Project Report
on
Minor Project

BACHELOR OF COMPUTER APPLICATIONS (BCA)

Submitted by:
Riaz Mohammad
(Roll No: 24/SCA/BCA(AI&ML)/39)
&
Pratyansh
(Roll No: 24/SCA/BCA(AI&ML)/52)
Under the Supervision of:

School of Computer Applications
Manav Rachna International Institute of Research and Studies Sector-43, Aravalli Hills, Faridabad – 121001
2025

Declaration

I do hereby declare that this project work entitled “My Minor Project Report” submitted by me for the partial fulfillment of the requirement for the award of BACHELOR OF COMPUTER APPLICATIONS (BCA) is a record of my own work. The report embodies the findings based on my study and observation and has not been submitted earlier for the award of any degree or diploma to any Institute or University.

Signature

Name: Riaz Mohammad Pratyansh

Roll No: 24/SCA/BCA(AI&ML)/39

Roll No: 24/SCA/BCA(AI&ML)/52

Date: 15

Acknowledgement

I gratefully acknowledge the assistance, cooperation, guidance and clarification provided by Kavita mam during the development of this internship report. My extreme gratitude to Taruna Chopra who guided me throughout the project. Without his willing disposition, spirit of accommodation, timely clarification and above all faith in me, this project could not have been completed in due time. I would like to extend my sincere gratitude to Prof. (Dr.) Suhail Javed Quraishi – HOD, Prof. (Dr.) Rashmi Agrawal – Associate Dean and Prof. (Dr.) Brijesh Kumar – Dean for their valuable teachings and advice. I want to thank all the department faculty members and non-teaching staff for their cooperation and support. This opportunity is a big milestone in my career development. I will strive to use the gained skills and knowledge in the best possible way and continue working on their improvement to attain my career objectives.

INDEX

1. Cover Page
2. Declaration
3. Acknowledgement
4. Introduction
5. Purpose & Objectives
6. Feasibility Study
7. Project Monitoring
8. System Analysis
9. System Design
10. Input/Output Form Design
11. System Testing
12. System Implementation
13. Documentation
14. Scope of the project
15. Bibliography

INTRODUCTION

This report details the design and implementation of a minor project: a Task Management System developed using Python. In today's fast-paced environment, effective organization and prioritization are essential for both personal productivity and team efficiency. Task management tools serve as crucial aids by providing a structured digital environment to define, track, and complete required activities.

Purpose & Objectives

Why build this?

Because productivity suffers when tasks are unorganized. A simple application can make a noticeable difference in efficiency.

Core Objectives

- Build a responsive task-tracking system
- Allow users to create, edit, update and delete tasks easily
- Store tasks permanently using browser storage
- Keep the UI clean, fast and beginner-friendly

Learning Goals

- Improve practical coding & debugging skills
- Understand state flow and rendering behavior
- Build something real that solves an everyday problem

Technologies Used

- Frontend -HTML, CSS, JavaScript, python
- Data Storage - localStorage
- Version Control - Git + GitHub
- Deployment - GitHub Pages

System Overview

The application follows a simple yet effective structure:

User → Creates Task → Task gets stored → UI updates instantly → User tracks progress

Key Features

- ❖ Add task with description
- ❖ Mark task as completed
- ❖ Edit tasks anytime
- ❖ Delete tasks if unnecessary
- ❖ Uses localStorage — data never disappears on refresh

System Design

When designing the application, I wanted the user experience to feel fast and smooth.
A task should take *seconds* to enter, not minutes.

So I structured the UI like this:

- A clear input area to enter a task
- A list view showing all tasks
- Buttons for `edit`, `delete`, `mark complete`

- Completed tasks visually look different (strikethrough or faded)
- Data saves automatically without asking the user

Implementation Summary

I began by setting up the interface — a task input box and display container.
Once the UI skeleton was ready, I implemented logic for:

