

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

(Hotel Information Management System using Database Encryption)

FINAL REVIEW REPORT

Prepared for

Software Engineering (CSE-3001) – PROJECT COMPONENT

Submitted to:

PROF. BHAVANI S

Assistant Professor (Senior)

Department of Digital Communication

School of Information Technology and Engineering



VIT[®]

Vellore Institute of Technology

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

School of Computer Science and Engineering
Fall 2020-21

TEAM MEMBER DETAILS

SL	NAME	REG NO.	MAIL ID:
1	Malla Jyotsna Sree Mahima	18BCE0912	jyotsna.sreemahima2018@vits tudent.ac.in
2	Preetham Lekkala	18BCE0854	satyapardha.preetham2018@vi tstudent.ac.in

TABLE OF CONTENTS

CHAPTER NO.	TITLE	PAGE NO.
1.	Title	1
2.	Team Members	2
3.	Problem Statement	3
4.	Proposed Work	7
5	Algorithm and Flowchart	13
6.	Lab Exercises	15
7.	UML	38
8.	Code	45
9.	Results and Discussion	50
10	Execution Screenshots	55

1. PROBLEM STATEMENT

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.

LIMITING OF THE EXISTING SYSTEM

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

2. PROPOSED WORK

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.

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 fulfills 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.

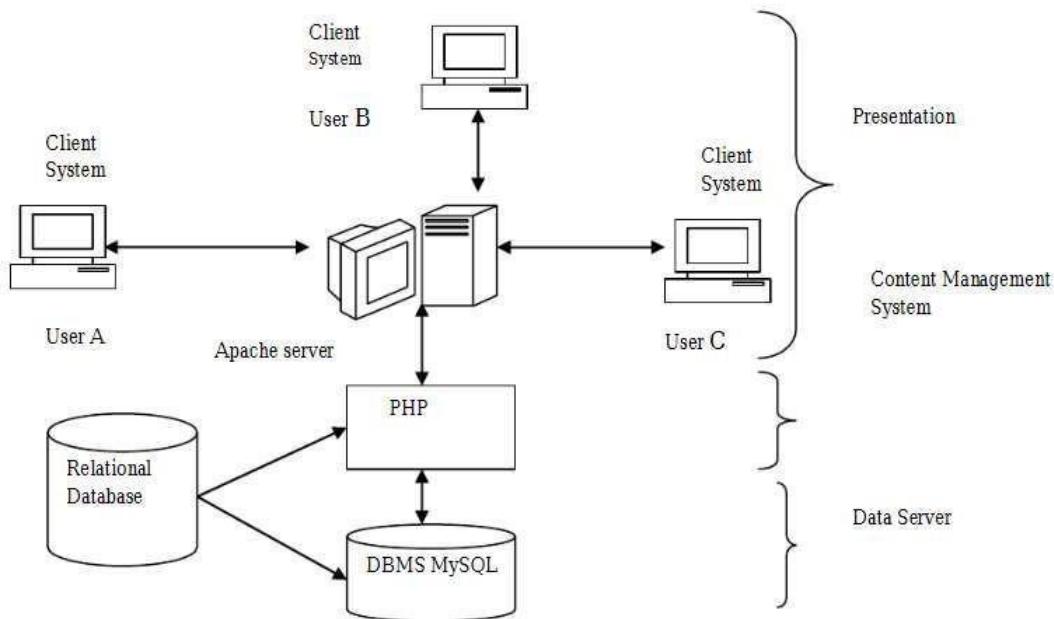


Figure 1: System Design

MODULE DESCRIPTION:

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

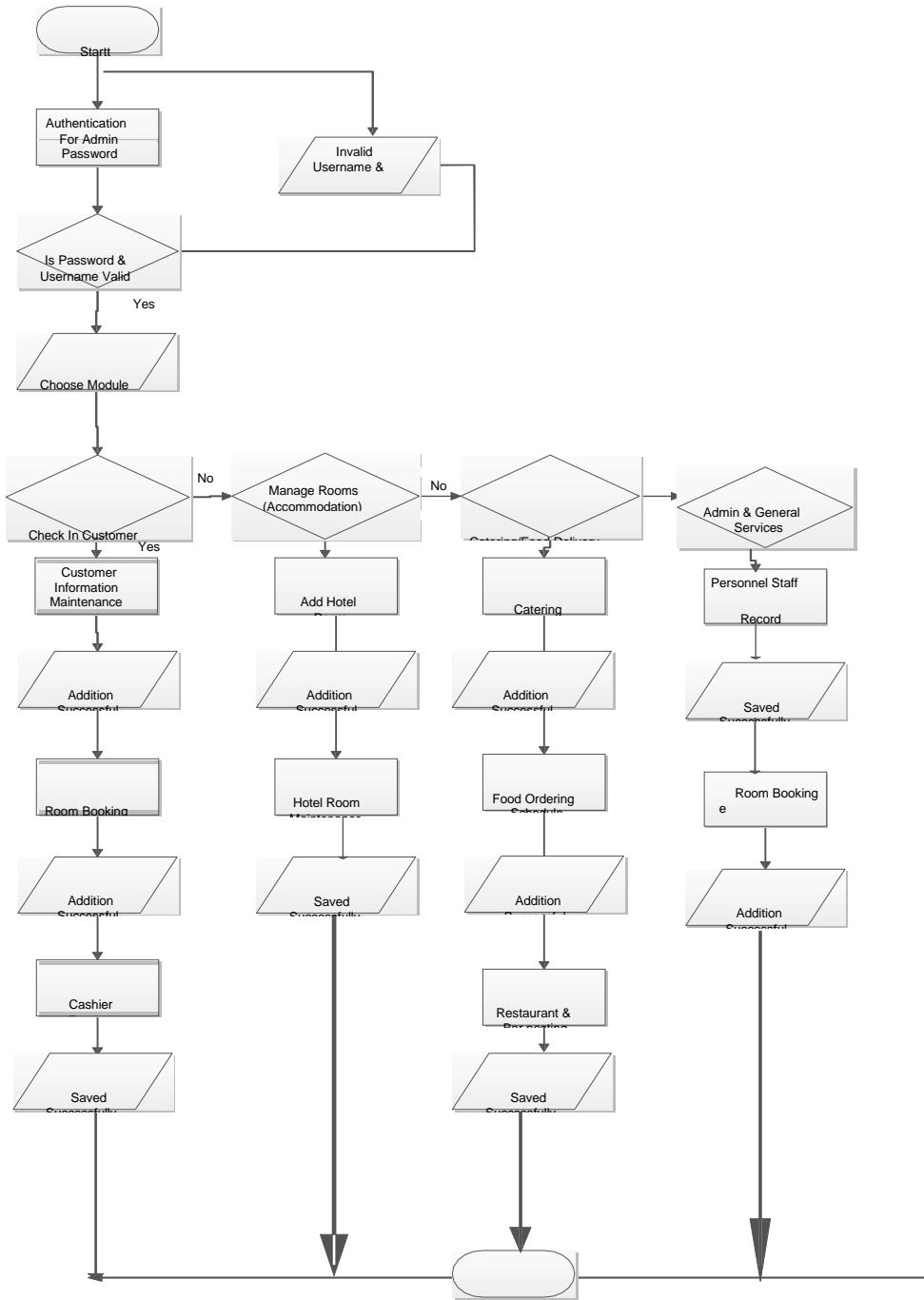
them and removing them. After the room is booked by the customer its manager's responsibility to confirm the bookings.

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.

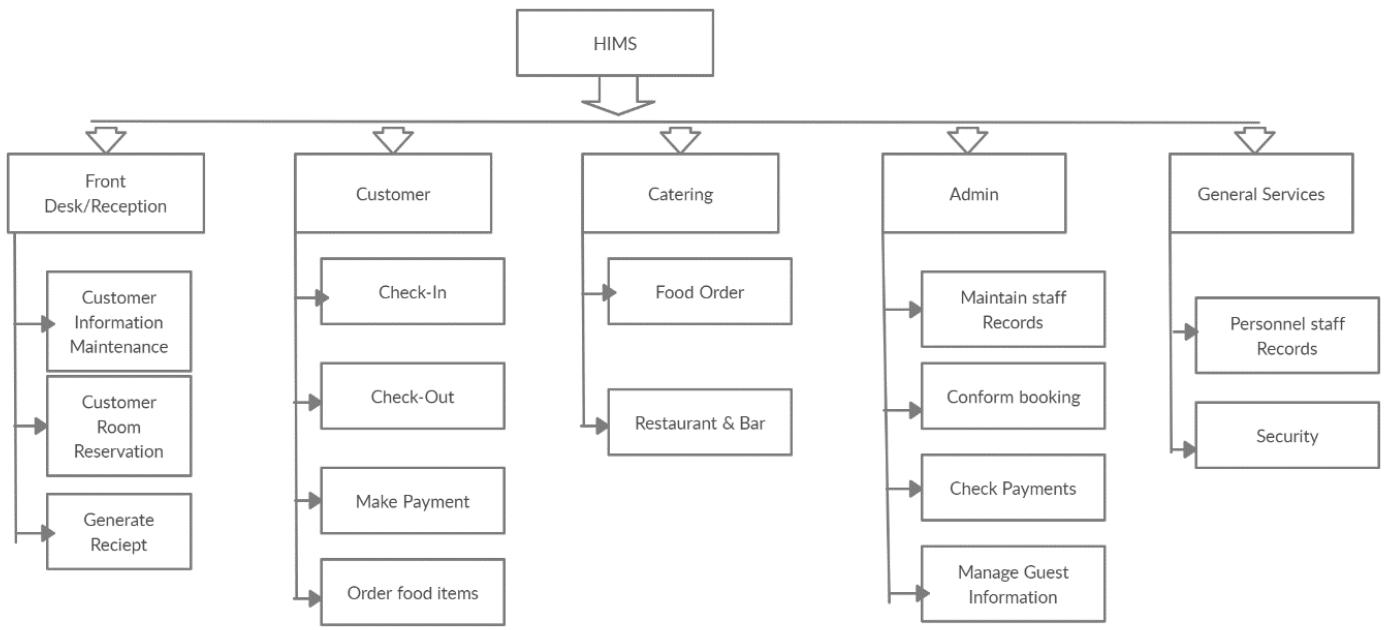
3. ALGORITHM AND FLOWCHART:



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.

4. LAB EXERCISES:

i. WBS Hotel Management System using Database Encryption:



ii. SRS

Software Requirements Specification

for

<HOTEL MANAGEMENT SYSTEM USING DATABASE ENCRYPTION>

Version <1.0>

Prepared by

Group Name: <HIMS>

**<Malla Jyotsna>
<Preetham
Lekkala>**

**<student1>
<student 2>**

**<jyotsna.sreemahima2018@vitstudent.ac.in>
<satyapardha.preetham2018@vitstudent.ac.in>**

Instructor: <Prof. Bhavani S>

Course: <Software Engineering>

Lab Section: <L37-L38>

Teaching Assistant: <Prof. Anil K>

Date: <21.07.20>

Contents

CONTENTS	XVII
REVISIONS	XVIII
1 INTRODUCTION.....	19
1.1 DOCUMENT PURPOSE	19
1.2 PRODUCT SCOPE.....	19
1.3 INTENDED AUDIENCE AND DOCUMENT OVERVIEW.....	20
1.4 DEFINITIONS, ACRONYMS AND ABBREVIATIONS.....	20
1.5 DOCUMENT CONVENTIONS	20
1.6 REFERENCES AND ACKNOWLEDGMENTS	20
2 OVERALL DESCRIPTION	20
2.1 PRODUCT OVERVIEW	20
2.2 PRODUCT FUNCTIONALITY	20
2.3 DESIGN AND IMPLEMENTATION CONSTRAINTS	24
2.4 ASSUMPTIONS AND DEPENDENCIES.....	24
3 SPECIFIC REQUIREMENTS	25
3.1 EXTERNAL INTERFACE REQUIREMENTS	25
3.2 FUNCTIONAL REQUIREMENTS	26
3.3 USE CASE MODEL	28
4 OTHER NON-FUNCTIONAL REQUIREMENTS.....	30
4.1 PERFORMANCE REQUIREMENTS	30
4.2 SAFETY AND SECURITY REQUIREMENTS	30
4.3 SOFTWARE QUALITY ATTRIBUTES	30
5 OTHER REQUIREMENTS	31
APPENDIX A – DATA DICTIONARY	ERROR! BOOKMARK NOT DEFINED.
APPENDIX B - GROUP LOG	ERROR! BOOKMARK NOT DEFINED.

Revisions

Version	Primary Author(s)	Description of Version	Date Completed
Draft 1.0	Malla Jyotsna, Preetham Lekkala	Version 1.0 Primary Version	21/07/2020

Introduction

Document Purpose

This document Software Requirements Specifications (SRS) document provide an overview of the entire SRS for the Hotel Management System using Database Encryption.

The Software Requirements Specification (SRS) will provide a detailed description of the requirements for the Hotel Management System (HMS). This SRS will allow for a complete understanding of what is to be expected of the HMS to be constructed. The clear understanding of the HMS and its' functionality will allow for the correct software to be developed for the end user and will be used for the development of the future stages of the project. This SRS will provide the foundation for the project. From this SRS, the HMS can be designed, constructed, and finally tested.

This SRS will be used by the software engineers constructing the HMS and the hotel end users. The software engineers will use the SRS to fully understand the expectations of this HMS to construct the appropriate software. The hotel end users will be able to use this SRS as a "test" to see if the software engineers will be constructing the system to their expectations. If it is not to their expectations the end users can specify how it is not to their liking and the software engineers will change the SRS to fit the end users' needs.

Product Scope

The software product to be produced is a Hotel Management System using Database Encryption which will automate the major hotel operations. This Hotel management will make sure to provide security on this web application. The design and implementation of an electronic hotel management system 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.

This system 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. It makes 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.

Aim is to design and 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.

Intended Audience and Document Overview

In the Hotel Management System using Database encryption the audience is divided into 3 categories:

- i. Primary stakeholders: Guests/Customers
- ii. Secondary stakeholders: Receptionists/Staff
- iii. Tertiary stakeholders: Admin

The SRS is organized into two main sections. The first is The Overall Description and the second is the Specific Requirements. The Overall Description will describe the requirements of the HMS from a general high level perspective. The Specific Requirements section will describe in detail the requirements of the system.

Definitions, Acronyms and Abbreviations

SRS – Software Requirements Specification HMS –
Hotel Management System

Subjective satisfaction – The overall satisfaction of the system

End users – The people who will be actually using the system

Document Conventions

<In general this document follows the IEEE formatting requirements. UseD Arial font size 11, or 12 throughout the document for text.

References and Acknowledgments

None

Overall Description

Product Overview

The HMS is an independent stand-alone system. It is totally self contained.

Hardware Interfaces

The HMS will be placed on PC's throughout the hotel.

