



Proiect

Utilizarea Bazelor de Date



Lanț Hotelier

Cătălan Cătălin-Florin

An academic 2020/2021

Proiect de Semestru

Specializarea: Tehnologia Informației

Disciplina: Utilizarea Bazelor de Date

Coordonator: Prof. univ. dr. ing. Györödi Cornelia Aurora

Cuprins

1. Introducere

- Introducere, argumente, scop și obiective specifice.

2. Analiza cerințelor utilizatorilor (specificațiile de proiect)

- Ipoteze specifice domeniului ales pentru proiect (cerințe, constrângeri);
- Organizare structurată a cerințelor utilizator;
- Determinarea și caracterizarea de profiluri de utilizatori (admin, user).

3. Modelul de date și descrierea acestuia

- Entități și atributele lor (descriere detaliată – implementarea fizică);
- Diagrama EER/UML pentru modelul de date complet;
- Normalizarea datelor.

4. Detalii de implementare

- Descrierea funcțională a modulelor (organizarea logică a acestora);
- Manual de utilizare/instalare.

5. Concluzii limitări și dezvoltări ulterioare

1. Introducere

Acest document are în vedere prezentarea în ansamblu a proiectului *Lanț Hotelier*, realizat în cadrul cursului de *Utilizarea Bazelor de Date*, semestrul I, an universitar 2020-2021. În acest document sunt descriși pașii urmați în realizarea proiectului, o descriere a versiunii finale a proiectului și o următoare etapă la care poate fi dus acest proiect.

Cerință:

- Dezvoltați o aplicație web pentru gestiunea unui lanț hotelier. Va exista o evidență a hotelurilor, camerelor, prețurilor, a clienților hotelului, a rezervărilor etc. Aplicația va cuprinde și funcționalități mai complexe pentru gestiunea informațiilor din baza de date, pe care le veți concepe și discuta la ora de proiect.

2. Analiza cerințelor utilizatorilor (specificațiile de proiect)

- *Ipoteze specifice domeniului ales pentru proiect (cerințe, constrângeri)*

O mare importanță o are aplicația web, iar aceasta trebuie construită astfel ca utilizatorul să o poată utiliza cu mare ușurință. Design-ul site-ului este foarte important fiindcă ajută utilizatorul să navigheze cât mai ușor, obținând astfel informațiile necesare.

- *Organizare structurată a cerințelor utilizator*

Informațiile din baza de date se păstrează în mai multe tabele, iar în cazul aplicației *Lanț Hotelier* avem nevoie de tabele pentru camerele disponibile, camerele închiriate, rezervări și administratori ai bazei de date.

3. Modelul de date și descrierea acestuia

Pentru ca informațiile puse la dispoziție de aplicația *Lanț Hotelier* să fie structurate cât mai clar, în baza de date exista 4 tabele:

