Smart Calendar Documentation:-

Introduction:

The **Smart Calendar with Event Prioritization** is an innovative application designed to help users manage their schedules efficiently. By integrating event prioritization features, this calendar goes beyond traditional time management tools. The app uses advanced data structures to handle events based on deadlines, importance, and user-defined criteria. The graphical user interface (GUI) ensures a seamless experience for users to add, edit, and organize their events. Features such as drag-and-drop scheduling, conflict detection, and real-time notifications make it an ideal solution for personal and professional time management.

Requirements:

1. **Functional Requirements**:
    - Users can add, edit, delete, and prioritize events.
    - Drag-and-drop functionality to rearrange events easily.
    - Notifications for upcoming events.
    - Conflict detection to avoid overlapping events.
2. **Non-Functional Requirements**:
    - User-friendly and responsive interface.
    - Secure data storage for event information.
    - Efficient event retrieval and management.
3. **Hardware Requirements**:
    - A system with at least 2 GB RAM.
    - 100 MB of available disk space.

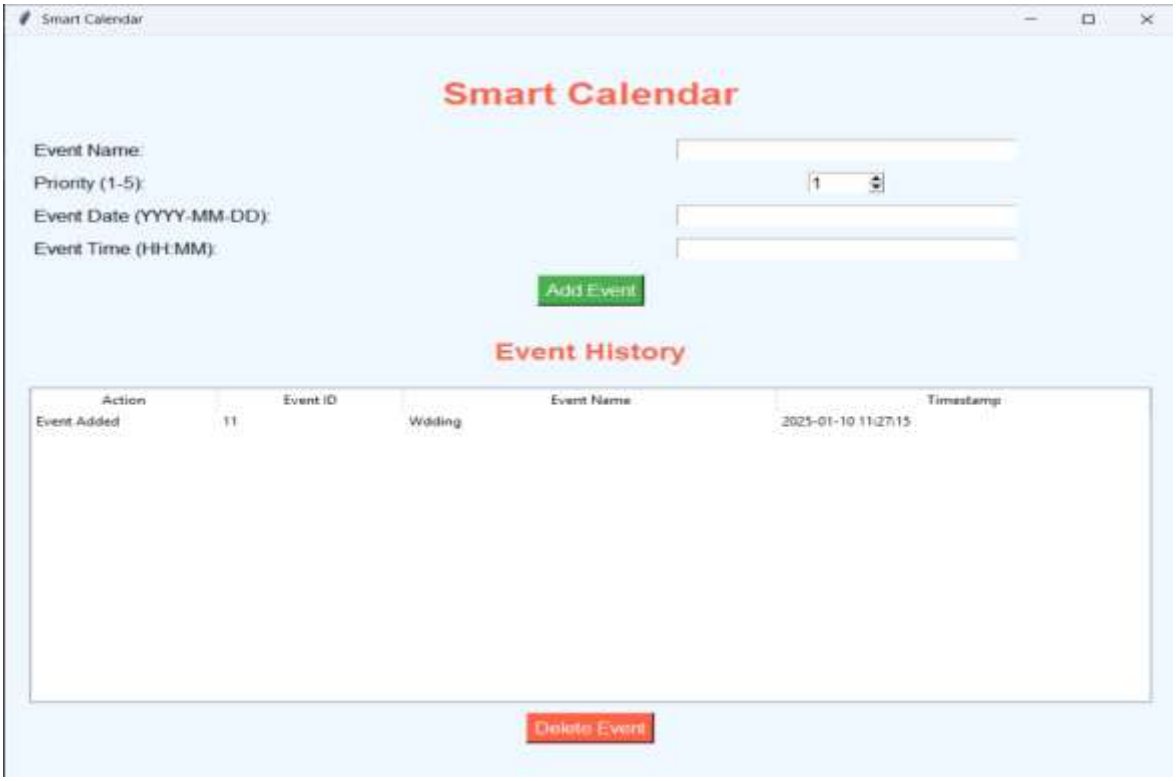
- Any operating system supporting Python.

