**Semester Project Documentation**

**Semester Project Title**:  **Event Management System**

**Student Details**

(Fill according to number of team/group members)

|  |  |  |  |
| --- | --- | --- | --- |
|  | Student Name | Student Reg # | Student Degree |
| Student-1 | Muhammad Talal Munir | 2024477 | MGS |
| Student-2 | Zarmina Khan | 2024682 | MGS |
| Student-3 | Manahil Dawood Shamsi | 2024278 | MGS |
| Student-4 | Muhammad Abubakar | 2024333 | MGS |

**Main Features**

*(Mention each by numbering as 1, 2, 3 … etc)*

1. Event Creation
2. Event Detailing
3. Buying Tickets
4. Registration
5. General UI Menu
6. Preloading Events

**Types of Users**

*(Just Mention the types of users For Example: Teacher, Student … etc)*

1. Administrator
2. Event Manager
3. Ticket Buyer

**Requirements Breakdown**

*(Write requirements of each feature by numbering. For example: Requirements of feature # 1. It will be mentioned as 1.1, 1.2. Similarly for feature # 2 it will be 2.1, 2.2, 2.3 … etc.)*

1. **Event Creation**
   1. **Administrator/Manager Role:** An admin or the event manager can add or create new events.
   2. **Input Required:** The manager or admin needs to input date, time, location and ticket fee for the event to create it.
   3. **Space Availability:** The program also checks if there’s space for more events as the max space is 100 events to be.
   4. **Event Details Storage:** The information of every event is stored in a 2D array (events[max\_events][fields]).
2. **Event Detailing**
   1. **Event Information:** The program shows all the information of the event such as date, time and location.
   2. **Display Events:** The user can navigate the present events in the system through the menu.
   3. **Event Lookup:** System also checks if the event is available or passed by.
3. **Buying Tickets**
   1. **Ticket Purchase:** The user can buy minimum of 1 and maximum of 5 tickets in one transaction.
   2. **Discount Application:** A discount of 16% is given when the user buys 5 tickets.
   3. **Receipt Generation:** When the transaction is processed the user has to input age and name to see the receipt of the transaction.
   4. **Ticket Restriction:** The system restricts to buy 5 tickets at most.
4. **Registration**
   1. **User Registration:** Users can register for an event which gives them a unique account ID to enter the event.
   2. **Registration Fee:** The system collects the registration fee and generates a receipt when the registration is successful.
5. **General UI Menu**
   1. **Menu Options:** The main menu of program includes options to navigate such as Event Creation, Buy Tickets, View Events, and Exit.
   2. **Looped Interface:** The program keeps running until the user decides to exit the program.
   3. **User Input:** The program uses cin to input values and cout to print receipts and show details of events etc.
6. **Preloading Events**
   1. **Preloaded Events:** Preloaded events are created as a basic idea of the programs they are helpful when you’re creating a new event you can edit the preloaded ones instead of registering the entire program.

**Features to Coding Matrix**

*(In the following table you will mention the following items for each feature, mention the items in each column for each feature of your application)*

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Sr**  **no.** | **Feature Name** | **Concept Used** | **Functions Created** | **Variables / Obj Created** | **Line of Code Written** |
| **1** | Event Creation | Conditional Logic, Data storage | create\_event() | events[max\_events][fields], current\_events, event\_details[] | 70 |
| **2** | Event Detailing | Data Retrieval, Looping | viewevents() | event\_id[], event\_info[] | 60 |
| **3** | Buying Tickets | Conditional Logic, Loops, Arithmetic | buy\_tickets() | ticketCount, purchaserName, purchaserAge, ticketPrice, tax, totalCost, netTotal | 110 |
| **4** | |  | | --- | |  |  |  | | --- | | Registering for Events | | Data Retrieval, User input, Output | Register\_for\_events() | eventID, DelegateName, DelegateAge, regPrice | 80 |
| **5** | General UI Menu | Switch-Case, Looping | Menu() | choice, user\_action | 100 |
| **6** | Preloading Events | Data Initialization | preloadEvents() | events[max\_events][fields] | 50 |