



Chess Engine

Sesiunea Iulie 2016

Autor
Balazs David
David.balazs1993@gmail.com

Coordonator Stiintific
s.l. ing Cosmina Ivan



Continut Prezenta

- Contextul proiectului
- Scop si obiective
- Studiu bibliografic – solutii existente
- Solutia propusa
- Tipuri de utilizatori
- Cerinte functionale
- Cerinte functionale – use case
- Cerinte nefunctionale
- Optiuni tehnologice
- Arhitectura conceptuala
- Detalii de implementare
- Dificultati – rezolvari
- Elemente de testare
- Interfata grafica
- Concluzii
- Dezvoltari viitoare
- Referinte bibliografice



Contextul proiectului

- Gradul ridicat de interes al jucatorilor de sah pentru folosirea tehnologiei in antrenamentele lor
- Interesul utilizatorilor de internet obisnuiti de a-si dezvolta logica, imaginatia si spiritul analitic



Scop si obiective

Scop principal

Realizarea unui sistem ce ofera suport de antrenament jucatorilor de sah, sistem ce respecta noile tehnologii si abordari din domeniul aplicatiilor WEB.



UNIVERSITATEA TEHNICĂ
DIN CLUJ-NAPOCA

Scop si obiective

Obiective principale

- Posibilitatea jucatorului virtual de a juca un meci la un anumit nivel de dificultate, cu timp de raspuns ce se incadreaza in limita specificata in cadrul cerintelor nefunctionale.
- Definirea proiectarea si construirea unui sistem capabil sa ofere suport de antrenament jucatorilor de sah prin expunerea de strategii si probleme de sah.
- Mod de relaxare pentru jucatorii mai putin experimentati.



Studiu bibliografic – solutii existente

Criteriu de evaluare	ChessEngine	Chess.com	Chess Master
Aplicatie WEB	✓	✓	✗
Permite interactiune cu administratorul direct din aplicatie	✓	✓	✗
Ofera tutoriale legate de jocul de sah	✓	✓	✓
Ofera un set de strategii de joc	✓	✓	✓
Interfata intuitiva	✓	✓	✓
Interfata simplificata, orientata catre functionalitatea esentiala	✓	✗	✓
Ofera un top al jucatorilor	✓	✓	✗
Permite vizualizare de statistici personale	✓	✓	✓
Performanta sistemului nu este dependenta de statia jucatorului	✓	✓	✗
Permite modul multiplayer	✗	✓	✓
Permite diferite nivele de dificultate	✓	✓	✓



Solutia propusa

- Sistem cu interfata grafica disponibila in browser web
- **Jucator virtual** – posibilitatea de a juca sah cu un jucator virtual cu diferite nivele de dificultate
- **Strategii de sah** – posibilitatea analizarii strategiilor de sah si descrierii mutarilor
- **Probleme de sah** – dezvoltarea capacitatii de a juca sah



Tipuri de utilizatori

Jucator neautentificat

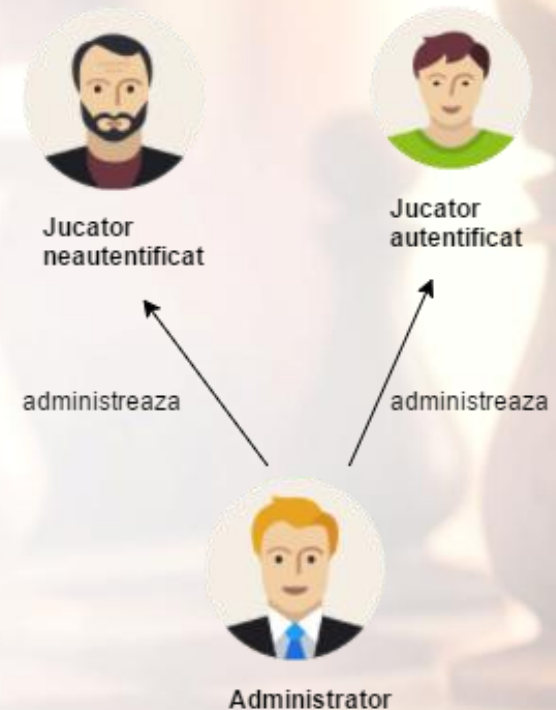
-> Este un jucator ocazional

Jucator autentificat

-> Are acces la functionalitati aditionale precum salvarea meciurilor, vizualizare de statistici

