

The task is: Design and implement a database for 2026 Winter Olympics

Robel Yohannes Wolie – S1119840

BUSINESS INFORMATION SYSTEMS

Prof. Alex Mircoli

April 29, 2025

Università Politecnica delle Marche

1. Introduction

This report outlines the database design for the **Milan–Cortina 2026 Winter Olympic Games Management System**. The aim is to develop a comprehensive and scalable relational database that supports the operations, logistics, and management of the upcoming Olympic and Paralympic Games in Italy. The system is expected to manage athletes, events, venues, accommodations, volunteers, transportation, and media coverage.

2. Purpose of the Database

The primary objectives of this database system are:

- Efficiently manage athlete information, event scheduling, and results tracking.
- Monitor accommodation, security, medical, and hospitality services.
- Track participation and ticket sales for event planning.
- Coordinate volunteers, media personnel, and transportation logistics.
- Support multiple stakeholders including the IOC, CONI, OCOG, IPC, and EOC.

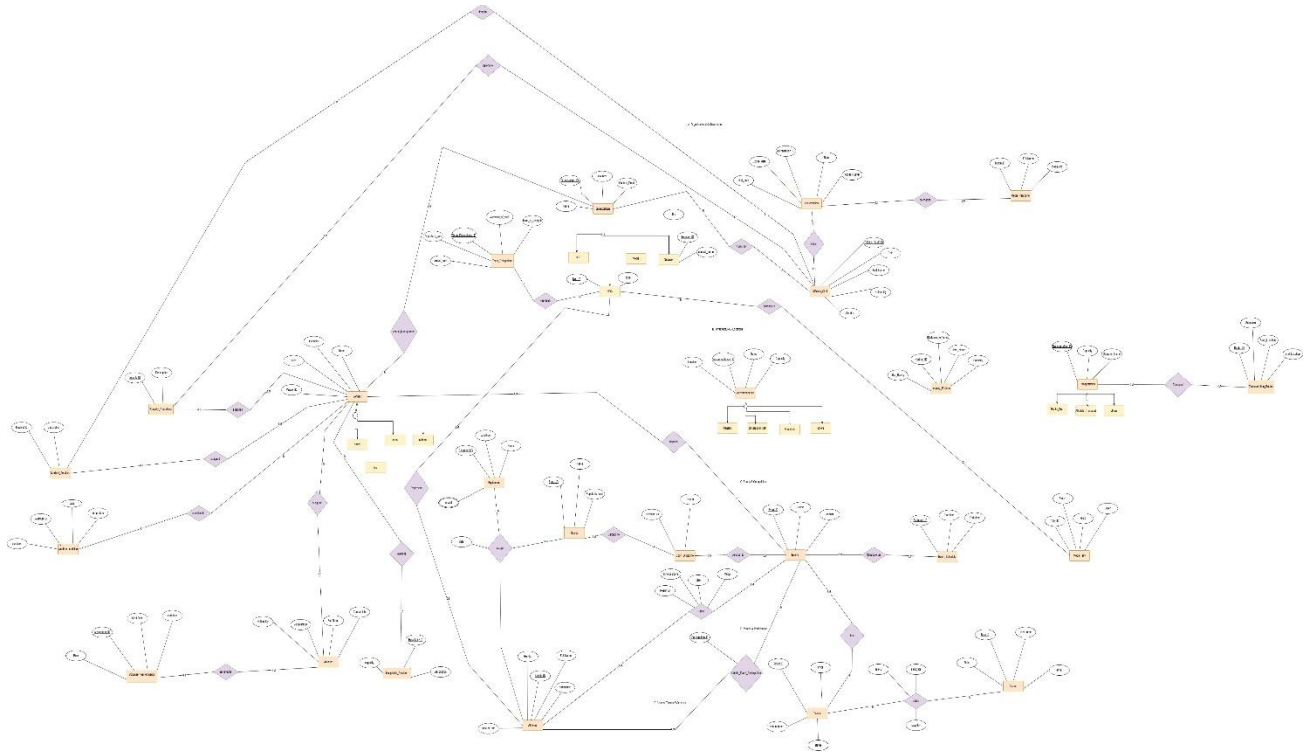
3. Data Requirements

Data Requirements: Milan–Cortina 2026 Winter Olympic Games Management System

- **Athletes** are uniquely identified by **AthleteID**. For each athlete, we record their **FullName**, **Nationality**, **Gender**, **Age**, and the **SportDiscipline** they are affiliated with.
- **Sport Events** are identified by **EventID** and described by their **Name**, **GenderCategory**, and the **VenueID** where they take place.
- **Event Schedules** are tracked using **ScheduleID**, and include **EventID**, **StartTime**, and **EndTime**. Each event may have multiple scheduled instances.
- **Athlete Participation** in events is tracked by **ParticipationID**, linking each **AthleteID** to an **EventID**.
- **Results** are uniquely identified by **ResultID**, and include **EventID**, **AthleteID**, **Rank**, **TimeOrScore**, and the awarded **Medal** (if any).
- **Venues** are uniquely identified by **VenueID**. Each venue includes a **Name**, **City**, **Capacity**, and **Type** (e.g., stadium, rink).
- **Accommodation** facilities are identified by **AccommodationID**, and include **Name**, **Type**, **Capacity**, and **Location**.
- **Lodging Partners** are recorded with **PartnerID**, and include **HotelName**, **StarRating**, **Capacity**, and **DistanceToVenue**.
- **Hospitality Services** are recorded per venue using **HospitalityID**, including **Description**, **VenueID**, and **Capacity**.
- **Weather Conditions** are tracked per venue with a unique **WeatherID**. Each record includes the **VenueID**, **Date**, **Temperature**, and **Condition** (e.g., snowy, sunny).
- **Medical Services** are uniquely identified by **MedicalID**, and include a **Description**, **StaffID**, and the **VenueID** where the service is provided.
- **Security Operations** are managed using **SecurityID**, and each entry includes **Description** and **VenueID**.
- **Officials and Staff** are uniquely identified by **Official_StaffID**. For each person, we store **FullName**, **Role**, **Gender**, **Nationality**, **OrganizationID**, and if applicable, associated **AccreditationID**, **MedicalID**, and **SecurityID**.