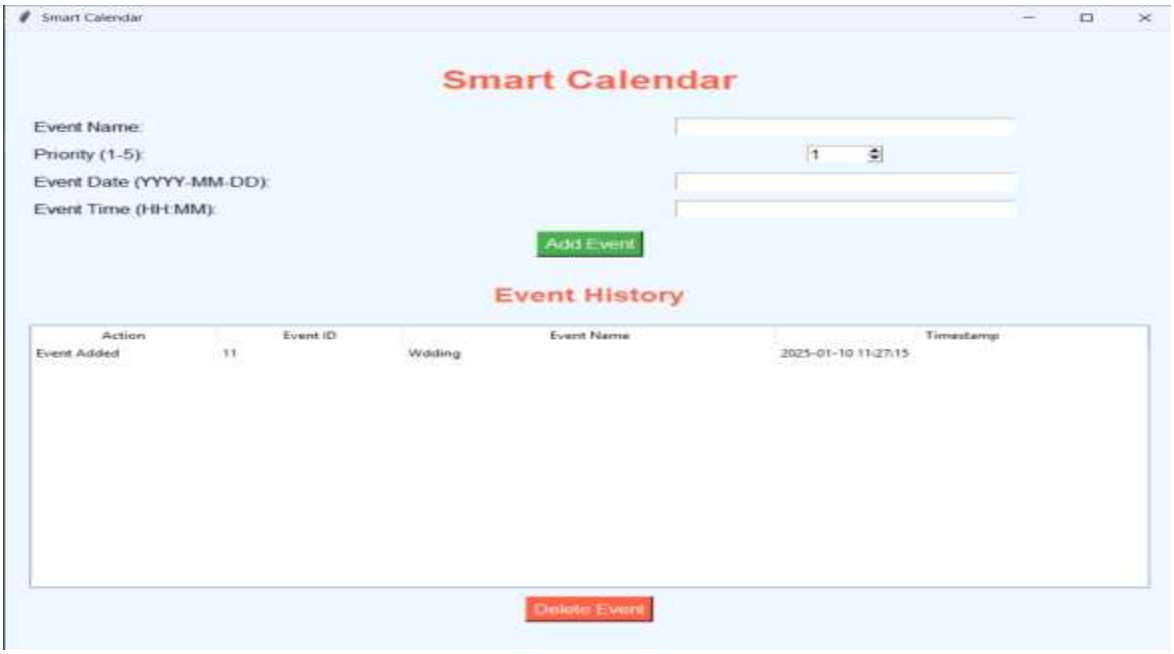
4. **Software Requirements**:
    - Python 3.x
    - GUI framework: Tkinter or PyQt
    - Libraries:
        - heapq for heap operations.
        - datetime for managing event schedules.
        - pickle or SQLite for data storage.

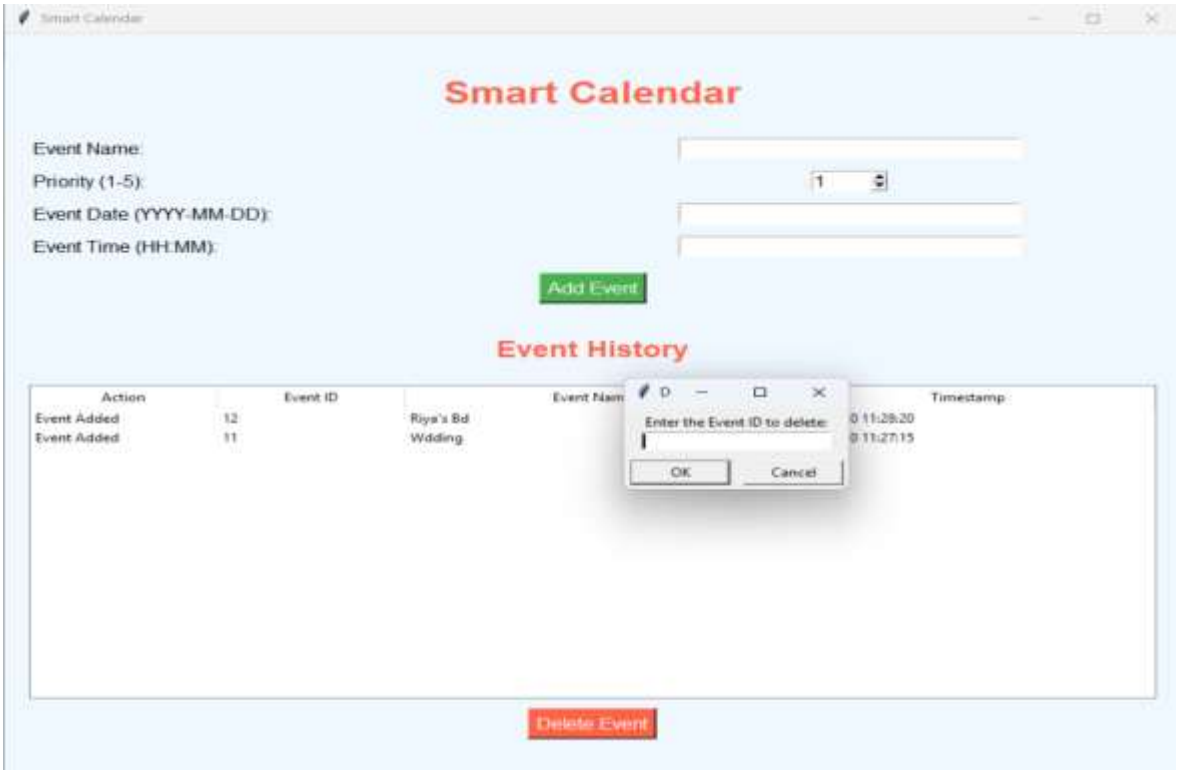
Technologies:

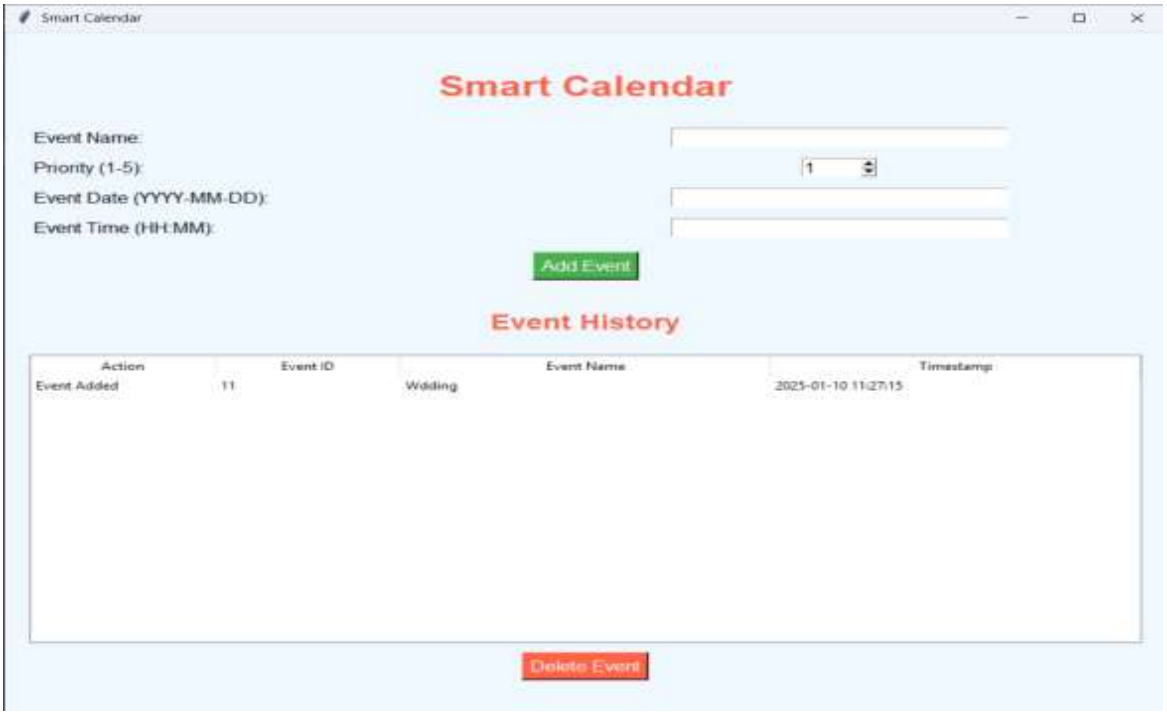
- **Programming Language**: Python
- **GUI Framework**:
    - **Tkinter** (for simplicity) or **PyQt** (for advanced UI features)
- **Data Structures**:
    - **Heap**: To prioritize events based on deadlines or importance.
    - **Linked List**: To maintain event reminders sequentially.
    - **Hash Table**: To store events for quick lookup and retrieval.
- **Database**: SQLite or file-based storage using Python's pickle.
- **Additional Libraries**:
    - time and datetime for scheduling and time calculations.
    - os for system notifications.

Screenshot:


Smart Calendar
Smart Calendar
Event Name:
Priority (1-5):
1
Event Date (YYYY-MM-DD):
Event Time (HH:MM):
Add Event
Event History
Action
Event ID
Event Name
Timestamp
Event Added
11
Wdding
2025-01-10 11:27:15
Delete Event


Smart Calendar
Smart Calendar
Event Name:
Priority (1-5):
1
Event Date (YYYY-MM-DD):
Event Time (HH:MM):
Add Event
Event History
Action
Event ID
Event Name
Timestamp
Event Added
11
Wdding
2025-01-10 11:27:15
Delete Event


Smart Calendar
Smart Calendar
Event Name:
Priority (1-5):
1
Event Date (YYYY-MM-DD):
Event Time (HH:MM):
Add Event
Event History
Action
Event ID
Event Nam
Timestamp
Event Added
12
Riya's Bd
0 11:28:20
Event Added
11
Wdding
0 11:27:15
D
Enter the Event ID to delete:
OK
Cancel
Delete Event


Smart Calendar
Smart Calendar
Event Name:
Priority (1-5):
1
Event Date (YYYY-MM-DD):
Event Time (HH:MM):
Add Event
Event History
Action
Event ID
Event Name
Timestamp
Event Added
11
Wdding
2025-01-10 11:27:15
Delete Event

Conclusion:

The Smart Calendar with Event Prioritization simplifies time management by prioritizing tasks and providing seamless event organization. With features like drag-and-drop scheduling, conflict detection, and notifications, the app ensures users stay on top of their schedules effortlessly. Utilizing robust data structures and a user-friendly GUI, the application delivers an efficient and responsive calendar solution. This project demonstrates the potential of combining advanced algorithms with intuitive design, paving the way for smarter personal organization tools.