Team Charter & Working Agreement

Team Members & Roles

Describe each member’s role, primary responsibilities, backups/rotation plan, and contact info.

Trinity Paulson:

- Role: Project manager
- Primary Responsibilities: organize communication and responsibilities.
- Rotation plan: Rotate based on the needs of the team.
- Contact info: paulsotr@oregonstate.edu

Hsun-Yu Kuo:

- Role: Meeting Coordinator / Developer
- Primary Responsibilities: Help schedule team meetings. Testing the code to ensure functionality. And will help develop project features.
- Rotation plan: Depending on team needs. May or may not rotate.
- Contact info: kuohsu@oregonstate.edu

Rithik Reddy:

- Role: Head of testing / Developer
- Primary Responsibilities: Work on project development and write test cases to ensure the code functions correctly and meets quality standards.
- Rotation plan: May rotate every 2 months or based on the team's needs.
- Contact info: nibbarar@oregonstate.edu

Norman O’Brien:

- Role: Documentation lead / head of architecture / tester
- Primary Responsibilities: Manage proper use of documentation and ensure project design specifications are met as well as second line of testing after Rithik
- Rotation plan: Open to rotating depending on the needs of the team
- Contact info: obrienno@oregonstate.edu, +12069724350

Decision-Making Model

Our team makes decisions by majority vote after open discussion during meetings or in our group chat. A quorum is reached when at least three out of four members are present. If there is a tie, our team will discuss further to reach consensus, or the project partner will make the final call if needed. And the technical direction will be reviewed with our instructor or project partner before implementation to ensure alignment with project expectations.

Meeting Cadence & Tools

Weekly Team Meeting: Sundays 4:00–6:00 PM PT via Discord, to review weekly accomplishments, identify blockers, and outline next steps for the following week.

Mentor Check-ins: Every Friday at 3:00 PM via Zoom to update the stakeholder on weekly progress, discuss future plans, and confirm approval of recent changes or gather feedback on new requirements.

Async Updates: Discord posts short daily progress updates.

Response Expectation: 6h for messages, 24h for pull-request reviews.

Core Tools:

- GitHub (repo, issues, Actions CI/CD)
- Discord (team communication)
- Google Docs / Drive (draft documents)
- Github Projects (sprint tracking & task board)

Risk Management & Escalation Path

Identify top risks, early warning indicators, and who escalates to whom with timelines and evidence to provide.

Risk	Early Warning Sign	Escalation Path	Timeline	Evidence
Missing deadlines	Incomplete tasks or delayed PRs before the sprint review	Private reminder →Discuss in team meeting → Escalate to mentor if repeated	Address within 3 days of the missed due date	GitHub issues, commit timestamps
Communication issues	No message replies or absence from meetings for more than 48 hours	Direct message → Team discussion → Escalate to instructor if no response after 1 week	Check in within 2–3 days	Discord chat logs, attendance records
Repeated CI failures	Consecutive failed	Developer	Within 2 days of	GitHub Actions logs

	runs in GitHub Actions on the main branch	investigates → Team debugging → Escalate to TA if persists after multiple failed runs	the first failure	
Code failure	Tests fail or integration breaks after the merge	Assign debugging pair → Review during next team meeting → Whole team tries to fix code → Escalate to TA /Stakeholder if unresolved	Within 1 day	Test results, PR review comments
Data issues	WRDS filings missing, incomplete, or API fetch errors	Notify team → Stakeholder → OSU WRDS manager	Immediately upon detection	Error logs, dataset validation script
AI API issues	Timeout errors, invalid responses, or API quota exceeded	Retry and debug → Report to TA/Stakeholder→ look for an alternate model or increase in funding	Same day as the issue	API logs, error reports

Conflict Resolution & Accountability

Define triggers, stepwise restorative actions, timelines, and how objective evidence (PRs, reviews, attendance) will be used.

Triggers:

- Missed deadlines or uncompleted assigned tasks
- Lack of communication or attendance at scheduled meetings
- Untested code merged without review
- Repeated unresponsiveness to team updates or feedback

Stepwise Restorative Actions:

- First instance: Private check-in between the affected members and team to clarify issues and reset expectations.
- Second instance: Discussion in a team meeting to redistribute workload and establish concrete action steps.
- Third instance: Formal documentation of the concern and escalation to the instructor or project partner if behavior continues.

Timelines:

- Each issue will be addressed within 3 days of occurrence.
- Progress and follow-up actions will be reviewed in the next weekly meeting.
- If unresolved after 1 week, responsibilities will be reassigned.

Objective Evidence Used:

- Pull Requests (PRs): Track code contributions and review for testing.
- Code Commits: Show engagement with team development and quality assurance.
- Meeting Logs: Record our meetings and attendance, and see if we maintain consistency in communication.

Definition of Done (DoD) & Quality Gates

Enumerate required checks before merge (tests, code review, security/static analysis, docs updates) and name the enforcing CI jobs.