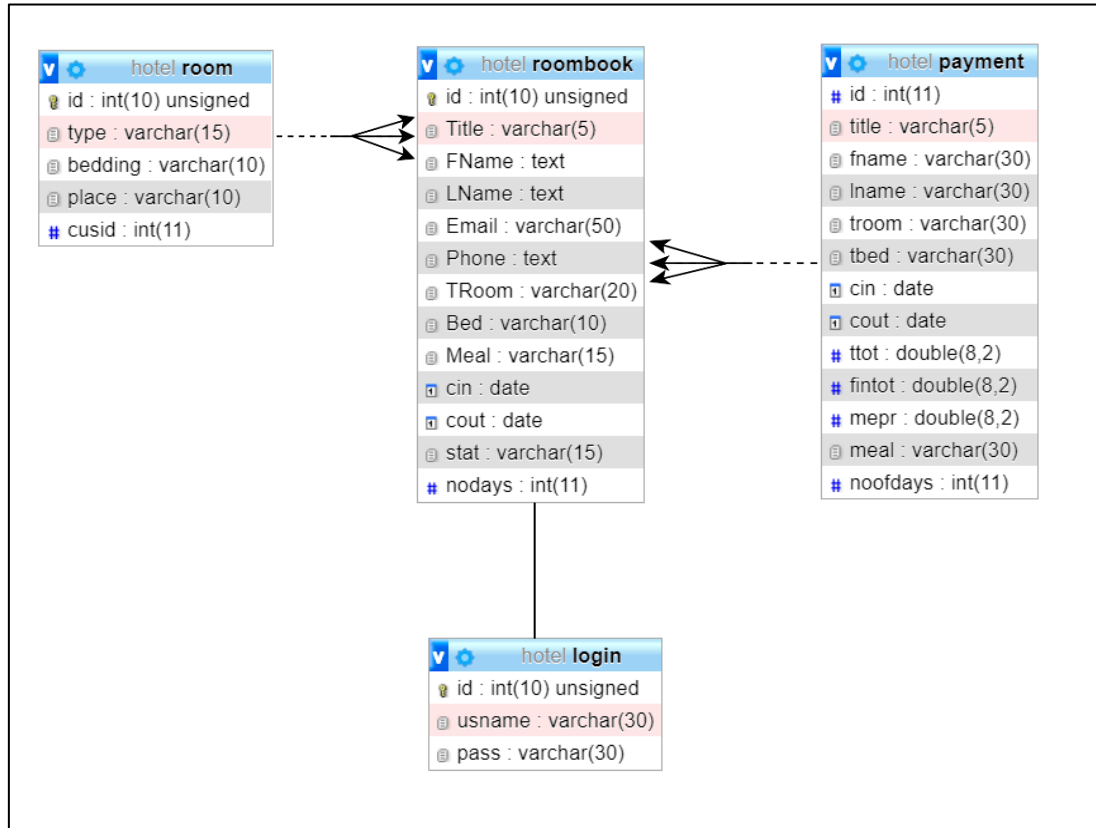


Tabela principală este **room** care are cheia primară *id* și care conține tipul camerei (*type*), tipul de pat (*bedding*) și un câmp care specifică dacă camera este dată sau nu în rezervare (*place*). Tot aici se găsește foreign-key-ul care face legătură cu id-ul clientului care face rezervarea. Tabela **roombook** stochează datele clientului (forma de adresare (*Title*), nume (*FName*), prenume (*LName*), email (*Email*), număr de telefon (*Phone*), tipul camerei (*TRoom*), tipul patului (*Bed*), tipul de meniu (*Meal*), datele de preluare și restituire a camerei (*cin*, *cout*), statusul camerei (*stat*), numărul de zile (*nodays*).

Tabela **payment** îi stochează pe acei clienți cărora administratorul le-a confirmat cererea de rezervare. Tabela conține în principiu datele clienților, respectiv prețul mesei (*meapr*), prețul camerei x numărul de zile de cazare (*ttot*), suma totală de plată (*fintot*). Fiecare bază de date trebuie să aibă și un administrator, prin urmare este nevoie să-i reținem într-o tabelă pe aceștia. Tabela **login** conține id-ul administratorului (*id*), numele de utilizator (*uname*) și parola (*pass*).

Normalizarea datelor

Pentru ca datele să poată fi relaționate și utilizate corect este necesară normalizarea bazei de date. Acest lucru presupune încadrarea în formele de normalizare de mai jos:

Prima formă normală:

- ➔ exclude posibilitatea existenței grupurilor repetitive cerând ca fiecare câmp într-o bază de date să cuprindă numai o valoare atomică;
- ➔ de asemenea, prima formă normală cere și ca fiecare înregistrare să fie definită astfel încât să fie identificată în mod unic prin intermediul unei chei primare.

A doua formă normală:

- ➔ cere ca toate elementele unei tabele să fie dependente funcțional de totalitatea cheii primare;
- ➔ dacă unul sau mai multe elemente sunt dependente funcțional numai de o parte a cheii primare, atunci ele trebuie să fie separate în tabele diferite.

A treia formă normală:

- ➔ Toate atributele non-cheie ale unei relații depind numai de chei candidate ale acelei relații;
- ➔ Toate atributele non-cheie sunt (trebuie să fie) mutual independente.^[1]

Proceduri utilizate asupra bazei de date:

- ➔ Următorul cod PHP realizează inserția datelor aferente introduse de client în tabela *roombook*. În momentul în care clientul apasă butonul de *submit* cererea lui este înregistrată în baza de date:

```
<?php
if (isset($_POST['submit'])) {
    $key_code = $_POST['key_code'];
    $generated_code = $_POST['generated_code'];
    if ($key_code != $generated_code)
        echo "<script type='text/javascript'> alert('Invalid code! Please, try again!')</script>";
    else{
        $connection = mysqli_connect("localhost", "catalin@yahoo.com", "W5q9yPF0C0PPu8E1", "hotel");
        $new = "Not Confirm Yet";
        $newUser = "INSERT INTO 'roombook' ('Title', 'FName', 'LName', 'Email', 'Phone', 'TRoom', 'Bed',
        'Meal', 'cin', 'cout', 'stat', 'noday') VALUES ('$_POST[title]', $_POST[first_name], $_POST[last_name]',
        '$_POST[email]', $_POST[phone_number], $_POST[type_room], $_POST[type_bed], $_POST[meal_plan]',
        '$_POST[check_in]', $_POST[check_out], $new, datediff('$_POST[check_out]', '$_POST[check_in]'))";
        if (mysqli_query($connection, $newUser))
            echo "<script type='text/javascript'> alert('Thank you! Your booking application has been sent!')</script>";
        else
            echo "<script type='text/javascript'> alert('I m sorry! Error adding user in database! Please try again!')</script>";
        }
    }
}
```

- ➔ Se creează o nouă sesiune de lucru (*session_start()*). În cazul în care la momentul accesării paginii nu suntem logați în cont, vom fi redirecționați către *Login_Admin.php*:

```
<?php
    session_start();
    if(!isset($_SESSION["user"]))
        header("location:Login_Admin.php");
?>
```

- ➔ Includem conexiunea către baza de date, selectăm din tabela *roombook* toți clienții și contorizăm în variabila *\$count* pe toți cei cărora încă nu li s-a confirmat rezervarea:

```
<?php
    include('Database_connection.php');
    $sql = "select * from roombook";
    $result = mysqli_query($con, $sql);
    $count = 0;
    while ($row = mysqli_fetch_array($result)) {
        $new = $row['stat'];
        $check_in = $row['cin'];
        $id = $row['id'];
        if ($new == "Not Confirm Yet")
            $count = $count + 1;
    }
?>
```

- ➔ Afișăm toți clienții cărora încă nu li s-a confirmat rezervarea folosind funcția *echo*:

```
<?php
    $sql = "select * from roombook";
    $tre = mysqli_query($con, $sql);
    while ($table_row = mysqli_fetch_array($tre)) {
        $status = $table_row['stat'];
        if ($status == "Not Confirm Yet") {
            echo "<tr>
                <th> . $table_row['id'] . "</th>
                <th> . $table_row['FName'] . " " . $table_row['LName'] . "</th>
                <th> . $table_row['Email'] . "</th>
                <th> . $table_row['TRoom'] . "</th>
                <th> . $table_row['Bed'] . "</th>
                <th> . $table_row['Meal'] . "</th>
                <th> . $table_row['cin'] . "</th>
                <th> . $table_row['cout'] . "</th>
                <th> . $table_row['stat'] . "</th>
                <th><a href='Room_Booking.php?id=' . $table_row['id'] . " ' class='btn btn-primary'>Action</a></th>
            </tr>";
        }
    }
?>
```

