**Name:** Pierre-Alexandre Delgado-arevalo
**Student ID:** 923689985
**Github username:** TheRealPad

# Health Management System

| Checkpoint # | Date Submitted |
|---|---|
| Checkpoint I | 09/07/2023 |
| Checkpoint II | 10/03/2023 |
| Checkpoint III | 10/23/2023 |

# Table of content

# Project Description

Centralized access to a patient's health data is important, first and foremost for the patient himself/herself, but also for doctors, to help them in their decision-making. We are also seeing the emergence of more and more tools to retrieve our health data, such as connected watches, like the Apple watch, to retrieve heart rate, and the evolution of our technologies, enabling us to retrieve data more precisely, like the phone which retrieves our position and enables us to know how fast we are running, for example, or we are also seeing the emergence of applications which will enable users to track their body mass and record the number of calories they eat per day.

By using our database system to store and centralize your health data, this will enable you and your doctor to have everything in one place, quickly. What's more, it will allow you to control what the app records without going through a third party.

The application will not only store classic health data, but also data on sports activities and meals taken by the user.

What’s more, the application will be able to connect with experts in sport (coach) or in food and giving you a lots of data about each field, like sports event (marathon), sport activities (run 3 times per week), food events, what type of food eat, which device use to get your data…

Today, all software storage for health data (except Google) store only 2 of the 3 types of data (health + sport or health + meal). our database will store. We will store everything in one place.

Currently on the market, we can find some applications like Fitbit or the Nike training club, which store the data about the sport activities and some about the user, but no more. With our database system they could access the meal data from other applications.

## Use case

1. **Use case :** too much data - lose time
   **Actor :** Doctors, patient (Pierre)
   **Description :** Pierre is a patient for whom everything was going well, but from one day to the next everything changed: he had a stroke. As someone young enough to have had a stroke and in good health, the doctors want to test every possibility to prevent it happening again. The only worry is that all his health data isn't centralized in one place, so it takes a while to retrieve it all, and Pierre isn't the type to record his sporting activities to see if there's been any change.
   With a system like our health data centralization system, doctors would have immediate access to all Pierre's medical data, so they could understand as quickly as possible why a young, healthy man might have had a stroke.

2. **Use case :** too many application - lose time and desire to use
   **Actor :** Sporty man (Pierre)
   **Description :** Pierre, a 30-year-old man who likes to keep track of his sporting performance and is constantly looking for ways to improve, would like to be able to record each of his sporting activities, as well as each of his meals, so that he can see how many calories he eats per day, and adapt his needs according to his activity. But the problem is that for each of these needs, he'll need a different application which, on top of that, will take up space on his phone. With a database system like ours, he'd have no problem recording all his needs in a single application.

3. **Use case :** diet follow-up
   **Actor :** Former overweight person (Pierre)
   **Description :** Pierre, a former obese man with a bad skin condition, has managed to stop being obese. But his old demons are still there, trying to bring him back to his old self. He'd like to be able to monitor his dietary data to prevent his old tendencies from returning, which he could do with our database by tracking his diet over time.

4. **Use case :** Hypochondriac - data management
   **Actor :** Hypochondriac (Pierre)
   **Description :** Pierre is a hypochondriac, he knows it's beyond comprehension, but he needs to be able to monitor his health data regularly to be reassured. That's why he uses our database system, in which he has centralized all his information so that he can have it all in one place, easily and without any headaches.

5. **Use case :** Health data management
   **Actor :** User (Pierre)
   **Description :** Pierre is a user of our database system. But being old-school, he doesn't have all the tools to retrieve his health data, he prefers to do it the old-fashioned way, by hand, which is why he likes our system because it allows him to enter his data without needing to go through other tools, he knows exactly what he's entering and what's coming out, without any third-party tools.

6. **Use case :** user Authentication
   **Actor :** User (Pierre)

**Description :** This use case begins when a user when to access his data, he don't want that any body can access it, so he must use his password which will be encrypted by the system

7. **Use case :** Access data from other software
   **Actor :** Start up in meal recommandation
   **Description :** We have a start up creating an application to recommend meals to its user in function of their activities, but because it's a start up and it's not their main feature, they will not implement an application for connected watch to get sport data of their user, so with our database, they can access the sport data of their users from other applications

