Figure 1: UET

UNIVERSITY OF ENGINEERING AND TECHNOLOGY LAHORE

FAISLABAD CAMPUS

Submitted To: Engr.Syed Muhammad Umer

Group 6

Submitted By: Abdullah Younas(2025-MC-165)

Hashir Ul Islam(2025-MC-181)

Fatimah Ahmed(2025-MC-205)

Umaiza Waseem Kiani(2025-MC-207)

Semester:1st

Session: Fall 2025

Course:AICT (MCT-107L)

DEPARTMENT OF MECHATRONICS,MECHANICAL AND MANUFACTURING ENGINEERING

0.1 INTRODUCTION

Event management is a key application of ICT, enabling efficient organization of events, participant tracking, and registration management. Manual methods often cause errors and duplicates. This project designs a small-scale event management system with 10 records, including a DBMS schema, ER diagram, and sample SQL queries. ICT tools like Google Sheets,Overleaf, GitHub, Canva, and Mendeley were used to complete the project.

0.2 PROBLEM DEFINITION

Manual event management and registration is time-consuming and error-prone. There is a need for an online system that allows easy event management and quick, accurate participant registration while reducing paperwork and effort

0.3 LITERATURE REVIEW

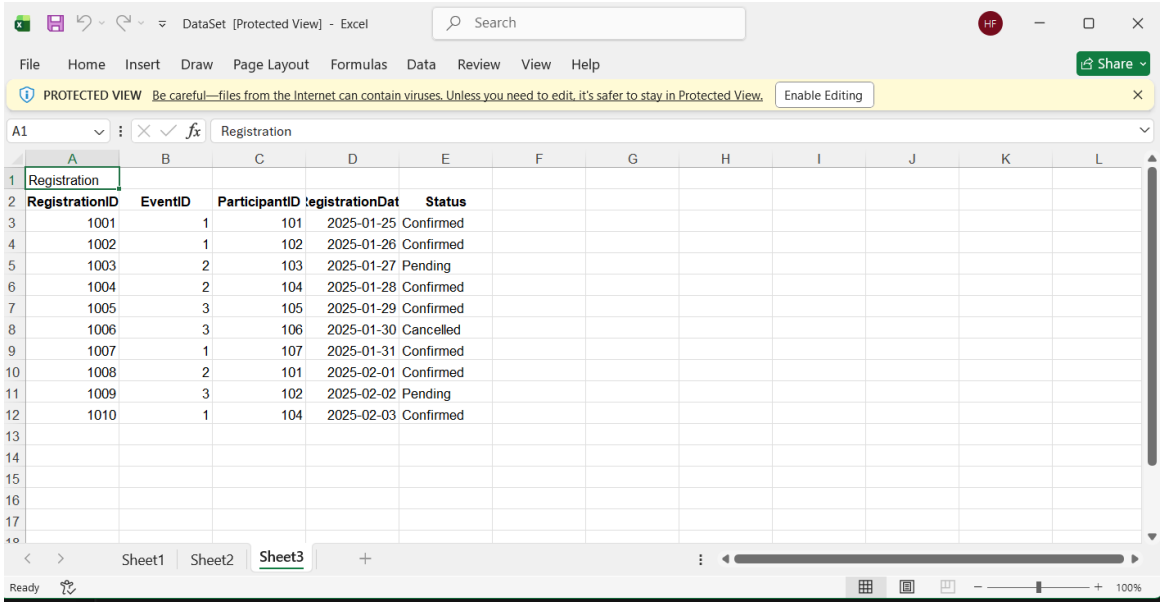
Event management systems are widely used to organize events, manage participant information, and handle registrations efficiently. Traditional manual registration methods often lead to data redundancy, errors, and difficulties in tracking participants, especially as the number of events and attendees increases [1]. Database Management Systems (DBMS) play a crucial role in modern event management applications by providing structured data storage, integrity constraints, and efficient data retrieval using SQL queries [2]. Relational databases help in maintaining relationships between events, participants, and registrations through primary and foreign keys. Recent studies also highlight the use of spreadsheet-based tools such as Google Sheets for dataset creation, basic analysis, and simulation before full system implementation [3]. These tools, combined with proper system design and documentation, improve the overall effectiveness and reliability of ICT-based management systems.

0.4 PROPOSE SYSTEM

The proposed system is a database-based Event Management and Registration system using three tables: Events, Participants, and Registrations. Primary and foreign keys ensure data integrity. Google Sheets manages the dataset, SQL queries handle

data retrieval and reporting, and Canva visualizes the system structure. The system provides a simple and reliable solution for managing event registrations digitally.

