Sistem de Securitate si Confort al unei Locuinte

# Documentul de proiectare

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

**Numele Proiectului**: Sistem de Securitate și Confort al unei Locuințe (SSCL)

**Acronim**: SSCL

**Evoluția Documentului**: Acest Document de Proiectare a Sistemului reprezintă următorul pas în dezvoltarea proiectului SSCL, urmând Documentului de Cerințe. Documentul de Proiectare are ca scop transformarea cerințelor funcționale și non-funcționale stabilite în Documentul de Cerințe în specificații tehnice de proiectare a sistemului. Detaliază proiectarea sistemului la nivel înalt și oferă specificații detaliate pentru fiecare componentă a sistemului, inclusiv hardware, software, comunicații, interfețe om-mașină și interfețe externe.

**Descriere**:

Acest document descrie proiectarea sistemului SSCL pentru a îndeplini cerințele stabilite în Documentul de Cerințe. Proiectarea sistemului este prezentată la un nivel înalt, detaliind arhitectura generală a sistemului, proiectarea datelor, interfețele om-mașină și scenariile operaționale. Proiectarea sistemului este ulterior descompusă în specificații detaliate pentru fiecare componentă a sistemului, pentru a asigura implementarea corectă și eficientă a acestora.

**Considerații Referitoare la Securitate și Confidențialitate**:

În cadrul proiectului SSCL, se acordă o atenție deosebită securității și confidențialității datelor utilizatorilor. Proiectul implică utilizarea unor tehnologii și dispozitive conectate la internet, ceea ce ridică preocupări referitoare la securitatea informațiilor. Prin urmare, în timpul proiectării sistemului, se vor lua măsuri pentru a asigura protejarea datelor și a preveni accesul neautorizat la sistem. De asemenea, se va acorda atenție securității fizice a sistemului, pentru a preveni accesul neautorizat la dispozitivele hardware și infrastructura asociată.

### Scopul documentului

Scopul acestui Document de Proiectare a Sistemului (SDD) este de a oferi o îndrumare detaliată și bine definită asupra arhitecturii și designului sistemului SSCL. Acest document are ca scop principal să documenteze și să urmărească informațiile necesare pentru a defini în mod eficient arhitectura și designul sistemului, furnizând echipei de dezvoltare orientări clare și coerență în implementarea acestora.

Documentul de proiectare este un livrabil incremental și iterativ, adaptat pentru nevoile specifice ale proiectului SSCL și pentru metodologia de dezvoltare a sistemului utilizată. Este conceput pentru a evolua pe parcursul ciclului de viață al dezvoltării sistemului, astfel încât să reflecte în mod corespunzător schimbările și cerințele proiectului pe măsură ce acesta progresează.

Publicul țintă al acestui document include managerul de proiect, echipa de proiect și echipa de dezvoltare. Anumite părți ale documentului, cum ar fi interfața cu utilizatorul (UI), pot fi împărtășite și cu clientul/utilizatorul și cu alte părți interesate ale căror contribuții sau aprobări sunt necesare în ceea ce privește UI-ul.

Prin urmare, scopul acestui document este de a servi drept ghid detaliat și autoritativ pentru dezvoltarea și implementarea sistemului SSCL, asigurând că toate aspectele arhitecturii și designului sunt bine definite și înțelese de către toți membrii implicați în proiect.

## Prezentare generală și abordări de proiectare

### Prezentare generală

Sistemul de Securitate și Confort al unei Locuințe (SSCL) este conceput pentru a oferi o soluție integrată pentru gestionarea securității și a confortului într-o locuință. Abordarea proiectării se axează pe integrarea componentelor hardware și software pentru a crea un mediu sigur, confortabil și inteligent pentru utilizatori.

### Presupuneri/ Constrângeri/ Riscuri

#### Presupuneri

1. Compatibilitate Hardware: Se presupune că toate componentele hardware utilizate în cadrul sistemului SSCL sunt compatibile și funcționează corespunzător. Orice incompatibilități sau probleme de interoperabilitate trebuie să fie abordate în timpul fazelor de proiectare și implementare.
2. Stabilitatea Conexiunii la Internet: Se presupune că locuința în care va fi implementat sistemul SSCL are o conexiune stabilă la internet. Dependența de internet este esențială pentru funcționarea corectă a anumitor aspecte ale sistemului, cum ar fi controlul de la distanță și monitorizarea prin intermediul aplicației mobile și a interfeței web.
3. Utilizatorii Finali: Se presupune că utilizatorii finali ai sistemului SSCL au cunoștințe de bază în utilizarea tehnologiei și a dispozitivelor inteligente. Cu toate acestea, interfața utilizatorului trebuie să fie intuitivă și ușor de înțeles pentru a permite utilizatorilor să interacționeze cu sistemul fără dificultăți semnificative.
4. Securitatea Datelor: Se presupune că toate datele și informațiile sensibile colectate și stocate de sistemul SSCL sunt protejate împotriva accesului neautorizat și a interceptării. Măsuri de securitate, cum ar fi criptarea datelor și autentificarea utilizatorilor, sunt luate în considerare pentru a asigura confidențialitatea și integritatea datelor utilizatorului.
5. Evoluția Tehnologiei: Se presupune că tehnologiile utilizate în cadrul sistemului SSCL sunt în continuă evoluție și dezvoltare. Astfel, sistemul trebuie să fie flexibil și adaptabil la schimbările tehnologice pentru a rămâne relevant și funcțional pe termen lung. Strategii de actualizare și modernizare a sistemului sunt luate în considerare pentru a permite integrarea noilor tehnologii în viitor.

#### Constrângeri

1. Cerințe de Performanță: Sistemul SSCL trebuie să funcționeze în mod eficient și rapid pentru a asigura o experiență plăcută și fără întârzieri pentru utilizatori. Constrângerile de performanță pot afecta viteza de răspuns a aplicației mobile și a interfeței web, precum și timpul de reacție al dispozitivelor hardware în cadrul sistemului.
2. Cerințe de Securitate: Securitatea datelor și a infrastructurii este o constrângere critică pentru proiectul SSCL. Sistemul trebuie să respecte standardele și reglementările privind protecția datelor și să implementeze măsuri robuste de securitate pentru a preveni accesul neautorizat și a proteja confidențialitatea informațiilor utilizatorilor.
3. Cerințe de Interoperabilitate: Sistemul SSCL trebuie să fie compatibil cu diverse dispozitive hardware și software existente într-o locuință tipică. Constrângerile de interoperabilitate pot afecta integrarea și comunicarea corectă între diferitele componente ale sistemului, necesitând adoptarea standardelor și protocoalelor de comunicare comune.
4. Cerințe de Stocare și Distribuție a Datelor: Sistemul SSCL trebuie să gestioneze eficient stocarea și distribuția datelor utilizatorilor, precum și a informațiilor de configurare și de operare a sistemului. Constrângerile legate de capacitatea de stocare și de lățimea de bandă pot influența modul în care datele sunt colectate, procesate și transmise în cadrul sistemului.

