Milestone Submissions

Capstone Computing Project 2

Executive Summary

During every sprint, you should be iterating your previous version of the product as per the Agile process. Each team member should be contributing to this.

Your documents will need to show the following:

1. [Team] Your current product, including what part of the project specification have been met. This will include links to the code on your repository and testing information that demonstrates your claims of correct functionality.
2. [Team] Evidence that the team has followed the agile process and otherwise worked in a suitable and sensible manner.
3. [Team] A joint statement of attribution of work.
4. [Individual] A brief summary on the parts of the team's work this sprint that were completed by you. This should emphasize any highlights of your work in the last sprint.
5. [Individual] A brief summary of any additional work you have undertaken that relates to the project, but not directly to the product.
6. [Individual] Evidence that the you have been following the Agile process.
7. [Individual] The time that you have spent on the project during this sprint.

Points 1-3 will be covered in a single document and points 4-7 will be covered in a second document. Note that this means that each team will need to make one submission jointly, and each team member will, in addition, need to submit a report themselves.

Milestone 1-4 Submission Requirements (10 marks each)

Both the group and individual documents will need to be submitted via the TurnItIn[1] links on Blackboard.

This means that you must submit a single document, which should be in PDF format.

Change: The last sprint report and handover document will be submitted using a Blackboard assignment. This means you can submit documents in separate files if needed. All files submitted must still be in the PDF format. All files names must still be named as below, with separate documents appended at the end if appropriate.

General Requirements

Each submission (team and individual) must be submitted via the appropriate Blackboard link. Each submission must be a single document[2] in PDF format[3]; any other submission or format will not be

[1] All documents are text-based, and thus must be submitted via TurnItIn as per Curtin policy.

[2] TurnItIn only allows one document to be submitted for an assessment. We are already using two submission links – one for group and one for individual – having one link per part of the submission isn't practical. The easiest thing may well be to create the separate parts as different documents, merge them in a word processor and only then convert to PDF. There are also free programs available that allow you to merge PDF files, and many Curtin Windows-based computers have Adobe PDF installed.

accepted. Note that you can submit as many times as you like, and only the last submission is counted. Be careful to avoid using public domain fonts that may not function on a Windows machine or with TurnItIn or not be easily available to the marker.

The documents must be named as follows[4]:

Group document: “<group>_Report_<milestone>.pdf”

Group document: “<group>_Report_<milestone>_SRS.pdf”

Individual document: “<group>_<student id>_<name>_Report_<milestone>.pdf”

where <group> is your group name/number (*e.g.*, E8, 21, SS01), <milestone> is the number of the report (*e.g.*, the first milestone report would be Report_1) and <student id> is … obvious. If a document is not named properly and is missed as a result, it may not be accepted. As such, proper naming is part of the submission requirement.

Note that, as per the unit outline, late submissions are only accepted if they do not impact the rest of the group. This means that if an individual submission is to be late, this should not involve any work on the documents that are submitted as a team. However, if the team document is submitted late, the lateness penalty will be applied to the entire team. You must notify your supervisor (in writing) if you are going to submit more than a few minutes late. This notification must be sent at the time that submission is due – not doing this could cause the submission to be ignored[5].

Your submission must be professional. That does not mean that you need clip-art, fancy fonts or other gimmicks, but your work *must* be properly checked for spelling and grammar and follow the normal rules of professional submissions. In particular it must be clear, concise, and any material that draws on the work of others must be properly referenced using a consistent referencing scheme.

1. Team Submission (70%)

Summary

The main purpose of the team submission is to present the current state of the product and project, provide evidence that the state is exactly what is claimed.

Every sprint must result in a product. It is permissible (but not advisable) for the first sprint to have a fully non-functional prototype. Every subsequent sprint must include a version of the product that implements more of the functionality than the previous one. The functionalities implemented should focus first on those required for project success, and only move onto “nice to have” functionality later.

