate suggestion was excluded here due to scope. Given the client's brief, the program is only expected to function in a single store, rendering the installability and adaptability of the system largely negligible.
Justification	Portability blankets over the subheading replaceability, which refers to the software's success in replacing the system it was designed to replace. It is clear from the client brief that this program is designed to replace a paper-based management system.
Conformation to ISO Standards	As mentioned, adaptability and installability of the ISO specifications can be considered negligible for this project due to them being out of scope. Replaceability is the final specification to address and we have done so by ensuring that no staff members are using the paper based system after the software is implemented.

Member Comments

Table 8. Member Comments

Name	Description
Dylan	Discussion was sufficient to clarify the teams definition of done. We believe this list of conditions to adequately represent a "good" outcome for our software and also to be in compliance with the ISO standards in ISO25010.
Simon	I believe that following extensive group debate and discussion we were ultimately successful in working out a DoD for the current project that every member of the team agreed with.
Rabya	The definition of done that our team has produced is pretty detailed and informative. We have discussed and debated on various topics which are shown as a part of our final definition of done file.
Cody	Our team has produced a definition of done that I'm happy with. We combined all our ideas and responses for the task and had a discussion to clarify and resolve any conflicts. The final result meets the expectations I have for our program and follows the ISO standards.
Thomas	I agree with what our team decided. The amount of information and in depth information that we got I am very happy with.
Nic	The team had an in-depth discussion about about we thought 'done' meant and ultimately agreed on the seven conditions mentioned above. Some of these may slightly change as we further the development process such as hardware requirements but ultimately I agree with the team. The end definition is one I believe is accurate and detailed.