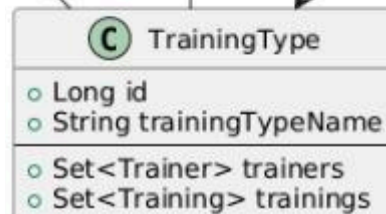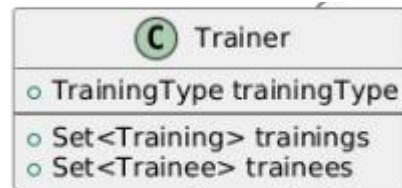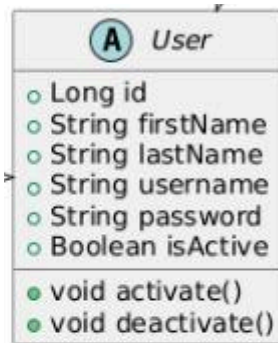# WORKSHOP-2

Estudiantes:

Adrian Yebid Rincon
Alirio Hernando Martinez Barreto
Andres Felipe Ramirez Fajardo
Juan Manuel Torres Leon
Manuel Santiago Mori Ardila
Pablo Felipe Sandoval Menjura

Facultad de Ingeniería de Sistemas e Industrial
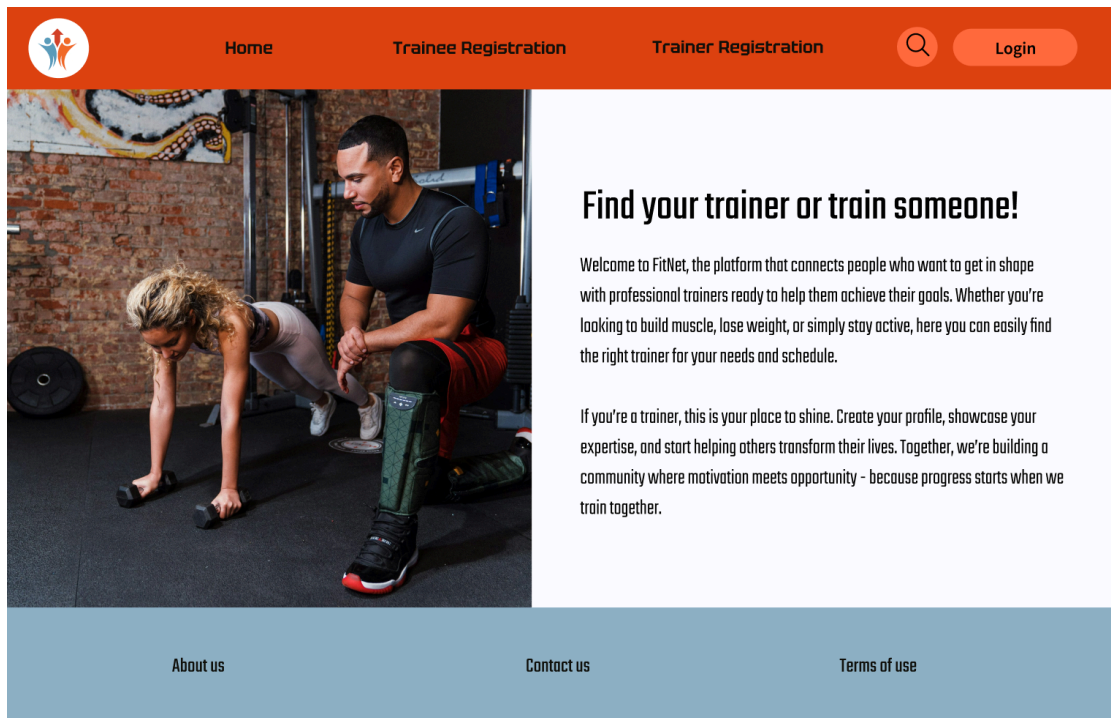**Ingeniería de Software II**
2025
Bogotá, Colombia