1. Adding a Task
   A new object pushes into an array → stored in localStorage → rendered live.

2. Editing a Task
   Fetch task → update text → save again → re-render everything.

3. Marking as Completed
   Toggle `status = done` → change display style → saved permanently.

4. Deleting a Task
   Remove object from list → update storage → refresh UI.

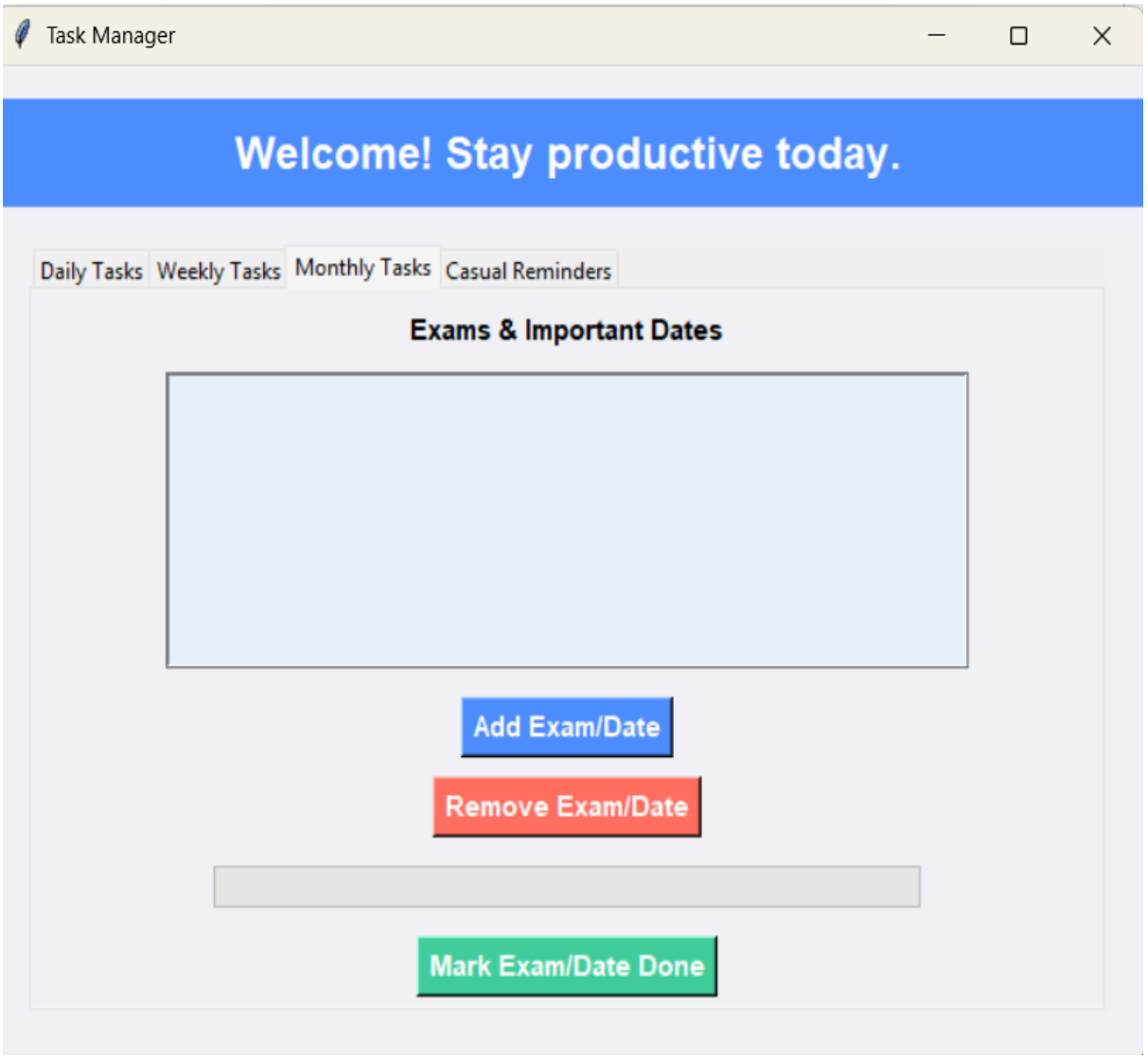
Every update is instant — no reload needed.