# Functional database requirements

1. User
   1.1. A user shall create only one account
   1.2. A user shall have at least one role
   1.3. A user shall access all of his data
   1.4. A user shall be linked to many sport activities
   1.5. A user shall be linked to one sport activity
   1.6. A user shall to be linked to many meals
   1.7. A user shall to be linked to one meal
   1.8. A user shall have many metrics health data
   1.9. A user shall link to many users to share sports data
   1.10. A user shall link to many users to share meals data
   1.11. A user shall link to many doctors
   1.12. Many users shall be linked to one doctor
   1.13. Many users shall be linked to many doctors
   1.14. Many users shall have the admin role
   1.15. A user shall be able to update his health data
   1.16. A user shall be able to update his meals
   1.17. A user shall be able to update his sport activities
   1.18. A user shall be a doctor, a patient or admin
2. Account
   2.1. An account shall be created by only one user
   2.2. An account shall have only one password
   2.3. An account shall have only one email
   2.4. An account shall be linked to zero or many actions
   2.5. An account shall be linked to one region
   2.6. An account shall be linked to zero or many event
   2.7. An account shall be linked to zero or many device
   2.8. An account shall be linked to zero or many training
   2.9. An account shall be linked to zero or many health activity
   2.10. An account shall be linked to zero or many food categories
   2.11. An account shall be linked to zero or many food expert
   2.12. An account shall be linked to zero or many coach
   2.13. An account shall be linked to zero or many food place
   2.14. An account shall be linked to zero or many sport place
   2.15. An account shall be linked to one and only one user health data
   2.16. An account shall be linked to one or many role
   2.17. An account shall be linked to one language
   2.18. An account shall be linked to one OS
3. Role
   3.1. A role shall be linked to many users
   3.2. A role shall be linked to zero or many actions
   3.3. Roles shall be linked to zero or many accounts

