CHECK AND BOOK HOTEL SERVICE

Project report submitted in partial fulfillment of the requirement for the degree of Bachelor of Technology

in

Computer Science and Engineering

by

Charul Gupta (141274)

Shubham Sharma (141300)

Ritwik Raj (141323)

Akshit Goel (141375)

Ritika Dogra (141369)

Shubham Arora (141372)

Under the supervision of Dr. Punit Gupta to

Department of Computer Science and Engineering & Information Technology Waknaghat, Solan-173234, Himachal Pradesh

Acknowledgement

We take this opportunity to express our profound gratitude and deep regards to our guide Dr. Punit Gupta for his exemplary guidance, monitoring and constant encouragement throughout the course of the project. The blessing, help and guidance given by her time to time shall carry us a long way in the journey of life on which we are about to embark.

At the end we would like to express our sincere thanks to all our friends and others who helped us directly or indirectly during this project work.

Date:

Charul Gupta (141274)

Shubham Sharma (141300)

Ritwik Raj (141323)

Akshit Goel (141375)

Ritika Dogra (141369)

Shubham Arora (141372)

Table of contents

List of figures

Abstract

The knowledge and understanding of quality standards of guests helps hotel managers improve the quality of hotel services and increase guests' satisfaction with the hotel stay. Different aspects of a hotel offer participate in the guests' evaluation of the hotel experience. The factors that influence guests' satisfaction level are also named "hotel attributes". There exist
a large number of factors that influence guests' impressions, but certainly not all of them have the same level of importance for guests. In order to be aware of the strengths and weaknesses of their businesses, hotel management has to identify which improvements in the hotel operations can bring additional value to their guests. One of the most reliable information sources for gaining customer knowledge is undoubtedly the feedback provided directly from customers. If analyzed properly, it can be exploited for the purpose of improving the hotel operations and raising profits. The environment in which hospitality businesses operate nowadays has become radically connected to the use of the Internet. An increase in the usage of Social Media triggered extreme changes in the information channels that hotel guests use in order to make booking decisions. Hence, the hotel management should be aware that travel information websites present a valuable source of information about customer preferences. They offer service providers a possibility to have an insight into the reasons for satisfaction or disappointmen of their guests. The aim of this thesis is to examine the relationship between different hotel attributes and the guests' overall satisfaction with the hotel stay. It tries to uncover the most influential hotel attributes for the formation of guests' satisfaction. For this purpose, a content analysis of hotel reviews available from TripAdvisor was used. The results of this research indicate that the attribute "service" has the greatest influence on customer satisfaction with the hotel stay. In addition, it is suggested that the way a service is provided has more influence on the satisfaction than some physical aspects of the hotel stay. Moreover, the value received for the price paid is perceived as a factor that influences the satisfaction of guests, regardless of the star category of a hotel.

.

Introduction

1.1 **Introduction:**

La Fiesta is a hotel booking service which provides the facilities for booking hotels for customers by comparing hotel prices from two different databases. Any customer who need to book hotel, need to visit La Fiesta website for checking the availability of rooms in the hotel. They start to search, for different hotels that are available or that are relevant for their specific trip, then they select the hotel that they really want to have for that specific trip. Here in this web service clients get transparency and also best price is offered for that specific hotel. Those are really the benefits for the user. This web service helps the customer to find the best price among different databases. In La Fiesta,we aim to simplify the search process so users can find their ideal hotel. When a traveller selects criteria in the filters (let's say Wi-Fi), zour engines "scan" the hotel database and present to the user only those hotels listing Wi-Fi. At this moment, it is important to display correct hotel information. Hotels that do not include this information in the system will not be displayed in the search results. In turn, they automatically lose the chance of being seen and being booked. Hotel details became growingly significant with the evolution of hotel metasearch engines. In a time where choice is an important factor for customers, these sites enable users to search vast inventories of hotels, filtering and tailoring the results to find the hotel whose attributes match their needs. 47% of travellers use metasearch sites to compare hotel pricesz (ResearchNow data from 2013-2015). Once a destination is chosen and a budget set, the next step inzthe quest to find the ideal hotel is to narrow down the options. We find that users have a specific image in mind for the hotel that adheres to certain preferences: e.g. type of accommodation, style, facilities, and services. But how do you show travellers that your hotel is the perfect fit? Well, the easiest and most concise way is through your hotel details, also known as metadata**.** Metadata is the technical term for all the information related to your hotel. It's the details that describe your assets and show the traveller what you have on offer:check-in policies, parking, Wi-Fi, etc.

Online hotel reservations are azpopular method for booking hotel rooms. Travelers can book rooms on a computer by using online security to protect their privacy and financial information and by using several online travel agents to compare prices and facilities at different hotels. Prior to the Internet, travelers could write, telephone the hotel directly, or use a travel agent to make a reservation. Nowadays, online travel agents have pictures of hotels and rooms, information on prices and deals, and even information on local resorts. Many also allow reviews of the traveler to be recorded with the online travel agent.

Online hotel reservations are also helpful for making last minute travel arrangements. Hotels may drop the price of a room if some rooms are still available. There are several websites that specialize in searches for deals on rooms.

La Fiesta is a hotel search with an extensive price comparison. The prices shown come from numerous hotels and booking websites. This means that while users decide on trivago which hotel best suits their needs, the booking process itself is completed through the booking sites (which are linked to our website). By clicking on the "view deal" button, you will be forwarded onto a booking site where you can complete the reservation for the hotel deal found on trivago.

1.2 **Problem Statement:**

Create a service oriented web-service that would compare the hotel room prices by multiple service providers from 2 different DBs, and help the consumer find the best deal possible based on price and feedback. The user should also be able to book the hotel rooms through the service.

1.3 **Objective:**

The main objective of La Fiesta is to search and compare real time prices of hotels offered by two different travel agencies.

Chapter 2: Literature Survey

2.1 Database Systems

Data is collection of facts about the object of interest. For e.g. data about an employee would include information like name, address, age, educational qualifications etc. Software Applications need to store data as it is required to answer a question e.g. how many employees are above 40 years of age? Data is also acquired to convey a story e.g. why are we successful as an organization?

Data is raw, just a set of facts which by itself does not convey anything. We need to understand patterns between factual data and give it a meaning. This is called information which helps us with answers to questions like who, when, what, where etc. Synthesis of data and information leads us to answer the how question and take business decisions. This is referred to as Knowledge. Software Applications that use data are expected to meet several requirements from end users.

Database Systems

A Database is a shared collection of logically related data and description of these data, designed to meet the information needs of an organization. A Database Management System is a software system that enables users to define, create, maintain, and control access to the database. Database Systems typically have high cost and they require high end hardware configurations. An Application Program interacts with a database by issuing an appropriate request (typically a SQL statement).

Types Of Database Systems

Database systems are categorized into four types based upon the underlying structure used to store data. These database systems in chronological order of their evolution are Hierarchical, Network, Relational and NoSql. We will now get a brief overview of these database management systems.

Candidate Key

A Candidate Key is a minimal set of columns/attributesathat can be used to uniquely identify a single tuple in a relation. Candidate Keys are determined during database designabased on the underlying business rules of the database.

Primary key

Primary Key is the candidate key that is selected to uniquely identify a tuple in a relation. When two or more columns together identify the unique row then it's referred to as Composite Primary Key.

Foreign Key

A foreign key is a set of one or more columns in the child table whose values are required to match with corresponding columns in the parent table. Foreign key establishes aarelationship between these two tables. Foreign key columns on child tables must be primary key or unique on the parent table. The child table can contain NULL values.

SQL Commands

Structured Query Language (SQL) is used to manage data in all relational databases likeaDB2, Oracle, SQL Server etc. SQL standards are maintained by ISO. While most database products comply with the ISO standard, they also offer additional proprietary features.

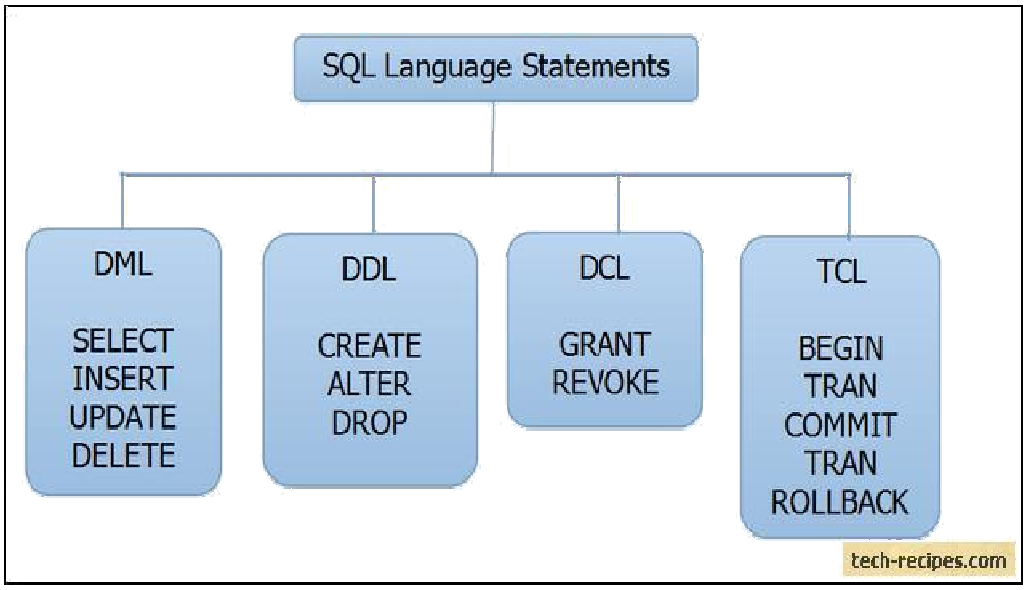


Figure1

DML

In Data Manipulation Language(DML), awe have four different SQL statements.

1. SELECT
2. INSERT
3. UPDATE

student Name is Kumar.

4. DELETE

   Delete statement is used to delete the existing record in the table, which is based on some condition.

   Eg., Delete from Student where StudentName=’Manoj’

   The query given above will delete the record which has StudentName Manoj.

DDL

In Data Definition Language (DDL), we have three different SQL statements.

1. CREATE
2. ALTER
3. DROP

DCL

In Data Control Language(DCL), it defines the control over the data in the database.

1. GRANT

   Grant is allowed to do the specified user to the specified tasks.

   Syntax

   GRANT privilege name
   ON object_name
   TO {user_name |PUBLIC |role_name}
   [WITH GRANT OPTION];

2. REVOKE

   It is used to cancel previously granted or denied permissions.

   Syntax

   REVOKE privilege name
   ON object_name
   FROM {user_name |PUBLIC |role_name}

TCL

1. COMMIT
2. ROLLBACK

3. SAVEPOINT

2.2 Fundamentals of java programming

JAVA was developed by Sun Microsystems Inc in 1991, later acquired by Oracle Corporation. It was developed by James Gosling and Patrick Naughton. It is a simple programming language. Writing, compiling and debugging a program is easy in java. It helps to create modular programs and reusable code.

JavaVirtualMachine(JVM)
This is generally referred as JVM. Before, we discuss about JVM lets see the phases of program execution. Phases are as follows: we write the program, then we compile the program and at last we run the program.
1) Writing of the program is of course done by java programmer like you and me.
2) Compilation of program is done by javac compiler, javac is the primary java compiler included in java development kit (JDK). It takes java program as input and generates java bytecode as output.
3) In third phase, JVM executes the bytecode generated by compiler. This is called program run phase.

Bytecode
As discussed above, javac compiler of JDK compiles the java source code into bytecode so that it can be executed by JVM. The bytecode is saved in a .class file by compiler.

JavaDevelopmentKit(JDK)
While explaining JVM and bytecode, I have used the term JDK. Let's discuss about it. As the name suggests this is complete java development kit that includes JRE (Java Runtime Environment), compilers and various tools like JavaDoc, Java debugger etc. In order to create, compile and run Java program you would need JDK installed on your computer.

