THE APPLICATION OF SOFTWARE TESTING TECHNOLOGY ON SECURITY IN WEB APPLICATION SYSTEM

(Hotel Information Management System using Database Encryption)

FINAL REVIEW REPORT

Submitted by

REEMA AGARWAL (18BCE2386)

ATHARVA MANGESHKUMAR AGRAWAL (18BCE2029)

Prepared for

Software Engineering (CSE-3001) – PROJECT COMPONENT

Submitted to:

PROF. MANJULA R

Associate professor Sr.

Department of Software Systems

School of Computer Science and Engineering

VIT®

Vellore Institute of Technology

(Deemed to be University under section 3 of UGC Act, 1956)

ABSTRACT

The software industry has achieved a solid recognition in this age. In the recent decade, however, the cyber-world seems to be even more dominating and driving force which is shaping up the new forms of almost every business. These days, websites are not meant only for publicity or marketing but these have been evolved into the stronger tools to cater to complete business needs. The websites are sold, the data is leaked and users or customers privacy is hampered. Therefore, there comes the need of **software/security testing** which is a type of software testing that intends to uncover vulnerabilities of the system and determine that its data and resources are protected from possible intruders. Web application demands, even more, security with respect to its access, along with data protection. A web developer should make the application immune to SQL Injections, Brute Force Attacks and XSS (cross-site scripting).

This project examines the aspect of the hospitality industry which is Hotel management and will make sure to provide security on this web application. This project is the design and implementation of an electronic hotel management system that provides proper management of data and transactions in a centralized and organized manner and also provides a user-friendly interface with which the user can interact easily with the just little or elementary knowledge of operating computers. Firstly, our project aims on providing security to the data being stored in the database i.e. information regarding the customers and other room reservation information by implementing application level encryption**.** Our project will make use of different software testing technologies and tools thereby to check the vulnerability and security. It will also look for penetration testing through the use of different software testing tools such as acunetix web vulnerability scanner.

This project is designed to create a platform that allows both the user and administrator to keep track of transactions like room reservations, room booking, financial administration of the hotel, online reservation and other day to day activities involved in the running and management of a hotel. The implementation is based on the requirements for a hotel management system. The project work is divided into five major categories which are; Front Desk, Accommodation, Catering, Finance & Account and Personnel Staff Record (Human resource management).

This project accomplished the task of building a system that ensures accurate record maintenance which was done through proper identification of customers and the proper designation of user functions with most of the processes being done automatically. An electronic hotel management information system is required to assist management of data in the hospitality industry and also to make the entire hotel management process easier.

KEYWORDS

- Hotel information Management System
- Application level Encryption
- Software/Security testing tools
- Vulnerability Scanning
- Security Scanning
- Penetration testing
- Brute Force attack
- SQL injection and XSS
- Owasp (Open web application security project)
- Acunetix Web Vulnerability Scanner

INTRODUCTION

Web-based application are not only being used by organizations but are also being sold as products today. This means that online applications have gained the trust of customers and users regarding their vital feature named as SECURITY. No doubt, the security factor is of primary value for desktop applications too. However, when we talk about the web, the importance of security increases exponentially. If an online system cannot protect the transaction data, no one will ever think of using it. Therefore, in our hotel information management system we implemented security testing in order to identify the threats in the system, to measure the potential vulnerabilities of the system and also it helps in detecting every possible security risks in the system

There are various types of security testing techniques such as:

- Vulnerability Scanning
- Security Scanning
- Penetration testing
- Risk assessment
- Security Auditing
- Ethical Hacking
- Posture Assessment
- Cross Site Scripting
- SQL injection
- Brute force attack

The above-mentioned security testing can be done using various security testing tool. Such as:

Owasp (Open web application Security project)- the tool to pen test various software environments and protocols.

Zed attack Proxy (ZAP- an integrated penetration testing tool)

The CROSS (Codenomicon Robust Open Source Software) program is designed to help open source projects, that are part of the infrastructure of the internet, fix critical flaws in their code.

Skipfish- Skipfish is an active web application vulnerability security scanning tool. Security professionals use this tool to scan their own sites for vulnerabilities.

WebScarab- A framework with multiple plugs in, written entirely in Java, for analyzing the applications that communicate through HTTP/HTTPS protocols.

Wireshark- It is used by network professionals around the globe for troubleshooting, analysis, software and protocol development.

Therefore, our project Hotel information management system which enables the management of customers' data, customers' registration, Customer accommodation or allocation into specific rooms, room reservation and Personnel staff management mainly focuses on the methods to provide maximum security on the web thereby making use of application based data encryption and decryption and also using various security tools such as owasp to show out the possibility of vulnerable scanning. Since making use of application level encryption and decryption in the database will prevent hacking of the data and misuse of one's private information, making use of security testing tools will therefore make it easier to check if still any vulnerability or risk of loss of information will occur.

BACKGROUND STUDY (LITERATURE REVIEW)

To use information system effectively, establishment of database systems and efficient management of information derived from databases have become increasingly important. There have been multiple studies and research regarding security on web-based application. On such research was made by Ivan Andrianto et at (2017) where they use fuzz testing method to test security on web application system. It is a software testing technique done by giving a set of invalid inputs to the application under test. In fuzz testing for web application, a set of HTTP requests will be sent to the application under test in order to see how the application behaves when getting various inputs. Similarly, Dimitris and their teams (2019) used a gray-box combinatorial testing method using domain specific granular attack grammars with the aim of detecting SQL injection vulnerabilities in web application. They also identified the flaws that were caused due to insufficient sanitization of user input. They used WAVSERP verification framework to establish the effectiveness of the proposed took so that they can analyze if there is any vulnerabilities being caused or not. Ibéria Medeiros et al used approach to detect web application vulnerabilities inspired in NLP in which static analysis tools learn to detect vulnerabilities automatically using machine learning. The approach uses a sequence model (HMM) that, first, learns to characterize vulnerabilities from a corpus composed of sequences of observations annotated as vulnerable or not. The approach was implemented in the DEKANT tool and evaluated experimentally with a set of open source PHP applications and WordPress plugins, finding 16 zero-day vulnerabilities.

Studies made previously regarding the problems of information security in the hotel industry are relatively rare. Some address issues such as security and privacy of hotel customers (Goh and Law,2007). But as stated by Hong-bumm kim (2012) frequent accidental incidents include unexpected hotel information system inaccessibility, destroyed data and physical damage to existing information. Hui Zhai, Hui Shi, Rui Zhai used the ASP.NET technology through software testing techniques to the check the flaws which were later found and then the improvements algorithm was given. ASP (Active Server Pages) is a server-side technology to create dynamic web pages. The users working interface was achieved through the IE browser under B/S mode (Browser/Server mode).

LIMITING OF THE EXISTING SYSTEMS

The phase of system analysis process deals with problems that are affecting the current manual system. The problems are those, which are affecting the hotel in its daily routine work. As the growing trend in most business in InfoTech World of Computers, need of accuracy, perfectness, speed and high memory data storage is a must. Each and every problem must be solved with least amount of time and energy.

- The problems faced by the existing system and hope to be solved by the Hotel Management System are described below:
- Difficulty in maintenance of records
- Time consuming
- Editing of data becomes a tedious job
- No security of data
- Mistakes occurring in Calculation of funds
- Lack of efficiency
- Data redundancy
- Data inconsistency
- Incidence of Fraud

The main purpose of the proposed system is to provide solutions to the problems and help the user to manage the hotel effectively and efficiently through:

Adequate Record Keeping: To eliminate manual record keeping and install an electronic record keeping thereby ensuring adequate record of transactions are kept. This ensures a centralized system where all necessary data and information can easily be accessed, Tracked, and monitored

Reduced Incidence of Fraud:

The program is envisaged to reduce the incidence of fraud both by staff and outsiders through proper record keeping, tracking and monitoring of transaction operations in the organization.

Provide Data Security: The study will install security measures by providing different access levels to various staff ensuring that the data stored is encrypted and cannot be therefore viewed by anyone except the admin.

Effective Resource Management: The Human Resource module (HR) and Finance & Account (F&A) module will enable effective utilization of financial and human resources by comparing the accounts receivable with the account payable and complete record of personnel through the nominal roll module will enhance staff deployment and productivity.

Reduced Time Consumption: A good search algorithm will be implemented on the web application to enhance the search facility whereby users of the system can search for all kinds of data using various criteria.

The system can be handy to the user in the following ways:

- To automatize the work such as gathering information, gathering Hotel Staff information, Workers' roster, food ordering and Hotel administration in general.
- Removal of Data Redundancy.
- To create a centralized system where all necessary data and information can be accessed easily.
- Data Consistency.

Design Approach / Materials & Methods

An online hotel information management system will be designed and implemented using MySQL as the database, Apache will be web server to provide basic functionality of the web services. PHP will be used as scripting language to program the server side that manipulates the knowledge in the database.

Different languages are used to design the portal:

- **HTML**

Hypertext markup
language Also
called web pages
A markup language is a set of markup tags

The Web pages created with HTML alone are static, meaning the user can't interact with the Web page. All users see the same Web page. Dynamic Web pages, on the other hand, allow the user to interact with the Web page. Different users might see different Web pages. For instance, one user looking at a furniture store's online product catalog might choose to view information about the sofas, whereas another user might choose to view information about coffee tables.

- **CSS**

Used to create the layout and look of pages

- **JavaScript**

One language widely used to make Web pages dynamic is JavaScript. JavaScript is useful for several purposes, such as mouse-overs (for example, to highlight a navigation button when the user moves the mouse pointer over it) or accepting and validating information that users type into a Web form.

- **PHP**

PHP is a language that is particularly well suited to interact with databases. PHP can accept and validate the information that users type into a Web form and can also move the information into a database. PHP, a scripting language designed specifically for use on the Web, is a dynamic tool for creating dynamic Web pages. PHP is rich in features that make Web design and programming easier. Its popularity continues to grow, meaning that it fulfils its function pretty well.

- **SQL**

Database is stored using SQL.

Design

The factors considered in designing the online hotel information management system are Interoperability and accessibility with minimum requirements on the user's side. Due to large flow of information delivery over the Internet, the system is implemented as a standard Internet application. The client side requires no more than standard Internet browser installed on the local computer, while the main application functionality is assured by the server side.

This includes, user interface made up of access services points at the remote site, a high speed, highly reliable and scalable regional network and content management gateway with database server. This architecture allows users to access the system via the Internet using hypertext transfer protocol and the user request is transformed into a structured query language using a PHP common content management gateway, which in turn passes it to the appropriate backend system. The common content management gateway provides a single point entry to the system.

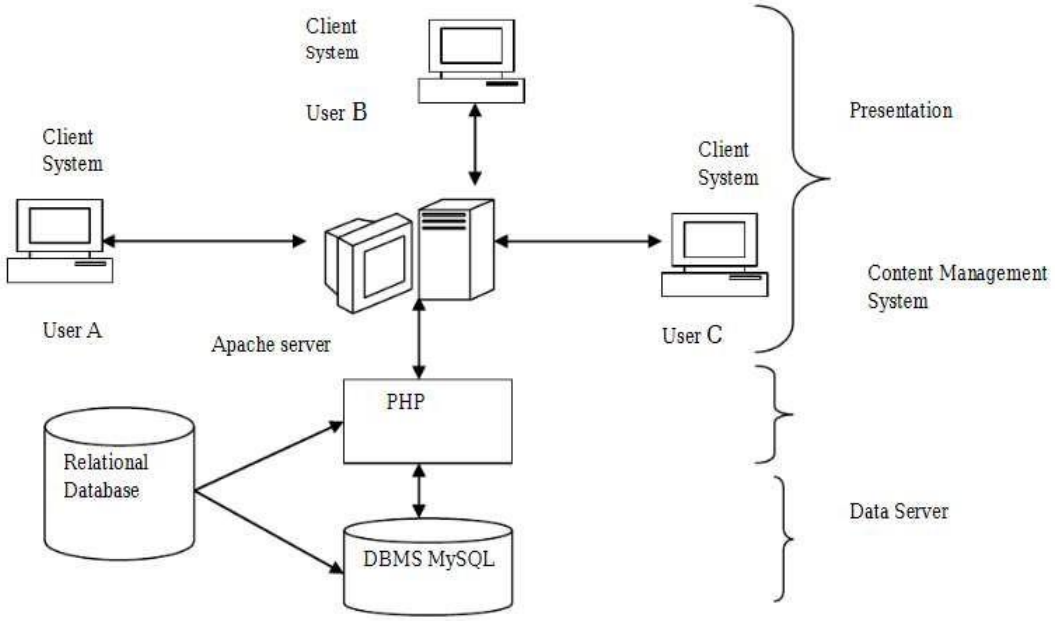


Figure 1: System Design

PROPOSED SYSTEM

We will be using Application level Encryption using Symmetric key to encrypt the database. In application-level encryption, the process of encrypting data is completed by the application that has been used to generate or modify the data that is to be encrypted. Essentially this means that data is encrypted before it is written to the database. This unique approach to encryption allows for the encryption process to be tailored to each user based on the information (such as entitlements or roles) that the application knows about its users. We have designed our own algorithm for the same. This will help us keep their customers information, user login protected from any threats from individuals with potentially malicious intentions, or any unforeseen hazards to the security of the data.

