

COMP3900 21T2

Task Master

COMP3900-T18A-Cooders

Background/Problem Overview	2
Competition Analysis	3
User Stories	4
Sprint Schedule	5
Objective satisfaction	5-6
Novel Functionality	6
Software Design and Architecture, User Flow	7-9
Storyboards	11-19
References	20

Lyric Wang	Claude Randall	Josh Mclellan	Sean He	Lachlan Pierce
z5260932	z5161197	z5163204	z5117101	z5194984
Scrum Master, Developer	Developer	Developer	Developer	Developer
lyricwang1901 @gmail.com	mrclauderandall @gmail.com	joshjgmclellan @gmail.com	seanhe514 @gmail.com	lach.p0 @gmail.com

1. Background

As work becomes increasingly rigorous, not only in tech but in all sectors of business, organization and efficient use of one's working day is more important than ever. The app we are building allows users to both organize themselves and delineate tasks to colleagues, as well as connect to other users as a 'friend'. As such, it is appropriate for, among others:

- Business managers
- Students and teachers
- Parents seeking to formalize household chores
- Secretaries/clerks
- Industrious laypeople

Problem Overview

Whilst there is no shortage of project and team management software or timetabling and checklist apps, there is a clear gap between the individual focused scheduling apps and the large scale project management apps. Almost all task management tools are designed with a large team in mind, completely ignoring the needs of smaller teams or individuals. Simultaneously, current organization tools such as traditional calendars are insufficient in supporting modern teamwork strategies. There are of course many checklist and personal organisation tools out there, however none that can integrate with a team. Our product aims to address these market gaps, through focusing on individual and team scalability and integration. This gap is particularly large with the huge number of startups and small teams that exist, all of which will at some point need to shift their attention as their businesses grow and teams increase in size. On top of this, social-networking features are underrepresented in the majority. Organization tools often function ancillary to other tools with social-networking features, but none have integrated them into the task management itself.

Competition Analysis

Product	Description	Advantages	Disadvantages
Trello	Atlassian's Kanban style project management tool	<ul style="list-style-type: none"> • Simpler and easier to use UI • Good for people working on their own • Multiple languages, great for international teams 	<ul style="list-style-type: none"> • Not great for larger teams • No Gantt Chart • Not as many features
Asana	Simpler task management and team member connection	<ul style="list-style-type: none"> • Good Milestone tracking • Quick set up • Better for smaller teams 	<ul style="list-style-type: none"> • Less customisable • Good automation options • Not optimal for single users • Steep learning curve • Best team member interaction
Monday	task management and time tracking web app designed with remote teams in mind	<ul style="list-style-type: none"> • Very customizable • Good UI • 2 to 1000 team members • More visualisation options 	<ul style="list-style-type: none"> • More complex and more 'clicks' required to do things • Not optimal for single users • Not as much third party integration

Sprint schedule

Sprint	Dates	Epics	Scope (CTC)
1	16/6 - 27/6	Backend: User Profile, Task Creation	7, 20, 21, 22, 23, 24, 29
2	14/7 - 25/7	Front end: User Interaction, Task Interaction	8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 25, 28

Objective satisfaction

Objectives	User Stories (CTC)
Task masters must be able to maintain a profile where they specify their name and contact details (email address).	7, 20
Each task master must be able to request a connection with 1 or more other task masters (by email address), and also accept or decline connection requests from other task masters (two task masters are connected if one of them accepted a connection request from the other).	8, 9
Taskmasters must be able to create a task, including a title, description, optional deadline, and assign it to either themselves or one of the task masters that they are connected with (each task must also have a system-assigned id that is shown, and is to be assigned to the creating task-master if not assigned to anyone else).	3, 21, 22, 23, 24, 29
In order to view the work to be done by a task master, their profile must also include an "Assigned Tasks List", showing all of their assigned tasks (with each task showing summary details of the task ID, title, and deadline (if specified)), sorted by deadline (earliest to latest deadline, with tasks not having a deadline appearing last).	10, 11, 12
Each task master must be able to view details on the profiles of any other connected task masters.	28
In order to progress the state of a task, a task master must be able to update the status of a task they have created or that they have been assigned (valid task states include "Not Started" (start state for new tasks), "In Progress", "Blocked", "Completed").	17, 18, 19
Over time, the number of tasks existing in the system will build up, so it must be possible for a task master to search, (through all tasks assigned to	13, 14, 15, 16, 31

