**ABSTRACT**

The D-Ads service is a business listing that users can access online anywhere. The D-Ads is used to simplify the day to day activities of peoples who are in searching for shops and services. This makes easy to locate and find detailed information about the particular shop or services. D-Ads the Local Search Engine provides comprehensive updated information of all the Business to business and business to customer products and services.

D-Ads help buyers to display advertising of their product/services on the platform and sellers to get in touch with each other and act as an intermediary between the buyer and the seller.

**Chapter 1**

**INTRODUCTION**

The D-Ads service is a business listing that users can access online anywhere. The D-Ads is used to simplify the day to day activities of peoples who are in searching for shops and services. This makes easy to locate and find detailed information about the particular shop or services. D-Ads the Local Search Engine provides comprehensive updated information of all the Business to business and business to customer products and services.

D-Ads help buyers to display advertising of their product/services on the platform and sellers to get in touch with each other and act as an intermediary between the buyer and the seller

**1.1 Objective**

The main objective of this project is to make available the detailed information about the Product/Services at one place so that the user need not worry about the other resources to find the required information.

**Chapter 2**

**LITERATURE SURVEY**

**2.1 Existing system:**

As of now, there are no local platforms available for this kind of service to collaborate with the buyers and sellers together. When a client needs some services or needs to find some product on the local market he/she has to survey the local shops/services and fulfill the requirement that is time-consuming and sometimes takes the wrong decision. All the product/service details are not available online at this time.

**Disadvantages of Existing system**

* Time-consuming.
* Clients cannot get all the information in a short time.
* Clients have to travel for getting details about the Product/services.
* No Records are available online.

**2.2 Proposed system**

To overcome the disadvantages of the existing we are developing a new computerized system called **D-Ads** (Digital Ads)

With the help of this project, we will reduce tasks deployed for information collection to a large extent and promote the buyers online also can reduce client efforts and time during the searching of products/services. It makes available on the local shops/services online.

**Advantages of proposed system**

* Customers can get full information about the business.
* Sellers can advertise their business to the Buyers.
* Buyers can rate the services provided by sellers (To verify the sellers and their services).
* All the records are stored in a secure database.
* If somehow data may be lost then we can restore it easily.
* We can get all the information in detail at our fingertips.

**2.3 Feasibility study**

A feasibility study is an analysis used to measure the ability expectation to complete a project successfully including all relevant factors that affects it such as operational, technical and economical.

Feasibility study is used to determine positive and negative outcomes of the project.

**2.3.1 Operational feasibility**

Vendors are ready to use this application software for managing their information and it is acceptable by administration as well to make available all the information at one place.

**2.3.2 Technical feasibility**

A number of issues we have to consider while doing a technical analysis.

* Understand the different technologies involved in the proposed system.

Before commencing the project, we have to be very clear about what are the technologies that are to be required for the development of the new system.

* Find out whether the organization currently processes the required technologies.

For this project we have used Dreamweaver as a front-end editor and MySQL as back-end.

**2.3.3 Economical feasibility**

In this study the cost and benefits of proposed system are compared.

The cost of proposed system is less as compared to the maintenance cost in the existing system in which more cost involved in maintenance. The system also reduces the administrative and other user to do various jobs that single software can do. So, this system is economically feasible.

**Chapter 3**

**SOFTWARE REQUIREMENT SPECIFICATION**

**3.1 Introduction**

This document describes the software requirements, hardware requirements, purpose and the nature of the software which are developing.

**3.1.1 Purpose**

Those principle reasons this project may be on mechanize the existing system of information in which data will get from the different data sources to reduce the time of data collection the proposed system came into exist.

**3.1.2 Scope**

Online Advertising Agency System is a system design for Reservation the advertisement. The scope of the system definition the system on which the system works. The system has a wide but out of the scope system is not work. The system is a made up of web application. But for web application computer must have window operating system and APACHE server and flash player installed to run this system. This system is used by all type of users. Users can register online on this site. All his detail can also see by them.. In this website you can create your login. Here u can show all advertisement.

**3.2 System Specification**

**3.2.1 Hardware requirements**

Processor : Pentium 4 or Above

RAM : 512 MB or Above

HDD : Minimum 20 GB

**3.2.2 Software requirements**

Operating System : Windows XP or above

IDE : Microsoft Visual Code

Designing Tools : HTML, CSS, Bootstraps

Server Side Programming language : PHP

Server : WAMP Server

Back End : MySQL

Web Browser : Google Chrome, Internet Explorer etc.

**3.3 Requirement specification**

**3.3.1 Functional requirements**

* **Administrative:**
* **Visitors (Users):**
* **Vender (Future Enhancement):**
* **Administrative:**

Digital Ads in the role of administrator will manage all day to day business activities. As well as it verify and organize the Sellers and their services.

**Responsibilities of Administrative**

* Add Verified Sellers and their Services to the Platform.
* Organize the sellers in the searched list as per their ratings.
* Ensure operations adhere to policies and regulations.
* Monitor day to day business activities.
* See all the feedbacks from the Vendors and Users and Make the user-friendly system.
* **Visitors (Users):**

Visitors as buyers can seek the products information available on the D-Ads portal if they want it for further clarification they can contact to the particular seller as related.

**Responsibilities of Visitors**

* Buyers can view the Products/services available on the D-Ads Platform.
* Check for the verified Sellers.
* Rate the sellers for their services.
* Contact the sellers from the platform for further information.
* Give genuine feedback that makes an even more useful system
* **Vender (Future Enhancement):**

A visitor satisfaction is depends on services provided by the system administrative.

**Role of Venders**

* Vender can have a separate panel to control advertise their product/service.
* Vender can add their product information.
* Vender panel has different plans with different features.
* Registered users can contact the vendor for more detail about the product/services.
* Vender can send promotional SMS to registered users for advertising their products/services.

**3.3.2 Nonfunctional requirements**

The non functional requirements describe the aspects of the system that do not relate to execution of the system.

Below are the nonfunctional requirements of the system.

* **Usability:** This defines how the system is used easily by the user. How system will create user friendly relation with the user. Here in D-Ads, this software is totally user friendly GUI based system. Anyone can get idea of how to use system by seeing itself.
* **Security:** This gives surety about the system access security by the unauthorized person. This system provides different login for all level of users.
* **Reliability:** It defines how the software will work without failure for a given time. D-Ads will respond properly with time period.
* **Performance:** performance will measure the response time of a system and accuracy of the result. D-Ads will respond at accurate time of request.
* **Modifiability:** this required to make the changes into the system accordingly. D-Ads is modifiable software, it can modify as required.

**3.4 Tools and Technologies used**

**3.4.1 Tool**

The tools used to develop this software are Android and Dreamweaver.

**Dreamweaver**:

Dreamweaver is an IDE (Integrated Development Environment). This is a tool that is designed for the web development field is used to create a particular website or application for different types of devices.

It helps huge numbers markup dialects including HTML, XML, CSS What's more JavaScript. Dreamweaver might have been formed What's more distributed Eventually Tom's perusing Macromedia clinched alongside 1997. The numerous characteristics for

Dreamweaver make it a versant web altering tool, the place it may be to making complex or basic destinations.

**Android Studio:**

Android studio is the IDE (Integrated Development Environment), for android application development. It is based on the IntelliJ IDEA, a java integrated development environment for software.

To support application development within the android operating system, Android studio uses a Gradle-based build system, emulator, code templates, and Github integration. Every project in android studio has one or more modalities with source code and resource files. These modalities include Android app module, library module, and Google App engine modules.

**3.4.2 Technologies**

**JSP:**

Java server Pages (JSP) may be a server-side modifying innovation that empowers the formation of dynamic, stage autonomous technique to building web based provision. JSP bring entry of the whole crew from claiming java API’s including those JDBC API’s to right enterprise databases.

JSP was developed by sun Microsystems and is an improved version of java servlets. JSP’s are normal HTML pages with embedded java code. To provide a JSP file, developers need a JSP engine, which is connected to a web server. The JSP page is then compiled into a servlet, which is handled by a servlet engine. The servlet engine then loads the servlet class and executes it to create dynamic HTML, which is then sent to the browser.

**3.4.3 Database**

**MySQL:**

MySQL is an open source relational database management system. MySQL runs on virtually all platforms including Linux, UNIX, and Windows. Although it can be used in a wide range of applications. MySQL is most often associated with web applications.

**Java Data Base Connectivity (JDBC):**

Java Data Base Connectivity is a standard SQL database access interface providing uniform access to a wide range of relational databases. It also provides a common base on which higher level tools and inter faces can be built. This comes with an “ODBCBridge”. That bridge is a library, which implements JDBC in terms of ODBC standard API.

Using JDBC, it is easy to send SQL statements to virtually any relational database. In other words, with the JDBC API, it isn't necessary to write one program to access a Sybase database, another program to access a MySQL database, another program to access an Informix database, and so on. One can write a single program using the JDBC API, and the program will be able to send SQL statements to the appropriate database. And, with an application written in the Java programming language, one also doesn't have to worry about writing different applications to run on different platforms. The combination of Java and JDBC lets a programmer write it once and run it anywhere.

**JDBC does the following things:**

* Establish a connection with a database.
* Send SQL statements.
* Process the results.

**Tomcat Server:**

JSP programs are executed by a JSP virtual machine that runs on a web server. Therefore, you need to have access to a JSP virtual machine to run your JSP program. You can download and install a JSP virtual machine.

One of the most popular JSP virtual machine is Tomcat, and it is downloadable at No charge from the Apache website. Apache is also a popular web server that you can also download at no cost. You will also need to have the Java Development Kit (JDK) installed on your computer, which you probably installed when you work with java.

**Wamp server:**

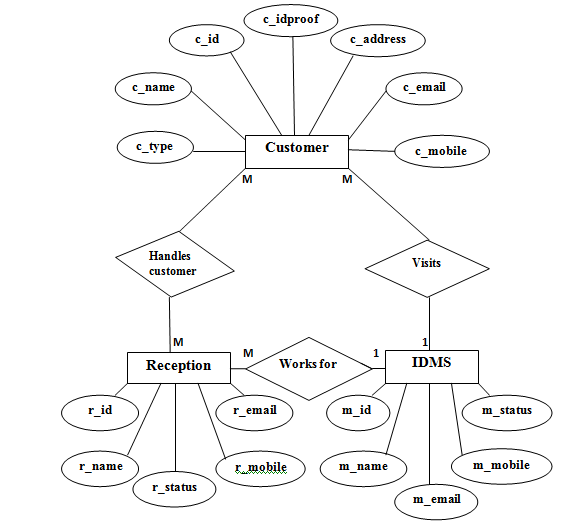
WAMP stands for Windows, Apache, MySQL and PHP/Python/Perl.

