

Fotball-IT
RAPORT DE
PROIECTARE

Februarie 2017

Cuprins

1. Scopul aplicației	3
2. Aria de acoperire a aplicației.....	3
3. Perspectiva funcțională	3
4. Perspectiva structurală	3
5. Perspectiva comportamentală	4
6. Modelul de date	4
7. Perspectiva fizică	5
8. Perspectiva de dezvoltare.....	5
9. Perspectiva de testare.....	5

1. Scopul aplicației

Proiectul nostru consta intr-o aplicatie web care are ca scop cresterea performantelor echipelor de fotbal prin evaluarea antrenamentelor.

Aplicatia "Fotball-IT" doreste sa usureze munca antrenorilor si jucatorilor de fotbal prin faptul ca ofera o analiza completa si un profil personalizat pentru fiecare utilizator in parte. In acelasi timp, ofera simplitate in utilizare si gestionare.

Utilizatorii aplicatiei sunt de trei tipuri: antrenor echipa, jucator si administrator. Exista o interdependenta intre cele doua roluri (antrenor si jucator), astfel ca antrenorul poate sa vada statusul si progresul fiecarui jucator, iar jucatorul poate in orice moment sa isi vada progresul in echipa si forma lui actuala din viziunea antrenorului. Administratorul poate sa gestioneze situatia echipelor si sa asigneze un antrenor pentru echipa sa.

2. Aria de acoperire a aplicației

Aplicatia noastra va fi web, pentru a putea fi la indemana jucatorilor atunci cand vor sa isi urmareasca performantele. Deoarece aplicatia noastra utilizeaza tehnica de gamification, implementam un sistem de progres si challenge-uri pentru a angrena cat mai mult utilizatorii pentru a-si indeplini scopul final, acela de a ajunge cat mai aproape de nivelul stabilit de antrenor la inceputul sezonului.

Aplicatia face parte din categoria celor sportive si doreste sa rezolve o problema de gestiune a antrenamentelor. Antrenorul impreuna cu jucatorii stabilesc obiective individuale la inceput de sezon, pe parcursul acestuia jucatorii pot vedea diferentele dintre situatia lor actuala si obiectivele stabilite, astfel incat acestia sunt motivati pentru a atinge performanta setata.

Aplicatia va avea o componenta: pregatire pe baza de antrenament. Pregatirea pe baza de antrenament presupune stabilirea unor obiective preliminare de catre antrenor si jucator inainte de pregatirea fizica, iar la final jucatorul isi va nota activitatea, in functie de implicarea si performanta din timpul antrenamentului. Acest raport pe care il primeste jucatorul la finalul pregatirii il poate influenta si motiva pe acesta, iar per total are un impact major in performanta jucatorului si valoarea lui.

Aplicatia nu garanteaza rezultate jucatorilor si nici titularizarea acestora in meciurile oficiale.

Antrenorul nu va avea posibilitatea de a schimba obiectivele pe termen scurt ale jucatorilor, ci doar a celor pe termen lung (la inceputul sezonului).

Aplicatia nu are ca scop oferirea de premii substantiale jucatorilor, ci doar incurajarea acestora pentru a atinge obiectivele setate.

3. Perspectiva funcțională

Este prezentata in documentul UML.html atasat.

4. Perspectiva structurală

Primul modul important al aplicatiei este reprezentat de sistemul de log-in in aplicatie. Utilizatorul va trebui sa introduca credentialele in interfata grafica, iar cand va apasa butonul de logare se va face un request catre server care va verifica si aproba datele introduse in formular cu cele aflate in baza de date. Odata logat, in functie de tipul de user (administrator, jucator sau antrenor) se va afisa o pagina cu datele fiecarui user in parte.

Daca acesta nu are un cont creat, isi va putea crea unul pe baza unui token primit de la managerul echipei. Aplicatia va putea face diferenta dintre un administrator, jucator sau antrenor pe baza acestui token.

Un alt modul prezent in aplicatie este cel de monitorizare obiective jucator. In momentul in care jucatorul s-a logat in aplicatie si va merge pe pagina sa, se va face un request catre server, iar acesta va interoga baza de date pentru a primi toate detaliile lui, de la obiective curente/de viitor, pana la attribute specifice.

O a treia componenta este data de sistemul de antrenamente prezent in aplicatie. Jucatorul apasand pe butonul "Incepere antrenament" se va putea inscrie in cadrul unui antrenament, cu scopul de a creste performantele

sale in joc. Se comunica catre server o cerere, iar acesta va inregistra o noua linie in baza de date, iar in acelasi timp antrenorul va fi instiintat de acest lucru. Cel din urma va putea sa selecteze tipul de antrenament (poate chiar sa defineasca un nou tip). O data ce antrenamentul s-a terminat jucatorul va introduce datele referitoare la antrenamentul efectuat, iar in functie de rezultatele jucatorului, acesta va primi diferite recompense.

