

Tema 1 – PSSC

*Analiza unui sistem software (Aplicatie de management a turneelor
de tenis de masa)*

Cuprins

Prezentarea cerintelor functionale si non-functionale si identificarea acelor cerinte care influenteaza arhitectura	3
Cerinte functionale si non-functionale	3
Cerinte care influenteaza arhitectura	3
Descompunerea in componente, definirea responsabilitatilor componentelor si a relatiilor dintre ele.....	4
Componente	4
Definirea responsabilitatilor componentelor	4
Relatiile dintre componente	4
Prezentarea sistemului software	5
Arhitectura privita din perspectiva logica	5
Arhitectura privita din perspectiva process	5
Identificarea celor mai importanti indicatori de calitate, specificarea masurii alese pentru fiecare indicator de calitate.....	6
Performanta.....	6
Fiabilitatea	6
Scalabilitatea.....	6
Toleranta la modificari	6
Identificarea tehnologiilor middleware folosite pentru a comunica intre componente	6
Identificarea principalelor module si stiluri arhitecturale folosite	7
Prezentarea scenariilor de validare a arhitecturii.....	7

Prezentarea cerintelor functionale si non-functionale si identificarea acelor cerinte care influenteaza arhitectura

Cerinte functionale si non-functionale

Cerinte functionale

Sistemul trebuie sa permita:

- a. Definirea si managementul unor turnee de tenis cu un numar variabil de echipe (min. 3): creare turneu, definirea unor echipe participante, definire reguli (ex: departajare prin golaveraj sau prin meci direct), generare de meciuri, adaugare de rezultate, vizualizare clasament.
- b. Managementul unei liste de jucatori – adaugare, editare, stergere jucatori, plus vizualizare statistici despre rezultatele jucatorilor si statistici de tip head to head.

Cerinte non-functionale:

- Folosirea unei baze de date SQL
- User interface trebuie sa fie in SWT
- Pentru lucrul cu baze de date se va folosi Hibernate

Cerinte care influenteaza arhitectura

Performanta

Puterea de procesare – aplicatia va fi capabila sa proceseze informatii despre turnee, chiar daca acestea vor fi de ordinul sutelor.

Deadline – aplicatia va avea timpii de raspuns sub 3 secunde pentru orice operatie efectuata.

Fiabilitatea

Nu este permisa pierderea de date, acestea reprezentand un lucru esential pentru aplicatia noastra.

Scalabilitatea

Aplicatia trebuie sa permita realizarea unor turnee complexe si gestionarea unei baze de date de dimensiuni gigantice.

Descompunerea in componente, definirea responsabilitatilor componentelor si a relatiilor dintre ele

Componente

- Acces date
- Database
- Export PDF
- PC

Definirea responsabilitatilor componentelor

Acces date – acest lucru presupune conexiunea cu baza de date si permite obtinerea acestora din baza de date

