# Aplicație web pentru gestionarea orelor dintr-o școală de ski

Cuprins

[1. Introducere 1](#_Toc160527836)

[1.1 Scopul documentului 1](#_Toc160527837)

[2. Prezentare generală și abordări de proiectare 2](#_Toc160527838)

[2.1 Prezentare generală 2](#_Toc160527839)

[2.2 Presupuneri/ Constrângeri/ Riscuri 2](#_Toc160527840)

[2.2.1 Presupuneri 2](#_Toc160527841)

[2.2.2 Constrângeri 2](#_Toc160527842)

[2.2.3 Riscuri 3](#_Toc160527843)

[3. Considerațiii de proiectare 4](#_Toc160527844)

[3.1 Obiective și linii directoare (ghiduri) 4](#_Toc160527845)

[3.2 Metode de dezvoltare 4](#_Toc160527846)

[3.3 Strategii de arhitectură 4](#_Toc160527847)

[4. Arhitectura Sistemului și Proiectarea Arhitecturii 6](#_Toc160527848)

[4.1 Vedere logică 6](#_Toc160527849)

[4.2 Arhitectură hardware 6](#_Toc160527850)

[4.3 Arhitectură software 6](#_Toc160527851)

[4.4 Arhitectura informațiilor 7](#_Toc160527852)

[4.5 Arhitectura de comunicații interne 7](#_Toc160527853)

[4.6 Diagrama de arhitectură a sistemului 8](#_Toc160527854)

[5. Proiectarea sistemului 9](#_Toc160527855)

[5.1 Proiectarea bazei de date 9](#_Toc160527856)

[5.1.1 Obiecte de date și structuri de date rezultante 9](#_Toc160527857)

[5.1.2 Fișiere și baze de date 9](#_Toc160527858)

[5.2 Conversii de date 9](#_Toc160527859)

[5.3 Interfețe utilizator 10](#_Toc160527860)

[5.3.1 Intrări 10](#_Toc160527861)

[5.3.2 Ieșiri 10](#_Toc160527862)

[5.4 Proiectarea interfețelor cu utilizatorul 10](#_Toc160527863)

[6. Scenarii de utilizare 11](#_Toc160527864)

[7. Proiectare de detaliu 12](#_Toc160527865)

[7.1 Proiectare hardware de detaliu 12](#_Toc160527866)

[7.2 Proiectare software de deatliu 12](#_Toc160527867)

[7.3 Proiectare detaliată de securitate 13](#_Toc160527868)

[7.4 Proiectare de detaliu pentru performanța sistemului 14](#_Toc160527869)

[7.5 Proiectare detaliată a comunicațiilor interne (între componente) 14](#_Toc160527870)

[8. Controale pentru verificarea integrității sistemului 15](#_Toc160527871)

[Anexa A: Gestiunea modificărilor documentului 16](#_Toc160527872)

[Anexa B: Acronime 17](#_Toc160527873)

[Anexa C Documente la care se face referire 18](#_Toc160527874)

## Introducere

Acest document descrie designul sistemului pentru o aplicație de management al orelor la o școală de schi. Include o descriere a arhitecturii sistemului, a componentelor sale și a modului în care acestea interacționează. De asemenea, descrie cerințele funcționale și non-funcționale ale sistemului..

### Scopul documentului

Scopul acestui document este de a oferi o descriere detaliată a designului sistemului pentru a facilita dezvoltarea, implementarea și testarea sa. De asemenea, va servi ca referință pentru echipa de proiect și pentru părțile interesate.

## Prezentare generală și abordări de proiectare

Această secțiune descrie principiile și strategiile care vor fi utilizate ca ghiduri în momentul proiectării și implementării sistemului.

### Prezentare generală

Acest document va utiliza o abordare bazată pe arhitectură pentru a descrie designul sistemului. Arhitectura sistemului va fi descrisă la nivel înalt, cu detalii suplimentare furnizate pentru componentele cheie.

### Presupuneri/ Constrângeri/ Riscuri

#### Presupuneri

Următoarele presupuneri au fost făcute în timpul dezvoltării acestui design:

* Toți instructorii și elevii au acces la un dispozitiv cu internet.
* Există o bază de date existentă cu informații despre instructori și elevi.
* Instructorii și elevii au conturi de utilizator în sistem.

