

Universitatea Tehnică “Gheorghe Asachi”, Iași  
Facultatea de Automatică și Calculatoare  
Specializarea Calculatoare și Tehnologia Informației

Nume și prenume:

- ★ Maieczki Petronela-Sînziana
- ★ Butnaru Raimond Eduard
- ★ Guriuc Vlad Ionuț
- ★ Lupu Andreea-Sabina

Grupa: 1306B

Disciplina: Ingineria Programării

Profesor coordonator: Corina Cîmpanu

## **Cerințe software și specificații pentru**

## **RoomWizard**

# CUPRINS

<b>Istoricul revizuirilor</b>	<b>3</b>
<b>1. Introducere</b>	<b>5</b>
1.1 Scop	5
1.2 Convențiile documentelor	5
1.3 Publicul vizat și sugestii de lectură	6
1.4 Domeniul de aplicare al produsului	6
1.5 Referințe	7
<b>2. Descriere generală</b>	<b>7</b>
2.1 Perspectiva produsului	7
2.2 Funcțiile produsului	7
2.3 Clasele și caracteristicile utilizatorilor	8
2.4 Mediul de operare	9
2.5 Constrângeri de proiectare și implementare	9
2.6 Documentația utilizatorului	10
2.7 Ipoteze și dependențe	10
<b>3. Cerințe de interfață externă</b>	<b>11</b>
3.1 Interfețe cu utilizatorul	11
3.2 Interfețe hardware	18
3.3 Interfețe software	18
<b>4. Caracteristici ale sistemului</b>	<b>19</b>
4.1 Vizualizarea rezervărilor clientului	19
4.2 Adăugarea unei noi rezervări	19
<b>4.3 Adăugarea unui manager</b>	<b>20</b>
<b>4.4 Ștergerea unui manager</b>	<b>21</b>
<b>4.5 Ștergerea unei camere</b>	<b>22</b>
<b>4.6 Fereastra ajutor</b>	<b>22</b>
<b>5. Cerințe non-funcționale</b>	<b>23</b>
5.1 Cerințe de performanță	23
5.2 Cerințe de siguranță	23
5.3 Cerințe de securitate	24
5.4 Atribute de calitate	24
5.5 Reguli de afaceri	24
<b>6. Alte cerințe</b>	<b>24</b>
6.1 Baza de date	24
6.2 Modelul datelor	25
6.3 Diagrame UML	25
6.4 Modul de utilizare al programului (Help)	33
6.5 Execuție program (capturi de ecran din diverse situații)	34
6.6 Test case	37

## Istoricul revizuirilor

Nume coechipieri	Data	Descriere modificare	Versiune
Butnaru Raimond	26-04-2024	Inițializare repository github + împărțire task-uri	0.1
Maieczki Sînziana	29-04-2024	Creare interfețe (GUI)	0.1
Lupu Sabina	17-05-2024	Creare documentație	0.1
Guriuc Vlad	17-05-2024	Creare Help	0.1
Maieczki Sînziana	17-05-2024	Actualizare interfețe	0.2
Butnaru Raimond	17-05-2024	Creare bază de date SQLite	0.3
Butnaru Raimond Guriuc Vlad	18-05-2024	Inițializare tabele + constrângeri + populare bază de date	0.3
Butnaru Raimond	19-05-2024	Adăugarea logicii de state pattern + logare și înregistrare utilizatori	0.4
Lupu Sabina	20-05-2024	Crearea unei „sesiuni” de logare pentru utilizatori	0.5
Maieczki Sînziana	21-05-2024	Aplicarea modelului strategie (strategy pattern) pentru diferite verificări	0.6
Guriuc Vlad	21-05-2024	Inițializarea de unit testing	0.7
Butnaru Raimond	21-05-2024	Rezolvare erori pe parte de client	0.8
Guriuc Vlad	22-05-2024	Extindere bază de date	0.9
Butnaru Raimond	22-05-2024	Rezolvare logică de creare rezervare	0.10
Butnaru Raimond	23-05-2024	Update creare rezervare	0.11
Guriuc Vlad	24-05-2024	Generare antet	0.11
Butnaru Raimond	25-05-2024	Atingere stadiu final al aplicației	1.0

Maieczki Sînziana Lupu Sabina	26-05-2024	Reutilizare cod folosind DLL	1.1
Maieczki Sînziana	27-05-2024	Comentarii duble și triple + Generare doxygen	1.1
Lupu Sabina Guriuc Vlad	27-05-2024	Finalizare modul de unit testing	1.2
Lupu Sabina	28-05-2024	Completare documentație	1.2
Guriuc Vlad Butnaru Raimond	28-05-2024	Rezolvare bug-uri și generare de diagrame	1.3

# 1. Introducere

## 1.1 Scop

Scopul acestei aplicații este de a furniza un sistem complet și integrat de gestionare a hotelului, care să optimizeze și să automatizeze procesele operaționale și administrative ale hotelului. Aplicația vizează îmbunătățirea eficienței și calității serviciilor oferite clienților, printr-o interfață prietenoasă și diverse funcționalități.. Prin implementarea acestei aplicații, se urmărește atingerea următoarelor obiective:

❖ ***Automatizarea procesului de rezervare:***

- Permite clienților să efectueze rezervări online printr-un portal web accesibil și intuitiv.
- Asigură actualizarea în timp real a disponibilității camerelor și a prețurilor.

❖ ***Gestionarea eficientă a datelor clienților:***

- Stocarea și administrarea datelor personale ale clienților într-un mod securizat.

Prin aceste funcționalități, aplicația urmărește să crească satisfacția clienților, să reducă erorile operaționale și să optimizeze resursele hotelului, contribuind astfel la succesul și rentabilitatea afacerii.

## 1.2 Convențiile documentelor

Acest document urmează formatarea standard IEEE pentru dezvoltarea de software. Standardul definește o formatare obișnuită pe care o urmează acest document, inclusiv scrierea care trebuie făcută la persoana a treia, voce pasivă, precum și text lizibil și corect din punct de vedere gramatical.

## 1.3 Publicul vizat și sugestii de lectură

Acest document nu este destinat utilizatorului final, deoarece oferă o specificație detaliată a modului în care urmează să fie implementată aplicația hotelului. Un utilizator are nevoie de informații despre cum să utilizeze aplicația, în loc de cum să o creeze, astfel, acest document este mai orientat către testerii și dezvoltatorii aplicației. Documentul începe cu o prezentare generală a funcțiilor și specificațiilor pentru aplicație în secțiunea 2, apoi trece la descrierea cerințelor pentru interfața cu hardware și software extern în secțiunea 3. Secțiunea 4 descrie funcțiile aplicației în detaliu, iar secțiunea 5 listează diverse cerințe pe care aplicația trebuie să le respecte după finalizare. Se sugerează ca toate categoriile de public ale acestui document să înceapă mai întâi cu secțiunea 2 pentru a avea o idee generală despre cerințele aplicației. Testerii ar trebui să citească în continuare secțiunile 5.1 până la 5.4 (performanță, siguranță, cerințe de securitate și atribute de calitate software). Acest lucru este pentru a vă face o idee despre modul în care aplicația îi va afecta și despre sistemul pe care îl rulează, precum și despre aspirațiile pentru calitate. În continuare, un tester ar trebui să citească secțiunea 3.1 (interfețe cu utilizatorul) urmată de toată secțiunea 4 (funcțiile sistemului). Citirea documentului în această ordine îi va oferi tester-ului o idee despre ce să se aștepte în interfață la prima vedere, iar apoi poate testa toate funcțiile individual pentru a se asigura că respectă specificațiile.

După ce au citit secțiunea 2, dezvoltatorii aplicației ar trebui să citească secțiunile rămase în ordine, deoarece acest document a fost conceput special în scopul dezvoltării aplicației. Dezvoltatorul trebuie să-și facă o idee generală în secțiunea 2 a aplicației, apoi, cum trebuie să interacționeze cu orice altceva din secțiunea 3 (astfel încât să aibă o idee despre ce instrumente să folosească și, eventual, cum ar trebui să le folosească). Secțiunea 4 este cea mai importantă pentru un dezvoltator, deoarece descrie toate funcțiile aplicației în detaliu și va ajuta la luarea deciziilor în scrierea codului real pentru aplicație. Secțiunea 5 este considerată cea mai puțin importantă, dar dezvoltatorul ar trebui să o citească pentru a se asigura că aplicația lor a aderat la idealurile date.

## 1.4 Domeniul de aplicare al produsului

RoomWizard este o aplicație de gestionare a rezervărilor, clienților și camerelor dezvoltată pe platforma .NET și utilizând limbajul de programare C#. Această aplicație oferă un mediu centralizat și intuitiv pentru personalul hotelier, facilitând procesele de rezervare și gestionare a informațiilor despre clienți și camere. Scopul principal al aplicației este de a îmbunătăți eficiența și experiența atât a personalului hotelier, cât și a clienților, prin automatizarea și optimizarea proceselor operaționale și oferirea diferitor servicii..

Această aplicație este destinată să ruleze pe sistemele de operare Windows și poate fi utilizată cu ușurință pe orice computer echipat cu un mediu de dezvoltare integrat, cum ar fi Visual Studio. Documentația și codul aplicației vor urma standardele și convențiile recomandate pentru dezvoltarea în limbajul C# și utilizarea framework-ului .NET, asigurând astfel un cod coerent, ușor de înțeles și întreținut. De asemenea, aplicația va beneficia de caracteristicile și avantajele oferite de .NET, inclusiv securitatea îmbunătățită, gestionarea memoriei și interoperabilitatea cu alte tehnologii Microsoft.

Prin implementarea acestei aplicații, se urmărește îmbunătățirea eficienței operaționale a hotelului, optimizarea proceselor de gestionare a rezervărilor și clienților și creșterea satisfacției clienților prin oferirea unor servicii de calitate.

## 1.5 Referințe

*IEEE. IEEE Std 830-1998 IEEE Recommended Practice for Software Requirements*

# 2. Descriere generală

## 2.1 Perspectiva produsului

RoomWizard reprezintă o aplicație de rezervări pentru hoteluri, dezvoltată în cadrul IDE-ului Visual Studio folosind .NET. Obiectivul principal al aplicației este de a facilita procesul de rezervare și de a îmbunătăți experiența atât pentru clienți, cât și pentru personalul hotelier. RoomWizard este o soluție software completă, care integrează funcționalități esențiale pentru gestionarea eficientă a rezervărilor și a resurselor hoteliere.

Aplicația este dezvoltată în C# și rulează pe sistemul de operare Windows, fiind implementată ca o aplicație de tip Windows Forms.

## 2.2 Funcțiile produsului

RoomWizard este o aplicație de rezervări pentru hoteluri, concepută pentru a îmbunătăți eficiența operațională și experiența clienților. Mai jos este un rezumat al principalelor funcții pe care aplicația trebuie să le îndeplinească sau să le permită utilizatorilor să le îndeplinească. Detalii suplimentare vor fi furnizate în *Secțiunea 3*.

### Funcții principale

#### ❖ *Rezervări online:*

- Clienții pot căuta camere disponibile.
- Clienții pot verifica prețurile și tipurile de camere.
- Clienții pot efectua rezervări în timp real.

- ❖ **Gestionarea profilului clientului:**
  - Crearea și gestionarea profilurilor personale de către clienți.
  - Vizualizarea și actualizarea informațiilor de contact.
  - Istoricul propriilor rezervări.
- ❖ **Gestionarea inventarului:**
  - Actualizarea în timp real a disponibilității camerelor.
- ❖ **Administrarea utilizatorilor:**
  - Gestionarea conturilor și permisiunilor utilizatorilor interni.
  - Monitorizarea activităților și accesului.

### Beneficii ale funcționalităților

- ❖ **Eficiență crescută:**
  - Automatizarea și integrarea proceselor reduc timpul și efortul manual.
- ❖ **Experiență îmbunătățită pentru clienți:**
  - Aplicația oferă o platformă intuitivă și accesibilă pentru realizarea rezervărilor.

Aceste funcții sunt esențiale pentru a asigura o experiență pozitivă atât pentru utilizatori, cât și pentru personalul hotelier, contribuind astfel la succesul și eficiența operațională a hotelului.

## 2.3 Clasele și caracteristicile utilizatorilor

RoomWizard este o aplicație de rezervări pentru hoteluri care va fi utilizată de diferite clase de utilizatori. Utilizatorii sunt diferențiați pe baza frecvenței de utilizare, setului de funcții utilizate, expertizei tehnice, nivelurilor de securitate sau privilegii, nivelului educațional sau experienței. Mai jos sunt descrise caracteristicile relevante ale fiecărei clase de utilizatori.

