Development Plan

Table 1: Revision History

Date	Developer(s)	Change
Date1	Name(s)	Description of changes
Date2	Name(s)	Description of changes
...	...	...

Put your introductory blurb here.

1 Team Meeting Plan

Our team will plan on doing a large amount of our work on this project asynchronous but we want to meet at least once a week as a group. The team created a when2meet page and decided on meeting as a whole team on Wednesdays at 4:30 and whenever else the team see fit. These meeting will be held over Discord. The team will also be meeting with Marie-Jean once a week to give her updates and be provided mentor ship and supervision. This meeting does not have a definite time each week and will depend on Marie-Jean's schedule. These meetings will be held over Zoom.

This project consists of two groups, one group (ours) that is responsible for the Natural Language Processing and another that is responsible for the Operations side of things. The team will be working closely with the Operations side of things and every couple of weeks we will meet with both teams to synchronize our meetings with professor Mosser and the Operations team. Every other week the team will also elect one representative to meet with a member of the Operations team without professor Mosser or Marie-Jean to support synchronization and to make sure the team all remains on the same page.

2 Team Communication Plan

As a team our main form of communication will be through our Discord group chat. The team will plan and host meetings there along with discussing any relevant topics. Discord also provides us the functionality to screen share which

will be very useful. The team also have a separate Discord group chat with the Operations team included to discuss relevant topics with them and plan meetings.
Our team will also be working closely with the team in Quebec and for communication with them we will be using Mattermost.

3 Team Member Roles

Matthew Curtis	Meeting Chair, Developer
Jessica Dawson	Developer
Michael Breau	GitHub Issue Tracker, Developer
Benjamin Chinnery	Team Liaison, Developer
Yaruo Tian	Developer

4 Workflow Plan

The version control we will be utilizing is git and GitHub. There will be a large repository that will also have the Operations team included in it and also material provided by Professor Mosser. This is where all of the coding will take place for our project. We also have a repository just with our team members which is where all of our documentation will be stored for the course. This is where we will work on files such as the Development Plan and the VnVReport for example. We will be using the Dev branch for all of our development and changes. Team members will be responsible for creating new branches off of the Dev branch for there specific work for any given milestone or for general development. They will then be responsible for creating a pull request for when they wish to add there work to the main branch in the repository.

This process will go as follows:

- Create a new branch off of the dev branch to do your work in
- Add and commit changes onto your branch
- Push to your branch
- Go to GitHub and click the green "Compare & pull request" button
- Write your description for your pull request of what you did and then click "Create pull request"

After being reviewed by the group your pull request will be merged.

It is strongly encouraged to create multiple commits per pull request with relatively detailed descriptions in case the team need to roll/revert back or to enable us to find the commit where certain changes were made or where something went wrong in our code. Also it is key that a tag issue is included in

commits with the relevant issue(s) in order to keep track of which commit corresponds to which issues. Your commits should be descriptive and also give a rational for the change.

GitHub will also be used for project management where the team will be utilizing the built-in Kanban board under the Projects tab for task organization.

We are currently using the issues tab in GitHub to organize our team meetings, all our deliverables and lecture attendance. For our team meetings we will use an issue to keep track of attendance and also jot down our agenda for the meeting and what was discussed. The team also will have all of our deliverables marked as issues with the corresponding dates attached in order for us to stay on top of things. We also are labelling all of our issues with the corresponding label being "meeting", "deliverable" or "lecture". For each actual coding related deliverable we will also have more in depth specific issues assigned to define tasks, bugs, and changes in order to systematically keep track of and address every step in our software development cycle for each deliverable. This will help us keep track of what issues (tasks) for a deliverable are still needing to be completed and which are closed. It also helps us track where we are and identify/work on any bugs that may arise. These will be labeled by what deliverable they are part of but will also have a label of "task" or "bug" or any other labels the team sees fit in the future as we progress.

5 Proof of Concept Demonstration Plan

The end goal of the team this year will be to complete and submit a working model during the 2024 eRisk competition. No information regarding the tasks to be completed during the competition are released until it starts. The primary challenge the team will face prior to the start of the competition will be gaining the knowledge of natural language processing necessary to build a successful model during the competition. The main risk the team faces during this time is being unprepared for the competition.

During the proof of concept demonstration the team plans to present a successful recreation of the gambling task in this past eRisk submission: https://ceur-ws.org/Vol-2936/paper-83.pdf. The goal is to demonstrate how much the team has learned and to set a clear timeline for where the team's knowledge should be in November.

6 Technology

GitHub Actions will be used to run Pytest modules as well as check the code with Pylint when code is commited to the repository. This will ensure the quality of code pushed to the repository.

7 Coding Standard

The Python code in this project will follow the PEP 8 coding standards. This will be ensured by Pylint through extensions on the developers IDEs as well as through the GitHub actions check when new code is pushed to the repository.

8 Project Scheduling

The project will largely follow the schedule outlined by the capstone course outline. The schedule will be organized through the use of the kanban board on the team's GitHub repository to ensure the team is on track. Team meetings will be scheduled through GitHub issues which contain information regarding the details of meeting as well as being used to keep track of attendance.

Table 2: Technology that will be used

Technology	**Use**
GitHub	will be used as version control for both the docs and code repositories
Python	will be used as the coding language for the majority of the project
Pytest	will be used to validate code
Pylint	will be used to ensure code quality
GitHub Actions	will be used for continuous integration with Pytest and Pylint
Stanza	Python library used for natural language processing