- **Volunteers** are identified by **VolunteerID**, and include **FullName**, **Nationality**, **VenueID**, and **ContactInfo**.
- **Volunteer Assignments** are tracked with **AssignmentID**, including **VolunteerID**, **Role**, **StartDate**, and **EndDate**.
- **Media Personnel** are uniquely identified by **MediaID**, and each record includes **FullName** and **MediaOrg**.
- **Media Accreditation** is stored separately using **AccreditationID**, which includes **Role**, **AccessLevel**, **IssueDate**, **ExpiryDate**, and a reference to **MediaID**.
- **National Olympic Committees (NOCs)** are uniquely identified by **NOC_ID**, and include the **Country**, **Code**, and **OrganizationID**.
- **Team Delegations** are tracked using **DelegationID**, with details including **NOC_ID**, **HeadOfDelegation**, **NumberOfStaff**, **ArrivalDate**, and **DepartureDate**.
- **Organizations** are identified by **OrganizationID**, and include **Name**, **Type**, **Country**, and **ContactEmail**.
- **Sponsors** are tracked using **SponsorID**, and each includes **Name**, **Tier**, **ContactPerson**, and **OrganizationID**.
- **Users** of the system are uniquely identified by **UserID**, and include **Username**, **Role**, and **Email**.
- **Tickets** are assigned a unique **TicketID**, and include **EventID**, **SeatNumber**, **Price**, and **Status**.
- **Sales Transactions** are tracked using **SaleID**, and include **TicketID**, **UserID**, **SaleDate**, and **Quantity** purchased.
- **Sports** are identified by **SportID**, and each entry includes **Name** and **Popularity_Level**.
- **Disciplines** are grouped under sports using **DisciplineID**, with **SportID** and **Name** recorded.
- **Equipment** is recorded using **EquipmentID**, and includes **Name**, **Model**, and **Condition**.
- **Usage** of equipment is tracked per date, identifying which **AthleteID** used which **EquipmentID** for which **SportID**, recorded by **Date**.
- **Transportation Resources** are uniquely identified by **TransportID**, and include **Type**, **Capacity**, and **OperatorName**.

4. Enhanced Entity Diagram



5. logical design (relations + FK)

Table	Attributes and foreign key
Accommodation	AccommodationID, Name, Type, Capacity, Location
Accreditation	AccreditationID, Role, AccessLevel, IssueDate, ExpiryDate, mediaId FK Accreditation.media_personnel= media_personnel.mediaId
Athlete_Event_Participation	ParticipationId, athleteID, EventID FK Athlete_Event_Participation.athlete= athlete. athleteID FK Athlete_Event_Participation.participation= participation.participationID
Result	ResultID, EventID, AthleteID, Rank, TimeOrScore, Medal FK Result.event=event.eventID FK Result.Athlet=athlete.athleteID
Event_Schedule	ScheduleID, EventID, StartTime, EndTime FK Event_Schedule.event=event.eventID
Event	EventID, Name, Gender, VenueID FK event.venue=venue.venueID
Hospitality_Services	HospitalityID, Description, VenueID, Capacity FK Hospitality_Services.venue=venue.venueID
Lodging_Partners	PartnerID, HotelName, Capacity, StarRating, DistanceToVenue
Medal_Tally	TallyID, NOC_ID, Gold, Silver, Bronze FK Medal_Tally.NOC=NOC.NOCID
Media_Personal	MediaID, FullName, MediaOrg
Medical_Services	MedicalID,description, StaffID, VenueID

