PLANIFY-TASK REMAINDER

Application development Report Submitted
In partial fulfillment of the requirements for the award of the degree of

Bachelor of Technology
In
Information Technology

By

R.NAMRATHA 22N31A12F2

Y.NISHANTH 22N31A12K1

R.BANTHILAL 22N31A12F1

Under the esteemed guidance of

Ms. A. JAYASREE

Department of computer science & Information Technology

Malla Reddy College of Engineering and Technology

(Autonomous Institution-UGC, Govt. of India)

(Affiliated to JNTUH, Hyderabad, Approved by AICTE, NBA&NAAC with 'A' Grade)
Maisammaguda, Kompally, Dhulapally, Secunderabad – 500100
Website: www.mrcet.ac.in

Malla Reddy College of Engineering and Technology

(Autonomous Institution-UGC, Govt. of India)

(Affiliated to JNTUH, Hyderabad, Approved by AICTE, NBA&NAAC with 'A' Grade)
Maisammaguda, Kompally, Dhulapally, Secunderabad – 500100
Website: www.mrcet.ac.in

CERTIFICATE

This is to certify that this is the Bonafide record of the Application development -2 entitled **" PLANIFY- TASK REMAINDER "** submitted by STUDENT **R.Namratha (22N31A12F2) , Y.Nishanth (22N31A12K1)** and **R.Banthilal (22N31A12F1)** of B. Tech in the partial fulfillment of the requirements for the degree of Bachelor of Technology in Information Technology, Department of Information Technology during the year 2024-2025.

Internal Guide **Head of the Department**

External Examiner

ABSTRACT

In today's fast-paced world, managing tasks efficiently has become essential for maintaining productivity and achieving goals. Planify - Task Reminder addresses this need by providing a seamless platform that helps users organize, prioritize, and track their daily activities .This website is an intuitive web-based application designed to help users organize and manage their daily tasks efficiently. It allows users to categorize tasks into long-term and short-term goals, set deadlines, and track progress. This website helps users by keeping them motivated by displaying daily motivational quotes. Additionally, if users become inactive or fail to enter tasks, the system sends suggestions like reading or cooking to encourage productivity. A standout feature of this platform is the community chat, where users can share tips, ask questions, and motivate each other. This collaborative approach creates a sense of shared responsibility and boosts user engagement. Moreover, the emergency mode highlights high-priority tasks with a red box, ensuring that urgent tasks get immediate attention. To keep users on track, the system sends notifications and reminders via push notifications, email, or SMS. These reminders help users stay updated on deadlines and encourage timely task completion. Once tasks are completed, the system celebrates the achievement by triggering an interactive balloon animation, adding a fun and rewarding element to the experience. Planify - Task Reminder is more than a productivity tool; it's a step towards a well-organized and balanced lifestyle..

TABLE OF CONTENTS

S. NO	TITLE	PG. NO
	ABSTRACT	
1	**INTRODUCTION**	
	1.1 PURPOSE AND OBJECTIVES	01
	1.2 EXISTING AND PROPOSED SYSTEM	02
2	**APPLICATION DESCRIPTION**	
	2.1 HARDWARE AND SOFTWARE REQUIREMENTS	03
	2.2 METHODOLOGY	04
3	**SOURCE CODE**	05
4	**RESULTS**	11
5	**CONCLUSION & ENHANCEMENTS**	18
	BIBLIOGRAPHY	

1. INTRODUCTION

1.1 PURPOSE AND OBJECTIVES

“Planify - Task Reminder” is a smart and efficient task management application designed to help users stay organized and productive. Whether it’s an urgent task, a long-term goal, or a personal commitment, Planify ensures that you never miss an important deadline. With features like emergency task prioritization, categorized task input, and an organized task list, users can seamlessly manage their daily activities. The application also includes progress tracking, calendar integration, and a chatbox for collaboration, making it an all-in-one productivity tool. Planify simplifies task management, helping users focus on what truly matters while staying on top of their schedules.

1.2 EXISTING AND PROPOSED SYSTEM

EXISTING SYSTEM

In traditional task management, people often rely on manual methods such as paper planners, sticky notes, or basic to-do lists on their phones. These methods lack efficiency, reminders, and proper categorization, making it easy to forget important tasks. Some digital applications provide task management features, but they may not prioritize urgent tasks, offer effective categorization, or provide real-time tracking. Additionally, collaboration and progress monitoring are often limited, making it difficult for users to stay consistently organized.

PROPOSED SYSTEM

Planify offers a structured and efficient way to manage tasks by integrating automated reminders, task categorization, and progress tracking in a single platform. It allows users to prioritize emergency tasks, classify tasks as short-term or long-term, and view them in an organized manner. With features like calendar integration, chatbox support, and a dedicated dashboard, Planify ensures seamless task management. The system enhances productivity by reducing the chances of missing deadlines while providing a user-friendly interface for easy navigation and task organization.

2. APPLICATION DESCRIPTION

2.1 HARDWARE AND SOFTWARE REQUIREMENTS

HARDWARE REQUIREMENTS:

- ✧ Processor: Intel i3 or higher (or equivalent AMD)
- ✧ RAM: Minimum 4GB (8GB recommended)
- ✧ Storage: Minimum 20GB free space

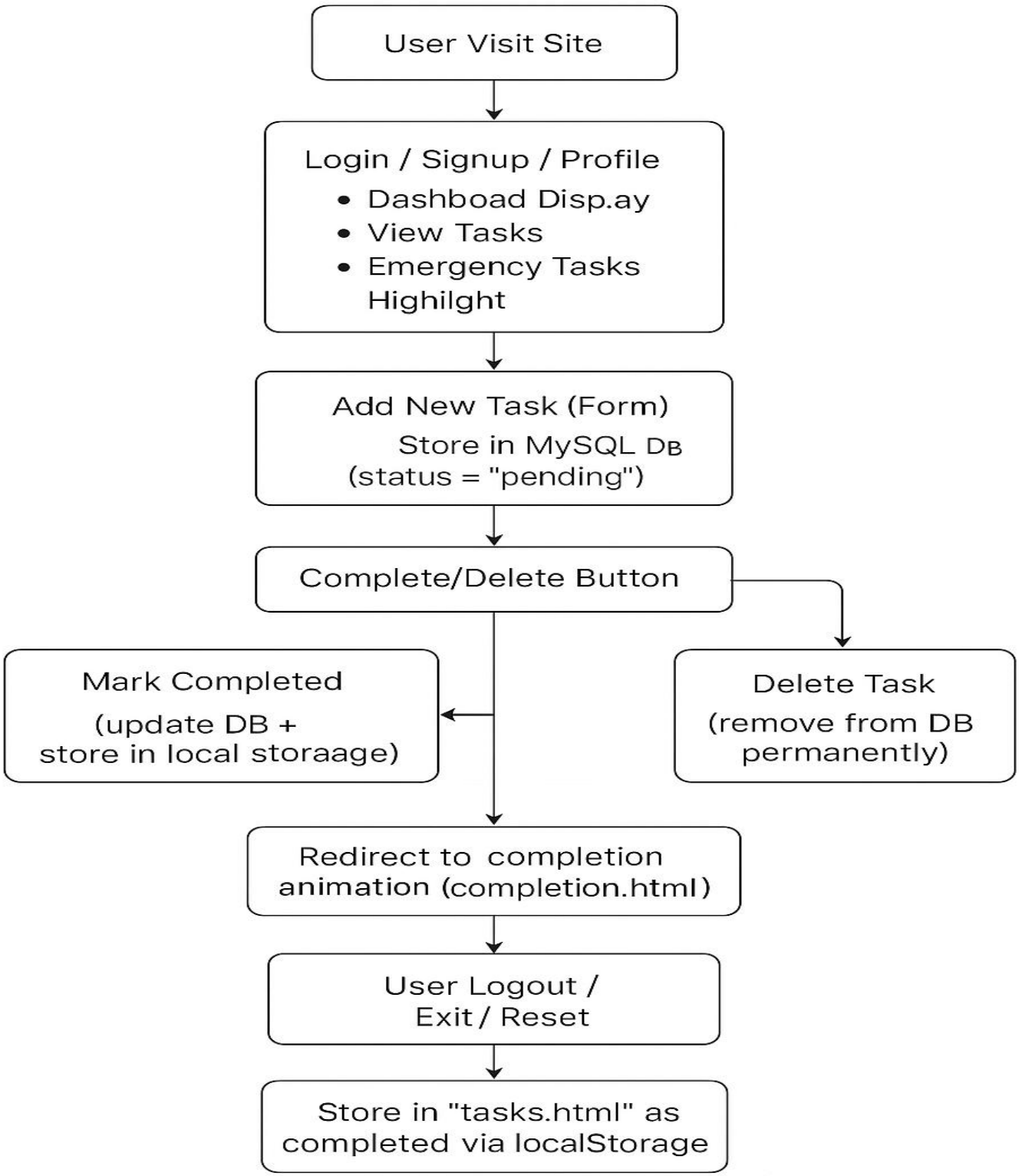
SOFTWARE REQUIREMENTS:

