Task Management System

Web - Application

```
  padding: 0;
  font-size: sans-serif;
  background: url(Cyber.jp
  background-size: 100vw 10
}
.box{
  position: absolute;
  top: 50%;
  left: 50%;
  transform: translate(-50%,
  width: 400px;
  padding: 40px;
  background: rgba(0, 0, 0,
  box-sizing: border-box;
  box-shadow: 0 15px 25px r
  border-radius: 10px;
}
.box h2{
  margin: 0 0 30px;
  padding: 0;
  color: #fff;
  text-align: center;
}
.box h3{
  margin: 0 0 10px;
  padding: 0;
  color: #fff;
  text-align: center;
}
.box .inputBox{
  position: relative;
}
```

Name : Jaimin Gajjar

Manage all your task in one place!

Cloud-Based Task Manager

TABLE OF CONTENTS

01 **Abstract:**
Project overview and tech used

02 **Introduction:**
Purpose and problem statement

03 **Basic Overview:**
Quick system summary

04 **Project Description:**
How the system works

05 **Flowchart:**
Workflow representation

06 **Tech Stack:**
Tools and technologies used

07 **Implementation Details:**
System architecture

TABLE OF CONTENTS

08 **Key Benefits:**
System advantages

09 **Screenshots and Images:**
UI/UX visuals

10 **Code Snapshots:**
Key code highlights

11 **Merits and Demerits:**
Pros and cons

12 **Features:**
Main functionalities

13 **Short Explanation:**
System in brief

14 **Conclusion:**
Final thoughts

01. Abstract

The Task Management System is a full-stack web application built using the MERN stack (MongoDB, Express.js, React.js, Node.js). It is designed to help teams manage tasks efficiently through features like task assignment, deadline tracking, real-time updates, and role-based access control. This project showcases the power of modern web technologies in solving real-world team coordination challenges.

02. Introduction

The Task Management System is aimed at helping teams and individuals organize, assign, and monitor tasks efficiently. With the increasing need for remote work and distributed teams, having a central tool to manage workflow becomes essential.

03. Basic Overview

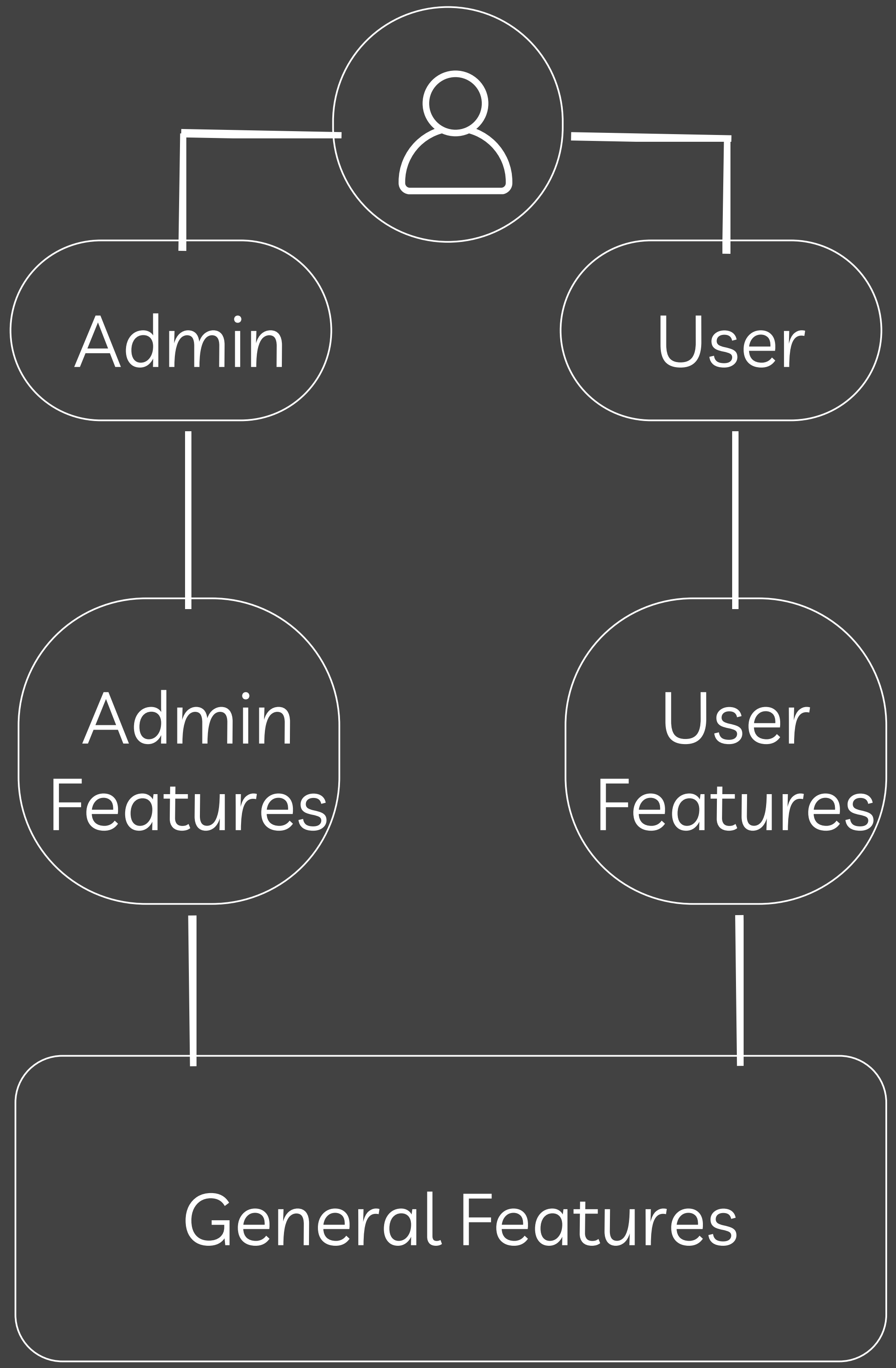
- A responsive web app built on MERN.
- Users can create, assign, edit, and delete tasks.
- Tasks have statuses (e.g., To-do, In Progress, Completed).
- Real-time updates, secure login, role-based access, and detailed analytics.