Aceste constrângeri reprezintă aspecte critice ale proiectului SSCL și trebuie luate în considerare în proiectarea și implementarea sistemului pentru a asigura conformitatea cu cerințele și standardele stabilite.

#### Riscuri

1. Risc de Securitate Cibernetică: Unul dintre principalele riscuri asociate cu proiectul SSCL este expunerea la amenințările cibernetice, cum ar fi hacking-ul, malware-ul și phishing-ul. Pentru a reduce acest risc, se vor implementa măsuri stricte de securitate, cum ar fi criptarea datelor, autentificarea cu doi factori și monitorizarea constantă a activității suspecte.
2. Risc de Dependabilitate a Internetului: Sistemul SSCL depinde în mare măsură de o conexiune la internet stabilă pentru funcționarea sa corectă. Riscul de întrerupere sau degradare a conexiunii la internet poate afecta funcționalitatea sistemului. Pentru a reduce acest risc, se vor explora soluții de backup și redundanță pentru a menține conectivitatea în caz de probleme cu rețeaua principală.
3. Risc de Integritate a Datelor: Există riscul ca datele utilizatorilor să fie corupte sau compromise din cauza erorilor de sistem sau a atacurilor cibernetice. Pentru a reduce acest risc, se vor implementa proceduri de backup și de restaurare a datelor, precum și mecanisme de validare și verificare a integrității datelor.
4. Risc de Compatibilitate Hardware/Software: Integrarea componentelor hardware și software din diverse surse poate genera riscul de incompatibilitate sau conflict între diferitele elemente ale sistemului. Pentru a reduce acest risc, se vor efectua teste riguroase de compatibilitate și interoperabilitate în timpul fazelor de dezvoltare și integrare a sistemului.
5. Risc de Creștere a Costurilor și a Calendarului de Implementare: Orice întârzieri sau probleme întâmpinate în timpul dezvoltării sau implementării sistemului pot duce la creșterea costurilor și a timpului necesar finalizării proiectului. Pentru a reduce acest risc, se vor implementa practici de gestionare a proiectelor eficiente și se vor monitoriza în mod constant progresul și costurile asociate proiectului.

## Considerațiii de proiectare

### Obiective și linii directoare (ghiduri)

Obiectivele și linile directoare ale sistemului de Securitate și Confort al unei Locuințe (SSCL) sunt fundamentale pentru asigurarea unei proiectări și implementări eficiente și coerente a sistemului. Acestea sunt descrise mai jos:

**Eficiență și Performanță:**

Obiectivul principal este de a crea un sistem care să fie eficient din punct de vedere al performanței, oferind în același timp o experiență rapidă și receptivă pentru utilizatori. Prioritatea este de a minimiza timpul de răspuns și de a maximiza eficiența operațională a sistemului.

**Securitate și Confidențialitate:**

Securitatea datelor și a infrastructurii este o prioritate majoră. Toate aspectele sistemului, inclusiv interfețele de utilizator, comunicările și stocarea datelor, trebuie să fie protejate împotriva accesului neautorizat și a amenințărilor cibernetice.

**Ușurința în Utilizare:**

Interfața utilizatorului trebuie să fie intuitivă și ușor de înțeles pentru a asigura o experiență plăcută pentru utilizatori. Ghidurile de utilizare și elementele de design vor fi orientate către facilitarea accesului la funcționalitățile sistemului fără a necesita instruire prealabilă extensivă.

**Flexibilitate și Scalabilitate:**

Sistemul trebuie să fie flexibil și scalabil pentru a permite adăugarea și extinderea ulterioară a funcționalităților în funcție de nevoile și preferințele utilizatorilor. Arhitectura sistemului va fi proiectată în modul cel mai modular și versatil posibil pentru a permite adaptarea la schimbările viitoare.

**Compatibilitate și Interoperabilitate:**

SSCL trebuie să fie compatibil cu diverse dispozitive hardware și software existente într-o locuință tipică. Interoperabilitatea cu alte sisteme și produse similare este esențială pentru a permite utilizatorilor să integreze ușor SSCL în mediul lor existent.

### Metode de dezvoltare

Pentru designul sistemului și al software-ului în cadrul proiectului SSCL, vom adopta o abordare orientată pe obiecte, însoțită de utilizarea modelării în UML (Unified Modeling Language). Această metodă a fost aleasă pentru capacitatea sa de a organiza și structura sistemul într-un mod modular și eficient, ceea ce este esențial pentru un proiect complex precum SSCL.

Utilizarea modelării în UML va permite echipelor de dezvoltare să creeze diagrame clare și comprehensibile ale diferitelor aspecte ale sistemului, inclusiv structura hardware, interacțiunile software și fluxurile de date. Aceasta va facilita comunicarea între membrii echipei și va oferi un cadru clar pentru dezvoltarea, implementarea și testarea sistemului.

În ceea ce privește contingentele posibile, există riscul ca interfețele cu agențiile externe sau arhitecturile să fie instabile în momentul pregătirii documentului. Pentru a face față acestor situații, vom avea un plan de rezervă care să ne permită să ajustăm sau să adaptăm designul și implementarea sistemului în funcție de schimbările care apar. Acest plan va implica evaluarea continuă a situației și luarea deciziilor strategice în timp util pentru a menține progresul și coerența proiectului.

