**Tema 1 – PSSC**

-Sistem software de plasare a pariurilor online-

**CAPITOLUL 1 –** Prezentarea cerințelor funcționale și non-funcționale și identificarea acelor cerințe ce influențează arhitectura.

**Descrierea Sistemului:**

Sistemul va consta într-un site web destinat pentru plasare de pariuri sportive online și va conține următoarele funcționalități:

1. Fiecare utilizator va avea un cont pentru identificare la accesarea site-ului.
2. În cazul în care utilizatorul nu deține un astfel de cont, sistemul va conține o funcționalitate de înregistrare.
3. După înregistrare se va cere printr-un email o dovada a faptului că utilizatorul are cel putin vârsta de 18 ani.
4. După ce autentificarea a avut loc cu success, utilizatorul va avea acces la toate evenimentele sportive din ziua curentă cât și cele din următoarele 6 zile.
5. Evenimentele sunt actualizate în fiecare zi astfel încât să fie puse la dispoziția utilizatorului toate evenimentele de pe parcursul unei săptămâni.
6. Utilizatorul va putea transfera bani pe cont direct din contul bancar.
7. Utilizatorul va putea retrage bani din cont direct pe contul bancar.
8. Un bilet odata plasat nu poate fi anulat.
9. Biletele plasate vor fi permanent verificate, în caz de câștig suma castigate va fi adaugată la contul current.
10. Utilizatorul va avea acces la istoricul biletelor.
11. Utilizatorul va putea plasa bilete cu miza de maxim valoarea contului său current.
12. Utilizatorul va putea adauga pe un singur bilet un numar maxim de 20 de evenimente.
13. Utilizatorul nu va putea plasa un bilet al cărui câstig depășește 500.000 lei.
14. Cotele evenimentelor sunt calculate în conformitate cu cei mai puternici traderi la nivel mondial.

**Cerințe ce înfluențează arhitectura:**

1. Sistemul va avea nevoie de o bază de date pentru stocarea conturilor utilizatorilor.
2. Sistemul va avea nevoie de o bază de date pentru stocarea evenimentelor și a cotelor acestora.
3. Sistemul va avea nevoie de un modul ce permite tranzacții între conturi bancare și conturi de pariuri.
4. Pentru validarea conturilor, acestea necesită o verificare de către admini a vârstei utilizatorului.

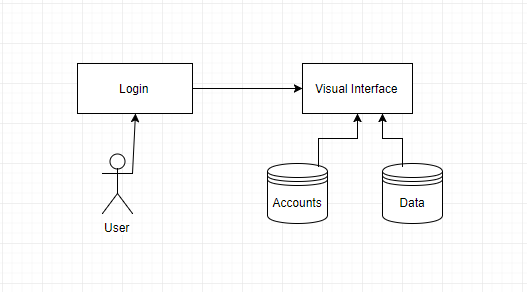
**CAPITOLUL 2 –** Descompunerea în componente.

**Definirea responsabilităților:**

Sistemul software este format din două baze de date în care sunt stocate datele. Una pentru conturile utilizatorilor și alta pentru cotele evenimentelor, acestea fiind componenetele de stocare a datelor.

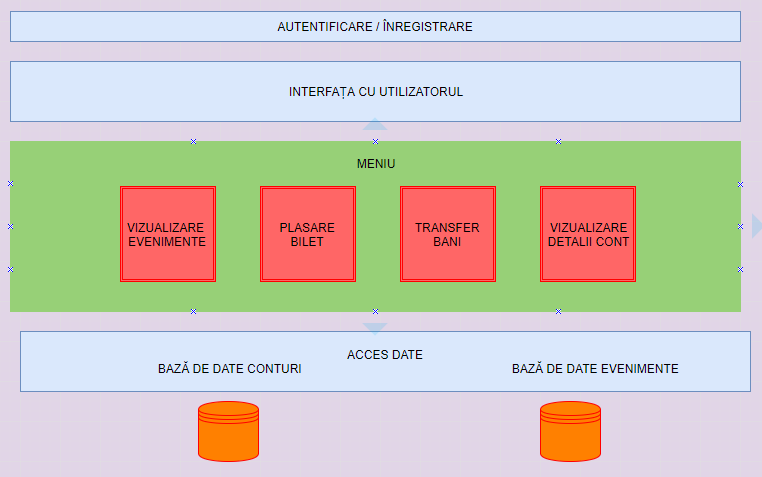
Interfața utilizatorului este un al component în care sunt afișate evenimentele iar utilizatorul are acces la datele contului său.

