Sprint 1 Materials

10B

Shreyas Kumar
Aviral Agarwal
Jack Payne
Ishaan Nigam
Dev Patel

[Delete all instructions in red before submitting.]

Retrospective Summary

Summarize your retrospective meeting. Discuss what went right and wrong during the sprint, what changes (if any) need to be made regarding procedure, and what adjustments might need to be made to the product backlog. If backlog goals were not met, include an explanation of why not. The summary should be approximately 250 words.

GitHub Release Link

Create a GitHub release of **a completely working version** of the software and include a link to your release here. Note that some features may not be included if they are scheduled for later sprints, but you must have a **minimum viable product (MVP)** that a user can give feedback on.

Remember to commit to your repository each time a new feature is added/modified. Items should not be marked as "complete" on your backlog until they have been pushed to your repository.

Product Backlog

Your product backlog contains the list of tasks with their:

- priorities,
- dependencies,
- user story points (an estimate of effort involved from the unitless set {1, 2, 3, 5, 8, 13, 21}), and
- status {not started, in progress, completed}.

It should be updated continually throughout the project. Include the snapshot of the product backlog at the end of the sprint here.

Sprint Backlog

Your sprint backlog is a subset of the product backlog. It includes the list of tasks that were scheduled for the sprint. For each task, it provides:

- user story point value (from the product backlog),
- who assigned to,
- actual time spent, and
- status {not started, in progress, completed}.

It should also contain a **summary table** that shows the total number of completed user story points for each team member. The summary table should have a column for each sprint completed to date. These will be a factor in your individual contribution assessment.

Burn-down Charts

Include two burn-down charts, one for the sprint and one for the product.

- Burn-down charts should be reported in **remaining user story points** (not number of tasks) and show both the user story points remaining and actual hours spent over time. Remember that user story points measure the *relative* estimated effort. They correlate to time spent (but are not equivalent to actual hours/specific units of time.)
- The initial sprint burn-down chart is created from the sprint backlog. It should be updated after each SCRUM meeting based on team member status and feedback. Thus it should contain at least 5 data points (the start of the sprint, 3 SCRUM meetings, and the end of the sprint).
- The product burn-down chart should be updated once during the sprint and again at the end of the sprint. Thus, by the end of the project's 3 sprints, there should be at least 7 data points on the product burn-down chart (the start of each sprint, the middle of each sprint, and the end of each sprint).

Next Sprint's SCRUM Meeting Schedule

Include a list of your planned SCRUM meetings for the next sprint. You should have at least 3 SCRUM meetings in each sprint. (This section does not apply for the final sprint.)

Appendix 1: SCRUM Meeting Agendas and Minutes

SCRUM Meeting 1 for Project 3
Prepared by: Ishaan Nigam
Meeting Date: 10/30/2023

Meeting Attendees

1. Ishaan Nigam
2. Shreyas Kumar
3. Dev Patel
4. Jack Payne

Meeting Agenda Items

- Go over frameworks for frontend and backend
- Plan out how to work during the week

Status Update Since Last Meeting

Accomplishments:

- Confirmed frameworks for frontend and backend
- Have a flexible plan for how the week work schedule will be

Tasks Completed:

Task Description	**Assigned to**	**Completed? (yes/no)**
Confirm framework languages	Entire Group	Yes
Plan the week's schedule	Entire Group	Yes
Take notes of meeting	Ishaan	Yes

Before The Next Meeting

Plans:

- Have a status check on where everyone is
- Address any new problems or concerns that have risen
- Test current code in place

Task Assignments:

Task Description	**Assigned to**
Connect Frontend code to database using Node.js	Shreyas
Begin Implementing layout construction for customer side of application	Dev and Jack
Upload meeting notes to github	Ishaan

Minutes from Previous Meeting

Summarize discussion in paragraph form from the previous meeting (NOT this current meeting).

In the previous meeting, the team discussed and solidified their choice of frameworks for the project. We've decided to use Node.js for the backend and Java React for the frontend. We found Node.js is a good framework. It will be utilized to connect the frontend code to the database and deal with API tasks. Additionally, the team collectively devised a flexible plan for the week's work schedule, and we've allocated tasks to our team members to kickstart the project. These decisions were made with the goal of ensuring a seamless project progression and ensuring that team members are aligned in terms of the chosen technologies and workflow.

SCRUM Meeting 2 for Project 3
Prepared by:
Meeting Date:

Meeting Attendees

5.
6.
7.
8.

Meeting Agenda Items

-
-

Status Update Since Last Meeting

Accomplishments:

-
-

Tasks Completed:

Task Description	**Assigned to**	**Completed? (yes/no)**

Before The Next Meeting

Plans:

-
-

Task Assignments:

Task Description	**Assigned to**

Minutes from Previous Meeting

Summarize discussion in paragraph form from the previous meeting (NOT this current meeting).

SCRUM Meeting 3 for Project 3
Prepared by:
Meeting Date:

Meeting Attendees

9.
10.
11.
12.

Meeting Agenda Items

-
-

Status Update Since Last Meeting

Accomplishments:

-
-

Tasks Completed:

Task Description	**Assigned to**	**Completed? (yes/no)**

Before The Next Meeting

Plans:

-
-

Task Assignments:

Task Description	**Assigned to**

Minutes from Previous Meeting

Summarize discussion in paragraph form from the previous meeting (NOT this current meeting).

Include additional SCRUM Agendas and Minutes on new pages if necessary (if you have more than 3 SCRUM meetings in a sprint).