Wamp act like virtual server on your computer, it allows you to test your website features on your computer without hosting it on web. In wamp you use Microsoft windows as an operating system, Apache as a web server, MySQL as a database and PHP/Python/Perl as a scripting language.

**Chapter 4**

**DESIGN DOCUMENTS**

**4.1 ER-Diagram**

****

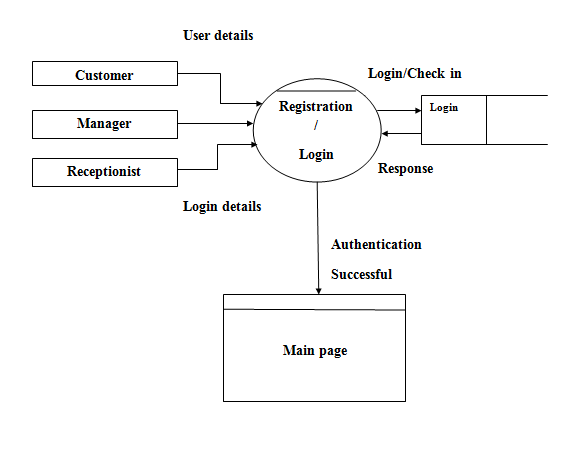
**Fig.2 ER-Diagram: Entity relationship diagram**

**4.2 Data-flow-diagram**

The Data Flow Diagram routes the flow of information for the system, by using the symbols like rectangle, circle and arrows, short text labels to show the data flow routes between each destination.

The data flow diagrams are functionally divided into, Zero level, First level and Second level data flow diagrams.

1. **Zero level DFD**

****

**Fig.3 Zero level data flow diagram**

1. **One level DFD**

**Manager**

**Receptionist**

**Customer**

**Database**

**Database**

**Database**

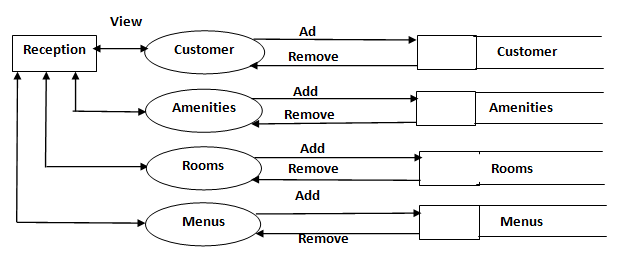
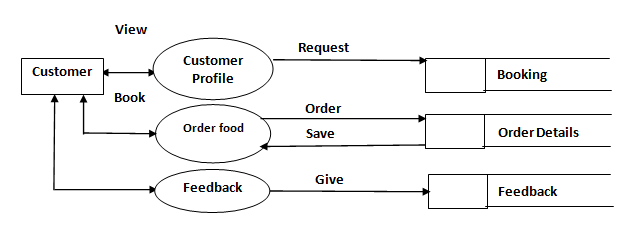
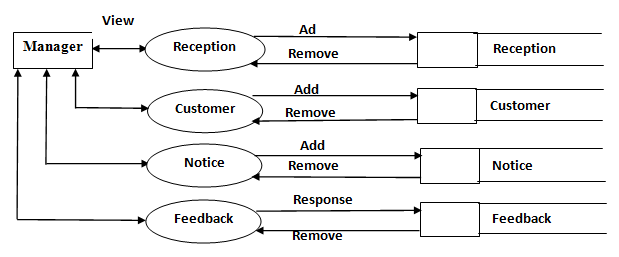
**Manages**

**Deals with**

**Can**

**Fig.4 One level data flow diagram**

1. **Two level DFD**

****

**Fig. 5 Two level data flow diagram**

**4.3 DATA DICTIONARY**

After carefully understanding the requirements of the client the entire data storage requirements are divided into below tables.

1. **b\_Category**

|  |  |  |
| --- | --- | --- |
| **Fields** | **Type** | **Description** |
| c\_id | int(100) | Primary key |
| c\_name | varchar(100) |  |
| c\_desc | varchar(255) |  |
| is\_Active | int(10) |  |
| Created\_dt | varchar(100) |  |

1. **b\_details**

|  |  |  |
| --- | --- | --- |
| **Fields** | **Type** | **Description** |
| b\_id | int(100) | Primary key |
| sc\_id | int(100) |  |
| b\_name | varchar(100) |  |
| b\_desc | varchar(225) |  |
| b\_estdate | varchar(100) |  |
| b\_mobile | varchar(100) |  |
| b\_altmob | varchar(100) |  |
| b\_email | varchar(100) |  |
| b\_website | varchar(100) |  |
| b\_address | varchar(100) |  |
| b\_city | varchar(100) |  |
| created\_dt | varchar(100) |  |

1. **b\_photos**

|  |  |  |
| --- | --- | --- |
| **Fields** | **Type** | **Description** |
| bp\_id | int(100) | Primary key |
| b\_id | int(100) |  |
| Bp\_photolink | varchar(100) |  |
| Bp\_slider1 | varchar(100) |  |
| Bp\_slider2 | varchar(100) |  |
| Bp\_slider3 | varchar(100) |  |
| created\_dt | varchar(100) |  |

1. **b\_subcat**

|  |  |  |
| --- | --- | --- |
| **Fields** | **Type** | **Description** |
| sc\_id | int(100) | Primary key |
| c\_id | varchar(100) |  |
| sc\_name | varchar(100) |  |
| sc\_desc | varchar(100) |  |
| sc\_icon | varchar(100) |  |
| is\_Active | int(10) |  |
| created\_dt | varchar(100) |  |

1. **contact\_us**

|  |  |  |
| --- | --- | --- |
| **Fields** | **Type** | **Description** |
| cs\_id | int(100) | Primary key |
| b\_id | int(100) |  |
| p\_name | varchar(100) |  |
| p\_desc | varchar(100) |  |
| p\_photos | varchar(100) |  |
| p\_status | varchar(100) |  |
| created\_dt | varchar(100) |  |

1. **faq**

|  |  |  |
| --- | --- | --- |
| **Fields** | **Type** | **Description** |
| id | int(100) | Primary key |
| faq\_question | varchar(300) |  |
| faq\_answer | varchar(1500) |  |
| created\_dt | varchar(20) |  |
| last\_updated | varchar(20) |  |
| status | varchar(10) |  |

1. **feedback\_details**

|  |  |  |
| --- | --- | --- |
| **Fields** | **Type** | **Description** |
| f\_id | int(100) | Primary key |
| feedback\_id | varchar(100) |  |
| fname | varchar(100) |  |
| lname | varchar(100) |  |
| email | varchar(100) |  |
| phone | varchar(100) |  |
| msg | varchar(100) |  |
| status | varchar(20) |  |
| created\_dt | varchar(100) |  |

1. **feedback\_reply**

|  |  |  |
| --- | --- | --- |
| **Fields** | **Type** | **Description** |
| feedback\_id | varchar(100) | Primary key |
| name | varchar(100) |  |
| email | varchar(100) |  |
| phone | varchar(100) |  |
| msg | varchar(1000) |  |
| status | varchar(20) |  |
| created\_dt | varchar(100) |  |

1. **lgn\_tbl**

|  |  |  |
| --- | --- | --- |
| **Fields** | **Type** | **Description** |
| lgn\_id | int(100) | Primary key |
| username | varchar(100) |  |
| password | varchar(100) |  |
| email | varchar(100) |  |
| u\_type | varchar(100) |  |
| s\_ques | varchar(100) |  |
| s\_ans | varchar(100) |  |
| u\_status | varchar(100) |  |

**10. login\_info**

|  |  |  |
| --- | --- | --- |
| **Fields** | **Type** | **Description** |
| id | int(10) | Primary key |
| name | varchar(100) |  |
| timestamp | varchar(100) |  |
| ip\_address | varchar(100) |  |
| mac\_address | varchar(100) |  |
| status | varchar(50) |  |

1. **otp**

|  |  |  |
| --- | --- | --- |
| **Fields** | **Type** | **Description** |
| id | int(10) | Primary key |
| username | varchar(100) |  |
| otp | int(10) |  |
| status | varchar(50) |  |
| timestamp | varchar(50) |  |
| hashvalue | varchar(100) |  |

1. **product\_details**

|  |  |  |
| --- | --- | --- |
| **Fields** | **Type** | **Description** |
| p\_id | int(100) | Primary key |
| b\_id | int(100) | Foreign key |
| p\_name | varchar(100) |  |
| p\_desc | varchar(100) |  |
| p\_photos | varchar(100) |  |
| p\_status | varchar(100) |  |
| created\_dt | varchar(100) |  |

1. **vender\_details**

|  |  |  |
| --- | --- | --- |
| **Fields** | **Type** | **Description** |
| v\_id | int(100) | Primary key |
| v\_code | varchar(100) |  |
| v\_fname | varchar(100) |  |
| v\_lname | varchar(100) |  |
| v\_email | varchar(100) |  |
| v\_mobile | varchar(100) |  |
| v\_alt\_mob | varchar(100) |  |
| v\_idproof | varchar(100) |  |
| v\_photos | varchar(100) |  |
| v\_reg\_date | varchar(100) |  |
| v\_type | varchar(100) |  |
| v\_status | int(10) |  |

1. **Vender\_type**

|  |  |  |
| --- | --- | --- |
| **Fields** | **Type** | **Description** |
| v\_id | int(10) | Primary key |
| v\_type | varchar(100) |  |
| Vtype\_desc | varchar(225) |  |
| Created\_dt | varchar(100) |  |
| Status | int(10) |  |