Table	Attributes and foreign key
	FK Medical_Services.VenueID= Venue. VenueID
National_Olympic_Committees (NOCs)	NOC_ID, Country, Code, OrganizationID, FK NOC.organization=organization. organizationID
Officials_Staff	Official_StaffID, FullName, role, OrganizationID, gender, Nationality, AccreditationID, medicalID, securityID FK Officials_Staff.accreditation= Accreditation. AccreditationID FK Officials_Staff.Organization=Organization.OrganizationID FK Officials_Staff. medical_service=medical_service.medicalID FK Officials_Staff.securiry operation= securiry_operation.securiryID
Organizations	OrganizationID, Name, Type, Country, ContactEmail
Security_Operations	SecurityID, Description, VenueID FK Security_Operations.venue=venue.venueID
Sponsors	SponsorID, Name, Tier, ContactPerson, OrganizationID FK Sponsors.organization=organization.OrganizationID
Sport_Discipline	DisciplineID, SportID, Name, FK Sport_Discipline.sport=sport.sportID
Equipment	EquipmentID, condition, name , model
Usage	Date, EquipmentID, sportID, AthletID FK Usage.equipment= equipment.EquipmentID FK Usage.sports=sport.sportID FK usage.athlet=athlet.athletID
Sports	SportID, Name, popularity_level

Table	Attributes and foreign key
Team_Delegations	DelegationID, NOC_ID, HeadOfDelegation, NumberOfStaff, ArrivalDate, DepartureDate FK Team_Delegations.NOC=NOC.NOCID
Sales	SaleID, TicketID, UserID, SaleDate, Quantity FK sales.ticket=ticket.ticketid FK sales=user.userID
Ticket	TicketID, EventID, Price, SeatNumber, Status FK Ticket.event=event.eventID
Transportation	TransportID, Type, Capacity, OperatorName
Transportation_Routes	RouteID, TransportID, StartLocation, EndLocation, Schedule FK Transportation_Routes.routeID= route.routeID
Users	UserID, Username, Role, Email
Venue_Management	venueID, OrganizationID FK Venue_Management.organization= Organization.OrganizationID FK Venue_Management. Venue=venue. venueID
Venues	VenueID, Name, City, Capacity, Type
Volunteer_Assignments	AssignmentID, VolunteerID, Role, StartDate, EndDate FK Volunteer_Assignments. Volunteer=Volunteer. VolunteerID
Volunteers	VolunteerID, FullName, Nationality, VenueID, ContactInfo FK Volunteers. Venue= Venue.VenueID
Weather_Conditions	WeatherID, VenueID, Date, Temperature, Condition FK Weather_Conditions.Venue=Venue. VenueID

6. SQL queries

a) DDL

-- Organizations

CREATE TABLE Organizations (

OrganizationID INT PRIMARY KEY,

Name VARCHAR(100),

Type VARCHAR(50),

Country VARCHAR(50),

ContactEmail VARCHAR(100)

);

-- Sports

CREATE TABLE Sports (

SportID INT PRIMARY KEY,

Name VARCHAR(100),

Popularity_Level VARCHAR(50)

);

-- Sport Discipline

CREATE TABLE Sport_Discipline (

DisciplineID INT PRIMARY KEY,

SportID INT,

Name VARCHAR(100),

FOREIGN KEY (SportID) REFERENCES Sports(SportID)

);

-- Venue

```
CREATE TABLE Venue (
```

```
VenueID INT PRIMARY KEY,
```

```
Name VARCHAR(100),
```

```
City VARCHAR(100),
```

```
Capacity INT,
```

```
Type VARCHAR(50)
```

```
);
```

```
-- Venue Management
```

```
CREATE TABLE Venue_Management (
```

```
VenueID INT,
```

```
OrganizationID INT,
```

```
PRIMARY KEY (VenueID, OrganizationID),
```

FOREIGN KEY (VenueID) REFERENCES Venue(VenueID),

FOREIGN KEY (OrganizationID) REFERENCES Organizations(OrganizationID)

);

-- National Olympic Committees

CREATE TABLE National_Olympic_Committees (

NOC_ID INT PRIMARY KEY,

Country VARCHAR(100),

Code VARCHAR(10),

OrganizationID INT,

FOREIGN KEY (OrganizationID) REFERENCES Organizations(OrganizationID)

);

-- Team Delegations

CREATE TABLE Team_Delegations (

DelegationID INT PRIMARY KEY,

NOC_ID INT,

HeadOfDelegation VARCHAR(100),

NumberOfStaff INT,

ArrivalDate DATE,

DepartureDate DATE,

FOREIGN KEY (NOC_ID) REFERENCES National_Olympic_Committees(NOC_ID)

);

-- Accommodation

CREATE TABLE Accommodation (

AccommodationID INT PRIMARY KEY,

Name VARCHAR(100),

Type VARCHAR(50),

Capacity INT,

Location VARCHAR(255)

);

-- Media Personal

CREATE TABLE Media_Personal (

MediaID INT PRIMARY KEY,

FullName VARCHAR(100),

MediaOrg VARCHAR(100)

);

-- Accreditation

CREATE TABLE Accreditation (

AccreditationID INT PRIMARY KEY,

Role VARCHAR(50),

AccessLevel VARCHAR(50),

IssueDate DATE,

ExpiryDate DATE,

MediaID INT,

FOREIGN KEY (MediaID) REFERENCES Media_Personal(MediaID)

);

-- Transportation

```
CREATE TABLE Transportation (  
  
    TransportID INT PRIMARY KEY,  
  
    Type VARCHAR(50),  
  
    Capacity INT,  
  
    OperatorName VARCHAR(100)  
  
);
```

