

Discord Focus Bot

Team Name: The Discordians

Team Members: Shreya Aravindan,
Justin Turkiewicz, Geethika Abhilash,
John Byman, Sheril George and Arav
Tripathi



Original Problem Statement

- Software engineering teams often face challenges in effectively managing deadlines and tracking code changes within their projects.
- Discord adds to the communication burden for software engineering teams.
- Manual tracking of deadlines and code changes leads to inefficiency.
- There is a need for automation to streamline communication and enhance productivity.

Proposed Solution

- Introducing the Discord bot, which we have designed to streamline communication and enhance productivity amongst teams of software engineers.
- The bot automates reminders for deadlines, reducing the risk of missed milestones.
- Git integration facilitates real-time notifications for code changes, which enhances collaboration.
- Accounts for customizable notifications that care to individual preferences and workflows.

Related Work

- [1] “The best calendar bot for discord,” The best calendar and event bot for Discord, <https://sesh.fyi/>.
- [2] A. Horesh, “The best discord bot,” LionBot, <https://lionbot.org>.
- [3] D. Falconer, “Falconerd/discord-bot-github: Github repo updates displayed in discord,” GitHub, <https://github.com/Falconerd/discord-bot-github>.

Features

1. Discord bot creates, edits and deletes a task board and its tasks
2. Discord bot sets, deletes, and sends reminders for deadlines
3. Discord bot sends notifications for different git actions
4. Discord bot supports the pomodoro technique

Use Case Diagram: Discord bot creates, edits and deletes a task board and its tasks

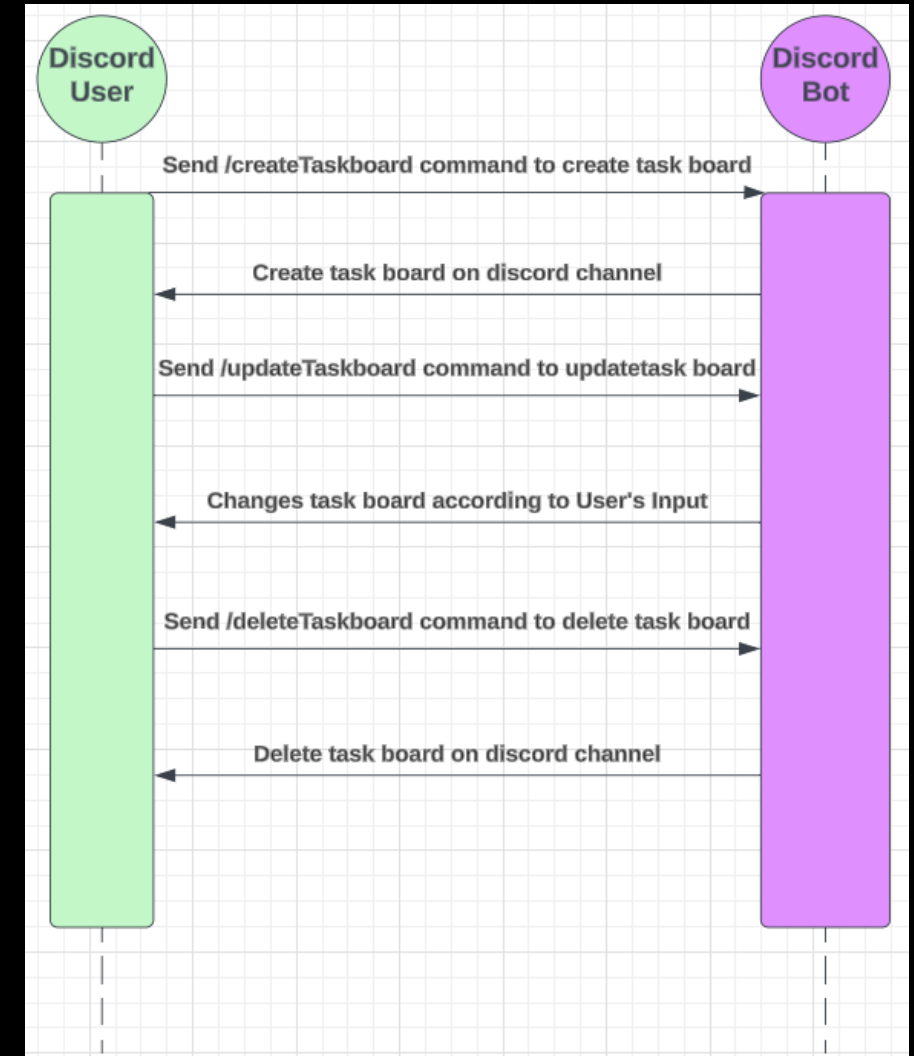
User creates a task board in a discord channel, updates it (e.g. title, font size) and then deletes it.