1. CRC Cards

| Class	Responsibilities	Collaborators
User (abstract)	- Represents a generic system user. - Stores basic information such as first name, last name, username, password, and active status. - Provides methods to activate or deactivate the user account.	Trainee, Trainer (subclasses)
Trainee	- Represents a gym member receiving training sessions. - Stores personal details such as date of birth and address. - Maintains a list of associated trainers.- Keeps track of all attended training sessions.	Training, Trainer
Trainer	- Represents a gym trainer or coach.- Manages a list of trainees being trained.- Keeps track of all training sessions conducted. - Linked to a specific training type.	Trainee, Training, TrainingType
Training	- Represents an individual training session.- Stores session details such as name, date, and duration. - Connects a trainee, a trainer, and a training type.	Trainee, Trainer, TrainingType
TrainingType	- Represents a category or type of training (e.g., Strength, Cardio, HIIT). - Keeps a list of trainers specialized in this training type. - Keeps a list of training sessions belonging to this type.	Trainer, Training

A *User*
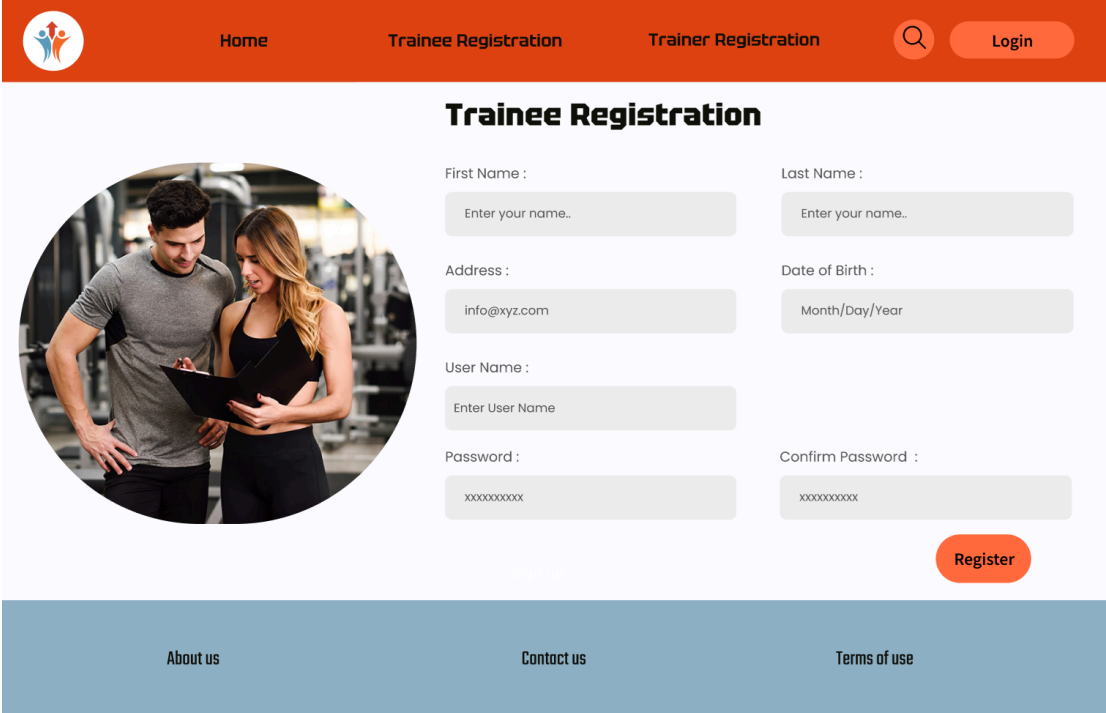
- Long id
- String firstName
- String lastName
- String username
- String password
- Boolean isActive

- void activate()
- void deactivate()

C Trainer

- TrainingType trainingType

- Set<Training> trainings
- Set<Trainee> trainees

C Trainee

- LocalDate dateOfBirth
- String address

- Set<Training> trainings
- Set<Trainer> trainers

C Training

- Long id
- String trainingName
- LocalDate trainingDate
- int trainingDuration

- Trainee trainee
- Trainer trainer
- TrainingType trainingType

C TrainingType

- Long id
- String trainingTypeName

- Set<Trainer> trainers
- Set<Training> trainings

2. Mockups

- Home page

The Home Page welcomes users and explains what the app does. New users can read about how the platform connects trainees who want to get fit with trainers who can guide them. The page shows links to register as a trainee or as a trainer and also has a login button to enter the system. At the bottom, there are links for "About us," "Contact us," and "Terms of use." It is the main introduction to the app.