#### Constrângeri

Următoarele constrângeri se aplică designului sistemului:

* Sistemul trebuie să fie disponibil 24/7.
* Sistemul trebuie să fie sigur și să protejeze datele personale.
* Sistemul trebuie să fie ușor de utilizat pentru instructori și elevi.

#### Riscuri

Următoarele riscuri au fost identificate pentru designul sistemului:

* Erori de software pot duce la pierderea datelor.
* Atacurile cibernetice pot compromite securitatea sistemului.
* Utilizatorii pot avea dificultăți în utilizarea sistemului.

## Considerațiii de proiectare

### Obiective și linii directoare (ghiduri)

Următoarele obiective și linii directoare au fost stabilite pentru designul sistemului:

* Sistemul trebuie să permită instructorilor să gestioneze programul lor de predare.
* Sistemul trebuie să permită elevilor să rezerve lecții de schi cu instructorii preferați.
* Sistemul trebuie să permită administratorului școlii să gestioneze conturile de utilizator și să monitorizeze performanța sistemului.

### Metode de dezvoltare

Următoarele metode de dezvoltare vor fi utilizate pentru a implementa sistemul:

* Dezvoltare agilă
* Metodologia Scrum

### Strategii de arhitectură

Următoarele strategii de arhitectură vor fi utilizate pentru a dezvolta sistemul:

* Arhitectură web 3-tier
* Microservicii

## Arhitectura Sistemului și Proiectarea Arhitecturii

Instrucțiuni: Descrieți arhitectura sistemului, modul în care aplicația interacționează cu alte aplicații. Nu este neapărat nevoie să detaliați cum funcționează aplicația în sine, ci mai degrabă cum datele corespunzătoare sunt corect transferate între componente. Furnizați o prezentare generală a modului în care funcționalitatea și responsabilitățile sistemului au fost partitionate și apoi atribuite sub-sisteme sau componente. Nu intrați prea mult în detalii despre componentele individuale în această secțiune. O secțiune ulterioară a documentului va furniza descrieri detaliate ale componentelor. Scopul principal aici este de a obține o înțelegere generală a modului și motivelor pentru care sistemul a fost descompus și cum părțile individuale lucrează împreună pentru a furniza funcționalitatea dorită.

Descrieți cum sistemul a fost dezmembrat în componente/subsistemele sale (identificând fiecare componentă/subsistem de nivel superior și rolurile/responsabilitățile atribuite acestuia). Descrieți modul în care componentele de nivel superior colaborează între ele pentru a obține rezultatele dorite. Furnizați o anumită justificare pentru alegerea acestei decompuneri particulare a sistemului (poate discutați alte decompuneri propuse și motivele pentru care au fost respinse).

Faceți uz de tiparele de proiectare ori de câte ori este posibil, fie pentru a descrie părți ale arhitecturii (în format de tipar), fie pentru a face referire la elementele arhitecturii care le utilizează. Furnizați o justificare pentru alegerea unui algoritm sau idiom de programare (sau tipar de proiectare) pentru a implementa părți ale funcționalității sistemului.

Această secțiune conturează designul arhitecturii sistemului și hardware-ului.

### Vedere logică

Instrucțiuni: Introduceți orice vederi logice relevante sau furnizați o referință către locul în care sunt stocate.

### Arhitectură hardware

Instrucțiuni: Descrieți hardware-ul și organizația generală a sistemului, indicând dacă sistemul de procesare este distribuit sau centralizat. Identificați tipul, numărul și locația tuturor componentelor hardware, inclusiv serverele de prezentare, aplicație și date și orice dispozitive periferice (de exemplu, load balancers, acceleratori SSL, comutatoare, firewall-uri), cu o descriere succintă a fiecărui element și diagrame care arată conectivitatea între componente, împreună cu firewall-urile, porturile și benzile de rețea utilizate (de exemplu, banda de management). Includeți estimările resurselor pentru capacitatea procesorului, memoria, stocarea online și stocarea auxiliară.