JavaRuntimeEnvironment(JRE)
JRE is a part of JDK which means that JDK includes JRE. When you have JRE installed on your system, you an run a java program however you won't be able to compile it. JRE includes JVM, browser plugins and applets support. When you only need to run a java program on your computer, you would only need JRE.

An **array** is a collection of similar data in contiguous memory locations referred to by the same name.

- Can be used to store data of primitive as well as reference data type
- Holds a fixed number of data, decided at the time of array declaration
- It is an object in Java and is created dynamically

A **class** is a design or blueprint that describes the characteristics and behaviors of a real-time entity.

- It starts with the keyword "class" followed by a name.
- It specifies attributes (characteristics) and methods (behaviors).
- Attributes are the elements (instance variables) which hold the values of a particular entity.
- Methods are the sets of instructions which define the behaviors of the entity.

A **method** is a set of statements which depicts the behavior of a class.

An **object** is an instance of a class.

A **constructor** is a special method which has the same name as the class and doesn't return any value.

Types of constructors:

- Default constructor
    - A constructor which doesn't take any parameter
    - If the programmer doesn't provide a default constructor, it will be provided by the compiler
- Parameterized constructor
    - A constructor which accepts parameters (arguments)

Garbage collection

Whenever there is an allocation, there must be a mechanism for deallocation too.

Sometimes, even though a resource in a program is unreachable *or* not in use, the memory used by that resource is not freed. This is called Memory leak and is undesirable.

Earlier languages like C/C++ made the programmer responsible for freeing the memory occupied by such resources. Java, on the other hand, has a garbage collector which automatically deallocates the memory used by these resources. This prevents memory leak.

When an object does not haveaany reference, it becomes eligible for garbage collection**.**

Object oriented

Java is an object oriented language.

Even in real life, we are surrounded by objects. We use them, we interact with them, awe get things done by them. Some of the object oriented principles are:

i. Abstraction

- The process of exposing the relevant details and hiding the irrelevant ones

  The dashboard of a car is exposed to the driver

ii. Encapsulation

- The process of **restricting access** to the members
- Data members (attributes) are usually made private to prevent direct access to them
- Methods (behaviors) are usually public to be accessible from outside the class

  The internals of a car are hidden from its drivers

iii. Inheritance

- A technique to create **specialized** classes from a general class
- Defines a class hierarchy

  Bike and car are automobiles

iv. Polymorphism

- An object's ability to **behave differently** depending upon the context
- Method overloading and method overriding are the two ways to achieve polymorphism

  An automobile behaves differently depending on whether it is a bike or a car

While we are learning the Object Oriented Concepts, it is important to understand the relationships between one object and another.

These relationships can be primarily classified as:

- Association:

When an object utilizes another object in order to perform its activities, aa relationship of association is established between them. This is termed as 'uses-a' relationship.

Example: A driver uses a car to travel Association (uses-a) is denoted by a line.

- Aggregation:

  When an object contains another object as its attribute, we have an aggregation between them. aIt is termed as "has-a" relationship. Example: A car has an engine Aggregation (has-a) isadenoted by a link with a diamond head.

- Inheritance:

  When an object is based on another object, i.e. acquires characteristics and behaviors from another object, the relationship is that of inheritance. Inheritance or "**Is-a**" relationship exhibits the concept of reusability through a valid relationship between any two real world objects. Example: A **projectamanager** is an employee, A car is an automobile.

Method overloading

When we need different implementations of the same behavior depending upon the context, we go for method overloading. Method overloading lets you have more than one method with the same name in a class. The methods accept parameters differing in their data types, the number of parameters, or their order.

Method overriding

If subclasssa (child magnificence) has the equal technique asadeclared in the determine elegance, ait isaknown as technique overriding in java. aIn different phrases, If subclassaprovides the specific implementation of the approach that hasabeen supplied via one among its apparent class, it's far acknowledged asamethod overriding.

Super keyword

The super keyword in java is a reference variable which is used to refer immediate parent class object.

Whenever you create the instance of subclass, an instance of parent class is created implicitly which is referred by super reference variable.

Usage of java super Keyword

1. super can be used to refer immediate parent class instance variable.

2. super can be used to invoke immediate parent class method.
3. super() can be used to invoke immediate parent class constructor.

The static can be:

1. variable(also known as class variable)
2. method (also known as class method)
3. block
4. nested class

static blocks are used to initialize static variables when it cannot be done in a single line. They can be used to add preprocessing if required. Static blocks get executed when a class gets loaded. If there are multiple static blocks, they will be executed in the order of their occurrence.

Final keyword

The final keyword in java is used to restrict the user. The java final keyword can be used in many context. Final can be:

1. variable
2. method
3. class

The final keyword can be applied with the variables, a final variable that have no value it is called blank final variable or uninitialized final variable. It can be initialized in the constructor only. The blank final variable can be static also which will be initialized in the static block only.

Interfaces

An interface in java is a blueprint of a class. It has static constants and abstract methodsIt is used to achieve abstraction and multiple inheritance in Java. In other words, you can say that interfaces can have methods and variables but the methods declared in interface contain only method signature, not body. Java Interface also represents IS-A relationship. It cannot be instantiated just like abstract class.

There are mainly three reasons to use interface. They are given below.

- It is used to achieve abstraction.
- By interface, we can support the functionality of multiple inheritance.
- It can be used to achieve loose coupling.

The object Class

The Object class is the parent class of all the classes in java by default.

Some of the methods are:

1. equals():

   The java string equals() method compares the two given strings based on the content of the string. If any character is not matched, it returns false. If all characters are matched, it returns true.

2. hashcode():

3. toString():

The method is used to get a String object representing the value of the Number Object. If the method takes a primitive data type as an argument, then the String object representing the primitive data type value is returned. If the method takes two arguments, then a String representation of the first argument in the radix specified by the second argument will be returned.

compareTo() Method

The method compareTo() is used for comparing two strings lexicographically. Each character of both the strings is converted into a Unicode value for comparison. If both the strings are equal then this method returns 0 else it returns positive or negative value. The result is positive if the first string is lexicographically greater than the second string else the result would be negative.

Strings

In java, string is basically an object that represents sequence of char values. An array of characters works same as java string. Java String class provides a lot of methods to perform operations on string such as compare(), concat(), equals(), split(), length(), replace(), compareTo(), intern(), substring() etc. The CharSequence interface is used to represent sequence of characters. It is implemented by String, StringBuffer and StringBuilder classes. The java String is immutable i.e. it cannot be changed. Whenever we change any string, a new instance is created. For mutable string, you can use StringBuffer and StringBuilder classes.

Wrapper classes

All the wrapper classes (Integer, Long, Byte, Double, Float, Short) are subclasses of the abstract class Number. The object of the wrapper class contains or wraps its respective primitive data type. Converting primitive data types into object is called boxing, and this is taken care by the compiler. Therefore, while using a wrapper class you just need to pass the value of the primitive data type to the constructor of the Wrapper class.

And the Wrapper object will beconverted back to a primitive data type, and this process is called unboxing. The Number class is part of the java.lang package.

Enum in java

Enum in java is a data type that contains fixed set of constants.

It can be used for days of the week (SUNDAY, MONDAY, TUESDAY, WEDNESDAY, THURSDAY, FRIDAY and SATURDAY) , directions (NORTH, SOUTH, EAST and WEST) etc. The java enum constants are static and final implicitly. It is available from JDK 1.5.

Java Enums can be thought of as classes that have fixed set of constants.

Important points in enum are given below:

- enum improves type safety
- enum can be easily used in switch
- enum can be traversed
- enum can have fields, constructors and methods
- enum may implement many interfaces but cannot extend any class because it internally extends Enum class

Types of design patterns

Advantage of design pattern:

1. They are reusable in multiple projects.
2. They provide the solutions that help to define the system architecture.
3. They capture the software engineering experiences.
4. They provide transparency to the design of an application.
5. They are well-proved and testified solutions since they have been built upon the knowledge and experience of expert software developers.
6. Design patterns dont guarantee an absolute solution to a problem. They provide clarity to the system architecture and the possibility of building a better system.

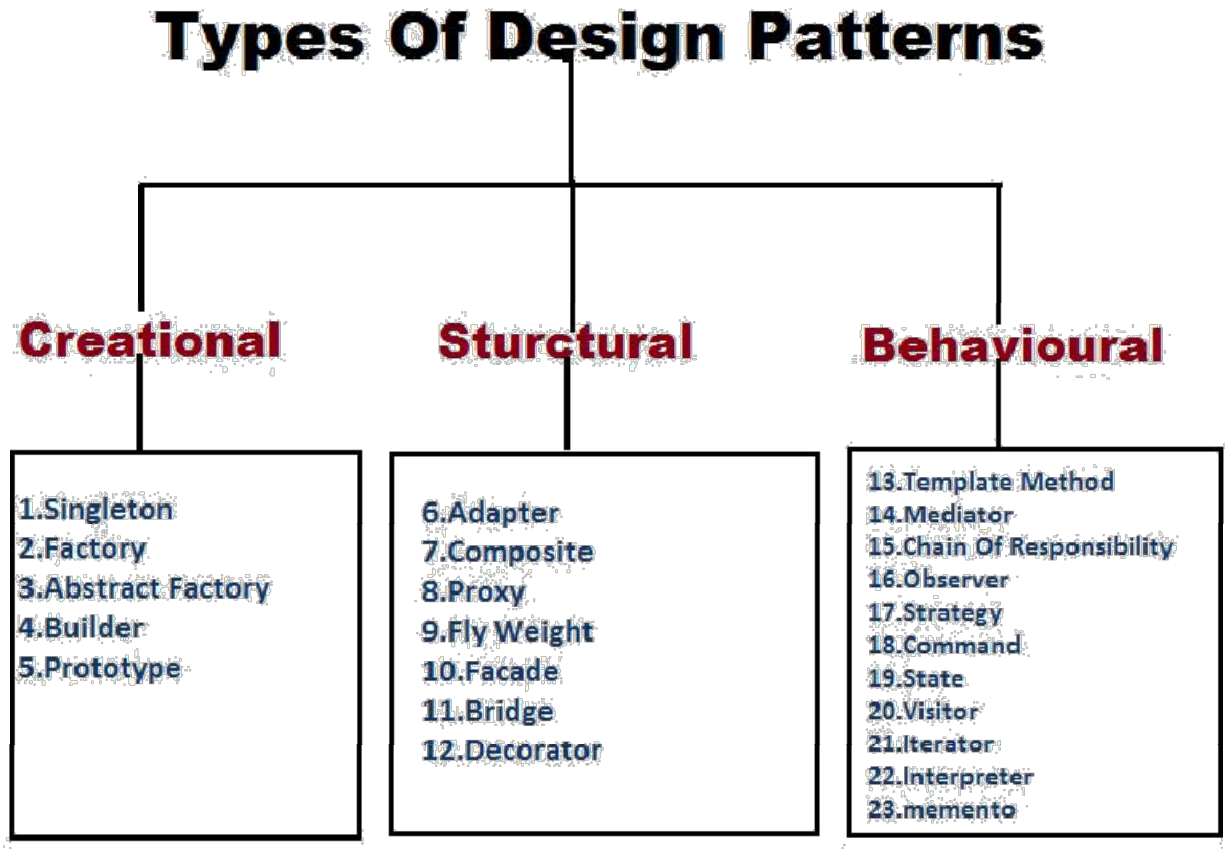


Figure:2

Creational patterns

In software engineering, creational design patterns are design patterns that deal with object creation mechanisms, trying to create objects in a manner suitable to the situation. The basic form of object creation could result in design problems or added complexity to the design. Creational design patterns solve this problem by somehow controlling this object creation.

