Team and Project Plan

Portfolio task 1

Unit code: COS40005

Unit Name: Computing Technology Project A

Submission date: 24/08/2025

|  |  |  |
| --- | --- | --- |
| Student Name | Student Id | Description of contribution in team and project planning |
| Connor Lack | 103992223 | Teamwork Roadmap  Document Management  Risk Mitigation  Problem Statement  Scope |
| Jen Mao | 103821194 | Quality Plan |
| Jason Vo | 103993653 | Solution Approach |
| Kavindu Bhanuka Weragoda | 104860525 | Stakeholder  High-level requirements |
| Md Hridoy Mia | 105077229 | Product Backlog |

## Acknowledgment of Country

Each team member identifies:

the [Traditional Owners](https://aiatsis.gov.au/explore/map-indigenous-australia) of the land they lived on while completing this work (if living in Australia).

We are honoured to recognise our connection to Wurundjeri Country, history, culture and spirituality through these locations, and strive to ensure that we operate in a manner that respects and honours the Elders and Ancestors of these lands.

We also respectfully acknowledge Swinburne’s Aboriginal and Torres Strait Islander staff, students, alumni, partners and visitors.

We also acknowledge and respect the Traditional Owners of lands across Australia, their Elders, Ancestors, cultures and heritage, and recognise the continuing sovereignties of all Aboriginal and Torres Strait Islander Nations.

## PART 1 Team Code of Conduct

## Team profile

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Student name | Technical skills | Soft skills | Communication | Teamwork |
| Connor Lack | Adequate in Python and C#. Experience with machine learning in relation to cleaning and using datasets. | Organized, Experience with sprints and project planning. | More introverted but will speak up when necessary. English first language. Comfortable with text or speech. | Usually the listener, I take into consideration all team-members opinions and ideas before responding and formulating a plan. |
| Jen Mao | Proficient in Python and C#. Experienced in front-end (HTML, CSS, JavaScript). | Ability to solve problems, adaptive, open to new learning, good at time management. | Kinda introverted but will try to communicate. Can speak English. Able to communicate online or in person if informed in advance. | Good at listening, responsive in group chats, collaborative, supportive. |
| Jason Vo | Sufficient in Python, C++, HTML, CSS, PHP. Some experience with Java, Node.js | Eagerness to learn, collaborating within a team, communicating authentic thoughts to the team | Able to communicate online or in person, may have long response time during working hours | Within a team, I will proactively listen to and respect my colleagues as well as express any thoughts and ideas I may have. I will try my best to meet deadlines and balance the team's needs with outside commitments. |
| Kavindu Bhanuka Weragoda | Good knowledge on c++. Skilled in HTML,CSS, and Java (not professional level).Fairly good at of python | Reliable,able to manage time effectively, and  committed to completing tasks within  deadlines. | English is not first language but able to  communicate and write well in academic/  professional contexts. | Experienced working in teams; cooperative  and adaptable with others, even while  balancing other commitments. |
| Md Hridoy Mia | Good knowledge on c++. Skilled in HTML,CSS,PHP and Java. | Prioritizing tasks effectively.  Meeting deadlines consistently. Balancing multiple responsibilities. | English is not my first but I am capable of effectively communicating and writing . | I believe teamwork is one of my strongest soft skills. I work well with others by listening to their ideas, respecting different perspectives, and contributing my own thoughts to reach a common goal. |