- Login Form
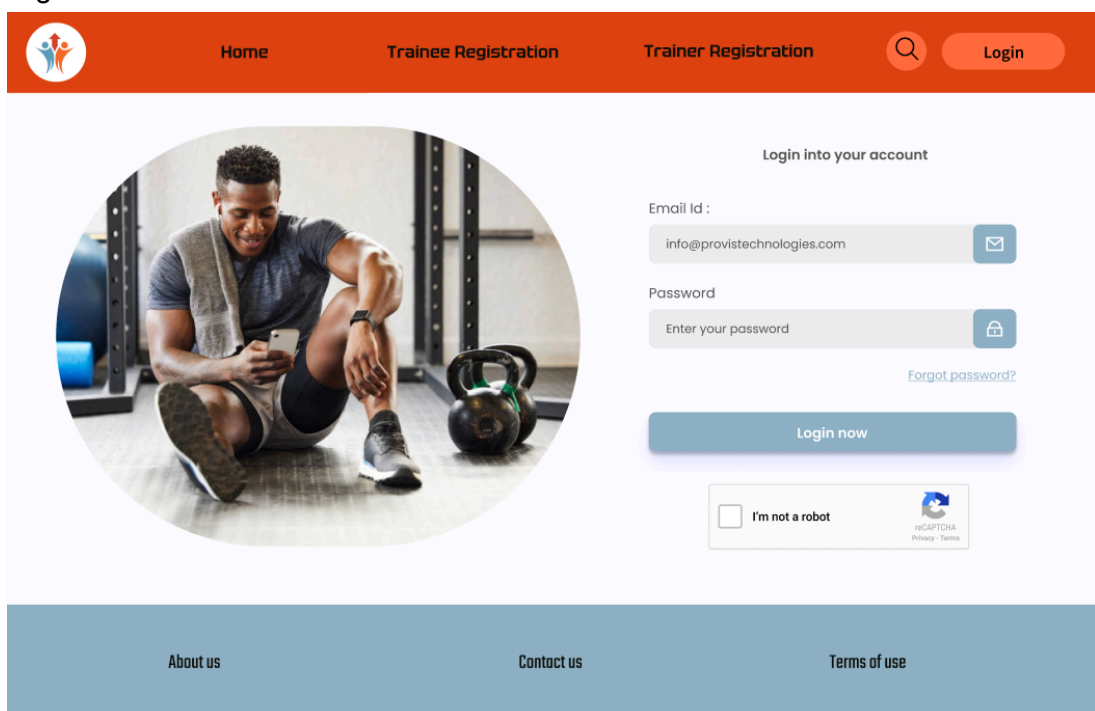
The Login Form screen allows both trainees and trainers to enter their accounts. Users type their email and password to access the platform safely. There is also a "Forgot password?" link to recover access if needed, and a CAPTCHA to verify that the user is not a robot. When users click "Login now," the system checks their information and opens their personal profile. This screen protects user data and gives secure entry to the application.
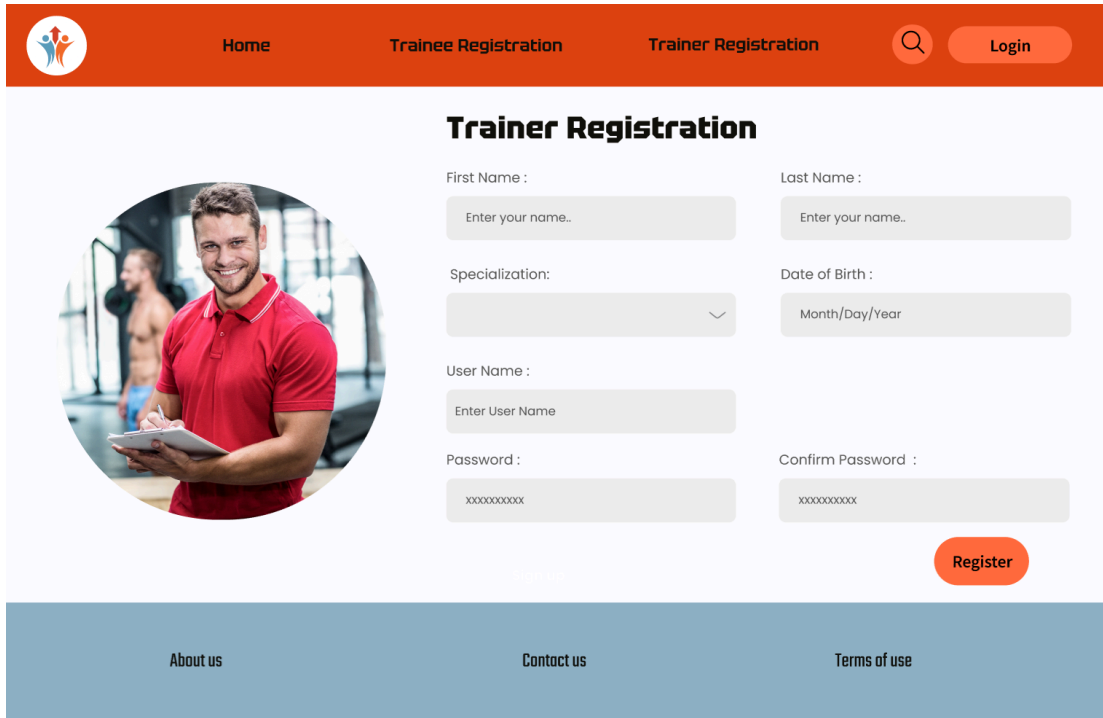
- Trainee registration screen