0.5 Dataset and Analysis



DataSet [Protected View] - Excel

Search

File Home Insert Draw Page Layout Formulas Data Review View Help

Share

PROTECTED VIEW Be careful—files from the Internet can contain viruses. Unless you need to edit, it's safer to stay in Protected View. Enable Editing

A1 Registration

	A	B	C	D	E
1	Registration				
2	RegistrationID	EventID	ParticipantID	egistrationDat	Status
3	1001	1	101	2025-01-25	Confirmed
4	1002	1	102	2025-01-26	Confirmed
5	1003	2	103	2025-01-27	Pending
6	1004	2	104	2025-01-28	Confirmed
7	1005	3	105	2025-01-29	Confirmed
8	1006	3	106	2025-01-30	Cancelled
9	1007	1	107	2025-01-31	Confirmed
10	1008	2	101	2025-02-01	Confirmed
11	1009	3	102	2025-02-02	Pending
12	1010	1	104	2025-02-03	Confirmed

Sheet1 Sheet2 Sheet3

Ready 100%

Figure 2: DATASET

Analysis: The dataset for the Event Management and Registration system was created using Google Sheets with three sheets: Events, Participants, and Registrations. It contains 10 registration records to simulate real scenarios. Basic analysis, such as counting registrations per event and listing participants for specific events, was performed and verified using SQL queries. This confirms the correctness of the database design and demonstrates efficient use of ICT tools for data management and reporting.

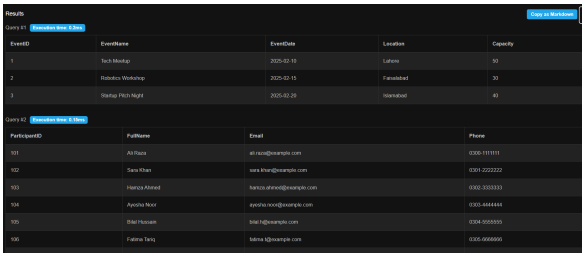
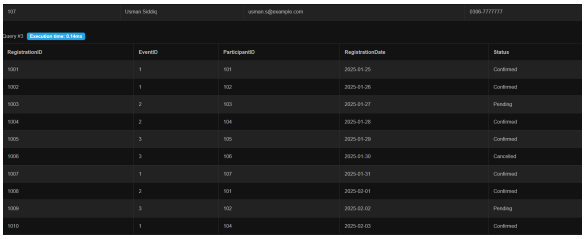
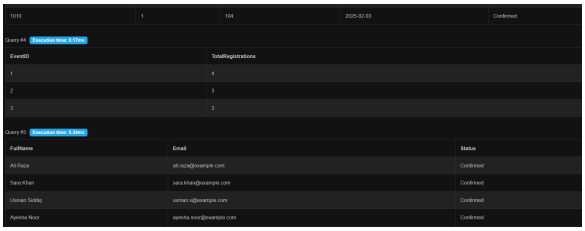


Figure 3: Result 1



RegistrationID	EventID	ParticipantID	RegistrationDate	Status
1001	1	101	2025-01-25	Confirmed
1002	1	102	2025-01-26	Confirmed
1003	2	103	2025-01-27	Pending
1004	2	104	2025-01-28	Confirmed
1005	3	105	2025-01-29	Confirmed
1006	3	106	2025-01-30	Cancelled
1007	1	107	2025-01-31	Confirmed
1008	2	101	2025-02-01	Confirmed
1009	3	102	2025-02-02	Pending
1010	1	104	2025-02-03	Confirmed

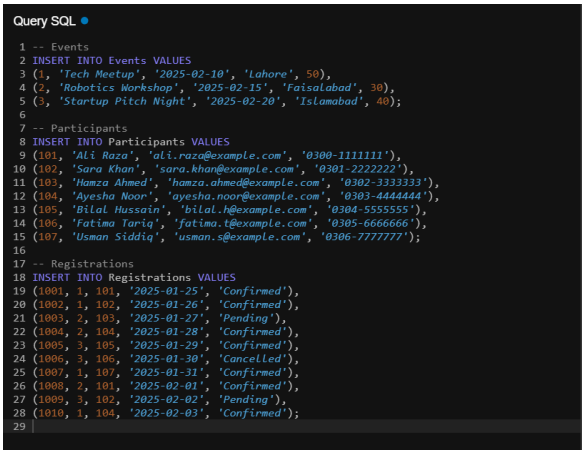
Figure 4: Result 2



EventID	TotalRegistrations
1	4
2	3
3	3

FullName	Email	Status
Ali Raza	ali.raza@example.com	Confirmed
Sara Khan	sara.khan@example.com	Confirmed
Usman Siddiq	usman.s@example.com	Confirmed
Ayesha Noor	ayesha.noor@example.com	Confirmed