- ➔ Selectăm din tabela *roombook* toți clienții și contorizăm în variabila *\$contor* pe toți cei cărora li s-a confirmat rezervarea:

```
<?php
$result_sql = "SELECT * FROM `roombook`";
$result = mysqli_query($con, $result_sql);
$contor = 0;
while ($row = mysqli_fetch_array($result)) {
    $statut = $row['stat'];
    if ($statut == "Confirm")
        $contor = $contor + 1;
}
?>
```

- ➔ Afișăm toți clienții cărora li s-a confirmat rezervarea folosind funcția *echo*:

```
<?php
$sql = "SELECT * FROM `roombook`";
$mre = mysqli_query($con, $sql);
while ($mrow = mysqli_fetch_array($mre)) {
    $statut = $mrow['stat'];
    if ($statut == "Confirm") {
        $fid = $mrow['id'];
        echo "<div class='col-md-3 col-sm-12 col-xs-12'>
            <div class='panel panel-primary text-center no-boder bg-color-blue'>
                <div class='panel-body'><i class='fa fa-users fa-5x'></i>
                <h3>". $mrow['FName'] . "</h3>
                <h3>". $mrow['LName'] . "</h3>
                </div>
                <div class='panel-footer back-footer-blue'> . "ID Client: " . $mrow['id'] . "<br> . "Type room: " . $mrow['TRoom'] .
                "<br> . "Type bed: " . $mrow['Bed'] . "</div>
            </div>";
    }
}
?>
```

- ➔ Includem conexiunea către baza de date, afișăm toți clienții din tabela *payment* și toate datele despre aceștia:

```
<?php
include ('Database_connection.php');
$sql="select * from payment";
$mre = mysqli_query($con,$sql);
while($row = mysqli_fetch_array($mre)) {
    $id = $row['id'];
    if($id % 2 ==1) {
        echo"<tr class='grader'>
            <td>". $row['title'] . " " . $row['fname'] . " " . $row['lname'] . "</td>
            <td>". $row['troom'] . "</td>
            <td>". $row['tbed'] . "</td>
            <td>". $row['cin'] . "</td>
            <td>". $row['cout'] . "</td>
            <td>". $row['meal'] . "</td>
            <td>". $row['ttot'] . "</td>
            <td>". $row['mepn'] . "</td>
            <td>". $row['fintot'] . "</td>
            <td><a href=Print_Invoice.php?pid=".$id ." <button class='btn btn-primary'> <i class='fa fa-print' ></i>Print</button></td>
            </tr>";
    }
    else {
        echo"<tr class='graderU'>
            <td>". $row['title'] . " " . $row['fname'] . " " . $row['lname'] . "</td>
            <td>". $row['troom'] . "</td>
            <td>". $row['tbed'] . "</td>
            <td>". $row['cin'] . "</td>
            <td>". $row['cout'] . "</td>
            <td>". $row['meal'] . "</td>
            <td>". $row['ttot'] . "</td>
            <td>". $row['mepn'] . "</td>
            <td>". $row['fintot'] . "</td>
            <td><a href=Print_Invoice.php?pid=".$id ." <button class='btn btn-primary'> <i class='fa fa-print' ></i>Print</button></td>
            </tr>";
    }
}
?>
```

- ➔ Afișăm id-ul, username-ul și password-ul fiecărui administrator existent în tabela *login* folosind funcția *echo*:

```
<?php
while ($row = mysqli_fetch_array($re)) {
    $id = $row['id'];
    $us = $row['username'];
    $ps = $row['pass'];
    if ($id % 2 == 0) {
        echo "<tr class='gradeC'>
            <td>" . $id . "</td>
            <td>" . $us . "</td>
            <td>" . $ps . "</td>
            <td><button class='btn btn-primary btn' data-toggle='modal' data-target='#myModal'>Update</button></td>
            <td><a href=Code_For_Delete_Admin.php?eid=" . $id . "<button class='btn btn-danger'><i class='fa fa-edit'></i>Delete</button></td>
        </tr>";
    } else {
        echo "<tr class='gradeU'>
            <td>" . $id . "</td>
            <td>" . $us . "</td>
            <td>" . $ps . "</td>
            <td><button class='btn btn-primary btn' data-toggle='modal' data-target='#myModal'>Update</button></td>
            <td><a href=Code_For_Delete_Admin.php?eid=" . $id . "<button class='btn btn-danger'><i class='fa fa-edit'></i>Delete</button></td>
        </tr>";
    }
}
?>
```

- ➔ Următorul cod PHP adaugă un nou administrator în baza de date:

```
<?php
if (isset($_POST['in'])) {
    $newus = $_POST['newus'];
    $newps = $_POST['newps'];
    $newsq = "Insert into login (username, pass) values ('$newus','$newps')";
    if (mysqli_query($con, $newsq))
        echo ' <script language="javascript" type="text/javascript"> alert("User name and password Added!")</script>';
    header("Location:Settings_Admin.php");
}
?>
```