Ultimul modul, dar cel mai important, este reprezentat de sistemul de gamification. Acesta apare integrat in toate celelalte module, mai mult sau mai putin, dar acesta va tine mereu jucatorul si antrenorul activi in aplicatie.

5. Perspectiva comportamentală

Aplicatia ofera trei tipuri de utilizatori: administrator, jucator si antrenor.

In momentul in care si va accesa site-ul aplicatiei va aparea pagina de home. Utilizatorul poate vedea informatii despre ce poate face aplicatia, iar, cel mai important, se poate loga sau crea cont in aceasta.

Userul va putea alege dintre 2 optiune: inregistrare (creare cont) si logare.

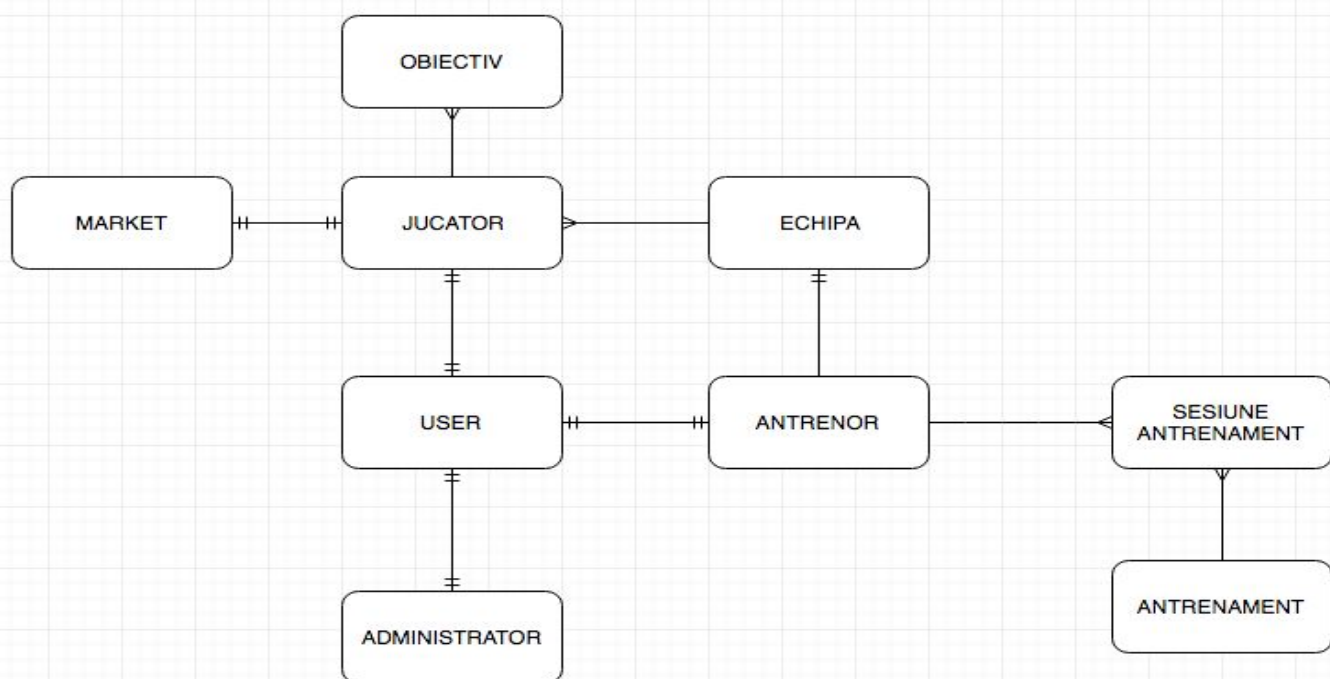
Daca va da click pe butonul de inregistrare i se va deschide o noua pagina unde va trebui sa completeze un formular pentru creare cont. Dupa ce introduce datele si tokenul aferent lui, aplicatia va decide pe baza acelui token ce tip de utilizator este. Daca datele sunt corecte, iar aplicatia valideaza acest lucru pe parte de server, utilizatorul va fi redirectionat pe principala existenta in aplicatie. Daca nu, va primi un mesaj de eroare.

Daca userul va da click pe butonul de logare, va fi redirectionat catre pagina de logare unde va completa datele cerute.