The Trainee Registration screen lets new users who want to train create their own account. Here, they fill in personal information like first name, last name, address, and date of birth. They also create a username and password to log in later. When they finish, they click the "Register" button to send their data and become part of the platform. This screen helps identify each trainee and saves their information safely in the system.
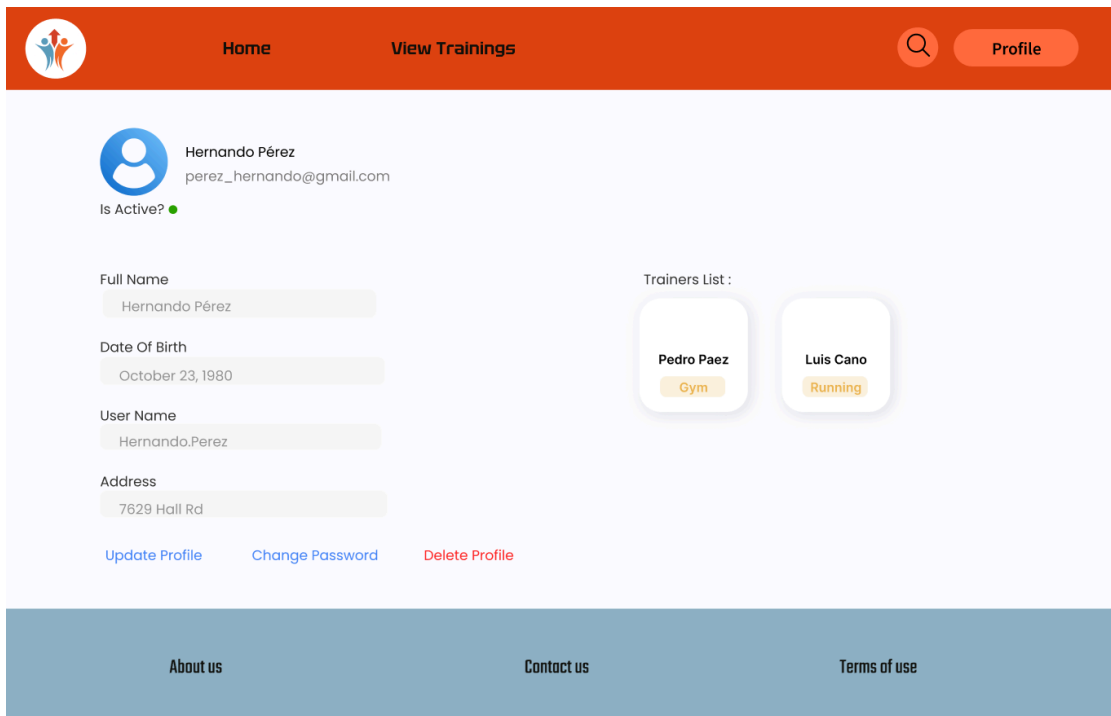
● Trainer registration screen

The Trainer Registration screen is for new trainers who want to join the platform. Trainers fill in their first and last name, date of birth, and choose their specialization, such as running or yoga. They also create a username and password to log in later. After entering all the information, they press "Register" to complete their account. This screen helps collect trainer details so they can start offering training sessions to trainees.