## Team role

|  |  |  |
| --- | --- | --- |
| Student name | Team role | Justification |
| Connor Lack | Team Leader  Research  Back End | I’m comfortable with organizing tasks and making sure deadlines are met, able to make sure that everyone has an opinion during decision making. I’m also happy to work on developing the backend of the project, which will involve implementation of rule-based and AI driven analysis of project artifacts - which are uploaded, collected, and analysed. |
| Jen Mao | Front End |  |
| Jason Vo | Scrum Master  Research  Back End | I’m able to work together with my colleagues to establish frequent and clear dialogue, so that we stay on track and deadlines are met. I am also interested in contributing to the major research needed to achieve this project and implementing a solution. |
| Kavindu Bhanuka Weragoda | Front End  UX/UI designer | I am confident in myself contributing towards the front end development of our project since I major in software development and have a keen interest in front end development myself. Also i am attempting advance web development makes me strengthen my capabilities in this area |
| Md Hridoy Mia | Back End | As a Data Science major working as a back-end developer, I understand the importance of teamwork in delivering successful projects. I always communicate clearly, complete my tasks on time, and adapt to the team’s needs. By sharing ideas and solving problems together, I contribute to building reliable back-end solutions that support the overall success of the project. |

## Teamwork roadmap

The team agreed upon two weekly meetings, with the time slots Monday 8:30pm and Saturday 11:00am fitting our schedules best. The team meetings will be conducted predominately on Discord. We came to the decision that discord will be our main informal communication platform. This will allow for files to be shared easily and for relevant information or announcements to be shared by all team members, as discord is also a very optimised application on smartphones - allowing all members to communicate even when they’re not on their laptops / desktops. Microsoft Teams will be used as the platform for formal communication between the team, our supervisor, and the client.

The expected response time between team members in regards to messages is within a few hours, as it’s likely team-members have casual employment or classes during the day. As stated prior, Discord is also an application on smartphones, so notifications for the team server shall always be on. This will improve response time and communication effectiveness between members. In terms of timeliness in regards to attending team meetings, all members are expected to show respect for their team-member’s and client’s time, being ready to join the meeting at the scheduled timeslot. If team-members fail to join a scheduled meeting without prior notice, it will be noted and discussed within the team. If the absences continue to occur without communication, and the team progress is negatively affected, then a supervisor will be contacted and the team-member will be reported. But, if a team-member gives a reasonable heads up that they will be unable to attend, their absence will be excused, and notes will be uploaded and shared with the absent member.

The team aims to complete weekly team submissions 3-4 days prior to their due-date to avoid cramming and submitting work that doesn’t reflect our full potential. The 3-4 day period between completion and submission will allow for daily revisions for the team to revise and review their own and other member’s individually written segments - allowing for constructive criticism and room for each member to voice their opinions. Destructive criticism will not be tolerated, we are all on the same team with the goal of helping one another and not tearing each other down.

Tasks will be assigned to team-members with the technical skills and team role suitable to take it on. Tasks will have different estimations on their completion date depending on the context of the task, its importance, and the size it’s estimated to be. Such as written documentation segments are easier and will require less time to complete in comparison to the implementation of a rule-based AI driven analysis. Team member task information and deadlines will be monitored using Trello. Trello will provide a visual board where tasks can be created, assigned, and tracked across different stages of development. Vital to tracking our sprint progress, Trello also includes checklists and due-dates which will make it easier for team-members to see progress and plan their work/life balance. If tasks end up falling behind, the team-member responsible should raise it as early as possible so deadlines can be altered or extended as a team.

All written documentation files are stored on a Google Drive which all team-members have access to. Google Drive was selected as all team-members can access it from any device, and it supports real-time collaboration - allowing all members of the team to work on a single document at the same time without requiring the document to be pushed or pulled to a repo. Document version history is also available for all Google documentation, which allows for us to keep track of which team-member contributed and when.

## Document management

Shared documents will be stored on a Google Drive, which all team members have access to. Google Docs grants the ability to see changes to documents and observe the document version history, which displays information in regards to team-member contributions, changes on specific dates, and restoration to earlier versions of the file if necessary. This will provide clear accountability and version control.

