Taskify: Collaborative To-do List Tracker/App

Project Overview:

Taskify is a web application designed to help individuals like students and teams manage their tasks efficiently and effectively. It provides a user-friendly interface for creating, assigning, and tracking tasks alongside a timer to keep track of progress. The project emphasizes collaborative features, allowing to keep track of assigned tasks with a timer to keep track of time efficiency.

Learning Objectives:

- Understanding open-source software and its significance.
- Collaboration and teamwork in software development.
- Basic software development principles and practices.
- Collaborative tools usage.
- Documentation and communication skills enhancement.

Project Description:

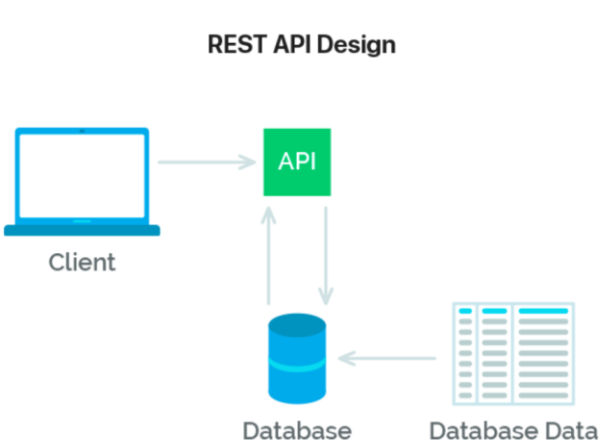
Taskify allows users to:

- Generate tasks with a “date added” attribute.
- Mark tasks as complete.
- View a dashboard with an overview of tasks.
- A functional clock including a timer.

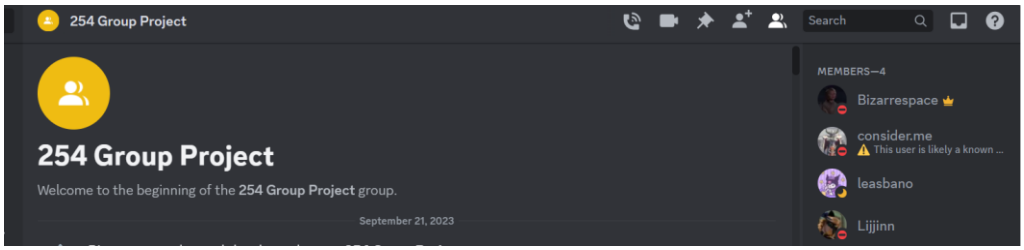
Technology Stack:

- Frontend: HTML / CSS / Javascript
- Framework: React
- Backend: Golang
- Database: Mongodb
  (REQUIREMENTS: 1. Download Golang Latest Version HOW TO USE DB: 1. Download backend and unzip 2. Open CMD 3. Run 'go mod init' 4. Run 'go mod tidy' 5. Run 'go build main.go')
- Collaborative Tools: GitHub

Server Infrastructure:



Collaboration and Teamwork:

1. Utilized Agile practices.
2. Tasks divided based on features and components.
   - Lea: Frontend Tasks
   - Long: Pomodoro Time
     (graphical timer with ability to start, stop, reset, adjust to different purposes & users liking)
   - Shadi: Database & Backend
   - Rilijin: Presentation, Documentation, and Testing/ Formatting (imitating a user)
3. Communication through daily communication and a dedicated discord channel.
   -


4. GitHub used for version control, with pull requests and code reviews for collaboration.

User Interface (UI) and User Experience (UX) Design:

1. Taskify's User interface is designed to contain simplicity.
2. Users can easily create, assign, and manage tasks through a built-in interface.
3. Design focuses on a clean and responsive layout for an optimal user experience.

Open Source Principle:

1. Sharing Documentation:
    - The project adheres to open source principles by publicly hosting documentation on GitHub. This enables widespread access.
2. Collaborative Editing:
    - Collaboration is through tools Git, allowing multiple contributors to work simultaneously on documentation. Edits, additions, and feedback are openly exchanged, promoting transparency between the group.
3. Community Involvement:
    - The project actively invites community participation. Users can report issues, suggest improvements, and contribute to the use of Taskify.
4. Licensing Transparency:
    - By an open source license, defining user rights. This ensures clarity on how Taskify can be used and shared within the community.
    - GNU Affero General Public License (AGPL)
    - GOLANG BY GOOGLE BSD-style license, specifically the 3-Clause BSD License.
5. Clear and Accessible Documentation:
    - Emphasis is placed on maintaining clear and accessible documentation. This aligns with open source principles, empowering users and contributors to engage effectively with the project.

Agile Practices:

1. Daily texting and bi-weekly in class talks to discuss progress, challenges, and resolutions.
2. Changes in project requirements accommodated through repeated development were notified and discuss for clear understanding.
3. Continuous feedback from team members improved overall collaboration.

4. Discussion on confusion and step by step process with everyone involved to understand the whole process as one.

Challenges and Learning:

Challenges:

- Port Allocations
- Cross Origin Resource Sharing (CORS)
- Procrastination
- Configuring database network access permissions, had to be set to universal and not specific IP addresses
- Learning React

Git Repository:

https://github.com/leaalbano/ToDoList.git

Conclusion:

Taskify provided valuable hands-on experience in open source development, collaboration, and Agile practices. The project enhanced the team's understanding of effective communication, adaptability in changing requirements, and the significance of collaborative tools in software development.