Key Components of the Project

1. **Participant Registration:**
   - Collect participant details like name, contact information, and payment method.
2. **Financial Tracking:**
   - Track the event's **arrangement costs** (venue, food, decoration, etc.).
   - Track **income** from registrations and payments.
   - Track the **expenses** related to organizing the event.
3. **Payment Tracking:**
   - Handle payments through **b-Kash** and **Nagad**.
   - Update participant status based on successful/failed transactions.
4. **Event Details:**
   - Store event-related data such as event date, location, description, and budget.

Revised Database Schema

We will need several tables to track participants, payments, event costs, and expenses.

1. Participants Table

This table stores participant details like name, contact information, and payment status.

sql
Copy code

```
CREATE TABLE Participants (
    participant_id INT AUTO_INCREMENT PRIMARY KEY,
    name VARCHAR(100) NOT NULL,
    email VARCHAR(100) UNIQUE NOT NULL,
    phone VARCHAR(15) NOT NULL,
    registration_date TIMESTAMP DEFAULT CURRENT_TIMESTAMP,
    payment_method ENUM('V-Kash', 'Nagad') NOT NULL,
    payment_status ENUM('Pending', 'Completed') DEFAULT 'Pending',
    payment_amount DECIMAL(10, 2) NOT NULL
);
```

2. Transactions Table

This table tracks the payment transactions for participants.

sql
Copy code
```
CREATE TABLE Transactions (
   transaction_id INT AUTO_INCREMENT PRIMARY KEY,
   participant_id INT,
   transaction_date TIMESTAMP DEFAULT CURRENT_TIMESTAMP,
   payment_method ENUM('V-Kash', 'Nagad') NOT NULL,
   amount DECIMAL(10, 2) NOT NULL,
   transaction_status ENUM('Success', 'Failed') DEFAULT 'Pending',
   FOREIGN KEY (participant_id) REFERENCES Participants(participant_id)
);
```

3. Event Table

This table stores details about the event, including name, date, location, and description.

sql
Copy code
```
CREATE TABLE Event (
   event_id INT AUTO_INCREMENT PRIMARY KEY,
   event_name VARCHAR(100) NOT NULL,
   event_date DATE NOT NULL,
   event_location VARCHAR(255) NOT NULL,
   event_description TEXT,
   event_budget DECIMAL(10, 2) NOT NULL
);
```

4. Expenses Table

This table stores all the expenses related to organizing the event (e.g., venue rental, food, decoration, etc.).

sql
Copy code
```
CREATE TABLE Expenses (
   expense_id INT AUTO_INCREMENT PRIMARY KEY,
```

```
    event_id INT,
    expense_type VARCHAR(100),
    expense_amount DECIMAL(10, 2) NOT NULL,
    expense_date TIMESTAMP DEFAULT CURRENT_TIMESTAMP,
    FOREIGN KEY (event_id) REFERENCES Event(event_id)
);
```

5. Income Table

This table tracks income generated from participant registrations (i.e., the payments received).

```
sql
Copy code
CREATE TABLE Income (
    income_id INT AUTO_INCREMENT PRIMARY KEY,
    event_id INT,
    income_amount DECIMAL(10, 2) NOT NULL,
    income_date TIMESTAMP DEFAULT CURRENT_TIMESTAMP,
    FOREIGN KEY (event_id) REFERENCES Event(event_id)
);
```

Example Data Insertion

1. Insert Event Data:

```
sql
Copy code
INSERT INTO Event (event_name, event_date, event_location, event_description,
event_budget)
VALUES ('Iftar Mahfil', '2024-04-05', 'Dhaka, Bangladesh', 'A community Iftar
gathering during Ramadan', 10000.00);
```

2. Insert Participant Data:

```
sql
Copy code
INSERT INTO Participants (name, email, phone, payment_method,
payment_status, payment_amount)
VALUES ('Ahmed Rafiq', 'ahmed.rafiq@example.com', '01812345678', 'V-Kash',
'Completed', 500.00);
```

```
INSERT INTO Participants (name, email, phone, payment_method,
payment_status, payment_amount)
VALUES ('Mariam Akter', 'mariam.akter@example.com', '01798765432', 'Nagad',
'Completed', 500.00);
```

3. Insert Transaction Data:

```
sql
Copy code
INSERT INTO Transactions (participant_id, payment_method, amount,
transaction_status)
VALUES (1, 'V-Kash', 500.00, 'Success');
INSERT INTO Transactions (participant_id, payment_method, amount,
transaction_status)
VALUES (2, 'Nagad', 500.00, 'Success');
```

4. Insert Expense Data:

```
sql
Copy code
INSERT INTO Expenses (event_id, expense_type, expense_amount)
VALUES (1, 'Venue Rental', 2000.00);
INSERT INTO Expenses (event_id, expense_type, expense_amount)
VALUES (1, 'Food Catering', 4000.00);
INSERT INTO Expenses (event_id, expense_type, expense_amount)
VALUES (1, 'Decoration', 1500.00);
INSERT INTO Expenses (event_id, expense_type, expense_amount)
VALUES (1, 'Sound System', 1000.00);
```

5. Insert Income Data:

```
sql
Copy code
INSERT INTO Income (event_id, income_amount)
VALUES (1, 1000.00);  -- Total amount from participants
```

Example Functionalities of the System

1. **Register Participant:**
    - **Input:** Name, Email, Phone, Payment Method, Payment Amount.