04. Project Description

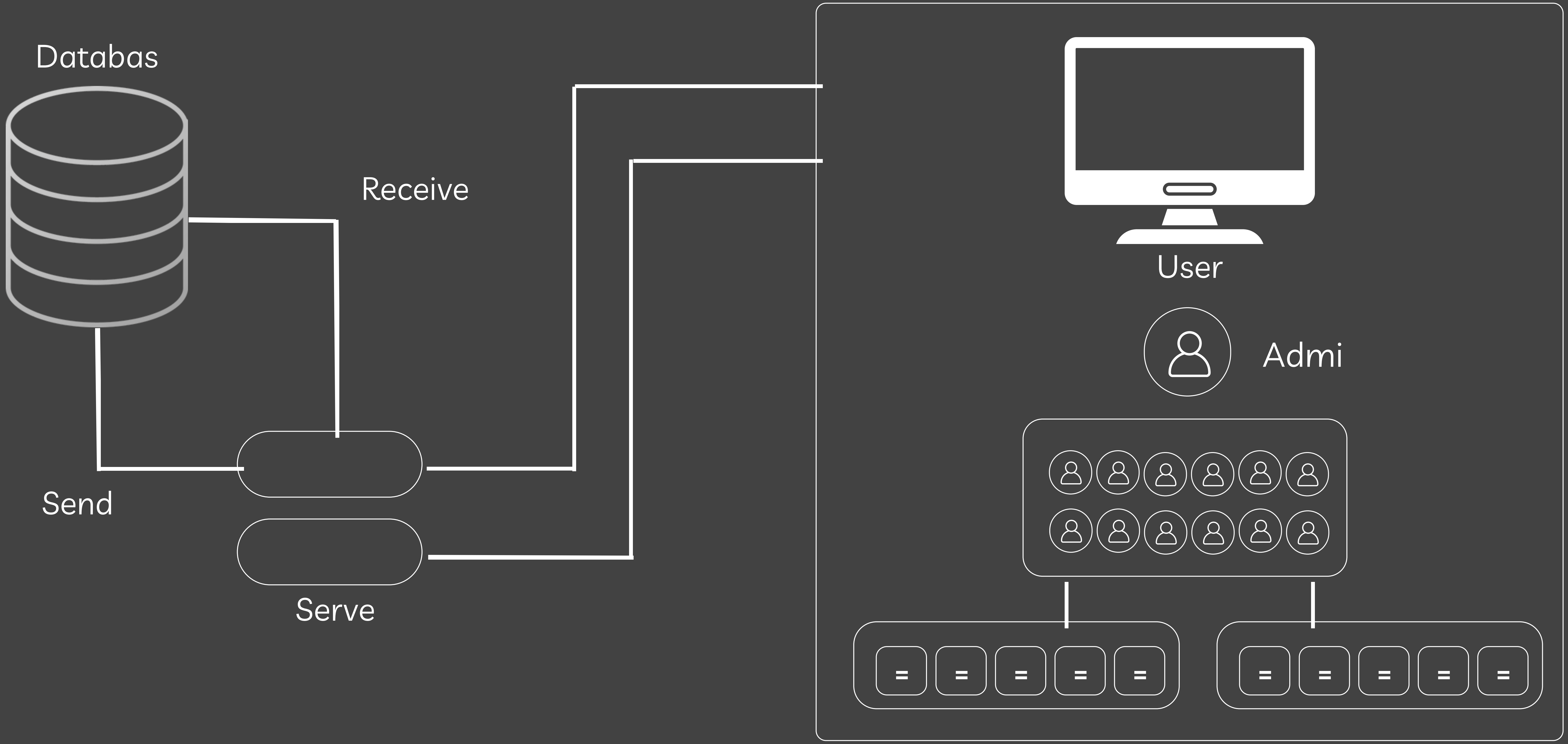
This system allows for:

- Creating user accounts and logging in securely.
- Assigning tasks with deadlines and priority levels.
- Monitoring task progress using visual indicators.
- Collaborating via comments and real-time updates.
- Admins controlling access with permissions.

