RepoRanger

The Bit Co.

Kanad Naleshwarkar   
 kanadn@vt.edu

Sanket Bhujbal   
 sanketb@vt.edu

Sarthak Banerjee   
 sarthakb@vt.edu

ABSTRACT

*In today's world, distributed teams have become a norm. The need for remote collaboration tools is on the rise, and Discord is one such tool that is widely used by software development teams for communication. This project proposes the development of a bot, RepoRanger, that leverages GitHub and Discord API to allow team members to manage the project repository, CI/CD pipeline as well as issue tracking through a Discord channel. The bot will enable team members to perform repository management tasks, such as creating branches, committing code, reviewing pull requests, project management tasks like creating, updating, and closing issues as well as CI/CD tasks like initiating a workflow, monitoring project build and deployment all from within the familiar environment of Discord. The aim of this project is to enhance the efficiency of software engineering teams by providing them with a more streamlined communication channel.*

1. **INTRODUCTION**

Remote work induced by the pandemic has increased software engineering teams’ reliance on communication apps like Slack [1], Discord [2] and Teams [3]. Even though these apps provide an excellent collaborative platform, there are many aspects of project management that can be improved. For example, it is particularly difficult for DevOps managers in a team to quickly get info on how long it takes for a new release of the product to be built and deployed on servers, especially if the manager lacks technical knowledge. Similarly, with the lack of a physical Kanban board, project managers find it troublesome to go back and forth with issue-tracking portals while discussing open issues with the team.

We believe that such problems can be solved, not by developing new software, but by integrating existing tools so that the combination can provide more value to engineering teams. Therefore, we propose an intuitive integration of GitHub [4] and Discord, and we call it ‘RepoRanger’. This application will provide an interface for team members, by means of a Discord bot [5], to access their GitHub repositories, projects as well as CI/CD pipelines (workflows) without having to switch between various portals.

This integration, for instance, can help managers get notified about recent issues, pull requests, etc., or fetch a particular issue from the backlog so that it can be discussed with the team in the same Discord channel. The RepoRanger bot will facilitate this communication by providing a set of slash commands [6] that abstract complex API calls and git commands, making it easier to quickly get info and perform operations on the user’s behalf. Hence, we believe RepoRanger will improve efficiency, reduce errors, and ensure that everyone is on the same page.

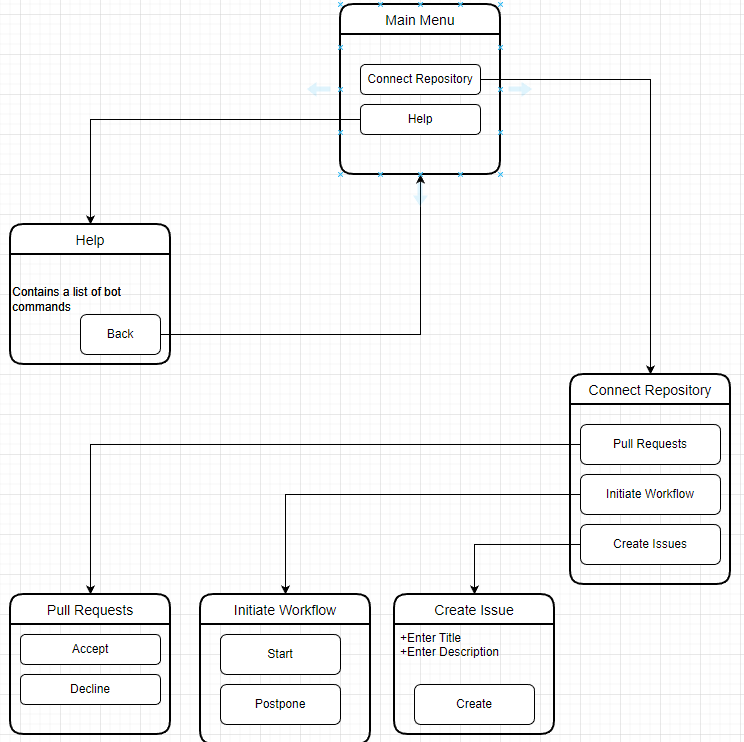
1. **RELATED WORK**

A similar integration for Slack and GitHub has already been developed [7]. Also, there are many services like Pipedream [8] that provide integration between GitHub API and the Discord Webhook API. Based on this inspiration, we plan to develop RepoRanger. We aim to achieve maximum usability of this bot by making it easy to configure with any GitHub repository and project.