- **Process:** Add a new record in the **Participants** table. Once payment is confirmed, a record will be added to the **Transactions** table.

2. **Track Event Expenses:**
   - **Input:** Expense type (venue, food, decoration, etc.), expense amount.
   - **Process:** Record the expense in the **Expenses** table.
3. **Track Event Income:**
   - **Input:** Payment details (participant payments).
   - **Process:** Update the **Income** table with the total income generated.
4. **Calculate Event Profit:**
   - **Process:** Calculate the profit for the event by subtracting total expenses from total income.

Example SQL to calculate the profit:

sql
Copy code

```
SELECT
   (SELECT SUM(income_amount) FROM Income WHERE event_id = 1)
AS total_income,
   (SELECT SUM(expense_amount) FROM Expenses WHERE event_id =
1) AS total_expenses,
   (SELECT SUM(income_amount) FROM Income WHERE event_id = 1) -
   (SELECT SUM(expense_amount) FROM Expenses WHERE event_id =
1) AS event_profit;
```

Full SQL File

Here is the complete SQL script for creating the tables and inserting example data:

sql
Copy code

```
-- Create Event Table
CREATE TABLE Event (
   event_id INT AUTO_INCREMENT PRIMARY KEY,
   event_name VARCHAR(100) NOT NULL,
   event_date DATE NOT NULL,
   event_location VARCHAR(255) NOT NULL,
   event_description TEXT,
   event_budget DECIMAL(10, 2) NOT NULL
```

```
);

-- Create Participants Table
CREATE TABLE Participants (
    participant_id INT AUTO_INCREMENT PRIMARY KEY,
    name VARCHAR(100) NOT NULL,
    email VARCHAR(100) UNIQUE NOT NULL,
    phone VARCHAR(15) NOT NULL,
    registration_date TIMESTAMP DEFAULT CURRENT_TIMESTAMP,
    payment_method ENUM('V-Kash', 'Nagad') NOT NULL,
    payment_status ENUM('Pending', 'Completed') DEFAULT 'Pending',
    payment_amount DECIMAL(10, 2) NOT NULL
);

-- Create Transactions Table
CREATE TABLE Transactions (
    transaction_id INT AUTO_INCREMENT PRIMARY KEY,
    participant_id INT,
    transaction_date TIMESTAMP DEFAULT CURRENT_TIMESTAMP,
    payment_method ENUM('V-Kash', 'Nagad') NOT NULL,
    amount DECIMAL(10, 2) NOT NULL,
    transaction_status ENUM('Success', 'Failed') DEFAULT 'Pending',
    FOREIGN KEY (participant_id) REFERENCES Participants(participant_id)
);

-- Create Expenses Table
CREATE TABLE Expenses (
    expense_id INT AUTO_INCREMENT PRIMARY KEY,
    event_id INT,
    expense_type VARCHAR(100),
    expense_amount DECIMAL(10, 2) NOT NULL,
    expense_date TIMESTAMP DEFAULT CURRENT_TIMESTAMP,
    FOREIGN KEY (event_id) REFERENCES Event(event_id)
);

-- Create Income Table
CREATE TABLE Income (
    income_id INT AUTO_INCREMENT PRIMARY KEY,
    event_id INT,
    income_amount DECIMAL(10, 2) NOT NULL,
```

```
    income_date TIMESTAMP DEFAULT CURRENT_TIMESTAMP,
    FOREIGN KEY (event_id) REFERENCES Event(event_id)
);

-- Insert Event Data
INSERT INTO Event (event_name, event_date, event_location, event_description,
event_budget)
VALUES ('Iftar Mahfil', '2024-04-05', 'Dhaka, Bangladesh', 'A community Iftar
gathering during Ramadan', 10000.00);

-- Insert Participant Data
INSERT INTO Participants (name, email, phone, payment_method,
payment_status, payment_amount)
VALUES ('Ahmed Rafiq', 'ahmed.rafiq@example.com', '01812345678', 'V-Kash',
'Completed', 500.00);
INSERT INTO Participants (name, email, phone, payment_method,
payment_status, payment_amount)
VALUES ('Mariam Akter', 'mariam.akter@example.com', '01798765432', 'Nagad',
'Completed', 500.00);

-- Insert Transaction Data
INSERT INTO Transactions (participant_id, payment_method, amount,
transaction_status)
VALUES (1, 'V-Kash', 500.00, 'Success');
INSERT INTO Transactions (participant_id, payment_method, amount,
transaction_status)
VALUES (2, 'Nagad', 500.00, 'Success');

-- Insert Expense Data
INSERT INTO Expenses (event_id, expense_type, expense_amount)
VALUES (1, 'Venue Rental', 2000.00);
INSERT INTO Expenses (event_id, expense_type, expense_amount)
VALUES (1, 'Food Catering', 4000.00);
INSERT INTO Expenses (event_id, expense_type, expense_amount)
VALUES (1, 'Decoration', 1500.00);
INSERT INTO Expenses (event_id, expense_type, expense_amount)
VALUES (1, 'Sound System', 1000.00);

-- Insert Income Data
INSERT INTO Income (event_id, income_amount)
```

```
VALUES (1, 1000.00); -- Total amount from participants
```

This SQL file will create the necessary tables and insert some example data related to participants, transactions, event expenses, and income. You can extend this project further by building a front-end to interact with this system.