

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. Consider including screen shots of other important results such as enabling CI. This document provides evidence of your accomplishments and is a major part of each sprint's grade.

Review

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.

1. Restate the sprint goal.
2. Discuss how much of your original sprint goal(s) was (were) achieved - that is, discuss which Stories (and other PBIs) you completed or didn't complete. Include any additional PBIs you added mid-sprint.
 - a. Discuss how many of the PBIs originally included in the sprint were finished.
 - b. Discuss which specific PBIs were included but not finished. It is assumed these will be finished in a future sprint, depending on priority.
 - c. Discuss any additional PBIs that your team included in the sprint.
3. Explain differences vs. original goals; what were the underlying reasons or causes for the differences?
4. Discuss the existing state of the product and future directions. Include listing the likely PBIs for the next sprint and a possible sprint goal.
5. Give screen shots capturing evidence of completing PBIs. This also gives the reader important context. Failing to include screen shots will reduce your score.
6. Include screen shots 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.

It is expected that each team member will be able to document one or more pull requests related to completed work. If a PBI cannot be completed for some reason, at least pull the completed portion into **dev** to minimize future merge conflicts and so you can get credit for your work in the given sprint. 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.

Effort

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, and failing to provide it will result in penalties. Team members may include an explanation of why their hours were significantly below that of others or why the numbers differ from those reported by online tools like AboutTime.

Either in the table or below it, each team member is to *summarize* their contributions:

- State major tasks, stories in which you took a significant role. Include pull requests in this discussion.
- Discuss your role in reviews, especially cases in which you identified issues that needed to be resolved before accepting the pull request.
- 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.

Retrospective

Discuss and analyze the project progress:

1. Present your story burndown chart in your report as the basis for the items that follow.
2. If your team uses the hourly burndown chart, include that as well. Note the hourly burndown chart does *not replace* the story burndown.
3. Discuss your progress through the sprint. This will likely include information from weekly standups and other Scrum ceremonies.
4. 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.

5. Analyze your team velocity based on story points completed. Give and defend the story point target for the next sprint.
6. Discuss the software quality activities and tasks your team practiced in this sprint (including design reviews, code reviews, unit tests, test plans).
7. 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).
8. 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.*