-- Transportation Routes

```
CREATE TABLE Transportation_Routes (  
  
    RouteID INT PRIMARY KEY,  
  
    TransportID INT,  
  
    StartLocation VARCHAR(100),  
  
    EndLocation VARCHAR(100),
```


Schedule VARCHAR(100),

FOREIGN KEY (TransportID) REFERENCES Transportation(TransportID)

);

-- Hospitality Services

CREATE TABLE Hospitality_Services (

HospitalityID INT PRIMARY KEY,

Description TEXT,

VenueID INT,

Capacity INT,

FOREIGN KEY (VenueID) REFERENCES Venue(VenueID)

);

-- Lodging Partners

CREATE TABLE Lodging_Partners (

PartnerID INT PRIMARY KEY,

HotelName VARCHAR(100),

Capacity INT,

StarRating INT,

DistanceToVenue DECIMAL(5,2)

);

-- Security Operations

CREATE TABLE Security_Operations (

SecurityID INT PRIMARY KEY,

Description TEXT,

VenueID INT,

FOREIGN KEY (VenueID) REFERENCES Venue(VenueID)

);

-- Medical Services

CREATE TABLE Medical_Services (

MedicalID INT PRIMARY KEY,

Description TEXT,

StaffID INT,

VenueID INT,

FOREIGN KEY (VenueID) REFERENCES Venue(VenueID)

);

-- Officials Staff

CREATE TABLE Officials_Staff (

Official_StaffID INT PRIMARY KEY,

FullName VARCHAR(100),

Role VARCHAR(50),

OrganizationID INT,

Gender CHAR(1),

Nationality VARCHAR(50),

AccreditationID INT,

MedicalID INT,

SecurityID INT,

FOREIGN KEY (OrganizationID) REFERENCES Organizations(OrganizationID),

FOREIGN KEY (AccreditationID) REFERENCES Accreditation(AccreditationID),

FOREIGN KEY (MedicalID) REFERENCES Medical_Services(MedicalID),

FOREIGN KEY (SecurityID) REFERENCES Security_Operations(SecurityID)

);

-- Sponsors

CREATE TABLE Sponsors (

SponsorID INT PRIMARY KEY,

Name VARCHAR(100),

Tier VARCHAR(20),

ContactPerson VARCHAR(100),

OrganizationID INT,

FOREIGN KEY (OrganizationID) REFERENCES Organizations(OrganizationID)

);

-- Equipment

CREATE TABLE Equipment (

EquipmentID INT PRIMARY KEY,

`Condition` VARCHAR(50),

Name VARCHAR(100),

Model VARCHAR(50)

);

-- Athlete

CREATE TABLE Athlete (

AthleteID INT PRIMARY KEY,

FullName VARCHAR(100),

Gender CHAR(1),

DOB DATE,

NOC_ID INT,

FOREIGN KEY (NOC_ID) REFERENCES National_Olympic_Committees(NOC_ID)

);

-- Event must come BEFORE Ticket

CREATE TABLE Event (

EventID INT PRIMARY KEY,

Name VARCHAR(100),

Gender CHAR(1),

VenueID INT,

FOREIGN KEY (VenueID) REFERENCES Venue(VenueID)

```
);
```

```
-- Ticket (after Event exists)
```

```
CREATE TABLE Ticket (
```

```
    TicketID INT PRIMARY KEY,
```

```
    EventID INT,
```

```
    Price DECIMAL(8,2),
```

```
    SeatNumber VARCHAR(10),
```

```
    Status VARCHAR(20),
```

```
    FOREIGN KEY (EventID) REFERENCES Event(EventID)
```

```
);
```

```
-- Users
```



```
CREATE TABLE Users (  
  
    UserID INT PRIMARY KEY,  
  
    Username VARCHAR(100),  
  
    Role VARCHAR(50),  
  
    Email VARCHAR(100)  
  
);
```

-- Sales (after Ticket exists)

```
CREATE TABLE Sales (  
  
    SaleID INT PRIMARY KEY,  
  
    TicketID INT,  
  
    UserID INT,  
  
    SaleDate DATE,
```

Quantity INT,

FOREIGN KEY (TicketID) REFERENCES Ticket(TicketID),

FOREIGN KEY (UserID) REFERENCES Users(UserID)

);

-- Event Schedule

CREATE TABLE Event_Schedule (

ScheduleID INT PRIMARY KEY,

EventID INT,

StartTime TIMESTAMP,

EndTime TIMESTAMP,

FOREIGN KEY (EventID) REFERENCES Event(EventID)

);

-- Weather Conditions

CREATE TABLE Weather_Conditions (

WeatherID INT PRIMARY KEY,

VenueID INT,

Date DATE,

Temperature DECIMAL(5,2),

`Condition` VARCHAR(50),

FOREIGN KEY (VenueID) REFERENCES Venue(VenueID)

);

-- Volunteers

CREATE TABLE Volunteers (

VolunteerID INT PRIMARY KEY,

FullName VARCHAR(100),

Nationality VARCHAR(50),

VenueID INT,

ContactInfo VARCHAR(100),

FOREIGN KEY (VenueID) REFERENCES Venue(VenueID)

);