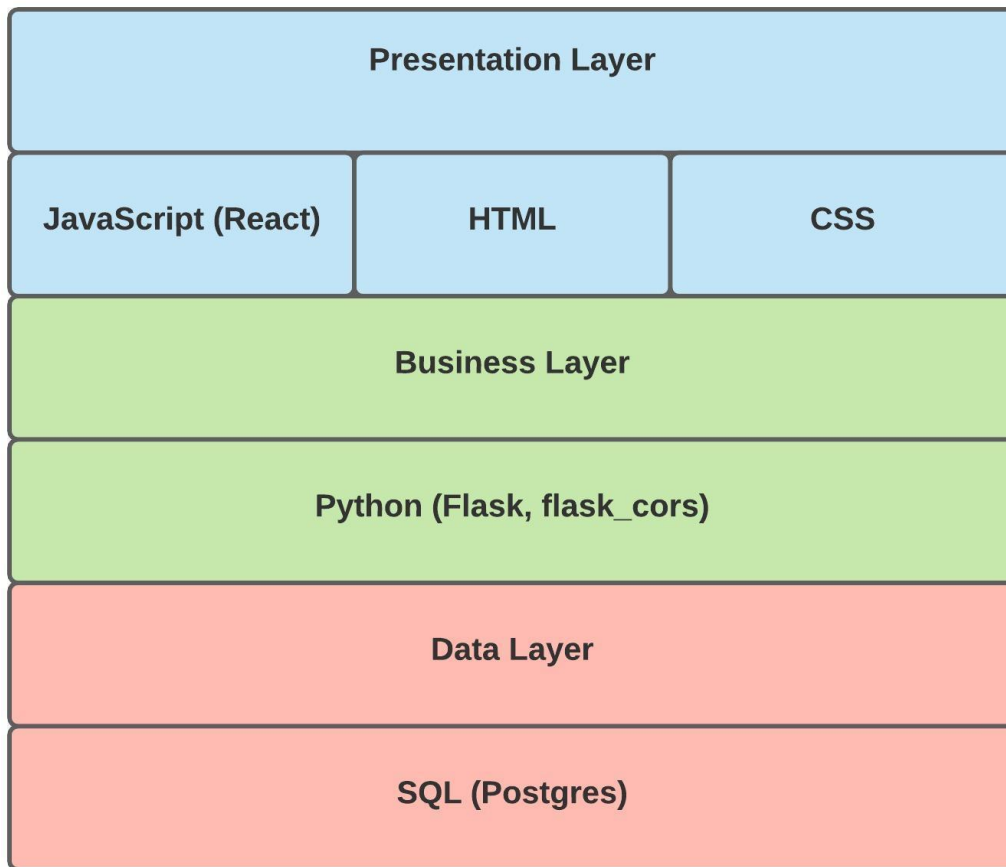
themselves or any task master they are connected to), by any combination of id, name, description and/or deadline, and view its full details.	
The system must be able to show a task master an estimate of how busy each of their connected task masters will be over the next week, (0% (min possible) through to 100% (maximum possible), or "100%+ (overloaded)" for any task masters it thought were overloaded), where this value can based on a combination of their assigned tasks, task states, task deadlines, how long similar tasks have taken to complete in the past, and/or any other variables that you wish to use/introduce for the purposes of estimating how busy each task master is.	25

Novel Functionality

While most team sharing products are built around rigid teams where members are added and removed by a central scrum master, our system is dynamic in that any taskmaster can connect with any other task master. This allows our product to be more scalable and dynamic for a changing work environment and changing team. The ability for all task masters to control which task masters they connect with and share tasks with is the main feature separating us from the competition, making our product more flexible than the others. On top of this, we also have a few additional functionality that is also not present in the competition;

- **Bump Feature:** allows the creator of a task to send push notifications to all other Task Masters that have been assigned that particular task, thus highlighting its urgency. This will mean that the task will be specifically highlighted on each Task Master's dashboard.
- **Task/Subtask Containers:** Allows the ability to create tasks that are made up of tasks, allowing for a better understanding of the scope that a particular task functions within.

3. SOFTWARE DESIGN AND ARCHITECTURE



Presentation Layer

We decided to use the JavaScript library of React for our frontend for a couple of reasons. Firstly, some members of our team were familiar with the language and library already, thus reducing the amount of learning we would need to do. Furthermore, React as an environment allows for quick renders and updates code changes quickly, simplifying the process of developing within it. Lastly, React Components are beneficial given our project, for example it allows us to develop a Task Component then reuse it easily. The fact that it also allows for HTML/CSS design is another positive. Task masters will interact with the system purely through the presentation layer.

Business Layer

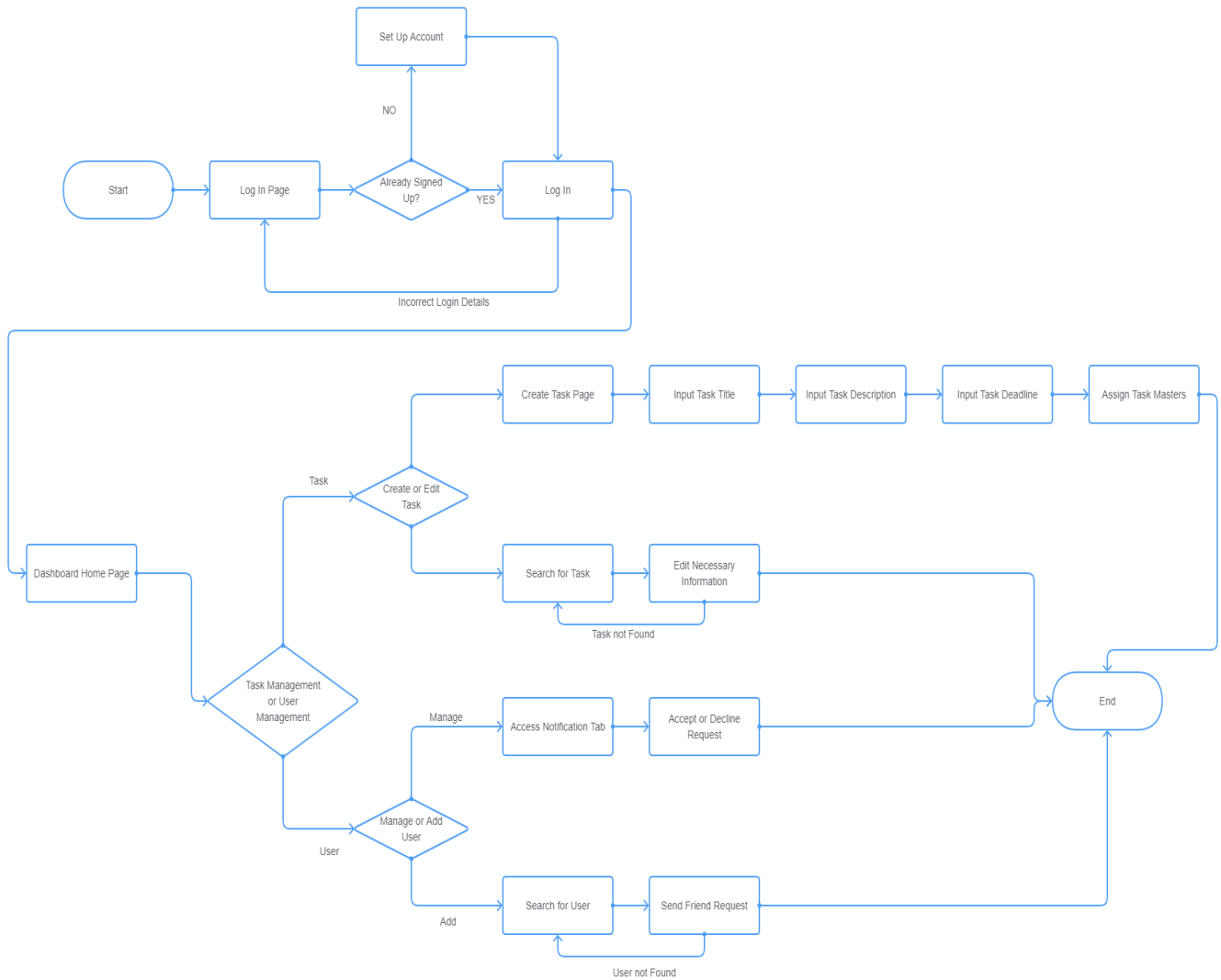
For hosting the backend, we decided to use Python, specifically the Flask library. Again, this was because most of the members of our group had extensive experience in both Python and Flask, and thus it makes it very easy for us to get a simple backend running in a short amount of time, which we can wrap the frontend over. In terms of third-party libraries and APIs we are using in this layer, currently we are using the psycopg2 library to handle requests to the Postgres database.