Figure 5: Result 3



Query SQL

```sql
-- Events
INSERT INTO Events VALUES
(1, 'Tech Meetup', '2025-02-10', 'Lahore', 50),
(2, 'Robotics Workshop', '2025-02-15', 'Faisalabad', 30),
(3, 'Startup Pitch Night', '2025-02-20', 'Islamabad', 40);

-- Participants
INSERT INTO Participants VALUES
(101, 'Ali Raza', 'ali.raza@example.com', '0300-1111111'),
(102, 'Sara Khan', 'sara.khan@example.com', '0301-2222222'),
(103, 'Hamza Ahmed', 'hamza.ahmed@example.com', '0302-3333333'),
(104, 'Ayesha Noor', 'ayesha.noor@example.com', '0303-4444444'),
(105, 'Bilal Hussain', 'bilal.h@example.com', '0304-5555555'),
(106, 'Fatima Tariq', 'fatima.t@example.com', '0305-6666666'),
(107, 'Usman Siddiq', 'usman.s@example.com', '0306-7777777');

-- Registrations
INSERT INTO Registrations VALUES
(1001, 1, 101, '2025-01-25', 'Confirmed'),
(1002, 1, 102, '2025-01-26', 'Confirmed'),
(1003, 2, 103, '2025-01-27', 'Pending'),
(1004, 2, 104, '2025-01-28', 'Confirmed'),
(1005, 3, 105, '2025-01-29', 'Confirmed'),
(1006, 3, 106, '2025-01-30', 'Cancelled'),
(1007, 1, 107, '2025-01-31', 'Confirmed'),
(1008, 2, 101, '2025-02-01', 'Confirmed'),
(1009, 3, 102, '2025-02-02', 'Pending'),
(1010, 1, 104, '2025-02-03', 'Confirmed');
```

Figure 6: SQL queries

```sql
CREATE TABLE Events (
  EventID INT PRIMARY KEY,
  EventName VARCHAR(100) NOT NULL,
  EventDate DATE NOT NULL,
  Location VARCHAR(100) NOT NULL,
  Capacity INT NOT NULL
);

CREATE TABLE Participants (
  ParticipantID INT PRIMARY KEY,
  FullName VARCHAR(100) NOT NULL,
  Email VARCHAR(100) UNIQUE NOT NULL,
  Phone VARCHAR(20)
);

CREATE TABLE Registrations (
  RegistrationID INT PRIMARY KEY,
  EventID INT NOT NULL,
  ParticipantID INT NOT NULL,
  RegistrationDate DATE NOT NULL,
  Status VARCHAR(20),
  FOREIGN KEY (EventID) REFERENCES Events(EventID),
  FOREIGN KEY (ParticipantID) REFERENCES Participants(ParticipantID)
);

-- Events
INSERT INTO Events VALUES
(1, 'Tech Meetup', '2025-02-10', 'Lahore', 50),
(2, 'Robotics Workshop', '2025-02-15', 'Faisalabad', 30),
(3, 'Startup Pitch Night', '2025-02-20', 'Islamabad', 40);

-- Participants
INSERT INTO Participants VALUES
(101, 'Ali Raza', 'ali.raza@example.com', '0300-1111111'),
(102, 'Sara Khan', 'sara.khan@example.com', '0301-2222222'),
(103, 'Hamza Ahmed', 'hamza.ahmed@example.com', '0302-3333333'),
```