### Clasele de utilizatori

- ❖ **Clienți**
  - **Descriere:** Persoane care doresc să facă rezervări de camere la hotel.
  - **Frecvență de utilizare:** Ocazional, în funcție de necesitățile de călătorie.
  - **Funcții utilizate:** Căutare camere, vizualizare detalii camere, rezervare camere.
  - **Expertiză tehnică:** Nivel scăzut spre mediu; utilizatori obișnuiți cu navigarea pe internet și efectuarea de rezervări online.
  - **Niveluri de securitate:** Acces limitat la propriile date personale și informațiile de rezervare.
- ❖ **Manager**
  - **Descriere:** Personalul de conducere al hotelului responsabil cu supervizarea operațiunilor și luarea deciziilor strategice.
  - **Frecvență de utilizare:** Zilnic sau periodic, în funcție de necesitățile operaționale.
  - **Funcții utilizate:** Vede toate rezervările, șterge rezervări, adaugă camere, șterge camere. Ultimul update, ștergerea unei camere rezulta automat ștergerea tuturor rezervărilor pe acea cameră.

- **Expertiză tehnică:** Nivel mediu spre avansat; familiarizați cu interpretarea datelor și utilizarea sistemelor de management.
- **Niveluri de securitate:** Acces ridicat.
- ❖ **Administrator**
  - **Descriere:** Personalul tehnic responsabil cu întreținerea și securitatea aplicației.
  - **Frecvență de utilizare:** Periodic, pentru întreținere și actualizări.
  - **Funcții utilizate:** Vede toate rezervările, șterge rezervări, adăuga manageri, șterge manageri și utilizatori. Orice ștergere a unui utilizator rezulta automat și ștergerea tuturor rezervărilor acelui utilizator.
  - **Expertiză tehnică:** Nivel avansat.
  - **Niveluri de securitate:** Acces complet la aproape toate funcțiile administrative și tehnice ale sistemului.

### Importanța claselor de utilizatori

- ❖ **Clienți:** Clasa cea mai importantă, deoarece reprezintă utilizatorii finali care generează veniturile prin rezervările efectuate.
- ❖ **Manager:** Critici pentru deciziile strategice și supervizarea generală a operațiunilor.
- ❖ **Administrator:** Importanți pentru menținerea funcționării și securității aplicației.

Aceste clase de utilizatori asigură funcționarea eficientă și securizată a RoomWizard, contribuind la satisfacția clienților și la succesul operațional al hotelului.

## 2.4 Mediul de operare

Din moment ce RoomWizard este implementat ca o aplicație de tip Windows Forms, scrisă în C# și folosind .NET, aceasta va putea fi compatibilă pe mai multe platforme. Utilizatorii de PC sau Mac nu ar trebui să întâmpine probleme în rularea programului.

RoomWizard este proiectată pentru a funcționa eficient într-un mediu de afaceri, oferind o compatibilitate largă cu platformele hardware și software utilizate în mod obișnuit în industria hotelieră.

## 2.5 Constrângeri de proiectare și implementare

RoomWizard este dezvoltat într-un context academic, ca proiect pentru o materie universitară, fără a implica costuri financiare. Cu toate acestea, există anumite constrângeri și factori care pot influența procesul de dezvoltare:

- ❖ **Timpul:** Proiectul trebuie să fie finalizat într-un anumit interval de timp, conform cerințelor academice și a termenelor limită stabilite de către profesorul coordonator al proiectului.
- ❖ **Platforma tehnologică:** RoomWizard va fi dezvoltat utilizând tehnologii specifice, precum .NET Framework și Visual Studio, ceea ce implică o familiarizare și expertiză adecvată cu aceste unelte și limbajul de programare C#.



- ❖ **Complexitatea aplicației:** Având în vedere natura proiectului și cerințele sale, este posibil să întâmpinăm provocări legate de gestionarea și implementarea funcționalităților complexe, cum ar fi sistemul de rezervări și interfața utilizatorului.
- ❖ **Experiența echipei:** Neexistând experiență anterioară în dezvoltarea unei aplicații hoteliere, ar putea apărea dificultăți în proiectarea și implementarea unor aspecte specifice, cum ar fi gestionarea bazelor de date, securitatea și aspectele legate de interfața utilizatorului.

## 2.6 Documentația utilizatorului

Pentru RoomWizard, va fi furnizată următoarea componentă de documentație pentru utilizator:

- ❖ **Ajutor online:** O secțiune de ajutor integrată în aplicație, disponibilă printr-un buton de ajutor ("Help"). Acest buton va fi plasat vizibil în interfața aplicației, oferind utilizatorilor acces rapid și facil la informații utile privind utilizarea aplicației.

Documentația pentru utilizator va fi livrată împreună cu software-ul RoomWizard în formate accesibile și ușor de înțeles pentru utilizatori. Documentația va fi disponibilă în limba română și va fi structurată în conformitate cu standardele de documentare a utilizatorului.

## 2.7 Ipoteze și dependențe

În dezvoltarea aplicației noastre de rezervări hoteliere, am făcut anumite presupuneri și identificat dependențe care ar putea afecta cerințele specificate în acest document SRS. Acestea includ:

- ❖ **Utilizarea componentelor terțe sau comerciale:** Aplicația se va baza pe anumite biblioteci și componente terțe pentru funcționalități precum autentificare, gestionarea bazelor de date, și interfața cu utilizatorul. Dacă aceste componente nu sunt disponibile sau suferă modificări semnificative, proiectul ar putea fi afectat.
- ❖ **Mediul de dezvoltare:** Aplicația este dezvoltată utilizând .NET și C# în Visual Studio. Presupunem că echipa de dezvoltare are acces continuu la aceste instrumente și că mediul de dezvoltare rămâne stabil pe durata proiectului.
- ❖ **Compatibilitatea cu platformele:** Aplicația este destinată să funcționeze pe sisteme de operare Windows, dar trebuie să fie compatibilă și cu alte platforme utilizând .NET Core. Eventuale probleme de compatibilitate ar putea afecta livrarea aplicației.
- ❖ **Experiența echipei:** Echipa de dezvoltare are experiență limitată în C# și proiectarea de software complex. Am presupus că echipa va putea învăța și adapta rapid noile tehnologii și metode necesare pentru acest proiect.

Dacă oricare dintre aceste ipoteze se dovedește a fi incorectă, nerealizabilă sau se schimbă pe parcursul dezvoltării, proiectul ar putea întâmpina întârzieri sau modificări ale cerințelor inițiale. Aceste aspecte trebuie monitorizate îndeaproape și ajustate conform noilor informații și situații apărute.

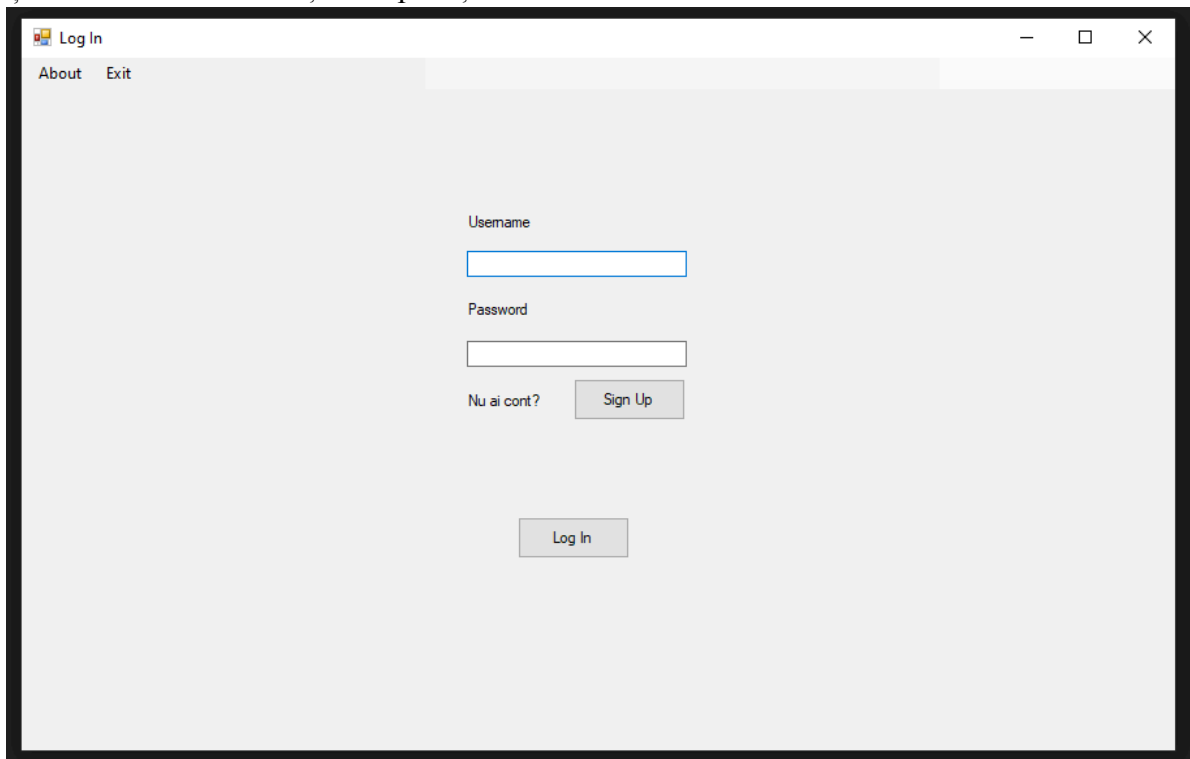
## 3. Cerințe de interfață externă

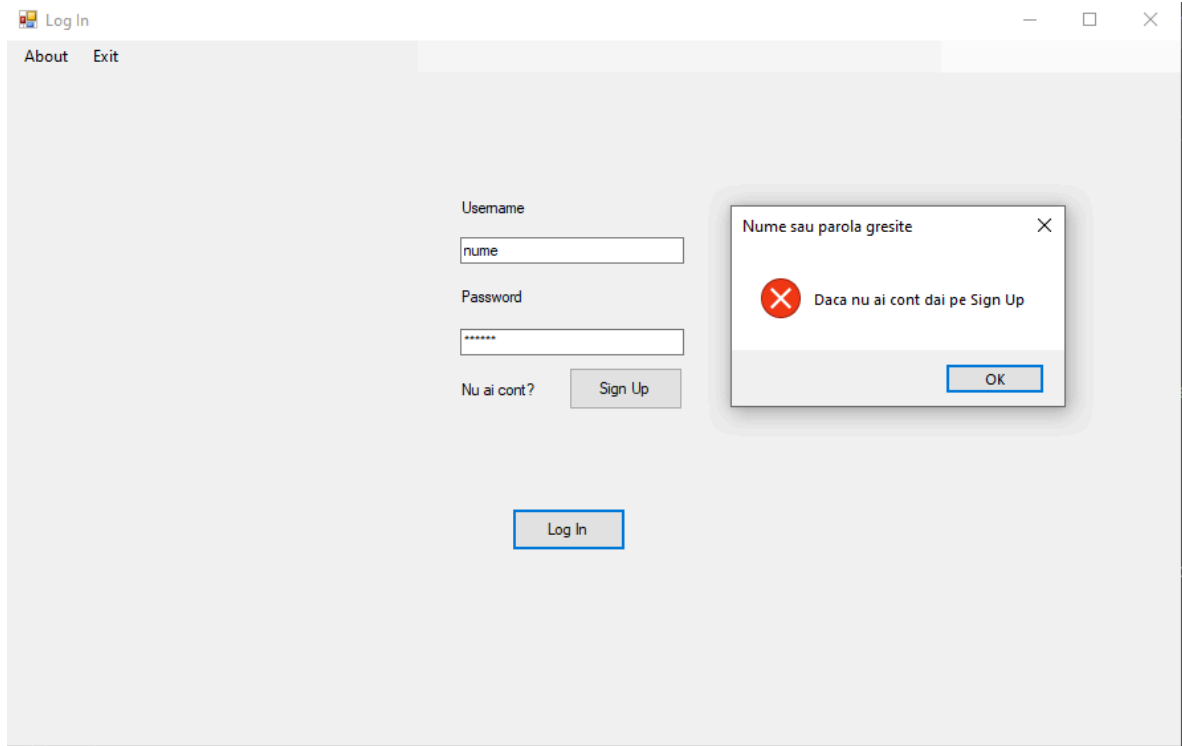
### 3.1 Interfețe cu utilizatorul

În această secțiune sunt descrise caracteristicile și constrângerile interfeței grafice (GUI) cu utilizatorul pentru aplicația de rezervări hoteliere.

**1. Fereastra principală (Log In):** Aceasta este formată din două câmpuri de completare: **Username** și **Password**, astfel dacă utilizatorul deja are un cont, va completa câmpurile cu informațiile necesare și va apăsa butonul **Log In** pentru a se conecta. În cazul în care utilizatorul a greșit datele, va fi întors la această pagină, primind un mesaj de atenționare: *"Nume sau parolă greșită!"*. De asemenea, la actionarea butonului de **Log In** se preiau datele din campul de username, se cauta existenta usernamului in baza de date, daca acesta exista, se face direct hash la parola (pentru a nu se stoca nicaieri nici macar temporar parola ne-hash-uita) dupa care se verifica daca hash-ul acelei parola corespunde cu ce se afla in baza de date. Toate cazurile alternative rezulta un pop-up care atentioneaza utilizatorul.

