1. Important notes

- Check the studio review checklist on Moodle (under My Assessment).
- Remember each iteration results in a "**done**" / shippable product.
- It is expected that you will meet with your client throughout the development, not just at acceptance testing - Please "show and tell" (showcase)- to get feedback immediately so changes can be made.
- Studio reviewCarried out regularly in the studio during the development of each iteration
- Used as part of the assessment for the final implemented iteration.

2. Aims

Students **successful** with this assessment will:

1. Work with clients and present effectively;
2. Turn user stories into detailed requirement documentation **for this iteration;**
3. Design a system meeting prioritised user requirements **for this iteration**;
4. Effectively manage ongoing changes to requirements;
5. Build on **previous working iterations** to meet adapted requirements **of this iteration;**
6. Gather feedback from clients and mentors and:
   - Record, circulate, and a document feedback
   - Act on feedback where appropriate
   - Discuss and justify when not acting on feedback
7. Follow the standards developed for the system and project;
8. Meet industry-strength standards for development including maintainability, testing, etc.
9. Show clients an iteration working **as expected** and obtain acceptance sign-off;
10. Work effectively as a team with a strong set of team processes;
11. Critically reflect on achievements/processes with retrospectives.

3. Due Dates & Weighting and Submission 18%

(See assessment guide on Moodle for any changes)
Trello Board and PGP up to date at all times with support artefacts. (At any given time, they should reflect the current state of your system). **Fixed artefcats go to your PGP**

Reviewed with you in studios	**Iteration 1 - 18 % Due**	**Iteration 2 - 20% Due**	**Iteration 3 - 20% Due**
Iteration Analysis & Design Report and Presentation -what is to be delivered	Week 2 Part of Bus. Vision report and presentation.	Week 6 Sun 28 th August 11.55pm Week 6 studio 1 client presentation	Week 9 Sun 18th Sept 11.55 Week 9 studio 1 for client presentation
Run sheet and Acceptance testing report	Week 4 Wed 17th August - Moodle	Week 7 Wed 7th Sept - Moodle	Week 10 Wed 5th Oct - Moodle
Completed build - to be integrity tested	Week 4 End of studio 2	Week 7 End of studio 2	Week 10 End of studio 2
Retrospective, after feedback - Retrospective Artifacts	Week 5 studio 1	Week 8 Studio 1	Week 11 studio 1
Client acceptance testing (from client folder, if not deployed on to Client's ISP)	By Week 6 studio 2	By Week 9 studio 1	By Week 12 studio 1
PIR (if implemented live)	Week 7	Week 10	Week 14

4. Guidelines

1. Requirements - on TRELLO Board

What to submit	**Details**	
Iteration 1 Analysis & design Report - Presentation Support artifacts	All changes made by client during presentation will be changed on the report (in PGP) and the new changed report sent to them (clients), or if report is already sent, changes can be minuted and sent to client. **Iteration 1 priorities labelled on Trello Board.**	already completed in Business Vision
User stories and acceptance criteria	Manage as you work through the iteration**.** Add to Trello as they happen. Your trello board will be constantly reviewed by studio mentors.	on TRELLO

AUDIENCE: development team, Mentors (for reviewing)
FORMAT:
Backlog

- High level epics/functions/client ideas with priorities for next iteration
    - Are highest priority (according to your client drawing on your advice)
    - Are numbered to match the overview and the acceptance testing doc

Waiting lane

- Functions/client ideas which are :
    - Broken down into user stories and then associated tasks (as you work through the iteration)
    - Are prioritised further for development within this iteration

To Do lane

- Above but with detailed acceptance criteria, validated from client discussion
  User Story Acceptance Criteria Examples and Definition in 2020

Acceptance Criteria which:

- Are specific and come from the client for that user story
- Allow the team to build to “Yes, this is what the client wants” or “No, this isn't” can indicate the iteration is done

Take cards from the to-do lane to build and test.

Note: As your user stories move into the “done” column of your kanban board, you will show your client, via the development folder, to see if this is what they want.
Requested changes should be added to your Trello board and documented as **requested changes**, in minutes sent to your client.