Data Layer

For managing data related to the application, we settled on SQL, namely PostgreSQL. We had decided that we wanted to structure the data in a database with multiple tables, and a couple of group members had experience with Postgres, and its scalability and search functions were also highly desirable.

User Flow Diagram

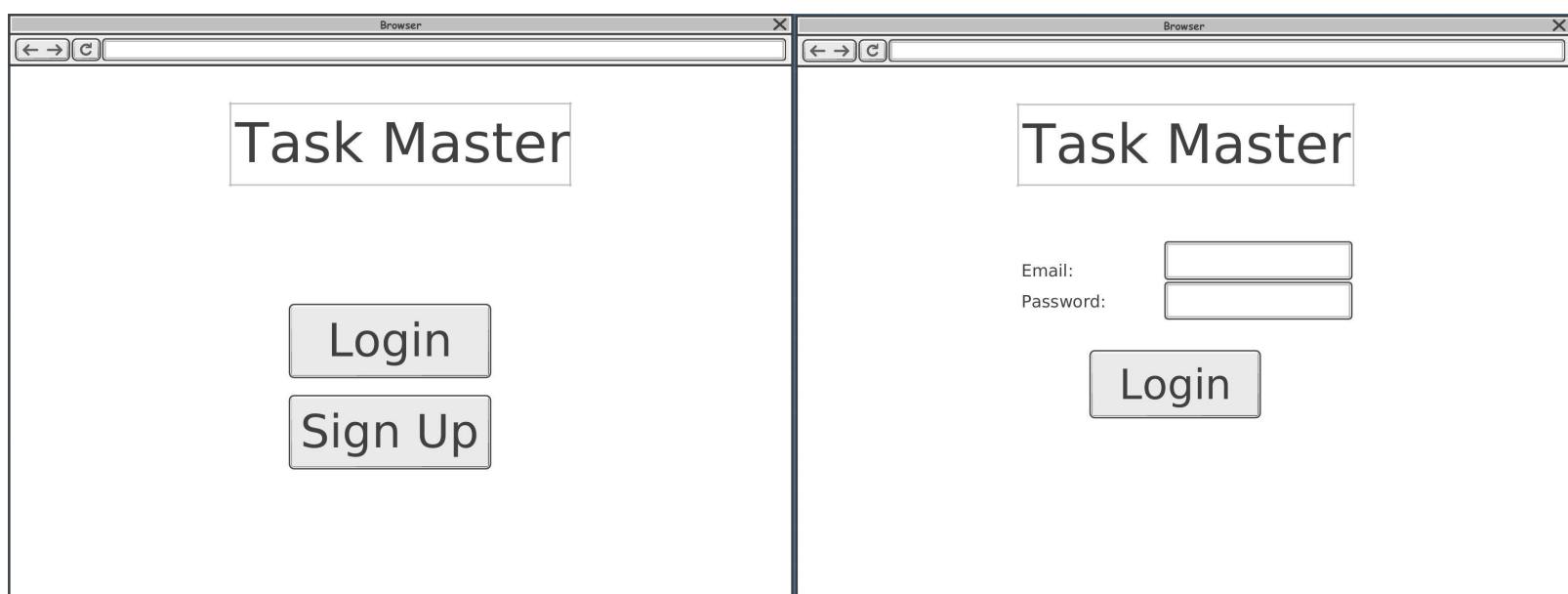
Task Master



11 June 2021

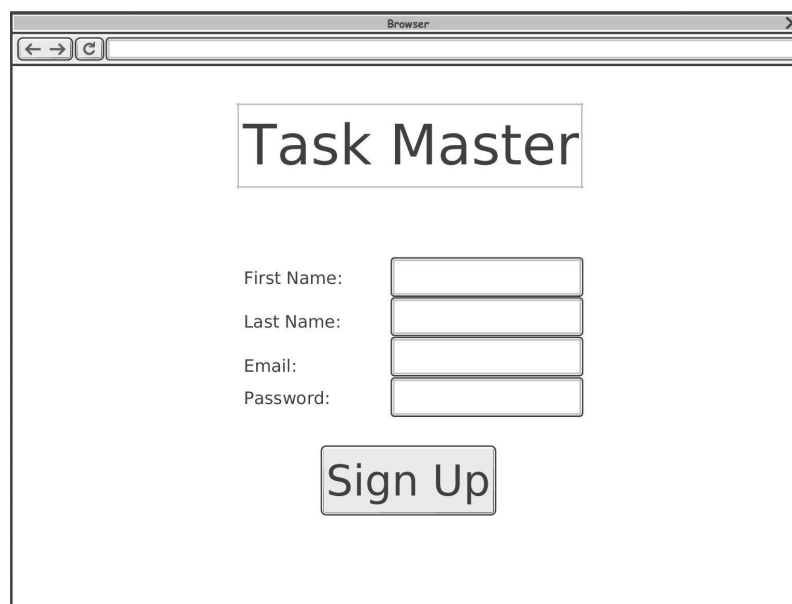
4. STORYBOARDS

1. Login/Signup



(Landing Page for users)