Dacă utilizatorul nu are un cont, acesta va fi dus prin apăsarea butonului **Sign Up** la o interfață pentru crearea contului. Fereastra principală conține și un buton de **About**, pentru informații suplimentare și un buton de **Exit** în cazul în care se dorește ieșirea din aplicație, precum și toate celelalte interfețe ale aplicației.





**2. Sign Up:** În momentul apăsării butonului **Sign Up** din pagina de **Log In**, utilizatorul va fi redirecționat către pagina de **Sign Up**, unde acesta va completa câmpurile **Username**, **Password**, **Nume**, **Prenume**, **Telefon**, **Email** și **Data Nașterii** cu informațiile personale pentru a își crea un cont. În cazul în care utilizatorul va avea un câmp completat greșit, de exemplu, câmpul **Email**, acesta va fi atenționat print-un **TextBox**. După completarea corectă a tuturor câmpurilor, utilizator poate să își creeze contul prin apăsarea butonului **Sign Up**, astfel acesta va primi un mesaj cum că va fi redirecționat către pagina principală. Pentru a se întoarce la pagina de autentificare în cazul în care nu mai dorește să își facă cont sau pentru că are deja unul, utilizator va apăsa butonul **Log In**. Un utilizator care nu are vârsta peste 18 ani, nu își va putea crea cont.

Sign Up

About Exit

Username

Password

Nume

Prenume

Telefon

Email

Data Nastere

[Ai deja cont?](#)

Sign Up

About Exit

Username

Password

Nume

Prenume


Telefon

Email

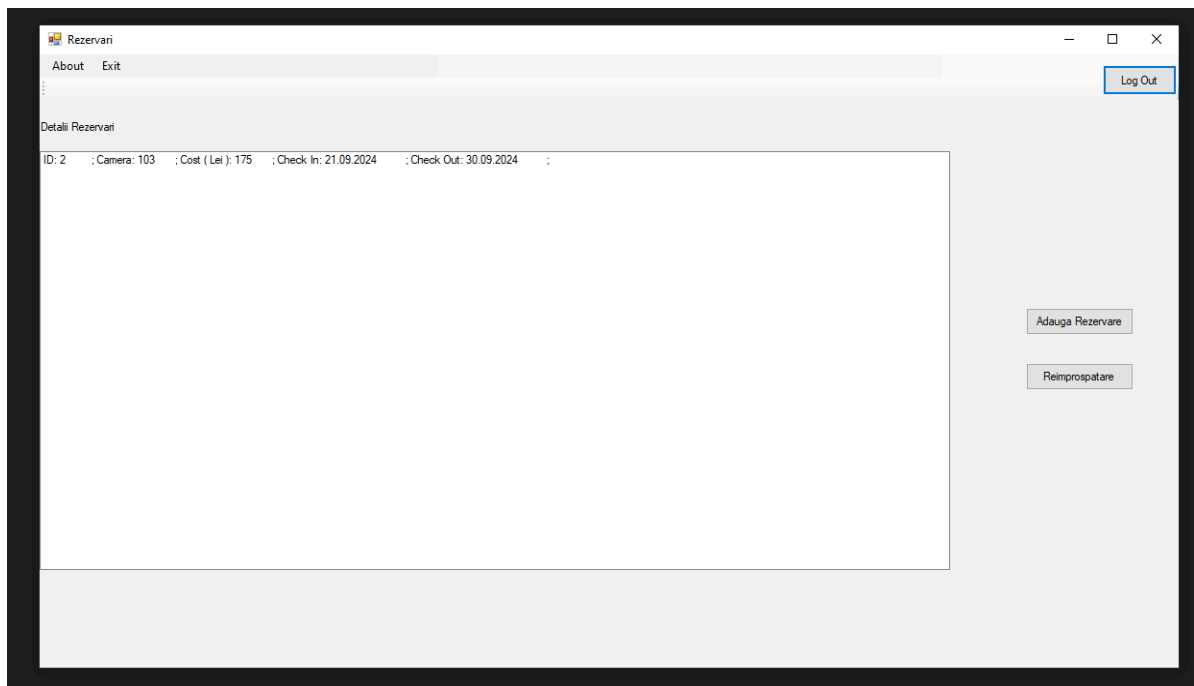
Data Nastere

[Ai deja cont?](#)

V-ati inregistrat cu succes, veti fi redirectionat la pagina de Log ...

 Succes

**3. Rezervări:** În cazul în care utilizatorul este un client, după autentificare, acesta va fi redirecționat către pagina *Rezervări*, unde are posibilitatea de a își vedea detaliile rezervărilor făcute de către acesta, prin intermediul ferestrei **Detalii Rezervări**, poate adăuga o rezervare prin apăsarea butonului **Adaugă Rezervare**. După adăugarea unei rezervări, pentru ca aceasta să apară în **Detalii Rezervări**, clientul trebuie să apese butonul **Reîmprospătare**. După efectuarea tuturor acțiunilor dorite, clientul se poate deloga prin apăsarea butonului **Log Out**, buton care îl va redirecționa spre pagina de autentificare.



**4. Adăugare Rezervare:** La acționarea imediată a butonului de **Adaugă Rezervare** (fără a completa câmpurile), platforma va atenționa clientul că nu a căutat camere, de aceea este necesar ca acesta să își exprime preferințele pentru camera dorită și să apese pe butonul **Caută** pentru a vedea ce camere sunt disponibile cu atributele selectate de acesta (**Tipul Camerei, Capacitate, Tip Balcon**). În cazul în care nu se găsește nici o cameră care să satisfacă dorințele clientului pentru perioada aleasă cu ajutorul calendarelor pentru **Check In** și **Check Out**, apare un mesaj ce notifică acest fapt.

După căutare, din nou, în momentul apăsării butonului **Adaugă Rezervare** se va emite un alt mesaj care îi impune clientului să selecteze din listă camera la care vrea să facă rezervare. Apoi la acționarea butonului de **Adăugare Rezervare**, după ce s-au îndeplinit toate condițiile, se va face o verificare cu intervalul de date selectat verificând dacă pe camera respectivă mai există și alte rezervații ce pot încurca această rezervare, notificând clientul atunci când nu se poate.

**Adaugare Rezervare**

About Exit

Rezervari

Tipul Camerei: Deluxe

Capacitate (persoane): 4

Tip Balcon: Propriu

Cauta

Log Out

Camere care se potrivesc selectiei

Numar: 401 ; Pret/Noapte: 200 ; Etaj: 4

Check In: mai 2024

Check Out: mai 2024

Today: 28.05.2024

Adauga Rezervare

\*\*\*Rezervarea nu poate fi stearsa decat de catre MANAGER\*\*\*

**5. Manager:** În cazul în care la logare, se observă contul unui manager, acesta va fi redirecționat către o altă interfață, ce conține **Detalii Rezervări**, unde, de data aceasta se pot vedea toate rezervările clienților, **Detalii Camere**, unde se află detalii despre toate camerele hotelului și butoanele de **Adaugă Cameră**, **Șterge Cameră**, **Reîmprospătare** și **Șterge Rezervare**. În momentul adăugării unei camere, a ștergerii unei camere sau a ștergerii unei rezervări, este foarte bine ca utilizatorul să apese butonul **Reîmprospătare** pentru a putea vedea modificările efectuate. La apăsarea butonului **Adaugă Cameră**, manager-ul va fi redirecționat către o altă interfață pentru adăugarea detaliilor camerei pe care dorește să o adauge. **Atentie:** Ștergerea unei camere înseamnă implicit ștergerea tuturor rezervarilor ce aparțin de acea cameră.

**Manager**

About Exit

Log Out

Detalii Rezervari

ID	Client (ID)	Camera (ID)	Plata	Check in	Check out
ID: 1	Client (ID): 1	Camera (ID): 2	Plata: 400	Check in: 2024-08-07	Check out: 2024-08-07
ID: 2	Client (ID): 4	Camera (ID): 5	Plata: 175	Check in: 2024-09-21	Check out: 2024-09-21

