# Implementarea unui sistem avansat de control pentru gestionarea rezervarilor in industria hoteliera

# Documentul de proiectare

Cuprins

[1. Introducere 3](#_Toc164072905)

[1.1 Scopul documentului 3](#_Toc164072906)

[2. Prezentare generală și abordări de proiectare 4](#_Toc164072907)

[2.1 Prezentare generală 4](#_Toc164072908)

[2.2 Constrângeri 4](#_Toc164072909)

[3. Considerațiii de proiectare 6](#_Toc164072910)

[3.1 Obiective și linii directoare (ghiduri) 6](#_Toc164072911)

[3.2 Metode de dezvoltare 6](#_Toc164072912)

[3.3 Strategii de arhitectură 7](#_Toc164072913)

[4. Arhitectura Sistemului și Proiectarea Arhitecturii 8](#_Toc164072914)

[4.1 Vedere logică 10](#_Toc164072915)

[4.2 Arhitectură hardware 10](#_Toc164072916)

[4.3 Arhitectură software 11](#_Toc164072917)

[4.4 Arhitectura informațiilor 11](#_Toc164072918)

[4.5 Diagrama de arhitectură a sistemului 12](#_Toc164072919)

[5. Proiectarea sistemului 13](#_Toc164072920)

[5.1 Proiectarea bazei de date 13](#_Toc164072921)

[5.2 Interfețe utilizator 14](#_Toc164072922)

[5.2.1 Intrări 14](#_Toc164072923)

[5.2.2 Ieșiri 14](#_Toc164072924)

[5.3 Proiectarea interfețelor cu utilizatorul 15](#_Toc164072925)

[6. Scenarii de utilizare 16](#_Toc164072926)

[7. Proiectare de detaliu 17](#_Toc164072927)

[7.1 Proiectare detaliată a comunicațiilor interne (între componente) 17](#_Toc164072928)

## Introducere

Documentul de Proiectare a Sistemului este dedicat sistemului automat de gestionare a rezervărilor și check-in-ului în regim hotelier. Scopul acestui document este de a transforma cerințele funcționale și non-funcționale, definite anterior, în specificații tehnice detaliate care vor ghida procesul de dezvoltare și implementare a sistemului.

În acest document, vor fi abordate următoarele aspecte:

* Obiectivele și considerațiile de proiectare
* Arhitectura generală a sistemului
* Proiectarea bazei de date
* Interfața cu utilizatorul
* Scenarii operaționale

De asemenea, proiectarea sistemului va fi descompusă în specificații detaliate pentru fiecare componentă majoră, inclusiv:

* Hardware (Raspberry Pi, ecran tactil, cititor RFID, Arduino, motor pas cu pas)
* Comunicații interne (între Raspberry Pi, Arduino și baza de date)
* Software (aplicația Python, baza de date MySQL, programul Arduino)
* Interfețe externe (cu sistemele hotelului, cu utilizatorii)

### Scopul documentului

Acest document va servi ca un ghid comprehensiv pentru echipa de dezvoltare, asigurând că sistemul este construit și implementat într-un mod eficient și în conformitate cu cerințele și obiectivele stabilite. Pe măsură ce proiectul avansează, documentul poate fi actualizat și rafinat pentru a reflecta orice modificări sau decizii de proiectare suplimentare.

Prin urmarea specificațiilor din acest document, sistemul automat de gestionare a rezervărilor și check-in-ului va fi dezvoltat pentru a îmbunătăți experiența oaspeților, a eficientiza operațiunile hotelului și a asigura o integrare fără probleme cu sistemele existente.

## Prezentare generală și abordări de proiectare

### Prezentare generală

Sistemul automat de gestionare a rezervărilor și check-in-ului este conceput pentru a optimiza procesul de cazare în regim hotelier, oferind oaspeților o modalitate rapidă, convenabilă și independentă de a-și confirma rezervările și de a obține acces în camere. Principalele funcționalități ale sistemului includ validarea codurilor de rezervare existente, crearea de noi rezervări direct la automat (în cazul disponibilității), inscripționarea cardurilor de acces și simularea deblocării ușilor camerelor.

