Aplicație de transferuri de jucători de fotbal

(mini Transfermarkt)

Marin Cristian-Alexandru

343C1

Pentru acest proiect am ales sa implementez un set de servicii care sa compuna o aplicatie care face management pe o baza de date cu jucatori de fotbal si clubul lor sportiv la care activeaza in mod curent.

Dupa cum stim, in functie de modul in care evolueaza/involueaza in cariera si performante sau inainteaza in varstra, jucatorilor le cresc, respectiv le scad preturile pe piata globala a transferurilor. Pentru o echipa de prestigiu ce doreste performanta in mod constant, dar caruia ii si permite bugetul, tinta principala vor fi mereu cei mai buni jucatori, cei mai tineri si cu cel mai mult talent.   
 Un alt aspect de trebuie luat in considerare este concurenta intre cluburi pentru anumiti jucatori populari intr-o perioada de transfer, acestia sunt monitorizati de scouteri cu mai mult timp inainte si mai multe oferte sunt plasate sub forma de licitatie pentru a indeplini doua conditii: un pret cel putin cat cel minim stabilit intern de club, si un contract rezonabil pentru jucator. De aceea multe dintre transferuri se opresc la stadiul de zvon.

Aplicatia propusa de mine va avea o instanta de client propiu pentru fiecare club pentru ca acesta sa confirme plecarea unui jucator de la clubul lui catre un altul pentru ca au fost indeplinite conditiile mentionate anterior, dar, dupa propria dorinta, pentru a atrage si atentia altor cluburi de a-si inainta ofertele, sa publice un zvon despre o oferta primita la acel moment de unde licitatia poate porni.

Toate transferurile sau zvonurile plasate de un club vor fi transferate catre un server care le va procesa si le va stoca in baza de date pentru a fi aprobate de autoritatea suprema(adminul) si expuse in Internet(serviciu de monitorizare web).

In aplicatia de web de final, diferite statistici vor putea fi analizate cu ajutorul mediului de monitorizare Grafana care va inspecta noile transferuri plasate in baza de date si fiecare utilizator ce o acceseaza poate vizualiza informatii precum: cel mai scump transfer din acea perioada, o lista cu cei mai valorosi jucatori “pe picior de plecare” si asa mai departe precum si multe altele.

Aplicatia web de monitorizare are ca public tinta, pe langa managerii cluburilor pentru a fi instiintati, dar si suporteri la modul general, oameni neimplicati direct in procesele de transfer, astfel ca informatiile expuse trebuie sa respecte anumite politici de securitate ale cluburilor deci acestea vor fi simplificate oferindu-se clientilor doar ceea ce poate fi publicat.

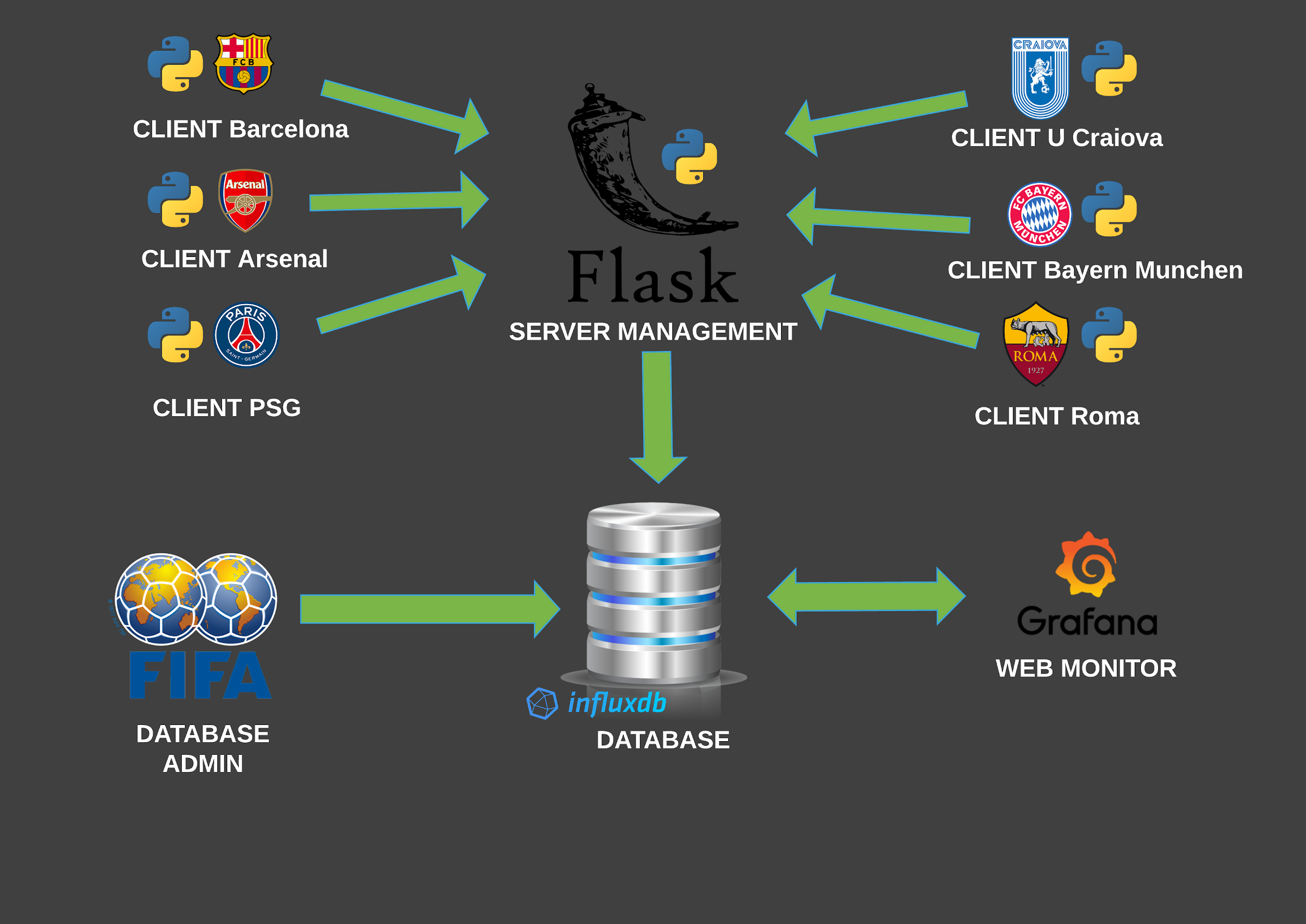
Baza de date va contine informatii despre jucatori precum: Nume, varsta(in ani), club actual, club viitor(sau ofertant in cazul unui zvon), valoare jucator, pret oferit si alte informatii relevante unui transfer si le va memora in ordinea plasarii unui transfer (timestamp database) pentru a pastra veridicitatea si actualitatea informatiei.