Software Interfaces

All databases for the HMS will be configured using MySQL. These databases include hotel rooms and customers information. These can be modified by the end users. The room database will include the room numbers and if they are vacant or occupied. The customers information database will contain all the information of the customer such as first name, last name, number of occupants, assigned room, default room rate(may be changed), phone number, whether or not the room is guaranteed, credit card number, confirmation number, automatic cancellation date, expected check in date and time, actual check in date and time, expected check out date and time, amount owed by customer, and abbreviated customer feedback.

Application level Encryption using Symmetric key is used 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

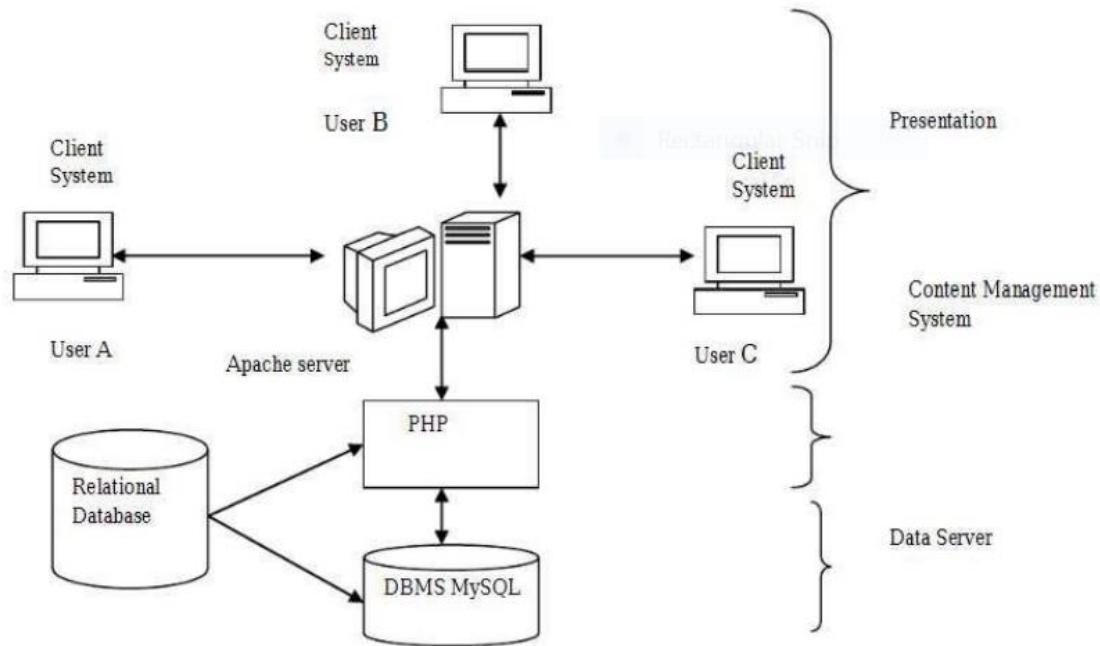


Figure 1: System Design

Product Functionality

Reservation and Booking System

- Allows for typing in customer information
- Has a default room rate that is adjustable
- Includes a description field for the changed rate
- When a customer checks in, the room number will be changed to occupied in the database.
- Ability to modify a reservation
- When no rooms are available and a customer would like to extend their reservation their information will be placed in a database and when there are rooms available the first customer on the list will have the room

-
- When a customer checks out the amount owed is displayed
 - If the internal clock states that is a customer's time to have checked out and customer has not checked out, adds an extra night to amount owed and provides a report
 - Records that room is vacant
 - Records payment
 - Allows for space to write customer's feedback.
 - Encrypt Customer and staff info.

Tracking and Selling Food System

- Tracks all meals purchased
- Charges the current room as necessary

Administration Services

- Decrypt Customer records
- Maintain Staff records
- Conform or delete Booking
- Conform or delete newsletters log.
- Add Rooms
- Delete Rooms
- Generate reports
- Analyze Profits
- Check Payments.

Design and Implementation Constraints

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.

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.

Assumptions and Dependencies

- The system is not required to save generated reports.
- Credit card payments are not included.

Specific Requirements

External Interface Requirements

The Hotel Management System will use the standard input/output devices for a personal computer. This includes the following:

- Keyboard
- Mouse
- Monitor
- Printer

User Interfaces

The User Interface Screens are described in table 1.

Table 1: Hotel Management User Interface Screens

Screen Name	Description
Login	Log into the system as a CSR or Manager
Reservation	Retrieve button, update/save reservation, cancel reservation, modify reservation, change reservation, adjust room rate, accept payment type/credit card
Check-in	Modify room stay (e.g., new credit card), check-in customer (with or without a reservation), adjust room rate, special requests, accept payment type/credit card
Checkout	Checkout customer, generate bill
Hotel Payment	Accept payment for room and food
Room Service/Restaurant	Create order, modify order, view order, cancel order, generate meal bill
Customer Record	Add or update customer records
Administer Rooms	Availability and rates
Administer User	Create, modify, and delete users; change password
Administer Meals	Create, modify, and delete meal items and prices
Reports	Select, view, save, and delete reports
Room Updation	Add or Delete rooms of any type and bedding

Hardware Interfaces

The system shall run on a Microsoft Windows based system.

Software Interfaces

The system shall interface with an Oracle or Access database.

Functional Requirements

Functional requirements define the fundamental actions that system must perform.

The functional requirements for the system are divided into three main categories, Reservation/Booking, Food, and Management. For further details, refer to the use cases.

1. Reservation/Booking

- 1.1. The system shall record reservations.
- 1.2. The system shall record the customer's first name.
- 1.3. The system shall record the customer's last name.
- 1.4. The system shall record the number of occupants.
- 1.5. The system shall record the room number.
- 1.6. The system shall display the default room rate.

- 1.6.1. The system shall allow the default room rate to be changed.
- 1.6.2. The system shall require a comment to be entered, describing the reason for changing the default room rate.

- 1.7. The system shall record the customer's phone number.
- 1.8. The system shall display whether or not the room is guaranteed.
- 1.9. The system shall generate a unique confirmation number for each reservation.
- 1.10. The system shall automatically cancel non-guaranteed reservations if the customer has not provided their credit card number by 6:00 pm on the check-in date.
- 1.11. The system shall record the expected check-in date and time.
- 1.12. The system shall record the expected checkout date and time.
- 1.13. The system shall check-in customers.
- 1.14. The system shall allow reservations to be modified without having to reenter all the customer information.
- 1.15. The system shall checkout customers.

- 1.15.1. The system shall display the amount owed by the customer.
- 1.15.2. To retrieve customer information the last name or room number shall be used
- 1.15.3. The system shall record that the room is empty.
- 1.15.4. The system shall record the payment.
- 1.15.5. The system shall record the payment type.

- 1.16. The system shall charge the customer for an extra night if they checkout after 11:00 a.m.
- 1.17. The system shall mark guaranteed rooms as "must pay" after 6:00 pm on the check-in date.
- 1.18. The system shall record customer feedback.
- 1.19. The system shall encrypt the customers' data and makes it appear in an encrypted form in the database.

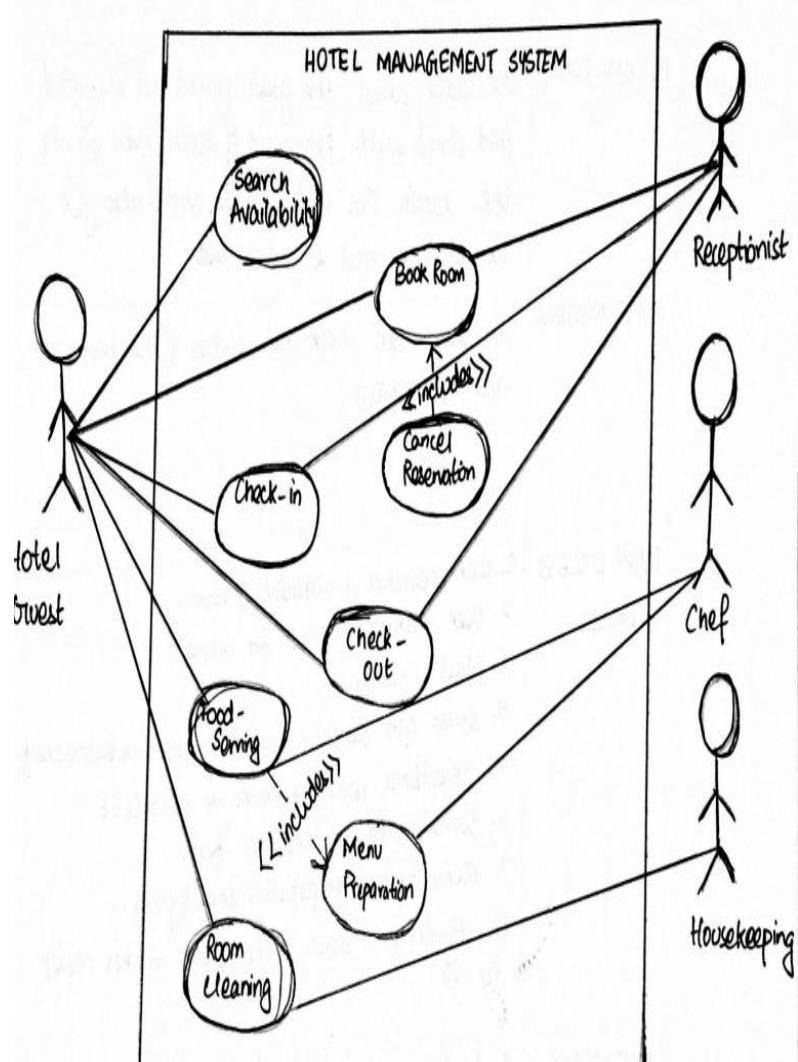
2. *Food*

- 2.1. The system shall track all meals purchased in the hotel (restaurant and room service).
- 2.2. The system shall record payment and payment type for meals.
- 2.3. The system shall bill the current room if payment is not made at time of service.
- 2.4. The system shall accept reservations for the restaurant and room service.

3. *Management*

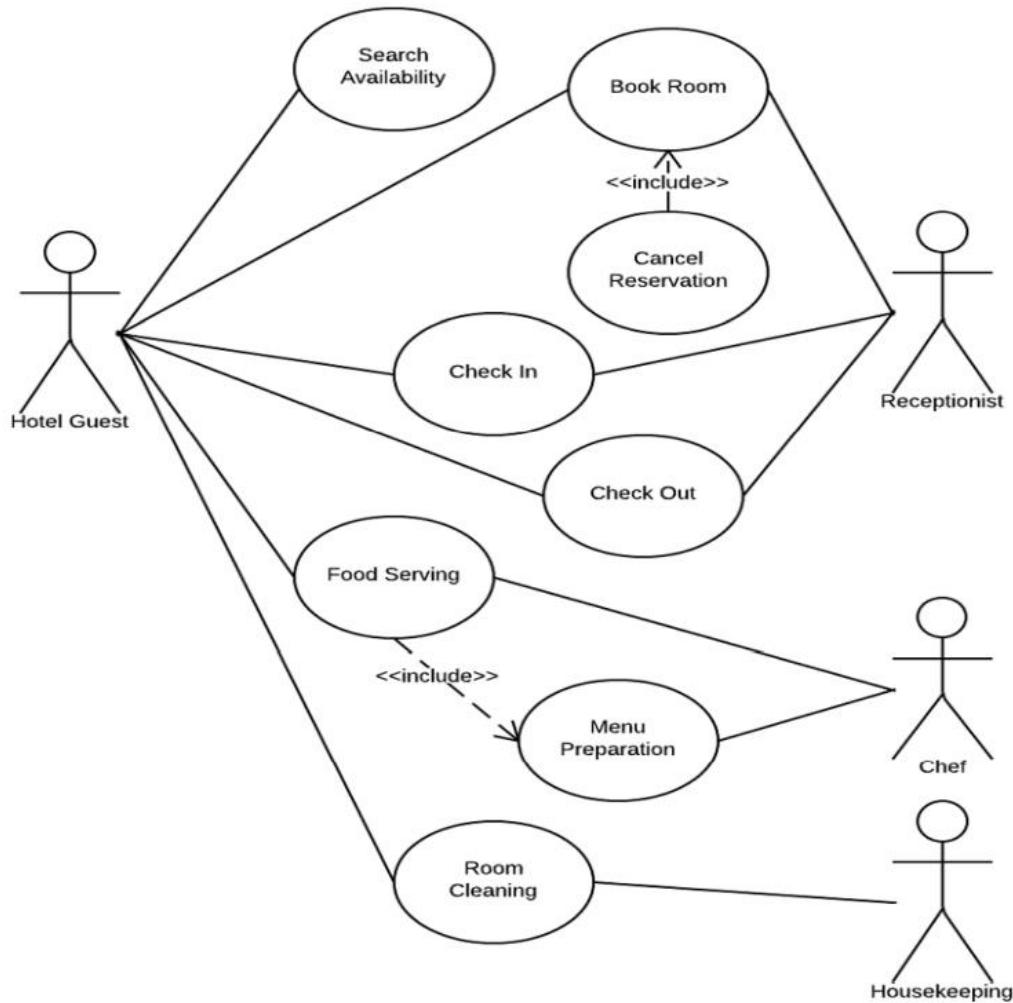
- 3.1. The system shall display the hotel occupancy for a specified period of time (days; including past, present, and future dates).
- 3.2. The system shall display projected occupancy for a period of time (days).
- 3.3. The system shall display room revenue for a specified period of time (days).
- 3.4. The system shall display food revenue for a specified period of time (days).
- 3.5. The system shall display an exception report, showing where default room and food prices have been overridden.
- 3.6. The system shall allow for the addition of information, regarding rooms, rates, menu items, prices, and user profiles.
- 3.7. The system shall allow for the deletion of information, regarding rooms, rates, menu items, prices, and user profiles.
- 3.8. The system shall allow for the modification of information, regarding rooms, rates, menu items, prices, and user profiles.
- 3.9. The system shall allow managers to assign user passwords.

Use Case Model
1. U1



Use Case Diagram for entire System

U1



Author – Malla Jyotsna, Preetham Lekkala

Purpose – Use case of Hotel Management System using Database Encryption.

Requirements Traceability – Admin required to conform bookings.

Priority - High

Preconditions – The data about the guest should get encrypted and permission of the access should be given only to be the Admin. The staff details should also

Post conditions – The Admin can conform the booking and the status is updated to the guest.

Actors – Guests, Receptionist, Staff, Housekeeping, Chef

Extends – None

Flow of Events

1. User searches availability of rooms.
2. User makes payment for room.
3. Receptionist conforms room-booking.
4. Guest checks-in.
5. Guest can make request for cancellation before check-in.
6. Guest makes request for meals and housekeeping.
7. Menu and food prepared by check.
8. Housekeeping done on request from user.

Other Non-functional Requirements

Performance Requirements

Performance requirements define acceptable response times for system functionality.

- The load time for user interface screens shall take no longer than two seconds.
- The log in information shall be verified within five seconds.
- Queries shall return results within five seconds.