05. Flow Chart - 1



05. Flow Chart - 2



06. Tech Stack

- Frontend: React.js
- Backend: Node.js, Express.js
- Database: MongoDB
- Other Tools: JWT (authentication), Redux (state management), CSS/Tailwind (styling), Mongoose (ODM)

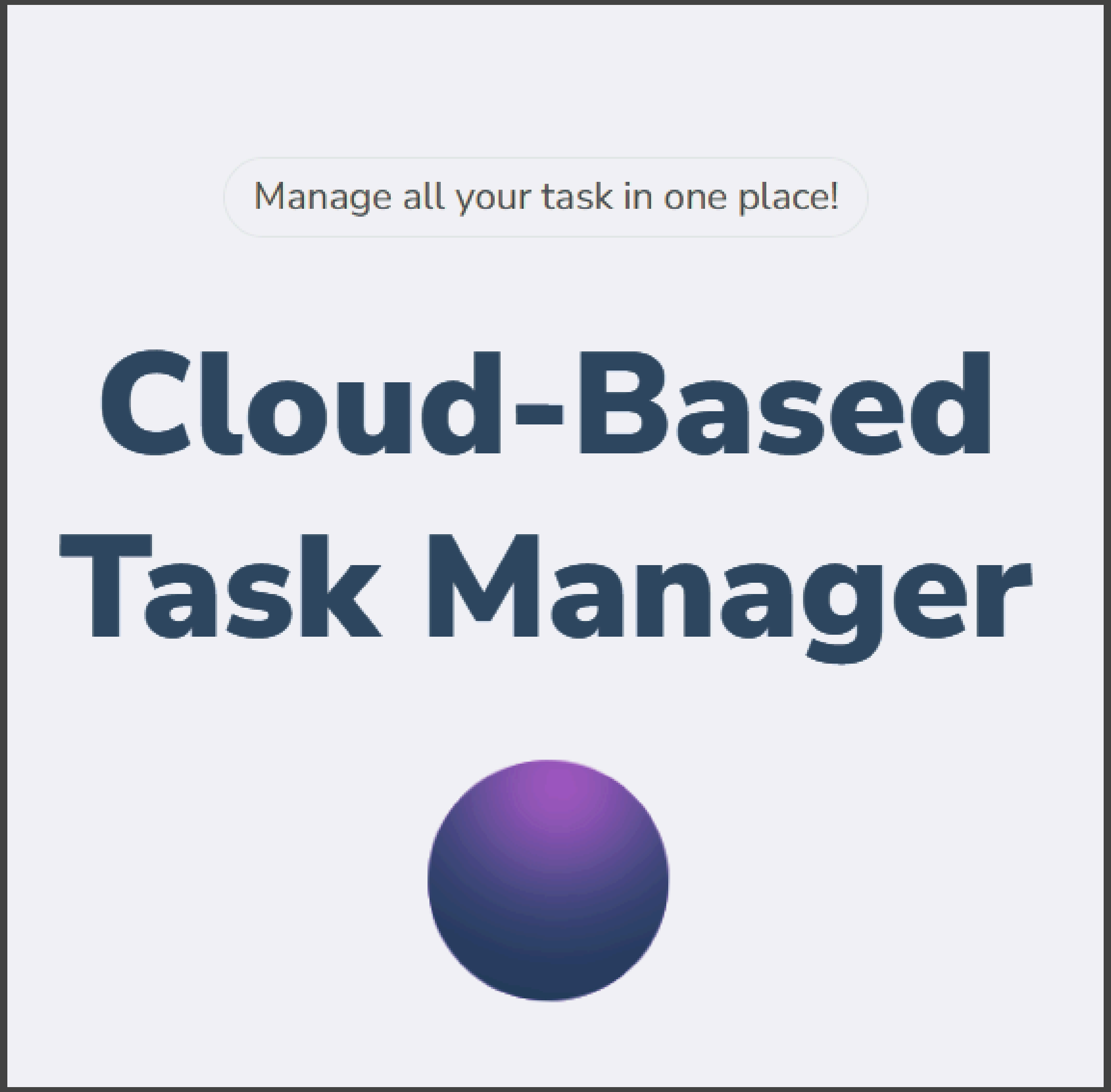
07. Implementation

- Frontend uses React components for a modular design.
- Backend API routes handle task CRUD operations and user authentication.
- MongoDB stores user data, task details, and activity logs.
- Authentication is handled with JWT tokens.
- Authorization ensures different access levels (Admin, Editor, Viewer).