## Risk mitigation

|  |  |  |
| --- | --- | --- |
| Risk | Impact on Project / Team | Mitigation strategy |
| Difficulty learning and applying ML and AI | Researching and learning artificial intelligence and implementation of machine learning integration into the software could take a substantial chunk of our time. This could result in delays in development or reduced quality of AI additions. | Mitigation could be working within a smaller scope which will prevent us from learning unnecessary information that doesn’t really have relevance to what we are creating. |
| Casual work schedule clashing with team meetings | As most of us team members work a casual job, shift schedules are not the same weekly, and will most likely result in us having to alter our meeting days and times to better suit each other. Creates potential for reduced coordination and could hinder team progress. | This can be mitigated through punctual communication amongst one another, letting the team know in advance that their work clashes with the designated meeting time - allowing us to select a new date. |
| Team member illness or personal issues | Team members could get sick or experience something in their personal lives which could result in them not having the mental or physical energy to complete their tasks on time. A result of this could be missed deadlines, uneven task distribution or reduced individual workload. | This can’t be controlled, so it’s important for all of us to show flexibility and support. Treating them with kindness and respect to ensure they aren’t burdened with any additional stress and can recover in a restful manner. |
| Missed Internal deadlines | Due to unforeseen circumstances or poor planning, internal deadlines for task completion could be missed, resulting in a rushed submission and less time to review our work. This would likely result in a lower grade. | The 3-4 day buffer before submission outlined in the teamwork roadmap allows additional time. Keeping tabs with the Trello visual board to be aware of which tasks are due in the coming week. If a team-member has doubts about finishing a task before the deadline, they should alert the team with as much notice as possible, so that other members can provide assistance to ensure the task gets completed. |
| Uneven contribution from team members | In group projects, oftentimes a scenario takes place where a few group members contribute more or less than others, which can create an unbalanced workload and lead to frustration from team-members. This can impact the quality of the project as the team cohesion lacks. | Mitigation for this problem stems from setting expectations. Team-members are expected to complete assigned tasks on time to the agreed upon standard. If a team-member has concerns, address them early and perhaps redistribute tasks to ensure that the workload amongst everyone is fair and balanced. |

## PART 2 Project overview and plan

## problem statement

**Purpose:**

The purpose of this project is to improve the system by which lecturers and tutors have to mark individual student contributions in group assessments. This typically is a very time consuming and complicated process, as multiple metrics have to be considered in order to grant each student a fair grade - team feedback, commits, worklogs, hours, and supervisor input all play a role in determining this. Bias is also always a possibility, as humans we all have a subjective judgement that can affect fairness in assessments.

**Objectives:**  
The project will aim to develop an AI-assisted system to help lecturers and tutors in determining the grade a student shall receive through careful evaluation of their individual contributions. The objectives include:

* A system summary of all the collected data of the students' contributions, perhaps on a dashboard.
* Assisting lecturers and tutors by analyzing the project artifacts, including worklogs, peer reviews, sprint reports, and documentation.
* Presenting statistics, including percentage of meetings attended, extensions asked for, and timeliness
* The system could suggest a weighted mark change later down the line depending on the impact of each student.
* Incorporation of client and peer feedback of each student, potentially through a checklist
* Separate ratings for peer-to-peer contribution i.e. code, writing, research efforts

This will save time, reduce bias, and provide lecturers and tutors with a succinct data-forward approach to determining individual contributions.

## Scope

The project will focus on the development of an AI tool that helps unit lecturers and tutors assess the individual contribution of a team-member throughout a group project. The main goal is to reduce the time that is typically spent to go through all the submitted project artifacts and rid potential biased marking. Typical project artifacts include commits, worklogs, code quality, documentation, sprint reports, and peer reviews. Non-technical contributions such as documentation and planning shall also be factored in.

In Scope:

* In the beginning, the AI tool can use rule-based analysis to aggregate structured data such as commits, worklogs, code metrics, and meeting attendance which will give us an understandable baseline for contribution scoring.
* Contribution scoring system, which includes statistics such as the percentage of meetings attended, timeliness, and extensions requested.
* Dashboard or summary system to assist lecturers and supervisors when determining the individual contributions.
* Trialing in the beginning with manual uploads of project files and reports for testing.
* Future exploration of using pre-trained Natural Language Processing (NLP) models to analyze unstructured text data, such as documentation, sprint reports, and peer reviews - assessing the effort and quality.
* Research of individual contributions in group projects to further our understanding of the weighing metrics and scoring required.