Adauga Camera

Sterge Camera

Reimprospatate

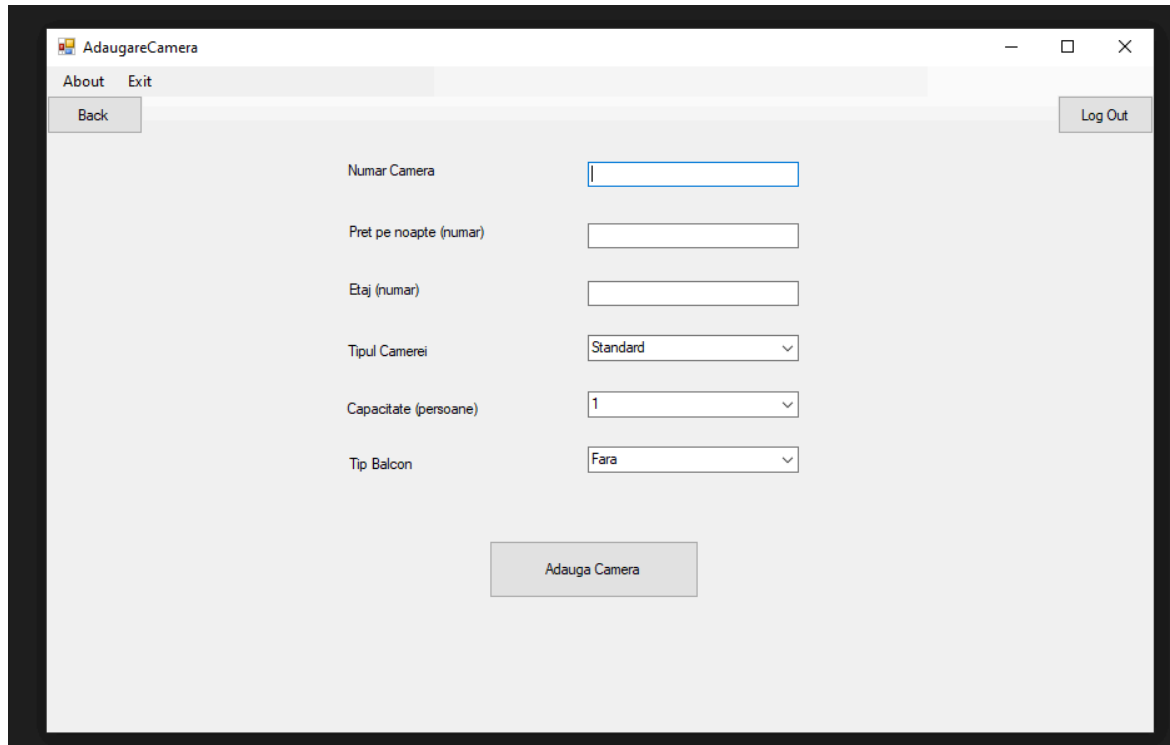
Sterge Rezervare

Detalii Camere

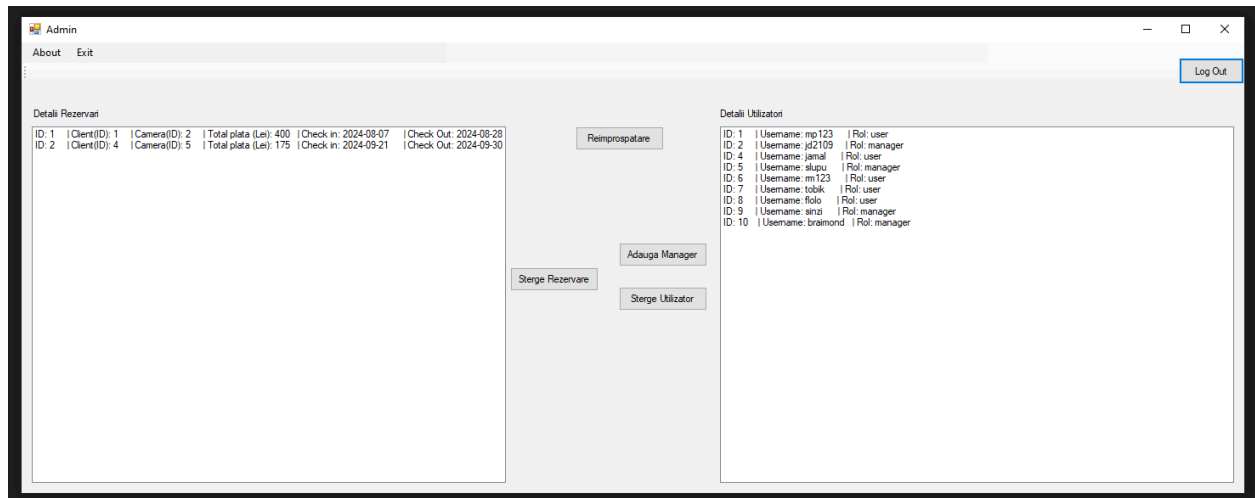
ID	Numar	Pret/Noapte	Etaj	Tip	Balcon	Nr. Persoane
ID: 1	Numar: 601	Pret/Noapte: 350	Etaj: 6	Tip: Deluxe	Balcon: Comun	Nr. Persoane: 4
ID: 2	Numar: 202	Pret/Noapte: 100	Etaj: 2	Tip: Standard	Balcon: Propriu	Nr. Persoane: 1
ID: 3	Numar: 101	Pret/Noapte: 175	Etaj: 1	Tip: Apartament	Balcon: Fara	Nr. Persoane: 2
ID: 4	Numar: 102	Pret/Noapte: 175	Etaj: 1	Tip: Apartament	Balcon: Fara	Nr. Persoane: 2
ID: 5	Numar: 103	Pret/Noapte: 175	Etaj: 1	Tip: Apartament	Balcon: Fara	Nr. Persoane: 2
ID: 6	Numar: 104	Pret/Noapte: 175	Etaj: 1	Tip: Apartament	Balcon: Fara	Nr. Persoane: 2
ID: 7	Numar: 201	Pret/Noapte: 100	Etaj: 2	Tip: Standard	Balcon: Propriu	Nr. Persoane: 1
ID: 8	Numar: 203	Pret/Noapte: 100	Etaj: 2	Tip: Standard	Balcon: Propriu	Nr. Persoane: 1
ID: 9	Numar: 204	Pret/Noapte: 100	Etaj: 2	Tip: Standard	Balcon: Propriu	Nr. Persoane: 1
ID: 10	Numar: 301	Pret/Noapte: 200	Etaj: 3	Tip: Apartament	Balcon: Comun	Nr. Persoane: 2
ID: 11	Numar: 302	Pret/Noapte: 200	Etaj: 3	Tip: Apartament	Balcon: Comun	Nr. Persoane: 2
ID: 12	Numar: 303	Pret/Noapte: 200	Etaj: 3	Tip: Apartament	Balcon: Comun	Nr. Persoane: 2
ID: 13	Numar: 304	Pret/Noapte: 200	Etaj: 3	Tip: Apartament	Balcon: Comun	Nr. Persoane: 2
ID: 14	Numar: 602	Pret/Noapte: 350	Etaj: 6	Tip: Deluxe	Balcon: Comun	Nr. Persoane: 4
ID: 15	Numar: 205	Pret/Noapte: 175	Etaj: 2	Tip: Apartament	Balcon: Fara	Nr. Persoane: 2
ID: 16	Numar: 206	Pret/Noapte: 175	Etaj: 2	Tip: Apartament	Balcon: Fara	Nr. Persoane: 2
ID: 17	Numar: 235	Pret/Noapte: 235	Etaj: 2	Tip: Standard	Balcon: Propriu	Nr. Persoane: 3
ID: 18	Numar: 401	Pret/Noapte: 200	Etaj: 4	Tip: Deluxe	Balcon: Propriu	Nr. Persoane: 4

**6. Adăugare Cameră:** Dacă din *Manager* se apasă butonul **Adaugă Cameră**, utilizatorul va fi redirecționat către această interfață *Adăugare Cameră*. Pentru adăugarea unei camere noi, trebuie completate câmpurile **Număr Cameră**, **Preț pe noapte**, **Etaj**, **Tipul Camerei**, **Capacitate** și **Tip Balcon**. După completarea corectă a tuturor câmpurilor, se va apăsa butonul **Adaugă Cameră** pentru ca aceasta să fie adăugată. La acționarea butonului de adăugare se verifică existența numărului camerei respective, notificând astfel manager-ul dacă există deja

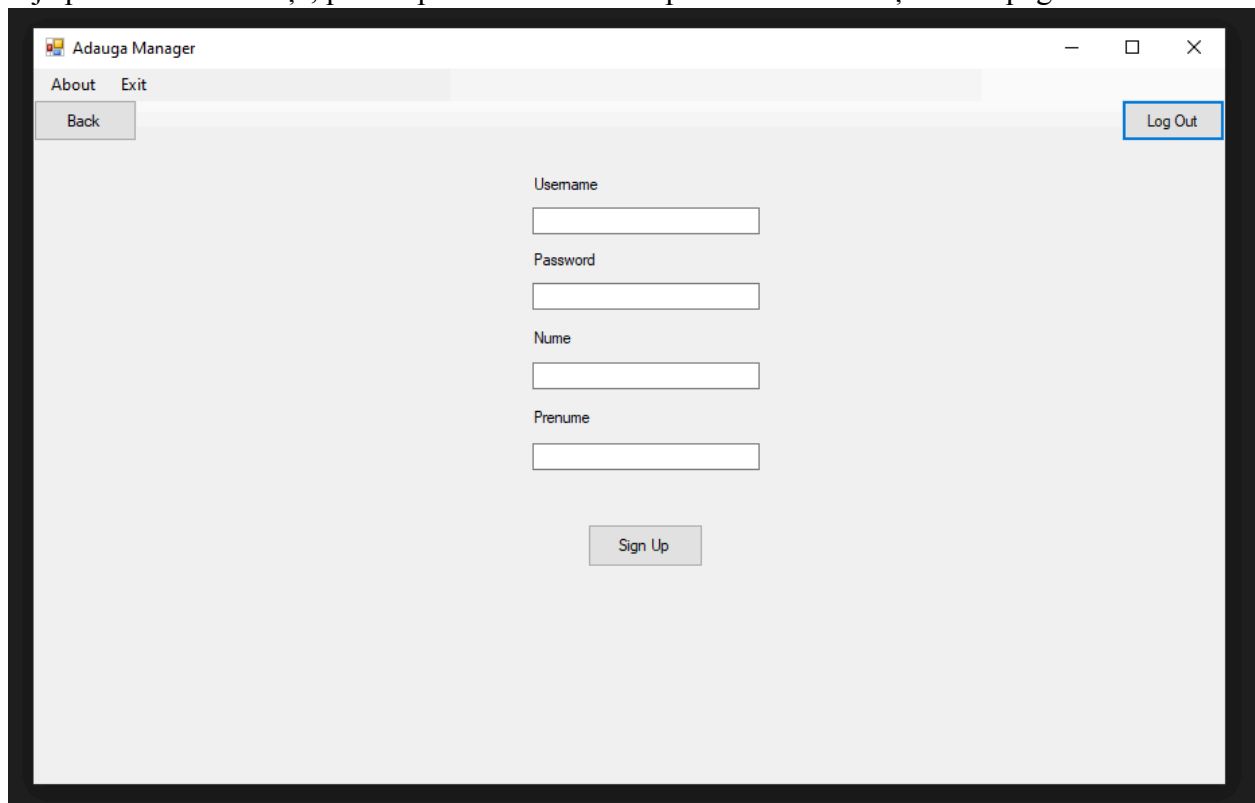
camera. În cazul în care manager-ul nu mai dorește adăugarea unei camere, poate reveni la pagina anterioară prin apăsarea butonului **Back**.



**7. Admin:** Dacă în momentul logării contul este al unui administrator, acesta va fi redirecționat către pagina *Admin*. Această pagină conține **Detalii Rezervări** (Detaliile tuturor rezervărilor), **Detalii Utilizatori** (Id-ul, Username-ul și rolul fiecărui utilizator existent) și butoanele **Adaugă Manager**, **Șterge Manager**, **Șterge Utilizator** și **Reîmprospătare**. Pe baza acestor funcționalități, administratorul poate vedea atât detaliile rezervărilor, cât și detaliile utilizatorilor, poate șterge sau adăuga manageri, poate șterge utilizatori și desigur poate reîmprospăta pagina pentru a putea vedea modificările efectuate. **Atentie:** Ștergerea unui utilizator cu rol de client duce implicit la ștergerea tuturor rezervărilor care au fost create de acel client.



**8. Adaugă Manager:** După apăsarea butonului **Adaugă Manager** din interfața *Admin*, utilizatorul va fi redirecționat către interfața *Adaugă Manager*. Această interfață conține câmpurile **Username**, **Password**, **Nume** și **Prenume** precum și butoanele **Sign Up** și **Back**. După completarea corectă a tuturor câmpurilor, la apăsarea butonului **Sign Up**, adminul adaugă un nou manager. În cazul în care acesta nu mai dorește să adauge un nou manager, dar a intrat deja pe această interfață, poate apăsa butonul **Back** pentru a fi redirecționat la pagina anterioară.





## 3.2 Interfețe hardware

Interfețele hardware pentru aplicația RoomWizard includ un mouse standard, un monitor de afișare și o tastatură. Nu este necesar un hardware complex pentru a rula aplicația, fiind suficient un computer obișnuit cu un monitor. Toată interacțiunea cu aplicația se realizează prin apăsarea click stânga pe mouse și completarea câmpurilor necesare cu ajutorul tastaturii. Aceste interfețe sunt proiectate pentru a asigura o utilizare eficientă și intuitivă a aplicației, oferind o experiență de utilizare optimă.

## 3.3 Interfețe software

Pentru a putea rula cu succes aplicația RoomWizard, utilizatorul trebuie să aibă instalat:

- ❖ **.NET Framework:** Minim versiunea 4.5.
- ❖ **Sistem de Operare:** Windows 10 sau versiuni ulterioare.
- ❖ **Bază de Date:** MySQL versiunea 8.0, pentru stocarea și gestionarea datelor.
- ❖ **Framework GUI:** Qt versiunea 5.15, pentru dezvoltarea interfeței grafice.

Toate funcționalitățile necesare lansării în execuție și utilizării corecte a aplicației sunt incluse în aceasta și nu necesită produse sau tehnologii software suplimentare. Această configurație asigură o interfață intuitivă și eficientă pentru utilizatori, facilitând gestionarea rapidă și sigură a rezervărilor hoteliere.

# 4. Caracteristici ale sistemului

*Această secțiune conține toate cerințele implementate în cadrul aplicației.*

## 4.1 Vizualizarea rezervărilor clientului

### 4.1.1 Descriere și Prioritate

Formularul pentru vizualizarea rezervărilor clientului permite utilizatorului să vadă rezervările curente. Informațiile despre rezervări sunt extrase din baza de date și afișate într-o interfață prietenoasă. Utilizatorul poate, de asemenea, să se deconecteze sau să adauge noi rezervări. Prioritatea acestei funcționalități este mare, deoarece oferă utilizatorului o modalitate de a gestiona rezervările.

### 4.1.2 Secvențe Stimulus/Răspuns

1. Utilizatorul accesează formularul de vizualizare a rezervărilor.
2. Sistemul extrage informațiile despre rezervări din baza de date.
3. Sistemul afișează informațiile în interfață.
4. Utilizatorul poate alege să se deconecteze.
5. Utilizatorul poate alege să adauge o nouă rezervare.

### 4.1.3 Cerințe Funcționale

- ❖ **REQ-1:** Sistemul trebuie să afișeze rezervările existente ale utilizatorului.
  - Intrare: Sistemul extrage datele de rezervare din baza de date.
  - Ieșire: Rezervările sunt afișate într-o listă pe interfața utilizatorului.
- ❖ **REQ-2:** Sistemul trebuie să permită deconectarea utilizatorului.
  - Intrare: Utilizatorul selectează opțiunea de deconectare.
  - Ieșire: Utilizatorul este redirecționat la ecranul de autentificare.
- ❖ **REQ-3:** Sistemul trebuie să permită adăugarea unei noi rezervări.
  - Intrare: Utilizatorul selectează opțiunea de adăugare a unei noi rezervări.
  - Ieșire: Este afișată interfața pentru introducerea detaliilor noii rezervări.

## 4.2 Adăugarea unei noi rezervări

### 4.2.1 Descriere și Prioritate

Formularul pentru adăugarea unei noi rezervări permite utilizatorului să introducă detaliile necesare pentru a crea o nouă rezervare. Prioritatea acestei funcționalități este mare, deoarece este esențială pentru gestionarea rezervărilor de către utilizatori.

### 4.2.2 Secvențe Stimulus/Răspuns

1. Utilizatorul accesează interfața pentru adăugarea unei noi rezervări.
2. Sistemul afișează formularul pentru introducerea detaliilor rezervării.
3. Utilizatorul completează detaliile rezervării și trimite formularul.
4. Sistemul salvează noile informații în baza de date.
5. Sistemul confirmă succesul operațiunii și actualizează lista de rezervări.

### 4.2.3 Cerințe Funcționale

- ❖ **REQ-1:** Sistemul trebuie să afișeze formularul pentru adăugarea unei noi rezervări.
  - Intrare: Utilizatorul selectează opțiunea de adăugare a unei noi rezervări.
  - Ieșire: Formularul pentru introducerea detaliilor este afișat.
- ❖ **REQ-2:** Sistemul trebuie să salveze detaliile noii rezervări.
  - Intrare: Utilizatorul completează și trimite formularul de rezervare.
  - Ieșire: Noile detalii ale rezervării sunt salvate în baza de date.
- ❖ **REQ-3:** Sistemul trebuie să confirme succesul adăugării rezervării.
  - Intrare: Sistemul salvează noile detalii ale rezervării.
  - Ieșire: Utilizatorul este notificat despre succesul operațiunii.

