# Sprint 2 Feedback

Team Name:

Members:

|  |  |  |
| --- | --- | --- |
| **Sprint 2 Marking Items** | **Worth** | **Grade** |
| Team Usage of GitHub and Google Drive | 1 |  |
| Product Backlog Updates (max 12 stories) and Sprint Backlog | 1 |  |
| Project Team Tracking document – Sprint Planning | 1 |  |
| Project Team Tracking document – Daily Scrum | 1 |  |
| Project Team Tracking document – Retrospective | 1 |  |
| Functional Executable (\*.apk) | 1 |  |
| Demo | 3 |  |
| Total | 9 |  |

**Suggestion:** This is a software engineering course, not an application development course.   
The learning objectives are learning software engineering processes. Teams should pay more attention to the processes.

### Team Usage of GitHub and Google Drive

* + Documents are organized into sprint folders in Google Drive.
* - Documents should be organized into folders for long-term sustainability.
* - Wrong Google Drive permission and privacy configurations.
  + Allow people outside of the project to access files.
  + Allow people who are not members to access files
  + Violated data privacy agreement with the product owner.
* + All members updated sprint documents in Google Drive promptly and frequently.
* - Sporadic updates to sprint documents by members.
* - Daily Scrums were created in batches and not updated continuously.
* - Documents in Google Drive were updated only once.
* - File names should reflect their purposes.
* - The team did not follow the Scrum processes.
* + The team followed the "commit, push, pull-request, merge, and pull" GitHub team collaboration process during development.
* - The team did not follow the "commit, push, pull-request, merge, and pull" GitHub process of the course.
  + Branches were not allowed.
  + Merges without pull requests are not allowed.
  + Merging one's own pull requests is not allowed.
* + Numerous pull requests in the sprint reflecting continuous development.
* - Numerous pull requests just before the demo is a risky practice.
* - Sporadic pull requests, not frequently merging source code into the team repository.
* + All members contributed to the repository.
* - Uneven contributions between members.
* - Only some members have significant contributions to the repository.
* - A significant amount of code was merged and removed from the repository.  
  The team may have problems in the review process.

### Product Backlog Updates and Sprint Backlog

* - There should be a separate sprint backlog file for each sprint.   
  A single sprint backlog file for all sprints is not sustainable.
* + The product backlog has been updated after the previous sprint.
* + Included a maximum of 12 active stories in the product backlog.
* - There should be a maximum of 12 active stories in the product backlog.
* - All stories should describe functionalities supporting the product owner's business.
* - Stories can be added, updated, split, or deprecated, but not deleted from the product backlog.
* - Some non-functional requirements are actually functional requirements.
* - When using a template, remove the samples.
* - Need better column formatting to improve document readability.
* - All descriptions should wrap and align left in the cells.
* - Should not strikethrough finished stories, which makes the documents hard to read.
* - IDs should never be changed or reused.
* + All stories have names of a maximum of three words.
* - Names of stories should have a maximum of three words.
* + All stories have importance and estimation.
* - All stories must have importance and estimation.
* - Almost all stories have high importance and cannot be prioritized.
* - Need a definition for importance (what is high and what is low).
* + All stories have proper test cases.
* - Test cases are necessary for all stories.
* - Some test cases are not clearly defined. It is not clear how the functions work.
* + States of stories have been updated after the previous sprint.
* - Should update the states of the stories at the end of the sprint.
* - The state of a story should be empty until the story is included in a sprint.
* - The state of a story should include the corresponding sprint number.
* + Sprint backlog includes selected sprint stories.
* - Sprint backlog not provided.
* - Should update the sprint numbers in the headers of the sprint backlog, task boards, and burndown chart.
* + Created a task board for each story.
* + Stories are properly broken down into implementation tasks.
* - Missing task boards for some sprint stories.
* - When using a template, remove the samples.
* - Should not alter the metadata fields in a template.   
  There may be programs that retrieve data from documents created from a template.
* - The first row of each task board should include the story ID, name, and estimated man-hours.
* - The man-hours on the task boards do not match those in the sprint backlog.
* + Created a burndown chart for the sprint.
* - The sprint period on the task boards does not match the data table in the burndown chart.
* - The total number of days in the burndown chart does not match the sprint period.
* - The total story man-hours in the burndown chart does not match those in the task boards.
* - In the burndown chart, the first day of the sprint should be Day 1, not Day 0.
* - The data table in the burndown chart should be adjusted to the total number of days.
* - The completion dates on the task boards do not match the story points completed in the burndown chart data table.
* - Wrong burndown chart title.
* - The horizontal scale of the burndown chart does not show the entire sprint period.
* - Better scaling should be used for the burndown chart.
* - The burndown chart should be a line chart.
* - Google Drive dis-form the burndown chart. The team should check and fix the burndown chart before submission.