Arhitectura sistemului implică o integrare strânsă între componente hardware (Raspberry Pi, ecran tactil, cititor RFID, Arduino, motor pas cu pas) și software (aplicația Python, baza de date MySQL). Raspberry Pi, echipat cu un ecran tactil, servește ca interfață principală cu utilizatorul, permițând oaspeților să introducă coduri de rezervare sau să creeze rezervări noi. Cititorul RFID este utilizat pentru inscripționarea cardurilor de acces, iar Arduino și motorul pas cu pas simulează funcționarea încuietorilor electronice ale ușilor.

Aplicația Python rulează pe Raspberry Pi și gestionează logica de funcționare a sistemului, interacționând cu baza de date MySQL pentru a valida rezervările și a actualiza disponibilitatea camerelor. De asemenea, aplicația controlează cititorul RFID pentru inscripționarea cardurilor și comunică cu Arduino pentru a simula deblocarea ușilor.

Obiectivul principal al proiectării sistemului este de a asigura o funcționare fiabilă, eficientă și ușor de utilizat, care să îmbunătățească experiența oaspeților și să optimizeze fluxul de lucru al personalului de la recepție. Proiectarea se concentrează pe crearea unei interfețe intuitive pentru utilizatori, pe integrarea fără probleme cu sistemele existente ale hotelului și pe asigurarea securității și integrității datelor.

Prin implementarea acestui sistem, hotelul va beneficia de o reducere a timpilor de așteptare la recepție, o îmbunătățire a satisfacției oaspeților și o eficientizare a procesului de cazare. De asemenea, sistemul va colecta date valoroase despre comportamentul și preferințele oaspeților, care pot fi utilizate pentru optimizarea operațiunilor hoteliere și pentru luarea deciziilor de afaceri.

### Constrângeri

Proiectarea și implementarea sistemului automat de gestionare a rezervărilor și check-in-ului se confruntă cu câteva constrângeri, atât din perspectivă hardware, cât și software:

Constrângeri hardware:

- Dimensiunea și poziționarea automatului: Automatul trebuie să fie suficient de compact pentru a se integra în zona de recepție a hotelului, fără a obstrucționa traficul sau a ocupa un spațiu excesiv. În același timp, acesta trebuie să fie poziționat într-un loc ușor accesibil și vizibil pentru oaspeți.

- Fiabilitatea componentelor: Componentele hardware utilizate, cum ar fi Raspberry Pi, ecranul tactil, cititorul RFID și motorul pas cu pas, trebuie să fie fiabile și durabile pentru a face față utilizării intense în mediul hotelier. Acestea trebuie să funcționeze corect și continuu, cu un timp minim de nefuncționare.

- Securitatea fizică: Automatul trebuie să fie proiectat și construit astfel încât să reziste la potențiale tentative de vandalism sau acces neautorizat la componentele interne. Acesta poate include măsuri precum încuietori securizate, camere de supraveghere sau senzori de mișcare. va fi responsabilă pentru întreținerea software-ului livrat).

Constrângeri software:

- Integrarea cu sistemele existente: Aplicația software a sistemului automat trebuie să se integreze fără probleme cu sistemele existente ale hotelului, cum ar fi software-ul de gestiune a rezervărilor și baza de date MySQL. Aceasta poate necesita dezvoltarea unor API-uri sau adaptoare specifice pentru a asigura o comunicare corectă și eficientă între sisteme.

- Securitatea datelor: Sistemul trebuie să fie proiectat cu măsuri de securitate adecvate pentru a proteja datele personale ale oaspeților și informațiile despre rezervări. Aceasta include criptarea datelor sensibile, implementarea de politici de autentificare și autorizare, precum și conformitatea cu reglementările privind protecția datelor, cum ar fi GDPR.