2. The Build

<table>
<tr><th>What to submit</th><th>Details</th><th>Where to submit</th></tr>
<tr><td>Acceptance testing document</td><td>Max length - 1-3 page</td><td>Moodle with link to build in review for mentors
After build review and fixing - to PGP with link to build in production for clients (and then out to clients)</td></tr>
<tr><td colspan="3">AUDIENCE: Mentors (for viewing that IT 1 hasn't changed from the business vision presentation and to complete integrity testing) and then, after agreement, the clients to accept the test (build link changed to production folder)
FORMAT:
An acceptance testing doc matching these high level requirements (for client use later)
○ Must be in client language - functional requirements/ goals
○ Perhaps use a diagram to help your client e.g functional decomposition diagram
○ Consider grouping by workflow to aid clients later</td></tr>
<tr><td>Details for Integrity test</td><td>Max length - 1 page</td><td>Moodle with Link to Build</td></tr>
<tr><td colspan="3">AUDIENCE: Mentors
FORMAT: Meaningful written notes
In PGP current link, passwords, user instructions etc for mentor
Concise clear instructions to review the latest version of the systems</td></tr>
<tr><td>The Build</td><td>The iteration build on review server</td><td>Link in PGP and acceptance test doc.</td></tr>
<tr><td colspan="3">AUDIENCE: mentors for quality control checking
FORMAT:
The latest version of the system which...
○ Matches Trello board
○ Matches client requirements (from presentation and feedback)
○ Is robust and complete (works from end to end)
Note: System URL does not change across iterations. You will be adding to the system as you develop the user stories. This build is cumulative and contains all current iteration user stories plus those from previous builds and fixes.
Corrected/fixed updated build on to client/production folder/server and URL to client, for review</td></tr>
<tr><td>Retrospective artefact</td><td>Submit after feedback on integrity test</td><td>Moodle</td></tr>
<tr><td colspan="3">Retrospective artefact max length - 1 page
AUDIENCE: Mentors, Team
FORMAT: Reflective artefact

A reflection on processes:
○ What went well,
○ What didn't go well
○ How can we improve?</td></tr>
</table>

3. After The Build

What to submit	Details	Where to submit
Refined Acceptance testing report for client	Should match A&D report	To PGP and to client for acceptance testing(on client folder or ISP, if client wants to implement/ deploy this iteration

AUDIENCE: Client
FORMAT: Client commented feedback on acceptance test report

- Acceptance Testing Report is given to clients (face to virtual face)
- Acceptance testing explained
- Iteration should ideally be on the client server but hidden from the general public
- You will leave the clients to test themselves and return results to you.
    - These will be shown to the mentor in the studio.
    - You will document and fix small issues and either...
        - the client will re-acceptance test
        - you and your client move larger changes to the next iteration..

Iterations once accepted by the client will be uploaded to the client's own hosting service (hidden), assuming this is what they want, for acceptance testing otherwise will be kept in your client review environment on the Monash web server for acceptance testing

Client wants Build to be Deployed	Onto Client's ISP - hidden until client says ready to go live Acceptance tested here.	Client ISP

Client Acceptance test results	Photos or scans of results from client	To PGP
PIR Minutes or Client review minutes, if not running live	After client has run system live (or on client Monash server) for 1-2 weeks	Moodle PGP and client

4. Project Governance Portfolio

AUDIENCE: Development team, Mentors (for reviewing)Project governance portfolio up to date at all times (at any given time, it should reflect the current state of your system and your work).

Client communications

- Minutes, emails and so on, recordings of phone calls.
- Changes from feedback from the acceptance criteria etc,
- Details from presentations.

Note: The client must have received a copy of everything in here. The client receipt of such will be checked.

Testing Details, this describes:

- The process used to know each user story/ function is correctly implemented.
- Kinds of testing data you used.

Note: All a sample of test cases should be here, ato indicate how you are testing and should::

- Use the acceptance criteria as the starting point
- Cover all eventualities that might arise for each of the acceptance criteria.

Development artefacts

Analysis and design artefacts, how did you get to your solution, may be user story mapping, flowcharts, procedure diagrams, activity diagrams,Interface mock-ups, rough sketches, user stories, mind maps etc. used to facilitate understanding of client's requirements and achieve this deliverable: what you used to get to this document. Photos of drawings are fine. Testing approaches should be here too, and what you tested for. There is no need to include 500 test cases, but you will have to have tested your system 500 times.

Feedback
Feedback noted from mentors

Maintenance/System/technical Doc - will be used for maintenance in the future and should enable modifications to be made easily.

- System Architecture (Architecture of technology use)
  - may be diagrams/ lists of modules used, system
  - activity diagrams, ER conceptual diagrams
  - Technologies used (and any APIs, plugins, ect.)
- Technical specification (see technical doc. guidelines for complete requirements)
  - Updated use cases (if used, due to changes in functionality requested by client from original use cases)
  - Working detailed data models (i.e. updated from last iteration, adding new areas that you have understood from further analysis) and so on.
  - Well labelled code (commented) with good naming conventions, formatting, understandability, scripts
  - and so on…

Extensions and late submissions

- Organise extensions through your mentor, otherwise the in-semester special consideration process
- Late penalty: 10% per day (inc. weekends) of marks received if no extension asked for or agreed t