---

[3] We want to standardise submissions as much as possible, and there have been previous issues with submissions that can’t be easily read/marked. In this case, we want to ensure that the marker can read the document(s) within TurnItIn so that they can apply the marking rubric.

[4] Some supervisors will have fifty or more reports to mark, which means downloading all reports from Blackboard as a batch and then sorting them into groups manually for record-keeping. Proper file naming makes this far easier.

[5] Because multiple submissions are enabled and we want supervisors to start marking as quickly as possible, notification must be given for late submissions. If a supervisor is not notified and marks one submission only to find that another submission has then been made, the later submission will be ignored.

Note that a student whose individual submission does not clearly show a strong contribution to the team may have their part of the team's submission mark decreased. If the team submission is obviously suffering because of particular members, the mark for students who are doing the right thing may be increased to make up for this. This requires proper reporting of these difficulties under the individual "additional work"

Part 1 - *The state of your product

You must provide a brief overview on the state of your product. This should include at the very least a link to the pull request for merging the parts of the product into a complete release branch and a brief discussion of the testing. The latter should, at the least, link to the test codes and any test scripts that you have used.

Everything that you are including in your "current state of the product" must be coded, pushed to Bitbucket, fully tested and merged into the release branch for the sprint. Anything that doesn't fit this description will be ignored. You won't need to address unit testing, because this will be covered in the individual reports, but should focus on your code working as a whole.

If a team is deploying via Docker or a similar system, they may choose to provide their current version to their client and supervisor via an agreed-upon-mechanism such as the submission of a USB drive or upload to an agreed location. This is in addition to the material on Bitbucket and must be completed before the due date/time.

If you gained client acceptance or received client feedback for any parts of the product, you should include a link to the minutes or e-mails that evidence this (see Agile below). It is strongly recommended that you arrange a client meeting shortly before the milestone is due to make this possible.

Industry students must submit this section of the report, but only a one or two sentence summary and the links to client feedback/acceptance. The aim is to give a rough overview while still requiring minimal reporting. Feel free to check with your supervisor for details.

Part 2 - *Agile

Your team must demonstrate that you have followed the Agile software development process. This specifically includes the following:

- That you are having regular meetings with the client to seek their feedback. This should be evidenced as follows:
    1. Minutes must be taken of all meetings. These minutes must be sent to the client and all attending group members via e-mail within 7 days of the meeting, with a CC to your supervisor. The e-mail must include a request for the client to raise anything that is not accurate, and a statement that if you don't hear from them in 7 days you will assume that they are accurate.
    2. Any e-mail communication should include a CC to your supervisor.
    3. In order to allow cross-marking, moderation or re-marking in case of an appeal minutes and important e-mail chains should be stored on a group folder that is linked from your submission document. If you don't do this you are relying on good communicating between the supervisor and all subsequent markers, which is a risk.

- That you are regularly communicating with each other, including meetings at the start and end of every sprint and regular stand-up meetings.
    1. The large meetings should be evidenced as above, with minutes sent to all attending group members, with a CC to your supervisor. These minutes should also be stored on the shared drive, as above.
    2. Smaller meetings (such as the stand-ups) should have very brief minutes, or a chat log if the meeting is fully online. These do not need to be sent by e-mail but should be stored on the folder as above.
- That you are adequately tracking your progress and are on-target for on-time completion.
    1. You should address metrics such as total team time and the number of features completed and outstanding. The features discussed should be those from your SRS or contract.
    2. You should communicate these metrics in appropriate ways such as burn-down and/or burn-up charts although you may also want to discuss them briefly if they aren't as expected. If you are using charts, you are encouraged to set up a spreadsheet that takes the information and creates the charts, in such a way that it can be re-used in future reports.
    3. If you are not on-target (which can easily happen) you must include a brief plan for getting back on track.
    4. Your reflections as a group on the last sprint, as per the Agile process. This should not be more than a couple of paragraphs at most (and can be as short as a sentence or two).