- AbstractFactory
  Creates an instance of several families of classes
- Builder
  Separates object construction from its representation
- FactoryMethod
  Creates an instance of several derived classes
- ObjectPool
  Avoid expensive acquisition and release of resources by recycling objects that are no longer in use
- Prototype
  A fully initialized instance to be copied or cloned

- Singleton
  A class of which only a single instance can exist

Structural patterns

In Software Engineering, Structural Design Patterns are Design Patterns that ease the design by identifying a simple way to realize relationships between entities.

- Adapter
  Match interfaces of different classes
- Bridge
  Separates an object's interface from its implementation
- Composite
  A tree structure of simple and composite objects
- Decorator
  Add responsibilities to objects dynamically
- Facade
  A single class that represents an entire subsystem
- Flyweight
  A fine-grained instance used for efficient sharing
- PrivateClassData
  Restricts accessor/mutator access
- Proxy
  An object representing another object

Behavioral patterns

In software engineering, behavioral design patterns are design patterns that identify common communication patterns between objects and realize these patterns. By doing so, these patterns increase flexibility in carrying out this communication.

- Chainofresponsibility
  A way of passing a request between a chain of objects
- Command
  Encapsulate a command request as an object
- Interpreter
  A way to include language elements in a program
- Iterator
  Sequentially access the elements of a collection
- Mediator
  Defines simplified communication between classes
- Memento
  Capture and restore an object's internal state
- NullObject
  Designed to act as a default value of an object

- Observer
  A way of notifying change to a number of classes
- State
  Alter an object's behavior when its state changes
- Strategy
  Encapsulates an algorithm inside a class
- TemplateMethod
  Defer the exact steps of an algorithm to a subclass
- Visitor
  Defines a new operation to a class without change

Java Calendar Class

The java.util.calendar class is an abstract class that provides methods for converting between a specific instant in time and a set of calendar fields such as YEAR, MONTH, DAY_OF_MONTH, HOUR, and so on, and for manipulating the calendar fields, such as getting the date of the next week.Following are the important points about Calendar −

- This class also provides additional fields and methods for implementing a concrete calendar system outside the package.
- Calendar defines the range of values returned by certain calendar fields.

Java local date and time Class

Java LocalDate class is an immutable class that represents Date with a default format of yyyy-MM-dd. It inherits Object class and implements the ChronoLocalDate interface.

Java LocalDateTime class is an immutable date-time object that represents a date-time, with the default format as yyyy-MM-dd-HH-mm-ss.zzz. It inherits object class and implements the ChronoLocalDateTime interface.

2.3 **Developing persistence tier of an enterprise application using hibernate**

Hibernate framework simplifies the development of java application to interact with the database. Hibernate is an open source, lightweight, ORM (Object Relational Mapping) tool.

An ORM tool simplifies the data creation, data manipulation and data access. It is a programming technique that maps the object to the data stored in the database.

Advantages of Hibernate Framework

There are many advantages of Hibernate Framework. They are as follows:

1) Opensource and Lightweight**:** Hibernate framework is opensource under the LGPL license and lightweight.

2) Fast performance: The performance of hibernate framework is fast because cache is internally used in hibernate framework. There are two types of cache in hibernate framework first level cache and second level cache. First level cache is enabled bydefault.

3) Database Independent query: HQL (Hibernate Query Language) is the object-oriented version of SQL. It generates the database independent queries. So you don't need to write database specific queries. Before Hibernate, If database is changed for the project, we need to change the SQL query as well that leads to the maintenance problem.

4) Automatic table creation: Hibernate framework provides the facility to create the tables of the database automatically. So there is no need to create tables in the database manually.

5) Simplifies complex join: To fetch data form multiple tables is easy in hibernate framework.

6) Provides query statistics and database status: Hibernate supports Query cache and provide statistics about query and database status.

Elements of Hibernate Architecture

For creating the first hibernate application, we must know the elements of Hibernate architecture. They are as follows:

SessionFactory

The SessionFactory is a factory of session and client of ConnectionProvider. It holds second level cache (optional) of data. The org.hibernate.SessionFactory interface provides factory method to get the object of Session.

Session

The session object provides an interface between the application and data stored in the database. It is a short-lived object and wraps the JDBC connection. It is factory of Transaction, Query and Criteria. It holds a first-level cache (mandatory) of data. The org.hibernate.Session interface provides methods to insert, update and delete the object. It also provides factory methods for Transaction, Query and Criteria.

Transaction

The transaction object specifies the atomic unit of work. It is optional. The org.hibernate.Transaction interface provides methods for transaction management.

ConnectionProvider

It is a factory of JDBC connections. It abstracts the application from DriverManager or DataSource. It is optional.

TransactionFactory

It is a factory of Transaction. It is optional.

CRUD Operations in hibernate

4 hibernate crud operations are:

1. Create a record
2. Read a record
3. Update a record
4. Delete a record

Hibernate persist and save operation

Hibernate saveOrUpdate results into insert or update queries based on the provided data. If the data is present in the database, update query is executed.

We can use saveOrUpdate() without transaction also, but again you will face the issues with mapped objects not getting saved if session is not flushed.

Hibernate saveOrUpdate adds the entity object to persistent context and track any further changes. Any further changes are saved at the time of committing transaction, like persist.

Hibernate persist is similar to save (with transaction) and it adds the entity object to the persistent context**,** so any further changes are tracked. If the object properties are changed before the transaction is committed or session is flushed, it will also be saved into database.

Second difference is that we can use persist() method only within the boundary of a transaction, so it's safe and takes care of any cascaded objects.

Finally, persist doesn't return anything so we need to use the persisted object to get the generated identifier value.

Hibernate life cycle

Hibernate defines different states and state transitions for a persistent object in its lifecyle.

The different states of a persistent object are:

- New/Transient State: A newly created object which is not associated with persistence context.
- Managed/Persistent State: An entity instance with a row representation in the database and it is currently associated with a persistence context.
- Detached State: An entity instance with or without a row identity in the database and which is no longer associated with a persistence context, usually because the persistence context was closed or the instance was evicted from the context.
- Removed State: An entity instance with a row identity in the database and associated with a persistence context, but scheduled for removal from the database.

Hibernate generator classes

The <generator> subelement of id used to generate the unique identifier for the objects of persistent class. There are many generator classes defined in the Hibernate Framework.

All the generator classes implements the org.hibernate.id.IdentifierGenerator interface. The application programmer may create one's own generator classes by implementing the IdentifierGenerator interface. Hibernate framework provides many built-in generator classes:

1. Assigned:

   It is the default generator strategy if there is no <generator> element . In this case, application assigns the id.

2. Increment:

   It generates the unique id only if no other process is inserting data into this table. It generates short, int or long type identifier. The first generated identifier is 1 normally and incremented as 1.

3. Sequence:

   It uses the sequence of the database. if there is no sequence defined, it creates a sequence automatically e.g. in case of Oracle database, it creates a sequence named HIBERNATE_SEQUENCE. In case of Oracle, DB2, SAP DB, Postgre SQL or McKoi, it uses sequence but it uses generator in interbase.

4. Hilo:

   It uses high and low algorithm to generate the id of type short, int and long.

5. Native:

   It uses identity, sequence or hilo depending on the database vendor.

6. Identity:

   It is used in Sybase, My SQL, MS SQL Server, DB2 and HypersonicSQL to support the id column. The returned id is of type short, int or long.

7. Seqhilo:

   It uses high and low algorithm on the specified sequence name. The returned id is of type short, int or long.

8. Uuid:

   It uses 128-bit UUID algorithm to generate the id. The returned id is of type String, unique within a network (because IP is used). The UUID is represented in hexadecimal digits, 32 in length.

9. Guid:

It uses GUID generated by database of type string. It works on MS SQL Server and MySQL.

10. Select:

    It uses the primary key returned by the database trigger.

11. Foreign:

    It uses the id of another associated object, mostly used with <one-to-one> association.

12. Sequence-identity:

    It uses a special sequence generation strategy. It is supported in Oracle 10g drivers only.

Hibernate - One-to-One Mappings

A one-to-one association is similar to many-to-one association with a difference that the column will be set as unique. For example, an address object can be associated with a single employee object.

Hibernate - Many-to-One Mappings

A many-to-one association is the most common kind of association where an Object can be associated with multiple objects. For example, the same address object can be associated with multiple employee objects.

Hibernate - Many-to-Many Mappings

A Many-to-Many mapping can be implemented using a Set java collection that does not contain any duplicate element. A Set is mapped with a <set> element in the mapping table and initialized with java.util.HashSet.

Hibernate Query Language (HQL)

Hibernate Query Language (HQL) is same as SQL (Structured Query Language) but it doesn't depends on the table of the database. Instead of table name, we use class name in HQL. So it is database independent query language.

Advantage of HQL

There are many advantages of HQL. They are as follows:

- database independent
- supports polymorphic queries
- easy to learn for Java Programmer

Query Interface

It is an object oriented representation of Hibernate Query. The object of Query can be obtained by calling the createQuery() method Session interface.

The query interface provides many methods. There is given commonly used methods:

1. public int executeUpdate() is used to execute the update or delete query.
2. public List list() returns the result of the relation as a list.
3. public Query setFirstResult(int rowno) specifies the row number from where record will be retrieved.
4. public Query setMaxResult(int rowno) specifies the no. of records to be retrieved from the relation (table).
5. public Query setParameter(int position, Object value) it sets the value to the JDBC style query parameter.
6. public Query setParameter(String name, Object value) it sets the value to a named query parameter.

Hibernate – Caching

Caching is a mechanism to enhance the performance of a system. It is a buffer memory that lies between the application and the database. Cache memory stores recently used data items in order to reduce the number of database hits as much as possible.

First-level Cache

The first-level cache is the Session cache and is a mandatory cache through which all requests must pass. The Session object keeps an object under its own power before committing it to the database.

If you issue multiple updates to an object, Hibernate tries to delay doing the update as long as possible to reduce the number of update SQL statements issued. If you close the session, all the objects being cached are lost and either persisted or updated in the database.

Second-level Cache

Second level cache is an optional cache and first-level cache will always be consulted before any attempt is made to locate an object in the second-level cache. The second level cache can be configured on a per-class and per-collection basis and mainly responsible for caching objects across sessions.

Any third-party cache can be used with Hibernate. An org.hibernate.cache.CacheProvider interface is provided, which must be implemented to provide Hibernate with a handle to the cache implementation.

Query-level Cache

Hibernate also implements a cache for query resultsets that integrates closely with the second-level cache.

This is an optional feature and requires two additional physical cache regions that hold the cached query results and the timestamps when a table was last updated. This is only useful for queries that are run frequently with the same parameters.

The Second Level Cache

Hibernate uses first-level cache by default and you have nothing to do to use first-level cache. Let's go straight to the optional second-level cache. Not all classes benefit from caching, so it's important to be able to disable the second-level cache.

The Hibernate second-level cache is set up in two steps. First, you have to decide which concurrency strategy to use. After that, you configure cache expiration and physical cache attributes using the cache provider.

Concurrency Strategies

A concurrency strategy is a mediator, which is responsible for storing items of data in the cache and retrieving them from the cache. If you are going to enable a second-level cache, we will have to decide, for each persistent class and collection, which cache concurrency strategy to use.

- Transactional – Use this strategy for read-mostly data where it is critical to prevent stale data in concurrent transactions, in the rare case of an update.
- Read-write – Again use this strategy for read-mostly data where it is critical to prevent stale data in concurrent transactions, in the rare case of an update.
- Nonstrict-read-write – This strategy makes no guarantee of consistency between the cache and the database. Use this strategy if data hardly ever changes and a small likelihood of stale data is not of critical concern.
- Read-only – A concurrency strategy suitable for data, which never changes. Use it for reference data only.

