**SW Engineering CSC648 Section 1**

Project title: TaskFlow

### Team #4

#### Team Leader: Bisum

#### Other Members: Josh, Zoe, Anisah, Sanna, Jun

#### MILESTONE 3

#### 11/04/2024

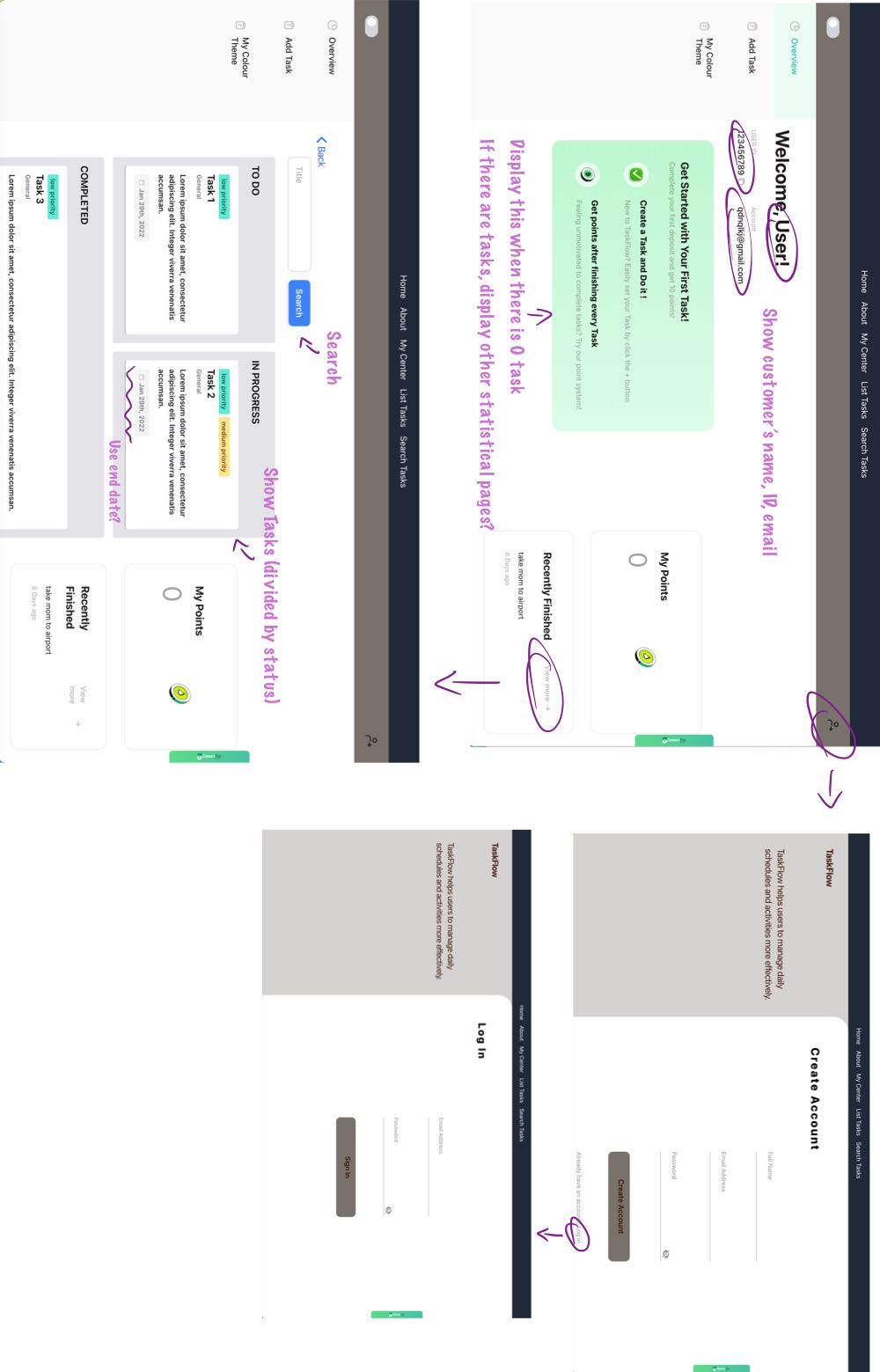
#### App: <https://csc648-01-fa24-csc648-01-fall24-team04.vercel.app/>

Table of contents

1. Overview
2. P1 List for M5
3. Architecture Documentation
4. Project Status
5. Github Organization

1. Overview

* **P1 Features:** Implemented 5 key features: including
  + Add Task: Users can create and add new tasks to the list.
  + Delete Task: Users can delete tasks as needed.
  + Complete Task: Tasks can be marked as completed, updating their status visually.
  + Display Tasks: Tasks are displayed in an organized, user-friendly layout.
  + Color Theme: Users can personalize their experience with customizable color themes that distinguish different priority levels of tasks
* **Software Development Progress:** Our components are integrated, and the major functionalities are operational. The files for our code are organized with separate sections for front-end and back-end to be able to navigate easier
  + Basic Process:
  + 1.UX & Figma Design: Designed the user experience and layout in Figma.
  + 2.Builder.io Integration: Converted each part of Figma designs into code and organized them together.
  + Key Focus Areas:
    - File Structure Organization:
      * Organized frontend files to ensure modularity and maintainability.
      * Structured components to facilitate easy collaboration with backend.
      * Color Theory - using brighter colors for better legibility
        + Using contrasting colors so that user can read website better without struggling