<u>Industry students must submit this section of the report.</u>

Part 3 - Repository

The submission after the first sprint must include a brief plan that describes what rules the team will use to manage its repository, especially with regards to branching and merging. This plan should be approved by your supervisor.

Sprints after the first must briefly demonstrate that this plan has been followed. This should be obvious from the repository links provided, but any difficulties or changes to the repository plan need to be specifically raised.

Part 4 - *Attribution

Where any work has been the work of multiple students, the work put in by each student must be clearly attributed. No code should be created entirely by two students, and the changes made by separate students should generally be seen in who carried out the commits. If this isn't the case (which could be the case with documentation or research, for example) your milestone submission must include a section that clearly attributes who did what. Any students disagreeing to this attribution should include a statement of dissent in their individual submission.

You will specifically need to attribute the writing of the milestone report. You can do that in this section, or by having a team member that writes a section put their name to that section.

<u>Industry students must submit this section of the report.</u>

How to submit

The team must choose a single student who is responsible for submitting the group document. Partially this is to avoid having competing submissions, but mostly this is because the whole group is potentially affected by a late or incorrect submission. The supervisor must be notified in writing of who the nominated person is. If this is done via e-mail, all group members must be included in the e-mail, to ensure that they are aware and can raise any issues that they have.

Note also that the team document is submitted on behalf of the whole team. That means that if one team member does something wrong, such as extremely poor referencing or plagiarism, the penalty will be applied (in at least some measure) to the whole group. We strongly recommend that the entire team check the group document before it is submitted.

Academic Integrity

Any documents must meet Curtin's academic integrity policy, meaning and the discipline code commenting and coding standards. You should reference sources and code as per the usual guidelines. You may definitely use code from other sources, but need to reference clearly. This means comments in your code AND a reference in this document.

2. Individual Submission (30%)

Summary

The main purpose of the individual submission is to evidence your contribution to the team's effort. While the unit has a group project, we want to ensure that all members contribute to a fair degree.

Part 1 - *Cover Sheet

The first page of your report should be the cover sheet that is used by all computing units. This cover sheet can be found on Blackboard. Submitting this to Blackboard is considered a digital signature, and the date of submission will be taken as the date of that signature. In other words, you can prepare one cover sheet and use it for each report.

<u>Industry students must submit this section of the report.</u>

Part 2 - *Status (No marks, but report not accepted without this)

We require a brief (generally one sentence) statement that you are currently working as planned and there is nothing major affecting your progress in this unit. If this is not the case, please instead let us know that there is an issue.

You don't even have to give us any personal details, but something like "I'm having some family issues that affected week x of this sprint." or similar is fine. Note that we strongly recommend that you get some evidence if anything is seriously affecting your study – a note from a counsellor or similar person is acceptable even if it says nothing other than that something was affecting your ability to work.

If there are issues with the team, please let us know even if you have probably resolved the issue. This could be as minor as there just being an issue with adjusting to a team member's work style to serious issues such as someone not pulling their weight or behaving inappropriately.

In general, at the end of the semester we want to be able to judge that you put in an appropriate amount of effort. If you recorded no difficulties during the semester, you will want to show that you've spent about 10 hours per week on this unit. If something made that impossible, give information as soon as possible and have evidence ready if needed.

Industry students must submit this section of the report.

Part 3- Contribution Highlights

You should briefly discuss your part of the team's submission. In particular, you should present links to your key commits during this sprint in such a way that your code contribution can be judged by the marker. One way to do this is to list all of your non-trivial commit and add a sentence to each, describing what it does.

You will be judged on the quality, commenting and other aspects of your code, so if some code is auto-generated you need to make this clear to the marker. (You should do this anyway, for reasons of academic integrity.)

You need to evidence that each code contribution works as intended. This means your unit testing needs to be referenced, as well as other evidence such as client acceptance. Any testing that is discussed in the group document should not be duplicated here, but you should mention that you were responsible for this in the attribution section of the group document.