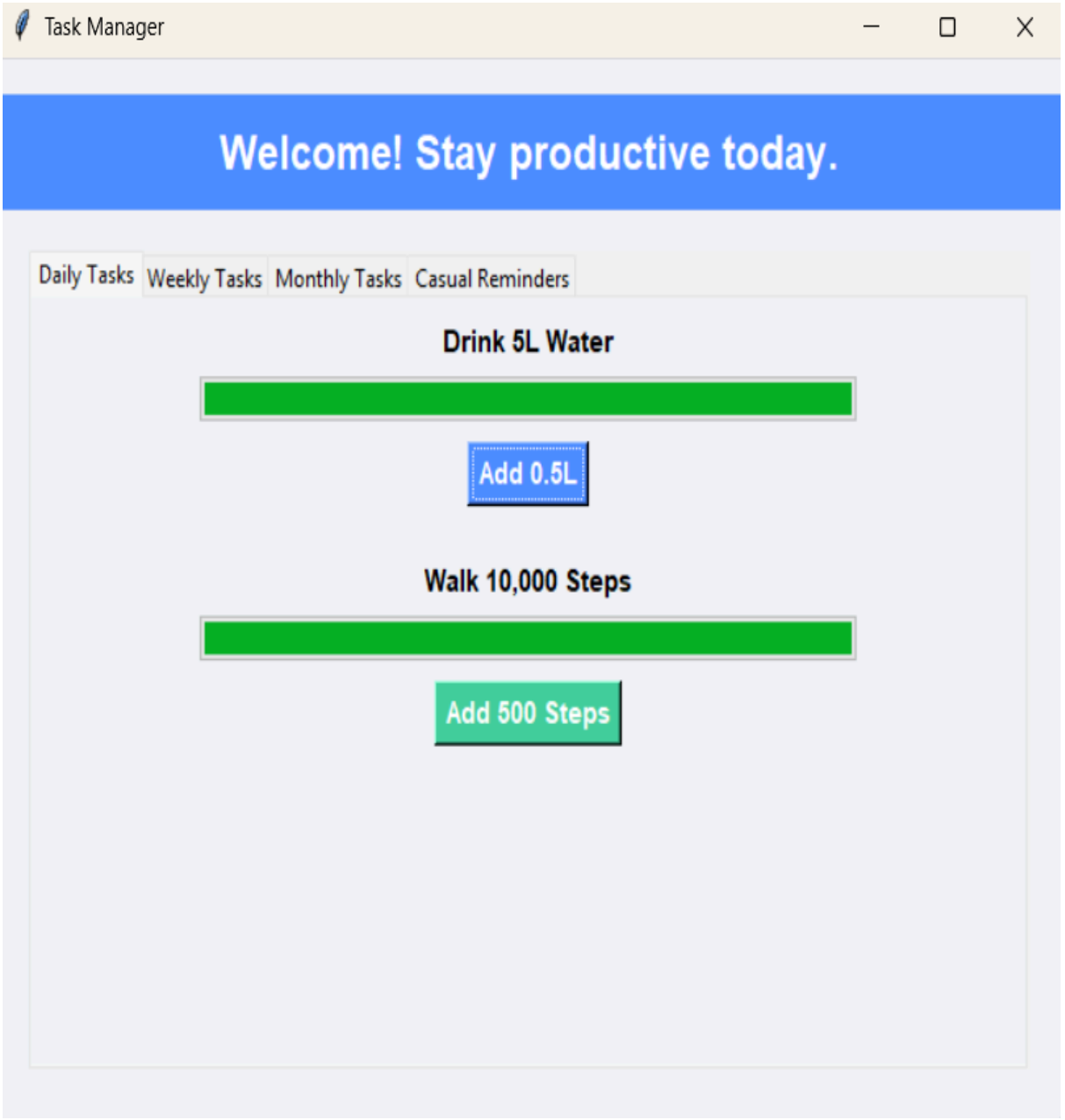
Testing & Observations

