



# Taskboard

Nishesh Saikrishna  
Peter Murphy  
Joel Buba  
Benjamin Hurt



# Problem

## Dividing Work

When working in teams, many software engineers have difficulty dividing up tasks between members

## Due Dates

With many deadlines to meet, many software engineers have trouble keeping track of due dates

## Details

When working on multiple projects, software engineers may forget minor details in each project



# Our Solution

## Taskboard

Allows management of users working on a project

Designate tasks to users to help spread the work on a project efficiently

Give due dates to certain tasks so members know how long they have as well as which tasks to prioritize

Allows users to add notes to individual tasks



# Why Our Taskboard?

## Similar Projects Exist



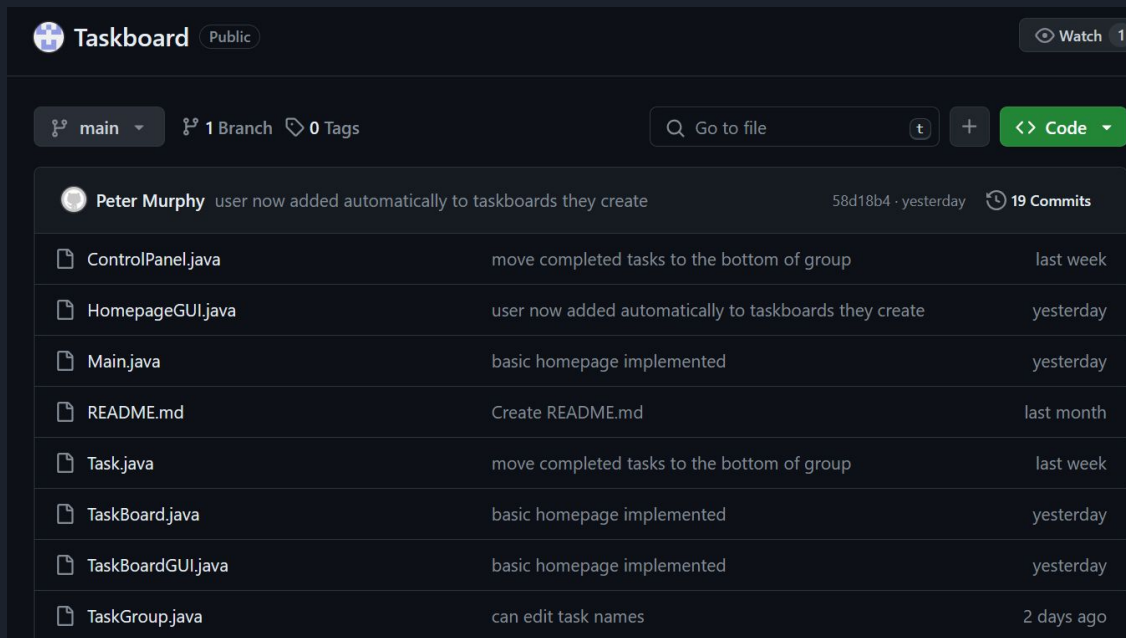
## Why Did We Make Our Own?

We were frustrated by the organization methods of existing offerings

We wanted a more clean, accessible layout without being bogged down by other connected services such as with Google

# Connected Class Topics

Cloud-Based Computing  
Our team used GitHub



The screenshot displays the GitHub interface for a repository named 'Taskboard', which is public. At the top, it shows the repository name, a 'Public' badge, and a 'Watch' button with a count of 1. Below this, the branch 'main' is selected, with 1 branch and 0 tags. A search bar for files and a 'Code' button are also visible. The commit history is shown, starting with a commit by Peter Murphy. The commit message is 'user now added automatically to taskboards they create', with a commit hash of 58d18b4 and a timestamp of 'yesterday'. There are 19 commits in total. Below the commit history, a table lists the files in the repository.

File	Commit Message	Time
ControlPanel.java	move completed tasks to the bottom of group	last week
HomepageGUI.java	user now added automatically to taskboards they create	yesterday
Main.java	basic homepage implemented	yesterday
README.md	Create README.md	last month
Task.java	move completed tasks to the bottom of group	last week
TaskBoard.java	basic homepage implemented	yesterday
TaskBoardGUI.java	basic homepage implemented	yesterday
TaskGroup.java	can edit task names	2 days ago



# Connected Class Topics

- Design process
  - Event Based Design Pattern
  - This is because the execution of one event leads to another
  - Example: The user created a task, so the task appears on the homescreen
  - Many different events can be triggered
    - Adding notes to tasks
    - Assigning users to tasks
    - Deleting a task
    - Naming a task



# Connected Class Topics

- Testing
  - What testing we did
    - Functionality
    - User test cases
  - How our testing impacted our future decisions on the design
    - Added the all tasks panel
    - Moving completed items to the bottom
    - Selecting group members and setting the task status for a tasks from a list instead of having to type it in