- ✧ Frontend: HTML, CSS, JavaScript
- ✧ Backend: Node.js
- ✧ Database: MySQL
- ✧ Development Tools: VS Code
- ✧ Notification Services -

  Email Notifications: Nodemailer

  Browser Notifications: JavaScript Notification API

2.2 METHODOLOGY



3. SOURCE CODE

INDEX.HTML

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Planify</title>
  <style>
    body {
      margin: 0;
      height: 100vh;
      display: flex;
      flex-direction: column;
      font-family: 'Arial', sans-serif;
      background: linear-gradient(to right, #6a11cb, #2575fc);
    }

    #upper-section {
      height: 40vh;
      background:  linear-gradient(to right, #51287c, #2575fc);
      color: white;
      box-shadow: 0 2px 6px rgba(0, 0, 0, 0.1);
    }

    h1 {
      font-size: 50px;
      color: #f1f8ff;
      margin: 0;
    }

    button {
      padding: 10px 20px;
      font-size: 16px;
      background-color: #5c9cd8;
      color: white;
      border: none;
      border-radius: 8px;
      cursor: pointer;
      transition: background-color 0.3s ease;
    }

    button:hover {
      background-color: #417cb8;
    }

    #content-section {
```

```
  height: 60vh;
  display: flex;
  padding: 20px 40px;
  background: linear-gradient(to bottom right,#f3f3f5, #7d95c0); /* Slightly darker blue gradient */
}

#text-container {
  width: 50%;
  font-size: 40px;
  color: #083759;
  display: flex;
  align-items: center;
  justify-content: flex-end;
  text-align: right;
  padding-right: 40px;
}

#logo-container {
  width: 50%;
  display: flex;
  align-items: center;
  justify-content: flex-start;
}

#logo-container img {
  max-width: 50%;
  height: auto;
}

#login-section {
  height: 20vh;
  display: flex;
  flex-direction: column;
  align-items: center;
  justify-content: center;
  gap: 10px;
}

.button-box {
  padding: 20px;
  background: rgba(255, 255, 255, 0.2);
  border-radius: 10px;
  box-shadow: 0 2px 4px rgba(0, 0, 0, 0.2);
}

iframe {
  width: 100%;
  height: 100%;
  border: none;
```

```
      display: none;
    }
  </style>
</head>
<body>
  <!-- Upper 40% Section -->
  <div id="upper-section">
    <!-- 20% Section with Heading -->
    <div style="height: 20vh; display: flex; justify-content: center; align-items: center;">
      <h1>PLANIFY - TASK REMINDER</h1>
    </div>

    <!-- 20% Section with Message and Buttons in a Box -->
    <div id="login-section">
      <div>Stay on top of your tasks. Start now!</div>
      <div class="button-box">
        <button onclick="loadPage('login.html')">Login</button>
        <button onclick="window.open('signup.html', '_self')">Sign Up</button>

      </div>
    </div>
  </div>

  <!-- 60% Section with Text and Logo -->
  <div id="content-section">
    <div id="text-container">
      ORDINARY PEOPLE <br> SPEND TIME. GREAT <br> PEOPLE USE IT.
    </div>
    <div id="logo-container">
      <img src="logo.png" alt="Logo">
    </div>

    <!-- Iframe (hidden by default) -->
    <iframe id="content-frame" src=""></iframe>
  </div>

  <script>
    function loadPage(page) {
      document.getElementById('text-container').style.display = 'none';
      document.getElementById('logo-container').style.display = 'none';
      const iframe = document.getElementById('content-frame');
      iframe.style.display = 'block';
      iframe.src = page;
    }
  </script>
</body>
</html>
```

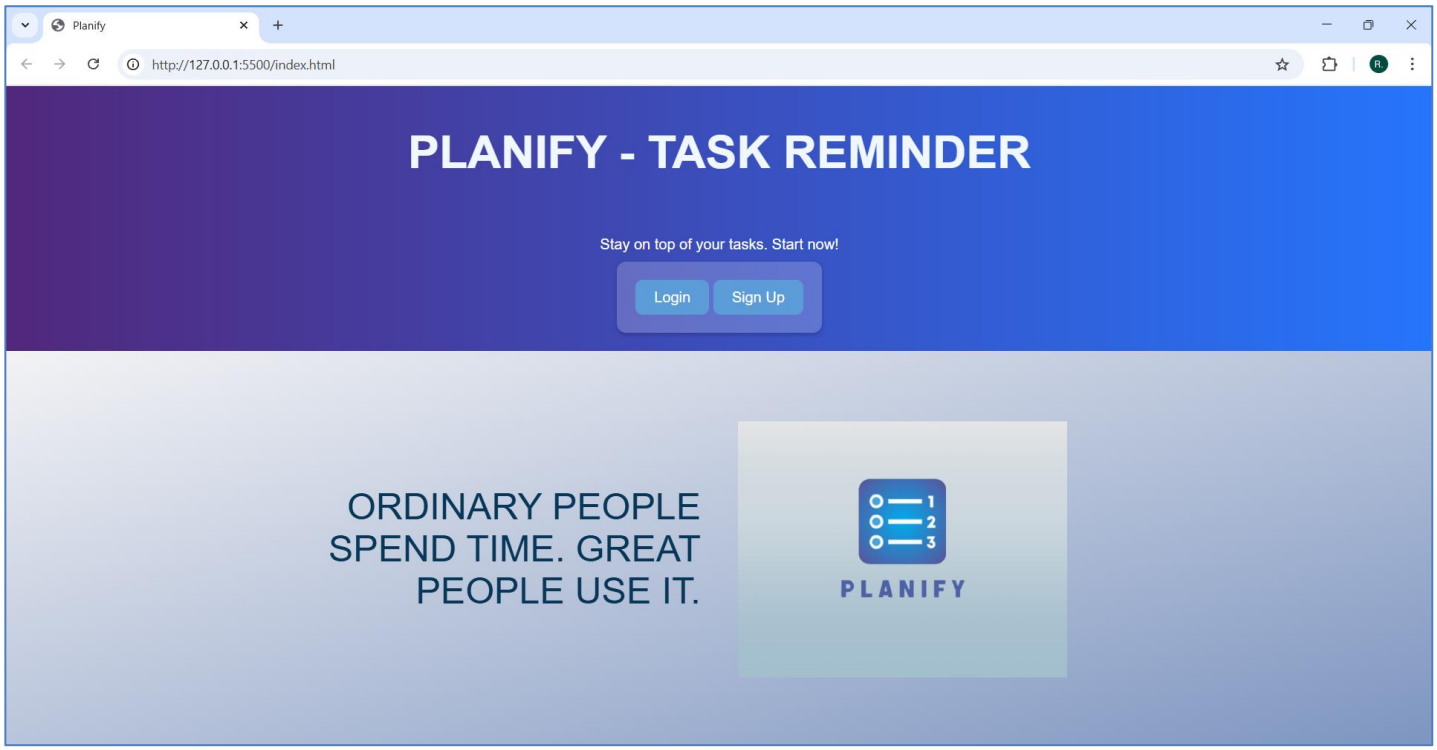
SERVER.JS

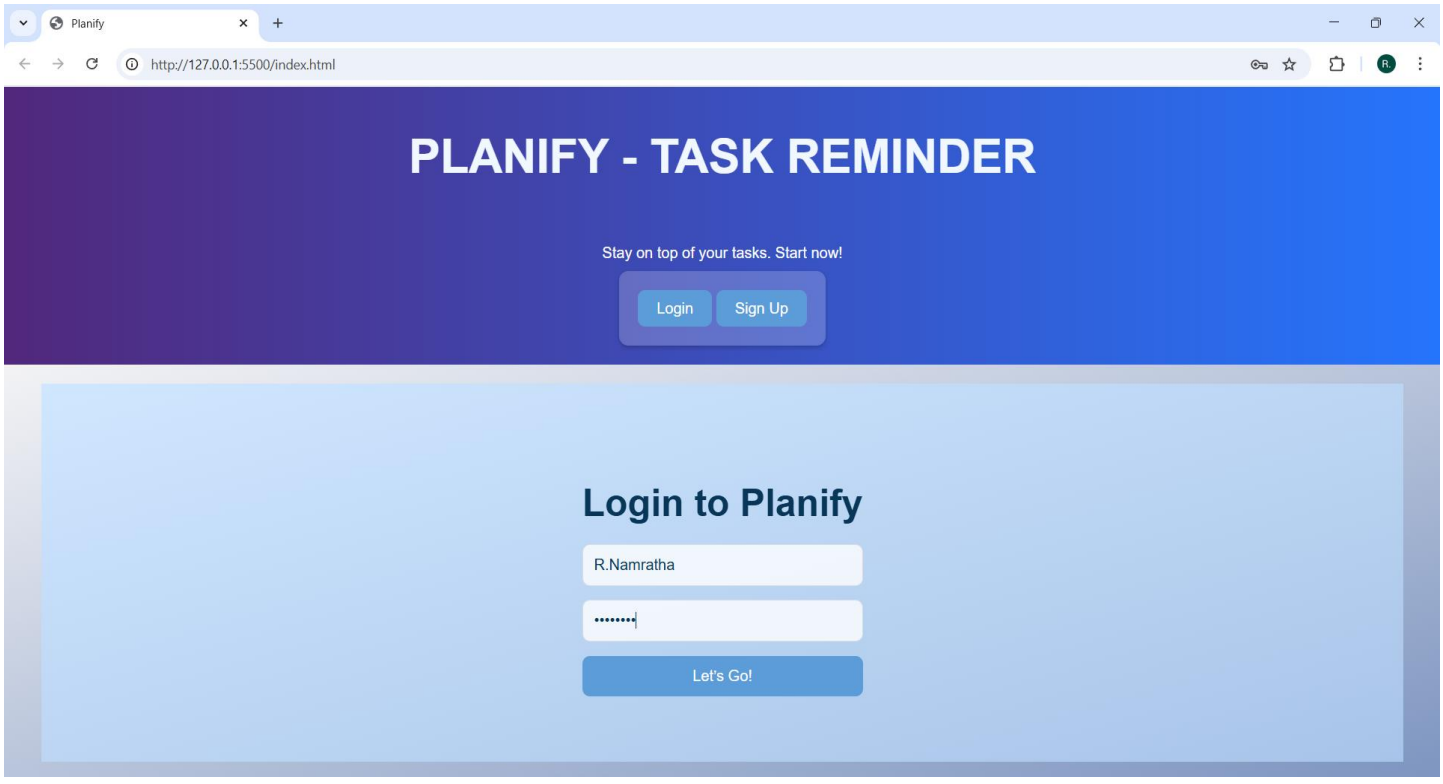
```
require("dotenv").config();
const express = require("express");
const mysql = require("mysql2");
const cors = require("cors");