● Trainee profile screen

The Trainee Profile screen shows the personal information of a trainee who is already registered. It displays details like the trainee's full name, date of birth, username, address, and if their account is active. On the right, the trainee can see the list of their trainers and each trainer's specialization. The trainee can also update their profile, change their password, or delete their account using the buttons below. This screen helps trainees manage their profile and trainer connections easily.

- Trainer profile screen

The Trainer Profile screen shows the personal information of a trainer who is registered in the system. It displays details like the trainer's full name, specialization, username, and if their account is active or inactive. On the right side, the trainer can see the list of trainees they are currently training, with each trainee's name and specialization. The trainer can also update their profile, change their password, or delete their account. This screen helps trainers manage their professional information and relationships with trainees.

- Add Training for Trainee


The Add Training for Trainee screen lets a trainer create a new exercise or training session for one of their trainees. The trainer can select the trainee's name, choose the type of exercise (for example, by time, reps, or distance), and pick which muscle group the workout focuses on. They can also add details about sets, weight, and duration. After entering everything, the trainer clicks "Create" to save the exercise. This screen helps trainers organize and assign personalized workouts to each trainee.
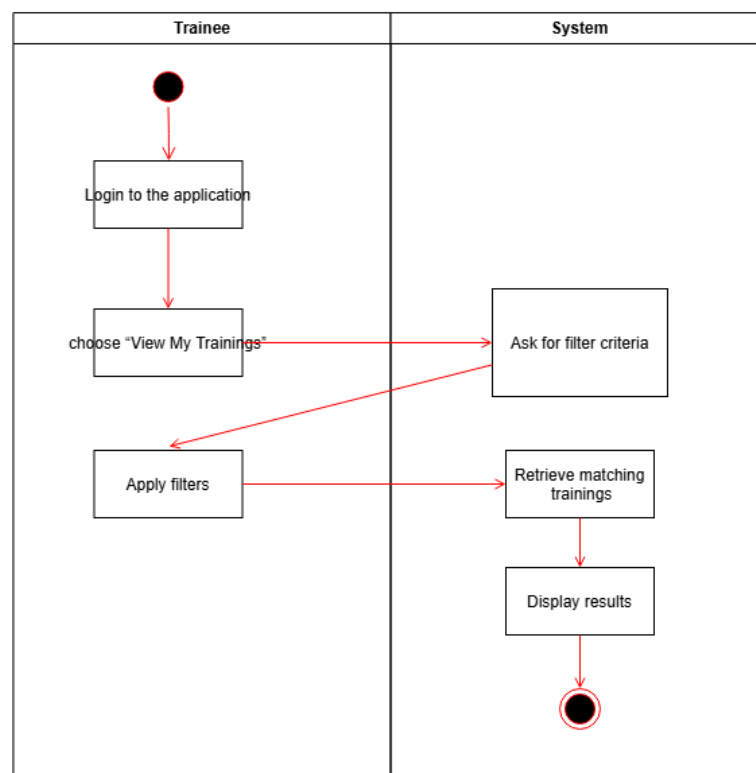
Herramienta usada: https://www.figma.com/