- **Code Review:** Code must be reviewed by at least one other team member before being merged to the main branch: feature/* → PR → ≥ 1 review → merge
- **Testing & CI:** Code must pass all automated tests and continuous integration (CI) checks before being merged into main.
- **Documentation Updates:** Team design documents must be updated to reflect any structural or functional changes.
  If new functionality is added, the GitHub README must also be updated with setup or usage instructions.
- **Enforcing CI jobs:** These requirements are validated through the CI workflow main_build.yml to ensure consistent testing and documentation standards.

Accessibility & Inclusion Practices

Specify meeting norms (time zones, turn-taking, note-taking), accommodation process, and how barriers will be surfaced/addressed.

- Team members will come prepared to meetings by reviewing the material to be discussed prior to the meetings.
- All meetings are scheduled in PST time.
- Trinity will take notes to be sent out after the meeting with the team's tasks and meeting highlights.
- During stand-up meetings, each team member will go around and discuss what they have done and any blockers. Then we will decide how to proceed with the task by assigning any assistance needed.

- If someone is unable to complete a task on time, they should communicate with the team and we will all work to accommodate them.
- Team members can privately request accessibility or scheduling accommodations from the project manager, and the team will adapt meeting practices as needed to accommodate all members.

Policy Owners & Review Dates

Section	Owner	Review Date	Update Process
Roles & Decision Model	Norman	Nov 9 2025	Propose edit via PR; vote required
Decision-Making Model	Hsun-Yu	Nov 9 2025	Propose edit via PR; discuss in meeting, and approve by majority vote
Meeting Cadence & Tools	Hsun-Yu	Nov 9 2025	Review with the team during sprint meetings and vote on changes if required.
Risk Management	Rithik	Nov 9 2025	If a teammate identifies a potential risk, they should post a message on Discord for the team to review and update the risk log accordingly if the concern is deemed valid.
Conflict Resolution	Rithik	Dec 1 2025	Review after Sprint reflection; update if new issues or escalation steps are identified
Definition of Done	Trinity	Dec 1 2025	Update as CI workflows, testing standards, or documentation requirements evolve

Accessibility	Trinity	Dec 1 2025	Review after stakeholder feedback or team accessibility check-ins

Links & Artifacts

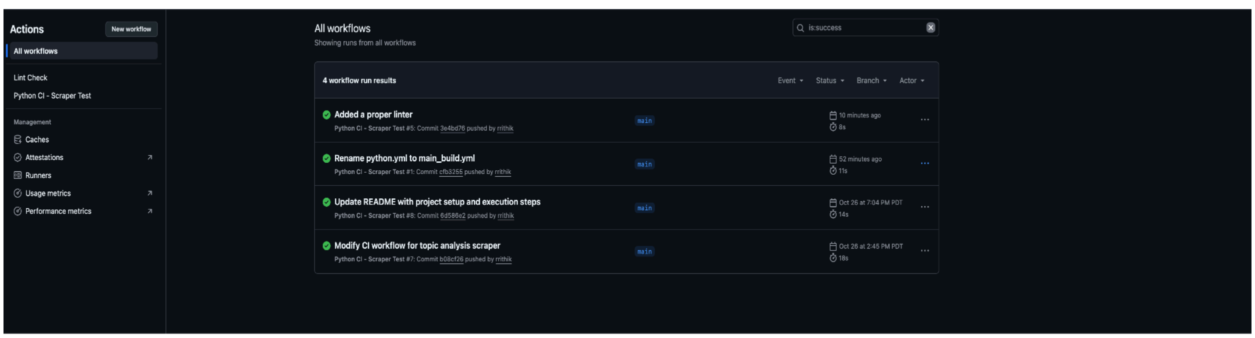
Link to meeting agenda/notes template, CI dashboard, linter/formatter config, and any external docs referenced.

Github Project board: https://github.com/users/rrithik/projects/4/views/1

CI Dashboard: https://github.com/rrithik/Natural-Language-Processing-of-Financial-Disclosures/actions

Linter: https://github.com/rrithik/Natural-Language-Processing-of-Financial-Disclosures/actions/workflows/linter-check.yml

Success:



Evidence

Rithik: Contributed to the Risk Management section, portions of the Definition of Done, and the Policy Owners table; participated in team meetings to finalize risk and quality guidelines.

Norman: Contributed largely to the "contributing.md" file; participated in team meetings.

Trinity: Contributed to Team Members and roles, Meeting Cadence & Tools, Definition of Done (DoD) & Quality Gates, and Accessibility & Inclusion Practices.

Hsun-Yu: Contributed to the Decision-Making Model, the Meeting Cadence, Conflict Resolution, and the Policy Owners table; participated in team meetings to finalize guidelines.

Changes by

H Hsun-Yu Kuo

T Trinity Paulson

Rithik Reddy Nibbaragandlla

N Norman William O'Brien

Version history

All versions

Today

November 2, 9:30 PM

Current version

- Trinity Paulson
- Rithik Reddy Nibbaragandlla
- Hsun-Yu Kuo

November 2, 8:06 PM

- Trinity Paulson
- Rithik Reddy Nibbaragandlla
- Hsun-Yu Kuo
- Norman William O'Brien

November 2, 5:56 PM

- Rithik Reddy Nibbaragandlla
- Trinity Paulson
- Hsun-Yu Kuo

November 2, 5:01 PM

- Rithik Reddy Nibbaragandlla