### Sprint Planning

* + Provided team and sprint planning information.
* - There should be a sprint tracking document for each sprint.
* - Should update the sprint number and the team information.
* - Should update the document header.
* - Should refresh the table of contents.
* - The sprint period does not match the period on the task boards.
* - Members' committed days should not exceed the sprint's total work days.
* + Calculated the total available man-hours correctly.
* - Incorrect total available man-hours calculation.
* - Calculated total available man-hours as <#>, but selected total story man-hours as <#>, with a significant gap without explanation.
* + Provided sprint goal.
* - Sprint goal should be a single sentence describing the sprint's high-level business focuses.
* - Technical description (e.g., database) should not be part of the sprint goal.
* - Should include the sprint backlog, not the product backlog.
* - Should include the sprint backlog, not the task boards.
* + Provided Daily Scrum schedule.
* - Should schedule at least eight Daily Scrums.
* - Daily Scrums should have more specific dates, times, and locations.

### Daily Scrum

* + Each member reported at least two Daily Scrums.
* - Should use "black" font for document content.  
  Other font colours should be used as special indicators.
* - From the document history, most Daily Scrums were reported in batches.  
  The team did not practice Daily Scrum regularly.
* - When the task boards or the burndown chart did not change from the previous Daily Scrum, document it as "same as the previous Daily Scrum."
* - The first Daily Scrum should include the task boards and burndown chart, so that the rest can be documented as "Same as the previous Daily Scrum."
* - Should include all sprint task boards, not just those with changes.
* - Should include the data table of the burndown chart.
* + Good Daily Scrum summaries.
* - Some Daily Scrums notes are taken without connection to the sprint's progress.
* - Some Daily Scrums notes have been taken but insufficient to help the team move forward.
* - Minimal Daily Scrums notes were taken. The document does not serve its purpose to acknowledge other parties about the progress.
* - Members have delegated specific man-hours to the sprint during the sprint planning.   
  If members cannot fulfill their commitments, clearly state the fact, the reason, and their new commitments; don't document personal excuses. Then, document how the team will adopt.
* - Product owners do not accept failure to keep commitments as an excuse for delay.
* - The team should document project obstacles (e.g., unexpected database connection issues), not personnel issues.
* - When obstacles are documented, corresponding actions to resolve the problems should be decided and documented (e.g., Member A will work with Member B to resolve the connection issues).
* - A sprint is not on target when the team believes they cannot finish all selected stories of the sprint.
* - The team should always be aware of whether the sprint is on target and make rectifications promptly.
* - When a sprint is not on target, the team needs to discuss and devise corrective actions to put the sprint back on target. The decisions should be documented in the corresponding Daily Scrum.
* - If the team has not been on target since the first Daily Scrum, then the team did not plan the sprint properly or did not understand Scrum.
* - When stories are unfinished, the causes should be discussed and recorded.

### Retrospective

* + Reflected on the sprint.
* - Minimal reflection on the sprint.
* + Calculated focus factor for the next sprint.
* - Did not show how the focus factor is calculated.
* - Wrong focus factor calculated.
* + Good dot voting exercise.
* + Done dot voting exercise.
* - Wrong usage of the dot voting exercise. Dot voting is a team-building exercise.
  + The team should suggest team-improving actions, not software-improving actions.
  + All actions should start with verbs.
* + Had tried to incorporate some dot-voting actions from the previous sprint.
* - No evidence shows that the team incorporated dot-voting actions from the previous sprint.
* - The team should note what will happen to the unfinished stories.

### Functional Executable (\*.apk)

* + APK is installable.
* Please submit a non-debugging APK for the final deliverable.  
  Otherwise, the performance may be affected.
* - APK is not installable.
* + Functions executable
* - Functions not executable.
* + Data persisted.
* - Data not persisted.

### Demo

* *<Refer to the Demo Marking Schema>*