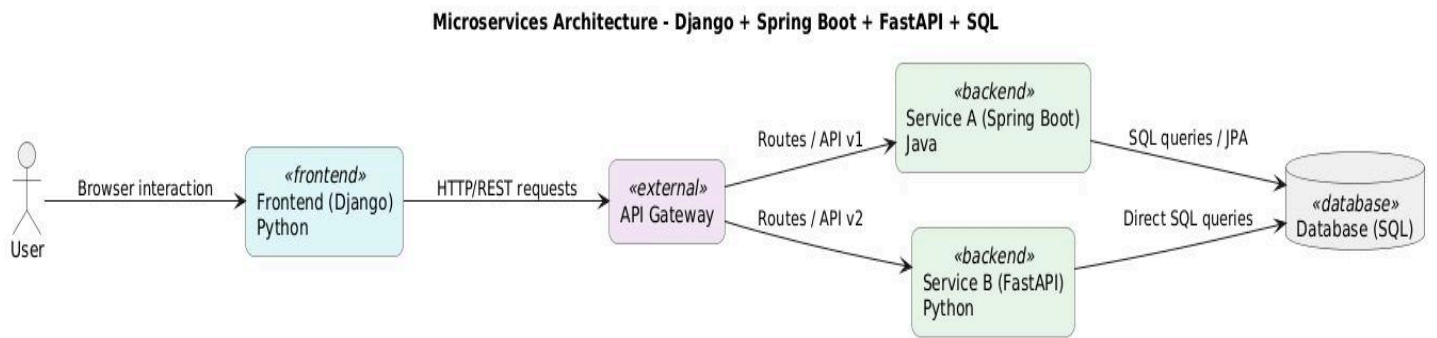## 3. Business Model Processes

Trainee Logs Training Activities
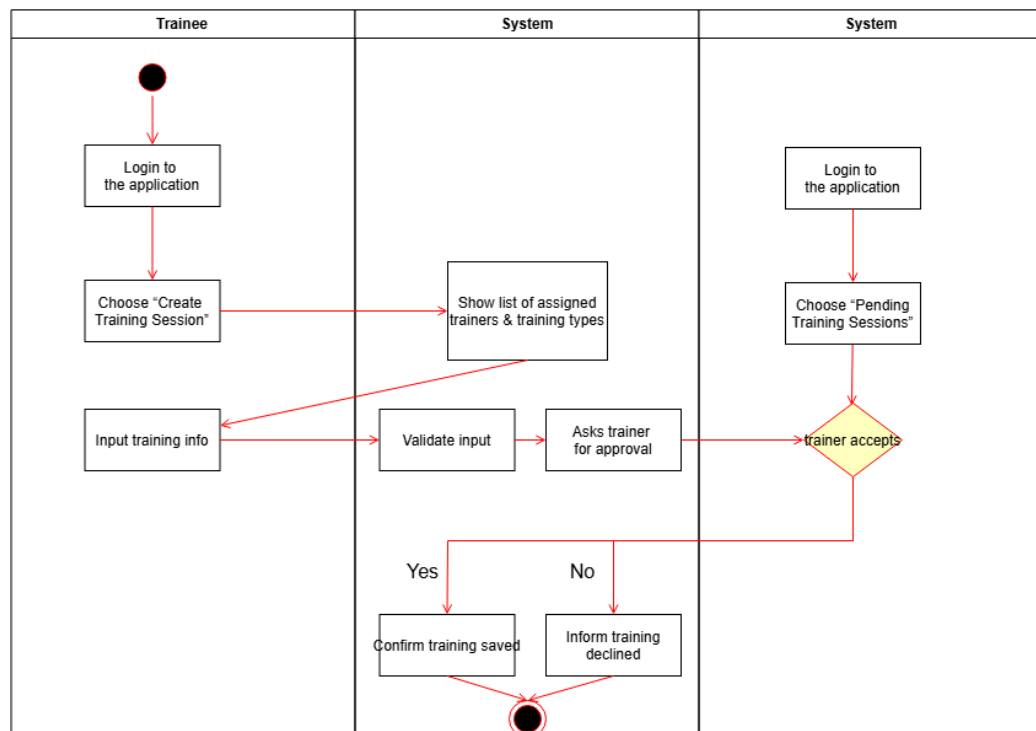
Trainee Views Training Activities

This process allows trainees to record their workout sessions and later review them.