## 4. Patient

4.1. A patient shall be linked to many users

## 5. Doctors

5.1. A doctor shall be linked to many users

5.2. A doctor shall have the same right as a patient

5.3. A doctor shall have only one license number

5.4. A doctor shall have one doctor license number to be considered as a doctor

5.5. Many doctors shall be linked to one patient

5.6. Many doctors shall be linked to many patient

5.7. A doctor shall have all the right of a user

5.8. A doctor shall be linked to a doctor

5.9. A doctor shall be able to update user linked health data

5.10. A doctor shall have one speciality

5.11. A doctor shall have access of a user sport, health and meals as one entity

## 6. Admin

6.1. An admin shall be linked to many users

## 7. Sport activity

7.1. A sport activity shall be linked to one user

7.2. Many activities shall be linked to many users

7.3. Many activities shall be linked to one user

7.4. An activity shall be implemented by one user

7.5. An activity shall have only one title

7.6. An activity shall have only one description

7.7. An activity shall have only one duration

## 8. Meals

8.1. A meal shall be linked to one user

8.2. Many meals shall be linked to one user

8.3. Many meals shall be linked to many users

8.4. A meal shall be implemented by one user

8.5. A meal shall have only one title

8.6. A meal shall have only one description

8.7. A meal shall have only one caloric value

8.8. A meal shall have only one date

## 9. Health metrics data

9.1. A Health metrics data shall be linked to one user

9.2. A health data shall have only on height

9.3. A health data shall have only one weight

9.4. A health data shall have only one blood pressure

9.5. A health data shall have only one breathing rate

9.6. A health data shall have only one current heart rate

9.7. A health data shall be linked to one heart rates

## 10. Doctor license number

10.1. A doctor license number shall be linked to only one doctor

## 11. Social number

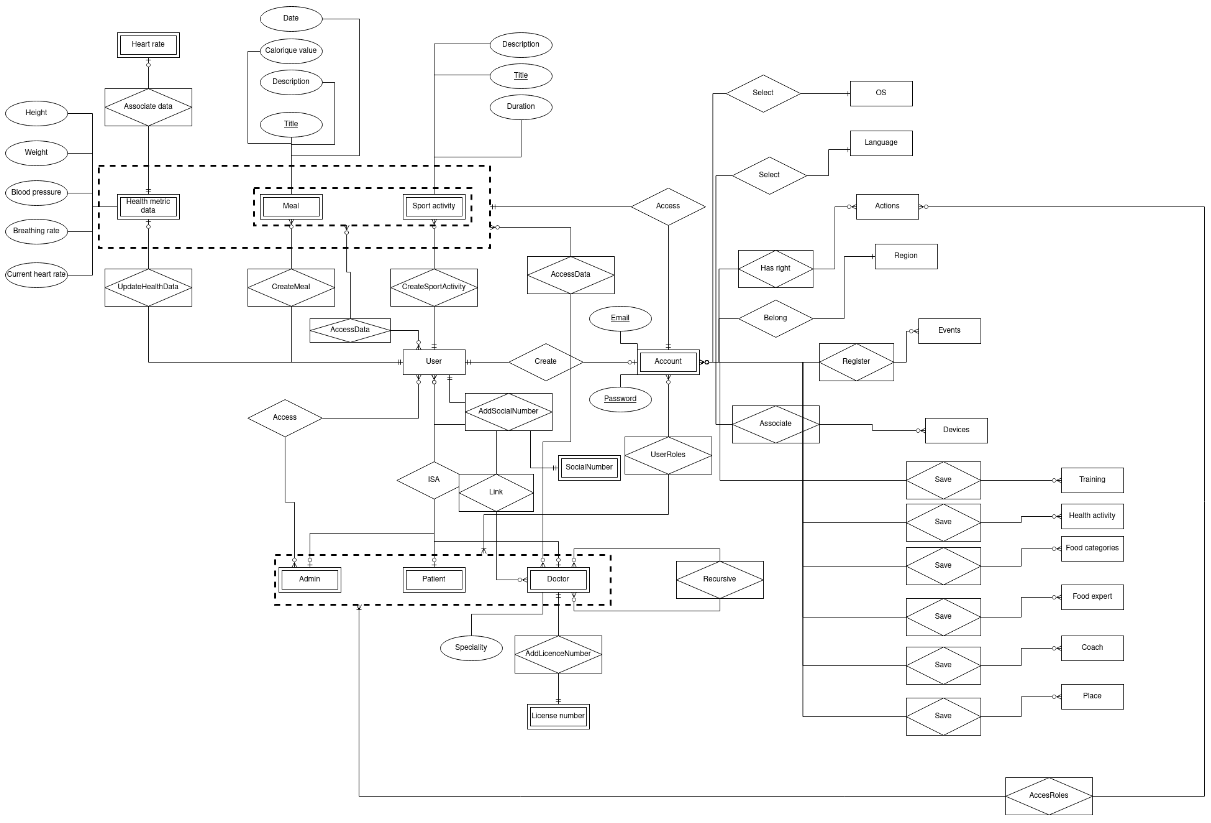
11.1. A social number shall be linked to only one user

12. Heart rates
    12.1. heart rate shall be linked to only one health metric data
13. Action
    13.1. an action shall be linked to zero or many account
    13.2. An action shall be linked to one or many roles
14. Region
    14.1. A region shall be linked to zero or many account
15. User data
    15.1. User data shall be linked to one and only one user
16. Event
    16.1. An event shall be linked to zero or many account
17. Device
    17.1. A device shall be linked to one or many account
18. Training
    18.1. A training shall be linked to zero or many account
19. Health activity
    19.1. A health activity shall be linked to zero or many account
20. Food categories
    20.1. A food category shall be linked to zero or many account
21. Food expert
    21.1. A food expert shall be linked to zero or many account
22. Coach
    22.1. A coach shall be linked to zero or many account
23. Place
    23.1. A food place shall be linked to zero or many account
24. Language
    24.1. A language shall be linked to zero or many account
25. OS
    25.1. An OS shall be linked to zero or many account

# Non-functional database requirements

1. Performance
   1.1. The database system shall support concurrent transactions.
2. Storage
   2.1. The database system shall assign 10 MB of memory per table.
   2.2. The database system should support persistent storage
3. Security
   3.1. Only encrypted passwords shall be supported by the database system
   3.2. All the values inserted into the database shall be consistent with the attribute's data type and domain.
   3.3. The database shall be automatically backed up everyday at 11:59 pm.
   3.4. Only the class instructor and the database developer can access the database
4. Compatibility
   4.1. The database shall work on MacOS and Linux systems
5. Availability
   5.1. The system should be available 99.9% of the time