BLOCK DIAGRAM

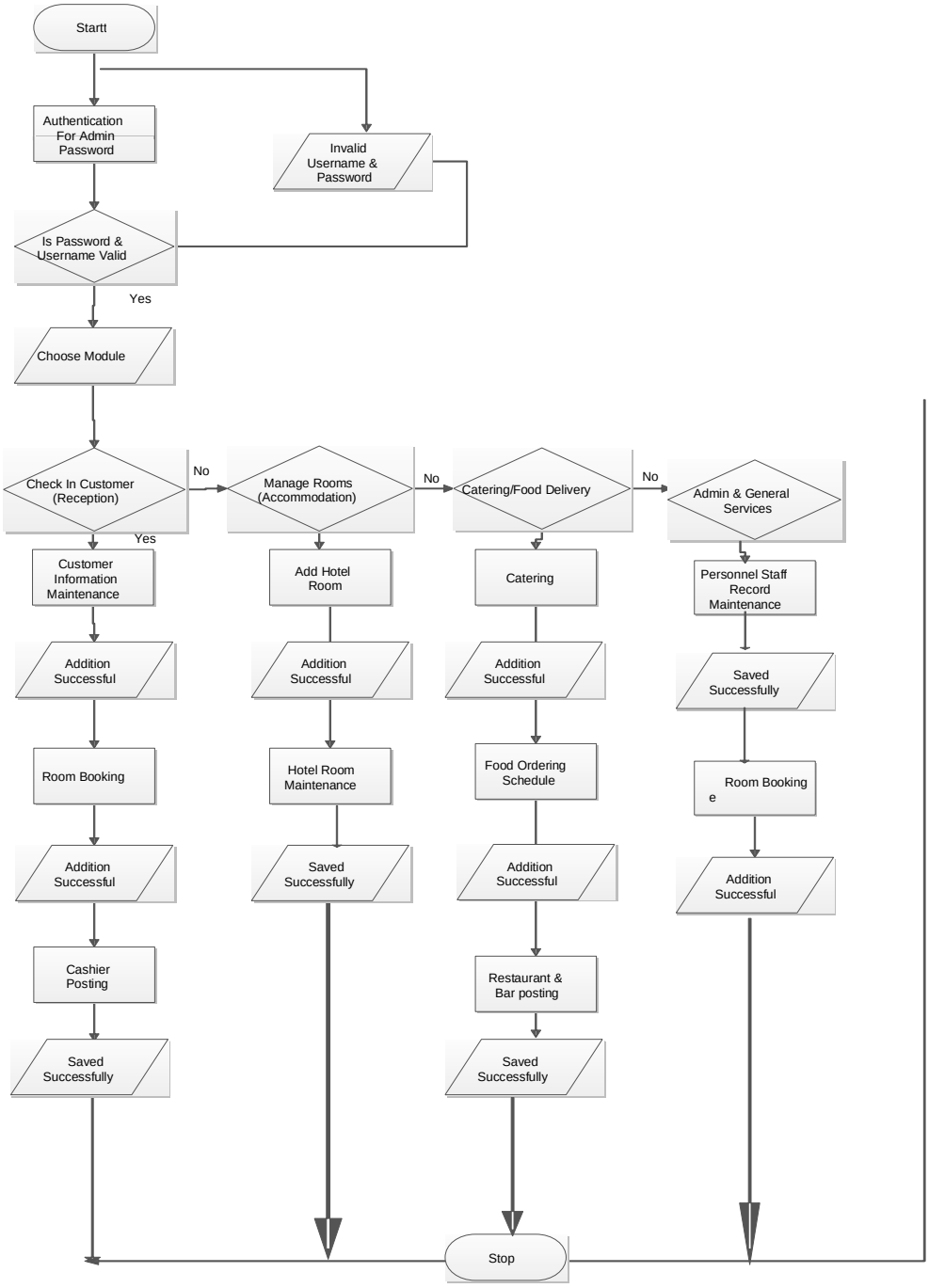


Figure 2: block diagram

The Hotel Information Management system designed by us is an online web portal. This system mainly consists of two user levels. These user levels which their entire functionality will enable us to know the exact system operates.

I) Manager

Hotel owner has the privilege of Monitoring and authorization of all the tasks handle by the system. He can access every function performed by the system. Meanwhile he will be able to take all the kinds of reports available in the system. As the manager of the system and the company he has the power to set room rates as well. Hotel manager has the sole right of deleting a staff member from the system database. He is also responsible for managing resources available in hotel management system. The user level, Manager has the authority to take all the reports available in the system and maintain the reports related to financial stuff, hotel income. Manager has other abilities such as, adding new staff member to the system, modifying them or removing them, adding new guests to the system, modifying them and removing them from the system, adding new inventory to the system, modifying them and removing them. Adding new room types to the system, modifying them and removing them. After the room is booked by the customer its manager's responsibility to confirm the bookings.

II) Customers

The customers can view all the details regarding the hotels such as the facilities being provided swimming, gym and all. They can also book the room of their choice online and even make payment on spot through net banking or credit card options. In this online portal itself they can give the details regarding their choice of food and everything. The main motive over this product is customers can be confident on providing their details on online booking as we will ensure the security to the customers regarding all their personal details which they provide while booking. The customers have to give their detailed information such as: their address, date of birth, aadhar card number, and check in, check out dates from the hotel and also their credit card or debit card details provided during the time of payment.

As soon as the customer opens the website, he/she can view the entire details of the hotel. If he then uses to book a room then can go under room reservation to book the room of his choice. During this method, he will have to fill many personal information's such as email id, phone number and nationality as well. He will be then directed to payment link to complete the payment procedure by giving the details of his debit card/ credit card. During this entire procedure he might fill ensure as he has to provide a lot of his private information which he might be willing to because of the insecurity of losing his data. Therefore, to remove this problem of data insecurity and privacy leakage, we came up with an idea of application level encryption where by your data is encrypted when it gets stored in the database such that no middle level hackers can come and hack in our data. And when the data is to be read by the admin for the detailed information it is decrypted and shown to the admin.

