

Milestone 1a: Requirements Analysis

The goal of the project is the creation of a database for a sports club, the Black Wings Linz.

Domain description

The database stores information on the employees, positions for employees, facilities (offices, stadium, flats), fans, teams, players, coaches, positions for coaches, games and sold and available tickets of the club.

The club is identified by its name. It also has a founding year.

Employees work for the club. Each employee can be identified by their social security number (SSN). Also, the name and birthday of each employee is stored.

The several positions at the club (like President, General Manager,...) are stored with their respective salaries and an indicator, whether the position supervises other positions (Yes/No). For positions which have a supervisor, the supervisor is also stored. A position can supervise several other positions, but each position has only one supervisor. A position is always located at a certain office.

Offices are one kind of the club's facilities. As all facilities, they can be identified by a unique address. They also have open hours, a construction year and a floorspace.

Flats, which are rented out to players (only one player per flat and vice versa) if needed, are another kind of facility. Additionally, to their address, they have a rent and a floorspace.

The third kind of facility is the club's stadium. It has address, capacity, construction year and a number of parking spots. It also has several dressing rooms which have a unique number within the building and a number of lockers.

The club has a fan club which can be joined by providing an email address (unique) and a name.

The club has several teams for different age ranges. Teams have a unique Team ID. The teams of course all belong to the club. Each team has a number of players.

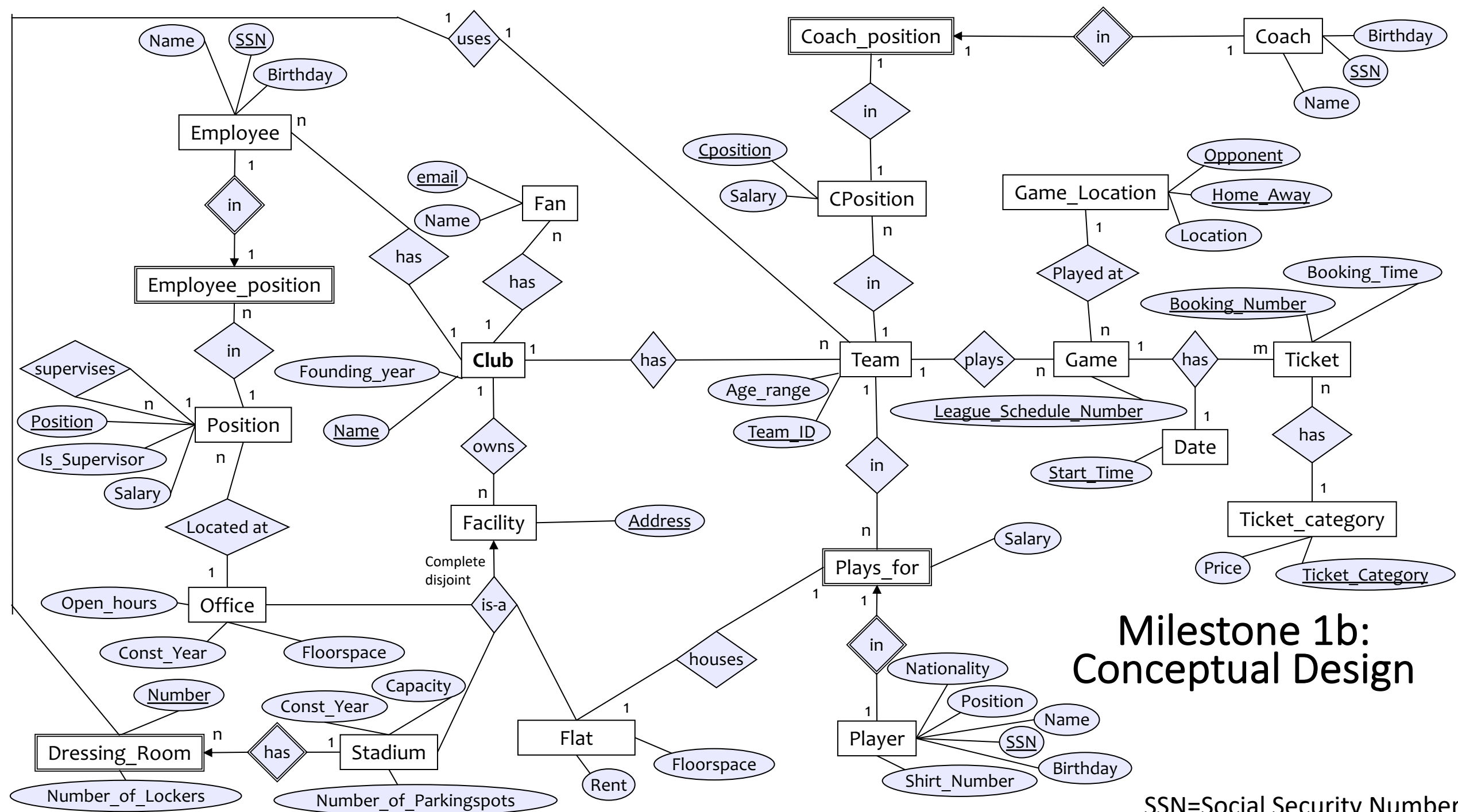
Players can be identified by their SSN. They also have a nationality, a birthday, a name and a salary as well as a shirt number and a position on the field. A player is part of only one team. Each team has several coaches in different positions, like head coach or assistant coach. A coach can be identified by SSN and has a name and a birthday. A coach only holds one coaching position.