# Entity Relationship Diagram (ERD)



# Entity description

1. User (Strong)
    a. user_id: key, numeric
    b. name: composite, alphanumeric
    c. email: alphanumeric, unique
    d. password: alphanumeric
    e. dob: multivalue, timestamp
    f. health_metric_data_id: numeric
    g. role_id: numeric
    h. account_id: numeric
2. Patient (Weak)
    a. patient_id: key, numeric
    b. user_id: numeric
3. Doctor (Weak)
    a. doctor_id: key, numeric
    b. user_id: numeric
    c. license_id: numeric
    d. speciality: multivalue, alphanumeric
4. Admin (weak)
    a. admin_id: key, numeric
    b. user_id: numeric
5. License number (Weak)
    a. license_id: key, numeric
    b. value: alphanumeric, unique
6. social number (Weak)
    a. social_number_id: key, numeric
    b. value: alphanumeric, unique
    c. user: numeric
7. Account (Weak)
    a. account_id: key, numeric
    b. user_id: numeric
    c. role: numeric
    d. region: numeric
    e. user_health_data: numeric
    f. os: numeric
    g. language: numeric
8. Meal (Weak)
    a. meal_id: key, numeric
    b. title: alphanumeric
    c. description: alphanumeric
    d. caloric_value: numeric
    e. date: multivalue, timestamp
    f. user_id: numeric
9. Health data (Weak)
    a. health_id: key, numeric

   b. height: numeric
   c. weight: numeric
   d. blood pressure: numeric
   e. breathing rate: numeric
   f. current_heart_rate: numeric

10. Sport activity (Weak)
    a. sport_id: key, numeric
    b. title: numeric
    c. description: numeric
    d. duration: numeric
    e. user_id: numeric
11. Heart rate (Weak)
    a. heart_id: key, numeric
    b. heart_rate: numeric
    c. date: multivalue, timestamp
    d. health_data_id: numeric
12. Role (Strong)
    a. role_id: key, numeric
    b. name: alphanumeric
    c. description: alphanumeric
    d. admin: number
    e. patient: numeric
    f. doctor: numeric
13. Action (strong)
    a. action_id: key, numeric
    b. title: alphanumeric
    c. description: alphanumeric
14. Region (strong)
    a. region_id: key, numeric
    b. name: alphanumeric
    c. code_area: alphanumeric
15. UserLinkToUser (strong)
    a. user_link_to_user_id: key, numeric
    b. userA: numeric
    c. userB: numeric
16. Event (strong):
    a. event_id: key, numeric
    b. title: alphanumeric
    c. description: alphanumeric
    d. date: multivalue, timestamp
    e. place: alphanumeric
17. Device (strong):
    a. device_id: key, numeric
    b. name: alphanumeric
    c. description: alphanumeric
18. Trainings (strong)
    a. training_id: key, numeric
    b. duration: numeric

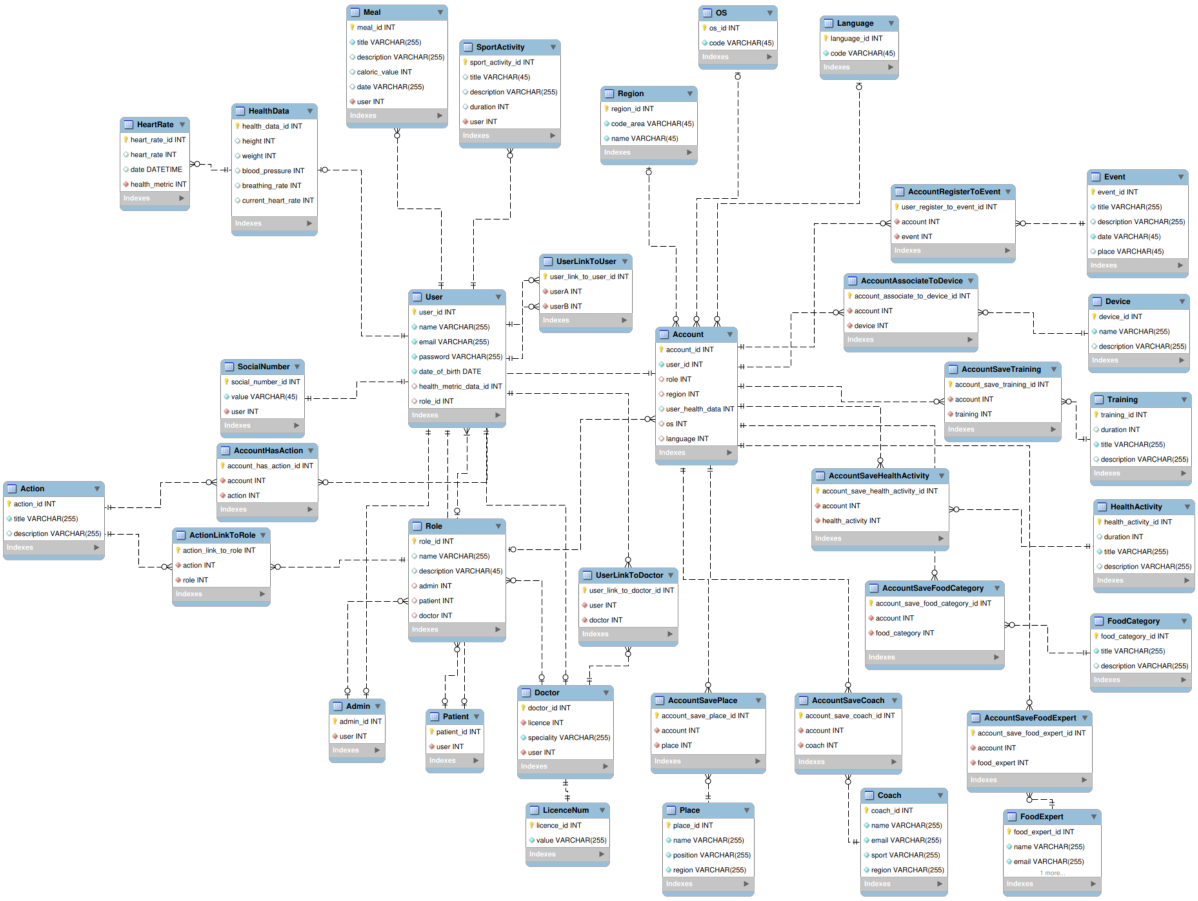
    c. title: alphanumeric
    d. description: alphanumeric