-- Volunteer Assignments

CREATE TABLE Volunteer_Assignments (

AssignmentID INT PRIMARY KEY,

VolunteerID INT,

Role VARCHAR(50),

StartDate DATE,

EndDate DATE,

FOREIGN KEY (VolunteerID) REFERENCES Volunteers(VolunteerID)

);

-- Medal Tally

CREATE TABLE Medal_Tally (

TallyID INT PRIMARY KEY,

NOC_ID INT,

Gold INT,

Silver INT,

Bronze INT,

FOREIGN KEY (NOC_ID) REFERENCES National_Olympic_Committees(NOC_ID)

```
);
```

```
-- Participation
```

```
CREATE TABLE Participation (
```

```
    ParticipationID INT PRIMARY KEY
```

```
);
```

```
-- Athlete Event Participation
```

```
CREATE TABLE Athlete_Event_Participation (
```

```
    ParticipationID INT PRIMARY KEY,
```

```
    AthleteID INT,
```

```
    EventID INT,
```

```
    FOREIGN KEY (AthleteID) REFERENCES Athlete(AthleteID),
```

FOREIGN KEY (ParticipationID) REFERENCES Participation(ParticipationID)

);

-- Result

CREATE TABLE Result (

ResultID INT PRIMARY KEY,

EventID INT,

AthleteID INT,

`Rank` INT,

TimeOrScore VARCHAR(50),

Medal VARCHAR(20),

FOREIGN KEY (EventID) REFERENCES Event(EventID),

FOREIGN KEY (AthleteID) REFERENCES Athlete(AthleteID)

);

-- Usage (Equipment <-> Sport <-> Athlete)

CREATE TABLE `Usage` (

Date DATE,

EquipmentID INT,

SportID INT,

AthleteID INT,

PRIMARY KEY (Date, EquipmentID, SportID, AthleteID),

FOREIGN KEY (EquipmentID) REFERENCES Equipment(EquipmentID),

FOREIGN KEY (SportID) REFERENCES Sports(SportID),

FOREIGN KEY (AthleteID) REFERENCES Athlete(AthleteID)

);

b) DML

INSERT INTO organizations (

OrganizationID, Name, Type, Country, ContactEmail

) VALUES

(1, 'International Olympic Committee', 'Olympic Committee', 'Switzerland',
'info@olympics.org'),

(2, 'Italian Olympic Committee', 'National Olympic Committee', 'Italy', 'contact@coni.it'),

(3, 'USA Track & Field', 'Sports Federation', 'USA', 'info@usatf.org'),

(4, 'German Ski Association', 'Sports Federation', 'Germany', 'info@ski.de'),

(5, 'Japan Skating Federation', 'Sports Federation', 'Japan', 'info@skating.jp');

INSERT INTO national_olympic_committees (

NOC_ID, Country, Code, OrganizationID

) VALUES

(1, 'Italy', 'ITA', 2),

(2, 'United States', 'USA', 3),

(3, 'Germany', 'GER', 4),

(4, 'Japan', 'JPN', 5),

(5, 'Switzerland', 'SUI', 1);

INSERT INTO venue (

VenueID, `Name`, City, Capacity, `Type`

) VALUES

(1, 'San Siro Stadium', 'Milan', 80000, 'Stadium'),

(2, 'Cortina Ice Rink', 'Cortina', 12000, 'Ice Rink'),

(3, 'Livigno Snow Park', 'Livigno', 5000, 'Snow Park'),

(4, 'Bormio Alpine Centre', 'Bormio', 15000, 'Ski Slope'),

```

(5, 'Fiera Milano', 'Milan', 10000, 'Indoor Arena');

INSERT INTO users (
    UserID, Username, Role, Email
) VALUES
    (1, 'mario_rossi', 'Spectator', 'mario.rossi@example.com'),
    (2, 'lisa_muller', 'Coach', 'lisa.muller@example.com'),
    (3, 'john_doe', 'Admin', 'john.doe@example.com'),
    (4, 'anna_bianchi', 'Sales', 'anna.bianchi@example.com'),
    (5, 'kenji_tanaka', 'Volunteer', 'kenji.tanaka@example.com');

-- Accommodation

INSERT INTO accommodation (AccommodationID, Name, Type, Capacity, Location) VALUES
    (1, 'Olympic Village Milan', 'Apartment', 5000, 'Milan'),
    (2, 'Cortina Lodge', 'Hotel', 400, 'Cortina'),
    (3, 'Bormio Chalet', 'Lodge', 300, 'Bormio'),
    (4, 'Livigno Residence', 'Apartment', 450, 'Livigno'),
    (5, 'City Hotel Milano', 'Hotel', 200, 'Milan');

-- Lodging Partners

INSERT INTO lodging_partners (PartnerID, HotelName, Capacity, StarRating, DistanceToVenue)
VALUES
    (1, 'Hotel Principe di Savoia', 500, 5, 3.0),
    (2, 'Cristallo Resort & Spa', 350, 5, 2.0),
    (3, 'QC Terme Hotel', 400, 4, 4.5),
    (4, 'Grand Hotel Bagni Nuovi', 300, 5, 5.0),
    (5, 'NH Collection Milano', 250, 4, 1.5);

-- Sports

INSERT INTO sports (SportID, Name, Popularity_Level) VALUES

