SE 3XA3: Development Plan
The Resume Shotgun

Team 5, Proper Mars Tribe
Gavin Jameson, jamesong
Jeremy Langner, langnerj
Sam Gorman, gormans

Contents

This document outlines the group's initial plans for maintaining efficient and effective progress on the project as a collaborative effort.

1   Team Meeting Plan

The team will be meeting during regular weekly lab sections to plan and develop collaboratively. Outside of class time, the team will be using Lettuce Meet to plan and schedule meeting times that work for everyone. From an initial assessment, Thursday and Friday nights will likely be the times allocated for any extra group work.

2   Team Communication Plan

Primary forms of communication will be through weekly lab sections and using a Discord group chat. The Discord group chat allows for text messaging, sharing files, sharing links, and unlimited voice calls.

3   Team Member Roles

A scribe will be designated for each significant team meeting, rotating between members each meeting. If a scribe can not attend a meeting, then the next group member will be chosen, with the original scribe taking up the role during the next meeting. It will be the role of the scribe to record what happens during meetings and update any plans accordingly. The order of rotation for scribes will be as follows: Jeremy Langner, Sam Gorman, Gavin Jameson.

Similarly, the team leader will rotate between meetings and be responsible for guiding and keeping meetings on track. The order of rotation for leaders will be as follows: Gavin Jameson, Jeremy Langner, Sam Gorman.

4   Git Workflow Plan

The team plans on using a similar Git Workflow plan as presented in lecture here. This outline has two main branches with an infinite lifetime throughout the project - *Master* and *Develop*. The Master branch is used for main deliverables while the Develop branch is used for the incremental and developmental design process that yields frequent changes. Finally, each member will have their own individual branch to exclusively work independently on features and fixes that can be merged into Develop upon completion.

5   Proof of Concept Demonstration Plan

Our demonstration plan will consist of creating several non-existent students with random resumes with varying backgrounds. Once these are created, the

next step is to use them to find compatible job matches with the application. During this demonstration however, there will not be a formal job application submission since these applicants do not exist. The function to actually apply will have to be demonstrated separately with a real applicant interested in the position. The part of the implementation plan that will be the most difficult is pulling the keywords from the sample resume to be used in comparison with the job application's keywords. The team will have to research potential APIs and other resources to use to make this task easier.

6 Technology

Since the original project is written in Python and it is a language that the team is comfortable using, the project will remain in Python. The supporting documentation will be generated with Doxygen as the team also has experience with it from previous classes. As mentioned in sections 4 and 7, the team will be following specific workflows and conventions - as Python has many capable IDEs, the team will use whichever is most comfortable that can maintain what was discussed in those sections.

7 Coding Style

A few conventions will be followed:

- Variables and functions will be named according to camel case, in which the first word begins with a lower case letter and all subsequent words are capitalized (ie. sumOfProducts()).

- Indentations will be achieved through four individual spaces rather than a \tab character.

- Equations will be formatted with spaces in between numbers and operations, but not between numbers and brackets (ie. [1 + 2] ** 6 is proper form).

- Single line comments will be formatted with two comment characters, followed by a space, followed by the message (ie. %% Example Comment).

- Comments will be placed above important blocks of code, including but not limited to: Function declarations, variable initialization blocks, large loops or conditional statements, complicated mathematical equations.
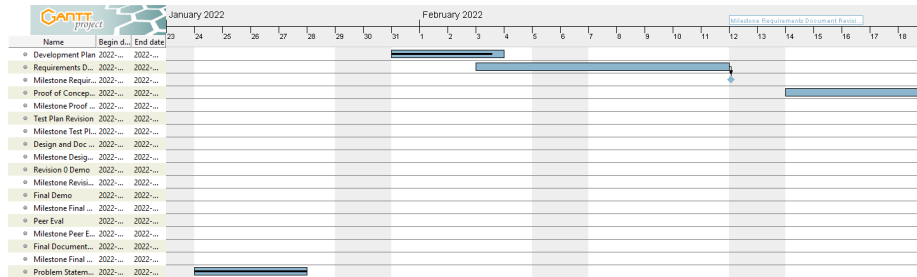
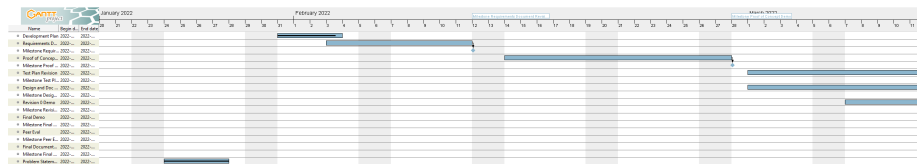# 8 Project Schedule

Figure 1: Gantt Chart Jan 23 - Feb 19

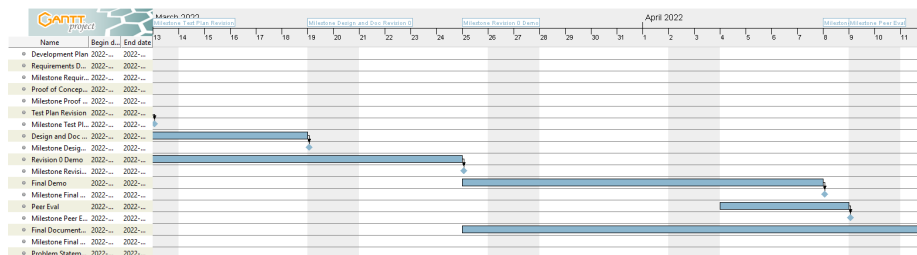Figure 2: Gantt Chart Feb 20 - March 11
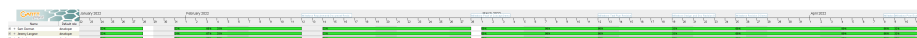

Figure 3: Gantt Chart March 12 - April 12

Figure 4: Gantt Chart Resource Allocation

9   Project Review

There is currently nothing to review.

Table 1: Revision History

Date	Developer(s)	Change
February 1	Everyone	Began out Team Meeting Plan, Team Communication Plan, Team Member Roles, Git Workflow Plan, Technology, and Coding Style.
February 3	Everyone	Project Schedule, Revising previous sections
...	...	...