

SE 3XA3: Development Plan ScrumBot

Team 304, ScrumBot

Arkin Modi, modial

Leon So, sol4

Timothy Choy, choyt2

Last Updated: April 6, 2020

Table 1: Revision History

Date	Developer(s)	Change
January 23, 2020	Arkin Modi	Copy template
January 28, 2020	Arkin Modi	Started Team Member Roles, Coding Style and Team Communication Plan sections
January 29, 2020	Leon So	Added to Team Meeting Plan section and Team Member Roles section table. Started Git Workflow Plan section and Technology section
January 30, 2020	Timothy Choy	Updated Proof of Concept Plan and Project Schedule sections
January 30, 2020	Leon So	Updated Introduction
January 30, 2020	Timothy Choy	Proofread document
January 30, 2020	Everyone	Complete, revise, and update development plan
April 4, 2020	Arkin Modi	Moved title to separate page
April 5, 2020	Arkin Modi	Separated the Team Meeting Plan in subsections
April 5, 2020	Arkin Modi	Added a meeting roles table, & project review

This document outlines the Development Plan for 3XA3 Team 304's ScrumBot. This document outlines the team meeting plan, team communication plan, team member roles, Git workflow plan, proof of concept demonstration plan, technology, coding style, project schedule, and project review.

1 Team Meeting Plan

1.1 Meeting Location & Time

Meetings will primarily take place within the course's two lab sessions each week. If necessary, the project lead will schedule any additional meeting outside of course time with the consent of all team members. The time and location will be decided by the Project Lead at the time of scheduling.

1.2 Meeting Agenda & Roles

For each meeting, the role each member will play is as described in the ~~Team Member Roles~~ section **Table 2**. The project lead will also ~~chair each meeting~~. The project lead is responsible for keeping the meeting on track and ensuring that the meeting is organized according to a meeting agenda. **fulfill the role of Meeting Lead**. The meeting agenda should be reviewed at the start of each meeting. During each meeting, one team member will be in charge of keeping track of meeting ~~minutes~~**notes**. The member in charge of keeping track of meeting ~~minutes~~**notes** will alternate between the two members who are not the project lead. **All members will also assume the role of participant**. At the end of each meeting, the project lead ~~is responsible for producing a written statement of all decisions made during the meeting~~**will assume the role of summarizer**. It is expected that by the end of each meeting, each member knows their role and upcoming tasks. At the end of each meeting, the team should also evaluate the effectiveness of the meeting.

Table 2: **Meeting Roles**

Roles	Description
Meeting Lead	Responsible for keeping the meeting on track and ensuring that the meeting is organized according to a meeting agenda
Recorder	Takes notes of decisions and action items that have been reached during the meeting
Summarizer	Compiles the meeting notes into meeting minutes
Participant	Responsible for contributing towards the meeting discussion

2 Team Communication Plan

Communication outside of class will predominantly be done through Facebook Messenger. Through Facebook Messenger, the team members will communicate any problems and inquires. The Gantt Chart will be used to keep track of the progress of all aspects of the project as well as plan out future work on a high level. Additionally, GitLab's issues feature will be used in parallel to the Gantt Chart to organize work assignment, deadlines, and work status (i.e. not started, in-progress, and completed).

3 Team Member Roles

The following table outlines the roles that each team member will be responsible for the course of this project.

Table 3: Team Member Roles

Role	Member(s)
Project Lead	Leon So
Scribe	Arkin Modi
Developer	Arkin Modi, Leon So, Timothy Choy
Documentation Expert	Arkin Modi, Leon So, Timothy Choy
Git Expert	Arkin Modi
LaTeX Expert	Timothy Choy
Technology Expert	Leon So

4 Git Workflow Plan

The master branch should always be a working branch which contains the most recent stable version of the application. For any development of the application, each push to master will have a tag to indicate the version of the project. Any development will be done on separate branches parallel to master, which will then be merged into master after new changes have been tested. Any hot-fixes will be done in a new branch parallel to master.

5 Proof of Concept Demonstration Plan

The main challenges for this project will be testing the Discord bot and other challenges associated with implementing and connecting to various technologies (i.e. APIs), since this project heavily relies on the Discord API. Testing may prove difficult because it is never possible to cover all cases in the code, and building a Discord bot can pose many more unknown problems that we have little to no prior experience with.

To tackle the testing challenge, we plan on using Pytest to unit test the code, more specifically specification testing through testing our written specifications. We also plan on using exploratory testing techniques to find errors by creating a Discord server and running commands there. This will be accomplished by acting as testers ourselves, as well as enlisting other peers to achieve how users would actually use the bot. With these methods of testing, we can be confident that the code has been sufficiently tested and covered.

~~To tackle~~ For the challenges of connecting to various APIs and systems, we plan on refining our requirements documents such that it outlines how each API is to be used and how our modules will interact with external interfaces. Detailed documentation will help us stay organized in the implementation stage of this project.

6 Technology

The main programming language used for this project will be Python3. The IDE used will be Visual Studio Code. The framework used for functional testing will be Pytest. All Python code written will be documented using the document generator Doxygen.

7 Coding Style

The project will be written in Python following the [Google Python Style Guide](#).

8 Project Schedule

The project schedule is written in the form of a Gantt Chart and found in our repository under ./ProjectSchedule. From the Gitlab repository, it is found [here](#).

9 Project Review

~~N/A for revision 0.~~ The development of this project went mostly as expected. As a team, the work was split relatively evenly and everyone was able to contribute. The end resulted in a working application that fulfills most of the initial requirements that the team had set out to complete. The newly redesigned application did add many new features and expanded far beyond what the original project entailed.

With that being said, there a few places that could have been improved. In terms of the final application, not all the requirements were fulfilled. The team had wished to implement more services (i.e. Google's API and Trello), however, there was not enough time. In terms of the team's workflow, time management could have been improved. While everything was completed within the planned time slot, many things that could have been finished early were left until the end of the allotted time.