Safety and Security Requirements

The system installs security measures by providing different access levels to various staff ensuring that the data stored is encrypted using Symmetric Key Encryption in the database and cannot be therefore viewed by anyone except the admin.

Customer Service Representatives and Managers will be able to log in to the Hotel Management System. Customer Service Representatives will have access to the Reservation/Booking and Food subsystems. Managers will have access to the Management subsystem as well as the Reservation/Booking and Food subsystems. Access to the various subsystems will be protected by a user log in screen that requires a user name and password.

Software Quality Attributes

Reliability

Specify the factors required to establish the required reliability of the software system at time of delivery.

Availability

The system shall be available during normal hotel operating hours.

Standards Compliance

There shall be consistency in variable names within the system. The graphical user interface shall have a consistent look and feel.

Maintainability

The Hotel Management System is being developed in Java. Java is an object oriented programming language and shall be easy to maintain.

Portability

The Hotel Management System shall run in any Microsoft Windows environment that contains Java Runtime and the Microsoft Access database.

Other Requirements

1.1.1 Logical Database Requirements

The logical database requirements include the retention of the following data elements. This list is not a complete list and is designed as a starting point for development.

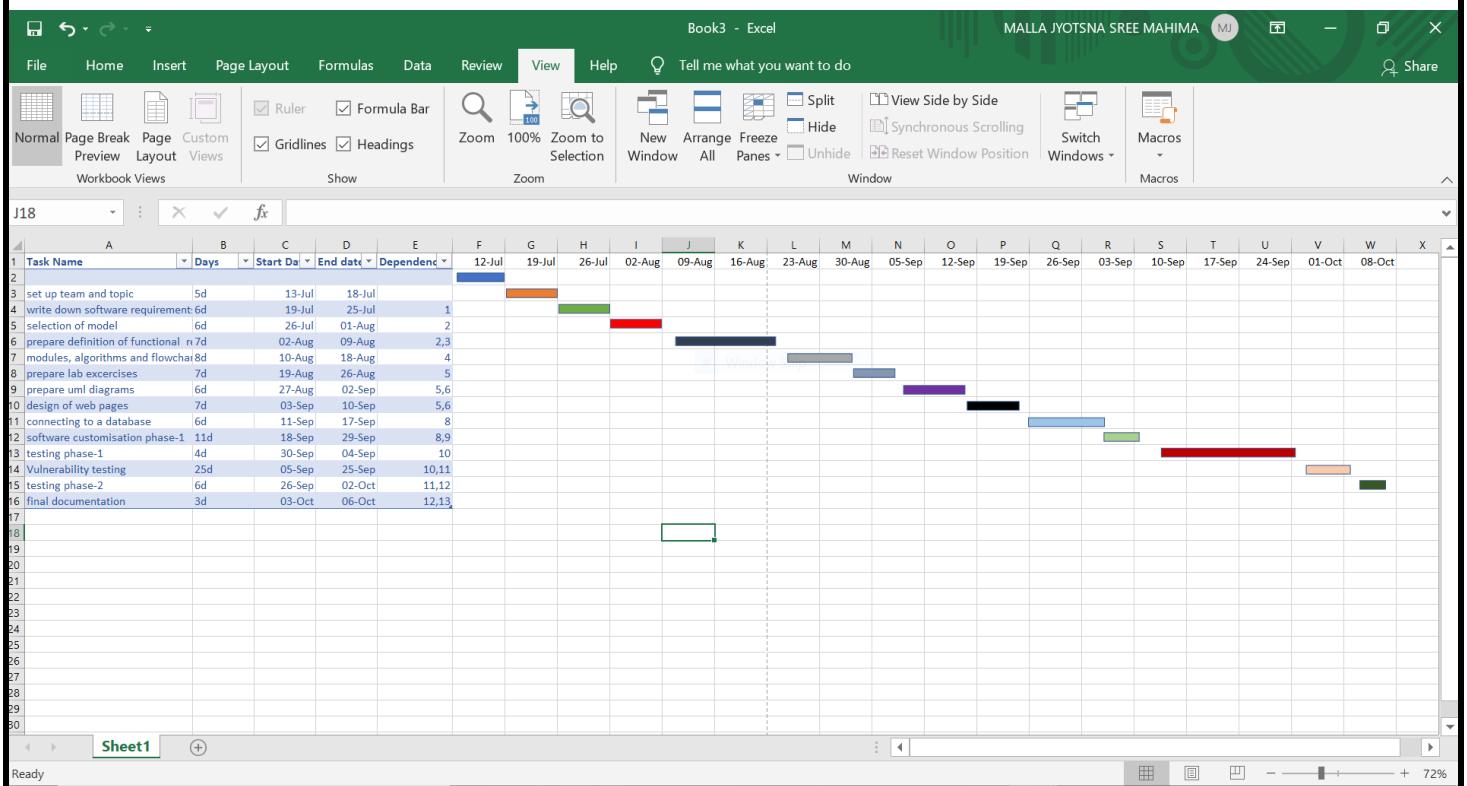
Booking/Reservation System

- Customer first name
- Customer last name
- Customer address
- Customer phone number
- Number of occupants
- Assigned room
- Default room rate
- Rate description
- Guaranteed room (yes/no)
- Credit card number
- Confirmation number
- Automatic cancellation date
- Expected check-in date
- Expected check-in time
- Actual check-in date
- Actual check-in time
- Expected check-out date
- Expected check-out time
- Actual check-out date
- Actual check-out time
- Customer feedback
- Payment received (yes/no)
- Payment type
- Total Bill

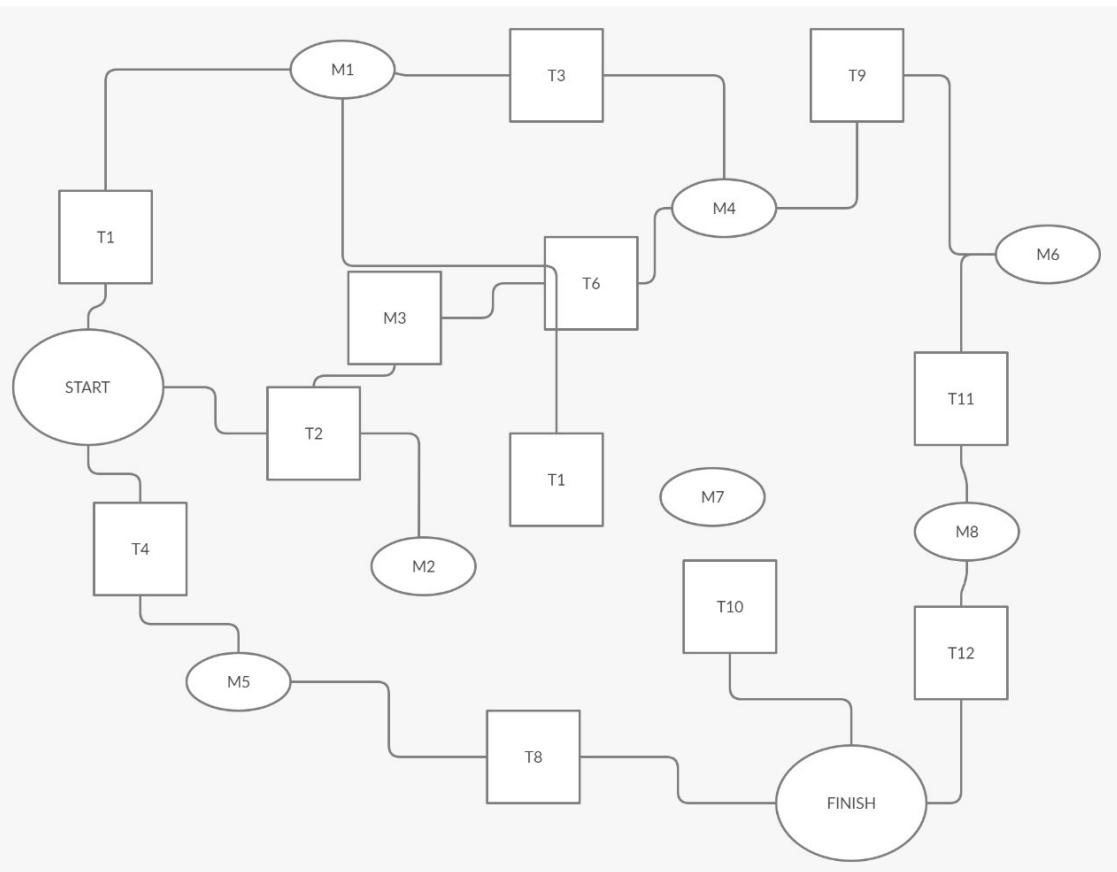
Food Services

- Meal
- Meal type
- Meal item
- Meal order
- Meal payment (Bill to room/Credit/Check/Cash)

iii. GANTT CHART:



PERT CHART :



T1: Picking a topic: 8 days

T2: Making the problem statement: 15 days

T3:WBS; 8 days

T4:SRS: 10 days

T5:PERT AND GANTT:10 days

T6:DFD AND ER: 5 days

T7:UML DIAGRAMS: 20 days

T8: IMPLEMENTATION:28 days

T9: DATABASE CONNECTION:15 days

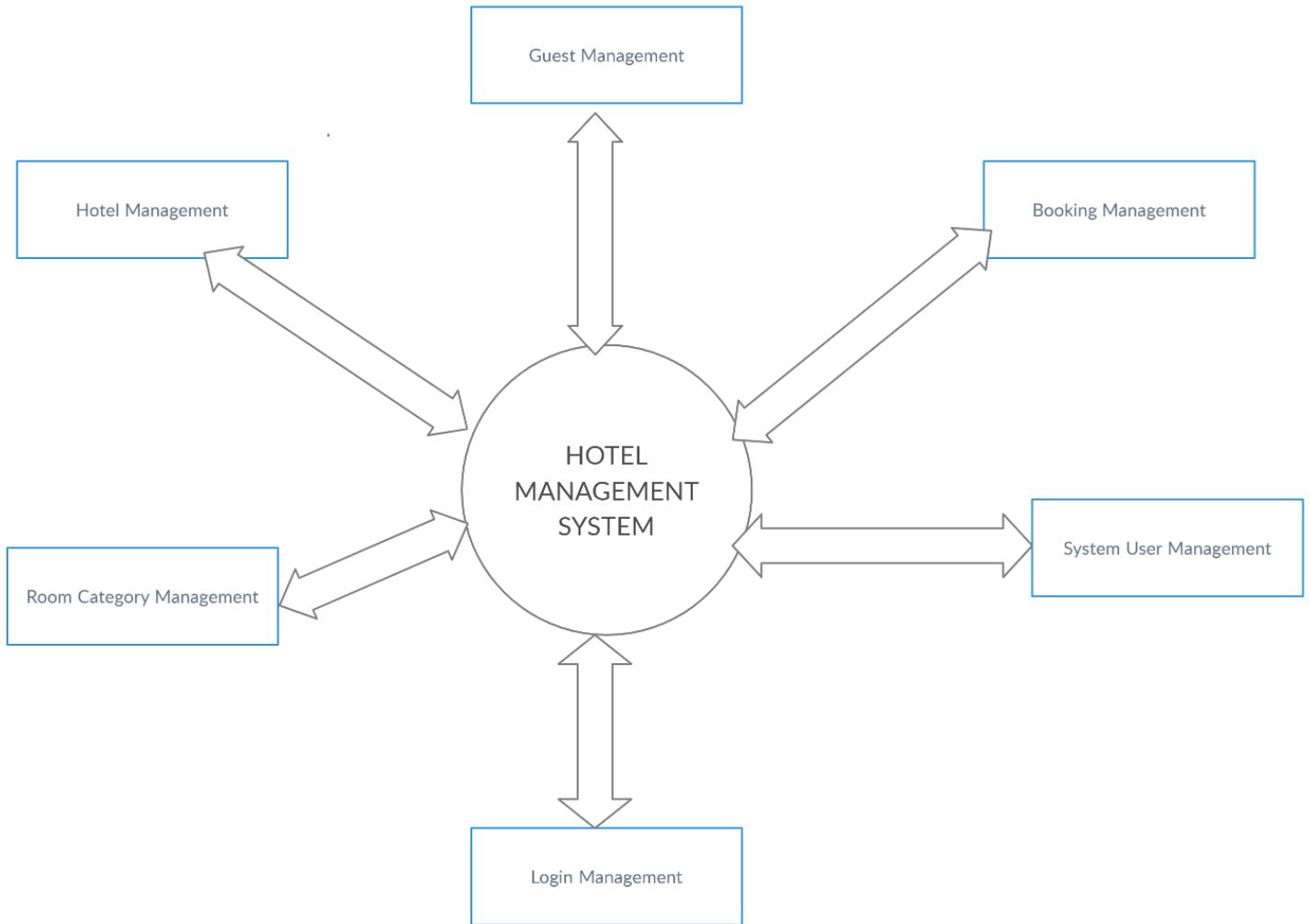
T10:ENCRYPTION: 15 days

T11:DECRYPTION:7days

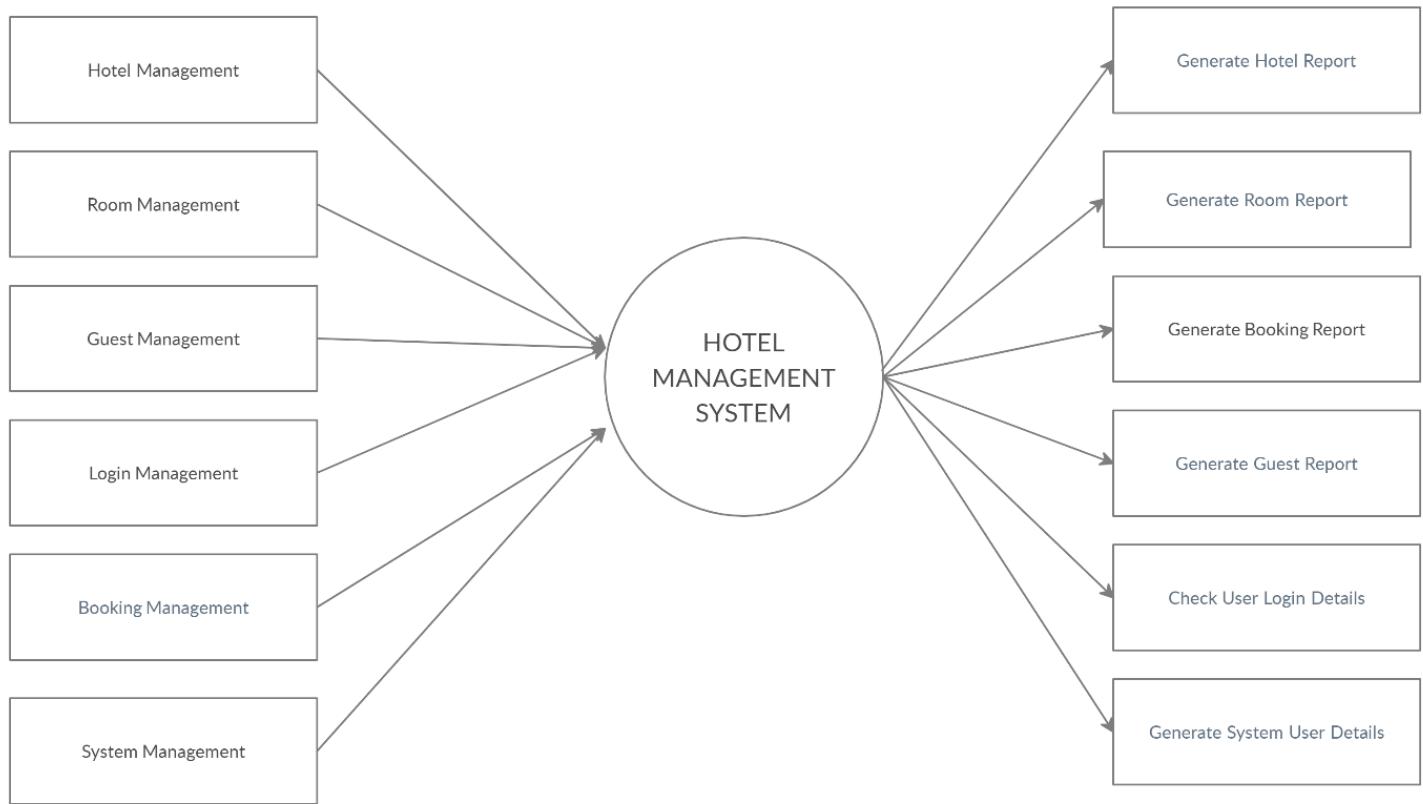
T12:TESTING: 10 days

iv.DFD & ER :