## 4.3 Adăugarea unui manager

### 4.3.1 Descriere și Prioritate

Formularul pentru adăugarea unui manager permite administratorilor să introducă detaliile necesare pentru a crea un nou cont de manager. Această funcționalitate este esențială pentru gestionarea și administrarea eficientă a sistemului, având o prioritate mare.

### 4.3.2 Secvențe Stimulus/Răspuns

1. Administratorul accesează formularul pentru adăugarea unui manager.
2. Sistemul afișează formularul pentru introducerea detaliilor noului manager.
3. Administratorul completează detaliile necesare și trimite formularul.
4. Sistemul verifică datele introduse.
5. Dacă datele sunt valide, sistemul salvează informațiile noului manager în baza de date.
6. Sistemul confirmă succesul operațiunii și notifică administratorul.

### 4.3.3 Cerințe Funcționale

- ❖ **REQ-1:** Sistemul trebuie să afișeze formularul pentru adăugarea unui manager.
  - Intrare: Administratorul selectează opțiunea de adăugare a unui manager.
  - Ieșire: Formularul pentru introducerea detaliilor managerului este afișat.
- ❖ **REQ-2:** Sistemul trebuie să verifice corectitudinea datelor introduse de administrator.
  - Intrare: Administratorul completează și trimite formularul de adăugare a managerului.
  - Ieșire: Sistemul validează datele introduse.
- ❖ **REQ-3:** Sistemul trebuie să salveze detaliile noului manager.
  - Intrare: Datele introduse de administrator sunt validate.
  - Ieșire: Informațiile noului manager sunt salvate în baza de date.
- ❖ **REQ-4:** Sistemul trebuie să confirme succesul adăugării managerului.
  - Intrare: Sistemul salvează informațiile noului manager.
  - Ieșire: Administratorul este notificat despre succesul operațiunii.

## 4.4 Ștergerea unui manager

### 4.4.1 Descriere și Prioritate

Butonul pentru ștergerea unui manager permite administratorilor să elimine conturile managerilor existenți. Această funcționalitate este importantă pentru menținerea securității și integrității sistemului, având o prioritate mare.

### 4.4.2 Secvențe Stimulus/Răspuns

1. Administratorul accesează lista managerilor existenți.
2. Administratorul selectează un manager pentru ștergere.
3. Sistemul solicită confirmarea ștergerii.
4. Administratorul confirmă ștergerea.
5. Sistemul verifică drepturile administratorului.
6. Dacă drepturile sunt valide, sistemul șterge contul managerului din baza de date.
7. Sistemul confirmă succesul operațiunii și actualizează lista managerilor.

### 4.4.3 Cerințe Funcționale

- ❖ **REQ-1:** Sistemul trebuie să afișeze lista managerilor existenți.
  - Intrare: Administratorul accesează secțiunea de administrare a managerilor.
  - Ieșire: Lista managerilor este afișată.
- ❖ **REQ-2:** Sistemul trebuie să permită selecția unui manager pentru ștergere.
  - Intrare: Administratorul selectează un manager din listă.

- Ieșire: Opțiunea de ștergere este activată.
- ❖ **REQ-3:** Sistemul trebuie să solicite confirmarea ștergerii.
  - Intrare: Administratorul selectează opțiunea de ștergere.
  - Ieșire: Sistemul afișează un mesaj de confirmare.
- ❖ **REQ-4:** Sistemul trebuie să verifice drepturile administratorului înainte de ștergere.
  - Intrare: Administratorul confirmă ștergerea.
  - Ieșire: Sistemul verifică drepturile de acces ale administratorului.
- ❖ **REQ-5:** Sistemul trebuie să șteargă contul managerului din baza de date.
  - Intrare: Drepturile administratorului sunt validate.
  - Ieșire: Contul managerului este șters din baza de date.
- ❖ **REQ-6:** Sistemul trebuie să confirme succesul operațiunii de ștergere.
  - Intrare: Contul managerului este șters.
  - Ieșire: Administratorul este notificat despre succesul operațiunii și lista managerilor este actualizată.

## 4.5 Ștergerea unei camere

### 4.5.1 Descriere și Prioritate

Funcționalitatea de ștergere a unei camere permite managerilor să elimine camerele existente din sistem. Această funcționalitate este esențială pentru menținerea și actualizarea inventarului de camere disponibile, având o prioritate mare.

### 4.5.2 Secvențe Stimulus/Răspuns

1. Managerul accesează lista camerelor existente.
2. Managerul selectează o cameră pentru ștergere.
3. Sistemul solicită confirmarea ștergerii.
4. Managerul confirmă ștergerea.
5. Sistemul șterge camera din baza de date.
6. Sistemul confirmă succesul operațiunii și actualizează lista camerelor.

### 4.5.3 Cerințe Funcționale

- ❖ **REQ-1:** Sistemul trebuie să afișeze lista camerelor existente.
  - Intrare: Managerul accesează secțiunea de administrare a camerelor.
  - Ieșire: Lista camerelor este afișată.
- ❖ **REQ-2:** Sistemul trebuie să permită selecția unei camere pentru ștergere.
  - Intrare: Managerul selectează o cameră din listă.
  - Ieșire: Opțiunea de ștergere este activată.
- ❖ **REQ-3:** Sistemul trebuie să solicite confirmarea ștergerii.
  - Intrare: Managerul selectează opțiunea de ștergere.
  - Ieșire: Sistemul afișează un mesaj de confirmare.
- ❖ **REQ-4:** Sistemul trebuie să verifice dacă a fost selectată o singură cameră.
  - Intrare: Managerul confirmă ștergerea.
  - Ieșire: Sistemul verifică dacă doar o cameră a fost selectată.
- ❖ **REQ-5:** Sistemul trebuie să șteargă camera selectată din baza de date.
  - Intrare: Camera selectată este validată.
  - Ieșire: Camera este ștearsă din baza de date.

- ❖ **REQ-6:** Sistemul trebuie să confirme succesul operațiunii de ștergere.
  - Intrare: Camera este ștearsă.
  - Ieșire: Administratorul este notificat despre succesul operațiunii și lista camerelor este actualizată.

## 4.6 Fereastra ajutor

### 4.6.1 Descriere și Prioritate

Această funcționalitate oferă utilizatorilor informații detaliate despre modul de utilizare a aplicației. Prioritatea acestei funcționalități este mare, deoarece ajută utilizatorii să înțeleagă cum să folosească eficient aplicația și să beneficieze de toate funcționalitățile acesteia.

### 4.6.2 Secvențe Stimulus/Răspuns

1. Utilizatorul accesează meniul de Ajutor din interfață.
2. Sistemul afișează fereastra care conține informațiile necesare.

### 4.6.3 Cerințe Funcționale

- ❖ **REQ-1:** Sistemul trebuie să ofere utilizatorului un meniu de Ajutor care să conțină informații despre utilizarea aplicației.
  - Intrare: Utilizatorul selectează butonul de ajutor din meniul principal.
  - Ieșire: Se deschide fereastra de ajutor care conține documentația necesară.

Aceste secțiuni sunt structurate pentru a oferi o descriere clară a funcționalităților și cerințelor asociate fiecărei caracteristici a sistemului.

## 5. Cerințe non-funcționale

### 5.1 Cerințe de performanță

Aplicația hotelieră funcționează eficient chiar și în condiții mai puțin favorabile, cum ar fi variații de temperatură ale hardware-ului pe care este rulată. Performanța aplicației nu depinde de o conexiune neîntreruptă la internet, asigurând astfel o funcționare stabilă și în medii cu conexiuni intermitente. Sistemul este proiectat să gestioneze eficient toate interacțiunile cu baza de date, garantând răspunsuri rapide pentru operațiuni precum adăugarea de camere, verificarea disponibilității și gestionarea rezervărilor. De exemplu, atunci când se adaugă o nouă cameră, aplicația verifică dacă aceasta există deja în baza de date și generează un ID unic pentru a preveni conflictele. Astfel, aplicația asigură o experiență fluidă și rapidă pentru utilizatori, indiferent de condițiile externe.

## 5.2 Cerințe de siguranță

Aplicația este concepută pentru a asigura protecția utilizatorilor și a datelor acestora. Toate operațiunile critice, cum ar fi adăugarea și actualizarea datelor, sunt supuse unor verificări pentru a preveni erorile și pierderile de date. Aplicația nu prezintă riscuri pentru utilizatori și este stabilă și fiabilă în orice condiții de utilizare.

Totodată, informațiile sensibile de tip *parola* nu ajung vreodată să fie vizualizate de către altcineva înafară de utilizator. Aplicația predis pune de un mod de ascundere a parolei, astfel, în baza de date nu se regăsește șirul de caractere scris de către utilizator ci un șir de caractere ce corespunde în mod unic acelei parole.

## 5.3 Cerințe de securitate

Securitatea aplicației hoteliere este esențială pentru protejarea datelor utilizatorilor și prevenirea accesului neautorizat. Fiecare utilizator are un cont individual protejat, asigurând astfel că datele personale și rezervările sunt accesibile doar de către utilizatorul autorizat. În timpul autentificării, aplicația verifică existența contului în baza de date și validează parolele folosind algoritmi de hash și criptare pentru a preveni furtul de informații sensibile.

## 5.4 Atribute de calitate

Aplicația trebuie să funcționeze consistent și să gestioneze corect toate operațiunile, prevenind pierderile de date sau erorile sistemice. Interfața aplicației trebuie să fie intuitivă și ușor de utilizat, reducând necesitatea instruirii utilizatorilor. Codul aplicației trebuie să fie structurat și documentat astfel încât să permită mentenanța și actualizarea facilă de către dezvoltatori.

## 5.5 Reguli de afaceri

Funcționalitățile aplicației trebuie să fie accesibile doar persoanelor autorizate, cu permisiuni definite pentru diferite roluri (client, manager, admin). Fiecare utilizator trebuie să aibă un cont individual cu roluri și permisiuni specifice pentru a asigura securitatea operațiunilor. Datele personale ale clienților trebuie să fie protejate și accesibile doar personalului autorizat, conform reglementărilor de protecție a datelor.

## 6. Alte cerințe

### 6.1 Baza de date

Sistemul va înregistra și actualiza instantaneu modificările adăugate de utilizator, garantând coerența și precizia datelor. Se va actualiza baza de date de câte ori este nevoie pentru a fi în timp real cu rezervările făcute de clienți sau șterse de către manageri.

#### Descrierea tabelelor

Baza de date a aplicației hoteliere are trei tabele, aceste sunt **Utilizatori**, **Cameră** și **Rezervare**.

**Tabela Utilizatori:** tabela are ca și componente o cheie primară (id\_utilizator), nume, prenume, username, vârsta ( $\leq 18$  ani), rolul (client, manager, admin), email și telefon.

**Tabela Cameră:** câmpurile tabelii sunt id\_cameră (cheia primară), număr\_cameră, preț\_per\_noapte, etaj, tip\_cameră, tip\_balcon și capacitate.

**Tabela Rezervare:** această tabelă conține câmpurile id\_rezervare, id\_client, id\_cameră, total\_plată, check\_in și check\_out.

### 6.2 Modelul datelor

#### 6.2.1 Structuri de date globale

Aplicația are o instanță globală a clasei de conexiune la baza de date, folosind șablonul de proiectare *Singleton*. Acest lucru va permite tuturor modulelor din aplicație să aibă acces la conexiunea la baza de date și să efectueze operațiuni CRUD pe tabelele respective. De asemenea, aplicația va utiliza o structură de date globală pentru gestionarea stărilor diverse, cum ar fi rezervările, implementată folosind șablonul de proiectare State.

#### 6.2.2 Structuri de date de legătură

Aplicația se va conecta la o bază de date SQL pentru a stoca și recupera date despre clienți, rezervări, camere disponibile și alte informații relevante pentru funcționarea hotelului. Modulul interfață grafică al aplicației va comunica cu modulul de baze de date prin intermediul argumentelor care conțin cheile primare din schema bazei de date (de exemplu, identificatorii de utilizatori sau de rezervări). Aplicația va utiliza structuri de date de legătură pentru a gestiona aceste comunicări între module și pentru a asigura integritatea datelor.

