



CSC104-Database Systems Mid Project Report (Spring 2025)



Student Details

Repository Name	DBS25P144
Registration Number	2024-CS-144

Project	
Project Title	Project A: ITEC Event Management System
Executive Summary	<p>The Information Technology Exhibition and Competition (ITEC) is an annual event organized by the Department of Computer Science, UET Lahore. It includes competitions, exhibitions, seminars, and non-tech events, managed by faculty and students through different committees. Currently, event management is done manually using spreadsheets, emails, and physical records, which leads to data errors, scheduling conflicts, financial tracking issues, and difficulty in managing committees. To solve these problems, a Windows Form Application using C# and MySQL are made to solve and managed these event management process. The main features are Manage ITEC Editions – Create and handle different editions like ITEC 2024,2023etc that are linking events, participants, and finances. Event Organization – Add, update, and delete or remove competitions, exhibitions, and seminars while assigning committees. Participant Registration & Fee Tracking – Register students, professionals, guest speakers, and judges, track event participation, and manage payments. Committee & Role Assignment – Assign faculty and students to different committees with different roles and responsibilities. Task Assignment & Tracking – Assign the duties for committee members. Financial Management – Record sponsorships, ticket sales, and event expenses while generating financial reports. Venue Scheduling – Allocate venues in event schedules. Event Results & Ranking – Store competition results. Automated Reports – Generate detailed reports on participants, committees, finances, and event schedules. This system will reduce manual effort, improve accuracy, and make event management easier and more efficient, ensuring a well-organized and successful ITEC event each year.</p>

ITEC	
------	--

<div> <div>ITEC</div> <div> <div>DASHBOARD</div> <div>EVENTS</div> <div>FINANCES</div> <div>COMMITTEES</div> <div>PARTICIPANT REGISTRATION</div> <div>DUTIES</div> <div>VENUES</div> <div>RESULTS</div> <div>REPORTS</div> <div>USER LOGIN</div> <div>EDITIONS</div> </div> <div> <div>ITEC EVENTS</div> <div> <div>EVENT MANAGERMENTS</div> <div>Participant Registration And Fee Tracking</div> <div>Committee And Role Management</div> <div>DUTIES ASSIGNMENT AND TRACKING</div> <div>VENUE ALLOCATIONS</div> <div>FINANCIAL MANAGERMENTS AND SPONSERSHIP</div> </div> </div> </div>			
UI Component Name	Type of UI component	Class Attribute name, Classname, attribute name	Linked to which database column/table
DASHBOARD	Button	button1, ITEC linked with itself	None
EVENTS	Button	button2, ITEC linked with Events	itec_events
FINANCES	Button	button3, ITEC linked with Finances	finances
COMMITTEES	Button	button4, ITEC linked with Committee	committees
PARTICIPANT REGISTRATION	Button	button5, ITEC linked with Participants	participants, event_participants
DUTIES	Button	button6, ITEC linked with Duty	duties
VENUES	Button	button7, ITEC linked with Venue	venues
RESULTS	Button	button8, ITEC linked with Result	event_results
REPORTS	Button	button9, ITEC linked with Report	All tables
USER LOGIN	Button	button10, ITEC linked with User	users
EDITIONS	Button	Button11, ITEC linked with ITEC editions	itec_editions

ITEC EDITIONS	
---------------	--

ITEC Editions

BACK

ITEC EDITIONS MANAGEMENT

SELECT EDITION

EDITION YEAR

THEME

DESCRIPTION

ADD

UPDATE

DELETE

UI Component Name	Type of UI component	Class Attribute name, Classname, attribute name	Linked to which database column/table
SELECT EDITION	ComboBox	combobox1, ITEC Editions	itec_id, year
EDITION YEAR	TextBox	textbox1, ITEC Editions	year
THEME	TextBox	textbox2, ITEC Editions	theme
DESCRIPTION	TextBox	textbox3, ITEC Editions	description
ADD	Button	button1, ITEC Editions	INSERT into itec_editions
UPDATE	Button	button2, ITEC Editions	DELETE from itec_editions
BACK	Button	button4, ITEC Editions	(Navigation only)
Grid view	DataGridView	Datagridview1, ITEC Editions	itec_editions (all columns)

```

INSERT INTO itec_editions (year, theme, description) VALUES (@year, @theme, @desc)
UPDATE itec_editions SET year = @year, theme = @theme, description = @desc WHERE itec_id = @id
DELETE FROM itec_editions WHERE itec_id = @id
SELECT itec_id, year FROM itec_editions
SELECT * FROM itec_editions ORDER BY year DESC

