

**DOCUMENTATIE**

Saca Victor-Valentin

Tabulac Daria-Mara

Mihaila Claudiu-Iulian

Inginerie software

1. Descrierea proiectului

Gymhelp este un program care ajuta clientii sa obtina rezultate foarte bune in vederea mentinerii, scaderii sau cresterii masei corporale. Programul se ridica la nivelul asteptarilor clientilor nostrii deoarece acesta tinteste spre dorinta acestora, indiferend daca doresc sa aibe un trup mai fit, plinut sau slim.

Limbajele de programare folosite pentru acest proiect sunt Java, Javascript, Html si SQL.

Fiecare program dintre acestea a avut un rol consecvent in alcatuirea produsului final si intemeierea unei baze solide pentru functionarea acestuia.

Html si Javascript alcatuiesc fatada prezenta pe website, o interfata prietenoasa, intuitiva

care ajuta utilizatorul sa foloseasca site-ul pus la dispozitie. Limbajul Java a fost ales pentru a crea legaturile dintre partea de front-end si cea de back-end, asigurand in acelasi timp si securitatea clientului in timpul navigarii pe pagina. Datele sunt preluate din baza de date pusa la dispozitie, facuta cu MySQL.

Packages:

Services:

Trainee/Trainer Service responsabil pentru a gestiona logica de afaceri legată de entitatea Trainee. Aceasta interacționează cu un repository (TraineeRepository) pentru a accesa datele stocate în baza de date și oferă diverse metode pentru a gestiona informațiile despre entitățile de tip Trainee. Clasa oferă o abstracție pentru operațiunile comune cu entitatea Trainee, cum ar fi gestionarea utilizatorilor în funcție de ID, username, sau autentificare. Este utilizată de alte componente ale aplicației (de exemplu, controlere) pentru a interacționa cu datele trainee-urilor într-un mod structurat și organizat.

Repository:

TraineeRepository este un repository Spring Data JPA, care gestionează operațiunile de acces la baza de date pentru entitatea Trainee. Aceasta extinde interfața JpaRepository, ceea ce îi oferă automat o serie de metode predefinite pentru manipularea entităților (cum ar fi salvare, ștergere, căutare etc.).

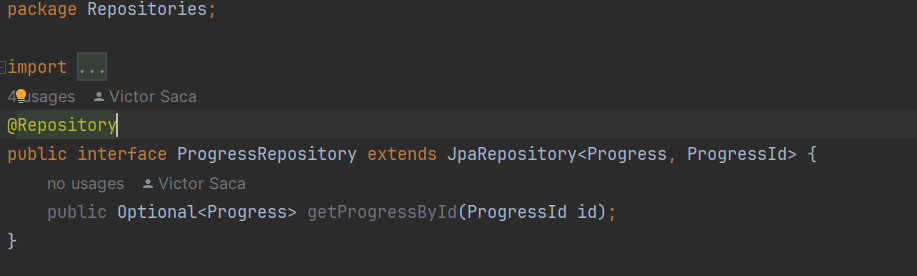
Models

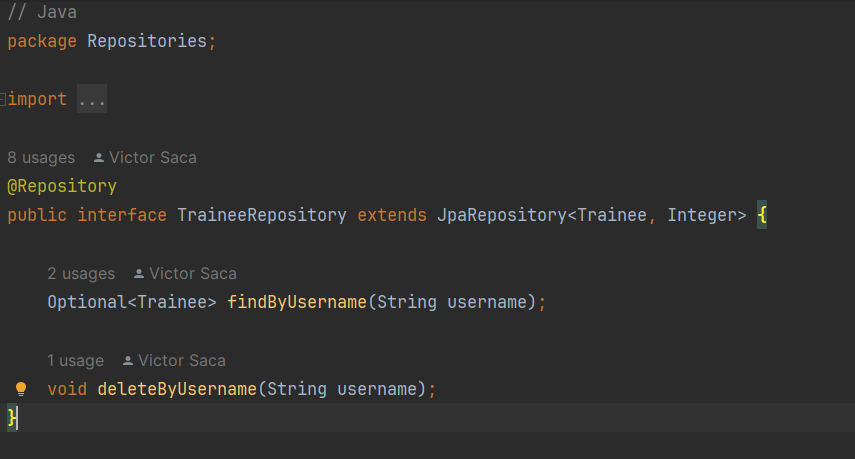
Clasa Trainee este baza pentru gestionarea datelor despre utilizatorii aplicației (trainees) în cadrul unei arhitecturi MVC (Model-View-Controller). Acesta este "Modelul" care definește datele și regulile acestora, fiind parte din stratul de persistență.

