**EVENTS MANGEMENT SYSEM**

**Group member:**

* Rao Huzaifa
* Muhammad Jamal
* Hassan Mehdi
* Muhammad Hasnain
* Muhammad Ahmad
* Ahmad Hassan

<https://github.com/group05E/EMS>

project name: Event -management-system

**Event Management System Report**

**Overview:**

The Event Management System (EMS) is a program designed to manage events of different categories such as conferences, workshops, and seminars. It allows users to add events with various details and display the events stored in the system.

**Features:**

1. Add Event: Users can add events by providing details such as event ID, name, category, venue, schedule date, time, and attendees.

2. Display Events: Users can view all the events stored in the system, along with their details including venue, schedule, and attendees.

3. Menu Interface: The program provides a menu-driven interface for users to interact with the system, offering options to add events, display events, and exit.

**Structure:**

Classes: The program is structured around several classes:

- `Attendee`: Represents an attendee with a name.

- `Venue`: Represents the venue of an event.

- `Schedule`: Represents the schedule (date and time) of an event.

- `Event`: Represents an event with attributes like ID, name, category, venue, schedule, and attendees.

- `Event-Management-System`: Manages events, allowing users to add events and display them.

Data Storage: Events are stored in an array within the `Event-Management-System` class.

**Functionality:**

Adding Events: Users can add events of different categories by providing relevant details through the console input.

Displaying Events: The program can display all the events stored in the system, including their details like venue, schedule, and attendees.

Input Handling: Input from users is handled using `cin` for standard input and `getline` for string input, with appropriate validation and error handling.

**Improvements:**

Input Validation: Implement input validation to ensure the correctness of user inputs and prevent errors.

Dynamic Memory Allocation: Consider using dynamic data structures like `std::vector` to allow for variable numbers of attendees and events.

Error Handling: Implement error handling and exception mechanisms to handle unexpected situations gracefully.

User Interface Enhancement: Improve user prompts and messages for better clarity and usability.

Code Documentation: Add comments to explain the purpose and functionality of classes, functions, and complex logic for easier understanding and maintenance.

**Conclusion:**

The Event Management System program provides basic functionality for managing events effectively. With further enhancements and refinements, it can be made more robust and user-friendly, catering to the needs of event organizers and participants.