WORK BREAKDOWN STRUCTURE

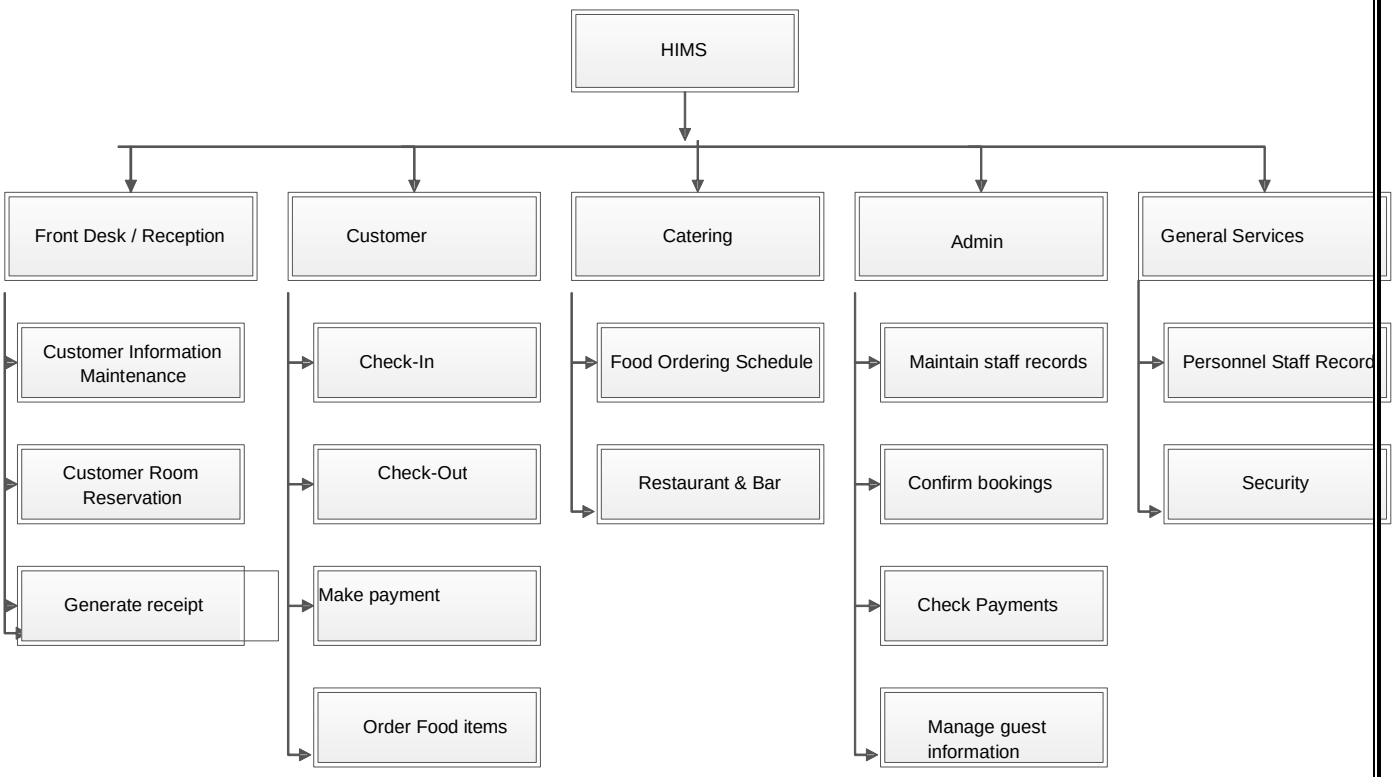


Figure 3: Work breakdown Structure

SYSTEM DESIGNS

USE CASE DIAGRAM

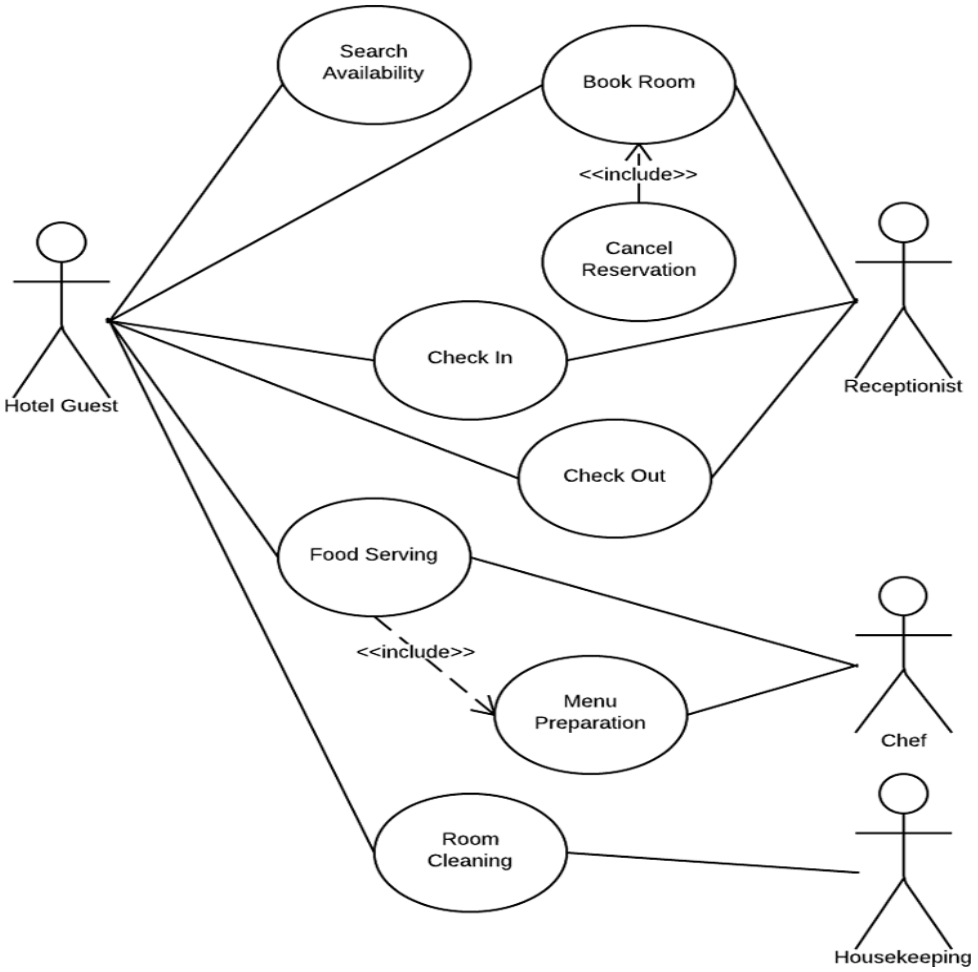


Figure 4: Use case diagram for entire system

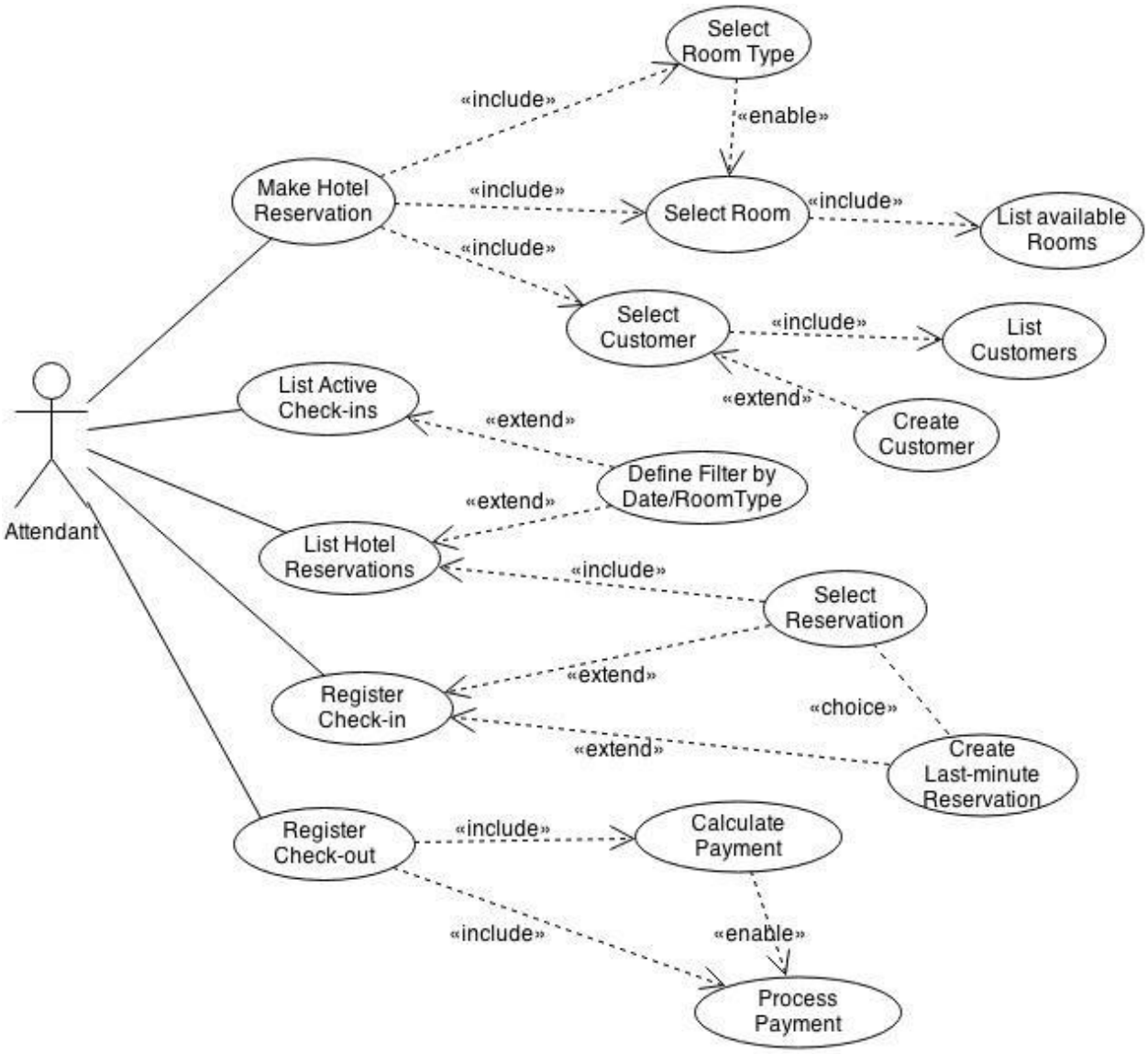


Figure 5: Use case diagram for the attendant/Customer/Hotel guest

CLASS DIAGRAM

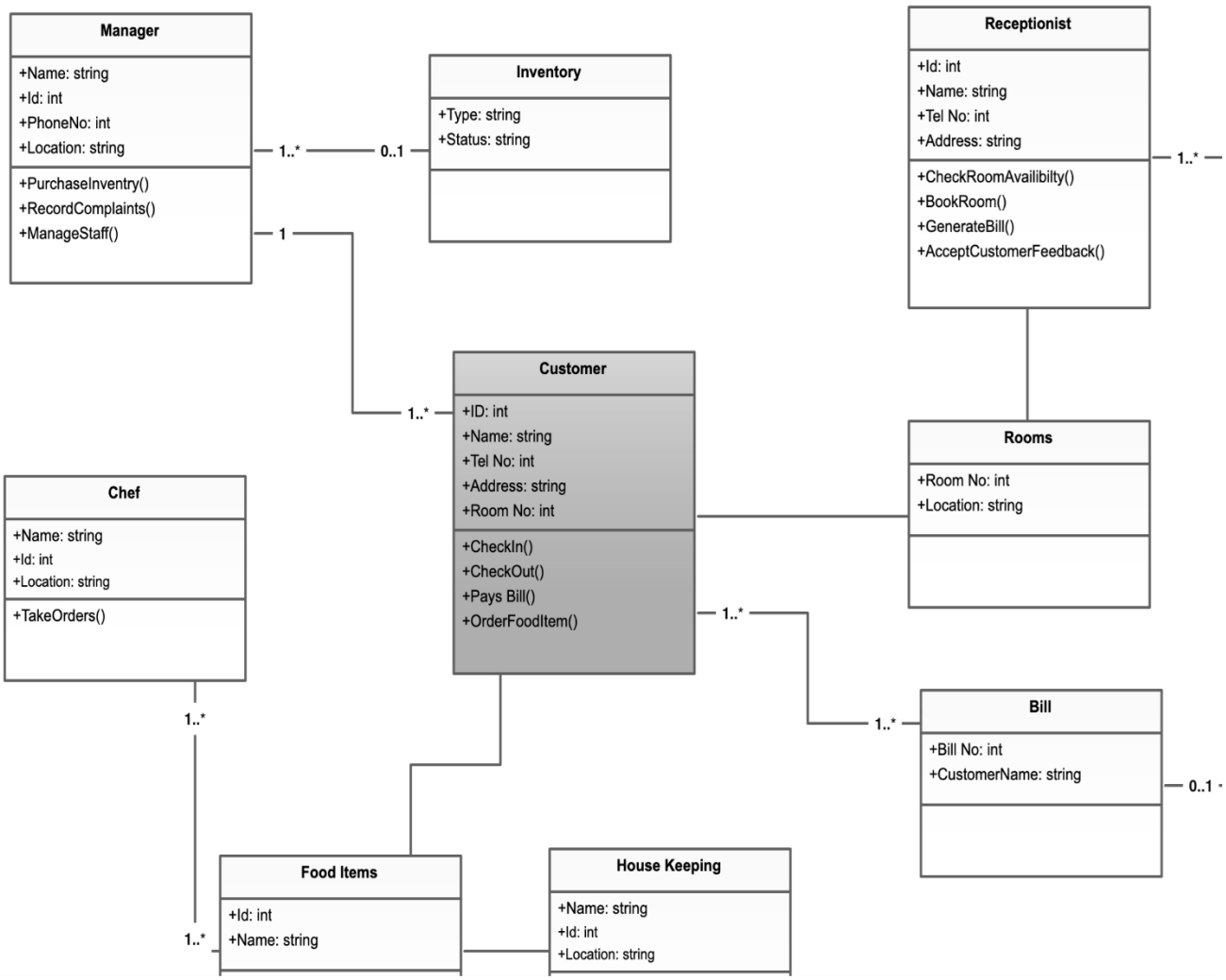


Figure 6: Class diagram

SEQUENCE DIAGRAM

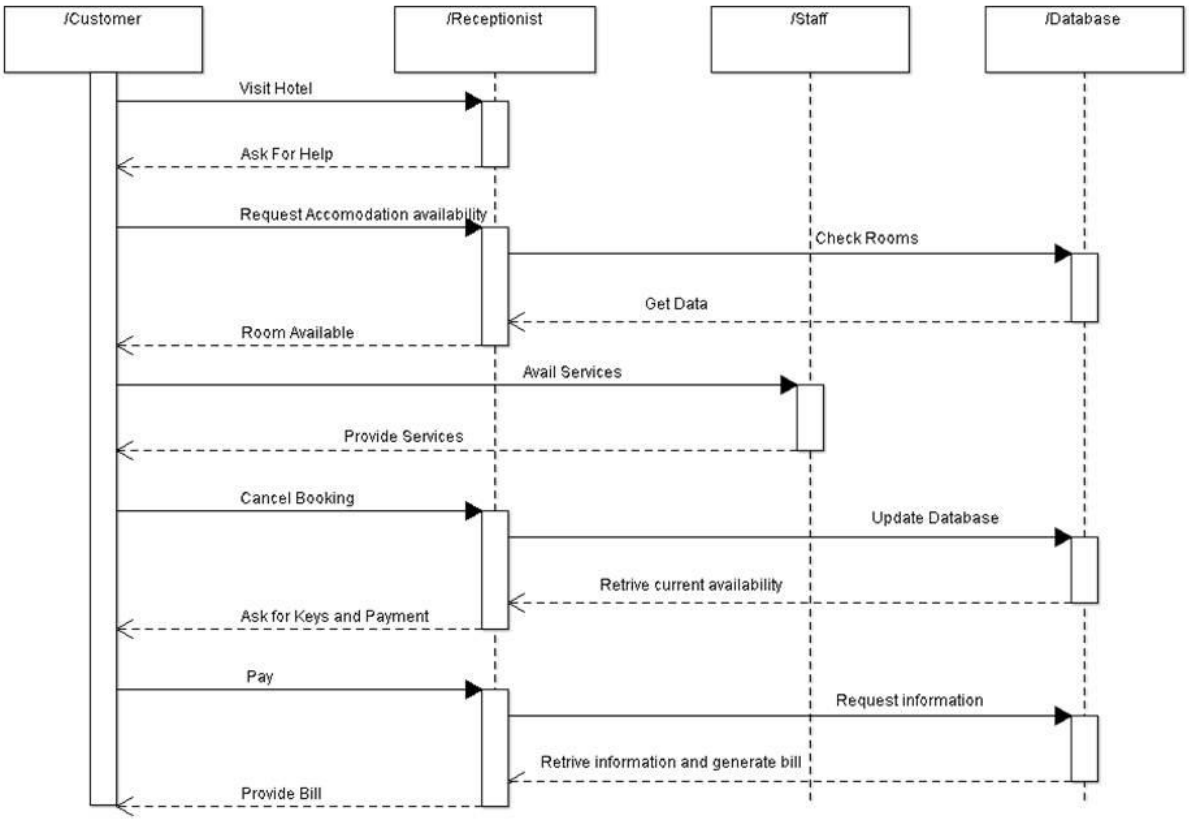


Figure 7: Sequence diagram

STATE CHART

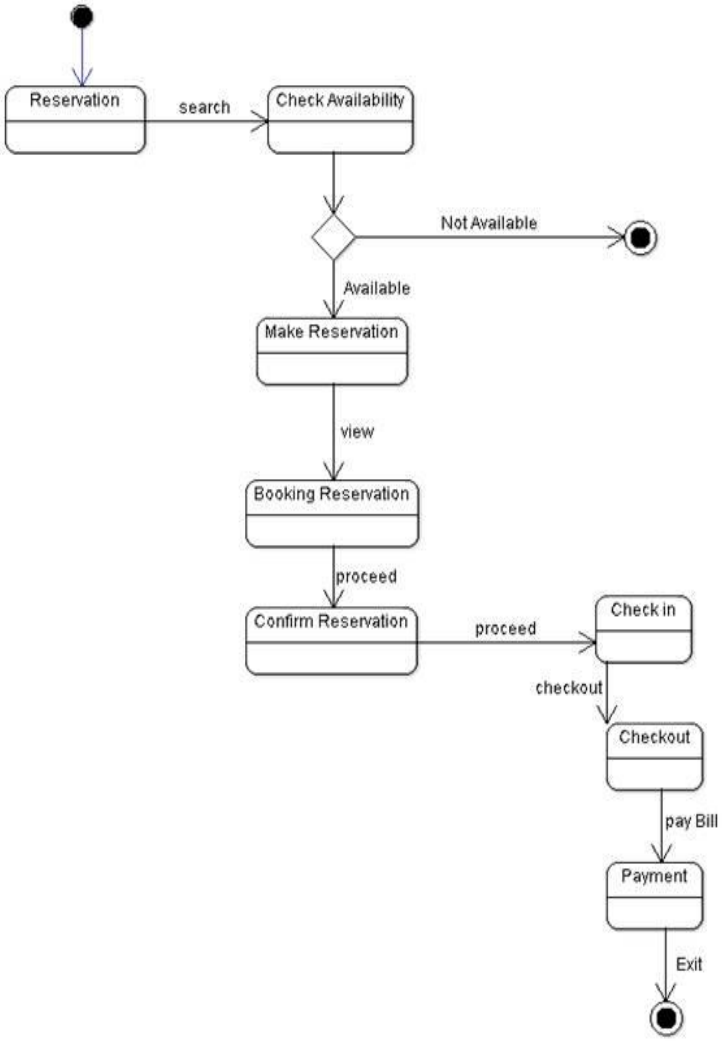


Figure 8: State chart

ACTIVITY DIAGRAM

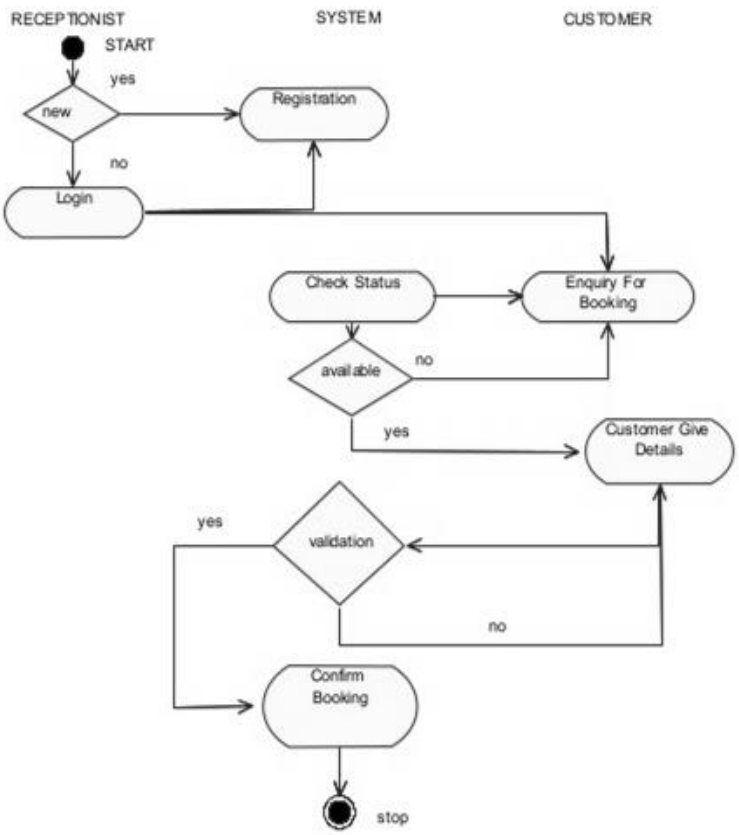


Figure 8: Activity diagram

CODE SNIPPET

This code is from the reservation.php file which basically includes the encryption of the data where Firstname (FName), Email, Nationality, Country, Phone are encrypted by making using application level encryption method and stored in the database. This is done basically to avoid any middle-level hacker to access the data and hack.

```
<div class="col-md-12 col-sm-12">
 <div class="well">
 <h4>HUMAN VERIFICATION</h4>
 <p>Type Below this code <?php $Random_code=rand(); echo$Random_code; ?> </p><br/>
<p>Enter the random code<br /></p>
<input  type="text" name="code1" title="random code" />
<input type="hidden" name="code" value="<?php echo $Random_code; ?>" />
<input type="submit" name="submit" class="btn btn-primary">
<?php
        if(isset($_POST['submit']))
        {
        $code1=$_POST['code1'];
        $code=$_POST['code'];
        if($code1!="$code")
        {
        $msg="Invalide code";
        }
        else
        {
                $con=mysqli_connect("localhost","root","","hotel");
                $check="SELECT * FROM roombook WHERE email = '$_POST[email]'";
                $rs = mysqli_query($con,$check);
                $data = mysqli_fetch_array($rs, MYSQLI_NUM);
                        if($_SERVER["REQUEST_METHOD"] == "POST") {
                         $Fname = mysqli_real_escape_string($con,$_POST['fname']);
                         $Lname= mysqli_real_escape_string($con,$_POST['lname']);
                         $Email=mysqli_real_escape_string($con,$_POST['email']);
                         $Nation=mysqli_real_escape_string($con,$_POST['nation']);
                         $Country=mysqli_real_escape_string($con,$_POST['country']);
                         $Phone=mysqli_real_escape_string($con,$_POST['phone']);
                         $Troom=mysqli_real_escape_string($con,$_POST['troom']);
                         $Btype=mysqli_real_escape_string($con,$_POST['bed']);
                         $Nroom=mysqli_real_escape_string($con,$_POST['nroom']);
                         $Meal=mysqli_real_escape_string($con,$_POST['meal']);
                         $Checkin=mysqli_real_escape_string($con,$_POST['cin']);
                         $Checkout=mysqli_real_escape_string($con,$_POST['cout']);


                         function codeData($value) {
                           $num=1;
                           $l=strlen($value);
                           $rev="";
                           for($i=0;$i<$l;$i++)
```

```
                    {
                     $a1=chr(ord(substr($value,$i,1))+$num);
                     $a1=mysqli_real_escape_string($GLOBALS['con'],$a1);
                     $rev=$a1.$rev;
                    }

                    return $rev;
                   }

                   function codeBg($value)
                   {
                    $l=strlen($value);
                    $ch="";
                    for($i=0;$i<$l;$i++)
                    {
                     $a1=ord(substr($value,$i,1));
                     $ch=$ch."$a1";
                    }
                    return $ch;
                   }

                   $Fname=codeData($Fname);
                   $Email=codeData($Email);
                   $Nation=codeData($Nation);
                   $Country=codeData($Country);
                   $Phone=codeBg($Phone);
                 }
$new ="Not Conform";
$newUser="INSERT INTO `roombook`(`Title`, `FName`, `LName`, `Email`, `National`,
`Country`, `Phone`, `TRoom`, `Bed`, `NRoom`, `Meal`, `cin`, `cout`,`stat`,`nodays`) VALUES
('$_POST[title]','$Fname','$Lname','$Email','$Nation','$Country','$Phone','$Troom','$Btype','$Nro
om','$Meal','$Checkin','$Checkout','$new',datediff('$_POST[cin]','$_POST[cout]'))";
if (mysqli_query($con,$newUser))
{
echo "<script type='text/javascript'> alert('Your Booking application has been sent')</script>";

}
else
{
echo "<script type='text/javascript'> alert('Error adding user in database')</script>";
}

$msg="Your code is correct";
            }
                }
              }
                        ?>
        </form>
        </div>
        </div>
        </div>
```

Home.php

This code is from the home.php file which basically includes the decryption of the data from the database where Firstname (FName), Email, Nationality, Country, Phone were encrypted by making using application level encryption method and stored in the database. This is done basically to avoid any middle-level hacker to access the data and hack. Now this data is decrypted and shown in the admin for confirming the room for the guest/customer

```
<div class="panel-body">
        <div class="table-responsive">
          <table class="table">
            <thead>
              <tr>
                <th>#</th>
                <th>Name</th>
                <th>Email</th>
                <th>Country</th>
                <th>Room</th>
                <th>Bedding</th>
                <th>Meal</th>
                <th>Check In</th>
                <th>Check Out</th>
                <th>Status</th>
                <th>More</th>
                 </tr>
                 </thead>
                <tbody>


            <?php
                    $tsql = "select * from roombook";
                     $tre = mysqli_query($con,$tsql);
                    function codeData($rev)
                    {
                       $l=strlen($rev);
                       $bk="";
                       for($i=0;$i<$l;$i++)
                      {
                       $a1=chr(ord(substr($rev,$i,1))-1);