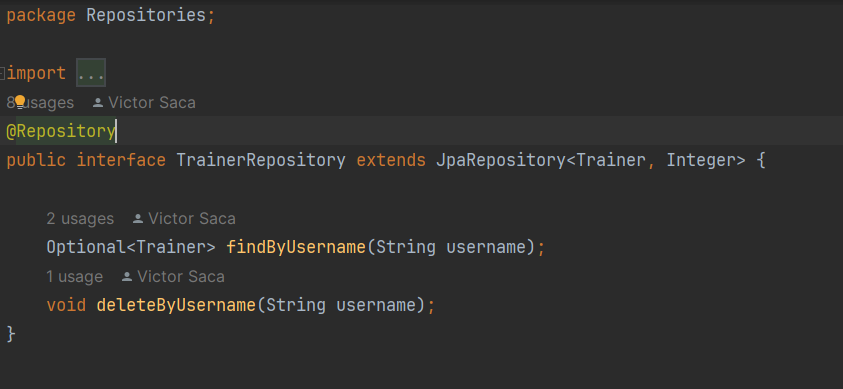
1. Design pattern-uri

Repository design pattern:

Am implementat repository design pattern, in care nu se mai fac interogari SQL pentru baza de date, ci sunt tratate prin interfete repositories, care foloseste nume suggestive pentru operatiile pe baza de date.







Un alt design pattern folosit este singleton, in care constructorii din clasele Service sunt private, si urmau sa fie instantiate o data cu metoda static.

Daca o metoda pentru baza de date nu se afla in repository, aceasta va fi specificata, si implementata in clasele de service.

Clasele de service reprezinta metodele care vor fi folosite din repository, si daca trebuie, si implementate.

Ultimul design pattern folosit este design patternul structural în dezvoltarea web, care se referă la organizarea și aranjarea componentelor unei aplicații astfel încât să fie ușor de înțeles, utilizat și întreținut. Acesta implică separarea logică a diferitelor părți ale interfeței utilizatorului și utilizarea unor structuri bine definite pentru a crea un layout clar și ordonat.

HTML este utilizat pentru a defini structura paginii web:

Titlul, butoanele, etichetele și câmpurile de input sunt organizate logic în elemente și clase.

CSS (legat prin login.css) este utilizat pentru a controla stilizarea:

Aspectul vizual, cum ar fi culorile, marginile sau dimensiunile elementelor, este gestionat separat de conținutul HTML.

JavaScript (legat prin login.js) este utilizat pentru comportamentul dinamic:

De exemplu, acțiunile legate de apăsarea butoanelor sau interacțiunea cu utilizatorul.

Această separare face codul mai ușor de întreținut și de modificat.