### Strategii de arhitectură

Pentru proiectul SSCL, au fost adoptate următoarele strategii de arhitectură pentru a asigura organizarea și structurarea eficientă a sistemului:

1. Utilizarea Arduino și ESP32: Decizia de a folosi platforme hardware cum ar fi Arduino și ESP32 a fost luată pentru a oferi o bază solidă și versatilă pentru implementarea sistemului. Aceste platforme oferă o gamă largă de funcționalități și sunt compatibile cu numeroase componente hardware necesare pentru proiectul SSCL.
2. Arhitectura Modulară și Scalabilă: Sistemul SSCL va fi proiectat cu o arhitectură modulară și scalabilă, permițând adăugarea facilă a noi funcționalități și extinderea sistemului în viitor. Această abordare va facilita adaptarea la schimbările tehnologice și cerințele utilizatorilor.
3. Utilizarea React Native pentru Aplicația Mobilă: Pentru dezvoltarea aplicației mobile, vom utiliza framework-ul React Native, care ne va permite să creăm o experiență de utilizare fluidă și consistentă pe diferite platforme mobile. Alegerea React Native se bazează pe obiectivul de a oferi o interfață utilizator intuitivă și ușor de utilizat pentru controlul sistemului SSCL.
4. Utilizarea Django pentru Baza de Date: Pentru gestionarea datelor și a bazei de date, vom utiliza framework-ul Django datorită robusteței și scalabilității sale. Alegerea Django va asigura o gestionare eficientă a datelor și o securitate sporită a informațiilor utilizatorilor.

Aceste strategii de arhitectură au fost alese în funcție de obiectivele și cerințele proiectului SSCL și vor servi drept fundament pentru dezvoltarea și implementarea eficientă a sistemului. Prin adoptarea acestor strategii, ne propunem să creăm un sistem flexibil, fiabil și ușor de utilizat pentru utilizatorii finali.

## Arhitectura Sistemului și Proiectarea Arhitecturii

### Vedere logică

În cadrul arhitecturii sistemului SSCL, vederea logică este reprezentată de structura modulară și interconectarea componentelor software și hardware care alcătuiesc sistemul. Această vedere este descrisă în detaliu în documentele de proiectare specifică a sistemului, inclusiv în diagramele UML și în specificațiile de interfață.

Pentru a asigura o înțelegere clară a vederii logice, se va elabora un document separat care va prezenta toate aspectele și interacțiunile sistemului SSCL într-un mod comprehensibil și accesibil. Acest document va include descrieri detaliate ale modulelor software, interfețelor de comunicare și a fluxurilor de date între diferitele componente.

Mai mult, vor fi create diagrame UML pentru a ilustra structura logică a sistemului, inclusiv diagrame de clasă pentru reprezentarea entităților software, diagrame de secvență pentru ilustrarea interacțiunilor între module și diagrame de activitate pentru a evidenția fluxurile de lucru și procesele sistemului.

Aceste vederi logice vor servi drept ghid pentru dezvoltatorii și proiectanții implicați în implementarea și integrarea sistemului SSCL, asigurând coerența și eficiența în proiectare și dezvoltare.

A diagram of a company

Description automatically generated

Img.1 Scema logica securitate

### Arhitectură hardware

Arhitectura hardware a sistemului SSCL este concepută pentru a oferi o structură fiabilă și scalabilă, care să susțină toate funcționalitățile și cerințele sistemului. Arhitectura este distribuită, cu componente hardware distribuite în întreaga locuință pentru a asigura o acoperire și o eficiență maximă.

Componentele hardware includ:

1. Arduino și ESP32: Aceste microcontrolere sunt utilizate pentru controlul diferitelor aspecte ale locuinței, cum ar fi sistemele de securitate, controlul temperaturii și umidității, controlul jaluzelelor și al incuietorilor.
2. Server de aplicație: Un server de aplicație va găzdui aplicația mobilă și va asigura comunicarea între aceasta și dispozitivele hardware din locuință. Acest server va fi dotat cu capacități adecvate de procesare și stocare pentru a gestiona volumul de date și interacțiunile cu utilizatorii.
3. Baza de date: O bază de date va stoca și gestiona toate datele relevante pentru funcționarea sistemului, inclusiv informații despre utilizatori, configurările sistemului și datele de senzori. Aceasta va fi configurată pe un server dedicat sau va folosi servicii de stocare în cloud pentru a asigura redundanța și accesibilitatea datelor.
4. Dispozitive Periferice: Acestea includ senzori, actuatori și alte dispozitive hardware necesare pentru funcționarea sistemului de securitate și confort al locuinței. Acestea sunt distribuite în întreaga locuință pentru a colecta date și a acționa în funcție de comenzi primite de la serverul de aplicație.

Diagrama de conectivitate va evidenția modul în care aceste componente sunt interconectate, precum și firewall-urile și porturile utilizate pentru comunicare. Estimările resurselor pentru fiecare componentă hardware vor fi furnizate pentru a asigura o dimensionare corespunzătoare a sistemului și pentru a îndeplini cerințele de performanță și scalabilitate.

A circuit board with wires and a display

Description automatically generated

Img.2 Arhitectura securitate

### Arhitectură software

Arhitectura software a sistemului SSCL este concepută pentru a oferi o structură modulară și flexibilă, care să faciliteze implementarea și gestionarea tuturor funcționalităților sistemului. Acesta cuprinde următoarele componente software:

Aplicație mobilă (React Native): O aplicație mobilă va fi dezvoltată folosind tehnologia React Native pentru a oferi utilizatorilor acces la funcționalitățile sistemului de securitate și confort al locuinței. Această aplicație va comunica cu serverul de aplicație pentru a prelua date și a trimite comenzi către dispozitivele hardware.

Server de aplicație (Django): Un server de aplicație va fi dezvoltat folosind framework-ul Django pentru a gestiona toate operațiunile backend ale sistemului. Acesta va gestiona autentificarea utilizatorilor, gestionarea datelor, comunicarea cu dispozitivele hardware și furnizarea de date către aplicația mobilă.

Bază de date (Django): O bază de date Django va fi utilizată pentru stocarea și gestionarea tuturor datelor sistemului, inclusiv date despre utilizatori, configurări și date de senzori. Această bază de date va fi accesată și gestionată de către serverul de aplicație pentru a asigura consistența și integritatea datelor.

APIs și interfețe de comunicare: Se vor dezvolta API-uri pentru a permite comunicația între aplicația mobilă, serverul de aplicație și dispozitivele hardware. Aceste interfețe vor folosi protocoale de comunicație standard pentru a facilita integrarea și interoperabilitatea între diferitele componente ale sistemului.