Figure 7: SQL Queries

```sql
(103, 'Hamza Ahmed', 'hamza.ahmed@example.com', '0302-3333333'),
(104, 'Ayesha Noor', 'ayesha.noor@example.com', '0303-4444444'),
(105, 'Bilal Hussain', 'bilal.h@example.com', '0304-5555555'),
(106, 'Fatima Tariq', 'fatima.t@example.com', '0305-6666666'),
(107, 'Usman Siddiq', 'usman.s@example.com', '0306-7777777');

-- Registrations
INSERT INTO Registrations VALUES
(1001, 1, 101, '2025-01-25', 'Confirmed'),
(1002, 1, 102, '2025-01-26', 'Confirmed'),
(1003, 2, 103, '2025-01-27', 'Pending'),
(1004, 2, 104, '2025-01-28', 'Confirmed'),
(1005, 3, 105, '2025-01-29', 'Confirmed'),
(1006, 3, 106, '2025-01-30', 'Cancelled'),
(1007, 1, 107, '2025-01-31', 'Confirmed'),
(1008, 2, 101, '2025-02-01', 'Confirmed'),
(1009, 3, 102, '2025-02-02', 'Pending'),
(1010, 1, 104, '2025-02-03', 'Confirmed');
```

Figure 8: SQL Queries

0.6 DBMS Schema

The system uses a relational database with three tables: *Events*, *Participants*, and *Registrations*. Primary keys ensure unique records, and foreign keys maintain relationships between events and participants. SQL queries are used to retrieve, insert, and analyze data efficiently, supporting event management and registration processes.

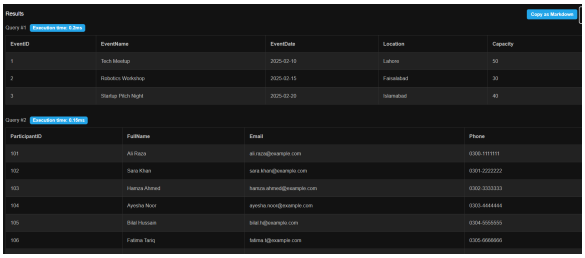


Figure 9: Result 1

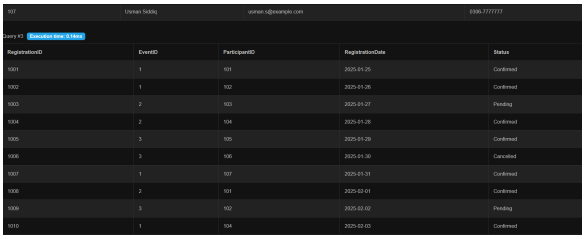


Figure 10: Result 2

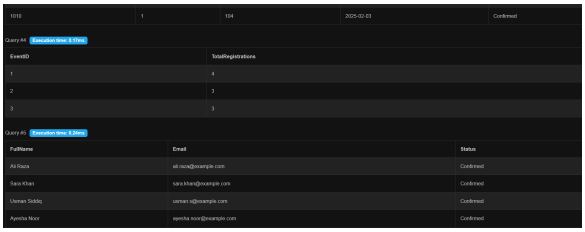


Figure 11: Result 3

0.7 System Diagram

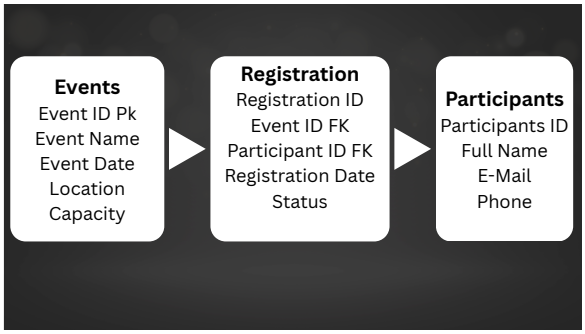


Figure 12: Enter Caption

0.8 Results

The system was tested using 10 records, and SQL queries were executed to retrieve event details, participant lists, and registration counts.

Key results:

- Total registrations per event were counted correctly.
- Participants of EventID = 1 were listed with their status.
- All queries returned accurate and consistent results.

These results confirm that the system effectively manages events and registrations.

0.9 Conclusion

The Event Management and Registration system provides a simple and efficient way to manage events, participants, and registrations using a relational database. The system ensures data accuracy, maintains relationships through primary and foreign keys, and allows easy retrieval and analysis using SQL queries. Integration with Google Sheets and visualization tools like Canva further supports data management and reporting. Overall, the system demonstrates an effective ICT-based solution for organizing and tracking events.

Bibliography

[1] K. C. Laudon and J. P. Laudon, *Management Information Systems*. Pearson, 2020.

[2] A. Silberschatz, H. F. Korth, and S. Sudarshan, *Database System Concepts*. McGraw-Hill, 2019.

[3] S. G. Powell and K. R. Baker, *Management Science: The Art of Modeling with Spreadsheets*. Wiley, 2018.