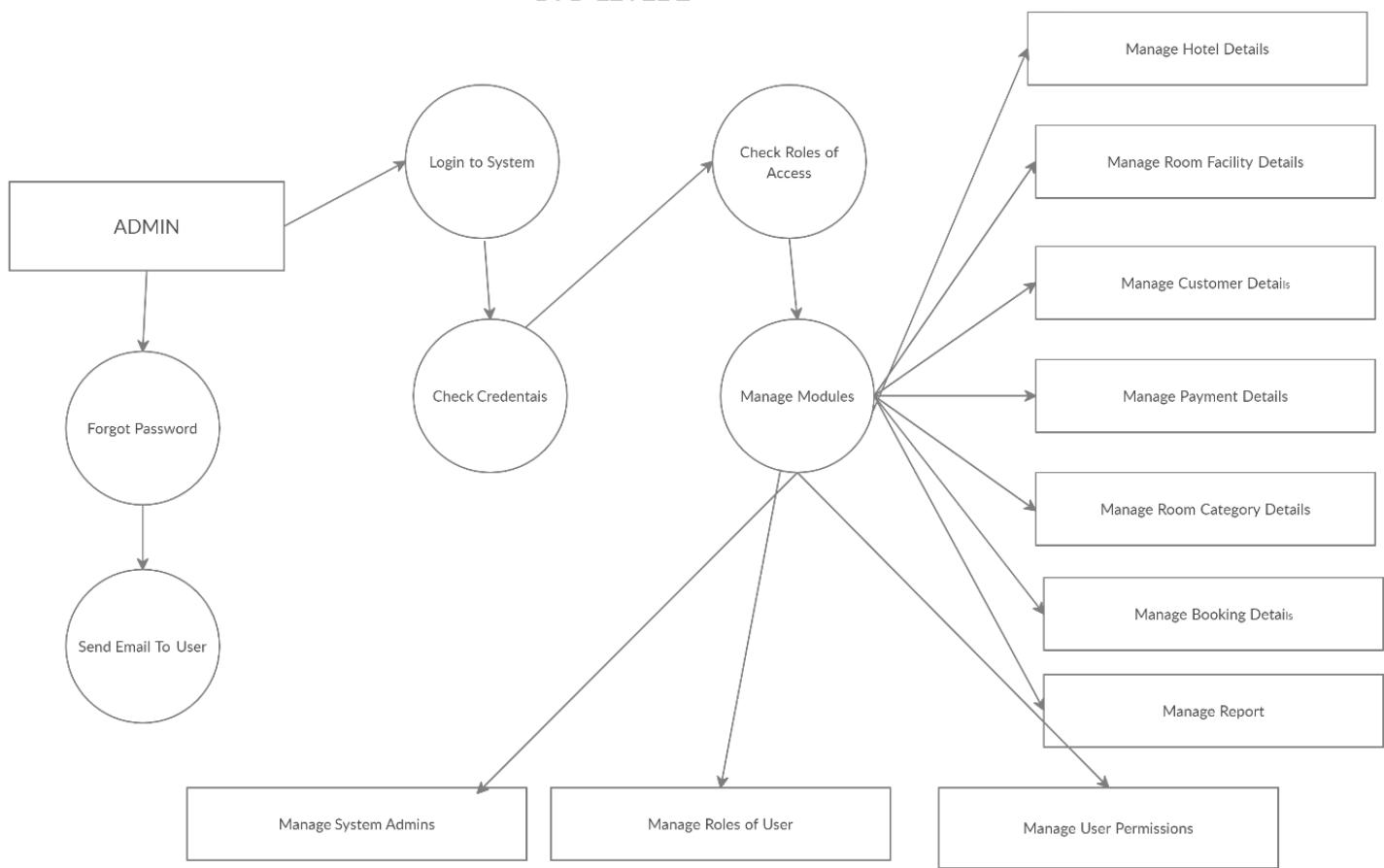
DFD LEVEL 0



DFD LEVEL 1

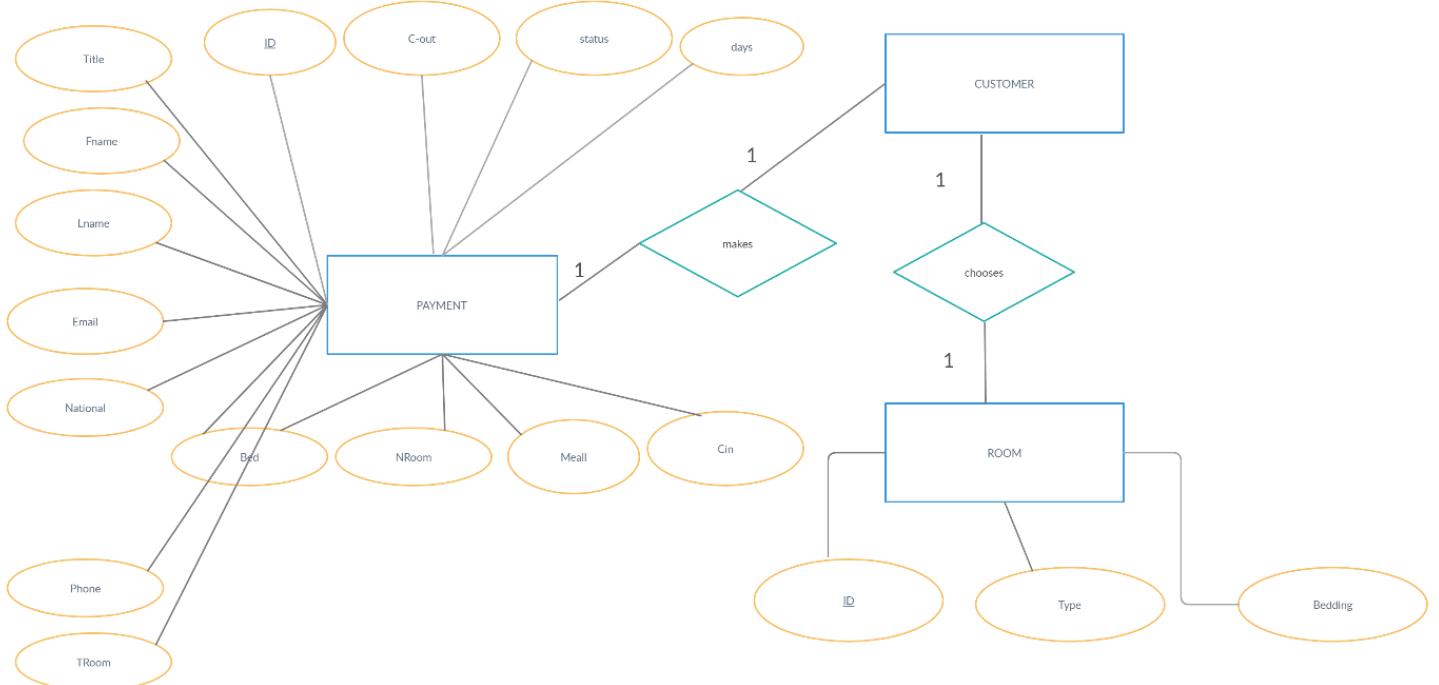


DFD LEVEL 2



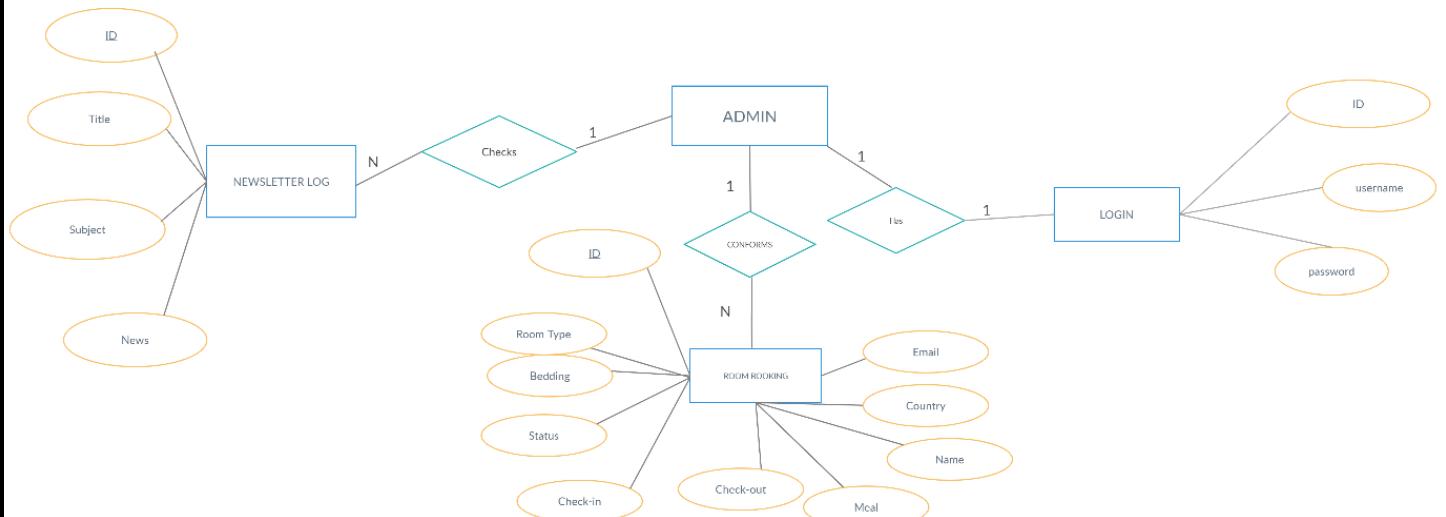
ER diagram for Guest:

ER DIAGRAM FOR GUEST



ER diagram for Admin:

E-R DIAGRAM FOR ADMIN



6.TEST CASES(Hotel Management System using Database Encryption):

Test Case ID	T01		
Priority	High		
Description	To test the Admin login component		
Module	Login		
Prepared By	Malla Jyotsna	Date Prepared	02 – Oct – 2020
Reviewed / Updated	Malla Jyotsna	Date Reviewed	02 – Oct – 2020
Tested By	Malla Jyotsna	Date Tested	02 – Oct – 2020
Test Cases			
Sl. No.	Class	Input Condition	Expected Result
1	C1 – User login with valid data	Username – Admin Password – 1234	User should login into the application
2	C2 – User login with invalid data	Username – blabla Password – BlaI@123	User should not login into the application, and
			an error message should appear.
3	C3 – User login with empty data.	Username – Password –	User should not login into the application, and an error message should appear.
Test Case Result		Pass	

Post-conditions:

Admin is validated with database and successfully login to account. The account session details are logged in database.

Test Case ID	T02			
Priority	High			
Description	To test the Successful Room Reservation			
Module	Room Reservation			
Prepared By	Preetham Lekkala	Date Prepared	03 – Oct – 2020	
Reviewed / Updated	Preetham Lekkala	Date Reviewed	03 – Oct – 2020	
Tested By	Preetham Lekkala	Date Tested	03 – Oct – 2020	
Test Cases				
Sl. No.	Class	Input Condition	Expected Result	Actual Result
1	C1 – Customer uploads all the details along with correct verification code and submits.	All details and correct verification code.	Customer should be informed about the successful booking.	Alert message “Your Booking Application has been sent”.
2	C2 – User Uploads details with wrong verification code and press submit button.	All details and incorrect verification code.	User has not uploaded the correct code and an error message is displayed	Alert prompt “Invalid Human Verification code”
3	C3 – User Uploads details with empty verification code and press submit button.	All details and empty verification code.	User has not uploaded the code and an error message is displayed	Alert prompt displayed “Enter the verification code”.
Test Case Result			Pass	

Post-conditions:

The customer application for room booking is sent to the Admin and stored in the database.

Test Case ID	T03			
Priority	High			
Description	To test the Successful Room Conformation by Admin			
Module	Room Reservation			
Prepared By	Malla Jyotsna	Date Prepared	04 – Oct – 2020	
Reviewed / Updated	Malla Jyotsna	Date Reviewed	04 – Oct – 2020	
Tested By	Malla Jyotsna	Date Tested	04 – Oct – 2020	
Test Cases				
Sl. No.	Class	Input Condition	Expected Result	Actual Result
1	C1 – Admin clicks on conform and the room is booked.	Click on the conform by the Admin	Message should be displayed to the Admin stating	Alert displayed to admin “Room is booked”.
			their room is booked.	
2	C2 – Admin clicks on the delete option in status and the room booking application is deleted.	Admin clicks on delete against the room booking application	Message should be displayed to the Admin stating that the room booking application has been deleted.	Alert prompted “Booking application Deleted”
Test Case Result			Pass	

Post-conditions:

The admin conforms the room and the room gets allocated to the applicant in the database.

Test Case ID	T04		
Priority	High		
Description	To test the Add and delete Rooms working		
Module	Add/Delete Rooms		
Prepared By	Preetham Lekkala	Date Prepared	02 – Oct – 2020
Reviewed / Updated	Preetham Lekkala	Date Reviewed	02 – Oct – 2020
Tested By	Preetham Lekkala	Date Tested	02 – Oct – 2020

Test Cases

Sl. No.	Class	Input Condition	Expected Result	Actual Result
1	C1 – Admin adds the room which already is vacant.	Type of room, Bedding type	Admin should be showed a message about the room already existing.	Alert Message: “Room already exists” displayed to the Admin.
2	C2 – Admin adds the room which is not vacant.	Type of room , Bedding type	A message should be displayed to indicate the updation of the new room in the database.	Alert Message “New Room Added to the database” shown to the Admin.
3	C3 – Admin deletes a room against the room ID	Room ID	A message should be displayed that the room has been deleted.	Alert message “Room deleted successfully” displayed to the Admin.