08. Key Benefits

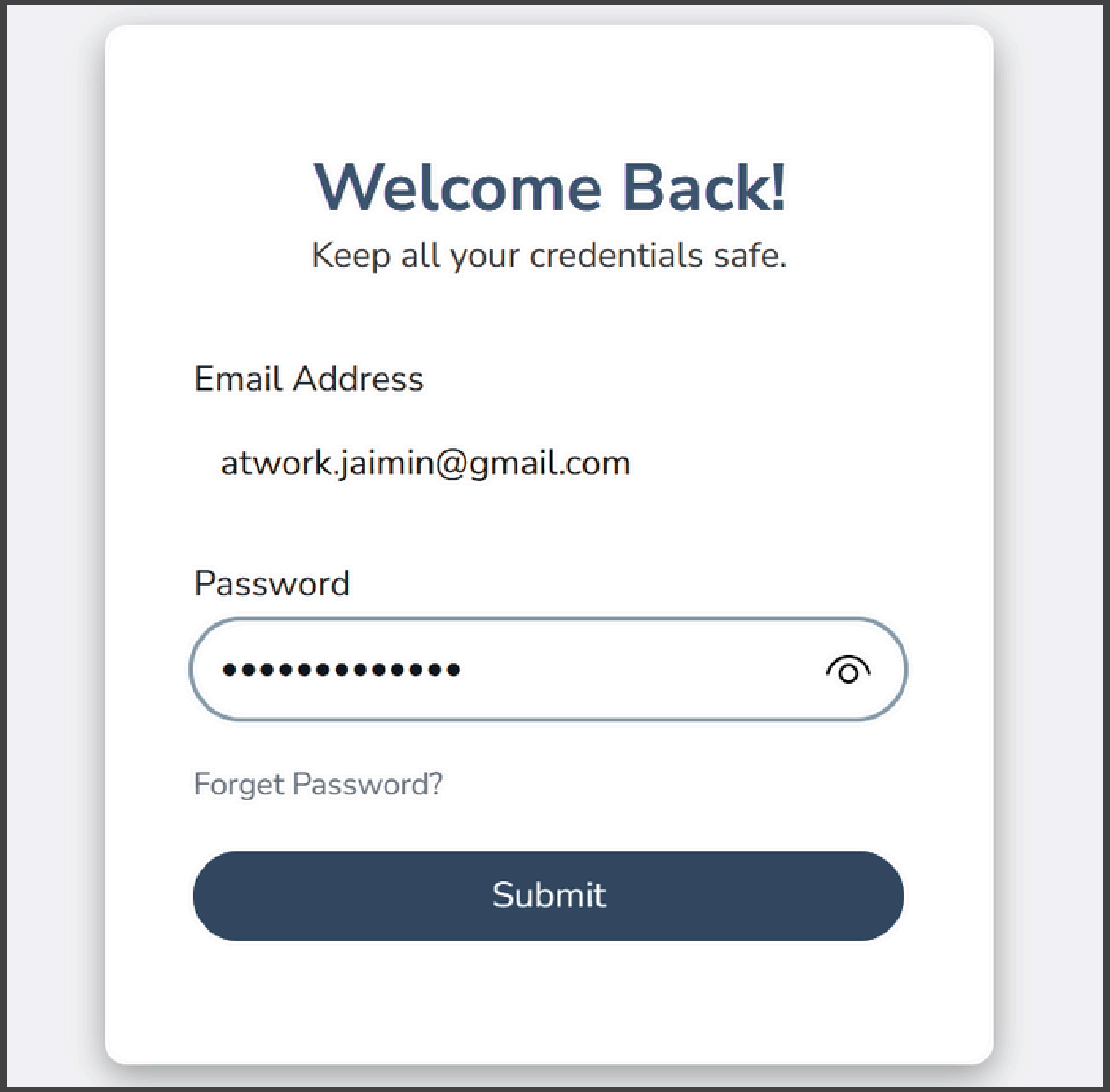
- Streamlined Task Workflow
- Real-Time Updates
- Role-Based Access Control
- Responsive UI
- Secure Data Handling
- Collaboration in One Place

09. User Interface - 1



Manage all your task in one place!

Cloud-Based Task Manager



Welcome Back!

Keep all your credentials safe.

