TaskSnap Journal: A Productivity Tracker

BSDCH ZC229T: Capstone Project

by

Saipa Abhinaya

202117bh109

Design Project work carried out at

HCLTech, Nagpur

BIRLA INSTITUTE OF TECHNOLOGY & SCIENCE
PILANI (RAJASTHAN)

October 2025

TaskSnap Journal: A Productivity Tracker

BSDCH ZC229T: Capstone Project

by

Saipa Abhinaya

202117bh109

Design Project work carried out at

HCLTech, Nagpur

Submitted in partial fulfillment of B.Sc. (Design and Computing) degree programme

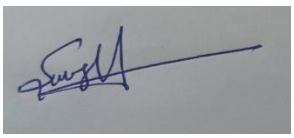
Under the Supervision of
Suresh Mamillapalli, Consultant
HCLTech, Bangalore

BIRLA INSTITUTE OF TECHNOLOGY & SCIENCE
PILANI (RAJASTHAN)

October 2025

CERTIFICATE

This is to certify that the Design Project entitled Productivity Tracker **TaskSnap Journal: A Productivity Tracker 2.0** submitted by **Saipa Abhinaya** having BITS ID No. **202117bh109** for the partial fulfilment of the requirements of the B.Sc. (Design and Computing) degree programme of BITS, embodies the bonafide work done by her under my supervision.



Place: Nagpur

Date: 04/10/25

Mentor's Signature:

Mentor's Name: Suresh Mamillapalli

Designation: Consultant

CONTENTS

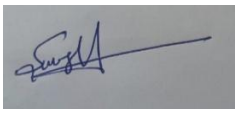
ABSTRACT

TaskSnap Journal: A Productivity Tracker 2.0

TASKSNAP JOURNAL is a desktop productivity application built in Python using the CustomTkinter framework. It was developed to directly address the organizational inefficiency of manual reporting (monthly productivity and weekly screen time) and the user pain point of task procrastination. The application provides a unified, Modern & Inviting platform that automatically tracks work metrics, monitors screen time, and enforces task completion. Key features include a compulsory To-Do List launch on login, proactive Productivity Module appearance before shift end, and a Screen Time Monitor with integrated break prompts. The application successfully demonstrates a modular architecture designed for efficiency and compliance.

.

Signature of Student:
Name: Saipa Abhinaya
Date: 4 Oct 2025
Place: Nagpur, India



Signature of Mentor:
Name: Suresh Mamilapalli
Date: 4 Oct 2025
Place: Nagpur, India

INTRODUCTION

1.1 Project Genesis

The TASKSNAP JOURNAL project originated from a direct organizational challenge: the constant requirement to report monthly productivity and weekly screen time. The reliance on fragmented, manual systems led to:

- Stressful end-of-month scramble and inaccurate data.
- Procrastination in task and data logging.
- Inefficient use of time that should have been spent on core work.

1.2 Our Solution

The core motivation is to create an assertive, automated desktop application that eliminates manual reporting friction. By integrating the tracking and reporting mechanism into the daily workflow, TASKSNAP JOURNAL ensures data is captured reliably and automatically. The application is built on a Modern & Inviting aesthetic designed for user comfort and consistent engagement.

DESIGN AND SYSTEM ARCHITECTURE

2.1 User Interface Design

The UI adheres to a "Modern & Inviting" aesthetic, defined by:

- ***Soft Color Palette***: Muted blues, greens, and grays for a comfortable, professional feel.
- ***Rounded Elements*:** CustomTkinter is used to create smooth, rounded buttons, cards, and input fields.
- ***Proactive UI**:* The design incorporates assertive elements to enforce adherence: the To-Do List auto-launches on login, and the Productivity module appears before shift end.

2.2 Technology Stack

The technical foundation of TASKSNAP JOURNAL is built on a highly maintainable, modular stack:

- ***Frontend***: Python with the CustomTkinter framework.
- Architecture: Modular Views Pattern (separated files like dashboard_view.py), crucial for scalability.
- ***Data Persistence***: Local File Storage (JSON/CSV) for immediate access, complemented by organizational reporting via the Google Sheets API.
- ***Cross-Platform Reporting***: Integration with the Google Sheets API to automatically update a single shared Excel sheet with all users' productivity and screen time data, eliminating manual consolidation.
- ***Automation***: Python Threading & Scheduling for continuous screen monitoring and automated events.