The coaching positions are identified by a name and each has a certain salary. A number of positions exist per team and each position is uniquely matched with a team.

Each team plays games against teams from other clubs. Games are identified by their schedule number in the league schedule and we know the status (like home or away game). From the opponent and the status, the game's location (address) can be found. (The club's own stadium if it is a home game.)

For the pro-team's games, the club sells tickets, uniquely identified by a booking number. A ticket is valid on a specific day for the pro game and every other game played on that day by the different teams of the club. (Several different teams have games on the same day.) The booking time is also stored.

Tickets are sold for specific seats (categories) within the stadium. Each category has an identifier and a price.



Milestone 1b: Conceptual Design

SSN=Social Security Number

Milestone 2: Logical Design

Club(Name, Founding_year)

Fan(Email, Name, *Fan.Fanclub* \diamond *Club.Name*)

Office(Address, Construction_year, Floorspace, Open_hours, *Office.Owner* \diamond *Club.Name*)

Employees(SSN, Name, Birthday, *Employee.Employer* \diamond *Club.Name*)

EPosition(Eposition, Salary, *EPosition.Supervisor* \diamond *EPosition.Eposition*, Is_Supervisor, *EPosition.Workspace* \diamond *Office.Address*)

Employee_Position(*Employee_Position.SSN* \diamond *Employees.SSN*, *Employee_Position.Eposition* \diamond *EPosition.Eposition*)

Stadium(Address, Construction_year, Capacity, Number_of_parkingspots, *Stadium.Owner* \diamond *Club.Name*)

Dressingroom(*Dressingroom.Address* \diamond *Stadium.Address*, Number, Number_of_lockers)

Flat(Address, Rent, Floorspace, *Flat.Owner* \diamond *Club.Name*)

Player(SSN, Name, Birthday, Position, Nationality, Shirt_number)

Team(TeamID, Age_range, *Team.Owner* \diamond *Club.Name*, *Team.DRAddress* \diamond *Dressingroom.Address*, *Team.DRNumber* \diamond *Dressingroom.Number*)

Plays_for(*Plays_for.Player* \diamond *Player.SSN*, *Plays_for.Team* \diamond *Team.TeamID*, Salary, *Plays_for.Flat* \diamond *Flat.Address*)

Coach(SSN, Name, Birthday)

CPosition(Cposition, Salary, *CPosition.Team* \diamond *Team.TeamID*)

Coach_Position(*Coach_Position.Coach* \diamond *Coach.SSN*, *Coach_Position.Position* \diamond *CPosition.Cposition*)

Game_Location(Opponent, HomeAway, Location)

Game(Schedulenummer, StartTime, *Game.Team* \diamond *Team.TeamID*, *Game.Opponent* \diamond *Game_Location.Opponent*, *Game.HomeAway* \diamond *Game_Location.HomeAway*)

Ticket_Category(TCategory, Price)

Ticket(Booking_number, Booking_Time, *Ticket.Category* \diamond *Ticket_Category.TCategory*, *Ticket.Schedulenummer* \diamond *Game.Schedulenummer*, *Ticket.StartTime* \diamond *Game.StartTime*)

Milestone 4: Implementation

The application has three different pages. The first page is the index.php page. Here a fan can sign up for the fan club. The 'Schedule & Tickets' button in the top right corner allows fans to view the game schedule and buy tickets through clicking on 'Buy Ticket' for the individual games (New tuple added to Tickets table).

Employees can enter the 'For Employees' section by clicking on the second button which appears in the top right corner on the index.php page. Here, employees can interact with the database by inserting, updating, searching and deleting tuples from the Player, Employee, EPosition and Coach tables in the database.

If any constraints are violated (e.g. when trying to insert a tuple), the application will show the oracle error message on the following feedback page. Otherwise, a success message is displayed. However, I tried to guide the user input through dropdown selections and form properties in a way that should avoid as many errors as possible.

There were no significant problems in implementing the php application.