const app = express();
app.use(express.json());
app.use(cors());
app.use(express.static(__dirname));  // Serve static files like HTML, CSS, JS


// ✅ Connect to MySQL
const db = mysql.createConnection({
  host: "localhost",
  user: "root",
  password: "hello123", // Change to your MySQL password
  database: "planify",
});

db.connect(err => {
  if (err) {
    console.error("❌ MySQL Connection Error:", err);
    process.exit(1);
  }
  console.log("✅ MySQL Connected...");
});

// ✅ Add Task (POST)
app.post("/add-task", (req, res) => {
  const { taskname, term, category, date, time } = req.body;
  if (!taskname || !term || !category || !date || !time) {
    return res.status(400).json({ error: "⚠ All fields are required!" });
  }

  const sql = "INSERT INTO tasks (taskname, term, category, date, time, status) VALUES (?, ?, ?, ?, ?, 'pending')";
  db.query(sql, [taskname, term, category, date, time], (err, result) => {
    if (err) return res.status(500).json({ error: err.message });
    res.json({ message: "✅ Task added successfully!", taskId: result.insertId });
  });
});

// ✅ Get All Tasks (GET)
app.get("/tasks", (req, res) => {
  const sql = "SELECT * FROM tasks ORDER BY date, time";

  db.query(sql, (err, results) => {
```

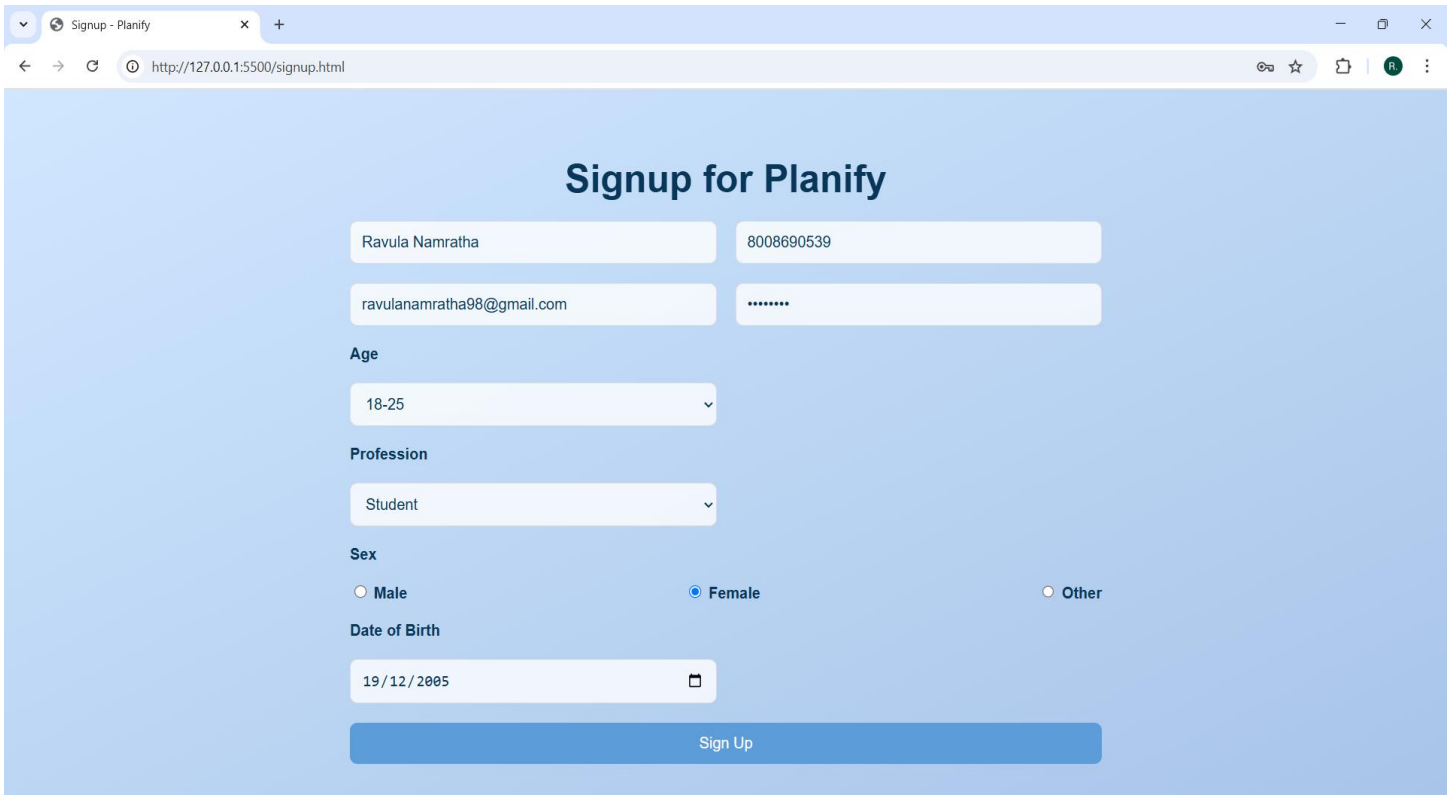
```
    if (err) return res.status(500).json({ error: err.message });
    res.json(results);
  });
});