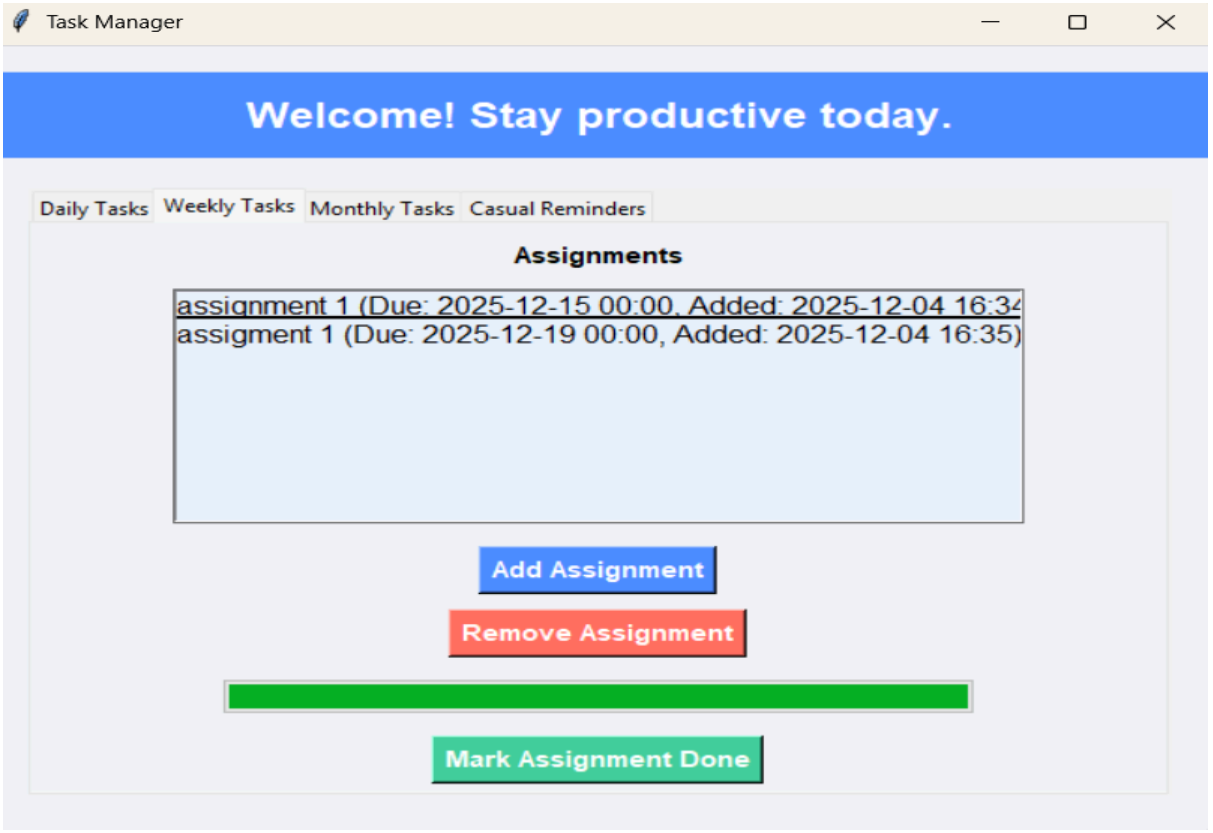
1. Add Task - Successfully stored & displayed