- Scalabilitatea și performanța: Aplicația software trebuie să fie scalabilă pentru a face față unui număr mare de utilizatori simultani și a unui volum crescut de date, fără a compromite performanța sau timpul de răspuns. Aceasta poate necesita optimizarea codului, utilizarea tehnicilor de caching sau implementarea unei arhitecturi distribuite.

- Ușurința în utilizare: Interfața utilizator a aplicației trebuie să fie intuitivă, ușor de navigat și accesibilă pentru o gamă largă de oaspeți, indiferent de nivelul lor de competență tehnică. Aceasta poate implica proiectarea unei interfețe grafice atractive, furnizarea de instrucțiuni clare și concise, precum și includerea de opțiuni multilingve.

## Considerațiii de proiectare

### Obiective și linii directoare (ghiduri)

Principalele obiective și linii directoare care guvernează designul sistemului și software-ului pentru proiectul de automat de gestionare a rezervărilor și check-in-ului în regim hotelier sunt următoarele:

* Eficiență și performanță:
  + Sistemul trebuie să fie capabil să gestioneze un volum mare de tranzacții (check-in-uri, rezervări) într-un timp scurt, fără a compromite calitatea serviciilor.
  + Timpul de răspuns al sistemului trebuie să fie minim, pentru a asigura o experiență fluidă și satisfăcătoare pentru oaspeți.
* Ușurință în utilizare:
  + Interfața cu utilizatorul trebuie să fie intuitivă, ușor de navigat și accesibilă pentru o gamă largă de oaspeți, indiferent de nivelul lor de competență tehnică.
  + Procesele de check-in și de creare a unei noi rezervări trebuie să fie simplificate și optimizate, cu un număr minim de pași necesari.
* Fiabilitate și disponibilitate:
  + Sistemul trebuie să funcționeze în mod fiabil, cu un timp de nefuncționare minim, pentru a asigura disponibilitatea serviciilor de check-in 24/7.
  + În cazul unor defecțiuni hardware sau software, sistemul trebuie să poată recupera rapid și să reia funcționarea normală, fără a afecta experiența oaspeților.
* Securitate și confidențialitate:
  + Sistemul trebuie să fie proiectat și implementat cu măsuri de securitate robuste, pentru a proteja datele personale ale oaspeților și informațiile de plată.
  + Accesul la funcționalitățile de administrare și la datele sensibile trebuie să fie strict controlat și limitat doar la personalul autorizat.
* Integrare și interoperabilitate:
  + Sistemul trebuie să se integreze fără probleme cu sistemele existente ale hotelului, cum ar fi software-ul de gestionare a rezervărilor și sistemele de control al accesului.
  + Interfețele și protocoalele de comunicare trebuie să fie standardizate și bine documentate, pentru a facilita integrarea cu alte sisteme sau servicii externe.

### Metode de dezvoltare

Pentru dezvoltarea software-ului sistemului automat de gestionare a rezervărilor și check-in-ului, se vor utiliza următoarele tehnologii și metodologii:

* Limbajul de programare Python:
  + Aplicația principală care va rula pe Raspberry Pi va fi dezvoltată în limbajul Python, datorită simplității, flexibilității și compatibilității sale cu ecosistemul Raspberry Pi.
  + Se vor utiliza framework-uri și biblioteci Python precum Tkinter pentru interfața grafică și MySQL Connector pentru interacțiunea cu baza de date.
* Baza de date MySQL:
  + Pentru stocarea și gestionarea datelor despre rezervări, camere și oaspeți, se va utiliza o bază de date MySQL.
  + Interacțiunea cu baza de date se va realiza prin intermediul MySQL Workbench, un mediu de dezvoltare vizual pentru baze de date MySQL.
* Limbajul de programare C++ pentru Arduino:
  + Programul care va rula pe placa Arduino, responsabil pentru controlul motorului pas cu pas și simularea deblocării ușilor, va fi scris în limbajul C++.
  + Se va utiliza mediul de dezvoltare Arduino IDE pentru scrierea, compilarea și încărcarea codului pe placa Arduino.