Un alt component este definit de o interfață de autentificare și înregistrare ce fac legătura între baze de date și înterfața utilizatorului.

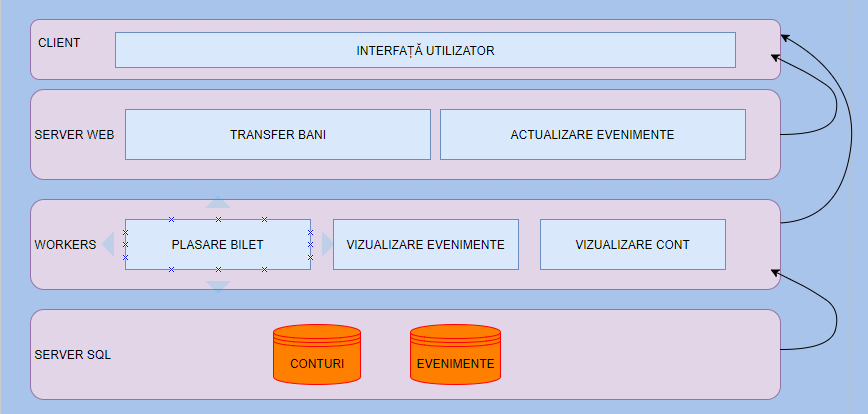


**CAPITOLUL 3 –** Prezentarea sistemului software.

**Perspectivă logică:**



**Perspectivă proces:**



**CAPITOLUL 4 –**Indicatori de calitate.

**Performață –** deoarece sistemul conține doar două baze de date, una fiind pentru autentificare iar cea de a doua fiind pentru importul de evenimente majoritatea operațiilor se vor face la nivel de server ceea ce facilitează viteza de răspuns la anumite taskuri.

**Securitatea –** datele sunt bine securizate deoarece se folosește o criptare de tip “one way” SHA256, asta făcând imposibilă decriptarea datelor chiar și la eventualele vulnerabilități ale bazei de date.

**Tolerața la modificări –** sistemul fiind relative simplu și bine împărțit în module, eventualele modificări la nivelul acestora nu vor genera schimbări majore c ear afecta întreg sistemul. Astfel o nouă funcționalitate fiind ușor de implementat.

**CAPITOLUL 5 –**Tehnologii middleware.

**Server SQL –** Pentru accesarea datelor din bazele de date se folosește o conexiune de tip SQL.

**CAPITOLUL 6 –**Modele Arhitecturale.

Pentru o mai bună organizare a site-ului și a modulelor am folosit **Model–view–controller**  sau **MVC.**

Acest model este util deoarece împarte sistemul în trei module, după cum urmează:

1. **Model** – este component central a sistemului, aceasta fiind component ce manipuleaza datele și controlează modul în care funcționează aplicația.
2. **View –** modul ce se ocupă cu ieșirile sistemului, output-urile generate de aplicație.
3. **Controller –** modul ce controlează atât “MODEL” cât și “VIEW” putând controla modul în care acționează acestea.

**CAPITOLUL 7 –**Prezentarea scenariilor de validare.

Validarea sistemului se va realiza prin creearea unui prototip și construirea unor scenarii pentru imitarea acțiunilor utilizatorilor.

Aceste scenarii au ca intenție trecerea prin toate stăile sistemului.

Odată ce aceste scenarii sunt realizate iar rezultatele sunt cele așteptate, sistemul se consideră a fi functional.