- ➔ Următorul cod PHP actualizează username-ul și password-ul unui administrator existent în tabela *login*:

```
<?php
if (isset($_POST['up'])) {
    $usname = $_POST['usname'];
    $passwr = $_POST['pasd'];
    $upsq = "UPDATE `login` SET `usname`='$usname`,`pass`='$passwr' WHERE id = '$id'";
    if (mysqli_query($con, $upsq))
        echo ' <script language="javascript" type="text/javascript"> alert("User name and password update") </script>';
    header("Location: Settings_Admin.php");
}
ob_end_flush();
?>
```

- ➔ Următorul cod PHP șterge username-ul și password-ul unui administrator existent în tabela *login* pe baza unui id:

```
<?php
include ('Database_connection.php');
$id = $_GET['eid'];
$newsq = "DELETE FROM `login` WHERE id = '$id' ";
if(mysqli_query($con,$newsq))
    echo ' <script language="javascript" type="text/javascript"> alert("User name and password deleted!") </script>';
header("Location: Settings_Admin.php");
?>
```


➔ Afișăm toate camerele disponibile pentru rezervare folosind funcția *echo* din tabela *room*:

```
<?php
include ('Database_connection.php');
$sql = "select * from room";
$res = mysqli_query($con,$sql)
?>

<div class="row">

    <?php
        while($row=mysqli_fetch_array($res))
        {
            $id = $row['type'];
            if($id == "Single Room") {
                echo<div class='col-md-3 col-sm-12 col-xs-12'>
                    <div class='panel panel-primary text-center no-boder bg-color-blue'>
                        <div class='panel-body'><img src='Images/single_room.jpg' alt='Single Room'><h3> . "ID Room: " . $row['id'] . " " . "<br> .
                        "Type room: " . $row['type'] . "</h3></div>
                        <div class='panel-footer back-footer-blue'> . "Type bed: " . $row['bedding'] . "</div>
                    </div>
                </div>; }
            else if ($id == "Deluxe Room"){
                echo<div class='col-md-3 col-sm-12 col-xs-12'>
                    <div class='panel panel-primary text-center no-boder bg-color-green'>
                        <div class='panel-body'><img src='Images/deluxe_room.jpg' alt='Deluxe Room'><h3> . "ID Room: " . $row['id'] . " " . "<br> .
                        "Type room: " . $row['type'] . "</h3></div>
                        <div class='panel-footer back-footer-green'> . "Type bed: " . $row['bedding'] . "</div>
                    </div>
                </div>; }
            else if($id == "Luxury Room") {
                echo<div class='col-md-3 col-sm-12 col-xs-12'>
                    <div class='panel panel-primary text-center no-boder bg-color-brown'>
                        <div class='panel-body'><img src='Images/luxury_room.jpg' alt='Luxury Room'><h3> . "ID Room: " . $row['id'] . " " . "<br> .
                        "Type room: " . $row['type'] . "</h3></div>
                        <div class='panel-footer back-footer-brown'> . "Type bed: " . $row['bedding'] . "</div>
                    </div>
                </div>; }
        }
    ?>
</div>
```

➔ Folosind următorul cod inserăm o nouă cameră în tabela *room*:

```
<?php
include('Database_connection.php');
if(isset($_POST['add'])) {
    $room = $_POST['troom'];
    $bed = $_POST['bed'];
    $place = 'Free';
    $check="SELECT * FROM room WHERE type = '$room' AND bedding = '$bed'";
    $sql="INSERT INTO `room` ( `type`, `bedding`,`place`) VALUES ('$room','$bed','$place')";
    if(mysqli_query($con,$sql))
        echo '<script>alert("New Room Added!") </script>' ;
    else
        echo '<script>alert("Sorry! Check The System!") </script>' ;
}
?>
```

- ➔ Cu ajutorul următorului cod PHP putem șterge o cameră existentă în tabela room pe baza id-ului:

```
<?php
    include('Database_connection.php');
    if(isset($_POST['del'])) {
        $did = $_POST['id'];
        $sql = "DELETE FROM `room` WHERE id = '$did' ";
        if(mysqli_query($con,$sql)) {
            echo '<script type="text/javascript">alert("Delete the Room!") </script>' ;
            header("Location:Delete_Rooms.php");
        }
        else
            echo '<script>alert("Sorry ! Check The System!") </script>' ;
    }
?>
```

- ➔ Primul cod permite logout-ul administratorului, în timp ce cel de-al doilea realizează login-ul. Funcția verifică dacă username-ul și parola introdusă coincide cu vreuna din datele existente din tabela login:

```
<?php
    session_start();
    unset($_SESSION["user"]);
    header("location:Login_Admin.php");
?>
```

```
<?php
    include('Database_connection.php');
    if ($_SERVER["REQUEST_METHOD"] == "POST") {
        $myusername = mysqli_real_escape_string($con, $_POST['user']);
        $mypassword = mysqli_real_escape_string($con, $_POST['pass']);
        $sql = "SELECT id FROM login WHERE usname = '$myusername' and pass = '$mypassword'";
        $result = mysqli_query($con, $sql);
        $row = mysqli_fetch_array($result, MYSQLI_ASSOC);
        $count = mysqli_num_rows($result);
        if ($count == 1) {
            $_SESSION['user'] = $myusername;
            header("location:Status_Room_Booking.php");
        }
        else
            echo '<script>alert("Your Login Name or Password is invalid! Please, try again!") </script>';
    }
?>
```