Database – permite stocarea informatiilor in baza de date

Export PDF – permite exportarea unui fisier PDF cu rezultatele turneului

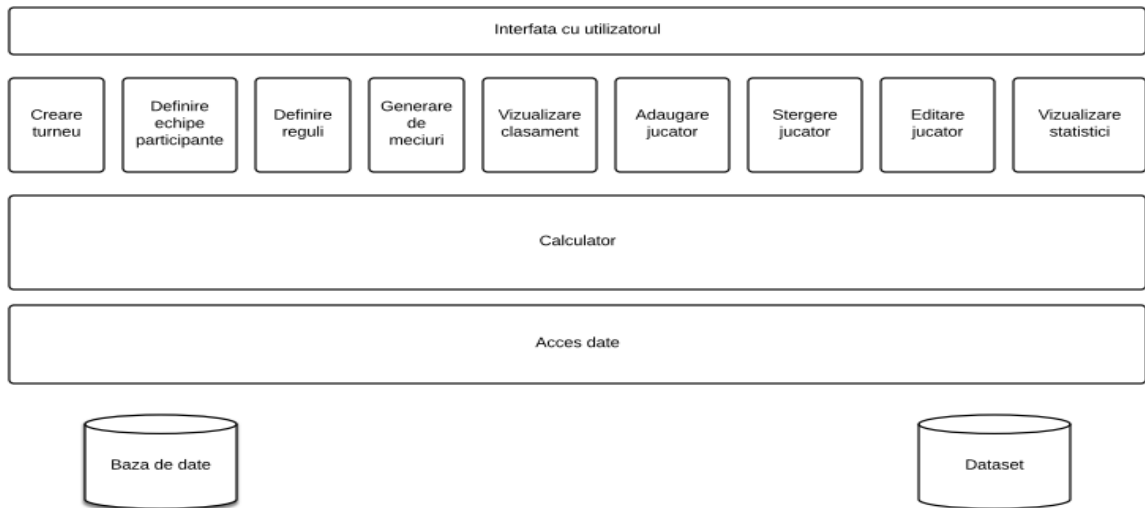
PC – ocupat cu prelucrarea datelor

Relatiile dintre componente

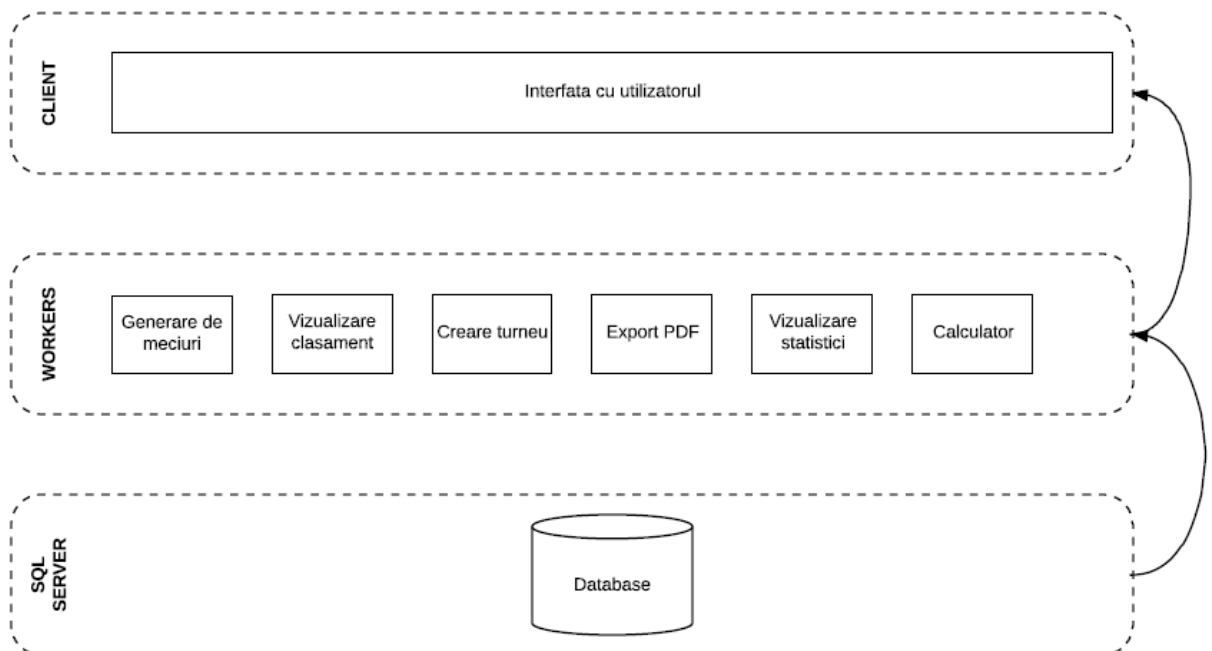
Componenta Acces date permite accesul la datele din baza de date, furnizandu-le componentei Calculator pentru a le prelucra, iar datele prelucrate vor fi transferate componentei Export PDF.

Prezentarea sistemului software

Arhitectura privita din perspectiva logica



Arhitectura privita din perspectiva process



Identificarea celor mai importanti indicatori de calitate, specificarea masurii alese pentru fiecare indicator de calitate

Performanta

Puterea de procesare – aplicatia va fi capabila sa proceseze informatii despre turnee, chiar daca acestea vor fi de ordinul sutelor.

Deadline – aplicatia va avea timpii de raspuns sub 3 secunde pentru orice operatie efectuata.

Fiabilitatea

Nu este permisa pierderea de date, acestea reprezentand un lucru esential pentru aplicatia noastra.

Scalabilitatea

Aplicatia trebuie sa permita realizarea unor turnee complexe si gestionarea unei baze de date de dimensiuni gigantice.

Toleranta la modificari

Aplicatia trebuie sa permita generarea de turnee pentru orice format al competitiei, indiferent de echipele participante sau de tipul de sport practicat.

Identificarea tehnologiilor middleware folosite pentru a comunica intre componente

OLE DB connection (*Object Linking and Embedding, Database*, sometimes written as **OLEDB** or **OLE-DB**)

Identificarea principalelor module si stiluri arhitecturale folosite

Module:

- User interface
- SQL
- Worker

Stiluri arhitecturale:

- Client – Server (cerere si obtinere de date)
- Coordonator de procese

Prezentarea scenariilor de validare a arhitecturii

- Utilizarea scenariilor – pentru o buna dezvoltare a aplicatiei se va face uz de scenariile simple de turnament: 4 echipe, 8 echipe si 16 echipe. Pe baza acestui scenariu se va putea scala generarea de turnamente la orice numar de echipe.