RewardBot - Demo HokieTechies

CS5704 - Spring '23 - May 6, 2023

Dhruveel Chouhan

Fasi Ullah Khan Mohammed

Ramaraja Ramanujan

Ramnath Raghu

Shaunak Juvekar

VIRGINIA TECH™

Problem Statement

What?
- Not able to quantify good quality work
- Lack of motivation to do good quality work
- Missing actionable objectives
- Lack of recognition

Motivation?
- "Quiet Quitting" and "The Great Resignation"
- Growing discontent
- Identify top performers
- Foster a healthy work culture

Proposed Solution - RewardBot

Features:
- Slack Bot that provides reward points
- Leaderboard
- External Integration - GitHub

Utility:
- Provides recognition for hard workers
- Provides tangible rewards for good quality work
- Promotes healthy competition which in turn drives up productivity
- Fosters a healthy work culture that keeps people content
- Helps the company identify and retain top performers

Points Reward System

Use Case

Use Case: Manually Awarding Points to Peers

Preconditions:

- The employee must have the necessary authorization to use the RewardBot.
- The employee should not have used the RewardBot to award points more than three times.

Main Flow:

The employee will input the award details, such as the recipient's name, the type of award, and the reason for giving the award, using the appropriate slash command [S1]. The RewardBot will verify the response and save the points in the database [S2]. The RewardBot will send a private message to the awarder confirming the award details[S3].

Subflows:

[S1] The employee triggers the RewardBot using the `/reward` slash command, and the bot will prompt them to input the recipient's name, the type of award, and the reason for the award.

[S2] The RewardBot will validate the response and update the recipient's points if the validation is successful.

[S3] The RewardBot will send a message with the award details to the public dedicated bot channel.
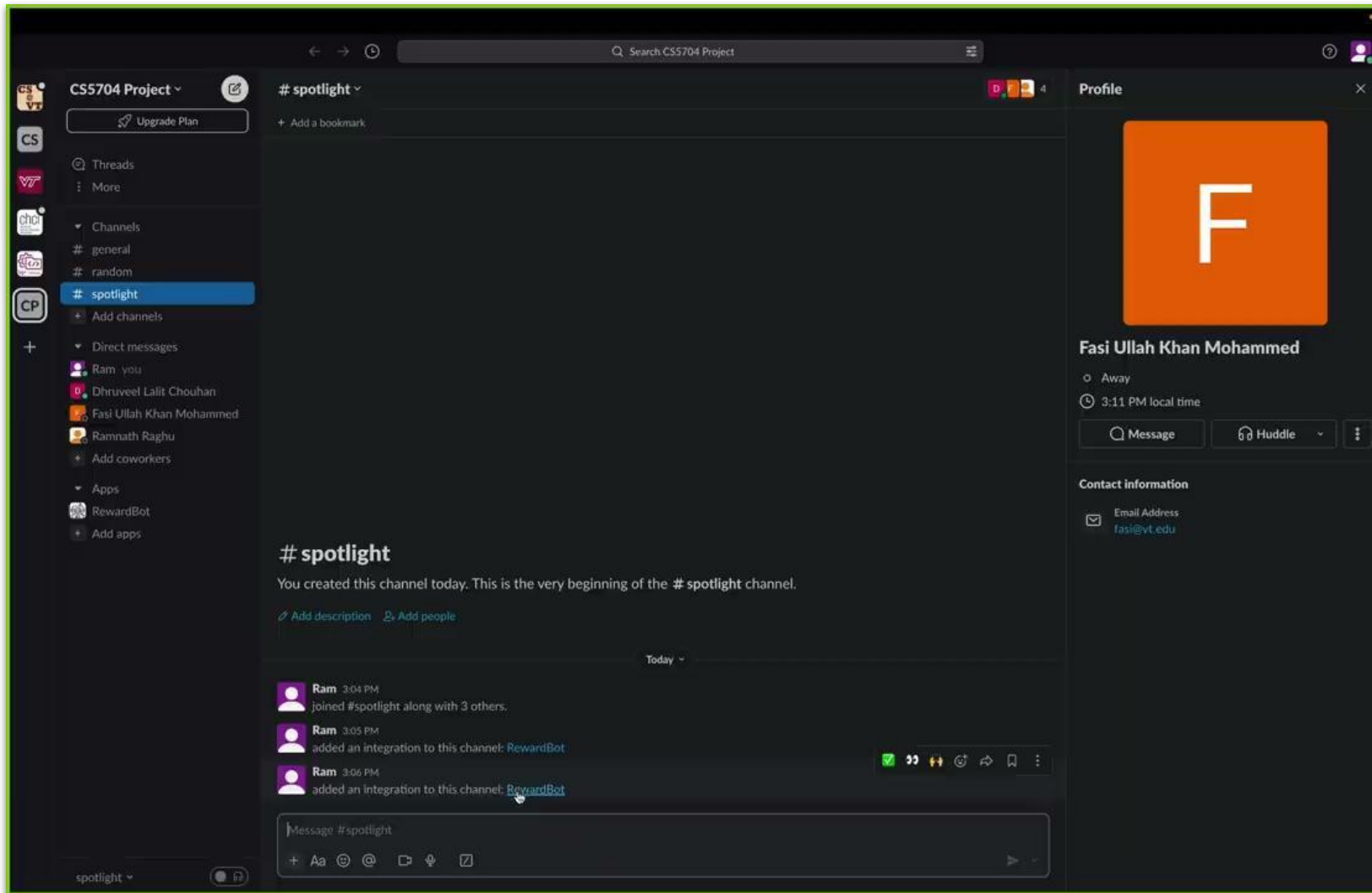
Postconditions:

- - The recipient's record is updated with the awarded points.
- - The awarder-to-awardee quarterly award count increases by one. (Only three awards can be given by a specific awarder to a specific awardee in a quarter)

User Diagram
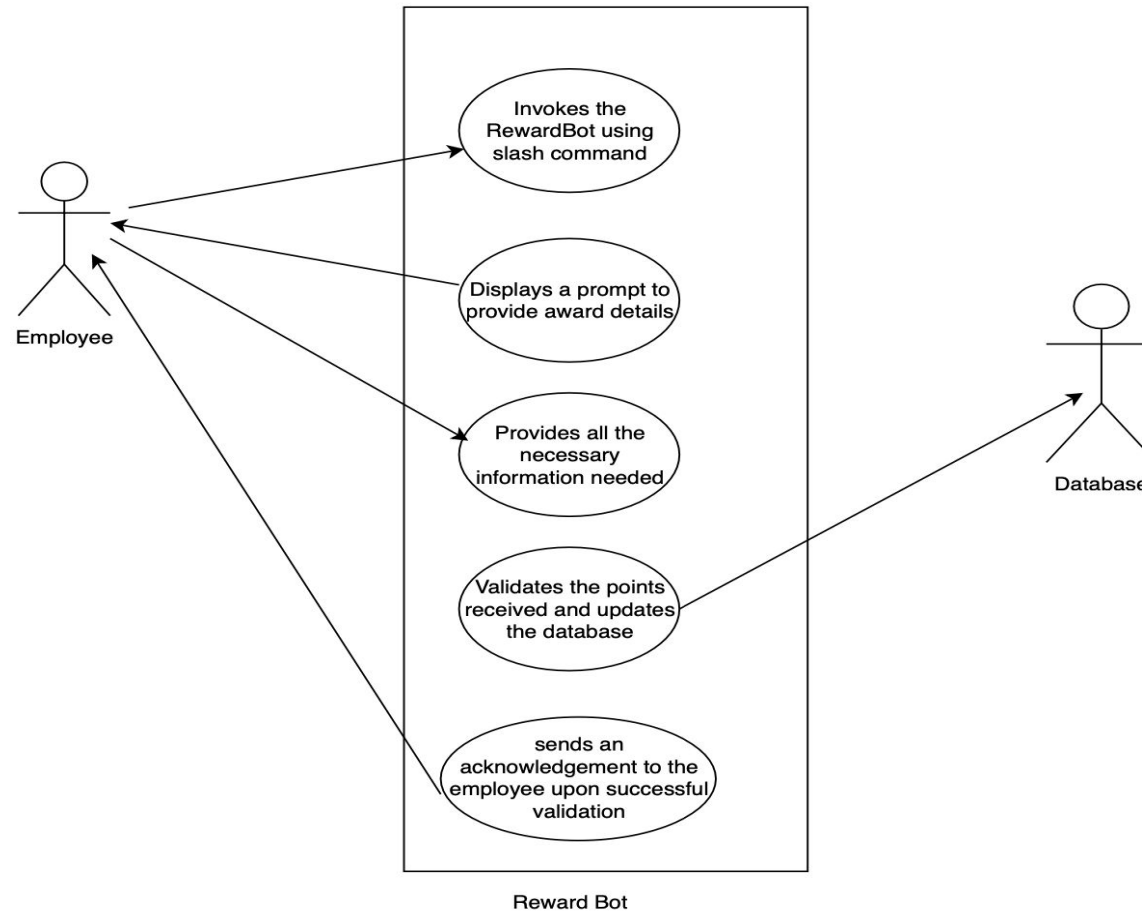
Demo

Leaderboard

Use Case

Use Case: Display the leaderboard showing the name of the top ten performers and their accumulated points.

Preconditions:

1. The employee should have the necessary permissions to interact with the bot.