* + - Backend-Ready Design:
      * Focused on designs that simplify backend interactions.
      * Ensured data flow compatibility for seamless API integration.
* **UI and Functionality Review:** Our updated code features our UI and core features, demonstrating the flow and usability, including the defined P1 features
* **Team Collaboration:** Utilizing GitHub branches to merge functionalities together
* **UI Implementation:** The UI follows the UX flow defined in Milestone 2, with the focus on consistency and ease of navigation. We currently have the main pages for our app including task view, an Add Task page with options for priority and category, and color theme customization.

2. P1 List for M5

The following P1 features are committed for delivery by Milestone 5:

1.Task Display Options: View tasks in monthly, weekly, and list formats to streamline task management.

2.Task Management: Easily create, update, delete, and check tasks to keep track of all activities.

3.Point System: Earn points for completing different tasks, motivating users to stay on top of their goals.

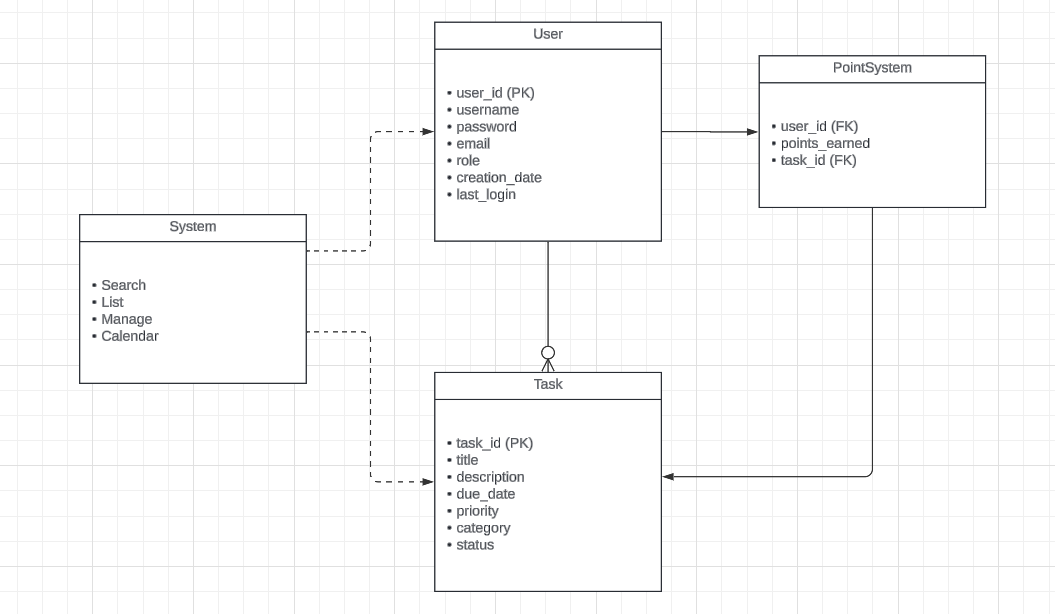
4.Personalized Color Theme: Enhance the interface with customizable color themes for a more engaging, personalized experience.

5.Task Statistics: Display user task statistics to provide insights into task completion and progress.

6.Collaborative Tasks (Optional): If time permits, enable collaboration by merging tasks among users.

These features are our primary focus and will form the core of the final product + potentially extra features if we have time

3. Architecture Documentation

* **High-Level Architecture and Database Organization:**Our MongoDB database is organized into separate collections for users, tasks, and notifications. Key tables include:
  + **User Table:** Stores user profiles, including ID, role, and login details.
  + **Task Table:** Contains details such as task title, due date, priority, and status.
  + **Notification Table:** Manages reminders and alerts associated with tasks.
* **UML Class Diagram:**

4. Project Status

* **Technical Risks:** Risks have been identified, and we are actively mitigating them.
  + **Schedule Risk:** Balancing schoolwork and schedules have impacted our timelines. Mitigation**:** Weekly mini-deadlines keep us on track and options for zoom calls for when others can’t meet (ex. Commuting or busy)
  + **Teamwork Risk:** Coordination challenges can arise due to busy schedules. Mitigation: Frequent meetings and group chats maintain alignment.
  + **Technical Skill Gaps:** Learning new tools like MongoDB and Next.js. Mitigation**:** Team members are completing tutorials and collaborating on challenges.
  + **DB not accessible to everyone**: Changed permissions and solved issue by adding manually
  + **Organization:** Not understanding the folders/file organization, got help from frontend to understand where everything is and re-organized file system structure in our code helping to distinguish frontend and backend functionalities
  + **Issue with pulling from the right branch**, being mindful of checking that you're on the right branch and pull before working on anything

5. Git/Github Organization

* Branching and Code Review Practices:
  + We use a ‘dev’ branch alongside individual ‘feature’ branches for each functionality.
  + Code Reviews: All team members meet to review the code in pull requests together, resolve merge conflicts, and then push the changes.