➔ Folosind următorul cod am realizat inserția a 15 camere în baza de date:

```
INSERT INTO `room` (`id`, `type`, `bedding`, `place`, `cusid`) VALUES  
(1, 'Luxury Room', 'Single', 'Free', NULL),  
(2, 'Luxury Room', 'Single', 'Free', NULL),  
(3, 'Luxury Room', 'Single', 'Free', NULL),  
(4, 'Luxury Room', 'Double', 'Free', NULL),  
(5, 'Luxury Room', 'Double', 'Free', NULL),  
(6, 'Deluxe Room', 'Double', 'Free', NULL),  
(7, 'Deluxe Room', 'Double', 'Free', NULL),  
(8, 'Deluxe Room', 'Double', 'Free', NULL),  
(9, 'Deluxe Room', 'Single', 'Free', NULL),  
(10, 'Deluxe Room', 'Single', 'Free', NULL),  
(11, 'Single Room', 'Single', 'Free', NULL),  
(12, 'Single Room', 'Single', 'Free', NULL),  
(13, 'Single Room', 'Single', 'Free', NULL),  
(14, 'Single Room', 'Double', 'Free', NULL),  
(15, 'Single Room', 'Double', 'Free', NULL);
```

4. Detalii de implementare

Pentru realizarea site-ului am folosit mai multe limbaje și medii de programare.

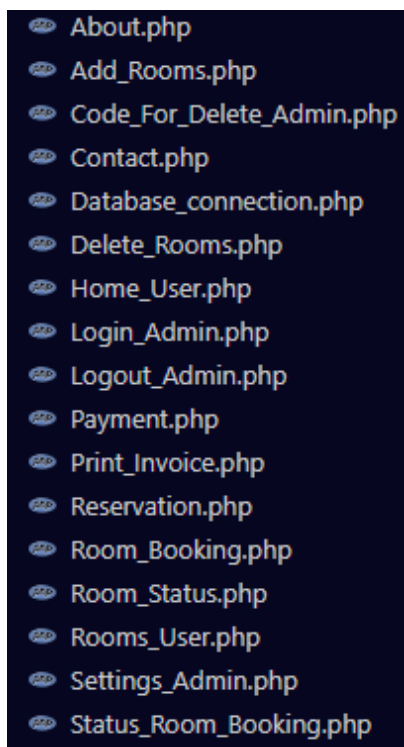
Site-ul a fost realizat folosind: HyperText Markup Language, Cascading Style Sheets, JavaScript, Hypertext Preprocessor și Structured Query Language.

Pentru crearea tabelelor și popularea lor am scris cod SQL în Data Modeler și am folosit phpMyAdmin ca server XAMPP. Structura a fost realizată folosind HTML, iar customizarea site-ului a fost realizată folosind CSS și JS. Mai jos sunt prezentate câteva detalii despre fiecare în parte:

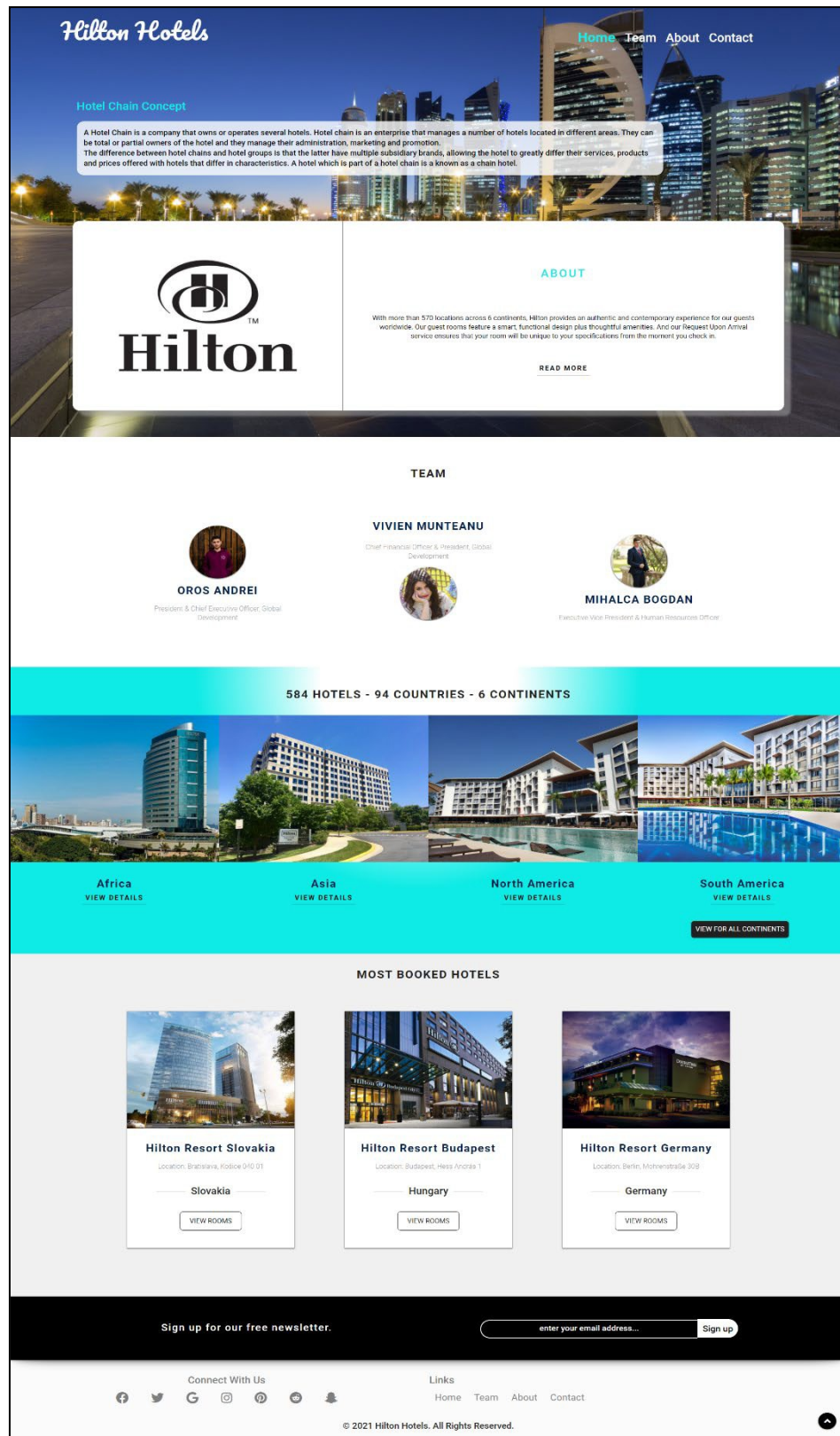
- Oracle SQL Developer **Data Modeler** este un instrument grafic gratuit care îmbunătățește productivitatea și simplifică sarcinile de modelare a datelor. Utilizând Oracle SQL Developer Data Modeler utilizatorii pot crea, răsfoi și edita modele logice, relaționale, fizice, multidimensionale și de tip date. Data Modeler oferă capabilități de inginerie directă și inversă și sprijină dezvoltarea colaborativă prin control integrat al codului sursă.^[2]
- **XAMPP** este un pachet de programe free software, open source și crossplatform web server, care constă în Apache HTTP Server, MySQL database și interpretoare pentru scripturile scrise în limbajele de programare PHP și Perl.^[3]