```

EVENTS	

UI Component Name	Type of UI component	Class Attribute name, Classname, attribute name	Linked to which database column/table
EVENT NAME	TextBox	textbox1, Events	event_name
EVENT CATEGORY	ComboBox	combobox1, Events	itec_events.event_category_id → FK to event_categories.event_category_id
DESCRIPTION	TextBox	textbox2, Events	description
EVENT DATE AND TIME	DateTimePicker	datetimepicker1, Events	event_date
VENUE	ComboBox	combobox2, Events	itec_events.venue_id → FK to venues.venue_id
ORGANIZING COMMITTEE	ComboBox	combobox3, Events	itec_events.committee_id → FK to committees.committee_id
ADD	Button	button1, Events	INSERT into itec_events
UPDATE	Button	button2, Events	UPDATE on itec_events
DELETE	Button	button3, Events	DELETE from itec_events
BACK	Button	button4, Events	Navigation only
Grid View	DataGridView	dataGridView1, Events	Shows itec_events joined with categories, venues, committee

```

SELECT * FROM itec_events
SELECT event_category_id, category_name FROM event_categories
SELECT venue_id, venue_name FROM venues
SELECT committee_id, committee_name FROM committees
SELECT
    e.event_id,
    e.event_name,
    ec.category_name AS category,
    e.description,

```

```

e.event_date,
v.venue_name AS venue,
c.committee_name AS committee,
e.event_category_id, e.venue_id, e.committee_id
FROM itec_events e
JOIN event_categories ec ON e.event_category_id = ec.event_category_id
JOIN venues v ON e.venue_id = v.venue_id
JOIN committees c ON e.committee_id = c.committee_id
INSERT INTO itec_events (event_name, event_category_id, description, event_date, venue_id, committee_id)
VALUES (@name, @category, @desc, @date, @venue, @committee)
UPDATE itec_events SET
    event_name = @name,
    event_category_id = @category,
    description = @desc,
    event_date = @date,
    venue_id = @venue,
    committee_id = @committee
WHERE event_id = @id
DELETE FROM itec_events WHERE event_id = @id

```

Participants

BACK **PARTICIPANT REGISTRATION FORM**

PARTICIPANT NAME

EMAIL

CONTACT NUMBER

INSTITUTE NAME

ROLE

EVENTS TO PARTICIPATE IN ☐ Coding

PAYMENT STATUS

ADD **UPDATE** **DELETE**

UI Component Name	Type of UI component	Class Attribute name, Classname, attribute name	Linked to which database column/table
PARTICIPANT NAME	TextBox	Participants, textBox1	name

EMAIL	TextBox	Participants, textBox2	email
CONTACT NUMBER	TextBox	Participants, textBox3	contact
INSTITUTE NAME	TextBox	Participants, textBox4	institute
ROLE	ComboBox	Participants, checkedListBox1	event_participants.event_id
EVENTS TO PARTICIPATE IN	CheckedListBox	Participants, checkedListBox1	event_participants.event_id
PAYMENT STATUS	ComboBox	Participants, comboBox2	event_participants.payment_status_id → FK to lookup
ADD	Button	Participants, button1	Inserts into participants and event_participants
UPDATE	Button	Participants, button2	Updates participants table based on name
DELETE	Button	Participants, button3	Deletes from participants table based on name
BACK	Button	Participants, button4	Navigation (back to Form1)
Grid View	DataGridView	Participants, dataGridView1	Displays participants table

```

INSERT INTO participants (itec_id, name, email, contact, institute, role_id) " +
        "VALUES (@itec_id, @name, @email, @contact, @institute, @role_id)"
INSERT INTO event_participants (event_id, participant_id, payment_status_id, fee_amount) " +
        "VALUES (@event_id, @participant_id, @payment_status_id,
@fee_amount)"
"UPDATE participants SET email=@email, contact=@contact, institute=@institute, role_id=@role_id " +
        "WHERE name=@name"
DELETE FROM participants WHERE name=@name
SELECT * FROM participants
SELECT value FROM lookup WHERE category = 'role'
SELECT value FROM lookup WHERE category = 'payment_status'
SELECT event_name FROM itec_events
SELECT lookup_id FROM lookup WHERE category = 'role' AND value = @value
SELECT lookup_id FROM lookup WHERE category = 'payment_status' AND value = @value
SELECT event_id FROM itec_events WHERE event_name = @value

```

Committee

Committee

BACK

COMMITTEE MANAGEMENT FORM

COMMITTEE NAME

MEMBERS (FACULTY/STUDENTS)