19. Health activity (strong)
    a. health_activity_id: key, numeric
    b. duration: numeric
    c. title: alphanumeric
    d. description: alphanumeric
20. Food categories (strong)
    a. food_categories_id: key, numeric
    b. title: alphanumeric
    c. description: alphanumeric
21. Food expert (strong)
    a. food_expert_id: key, numeric
    b. name: composite, alphanumeric
    c. email: alphanumeric
    d. region: alphanumeric
22. Coach (strong)
    a. coach_id: key, numeric
    b. name: composite, alphanumeric
    c. email: alphanumeric
    d. sport: alphanumeric
    e. region: alphanumeric
23. Place (strong)
    a. place_id: key, numeric
    b. name: alphanumeric
    c. position: alphanumeric
    d. region: alphanumeric
    e. Description: alphanumeric
24. UserLinkToDoctor (strong)
    a. user_link_to_doctor_id: key, numeric
    b. user: numeric
    c. doctor: numeric
25. Language (strong):
    a. language_id: key, numeric
    b. code: alphanumeric
26. OS (strong):
    a. os_id: key, numeric
    b. code: alphanumeric
27. ActionLinkToRole (weak):
    a. action_link_to_role_id: key, numeric
    b. action: numeric
    c. role: numeric
28. AccountHasAction (weak):
    a. accound_has_action_id: key, numeric
    b. account: numeric
    c. action: numeric
29. AccountSavePlace (weak):
    a. account_save_place_id: key, numeric
    b. account: numeric

    c. place: numeric

30. AccountSaveCoach (weak):
    a. account_save_coach_id: key, numeric
    b. account: numeric
    c. coach: numeric
31. AccountSaveFoodExpert (weak):
    a. account_save_food_expert_id: key, numeric
    b. account: numeric
    c. food_expert: numeric
32. AccountSaveFoodCategory (weak):
    a. account_save_food_category_id: key, numeric
    b. account: numeric
    c. food_category: numeric
33. AccountSaveHealthActivity (weak):
    a. account_save_health_activity_id: key, numeric
    b. account: numeric
    c. health_activity: numeric
34. AccountSaveTraining (weak):
    a. account_save_training_id: key, numeric
    b. account: numeric
    c. training: numeric
35. AccountAssociateToDevice (weak):
    a. account_associate_device_id: key, numeric
    b. account: numeric
    c. device: numeric
36. AccountRegisterToEvent (weak):
    a. account_register_event_id: key, numeric
    b. account: numeric
    c. event: numeric

