****

**PURBANCHAL UNIVERSITY**

KULESHWOR AWAS CAMPUS

FIRST SEMESTER PROJECT

ON

“CALENDAR MANAGEMENT SYSTEM”

In the partial fulfillments for the requirement of the first Semester Project-I (Subject code-………) in the completion of Bachelor of Information Technology (BIT) degree at Kuleshwor Awas Campus, under Purbanchal University.

Submitted By: Submitted To:

Name: Pasang Dorje Tamang Purbanchal University

Name: Prasanna Katuwal

2025-6-9

# STUDENT’S DECLARATION

We following students, hereby declare that the Project Report is titled

“Calendar Management System” is a result of our own work and our indebtedness to other work publications, references, if any, have been dully acknowledged. If we are found guilty of copying any other report or published information and showing as our original work, we understand that we shall be liable and punishable by Purbanchal University, which may include fail in examination, ‘Repeat study and re-submission of the report’ or any other punishment the Purbanchal University may decide.

We further certify that this Project submitted in partial fulfillment of the requirement for the award of Bachelor in Information Technology (BIT) of the Purbanchal University is our original work and has not been submitted for award of any other degree or other similar title or prizes.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| S.N | Name | Redg | Roll no. | Signature |
| 1. | Pasang Dorje Tamang |  |  |  |
| 2. | Prasanna Katuwal |  |  |  |

# 

# EXAMINER’S CERTIFICATION

This is to certify that the project titled **“Calendar Management System”** has been successfully completed by **Mr. Pasang Dorje Tamang (S/N: ……)** and **Mr. Prasanna Katuwal (S/N: ……)** as part of the requirements for the **Bachelor of Information Technology** degree under **Purbanchal University.**The project was carried out during the academic year **2025**, under the supervision of **Mr. Narayan G.C.**

# ACKNOWLEDGEMENT

We take this opportunity to express our profound appreciation and unfathomable regards to the Information Technology (IT) department for this commendable guidance, monitoring and constant encouragement throughout the course of this project. The help and guidance given by shall carry us a long way, in the journey on which we are about to embark.

We also take this opportunity to express a deep sense of gratefulness to Mr. Narayan GC for his amiable support, valuable information and guidance, which helped us in completing this task throughout its various stages. We also want to thank our program director Mr. Narayan G.C. who helped us a lot during the complete project by giving us his precious time. We are also indebted to all members of Kuleshwor Awas Campus, for the help provided by them in their in their respective fields. We are grateful for their cooperation during the period of our project.

Finally we would also like to express lots of thanks to PURBANCHAL UNIVERSITY for designing such a wonderful course structure. It will help us to get more knowledge in the field of Information Technology & help us to have a bright future in the field of technology.

# ABSTRACT

Table of Contents

[STUDENT’S DECLARATION I](#_Toc201125980)

[EXAMINER’S CERTIFICATION II](#_Toc201125982)

[ACKNOWLEDGEMENT III](#_Toc201125983)

[1. BACKGROUND 1](#_Toc201125984)

[I. Abstract of the project 1](#_Toc201125985)

[II. INTRODUCTION 2](#_Toc201125986)

[III. OBJECTIVE OF THIS PROJECT 2](#_Toc201125987)

[IV. FUTURE IMPLEMENTATION FOR THE PROJECT 3](#_Toc201125988)

[V. PROCESS IN THE PROJECT USE 4](#_Toc201125989)

[VI. INTRODUCTION TO SOFTWARE / LANGUAGE USED 4](#_Toc201125990)

[2. SYSTEM RECOMMENDATIONS: 5](#_Toc201125991)

[3. PROBLEM FACED 5](#_Toc201125992)

[4. System design 6](#_Toc201125993)

[Step 1: Display main menu with options: 6](#_Toc201125994)

[Step 2: Wait for user input (1–6) and process based on selected option. 6](#_Toc201125995)

[Step 3: **If Option 1 (View Calendar):** 6](#_Toc201125996)

[Step 4: If Option 2 (Add Event): 6](#_Toc201125997)

[Step 5: If Option 3 (View Event): 6](#_Toc201125998)

[Step 6: If Option 4 (Delete Event): 7](#_Toc201125999)

[Step 7: If Option 5 (To-Do List): Display submenu: 7](#_Toc201126000)

[Step 8: If Option 6 (Exit): 7](#_Toc201126001)

[Step 9: Stop 7](#_Toc201126002)

[5. SNAPSHOTS OF THE SYSTEM 9](#_Toc201126003)

[6. LIMITATIONS 11](#_Toc201126004)

[7. CONCLUSIONS 12](#_Toc201126005)

[8. BIBLIOGRAPHY 13](#_Toc201126006)

# BACKGROUND

## Abstract of the project

The **Calendar Management System** project is developed using the **C programming language** to provide an efficient and user-friendly solution for organizing and managing schedules. The main goal of this system is to simplify the process of scheduling events, appointments, and to-do lists, which can be cumbersome with manual methods.

