Team Contributions: POC
Software Engineering

Team 2, SyntaxSentinals
Mohammad Mohsin Khan
Lucas Chen
Dennis Fong
Julian Cecchini
Luigi Quattrociocchi

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

[For each team member how many team meetings have they attended over the time period of interest. This number should be determined from the meeting issues in the team's repo. The first entry in the table should be the total number of team meetings held by the team. —SS]

Student	**Meetings**
Total	Num
Mohammad Mohsin Khan	Num
Lucas Chen	Num
Dennis Fong	Num
Julian Cecchini	Num
Luigi Quattrociocchi	Num

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

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]

Student	Meetings
Total	Num
Mohammad Mohsin Khan	Num
Lucas Chen	Num
Dennis Fong	Num
Julian Cecchini	Num
Luigi Quattrociocchi	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. The first entry in the table should be the total number of lectures since the beginning of the term. —SS]

Student	Lectures
Total	Num
Mohammad Mohsin Khan	Num
Lucas Chen	Num
Dennis Fong	Num
Julian Cecchini	Num
Luigi Quattrociocchi	Num

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

5 TA Document Discussion Attendance

[For each team member how many of the informal document discussion meetings with the TA were attended over the time period of interest. —SS]

Student	Lectures
Total	Num
Mohammad Mohsin Khan	Num
Lucas Chen	Num
Dennis Fong	Num
Julian Cecchini	Num
Luigi Quattrociocchi	Num

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

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%
Mohammad Mohsin Khan	Num	%
Lucas Chen	Num	%
Dennis Fong	Num	%
Julian Cecchini	Num	%
Luigi Quattrociocchi	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)
Mohammad Mohsin Khan	Num	Num
Lucas Chen	Num	Num
Dennis Fong	Num	Num
Julian Cecchini	Num	Num
Luigi Quattrociocchi	Num	Num

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

8 CICD

The project will use continuous integration and continuous deployment (CICD) to run tests and deploy the software. The steps in the CICD pipeline are as follows:

1. Developer creates PR and CICD pipeline will trigger phase 0.
2. Phase 0 will run build and check if the build is successful.
3. If the build is successful, reviewer will review the PR.
4. If the reviewer approves the PR, the CICD pipeline will trigger phase 1.
5. Phase 1 will run tests and check if the tests are successful.
6. If the tests are successful, the CICD pipeline will trigger phase 2.
7. Phase 2 will merge and deploy the software.