Out of Scope:

* Development of new machine learning models from scratch.
* Use outside of academic or student team project contexts.
* **During initial development phase**, the fully automated integration of third-party platforms (i.e. Canvas, Trello)
* Launch of a full production-ready system, in this current context the focus is on creating a functioning prototype.

Initially the tool will utilize manual uploads of the project artifacts. After further development, the long-term goal would be to implement data collection autonomously through integration with platforms such as Canvas, Trello, and GitHub. However, these data sources are private and need permission for supervisors to access. The system should still run optimally even if integrations aren’t available.

## stakeholder

* Team Supervisor - Vasudha Malhorta

Our supervisor acts as an internal stakeholder and serves as the key link between our team and the faculty. She monitors our progress, provides guidance, and ensures the project aligns with unit expectations and academic requirements.

* Client - Tanjila Kanij

The client is an internal stakeholder with a direct interest in the project's success. She expects the development of a functional prototype that can be used to demonstrate how student contributions can be evaluated more accurately and fairly.

* Our Student Team

We, as the project team, are also internal stakeholders. We hold direct control over the development and implementation of the contribution assessment platform. We are responsible for designing, coding, documenting, and presenting the system. Our learning outcomes and grades are also directly dependent on the project's success.

* University Staff (Teachers and unit coordinators)

These are internal stakeholders who will directly use the platform to assess student contributions in group projects. They benefit from improved accuracy, fairness, and transparency in evaluation.

* Students in Project Group

Students across different courses are indirect internal stakeholders. Their work will be evaluated more fairly by reducing biases inherent in traditional peer evaluation.

* University Administrators

These are external stakeholders who may adopt or scale the platform across multiple units. They hold a strategic interest in maintaining fairness and consistency in group assessments.

* Wider Academic Community

External stakeholders may include other universities, supervisors, or researchers in the field of education technology. If successful, the project could serve as a model or prototype for fair contribution assessment beyond our university.

## High-level requirements

The proposed system is a new prototype web application design to support lecturers and tutors in fairly assessing individual contributions within student group projects. It is not an upgrade or replacement of an existing system.

Initially, the system will operate as a rule-based prototype, aggregating structured data (commits, worklogs, meeting attendance) through manual uploads. In the long term, the project may incorporate Al/NLP models to analyze unstructured data (documentation, sprint reports, peer reviews). The system is not part of a larger platform but can be extended in the future to integrate with third-party tools (e.g., GitHub, Trello, Canvas).

Functional Requirements (FR)

* The system must support user accounts for students, lecturers/tutors, and supervisors with role-based access.
* The system must allow data collection from project artifacts (GitHub commits, worklogs, sprint reports, documentation, meeting attendance).
* The system must calculate and display an aggregated contribution score for each student.
* The system must provide a dashboard summarizing contribution data in charts and percentages.
* The system must allow lecturers to drill down into specific data sources (e.g., commits, edits).
* Reports must be exportable in PDF/Excel for record-keeping.

Non-Functional Requirements (NFR)

* Security - All contribution data must be stored securely and accessible only by authorized users.
* Usability - The interface should be intuitive, requiring minimal technical knowledge.
* Reliability - Data from different sources should be consistent, accurate, and updated regularly.
* Scalability - The system should support multiple units, teams, and projects simultaneously.
* Transparency - Contribution scores should be accompanied by a breakdown of how they were calculated.