Implementation Details: Automation and Features

3.1 Initial User Setup

The initial module collects essential data necessary to enable automation and reporting:

- ***Information***: User Name, Email, Shift Time, Weekly Week-Offs.
- ***Reporting Chain***: Manager's Email Address.
- ***Module***: The Update Info section retains this form for ongoing detail modification.

3.2 Productivity Tracker

This module is the core component for compliance and reporting:

- ***Core Input***: Users update discrete counters for QA, Incidents, Packages, and PRF Created.
- ***Automation Trigger***: The module is configured to automatically appear on the user's screen 15 minutes before the shift ends, forcing data finalization.
- ***Hamburger Menu***: Provides essential extensions: Send Email (sends monthly summary via API), Misc Tasks (log for uncounted tasks), Edit Data, and Summary (visual charts).
- ***Reporting Integration***: Daily productivity data is immediately pushed to the shared Google Sheet via API.

3.3 To-Do List

This module is designed for compulsory task adherence:

- ***Assertive Launch***: The To-Do window automatically launches upon user login.
- ***Enforced Completion***: The module remains visible until the user has actively checked off all tasks.

3.4 Screen Time Monitoring

This module supports digital wellness and compliance reporting:

- ***Tracking***: Shows Total Screen Time, detailed daily usage bars, an average usage bar, a pie chart by category, and a list of all apps used.
- ***Digital Wellness***: The app automatically prompts the user to take a break after every 1 hour of continuous screen usage.
- ***Reporting Integration***: Weekly screen time summaries are automatically uploaded to the shared Google Sheet.

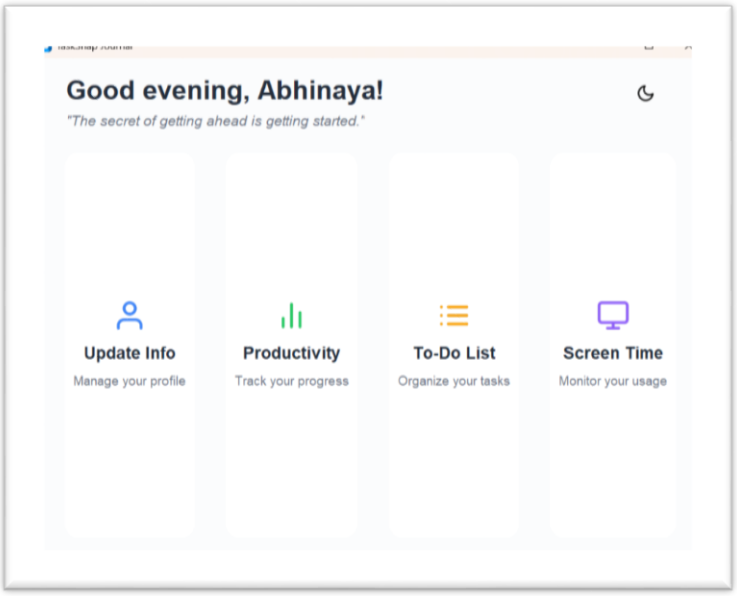
Conclusion and Future Work

TASKSNAP JOURNAL successfully delivers a unified, automated, and visually appealing solution to the problem of manual corporate reporting. The implementation leveraging Python/CustomTkinter and the Google Sheets API ensures a stable, scalable, and fully integrated platform.

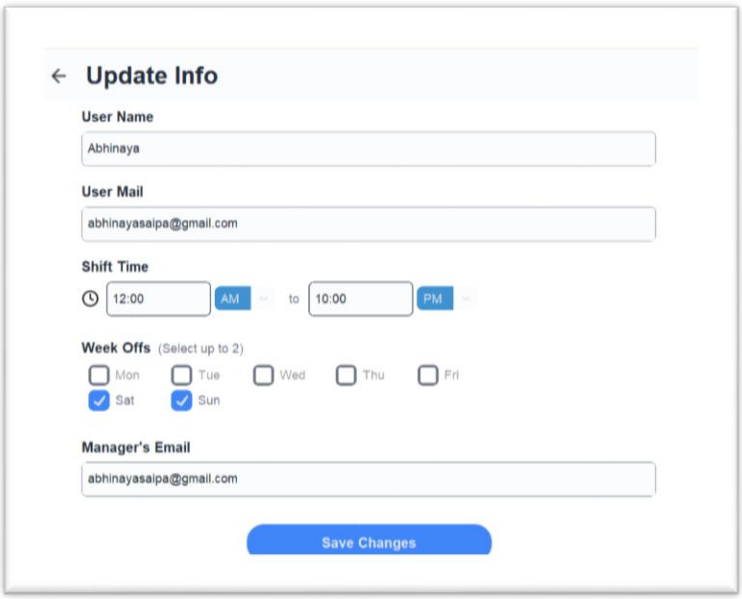
Future work includes integrating a persistent database and fully implementing the planned AI Mood Tracker for personalized, proactive user support