,Part 4- Additional Work

You should have credit for all of the work you've done during the last sprint. For example, if you've written the minutes this should have been noted in the group Attribution section and in your highlights section. However, if there is anything that you have contributed that doesn't fit into anything else, mention it here. You will have to evidence it appropriately – check with your supervisor well before submission of this document if you're not sure what do offer as evidence.

Part 5- *Agile

Much of what you have done should be mentioned in the group document. For your individual Agile evidence you should present your time tracking (as a Toggl report included in your document) and individual progress evidence. Again, experiment with this for the first report, but it should be set up in such a way so as to not take too much time for later reports.

Industry students must submit this section of the report.

Marking

Your milestone submissions will be marked as shown below. Note that if the individual report does not have a cover sheet, it will be considered as not submitted. Also, if an individual report is not submitted, the student will not get their share of the group report. Any lateness penalty is applied equally to both reports (since they are effectively meant to be a single submission, but TurnItIn doesn't make that easy). This means that if the group report is late, every student will get a lateness penalty, but if the group report is on time and a student submits their individual report late then the penalty for that student will be applied to both reports.

Group

Product (60%)

Not Present (0%)	Insufficient (25%)	Marginal (50%)	Good (75%)	Excellent (100%)
The product has either not advanced since the last sprint or there is no substantial evidence to support its advancement. For the first milestone, this would mean that there is no evidence of progression towards a product. This is the only category available without a working proto-type from milestone 2 onwards.	While there is some evidence that product iteration has occurred, it is unconvincing. This is the highest category possible without links to appropriate commits on BitBucket.	While work has obviously been undertaken, there is insufficient evidence to show that the progress has been well tested and aligned with the product requirements. This is the highest category possible without testing that is relatively complete.	Evidence has been given that work has progressed in the right direction, and testing has been adequate. There are still shortcomings in either how the repository is managed or similar things. This is the highest category possible without evidence that the team is complying with an agreed branching plan for the repository.	Progress has been made and evidence has been given that the work has been adequately tested, that the branching plan is being followed and the work is going in the correct direction. This level includes things like the attributions being clear and the tasks chosen making sense given requirement priorities.

Agile (20%)

Not Present (0%)	Insufficient (25%)	Marginal (50%)	Good (75%)	Excellent (100%)
There is no more than a token attempt to evidence use of an Agile process.	While some attempt has been made to demonstrate the use of an Agile process, it is unconvincing. Either there is little evidence or some categories have been ignored entirely. This is the highest category possible with no minutes at all.	An attempt has been made to address all categories, and most of them have some form of evidence. Either some of the evidence or some of the way it is addressed is still lacking.	All categories have been addressed and evidence has been given. Some parts are not fully convincing.	All categories have been fully addressed and supported by proper evidence.

Professionalism (20%)

Not Present (0%)	Insufficient (25%)	Marginal (50%)	Good (75%)	Excellent (100%)
The document has enough errors that it has little or no use as a legal protection. The client is very likely to form a negative impression of the group based on this document.	The document has many errors that should have been found and corrected. The client is likely to form a negative impression of the group based on this document.	There are several errors or signs of poor judgement, but much of the document is sufficiently professional. The errors are not likely to detract from the legal value of the document.	The document is professionally laid out and presented, although there are one or two minor errors or omissions that do not greatly detract from it.	The document is professionally laid out and presented. This means that it makes use of bookmarks (the F4 pane in PDF documents) for all major headings, follows a logical layout and as had few or no errors.