1. Diagrama de usecase

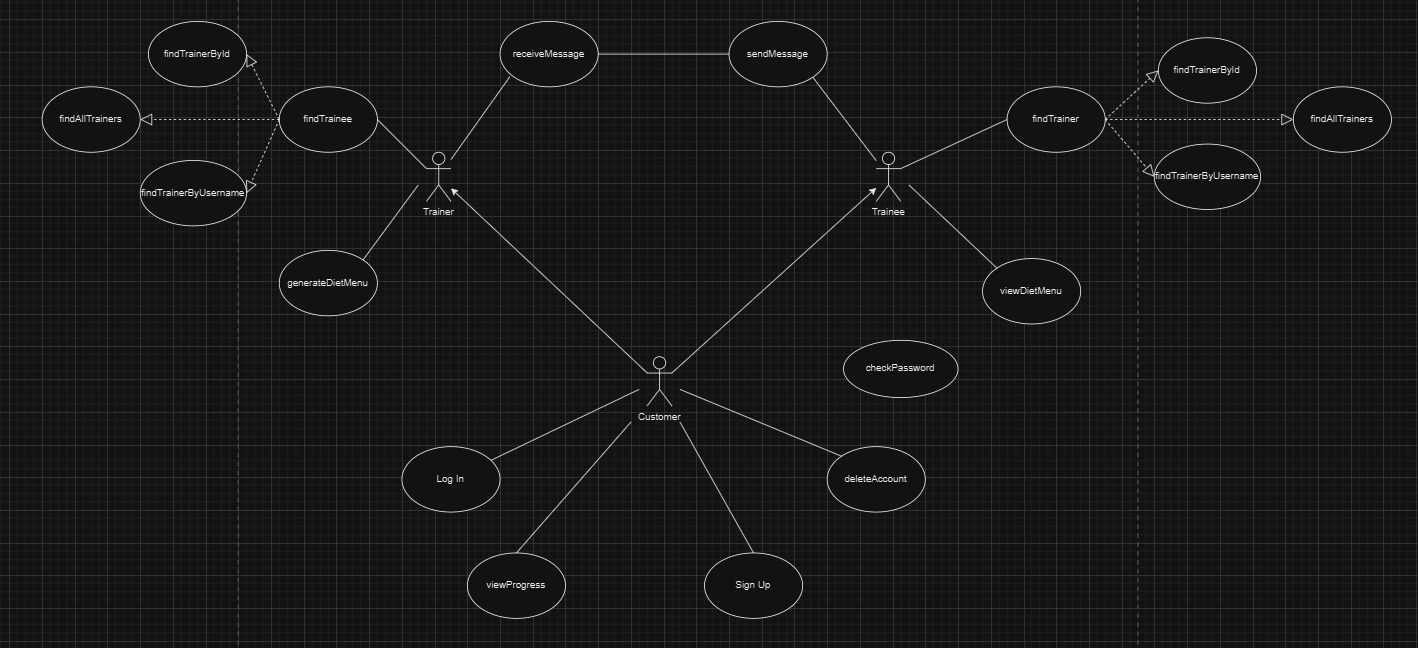