RESULTS

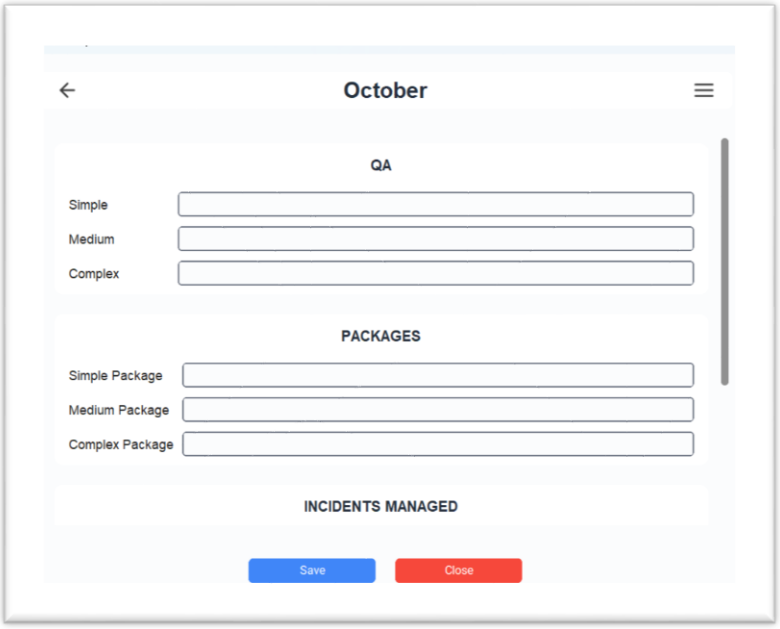
Main Window Overview



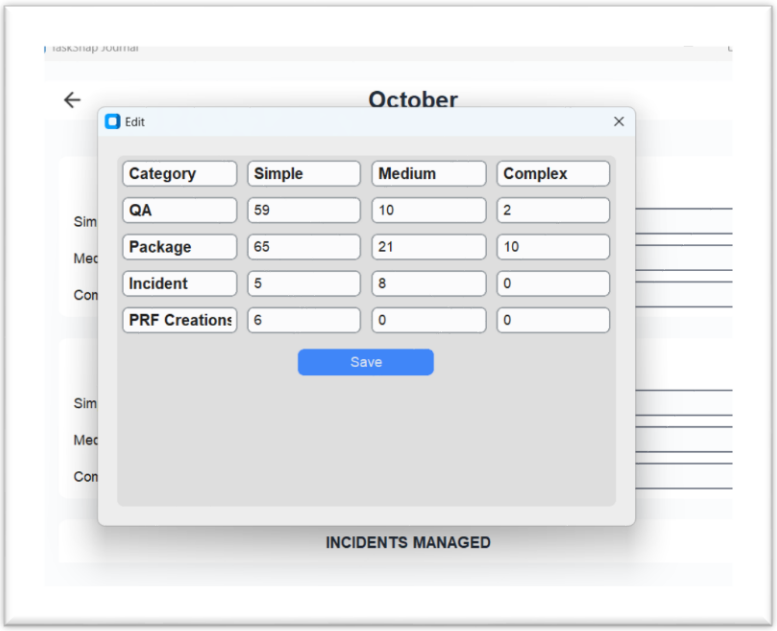
Update Info



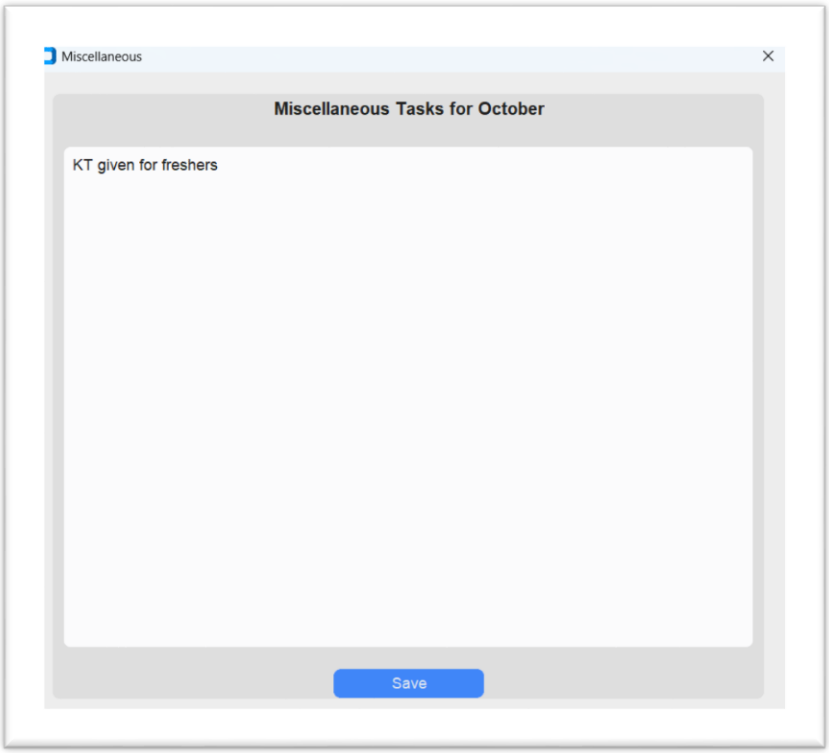
Productivity



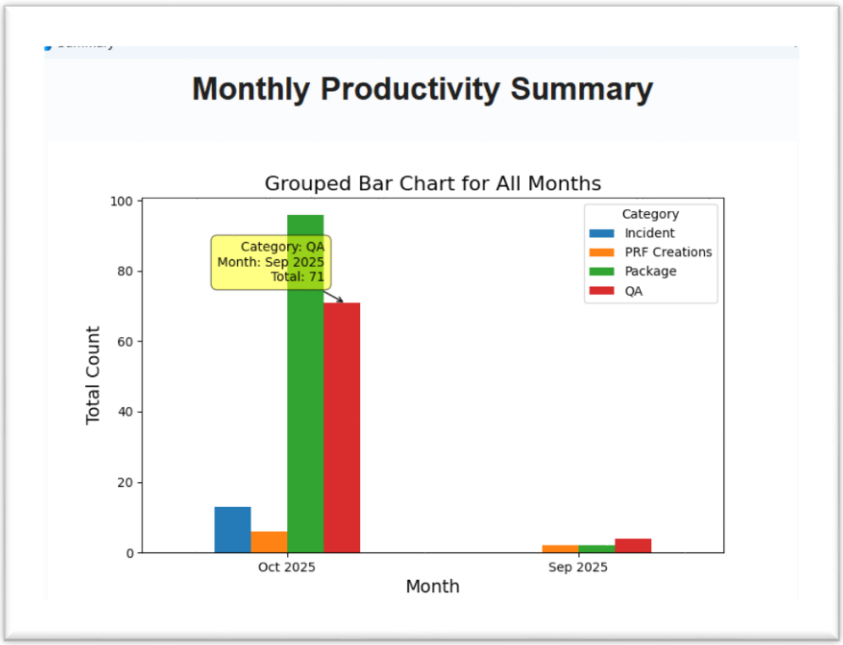
Edit Window