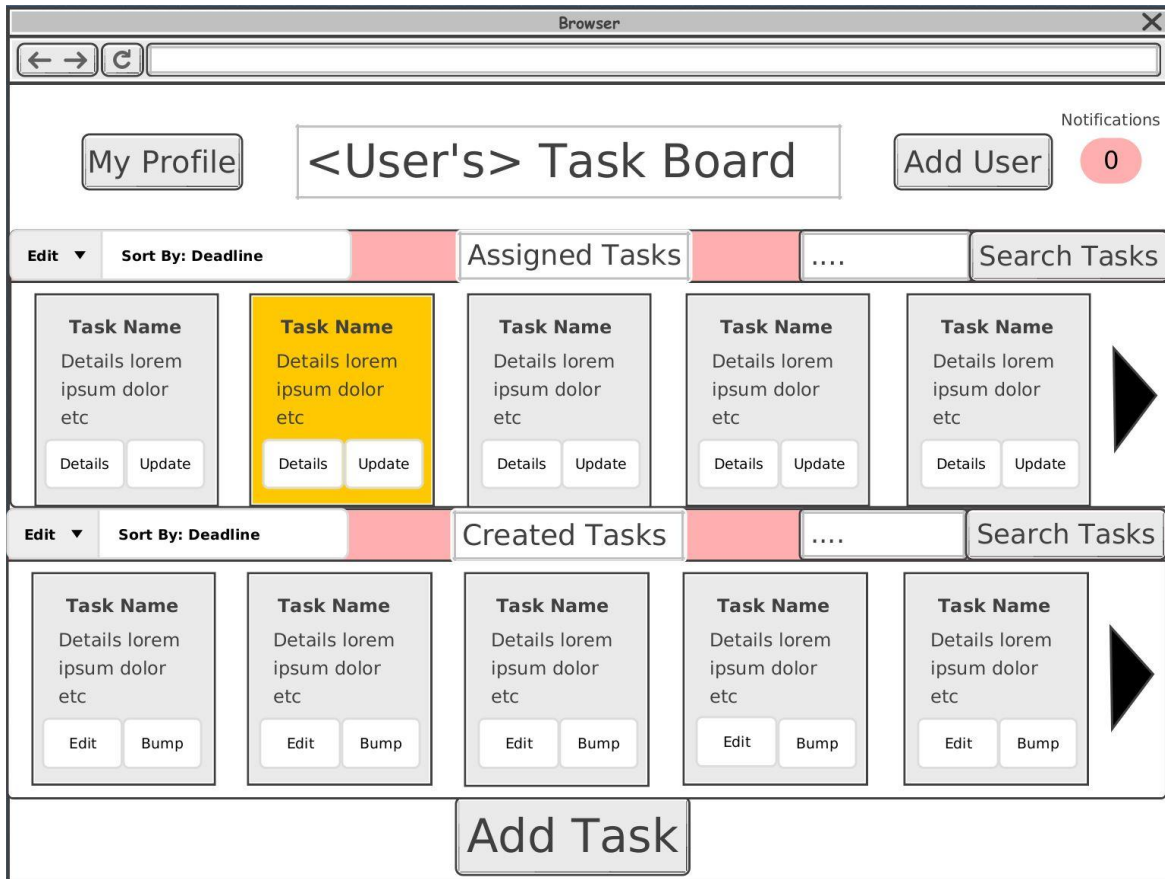
(Login Page)



(Sign up form for new users)

This section is fairly straightforward. It covers the user stories relating to login/signup of users, and thus we envision a simple landing page to catch both cases.