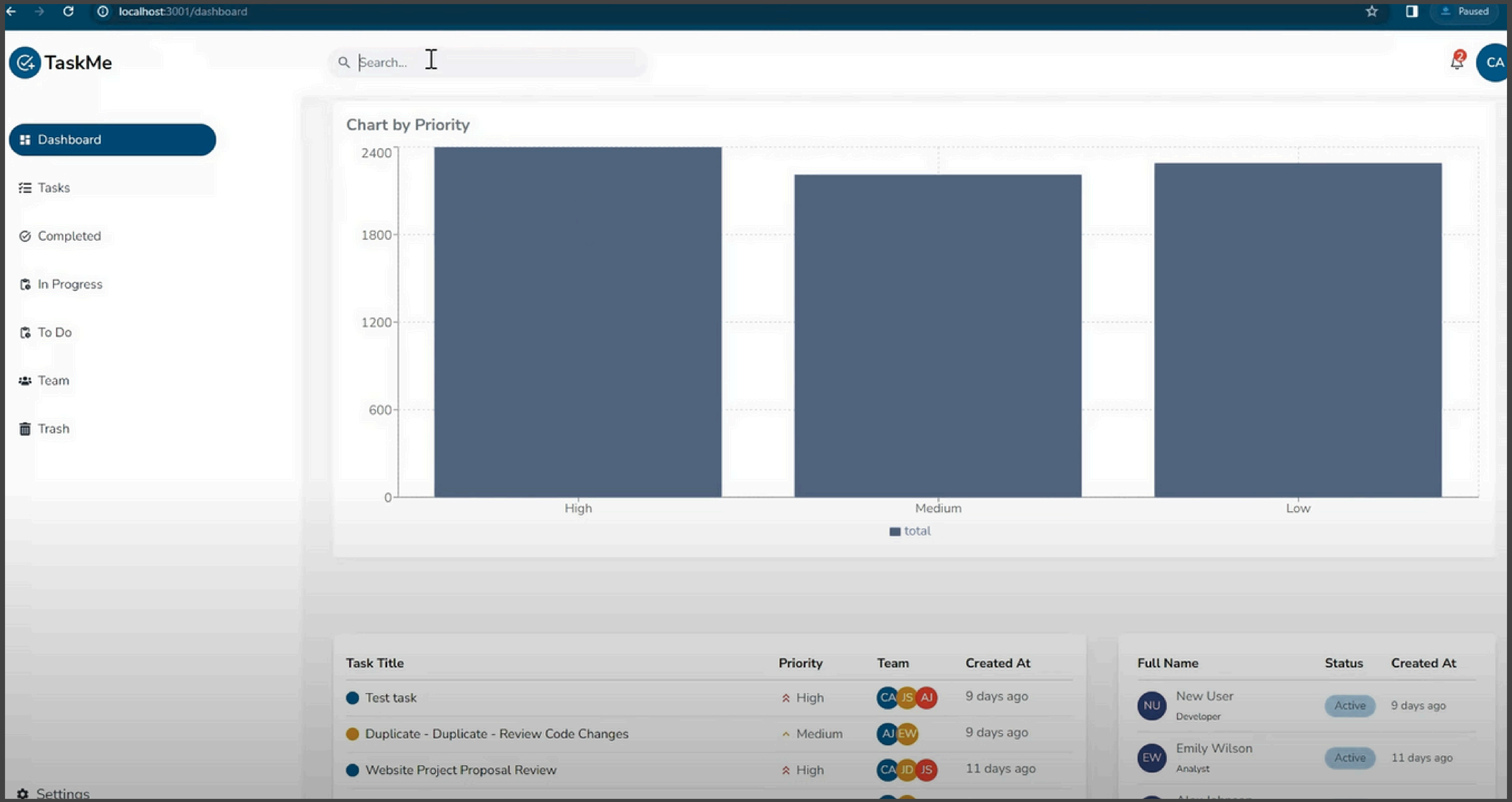
Email Address

atwork.jaimin@gmail.com

Password

•••••••••••••

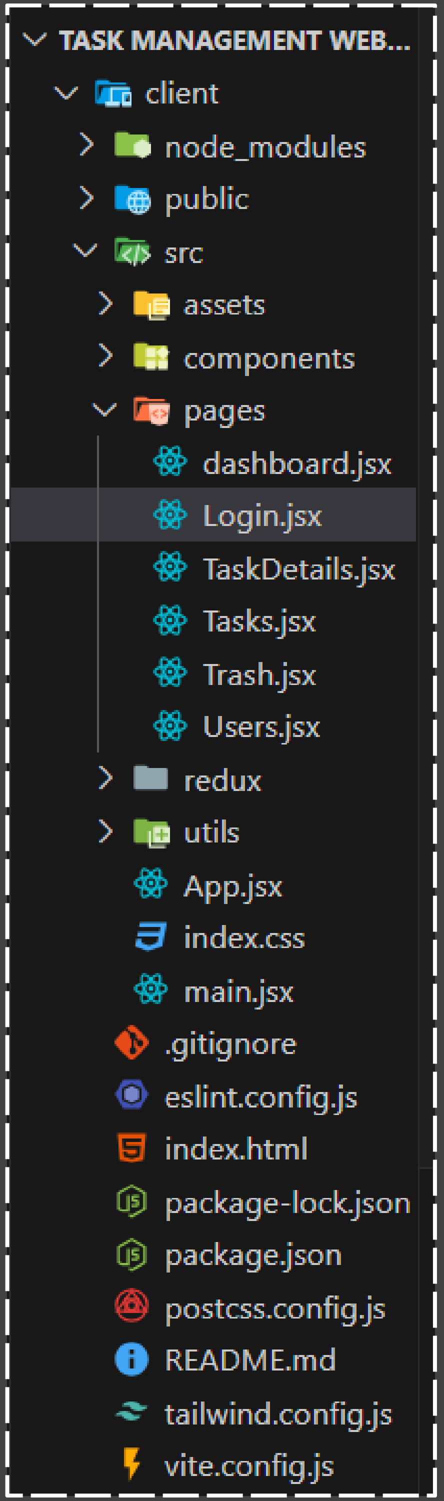
09. User Interface - 2



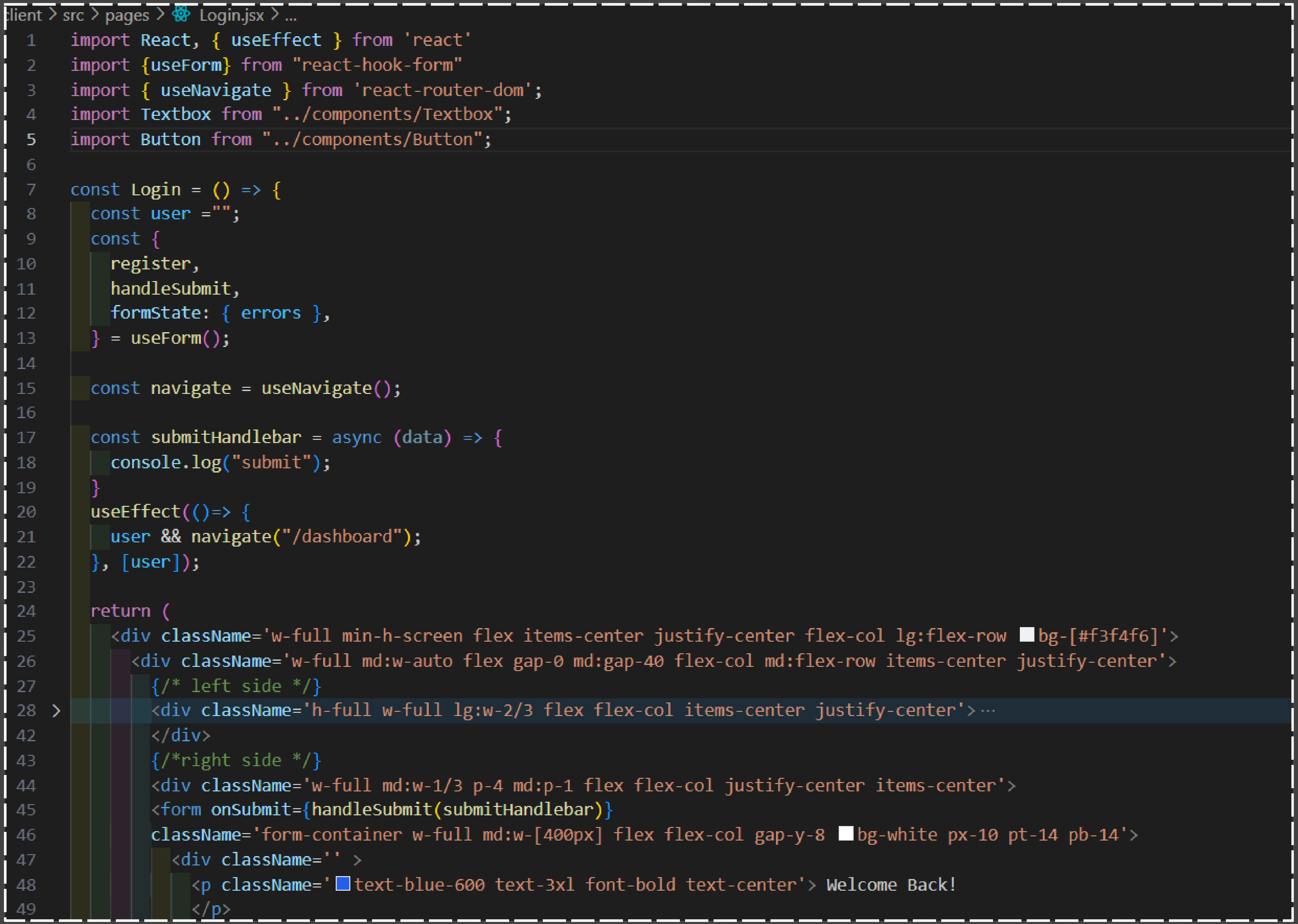
Task Title	Priority	Team	Created At
Test task	High	CA JS AJ	9 days ago
Duplicate - Duplicate - Review Code Changes	Medium	AJ EW	9 days ago
Website Project Proposal Review	High	CA JD JS	11 days ago

Full Name	Status	Created At
NU New User Developer	Active	9 days ago
EW Emily Wilson Analyst	Active	11 days ago