High Level Requirement 1 - User Accounts & Access

|  |  |
| --- | --- |
| Story | Description |
| A lecturer/tutor wants to log in and access student contribution reports. | Lectures and tutors must have secure accounts with permissions to view contribution dashboards and generate reports |
| A student wants to log in and see their own contribution data. | Students should be able to log in and access their personal contribution breakdown but not those of other students. |
| A supervisor wants to manage system permissions. | Supervisors should have accounts with privileges to manage teacher access and monitor system activity. |
| The system must ensure security. | Role-based access control should protect sensitive contribution data from unauthorized users. |

High Level Requirement 2 - Contribution Data Collection & Tracking

|  |  |
| --- | --- |
| Story | Description |
| A lecturer wants to know how much code a student has written. | The system should track GitHub commits and code metrics (e.g., number of commits, lines of code. |
| A lecturer wants to see contributions beyond coding. | The system should analyze documentation, worklogs, sprint reports, and peer reviews. |
| A supervisor wants to consider attendance and timeliness. | The system should log attendance at meetings, extensions requested, and on-time submission of work. |
| The system should produce a contribution score. | All collected data should be aggregated into a weighted percentage score for each student. |

High Level Requirement 3 - Reporting & Dashboard

|  |  |
| --- | --- |
| Story | Description |
| A lecturer wants to quickly see an overview of team contributions. | The system should provide a dashboard summarizing contributions with clear visualizations (charts, graphs, percentages). |
| A lecturer wants detailed evidence of contributions. | The dashboard should allow drilling down into specific artifacts (e.g., code commits, document edits, meeting logs). |
| A lecturer wants exportable results. | Reports should be downloadable in PDF/Excel for grading and record keeping. |
| A student wants transparency in marking. | Students should be able to view their individual contribution summary for fairness and accountability. |

High Level Requirement 4 - System Quality and AI Features

|  |  |
| --- | --- |
| Story | Description |
| The system should be secure and reliable. | All stored and uploaded data should be encrypted, with accurate and consistent retrieval. |
| The system should be scalable. | It should support multiple units, projects, and teams at the same time. |
| The system should provide explainability. | Contribution scores should include a breakdown of how they were calculated. |
| The system should evolve with AI features. | Initially rule-based, but later capable of using NLP models to analyze unstructured text such as peer reviews and documentation. |

## Solution approach

Problem at hand

To individually assess a large variety of student’s contributions in a group project can be very time consuming and difficult to mark. Our objective is to create a tool that assessors can use to aid them in their marking process by using or trialing various technologies such as AI, LLM’s, NLP’s and/or others to analyse quality of code, importance of code, peer reviews, supervisor reviews, contribution towards documentation, etc.

Proposed Solution

Our approach is to design a solution that collects data such as peer reviews, weekly work log reports, other reports/documentation and code, for our AI solution to determine how much a student has contributed to the overall project from certain key metrics. As our solution will be a tool used to assist assessors in their grading process, it will show the multiple scores of different sections from the rubric. For example, it will show a score for contribution to the development of the project solution, research, documentation, peer review score etc. These higher-level scores can then be broken down into lower-level scores so the assessor can better understand how the system has calculated these scores. For example, contribution to the development of the project solution can be broken down into code quality, commit frequency, importance of code written to the overall system etc. These scores give the assessor a general idea of the contributions a student has given as well as the quality of their work, and lets them make the final decision.

As we are still in the early development stages of this project, this approach is subject to change.

To achieve this, we must break down our projects into the following main segments:

Research

Trialing

Implementation

Testing

Research

As a large amount of research will be needed to develop the solution, we will have to prioritise finding the correct technologies and papers that can assist us in development. Our research component will consist of reviewing current research papers and technologies that can guide/aid us to achieve our goal. Topics such as AI, Machine Learning and existing frameworks will be heavily focused and delved into.

Trialing

With the new technologies, frameworks and knowledge we have gathered, we will then trial these new methods to see which approach will work best for our project. This will help us narrow down which technology and methodology to use when implementing our solution.

Implementation

After developing a solid idea of what our solution will look like and how it will be implemented, we can then focus on building our actual solution. This will be the most difficult part of our project and a lot of time and resources must be put into this section.

Testing

When our solution is in the late stages of its development, we will want to put it through many test cases to ensure it is working as intended. Ideally, we will be testing along the way but by having a testing stage we can pour more resources into fixing bugs and cleaning up. We can also use this period to tweak our algorithm by comparing our systems output with the output of real assessors. It is vital that our AI tool provides the most accurate and fair grade possible to not give any unfair disadvantages to any students with the use of this tool.