2.4: Creating UI for web using HTML5

HTML (**H**yper **T**ext **M**arkup **L**anguage)

- Hypertext is the text displayed on a computer which refers to other text or image that the user can access
- Markup language is a set of tags which helps the web browser in presenting text in a document

HTML can:

- Publish documents with text, headlines, images etc
- Create forms to collect user data
- Include videos, audio clips, flash movies etc. inside an HTML document
- Access online information via hyperlinks

Block-Level and Inline Elements

HTML is made up of various elements that act as the building blocks of web pages. For the purpose of styling, elements are divided into two categories: *block-level* elements and *inline* elements.
A element is used as an inline element and a element as a block level element.
An inline element does not cause a line break (start on a new line) and does not take up the full width of a page, only the space bounded by its opening and closing tag. It is usually used within other HTML elements.

A block-level element always starts on a new line and takes up the full width of a page, from left to right. A block-level element can take up one line or multiple lines and has a line break before and after the element.

Basic HTML Elements

The basic elements of an HTML page are:

- A text header, denoted using the <h1>,<h2>,<h3>,<h4>,<h5>,<h6>tags.
- A paragraph, denoted using the <p> tag
- A horizontal ruler, denoted using the <hr> tag
- A link, denoted using the <a>(anchor) tag
- A list, denoted using the (unordered list), <ol> (ordered list) and <li> (list element) tags
- An image, denoted using the <img> tag
- An divisor, denoted using the tag
- A text span, denoted using the tag

HTML Form Elements

The <input> Element

The most important form element is the <input> element.

The <input> element can be displayed in several ways, depending on the type attribute.

The <select> Element

The <select> element defines a drop-down list**.**

The <textarea> Element

The <textarea> element defines a multi-line input field (a text area). The rows attribute specifies the visible number of lines in a text area. The cols attribute specifies the visible width of a text area.

The <button>Element

The <button> element defines a clickable button.

HTML5 <datalist> Element

The <datalist> element specifies a list of pre-defined options for an <input>element.

Users will see a drop-down list of the pre-defined options as they input data.

The list attribute of th <input> .element, must refer to the id attribute of the <datalist> element.

HTML5 <output> Element

The <output>element represents the result of a calculation.

HTML5 - Audio & Video

HTML5 features include native audio and video support without the need for Flash.

The HTML5 <audio> and <video> tags make it simple to add media to a website.

The HTML5 video tag can have a number of attributes to control the look and feel and various functionalities of the control. HTML5 supports <audio> tag which is used to embed sound content in an HTML or XHTML document

HTML Text Formatting

Formatting elements were designed to display special types of text:

- <b>- Bold text
  - Important text
- <i>- Italic text
- - Emphasized text
- <mark>- Marked text
- <small>- Small text
- <del>- Deleted text
- <ins>- Inserted text
- <sub>- Subscript text
- <sup>- Superscript text

HTML Lists

Unordered HTML List

An unordered list starts with the <ul> tag. Each list item starts with the <li> tag.

Ordered HTML List

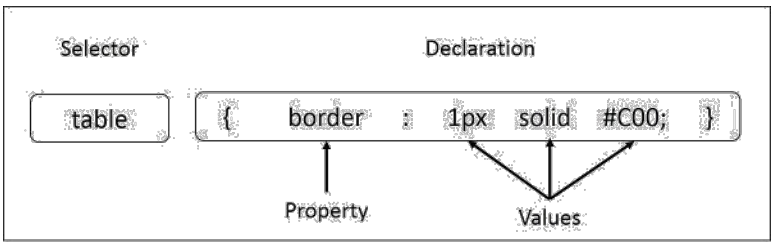
An ordered list starts with the <ol> tag. Each list item starts with the <li> tag.

Chapter:2.5: Styling web pages using css3

CSS (or Cascading Style Sheets) is a language/technology that is used for defining the presentation of an HTML page**.**

It was introduced primarily to separate the page content from the presentation. This provides greater flexibility and control over the styling. It helps in separating the style information and the HTML content.

CSS Syntax



The selector points to the HTML element you want to style.

The declaration block contains one or more declarations separated by semicolons.

Each declaration includes a CSS property name and a value, separated by a colon.

A CSS declaration always ends with a semicolon, and declaration blocks are surrounded by curly braces.

CSS Selectors

The element Selector

The element selector selects elements based on the element name.

The id Selector

The id selector uses the id attribute of an HTML element to select a specific element.

The id of an element should be unique within a page, so the id selector is used to select one unique element!

To select an element with a specific id, write a hash (#) character, followed by the id of the element.

The class Selector

The class selector selects elements with a specific class attribute.

To select elements with a specific class, write a period (.) character, followed by the name of the class.

Three Types of CSS

External style sheets are separate files full of CSS instructions (with the file extension .css). When any web page includes an external stylesheet, its look and feel will be controlled by this CSS file (unless you decide to override a style using one of these next two types). This is how you change a whole website at once. And that's perfect if you want to keep up with the latest fashion in web pages without rewriting every page!

Internal styles are placed at the top of each web page document, before any of the content is listed. This is the next best thing to external, because they're easy to find, yet allow you to 'override' an external style sheet -- for that special page that wants to be a nonconformist!

Inline styles are placed right where you need them, next to the text or graphic you wish to decorate. You can insert inline styles anywhere in the middle of your HTML code, giving you real freedom to specify each web page element. On the other hand, this can make maintaining web pages a real chore!

CSS Text

Text Color

The color's property is used to set the color of the text. The color is specified by:

- a color name - like "red"
- a HEX value - like "#ff0000"
- an RGB value - like "rgb(255,0,0)"

Text Alignment

The text-align property is used to set the horizontal alignment of a text.

A text can be left or right aligned, centered,

or justified. Text Decoration

The ext-decoration property is used to set or remove decorations from text.

The value text-decoration:none; is often used to remove underlines from links.

Text Transformation

The text-transform property is used to specify uppercase and lowercase letters in a text.

It can be used to turn everything into uppercase or lowercase letters, or capitalize the first letter of each word.

Text Indentation

The text-indent property is used to specify the indentation of the first line of a text.

Letter Spacing

The letter-spacing property is used to specify the space between the characters in a text.

Line Height

The line-height property is used to specify the space between lines.

Text Direction

The direction property is used to change the text direction of an element.

Word Spacing

The word-spacing property is used to specify the space between the words in a text.

CSS Lists

In HTML, there are two main types of lists:

- unordered lists (<ul>) - the list items are marked with bullets
- ordered lists (<ol>) - the list items are marked with numbers or letters

DifferentList Item Markers

The list property specifies the type of list item marker.

An Image as The List Item Marker

The list-type-image property specifies an image as the list item marker:

Position The List Item Markers

The list-style-position property specifies the position of the list-item markers (bullet points).

"list-style-position: outside;" means that the bullet points will be outside the list item. The start of each line of a list item will be aligned vertically.

"list-style-position: inside;" means that the bullet points will be inside the list item. As it is part of the list item, it will be part of the text and push the text at the start.

Remove Default Settings

The list-style-type:none property can also be used to remove the markers/bullets. The list also has default margin and padding.

List - Shorthand property

The list-style property is a shorthand property.

CSS Borders

Border Style

The border-style property specifies what kind of border to display.

The following values are allowed:

- dotted - Defines a dotted border
- dashed - Defines a dashed border
- solid - Defines a solid border
- double - Defines a double border
- groove - Defines a 3D grooved border. The effect depends on the border-color value
- ridge - Defines a 3D ridged border. The effect depends on the border-color value
- inset - Defines a 3D inset border. The effect depends on the border-color value
- outset - Defines a 3D outset border. The effect depends on the border-color value
- none - Defines no border
- hidden - Defines a hidden border

Border Width

The border-width property specifies the width of the four borders.

The width can be set as a specific size (in px, pt, cm, em, etc) or by using one of the three pre-defined values: thin, medium, or thick.

The border-width property can have from one to four values (for the top border, right border, bottom border, and the left border).

Border Color

The border-color property is used to set the color of the four borders.

The color can be set by:

- name - specify a color name, like "red"
- Hex - specify a hex value, like "#ff0000"
- RGB - specify a RGB value, like "rgb(255,0,0)"
- transparent

The border-color property can have from one to four values (for the top border, right border, bottom border, and the left border).

CSS Fonts

In CSS, there are two types of font family names:

- generic family - a group of font families with a similar look (like "Serif" or "Monospace")
- font family - a specific font family (like "Times New Roman" or "Arial")

Font Family

The font family of a text is set with the font-family property.

The font-family property should hold several font names as a "fallback" system. If the browser does not support the first font, it tries the next font, and so on.

Font Style

The font-style property is mostly used to specify italic text.

This property has three values:

- normal - The text is shown normally
- italic - The text is shown in italics
- oblique - The text is "leaning" (oblique is very similar to italic, but less supported)

Font Size

The font-size property sets the size of the text.

Being able to manage the text size is important in web design. However, you should not use font size adjustments to make paragraphs look like headings, or headings look like paragraphs.

Always use the proper HTML tags, like <h1> - <h6> for headings and <p> for paragraphs.

The font-size value can be an absolute, or relative size.

Absolute size:

- Sets the text to a specified size
- Does not allow a user to change the text size in all browsers (bad for accessibility reasons)
- Absolute size is useful when the physical size of the output is known

Relative size:

- Sets the size relative to surrounding elements
- Allows a user to change the text size in browsers

Font Weight

The font-weight property specifies the weight of a font.

CSS Layout - The position Property

The position Property

The position property specifies the type of positioning method used for an element.

There are four different position values:

- static
- relative
- fixed
- absolute

position: static;

HTML elements are positioned static by default.

Static positioned elements are not affected by the top, bottom, left, and right properties.

An element with position: static; is not positioned in any special way; it is always positioned according to the normal flow of the page:

position: relative;

An element with position: relative; is positioned relative to its normal position.

Setting the top, right, bottom, and left properties of a relatively-positioned element will cause it to be adjusted away from its normal position. Other content will not be adjusted to fit into any gap left by the element.

position: fixed;

An element with position: fixed; is positioned relative to the viewport, which means it always stays in the same place even if the page is scrolled. The top, right, bottom, and left properties are used to position the element.

A fixed element does not leave a gap in the page where it would normally have been

located. position: absolute;

An element with position: absolute; is positioned relative to the nearest positioned ancestor (instead of positioned relative to the viewport, like fixed).

However; if an absolute positioned element has no positioned ancestors, it uses the document body, and moves along with page scrolling.

Chapter:2.6: Bootstrap

What is Bootstrap?

- Bootstrap is a free front-end framework for faster and easier web development
- Bootstrap includes HTML and CSS based design templates for typography, forms, buttons, tables, navigation, modals, image carousels and many other, as well as optional JavaScript plugins
- Bootstrap also gives you the ability to easily create responsive designs

Responsive web design

The basic features of responsive web design are as follows:

- Fluid Layout : Layout grows and shrinks based on the size of the device browser
- Flexible Images : Images adapt to the size of the device browser
- Responsiveness : Selectively apply CSS based on size of browser / device using media queries

Bootstrap Container

The container class is used to create boxed content.

There are two container classes in Bootstrap:

1. container
2. container-fluid

Bootstrap Grid System

Bootstrap's grid system is responsive, and the columns will re-arrange depending on the screen size: On a big screen it might look better with the content organized in three columns, but on a small screen it would be better if the content items were stacked on top of each other.