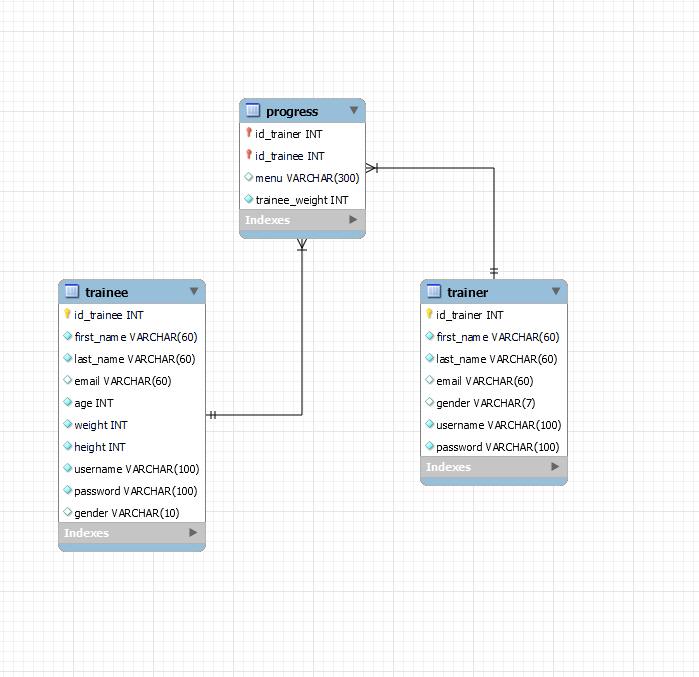
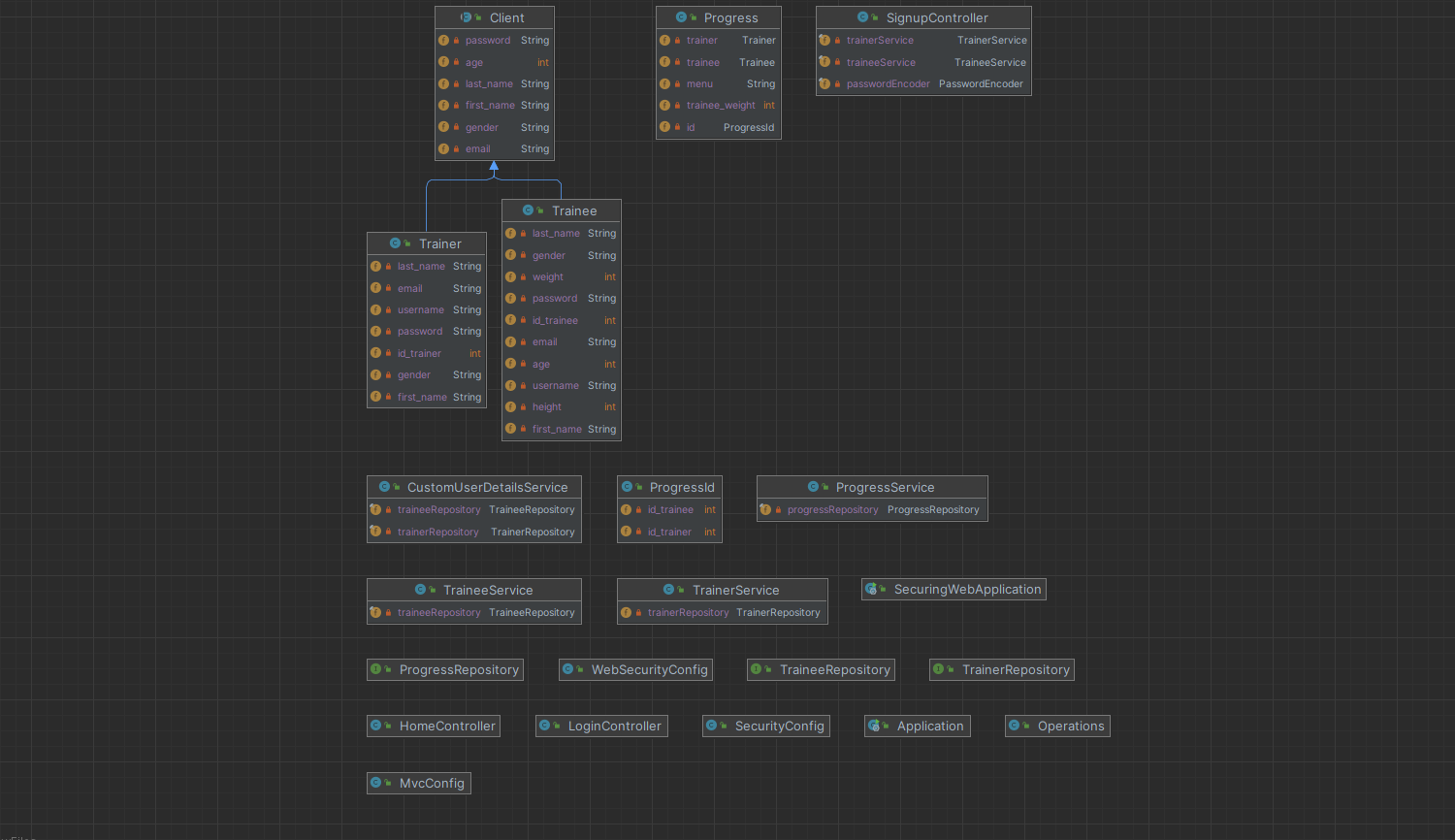