Administratorul

-> Administreaza jucatorii

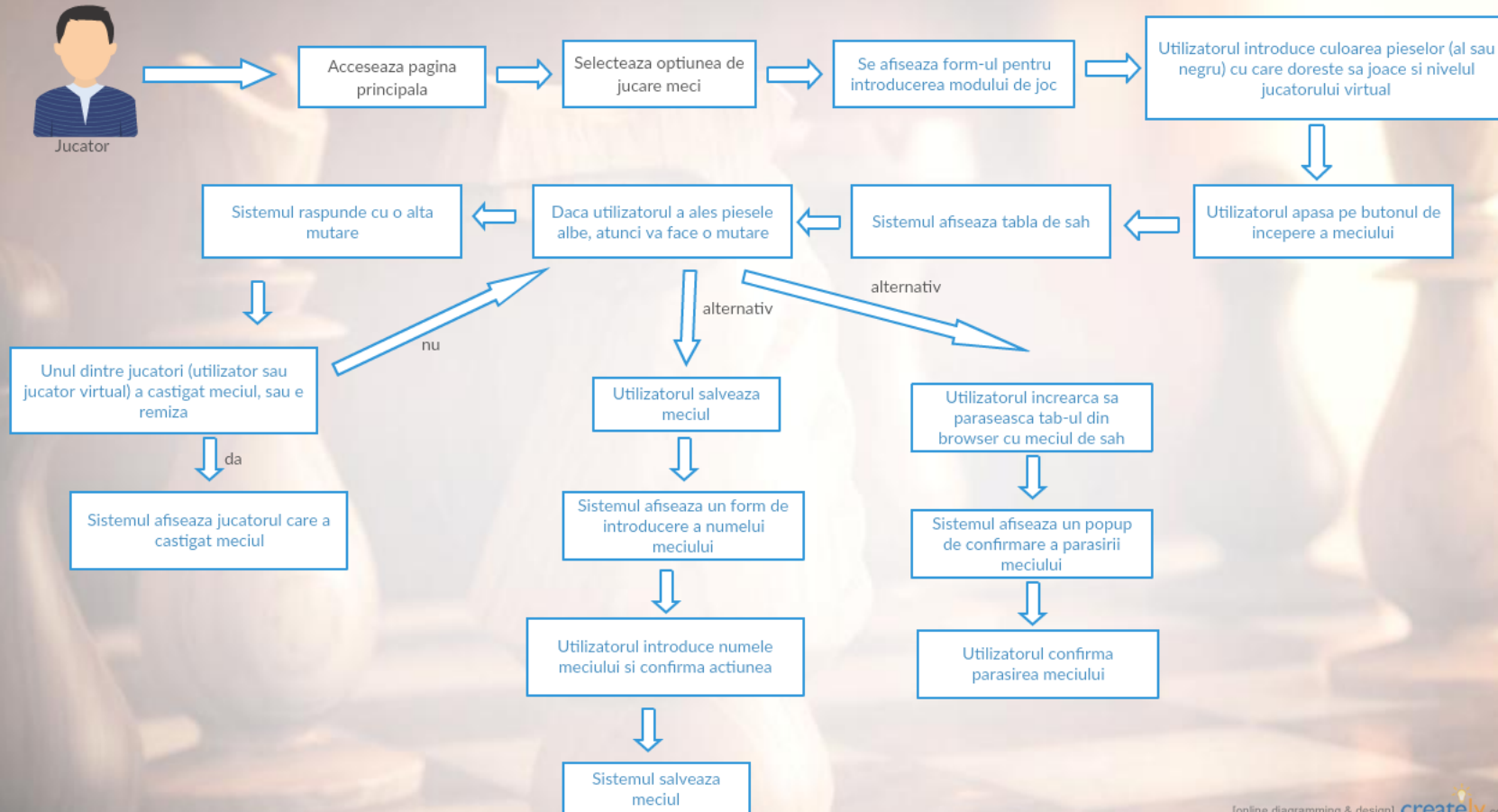


Cerinte functionale

1. Jucarea unui meci de sah
2. Vizualizare strategii joc
3. Testare cunoaștere strategii de joc
4. Vizualizare de statistici
5. Managementul meciurilor
6. Topul utilizatorilor
7. Managementul utilizatorilor



Cerinte functionale – flow



Cerinte nefunctionale

Utilizabilitatea

Gradul in care sistemul poate fi folosit de catre utilizatorii specificati pentru a atinge obiectivele dorite cu eficienta si satisfactie.

Performanta

Timpul de raspuns la paginile ce nu implica actiunile jucatorului virtual < 1 secunda

Timpul de raspuns la paginile ce implica jucatorul virtual < 15 secunde

Disponibilitatea