## Product backlog

The product backlog includes all the features and tasks required to develop the Contribution Assessment prototype for Team Projects. The items are prioritized according to their value, interdependencies, and considerations for sprint planning.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| ID | User Story | Priority | Dependencies | Sprint |
| PB1 | As a user, I can upload a dataset of team contributions (e.g., commits, tasks completed) so that I can build a contribution record. | High | - | Sprint 1 |
| PB2 | As a user, I can see a visual dashboard of team contributions (charts/graphs) so I can quickly understand workload distribution. | High | PB1 | Sprint 1 |
| PB3 | As a user, I can view metrics such as task count, time spent, and % contribution so I can evaluate fairness. | High | PB1 | Sprint 2 |
| PB4 | As a user, I can compare contributions across team members to identify under/over-contributors. | High | PB1 | Sprint 2 |
| PB5 | As a user, I can see a timeline of contributions (e.g., per week) so I can analyse trends over the project. | Medium | PB1 | Sprint 3 |
| PB6 | As an administrator, I can assign roles and permissions so that only authorised users can view/edit sensitive data. | Medium | PB2 | Sprint 3 |
| PB7 | As a user, I can export reports (CSV/PNG/PDF) so I can include results in project reviews. | Medium | PB2 | Sprint 3 |
| PB8 | As a user, I can view recommendations or alerts (e.g., uneven workload, inactive members) so I can take action. | Medium | PB3, PB4 | Sprint 4 |
| PB9 | As a user, I can anonymise member names in reports so privacy is preserved. | High | PB1 | Sprint 1 |
| PB10 | As a developer, I want an automated test suite and CI pipeline so that the system remains reliable. | High | - | Sprint 1-4 |

Alpha Release (Semester 1): PB1–PB4, PB9, PB10 (core upload, metrics, comparison, privacy, testing)

Beta Release (Semester 2): PB5–PB8 (timeline, permissions, exports, recommendations)

## Quality plan

The quality plan ensures that the deliverables of this project meet the technical, functional, and academic expectations outlined by the supervisor and client

**Objective**:

* Accuracy and Fairness: Ensure the tool evaluates contributions reliably, reducing bias by considering multiple metrics (commits, work logs, sprint reports, documentation, peer reviews).
* Transparency: Present results in a clear and interpretable format (dashboard with statistics, contribution breakdowns) that supports, rather than replaces, lecturer judgement.
* User-Centric Design: Align outputs with lecturer and supervisor needs by incorporating their feedback at each sprint milestone (Weeks 6, 9, 12).
* Timeliness: Ensure all internal deadlines are met 3–4 days prior to official submission dates, allowing revision and quality check

**Project Standards:**

* Coding : All code must follow consistent structure, naming conventions, and commenting guidelines.
* Documentation Standards: Written deliverables will be reviewed weekly for clarity, completeness, and alignment with academic requirements**.**
* Version Control & Task Tracking: GitHub (for code) and Trello (for task management)
* Supervisor & Client Feedback: Regular updates and sprint demonstrations will ensure external validation of quality and alignment with project goals.

**Testing:**

* Unit Testing: Each functional component (file upload, contribution scoring, dashboard rendering) will be tested for correctness.
* Integration Testing: Modules (front end, back end, rule-based AI engine) will be progressively integrated and tested to confirm data flow and system interoperability.
* System Testing: The complete prototype will be evaluated against requirements, ensuring it processes project artifacts (logs, code, documentation) as intended.

**Improvement:**

* Weekly: Each meeting will conclude with a short review of what worked, what needs improvement, and actions for the next sprint.
* Risk Tracking: Quality risks (e.g., uneven contributions, delayed tasks, biased scoring) will be logged in Trello and addressed early.
* Adaptive Updates: Feedback from sprint demos will be integrated into subsequent iterations to ensure the tool meets client needs.