2. There should be valid entries in the database of the respective employees and their details.
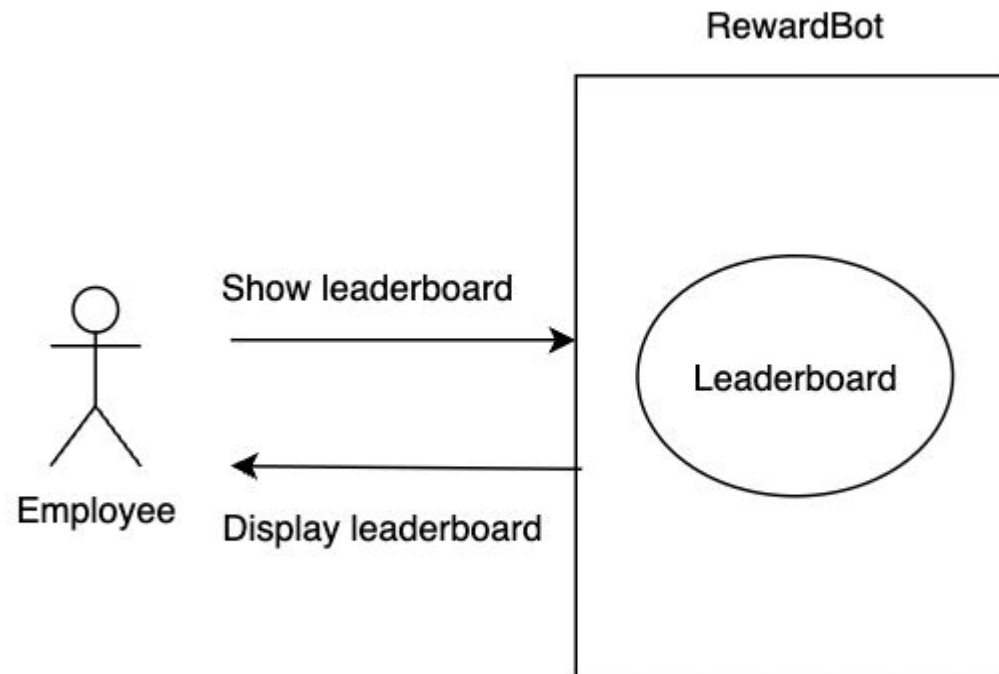
Main Flow:

The employee will invoke a slash command, triggering the Reward Bot [S1]. The Reward Bot will then show the leaderboard to the employee on a private channel [S2].

Sub Flow:

[S1] Employee invokes the bot using the `/leaderboard` slash command.

[S2] The bot then sends a direct message containing the leaderboard of the top ten performers to the employee who invoked the command.