                       $bk=$a1.$bk;
                      }
                       return $bk;
                    }
                    while($trow=mysqli_fetch_array($tre) )
                    {
                              $co =$trow['stat'];
                              if($co=="Not Conform")
                    {
```

```
                                $Fname=codeData($trow['FName']);
                                 $Email=codeData($trow['Email']);
                                 $Country=codeData($trow['Country']);

                                echo"<tr>
                               <th>".$trow['id']."</th>
                               <th>".$Fname." ".$trow['LName']."</th>
                               <th>".$Email."</th>

<th>".$Country."</th>
<th>".$trow['TRoom']."</th>
<th>".$trow['Bed']."</th>
<th>".$trow['Meal']."</th>
<th>".$trow['cin']."</th>
<th>".$trow['cout']."</th>
<th>".$trow['stat']."</th>

<th><a href='roombook.php?rid=".$trow['id']." ' class='btn btn-primary'>Action</a></th>

</tr>";
}
}
  </tbody>
                      </table>

                    </div>
                  </div>
                </div>
    </div>
                        </div>
                      </div>
                    <?php

                    $rsql = "SELECT * FROM `roombook`";
          $rre = mysqli_query($con,$rsql);
                    $r =0;
                    while($row=mysqli_fetch_array($rre) )
                    {
                    $br = $row['stat'];
                    if($br=="Conform")
                    {
                    $r = $r + 1;
                    }
          }

          ?>
```

Testing

Testing is the process of detecting errors. Testing performs a very critical role for quality assurance and for ensuring the reliability of software. The results of testing are used later on during maintenance also.

The main objectives of the testing are:

- A successful test is one that uncovers an as yet undiscovered error.
- A good test case is one that has a high probability of finding error, if it exists.
- The tests are inadequate to detect possibly present errors.
- The software more or less confirms to the quality and reliable standards.

Levels/Types of Testing:

System Testing

The philosophy behind testing is to find errors. Test cases are devised with this in mind. A strategy employed for system testing is code testing.

Code Testing:

his strategy examines the logic of the program. To follow this method we developed some test data that resulted in executing every instruction in the program and module i.e. every path is tested

Unit Testing

Unit testing focuses verification effort on the smallest unit of software i.e. the module. Using the detailed design and the process specifications testing is done to uncover errors within the boundary of the module. All modules must be successful in the unit test before the start of the integration testing begins.

Link Testing

Link testing does not test software but rather the integration of each module in system. The primary concern is the compatibility of each module.

Integration Testing

After the unit testing, we have to perform integration testing. The goal here is to see if modules can be integrated properly, the emphasis being on testing interfaces between modules.

System Testing

Here the entire software system is tested. The reference document for this process is the requirements document, and the goal is to see if software meets its requirements.

Acceptance Testing

Acceptance Test is performed with realistic data of the client to demonstrate that the software is working satisfactorily. Testing here is focused on external behavior of the system; the internal logic of program is not emphasized.

White Box Testing

This is a unit testing method where a unit will be taken at a time and tested thoroughly at a statement level to find the maximum possible errors.

Black Box Testing

This testing method considers a module as a single unit and checks the unit at interface and communication with other modules rather getting into details at statement level.

DESIGN OF TEST CASES

A test case is a document, which has a set of test data, preconditions, expected results and postconditions, developed for a particular test scenario in order to verify compliance against a specific requirement.

Test Case acts as the starting point for the test execution, and after applying a set of input values, the application has a definitive outcome and leaves the system at some end point or also known as execution postcondition.

Typical Test Case Parameters:

- Test Case ID
- Test Scenario
- Test Case Description
- Test Steps
- Prerequisite
- Test Data
- Expected Result
- Test Parameters
- Actual Result
- Environment Information
- Comments

1. LOGIN FORM TESTING

Test Case ID	Module Name	Expected Input	Expected Result	Actual Input	Actual Output	Status Pass/Fail
1.	Login	Username, Password	Admin page	Admin id Admin password	Admin page	Pass

2. ROOM RESERVATION TEST

Test Case ID	Module Name	STEPS	Expected Input	Expected Result	Actual Input	Actual Output	Status Pass/Fail
2	Room reservation by customer	1.Fill in the required information in the form 2.Complete the verification step 3. Submit the form	Form to be filled for details regarding customer or guest t	Form is successfully submitted	Name, email id, nationality, address, type of room, type of meal and phone number	Form is successfully submitted	Pass

3. CHECKOUT

Test Case ID	Module Name	Steps	Expected Input	Expected Result	Actual Input	Actual Output	Status Pass/Fail
3	Checkout	1.Fill in the required information 2.Make a payment at checkout	Entering customer/guest details for payment	Payment successfully done	Credit card/debit cards details	Payment successfully done	Pass

4. CUSTOMER RECORD

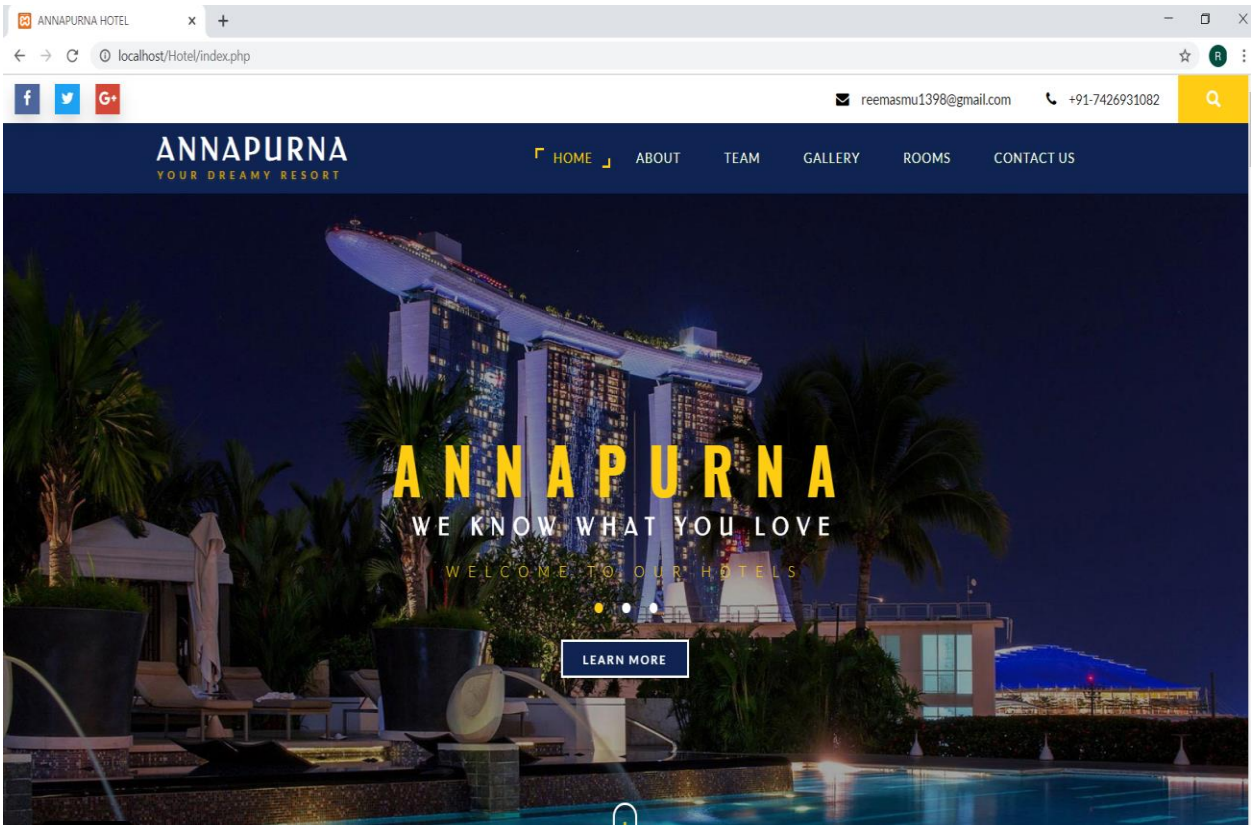
Test Case ID	Module Name	Steps	Expected Input	Expected Result	Actual Input	Actual Output	Status Pass/Fail
4	Customer record	1.Retrieve Customer record 2.Clear the customer record screen 3.Save newly entered customer record	Customer Record use case	All the required customer record and information in the database	Checking the customer record	Displayed customer record	Pass