Grid Classes

The Bootstrap grid system has four classes:

- xs (for phones - screens less than 768px wide)

- sm (for tablets - screens equal to or greater than 768px wide)
- md (for small laptops - screens equal to or greater than 992px wide)
- lg (for laptops and desktops - screens equal to or greater than 1200px wide)

Grid System Rules

Some Bootstrap grid system rules:

- Rows must be placed within a .container (fixed-width) or .container-fluid (full-width) for proper alignment and padding
- Use rows to create horizontal groups of columns
- Content should be placed within columns, and only columns may be immediate children of rows
- Predefined classes like .row and .col-sm-4 are available for quickly making grid layouts
- Columns create gutters (gaps between column content) via padding. That padding is offset in rows for the first and last column via negative margin on rows
- Grid columns are created by specifying the number of 12 available columns you wish to span. For example, three equal columns would use three .col-sm-4
- Column widths are in percentage, so they are always fluid and sized relative to their parent element

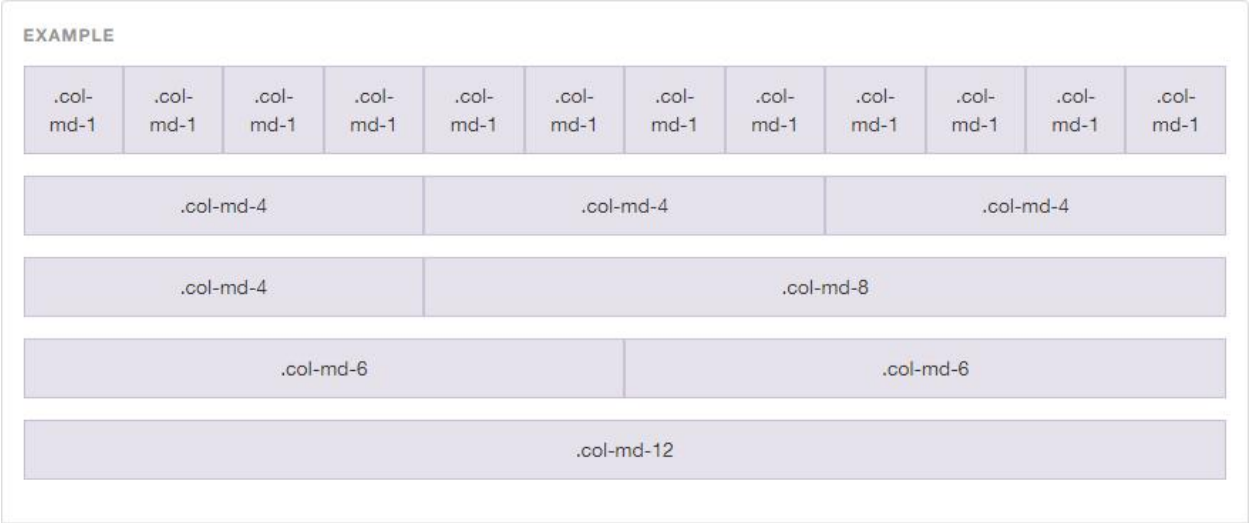


Figure:3

In Bootstrap, each webpage can be divided into rows and column-grids, and each row has 12 column-grids.

Chapter-3 System Development

a. **Development:**

The system development of “Check and book a hotel service” is as follows:

b. System Design:

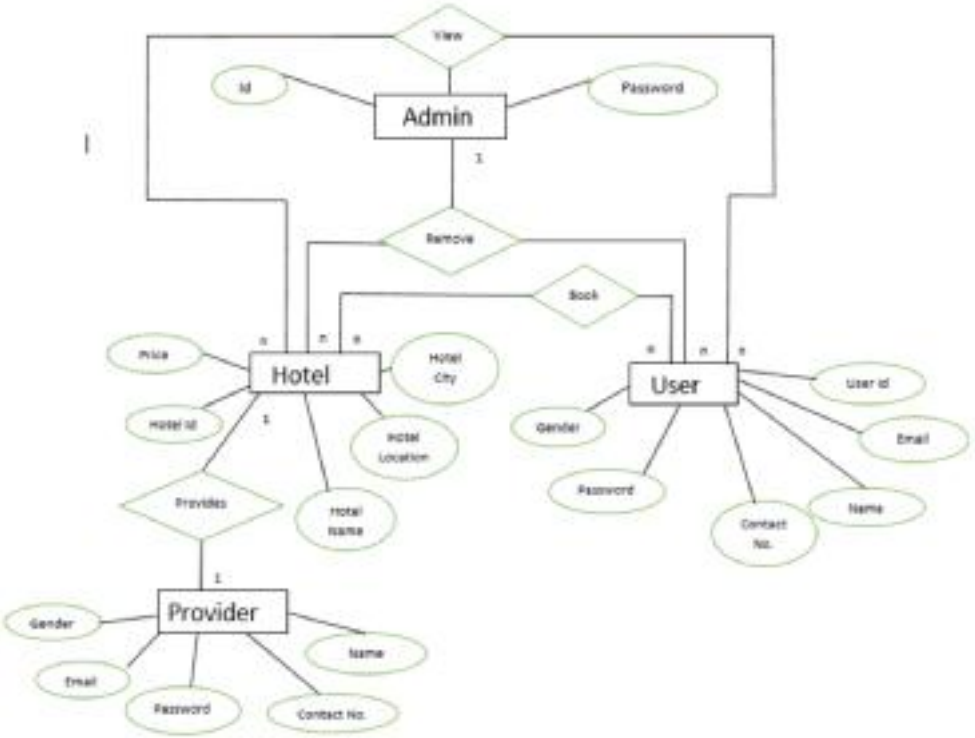


Figure:4

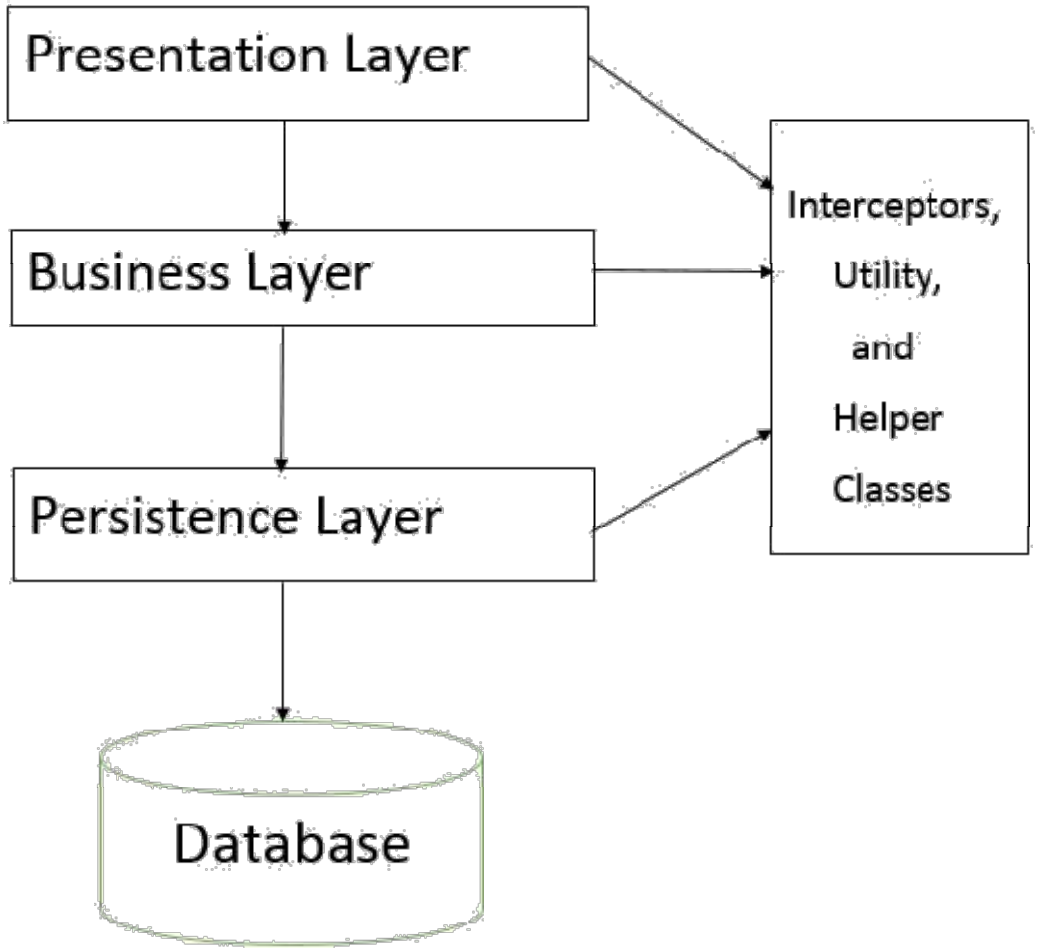
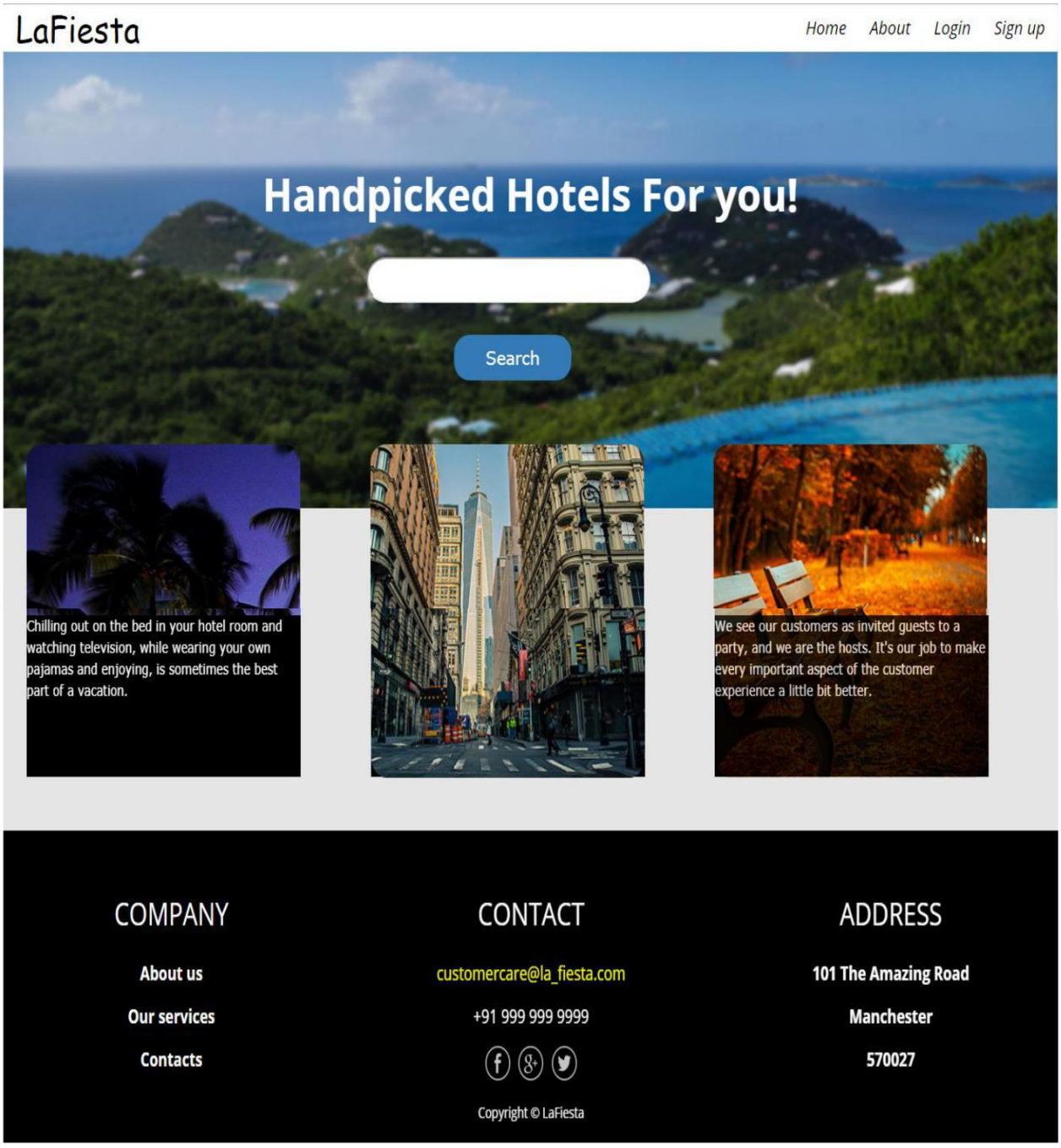