**4.4 Use case diagram**

* **Use case diagram**

**Manager**

**Customer**

**Receptionist**

**Fig. 6 System use case diagram**

* **Manager use case diagram**

**Manager**

**Fig. 7 Manager level use case diagram**

* **Customer use case diagram**

**Customer**

**Fig. 8 Customer level use case diagram**

* **Receptionist use case diagram**

**Fig. 9 Receptionist level use case diagram**

**4.5 Activity diagram**

* **Registration page activity diagram**

**Get details**

**Validate details**

**Accepted**

**Rejected**

**[Enter registration details]**

**Submit**

**No**

**Yes**

**[Registered successfully]**

**Fig. 10 Registration page activity diagram**

* **Login page activity diagram**

**Get details**

**Validate details**

**Accepted**

**Rejected**

**[Enter login details]**

**Submit**

**No**

**Yes**

**[Login successful]**

**Fig. 11 Login page activity diagram**

* **Manager activity diagram**

**Get data**

**Validate data**

**Customer details**

**Send notice**

**Booking details**

**Feedback**

**Add menus**

**Add/Remove Receptionist**

**Add/Remove Rooms**

**Receptionist details**

**Bill/Payment**

**[Enter login details]**

**No**

**Yes**

**[View activities] [Process activities]**

**Fig. 12 Manager Activity diagram**

* **Customer activity diagram**

**Get data**

**Validate data**

**Profile**

**Feedback**

**My booking**

**Food order**

**My order**

**Room details**

**Bill / Payment**

**Notice**

**[Enter login details]**

**Submit**

**No Yes**

**[View activities] [Process activities]**

**Fig. 13 Customer activity diagram**

* **Receptionist activity diagram**

**Get data**

**Validate data**

**Profile**

**Room details**

**Order details**

**Feedback**

**Booking details**

**Add menus**

**Notice**

**Customer details**

**Bill/Payment**

**[Enter login details]**

**No Yes**

**[View activities] [Process activities]**

**Fig. 14 Receptionist activity diagram**

**Chapter 5**

**VERIFICATION AND VALIDATION**

**6.1 Introduction**

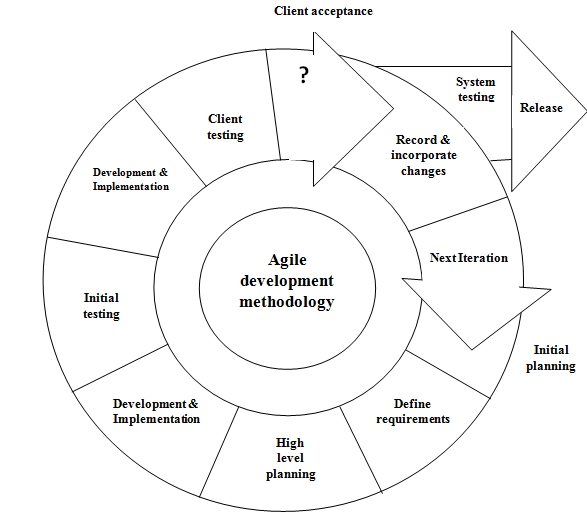
The verification and validation can also referred as software quality control. It is the process of checking whether the software fulfills its purpose and meets all the requirements.

**6.2 Methodology used**

**Agile methodology**

While each of the agile mythologies is unique in its specific approach, they all share a common vision and core values. They all fundamentally incorporate iterations and the continuous feedback that it provides to successively refine and deliver a software system. They all involve continues planning, continues testing, continues integration, and other forms of continues evolution of software. They are lightweight, especially compared to traditional waterfall style processes, and inherently adaptable. What is more important about agile method is that they all focus on empowering people to collaborate and make decisions together quickly and effectively.

* Test first programming
* Refactoring
* Continues iterations
* Simple design
* Coding standards

****

**Fig. 15 Agile model.**

**6.3 Testing technology**

**Black Box Testing**

Black Box Testing is one of the approaches to the testing scenario or we can call it as the type of testing. In block box testing will do the test engineers, where the framework functionality will be checked. In the black box testing code will not visible to the test engineers. Here we are checking the missing function if any interface error, we will be checking the performance of the software and its behaviors. Here we also check any external error regarding the database access, etc.

**White Box Testing**

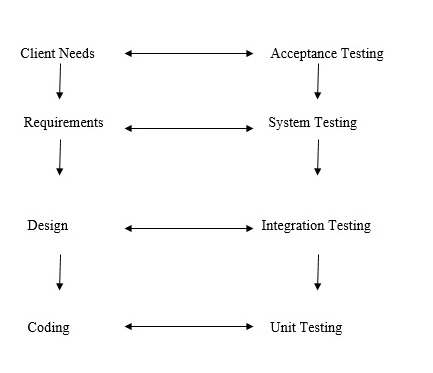
White Box Testing is also called as Code testing. This testing is done by the developer while developing the code for the software. This testing is concerned with the developer where the code will be visible to perform the testing, this testing involves step by step procedure to perform the testing .In this testing, we check all the independent condition and also the path in the code whether all the related path is executed at least once or not.

We also check whether the necessary loop and the condition were checked are not ,here we are checking the boundaries of the possibilities for the logical condition or decision for the true and false .here we will check the data structure is valid or not.

Ensure whether the possible validity check and validity lookups have been provided validate data entry.

**7.2Testing Strategies:**

**Levels of testing:**



**Fig. 16 Levels of testing.**

The different testing is carried out which reflects its effectiveness and Efficiency of different phases of the software development where these test help to uncover the error in the corresponding phase.

There are two general strategies for testing software. There are as follows

**Code Testing:**

Code Testing where we will check the program logic is correct or not. To do this we are using the test case which is developed by the tester in order to check each and every path of the code logic as well as the flow of the program.

**Specification Testing:**

In this testing approach, we are using for the specification that is required to test the various behaviors of the application in various conditions. To perform the specification testing developer writes the testcase with all the combination of condition to perform the specification testing.

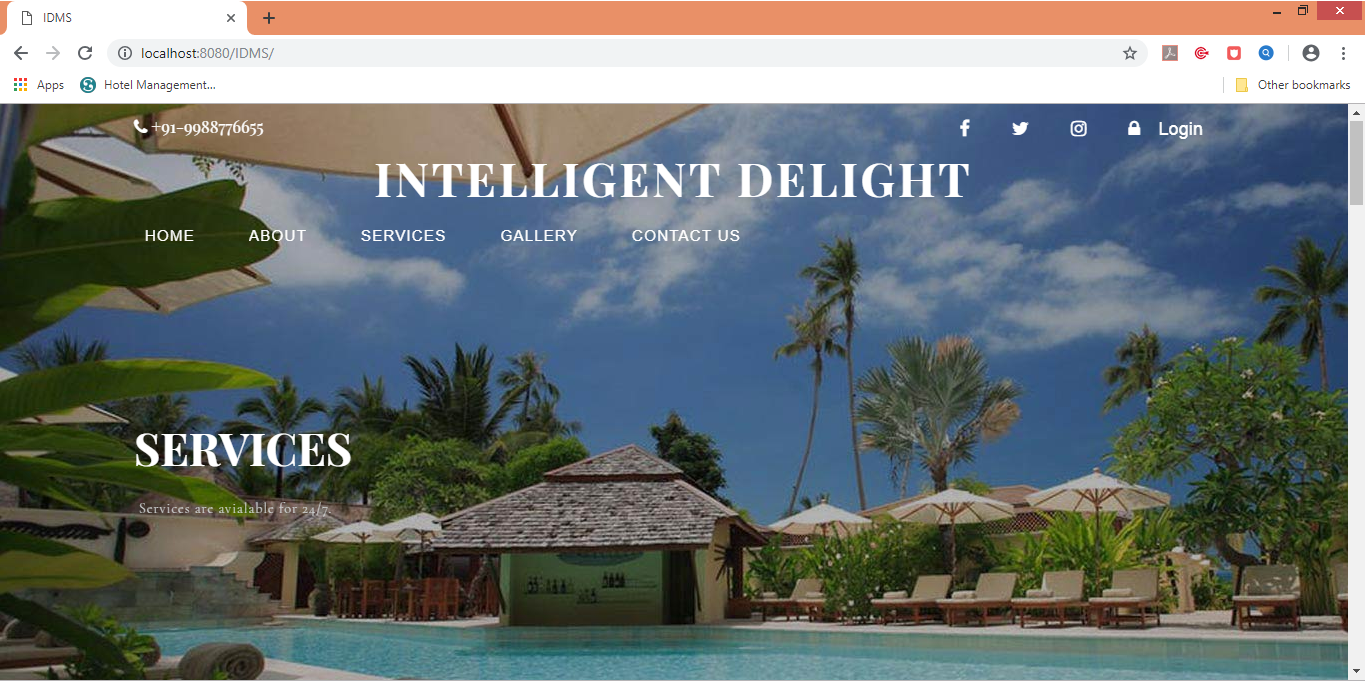
**Integration Testing:**

Integration Testing is performed after the unit testing, where the testing can involve the integration i.e. all the unit which are done with the testing those will be combined, integrated with the other entire module and performs the testing order to maintain the consistency.

