# Entity-Relationship (ER) Table Design

## 1. Entity-Relationship (ER) Table Design

**1. Users Table** (Stores user details and authentication information)

| **Field Name** | **Data Type** | **Width** | **Description** |
|---|---|---|---|
| **user_id** (PK) | String (UID) | 28-36 | Unique identifier for each user (Primary Key). |
| name | String | 50 | Full name of the user. |
| email | String | 100 | User’s email (also used for authentication). |
| phone_number | String | 15 | Contact number of the user. |
| role | String | 10 | Defines user type (Admin or Normal User). |
| registered_events | Array | Varies | List of event IDs the user has registered for. |
| created_at | Timestamp | 8 | Date and time when the user signed up. |

**2. Events Table** (Stores details of all events)

| **Field Name** | **Data Type** | **Width** | **Description** |
|---|---|---|---|
| **event_id** (PK) | String | 28-36 | Unique identifier for each event (Primary Key). |
| title | String | 100 | Name of the event. |
| date_time | Timestamp | 8 | Date and time of the event. |
| venue | String | 100 | Location where the event will take place. |
| competitions | String | 50 | Type of competition (Group, Individual, or both). |
| prizes | String | 50 | Prizes awarded to winners. |
| organizers | String | 255 | Names of people organizing the event. |
| poster | String (URL) | 500 | URL of the event poster image. |
| past_images | Array (URLs) | Varies | Images from past events. |
| created_by (FK) | String (User ID) | 28-36 | User ID of the admin who added the event. |
| created_at | Timestamp | 8 | Date and time the event was added. |

**3. Registrations Table** (Stores event registrations)

| **Field Name** | **Data Type** | **Width** | **Description** |
|---|---|---|---|
| **registration_id** (PK) | String | 28-36 | Unique identifier for each registration (Primary Key). |
| user_id (FK) | String | 28-36 | ID of the user who registered. |
| event_id (FK) | String | 28-36 | ID of the event the user registered for. |
| registered_at | Timestamp | 8 | Date and time of registration. |

**4. Event Results Table** (Stores winners and event results)

| **Field Name** | **Data Type** | **width** | **Description** |
|---|---|---|---|
| **result_id** (PK) | String | 28-36 | Unique identifier for each event result (Primary Key). |
| event_id (FK) | String | 28-36 | ID of the event the result belongs to. |
| winner_id (FK) | String | 28-36 | ID of the user who won the event. |
| position | String | 10 | 1st, 2nd, 3rd place, etc. |
| prize_awarded | String | 50 | Prize given to the winner. |
| uploaded_by (FK) | String | 28-36 | Admin ID who uploaded the results. |
| uploaded_at | Timestamp | 8 | Date and time when the result was uploaded. |

**5. Admin Table** (Stores admin details)

| **Field Name** | **Data Type** | **Width** | **Description** |
|---|---|---|---|
| **admin_id** (PK) | String | 28-36 | Unique identifier for each admin (Primary Key). |
| name | String | 50 | Admin’s full name. |
| email | String | 100 | Admin’s email (used for login). |
| phone_number | String | 15 | Contact number. |
| role | String | 10 | Always set as “Admin”. |
| created_at | Timestamp | 8 | Date and time the admin was added. |

---

## 2. Data Flow Diagram (DFD) Explanation

The **Data Flow Diagram (DFD)** describes the flow of data within your system. The key processes and data stores are:

### 1️⃣ User Signup/Login

- User signs up → Data is stored in **Users Table**.
- If user is an admin, the role is set as "Admin" in the database.

### 2️⃣ User Views Events

- The app fetches event details from **Events Table**.

### 3️⃣ User Registers for an Event

- User selects an event and registers → Data is stored in **Registrations Table**.

### 4️⃣ Admin Adds/Edits/Removes Events

- Admin adds a new event → Data is stored in **Events Table**.
- Admin can edit or delete event details.

### 5️⃣ Admin Views Registered Users

- Admin checks event registrations from **Registrations Table**.

### 6️⃣ Admin Uploads Event Results

- Admin selects an event and uploads winners → Data is stored in **Event Results Table**.

---

## Final Notes:

- **Primary Keys (PK)**: Unique identifiers for each record.
- **Foreign Keys (FK)**: Used to create relationships between tables.
- Firestore is a NoSQL database, so actual relationships are managed through document references rather than traditional relational constraints.
- Indexing should be enabled on frequently queried fields like `event_id`, `user_id`, and `email` for better performance.
- The **width** column specifies the expected size for each field.