4. Architecture Diagram

Microservices Architecture - Django + Spring Boot + FastAPI + SQL

User
Browser interaction →
«frontend»
Frontend (Django)
Python
HTTP/REST requests →
«external»
API Gateway

Routes / API v1 →
«backend»
Service A (Spring Boot)
Java
SQL queries / JPA →

Routes / API v2 →
«backend»
Service B (FastAPI)
Python
Direct SQL queries →

«database»
Database (SQL)

5. Class Diagram
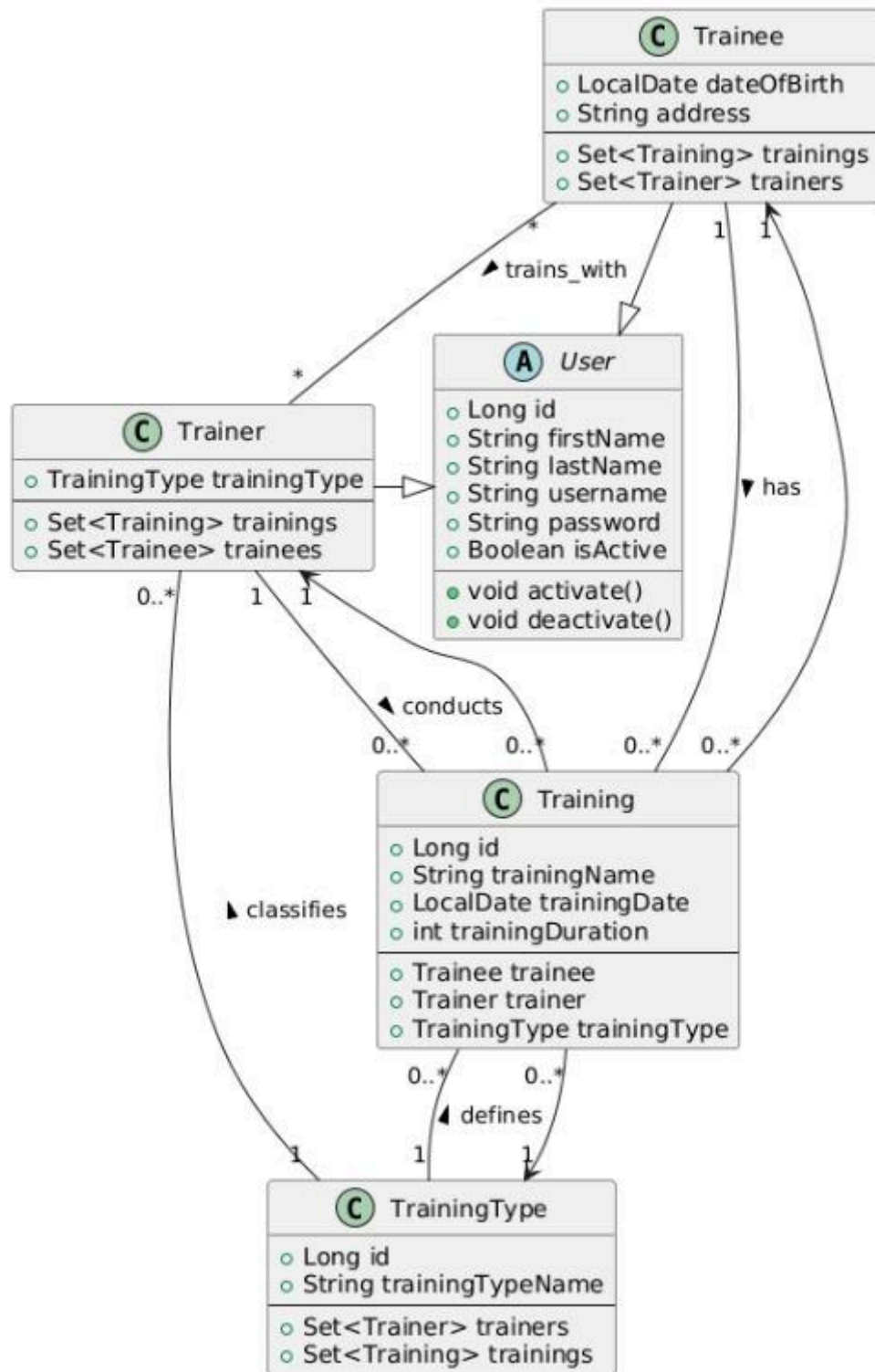
6. Relational Database Model