ROLES

ADD

UPDATE

DELETE

ASSIGNED TO

COMMITTEE LIST

COMMITTEE MEMBERS LIST

UI Component Name	Type of UI component	Class Attribute name, Classname, attribute name	Linked to which database column/table
COMMITTEE NAME	TextBox	Committee, textBox1	committee_name
MEMBERS (FACULTY/STUDENTS)	ComboBox	Committee, comboBox1	Used for selecting existing committee to which members will be assigned (binds to committee_id)
ROLES	ComboBox	Committee, comboBox2	committee_members.role_id → FK to lookup (where category = 'Role')
COMMITTEE LIST	DataGridView	Committee, dataGridView1	Displays committees table
COMMITTEE MEMBERS LIST	DataGridView	Committee, dataGridView2	Displays committee_members with name and joined role from lookup
ADD	Button	Committee, button1	Adds new record to committees table
UPDATE	Button	Committee, button2	Updates committees.committee_name based on selected row from dataGridView1
DELETE	Button	Committee, button3	Deletes record from committees (code mistakenly refers to ID instead of committee_id)
ASSIGNED TO	Button	Committee, button4	Adds member (name, role_id) to committee_members for selected committee
BACK	Button	Committee, button5	Returns to Form1

SELECT committee_id, committee_name FROM Committees
SELECT cm.member_id, cm.name, l.value AS role
FROM CommitteeMembers cm

```
JOIN lookup_l ON cm.role_id = l.lookup_id
WHERE cm.committee_id = " + committeeId
```

```
SELECT lookup_id, value FROM lookup WHERE category = 'Role'
```

```
INSERT INTO Committees (committee_name) VALUES ('{name}')
```

```
UPDATE Committees SET committee_name = '{name}' WHERE committee_id = {id}
```

```
INSERT INTO CommitteeMembers (committee_id, name, role_id) VALUES (@committee_id, @name, @role_id)
```

Duty

UI Component Name	Type of UI component	Class Attribute name, Classname, attribute name	Linked to which database column/table
COMMITTEE	ComboBox	Duty, comboBox1	duties.committee_id → FK to committees.committee_id
ASSIGNED TO	ComboBox	Duty, comboBox2	duties.assigned_to
TASK DESCRIPTION	TextBox	Duty, textBox1	duties.task_description
DEADLINE	DateTimePicker	Duty, dateTimePicker1	duties.deadline
STATUS	ComboBox	Duty, comboBox3	duties.status_id → FK to lookup.lookup_id where category = 'task_status'
TASK LIST GRID	DataGridView	Duty, dataGridView1	Displays all rows from duties table
SAVE TASK	Button	Duty, button1	Executes INSERT INTO duties
MODIFY TASK	Button	Duty, button2	Executes UPDATE duties
REMOVE TASK	Button	Duty, button3	Executes DELETE FROM duties
BACK	Button	Duty, button4	Navigates to Form1

```
SELECT * FROM duties
```



```

SELECT committee_id, committee_name FROM committees
SELECT lookup_id, value FROM lookup WHERE category = 'task_status'
INSERT INTO duties (committee_id, assigned_to, task_description, deadline, status_id) " +
    "VALUES (@committee_id, @assignedTo, @taskDescription, @deadline, @status_id)"
UPDATE duties SET committee_id = @committee_id, assigned_to = @assignedTo, " +
    "task_description = @taskDescription, deadline = @deadline, status_id = @status_id " +
    "WHERE duty_id = @dutyId"
DELETE FROM duties WHERE duty_id = @taskId

```

Finances

UI Component Name	Type of UI component	Class Attribute name, Classname, attribute name	Linked to which database column/table
TRANSACTION TYPE	ComboBox	comboBox1, Finances	type_id, look_up table and finances table
AMOUNT	TextBox	textBox1, Finances	amount, finances table
FROM ENTITY	ComboBox	comboBox2, Finances	from_entity_id ,sponsors table
TO ENTITY	ComboBox	comboBox3, Finances	to_entity_id, committees
DESCRIPTION	TextBox	textBox2, Finances	description, finances table
DATE RECORDED	DateTimePicker	dateTimePicker1, Finances	date_recorded, finances table
SAVE FINANCIAL RECORD	Button	button1, Finances	INSERT operation, finances
MODIFY EXISTING RECORDS	Button	button2, Finances	UPDATE operation, finances

DELETE TRANSACTION	Button	button3, Finances	DELETE operation, finances
BACK	Button	button4, Finances	navigation
Data Grid View	DataGridView	dataGridView1, Finances	finances (joined with lookup, sponsors, committees, vendors)