10. Code Snapshots - 1



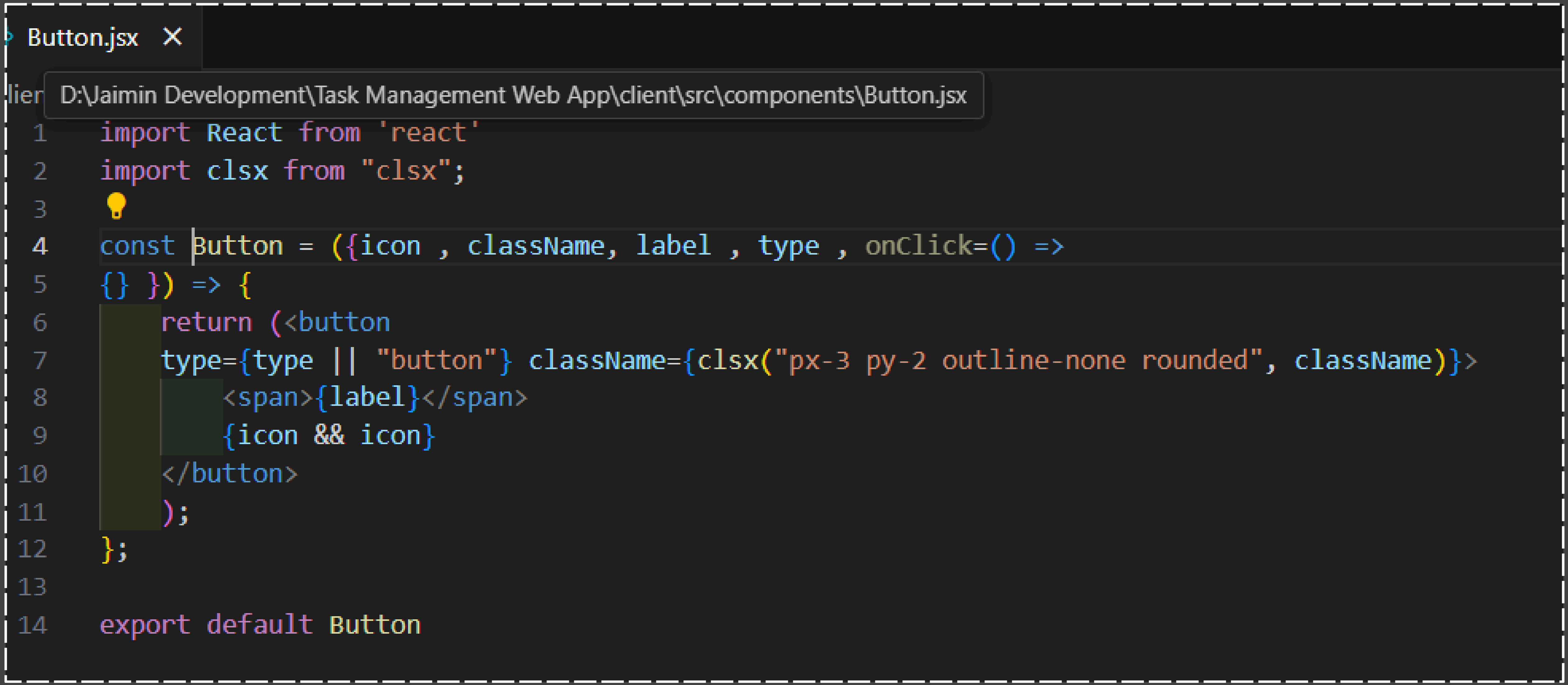
```
TASK MANAGEMENT WEB...
  client
    node_modules
    public
    src
      assets
      components
      pages
        dashboard.jsx
        Login.jsx
        TaskDetails.jsx
        Tasks.jsx
        Trash.jsx
        Users.jsx
      redux
      utils
      App.jsx
      index.css
      main.jsx
    .gitignore
    eslint.config.js
    index.html
    package-lock.json
    package.json
    postcss.config.js
    README.md
    tailwind.config.js
    vite.config.js
```



lient > src > pages > Login.jsx > ...