1. **DESIGN**

RepoRanger will be based on the message broker architectural pattern as it will facilitate communication between Discord and GitHub and by doing so, it can simplify commands for the user. For example, a complex git command to list all the pending pull requests can be simplified to just one request to the bot. This will improve the accessibility of GitHub even for the stakeholders that are not well versed with the console.

This project will have the following 3 major components:

* 1. **GitHub App**

GitHub Apps are the recommended way to communicate with GitHub as they offer better security. For this project, we plan to develop a GitHub app called RepoRanger that can be installed on the organization or personal GitHub accounts and can be given access to specific repositories. This app can act on its own, meaning we don’t need to create a separate bot or user to perform actions on the repository. But for tasks like modifying workflow files, the app can authenticate on behalf of a user through user-to-server interaction.

* 1. **Discord Bot**

Users of a Discord server will communicate with RepoRanger through a Discord bot. This bot can be invoked through a slash command, for example “/pull-requests”. These commands can abstract complex operations of managing a repository, additionally, we can fetch project information about pending issues and CI/CD workflows without leaving the communication channel. Thus, the bot will provide a simpler interface for teammates to manage repositories and projects.

Figure : Storyboard for RepoRanger with sample use cases.

* 1. **Remote Server**

The GitHub App (RepoRanger) will be hosted on a remote server so that it is available 24x7. We are planning to use Replit to host this app as it provides free hosting, easy setup and built-in tools. The remote server will be installed with JavaScript frameworks like Node.js so that the GitHub App can listen to events related to a repository and can perform REST API operations.

Considering the need for multiple classes so that responsibilities can be distributed, we can make use of the behavioral design pattern. In this way, one class will be responsible for mapping Discord bot commands with corresponding APIs, and another class can execute actual API calls and so on.

Moreover, to ensure RepoRanger truly helps software engineering teams and boosts their productivity, it should satisfy the following requirements:

1. Integrability: This bot should seamlessly integrate with any repository.
2. Extensibility: The bot should be designed in such a way that features can be added with minimum friction.
3. Good response time: The bot should fetch info and respond to requests as quickly as possible.

# References

|  |  |
| --- | --- |
| [1] | "slack.com," Slack, [Online]. Available: https://slack.com/. |
| [2] | "discord.com," Discord, [Online]. Available: https://discord.com/. |
| [3] | "microsoft-teams," Microsoft, [Online]. Available: https://www.microsoft.com/en-us/microsoft-teams/group-chat-software. |
| [4] | "github.com," GitHub, [Online]. Available: https://github.com/. |
| [5] | "discord-bot," Discord, [Online]. Available: https://discord.com/developers/docs/intro#bots-and-apps. |
| [6] | "slash-commands," Discord, [Online]. Available: https://support.discord.com/hc/en-us/articles/1500000368501-Slash-Commands-FAQ#:~:text=Slash%20Commands%20are%20the%20new,command%20right%20the%20first%20time.. |
| [7] | "slack.github," GitHub Inc., [Online]. Available: https://slack.github.com/. |
| [8] | "pipedream," [Online]. Available: https://pipedream.com/apps/github/integrations/discord-webhook. |

Conference Name:ACM Woodstock conference

Conference Short Name:WOODSTOCK’18

Conference Location:El Paso, Texas USA

ISBN:978-1-4503-0000-0/18/06

Year:2018

Date:June

Copyright Year:2018

Copyright Statement:rightsretained

DOI:10.1145/1234567890

RRH: F. Surname et al.

Price:$15.00