Individual

Product (60%)

Not Present (0%)	Insufficient (25%)	Marginal (50%)	Good (75%)	Excellent (100%)
The team member's	While there is some	While work has	Evidence has been	Progress has been

part of the product has either not advanced since the last sprint or there is no substantial evidence to support its advancement. For the first milestone, this would mean that there is no evidence of progression towards a product. This is the only category available without a working proto-type from milestone 2 onwards although exceptions may be made if the lack of prototype is due to another team member.	evidence that iteration of the student's component has occurred, it is unconvincing. This is the highest category possible without links to appropriate commits on BitBucket and proper attribution of work. If a significant proportion of the work is not attributed solely to this team member that also is a reason for selecting this category – individual progress is necessary.	obviously been undertaken, there is insufficient evidence to show that the team member's progress has been well tested and aligned with the product requirements. This is the highest category possible without testing that is relatively complete.	given that the team member's work has progressed in the right direction, and testing has been adequate. There are still shortcomings in either how the repository is managed or similar things. This is the highest category possible without evidence that the individual is complying with an agreed branching plan for the repository.	made and evidence has been given that the team member's work has been adequately tested, that the branching plan is being followed and the work is going in the correct direction. This level includes things like the attributions being clear and the tasks chosen making sense given requirement priorities.

Agile (20%)

Not Present (0%)	Insufficient (25%)	Marginal (50%)	Good (75%)	Excellent (100%)
There is no more than a token attempt to evidence use of an Agile process.	While some attempt has been made to demonstrate the use of an Agile process, it is unconvincing. Either there is little evidence or some categories have been ignored entirely. This is the highest category possible with no minutes at all.	An attempt has been made to address all categories, and most of them have some form of evidence. Either some of the evidence or some of the way it is addressed is still lacking.	All categories have been addressed and evidence has been given. Some parts are not fully convincing.	All categories have been fully addressed and supported by proper evidence.

Professionalism (20%)

Not Present (0%)	Insufficient (25%)	Marginal (50%)	Good (75%)	Excellent (100%)
The document has enough errors that it has little or no use as a legal protection. The client is very likely to form a negative impression of the group based on this document.	The document has many errors that should have been found and corrected. The client is likely to form a negative impression of the group based on this document.	There are several errors or signs of poor judgement, but much of the document is sufficiently professional. The errors are not likely to detract from the legal value of the document.	The document is professionally laid out and presented, although there are one or two minor errors or omissions that do not greatly detract from it.	The document is professionally laid out and presented. This means that it makes use of bookmarks (the F4 pane in PDF documents) for all major headings, follows a logical layout and as had few or no errors.

Handover (10 marks)

Overview

When your project is finished (which hopefully means before you submit Milestone 4) you will need to hand over the product to your client. You will have specified in your SRS or another document what form this handover will take[6] so you know what is expected of you. You will need to arrange a meeting with the client at some point before the due date of the handover submission to carry out the handover.[7]

During or before the handover you may need to install software on a machine specified by the client. If not, you may need to make your product available before the handover for the client to install. In any case, during the handover meeting you will ensure that your client has received all deliverables specified in the SRS (including any documentation) and you will answer any questions that they have.

After the handover, the client will be asked to provide a degree of agreement on how well your product solves every functional requirement in the SRS. This will be done by them filling out a handover document that you create based on the sample fragment below and will supply to them. Note that the student(s) column can be absent from the version given to the client if desired but must be present in the version submitted to Blackboard and must be completed with the names of the students responsible for that requirement. The client will need to check every requirement and give their opinion on how well your solution meets it.

It may be worthwhile walking your client through the functions in such a way that you can be sure that the client understands the functionality, but the client should also be given time to experiment on their own to satisfy themselves that all aspects of the requirement are covered. The presentation of the handover document to the client is part of the handover procedure and the client is asked to submit the document (via e-mail to the supervisor) within one week of the handover meeting but you won't be penalized if they submit late for reasons that are not connected to your group.

Steps Required

1. The group completes all deliverables and testing.
2. The group delivers all agreed upon deliverables to the client in the handover meeting.
3. The group prepares a Handover Document to the client in the handover meeting.
4. The group submits a copy of the deliverables that are not on the repository, a copy of the Handover Document and a copy of the minutes of the handover meeting to Blackboard using the submission link. (There is no individual submission.)
5. The client raises any immediate issues with the group during the handover meeting.