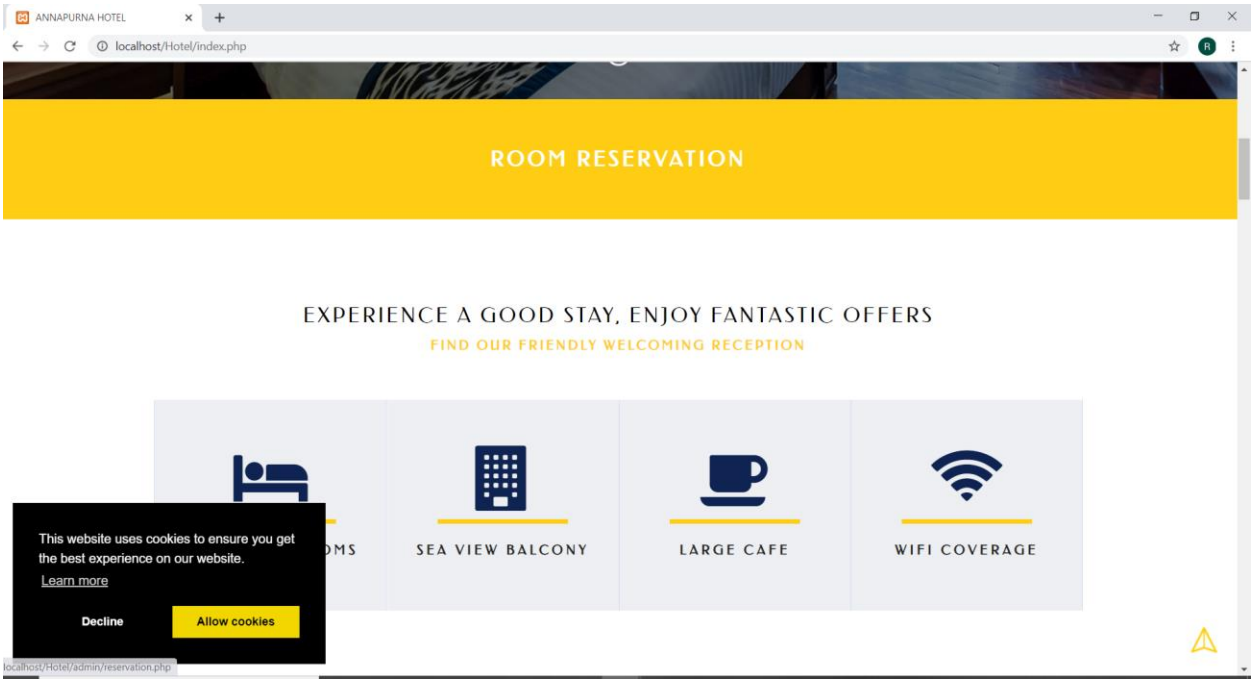
5. PAYMENT

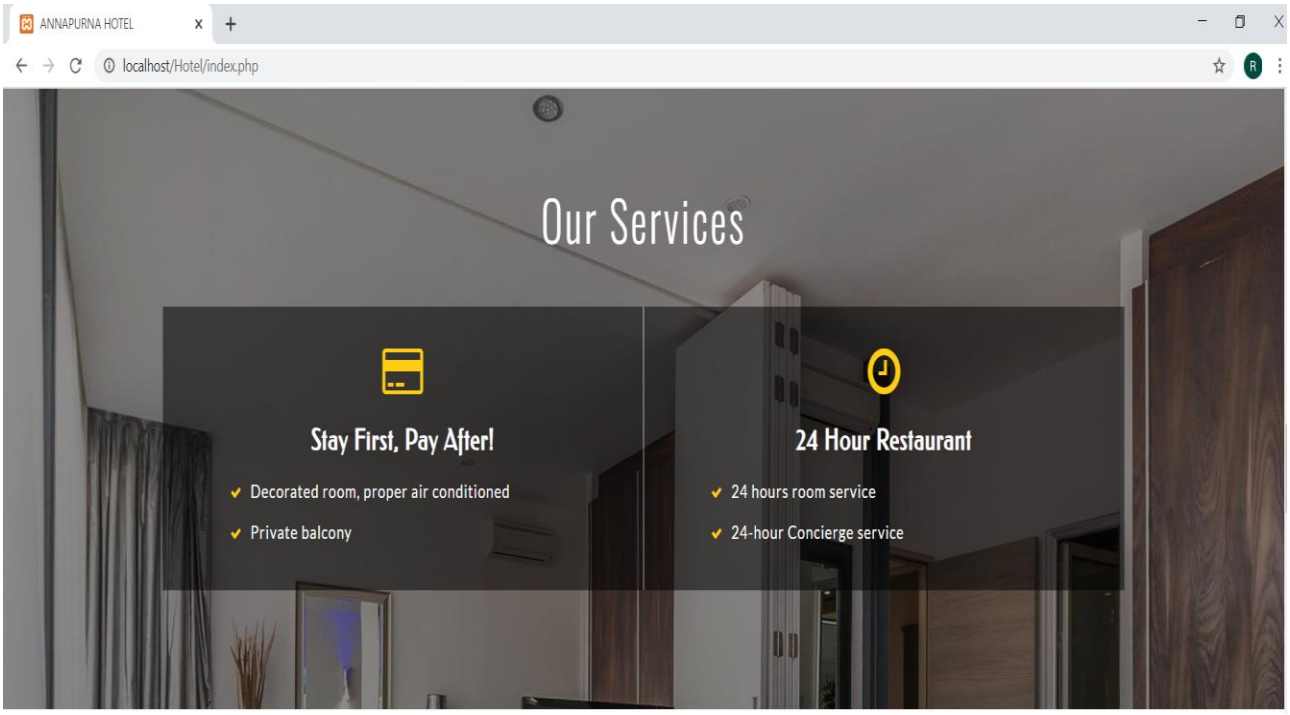
Test Case ID	Module Name	Steps	Expected Input	Expected Result	Actual Input	Actual Output	Status Pass/Fail
5	Payment	1.Generate the bill 2. Retrieve payment information 3. Clear the payment screen	Entering all information for payment and Checking if payment done successfully	Payment done successfully	Entering all the required information	Payement done Room is alloted	Pass