Test Case Result	Pass
-------------------------	------

Post-conditions:

The admin adds and deleted room and the status of the rooms available gets updated in the database and is reflected in the admins' account.

Test Case ID	T05		
Priority	High		
Description	To test the Encryption of the Sensitive Info of the Customers		
Module	Encryption		
Prepared By	Malla Jyotsna	Date Prepared	02 – Oct – 2020
Reviewed / Updated	Malla Jyotsna	Date Reviewed	02 – Oct – 2020
Tested By	Malla Jyotsna	Date Tested	02 – Oct – 2020

Test Cases

Sl. No.	Class	Input Condition	Expected Result	Actual Result
1	C1 – Admin sends a booking application.	Customer Information	The Customers; sensitive data should be stored in the database in an encrypted form.	The customer details gets stored in the roombook database in an encrypted manner.
2	C2 – Admin conforms the booking application .	Clicks on the conform button.	The sensitive data of the customer should get encrypted.	Once the booking is conformed the sensitive info again gets encrypted.
Test Case Result			Pass	

Post-conditions:

The sensitive info is encrypted before and after the conformation in the database to ensure security of information of customers

Test Case ID	T05		
Priority	High		
Description	To test the Decryption of the Sensitive Info		
Module	Decryption		
Prepared By	Preetham Lekkala	Date Prepared	02 – Oct – 2020
Reviewed / Updated	Preetham Lekkala	Date Reviewed	02 – Oct – 2020
Tested By	Preetham Lekkala	Date Tested	02 – Oct – 2020

Test Cases

Sl. No.	Class	Input Condition	Expected Result	Actual Result
1	C1 – Admin logs into his account and goes to the status page.	Clicks on Status page.	The decrypted information should be available to the user.	The decrypted information of the booking applications to be conformed are displayed to the Admin.
2	C2 – Admin conforms the booking application .	Clicks on the conform button.	The sensitive data of the customer should get encrypted.	Once the booking is conformed the sensitive info again gets encrypted.
Test Case Result			Pass	

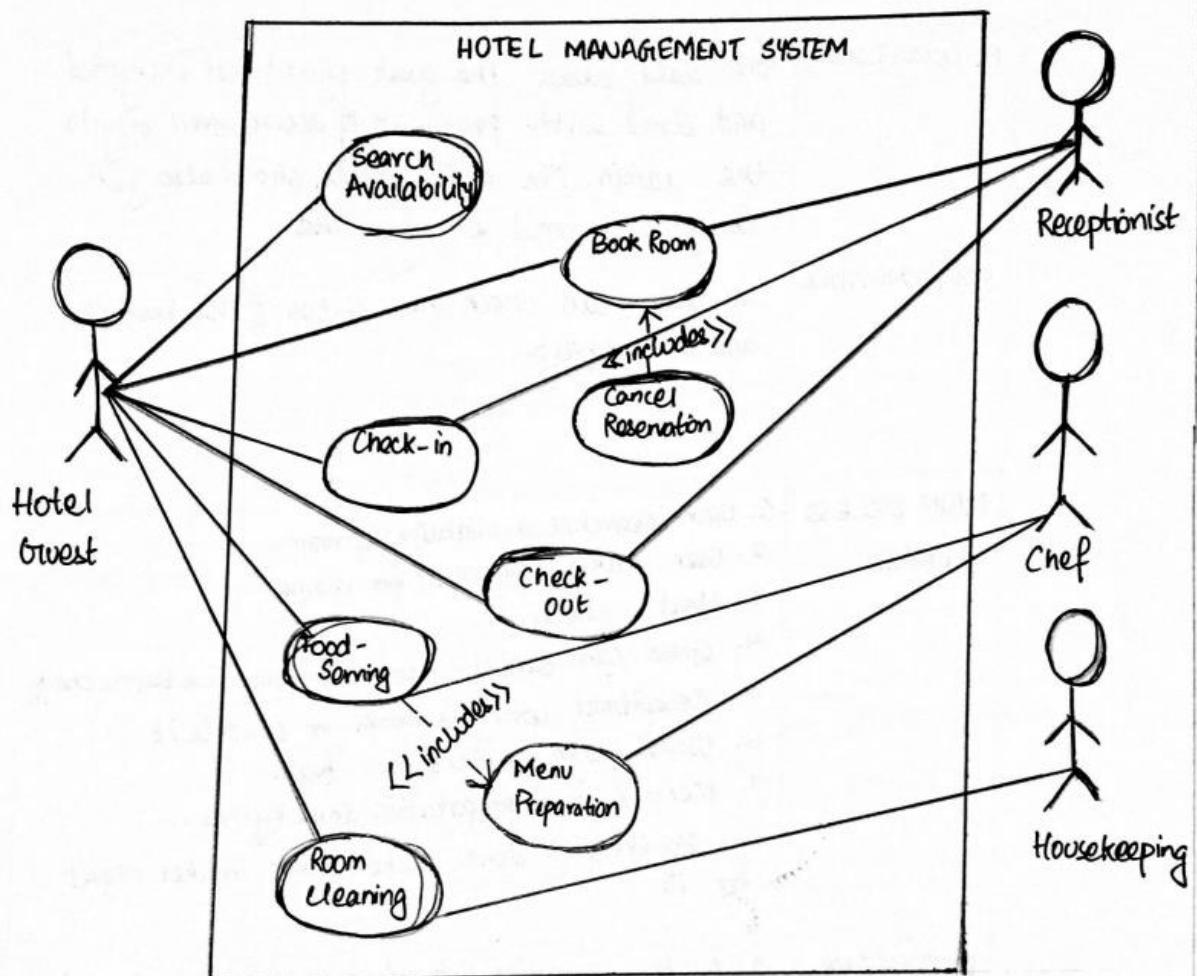
Post-conditions:

The admin can see the decrypted information about the customers' only during the conformation time.

5 .UML DIAGRAMS:

1.USE CASE DIAGRAM:

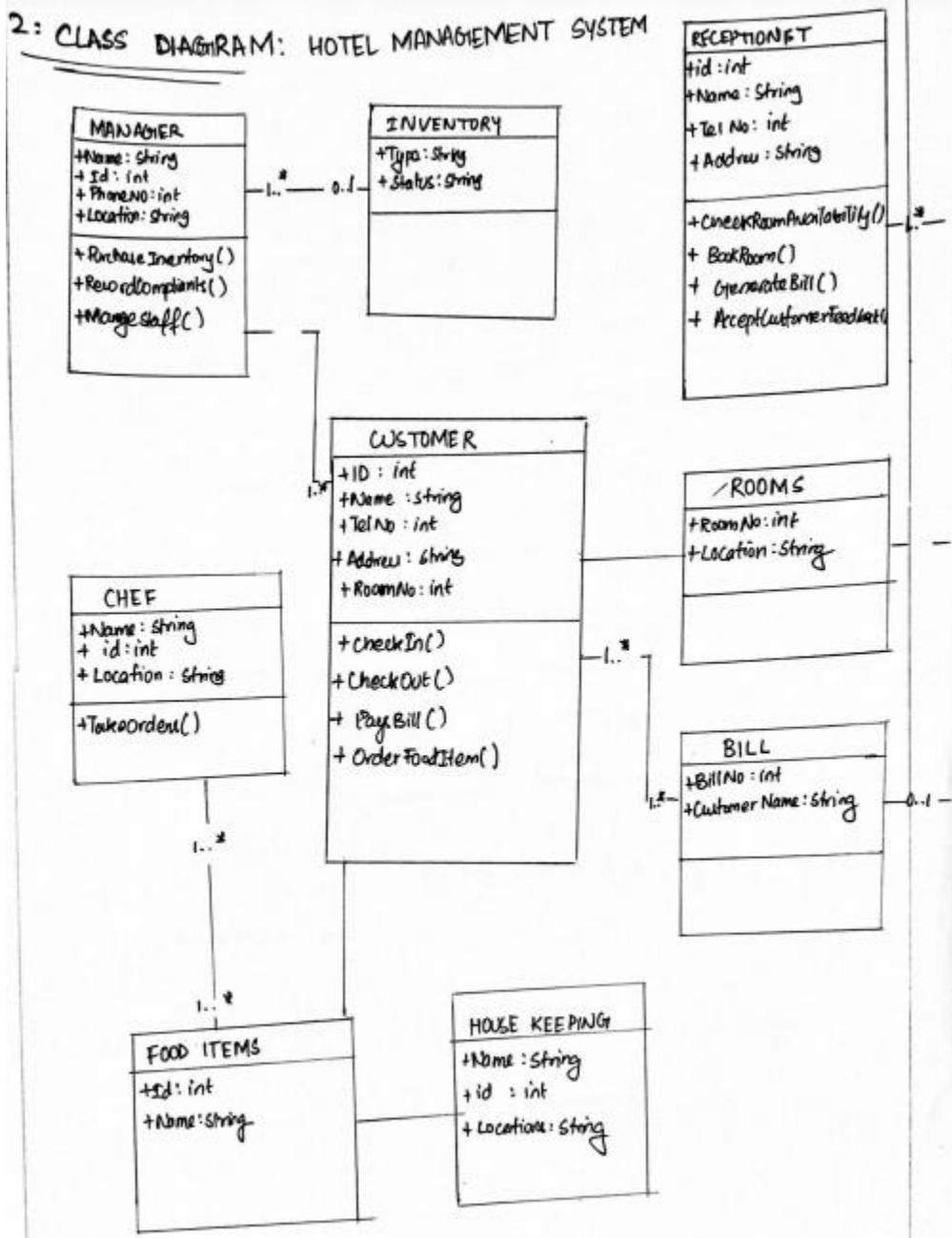
1: USE CASE DIAGRAM:



USE CASE DESCRIPTION IN TABULAR FORMAT

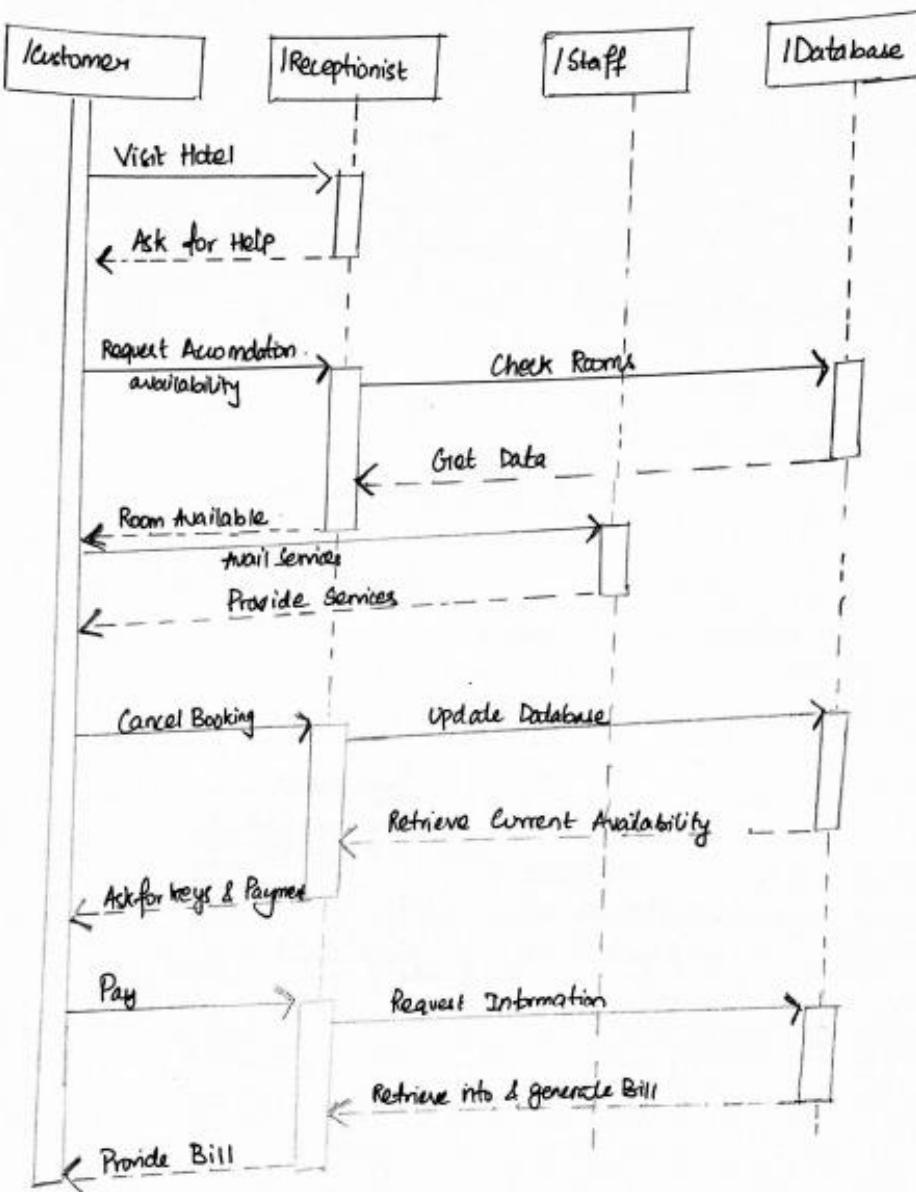
USE CASE TITLE	HOTEL MANAGEMENT SYSTEM
DESCRIPTION	A guest can search availability of desired rooms, check-in and checkout, make payments, make food orders and make requests for room cleaning.
ACTOR(S)	Hotel Guest, Receptionist, Chef, Housekeeping.
PRECONDITIONS	The data about the guest should get encrypted and stored with permission of access given only to the admin. The staff details should also get encrypted & stored for admin use.
POST CONDITIONS	The user can check the status of his booking and his requests.
MAIN SUCCESS SCENARIO	<ol style="list-style-type: none"> 1. User searches availability of room. 2. User makes payment for room. 3. Guest checks-in. 4. Guest can cancel reservation anytime before check-in. 5. Receptionist confirms book or cancels it. 6. Users make orders for food. 7. Menu & food preparation done by chef. 8. Housekeeping done when guest makes request for it.
EXTENSIONS	<ol style="list-style-type: none"> 1. All the sensitive info of customer gets stored in an encrypted format which can only be accessed by the admin. 2. Admin can remove guests, staff etc.