```

```
(1, 'Skiing', 'High'),  
(2, 'Ice Hockey', 'High'),  
(3, 'Figure Skating', 'Medium'),  
(4, 'Snowboarding', 'Medium'),  
(5, 'Speed Skating', 'Medium');
```

-- Sport Discipline

```
INSERT INTO sport_discipline (DisciplineID, SportID, Name) VALUES
```

```
(1, 1, 'Alpine Skiing'),  
(2, 2, 'Men\'s Ice Hockey'),  
(3, 3, 'Pairs Figure Skating'),  
(4, 4, 'Snowboard Halfpipe'),  
(5, 5, '500m Speed Skating');
```

-- Event

```
INSERT INTO event (EventID, Name, Gender, VenueID) VALUES
```

```
(1, 'Downhill', 'M', 4),  
(2, 'Slalom', 'F', 4),  
(3, 'Ice Hockey Final', 'M', 2),  
(4, 'Figure Skating Final', 'F', 2),  
(5, 'Halfpipe Final', 'M', 3);
```

```
INSERT INTO participation (ParticipationID, Date, EventID, VenueID) VALUES
```

```
(1, '2026-02-05', 1, 2),  
(2, '2026-02-06', 2, 2),  
(3, '2026-02-10', 3, 1),  
(4, '2026-02-12', 4, 1),  
(5, '2026-02-15', 5, 2);
```

```
ALTER TABLE participation
```

```
ADD CONSTRAINT fk_participation_venue
```

```
FOREIGN KEY (VenueID) REFERENCES venue(VenueID);
```

```
INSERT INTO athlete (AthleteID, FullName, Gender, DOB, NOC_ID) VALUES
```

```
(1, 'Mikaela Shiffrin', 'F', '1995-03-13', 2),
```

```
(2, 'Yuzuru Hanyu', 'M', '1994-12-07', 4),
```

```
(3, 'Federica Brignone', 'F', '1990-07-14', 1),
```

```
(4, 'Felix Neureuther', 'M', '1984-03-26', 3),
```

```
(5, 'Beat Feuz', 'M', '1987-02-11', 5);
```

```
INSERT INTO result (ResultID, EventID, AthleteID, `Rank`, TimeOrScore, Medal) VALUES
```

```
(1, 1, 1, 1, '1:39.56', 'Gold'),
```

```
(2, 2, 3, 2, '52.23', 'Silver'),
```

```
(3, 4, 2, 1, '149.20', 'Gold'),
```

```
(4, 1, 4, 3, '1:41.22', 'Bronze'),
```

```
(5, 5, 5, 2, '92.5', 'Silver');
```

```
-- Event_Schedule
```

```
INSERT INTO event_schedule (ScheduleID, EventID, StartTime, EndTime) VALUES
```

```
(1, 1, '2026-02-05 10:00:00', '2026-02-05 12:00:00'),
```

```
(2, 2, '2026-02-06 14:00:00', '2026-02-06 16:00:00'),
```

```
(3, 3, '2026-02-10 18:00:00', '2026-02-10 20:00:00'),
```

```
(4, 4, '2026-02-12 19:00:00', '2026-02-12 21:00:00'),
```

```
(5, 5, '2026-02-15 15:00:00', '2026-02-15 17:00:00');
```

```
-- 15. Hospitality_Services
```

```
INSERT INTO hospitality_services (HospitalityID, Description, VenueID, Capacity) VALUES
(1, 'VIP Lounge', 1, 300),
(2, 'Athlete Rest Area', 2, 200),
(3, 'Sponsor Lounge', 3, 150),
(4, 'Press Room', 5, 100),
(5, 'Medical Station', 4, 120);
```

-- 16. Medical_Services

```
INSERT INTO medical_services (MedicalID, Description, StaffID, VenueID) VALUES
(1, 'Emergency Medical Team', 1, 1),
(2, 'First Aid Station', 2, 2),
(3, 'Athlete Physiotherapy', 3, 4),
(4, 'Paramedic Unit', 4, 5),
(5, 'General Clinic', 5, 3);
```

-- Security_Operations

```
INSERT INTO security_operations (SecurityID, Description, VenueID) VALUES
(1, 'VIP Security', 1),
(2, 'Crowd Control', 2),
(3, 'Athlete Escort', 3),
(4, 'Emergency Evacuation Team', 4),
(5, 'Venue Patrol', 5);
```