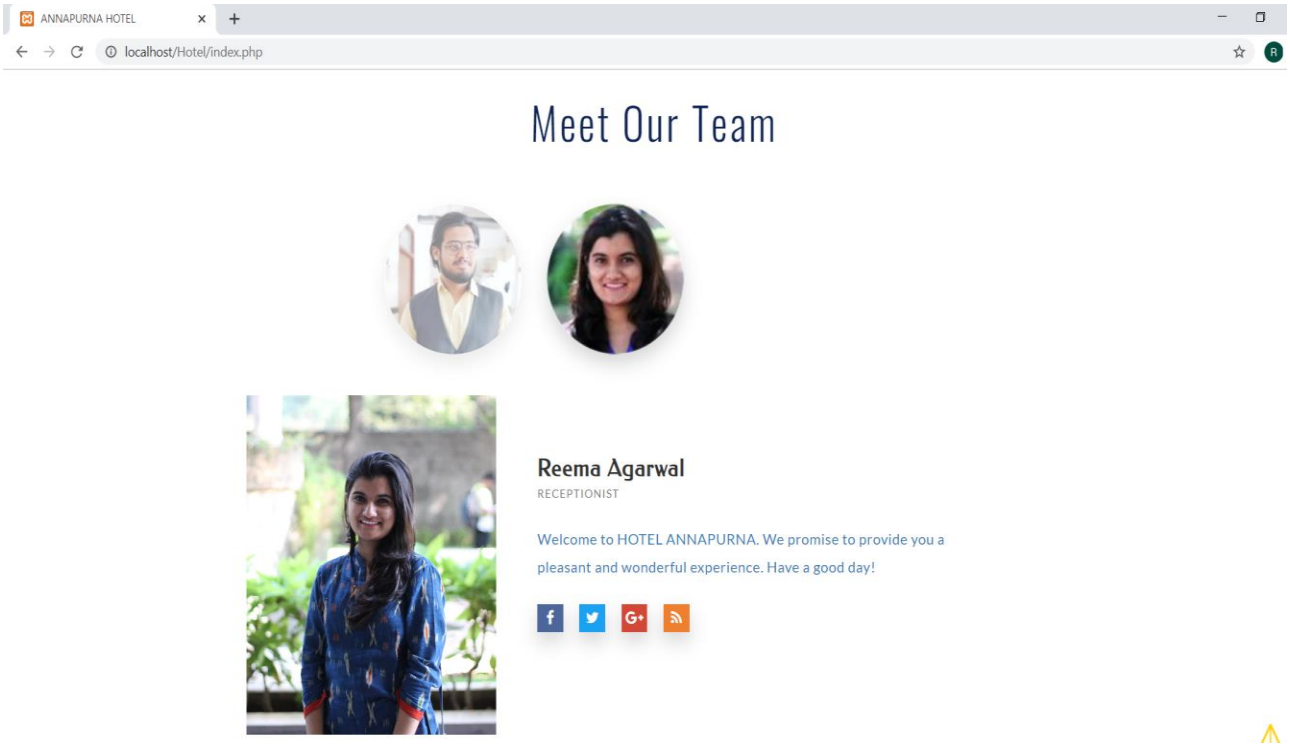
RESULT ANALYSIS AND DISCUSSION

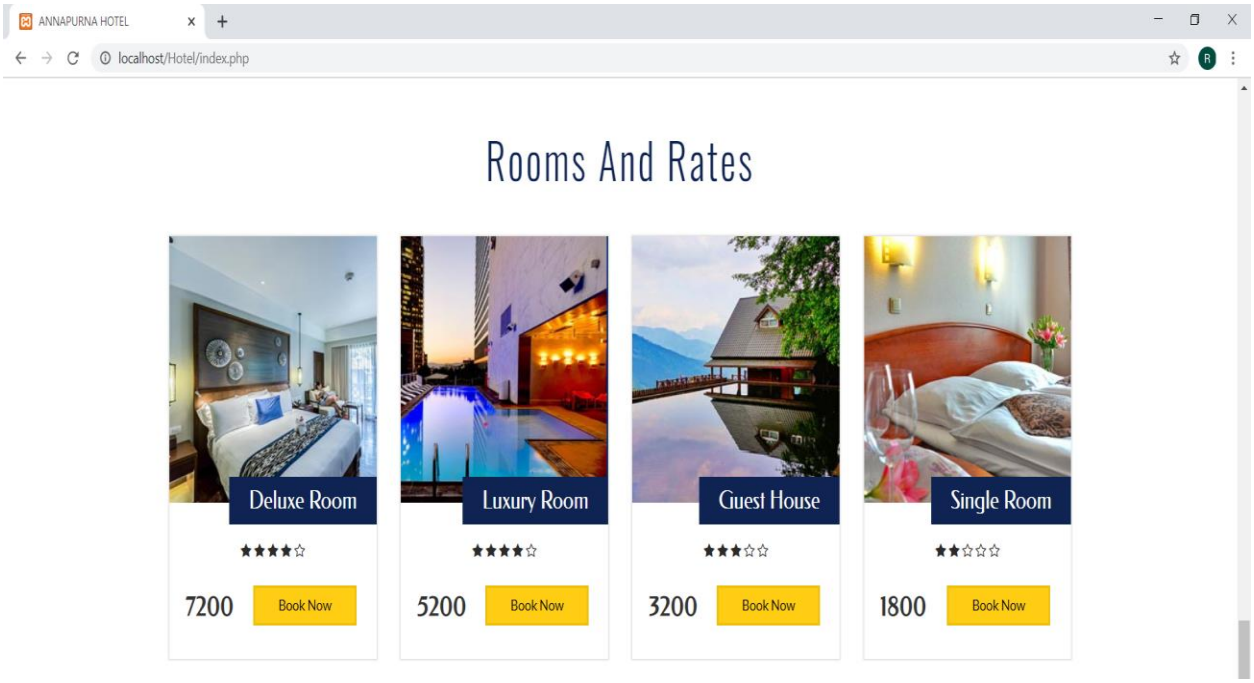
HOME PAGE

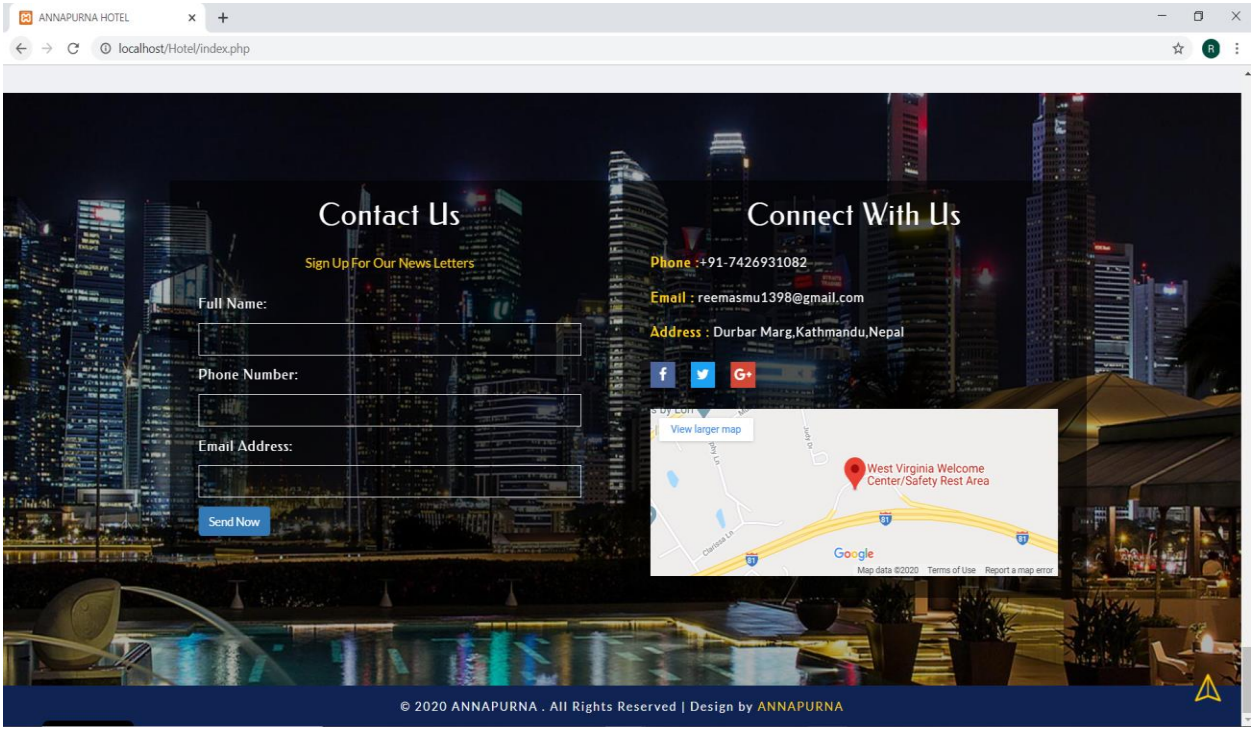





ANNAPURNA HOTEL
localhost/Hotel/index.php
Our Services
Stay First, Pay After!
Decorated room, proper air conditioned
Private balcony
24 Hour Restaurant
24 hours room service
24-hour Concierge service

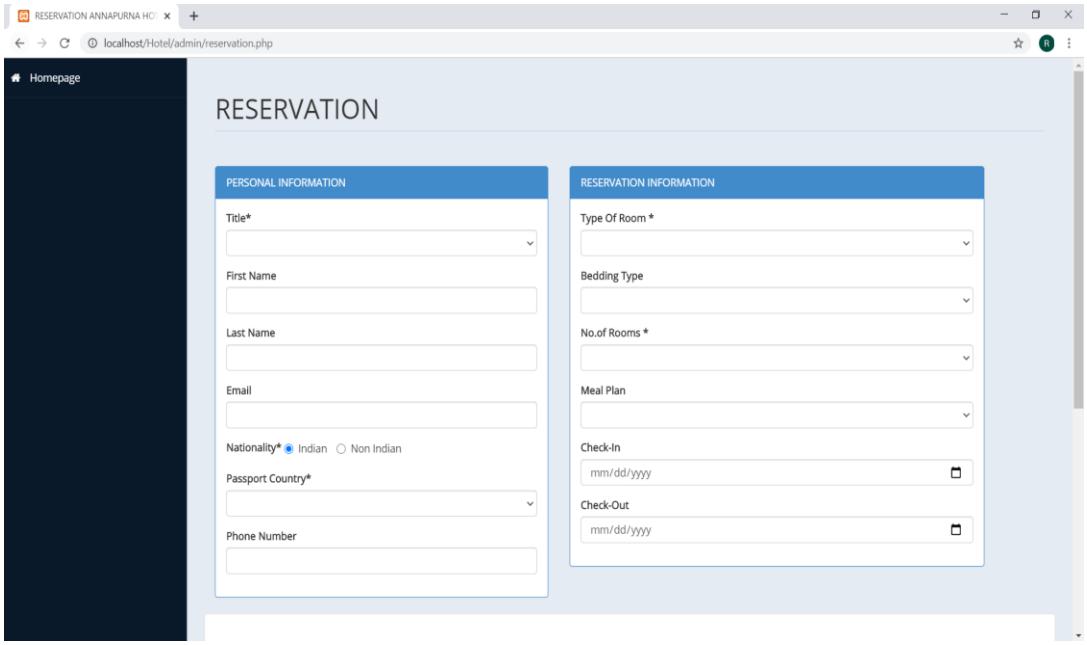

ANNAPURNA HOTEL
localhost/Hotel/index.php
Meet Our Team
Reema Agarwal
RECEPTIONIST
Welcome to HOTEL ANNAPURNA. We promise to provide you a pleasant and wonderful experience. Have a good day!





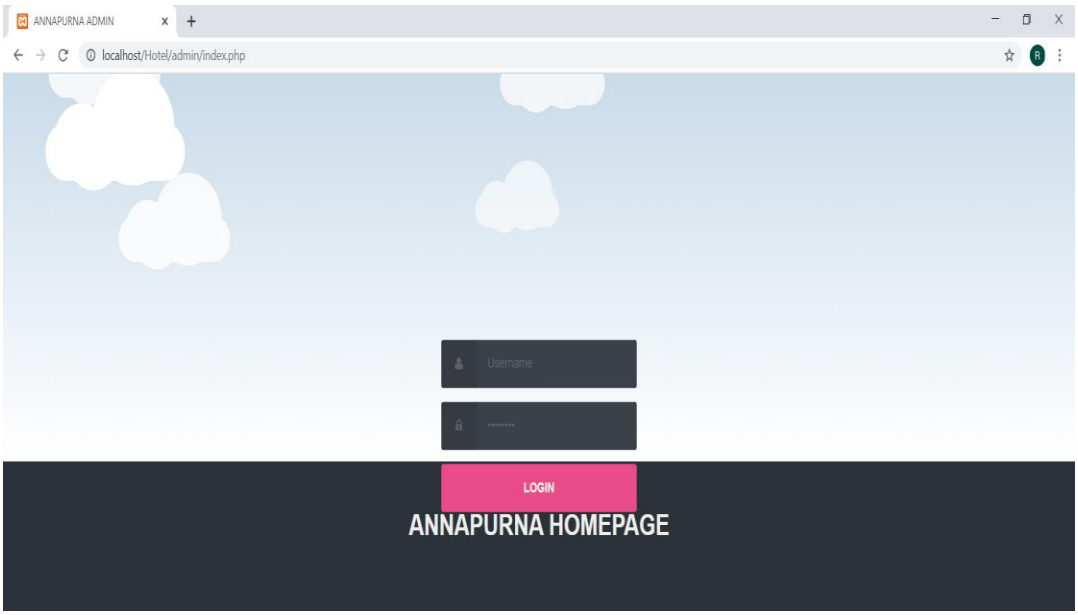
These entire screenshots are the main/home page of the website which includes entire details regarding the hotel Annapurna. User/Guest can visit the website and can go under room reservation in order to book their room.

ROOM RESERVATION



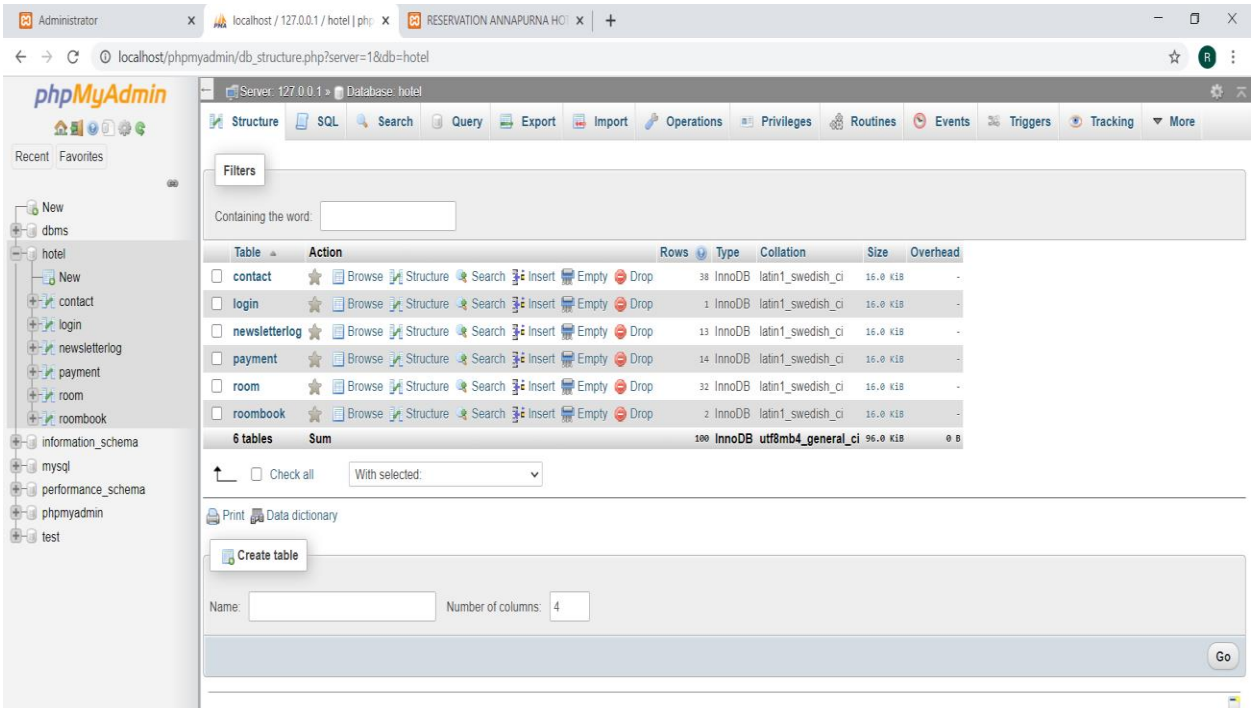
This is the form which the user/customer has to fill when they want to book their room online through our website.

ADMIN PAGE



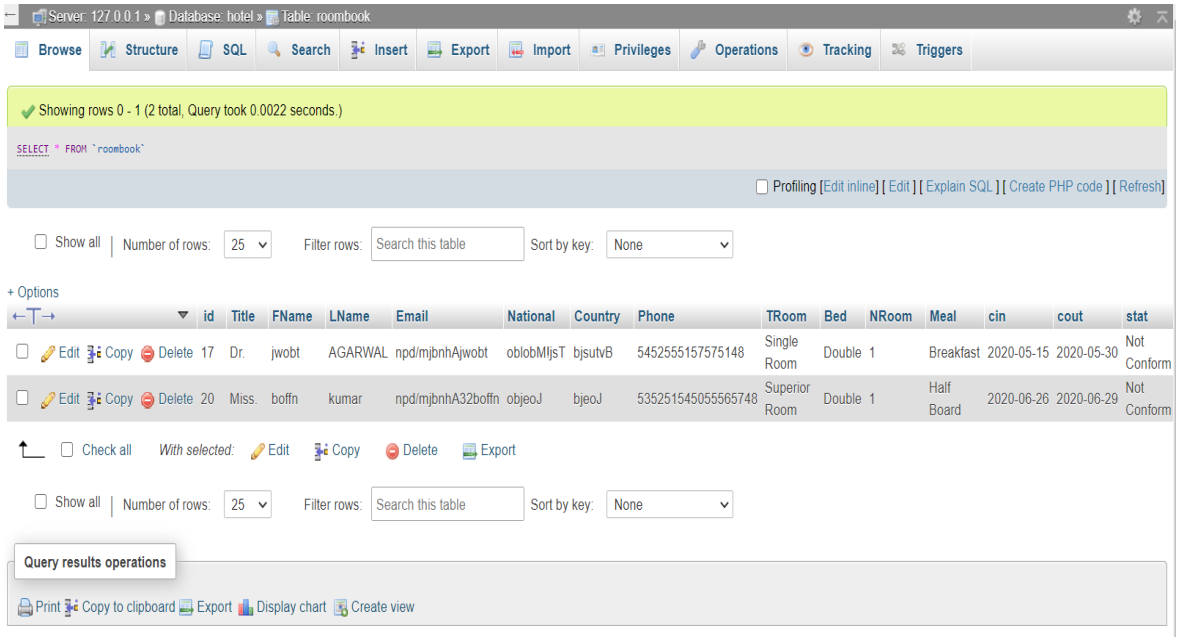
This is the admin page where admin has to enter his id and password to get logged in. The admin will be directed to his own admin page whereby he can make all the changes.