Diagrama ER a bazei de date:



Diagramele de clase:

Services si Repositories



Controller si security:

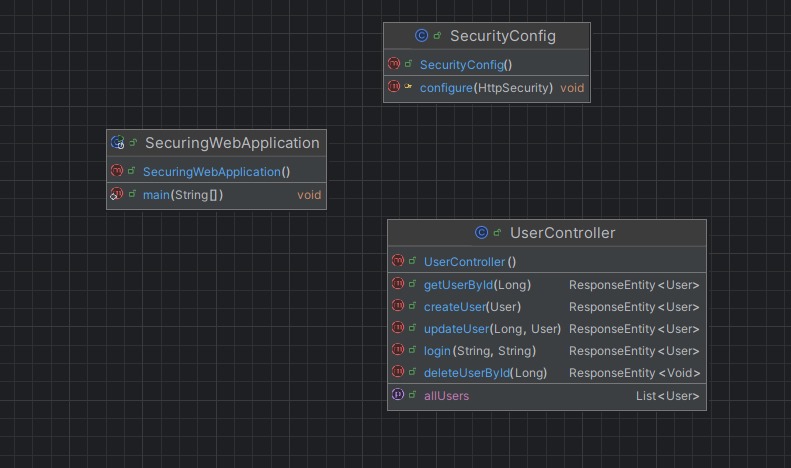


Diagrama secventiala: (Daria)

