Team Contributions: POC
Software Engineering

Team #12, Streamliners
Mahad Ahmed
Abyan Jaigirdar
Prerna Prabhu
Farhan Rahman
Ali Zia

This document summarizes the contributions of each team member up to the POC Demo. The time period of interest is the time between the beginning of the term and the POC demo.

1 Demo Plans

[What will you be demonstrating —SS]

2 Team Meeting Attendance

Student	**Meetings**
Total	7
Mahad Ahmed	7
Abyan Jaigirdar	7
Prerna Prabhu	7
Farhan Rahman	7
Ali Zia	7

Our team meets weekly at the start of each deliverable, with additional meetings scheduled as needed, typically in the days leading up to major deadlines. Since the team was formed during Week 3, we have held a total of seven meetings to date, with the exception of reading week. Every team member has been able to attend these meetings, even if occasionally arriving slightly late or needing to leave early.

3 Supervisor/Stakeholder Meeting Attendance

[For each team member how many supervisor/stakeholder team meetings have they attended over the time period of interest. This number should be determined from the supervisor meeting issues in the team's repo. The first entry in the table should be the total number of supervisor and team meetings held by the team. If there is no supervisor, there will usually be meetings with stakeholders (potential users) that can serve a similar purpose. —SS]
Supervisor's Name: [fill in this information]

Student	**Meetings**
Total	Num
Name 1	Num
Name 2	Num
Name 3	Num
Name 4	Num
Name 5	Num

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

4 Lecture Attendance

[For each team member how many lectures have they attended over the time period of interest. This number should be determined from the lecture issues in the team's repo. You can find the number of lectures in the time period of interest by looking at the Google calendar for the capstone course. —SS]

[NOTE: There will be approximately 13 lectures between the start of class and the POC demos —SS]

Student	**Lectures**
Total	Num
Name 1	Num
Name 2	Num
Name 3	Num
Name 4	Num
Name 5	Num

[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
Ali Zia	2
Abyan Jaigirdar	2
Mahad Ahmed	2
Farhan Rahman	2
Prerna Prabhu	2

There were approximately three meetings held with the TA during this period. Our team attended two of these meetings together. By group consensus, we decided not to attend the most recent session due to a midterm scheduled that evening, as we prioritized the additional study and commute time required on that day.

6 Commits

[For each team member how many commits to the main branch have been made over the time period of interest. The total is the total number of commits for the entire team since the beginning of the term. The percentage is the percentage of the total commits made by each team member. —SS]

Student	Commits	Percent
Total	Num	100%
Name 1	Num	%
Name 2	Num	%
Name 3	Num	%
Name 4	Num	%
Name 5	Num	%

[If needed, an explanation for the counts can be provided here. For instance, if a team member has more commits to unmerged branches, these numbers can be provided here. If multiple people contribute to a commit, git allows for multi-author commits. —SS]

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

Continuous Integration and Continuous Deployment (CI/CD) will be used to keep the project stable and easy to maintain as new features are added. The team will use GitHub Actions to automatically build, test, and deploy the system.

For **Continuous Integration (CI)**, every time a pull request is opened, the pipeline will automatically run linting, type checks, and unit tests. This helps catch issues early and ensures that the codebase stays clean and consistent. It will also build the project to make sure that new changes do not break existing functionality before they are merged.

For **Continuous Deployment (CD)**, after merging into the `dev` branch, a staging build will be automatically deployed for testing and feedback. When changes are merged into the `main` branch, a production build will be deployed to the university-hosted environment. This keeps updates smooth and reduces the chance of deployment errors.

GitHub Actions will also send build or test failure notifications to the team so that problems can be fixed quickly. Environment variables and API keys will be stored securely using GitHub's built-in secret management. Over time, the team may expand the pipeline to include integration or end-to-end tests, but for now the main goal is to automate testing and deployment to save time and improve reliability.

9 Team Charter Trigger Items

[Provide a summary of the quantified triggers identified in the team's charter. —SS]

[Provide a list of any violations of the triggers. If the team wishes, the violations can be summarized on aggregate, instead of naming specific team members. —SS]

[Provide a plan to address the violations. This could include revising the triggers, if they are found to be too weak, strong or ambiguous. —SS]

10 Additional Productivity Metrics

[If your team has additional metrics of productivity, please feel free to add them to this report. —SS]