// ✅ Mark Task as Completed (PUT)
app.put("/complete-task/:id", (req, res) => {
  const { id } = req.params;
  const sql = "UPDATE tasks SET status = 'completed' WHERE id = ?";

  db.query(sql, [id], (err, result) => {
    if (err) return res.status(500).json({ error: err.message });
    res.json({ message: "✅ Task marked as completed!" });
  });
});



// ✅ Delete Task (DELETE)
app.delete("/delete-task/:id", (req, res) => {
  const { id } = req.params;
  const sql = "DELETE FROM tasks WHERE id = ?";
  db.query(sql, [id], (err, result) => {
    if (err) return res.status(500).json({ error: err.message });
    res.json({ message: "✅ Task deleted successfully!" });
  });
});

const nodemailer = require("nodemailer");

// ✅ Configure Email Transporter (Use Your Gmail)
const transporter = nodemailer.createTransport({
   service: "gmail",
   auth: {
      user: "your-email@gmail.com", // Replace with your email
      pass: "your-app-password" // Use an App Password, NOT your real password
   }
});

// ✅ Function to Send Email Notifications
async function sendEmailNotification(to, subject, text) {
   try {
      await transporter.sendMail({
         from: "your-email@gmail.com",
         to: to,
         subject: subject,
         text: text
```

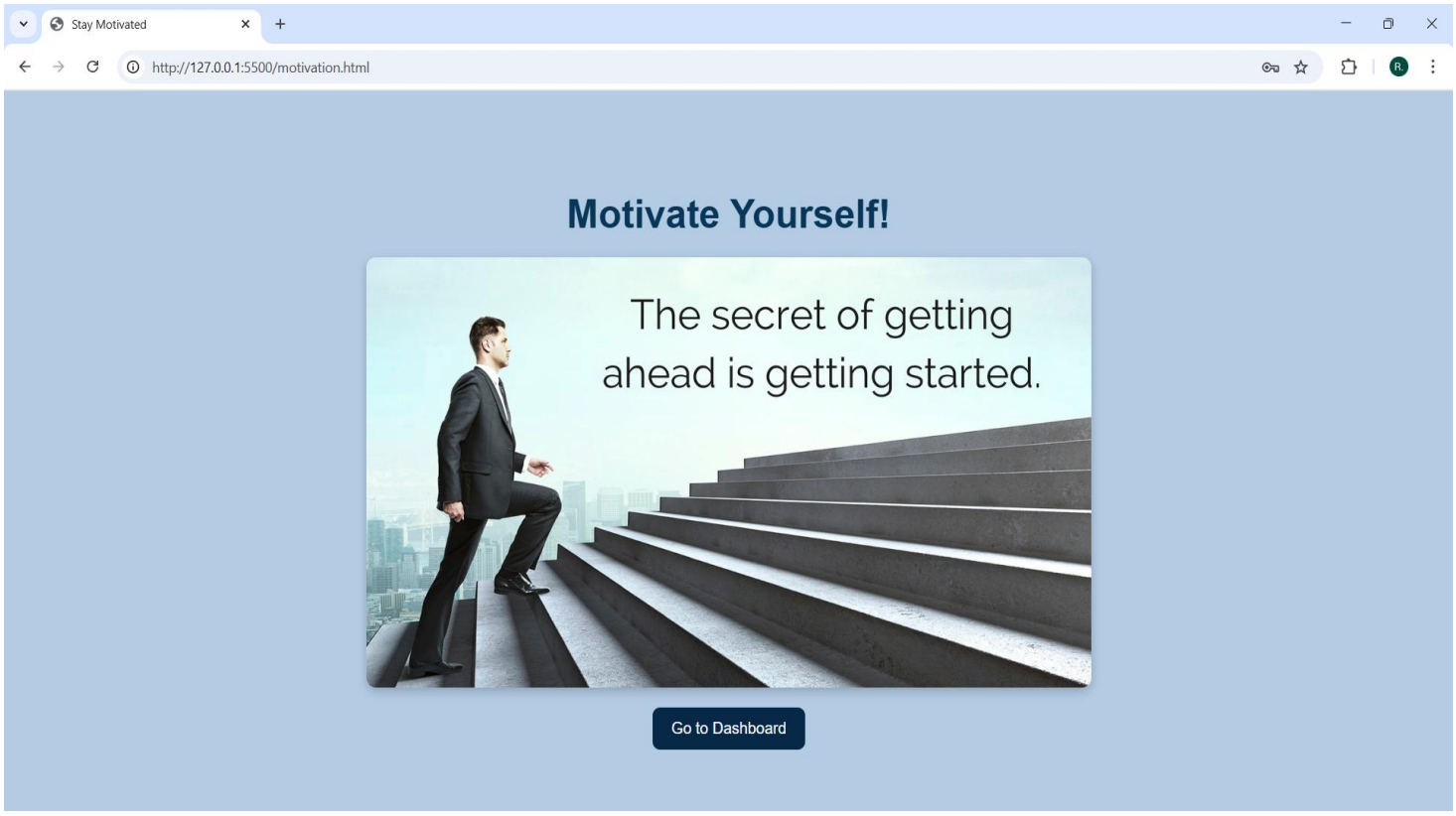
```
        });
        console.log("✅ Email Sent Successfully!");
    } catch (error) {
        console.error("❌ Error Sending Email:", error);
    }
}

// ✅ Example Usage
sendEmailNotification("recipient@example.com", "Task Reminder", "Don't forget to complete your task!");

const cron = require("node-cron");

// ✅ Run Every Minute (for testing) OR Change to "0 9 * * *" for 9 AM daily
cron.schedule("* * * * *", async () => {
    const now = new Date();
    const formattedDate = now.toISOString().split("T")[0]; // Get today's date

    db.query("SELECT * FROM tasks WHERE date = ?", [formattedDate], async (err, results) => {
        if (err) {
            console.error("❌ Error fetching tasks:", err);
            return;
        }

        for (let task of results) {
            await sendEmailNotification(task.email, "Task Reminder", `Reminder: '${task.taskname}' is scheduled for today at ${task.time}`);
        }
    });
});