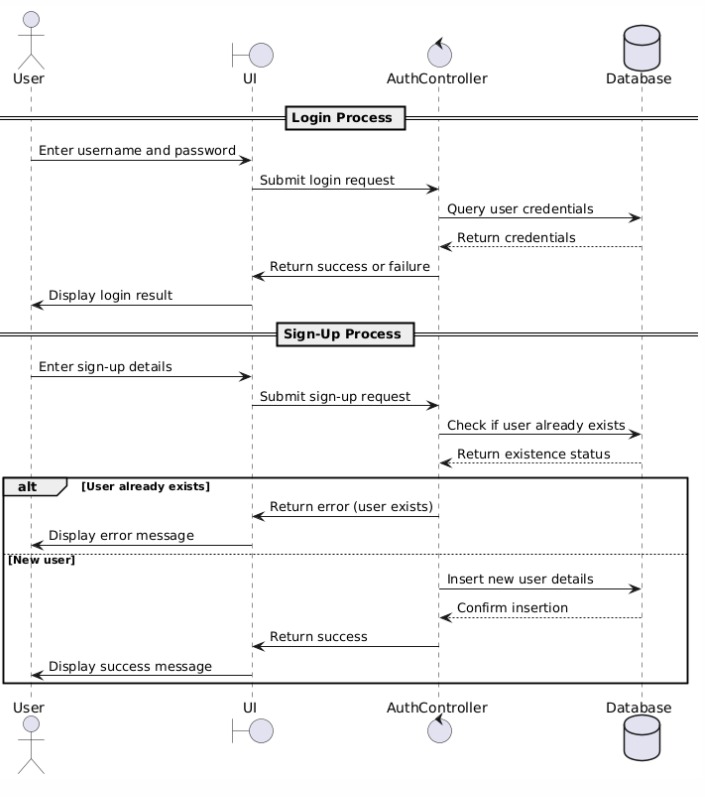
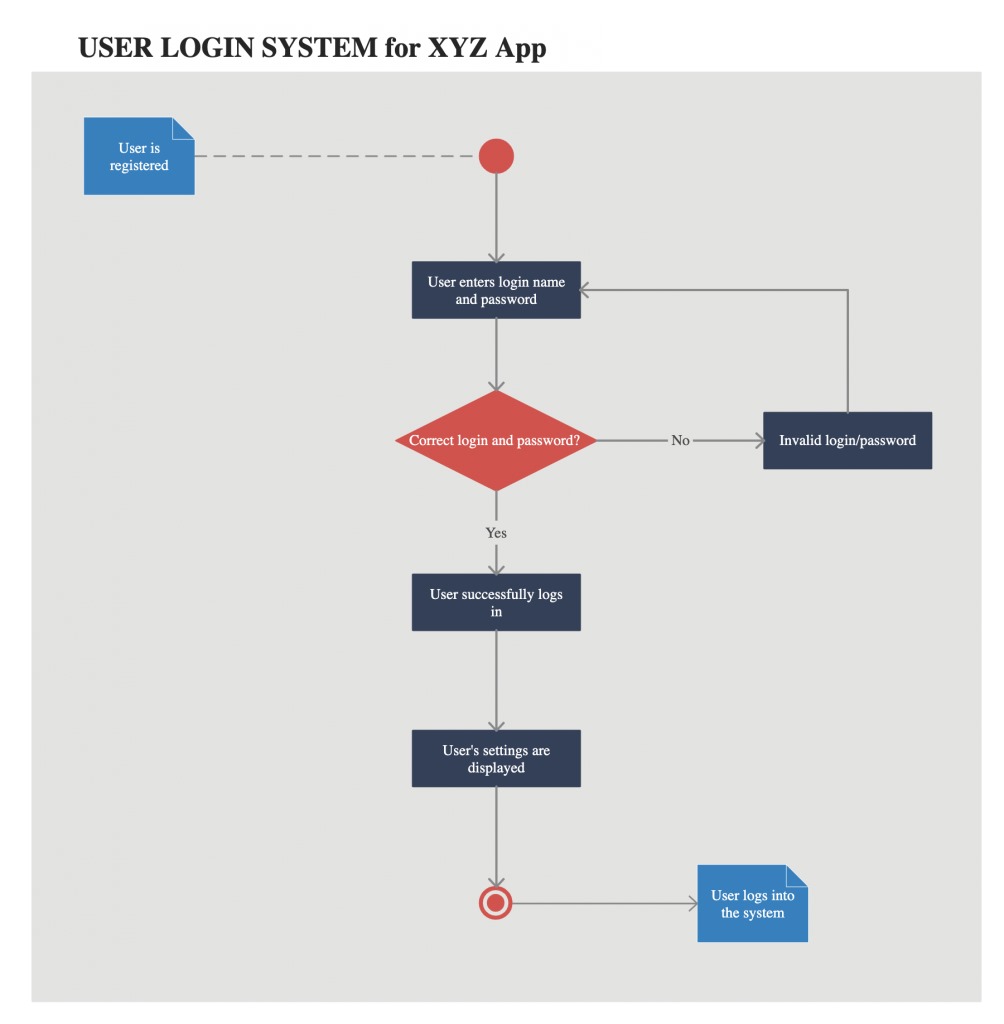
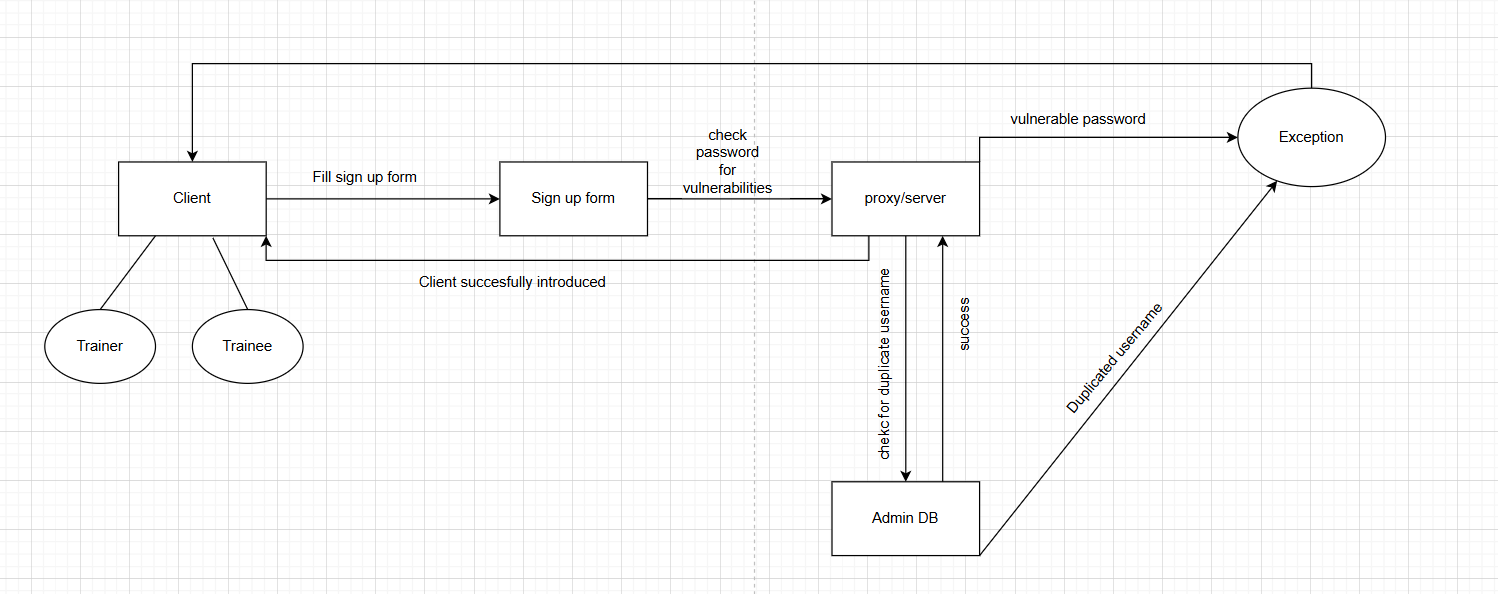