Un alt serviciu il va reprezenta un administrator de baza de date care va fi instanta unei autoritati supreme (FIFA spre exemplu) care va putea interzice anumite transferuri chiar si daca acestea au fost in prima faza acceptate de ambele parti, din varii motive (un jucator deja a fost transferat in aceeasi perioada de transferuri si nu poate sa mai fie transferat, fairplay financiar: o echipa ce are surse de venit incomparabile cu altele, trebuie limitata prin realizarea unor transferuri pentru a putea pastra farmecul jocului).

Tehnologies & TOOLS

* DOCKER
* FLASK
* PYTHON
* GRAFANA
* INFLUXDB
* LUCIDCHART
* GIT

FLOWCHART APLICAȚIE:



Conturi:

1. Docker HUB

<https://hub.docker.com/repository/docker/alexmarin88/idp_project>

1. GITHUB

<https://github.com/marin-cristian-alexandru/Proiect_IDP>

Servicii:

* Clientul

Ce va reprezenta un script de python in care se vor prelua datele de la user si se vor pune folosind un request la una din caile la care asculta serverul. Interfata acestuia va fi una minimala realizata in terminal.

* Serverul

Un serviciu ce are si rolul de adaptor intre clienti si serviciul de persistenta. Acesta va fi implementat cu Flask pentru a asculta la mai multe endpointuri pentru a primi date de la clienti si a le adauga la baza de date cu amprenta de timp curenta

* Baza de date

Va fi un TSDB(cheia de memorare va fi un timestamp) pentru a facilita scopul de a retine ordinea transferurilor. Va stoca intr-o tabela cu jucatori datele transferului acestora. Pentru aceasta voi folosi o imagine deja existenta de InfluxDB.

* Administrator baza de date

Acesta va fi o autoritate suprema si va fi cel de-al treilea serviciu creat de mine, dupa server si client si va avea dreptul de a analiza transferurile realizate si in caz ca anumite reguli nu sunt respectate, sa nu le permita publicarea, deci o citire si o stergere directa a bazei de date.

* Serviciu de monitorizare

Un serviciu pentru care voi folosi Grafana pentru monitorizare si pe care il voi lega la baza de date pentru a prelua informatiile si pentru a urmari statistici si informatii generale despre transferurile realizate.

Lucrul cu dockere:

!Aceasta parte va mai suferi modificari pe parcurs in functie de modul in care voi ajunge sa folosesc anumite componente.

Imagini oficiale folosite: Grafana si InfluxDB

Imagini scrise: Client, Server, AdminDB

Voi lega in retele imaginile pentru a avea conexiune doar cele care trebuie sa comunice conform flowchartului de mai sus.

Fiecare imagine creata de mine va fi creata cu un dockerfile si se va putea porni impreuna cu stiva de servicii ce va exista la final.

Pentru Grafana se va realiza autentificare personalizata si un set de templateuri de grafice prestabilite.

Fisierul docker\_compose.yml va contine toate informatiile necesare pentru un build automat al serviciilor si pornirea acestora.