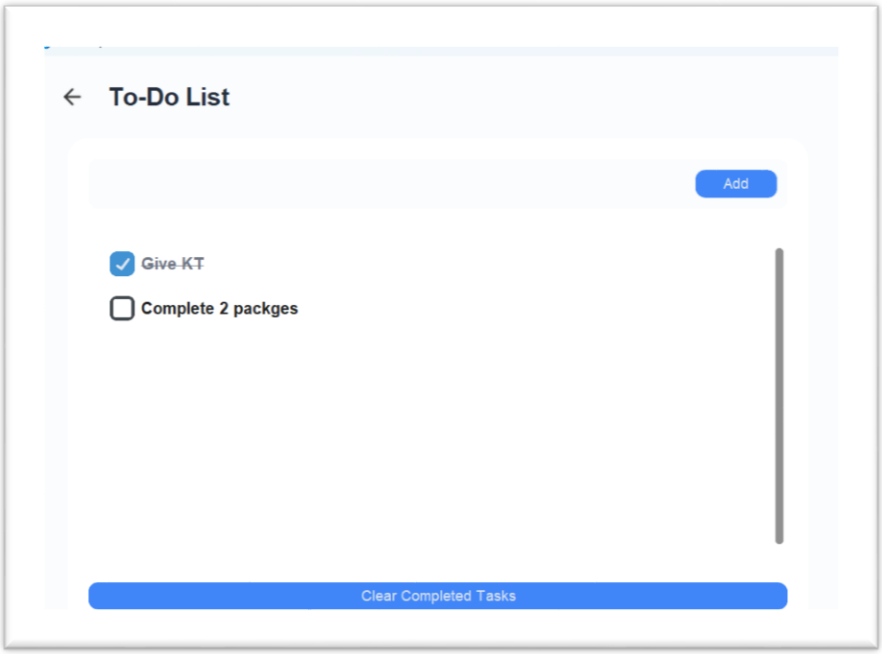
Miscellaneous Task Window



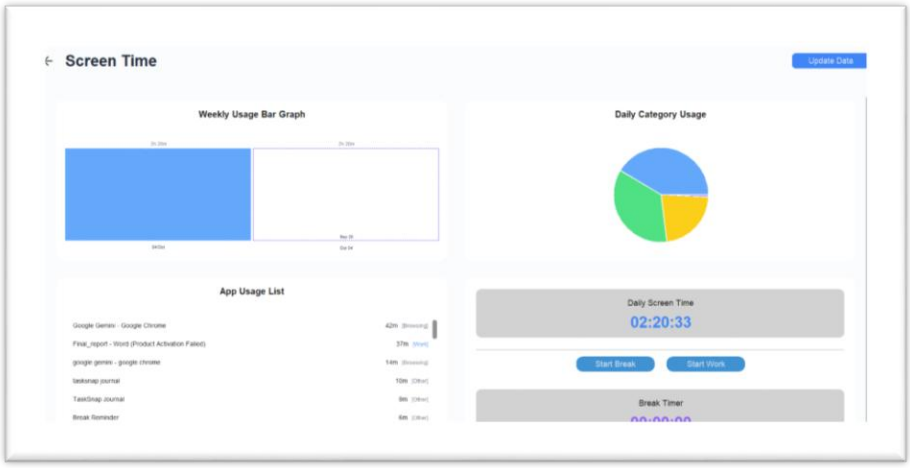
Summary



To do List



Screen Time



Scheduled Tasks

Successful creation of scheduled tasks using the Windows Task Scheduler ensures timely reminders for users.