Diagrama de stari: (Claudiu)



Data flow diagram: (Victor)



1. Dezvoltari ulterioare:

* Conectarea la email pentru recuperarea parolei.
* Pagina de home pentru trainer, individuala de cea a trainee-ului.
* Monitorizarea progresului trainee-ului prin grafice si inregistrarile datelor.
* Pagina de meal plan update pentru a permite antrenorului sa schimbe meniul antrenatului.
* Optiuni de logout si delete account care exista deja in baza de date, dar n-au fost implementate.

1. Ghidul utilizatorului

Aplicatia se deschide in pagina de login. Aici utilizatorul se poate loga in contul sau, sau se poate inregistra in aplicatie.

La logare, utilizatorul are optiunea de recuperare a parolei.

Butoanele de Sign Up redirectioneaza utilizatorul catre pagini de sign-up pentru trainer si pentru trainee.

Pagina de sign-up pentru trainer, contine informatiile necesare crearii unui cont clasic. Butonul de sign-up va redirectiona utilizatorul catre pagina home a aplicatiei. In momentul de fata, pagina de home pentru trainer si pentru trainee este comuna.

Pagina de sign-up pentru trainee contine informatii specifice despre acesta, precum greutate, inaltime, etc. Butonul de sign-up ne va redirectiona de asemenea catre pagina de home.

Pagina de home contine, in momentul actual, doar 2 butoane – unul de vizualizare a meniului propus de antrenor catre antrenat si unul de actualizare a greutatii, dupa care, in viitor se va evalua progresul.

Butonul de “Got to meal plan” va deschide o noua pagina care va avea afisate zilele saptamanii si utilizatorul va putea alege pentru ce zi isi doreste sa vada meniul.

Pagina de meal plan se deschide si se prezinta toate cele 7 zile ale saptamanii. Pe orice buton vom apasa, se va deschide pagina cu meniul acelei zile. Fiecare zi va avea unul sau mai multe breakfast-uri, unul sau mai multe lunch-uri si una sau mai multe dinner-uri. In aceasta pagina sunt incluse si snack-urile din acea zi.

Pentru fiecare “masa” din zi avem o imagine representativa si numele preparatului. Butonul de “Read More” aferent aceteia ne va duce la pagina care prezinta ingredientele si modul de preparare pentru masa respectiva.

In aplicatie se poate merge inapoi fie din iconita de back din browser, fie din combinatia de taste Alt + Right Arrow.  
  
 Ocupatii studenti:

Tabulac Daria – Front end (Javascript, css, html)  
Saca Victor – Back End (DataBase Relationships – repositories, services, entities)

Mihaila Claudiu – Back End(Security, Comunicarea Back-endBD-Frontend, entities, SecurityConfig, Controllers)

Resurse:

<https://spring.io/projects/spring-boot>

<https://www.plantuml.com/plantuml/uml/SyfFKj2rKt3CoKnELR1Io4ZDoSa70000>

<https://app.diagrams.net/>

<https://www.pexels.com/search/healthy%20food/>

<https://www.w3schools.com/css/>

<https://web.dev/learn/css>

<https://www.w3schools.com/html/>

<https://web.dev/learn/html>

<https://www.geeksforgeeks.org/spring-boot/>

<https://spring.io/guides/gs/spring-boot>

<https://spring.io/guides/gs/securing-web>