### Strategii de arhitectură

Arhitectura sistemului automat de gestionare a rezervărilor și check-in-ului se bazează pe o abordare client-server, cu Raspberry Pi acționând ca server central și interfață principală cu utilizatorul, iar Arduino ca dispozitiv periferic pentru controlul accesului în camere.

* Arhitectura client-server:
  + Raspberry Pi va găzdui aplicația principală și va servi ca server, gestionând logica de afaceri, interacțiunea cu baza de date și comunicarea cu alte componente ale sistemului.
  + Ecranul tactil conectat la Raspberry Pi va acționa ca interfață client, permițând oaspeților să interacționeze cu sistemul și să efectueze operațiuni de check-in și rezervare.
* Stocarea datelor într-o bază de date MySQL:
  + Pentru a gestiona eficient datele despre rezervări, camere și oaspeți, se va utiliza o bază de date MySQL găzduită pe un server separat.
  + Raspberry Pi se va conecta la baza de date prin intermediul unei conexiuni de rețea securizate, utilizând protocolul TCP/IP și biblioteci Python specifice (precum MySQL Connector).
* Comunicarea între Raspberry Pi și baza de date:
  + Arduino va fi responsabil pentru controlul motorului pas cu pas și simularea deblocării ușilor camerelor.
  + Arduino va fi echipat cu un modul Wi-Fi (precum ESP8266) pentru a se conecta la rețeaua locală și a comunica direct cu baza de date MySQL.
  + Interogarea va implica o comparație între data curentă și data de check-out a fiecărei rezervări, pentru a determina dacă o rezervare este încă valabilă.

## Arhitectura Sistemului și Proiectarea Arhitecturii

Arhitectura sistemului automat de gestionare a rezervărilor și check-in-ului este proiectată pentru a asigura o interacțiune eficientă între componentele hardware și software, precum și pentru a facilita comunicarea cu baza de date și integrarea cu alte sisteme ale hotelului. Sistemul este structurat în patru componente principale: Raspberry Pi, Arduino, baza de date MySQL și aplicația Python.

Raspberry Pi servește ca unitate centrală de procesare și interfață cu utilizatorul, găzduind aplicația Python și interacționând cu ecranul tactil, cititorul RFID și alte periferice. Acesta se ocupă de logica de afaceri, validarea datelor și comunicarea cu baza de date MySQL pentru a gestiona rezervările și a actualiza disponibilitatea camerelor.

Arduino este responsabil pentru controlul accesului în camere, acționând motorul pas cu pas pentru a simula deblocarea ușilor. Acesta comunică direct cu baza de date MySQL prin intermediul unui modul Wi-Fi, interogând periodic tabelul de rezervări pentru a verifica valabilitatea acestora și a actualiza starea încuietorilor în consecință.

Aplicația Python este structurată în mai multe module, fiecare având responsabilități specifice:

* Interfața cu utilizatorul: Acest modul gestionează interacțiunea cu oaspeții prin intermediul ecranului tactil, afișând informații, primind input-uri și oferind feedback vizual.
* Gestionarea rezervărilor: Acest modul se ocupă de procesarea rezervărilor, atât a celor existente, cât și a celor noi. Acesta validează datele introduse, interogează baza de date și actualizează informațiile despre disponibilitatea camerelor.
* Controlul cititorului RFID: Acest modul interacționează cu cititorul RFID pentru a inscripționa cardurile de acces cu datele corespunzătoare rezervărilor confirmate.
* Comunicarea cu baza de date: Acest modul facilitează interacțiunea cu baza de date MySQL, executând interogări și actualizări pentru a menține consistența datelor între sistem și baza de date centrală.
* Integrarea cu alte sisteme: Acest modul gestionează comunicarea cu alte sisteme ale hotelului, cum ar fi software-ul de management al proprietății (PMS), prin intermediul API-urilor și protocoalelor de integrare.