// ✅ Start Server
const PORT = process.env.PORT || 5002;
app.listen(PORT, () => console.log(`    Server running on port ${PORT}`));
```

4. RESULTS

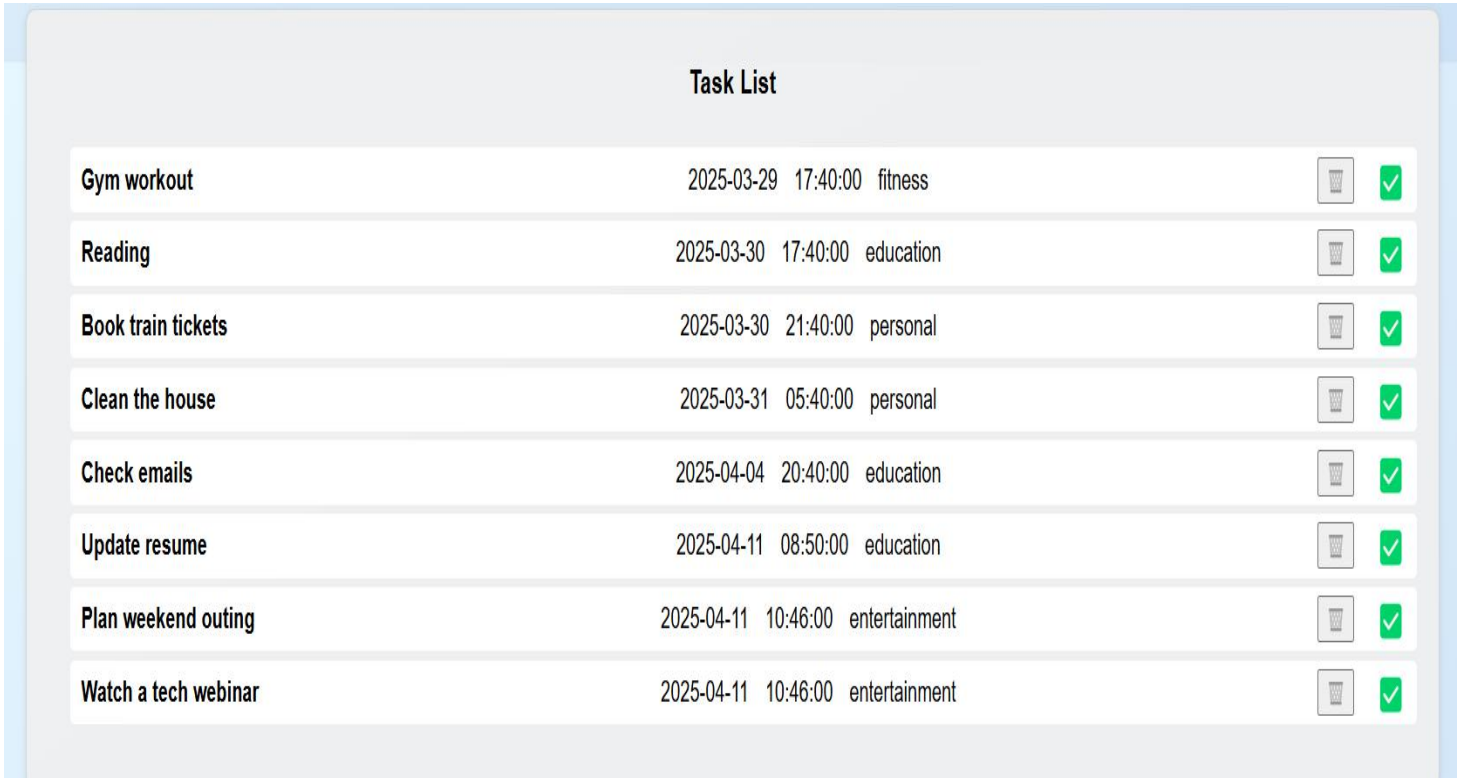

Planify
http://127.0.0.1:5500/index.html
PLANIFY - TASK REMINDER
Stay on top of your tasks. Start now!
Login
Sign Up
ORDINARY PEOPLE SPEND TIME. GREAT PEOPLE USE IT.
PLANIFY

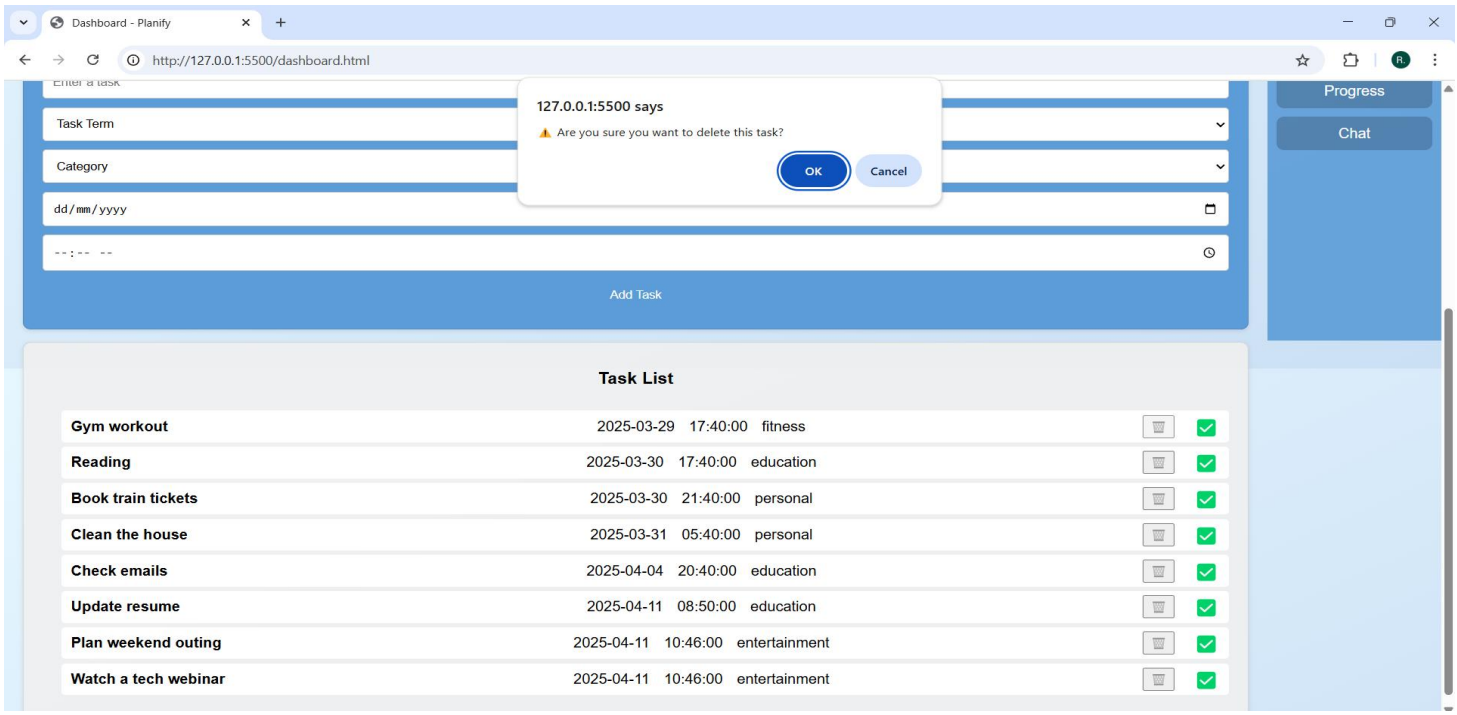

Planify
http://127.0.0.1:5500/index.html
PLANIFY - TASK REMINDER
Stay on top of your tasks. Start now!
Login
Sign Up
Login to Planify
R.Namratha
Let's Go!

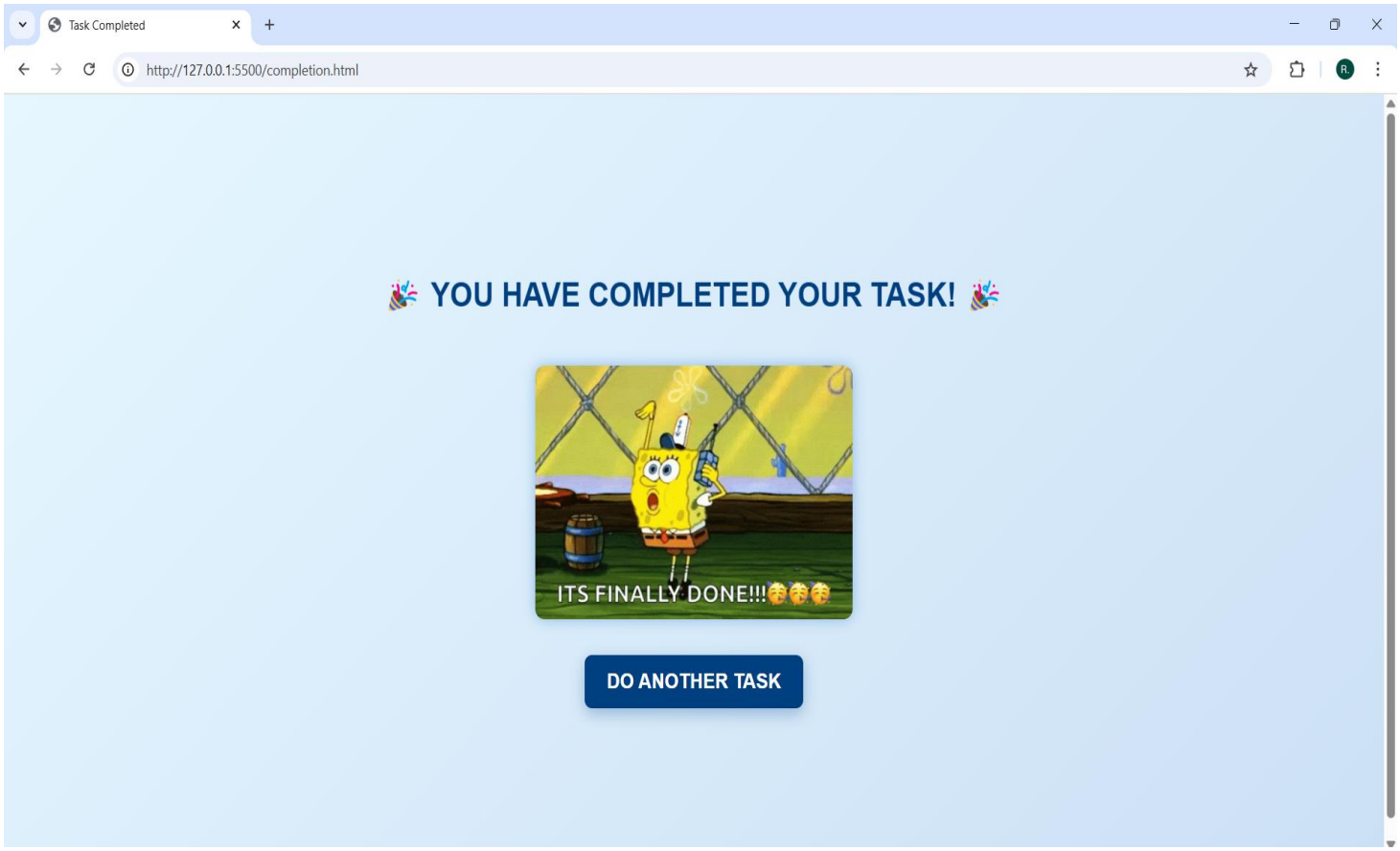

Signup - Planify
http://127.0.0.1:5500/signup.html
Signup for Planify
Ravula Namratha
8008690539
ravulanamratha98@gmail.com
Age
18-25
Profession
Student
Sex
Male
Female
Other
Date of Birth
19/12/2005
Sign Up

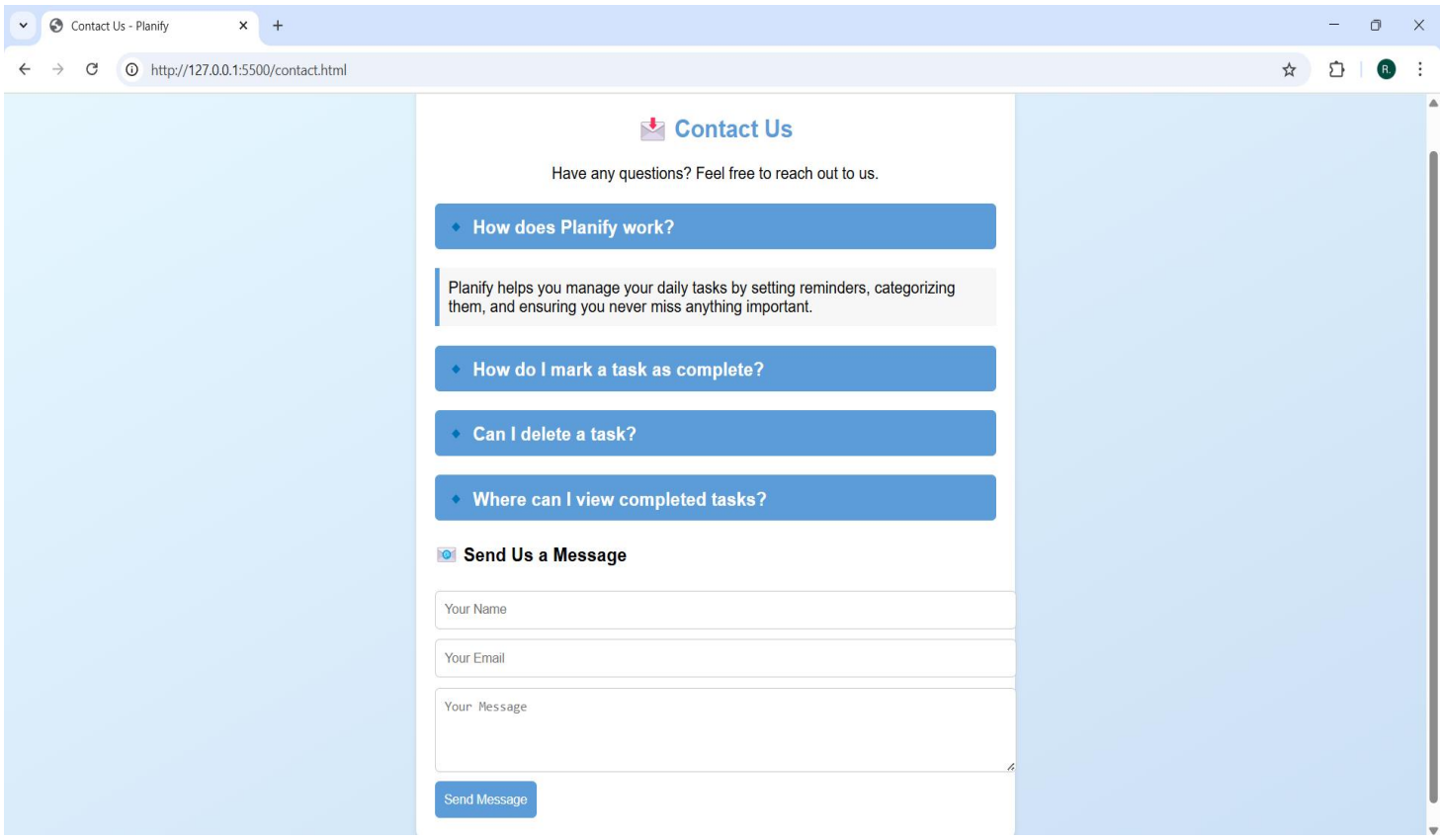

Stay Motivated
http://127.0.0.1:5500/motivation.html
Motivate Yourself!
The secret of getting ahead is getting started.
Go to Dashboard

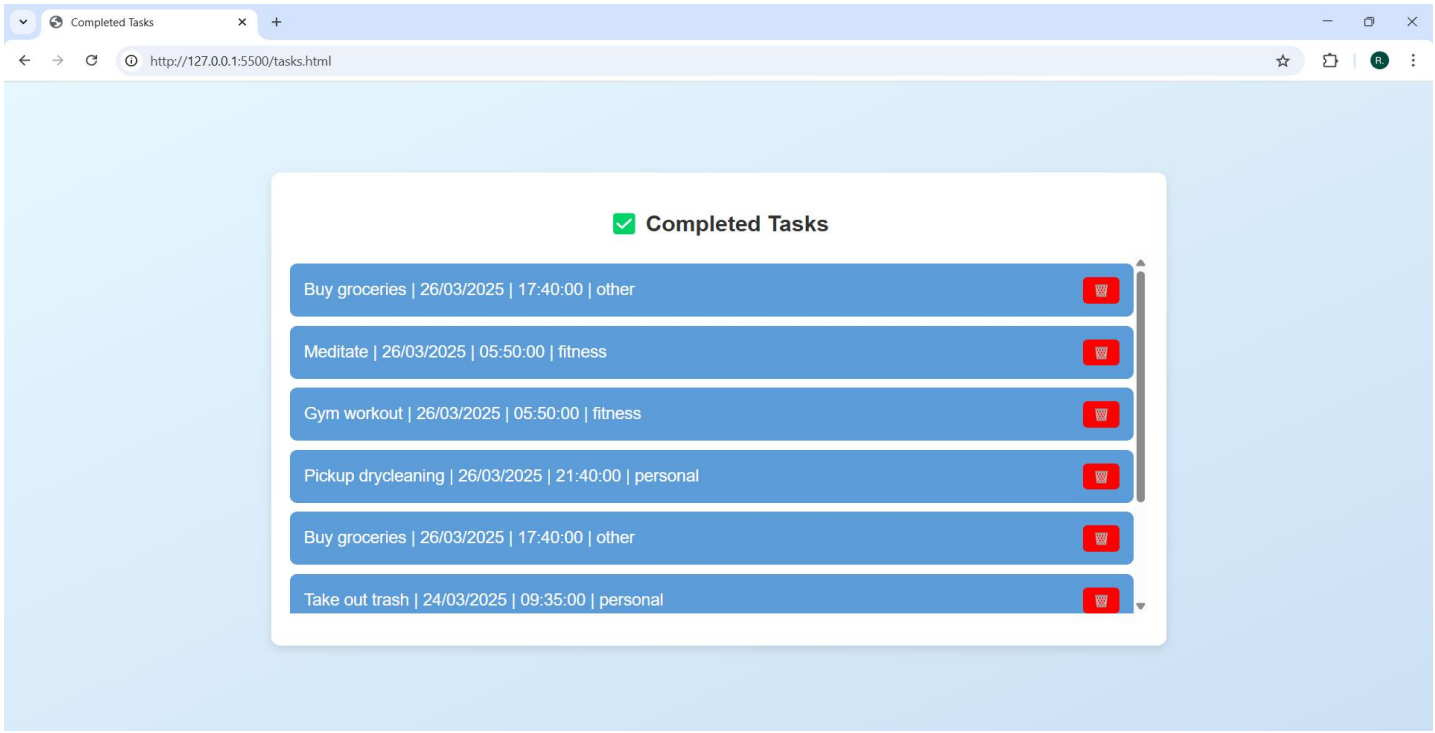

Dashboard - Planify
http://127.0.0.1:5500/dashboard.html
PLANIFY - TASK REMINDER
Emergency Tasks
Submit project report 2025-03-26 11:40:00 work
Doctor's appointment 2025-03-27 04:40:00 fitness
Pay electricity bill 2025-03-27 05:40:00 personal
Add Task
Enter a task
Task Term
Category
dd/mm/yyyy
--:-- --
Add Task
PLANIFY
Contact
Logout
Tasks
Calendar
Progress
Chat