This software enables users to easily add, view, and modify calendar entries, ensuring quick access to important dates and tasks. Designed with simplicity and interaction in mind, it aims to enhance productivity by helping users manage their time effectively. By automating the calendar management process, this project offers a reliable and practical alternative to traditional paper-based systems.

As we are beginners with limited practical experience in software development, and considering that calendar-based systems can have a wide range of features, we decided to keep the scope of our project simple and focused. Our Calendar Management System covers the following key areas:

Viewing calendar by month and year

Adding, viewing, and deleting scheduled events

Managing a personal to-do list

Marking and viewing completed tasks

## INTRODUCTION

Time management plays a crucial role in both personal and professional life. As our daily tasks, responsibilities, and deadlines continue to grow, having a well-organized scheduling system has become more important than ever. A Calendar Management System helps users keep track of important dates, plan events, and manage tasks efficiently.

Traditionally, people rely on notebooks or basic reminders, which can be unreliable and hard to maintain over time. Our project, "Calendar Management System", aims to provide a simple yet effective digital solution to these problems. It allows users to view monthly calendars, add and manage events, and maintain a to-do list with the ability to mark tasks as completed.

This system is especially helpful for students, professionals, and anyone looking to organize their daily life more efficiently without relying on external applications. It reduces the chances of missing important dates and helps users stay productive and on schedule.

## OBJECTIVE OF THIS PROJECT

* To provide a simple and efficient way for users to manage dates, events, and daily tasks.
* To help users schedule and view events on specific dates with ease.
* To improve time management by allowing users to maintain and track a personal to-do list.
* To reduce the chances of missing important dates or tasks.
* To minimize the use of paper for planning and organizing schedules.
* To ensure faster access to past and upcoming events and tasks through a user-friendly interface.
* To avoid data redundancy by storing events and tasks in organized digital format using file handling.

## FUTURE IMPLEMENTATION FOR THE PROJECT

The Calendar Management System has strong potential for future improvements and expansion. While this version focuses on basic calendar viewing, event scheduling, and task management, there are many possibilities for enhancing the system further. As technology evolves, the system can be updated to offer more advanced features, improve user experience, and increase flexibility. The current limitations can be overcome with better tools and techniques in the future.

Below are some areas for future enhancement:

* Developing the system as a web-based or mobile application to allow access from anywhere, at any time
* Adding multi-user support so multiple people can manage and share calendars or tasks
* Integrating a database to store events and to-do lists more securely and efficiently
* Improving the user experience by implementing a graphical user interface
* Enabling automatic reminders or notifications for upcoming events or deadlines
* Allowing custom reports and statistics for task completion and scheduling habits
* Enhancing the system with login and authentication features for better privacy and security

These improvements will not only make the system more powerful but also more useful in real-world applications. As users' needs grow, the system can be scaled and adapted to meet those requirements.

## PROCESS IN THE PROJECT USE

This software is designed to be used by individuals who want to manage their daily schedules, events, and tasks. The system provides simple menu-driven access without requiring a login, making it user-friendly and easily accessible for beginners. Once the user starts the program, they can view calendars, add or delete events, and manage a personal to-do list. Each section of the system is organized to handle different types of information, such as scheduled events, pending tasks, and completed activities. The goal is to help users stay organized and manage their time more effectively.

## INTRODUCTION TO SOFTWARE / LANGUAGE USED

C is a powerful general-purpose programming language developed in the early 1970s by Dennis Ritchie at Bell Labs. It is widely used for system programming and application development due to its efficiency and ability to work closely with hardware. C is considered a middle-level language because it combines features of both high-level and low-level programming. Its simplicity and speed make it well-suited for developing software like this calendar management system, which involves handling dates, events, and task management efficiently.

