

Team Contributions: Rev 0

Software Engineering

Team 4, EvENGage
Virochaan Ravichandran Gowri
Omar Al-Asfar
Rayyan Suhail
Ibrahim Quraishi
Mohammad Mahdi Mahboob

This document summarizes the contributions of each team member for the Rev 0 Demo. The time period of interest is the time between the PoC demo and the Rev 0 demo; the contributions prior to the PoC are NOT included.

1 Demo Plans

2 Team Meeting Attendance

Student	Meetings
Total	8
Ibrahim Quraishi	8
Virochaan Ravichandran Gowri	8
Mohammad Mahdi Mahboob	8
Rayyan Suhail	8
Omar Al-Asfar	8

3 Supervisor/Stakeholder Meeting Attendance

Supervisor's Name: Luke Schuurman

Student	Meetings
Total	4
Ibrahim Quraishi	4
Virochaan Ravichandran Gowri	4
Mohammad Mahdi Mahboob	4
Rayyan Suhail	4
Omar Al-Asfar	4

4 Lecture Attendance

Student	Lectures
Total	9
Ibrahim Quraishi	7
Virochaan Ravichandran Gowri	3
Mohammad Mahdi Mahboob	3
Rayyan Suhail	3
Omar Al-Asfar	3

[If needed, an explanation for the lecture attendance can be provided here. —SS]

5 TA Document Discussion Attendance

TA's Name: Tiago de Moraes Machado

Student	Lectures
Total	3
Ibrahim Quraishi	3
Virochaan Ravichandran Gowri	3
Mohammad Mahdi Mahboob	3
Rayyan Suhail	3
Omar Al-Asfar	3

The final TA meeting for the Design Documentation was cancelled due to a snow day.

6 Commits

Student	Commits	Percent
Total	180	100%
Ibrahim Quraishi	48	26.7%
Virochaan Ravichandran Gowri	58	32.2%
Rayyan Suhail	13	7.2%
Mohammad Mahdi Mahboob	28	15.6%
Omar Al-Asfar	32	17.8%

While there are visible differences in contribution percentages, there is a similar pattern to the POC Team Contribution report. The differences can be attributed to the same reasoning as before, with members committing at different frequencies. For instance, Virochaan and Ibrahim tend to commit more frequently than the rest. Omar ranks 3rd in commit count, but contributed the highest in terms of lines of code, with Mohammad following a similar pattern. The line contributions can also be misleading since Virochaan and Omar were responsible for app setup and integration of provided templates, resulting in a higher number of lines than the rest. Rayyan focused more on testing and developed components with less volume, but still essential to the project. Overall, while commit counts vary, all members are comfortable with the workload distribution.

7 Issue Tracker

[For each team member how many issues have they authored (including open and closed issues (O+C)) and how many have they been assigned (only counting closed issues (C only)) over the time period of interest. —SS]

Student	Authored (O+C)	Assigned (C only)
Name 1	Num	Num
Name 2	Num	Num
Name 3	Num	Num
Name 4	Num	Num
Name 5	Num	Num

[If needed, an explanation for the counts can be provided here. —SS]

8 CICD

CI/CD is still intended to be implemented in a later phase of the project. It's primary application remains to be to support regression testing, with the pipeline accomodating verified unit tests to ensure functionality over time. It will also be used to ensure compliance with Airbnb JavaScript and JSX guidelines. Overall, the team has made progress in setting up the necessary infrastructure and is actively working towards the deployment of the CI/CD pipeline.

9 Team Charter Trigger Items

The team charter triggers are defined in the Development Plan, aiming to assess team performance. The triggers are as follows:

- **Attendance:** All members are expected to attend meetings punctually for the entire agreed upon duration. Repeated unexcused absences constitute a trigger violation.
- **Communication:** All members are expected to respond to team communications within 24 hours. Failure to respond within this timeframe is considered a trigger violation. Moreover, failure to communicate scheduling difficulties or absences in advance is also considered a communication violation.
- **Quality:** All members are expected to complete their assigned tasks to an agreed upon standard. Repeated and blatant lack of effort or quality in completed tasks is considered a trigger violation. Members are expected to seek help from the team if they are struggling to complete their tasks to the expected standard. Failure to do so reflects on the entire team and is therefore a quality violation.
- **Attitude:** All members are expected to maintain a respectful and open-minded attitude towards each other. A good attitude facilitates effective teamwork and communication. Repeated negative or disrespectful behaviour is considered a trigger violation. Differences of opinion and conflict are not excusable under this trigger. All members are expected to address any issues maturely and professionally based on the team charter guidelines.
- **Contribution:** All members are expected to contribute equitably to the workload. Members are expected to proactively seek out tasks and responsibilities to support the team. Failure to achieve agreed upon performance metrics is a trigger violation. These expectations differ depending on the task, and may include number of commits, software features, lines of code, or even idea generation. The team will agree upon these metrics and expectations at the start of each phase.

All members agree that expectations have been met and no trigger violations have occurred thus far. We regularly communicate to ensure universal satisfaction with workload distribution, task quality, and the general direction of the project. Overall, the team is functioning well and there are no concerns regarding team performance at this time.

10 Additional Productivity Metrics

The additional productivity metrics we considered previously were pull requests and task completion time. The latter was deemed an ineffective measure due to the varying complexity of tasks, which made it difficult to draw meaningful conclusions. The pull request contributions since the POC are as follows:

Student	Pull Requests	Percent
Total	22	100%
Ibrahim Quraishi	5	22.7%
Virochaan Ravichandran Gowri	4	18.2%
Rayyan Suhail	3	13.6%
Mohammad Mahdi Mahboob	4	18.2%
Omar Al-Asfar	6	27.3%

We can see that this metric demonstrates a far more balanced contribution percentage compared to commits and issues. Pull requests are a more broad action, generally created per major task, and consisting of multiple commits and issues. This makes it a more reliable indicator of team contributions. There are still discrepancies due to varying task complexities, but it is more resistant to inflation than the other metrics.

In the future, it may be beneficial to separately track coding and documentation pull requests to more fairly assess contributions. Coding tasks typically require more effort and time, so they should be valued more heavily than documentation tasks.