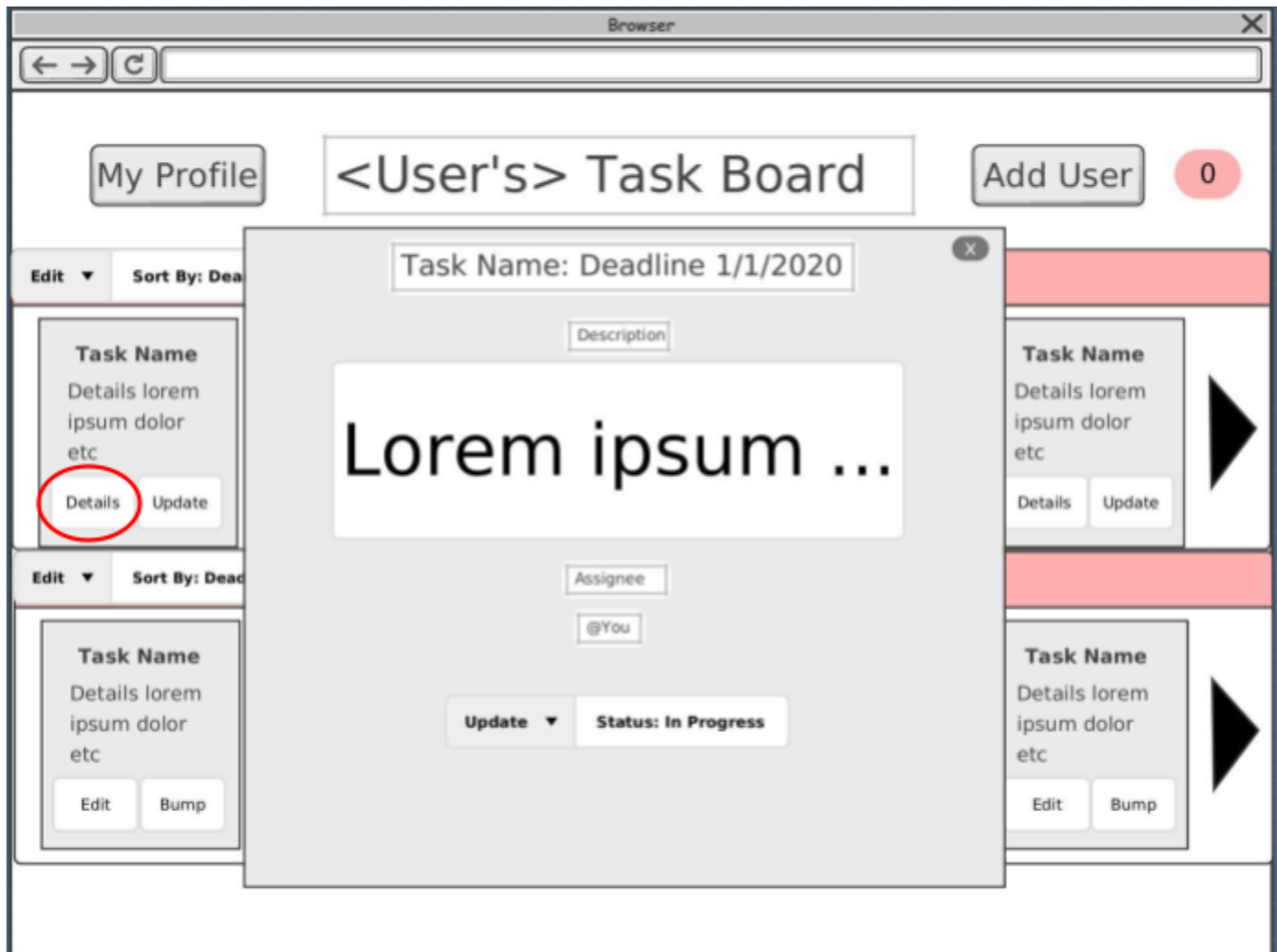
2. User Dashboard (Task Board)



Effectively the home page, this provides the user with the ability to:

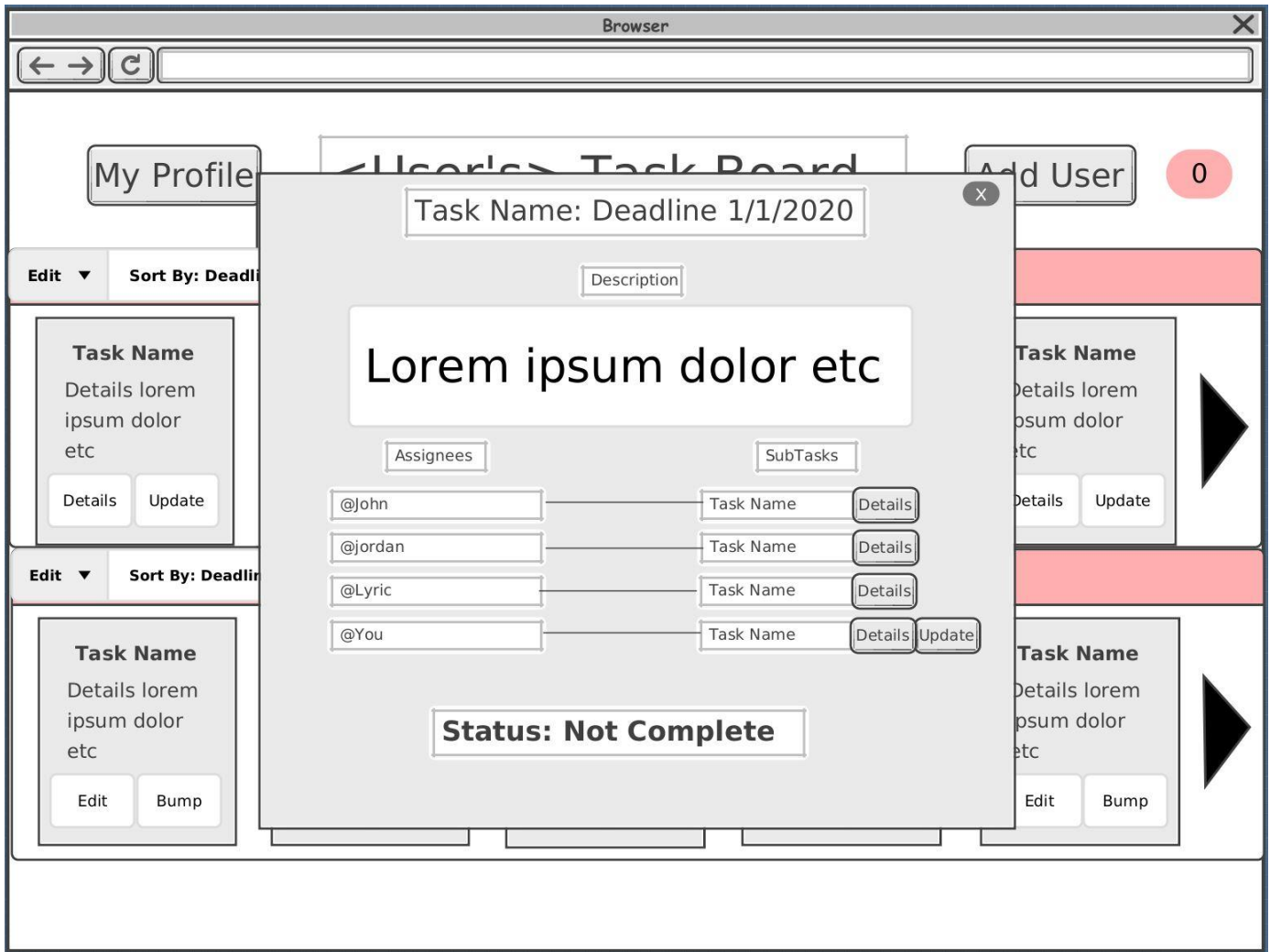
- View tasks that they have been assigned, with the option to view more details about that task or update its status. The task that has been highlighted is a task that has been bumped, and thus there is a visual change to highlight its importance. Tasks are portrayed in a horizontal scrolling manner. Tasks can be sorted by deadline, or searched by task title/description.
- View tasks that they have created, so that they can easily be accessed to be updated or bumped.
- Initiate the creation of a task
- Initiate connection with a user
- View/edit their profile
- View notifications: the notification badge will update automatically when new notifications appear, and will display a small popup of that notification