DATABASE



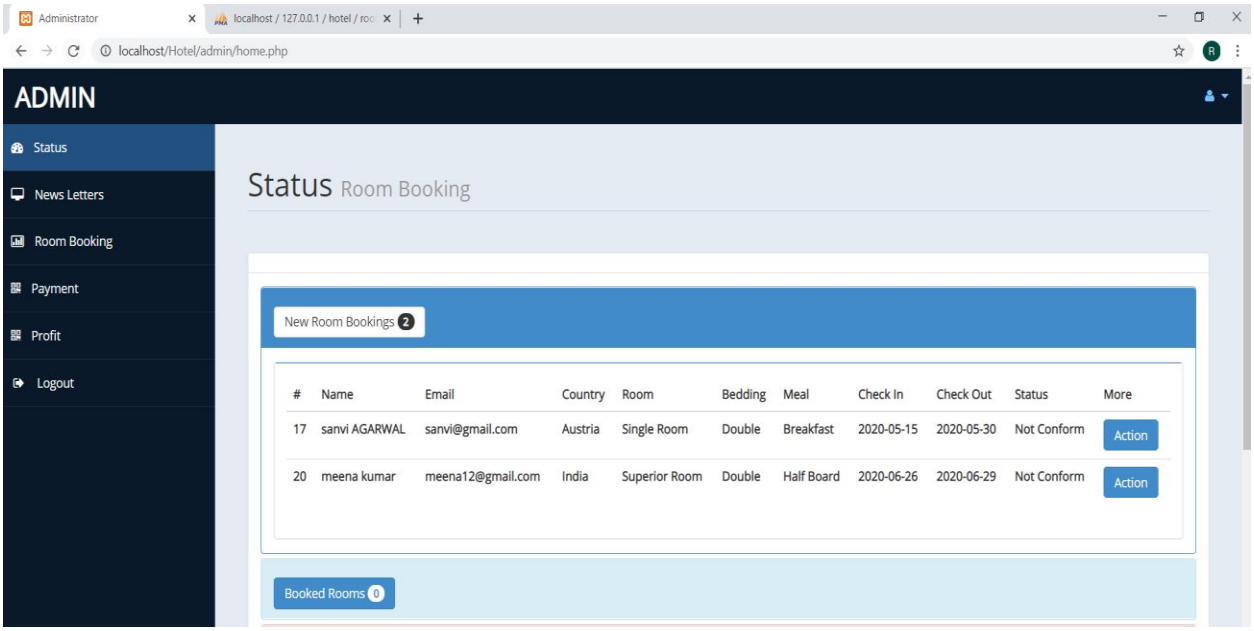
This are the different data's being stored in our database which contains all the details regarding the payment, room, room reservation, login and contact information.

ENCRYPTED DATA



Using the application level encryption we encrypted the customers data such as their first name, country, nationality, phone number and email id such there is no leakage of any data.

DECRYPTED DATA



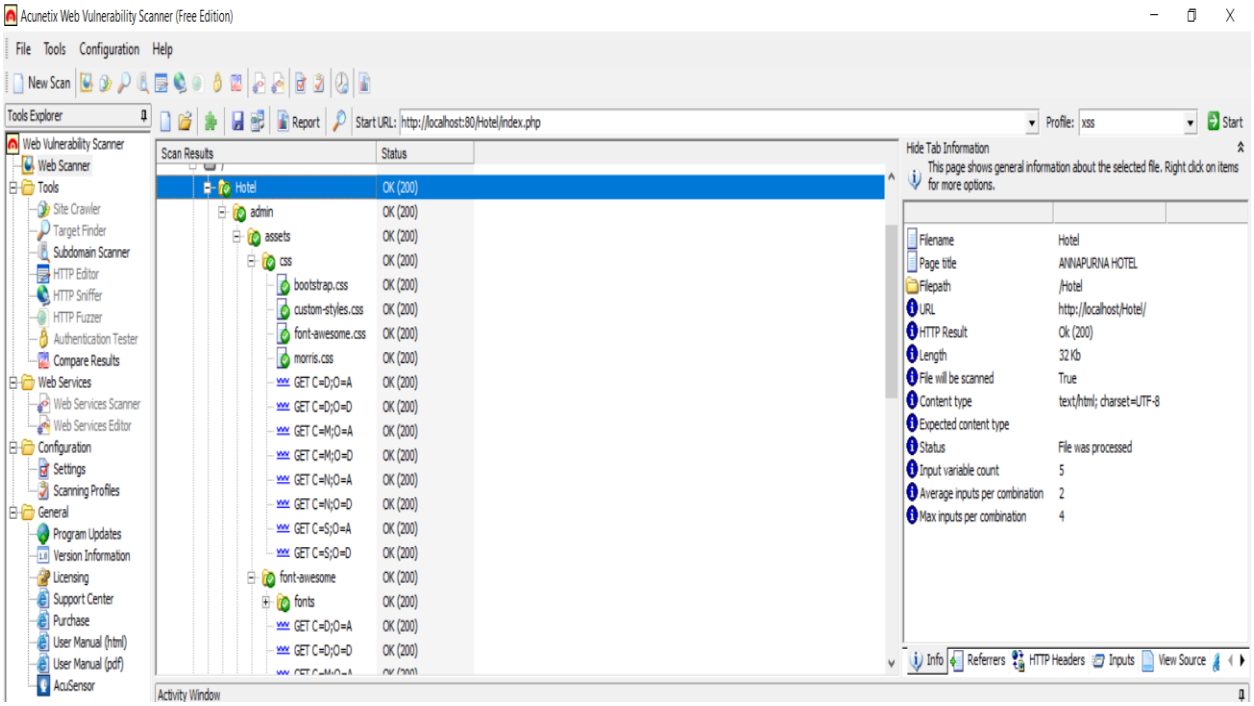
Since the admin has to confirm the information about their customer and their stay, the customers data are decrypted and only visible to admin when he log in into his account to confirm the stay of the guest.

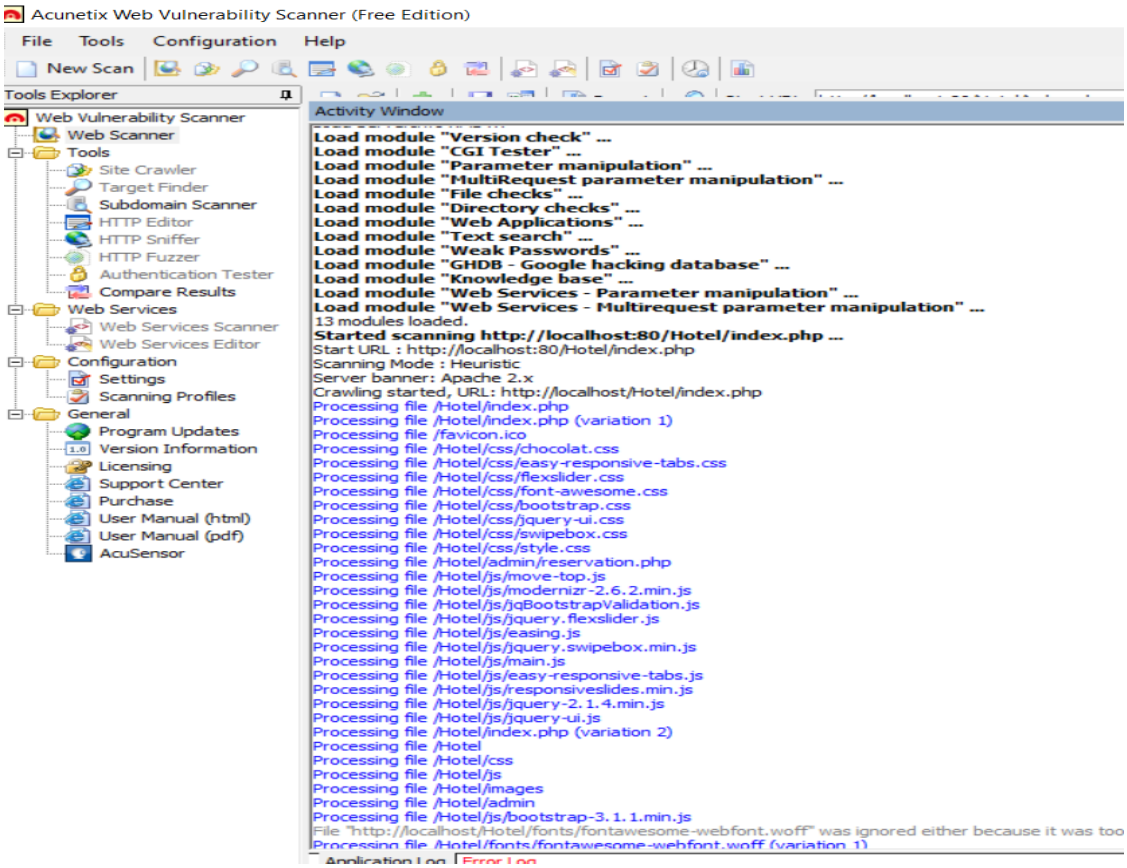
Unit Test Summary and Results

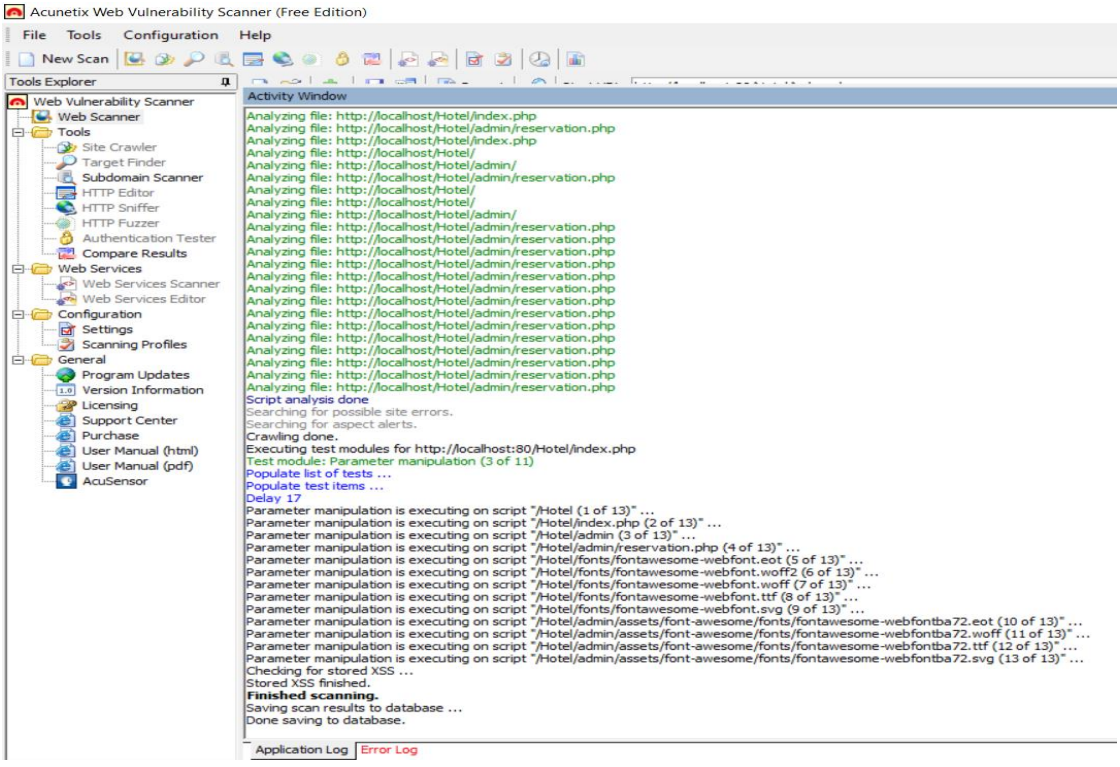
Test	Use Case Derivation	Test Result
A.Reports		
1. Generate a Report on Hotel Occupancy	Generate a Report Use Case	Passed
2. Generate a Report on Hotel Revenue	Generate a Report Use Case	Passed
3. Generate a Report on Default Price Exceptions	Generate a Report Use Case	Passed
4. Generate a Report on Pending Check Outs	Generate Pending CheckOut Report Use Case	Passed
B.User Administration		Passed
1. Find a User	Administer User Profile Use Case	Passed
2. Add a New User	Administer User Profile Use Case	Passed
3. Update a Users Information	Administer User Profile Use Case	Passed
4. Clear information in Text Field	Administer User Profile Use Case	Passed
5. Delete a User	Administer User Profile Use Case	Passed
6. View Updated List of Users	Administer User Profile Use Case	Passed
C.Room Administration		Passed
1. Find a Room	Administer Hotel System Use Case	Passed
2. Add a Room	Administer Hotel System Use Case	Passed
3. Update a Room	Administer Hotel System Use Case	Passed
4. Clear the Text Information	Administer Hotel System Use Case	Passed
5. Update Room Rate and Number of Beds	Administer Hotel System Use Case	Passed
6. Clear Text Field Information	Administer Hotel System Use Case	Passed
D.Reservations		Passed
1. Make a Reservation	Make a Reservation Use case	Passed
2. Apply a Room Rate Discount	Adjust Default Room Rate Extention Use Case	Passed
3. Remove Room Rate Discount	Adjust Default Room Rate Extention Use Case	Passed
4. Cancel Room Rate Discount	Adjust Default Room Rate Extention Use Case	Passed
5. Clear the Reservation Screen	Make a Reservation Use case	Passed
6. Retrieve a Reservation	Retrieve Reservation Use Case	Passed
7. Cancel a Reservation	Make a Reservation Use case	Passed