### Arhitectură software

Instrucțiuni: Descrieți toate componentele software. Enumerați elemente logice precum componente (de exemplu, JSP în stratul de prezentare, JNDI în stratul de aplicație, EJB și JDBC în stratul de date), platforme de baze de date, limbaje de programare, compilatoare, utilitare, sisteme de operare, software de comunicații, instrumente de inginerie software asistate de calculator, produse software externe – software comercial disponibil pe piață, framework-uri open source etc., cu o scurtă descriere a funcției fiecărui element și a oricăror informații de identificare, cum ar fi producătorul, numărul versiunii, numărul și tipurile de licențe necesare etc., dacă este cazul. Identificați toate elementele de configurare a software-ului computerului și interfețele de programare a aplicațiilor (APIs), inclusiv numele, tipul, scopul și funcția pentru fiecare; interfețele, mesajele și protocoalele pentru acele elemente; și raționamentul pentru designul arhitectural al software-ului.

Includeți module de software care sunt funcții, subrutine sau clase. Utilizați diagrame ierarhice funcționale, diagrame de organizare structurată (de exemplu, diagrame de structură), sau diagrame orientate pe obiecte care arată diferitele niveluri de segmentare până la nivelul cel mai jos. Toate caracteristicile din diagrame ar trebui să aibă numere de referință și nume.

Dacă este necesar, descrieți cum o componentă a fost împărțită în subcomponente, precum și relațiile și interacțiunile dintre subcomponente. Continuați în cât mai multe niveluri/subsecțiuni de discuție cât este necesar pentru a oferi o înțelegere de nivel înalt a întregului sistem sau subsistem, lăsând detaliile pentru includerea într-o secțiune ulterioară a documentului. Includeți diagrame de flux de date și furnizează fluxul fizic al proceselor și datelor legate de fluxul logic al proceselor și datelor dezagregat la nivelul procesului primitiv (descriind cum fiecare intrare este procesată/transformată în ieșirea rezultată). Dacă există părți ale sistemului care au existat deja înainte de începerea acestui efort de dezvoltare, atunci descrieți doar relațiile și interacțiunile dintre părțile vechi și cele noi. Părțile preexistente care sunt modificate sau îmbunătățite trebuie descrise doar în măsura în care este necesar pentru a oferi o înțelegere suficientă a naturii modificărilor care se fac.

### Arhitectura informațiilor

Instrucțiuni: Descrieți informațiile care vor fi stocate în sistem (de exemplu, informații despre beneficiari, date despre cereri etc.). Identificați dacă vreuna dintre informații este informație cu caracter sensibil.

Identificați toate datele (precum și formatul datelor - hârtie, introducere manuală, date electronice) furnizate sistemului, precum și cine/ce furnizează datele.

### Arhitectura de comunicații interne

Instrucțiuni: Furnizați o descriere detaliată a rețelei de comunicații a sistemului, indicând arhitectura de comunicații implementată și modul în care componentele sistemului sunt conectate. Includeți descrieri ale echipamentelor necesare (de exemplu, hub-uri, routere, transmițătoare, module de comunicații, firewall-uri, porturi etc.). Furnizați o diagramă care să ilustreze fluxul de comunicații între componentele sistemului și subsisteme.

Includeți estimări ale resurselor necesare pentru capacitatea rețelei de comunicații (LAN și WAN) necesară pentru instalarea și executarea fiecărei aplicații pe fiecare platformă.

### Diagrama de arhitectură a sistemului

Instrucțiuni: Utilizând proiectarea hardware, software, de comunicații și de informații descrise mai sus, redați structura generală și integrată a sistemului.

## Proiectarea sistemului

### Proiectarea bazei de date

Model de date:

* Entități:
* Utilizator (ID, nume, prenume, email, parola, tip\_utilizator)
* Instructor (ID\_utilizator, specializare, experienta)
* Elev (ID\_utilizator, nivel\_schi)
* Lectie (ID, data, ora, durata, ID\_instructor, ID\_elev)

Relații:

* Un instructor poate avea mai multe lecții.
* Un elev poate avea mai multe lecții.
* lecție are un singur instructor.
* lecție are un singur elev.

