Requirements Workshop

0.a Team name: The Discordians

0.b Names and pid

Shreya Aravindan – saravindan

Justin Turkiewicz – justint417

Geethika Abhilash – geethika

John Byman – johnbyman

Sheril George – sherilhannah

Arav Tripathi – aravt

0.c Preliminary project idea(s)

Discord focus bot that can send discord group/server members reminders on events and tasks as well as send messages when specific activities for a git repo occur such as a git commit, pipeline passing or failing and merge/pull request. The bot will also support additional features such as applying pomodoro techniques and displaying deadlines. The bot can be configurable via commands sent in messages on Discord.

1. Nonfunctional requirements

1. Usability - Bot displays list of possible focus items to choose from (Calendar, reminders, incomplete tasks).
2. Reliability – Bot cannot fail to give important reminders such as for a project or homework deadline.
3. Performance – Bot should respond quickly to user input
4. Supportability – The bot should be customizable in the amount it notifies user
5. implementation/Constraints - Bot is made in Python and is only used for Discord

2. Functional requirements:

1. Bot should send reminders 1 day before deadlines
2. There should be a discord command to set reminders
3. Bot could display if some made a git commit or a ci/cd pipeline passed/failed
4. Bot can display list of deadlines
5. Bot could support Pomodoro technique, 25 min work sessions with short breaks in between

3. Tasks and values

Values: 0, 0.5, 1, 2, 3, 5, 8, 13, 20, 40, 100

1. <u>Non-Functional Requirement 1:</u> Bot displays items when user enters a certain command.

   Value: 1

   Explanation: Numerous commands to support and must decide on what items to display but should not be too difficult to implement.

2. <u>Non-Functional Requirement 2:</u> Bot is consistent in sending reminders:

   Value: 2

   Explanation: Important from an end user perspective so need to test with care though it should be simple to test.

3. <u>Non-Functional Requirement 3:</u> Ensure bot is snappy and quick when responding to user commands.

   Value: 3

   Explanation: Tricky to get the bot to respond in a hastily manner. Would require checking the efficiency of the code and optimizing it as much as possible.

4. <u>Non-Functional Requirement 4:</u> Bot is configurable with which specific notifications it sends.

   Value: 5

   Explanation: Sufficient coding is required to implement the various settings for notifications.

5. <u>Non-Functional Requirement 5:</u> Initialize bot for Discord with Python.

   Value: 5

   Explanation: Understanding how to build a bot to work with Discord will be the main challenge for this task.

6. <u>Functional Requirement 1:</u> Reminders are sent 1 day before the deadline.

   Value: 1

   Explanation: Simple check to ensure that reminders are sent at a specific time.

7. <u>Functional Requirement 2:</u> Create a command to set reminders.

   Value: 3

   Explanation: Setting a reminder involves various different forms of input, such as what the reminder is, when to send it and if it is a repeatable reminder.

8. <u>Functional Requirement 3:</u> Notification for git commits pipeline statues and merge/pull requests.

   Value: 13

   Explanation: First the bot must be configured with the git repo and then it must support notifications for commits pipeline status and merge/pull requests which have a significant number of features to implement.

9. <u>Functional Requirement 4:</u> Displaying deadlines.

   Value: 1

   Explanation: Printing and formatting output should not be a challenging task at all.

10. <u>Functional Requirement 5:</u> Pomodoro technique.

    Value: 8

    Explanation: Including the various settings and configurations for the Pomodoro technique could prove somewhat challenging. A significant amount of code must be written.

4. Three User Stories

   1. Discord server leader needs to support notifications from multiple git repositories
      Acceptance Criteria:
      a. Bot is configurable to add multiple git repositories that it can watch
      b. Bot reports the notification for each git repository in a separate Discord channel

   2. Discord member can view, create, update and delete deadline reminders
      Acceptance Criteria:
      a. Bot supports commands to view, create, update and delete a deadline reminder
      b. Discord member has permissions to use the commands built from acceptance criteria #1

   3. Discord server leader can configure what git notifications the bot sends in the discord

      Acceptance Criteria:

      a. Bot is configurable to enable/disable git commit messages to all or specific branches
      b. Bot is configurable to send/not send git messages if the pipeline passes/fails
      c. Bot is configurable to send/not send messages when a merge/pull request is made
      d. Bot is configurable to mention discord members linked from their GitLab/GitHub account if their specific git commit caused the pipeline to pass/fail

5. Two Examples of Risk

   1. The bot could potentially mis-categorize or not provide proper notification about a key deadline

        a. We could minimize this risk by conducting rigorous bug testing to ensure that all notifications are delivered on time

  2. Bot create an unnecessary number of notifications causing disengagement by users

        a. We can minimize this risk by allowing for notification customization and different rankings of importance

6. Process for Requirements Elicitation

Because students like us and our group are the primary users of the bot, the primary form of elicitation can be brainstorming. We can also include interviews with students outside of our group. We could start by conducting surveys with students to gather basic information regarding requirements and desired objectives for the focus bot.