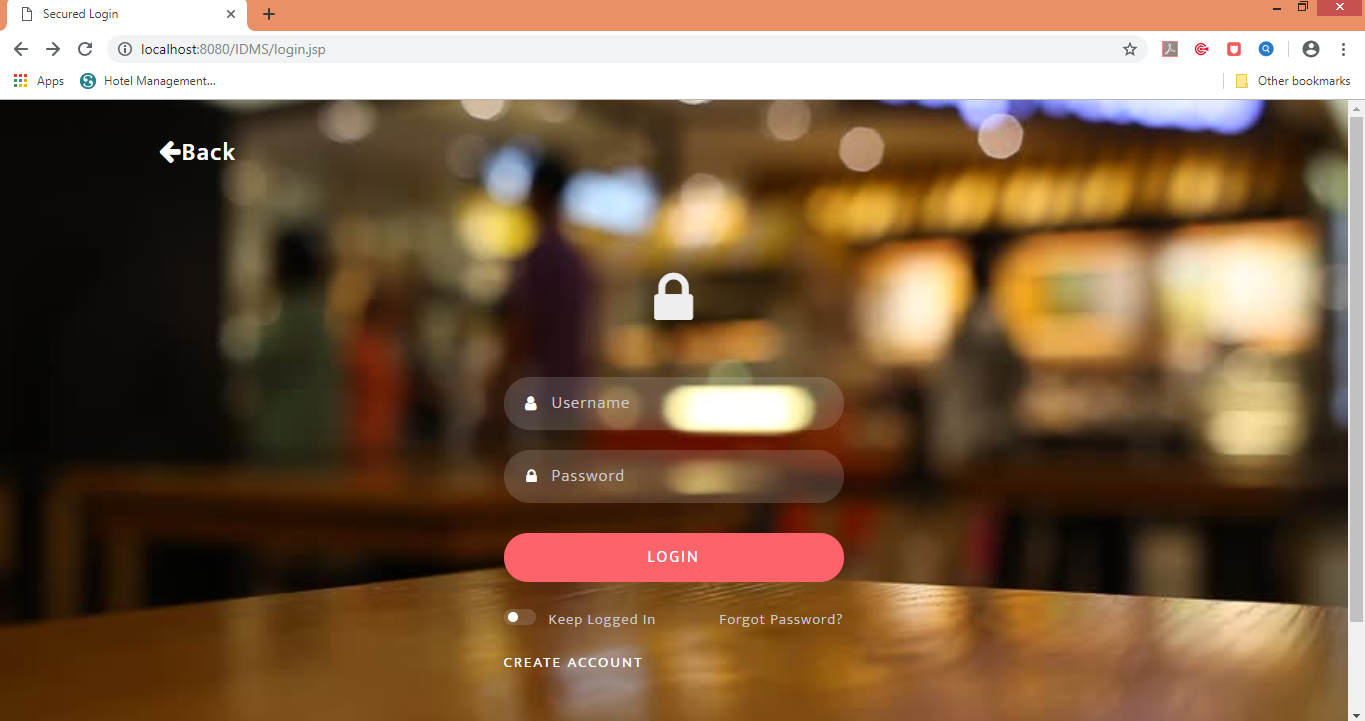
**Chapter 6**

**IMPLEMENTATION RESULT**

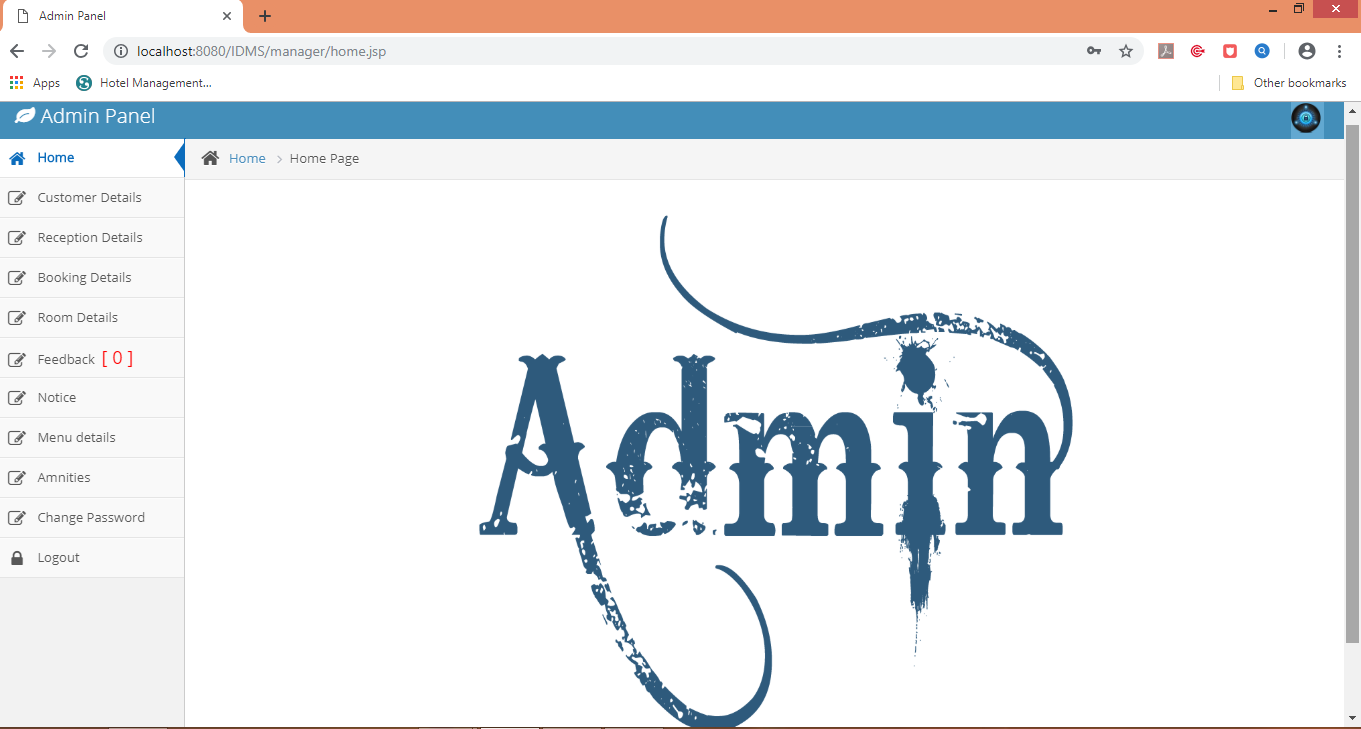
1. **Home page**

****

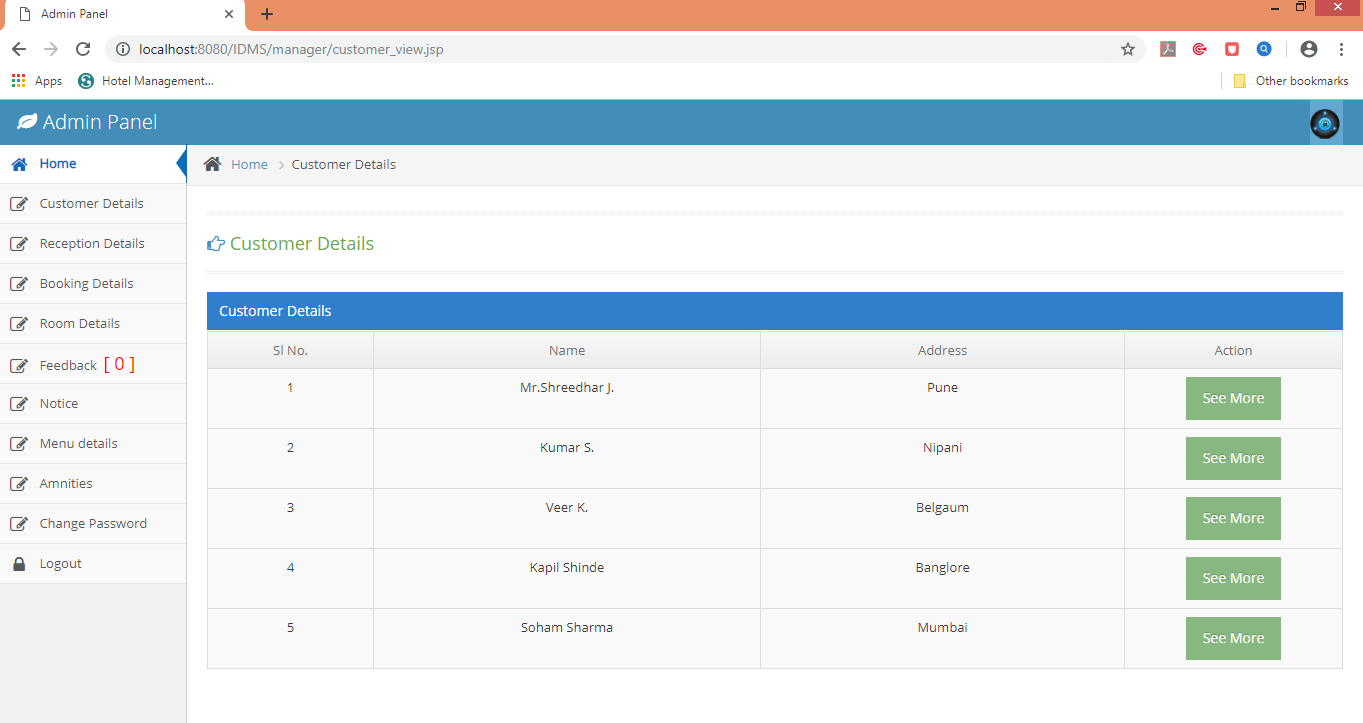
1. **Login page**

****

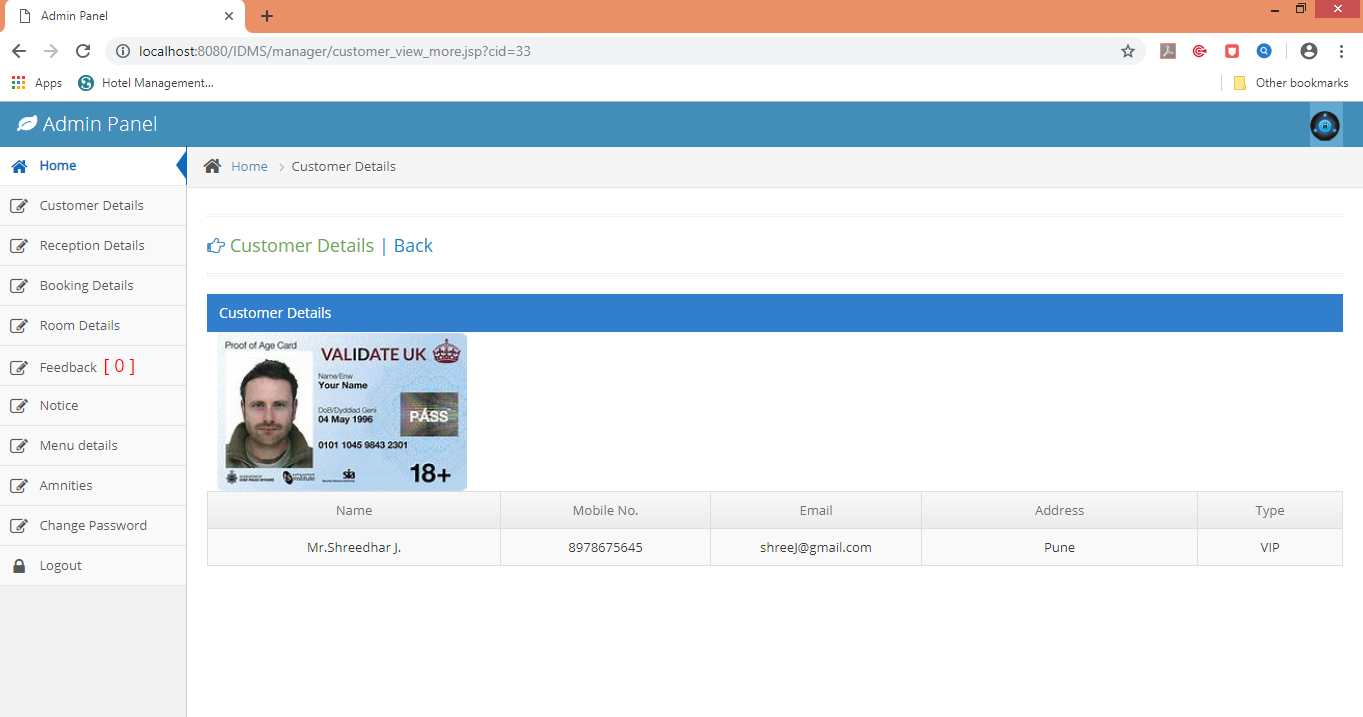
1. **Admin page**

****

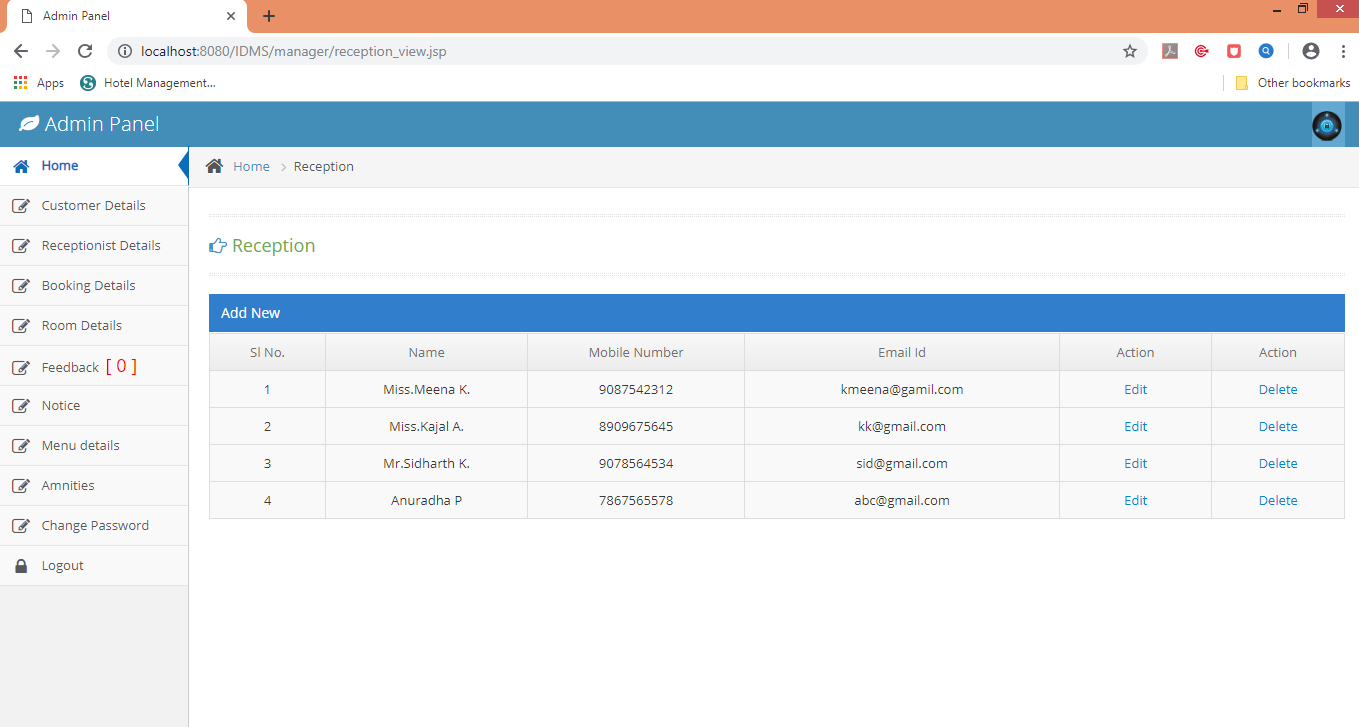
1. **Customer list**

****

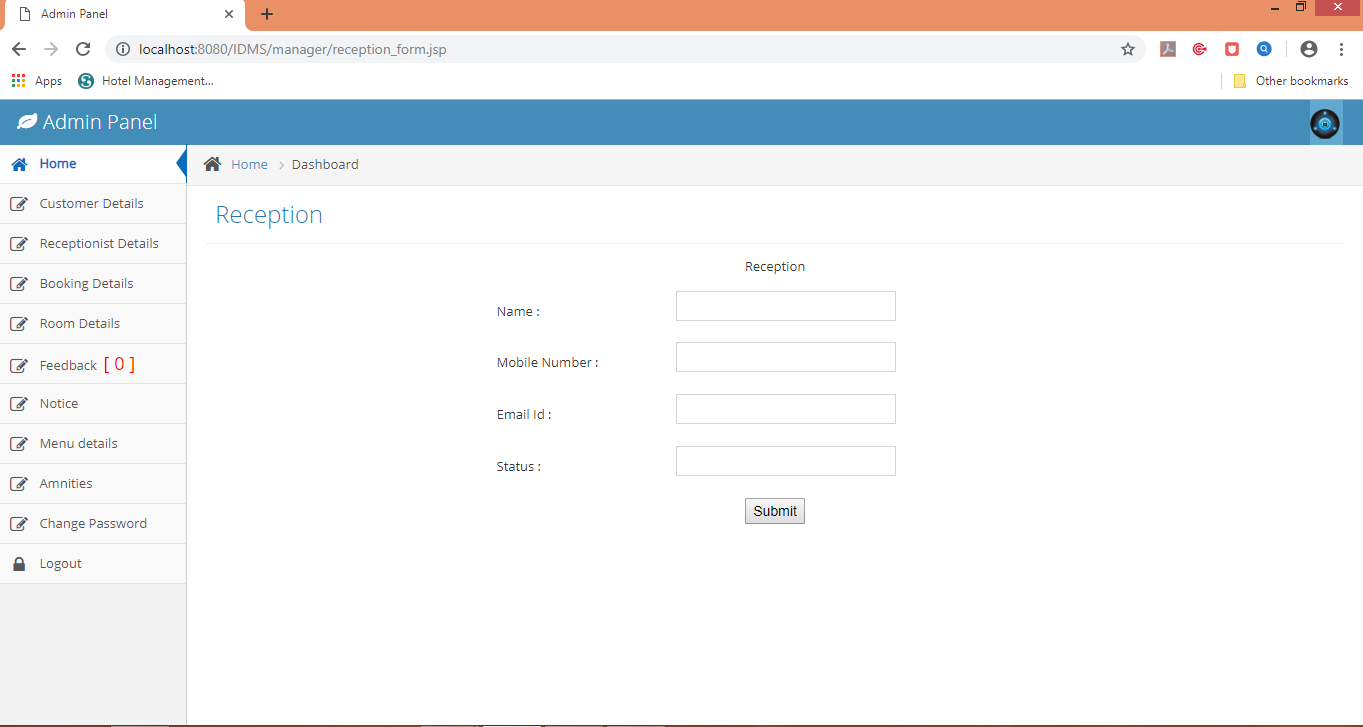
**4.1 Customer details**

****

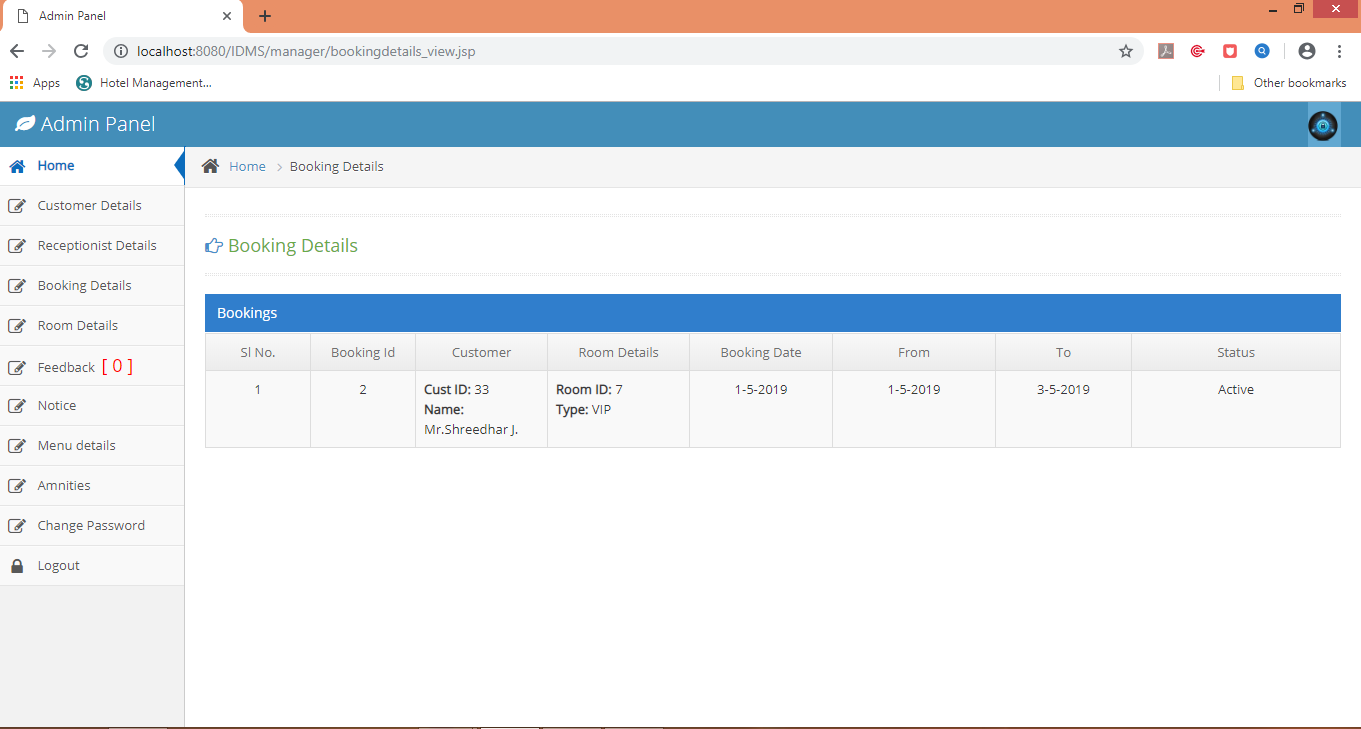
1. **List of receptionist working.**

****

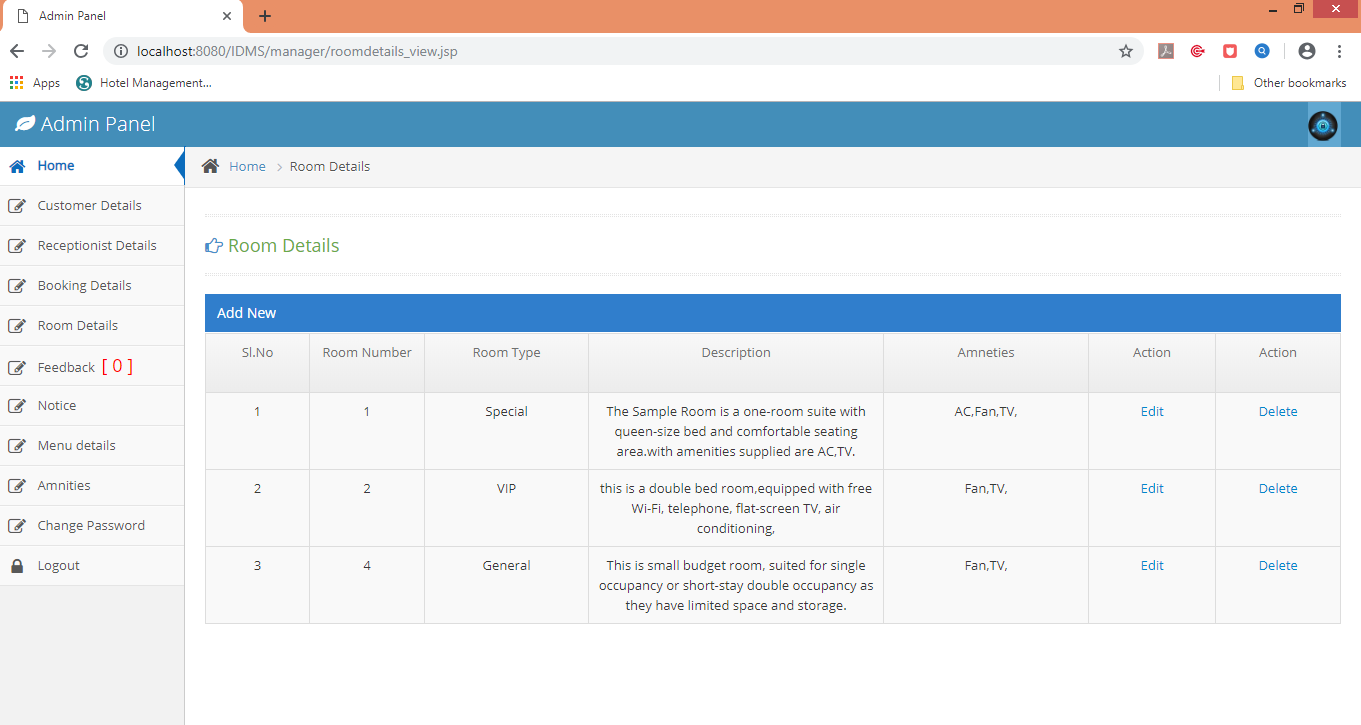
**5.2 Admin will add new receptionist**



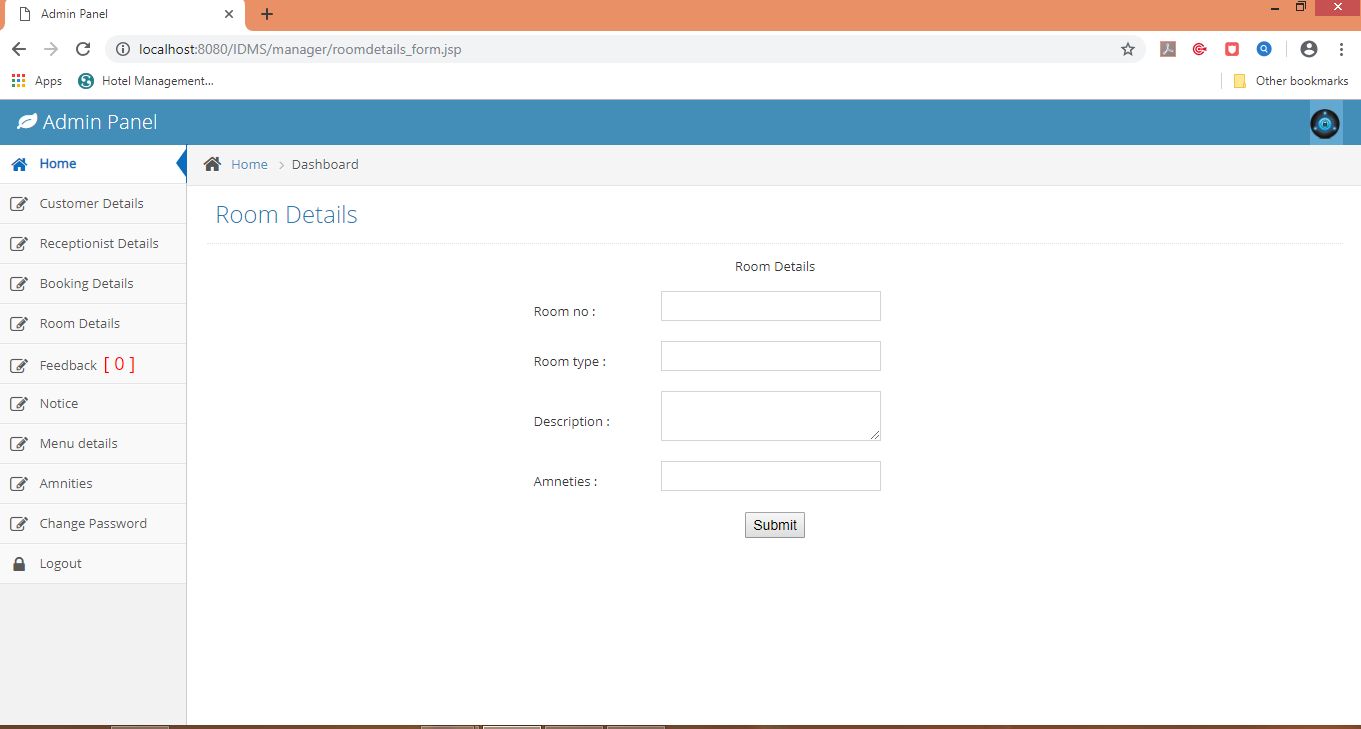