Dicționar de date:

* ID: identificator unic
* Nume: nume complet al utilizatorului
* Prenume: prenumele utilizatorului
* Email: adresa de email a utilizatorului
* Parola: parola utilizatorului
* Tip\_utilizator: tipul de utilizator (instructor, elev, administrator)
* Specializare: specializarea instructorului (schi alpin, snowboard, etc.)
* Experienta: experiența instructorului (în ani)
* Nivel\_schi: nivelul elevului (începător, intermediar, avansat)
* Data: data lecției
* Ora: ora lecției
* Durata: durata lecției (în minute)

#### Obiecte de date și structuri de date rezultante

Utilizatori:

* Structură: Entitate
* Atribute:
  + ID (int, PK)
  + Nume (string)
  + Prenume (string)
  + Email (string)
  + Parola (string)
  + Tip\_utilizator (string)

Instructori:

* Structură: Entitate moștenită din Utilizatori
* Atribute suplimentare:
  + Specializare (string)
  + Experienta (int)

Elevi:

* Structură: Entitate moștenită din Utilizatori
* Atribute suplimentare:
  + Nivel\_schi (string)

Lecții:

* Structură: Entitate
* Atribute:
  + ID (int, PK)
  + Data (date)
  + Ora (time)
  + Durata (int)
  + ID\_instructor (int, FK)
  + ID\_elev (int, FK)

Structuri de date majore:

* Lista de utilizatori: O listă care conține toți utilizatorii din sistem.
* Lista de instructori: O listă care conține toți instructorii din sistem.
* Lista de elevi: O listă care conține toți elevii din sistem.
* Lista de lecții: O listă care conține toate lecțiile programate.
* Calendar: O reprezentare vizuală a programului de lecții.

#### Fișiere și baze de date

Model de date fizic:

* Sistemul va utiliza o bază de date MySQL pentru a stoca toate datele.
* Baza de date va avea următoarele tabele:
  + Utilizatori
  + Instructori
  + Elevi
  + Lecții

Structuri de fișiere:

* Fișierele de log vor fi utilizate pentru a înregistra toate activitățile din sistem.
* Fișierele de configurare vor fi utilizate pentru a stoca setările de sistem.

##### Baze de date

Tabelul Utilizatori:

* ID: Int, Primary Key
* Nume: String(255)
* Prenume: String(255)
* Email: String(255)
* Parola: String(255)
* Tip\_utilizator: String(255)

Tabelul Instructori:

* ID: Int, Primary Key, Foreign Key (References Utilizatori.ID)
* Specializare: String(255)
* Experienta: Int

Tabelul Elevi:

* ID: Int, Primary Key, Foreign Key (References Utilizatori.ID)
* Nivel\_schi: String(255)

Tabelul Lecții:

* ID: Int, Primary Key
* Data: Date
* Ora: Time
* Durata: Int
* ID\_instructor: Int, Foreign Key (References Instructori.ID)
* ID\_elev: Int, Foreign Key (References Elevi.ID)

##### Fișiere non-DBMS

Fișiere de log:

* Locație: /var/log/aplicatie
* Tip: Text
* Utilizare: Înregistrarea tuturor activităților din sistem
* Acces: Citire/scriere
* Module: Toate modulele

Fișiere de configurare:

* Locație: /etc/aplicatie
* Tip: INI
* Utilizare: Stocarea setărilor de sistem
* Acces: Citire/scriere
* Module: Toate modulele

### Conversii de date

Tipuri de conversii:

* Importul datelor despre instructori și elevi din baza de date existentă a școlii de schi.
* Conversia datelor din formatul existent în formatul compatibil cu sistemul nou.
* Validarea datelor pentru a se asigura acuratețea și completitudinea.

Instrumente și metode:

* Se vor utiliza scripturi SQL pentru a extrage datele din baza de date existentă.
* Se va utiliza un program de conversie de date pentru a converti datele în formatul nou.
* Se vor utiliza reguli de validare pentru a verifica acuratețea și completitudinea datelor.

Planificare:

* Conversiile de date vor fi efectuate înainte de lansarea sistemului nou.
* Se va efectua o testare completă a conversiilor de date pentru a se asigura că datele sunt transferate cu succes.

### Interfețe utilizator

Clase de utilizatori:

* Instructori
* Elevi
* Administrator

#### Intrări

Instructori:

* Formular de conectare
* Formular de înregistrare
* Calendar pentru programarea lecțiilor
* Formular de rezervare a lecțiilor
* Sistem de mesagerie

Elevi:

* Formular de conectare
* Formular de înregistrare
* Calendar pentru vizualizarea programului instructorilor
* Formular de rezervare a lecțiilor
* Sistem de mesagerie

Administrator:

* Formular de conectare
* Interfață de gestionare a utilizatorilor
* Interfață de monitorizare a performanței sistemului
* Interfață de generare a rapoartelor
* Interfață de configurare a setărilor sistemului

