**Peer Review Assessment Team Template**

Questions should be focused on these areas to provide content for the business letter that needs to be written up within 3 days:

Things that should be shown during the presentation:

* Sprint and release plan for sprint 1
* User stories for sprint 1, including acceptance criteria
* Description of their system architecture and their rationale for choosing this architecture
* Product

Preparedness:

* Was the team prepared for the demonstration?

The team didn’t have a presentation ready, they weren’t 100% confident in their demonstration

* Was the product able to be demonstrated, if not, investigate why the product was not in a demonstrable form in a non-judgemental fashion. Compare their sprint items directly to their product and make notes of where there are discrepancies.

Their work in progress was not able to be demonstrated due to an underlying server issue at the time

Expectations

* Note how well the product meets the acceptance criteria of stories in the sprint. Note if the acceptance criteria is partly met, and ask questions based on missing acceptance criteria pieces (if any).

A lot of part acceptances, code has been written for the stories but hasn’t been deployed or implemented in the current model

* Does the product developed thus far deliver business value for the client?

With minor fixes to the web server, the product developed will provide business value, however at the time of the demonstration, it does not.

* Note any deviations from the sprint plan, Did the team plan for deviations before the end of the sprint?

The team pushed some user stories back to sprint two, they said they overestimated what they could achieve during sprint one,

* How many stories passed all of their acceptance criteria

Few. There were more incomplete than complete at the time of the demonstration due to the webserver not correctly working and code that had not been deployed.

Technical

* Was the presentation made at the
* Is the developer team able to articulate the system architecture and their reasoning for choosing it
* Any bugs in the current product related to the first sprint.

Database related.

Professionalism

* Was the product presentation clear, coherent

For what was presented, explanations and reasoning was clear and concise. The developer team understood their current downfalls and how they were going to amend them to make up for issues in the current sprint and next sprint

* Were explanations clear and concise

Yes

* Did the presentation appear to be prepared in advance or was it improvised?

The team understood what had to be performed in the presentation, however weren’t completely prepared for the presentation

* Was the presentation confidently performed, how smoothly did the team recover from problems during the presentation?

The team were confident in their explanations and their current model

* Were all team members involved in the project and presentation?

Work load seemed evenly distributed, all team were involved in the presentation

* How confident did the entire team seem in their answers to any questions?

The team were confident in almost every question asked.

* Comment on developer team’s process as seen from the perspective of the client.

The developer team were self-described “complacent” early in the project, which has created issues leading to the end of sprint 1. If these issues are not sorted early into sprint two, they could possibly lead to further issues leading to final product distribution. However the developer team seems confident in their ability to fix these issues and recover from early complacency. The team’s prioritisation seems to be in check and it looks like they will recover and produce a working product.

Shown sprint plan

Introduced teams

Database not working for demo

Pushing agents back to sprint 2

2-3 story points pushed back

Burndown accurate, well explained, clear and concise

Explanation of work discrepancies

Story cards:

Well noted in story cards

1. Incomplete
2. Completed
3. Pushed to sprint 2

13- implemented but unverified

Prototyping well documented, original

System architecture not shown,

Using remote access

Teamwork sufficient

Partial, Story 8, code is written but not implemented

Story 16 and 10 are written but not deployed

Rationale was based on common convention