Presentation Layer
Business Layer
Persistence Layer
Database
Interceptors,
Utility,
and
Helper
Classes

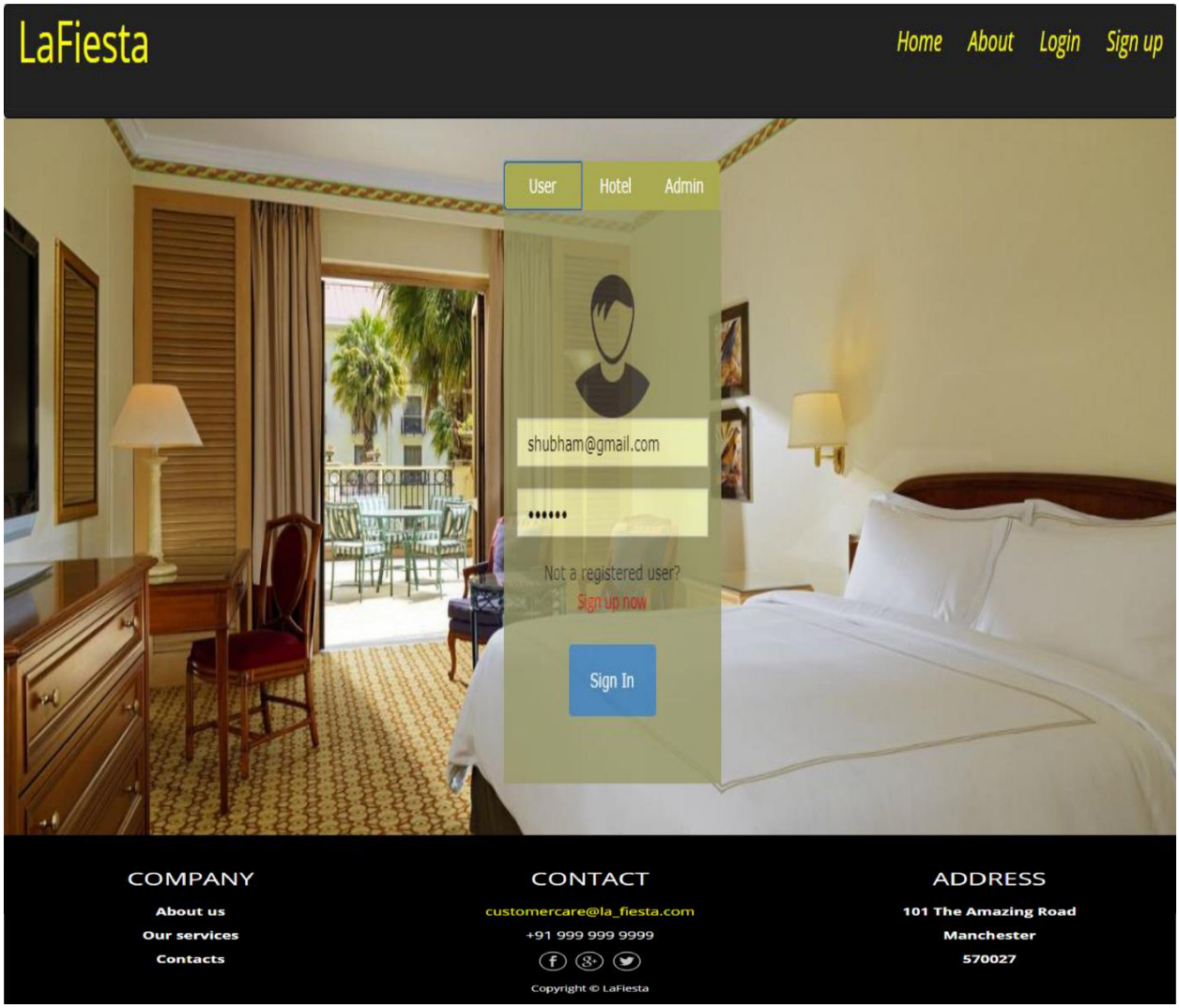
Figure:5

Chapter 4: Performance Analysis

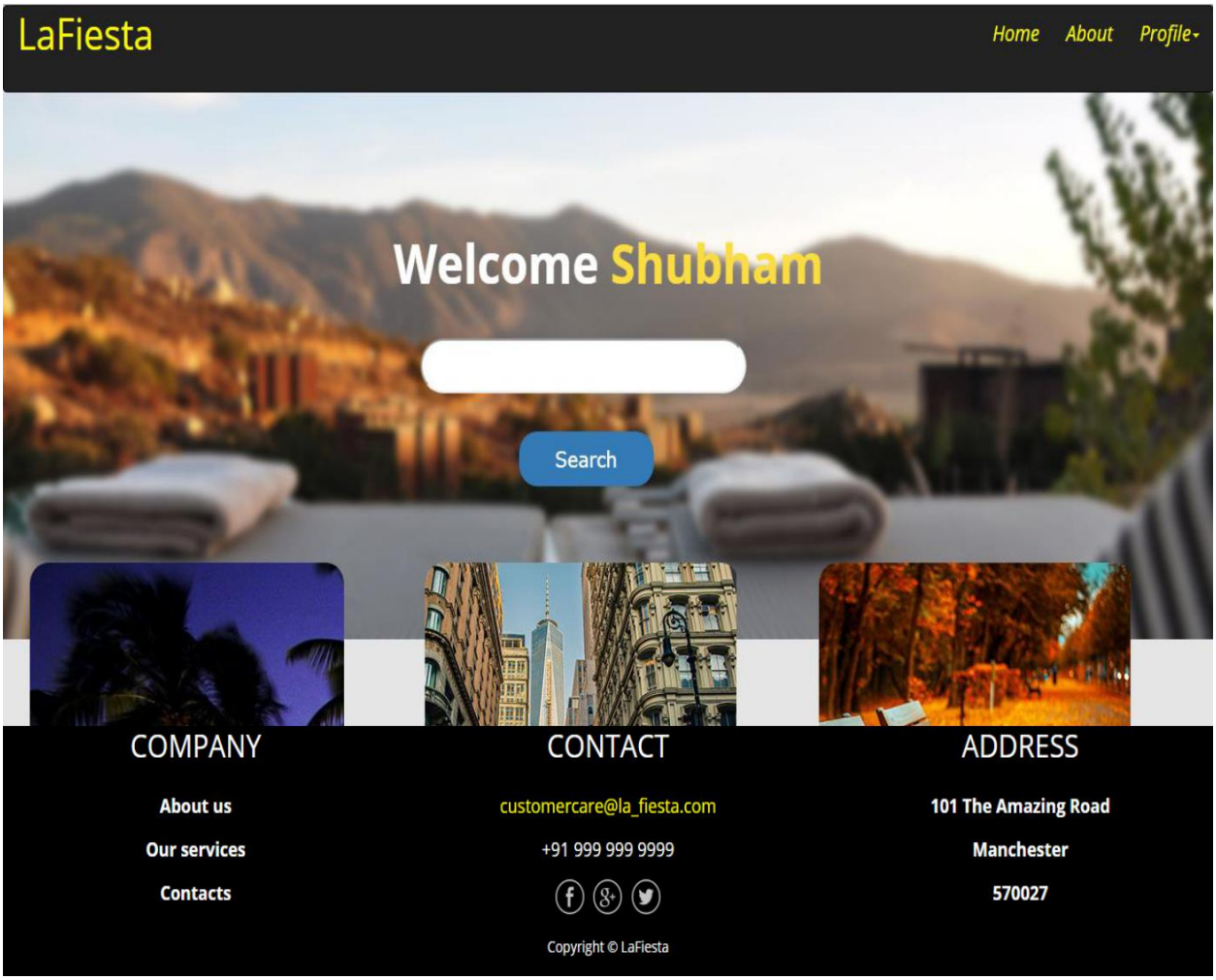
Index Page



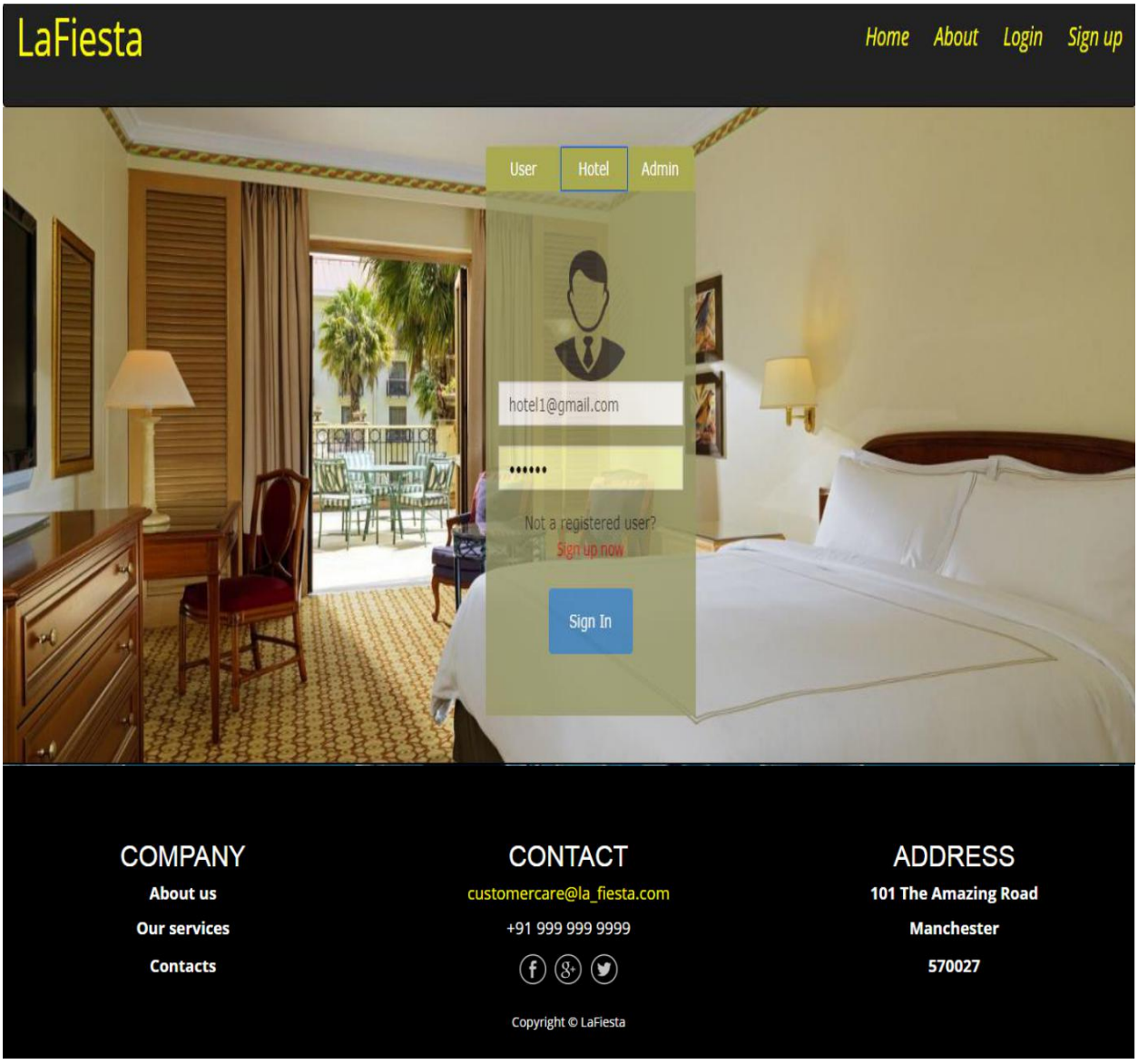
User Login Page



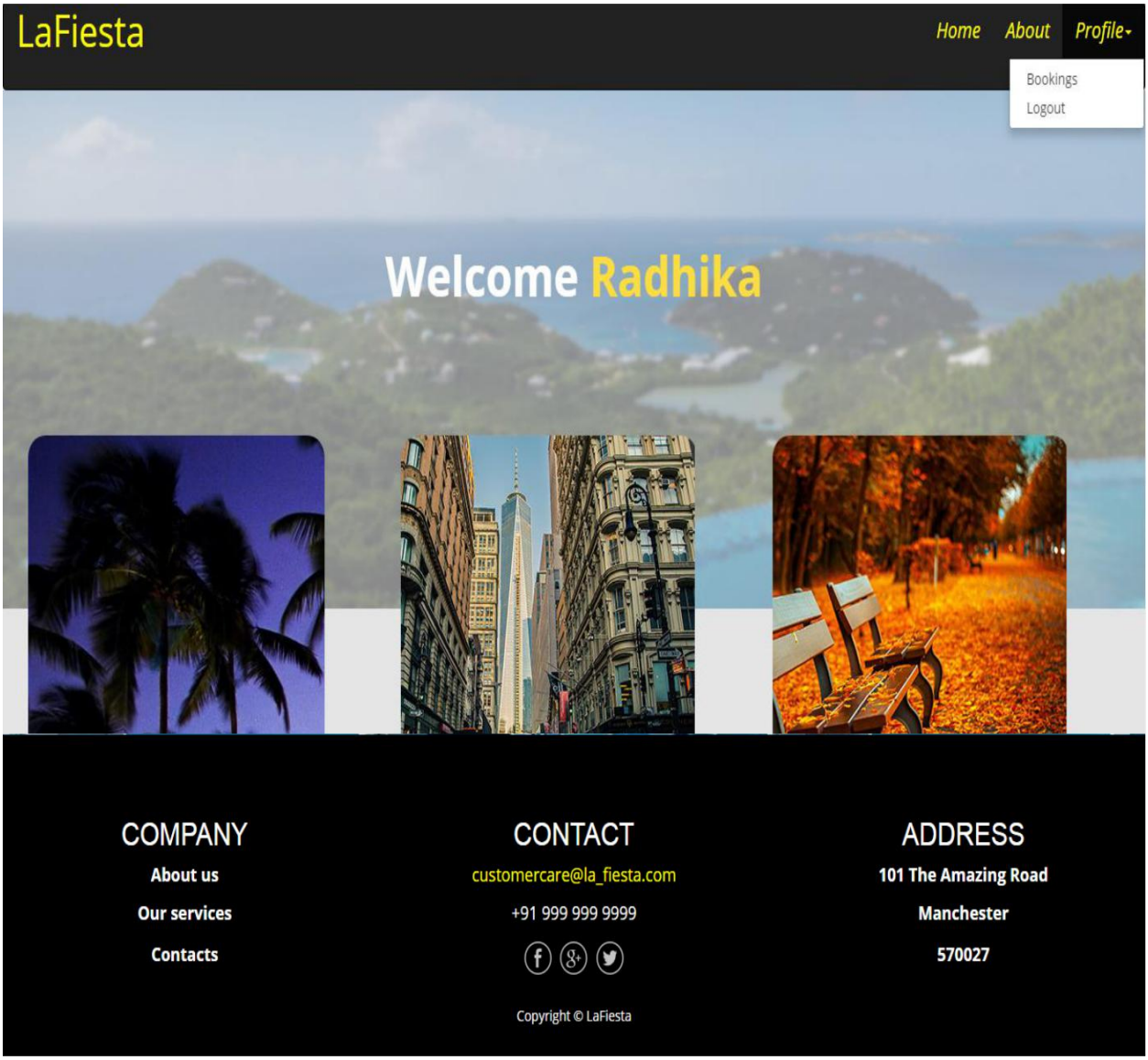
Inside User Login