# SYSTEM RECOMMENDATIONS:

Minimum System Requirements

**Hardware:**

* PC with Pentium II Processor (260 MHz) or higher
* 32 MB RAM or more
* Color Monitor (preferred)
* Hard disk with at least 50 MB of free space

**Software:**

Windows Operating System (Windows XP, Windows 7, Windows 8, Windows 10)

# PROBLEM FACED

The main challenges faced during the development of this software were:

* Complexity in coding and designing the system logic.
* Limited prior experience with C programming made it initially difficult to understand some programming concepts and manage file handling.
* Difficulty in working with input/output operations and managing data storage using text files.
* Due to limited features and tools, the software could not include advanced functionalities or a more attractive interface.
* Managing date and time-related operations accurately posed some challenges in ensuring the calendar and event system worked smoothly.

# System design

* 1. **Algorithm:**

The system consist of different sets of algorithms:

### **Step 1:** Display main menu with options:

* View Calendar
* Add Event
* View Event
* Delete Event
* To-Do List
* Exit Program

### **Step 2:** Wait for user input (1–6) and process based on selected option.

### **Step 3: If Option 1 (View Calendar):**

* Ask for year and month.
* Calculate first weekday of the month.
* Display calendar for selected month.

### **Step 4: If Option 2 (Add Event):**

* Ask for event date.
* Validate that it’s a future date.
* Ask for event description.
* Save to event.txt.

### **Step 5: If Option 3 (View Event):**

* Read and display all events from event.txt.
* If none, show “No event found”.

### **Step 6: If Option 4 (Delete Event):**

* Show existing events.
* Ask for date to delete.
* Remove matching entry from file.

### **Step 7: If Option 5 (To-Do List):** Display submenu:

1. Add To-Do
2. View To-Do
3. Mark as Completed
4. View Completed
5. Return to Main Menu

Each sub-option performs corresponding file operations (todo.txt, complete.txt).

### **Step 8: If Option 6 (Exit):**

* Terminate program using return 0.