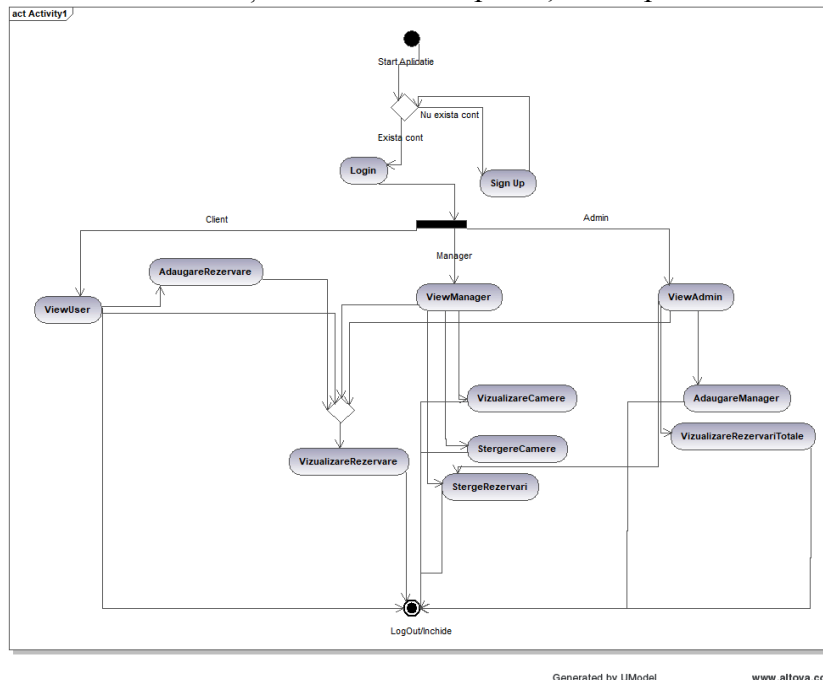
#### 6.2.3 Structuri de date temporare

Aplicația nu va utiliza structuri de date temporare care să presupună un consum semnificativ de resurse de memorie.

## 6.3 Diagrame UML

### 6.3.1 Diagrama de activități

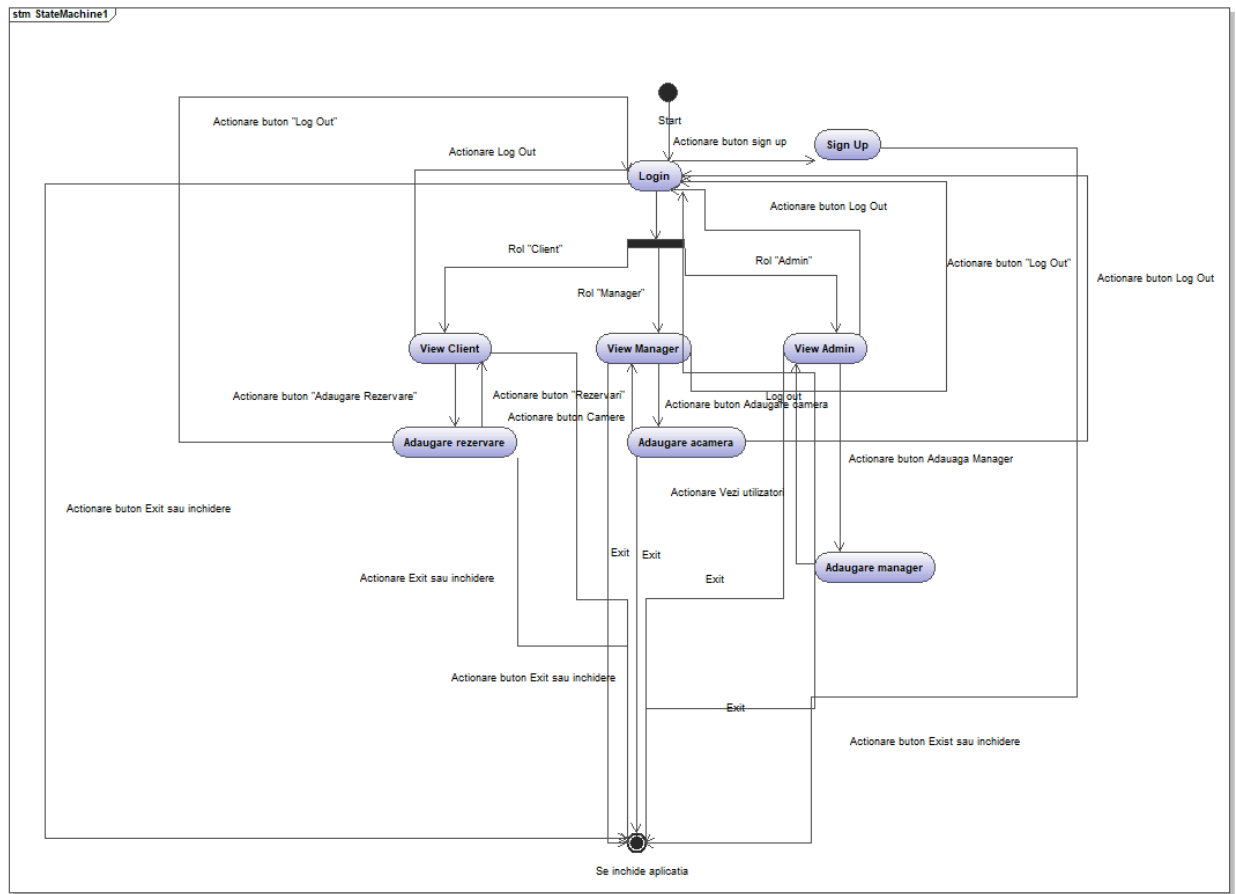
Această diagramă arată pas cu pas activitățile implicate în diferite funcții ale aplicației, cum ar fi procesul de autentificare, view-urile pentru fiecare utilizator în parte (client, admin, manager), cum se adaugă o rezervare sau o cameră, cum se adaugă un manager și, de asemenea, cum se șterge o rezervare, o cameră sau un manager. Acest lucru ajută la înțelegerea precisă a fluxurilor de lucru și la identificarea potențialelor puncte de ineficiență sau erori.



### 6.3.2 Diagrama StateMachine

Aceasta diagramă are ca scop descrierea modului de funcționare a modelului de proiectare intitulat „State Pattern”. Fiecare acțiune a utilizatorului îl poate trimite într-o altă stare. Dar mare parte din stări depind și de rolul pe care îl au utilizatorii (client, manager, sau admin).



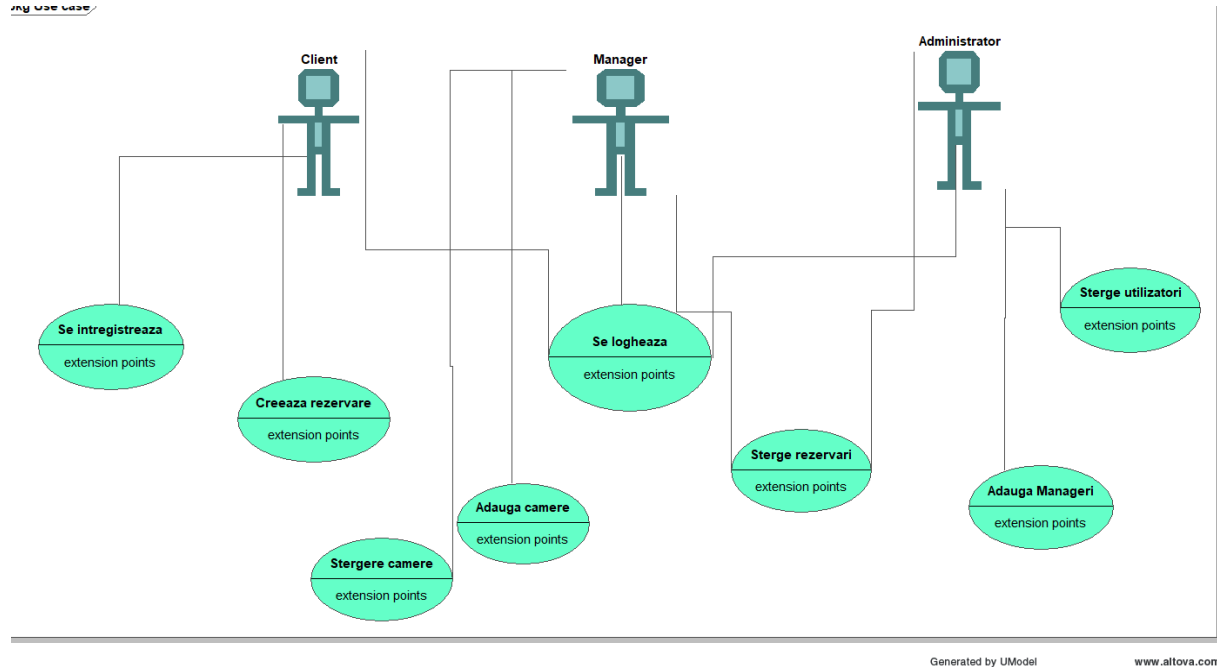


Generated by UModel

www.altova.com

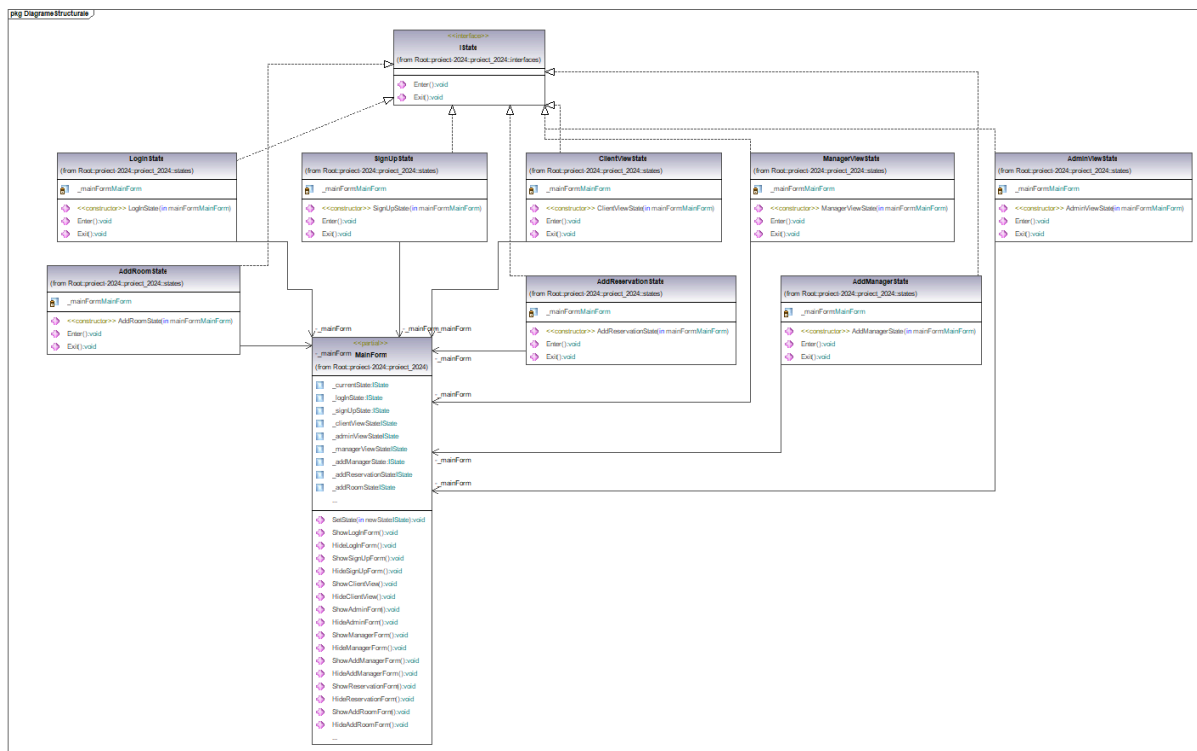
### 6.3.3 Diagrama cazurilor de utilizare

Diagrama cazurilor de utilizare are ca scop descrierea interacțiunii dintre utilizatori și sistem. Prin construirea cazurilor de utilizare, putem descrie întregul sistem într-o manieră clară și pentru a defini și a organiza cerințele funcționale ale aplicației. Astfel, se clarifică modul în care diferiți utilizatori interacționează cu aplicația și funcțiile principale pe care clientul, manager-ul și admin-ul le pot exercita în cadrul aplicației.