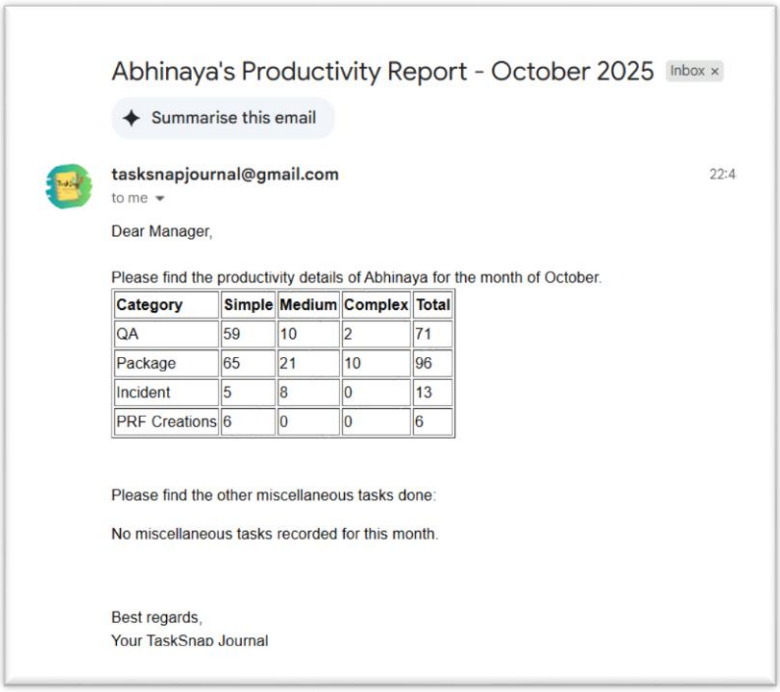
1. Daily Launch – triggers the launch of the productivity module 15mins before the users shift end time.
2. TODOViewLaunch – triggers the todo module when the user logs in.