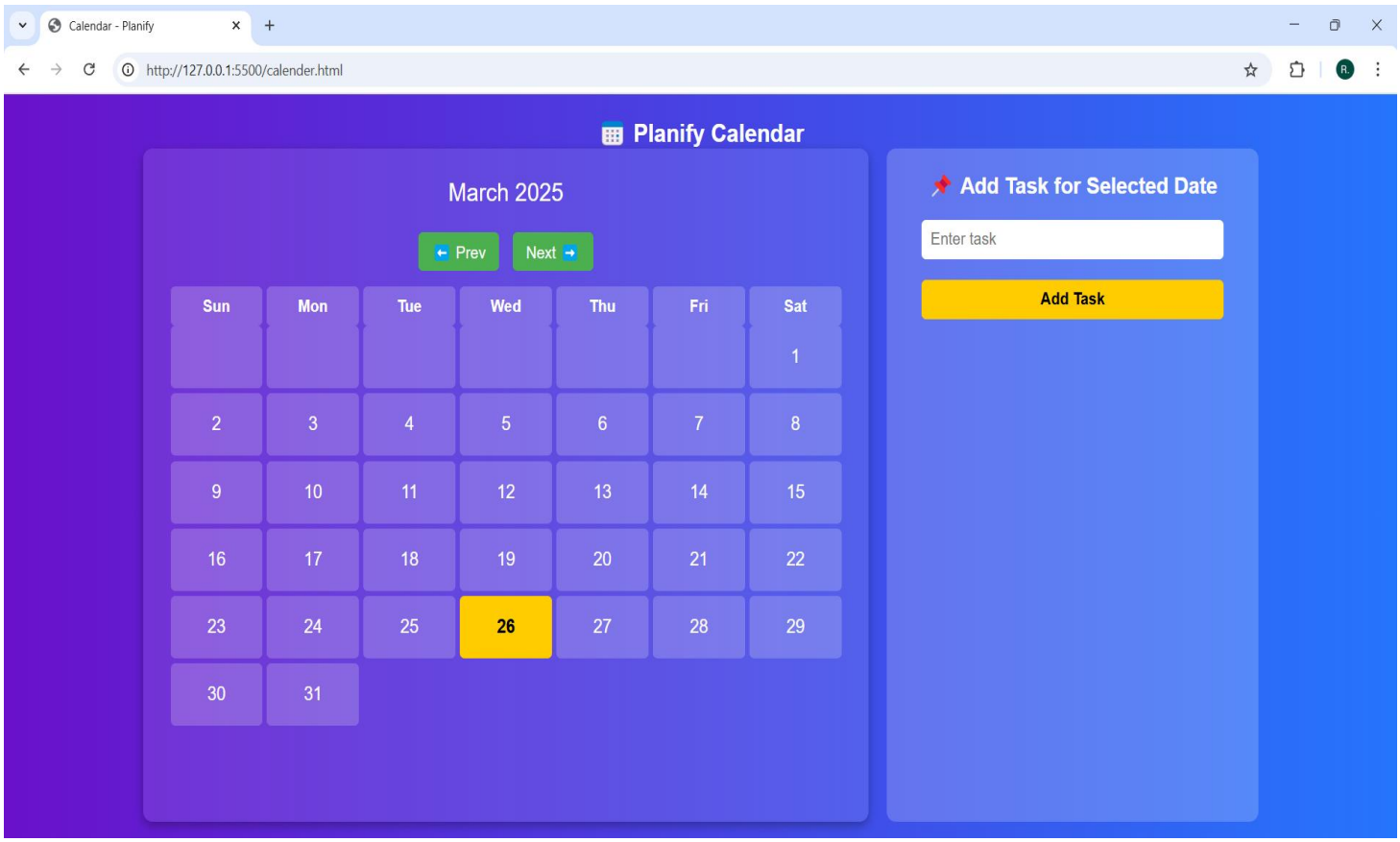

Task List
Gym workout 2025-03-29 17:40:00 fitness
Reading 2025-03-30 17:40:00 education
Book train tickets 2025-03-30 21:40:00 personal
Clean the house 2025-03-31 05:40:00 personal
Check emails 2025-04-04 20:40:00 education
Update resume 2025-04-11 08:50:00 education
Plan weekend outing 2025-04-11 10:46:00 entertainment
Watch a tech webinar 2025-04-11 10:46:00 entertainment

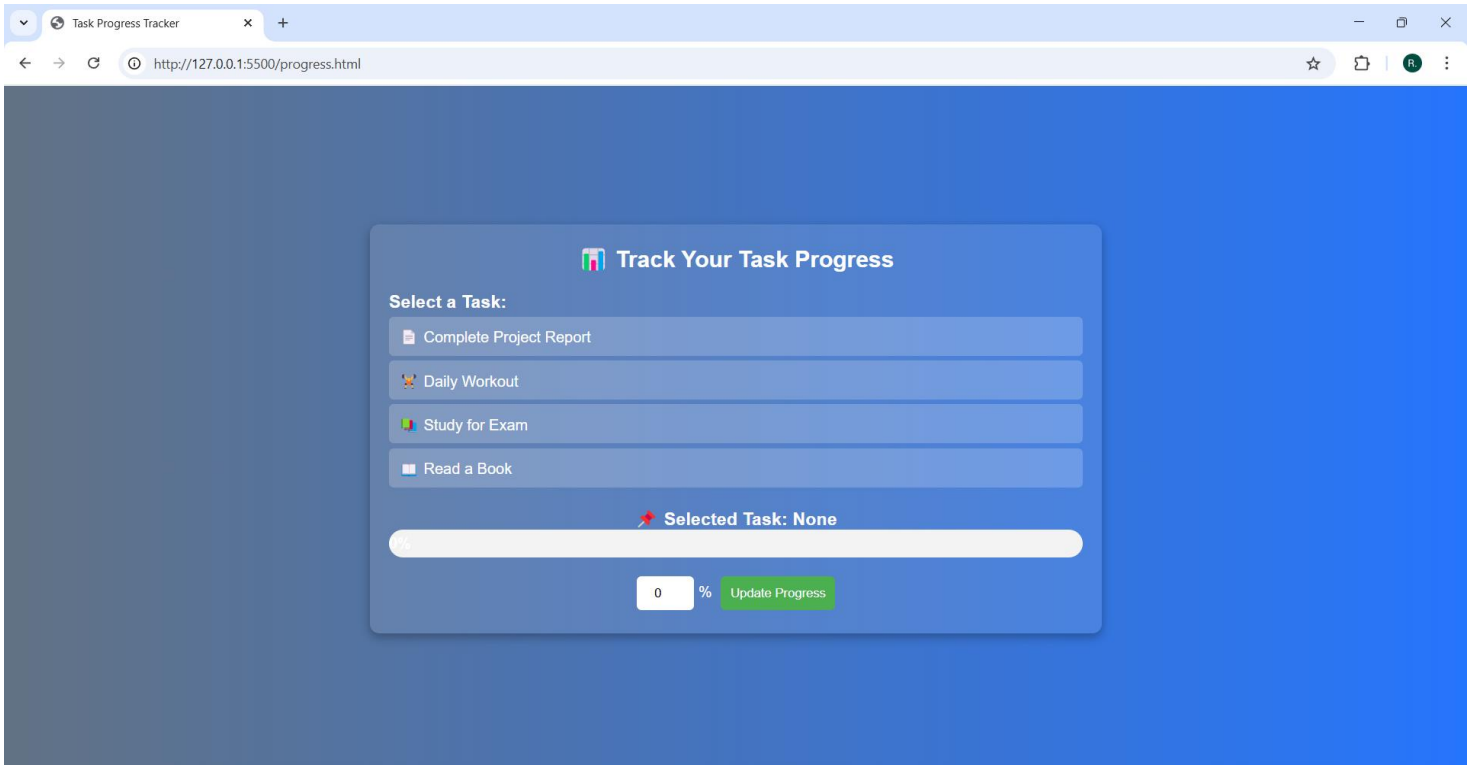

Dashboard - Planify
http://127.0.0.1:5500/dashboard.html
127.0.0.1:5500 says
Are you sure you want to delete this task?
OK
Cancel
Task Term
Category
dd/mm/yyyy
--:-- --
Add Task
Progress
Chat
Task List
Gym workout 2025-03-29 17:40:00 fitness
Reading 2025-03-30 17:40:00 education
Book train tickets 2025-03-30 21:40:00 personal
Clean the house 2025-03-31 05:40:00 personal
Check emails 2025-04-04 20:40:00 education
Update resume 2025-04-11 08:50:00 education
Plan weekend outing 2025-04-11 10:46:00 entertainment
Watch a tech webinar 2025-04-11 10:46:00 entertainment

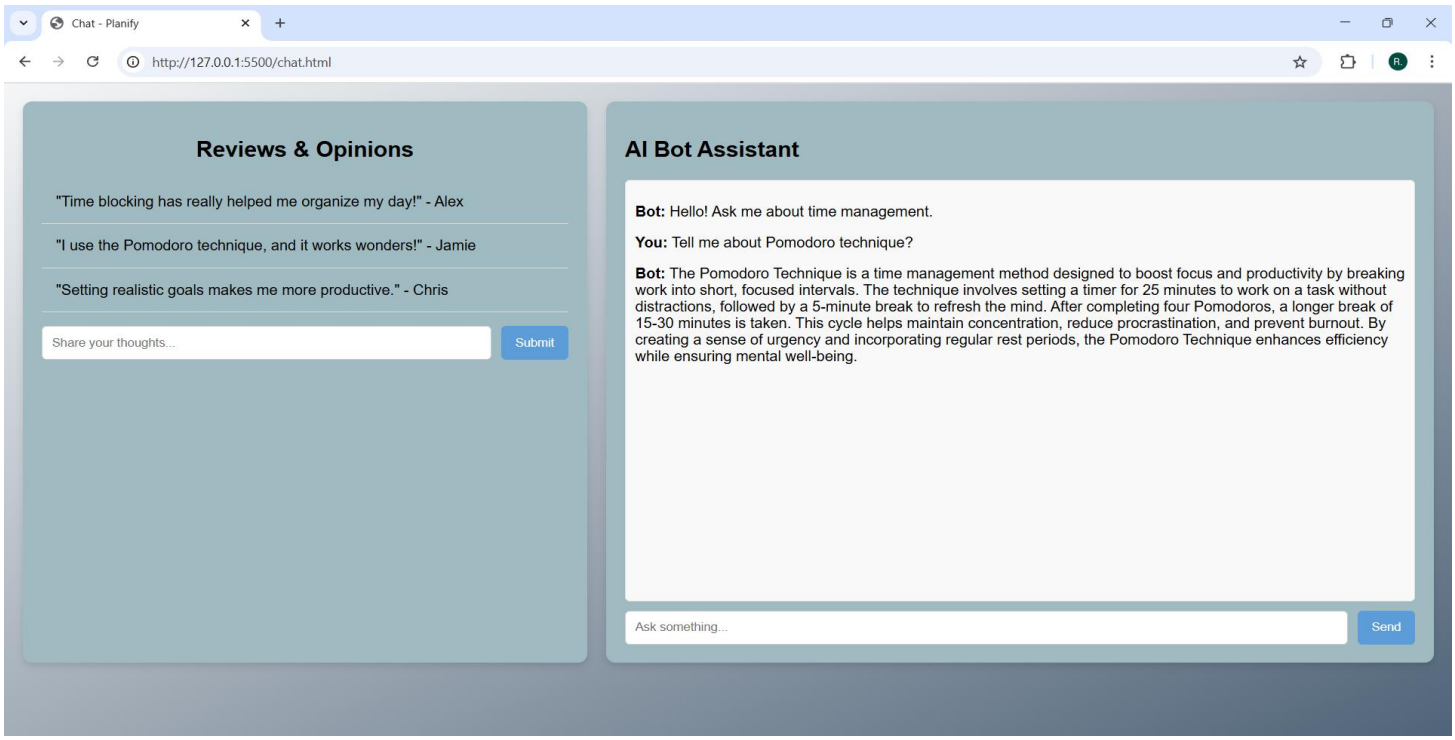

Task Completed
http://127.0.0.1:5500/completion.html
YOU HAVE COMPLETED YOUR TASK!
ITS FINALLY DONE!!!
DO ANOTHER TASK


Contact Us - Planify
http://127.0.0.1:5500/contact.html
Contact Us
Have any questions? Feel free to reach out to us.
How does Planify work?
Planify helps you manage your daily tasks by setting reminders, categorizing them, and ensuring you never miss anything important.