2. CLASS DIAGRAM:



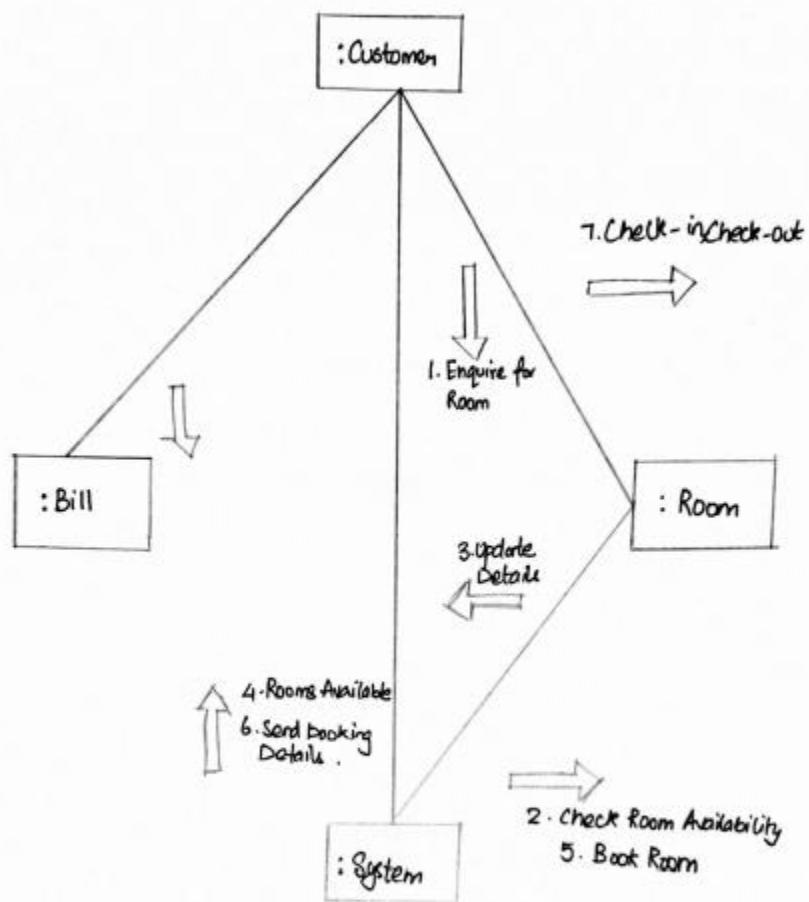
3. SEQUENCE DIAGRAM:

3: SEQUENCE DIAGRAM: HOTEL MANAGEMENT SYSTEM



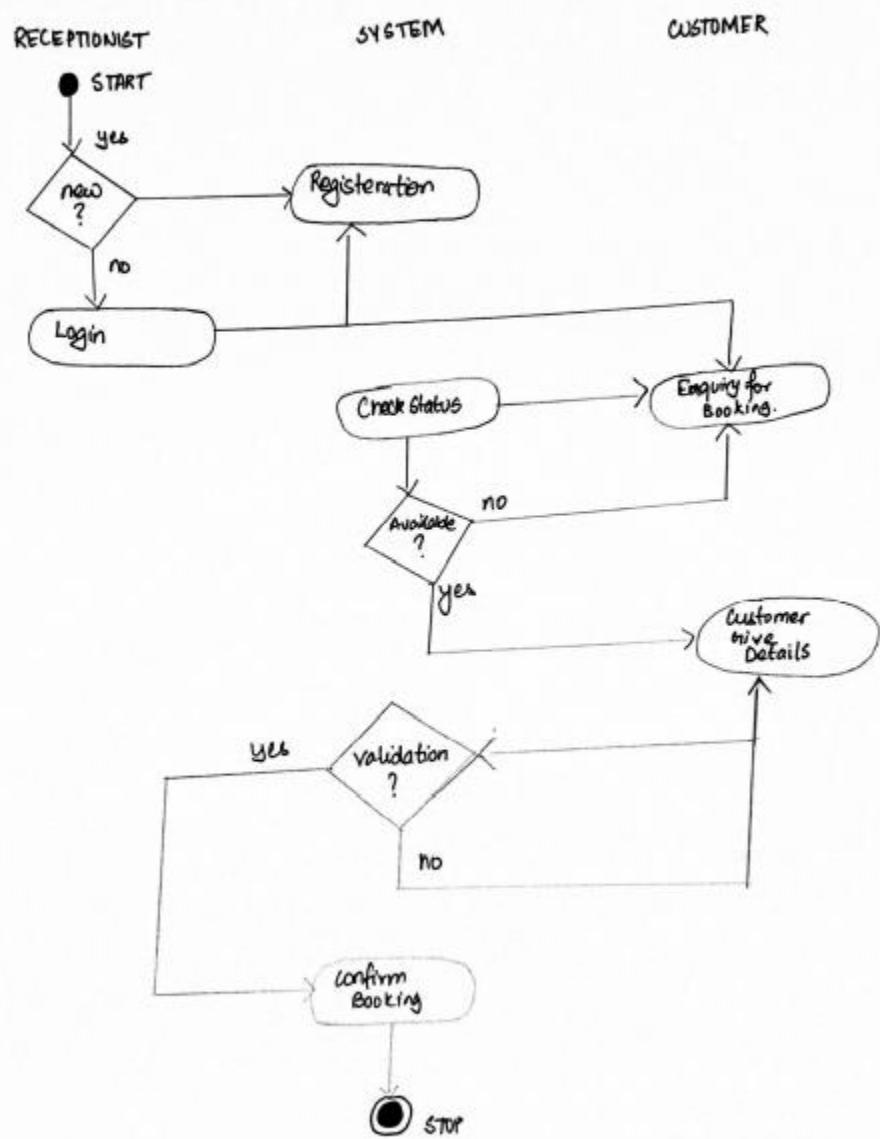
4. COLLABORATION DIAGRAM:

4: COLLABORATION DIAGRAM: HOTEL MANAGEMENT SYSTEM



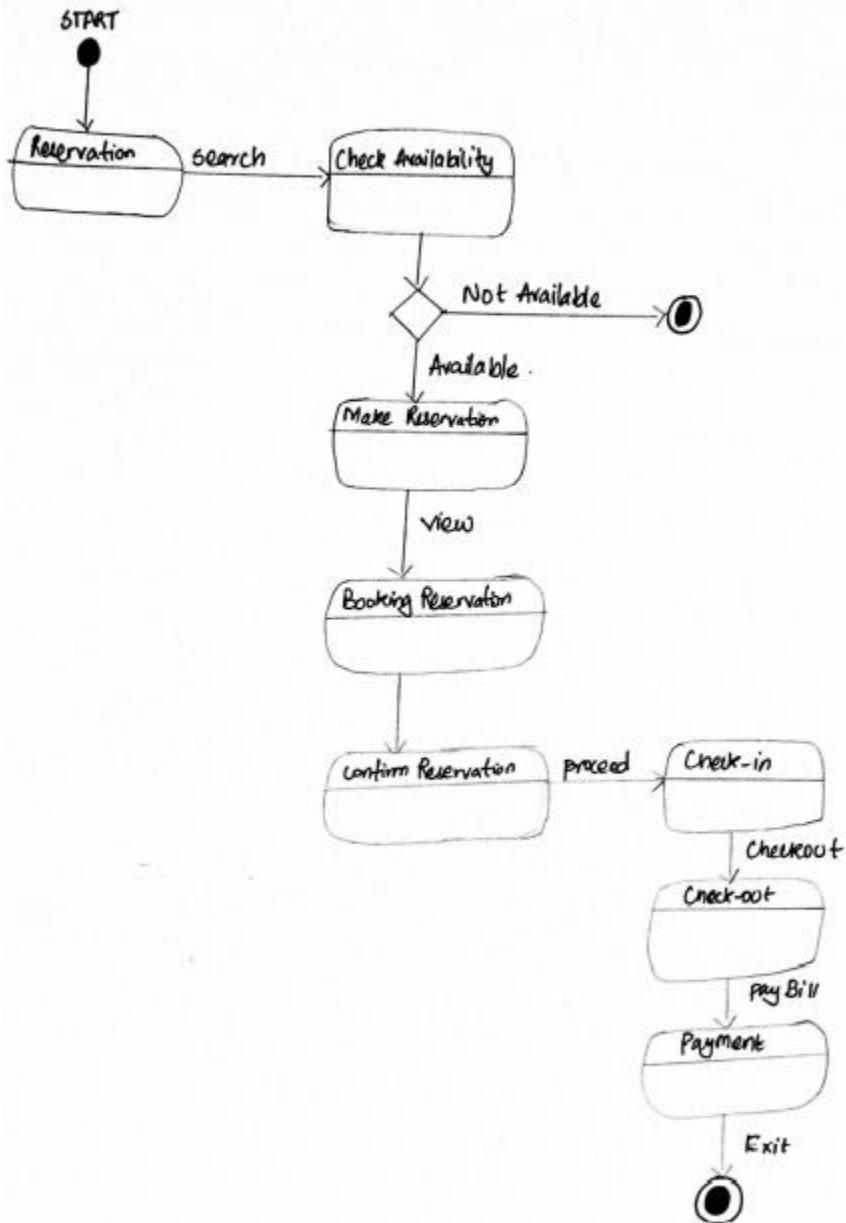
5.ACTIVITY DIAGRAM:

5. ACTIVITY DIAGRAM: HOTEL MANAGEMENT SYSTEM

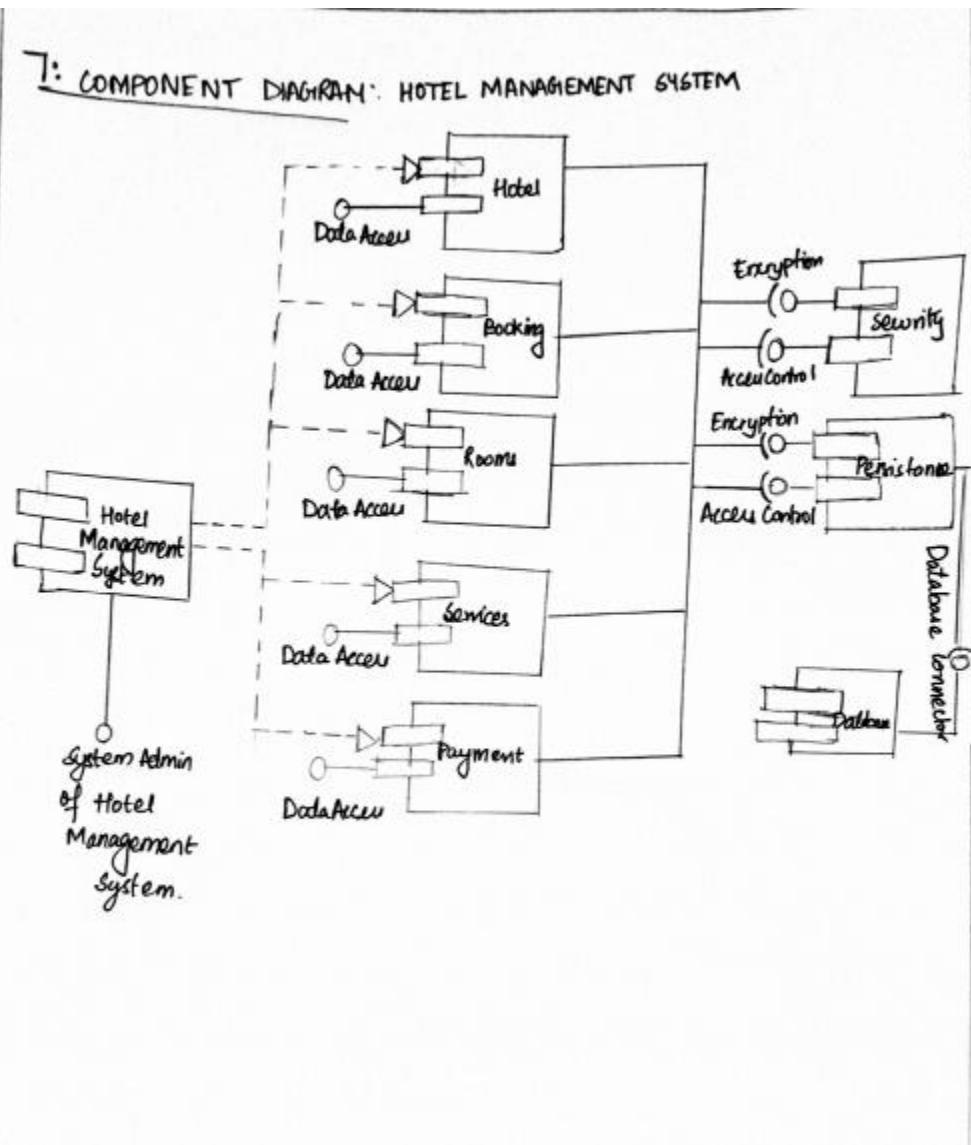


6.STATE DIAGRAM:

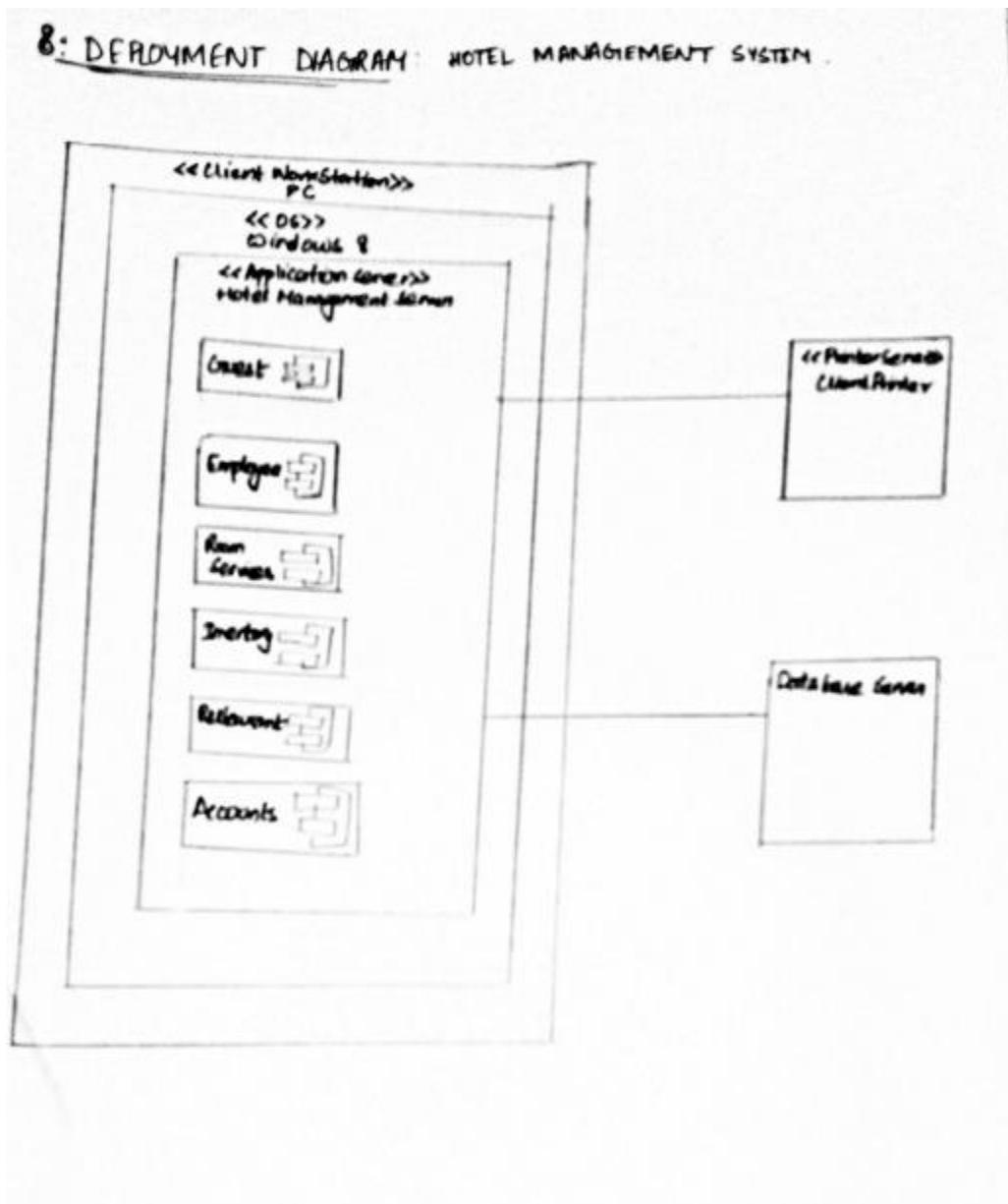
6: STATE DIAGRAM: HOTEL MANAGEMENT SYSTEM