- **SQL** (Structured Query Language - Limbaj Structurat de Interogare) este un limbaj de programare specific pentru manipularea datelor în sistemele de manipulare a bazelor de date relaționale (RDBMS), iar la origine este un limbaj bazat pe algebra relațională. Acesta are ca scop inserarea datelor, interogări, actualizare și ștergere, modificarea și crearea schemelor, precum și controlul accesului la date. A devenit un standard în domeniu (standardizat ANSI-ISO), fiind cel mai popular limbaj utilizat pentru crearea, modificarea, regăsirea și manipularea datelor de către SGBD-urile (Sistemele de Gestiune a Bazelor de Date) relaționale.^[4]
- **HTML** (HyperText Markup Language) este un limbaj de marcare utilizat pentru crearea paginilor web ce pot fi afișate într-un browser (sau navigator). Scopul HTML este mai degrabă prezentarea informațiilor – paragrafe, fonturi, tabele ș.a.m.d. – decât descrierea semanticii documentului.^[5]
- **CSS** (Cascading Style Sheets) este un limbaj (style language) care definește "layout-ul" pentru documentele HTML. CSS acoperă culori, font-uri, margini (borders), linii, înălțime, lățime, imagini de fundal, poziții avansate și multe alte opțiuni. CSS folosește stiluri, acestea înglobează, sub un anumit nume, attribute de formatare care se aplică asupra unui element individual din pagină, asupra unui grup de elemente sau la nivelul întregului document.^[6]
- **JS** (JavaScript) este un limbaj de programare orientat obiect bazat pe conceptul prototipurilor. Este folosit mai ales pentru introducerea unor funcționalități în paginile web, codul JavaScript din aceste pagini fiind rulat de către browser. Limbajul este binecunoscut pentru folosirea sa în construirea siturilor web, dar este folosit și pentru acesul la obiecte încastate (embedded objects) în alte aplicații.^[7]
- **PHP** (Hypertext Preprocessor) este un limbaj de programare folosit inițial pentru a produce pagini web dinamice. Astăzi este folosit pe scară largă în dezvoltarea paginilor și aplicațiilor web. Se folosește în principal înglobat în codul HTML, dar începând de la versiunea 4.3.0 se poate folosi și în mod „linie de comanda” permițând crearea de aplicații independente.^[8]

Paginile .php ale aplicației *Lanț Hotelier* sunt următoarele:



Pagina principală este *Home_User.php*:



Cătălan Cătălin-Florin 1631A – *Lanț Hotelier*

În momentul în care utilizatorul dă click pe opțiunea *VIEW ROOMS* va fi redirecționat către pagina *Rooms_User.php*:


Hilton Hotels

Home Team About Contact


THE MOST LUXURIOUS HOTELS ONLY HERE ON HILTON HOTELS.COM

Our Hotels offer includes only five-star hotels from different locations: either on sunny beaches of the high mountains. You can chose from our offer. We handle all the paperwork for your reservation in order for your room to be ready for your arrival. When it's time for a getaway, Hilton offers unique travel packages and hotel discounts. Search our latest hotel special offers by location, date of travel, package type or by what interests you.

Note: the prices displayed include three-days reservations!



HILTON HOTELS OFFERS




Single room

description room

\$30

BOOK NOW




Single room

description room

\$25

BOOK NOW




Single room

description room

\$40

BOOK NOW




Single room

description room

\$50

BOOK NOW




Deluxe room

description room

\$70

BOOK NOW




Deluxe room

description room

\$75

BOOK NOW




Deluxe room

description room

\$67

BOOK NOW




Deluxe room

description room

\$80

BOOK NOW




Luxury room

description room

\$75

BOOK NOW




Luxury room

description room

\$68

BOOK NOW




Luxury room

description room

\$100

BOOK NOW



Luxury room

description room

\$90

BOOK NOW

Sign up for our free newsletter.

enter your email address... Sign up

Connect With Us

Links

Home Services About Contact

© 2021 Hilton Hotels. All Rights Reserved.