# Planned Features

## Functionality Features

Login Capabilities/User database

Save/Load existing Task Boards

Online Access

## Additional Features

Customizable interface

Think of a more marketable  
name

AI Integration

Canvas API Integration





Demo

# Use Case: Create a Task

## Preconditions

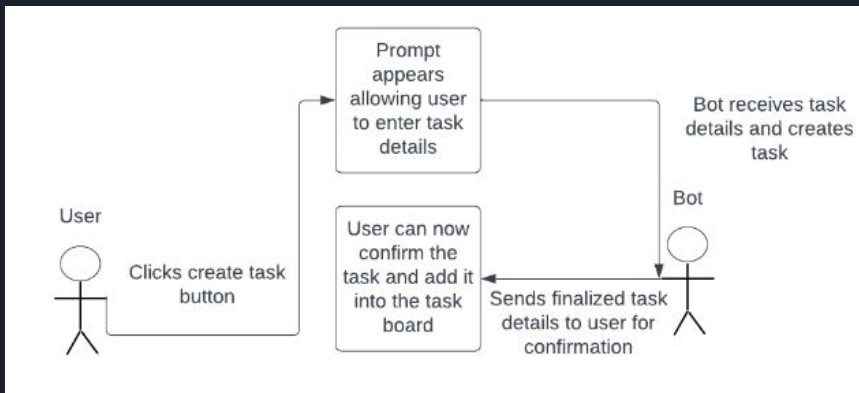
- System must be up and running with 1+ user on view

## Main Flow

- User will click on the create task button and enter task name/description in prompt [S1].
- User will click on the submit task button [S2].
- If task list has reached maximum entries, prompt will display asking to delete one [E1].

## Subflows/Alternative

- [S1] The system opens up a prompt allowing the user to enter the details of the task.
- [S2] The system finalizes the task and presents it on the board of tasks.
- [E1] Board of tasks is full and system cannot add any more tasks. Will ask user to remove task in order to add new one.



# Use Case: Assign Task to user

## Preconditions

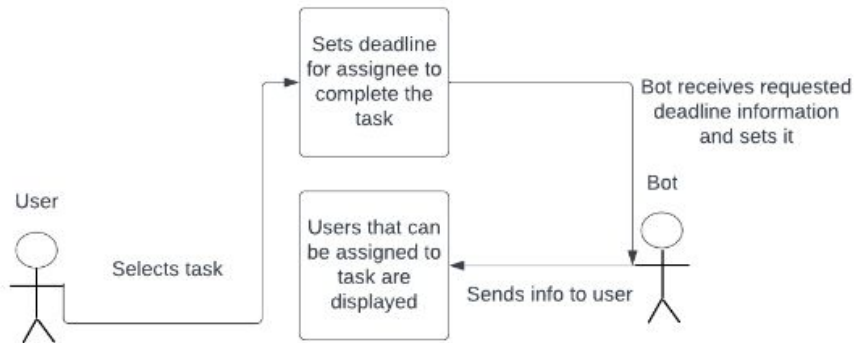
- 1+ task must be created
- At least one user must be on system

## Main Flow

- User will select a task [S1]
- Bot will ask the user for the deadline for the assignee to complete the task [S2].

## Subflows/Alternative

- [S1] Bot displays the users that can potentially be assigned the task
- [S2] User sets the deadline for the assignee to complete the task
- [E1] No users are available to assign a task to.



# Use Case: User can designate task as completed

## Preconditions

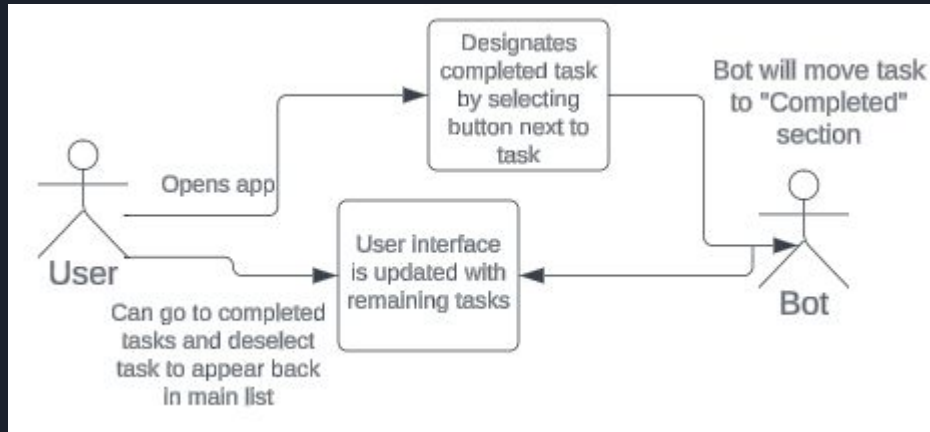
- System must be up and running with 1+ user on view

## Main Flow

- User will click on task checkmark/button to signify completion [S1]
- Upon exiting, bot will move task to separate "Completed" section [S2]

## Subflows/Alternative

- [S1] Bot can accept task completion selection from any member of team
- [S2] User can go to Completed section and deselect task to appear back in main list





# References

Taskboards logo owned by Google: <https://tasksboard.com/img/social.jpg>