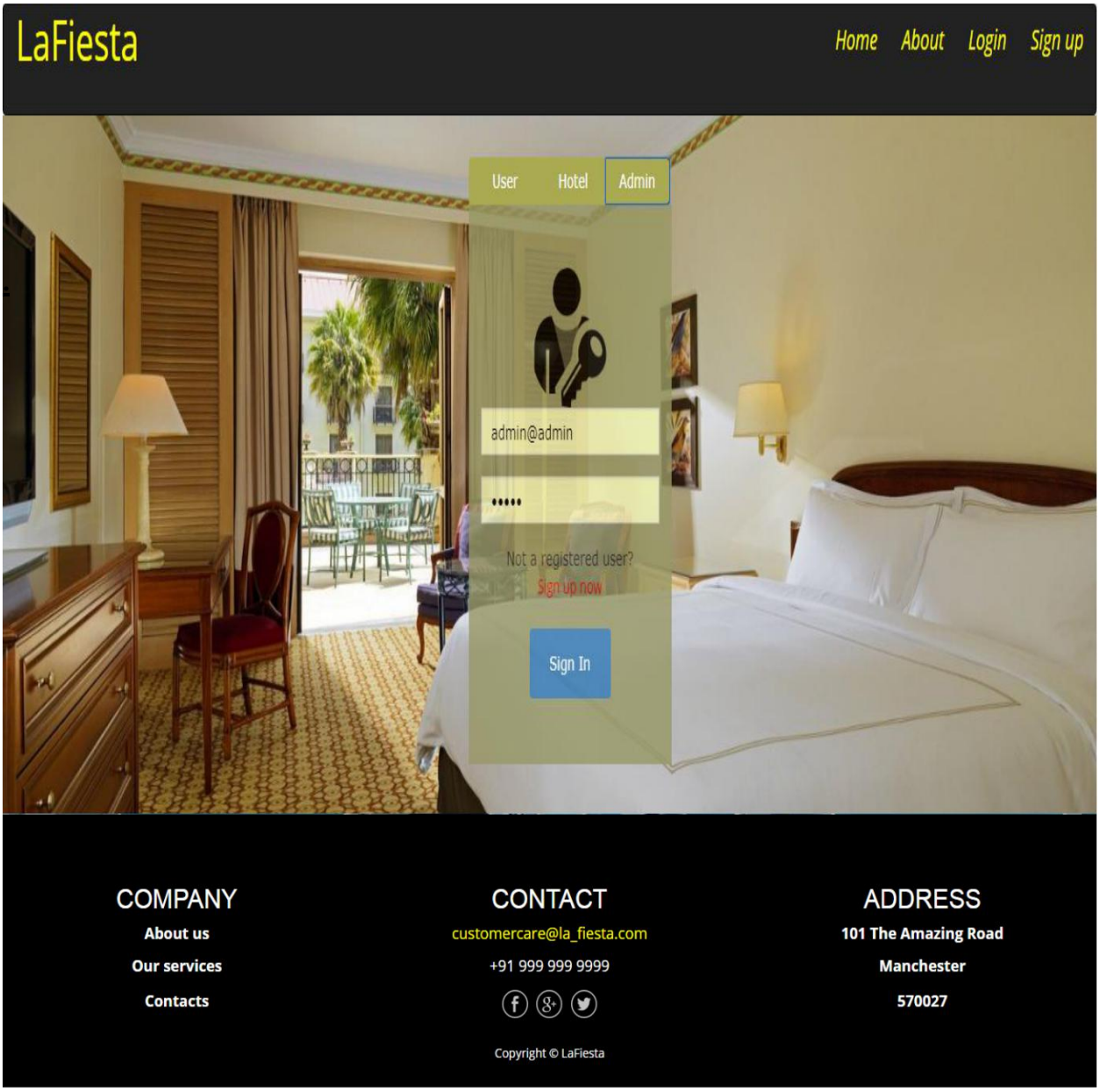
Hotel Manager Login



Hotel Manager Functionality

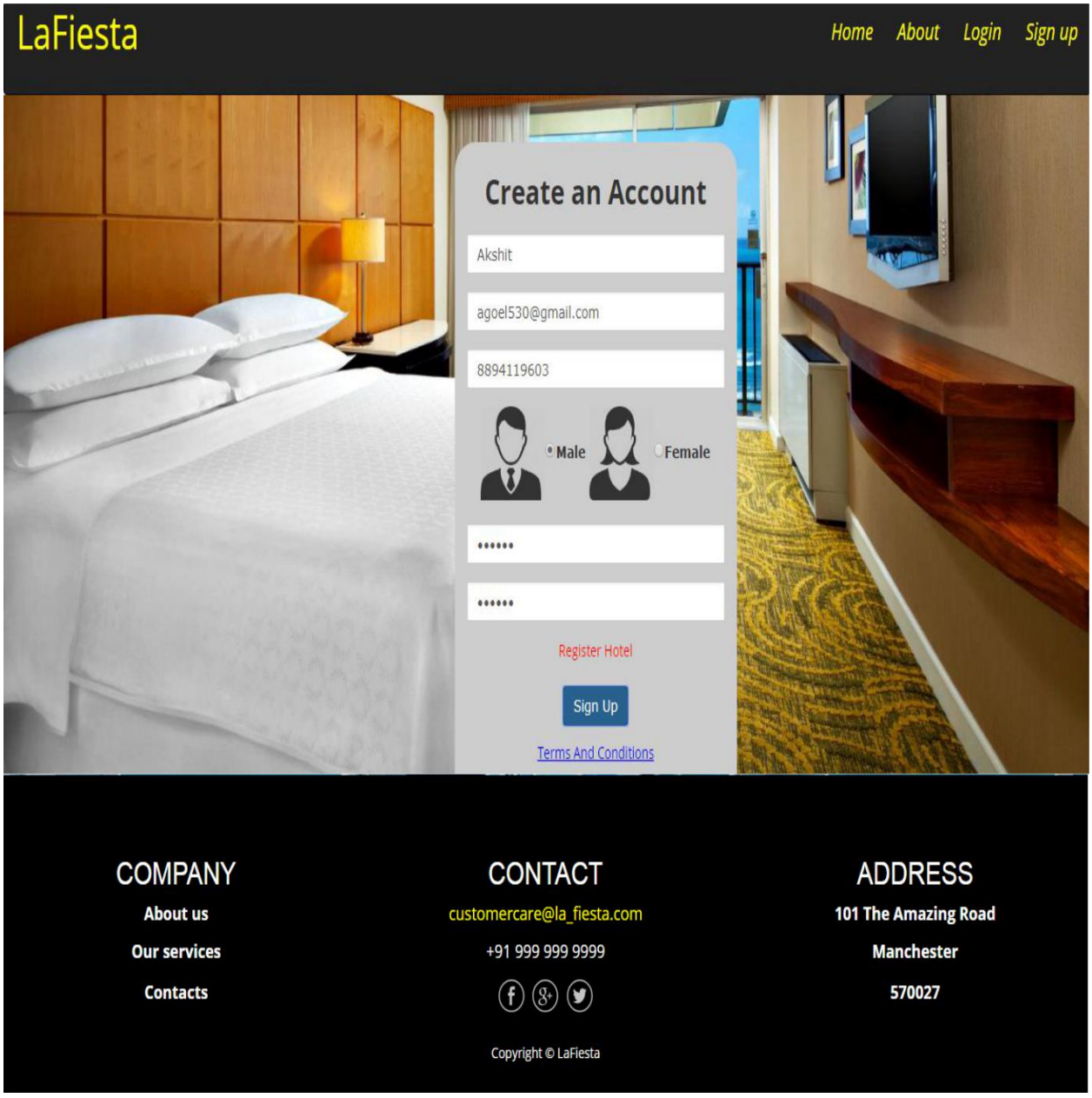


Admin Login Page

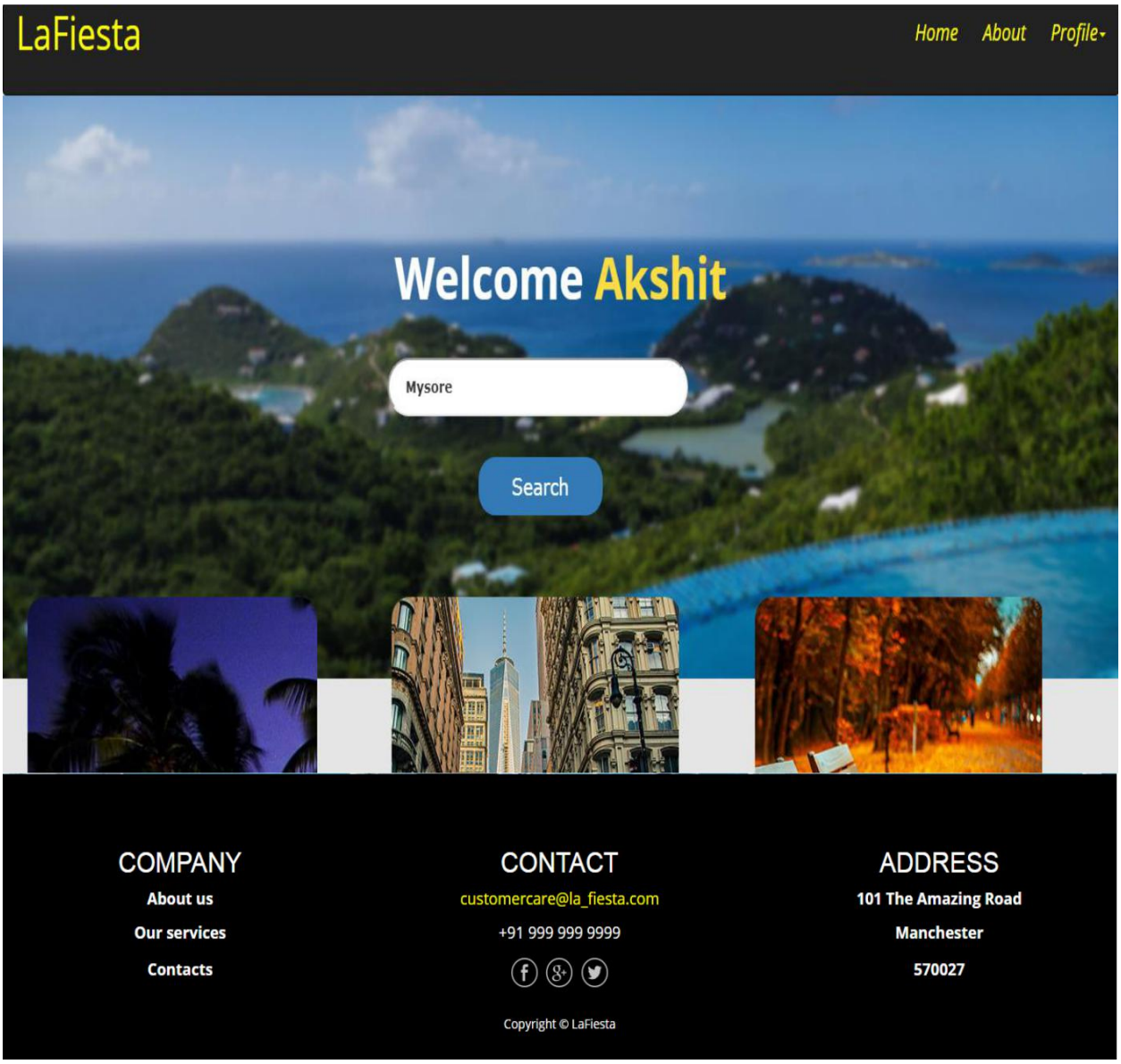


Inside Admin Portal

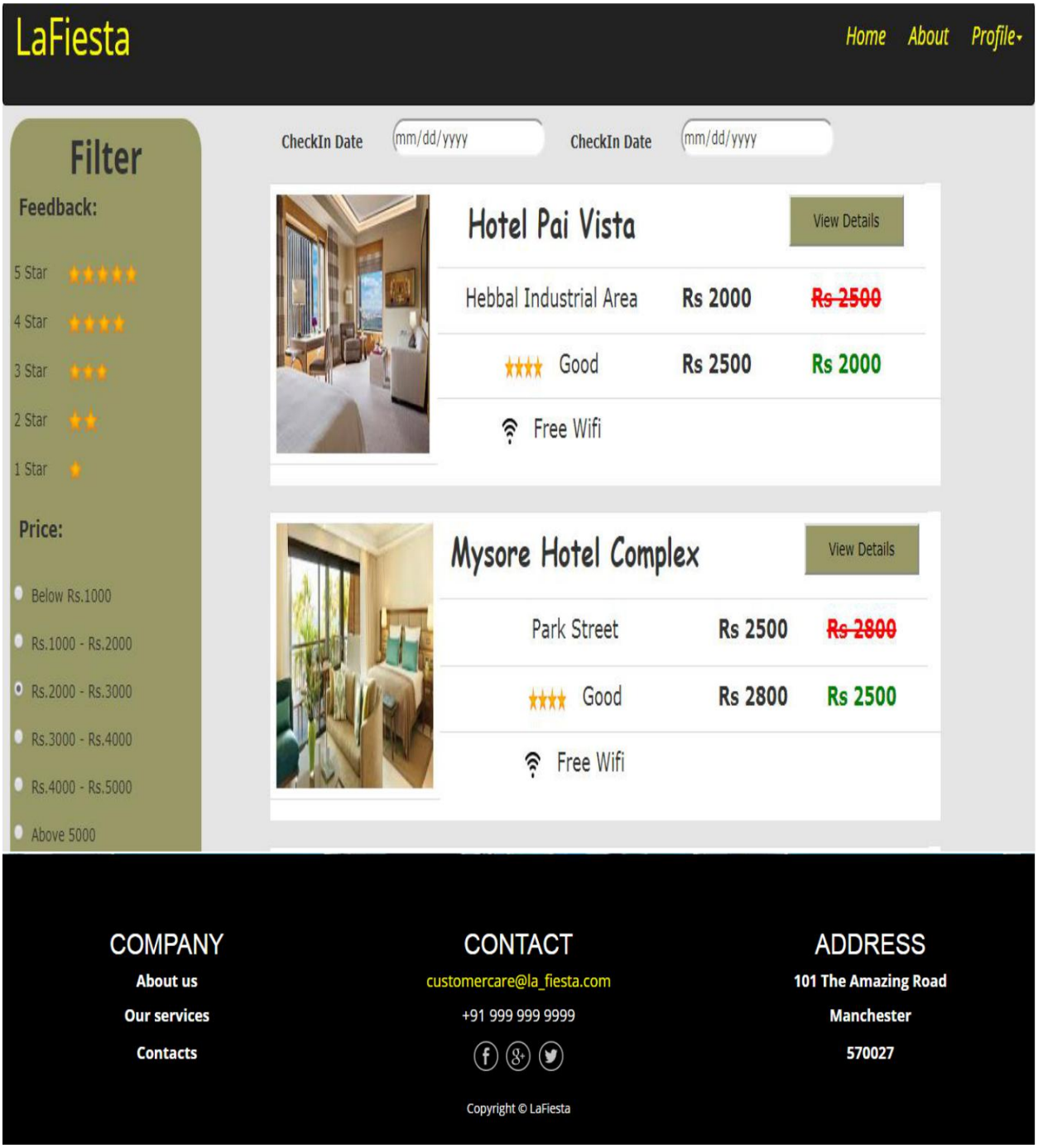
New User Sign Up



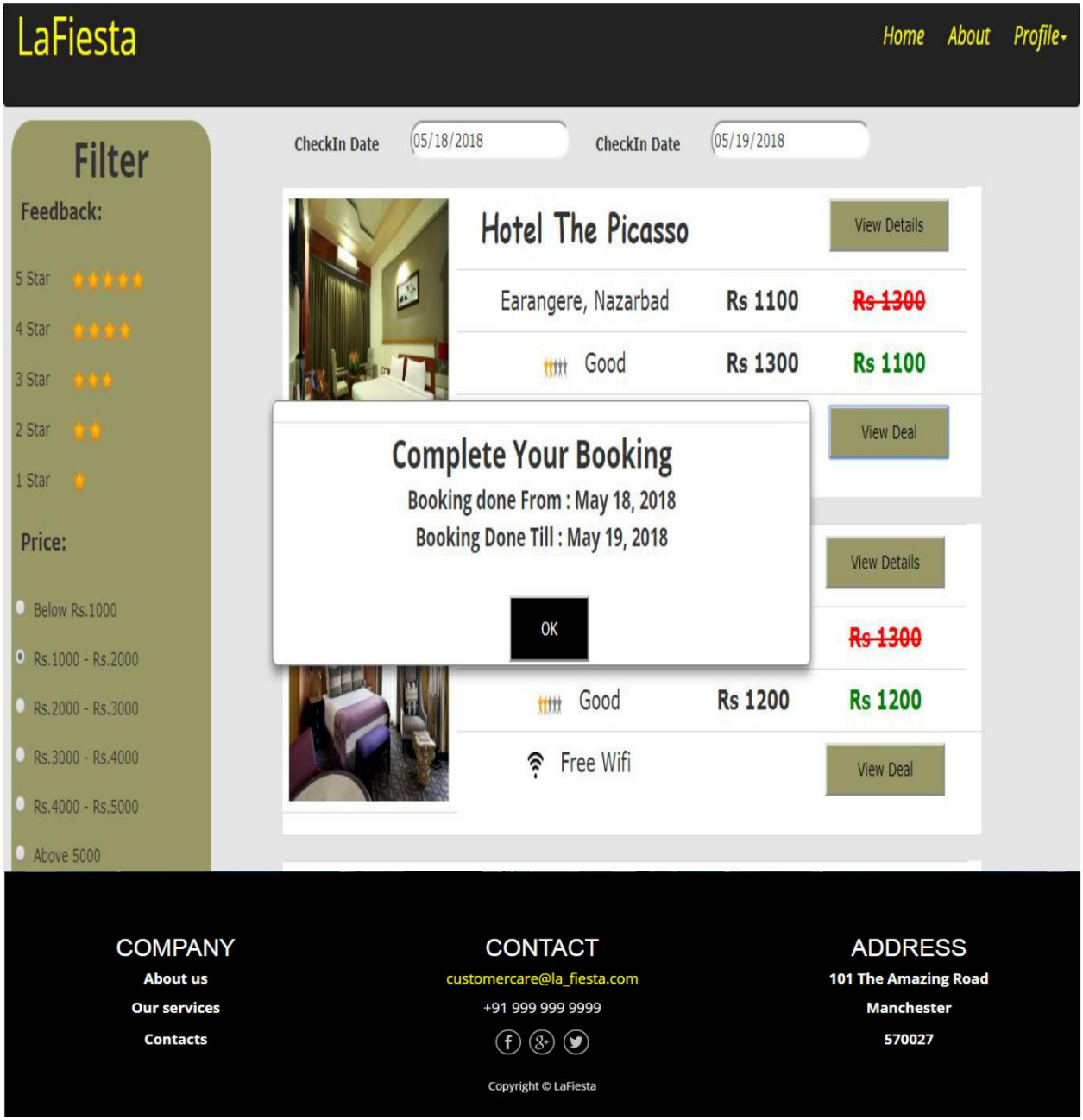
After Successful Sign Up



Search Result



Hotel Bookings



Chapter 5: Conclusion

This system will provide the booking facilities to customers. In register page customers can put their email address or phone number for booking hotel according to a particular location. Customers no need to worry for payment, they can pay at the hotel also. So, overall this system will give the customers a better user friendly environment to book online hotel .