Odata logat, in functie de tipul de utilizator vor exista 3 pagini de redirectionare:

- administrator - va avea acces la pagina de unde poate adauga/edita/sterge date din baza de date (referitoare la antrenorii echipelor din aplicatie). Acesta are rolul de a mentine aplicatia din punct de vedere al consistentei datelor.
- jucator - va fi redirectionat spre pagina lui cu datele sale: poza, obiective, attribute, grafice de performanta. Daca doreste sa inceapa antrenament va apasa pe butonul respectiv, iar dupa terminarea acestuia va putea sa se autoevalueze.
- antrenor - va fi redirectionat catre pagina persoanal, unde va vedea date despre el, poza sa si jucatorii pe care ii antreneaza. Daca va da click pe butonul "Antrenament" va fi trimis catre pagina speciala de antrenamente, unde va putea selecta tipul de antrenament pentru jucatorii inscrisi la sesiunea curenta de antrenament. Tot in aceasta pagina se afla optiunea de Adaugare obiective pentru jucatorii echipei pe care o antreneaza si vizualizarea tuturor obiectivelor setate la un moment de timp.

6. Modelul de date



7. Perspectiva fizică

Aplicatia va avea doua componente principale din perspectiva modului de scriere:

- partea de back-end: limbajul de programare folosit este PHP, versiunea 5.6. Pentru a scrie intr-un mod mult mai simplu aplicatia si a fi modulara vom folosi un framework pentru PHP numit Laravel. Acesta contine la randul lui si un framework pentru maparea tabelor din baza de date (ORM - Object Relational Mapping), numit Eloquent, ce ne va ajuta sa interactionam mult mai usor si flexibil cu baza de date relationala.
- partea de front-end: limbajul de marcare folosit este HTML (Hypertext Markup Language), pentru stiluri CSS (Cascade Style Sheet) impreuna cu Bootstrap ca framework. Pentru ca aplicatia sa fie cat mai simplu de folosit, unele apeluri catre server se vor face asincron, de aceea vom folosi AngularJS (Javascript) pentru animatii si AJAX (Asynchronous Javascript and XML).

Din perspectiva puterii de procesare, aplicatia va rula intr-o instanta de server cu PHP intr-un mediu “Cloud”, mai exact Heroku. Aceasta resursa se va adapta in mod dinamic in functie de cererile ce se vor face asupra ei. De exemplu, in mod normal instanta ofera 1 Procesor cu 2 Thread-uri si 1 GB RAM, dar in cazul in care aplicatia va primi un numar de cereri foarte mare, resursele fizice vor creste.

Baza de date ruleaza si ea, de asemenea, in Heroku si la fel ca in cazul instantei aplicatiei poate fi scalata in functie de necesitate.

Din punct de vedere al modului de sincronizare a membrilor echipei asupra codului, vom folosi GIT pentru controlul versiunilor de cod (varianta implementata de Bitbucket). Fiecare va avea clonat repository-ul de pe server si va lucra in functie de task pe una sau mai multe branch-uri. In acelasi timp, pentru testare de functionalitati, fiecare membru va avea propriul mediu de programare, de la server PHP pana la baza de date locala.

Mediul de programare (IDE) va fi PhpStorm, un produs al JetBrains. Am ales acest IDE, deoarece simplifica scrierea oferind ajutor din punct de vedere sintactic.

8. Perspectiva de dezvoltare

Fisierele privind perspectiva de dezvoltare sunt atasate documentului. Am realizat aceasta perspectiva de dezvoltare a aplicatiei folosind planul general de dezvoltare, dar si planul saptamanal de activitate.

Planul general de dezvoltare este alcatuit din impartirea pe doua intervale a tuturor componentelor care urmeaza a fi dezvoltate:

1. saptamanile 1-5 (cuprinse in perioada 14.11 - 18.12)
2. saptamanile 9-10 (cuprinse in perioada 9.01-22.01)

Planul saptamanal de activitate cuprinde activitatile detaliate sub forma de user stories, alaturi de termenul de livrare a acestora, dar si de persoana alocata pentru acea activitate.

9. Perspectiva de testare

Dupa fiecare activitate livrata in aplicatie se va testa noua functionalitate; testarea se va face mai ales prin formulare de introducere a datelor, acesta fiind un subiect sensibil intr-o aplicatie.

Testarea se va face in doua moduri:

1. testare functionala: folosindu-ne de scenariile de calitate definite in raportul de analiza;
2. testare tehnica: folosindu-ne de cazurile de utilizare definite in raportul de analiza.

Doi membri ai echipei se vor ocupa peste 50% din timp pentru a testa aplicatia si a valida functionalitatile ei.