7. COMPONENT DIAGRAM:



8. DEPLOYMENT DIAGRAM:



6.CODE:

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;
}
}

?>
```

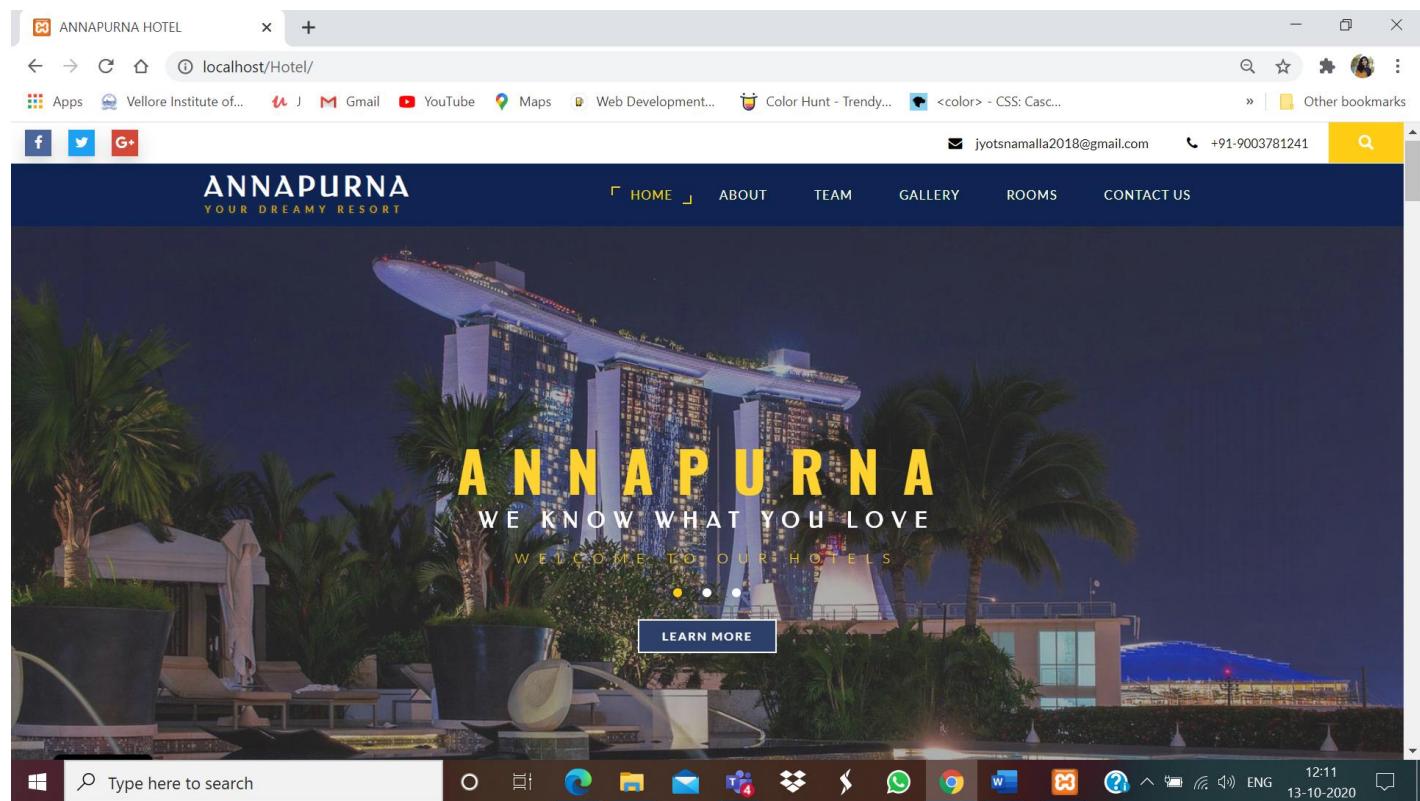
7.RESULTS AND DISCUSSION:

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 application level encryption to make sure 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 kind 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.

8.EXECUTION SCREENSHOTS:

HOME PAGE



ANAPURNA HOTEL x +

localhost/Hotel/

Apps Vellore Institute of... J Gmail YouTube Maps Web Development... Color Hunt - Trendy... <color> - CSS: Casc...

About Our ANAPURNA

Cookie Policy

ANAPURNA HOTEL x +

localhost/Hotel/

Apps Vellore Institute of... J Gmail YouTube Maps Web Development... Color Hunt - Trendy... <color> - CSS: Casc...



ANAPURNA HOTEL x +

localhost/Hotel/

Apps Vellore Institute of... J Gmail YouTube Maps Web Development... Color Hunt - Trendy... <color> - CSS: Casc...

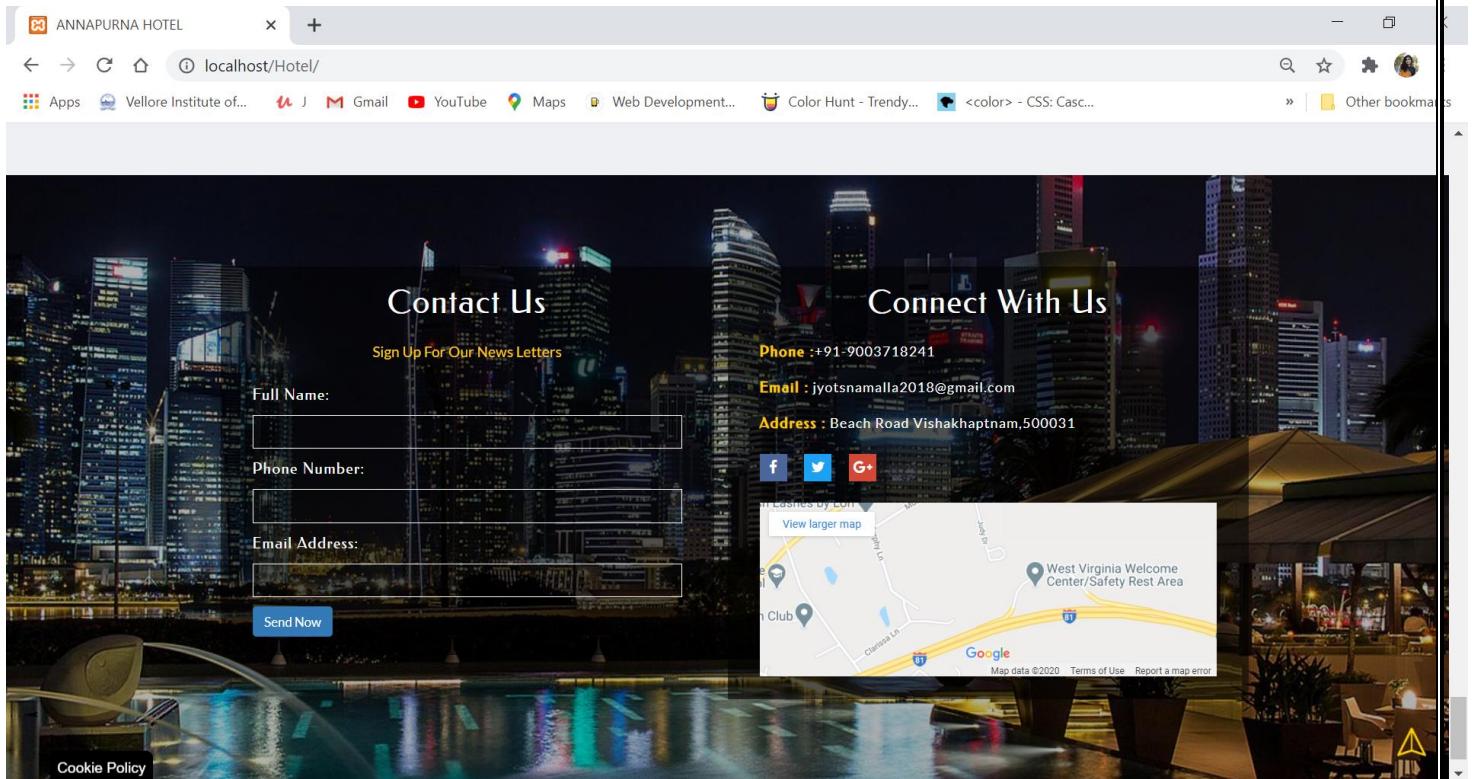
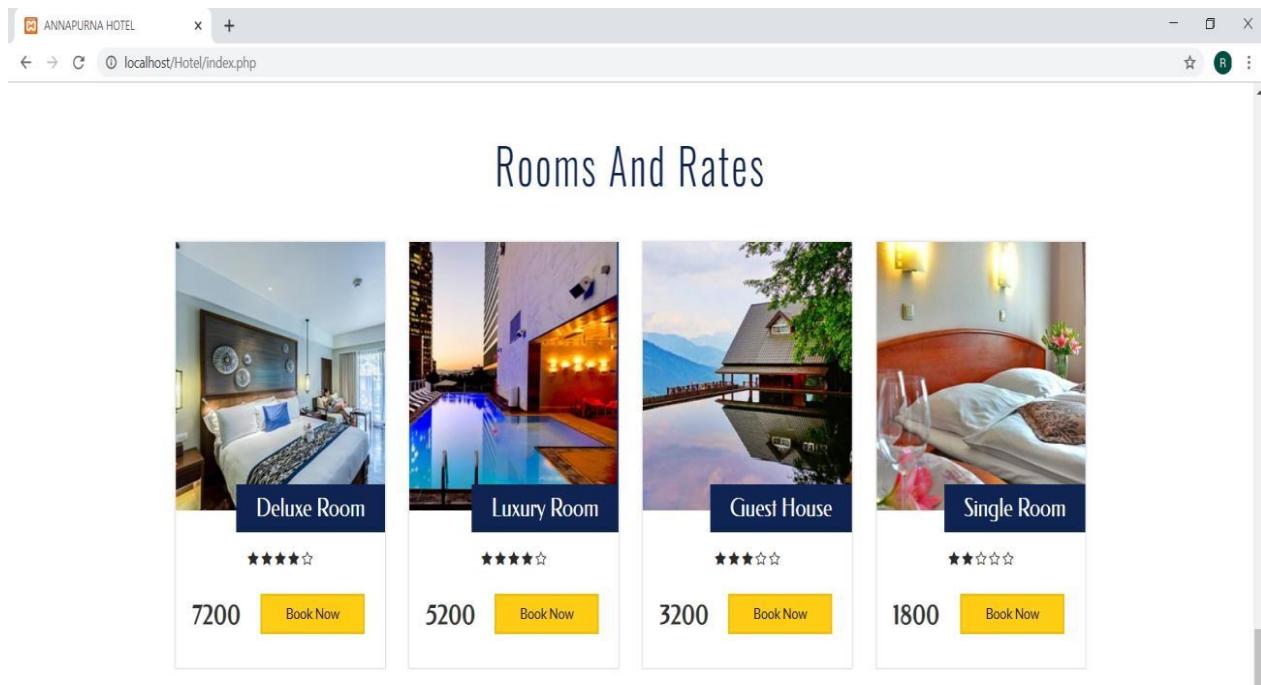
Meet Our Team



Malla Jyotsna
MANAGER

Welcome to HOTEL ANNAPURNA. We promise to provide you a pleasant and wonderful experience. Have a good day! .

f t G+ r



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

The screenshot shows a web browser window titled "RESERVATION ANnapurna HO". The URL is "localhost/Hotel/admin/reservation.php". The page has a dark blue header with the word "RESERVATION" in white. Below the header, there are two main sections: "PERSONAL INFORMATION" and "RESERVATION INFORMATION".

PERSONAL INFORMATION:

- Title* (dropdown menu)
- First Name (text input)
- Last Name (text input)
- Email (text input)
- Nationality* (radio buttons: Indian (checked) and Non Indian)
- Passport Country* (dropdown menu)
- Phone Number (text input)

RESERVATION INFORMATION:

- Type Of Room * (dropdown menu)
- Bedding Type (dropdown menu)
- No.of Rooms * (text input)
- Meal Plan (dropdown menu)
- Check-In (text input: mm/dd/yyyy)
- Check-Out (text input: mm/dd/yyyy)

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

ADMIN PAGE

The screenshot shows a web browser window titled "ANnapurna ADMIN". The URL is "localhost/Hotel/admin/index.php". The page features a light blue background with white clouds. In the center, there are two input fields: "Username" and "Password" (with a visibility icon). Below these is a pink "LOGIN" button. At the bottom of the page, the text "ANNAPURNA HOMEPAGE" is displayed in white on a dark background.

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

The screenshot shows the phpMyAdmin interface for the 'hotel' database. The left sidebar lists various databases and their structures. The main area displays the tables within the 'hotel' database, including 'contact', 'login', 'newsletterlog', 'payment', 'room', and 'roombook'. A table summary at the bottom indicates there are 52 rows and 6 tables.

Table	Action	Rows	Type	Collation	Size	Overhead
contact	Browse Structure Search Insert Empty Drop	25	InnoDB	latin1_swedish_ci	16.0 Kib	-
login	Browse Structure Search Insert Empty Drop	2	InnoDB	latin1_swedish_ci	16.0 Kib	-
newsletterlog	Browse Structure Search Insert Empty Drop	2	InnoDB	latin1_swedish_ci	16.0 Kib	-
payment	Browse Structure Search Insert Empty Drop	4	InnoDB	latin1_swedish_ci	16.0 Kib	-
room	Browse Structure Search Insert Empty Drop	15	InnoDB	latin1_swedish_ci	16.0 Kib	-
roombook	Browse Structure Search Insert Empty Drop	4	InnoDB	latin1_swedish_ci	16.0 Kib	-
6 tables	Sum	52	InnoDB	utf8mb4_general_ci	96.0 Kib	0 B

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

The screenshot shows the phpMyAdmin interface for a MySQL database named 'hotel'. The 'roombook' table is selected. The table contains the following data:

	id	Title	FName	LName	Email	National	Country	Phone	TRoom	Bed	NRoom
<input type="checkbox"/>	3	Mr.	nbuiuffsq	Iekkala	oj/db/uofevutujwA9213bnjibnffst/botupzk	objeoJ	bmmjvhOB	5051515152535353	Deluxe Room	Quad	1
<input type="checkbox"/>	5	Prof.	botupzKlbmmbn	Sree Malima	npd/mjbnhA9213bmmnbtopuzk	objeoJ	ojbsibC	4950515253545556	Guest House	Double	1
<input type="checkbox"/>	6	Mrs.	bitsbl	roy	npd/mjbnhApbsowm	objeoJ	bjofnsB	49505152535455	Deluxe Room	Triple	1
<input type="checkbox"/>	7	Mrs.	bitsbl	roy	npd/mjbnhApbsowm	objeoJ	bjofnsB	49505152535455	Single Room	Single	1

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.

/Type here/

DECRYPTED DATA

The screenshot shows a web browser window titled "ANNA PURNA HOTEL" with the URL "localhost/admin/home.php". The page has a dark blue header with the word "ADMIN" in white. On the left, there is a sidebar with links: "Status", "News Letters", "Room Booking", "Payment", and "Logout". The main content area is titled "Status Room Booking" and contains a table titled "New Room Bookings". The table has columns: #, Name, Email, Country, Room, Bedding, Meal, Check In, Check Out, Status, and More. It lists two entries:

#	Name	Email	Country	Room	Bedding	Meal	Check In	Check Out	Status	More
5	Malla Jyotsna Sree Mahima	jyotsnamalla2018@gmail.com	Bahrain	Guest House	Double	Breakfast	2020-10-27	2020-10-28	Not Conform	Action