Timpul in care aplicatia este disponibila utilizatorilor: 99%.

Numarul minim de utilizatori concurenti suportat

Pentru o arhitectura cu un singur nod de procesare, sistemul trebuie sa suporte minim 20 de utilizatori concurenti.



Optiuni tehnologice

Tomcat

Container-ul WEB al aplicatiei

MySQL

Server-ul dedicat bazei de date

Maven

Suportul pentru construirea proiectului si management-ul bibliotecilor utilizate

Java

Limbajul de dezvoltare pe partea de server

Spring

Managementul obiectelor Java folosite

Spring MVC

Organizarea interfetei grafice a aplicatiei sub forma arhitecturii Model-View-Controller

Hibernate

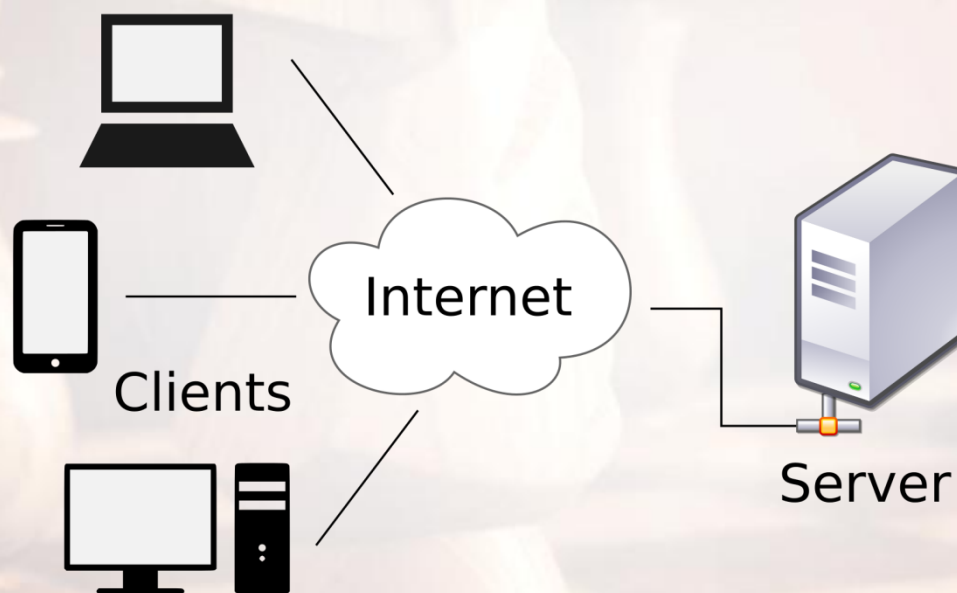
Maparea obiectelor java la tabele din baza de date

