



Bring in changes made to the Google Docs version.

Josiah authored 5 days ago

62c45428

rr\_instructions.md 13.2 KB

# Formatting Instructions for the Review and Retrospective Report

Revision 22w3

Dr. Yoder

Winter 2020-2021

(Based on Dr. Hasker's instructions)

## Sprint Review and Retrospective

The Sprint Review and Retrospective is a written summary of how much of your sprint goals were met (the "review" part), as well as how your team performed (the "retrospective" part) in achieving those goals.

Some of this material may duplicate your other planned deliverables. Your instructor may ask that material to be in your other documents but not this one.

Prepare this document carefully! Be sure this document makes it obvious what you and the team did and how the process will improve in the next sprint. Screen shots capturing your progress are critical. This document provides evidence of your accomplishments and is a major part of each sprint's grade.

Throughout this document, the term "prose" is used to indicate [ordinary speech or writing](#) as distinct from bulleted lists and table formats.

## Review

The review should come first in the report. I recommend writing the review and retro together as a single markdown document in your wiki.

In this section, review what functionality your team completed. Include screen shots of your application's functionality. Stories primarily represent real progress, but you should also include the other high-level PBIs (Defects, Knowledge Acquisitions, and Internal Improvements) you included in your sprint. You may also be required to demonstrate your progress to your instructor.

The review MUST have the following sections:

- Sprint Goal Recap
- PBI Status Report
- Plans for Next Sprint
- Screenshots
- Hours Worked
- Individual Contribution Reports

You are welcome to include additional sections, but must keep these sections in the order above.

## Sprint Goal

Restate the sprint goal. Do not mention individual PBIs in this section but rather focus on the themes that motivate the majority of the PBIs.

## PBI Status Report

This section of the report must include three bulleted lists:

- PBIs Planned and Completed:
- PBIs Planned and Not Completed:
- PBIs Not Planned and Completed:

Each item in these lists should be a single PBI. List the PBI number (#nn) and a brief description for each PBI. You could optionally use the titles for the brief descriptions. For example, you could write:

- PBIs Planned and Completed:
  - [#1](#) Create first stage rocket
  - [#2](#) Create second-stage rocket
  - [#4](#) Create capsule
- PBIs Planned and Not Completed:
  - [#3](#) Integrate first and second stage rockets
- PBIs Not Planned and Completed:
  - [#5](#) Plan moon orbit

If the PBIs Planned and Not Completed list is empty, you may replace it with a prose statement that all PBIs were complete. You may omit the PBIs Not Planned and Completed at all if this list is empty without mentioning it at all. You are welcome to include other lists in addition to these (such as PBIs Not Planned and Not Completed) if you wish.

You may give these lists different names, but it must be clear for every PBI whether it was on the original sprint plan and whether it was completed. The PBIs Not Planned and Completed list includes PBIs added mid-sprint that were completed before the end of the sprint. The PBIs Planned and Not Completed are PBIs that were included in the sprint plan but not finished. It is assumed these will be finished in a future sprint, depending on priority.

After the lists, explain in prose any differences vs. original goals; what were the underlying reasons or causes for the differences?

This is also a good place to discuss the existing state of the product.

## Future Directions

List the likely PBIs (with PBI numbers and descriptions) for the next sprint and a possible sprint goal. You may also wish to discuss tentative plans that go beyond the next sprint, such as your tentative release schedule for software.

### Plans for Sprint 2

The goal for next sprint is to begin the actual trip to the moon.

- [#7](#) Takeoff
- [#8](#) Orbit
- [#9](#) Land
- [#10](#) Moonwalk

## Screenshots

Screen shots capturing your progress are critical.