SELECT

f.transaction_id,
l.value AS transaction_type,
f.amount,
s1.sponsor_name AS from_sponsor,
v1.vendor_name AS from_vendor,
c1.committee_name AS from_committee,
s2.sponsor_name AS to_sponsor,
v2.vendor_name AS to_vendor,
c2.committee_name AS to_committee,
f.description,
f.date_recorded

FROM finances f

LEFT JOIN lookup l ON f.type_id = l.type_id

LEFT JOIN sponsors s1 ON f.from_entity_type = 'Sponsor' AND f.from_entity_id = s1.sponsor_id

LEFT JOIN vendors v1 ON f.from_entity_type = 'Vendor' AND f.from_entity_id = v1.vendor_id

LEFT JOIN committees c1 ON f.from_entity_type = 'Committee' AND f.from_entity_id = c1.committee_id

LEFT JOIN sponsors s2 ON f.to_entity_type = 'Sponsor' AND f.to_entity_id = s2.sponsor_id

LEFT JOIN vendors v2 ON f.to_entity_type = 'Vendor' AND f.to_entity_id = v2.vendor_id

LEFT JOIN committees c2 ON f.to_entity_type = 'Committee' AND f.to_entity_id = c2.committee_id;

SELECT lookup_id AS type_id, value FROM lookup WHERE category = 'TransactionType'

SELECT sponsor_id, sponsor_name FROM sponsors

SELECT committee_id, committee_name FROM committees

INSERT INTO finances

(itec_id, event_id, type_id, amount, from_entity_type, from_entity_id, to_entity_type, to_entity_id, description, date_recorded)

VALUES (1, 1, @type_id, @amount, @from_entity_type, @from_entity_id, @to_entity_type, @to_entity_id, @description, @date_recorded);

UPDATE finances SET type_id = @type_id, amount = @amount, from_entity_type = @from_entity_type, from_entity_id = @from_entity_id, " +

"to_entity_type = @to_entity_type, to_entity_id = @to_entity_id, description = @description, date_recorded = @date_recorded " +

"WHERE transaction_id = @transaction_id;

DELETE FROM finances WHERE transaction_id = @transaction_id

Venue

UI Component Name	Type of UI component	Class Attribute name, Classname, attribute name	Linked to which database column/table
Event	ComboBox	comboBox1, Venue	event_id, event_name from itec_events
Venue	ComboBox	comboBox2, Venue	venue_id, venue_name from venues
Assigned Date	DateTimePicker	dateTimePicker1, Venue	assigned_date from venue_allocations
Assigned Time	DateTimePicker	dateTimePicker2, Venue	assigned_time from venue_allocations
Validate Scheduling Conflicts	Button	button1, Venue	Validates using query on venue_allocations
Save Venue Assignment	Button	button2, Venue	Inserts data into venue_allocations
Modify Existing Allocation	Button	button3, Venue	Updates record in venue_allocations
Delete Venue Assignment	Button	button4, Venue	Deletes record from venue_allocations
Back	Button	button5, Venue	Back to Form1
DataGrid	DataGridView	dataGridView1, Venue	Display data

SELECT event_id, event_name FROM itec_events;

SELECT venue_id, venue_name FROM venues;

SELECT va.venue_allocation_id, ev.event_name, v.venue_name, va.assigned_date, va.assigned_time
FROM venue_allocations va

JOIN itec_events ev ON va.event_id = ev.event_id

JOIN venues v ON va.venue_id = v.venue_id;

SELECT COUNT(*) FROM venue_allocations WHERE venue_id={venueId} AND assigned_date='{date}' AND
assigned_time='{time}';

INSERT INTO venue_allocations (event_id, venue_id, assigned_date, assigned_time) VALUES ({eventId},

```
{venueId}, '{date}', '{time}');
```

```
UPDATE venue_allocations SET event_id={eventId}, venue_id={venueId}, assigned_date='{date}',  
assigned_time='{time}' WHERE venue_allocation_id={id};
```

```
DELETE FROM venue_allocations WHERE venue_allocation_id={id};
```