Baza de date MySQL stochează toate informațiile relevante despre rezervări, camere și oaspeți, servind ca depozit central de date pentru întregul sistem. Aceasta este accesată atât de Raspberry Pi, cât și de Arduino, precum și de alte sisteme ale hotelului, asigurând o sursă unică și consistentă de informații.

Decizia de a adopta această arhitectură a fost motivată de necesitatea de a avea o separare clară a responsabilităților între componentele hardware și software, precum și de a asigura o comunicare eficientă și securizată cu baza de date. De asemenea, structurarea aplicației Python în module distincte favorizează modularitatea, reutilizarea codului și ușurința de întreținere pe termen lung.

Această arhitectură permite o scalabilitate orizontală, putând fi extinse cu ușurință prin adăugarea de noi dispozitive Raspberry Pi și Arduino pentru a gestiona mai multe puncte de check-in sau camere. De asemenea, integrarea cu alte sisteme ale hotelului este facilitată prin intermediul unei baze de date centralizate și al unor interfețe de comunicare standardizate.

### Vedere logică

A diagram of a flowchart

Description automatically generated

### Arhitectură hardware

Sistemul automat de gestionare a rezervărilor și check-in-ului folosește o arhitectură centralizată, bazată pe următoarele componente hardware principale:

* Raspberry Pi 4 Model B (procesor quad-core, 4GB RAM) - unitatea centrală de procesare
* Ecran tactil de 7 inch - interfața cu utilizatorul
* Cititor RFID RC522 - inscripționarea cardurilor de acces
* Arduino UNO cu modul Wi-Fi ESP8266 - controlul accesului în camere
* Motor pas cu pas - simularea încuietorilor ușilor

Raspberry Pi găzduiește aplicația Python și comunică cu baza de date MySQL. Ecranul tactil și cititorul RFID sunt conectate direct la Raspberry Pi. Arduino comunică cu baza de date prin Wi-Fi pentru a verifica starea rezervărilor și controlează motorul pas cu pas.

### Arhitectură software

Componentele software principale:

* Aplicația Python pe Raspberry Pi
  + Framework-uri: Flask, Tkinter, MySQL Connector
  + Module: main, interfață utilizator, gestionare rezervări, control RFID, comunicare bază de date
  + Interacțiuni: comunică cu baza de date MySQL prin TCP/IP
* Baza de date MySQL
  + Stochează informații despre rezervări, camere, oaspeți
  + Interacționează cu aplicația Python și programul Arduino
* Programul Arduino (C++)
  + Biblioteci: WiFi, MySQL, Stepper
  + Funcții: configurare, verificare periodică a stării rezervărilor
  + Interacțiuni: comunică cu baza de date MySQL prin Wi-Fi

Arhitectura se bazează pe separarea responsabilităților, o bază de date centralizată și comunicarea prin Wi-Fi pentru flexibilitate și scalabilitate.

### Arhitectura informațiilor

Sistemul automat de gestionare a rezervărilor și check-in-ului va stoca și gestiona informații despre rezervări, camere și cardurile de acces. Datele despre rezervări includ ID-ul rezervării, codul rezervării, datele de check-in și check-out, ID-ul camerei, date relevante despre oaspete, precum și suma totală. Datele despre camere cuprind ID-ul camerei, numărul camerei, tipul camerei, disponibilitatea. Informațiile despre cardurile de acces includ ID-ul cardului, codul cardului, ID-ul rezervării asociate și data expirării.

Aceste date vor fi stocate într-o bază de date MySQL, cu tabele corespunzătoare și relațiile necesare între ele. Datele vor fi introduse în sistem prin intermediul interfeței cu utilizatorul și vor fi actualizate de către sistem în urma procesării rezervărilor și a inscripționării cardurilor de acces. Accesul la aceste date va fi controlat și limitat în funcție de rolurile utilizatorilor pentru a asigura confidențialitatea și securitatea informațiilor.