Diagrama arhitecturală va evidenția modul în care aceste componente software sunt interconectate și vor descrie fluxurile de date între ele. De asemenea, va include detalii despre modul în care aceste componente își îndeplinesc funcțiile și interacțiunile între ele.

### Arhitectura informațiilor

Sistemul va stoca o varietate de informații esențiale pentru funcționarea sa corectă și pentru a oferi utilizatorilor experiența dorită. Aceste informații includ:

* Datele contului utilizatorului: Acestea includ informații precum numele utilizatorului, adresa de e-mail, parola (criptată), preferințele și setările contului.
* Starea ușilor și ferestrelor: Informațiile despre starea ușilor și ferestrelor vor fi stocate pentru a monitoriza dacă acestea sunt deschise sau închise. Acest lucru ajută la asigurarea securității locuinței și la furnizarea de notificări către utilizatori în cazul unor situații neașteptate.
* Datele de temperatură și umiditate: Sistemul va colecta și stoca date despre temperatura și umiditatea din interiorul locuinței pentru a permite utilizatorilor să monitorizeze și să controleze climatul în interiorul casei.
* Sisteme presetate pentru lumină: Informațiile referitoare la sistemele presetate pentru lumină vor fi stocate pentru a permite utilizatorilor să configureze și să controleze iluminatul în diferite zone ale locuinței.
* Datele cartelelor acceptate de RFID: Pentru controlul accesului, sistemul va stoca informații despre cartelele RFID acceptate, inclusiv identificatorii acestora și drepturile de acces asociate.

Toate datele vor fi stocate în format electronic în baza de date MySQL asociată sistemului. Accesul la aceste date va fi restricționat și se va aplica criptarea pentru a asigura confidențialitatea și securitatea informațiilor, în special în cazul datelor cu caracter sensibil, cum ar fi datele de cont și informațiile despre accesul în locuință.

### Arhitectura de comunicații interne

Arhitectura de comunicații interne a sistemului este esențială pentru asigurarea funcționării corespunzătoare și a comunicării eficiente între diferitele componente. Sistemul utilizează o rețea locală (LAN) pentru comunicarea internă între dispozitivele conectate și o rețea extinsă (WAN) pentru accesul extern și comunicarea cu dispozitivele externe.

**Componente ale rețelei:**

1. **Router:** Un router este utilizat pentru gestionarea traficului între rețelele locale și externe. El asigură rutarea corectă a datelor către și dinspre internet și alte rețele externe.
2. **Switch-uri:** Switch-urile sunt utilizate pentru conectarea dispozitivelor locale în rețeaua LAN. Ele facilitează comunicația directă între diferitele dispozitive, cum ar fi Arduino, ESP32, camere de supraveghere și alte dispozitive conectate.
3. **Module de comunicații:** Modulele de comunicații sunt utilizate pentru conectarea dispozitivelor fără fir la rețea, inclusiv ESP32 și ESP32-CAM. Acestea permit comunicarea fără fir între dispozitive și baza de date centrală.
4. **Firewall:** Un firewall este implementat pentru a asigura securitatea rețelei împotriva accesului neautorizat și a atacurilor cibernetice.

**Resurse necesare pentru capacitatea rețelei:**

* **Lățimea de bandă:** Estimarea resurselor de lățime de bandă necesare pentru a asigura o comunicare fluentă între toate dispozitivele conectate, inclusiv transferul de date video și actualizările de stare.
* **Latenta:** Estimarea timpului de răspuns al rețelei, inclusiv întârzierile de comunicație între dispozitivele conectate și serverele de bază de date.
* **Fiabilitatea:** Evaluarea fiabilității și redundanței rețelei pentru a asigura funcționarea continuă a sistemului, chiar și în cazul unor eșecuri ale echipamentelor sau atacurilor cibernetice.

Top of Form

### Diagrama de arhitectură a sistemului

A diagram of a circuit board

Description automatically generated

Img.3 Diagrama de arhitectura

## Proiectarea sistemului

### Proiectarea bazei de date

Proiectarea bazei de date este responsabilă pentru definirea structurii și relațiilor dintre datele stocate în sistemul nostru. Aceasta include tipurile de date, cheile primare și străine, constrângerile de integritate și modalitățile de accesare a datelor. Scopul este de a asigura eficiența, siguranța și coerența datelor pe care sistemul le utilizează.

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

**Leduri**:

* ID Led (int) - identificator unic pentru fiecare LED
* Culoare (varchar) - culoarea asociată LED-ului (roșu, verde, albastru etc.)
* Stare (boolean) - indică dacă LED-ul este pornit sau oprit

**Uși**:

* ID Ușă (int) - identificator unic pentru fiecare ușă
* Stare (varchar) - indică dacă ușa este deschisă sau închisă
* Dată/Timp (timestamp) - momentul la care a fost înregistrată ultima modificare a stării ușii

**Temperatură**:

* ID Temp (int) - identificator unic pentru fiecare înregistrare a temperaturii
* Temperatură (decimal) - valoarea temperaturii măsurate
* Dată/Timp (timestamp) - momentul la care a fost înregistrată temperatura

**Securitate** **(RFID):**

* ID Card (int) - identificator unic pentru fiecare card RFID
* Cod Card (varchar) - codul unic asociat fiecărui card RFID
* Nume Utilizator (varchar) - numele utilizatorului asociat cardului RFID
* Permisiune Acces (varchar) - nivelul de permisiune al utilizatorului (de exemplu, administrator, utilizator standard etc.)

**Cont**:

* ID Utilizator (int) - identificator unic pentru fiecare utilizator al aplicației
* Nume Utilizator (varchar) - numele de utilizator al utilizatorului
* Parolă (varchar) - parola utilizatorului (criptată)
* Email (varchar) - adresa de email a utilizatorului
* Permisiuni (varchar) - nivelul de permisiune al utilizatorului în cadrul aplicației (de exemplu, utilizator standard, administrator etc.)

#### Fișiere și baze de date

Pentru stocarea și manipularea datelor în cadrul arhitecturii sistemului, vom avea următoarele structuri de fișiere și baze de date:

Baza de date principală:

Nume: "SistemSmartHomeDB"

Locație: Serverul central

Conținut: Va conține toate tabelele de date menționate anterior, precum și alte tabele necesare pentru funcționarea sistemului.

Tabelele de date:

Fiecare tabel menționat anterior va avea un fișier asociat în cadrul bazei de date principale.

