

# Cal-o-Web

## Introducere:

Sa se dezvolte o aplicatie Web care sa permita utilizatorilor autentificati sa-si monitorizeze caloriile consumate pe parcursul unei unitati de timp (zi, saptamina, luna, an) in functie de ce au consumat in acea unitate de timp. De asemenea, se vor putea propune schimbari in dieta fiecarui utilizator pe baza preferintelor personale (e.g., daca pentru Tuxy se constata ca numarul de calorii lunare a scazut, atunci i se poate propune ca duminica sa consume 3 linguri de dulceata sau o banana). Aplicatia va oferi statistici, inclusiv vizualizari de interes, pe baza caloriilor acumulate si (evolutiei) greutatii – introduse manual de utilizator. Rapoartele generate vor fi exportate in format XML.

## Arhitectura Aplicatie

### *1. Tehnologii Folosite*

Partea de front-end a fost implementata folosind HTML5, CSS si putin JavaScript pentru dinamica paginii. Pentru partea de back-end vom folosi PHP fiind usor de integrat intr-o pagina web HTML. Aceasta componenta(back-end) implementeaza functionalitatile front-endului, executa operatii CRUD in baza de date.

Ca stil arhitectural, vom folosi paradigma REST care defineste un set de constrangeri si proprietati bazate pe HTTP. De asemenea aceasta paradigma presupune separarea serverului de client la nivel de implementare, aceasta modularizare ajutand la o modificare eficienta a codului si gestionarea facila a aplicatiei web. Asadar, toate datele fiind pastrate in baza de date la care este conectat serverul.

### *2.Structura Bazei de Date*

Vom utiliza o baza de date MySql pentru a retine datele utilizatorilor.

#### 4.Design Patternul folosit

Aplicația este structurată pe 3 niveluri: *prezentare*, *logică*, *model*.

Nivelul *prezentare*(view) este reprezentat de interfața aplicației. Când utilizatorii accesează aplicația aceștia își pot crea un cont de utilizator sau se pot autentifica. La crearea unui nou cont de utilizator, aplicația validează datele introduse de ei. După ce utilizatorul s-a conectat folosind datele contului acesta poate adăuga alimentele consumate în ziua respectivă sau poate vizualiza un istoric al meselor.

Nivelul *logică* este reprezentat de *controllerul* aplicației care preia inputul de la user. Controllerul va prelua date de la nivelul model pe care le va prelucra pentru ca apoi să fie afișate utilizatorului.

Nivelul *model* este responsabil cu preluarea și prelucrarea datelor care vor fi reținute într-o bază de date.

*Exemplu de comunicare între view, controller și model în timpul interacționării cu aplicația:*

Ne aflăm în pagina principală și dorim să accesăm contul nostru de utilizatori(LogIn), apăsăm linkul de LogIn sau register care ne va redirectiona către pagina corespunzătoare. Introducem datele în casutele username și password și apăsăm butonul LogIn, în acest moment componenta controller primește de la view aceste date, username-ul și parola pe care le-am introdus, controllerul pasează aceste date modelului iar modelul verifică în baza de date, mai exact în tabela user dacă aceste date introduse sunt unice, dacă da, atunci modelul ia din tabela datele necesare afișării următorului page, le pasează controllerului, iar acesta le dă mai departe view-ului pentru ca acesta să afișeze pagina corespunzătoare(History/AddFood), dacă nu, atunci controllerul dă mai departe un mesaj de eroare care este mai apoi afișat în view, în pagina Log in.

#### 6.Use Cases

##### 6.1 Utilizatorul dorește să vizualizeze recomandările

1. Accesează site-ul
2. Accesează linkul care îl va redirectiona către pagina de LogIn sau Register
3. Selectează LogIn
4. Introduce Username-ul și Parola
5. Accesează pagina de History
6. În partea de sus a paginii vor fi afișate recomandările

##### 6.2 Utilizatorul înregistrat dorește să adauge un nou aliment

1. Accesează aplicația
2. Accesează linkul care îl va redirectiona către pagina de LogIn sau Register
3. Selectează opțiunea LogIn
4. Accesează linkul Add Food
5. Utilizatorul completează formularul pentru adăugarea unui nou fel de mâncare
6. Datele introduse de utilizator sunt stocate în baza de date

### 6.3 Utilizatorul inregistrat doreste sa adauge alimentele consumate

1. Acceseaza aplicatia
2. Acceseaza linkul care il va redirectiona catre pagina de LogIn sau Register
3. Selecteaza optiunea de LogIn
4. Acceseaza linkul AddFood
5. Utilizatorul completeaza formularul pentru adaugarea alimentelor consumate in cadrul unei mese
6. Apasa butonul Save
7. Datele sunt salvate in baza de date