#### Ieșiri

Instructori:

* Profilul instructorului
* Lista lecțiilor programate
* Confirmarea rezervărilor
* Mesaje de notificare

Elevi:

* Profilul elevului
* Lista lecțiilor disponibile
* Confirmarea rezervărilor
* Mesaje de notificare

Administrator:

* Rapoarte despre utilizatori
* Rapoarte despre performanța sistemului
* Setări de sistem

### Proiectarea interfețelor cu utilizatorul

Principii de design:

* Ușurință de utilizare
* Interfață intuitivă
* Design receptiv (adaptabil la diferite dispozitive)
* Accesibilitate

Instrumente de design:

* Figma
* Adobe XD
* Prototipuri:
* Se vor crea prototipuri interactive pentru a testa și valida designul interfețelor.

Testare:

* Se vor efectua teste de utilizare cu utilizatori reali pentru a se asigura că interfețele sunt ușor de utilizat și intuitive.

## Scenarii de utilizare

Scenariu 1: Rezervare lecție de către un elev

Utilizator: Elev

Descriere:

* Elevul se conectează la sistem folosind formularul de conectare.
* Elevul accesează calendarul pentru a vizualiza programul instructorilor.
* Elevul selectează un instructor și o zi și oră dorită pentru lecție.
* Elevul completează formularul de rezervare a lecției, introducând detaliile personale și preferințele pentru lecție.
* Sistemul verifică disponibilitatea instructorului și validează datele introduse de elev.
* Dacă rezervarea este validă, sistemul o confirmă și trimite un email de confirmare elevului.
* Lecția este programată în calendarul instructorului.

Scenariu 2: Anulare lecție de către un elev

Utilizator: Elev

Descriere:

* Elevul se conectează la sistem folosind formularul de conectare.
* Elevul accesează profilul său și vizualizează lista lecțiilor programate.
* Elevul selectează lecția pe care dorește să o anuleze.
* Sistemul afișează o confirmare pentru anularea lecției.
* Elevul confirmă anularea lecției.
* Sistemul anulează rezervarea din calendarul instructorului și trimite un email de confirmare elevului.

Scenariu 3: Vizualizare program de către un instructor

Utilizator: Instructor

Descriere:

* Instructorul se conectează la sistem folosind formularul de conectare.
* Instructorul accesează profilul său și vizualizează programul de predare.
* Programul afișează detaliile lecțiilor programate, precum și numele elevilor.
* Instructorul poate accepta sau respinge rezervările de lecții.
* Instructorul poate marca lecțiile ca fiind finalizate.

Scenariu 4: Generare raport de către administrator

Utilizator: Administrator

Descriere:

* Administratorul se conectează la sistem folosind formularul de conectare.
* Administratorul accesează interfața de generare a rapoartelor.
* Administratorul selectează tipul de raport dorit (ex: raport cu lecțiile programate, raport cu elevii înscriși).
* Sistemul generează raportul în format PDF.
* Administratorul poate descărca și vizualiza raportul.

## Proiectare de detaliu

### Proiectare hardware de detaliu

Server: Dell PowerEdge R7525

* Procesor: AMD EPYC 7773X 64-core
* Memorie RAM: 256 GB DDR4 ECC
* Stocare: 2 x 8 TB HDD SATA

Stocare în rețea: Synology NAS DS1821+

* 8 bay-uri pentru HDD
* Capacitate totală de stocare: 48 TB

Switch de rețea: Cisco Catalyst 3750E

* 24 de porturi Gigabit Ethernet
* 2 porturi Gigabit Ethernet SFP

**Specificații:**

* **Server:**
  + Cerințe de alimentare: 800W
  + Impedanțe și stări logice ale semnalului: Conform specificațiilor PCIe 4.0
  + Specificații ale conectorilor: 2 x RJ-45 1GbE, 1 x RJ-45 10GbE
  + Specificații ale spațiului de memorie: 256 GB DDR4 ECC
  + Cerințe ale procesorului: AMD EPYC 7773X 64-core
* **Stocare în rețea:**
  + Cerințe de alimentare: 300W
  + Specificații ale conectorilor: 2 x RJ-45 1GbE
  + Specificații ale spațiului de stocare: 8 bay-uri pentru HDD SATA
* **Switch de rețea:**
  + Cerințe de alimentare: 450W
  + Specificații ale conectorilor: 24 x RJ-45 1GbE, 2 x SFP Gigabit Ethernet