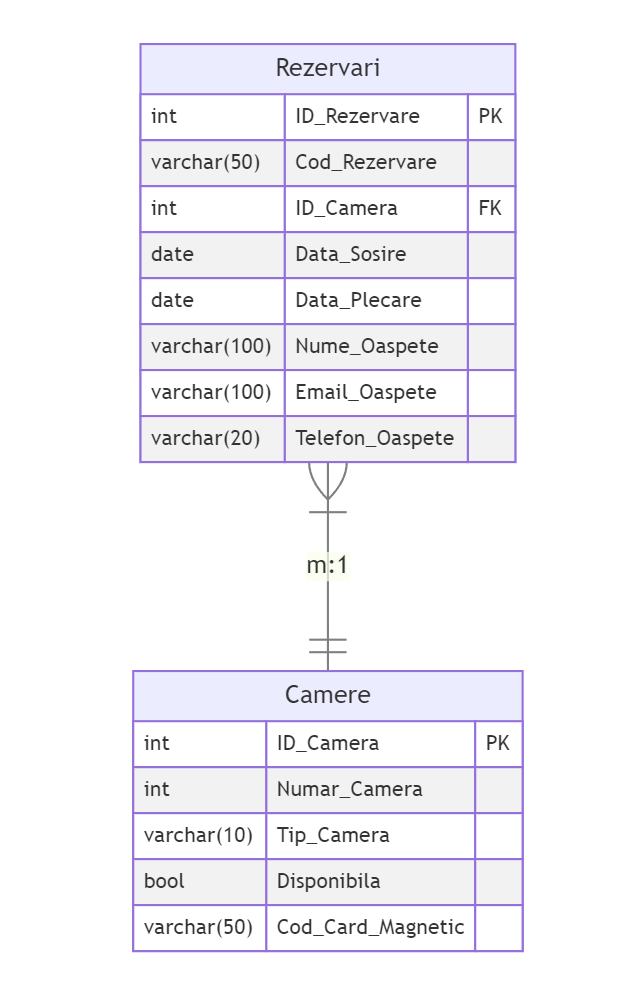
### Diagrama de arhitectură a sistemului

A diagram of a system

Description automatically generated

## Proiectarea sistemului

### Proiectarea bazei de date



##### Fișiere non-DBMS

Exista mai multe fișiere non-DBMS utilizate pentru diferite scopuri. Acestea includ fișiere de configurare, fișiere de cod sursă, fișiere de imagini. Fișierele de configurare sunt utilizate pentru a stoca setările și parametrii specifici proiectului, cum ar fi detaliile de conectare la baza de date MySQL, setările pentru ecranul tactil și alte configurații relevante. Aceste fișiere sunt citite de aplicația Python la pornire pentru a configura comportamentul sistemului.

Fișierele de cod sursă Python conțin logica principală a aplicației de automatizare a check-in-ului. Acestea includ module pentru interfața grafică dezvoltată cu tkinter, module pentru comunicarea cu baza de date MySQL, module pentru gestionarea procesului de check-in și alte funcționalități specifice proiectului.

Fișierele de imagini sunt utilizate în interfața grafică a aplicației dezvoltate cu tkinter. Acestea includ logo-ul hotelului, imagini de fundal, iconițe și alte elemente vizuale necesare pentru a crea o interfață atractivă și ușor de utilizat. Fișierele de imagini sunt încărcate și afișate în aplicație folosind funcționalitățile tkinter.

### Interfețe utilizator

Oaspeții interacționează cu sistemul prin intermediul ecranului tactil al aplicației Python dezvoltate cu tkinter. Ei pot introduce codul de rezervare existent sau pot crea o nouă rezervare dacă sunt camere disponibile. După finalizarea procesului, oaspeții primesc un card magnetic inscripționat cu detaliile camerei rezervate.