How do I mark a task as complete?
Can I delete a task?
Where can I view completed tasks?
Send Us a Message
Your Name
Your Email
Your Message
Send Message


Completed Tasks
http://127.0.0.1:5500/tasks.html
Completed Tasks
Buy groceries | 26/03/2025 | 17:40:00 | other
Meditate | 26/03/2025 | 05:50:00 | fitness
Gym workout | 26/03/2025 | 05:50:00 | fitness
Pickup drycleaning | 26/03/2025 | 21:40:00 | personal
Buy groceries | 26/03/2025 | 17:40:00 | other
Take out trash | 24/03/2025 | 09:35:00 | personal


Calendar - Planify
http://127.0.0.1:5500/calender.html
Planify Calendar
March 2025
Prev
Next
Sun
Mon
Tue
Wed
Thu
Fri
Sat
1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28
29
30
31
Add Task for Selected Date
Enter task
Add Task


Task Progress Tracker
http://127.0.0.1:5500/progress.html
Track Your Task Progress
Select a Task:
Complete Project Report
Daily Workout
Study for Exam
Read a Book
Selected Task: None
0
%
Update Progress


Chat - Planify
http://127.0.0.1:5500/chat.html
Reviews & Opinions
"Time blocking has really helped me organize my day!" - Alex
"I use the Pomodoro technique, and it works wonders!" - Jamie
"Setting realistic goals makes me more productive." - Chris
Share your thoughts...
Submit
AI Bot Assistant
Bot: Hello! Ask me about time management.
You: Tell me about Pomodoro technique?
Bot: The Pomodoro Technique is a time management method designed to boost focus and productivity by breaking work into short, focused intervals. The technique involves setting a timer for 25 minutes to work on a task without distractions, followed by a 5-minute break to refresh the mind. After completing four Pomodoros, a longer break of 15-30 minutes is taken. This cycle helps maintain concentration, reduce procrastination, and prevent burnout. By creating a sense of urgency and incorporating regular rest periods, the Pomodoro Technique enhances efficiency while ensuring mental well-being.
Ask something...
Send

5. CONCLUSION AND ENHANCEMENTS

The “Planify -Task Reminder” Website serves as a smart and efficient personal productivity tool, enabling users to create, manage, and track tasks with ease. With features like categorized task organization, real-time reminders, a visual calendar, and engaging task completion animations, the platform ensures an intuitive user experience. The addition of modules such as emergency task handling, user reviews, and an AI bot for engagement further enhances its functionality. Overall, "Planify" promotes better time management and encourages users to stay organized, productive, and motivated in their daily routines.

BIBLIOGRAPHY

- https://github.com/topics/todo-list-project
- https://www.geeksforgeeks.org/features-importance-and-limitations-of-planning/
- https://www.managementstudyguide.com/planning-function-articles.htm