### Proiectare software de deatliu

Servicii software:

* Serviciul de autentificare:
  + Identificator: AuthService
  + Clasificare: Serviciu de date
  + Definiție: Autentificarea utilizatorilor în sistem
  + Cerințe:
    - Să utilizeze un algoritm de criptare puternic pentru parole
    - Să permită autentificarea cu mai mulți factori
  + Structuri de date interne:
    - Utilizatori (nume, parolă, rol)
  + Constrângeri:
    - Parolele trebuie să aibă o lungime minimă de 8 caractere
    - Parolele trebuie să conțină cel puțin o majusculă, o minusculă și o cifră
  + Compoziție:
    - N/A
  + Utilizatori/Interacțiuni:
    - Interacționează cu interfața de conectare
    - Interacționează cu alte servicii care necesită autentificare
  + Procesare:
    - Verifică parola utilizatorului
    - Generează un jeton de autentificare
  + Interfețe/Exporturi:
    - Funcție de autentificare(nume, parolă)
    - Funcție de validare jeton(jeton)
* Serviciul de gestionare a programului:
  + Identificator: ScheduleService
  + Clasificare: Serviciu de aplicație
  + Definiție: Permite utilizatorilor să programeze și să gestioneze lecții
  + Cerințe:
    - Să permită utilizatorilor să vizualizeze programul instructorilor
    - Să permită utilizatorilor să rezerve lecții
    - Să permită instructorilor să accepte sau să respingă rezervările
  + Structuri de date interne:
    - Lecții (data, ora, durata, instructor, elev)
  + Constrângeri:
    - lecție nu poate fi rezervată de mai mulți elevi în același
* Utilizatori/Interacțiuni:
  + Interacționează cu interfața de vizualizare a programului
  + Interacționează cu interfața de rezervare a lecțiilor
  + Interacționează cu interfața de gestionare a rezervărilor (pentru instructori)
* Procesare:
  + Verifică disponibilitatea instructorului
  + Creează o rezervare pentru o lecție
  + Trimite notificări prin email pentru confirmări și anulări
* Interfețe/Exporturi:
  + Funcție de vizualizare program(instructor)
  + Funcție de rezervare lecție(data, ora, instructor)
  + Funcție de gestionare rezervări(instructor)
* Alte servicii software:
  + Serviciul de gestionare a utilizatorilor
  + Serviciul de notificări
  + Serviciul de raportare

### Proiectare detaliată de securitate

* Autentificare:
  + Utilizarea parolelor criptate
  + Autentificare cu mai mulți factori (opțional)
* Autorizare:
  + Controlul accesului bazat pe roluri
  + Permisiuni specifice pentru fiecare funcție a sistemului
* Jurnalizare și auditare:
  + Jurnalizarea detaliată a tuturor activităților din sistem
  + Auditarea regulată a jurnalelor de securitate
* Criptare:
  + Criptarea parolelor
  + Criptarea datelor sensibile (de exemplu, informații financiare)
* Utilizarea porturilor de rețea:
  + Utilizarea porturilor standard pentru HTTP, HTTPS și SSH
  + Restricționarea accesului la porturile neutilizate
* Detectare și prevenire a intruziunilor:
  + Implementarea unui sistem de detectare a intruziunilor (IDS)
  + Implementarea unui sistem de prevenire a intruziunilor (IPS)
* Cerințe/estimări de capacitate și volum:
  + Număr de utilizatori: 100
  + Număr de lecții pe zi: 50
  + Volum de stocare: 50 GB
* Așteptări de performanță:
  + Timpul de răspuns pentru autentificare: < 2 secunde
  + Timpul de răspuns pentru rezervarea unei lecții: < 5 secunde
  + Disponibilitatea sistemului: 99,9%
* Cerințe de disponibilitate:
  + Sistemul trebuie să fie disponibil 24/7
  + Timpul de nefuncționare planificat: < 1 oră pe an
* Proiectare de performanță:
  + Utilizarea unui server web performant
  + Utilizarea unei baze de date scalabile
  + Implementarea cache-ului pentru a reduce timpii de răspuns
  + Optimizarea codului pentru a îmbunătăți performanța
* Proiectare de fiabilitate:
  + Utilizarea componentelor hardware de încredere
  + Implementarea redundanței pentru componentele critice
  + Utilizarea unui sistem de backup și recuperare
* Proiectare de backup, recuperare și arhivare:
  + Backup zilnic complet al bazei de date
  + Backup incremental al bazei de date la fiecare oră
  + Arhivare lunară a datelor istorice
* Puncte unice de eșec:
  + Serverul principal
  + Baza de date
* Proiectare de disponibilitate ridicată:
  + Clustering pentru serverul principal
  + Replic
* Clustering pentru serverul principal:
  + Utilizarea unui cluster de servere pentru a oferi redundanță
  + În caz de eșec al unui server, celelalte servere din cluster vor prelua sarcina
* Replicare pentru baza de date:
  + Utilizarea replicării bazei de date pentru a oferi redundanță
  + În caz de eșec al bazei de date principale, baza de date secundară va prelua sarcina