HTML, CSS, Java script, JQuery, Ajax

Interfata grafica si comunicarea cu serverul

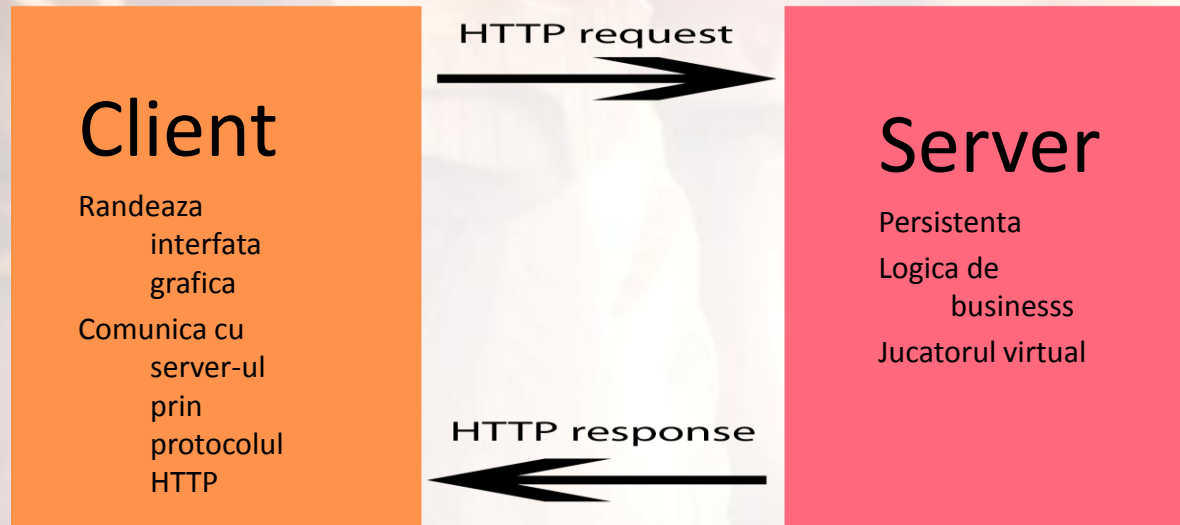


Arhitectura conceptuala

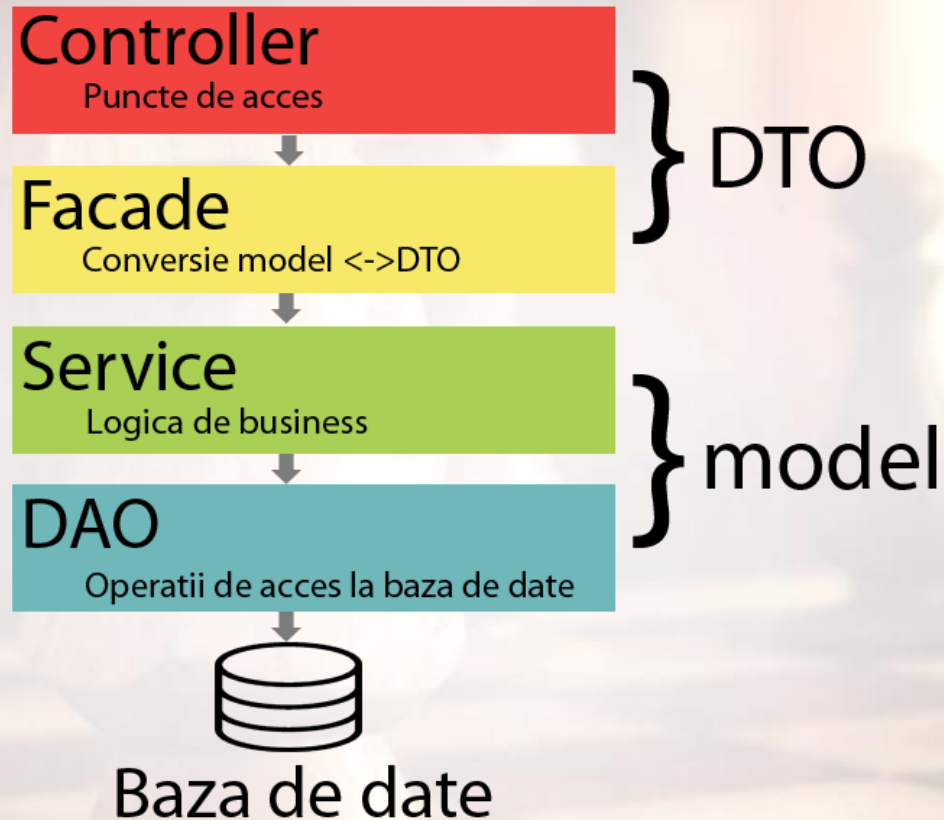


UNIVERSITATEA TEHNICĂ
DIN CLUJ-NAPOCA

Arhitectura conceptuala



Arhitectura server



Detalii de implementare

Comunicarea intre client si server se realizeaza utilizand protocolul HTTP



Detalii de implementare