2.1. Viewing Task Details



By clicking on the “Details” button of a particular task from their Task Board (highlighted in red), a user can view the task details of that task. This includes the Description, the assignee, as well the ability to update the Status of that task.

2.2. Viewing Task Details with Sub-Tasks



If a task has multiple subtasks, then pressing the “Details” button will display a slightly different pop-up, in which the user can view which users are connected to the sub-tasks that make up that particular task, as well as the option to view their “Details”, and update the status of the one/s that they have been assigned.

3. Creating a New Task

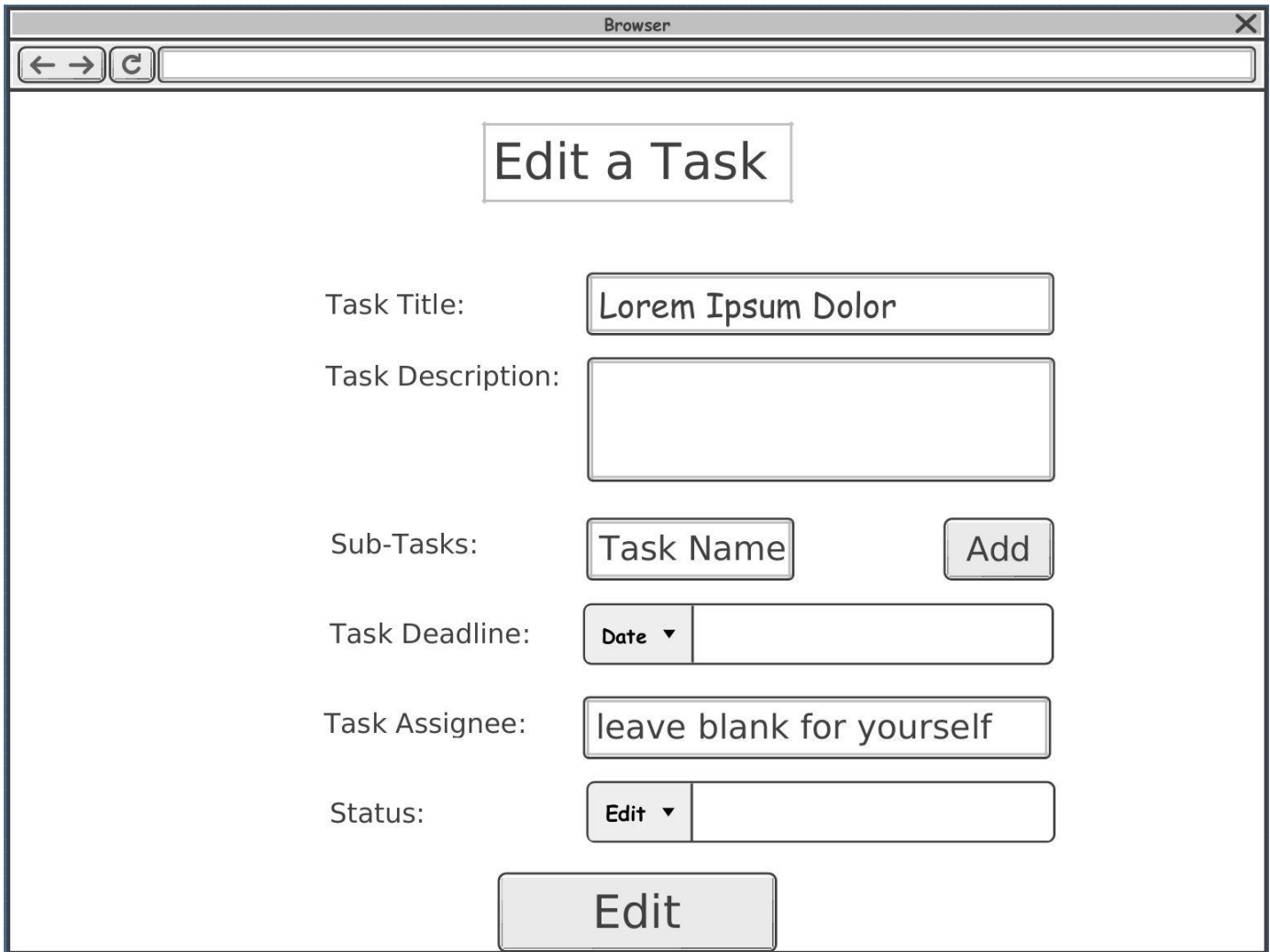
The screenshot shows a web browser window with the title 'Browser'. The main heading is 'Create a New Task'. Below the heading, there are several form fields:

- Task Title:** A single-line text input field.
- Task Description:** A multi-line text area.
- Task Deadline:** A date selector with a dropdown menu labeled 'Date' and a text input field.
- Task Assignees:** A text input field containing the placeholder text 'leave blank for yourself'.
- Sub-Tasks:** A section with a text input field labeled 'Task Name' and a button labeled 'Add'.