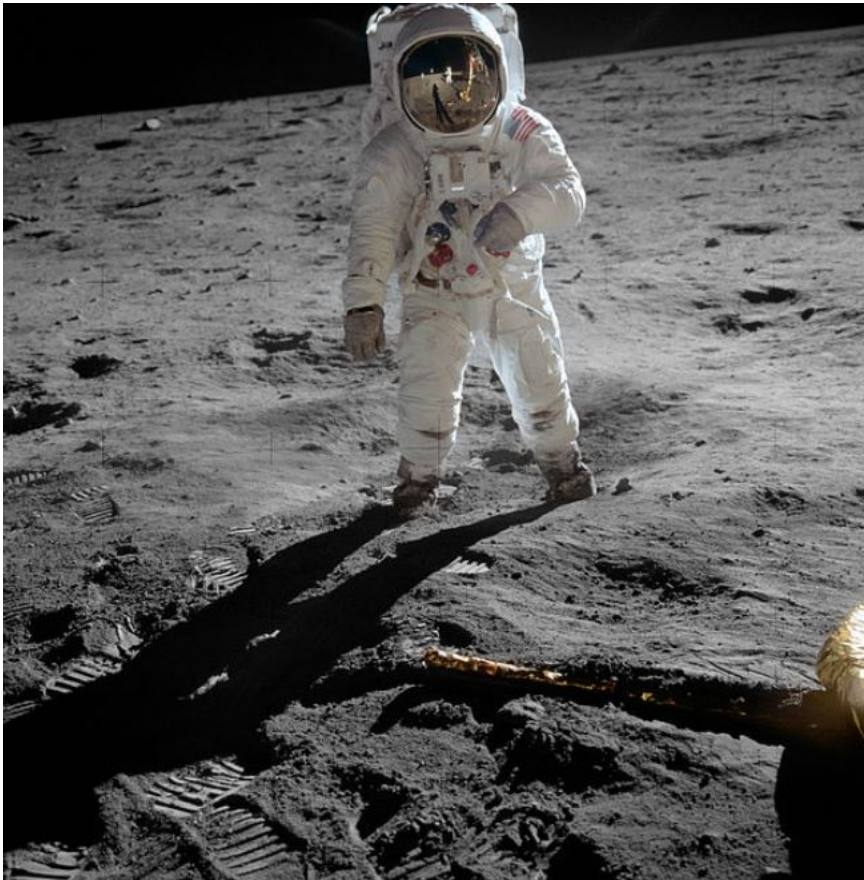
Give screen shots capturing evidence of completing PBIs. This evidence should be of your running software as experienced by the user. Include PBI numbers and descriptions as headings before each section of screenshots. You may optionally include team member's names in the title for each subsection as well. Some features require several screenshots to fully document.

Each screenshot should include a brief descriptive text describing what the screenshot illustrates. Capture full windows and make the screenshots as large as possible within the document so that the fonts in the screenshot are comparable to the main text font in the document. Canvas does not show the full resolution of the screenshot making interpretation of small text within the screenshot difficult even when zooming.

It can also help if you link to the original image from your image. This can allow the full-resolution version to be seen easily:

## Sprint 2 Review - Screenshots

### #10 Moonwalk



Buzz Aldrin demonstrating that we actually landed on the moon. The food was great, but the atmosphere was terrible.

You may optionally include screenshots of other important items such as working CI systems (including number of tests passing and not passing), user documentation, etc.

Note that primary material must be in the document. Do not link to other documents unless the link is just a footnote to something that is not important to your grade. This document is a key portion of the assessment for this course, and links are too volatile.

### Hours Worked

Give a table capturing the time spent by each team member on the sprint (to the closest tenth of an hour). This table must include at least three columns, one for ceremony hours, one for on-task hours, and one for the sum. It is also helpful to add a comments column.

Each team member is expected to enter their own information.

If hours differ significantly from what is reported by AboutTime, team members must include an explanation adjacent to this table explaining why the numbers differ from those reported by online tools like AboutTime. Team members may also wish to include an explanation at this point of why their hours were significantly below that of others.

### Individual Contribution Reports

Below the table, each team member is to summarize their contributions:

Number (#nn) and describe the major PBIs to which you contributed and describe for shared PBIs your unique contributions to the PBI. It is not necessary to include titles for PBIs in this section. Your contribution narrative itself may make it clear what the topic of the PBI is. For example, you may write:

I helped Buzz Aldrin adjust the nozzle for the first stage of the rocket (#1).

Even though the title of the PBI isn't included in this discussion, its topic is clear -- and your specific contributions to the creation of the first stage rocket would be clear as well.

It is expected that each team member will be able to document one or more completed PBIs on which they made code contributions. On shared PBIs it can be useful to list not only the PBI number (#nn) but also the merge request number (!nn) in which your contributions were merged into a shared code-base.