Aceste fișiere vor fi stocate pe discul dur al serverului de bază de date.

Fișiere de configurare:

Nume: "config\_leds.ini", "config\_usi.ini", "config\_temperatura.ini", "config\_securitate.ini"

Locație: Pe serverul central sau pe dispozitivele locale, în funcție de necesități.

Conținut: Aceste fișiere vor conține setările de configurare pentru LED-uri, uși, temperatură și securitate. De exemplu, culorile disponibile pentru LED-uri, adresele IP ale dispozitivelor de temperatură, lista de cartele RFID acceptate etc.

Fișiere de jurnalizare (loguri):

Nume: "log\_sistem.txt", "log\_utilizatori.txt", "log\_acces\_usi.txt", etc.

Locație: Pe serverul central sau pe dispozitivele locale, în funcție de necesități.

Conținut: Aceste fișiere vor stoca înregistrările evenimentelor importante din sistem, precum accesul utilizatorilor, modificările stării ușilor sau temperaturii etc.

##### Baze de date

Proiectarea detaliată a fișierelor DBMS pentru baza de date principală "SistemSmartHomeDB" va include următoarele elemente:

Tabelul "Leduri":

Structura:

id\_led (cheie primară)

nume\_led

culoare

stare

Tabelul "Usi":

Structura:

id\_usa (cheie primară)

nume\_usa

stare

Tabelul "Temperatura":

Structura:

id\_temperatura (cheie primară)

temperatura\_actuala

data\_ultimei\_actualizari

Tabelul "Securitate":

Structura:

id\_cartela (cheie primară)

nume\_proprietar

cod\_cartela

stare\_valabilitate

Tabelul "Conturi":

Structura:

id\_cont (cheie primară)

utilizator

parola

nivel\_acces

Aceste tabele vor fi create în cadrul bazei de date "SistemSmartHomeDB", fiecare având un identificator unic (cheie primară) și câmpuri specifice care vor stoca informațiile relevante pentru gestionarea LED-urilor, ușilor, temperaturii, securității și conturilor utilizatorilor.

##### Fișiere non-DBMS

**Fișierul de configurare a LED-urilor:**

* Descriere: Acest fișier este utilizat pentru a stoca configurațiile LED-urilor, cum ar fi culorile și starea lor.
* Utilizare: Fișierul este utilizat pentru intrare și ieșire, deoarece conține informații despre configurațiile LED-urilor și poate fi actualizat de către utilizator sau de către sistem.
* Structura de fișier: Format text sau JSON, cu înregistrări structurate pentru fiecare LED, incluzând informații despre culoare și stare.

**Fișierul de jurnalizare a evenimentelor:**

* Descriere: Acest fișier este utilizat pentru a înregistra evenimentele importante care apar în sistem, cum ar fi accesul utilizatorilor sau schimbările de starea a ușilor.
* Utilizare: Fișierul este utilizat pentru ieșire și poate fi consultat pentru a urmări evenimentele înregistrate în sistem.
* Structura de fișier: Fișier de tip text, cu înregistrări timestamp și detalii despre evenimentele înregistrate.

**Fișierul de configurare a temperaturii:**

* Descriere: Acest fișier stochează setările de temperatură pentru activarea automată a ventilatorului sau a sistemului de încălzire/răcire.
* Utilizare: Fișierul este utilizat pentru intrare, deoarece conține configurațiile de temperatură stabilite de utilizator.
* Structura de fișier: Format text sau JSON, cu înregistrări structurate pentru fiecare setare de temperatură.

Aceste fișiere non-DBMS sunt esențiale pentru funcționarea sistemului și contribuie la gestionarea configurațiilor, înregistrarea evenimentelor și stabilirea parametrilor de temperatură.

### Conversii de date

Pentru proiectul nostru, nu sunt necesare conversii de date specifice sau documente suplimentare pentru acest aspect al sistemului.

### Interfețe utilizator

1. Pentru proiectul nostru, clasele de utilizatori includ:
2. Utilizatori Obișnuiți: Aceștia sunt utilizatorii finali care vor accesa aplicația pentru a controla luminile, a verifica starea ușilor, a seta temperaturile și a gestiona accesul cu cartele RFID. Se estimează un număr total de utilizatori obișnuiți de aproximativ 100 de persoane, cu un număr maxim de utilizatori simultani de 20.
3. Administratori: Aceștia sunt responsabili pentru administrarea și configurarea sistemului. Vor avea privilegii extinse pentru a gestiona utilizatorii, a monitoriza datele și a efectua întreținerea sistemului. Se estimează că vor exista aproximativ 5 administratori, dintre care doar unul va accesa simultan sistemul.
4. Suport Tehnic: Acest grup este responsabil pentru rezolvarea problemelor tehnice și pentru furnizarea de asistență utilizatorilor în caz de dificultăți. Numărul estimat de membri ai echipei de suport tehnic este de 3 persoane, dintre care unul sau doi pot accesa sistemul simultan pentru a oferi suport.
5. Personal de Întreținere: Aceștia sunt responsabili pentru mentenanța fizică a echipamentelor, inclusiv a senzorilor de temperatură, a luminilor LED și a sistemului de acces cu cartelă RFID. Numărul de personal de întreținere este estimat la 2 persoane. Accesul lor simultan la sistem este posibil, dar este limitat la nevoile de întreținere.

#### Intrări

1. Interfața Utilizatorului (UI) a aplicației mobile: Utilizatorii vor furniza informații sistemului prin intermediul unei aplicații mobile. Acesta va include ecrane de introducere a datelor pentru controlul luminilor, verificarea stării ușilor, setarea temperaturii și gestionarea accesului cu cartele RFID.
2. Ecrane de introducere a datelor: Acestea vor include butoane, comutatoare și casete de text pentru a permite utilizatorilor să interacționeze cu sistemul. De exemplu, un utilizator poate selecta o culoare pentru iluminare folosind o paletă de culori pe ecran.
3. Cititoare RFID: Pentru gestionarea accesului, utilizatorii vor folosi cititoare RFID pentru a introduce sau a scana cartelele lor de acces în sistem.
4. Sensori de temperatură: Pentru setarea temperaturii, utilizatorii pot interacționa cu senzorii de temperatură pentru a regla valorile dorite.