At the bottom of the form is a large, prominent button labeled 'Create'.

By pressing the “Add Task” button on the user’s task board, the user will be routed to a new page, in which they can create a new task. They are able to specify the Title of the task, its description, a deadline, the assignee (the option for self-assignment is also made clear), as well as the ability to add sub-tasks if they wish. Adding a sub-task will route to a new form of the exact same layout etc. Pressing Create routes the user back to their dashboard/back to the main Task if they created a sub-task.

4. Editing an Existing Task

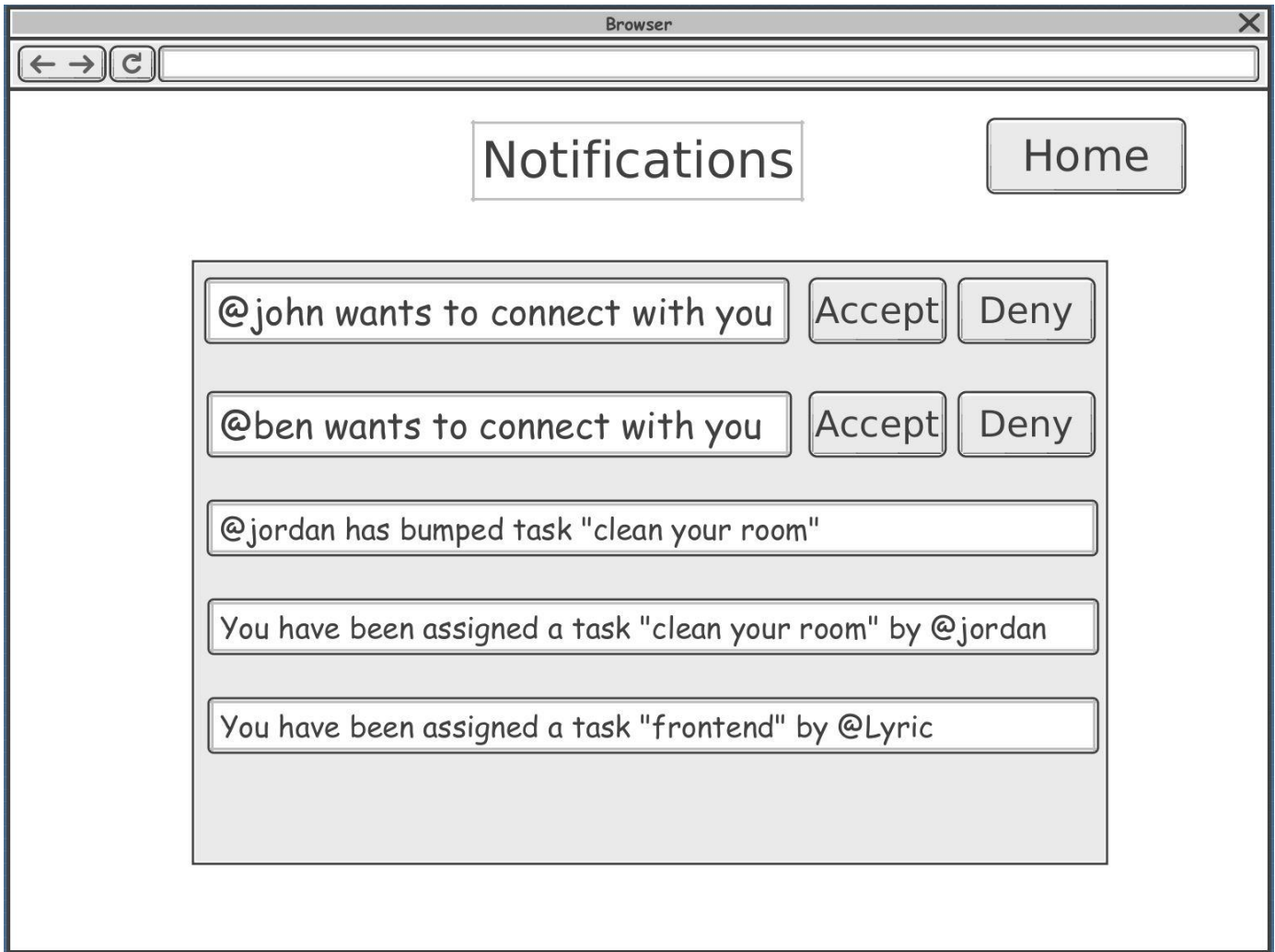


The screenshot shows a web browser window titled "Browser" with a standard address bar. The main content area displays a form titled "Edit a Task". The form includes several input fields and buttons:

- Task Title:** A text input field containing "Lorem Ipsum Dolor".
- Task Description:** A large, empty text area.
- Sub-Tasks:** A section with a text input field labeled "Task Name" and a button labeled "Add".
- Task Deadline:** A date picker control showing "Date" with a dropdown arrow and an empty input field.
- Task Assignee:** A text input field containing "leave blank for yourself".
- Status:** A dropdown menu showing "Edit" with a dropdown arrow and an empty input field.
- Edit Button:** A large button at the bottom of the form labeled "Edit".

By pressing the "Edit" button from the "Created Tasks" section of a users Task Board, they are able to edit the details of an existing task. This includes the title, description, deadline, assignee, status as well as the option to add sub-tasks. The process of adding a sub-task is the same as mentioned above.

5. Viewing Notifications



By pressing on the notification badge at the top right of the users Task Board, they are able to view their Notifications. This includes any other Task Masters who want to connect to the user, as well as Bump notifications, and task assignment notifications.

6. Editing Profile

The image shows a web browser window titled "Browser". The address bar is empty. The main content area displays a "My Profile" page. At the top, there is a "My Profile" button and a "Home" button. Below these, there are three input fields labeled "First Name:", "Last Name:", and "Password:". At the bottom, there is a large "Edit" button.