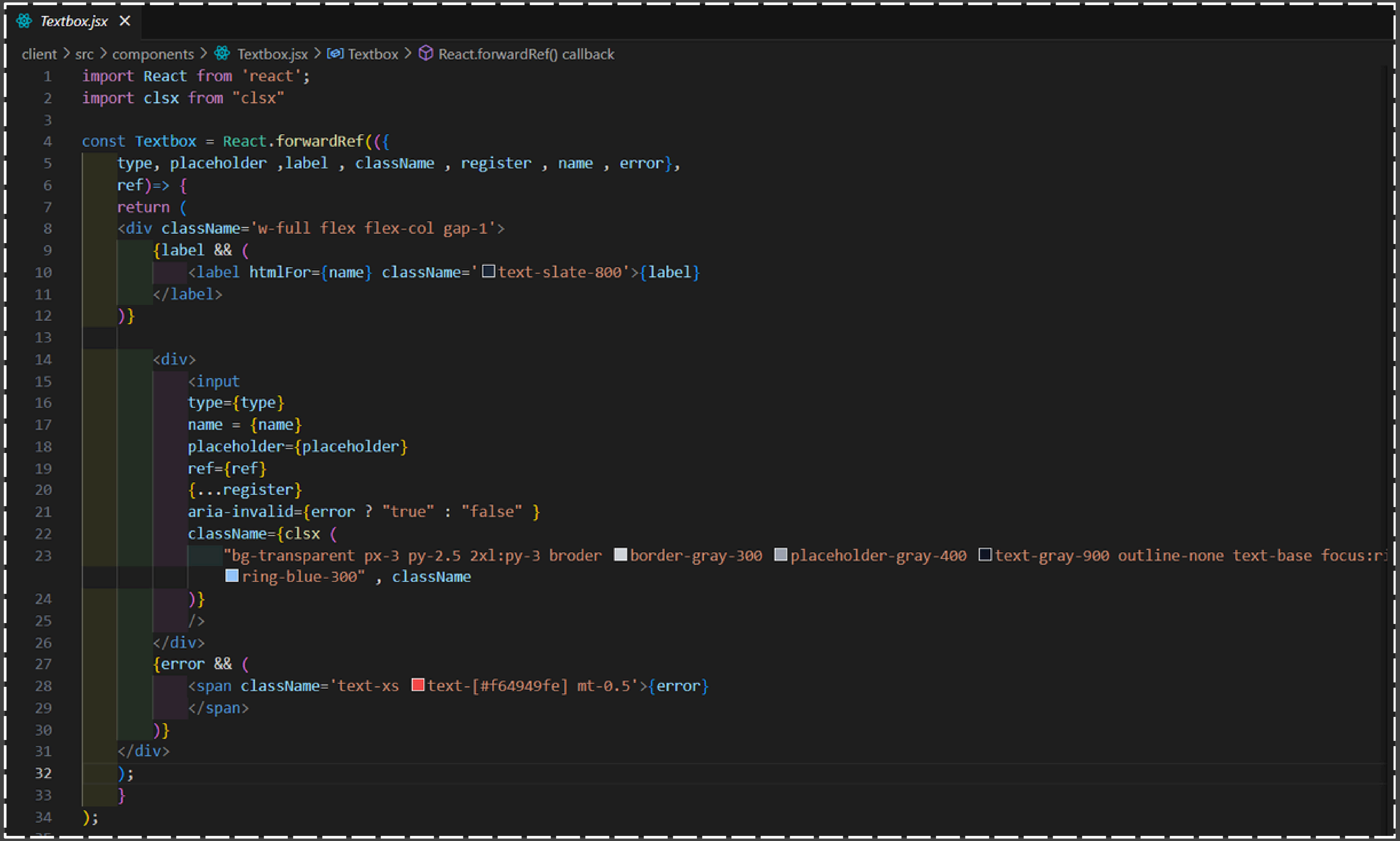
```jsx
import React, { useEffect } from 'react'
import {useForm} from "react-hook-form"
import { useNavigate } from 'react-router-dom';
import Textbox from "../components/Textbox";
import Button from "../components/Button";

const Login = () => {
  const user ="";
  const {
    register,
    handleSubmit,
    formState: { errors },
  } = useForm();

  const navigate = useNavigate();

  const submitHandlebar = async (data) => {
    console.log("submit");
  }
  useEffect(()=> {
    user && navigate("/dashboard");
  }, [user]);

  return (
    <div className='w-full min-h-screen flex items-center justify-center flex-col lg:flex-row bg-[#f3f4f6]'>
      <div className='w-full md:w-auto flex gap-0 md:gap-40 flex-col md:flex-row items-center justify-center'>
        {/* left side */}
        <div className='h-full w-full lg:w-2/3 flex flex-col items-center justify-center'>···
        </div>
        {/*right side */}
        <div className='w-full md:w-1/3 p-4 md:p-1 flex flex-col justify-center items-center'>
        <form onSubmit={handleSubmit(submitHandlebar)}
        className='form-container w-full md:w-[400px] flex flex-col gap-y-8 bg-white px-10 pt-14 pb-14'>
          <div className='' >
            <p className='text-blue-600 text-3xl font-bold text-center'> Welcome Back!
            </p>
```

10. Code Snapshots - 2


Button.jsx
lier
D:\Jaimin Development\Task Management Web App\client\src\components\Button.jsx

```jsx
import React from 'react'
import clsx from "clsx";

const Button = ({icon , className, label , type , onClick=() =>
{} }) => {
    return (<button
    type={type || "button"} className={clsx("px-3 py-2 outline-none rounded", className)}>
        <span>{label}</span>
        {icon && icon}
    </button>
    );
};

export default Button
```

10. Code Snapshots - 3


Textbox.jsx
client > src > components > Textbox.jsx > Textbox > React.forwardRef() callback

```jsx
import React from 'react';
import clsx from "clsx"

const Textbox = React.forwardRef(({
    type, placeholder ,label , className , register , name , error},
    ref)=> {
    return (
    <div className='w-full flex flex-col gap-1'>
        {label && (
            <label htmlFor={name} className='text-slate-800'>{label}
        </label>
    )}

        <div>
            <input
            type={type}
            name = {name}
            placeholder={placeholder}
            ref={ref}
            {...register}
            aria-invalid={error ? "true" : "false" }
            className={clsx (
                "bg-transparent px-3 py-2.5 2xl:py-3 broder border-gray-300 placeholder-gray-400 text-gray-900 outline-none text-base focus:ri
                ring-blue-300" , className
            )}
            />
        </div>
        {error && (
            <span className='text-xs text-[#f64949fe] mt-0.5'>{error}
            </span>
        )}
    </div>
    );
    }
);
```

11. Merits & Demerits

- Merits:
- Efficient task distribution
- Centralized management
- Scalable architecture
- Intuitive UI
- Secure and reliable

- Demerits:
- Limited offline functionality
- Requires initial setup and hosting
- No mobile app (web-only)

12. Features

- Task Assignment & Prioritization:
- Deadline Tracking & Notification:
- Progress Reporting:
- Role-Based Permissions:
- Real-Time Collaboration:
- Secure Authentication & Authorization:

13. Short Explanation

The Task Management System Web App helps individuals and teams manage their daily tasks efficiently. It focuses on transparency, accountability, and productivity through a combination of intuitive UI and powerful backend logic using the MERN stack.

14. Conclusion

The Task Management Web-App project demonstrates the use of full-stack web development to solve real-world productivity challenges. With flexibility, scalability, and security, it can serve teams of any size, making it a valuable tool in today’s work environments.

Thank You !