Numărul total de utilizatori anticipați depinde de capacitatea și gradul de ocupare al hotelului. Numărul maxim de utilizatori simultani este limitat la o singură persoană la un moment dat, deoarece sistemul este proiectat pentru a fi utilizat individual de către fiecare oaspete în timpul procesului de check-in. Nu există utilizatori externi ai sistemului, acesta fiind destinat exclusiv utilizării interne de către oaspeții hotelului.

#### Intrări

Ecranul tactil afișează câmpuri de introducere a datelor, cum ar fi câmpul pentru codul de rezervare sau câmpurile pentru crearea unei noi rezervări (nume, date de sosire/plecare etc.). Aceste câmpuri sunt proiectate pentru a fi intuitive și ușor de completat, cu validări adecvate pentru a asigura corectitudinea datelor introduse.

Interfața grafică (GUI) a aplicației este proiectată pentru a fi atractivă și ușor de navigat, cu butoane, meniuri și alte elemente de control care ghidează oaspeții prin procesul de check-in.

Criteriile de editare sunt definite pentru fiecare câmp de introducere a datelor pentru a asigura integritatea și validitatea informațiilor furnizate de oaspeți. Aceste criterii pot include verificări precum câmpuri obligatorii, limite de lungime, formate specifice (de exemplu, pentru datele de sosire/plecare) și alte reguli de validare.

#### Ieșiri

Principala ieșire a sistemului este cardul magnetic inscripționat cu detaliile camerei rezervate. După ce oaspetele introduce un cod de rezervare valid sau creează o nouă rezervare, sistemul preia un card magnetic din interior și îl inscripționează folosind RFID. Acest card magnetic permite oaspetelui accesul în camera rezervată.

O altă formă de ieșire a sistemului sunt mesajele de succes sau de eroare afișate pe ecranul tactil al aplicației. După finalizarea cu succes a procesului de check-in, se afișează un mesaj de confirmare, informând oaspetele că rezervarea a fost validată și că poate ridica cardul magnetic inscripționat. În cazul în care apar erori, cum ar fi introducerea unui cod de rezervare invalid sau probleme tehnice, se afișează mesaje de eroare corespunzătoare pentru a informa oaspetele despre problemă și a oferi instrucțiuni suplimentare.

### Proiectarea interfețelor cu utilizatorul

Interfața aplicației de automatizare a procesului de check-in în regim hotelier este proiectată pentru a oferi oaspeților o experiență intuitivă și fără probleme. Când oaspetele se apropie de sistemul de check-in, este întâmpinat de un ecran de pornire care afișează un mesaj de bun venit și instrucțiuni clare pentru a începe procesul.

Odată ce oaspetele interacționează cu sistemul, este ghidat către ecranul de introducere a codului de rezervare. Aici, oaspetele are posibilitatea de a introduce codul de rezervare existent, dacă acesta a fost obținut în prealabil. După introducerea codului, sistemul validează informațiile și, în cazul unui cod valid, oaspetele este direcționat către ecranul de confirmare a check-in-ului.

În cazul în care oaspetele nu are un cod de rezervare prealabil, interfața oferă opțiunea de a crea o nouă rezervare direct de la sistemul de check-in. Pe ecranul de creare a unei noi rezervări, oaspetele poate introduce detaliile necesare, cum ar fi numele, datele de sosire și plecare, și alte informații relevante. După completarea tuturor câmpurilor necesare, oaspetele poate finaliza procesul de creare a rezervării, iar sistemul îl va dirija către ecranul de confirmare a check-in-ului.

Pe ecranul de confirmare a check-in-ului, oaspetele primește o confirmare vizuală a rezervării sale, împreună cu detaliile camerei alocate. Aici, oaspetele este instruit să ridice cardul magnetic inscripționat, care îi va permite accesul în camera rezervată. După o scurtă perioadă de timp sau la confirmarea oaspetelui, sistemul revine la ecranul de pornire, gata pentru următorul oaspete.

În cazul apariției unor erori sau probleme tehnice pe parcursul procesului de check-in, interfața afișează un ecran de eroare. Acest ecran prezintă un mesaj clar care explică natura problemei și oferă instrucțiuni oaspetelui pentru a contacta personalul hotelului pentru asistență suplimentară.