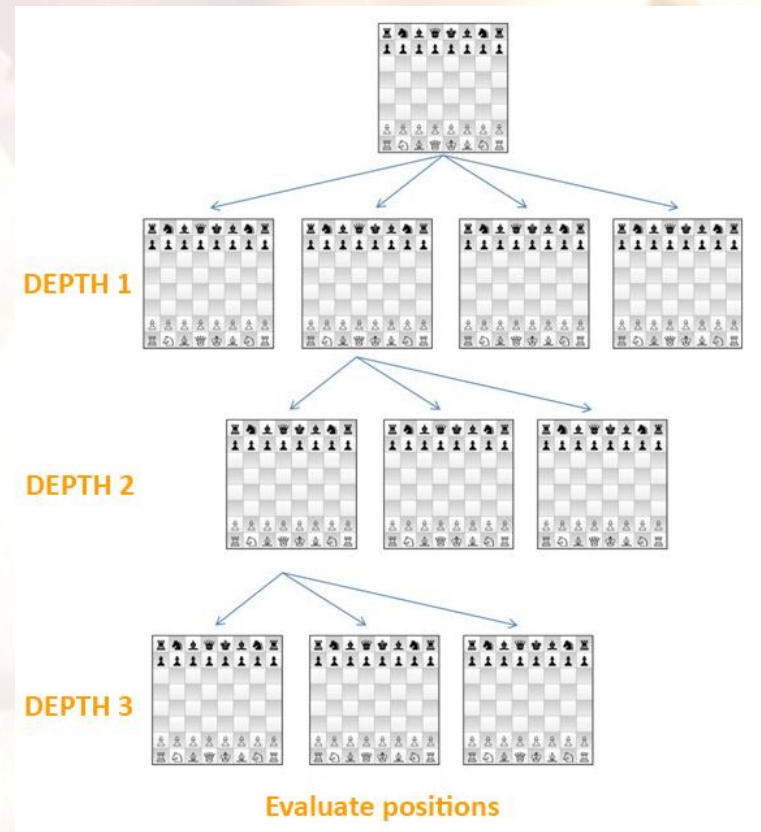
Evitarea introducerii datelor eronate pe server
se face folosind validare in layer-ul de
Controller



UNIVERSITATEA TEHNICĂ
DIN CLUJ-NAPOCA

Detalii de implementare jucator virtual

Algoritmul folosit pentru
generarea mutarii
urmatoare: MINIMAX



Detalii de implementare jucator virtual

Functia de evaluare se bazeaza pe urmatoarele criterii:

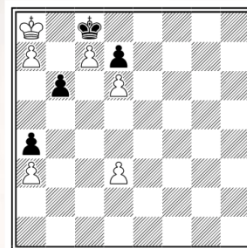
- Avantajul material
- Pozitia pieselor pe tabla



UNIVERSITATEA TEHNICĂ
DIN CLUJ-NAPOCA

Detalii de implementare jucator virtual

Reprezentarea
interna a unei pozitii
de sah



bitboard rege alb



bitboard pionii negrii



bitboard rege negru



Detalii de implementare jucator virtual

- Generarea mutarilor posibile pentru o anumta piesa: s-au folosit pozitii precalculate pentru a evita calcule redundante la runtime



Dificultati – rezolvari

Timpul de raspuns al jucatorului virtual – folosirea bitboard-urilor precomutate pentru generarea mutarii urmatoare.

Functia de evaluare – stabilirea criteriilor de evaluare a unei pozitii de sah in cadrul jucatorului virtual



Elemente de testare

Proces de dezvoltare iterativ -> testare manuala
la finalizarea implementarii unei componente



UNIVERSITATEA TEHNICĂ
DIN CLUJ-NAPOCA

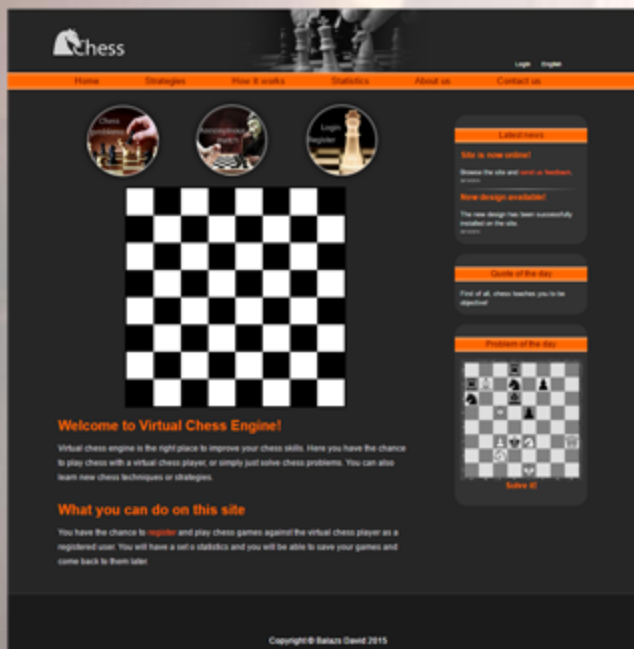
Elemente de testare

Testarea utilizabilitatii sistemului – sondaj facut pentru jucatorii experimentati dar si pentru jucatorii ocazionali, neexperimentati.



UNIVERSITATEA TEHNICĂ
DIN CLUJ-NAPOCA

Interfata grafica



UNIVERSITATEA TEHNICĂ
DIN CLUJ-NAPOCA

Concluzii

Sistemul reuseste sa isi atinga scopul, de a putea juca un meci de sah pe diferite nivele de dificultate cu timp de raspuns ce se incadreaza in cel stabilit in cadrul cerintelor nefunctionale

Sistemul ofera suport pentru de antrenament jucatorilor de sah prin expunerea de strategii de joc si probleme.



Dezvoltari viitoare

Extragerea jucatorului virtual intr-un serviciu WEB separat

Folosirea arhitecturii cluster pentru jucatorul virtual – instalarea serviciului WEB cu jucatorul virtual pe mai multe servere si folosirea unui load balancer pentru a obtine performanta dorita la un numar ridicat de jucatori ce utilizeaza aplicatia simultan.

Adaugarea unui dictionar de mutari pentru jucatorul virtual



Referinte bibliografice



UNIVERSITATEA TEHNICĂ
DIN CLUJ-NAPOCA