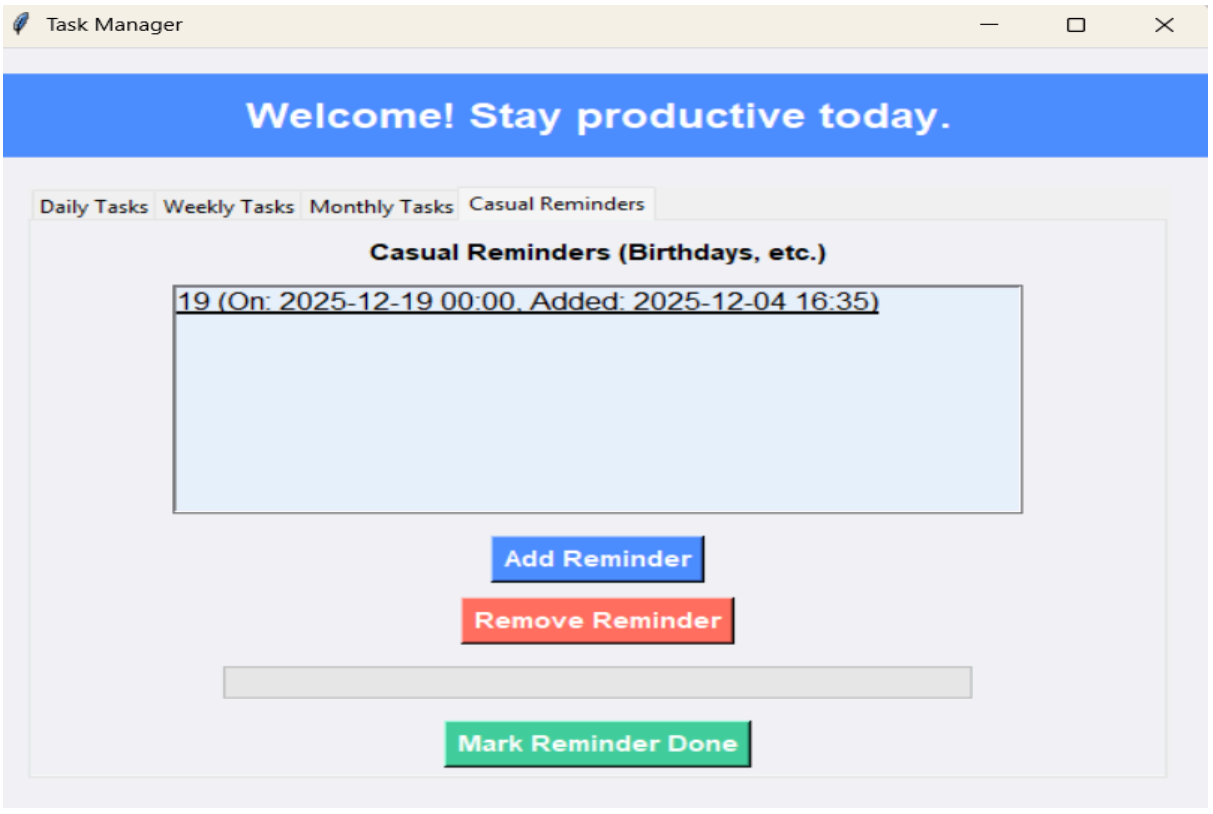
2. Edit Task - Updates correctly in interface & storage
3. Delete Task - Task disappears immediately
4. Refresh Browser - Data remains — localStorage works perfectly
5. Multiple Edits - No conflicts or overwrites

Input/Output Form Design




Task Manager
Welcome! Stay productive today.
Daily Tasks
Weekly Tasks
Monthly Tasks
Casual Reminders
Drink 5L Water
Add 0.5L
Walk 10,000 Steps
Add 500 Steps


Task Manager
Welcome! Stay productive today.
Daily Tasks
Weekly Tasks
Monthly Tasks
Casual Reminders
Assignments
assignment 1 (Due: 2025-12-15 00:00, Added: 2025-12-04 16:34
assigment 1 (Due: 2025-12-19 00:00, Added: 2025-12-04 16:35)
Add Assignment
Remove Assignment
Mark Assignment Done


Task Manager
Welcome! Stay productive today.
Daily Tasks
Weekly Tasks
Monthly Tasks
Casual Reminders
Casual Reminders (Birthdays, etc.)
19 (On: 2025-12-19 00:00, Added: 2025-12-04 16:35)
Add Reminder
Remove Reminder
Mark Reminder Done

Future Upgrades

I plan to enhance this application further with:

- Notifications & remainder alerts
- Calendar view integration
- Cloud sync & login system
- Task analytics (productivity graph)
- Mobile app version using React Native
- Dark mode, drag-and-drop sorting, voice add command

Bibliography

1. W3Schools. *HTML, CSS & JavaScript Tutorials and Syntax References.* Retrieved from: w3schools.com
2. GitHub Documentation. *Version Control, Repository Management & Source Deployment Guides.* Retrieved from: github.com
3. GeeksForGeeks. *Concept Articles on Web Development & JavaScript Implementation Techniques.* Retrieved from: geeksforgeeks.org
4. Your GitHub Project Repository. *Task Manager App Source Code — Development, Logic & UI Implementation.*
5. Link: github.com/Riaz1909/task-manager-app