-- Media Personal

```
INSERT INTO media_personal (MediaID, FullName, MediaOrg) VALUES
```

```
(1, 'Francesca Neri', 'RAI'),  
(2, 'David Thompson', 'NBC Sports'),  
(3, 'Yuki Sato', 'NHK'),  
(4, 'Laura Gonzalez', 'Telemundo Deportes'),  
(5, 'Hans Meier', 'ARD');
```

```
INSERT INTO accreditation (AccreditationID, Role, AccessLevel, IssueDate, ExpiryDate, MediaID)  
VALUES
```

```
(1, 'Journalist', 'Media', '2026-01-01', '2026-03-31', 1),  
(2, 'Photographer', 'Media', '2026-01-01', '2026-03-31', 2),  
(3, 'Camera Operator', 'Media', '2026-01-01', '2026-03-31', 3),  
(4, 'Reporter', 'Media', '2026-01-01', '2026-03-31', 4),  
(5, 'Analyst', 'Media', '2026-01-01', '2026-03-31', 5);
```

```
INSERT INTO officials_staff (Official_StaffID, FullName, Role, OrganizationID, Gender, Nationality,  
AccreditationID, MedicalID, SecurityID) VALUES
```

```
(1, 'Giovanni Russo', 'Judge', 2, 'M', 'Italy', 1, 1, 1),  
(2, 'Susan White', 'Referee', 3, 'F', 'USA', 2, 2, 2),  
(3, 'Akira Yamamoto', 'Judge', 5, 'M', 'Japan', 3, 3, 3),  
(4, 'Hans Keller', 'Timekeeper', 4, 'M', 'Germany', 4, 4, 4),  
(5, 'Maria Gonzalez', 'Referee', 1, 'F', 'Spain', 5, 5, 5);
```

```
-- Sponsors
```

```
INSERT INTO sponsors (SponsorID, Name, Tier, ContactPerson, OrganizationID) VALUES
```

```
(1, 'Coca-Cola', 'Gold', 'James Parker', 1),
```

```
(2, 'Samsung', 'Silver', 'Anna Kim', 2),  
(3, 'Visa', 'Platinum', 'Richard Smith', 3),  
(4, 'Toyota', 'Gold', 'Hiroshi Tanaka', 4),  
(5, 'Omega', 'Official Timekeeper', 'Max Müller', 5)
```

-- Transportation

```
INSERT INTO transportation (TransportID, Type, Capacity, OperatorName) VALUES
```

```
(1, 'Bus', 50, 'ATM Milan'),  
(2, 'Shuttle', 20, 'Dolomiti Express'),  
(3, 'Minivan', 8, 'Bormio Transfers'),  
(4, 'Train', 200, 'Trenord'),  
(5, 'Helicopter', 5, 'HeliService Milan');
```

-- 21. Transportation_Routes

```
INSERT INTO transportation_routes (RouteID, TransportID, StartLocation, EndLocation,  
Schedule) VALUES
```

```
(1, 1, 'Olympic Village', 'San Siro Stadium', '2026-02-05 08:00:00'),  
(2, 2, 'Cortina Village', 'Ice Rink', '2026-02-06 10:00:00'),  
(3, 3, 'Bormio Hotels', 'Alpine Centre', '2026-02-07 09:00:00'),  
(4, 4, 'Milano Centrale', 'Venues', '2026-02-08 07:30:00'),  
(5, 5, 'Milan Airport', 'Livigno Park', '2026-02-09 11:00:00');
```

-- Venue_Management

```
INSERT INTO venue_management (VenueID, OrganizationID) VALUES
```

```
(1, 2),  
(2, 1),  
(3, 4),
```

(4, 3),

(5, 5);

INSERT INTO volunteers (VolunteerID, FullName, Nationality, VenueID, ContactInfo) VALUES

(1, 'Elena Ferrari', 'Italy', 1, 'elena.ferrari@example.com'),

(2, 'Lukas Schmidt', 'Germany', 2, 'lukas.schmidt@example.com'),

(3, 'Sakura Tanaka', 'Japan', 3, 'sakura.tanaka@example.com'),

(4, 'Emily Johnson', 'USA', 4, 'emily.johnson@example.com'),

(5, 'Carlos Martinez', 'Spain', 5, 'carlos.martinez@example.com');

INSERT INTO volunteer_assignments (AssignmentID, VolunteerID, Role, StartDate, EndDate)
VALUES

(1, 1, 'Usher', '2026-01-20', '2026-02-25'),

(2, 2, 'Information Desk', '2026-01-22', '2026-02-26'),

(3, 3, 'Medical Support', '2026-01-25', '2026-02-28'),

(4, 4, 'Security Assistance', '2026-01-27', '2026-03-01'),

(5, 5, 'VIP Assistance', '2026-01-30', '2026-03-05');