User Diagram

Demo

External Integration - GitHub

Use Case

Use Case: Integrating our RewardBot with GitHub

Preconditions:

1. The employee should be present in the database.
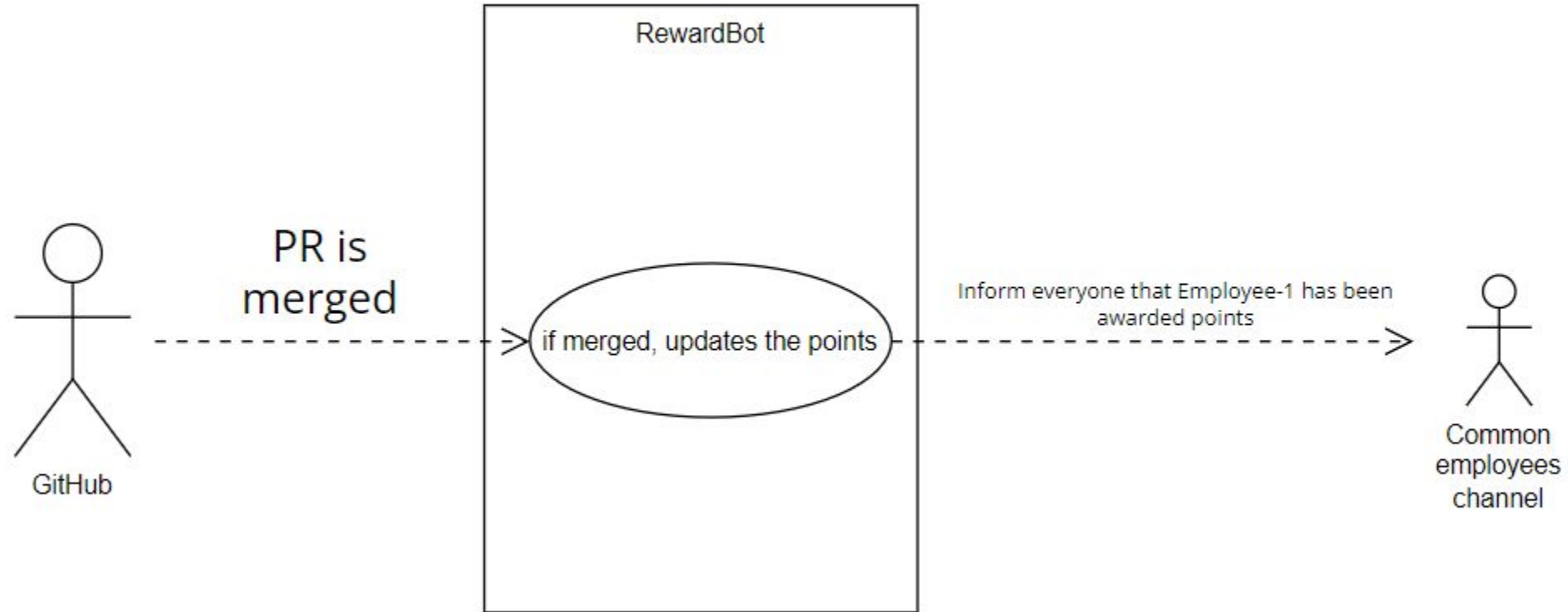2. The employee should have raised a PR.

Main Flow:

The bot will award the points to the employee who raised the PR once it is merged[S1]. The bot will also post a message on the common channel and inform everyone about the awarded points[S2].

Subflows::

[S1] Once the PR is merged, it automatically awards points to the employee and will update the database

[S2] It will post a message on the Slack channel saying Employee ABC has been awarded 10 points for their contribution.

User Diagram

Demo

Limitations

- Available only on Slack
- Only one external integration i.e. Github

- As effective as the frequency of the user interactions

- Lack of user study

Future Work

- Further external integrations such as JIRA, Salesforce, StackOverflow, etc.

- Improve gamification - adding levels and badges

- Rules - manager approves the peer-to-peer rewards

- Leaderboard 2.0 - metrics and visualizations

- Customizability - to tailor according to the organization's culture

- Points Redemption - exchange points for rewards

Lesson Learnt

What went well?
- Develop an app end-to-end from scratch
- Collaboration as a team
- Usage of tools for Software Engineering

What did not go well?
- Time management
- Learning curve
- Cloud deployment

Thank You!
Questions?