Result

UI Component Name	Type of UI component	Class Attribute name, Classname, attribute name	Linked to which database column/table
EVENT	ComboBox	comboBox1, Result	event_id, event_results → itec_events
PARTICIPANT/TEAM	ComboBox	comboBox2, Result	team_id, event_results → teams
POSITION	ComboBox	comboBox3, Result	position, event_results
SCORE	TextBox	textBox1, Result	Score, event_results
REMARKS	TextBox	textBox2, Result	Remarks, event_results
SAVE RESULT	Button	button1, Result	(Triggers insert) event_results
MODIFY EXISTING RESULT	Button	button2, Result	(Triggers update) event_results
REMOVE A RESULT	Button	button3, Result	(Triggers delete) event_results
CALCULATE AND DISPLAY RANKING	Button	button4, Result	(Triggers data reload) event_results
BACK	Button	button5, Result	navigation
Data Grid	DataGridView	dataGridView1, Result	Shows results from all columns event_results + itec_events + teams

```
SELECT event_id, event_name FROM itec_events;
```

```
SELECT team_id, team_name FROM teams;
```

```

SELECT
    er.result_id,
    ev.event_id,
    ev.event_name,
    t.team_id,
    t.team_name,
    er.position,
    er.score,
    er.remarks
FROM event_results er
JOIN itec_events ev ON er.event_id = ev.event_id
JOIN teams t ON er.team_id = t.team_id";
INSERT INTO event_results
    (event_id, team_id, position, score, remarks)
VALUES (@event_id, @team_id, @position, @score, @remarks);
UPDATE event_results
    SET event_id=@event_id, team_id=@team_id,
        position=@position, score=@score, remarks=@remarks
    WHERE result_id=@result_id;
DELETE FROM event_results WHERE result_id=@result_id;

```

Report

Report

BACK

AUTOMATED REPORTS FORM

REPORT TYPE

FILTERS

EVENT(IF APPLICABLE)

DATE RANGE

COMMITTEE(IF APPLICABLE)

DISPLAY REPORT **EXPORT AS PDF** **EXPORT AS EXCEL**

UI Component Name	Type of UI component	Class Attribute name, Classname, attribute name	Linked to which database column/table
Report Type	ComboBox	comboBox1, Report	not linked to DB
Event	ComboBox	comboBox2, Report	itec_events.event_name
Date Range (From)	DateTimePicker	dateTimePicker1, Report	event_date (for event reports),

			date_recorded (for finances), etc.
Date Range (To)	DateTimePicker	dateTimePicker2, Report	event_date (for event reports), date_recorded (for finances), etc.
Committee	ComboBox	comboBox3, Report	committees.committee_name
Display Report	Button	button1, Report	Triggers report generation from various tables
Export as PDF	Button	button2, Report	Save as PDF
Export as Excel	Button	button3, Report	Save as excel
Back	Button	button4, Report	Navigates to Form1
Data Grid	DataGridView	dataGridView1, Report	Displays results

SELECT DISTINCT event_name FROM itec_events;

SELECT p.name, p.email, p.institute, ep.event_id, e.event_name " +
"FROM participants p " +
"JOIN event_participants ep ON p.participant_id = ep.participant_id " +
"JOIN itec_events e ON ep.event_id = e.event_id " +
"WHERE e.event_date BETWEEN @start AND @end;

SELECT f.transaction_id, f.amount, f.type_id, f.description, f.date_recorded, l.value AS type " +
"FROM finances f " +
"JOIN lookup l ON f.type_id = l.lookup_id " +
"WHERE f.date_recorded BETWEEN @start AND @end";

SELECT c.committee_name, p.name AS member_name, l.value AS role " +
"FROM committees c " +
"JOIN committee_members cm ON c.committee_id = cm.committee_id " +
"JOIN participants p ON cm.participant_id = p.participant_id " +
"JOIN lookup l ON cm.role_id = l.lookup_id " +
"WHERE l=1";

SELECT event_name, event_date, location, description " +
"FROM itec_events " +
"WHERE event_date BETWEEN @start AND @end;

User

User

BACK

USER MANAGEMENT FORM

USERNAME

EMAIL

PASSWORD

ROLE

SAVE NEW USER

MODIFY EXISITING USER

REMOVE USER

UI Component Name	Type of UI component	Class Attribute name, Classname, attribute name	Linked to which database column/table
Username	TextBox	textBox1, User	users.username
Email	TextBox	textBox2, User	users.email
Password	TextBox	textBox3, User	users.password_hash
Role	ComboBox	comboBox1, User	users.role_id → roles.role_id
Save New User	Button	button1, User	Triggers the INSERT users
Modify Existing User	Button	button2, User	Triggers the UPDATE users
Remove User	Button	button3, User	Triggers the DELETE users
Back	Button	button4, User	Navigates back to Form1
Grid View	DataGridView	dataGridView, User 1	Displays: user_id, username, email, role_id from users

```

SELECT role_id, role_name FROM roles;
SELECT user_id, username, email, role_id FROM users;
INSERT INTO users (username, email, password_hash, role_id) VALUES (@username, @email, @password, @role_id);
UPDATE users SET email = @email, password_hash = @password, role_id = @role_id WHERE username = @username;
DELETE FROM users WHERE username = @username;

```