-- Equipment

INSERT INTO equipment (EquipmentID, `Condition`, Name, Model) VALUES

(1, 'Excellent', 'Ski Boots', 'Atomic Redster'),

(2, 'Good', 'Snowboard', 'Burton Custom'),

(3, 'Excellent', 'Ice Skates', 'Riedell 435 Bronze Star'),

(4, 'Fair', 'Speed Skates', 'Viking Icon'),

(5, 'Good', 'Hockey Stick', 'Bauer Vapor FlyLite');


```
INSERT INTO `usage` (Date, EquipmentID, SportID, AthleteID) VALUES
```

```
('2026-02-05', 1, 1, 1),
```

```
('2026-02-06', 2, 4, 5),
```

```
('2026-02-10', 3, 3, 2),
```

```
('2026-02-12', 4, 5, 5),
```

```
('2026-02-15', 5, 2, 4);
```

```
-- Ticket
```

```
-----
```

```
INSERT INTO ticket (TicketID, EventID, Price, SeatNumber, Status) VALUES
```

```
(1, 1, 150.00, 'A01', 'Available'),
```

```
(2, 2, 120.00, 'B02', 'Available'),
```

```
(3, 3, 200.00, 'C03', 'Available'),
```

```
(4, 4, 180.00, 'D04', 'Available'),
```

```
(5, 5, 130.00, 'E05', 'Available');
```

```
-- 26. Weather_Conditions
```

```
INSERT INTO weather_conditions (WeatherID, VenueID, Date, Temperature, `Condition`) VALUES
```

```
(1, 1, '2026-02-05', 5, 'Clear'),
```

```
(2, 2, '2026-02-06', -2, 'Snow'),
```

```
(3, 3, '2026-02-07', -6, 'Blizzard'),
```

```
(4, 4, '2026-02-08', -4, 'Cloudy'),
```

```
(5, 5, '2026-02-09', 3, 'Sunny');
```

```
-- Users
```

```
-----
```

```
INSERT INTO users (UserID, Username, Role, Email) VALUES
```

```
(1, 'mario_rossi', 'Spectator', 'mario.rossi@example.com'),
```

```
(2, 'lisa_muller', 'Coach', 'lisa.muller@example.com'),  
(3, 'john_doe', 'Admin', 'john.doe@example.com'),  
(4, 'anna_bianchi', 'Sales', 'anna.bianchi@example.com'),  
(5, 'kenji_tanaka', 'Volunteer', 'kenji.tanaka@example.com');
```

-- 27. Sales

```
INSERT INTO sales (SaleID, TicketID, UserID, SaleDate, Quantity) VALUES
```

```
(1, 1, 1, '2026-01-15', 2),  
(2, 2, 2, '2026-01-16', 3),  
(3, 3, 3, '2026-01-17', 1),  
(4, 4, 4, '2026-01-18', 4),  
(5, 5, 5, '2026-01-19', 2);
```

```
INSERT INTO Team_Delegations (DelegationID, NOC_ID, HeadOfDelegation, NumberOfStaff,  
ArrivalDate, DepartureDate) VALUES
```

```
(1, 1, 'Giuseppe Rossi', 20, '2026-02-01', '2026-02-25'),  
(2, 2, 'John Doe', 25, '2026-02-02', '2026-02-26'),  
(3, 3, 'Max Müller', 30, '2026-02-03', '2026-02-27'),  
(4, 4, 'Taro Yamada', 22, '2026-02-04', '2026-02-28'),  
(5, 5, 'Peter Meier', 18, '2026-02-05', '2026-03-01');
```

```
INSERT INTO Medal_Tally (TallyID, NOC_ID, Gold, Silver, Bronze) VALUES
```

```
(1, 1, 10, 5, 3), -- Example: Italy (NOC_ID = 1)  
(2, 2, 8, 6, 4), -- Example: United States (NOC_ID = 2)  
(3, 3, 12, 8, 7), -- Example: Germany (NOC_ID = 3)  
(4, 4, 9, 7, 5), -- Example: Japan (NOC_ID = 4)  
(5, 5, 15, 10, 6); -- Example: Switzerland (NOC_ID = 5)
```

c) complex operations

-- Get all medal winners

```
SELECT a.FullName, r.Medal, e.Name AS EventName
FROM result r
JOIN athlete a ON r.AthleteID = a.AthleteID
JOIN event e ON r.EventID = e.EventID
WHERE r.Medal IS NOT NULL;
```

-- View ticket sales per user

```
SELECT u.Username, t.SeatNumber, s.Quantity, s.SaleDate
FROM sales s
JOIN users u ON s.UserID = u.UserID
JOIN ticket t ON s.TicketID = t.TicketID;
```

-- Medal Summary Per Country with Athlete Names

```
SELECT
    n.country,
    a.FullName AS Athlete,
    e.Name AS Event,
    r.Medal
FROM result r
JOIN athlete a ON r.AthleteID = a.AthleteID
JOIN national_olympic_committees n ON a.NOC_ID = n.NOC_ID
```

```
JOIN event e ON r.EventID = e.EventID
```

```
WHERE r.Medal IS NOT NULL
```

```
ORDER BY n.country, r.Medal DESC;
```

```
-- Total Medals by Country with Aggregation
```

```
SELECT
```

```
    n.country,
```

```
    SUM(CASE WHEN r.Medal = 'Gold' THEN 1 ELSE 0 END) AS Golds,
```

```
    SUM(CASE WHEN r.Medal = 'Silver' THEN 1 ELSE 0 END) AS Silvers,
```

```
    SUM(CASE WHEN r.Medal = 'Bronze' THEN 1 ELSE 0 END) AS Bronzes,
```

```
    COUNT(r.Medal) AS Total
```

```
FROM national_olympic_committees n
```

```
LEFT JOIN athlete a ON a.NOC_ID = n.NOC_ID
```

```
LEFT JOIN result r ON a.AthleteID = r.AthleteID
```

```
GROUP BY n.country
```

```
ORDER BY Golds DESC, Silvers DESC, Bronzes DESC
```

7. Conclusion

The proposed database for the Milan–Cortina 2026 Winter Olympic Games is designed to efficiently manage key operations such as athlete tracking, event scheduling, accommodations, and logistics. With its scalable structure and support for multiple stakeholders, the system ensures smooth coordination, real-time data access, and reliable management of the Games.