# Entity Establishment Relationship Diagram (EER)



# Constraints description

| Table | FK | ON DELETE | ON UPDATE | Comment |
|---|---|---|---|---|
| HeartRate | HealthData | ON CASCADE | ON CASCADE | If the health data is delete, the heart rate too |
| User | HealthData | SET NULL | ON CASCADE | If the health data is deleted, The user health data will be set to null |
| User | SocialNumber | ON CASCADE | ON CASCADE | A user can't not have a social number |
| User | Role | NO ACTION | ON CASCADE | If the role is deleted, the user will have no roles |
| Meal | User | ON CASCADE | ON CASCADE | If the user is deleted, the meal too |
| SportActivity | User | ON CASCADE | ON CASCADE | If the user is deleted, the sport activity too |
| Role | Patient | SET NULL | ON CASCADE | if a patient role in deleted, the role is net to null |
| Role | Admin | SET NULL | ON CASCADE | if a admin role in deleted, the role is net to null |
| Role | Doctor | SET NULL | ON CASCADE | if a doctor role in deleted, the role is net to null |
| Doctor | LicenceNum | ON CASCADE | ON CASCADE | if a social number is deleted, the doctor role is deleted too |
| UserLinkDoctor | User | ON CASCADE | ON CASCADE | if the user is deleted, the link in deleted too |
| UserLinkDoctor | Doctor | ON CASCADE | ON CASCADE | if the doctor is deleted, the link in deleted too |
| UserLinkToUser | User | ON CASCADE | ON CASCADE | if one user is deleted, the link too |
| Account | Region | SET NULL | ON CASCADE | if a region is deleted, it's set to null in the account |
| Account | Role | SET NULL | ON CASCADE | if a role is deleted, it's set to null in the account |
| Account | OS | SET NULL | ON CASCADE | if a os is deleted, it's set to null in the account |

| | | | | |
|---|---|---|---|---|
| Account | Language | SET NULL | ON CASCADE | if a language is deleted, it’s set to null in the account |
| AccountHasAction | Account | ON CASCADE | ON CASCADE | If one element is deleted, the link is deleted too |
| AccountHasAction | Action | ON CASCADE | ON CASCADE | If one element is deleted, the link is deleted too |
| ActionLinkToRole | Action | ON CASCADE | ON CASCADE | If one element is deleted, the link is deleted too |
| ActionLinkToRole | Role | ON CASCADE | ON CASCADE | If one element is deleted, the link is deleted too |
| AccountSavePlace | Account | ON CASCADE | ON CASCADE | If one element is deleted, the link is deleted too |
| AccountSavePlace | Place | ON CASCADE | ON CASCADE | If one element is deleted, the link is deleted too |
| AccountSaveCoach | Account | ON CASCADE | ON CASCADE | If one element is deleted, the link is deleted too |
| AccountSaveCoach | Coach | ON CASCADE | ON CASCADE | If one element is deleted, the link is deleted too |
| AccountSaveFoodExpert | Account | ON CASCADE | ON CASCADE | If one element is deleted, the link is deleted too |
| AccountSaveFoodExpert | FoodExpert | ON CASCADE | ON CASCADE | If one element is deleted, the link is deleted too |
| AccountSaveFoodCategory | Account | ON CASCADE | ON CASCADE | If one element is deleted, the link is deleted too |
| AccountSaveFoodCategory | FoodCategory | ON CASCADE | ON CASCADE | If one element is deleted, the link is deleted too |
| AccountSaveHealthActivity | Account | ON CASCADE | ON CASCADE | If one element is deleted, the link is deleted too |
| AccountSaveHealthActivity | HealthActivity | ON CASCADE | ON CASCADE | If one element is deleted, the link is deleted too |
| AccountSaveTraining | Account | ON CASCADE | ON CASCADE | If one element is deleted, the link is deleted too |
| AccountSaveTraining | Training | ON CASCADE | ON CASCADE | If one element is deleted, the link is deleted too |
| AccountAssociateToDevice | Account | ON CASCADE | ON CASCADE | If one element is deleted, the link is deleted too |

| | | | | |
|---|---|---|---|---|
| AccountAssociateToDevice | Device | ON CASCADE | ON CASCADE | If one element is deleted, the link is deleted too |
| AccountRegisterToEvent | Account | ON CASCADE | ON CASCADE | If one element is deleted, the link is deleted too |
| AccountRegisterToEvent | Event | ON CASCADE | ON CASCADE | If one element is deleted, the link is deleted too |
| Admin | User | ON CASCADE | ON CASCADE | If the user is deleted, this role too |
| Patient | User | ON CASCADE | ON CASCADE | If the user is deleted, this role too |
| Doctor | User | ON CASCADE | ON CASCADE | If the user is deleted, this role too |