### Proiectare detaliată a comunicațiilor interne (între componente)

* Numărul de servere și clienți:
  + 1 server principal
  + 1 server de rezervă
  + 20 de clienți (stații de lucru)
* Specificații pentru cerințele de sincronizare și control al busului:
  + Sincronizare NTP pentru a menține ceasurile tuturor serverelor și clienților sincronizate
  + Controlul accesului la magistrală pentru a preveni coliziunile de date
* Format(e) pentru datele schimbate între componente:
  + JSON
  + XML
* Topologia LAN:
  + Topologie stea
  + Toate serverele și clienții sunt conectați la un switch central
* Considerații suplimentare:
  + Securitate:
    - Implementarea unui firewall pentru a restricționa accesul la rețea
    - Utilizarea criptării pentru a proteja datele în tranzit
  + Performanță:
    - Optimizarea rețelei pentru a minimiza latența
    - Utilizarea unui switch de rețea performant

## Controale pentru verificarea integrității sistemului

* Securitate Internă:
  + Controlul accesului bazat pe roluri (RBAC):
    - Fiecare utilizator va fi asignat unui rol cu permisiuni specifice.
    - Accesul la datele critice va fi restricționat la rolurile care au nevoie de ele.
  + Autentificare:
    - Toți utilizatorii vor fi autentificați prin parole criptate sau autentificare cu mai mulți factori.
  + Criptare:
    - Datele critice vor fi criptate la rest și în tranzit.
  + Firewall:
    - Un firewall va fi implementat pentru a restricționa accesul la rețea.
* Proceduri de Audit:
  + Jurnalizare detaliată:
    - Toate activitățile din sistem vor fi jurnalizate, inclusiv accesul la datele critice.
    - Jurnalele vor fi stocate pentru o perioadă de timp specificată.
  + Auditare regulată:
    - Jurnalele vor fi auditate regulat pentru a identifica orice activitate suspectă.
  + Rapoarte de audit:
    - Rapoarte de audit vor fi generate periodic și distribuite părților interesate.
* Piste de Auditare a Aplicațiilor:
  + Piste de auditare detaliate:
    - Pistele de auditare vor fi create pentru toate accesările la datele critice.
    - Pistele de auditare vor include identificatorul utilizatorului, identificatorul terminalului de rețea, data, ora și datele accesate.
* Tabele Standard pentru Validarea Datelor:
  + Tabele standard:
    - Tabele standard vor fi utilizate pentru a valida câmpurile de date.
    - Tabelele standard vor include liste de valori valide și reguli de validare.
* Procese de Verificare:
  + Verificare încrucișată:
    - Datele critice vor fi verificate încrucișat cu alte surse de date pentru a se asigura acuratețea.
  + Reconciliere:
    - Datele critice vor fi reconcileate periodic pentru a se asigura integritatea.
* Identificarea Informațiilor de Auditare:
  + Informații de identificare:
    - Toate informațiile de auditare vor include identificatorul utilizatorului, identificatorul terminalului de rețea, data, ora și datele accesate.
  + Considerații suplimentare:
    - Securitate fizică:
      * Accesul la serverele care stochează datele critice va fi restricționat.
  + Backup și recuperare:
    - Datele critice vor fi salvate în mod regulat.
  + Plan de răspuns la incidente:
    - Un plan de răspuns la incidente va fi implementat pentru a face față incidentelor de securitate.