<Car Wash>

Document de analiza si design

Studenti:

* Brăguși Vlad
* Pop Sergiu
* Ținteșan Cristiana

**Grupa: 30237**

**Contents**

[I Specificatia proiectului 3](#_Toc61373404)

[1.1 Diagrama Domain Model 3](#_Toc61373405)

[II Modelul Use-Case 4](#_Toc61373406)

[2.1 Utilizatori 4](#_Toc61373407)

[2.2 Identificarea Use-Case-uri 5](#_Toc61373408)

[2.3 Diagrama UML Use-Case 6](#_Toc61373409)

[III Design arhitectural 6](#_Toc61373411)

[3.1 Arhitectura conceptuala 7](#_Toc61373412)

[3.2 Diagrama bazei de date 7](#_Toc61373413)

[IV Specificatii suplimentare 7](#_Toc61373414)

[4.1 Specificatii non-functionale 7](#_Toc61373415)

[4.2 Constrangeri de design 8](#_Toc61373416)

[V Testare 8](#_Toc61373417)

[5.1 Testarea functionalitatii aplicatiei 8](#_Toc61373418)

[5.2 Dezvolatari ulterioare 13](#_Toc61373419)

[VI Bibliografie 13](#_Toc61373420)

# I Specificatia proiectului

Proiectul presupune implementarea unei aplicatii pentru o spalatorie de masini. Aceasta va avea rolul de a stoca, prelucra si afisa informatiile speficifice unei rezervari.

Aplicatia dispune de doua tipuri de utilizatori:

* User-ul(clientul). Acesta are posibilitatea de a se loga si de a efectua o programare la o ora si zi stabilita.
* Administratorul. Acesta are posibilitatea de a vizualiza toate rezervarile efectuate de client, de a vizualiza totalul incasarilor dintr-o anumita zi si de a adauga/ scoate din functiune o boxa din salatorie.

## Diagrama Domain Model

A picture containing text, sky, map

Description automatically generated

# II Modelul Use-Case

Diagrama Use-Case prezinta o colectie de cazuri de utilizare si actori. Aceasta indica interectiunea dintre sistem si utlizatorii sai, ofera o descriere generala a modului in care poate fi utilizat proiectul, creeaza o privire de ansamblu a fuctionalitatilor asteptate de sistem.

## 2.1 Utilizatori

Utilizatorii aplicatiei se impart in doua categorii:

* User (clientul). Drepturile acestuia sunt limititate in cadrul aplicatiei. Pentru a putea efectua o rezervare, clientul trebuie sa detina un cont valid si sa fie logat. Acesta are optiunile de a vizualiza lista preturilor si de a efectua o rezervare selectand data, ora exacta si tipul de spalare al masinii pe care il doreste.
* Administratorul. Acesta are posibilitatea de a vizualizata tabelul rezervarilor, de a introduce/sterge o boxa si de a vizualiza incasarile care s-au produs in ziua selectata.

## 2.2 Identificarea Use-Case-uri

**Nume Use case:** Inregistrare utilizator nou

**Nivel:** User-Goal

**Actor principal:** Utilizatorul

**Scenariul principal de success:** Aplicatia necesita folosirea unui cont pentru utilizatori. In cazul in care clientul nu detine un cont creeat anterior, aceasta are posibilitatea de a-si crea unul prin accesarea butonului “New user”, care va deschide fereastra de creeare cont, unde va introduce datele cerute. In cazul in care, contul a fost realizat cu success, clientul se poate inregitra in aplicatie si va avea posibilitatea de a efectua rezervari.

**Extensie:** Trimirea unui mesaj de confirmare, in care sa fie specificat daca contul clientului a fost creeat cu scces.

**Nume Use case:** Efectuare rezervare

**Nivel:** User-Goal

**Actor principal:** Utilizatorul

**Scenariul principal de success:** Pentru a ajunge la fereastra de rezervari, utilizaotrul trebuie sa fie logat si sa selecteze butonul “Book Box”, care va deschide fereastra de rezervari. Aici, user-ul trebuie sa selecteza data in care doreste sa faca rezervarea, ora exacta si tipul de spalare al masinii pentru care opteaza. In cazul in care, date introduse sunt corecte si mai exista locuri disponibile, clientul va primi o confirmare a rezervarii.

**Nume Use case:** Vizualizare rezervari

**Nivel:** User-Goal

**Actor principal:** Administrator

**Scenariul principal de success:** Admin-ul trebuie sa se logheze cu contul lui pentru a putea vizualiza tabelul rezervarilor. Dupa logare, prin accesarea butonul “View Bookings”, se va deschide o noua fereastra, oferind administratorului posibilitatea de a vizualiza tabelul care contine datele despre rezervari: id-ul boxei, ora de inceput si de final, data si tipul.

## 2.3 Diagrama UML Use-Case

# Diagram Description automatically generated

# III Design arhitectural

## 3.1 Arhitectura conceptuala

Aceasta aplicatie este una desktop, continand o baza de date care stocheaza informatii despre utilizatori (clienti) si despre rezervarile pe care acestia le efectueaza in cadrul aplicatiei.

Pentru comunicarea cu baza de date este utilizat instrumental de mapare relational obiect-obiect “Hibernate.Atat getionarea propriu-zisa a datelor, cat si interfata grafica au fost realizate cu ajutorul limbajul de programare “Java”. Pentru realizarea interfetei grafice au fost utilizate functiile specifice din pachetul swing.