For incomplete PBIs, including both the PBI number (#nn) and the merge request number (!nn) could be helpful for partial credit reasons. However, please bear in mind the Scrum perspective that incomplete PBIs should not be considered partial progress for the team. One of the main purposes of the short Scrum cycle is to have real, frequent deadlines and PBIs which are only time-limited should be wrapped up by the end of the sprint so the team can count that progress.

Discuss your role in reviews, especially cases in which you identified issues that needed to be resolved before accepting the merge request. Include merge request numbers (!nn) in these cases.

State your team role (product owner proxy, ScrumMaster, note taker, devops lead, etc.) and document activities you have done in support of your role. Be sure to go beyond restating definitions! Give enough specifics so that I or a hiring manager could understand how capably you filled your role. Consider including such details as dates, circumstances involved, and outcomes.

If your accomplishment is a spike solution that will not be pulled into the product, note that the spike solution must have a report associated with it. Be sure to link that report into the sprint review document. This type of link is an exception to the above rule about not using links, but it is appropriate in this case. Include a title with the link in case the document is moved at a later date.

## Retrospective

---

The retrospective **MUST** have the following sections:

- Burndown chart(s)
- Progress and Velocity
- Process and Process Improvement

### Burndown chart(s)

---

Present your issue-count burndown chart in your report as the basis for the items that follow. Once your team starts using story-points, include the weighted burndown chart as well. The weighted chart does not replace the issue-counting burndown chart. In addition to these required charts, you may choose to include burnup charts as well.

Immediately after the burndown chart, describe any abnormalities in the chart (such as spikes or burn-cliffs) and what you think the sources of those abnormalities are. As of Winter 2020-2021, GitLab has a bug in which the burndown chart curves backwards one day at the end -- you may ignore this bug. If you think other GitLab bugs are influencing your chart, this is a good place to describe how the chart should look.

### Progress and Velocity

---

Briefly discuss your progress through the sprint. This will likely include information from weekly standups and other Scrum ceremonies, but should not be an exhaustive week-by-week summary. Focus on themes and trends in progress.

Compare your planned progress against your actual progress. This should start by comparing the target number of story points against the delivered number of story points, but could include discussions of process issues that slowed progress.

Analyze your team velocity based on story points completed. Starting in the second story-pointed sprint, this should number exactly how many story-points were completed for each of the last two sprints and list the average velocity. More sprints can be included if desired. Give and defend the story point target for the next sprint.

### Process and Process Improvement

---

Discuss the software quality activities and tasks your team practiced in this sprint (including design reviews, code reviews, unit tests, test plans).

Discuss improvements your team has made to your software development process, including any changes you made to reduce technical debt (such as automating your team's testing and build processes).

Early in the project, discuss the groundwork you are laying for good process. There is no need to mention that you don't have a process to improve yet.

Add additional sections as needed. You might consider discussing tasks which were either over or under estimated by large amounts, project pivot reports, cumulative flow diagrams, epic reports, control charts, etc.

If you engage in an activity (such as dot voting) as part of your retrospective, include notes from that activity and discuss the results.

Close with discussions of what went well, what could be improved, and process-related action items assigned to specific team members:

#### What did we do well?

List items you did well. Give some detail; for example, don't just say "estimated well" but briefly discuss how your process lead to good estimates.

#### What should we have done better?

List all the items you should have done better. For example, describe the sorts of tasks you had difficulty estimating or the stories that exhibited poor communication. Attempt to identify causes.

#### Actions

Identify process improvements, giving specifics about how you will address each. Ensure these are measurable. For example, set a goal of completing X% of tasks within 20% of the estimated time or moving a third of the story points to DONE each week. [@mention](#) team members with assigned process-improvement tasks.

## **Publishing the Review and Retro Report**

---

Include, either at the top of your markdown file, or on a separate cover page, the names of all the team members, the sprint code (e.g. 22w1), and a title (e.g. Sprint 2 Review and Retrospective).

One way to create a title page is to copy the markdown report into a Microsoft Word document or Google document and then add a title page to that document. The 2020-2021 tutoring team reports that sometimes images copy incorrectly.