Cătălan Cătălin-Florin 1631A – Lanț Hotelier

Utilizatorul alege tipul camerei din opțiunile prezentate dând click pe *BOOK NOW*.
În continuare se va face salt la pagina de rezervări *Reservation.php*:

Home

Rooms

Team

About

Contact

Reservation Hilton Hotels

PERSONAL INFORMATION

Title

MR. ▾

First Name

Last Name

Email

Phone Number

RESERVATION INFORMATION

Type Of Room

SINGLE ROOM ▾

Bedding Type

SINGLE ▾

Meal Plan

ROOM ONLY ▾

Check-In

ZZ -- -- 0000

Check-Out

ZZ -- -- 0000

HUMAN VERIFICATION

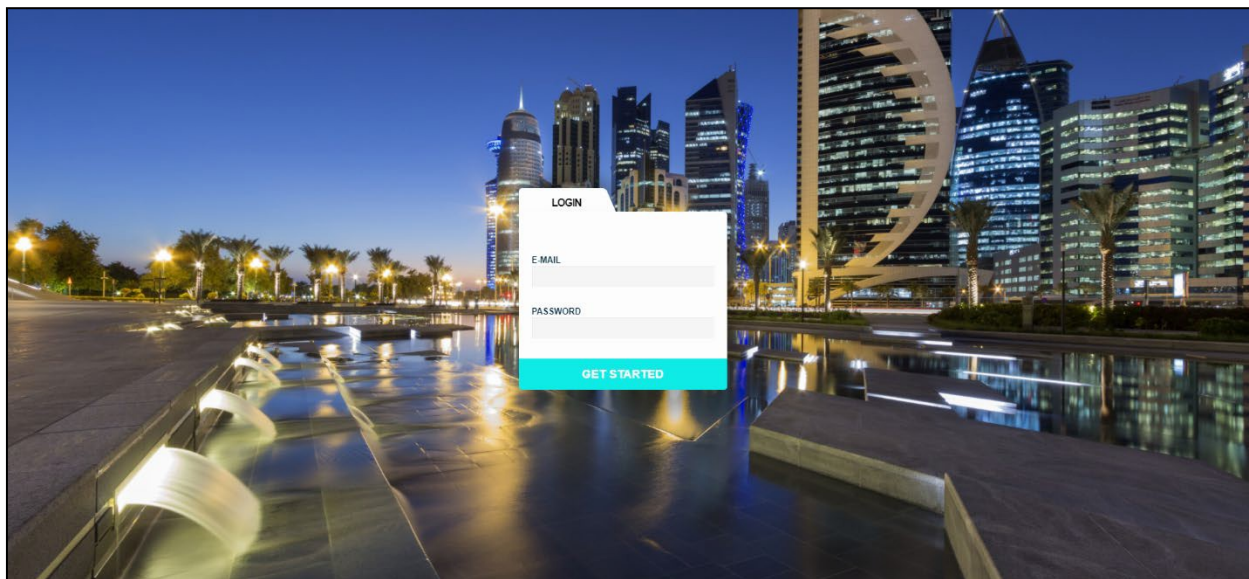
Type Below this code 1635975213

Enter the random code

Trimite

În această pagină clientul completează datele necesare pentru formular, introduce codul de verificare în câmpul specificat și apasă butonul *Trimite*. În acest moment cererea lui este înregistrată în baza de date.

În pagina *Login_Admin.php* administratorul introduce datele de conectare. Dacă datele de conectare introduse sunt corecte va fi redirecționat către pagina *Status_Room_Booking.php*. În această pagină administratorul poate vedea cererile de cazare și clienții existenți care au deja o cameră confirmată:



CATALIN_CATALAN@YAHOO.COM

Status Rooms

Payment

User Profile

Settings

Logout

Status Room Booking

New Room Bookings 0

Booked Rooms 0

Catalan
Catalin

ID Client: 2
Type room: Single Room
Type bed: Single

Popescu
Ioana

ID Client: 3
Type room: Luxury Room
Type bed: Double

Ionescu
Alexandru

ID Client: 4
Type room: Deluxe Room
Type bed: Double

Pentru aconfirma o cerere de cazare administratorul dă click pe butonul *Action*, iar de aici va fi redirectionat către pagina *Room_Booking.php*. Se selectează opțiunea *CONFIRM* pentru confirmare.

CATALIN_CATALAN@YAHOO.COM

Status Rooms

Payment

User Profile

Settings

Logout

Room Booking2020/12/17

Booking Confirmation

DESCRIPTION	INFORMATION
Name	Mr. Moisi Emanuel
Email	m_emanuel@gmail.com
Phone	0748509316
Type Of the Room	Luxury Room
Meal Plan	Half Board
Bedding	Double
Check-in Date	2020-12-17
Check-out Date	2020-12-24
Numbers of days	7
Status Level	Not Confirm Yet

Select the Confirmation

CONFIRM

Confirm Reservation

Available Rooms Details

Single Room	1
Deluxe Room	3
Luxury Room	4

În pagina *Payment.php* administratorul poate vizualiza sumele de plată ale clienților, iar prin apăsarea butonului Print poate printa o factură fiscală cu numele clientului, data de primire și restituire precum și eventualele sume de achitat:

CATALIN_CATALAN@YAHOO.COM

Status Rooms

Payment

User Profile

Settings

Logout

Payment Details Room Booking

Name	Room type	Bedding Type	Check in	Check out	Meal Type	Room Rent	Price Meal	Total Price	Print
Mr. Catalan Catalin	Single Room	Single	2020-12-16	2020-12-23	Room only	210.00	0.00	210.00	Print
Miss. Popescu Ioana	Luxury Room	Double	2020-12-16	2020-12-23	Full Board	700.00	280.00	980.00	Print
Mrs. Ionescu Alexandru	Deluxe Room	Double	2020-12-17	2020-12-24	Full Board	560.00	280.00	840.00	Print