## Scenarii de utilizare

Sistemul de automatizare a procesului de check-in în regim hotelier este conceput pentru a oferi oaspeților o modalitate eficientă și convenabilă de a realiza check-in-ul și de a obține acces la camerele lor fără a fi nevoie de interacțiunea directă cu personalul hotelului. Următoarele scenarii de utilizare descriu funcționalitatea și fluxul operațional al sistemului:

Scenariul 1: Check-in cu cod de rezervare prealabil

* Oaspetele sosește la hotel și se apropie de sistemul de check-in automatizat.
* Pe ecranul de pornire, oaspetele selectează opțiunea de a introduce un cod de rezervare.
* Oaspetele introduce codul său de rezervare utilizând interfața tactilă.
* Sistemul validează codul de rezervare și preia informațiile asociate din baza de date.
* Dacă codul de rezervare este valid, sistemul afișează un mesaj de confirmare și detaliile camerei rezervate.
* Sistemul preia un card magnetic gol din interiorul dispozitivului și îl inscripționează cu detaliile camerei folosind tehnologia RFID.
* Cardul magnetic inscripționat este eliberat către oaspete, împreună cu instrucțiunile de utilizare.
* Oaspetele își ridică cardul magnetic și se îndreaptă către camera sa, utilizând cardul pentru a accesa ușa camerei.

Scenariul 2: Crearea unei noi rezervări la fața locului

* Oaspetele sosește la hotel fără o rezervare prealabilă și se apropie de sistemul de check-in automatizat.
* Pe ecranul de pornire, oaspetele selectează opțiunea de a crea o nouă rezervare.
* Sistemul afișează un formular de rezervare pe care oaspetele îl completează cu detaliile necesare, cum ar fi numele, datele de sosire și plecare, și preferințele camerei.
* Oaspetele trimite formularul de rezervare.
* Sistemul verifică disponibilitatea camerelor pe baza datelor introduse și afișează opțiunile disponibile.
* Oaspetele confirmă rezervarea.
* Sistemul procesează rezervarea, actualizează baza de date și generează un cod de rezervare unic.
* Un card magnetic gol este preluat din interiorul dispozitivului și inscripționat cu detaliile camerei folosind tehnologia RFID.
* Cardul magnetic inscripționat este eliberat către oaspete, împreună cu instrucțiunile de utilizare și detaliile rezervării.
* Oaspetele își ridică cardul magnetic și se îndreaptă către camera sa, utilizând cardul pentru a accesa ușa camerei.

## Proiectarea detaliată a comunicațiilor interne

A diagram of a network

Description automatically generated

Sistemul este compus din mai multe componente interconectate printr-o rețea dedicată. Comunicarea internă între aceste componente este esențială pentru funcționarea corectă și eficientă a sistemului.

Componente principale:

* Automatul de check-in (Raspberry Pi cu ecran tactil și aplicația Python)
* Sistemele de blocare ale camerelor (Arduino UNO pentru demonstratie)
* Serverul central (găzduiește baza de date MySQL)

Toate componentele sunt conectate la o rețea locală dedicată (LAN) prin intermediul unei conexiuni Wi-Fi securizate. Automatul de check-in și sistemele de blocare ale camerelor acționează ca clienți în rețea, comunicând cu serverul central. Serverul central gestioneaza stocarea datelor în baza de date MySQL.

Fluxul de date:

* Automatul de check-in trimite cereri către serverul central pentru a valida codurile de rezervare, a crea noi rezervări și a obține informații despre camere.
* Serverul central procesează cererile, interacționează cu baza de date MySQL și trimite răspunsuri înapoi către automatul de check-in.
* Sistemele de blocare ale camerelor interoghează baza de date la intervale de timp prestabilite pentru a-și updata codurile inscripționate pe cardurile magnetice la apropierea cărora vor debloca ușile