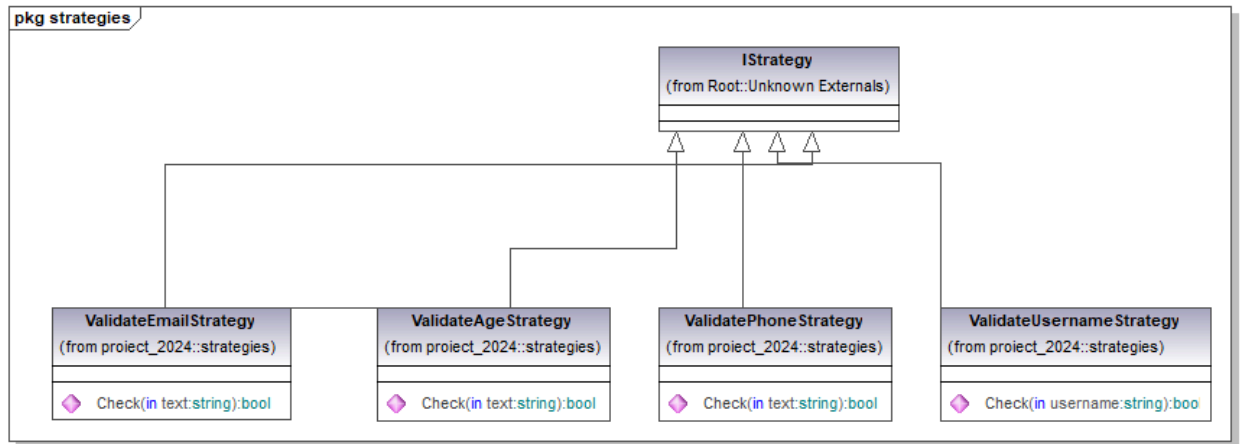


## 6.3.4 Diagramele modelelor de programare

### ❖ Diagrama de implementare a modelului „State Pattern”



## ❖ Diagrama de implementare a modelului strategie

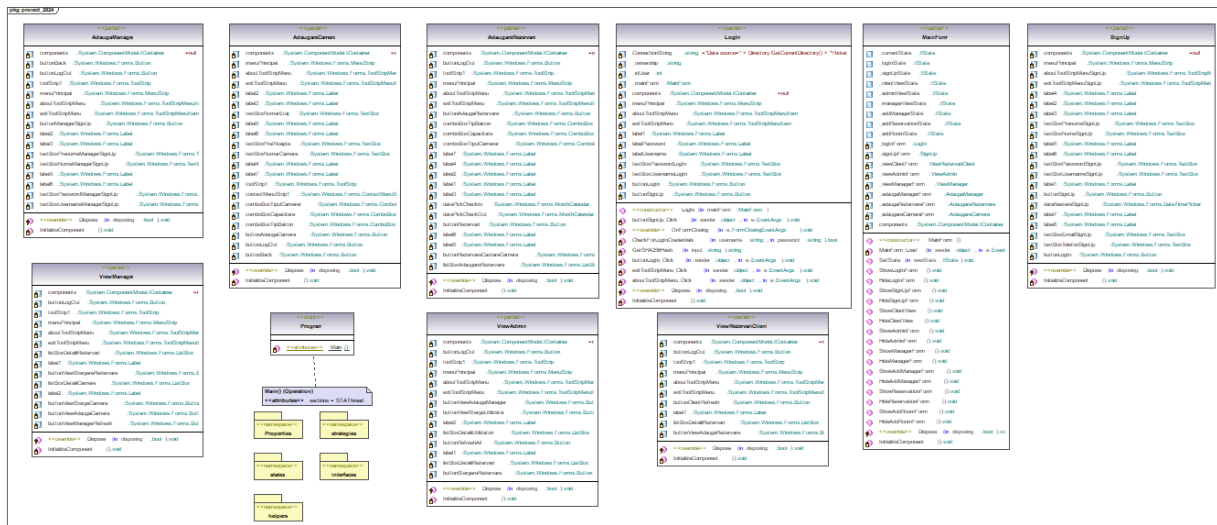


Generated by UModel

www.altova.com

## 6.3.5 Diagrama de clase per namespace-uri

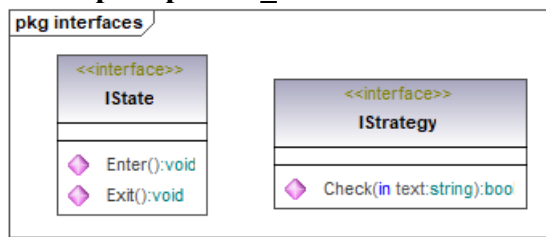
### ❖ Namespace proiect\_2024



Generated by UModel

www.altova.com

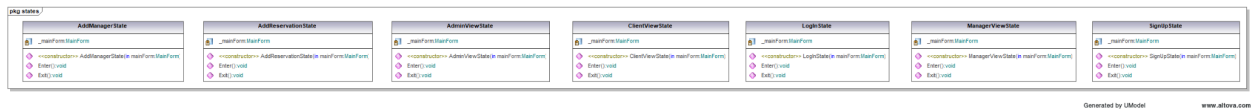
### ❖ Namespace project\_2024.interfaces



Generated by UModel

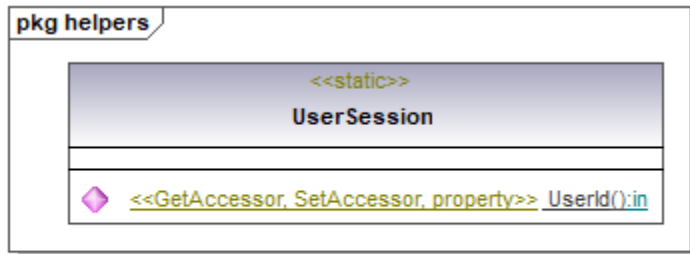
www.altova.com

## ❖ Namespace project\_2024.states



Generated by UModel [www.altova.com](http://www.altova.com)

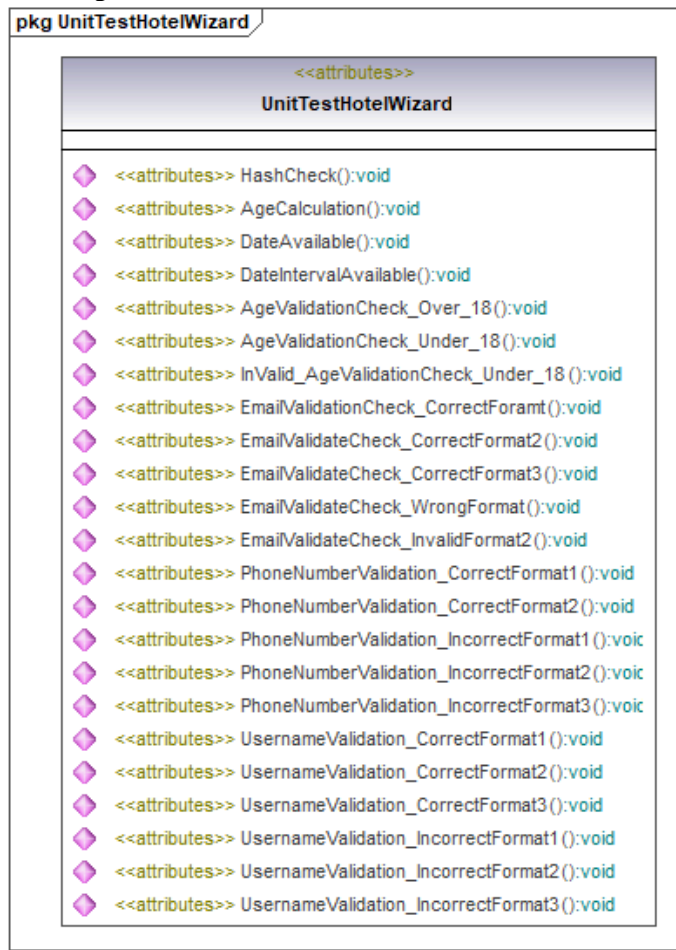
## ❖ Namespace project\_2024.helpers



Generated by UModel

[www.altova.com](http://www.altova.com)

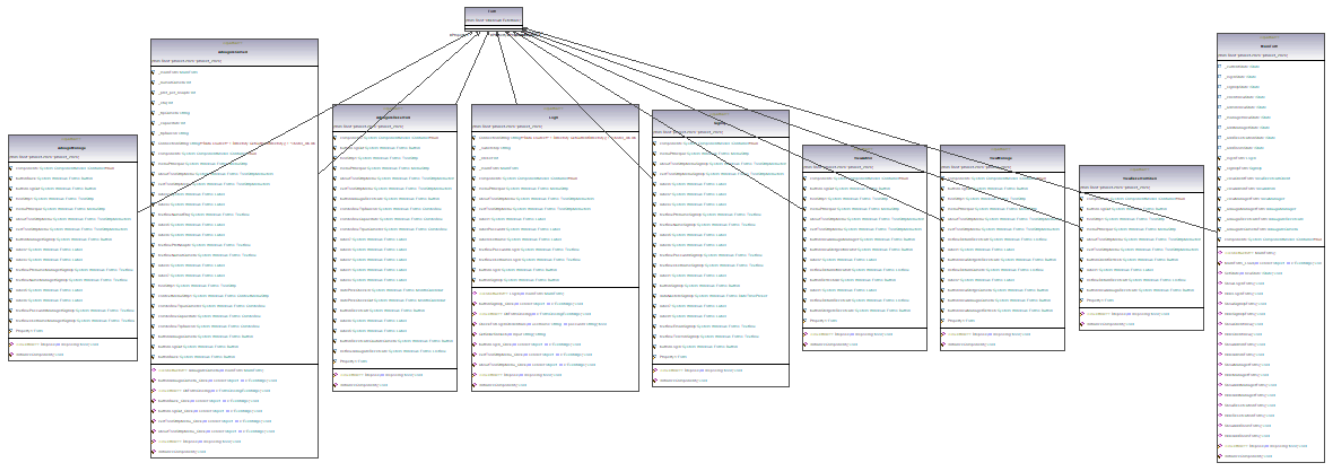
## ❖ Namespace UnitTestHotelWizard



Generated by UModel

[www.altova.com](http://www.altova.com)

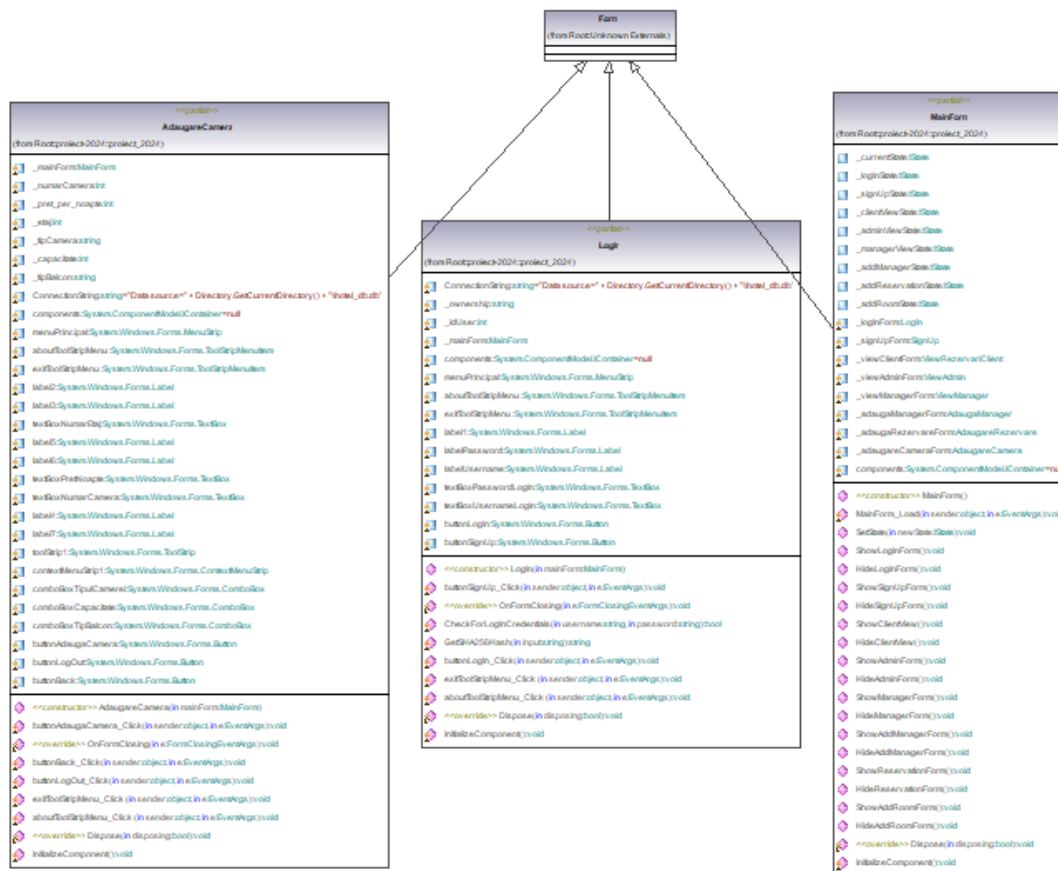
### 6.3.6 Diagrama SInForm



Generated by UModel

www.altova.com

### 6.3.7 Diagrama DependenteForm



Generated by UModel

www.altova.com

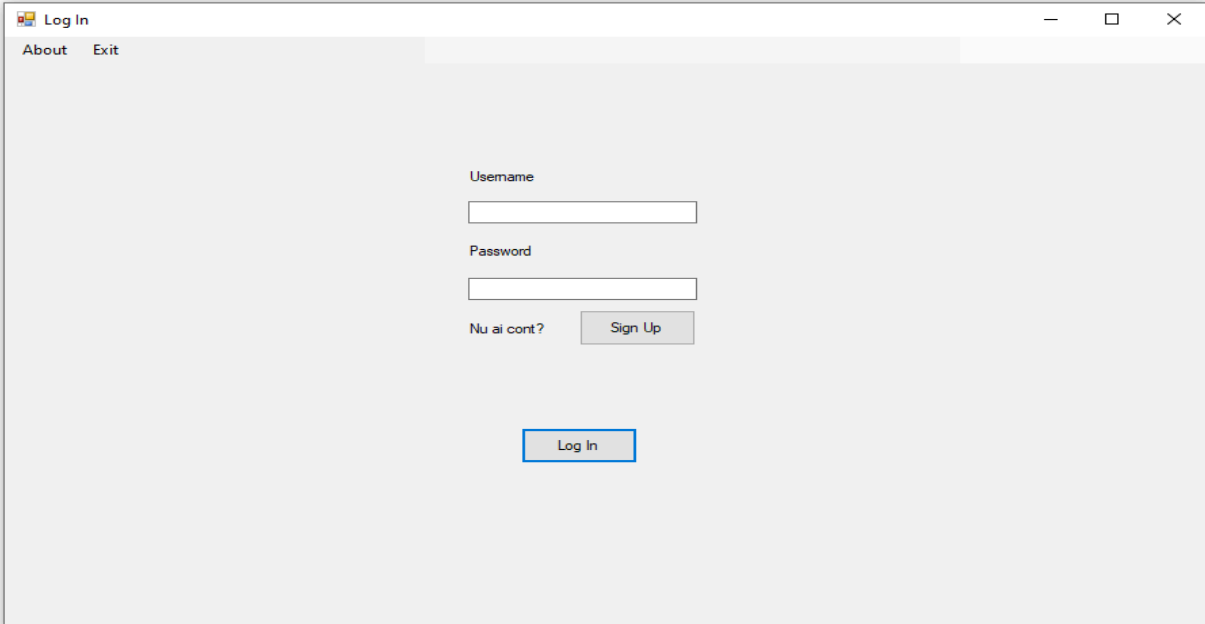
## 6.4 Modul de utilizare al programului (*Help*)

Acest *Help* a fost realizat cu intenția de a ghida utilizatorul în momentele în care aplicația îi poate crea dificultăți. Să numim acest *Help* „Manualul de utilizare” al aplicației.