INVOICE

Hilton Hotels
Street: E Lamar, Arlington, USA
Phone: +1 817-640-3322, +1 952-991-4411

Hilton
HONORS

Mr. Catalan Catalin

Invoice # 2

Date check-in 2020-12-16

Date check-out 2020-12-23

Item	No of Days	Price	Total Price
Type room: Single Room	7	\$30	\$210
Type bed: Single Bed	7	\$0	\$0
Type meal: Room only	7	\$0	\$0

Total Payment \$210

CONTACT US

Email: hilton_hotels@yahoo.com, h_hotels@gmail.com

Web: www.HiltonHotels.com

În pagina *Settings_Admin.php* se poate vizualiza lista de administratori ai bazei de date, de asemenea se poate crea, șterge și actualiza această listă:

CATALIN_CATALAN@YAHOO.COM

Status Rooms
Payment
User Profile
Settings
Logout

ADMINISTRATOR accounts

User ID	User name	Password	Update	Remove
1	catalin_catalan@yahoo.com	1234	Update	Delete
3	Catalin98	1234	Update	Delete


[Add New Admin](#)

În pagina *Room_Status.php* putem vizualiza lista de camere disponibile pentru închirieri:


CATALIN_CATALAN@YAHOO.COM

Available Rooms
Add Rooms
Delete Rooms
Status Rooms
Payment
User Profile
Logout


Available Rooms




ID Room: 1
Type room: Luxury Room
Type bed: Single




ID Room: 2
Type room: Luxury Room
Type bed: Single




ID Room: 3
Type room: Luxury Room
Type bed: Single




ID Room: 6
Type room: Deluxe Room
Type bed: Double




ID Room: 7
Type room: Deluxe Room
Type bed: Double




ID Room: 8
Type room: Deluxe Room
Type bed: Double




ID Room: 9
Type room: Deluxe Room
Type bed: Single




ID Room: 11
Type room: Single Room
Type bed: Single



ID Room: 12
Type room: Single Room
Type bed: Single



ID Room: 17
Type room: Luxury Room
Type bed: Single



ID Room: 18
Type room: Luxury Room
Type bed: Single

De asemenea se poate adăuga (*Add_Rooms.php*) și șterge camere în și din baza de date (*Delete_Rooms.php*):

CATALIN_CATALAN@YAHOO.COM

Available Rooms

ADD NEW ROOMS

ADD NEW ROOMS

Type Of Room
SINGLE ROOM

Bedding Type
SINGLE

Add New

CATALIN_CATALAN@YAHOO.COM

Available Rooms












Add Rooms

DELETE ROOM

Delete room

Select the Room ID
1

Delete Room

 ID Room: 1 Type room: Luxury Room Type bed: Single	 ID Room: 2 Type room: Luxury Room Type bed: Single	 ID Room: 3 Type room: Luxury Room Type bed: Single	 ID Room: 6 Type room: Deluxe Room Type bed: Double
 ID Room: 7 Type room: Deluxe Room Type bed: Double	 ID Room: 8 Type room: Deluxe Room Type bed: Double	 ID Room: 9 Type room: Deluxe Room Type bed: Single	 ID Room: 11 Type room: Single Room Type bed: Single
 ID Room: 12 Type room: Single Room Type bed: Single	 ID Room: 17 Type room: Luxury Room Type bed: Single	 ID Room: 18 Type room: Luxury Room Type bed: Single	

5. Concluzii, limitări și dezvoltări ulterioare

Dezvoltarea ulterioară poate apărea atât în partea bazei de date prin introducerea unor noi tabele sau câmpuri, cât și în partea de interfață.

În partea de interfață se mai poate realiza un Log pentru client cu un username și parolă.

De asemenea se pot realiza mai multe proceduri pentru introducerea, ștergerea sau modificarea datelor din baza de date, de exemplu: un client poate să anuleze rezervarea, să primească notificare conform căreia cererea lui a fost acceptată sau respinsă, ștergerea unui user din tabelă etc. Toate acestea fiind posibile dezvoltări ulterioare.

Bibliografie

Surse de informare pentru descrierea teoretică a lucrării:

- http://www.elth.pub.ro/~preda/teaching/BDE/BDE_5.pdf
- https://ro.wikipedia.org/wiki/HyperText_Markup_Language
- https://ro.wikipedia.org/wiki/Cascading_Style_Sheets
- <https://ro.wikipedia.org/wiki/JavaScript>
- <https://www.oracle.com/ro/database/technologies/appdev/datamodeler.html>
- <https://ro.wikipedia.org/wiki/SQL>
- <https://ro.wikipedia.org/wiki/XAMPP>
- <https://ro.wikipedia.org/wiki/PHP>

Pozele din site au fost preluate de pe:

- <https://pixabay.com/ro/>
- <https://unsplash.com>

Surse pentru descărcarea mediilor de programare necesare pentru realizarea și rularea proiectului:

- <https://code.visualstudio.com>
- <https://www.oracle.com/ro/database/technologies/appdev/datamodeler.html>
- <https://www.apachefriends.org/ro/index.html>
- <https://www.jetbrains.com/phpstorm/>

(Observație: data la care au fost accesate link-urile este 17.12.2020)

Pentru rularea proiectului consultați fișierul Read_Me.txt și urmați pașii de acolo.