Criteriile de editare pentru datele introduse includ validarea tipului de date (de exemplu, verificarea dacă o valoare introdusă este numerică sau alfanumerică), limitarea intervalului de valori (de exemplu, temperatura trebuie să fie între anumite limite) și furnizarea de mesaje de eroare în cazul în care datele introduse sunt invalide sau lipsesc. Interfețele vor fi proiectate pentru a fi intuitive și ușor de utilizat, cu mesaje de feedback clare pentru a ghida utilizatorii în timpul interacțiunii.

#### Ieșiri

Pentru proiectul nostru, ieșirile sistemului includ:

1. Ecrane de afișare a datelor: Acestea prezintă informațiile sistemului utilizatorilor prin intermediul aplicației mobile. De exemplu, un ecran poate afișa starea actuală a luminilor, temperatura ambientală și starea ușilor.
2. Rapoarte de acces: Aceste rapoarte pot fi generate pentru a afișa istoricul accesului utilizatorilor cu cartele RFID. Ele pot conține informații precum data și ora accesului, identificatorul utilizatorului și tipul de acces (de exemplu, intrare sau ieșire).
3. Mesaje de notificare: Sistemul poate genera mesaje de notificare pentru a alerta utilizatorii cu privire la evenimente importante sau condiții anormale, cum ar fi o ușă lăsată deschisă sau o temperatură ambientală anormală.

Pentru fiecare ieșire, se vor specifica următoarele:

* Codurile și numele asociate cu fiecare ecran sau raport.
* Conținutul fiecărui ecran sau raport, inclusiv elementele de date asociate.
* Scopul fiecărei ieșiri și identificarea principalilor utilizatori ai acestora.
* Orice restricții de acces sau considerații de securitate asociate cu fiecare ieșire, cum ar fi necesitatea de autentificare pentru a vizualiza anumite informații.

### Proiectarea interfețelor cu utilizatorul

1. Interfața aplicației mobile: Această interfață permite utilizatorilor să controleze sistemul de monitorizare și control al locuinței lor folosind dispozitive mobile. Interfața va include ecrane pentru controlul luminilor, monitorizarea temperaturii, verificarea stării ușilor și gestionarea accesului prin intermediul cartelelor RFID.
2. Interfața web: Această interfață va fi accesată prin intermediul unui browser web și va oferi o vizualizare mai detaliată a datelor și a configurațiilor sistemului. Utilizatorii pot accesa și gestiona setările sistemului, precum și pot vizualiza rapoarte și istoricul evenimentelor.

## Scenarii de utilizare

**1. Activare/Dezactivare lumină automată:**

* Utilizatorul intră în locuință, iar senzorii de mișcare detectează prezența.
* Sistemul verifică dacă este întuneric în cameră.
* Dacă este întuneric, luminile sunt activate automat.
* După o perioadă predefinită de inactivitate, luminile se sting automat.

**2. Setarea temperaturii camerei:**

* Utilizatorul deschide aplicația mobilă sau interfața web și navighează la secțiunea de control al temperaturii.
* Utilizatorul selectează temperatura dorită pentru camera respectivă.
* Sistemul trimite comanda către termostatul inteligent din cameră.
* Termostatul ajustează temperatura camerei conform setării utilizatorului.

**3. Monitorizarea stării ușii:**

* Utilizatorul primește o notificare pe telefonul mobil că ușa de la intrare a fost deschisă.
* Utilizatorul accesează aplicația mobilă pentru a verifica starea ușii.
* Sistemul afișează starea ușii și oferă opțiuni pentru închiderea acesteia în cazul în care este necesar.

**4. Gestionarea accesului cu cartele RFID:**

* Utilizatorul adaugă o nouă cartelă RFID în sistem folosind interfața web.
* Sistemul solicită utilizatorului să introducă ID-ul cartelei RFID și să atribuie un nume asociat acesteia.
* După confirmare, cartelă RFID este adăugată în baza de date a sistemului și este activată pentru accesul în locuință.

**5. Autentificarea în aplicație:**

* Utilizatorul deschide aplicația mobilă și introduce numele de utilizator și parola.
* Sistemul verifică credențialele și autentifică utilizatorul.
* După autentificare, utilizatorul are acces la toate funcțiile și setările aplicației.

## Proiectare de detaliu

### Proiectare hardware de detaliu

Pentru proiectul nostru de monitorizare și control al locuinței, proiectarea hardware detaliată ar putea include următoarele componente:

Microcontroller sau Sistem de Procesare:

Model: Raspberry Pi 4 Model B

Alimentare: 5V DC

Conectori: HDMI, USB, GPIO

Spațiu de stocare: MicroSD Card

Cerințe ale procesorului: Quad-core ARM Cortex-A72 CPU

Senzori de Mișcare:

Model: PIR Motion Sensor HC-SR501

Alimentare: 5V DC

Stări logice ale semnalului: ON/OFF

Conectori: Pin de ieșire digitală

Distanță de detectare: Până la 7 metri

Termostat Inteligent:

Model: Nest Learning Thermostat

Alimentare: 24V AC

Compatibilitate: WiFi

Funcționalitate: Control automat al temperaturii

Cititor RFID:

Model: MFRC522

Alimentare: 3.3V DC

Stări logice ale semnalului: HIGH/LOW

Comunicare: SPI

Conectori: Antenă, pinuri de date

Ecrane de Afișare:

Model: LCD 16x2 Character Display

Alimentare: 5V DC

Interfață: I2C

Număr de caractere: 16x2

LED-uri pentru Iluminare:

Model: LED RGB Strips

Alimentare: 12V DC

Tip: SMD 5050

Control: Wireless (Bluetooth sau WiFi)

Funcționalitate: Schimbarea culorii și intensității luminii

Senzori de Ușă:

Model: Magnetic Reed Switch

Alimentare: 5V DC

Stări logice ale semnalului: Deschis/Închis

Conectori: Pin de ieșire digitală

Distanta de detectare: Până la 2 cm

Ventilator inteligent:

Model: Dyson Pure Cool TP04

Alimentare: 120V AC

Funcționalitate: Ventilare și purificare a aerului

Control: WiFi și aplicație mobilă

### Proiectare software de deatliu

**Serviciu de Control al Iluminatului:**

Identificator serviciu: LightControlService

Clasificare: Aplicație

Definiție: Acest serviciu controlează culorile și intensitatea luminii pentru LED-urile de iluminat din locuință.