Commands issued to Bot:

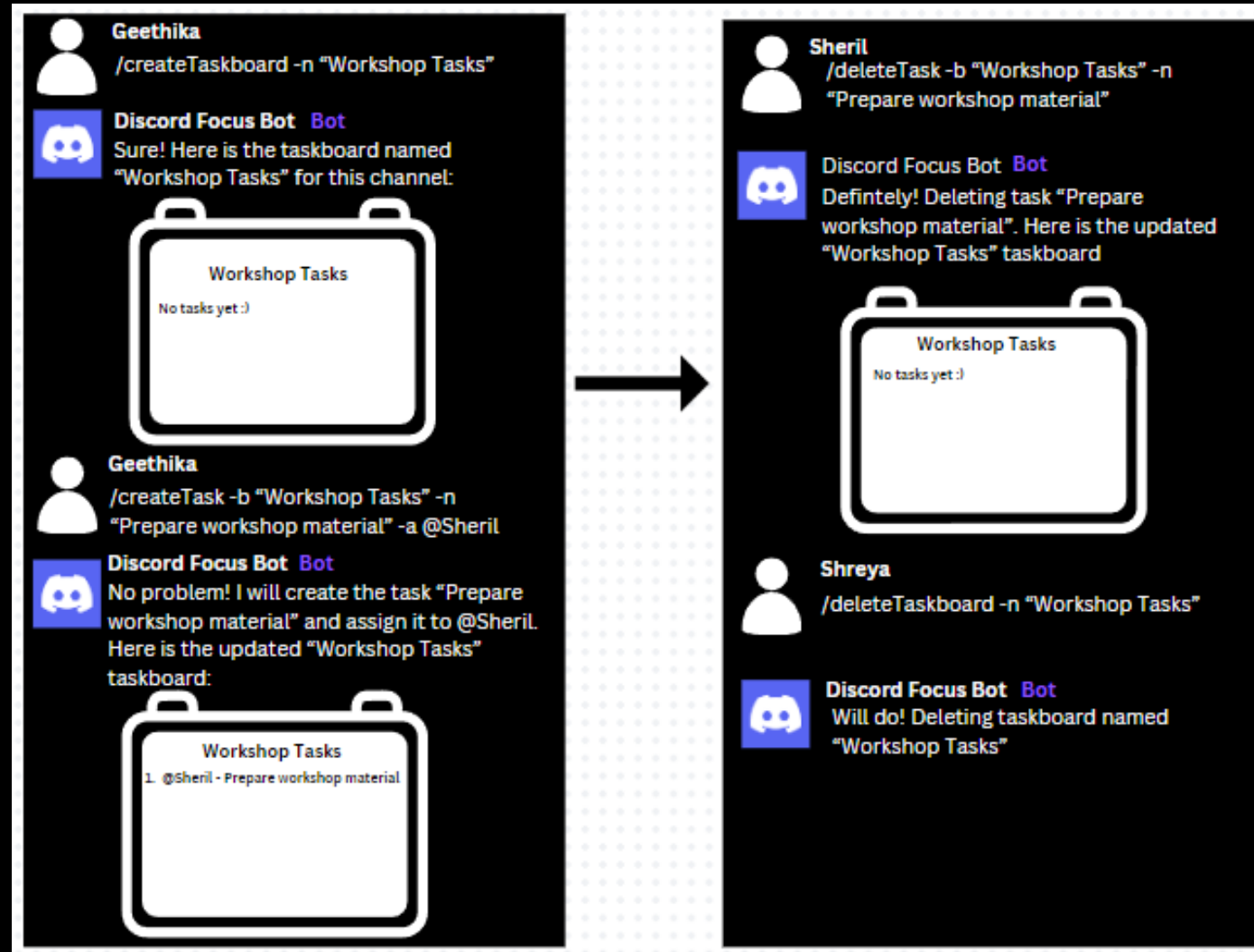
- /createTaskboard - Creates taskboard
- /updateTaskboard - Updates taskboard
- /deleteTaskboard - Deletes taskboard

Similar API for Tasks:

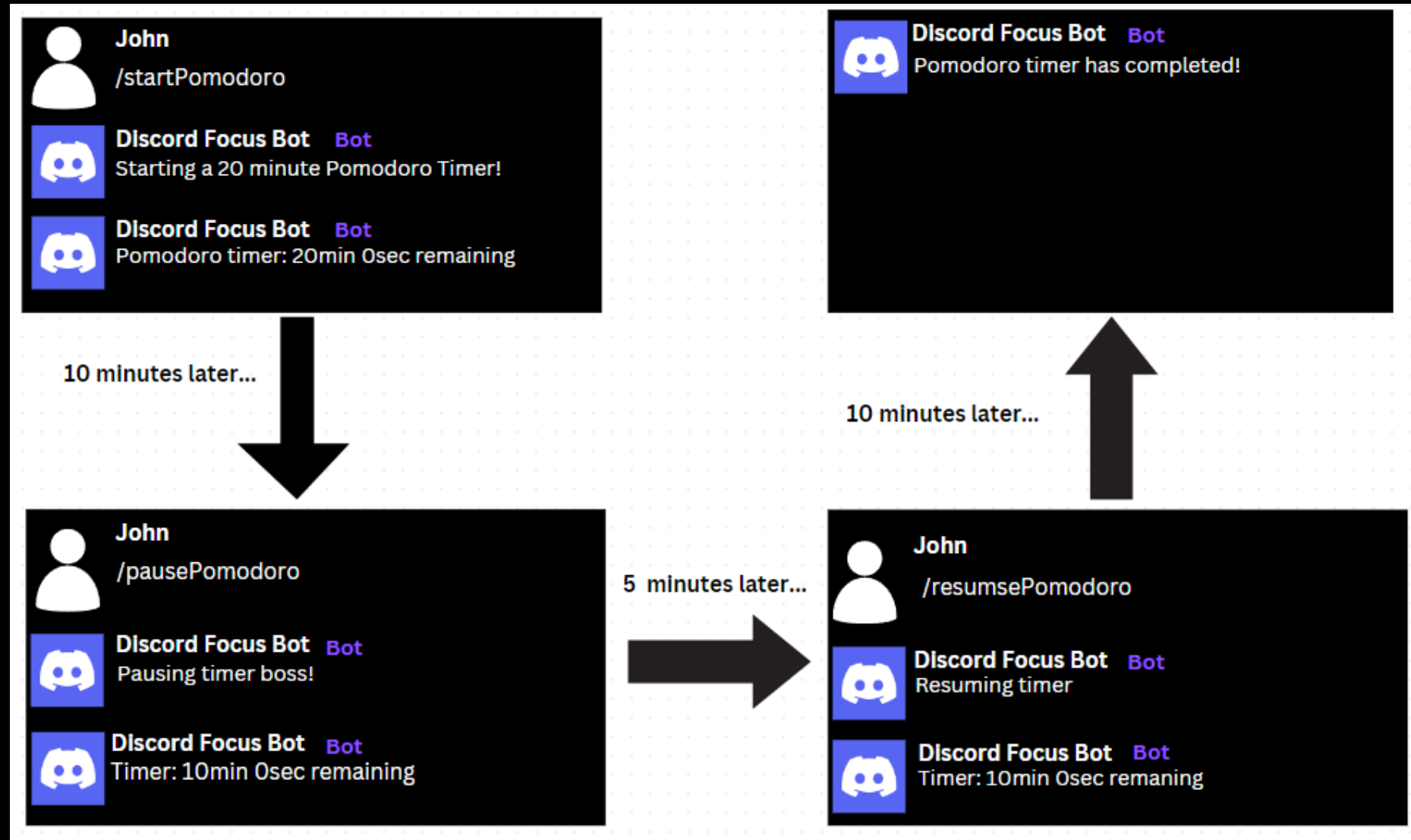
- /createTask, /updateTask, /deleteTask



Mock User Interface Example 1: Create & Delete Taskboard and Tasks



Mock User Interface Example 2: Using Pomodoro Timer



Concepts Used from Class

1. We used **Scrum Meetings** to track team members progress for the Process Deliverable Assignments
2. For the Mock User Interfaces, we created **storyboards** to detail the typical user flow for interacting with the bot
3. We followed the **first golden rule of user interface design**: "Place the user in control". User can facilitate the bot as they like via issuing commands to the bot [1]
4. The bot uses multiple **Bot Design Patterns** such as Notifier (sends reminder messages) and Responder (Responds to commands based on the memory of the task board)

Lessons Learned

1. **Deepened Understanding of Key Concepts:** Enhanced grasp of classroom theories through their practical application, enabling a real-world connection to theoretical knowledge.
2. **Focus on User Needs:** Emphasized learning the critical importance of designing features that effectively address and fulfill specific user requirements, ensuring usability and satisfaction.
3. **Benefits of Iterative Design and Feedback:** Continually improved the bot's functionality and user experience by implementing iterative design processes and integrating ongoing user feedback, leading to a more refined product.
4. **Essential Project Management Skills:** Developed essential project management skills by clearly defining project scopes, establishing precise deadlines, and efficiently allocating tasks among team members, which streamlined project execution and enhanced team coordination.

Future Work and Additional Features

1. Focus mode
 - Mutes notifications, hides distracting channels
2. Integrations with other productivity tools
 - Google Calendar
3. Progress Tracking
 - Provide stats on focus sessions, completed tasks
4. Progress Visualization
 - Productivity trends over time
 - Charts and graphs of completed tasks

Limitations of the Discord Focus Bot

- Rate Limits – Bot performs actions at a certain rate, if too many commands sent too quickly bot may slow down and stop responding
- Internet Connection Dependent – Bot needs stable internet connection to function, won't work without it
- Limited Multitasking – May struggle to do any tasks simultaneously, which could lead to a slower response time
- Learning Curve - Users need to learn specific commands to communicate with the bot

References

- [1] T. Mandel, “Chapter 5 The Golden Rules of User Interface Design,” in The Elements of User Interface Design, John Wiley & Sons
- [2] “The best calendar bot for discord,” The best calendar and event bot for Discord, <https://sesh.fyi/> (accessed Feb. 18, 2024).
- [3] A. Horesh, “The best discord bot,” LionBot, <https://lionbot.org/> (accessed Feb. 18, 2024).
- [4] D. Falconer, “Falconerd/discord-bot-github: Github repo updates displayed in discord.,” GitHub, <https://github.com/Falconerd/discord-bot-github> (accessed Feb. 18, 2024).