## 3.2 Diagrama bazei de date

Box

 User

Graphical user interface, text, application

Description automatically generated

Graphical user interface, application

Description automatically generated

Booking

# IV Specificatii suplimentare

In continuare, se vor evidentia speficicatiile suplimentare non-functionale ale acestui proiect, precum si constrangerile impuse de aplicatie.

## 4.1 Specificatii non-functionale

* **Securitatea.** Aceasta cerinta non-functionala este una dintre cele mai importante deoarece presupune protectia impotriva atacurilor nedorite. Fiecare client trebuie sa treaca de fereastra de Log In pentru a putea efectua rezervari in cadrul aplicatiei. Clientii au acces la fereastra de rezervari si de vizualizare a preturilor, dar nu au acces la operatiile specifice administratorului.
* **Disponibilitatea.** Aceasta este e alta cerinta non-functionala si este indicata de momentele in care sistemul poate fi utilizat. Aplicatia este disponibila permanent, oferind clintiilor timp de acces nelimitat.
* **Extensibilitatea.** Aceasta proprietate non-functionala a sistemului indica cat de usor se pot adauga noi functionalitati sistemului. Implementarea aplicatiei a fost realizata astfel incat, dezvoltarile viitoare sa poate fi mapate cu usurinta, fara a afecta modul de functionare al celor deja existente.
* **Utilizabilitatea.** Interfata grafica a proiectul trebuie realizata astfel incat, clientii sa o poata utiliza cu usurinta, fara a exista necesitatea consultarii unui manual de utilizare. Designul interfetei, plasarea elementelor in locatii specifice sunt cateva dintre caracteristicile necesare pentru o utilizare preferata de client.
* **Performanta.** Performanta reprezinta o masura care defineste volumul de procesari pe care o aplicatie trebuie sa le poata realiza pe o unitate de timp. Aceasta proprietate este necesara pentru a garanta functionarea aplicatiei in conditii optime. Implementarea proiectului are la baza comunirea in timp real cu clientul.

## 4.2 Constrangeri de design

Aplicatia noastra este o aplicatie desktop, ceea ce ofera utilizatorului un avataj deoarece nu este necesara conexiunea la internet pentru utilizarea ei.

Datorita naturii aplicatiei, aceasta nu este scalabila. Ea este destinata utilizarii unui numar relative mic de clienti. In cazul in care se doreste extinderea numarului de utilizatori este necesara optimizarea acceselui la baza de date.

# V Testare

Pentru testarea aplicatiei s-au efectuat mai multe verificari: s-a verificat faptul ca datele sunt introduse corespunzator in baza de date, ca nu se poate efectua o rezervare la o data precedenta zilei curente.

## 5.1 Testarea functionalitatii aplicatiei

Aplicatia incepe cu deschiderea ferestre de “Log in”. In cadrul acesteia fiecare client isi poate indroduce datele personale sau, in cazul in care, este un utilizator nou, acesta poate selecta butonul “New user”, unde va introduce datele necesare pentru a creea noul cont.

Graphical user interface, application

Description automatically generated

Graphical user interface, application

Description automatically generated

In cazul in care, utilizatorul incerca sa intrduca un Username sau o Parola invalida, va primi un mesaj de avertizare, iar autentificarea nu va fi posibila.

Graphical user interface, application

Description automatically generated

O data autentificat, clientul poate sa vizualizele lista preturilor pentru servicile de care dispune spalatoria sau sa efectueze o rezervare.

Graphical user interface, website

Description automatically generated

Daca clientul opteaza pentru “Book box”, va accesa fereastra detinata rezervarii. Daca datele introduse in scopul efectuarii unei rezervari, atunci clientul va primi o chitanta cand acesta apasa buton “Confirm”

Graphical user interface, application

Description automatically generatedGraphical user interface, text, application

Description automatically generated

Administratorul aplicatiei dispune de patru optiuni: “View Booings” , “View Income”, “Add Box” si “ Remove Box”.

Graphical user interface

Description automatically generated

Optiunea “View Bookings” permite administratorului sa vizualizeze tabelul in care au fost inregistrate toate rezervarile clientilor.

Table

Description automatically generated

Optiunile Add/Remove Box ofera administratorului posibilitatea de a adauga sau de a inchide o boxa din cadrul spalatoriei.

Butonul “View Income” va deschide o fereatra in care administratorul are posibilitatea de introduce ziua pentru care doreste ca afle totalul produs de spalatorie.

Graphical user interface

Description automatically generatedGraphical user interface

Description automatically generated

Angajatul spalatoriei are optiunea de a face o rezervare pe baza locurilor ce se ocupa in timp real, pentru o gestiune mai buna a timpului de asteptare.Graphical user interface, application

Description automatically generatedGraphical user interface, website

Description automatically generated

## 5.2 Dezvolatari ulterioare

Aplicatia prezinta multiple posibilitati de dezvoltare ulterioare:

* Relizarea unei pagini web;
* Posibilitatea de a vizualiza boxele disponibile in timp real cu scopul de a permite clientilor sa vina la spalatorie si fara rezervare;
* Stergerea rezervarii, in cazul in care clientul se razgandeste

# VI Bibliografie

<https://www.javatpoint.com/java-swing>

<https://jdatepicker.org/>

<https://www.javatpoint.com/java-joptionpane>

<https://www.codejava.net/java-se/swing/how-to-use-jdatepicker-to-display-calendar-component>