Cerințe: Interfața utilizatorului pentru a selecta culorile și intensitatea luminii, integrare cu senzorii de lumină și programare a presetărilor de lumină.

Structuri de date interne: Lista de presetări de lumină, starea curentă a LED-urilor.

Constraingeri: Limitări de frecvență a actualizărilor culorilor și intensității.

Compoziție: Interacționează cu serviciul de monitorizare a luminii și cu interfața utilizatorului.

Utilizatori/Interacțiuni: Utilizatorul interacționează cu acest serviciu prin intermediul interfeței utilizatorului și a senzorilor de lumină.

Procesare: Utilizează algoritmi de conversie a culorilor și logica pentru a controla LED-urile de iluminat.

Interfețe/Exporturi: Metode pentru setarea culorii și intensității luminii.

**Serviciu de Monitorizare a Ușilor:**

Identificator serviciu: DoorMonitoringService

Clasificare: Serviciu de date

Definiție: Acest serviciu monitorizează starea ușilor din locuință (deschis/închis).

Cerințe: Detectarea stării ușilor, notificarea utilizatorului în cazul unei modificări a stării.

Structuri de date interne: Starea curentă a fiecărei uși.

Constraingeri: Frecvența de actualizare a stării ușilor.

Compoziție: Interacționează cu interfețele utilizatorului și cu alte servicii de control.

Utilizatori/Interacțiuni: Interfața utilizatorului interacționează cu acest serviciu pentru a primi actualizări despre starea ușilor.

Procesare: Monitorizează continuu starea ușilor și generează notificări în funcție de modificările detectate.

Interfețe/Exporturi: Metode pentru accesarea stării ușilor.

**Serviciu de Control al Temperaturii:**

Identificator serviciu: TemperatureControlService

Clasificare: Aplicație

Definiție: Acest serviciu controlează activarea și dezactivarea ventilatorului în funcție de temperatura setată de utilizator.

Cerințe: Setarea temperaturii dorite, integrare cu senzorii de temperatură.

Structuri de date interne: Temperatura setată de utilizator, temperatura actuală măsurată de senzorul de temperatură.

Constraingeri: Limitări ale temperaturilor minime și maxime acceptate.

Compoziție: Interacționează cu senzorii de temperatură și interfața utilizatorului.

Utilizatori/Interacțiuni: Utilizatorul setează temperatura dorită prin intermediul interfeței utilizatorului.

Procesare: Verifică temperatura actuală și activează/dezactivează ventilatorul în consecință.

Interfețe/Exporturi: Metode pentru setarea temperaturii și monitorizarea temperaturii actuale.

Aceste servicii acoperă funcționalitățile esențiale ale sistemului nostru de monitorizare și control al locuinței și oferă o bază solidă pentru dezvoltarea și integrarea ulterioară a altor funcționalități. Detaliile complete despre aceste servicii pot fi documentate în antetele și comentariile codului sursă, precum și în diagramele și documentele de proiectare software.

### Proiectare detaliată de securitate

Autentificare:

Implementarea unui sistem de autentificare bazat pe parole pentru accesul la aplicația de control a locuinței.

Utilizarea unui mecanism de autentificare cu factori multipli pentru a adăuga un nivel suplimentar de securitate.

Autorizare:

Definirea unor roluri și permisiuni clare pentru diferitele funcționalități ale aplicației.

Asigurarea că doar utilizatorii autorizați au acces la anumite funcționalități sau date sensibile.

Jurnalizare și auditare:

Implementarea unui sistem robust de jurnalizare a activităților pentru a înregistra toate acțiunile utilizatorilor și evenimentele de sistem.

Asigurarea că jurnalele sunt protejate împotriva modificărilor neautorizate și sunt disponibile doar pentru administratorii de sistem autorizați.

Criptare:

Utilizarea criptării pentru protejarea datelor sensibile, cum ar fi parolele utilizatorilor și datele de autentificare.

Implementarea unei conexiuni securizate cu dispozitivele periferice pentru a proteja datele transmise prin rețea.

Utilizarea porturilor de rețea:

Configurarea corectă a porturilor de rețea pentru a permite accesul numai de la dispozitive autorizate și pentru a bloca traficul nedorit.

Utilizarea unor mecanisme de control al accesului la nivel de port pentru a preveni atacurile de tip port scanning și spoofing.

Detectare și prevenire a intruziunilor:

Implementarea unui sistem de detecție a intruziunilor pentru a monitoriza activitatea rețelei și a sistemului și a detecta eventualele anomalii sau atacuri.

Configurarea unor firewall-uri și filtre de acces pentru a bloca traficul nedorit și pentru a preveni intruziunile.

Aceste componente de securitate hardware sunt esențiale pentru protejarea integrității, confidențialității și disponibilității sistemului nostru de monitorizare și control al locuinței. Implementarea acestora în mod corespunzător va asigura că sistemul este rezistent la atacuri și că datele utilizatorilor sunt protejate în mod eficient.

### Proiectare de detaliu pentru performanța sistemului

Componenta Hardware - Server:

Cerințe de capacitate și volum: Serverul trebuie să gestioneze simultan accesul și procesarea datelor de la multiple dispozitive periferice și să asigure o performanță optimă în timp real.

Așteptări de performanță: Se așteaptă ca serverul să ofere un timp de răspuns rapid și să suporte un volum mare de solicitări simultane.

Cerințe de disponibilitate: Serverul trebuie să fie disponibil în mod continuu pentru a asigura funcționarea neîntreruptă a sistemului.

Proiectare de performanță și fiabilitate: Serverul va fi echipat cu procesoare puternice și cu suficientă memorie RAM pentru a gestiona sarcinile cu eficiență. De asemenea, va fi configurat cu o soluție de stocare RAID pentru a asigura redundanța și pentru a preveni pierderea datelor în caz de defectare a discului.

Backup, recuperare și arhivare: Se va implementa o strategie automată de backup pentru datele critice, iar datele vor fi arhivate periodic pentru a asigura disponibilitatea și integritatea acestora.

Componenta Hardware - Dispozitive Periferice:

Cerințe de capacitate și volum: Dispozitivele periferice trebuie să poată trimite date către server și să primească comenzi de la acesta în timp real.

Așteptări de performanță: Se așteaptă ca dispozitivele să răspundă prompt la comenzile primite și să transmită datele fără întârzieri semnificative.