6	Harsha roy	lvnrao@gmail.com	Armenia	Deluxe Room	Triple	Breakfast	2020-10-22	2020-10-22	Not Conform	Action

Below the table, there is a section titled "Booked Rooms" with a count of 2.

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.

AVAILABILITY OF ROOMS:

The screenshot shows a web browser window titled "ANNA PURNA HOTEL" with the URL "localhost/admin/settings.php". The page has a dark blue header with the word "MAIN MENU" in white. On the left, there is a sidebar with links: "Room Status", "Add Room", and "Delete Room". The main content area is titled "Available Rooms" and displays a grid of room icons and labels. The grid is organized into three rows and four columns. Each icon represents a room type (Single, Double, Triple, Quad) and its availability status (Superior Room or Deluxe Room). The colors of the icons and labels indicate the status: blue for Superior Room and green for Deluxe Room.

Room Type	Status
Single	Superior Room
Double	Superior Room
Triple	Superior Room
Quad	Superior Room
Single	Deluxe Room
Double	Deluxe Room
Triple	Deluxe Room
Quad	Deluxe Room

Availability of rooms is visible to the Admin.

/Type here/

ADD ROOMS:

The screenshot shows a web browser window with three tabs: 'ANAPURNA HOTEL', 'SUNRISE HOTEL', and 'localhost / 127.0.0.1 / hotel / room'. The main content area is titled 'NEW ROOM'. It contains two forms: 'ADD NEW ROOM' with fields for 'Type Of Room *' (dropdown) and 'Bedding Type' (dropdown), and a 'Rooms Information' table with 11 rows. The table columns are 'Room ID', 'Room Type', and 'Bedding'. The data is as follows:

Room ID	Room Type	Bedding
1	Superior Room	Single
2	Superior Room	Double
3	Superior Room	Triple
5	Superior Room	Quad
6	Deluxe Room	Single
7	Deluxe Room	Double
8	Deluxe Room	Triple
9	Deluxe Room	Quad
10	Guest House	Single
11	Guest House	Double

Depending on the status of the rooms ,the Admin can add rooms.

DELETE ROOMS:

The screenshot shows a web browser window with three tabs: 'ANAPURNA HOTEL', 'SUNRISE HOTEL', and 'localhost / 127.0.0.1 / hotel / room'. The main content area is titled 'DELETE ROOM'. It contains a 'Delete room' form with a dropdown for 'Select the Room ID *' and a 'Delete Room' button. Below the form are eight room icons arranged in two rows of four. Each icon has a blue or green background and a white silhouette of three people. The labels are: 'Single Superior Room' (top row, first two), 'Double Superior Room' (top row, last two), 'Single Deluxe Room' (bottom row, first two), 'Double Deluxe Room' (bottom row, last two). The 'Single Superior Room' and 'Double Superior Room' icons have blue backgrounds, while the others have green backgrounds.

/Type here/

TESTING THE SAFETY OF THE WEBSITE:

Acunetix WVS Scheduler interface

Add Scheduled Scan Refresh Settings Start/Stop Service Service is running

Name	Details
queue1	Run once on 10/11/2020 at 22:07 finished
queue2	Run once on 10/11/2020 at 22:06 finished
queue3	Run once on 10/13/2020 at 12:20 started
http://localhost:80/Hotel/index.php	

Date Time Info

Oct 13 12:20:29 (B3CC) Progress: 76%
Oct 13 12:20:29 (B3CC) Progress: 77%
Oct 13 12:20:29 (B3CC) Progress: 78%
Oct 13 12:20:29 (B3CC) Progress: 79%
Oct 13 12:20:29 (B3CC) Progress: 80%
Oct 13 12:20:29 (B3CC) Progress: 81%
Oct 13 12:20:29 (B3CC) Progress: 82%
Oct 13 12:20:29 (B3CC) Progress: 83%
Oct 13 12:20:29 (B3CC) Progress: 84%
Oct 13 12:20:29 (B3CC) Progress: 85%
Oct 13 12:20:29 (B3CC) Progress: 86%
Oct 13 12:20:29 (B3CC) Progress: 87%
Oct 13 12:20:29 (B3CC) Progress: 88%
Oct 13 12:20:30 (B3CC) Progress: 89%
Oct 13 12:20:30 (B3CC) Progress: 90%
Oct 13 12:20:30 (B3CC) Scan Thread 1 [http://localhost:80/Hotel/index.php] - No alerts found.
Oct 13 12:20:30 (B3CC) Save scan results to database ...

Scheduler logs

Ready 3 queue(s), 3 scan(s)

Acunetix WVS Scheduler interface

Add Scheduled Scan Refresh Settings Start/Stop Service Service is running

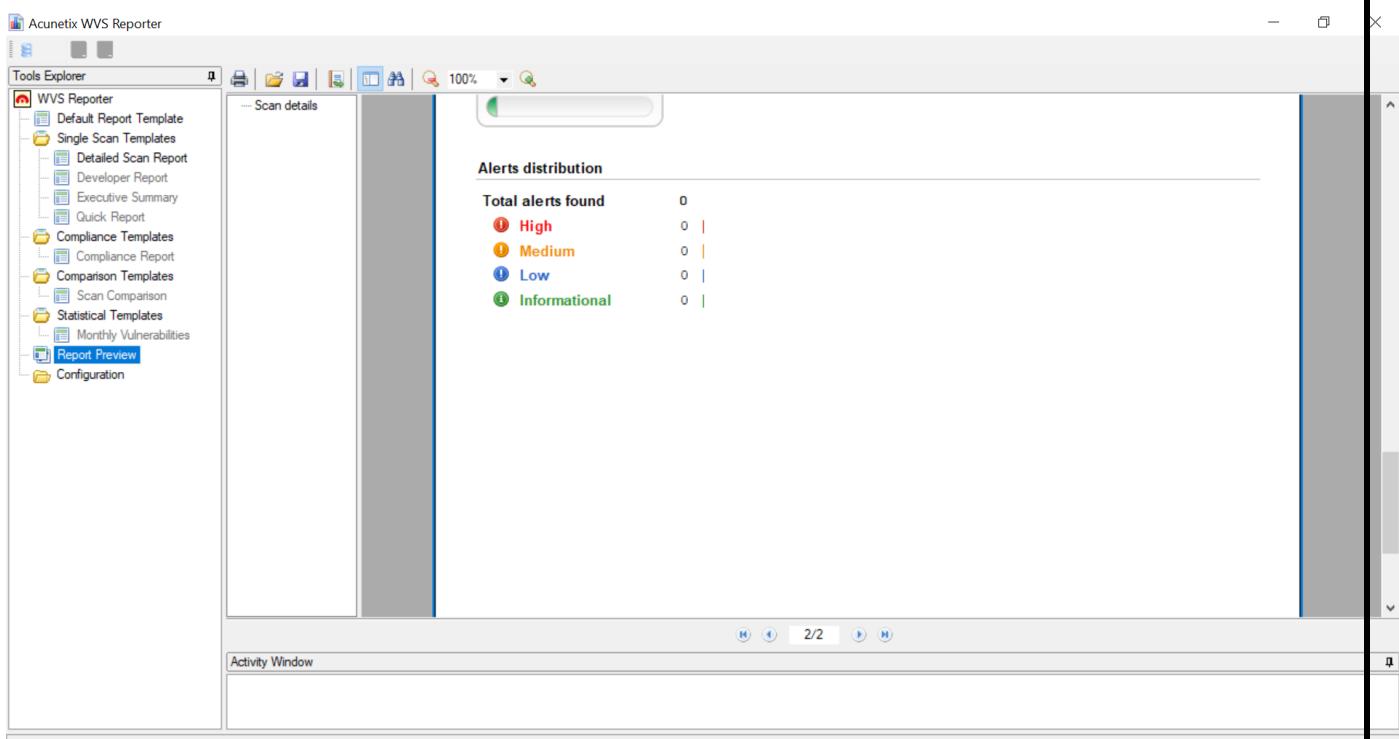
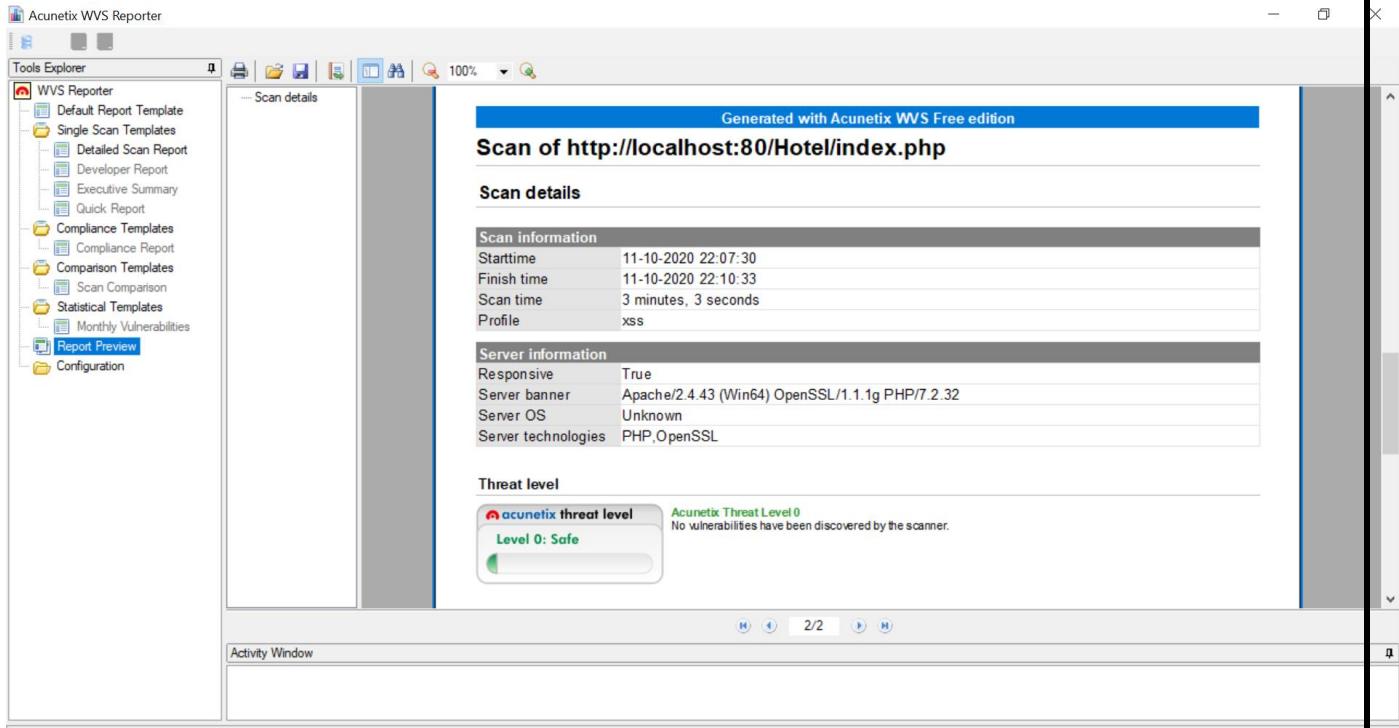
Name	Details
queue1	Run once on 10/11/2020 at 22:07 finished
queue2	Run once on 10/11/2020 at 22:06 finished
queue3	Run once on 10/13/2020 at 12:20 finished
http://localhost:80/Hotel/index.php	

Date Time Info

Oct 13 12:20:29 (B3CC) Progress: 81%
Oct 13 12:20:29 (B3CC) Progress: 82%
Oct 13 12:20:29 (B3CC) Progress: 83%
Oct 13 12:20:29 (B3CC) Progress: 84%
Oct 13 12:20:29 (B3CC) Progress: 85%
Oct 13 12:20:29 (B3CC) Progress: 86%
Oct 13 12:20:29 (B3CC) Progress: 87%
Oct 13 12:20:29 (B3CC) Progress: 88%
Oct 13 12:20:30 (B3CC) Progress: 89%
Oct 13 12:20:30 (B3CC) Progress: 90%
Oct 13 12:20:30 (B3CC) Scan Thread 1 [http://localhost:80/Hotel/index.php] - No alerts found.
Oct 13 12:20:30 (B3CC) Save scan results to database ...
Oct 13 12:20:30 (B3CC) Generate report ...
Oct 13 12:20:30 (B3CC) Generate report for http://localhost:80/Hotel/index.php ...
Oct 13 12:20:30 (B3CC) ExecuteCommand (c:\program files (x86)\acunetix\web vulnerability scanner 7\reporter.exe /r %DEFAULT% /t 2 /a REP /o "C:\P...
Oct 13 12:20:30 (B3CC) ExitCode = 0
Oct 13 12:20:31 (B3CC) Scan 'http://localhost:80/Hotel/index.php' finished.

Scheduler logs

Ready 3 queue(s), 3 scan(s)



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.

[Type here]