1. **Booking details**

****

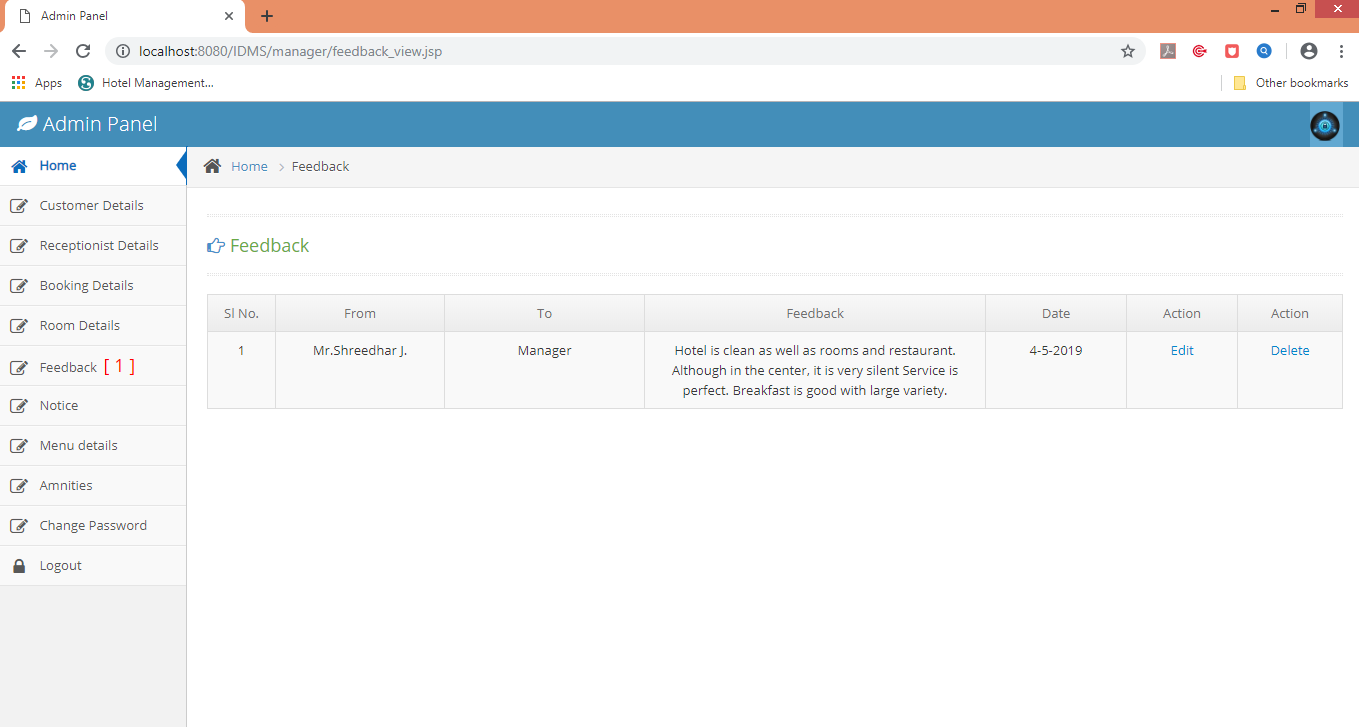
1. **Room details**

****

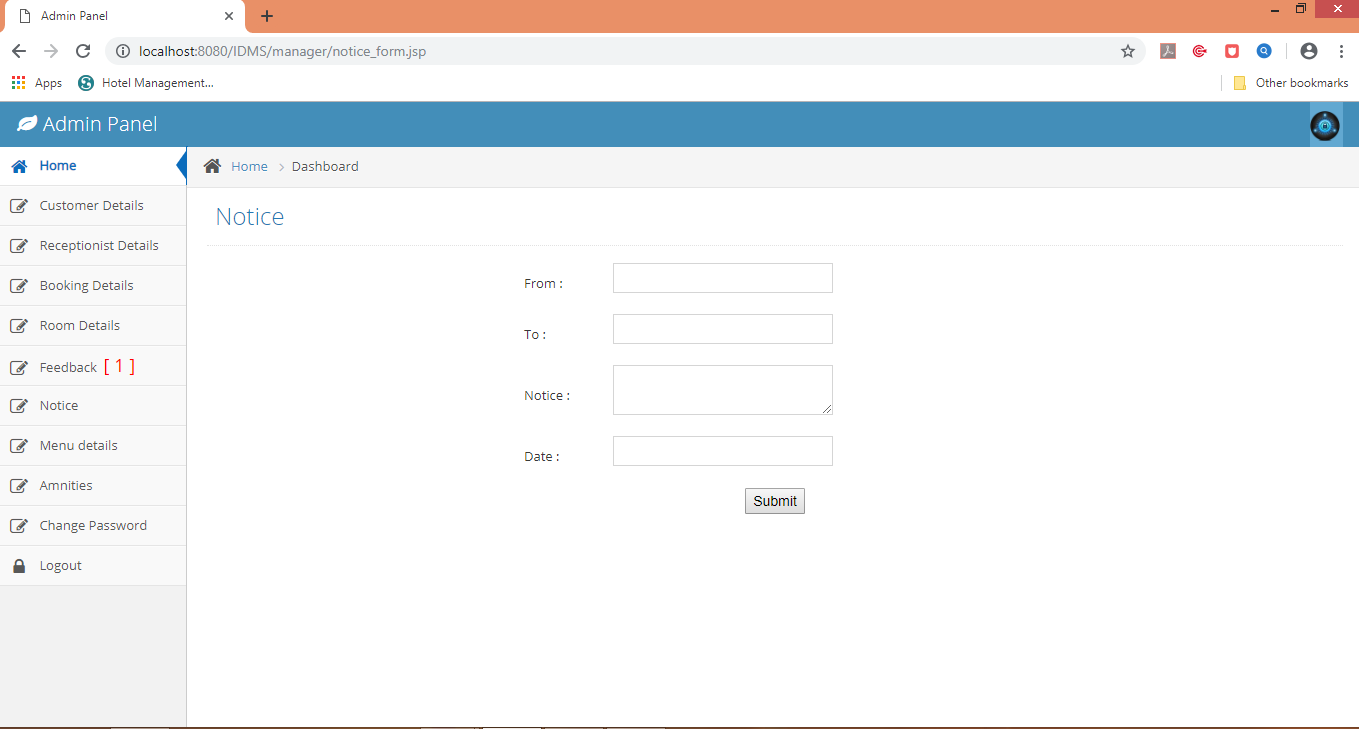
* 1. **Admin can add new room details**

****

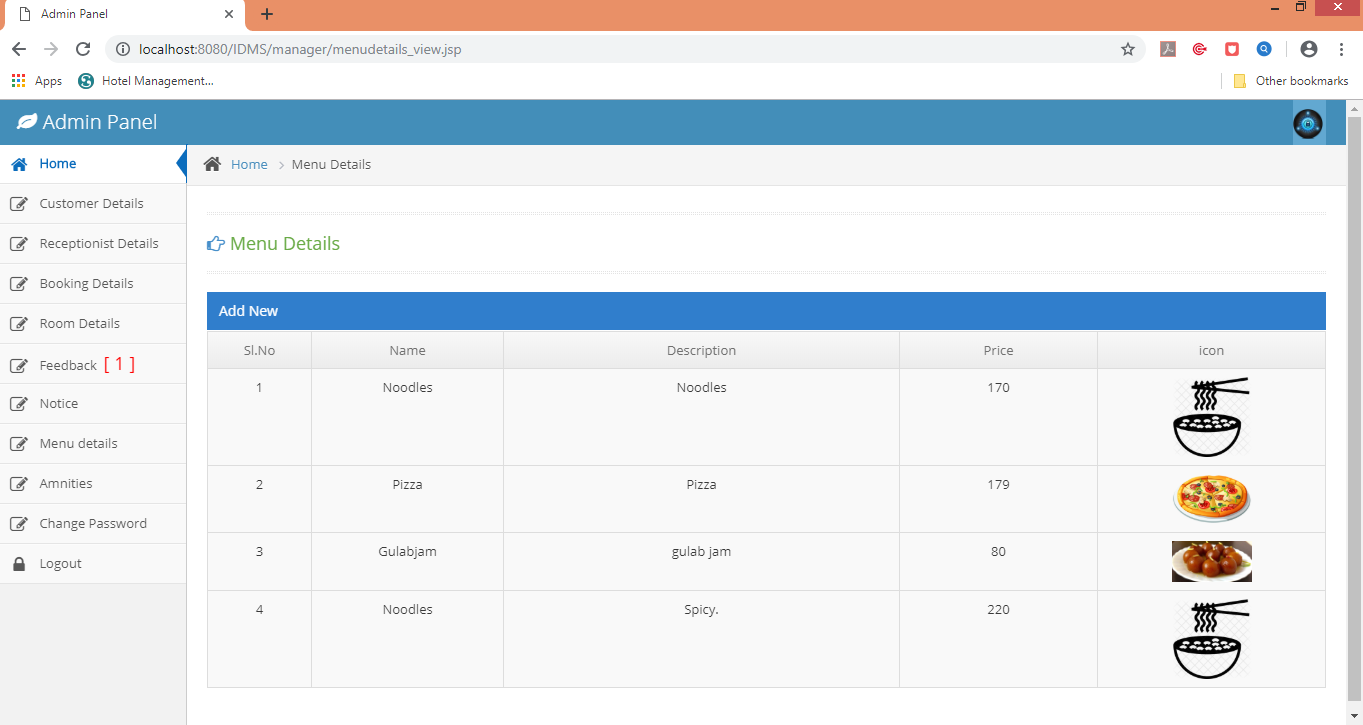
1. **Feedback**

****

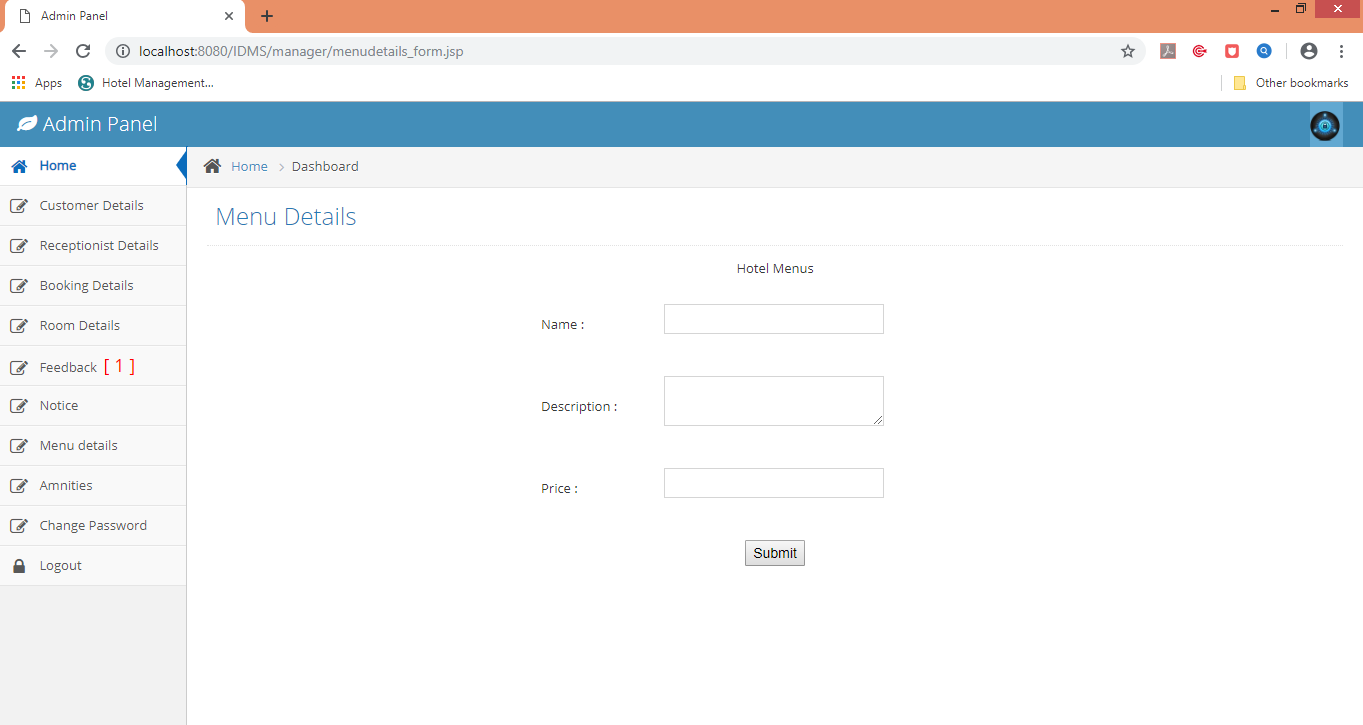
1. **Admin can send notice**

****

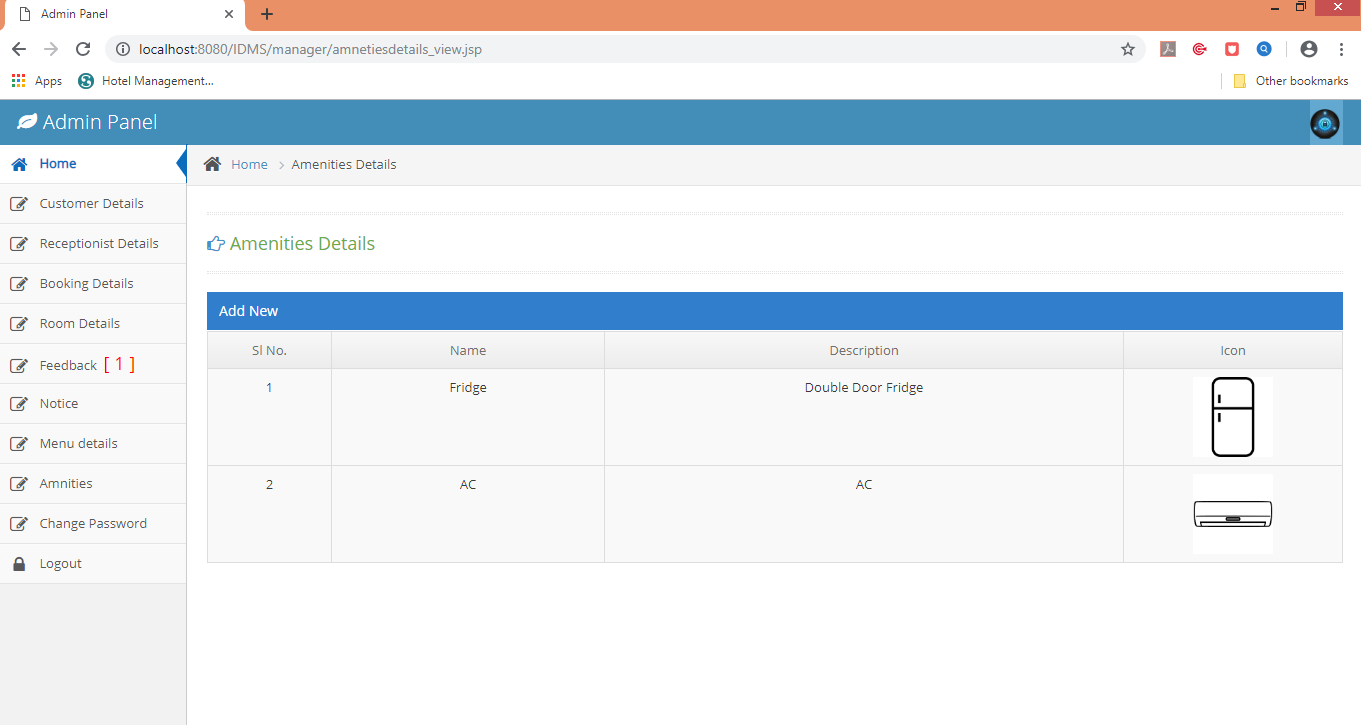
1. **Menu details**

****

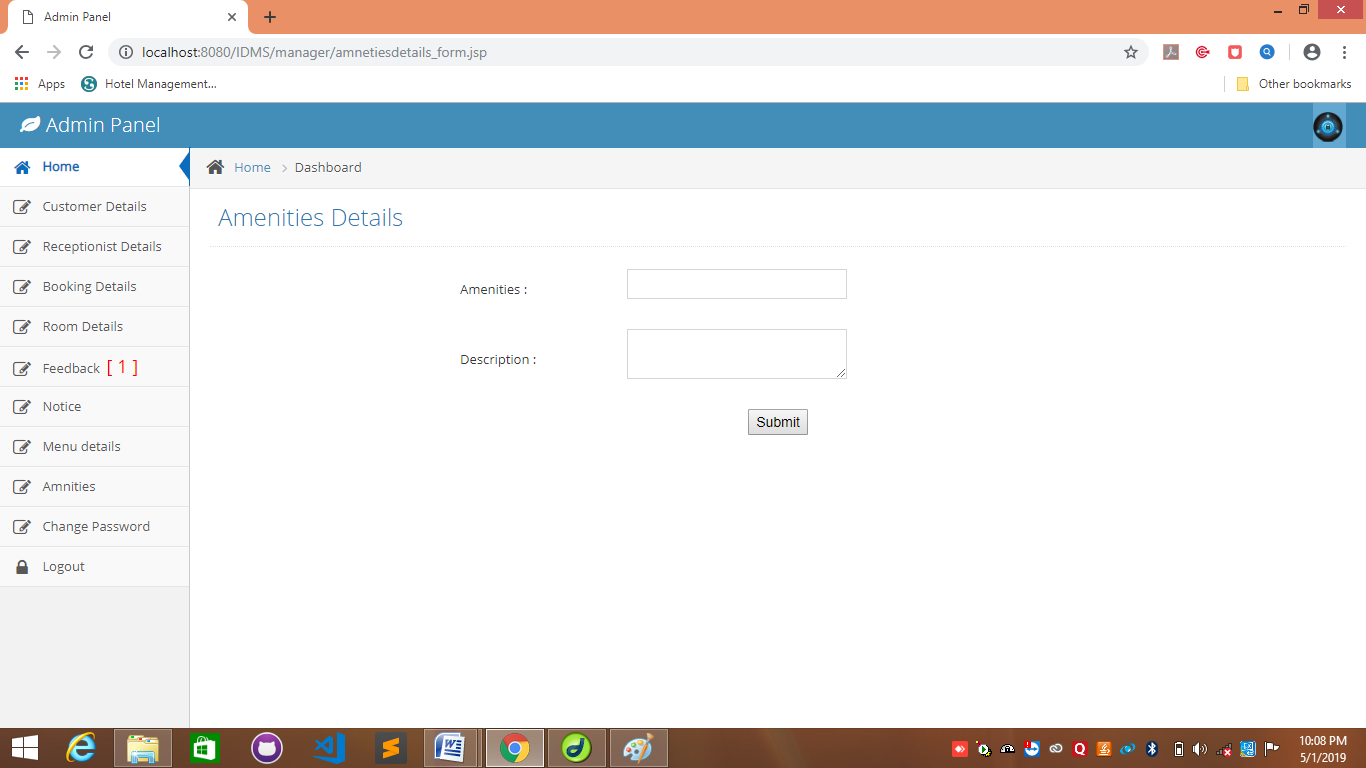
**10.1 Admin can add new menu to list**

****

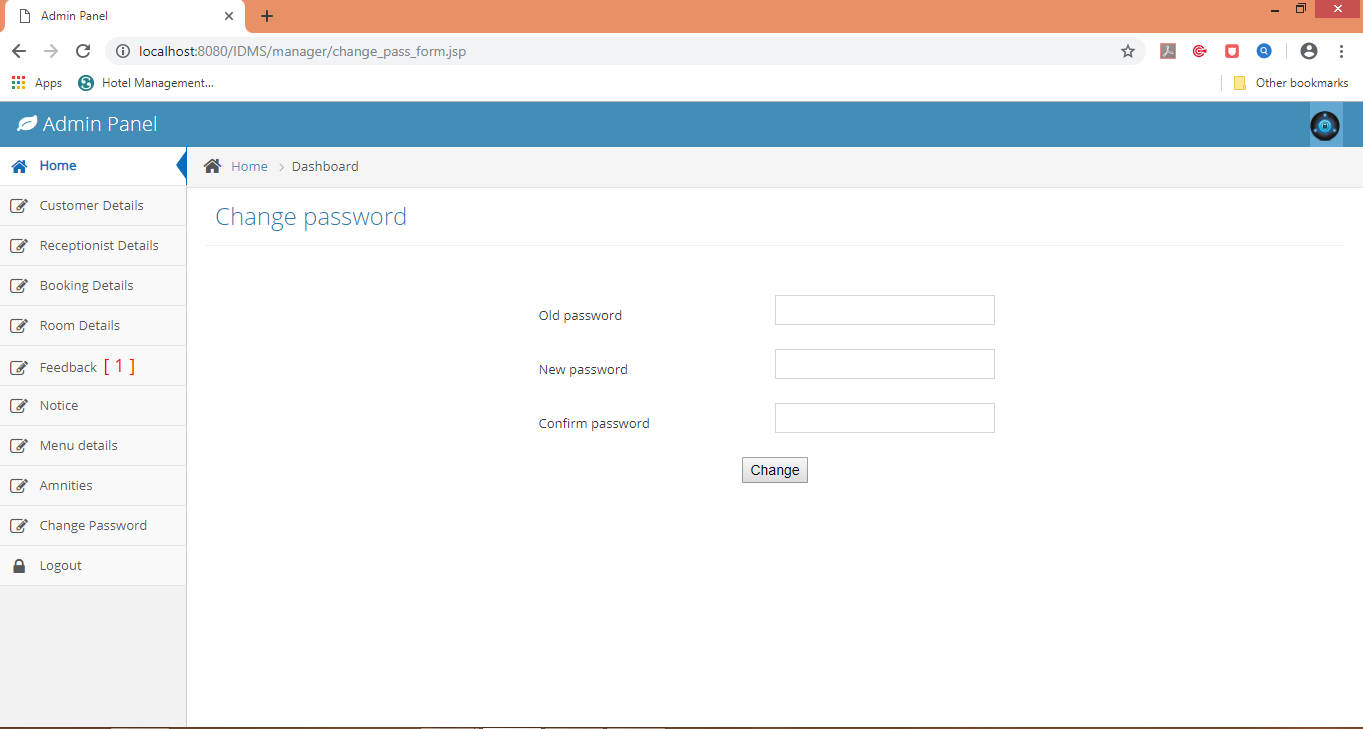
1. **Amenity details**

****

**11.1 Admin can add new amenities**



1. **Change password**