Cerințe de disponibilitate: Dispozitivele trebuie să fie disponibile în mod continuu pentru a permite monitorizarea și controlul eficient al locuinței.

Proiectare de performanță și fiabilitate: Dispozitivele vor fi proiectate și fabricate pentru a rezista la condiții de mediu variabile și pentru a funcționa în mod fiabil în timp îndelungat.

Backup, recuperare și arhivare: Datele importante generate de către dispozitive vor fi stocate și backup-ate în mod regulat pentru a preveni pierderea acestora în caz de defectare sau incident.

Componenta Hardware – Rețea:

Cerințe de capacitate și volum: Rețeaua trebuie să suporte un volum mare de trafic de date între dispozitivele periferice și server, asigurând o latență minimă.

Așteptări de performanță: Se așteaptă ca rețeaua să ofere o viteză de transfer ridicată și o conectivitate stabilă între toate componentele sistemului.

Cerințe de disponibilitate: Rețeaua trebuie să fie disponibilă în mod continuu pentru a asigura comunicarea neîntreruptă între dispozitive și server.

Proiectare de performanță și fiabilitate: Rețeaua va fi configurată cu echipamente de calitate și va utiliza protocoale și tehnologii care să optimizeze performanța și fiabilitatea, cum ar fi switch-urile redundante și protocolul Spanning Tree pentru a preveni buclele de rețea.

Backup, recuperare și arhivare: Configurația rețelei va include un mecanism automat de backup pentru a asigura că datele de configurare și log-urile sunt protejate împotriva pierderii în caz de defectare sau incident.

Puncte de eșec și proiectare de disponibilitate ridicată:

Punctele de eșec pot include defectarea serverului principal, a unui switch de rețea critic sau a unui dispozitiv periferic important.

Pentru a asigura o disponibilitate ridicată

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

Numărul de servere și clienți pe fiecare rețea de zonă:

Avem un server central care gestionează și coordonează toate operațiunile sistemului. Acesta va fi conectat la toate dispozitivele periferice din rețea.

Fiecare dispozitiv periferic (LED-uri, senzori de uși, senzori de temperatură, etc.) va acționa ca un client care comunică cu serverul central pentru a trimite date și a primi comenzi.

Specificații pentru cerințele de sincronizare și control al busului:

Comunicațiile între server și clienți vor fi gestionate printr-un protocol de comunicație standard, cum ar fi TCP/IP sau MQTT, pentru a asigura sincronizarea și controlul adecvat al fluxului de date.

Format(e) pentru datele schimbate între componente:

Datele schimbate între componente vor fi în format digital, utilizând protocoale de comunicație adecvate pentru fiecare tip de dispozitiv și funcționalitate.

Reprezentare grafică a conectivității între componente:

Serverul central va fi conectat direct la fiecare dispozitiv periferic prin intermediul unei rețele LAN. Conexiunile vor fi realizate fie prin cabluri Ethernet, fie prin tehnologii wireless, în funcție de configurarea fizică a sistemului.

Fiecare dispozitiv periferic va fi echipat cu o interfață de rețea (Ethernet sau Wi-Fi) pentru a permite comunicarea bidirecțională cu serverul central.

Topologia LAN:

Topologia rețelei locale (LAN) va fi de tip staționară, cu serverul central plasat central și dispozitivele periferice conectate direct la acesta. Aceasta va asigura un timp de răspuns redus și o comunicare eficientă între componente.

## Controale pentru verificarea integrității sistemului

Pentru a asigura integritatea și securitatea sistemului nostru, următoarele controale sunt necesare:

Securitate internă:

Implementarea unui sistem de gestionare a drepturilor de acces (RBAC) pentru a restricționa accesul la datele critice doar pentru utilizatorii sau rolurile care necesită acest acces.

Criptarea datelor sensibile pentru a proteja informațiile în tranzit și în repaus.

Proceduri de audit:

Dezvoltarea unor proceduri de audit care să respecte cerințele de control și raportare, incluzând perioadele de reținere a rapoartelor operaționale și de management.

Implementarea unui sistem de jurnalizare robust pentru a înregistra toate acțiunile utilizatorilor și modificările de date.

Piste de auditare a aplicațiilor:

Crearea de piste de auditare a aplicațiilor pentru a monitoriza dinamic accesul și modificările la datele critice designate, inclusiv identificarea utilizatorului, terminalului de rețea, dată, oră și datele accesate sau modificate.

Tabele standard pentru validarea câmpurilor de date:

Definirea de tabele standard care să fie utilizate pentru validarea câmpurilor de date, astfel încât să se asigure integritatea datelor introduse în sistem.

Procese de verificare a datelor critice:

Implementarea unor procese de verificare riguroase pentru adăugarea, ștergerea sau actualizarea datelor critice, inclusiv autentificare și autorizare adecvată.

Capacitatea de identificare a informațiilor de auditare:

Dezvoltarea unui sistem robust de identificare a informațiilor de auditare, care să permită identificarea utilizatorului, terminalului de rețea, dată, oră și datele accesate sau modificate în mod precis și fiabil.

Anexa A: Gestiunea modificărilor documentului

Tabel 1 – Înregistrarea modificărilor asupreaa documentului curent

| versiune | Data | Autorul/Deținătorul | Descriere |
| --- | --- | --- | --- |
| <0.8> | <14.04.2024> | Albert-Dorin | Completat capitol 1-3 |
| <0.9> | <15.04.2024> | Albert-Dorin | Completat capitol 4-6 |
| <1.0> | <16.04.2024> | Albert-Dorin | Completat intreg documentul |

Anexa B: Acronime

Tabel 2 - Acronime

| Acronim | Forma completă |
| --- | --- |
| <SSCL> | Sistem de Securitate si Confort al unei Locuinte |
| <Acronim> | <Forma completă> |
| <Acronim> | <Forma completă> |

Anexa C Documente la care se face referire

Tabel 3 – Documente la care se facce referire

| Nume document | Locație sau URL | Dată emitere document |
| --- | --- | --- |
| < Documentatie Django> | <https://docs.djangoproject.com/> | <16/04/2024> |
| < Documentatie React Native > | <https://reactnative.dev> | <16/04/2024> |
| < Nume document > | <Locație sau URL> | <ZZ/LL/AAAA> |