---

[6] If your project did not have a first half (CCP 1) for whatever reason, then you may not have a written agreement on how the client wants the product to be handed over. In that case you are strongly encourage to either add that to the SRS or to discuss it in a client meeting. In either case you will want written evidence that the client accepted it, which is either an e-mail or an acceptance of the meeting minutes where acceptance took place.

[7] Do not leave this too late, because trying to organize a meeting with the client at the last minute and not being able to get an appointment won't be an acceptable reason for submitting late. In fact, it is advisable to schedule the meeting at least a couple of days before the documents are due, which gives you a chance to re-schedule if something comes up.

6. The client fills in the Handover Document and sends it to the supervisor (and optionally to the team).
7. The group addresses any last-minute issues (either raised in the meeting or from the Handover Document – if applicable) in preparation for the final submission.

Rubric: For the handover documents, the group will receive a mark based on the percentage of the project that was completed. If a student's part of the project was fully completed (and tested) and they put in sufficient hours, they get full marks. As either criterion decreases, they will lose marks proportional to that decrease. Thus, a student whose component(s) are considered to be 80% completed and who has put in only 50% of the time expected can expect to score around 65% (80/2 + 50/2) of the marks for this assessment.

Because the completeness is somewhat judgement based and there can be complicating factors for the time expected, the mark for this assessment will not be released until moderated by the co-examiner (and, if applicable, the UC). This mark is summative (i.e., not intended as feedback) so may not be ready before the exam period starts.

Sample Handover Document Fragment

Group: 007 Secret Agents

Client: Dr Yes

Functional Requirements	Student(s)	Non-Functional	Occasionally Functional	Somewhat Functional	Mostly Functional	Fully Functional
FR 1.1						
FR 1.2						
FR 1.3						
…						

Non-Functional Requirements	Student(s)	Not Observed	Occasionally Observed	Somewhat Observed	Mostly Observed	Fully Observed
FR 1.1						
FR 1.2						
FR 1.3						
…						

Final Submission (25 marks)

The final submission is very similar to a milestone submission, in that it includes both a group and a general part. Apart from the below modifications, use the format for Milestone Reports. In general, all parts in the final submission concern the overall completion of the product, not the progress of a particular sprint.

In this case the state of the product addresses the final state of the product, which is also an opportunity to explain any shortcomings or to highlight any strengths. Links to Bitbucket pull requests are only needed if they occurred after Milestone 4.

The Agile section should focus on how the measures used in previous group reports measured the actual progress of the project. If they indicated that the project was going to be completed on time and this did not occur, why not? If they indicated problems with completion but the product was completed anyway, what was done to achieve this? This does still needs to include links to the minutes from meetings after Milestone 4 and the hours of team members in these two weeks.

The repository should now be complete. The final submission must link to each sprint branch product and address how the team has followed the branching plan for the semester.

Attribution is similar to the previous reports, documenting the work of group members in the final two weeks. Note, if your product was completed at the end of milestone 4 if is very likely that team members will not be putting in the full 10 hours per week in these final two weeks, and in this case it is permitted. Where the work of one or more students was not completed, those students may want up needing to work additional time to get their parts working at their discretion.

Overall, the aim of the project is to deliver a fully working product to the client in the manner specified. The main task of this report is to provide evidence that this has occurred. Part of the judgement of this will come from the documents submitted for the handover milestone; if any part of those indicated less than complete completion and this has been corrected since then, the group is encouraged to provide specific evidence of this.

The group must also (re)submit the following as part of their final submission:

- SRS
- Branching Plan
- If repairs have been made to the product after handover and the client agrees to fill in an updated Handover Document, this may be included. Otherwise the acceptance of these repairs can be submitted through testing and minutes as evidence.

Team Rubric (70%)

Individual Rubric (30%)