Acest manual de utilizare este disponibil oricând în momentul rulării aplicației la acționarea butonului de „About”. Acesta conține informații detaliate a modului de funcționare a aplicației atât în scris cât și într-un mod vizual ușor de înțeles.

## 6.5 Execuție program (capturi de ecran din diverse situații)

Log In:



The screenshot shows a web browser window titled "Log In". The page has a light gray background. At the top left, there are two links: "About" and "Exit". In the center, there is a login form with two input fields: "Username" and "Password". Below the "Password" field, there is a link "Nu ai cont?" and a "Sign Up" button. At the bottom center, there is a "Log In" button. The browser window has standard Windows-style window controls (minimize, maximize, close) in the top right corner.

Sign Up:

Sign Up

About Exit

Username

Password

Nume

Prenume

Telefon

Email

Data Nastere

[Ai deja cont?](#)

View Client:

Rezervari

About Exit

Detalii Rezervari

ID: 2 ; Camera: 103 ; Cost ( Lei ): 175 ; Check In: 21.09.2024 ; Check Out: 30.09.2024 ;

View Adaugare rezervare:

Adaugare Rezervare

AboutExit

Rezervari

Tipul CamereiStandard

Capacitate (persoane)1

Tip BalconFara

Cauta

Log Out

Camere care se potrivesc selectiei

Check In

mai 2024

lun. mar. mie. joi vin. sam. dum.

29 30 1 2 3 4 5

6 7 8 9 10 11 12

13 14 15 16 17 18 19

20 21 22 23 24 25 26

27 28 29 30 31 1 2

3 4 5 6 7 8 9

Today: 29.05.2024

Check Out

mai 2024

lun. mar. mie. joi vin. sam. dum.

29 30 1 2 3 4 5

6 7 8 9 10 11 12

13 14 15 16 17 18 19

20 21 22 23 24 25 26

27 28 29 30 31 1 2

3 4 5 6 7 8 9

Today: 29.05.2024

Adauga Rezervare

\*\*\*\*Rezervarea nu poate fi stearsa decat de catre MANAGER\*\*\*\*

View Manager:

Manager

AboutExit

Log Out

Detali Rezervari

ID: 1 | Client (ID): 1 | Camera (ID): 2 | Plata: 400 | Check in: 2024-08-07 | Check out: 2024-08-07

ID: 2 | Client (ID): 4 | Camera (ID): 5 | Plata: 175 | Check in: 2024-09-21 | Check out: 2024-09-21

ID: 3 | Client (ID): 11 | Camera (ID): 9 | Plata: 300 | Check in: 2024-07-24 | Check out: 2024-07-24

Adauga Camera

Sterge Camera

Reinprospatare

Sterge Rezervare

Detali Camere

ID: 1 : Numar: 601 : Pret/Noapte: 350 : Etaj: 6 : Tip: Deluxe : Balcon: Comun : Nr. Persoane: 4

ID: 2 : Numar: 202 : Pret/Noapte: 100 : Etaj: 2 : Tip: Standard : Balcon: Propriu : Nr. Persoane: 1

ID: 3 : Numar: 101 : Pret/Noapte: 175 : Etaj: 1 : Tip: Apartament : Balcon: Fara : Nr. Persoane: 2

ID: 4 : Numar: 102 : Pret/Noapte: 175 : Etaj: 1 : Tip: Apartament : Balcon: Fara : Nr. Persoane: 2

ID: 5 : Numar: 103 : Pret/Noapte: 175 : Etaj: 1 : Tip: Apartament : Balcon: Fara : Nr. Persoane: 2

ID: 6 : Numar: 104 : Pret/Noapte: 175 : Etaj: 1 : Tip: Apartament : Balcon: Fara : Nr. Persoane: 2

ID: 7 : Numar: 201 : Pret/Noapte: 100 : Etaj: 2 : Tip: Standard : Balcon: Propriu : Nr. Persoane: 1

ID: 8 : Numar: 203 : Pret/Noapte: 100 : Etaj: 2 : Tip: Standard : Balcon: Propriu : Nr. Persoane: 1

ID: 9 : Numar: 204 : Pret/Noapte: 100 : Etaj: 2 : Tip: Standard : Balcon: Propriu : Nr. Persoane: 1

ID: 10 : Numar: 301 : Pret/Noapte: 200 : Etaj: 3 : Tip: Apartament : Balcon: Comun : Nr. Persoane: 2

ID: 11 : Numar: 302 : Pret/Noapte: 200 : Etaj: 3 : Tip: Apartament : Balcon: Comun : Nr. Persoane: 2

ID: 12 : Numar: 303 : Pret/Noapte: 200 : Etaj: 3 : Tip: Apartament : Balcon: Comun : Nr. Persoane: 2

ID: 13 : Numar: 304 : Pret/Noapte: 200 : Etaj: 3 : Tip: Apartament : Balcon: Comun : Nr. Persoane: 2

ID: 14 : Numar: 602 : Pret/Noapte: 350 : Etaj: 6 : Tip: Deluxe : Balcon: Comun : Nr. Persoane: 4

ID: 15 : Numar: 205 : Pret/Noapte: 175 : Etaj: 2 : Tip: Apartament : Balcon: Fara : Nr. Persoane: 2

ID: 16 : Numar: 206 : Pret/Noapte: 175 : Etaj: 2 : Tip: Apartament : Balcon: Fara : Nr. Persoane: 2

ID: 17 : Numar: 235 : Pret/Noapte: 235 : Etaj: 2 : Tip: Standard : Balcon: Propriu : Nr. Persoane: 3

ID: 18 : Numar: 401 : Pret/Noapte: 200 : Etaj: 4 : Tip: Deluxe : Balcon: Propriu : Nr. Persoane: 4

View Admin:

Admin

AboutExit

Log Out

Detali Rezervari

ID: 1 | Client(ID): 1 | Camera(ID): 2 | Total plata (Lei): 400 | Check in: 2024-08-07 | Check Out: 2024-08-28

ID: 2 | Client(ID): 4 | Camera(ID): 5 | Total plata (Lei): 175 | Check in: 2024-09-21 | Check Out: 2024-09-30

ID: 3 | Client(ID): 11 | Camera(ID): 9 | Total plata (Lei): 300 | Check in: 2024-07-24 | Check Out: 2024-07-26

Reinprospatare

Adauga Manager

Sterge Rezervare

Sterge Utilizator

Detali Utilizatori

ID: 1 | Username: mp123 | Rol: user

ID: 2 | Username: jc2109 | Rol: manager

ID: 4 | Username: jama | Rol: user

ID: 5 | Username: slupu | Rol: manager

ID: 6 | Username: m123 | Rol: user

ID: 7 | Username: tobik | Rol: user

ID: 8 | Username: folo | Rol: user

ID: 9 | Username: snai | Rol: manager

ID: 10 | Username: braimond | Rol: manager

ID: 11 | Username: ss24 | Rol: user

View Adaugare Camera:



The screenshot shows a window titled "AdaugareCamera" with a standard Windows-style title bar (minimize, maximize, close buttons). Below the title bar is a menu bar with "About" and "Exit" options. The main content area has a light gray background. On the left side, there is a "Back" button. On the right side, there is a "Log Out" button. The central part of the window contains a form with the following fields and controls:

- Numar Camera**: A text input field.
- Pret pe noapte (numar)**: A text input field.
- Etaj (numar)**: A text input field.
- Tipul Camerei**: A dropdown menu with "Standard" selected.
- Capacitate (persoane)**: A dropdown menu with "1" selected.
- Tip Balcon**: A dropdown menu with "Fara" selected.

At the bottom center of the form area is a button labeled "Adauga Camera".

View Adaugare Manager:

The screenshot shows a window titled "Adauga Manager" with a standard Windows-style title bar (minimize, maximize, close buttons). Below the title bar is a menu bar with "About" and "Exit" options. The main content area has a light gray background. On the left side, there is a "Back" button. On the right side, there is a "Log Out" button. The central part of the window contains a form with the following fields and controls:

- Username**: A text input field.
- Password**: A text input field.
- Nume**: A text input field.
- Prenume**: A text input field.

At the bottom center of the form area is a button labeled "Sign Up".

## 6.6 Test Case

Nr crt.	Ce se testează	Denumire test	Rezultat asteptat	Rezultat obținut	Valoare testata	Verdict test case
1,	Calculator varsta (DLL)	AgeCalculation	21	21	21-09-2002	Passed
2.	Validare varsta	AgeValidationCheck_Over_18	True	True	21-09-2002	Passed
3.	Validare varsta	AgeValidationCheck_Under_18	False	False	07-10-2008	Passed
4.	Valabilitate date calendaristice	DateAvailable	True	True	21-09-2027 06-09-2027 23-09-2027	Passed
5.	Validare intervale de date calendaristice	DateIntervalAvailable	True	True	07-08-2025 11-08-2025 02-08-2025 12-08-2025	Passed
6.	Validare email (model strategie)	EmailValidateCheck_CorrectFormat	True	True	raimond.butnaru21@gmail.com	Passed
7.	Validare email (model strategie)	EmailValidateCheck_CorrectFormat2	True	True	jamal_africanu@yahoo.com	Passed
8.	Validare email	EmailValidateCheck_WrongFormat	False	False	un.email.invalid	Passed
9.	Validare email	EmailValidateCheck_InvalidFormat2	False	False	un.alt.email-invalid	Passed
10.	Verificare corectitudine hash( a se vedea figura de mai jos)	HashCheck	c910384c4455ffe36e76fd32a03aa3c7eee9c23f58e02804445fec2da6c8bb48	c910384c4455ffe36e76fd32a03aa3c7eee9c23f58e02804445fec2da6c8bb48	IP2024	Passed
11.	Validare varsta	InValid_AgeValidationCheck_Unde	False	System.FormatException	2010-100	Failed

		r_18				
12.	Validare numar de telefon	PhoneNumberValidation_CorrectFormat1	True	True	0758387068	Passed
13.	Validare numar de telefon	PhoneNumberValidation_CorrectFormat2	True	True	0234266077	Passed
14.	Validare numar de telefon	PhoneNumberValidation_IncorrectFormat1	False	False	Numar de telefon	Passed
15.	Validare numar de telefon	PhoneNumberValidation_IncorrectFormat2	False	False	07 n-am cartela	Passed
16.	Validare numar de telefon	PhoneNumberValidation_IncorrectFormat3	True	False	” ”	Failed
17.	Validare username	UsernameValidation_CorrectFormat1	True	True	UnUsername Oarecare	Passed
18.	Validare username	UsernameValidation_CorrectFormat2	True	True	Programatorul Notoriu69	Passed
19.	Validare username	UsernameValidation_CorrectFormat3	True	True	User1	Passed
20.	Validare username	UsernameValidation_IncorrectFormat1	False	False	Nume de utilizator	Passed
21.	Validare username	UsernameValidation_IncorrectFormat2	False	False	Vladut Guriuc	Passed
22.	Validare username	UsernameValidation_IncorrectFormat3	False	False	“ “	Passed

Total teste: 22

Passed: 20

Failed: 2

IP2024

☐ Remember Input

☐ Enable HMAC

Hash

☒ Auto Update

Output Type

Hex

c910384c4455ffe36e76fd32a03aa3c7eee9c23f58e02804445fec2da6c8bb48

Copy