### **Step 9: Stop**

* 1. **FLOW CHART**

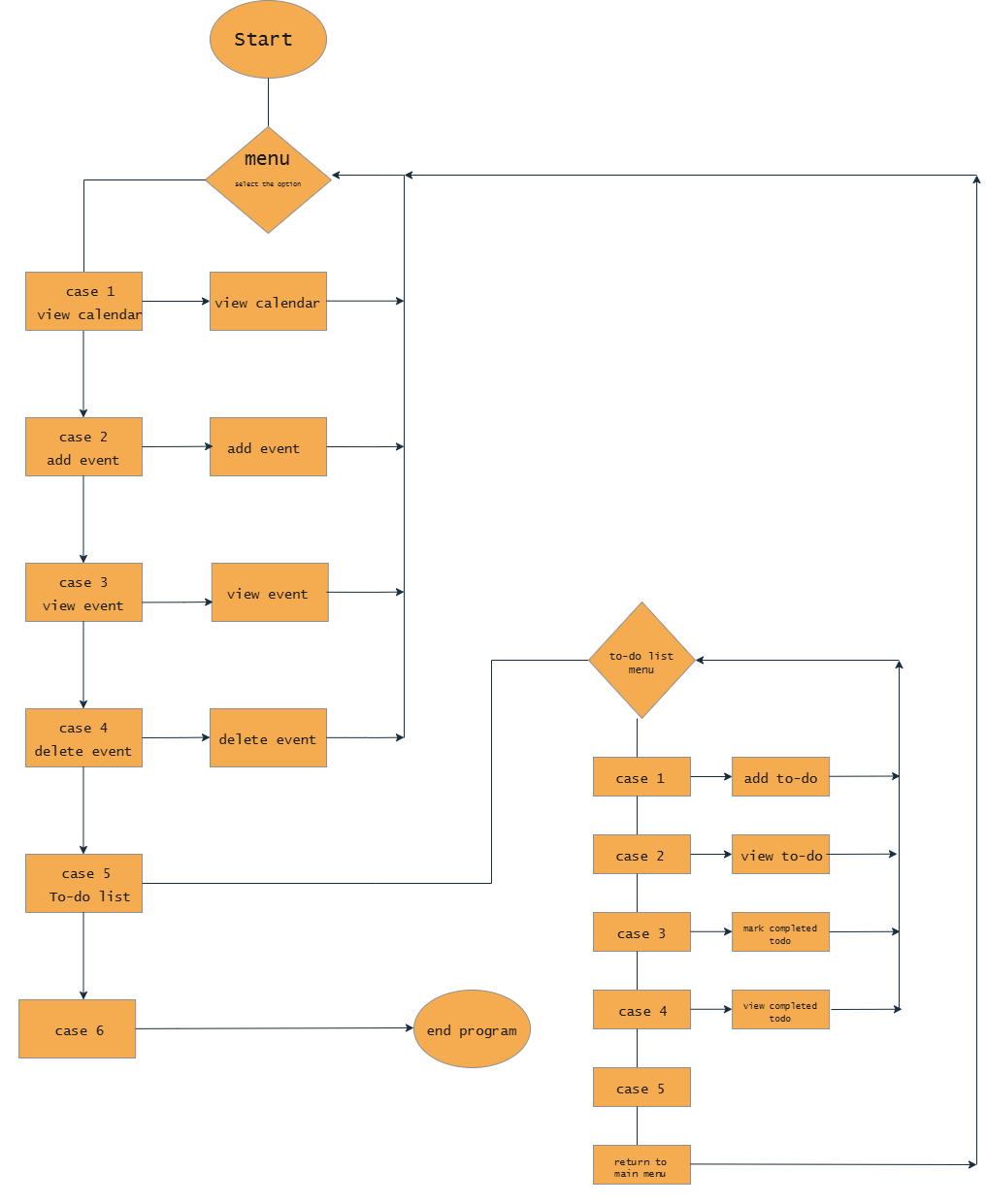


Figure 1 Flow Chart

# SNAPSHOTS OF THE SYSTEM

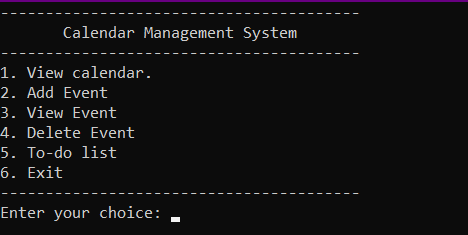


Figure 2 Main Menu

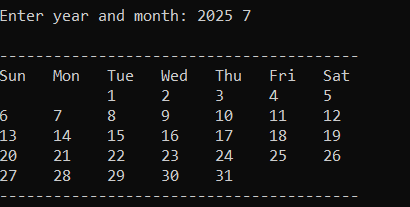


Figure 3 Calendar

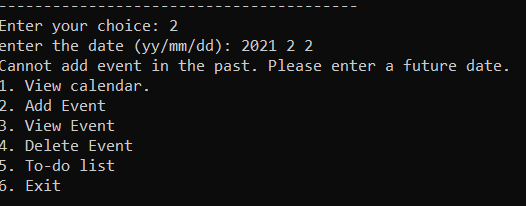


Figure 4 Can;t add event in Past Dates

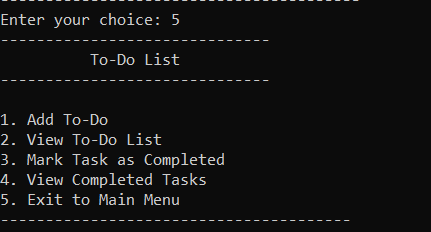


Figure 5 To-do list menu

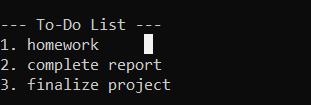


Figure 6 View To-do list

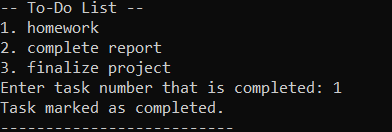


Figure 7 Mark To-do as complete

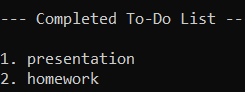


Figure 8 Completed TO-DO

# LIMITATIONS

The following are the Limitations of the project-

* No graphical user interface; only runs in command-line mode.
* No reminder pop-up or event notification system.
* Does not validate invalid dates like 31/02/2025.
* No option to search or filter events.
* No login, password protection, or data security.
* Single-user only; does not support multiple user profiles.
* To-do tasks lack deadlines, priorities, and categories.
* Calendar view does not highlight event dates.
* Uses plain text files with no backup or recovery system.
* Not accessible via mobile or web platforms.

# CONCLUSIONS

Working on this Calendar Management System project has been a great learning experience for us. It gave us the chance to explore different concepts in C programming, especially related to file handling, modular functions, and date management. We really enjoyed building something that’s practical and useful, even if it runs in a simple command-line interface.

With the support of our teachers and friends, we were able to complete the project successfully. It wasn’t just about writing code — we also learned how to structure a complete program, handle errors, and think about how users will interact with it. These lessons will definitely help us in future academic projects and professional work.

At the same time, we understand that the project isn’t perfect. There are still many features we’d like to add — like reminders, a proper user interface, and maybe even online access. But as a starting point, this project helped us build confidence and gave us a strong foundation to work on more complex applications in the future.

# BIBLIOGRAPHY