By pressing the “My Profile” button on the users Task Board, the user will be routed to a new page which allows the user to edit their personal details. Right now, this is just standard personal information and password changes.

7. Connecting to Task Masters

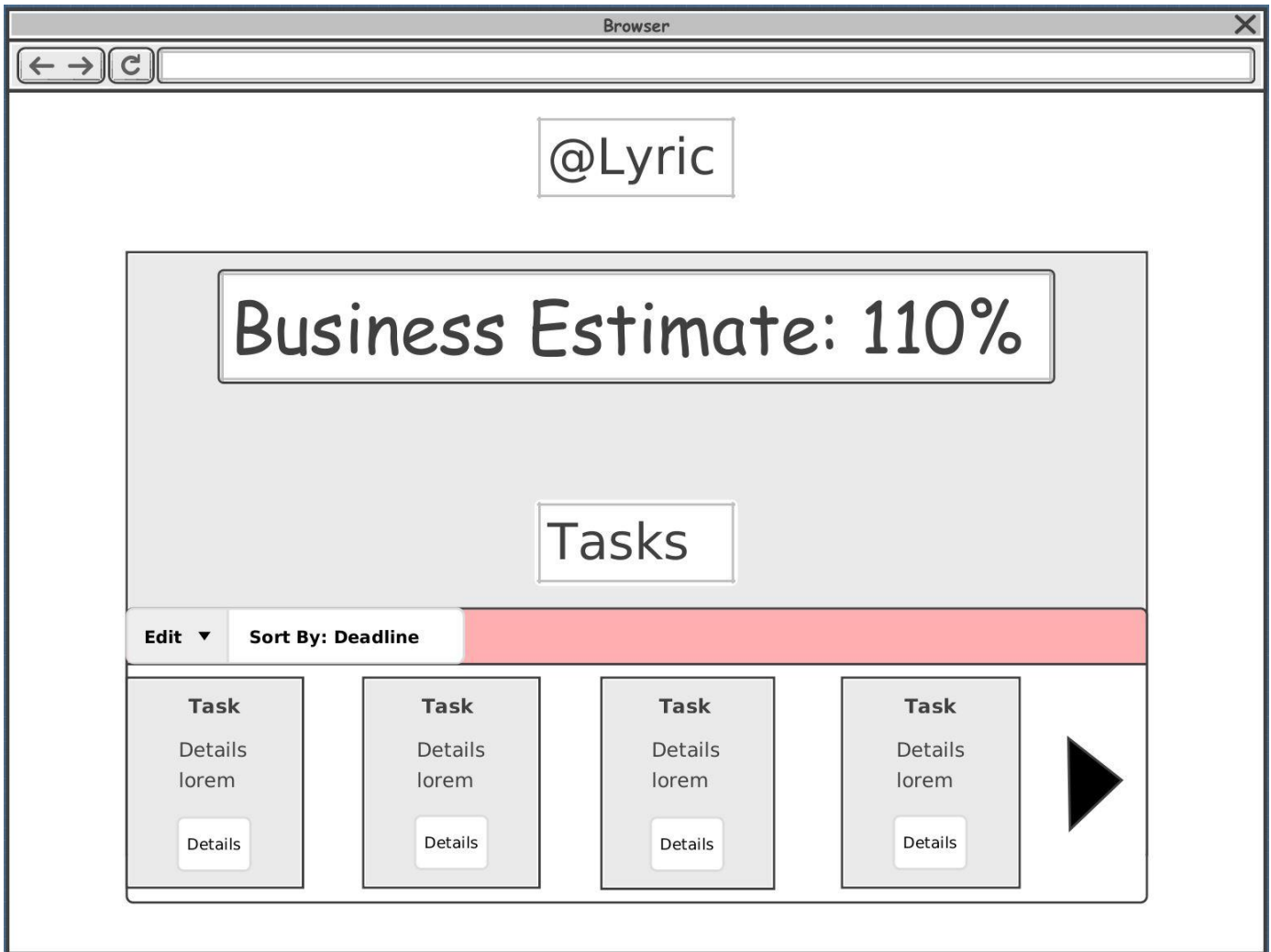


The image shows a web browser window with a title bar labeled "Browser". Inside the window, there is a form with the following elements:

- A button labeled "Add a User" at the top center.
- A label "Enter User's Email:" followed by a text input field.
- A button labeled "Send Request" below the input field.

By pressing the “Add User” button on the users Task Board, the user will be routed to a new page from which they can add other Task Masters. Currently, this relies on the user knowing the email address of the Task Master that they want to connect to. Once they press “Send Request”, they will be routed back to the Task Board, and the Task Master that they sent the request to will be notified.

8. Viewing other Task Master's Profiles



By pressing on the name of a Task Master, via the task details popup, the user can view the details of that Task Master. This includes an estimate of how busy they are, as well as a similar horizontal scrolling representation of the tasks that they have had assigned to them.

5. References

Stewart Gauld. (2020, August 11). How to use Monday.com | Day-to-day Project Management (Monday Tutorial for Beginners). YouTube. <https://www.youtube.com/watch?v=piDQKQmr2H4>

HOW TO USE ASANA | Asana Tutorial for Beginners (Project Management Software) 2021. (2020, June 19). YouTube. <https://www.youtube.com/watch?v=iKsO9zx9n2Q>

HOW TO USE TRELLO | Project Management Software for Beginners (Trello Tutorial). (2020, April 21). YouTube. <https://www.youtube.com/watch?v=WlzWKP0Dp3E>

Francis, F. (2021, February 23). Asana vs Monday vs Trello: A Comprehensive Comparison Guide. SoftwareSuggest Blog. <https://www.softwaresuggest.com/blog/asana-vs-monday-vs-trello/>

J. (2020, December 31). Monday vs Asana vs Trello - Compared 2021. Spacehop. <https://spacehop.com/monday-vs-asana-vs-trello/>