TaskSnap-DailyLaunch	Ready	At 21:45 every Sunday, Monday, Tuesday, Wednesday, Thursday, Friday, Saturday of every week, starting 04-10-2025	05-10-2025 21:45:00	3
TaskSnap-ToDOViewLaunch	Ready	At 00:15 every Sunday, Monday, Tuesday, Wednesday, Thursday, Friday, Saturday of every week, starting 04-10-2025	05-10-2025 00:15:00	3

Email Configuration

The snapshot below showcases the email format, emphasizing the structure and content of the emailed productivity reports.

Fig-6



Abhinaya's Productivity Report - October 2025 Inbox ×

Summarise this email

tasksnapjournal@gmail.com 22:4

to me

Dear Manager,

Please find the productivity details of Abhinaya for the month of October.

Category	Simple	Medium	Complex	Total
QA	59	10	2	71
Package	65	21	10	96
Incident	5	8	0	13
PRF Creations	6	0	0	6

Please find the other miscellaneous tasks done:

No miscellaneous tasks recorded for this month.

Best regards,
Your TaskSnap Journal

REFERENCES

- Custom Tkinter Documentation
  https://customtkinter.tomschimansky.com/documentation/

- Python Official Documentation
  https://docs.python.org/

- Matplotlib Documentation
  https://matplotlib.org/stable/contents.html

- Pandas Documentation
  https://pandas.pydata.org/pandas-docs/stable/index.html

- win32com.client Documentation
  https://docs.microsoft.com/en-us/office/vba/api/overview/excel

- smtplib Documentation
  https://docs.python.org/3/library/smtplib.html

- Task Scheduler API Documentation
  https://docs.microsoft.com/en-us/windows/win32/taskschd/task-scheduler-start-page

- OpenAI GPT-3 Documentation
  https://beta.openai.com/docs/

- Feather Icons
  https://feathericons.com/

- Stack Overflow and other community forums for problem-solving.

CHECKLIST OF ITEMS

1.	**Is the final report neatly formatted with all the elements required for a technical Report?**	Yes
2.	Is the Cover page in proper format as given in Annexure A?	Yes
3.	Is the Title page (Inner cover page) in proper format?	Yes
4.	(a) Is the Certificate from the Mentor in proper format? (b) Has it been signed by the Mentor?	Yes Yes
5.	Is the Abstract included in the report properly written within one page? Have the technical keywords been specified properly?	Yes Yes
6.	Is the title of your report appropriate? **The title should be adequately descriptive, precise and must reflect scope of the actual work done.** Uncommon abbreviations / Acronyms should not be used in the title	Yes
7.	Have you included the List of abbreviations / Acronyms, if any?	No
8.	Does the Report contain a summary of the literature survey, if any?	No
9.	**Does the Table of Contents include page numbers?** (i). Are the Pages numbered properly? (Ch. 1 should start on Page # 1) (ii). Are the Figures numbered properly? (Figure Numbers and Figure Titles should be at the bottom of the figures) (iii). Are the Tables numbered properly? (Table Numbers and Tabl e Titles should be at the top of the tables) (iv). Are the Captions for the Figures and Tables proper? (v). Are the Appendices numbered properly? Are their titles appropriate	Yes Yes Yes Yes Yes Yes
10.	Is the conclusion of the Report based on discussion of the work?	Yes
11.	Are References or Bibliography given at the end of the Report? Have the References been cited properly inside the text of the Report? Are all the references cited in the body of the report	Yes Yes Yes
12.	Is the report format and content according to the guidelines? The report should not be a mere printout of a Power Point Presentation, or a user manual. Source code of software need not be included in the report.	Yes

Declaration by Student:

I certify that I have properly verified all the items in this checklist and ensure that the report is in proper format as specified in the course handout.

Place: Nagpur

Date: 4/10/2025

Signature of the Student

Name: Saipa Abhinaya

ID No.: 202117bh109