****

**Chapter 7**

**BENEFITS**

Result of benefits of IDMS over the manual based hotel management system.

**Fig. 17 IDMS benefits over the manual system.**

**CONCLUSION AND FUTURE SCOPE**

This project has got clear advantage over the manual system. The computerized system is more reliable, efficient and fast and does maximum work within minimum time. It provides comfort and suitability to everyone. Providing maximum facilities and comfort to customers.

**FUTURE SCOPE**

For the above developed system to function to give more capabilities than the existing, the below features can implement in future.

* Multiple system can connected for communication through networking.
* The hotel server can be connected to the internet, to have the web site for hotel through which online booking page will be loaded to enable online booking over the internet.
* IOT’s can be applied for the future services.
* More security will update as the time passes.

**REFERENCES**

1. [www.google.com](http://www.google.com)
2. [www.googleScholar.com](http://www.googleScholar.com)
3. [www.softwareadvice.com](http://www.softwareadvice.com)
4. <https://projectsgeek.com>
5. <https://www.academia.edu>
6. <https://www.slideshare.net>
7. Grzegorz Golembski (The Poznan University of Economics)"The impact of modern management methods on hotel operational performance", Tourism Review, Vol. 62 Issue: 2, pp.31-36.
8. James A O'Brien, George M.Marakas 7th Edition, Tata Mc-Graw Hills publication.
9. Kevin Donaghy, Una McMahon-Beattie, David McDowell, (1997) "Implementing yield management: lessons from the hotel sector", International Journal of Contemporary Hospitality Management.
10. Ruggero Sainaghi, (2010) "Hotel performance: state of the art", International Journal of Contemporary Hospitality Management,