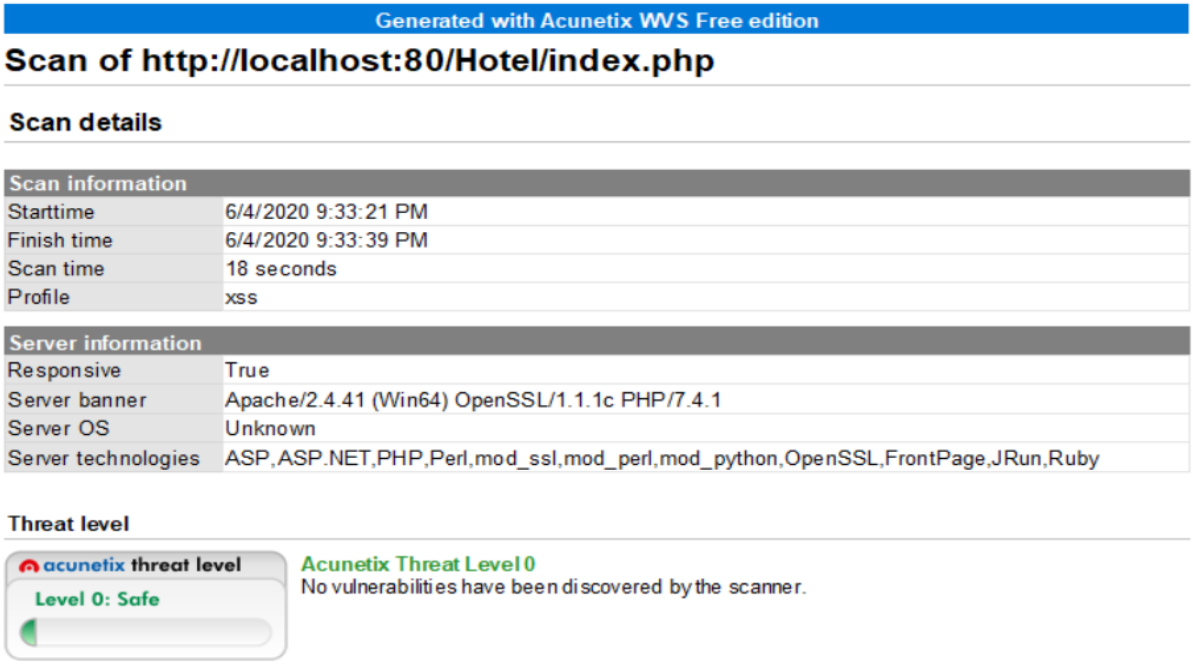
Unit Test Summary and Results (continued)

Test	Use Case Derivation	Test Result	Corrections
E.Room Availability			
1. Check Room Availability	Room Use Case	Passed	
2. Clear Information in Text Field	Room Use Case	Passed	
F.Customer Record		Passed	
1. Retrieve a customer Record	Customer Record Use Case	Passed	
2. Clear the Customer Record Screen	Customer Record Use Case	Passed	
3. Save Newly Entered Customer Record	Customer Record Use Case	Passed	
G.Check In		Passed	
1. Check In a Customer With No Reservation	Check In Use Case	Failed	Passed correct parameters
2. Apply a Discount at Check In	Check In Use Case	Passed	
3. Remove Discount at Check In	Check In Use Case	Passed	
4. Cancel Discount Information at Check In	Check In Use Case	Passed	
5. Pay at Check In	Check In Use Case	Passed	
6. Check In a Customer With a Reservation	Check In Use Case	Passed	
7. Clear the Check In Screen	Check In Use Case	Passed	
H.Check Out		Passed	
1. Make a Payment at Check out	Check Out Use Case	Passed	
2. Retrieve a Customer at Check Out	Check Out Use Case	Passed	
3. Clear the Check Out Screen	Check Out Use Case	Passed	
I.Payment		Passed	
1. Generate Bill	Make Payment Use Case	Passed	
2. Retrieve Payment Information	Make Payment Use Case	Passed	
3. Clear the Payment Screen	Make Payment Use Case	Passed	




Acunetix Web Vulnerability Scanner (Free Edition)
File Tools Configuration Help
New Scan
Tools Explorer
Web Vulnerability Scanner
Web Scanner
Tools
Site Crawler
Target Finder
Subdomain Scanner
HTTP Editor
HTTP Sniffer
HTTP Fuzzer
Authentication Tester
Compare Results
Web Services
Web Services Scanner
Web Services Editor
Configuration
Settings
Scanning Profiles
General
Program Updates
Version Information
Licensing
Support Center
Purchase
User Manual (html)
User Manual (pdf)
AcuSensor
Activity Window
Load module "Version check" ...
Load module "CGI Tester" ...
Load module "Parameter manipulation" ...
Load module "MultiRequest parameter manipulation" ...
Load module "File checks" ...
Load module "Directory checks" ...
Load module "Web Applications" ...
Load module "Text search" ...
Load module "Weak Passwords" ...
Load module "GHDB - Google hacking database" ...
Load module "Knowledge base" ...
Load module "Web Services - Parameter manipulation" ...
Load module "Web Services - Multirequest parameter manipulation" ...
13 modules loaded.
Started scanning http://localhost:80/Hotel/index.php ...
Start URL : http://localhost:80/Hotel/index.php
Scanning Mode : Heuristic
Server banner: Apache 2.x
Crawling started, URL: http://localhost/Hotel/index.php
Processing file /Hotel/index.php
Processing file /Hotel/index.php (variation 1)
Processing file /favicon.ico
Processing file /Hotel/css/chocolat.css
Processing file /Hotel/css/easy-responsive-tabs.css
Processing file /Hotel/css/flexslider.css
Processing file /Hotel/css/font-awesome.css
Processing file /Hotel/css/bootstrap.css
Processing file /Hotel/css/jquery-ui.css
Processing file /Hotel/css/swipebox.css
Processing file /Hotel/css/style.css
Processing file /Hotel/admin/reservation.php
Processing file /Hotel/js/move-top.js
Processing file /Hotel/js/modernizr-2.6.2.min.js
Processing file /Hotel/js/jqBootstrapValidation.js
Processing file /Hotel/js/jquery.flexslider.js
Processing file /Hotel/js/easing.js
Processing file /Hotel/js/jquery.swipebox.min.js
Processing file /Hotel/js/main.js
Processing file /Hotel/js/easy-responsive-tabs.js
Processing file /Hotel/js/responsiveslides.min.js
Processing file /Hotel/js/jquery-2.1.4.min.js
Processing file /Hotel/js/jquery-ui.js
Processing file /Hotel/index.php (variation 2)
Processing file /Hotel
Processing file /Hotel/css
Processing file /Hotel/js
Processing file /Hotel/images
Processing file /Hotel/admin
Processing file /Hotel/js/bootstrap-3.1.1.min.js
File "http://localhost/Hotel/fonts/fontawesome-webfont.woff" was ignored either because it was too
Application Log
Error Log


Acunetix Web Vulnerability Scanner (Free Edition)
File Tools Configuration Help
New Scan
Tools Explorer
Web Vulnerability Scanner
Web Scanner
Tools
Site Crawler
Target Finder
Subdomain Scanner
HTTP Editor
HTTP Sniffer
HTTP Fuzzer
Authentication Tester
Compare Results
Web Services
Web Services Scanner
Web Services Editor
Configuration
Settings
Scanning Profiles
General
Program Updates
Version Information
Licensing
Support Center
Purchase
User Manual (html)
User Manual (pdf)
AcuSensor
Activity Window
Analyzing file: http://localhost/Hotel/index.php
Analyzing file: http://localhost/Hotel/admin/reservation.php
Analyzing file: http://localhost/Hotel/index.php
Analyzing file: http://localhost/Hotel/
Analyzing file: http://localhost/Hotel/admin/
Analyzing file: http://localhost/Hotel/admin/reservation.php
Analyzing file: http://localhost/Hotel/
Analyzing file: http://localhost/Hotel/
Analyzing file: http://localhost/Hotel/admin/
Analyzing file: http://localhost/Hotel/admin/reservation.php
Analyzing file: http://localhost/Hotel/admin/reservation.php
Analyzing file: http://localhost/Hotel/admin/reservation.php
Analyzing file: http://localhost/Hotel/admin/reservation.php
Analyzing file: http://localhost/Hotel/admin/reservation.php
Analyzing file: http://localhost/Hotel/admin/reservation.php
Analyzing file: http://localhost/Hotel/admin/reservation.php
Analyzing file: http://localhost/Hotel/admin/reservation.php
Analyzing file: http://localhost/Hotel/admin/reservation.php
Analyzing file: http://localhost/Hotel/admin/reservation.php
Analyzing file: http://localhost/Hotel/admin/reservation.php
Analyzing file: http://localhost/Hotel/admin/reservation.php
Analyzing file: http://localhost/Hotel/admin/reservation.php
Analyzing file: http://localhost/Hotel/admin/reservation.php
Script analysis done
Searching for possible site errors.
Searching for aspect alerts.
Crawling done.
Executing test modules for http://localhost:80/Hotel/index.php
Test module: Parameter manipulation (3 of 11)
Populate list of tests ...
Populate test items ...
Delay 17
Parameter manipulation is executing on script "/Hotel (1 of 13)" ...
Parameter manipulation is executing on script "/Hotel/index.php (2 of 13)" ...
Parameter manipulation is executing on script "/Hotel/admin (3 of 13)" ...
Parameter manipulation is executing on script "/Hotel/admin/reservation.php (4 of 13)" ...
Parameter manipulation is executing on script "/Hotel/fonts/fontawesome-webfont.eot (5 of 13)" ...
Parameter manipulation is executing on script "/Hotel/fonts/fontawesome-webfont.woff2 (6 of 13)" ...
Parameter manipulation is executing on script "/Hotel/fonts/fontawesome-webfont.woff (7 of 13)" ...
Parameter manipulation is executing on script "/Hotel/fonts/fontawesome-webfont.ttf (8 of 13)" ...
Parameter manipulation is executing on script "/Hotel/fonts/fontawesome-webfont.svg (9 of 13)" ...
Parameter manipulation is executing on script "/Hotel/admin/assets/font-awesome/fonts/fontawesome-webfontba72.eot (10 of 13)" ...
Parameter manipulation is executing on script "/Hotel/admin/assets/font-awesome/fonts/fontawesome-webfontba72.woff (11 of 13)" ...
Parameter manipulation is executing on script "/Hotel/admin/assets/font-awesome/fonts/fontawesome-webfontba72.ttf (12 of 13)" ...
Parameter manipulation is executing on script "/Hotel/admin/assets/font-awesome/fonts/fontawesome-webfontba72.svg (13 of 13)" ...
Checking for stored XSS ...
Stored XSS finished.
Finished scanning.
Saving scan results to database ...
Done saving to database.
Application Log
Error Log



Generated with Acunetix WVS Free edition

Scan of http://localhost:80/Hotel/index.php

Scan details

Scan information	
Starttime	6/4/2020 9:33:21 PM
Finish time	6/4/2020 9:33:39 PM
Scan time	18 seconds
Profile	xss

Server information	
Responsive	True
Server banner	Apache/2.4.41 (Win64) OpenSSL/1.1.1c PHP/7.4.1
Server OS	Unknown
Server technologies	ASP,ASP.NET,PHP,Perl,mod_ssl,mod_perl,mod_python,OpenSSL,FrontPage,JRun,Ruby

Threat level

acunetix threat level

Level 0: Safe

Acunetix Threat Level 0

No vulnerabilities have been discovered by the scanner.

We made use of Acunetix Web Vulnerability Scanner in order to check the vulnerability in our website. Acunetix is an automated web application security testing tool that audits your web applications by checking for vulnerabilities like SQL Injection, Cross site scripting and other exploitable vulnerabilities. In general, Acunetix scans any website or web application that is accessible via a web browser and uses the HTTP/HTTPS protocol.

Acunetix offers a strong and unique solution for analyzing off-the-shelf and custom web applications including those utilizing JavaScript, AJAX and Web 2.0 web applications. Acunetix has an advanced crawler that can find almost any file. This is important since what is not found cannot be checked.

CONCLUSION AND FUTURE ENHANCEMENT

Our project hotel information management system using database encryption has helped to solve the problem of security threats, vulnerability issues, data redundancy, incidence of frauds and data inconsistency. We used to application level encryption to make sure data our data is encrypted and stored in database such that no hackers can attack and access the data of our customers. We also decrypted the data such that it will be visible only to admin and he can verify all the information of customer's booking and confirm it. Moreover, in order to make sure our website is not vulnerable to any king of threats we used acunetix vulnerability scanner which helped to scan our entire website and created a report to show if there are any issues or threats. It tested each and every module of our websites and reported 0 vulnerability in our website. Moreover, we also have presented unit testing summary report indicating all the test cases.

For future purpose we aim to work on other security tools in order to check the security threats of our website. We will try to impose all the testing mechanisms and tools in one website and provide report on the same. We also aim to build a tool of our own and implement it and make accessible to others such that other people can also get benefitted out of it and use it.