# 2. Analysis

# 2.1 Introduction

The systematic evaluation of any information or data for understanding cause-effect or relation which is done for problem solving or decision making. It is a process of examination of data as well as facts done by breaking down it into several parts to understand its interrelationship. However, analysis is done to know more about the relevant topic. This process helps to generate the sample of the system that can be easily understandable by the clients or developers about what kind of system is going to be built.

## 2.1.1 Analysis methodology:

The procedure that we perform to application performance or analyze system is analysis methodology. Analysis without a methodology is generally considered as valueless in the present world. Since there are many methodologies to perform. Here, we are going to perform Hotel Management System by using soft system methodology.

Soft System Methodology is the way of tackling the problem situation in which there is involvement of social, political and human activity components. I have used this methodology as it more people oriented and study human behavior which shows the high interaction between them. It consists of three steps

1. **Rich Pictures**

The process of presenting situation as a picture using symbols, diagrams, cartoon and words which can be drawn by hands or electronically. It should reflect as much going on as possible without privilege or presuming any point of view.

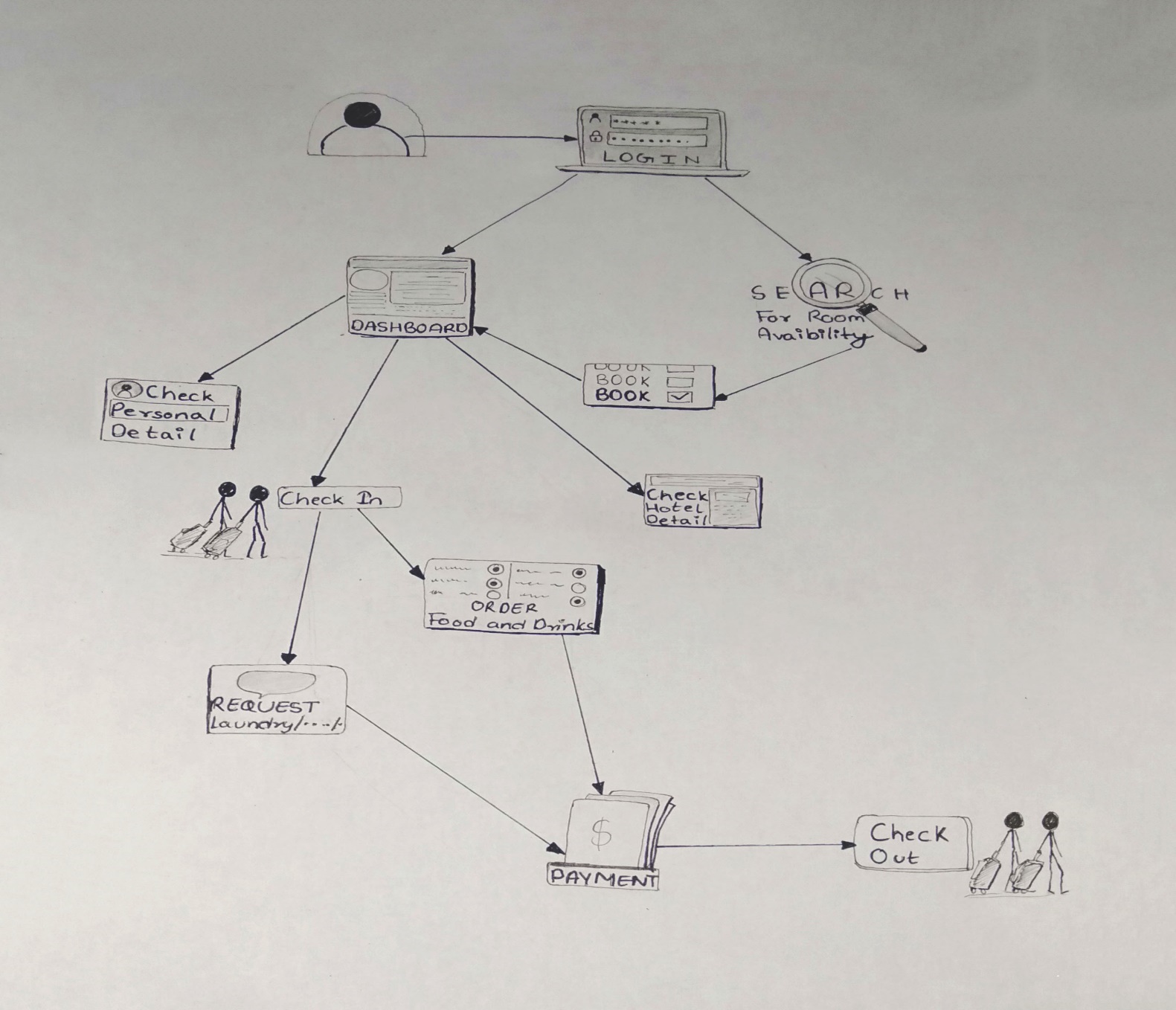


Figure 1: Rich picture of Hotel management system

1. **Root definition**

Defining about the proposed system, user should register himself/ herself before reserving the room. After login the user can choose required room type, view dashboard, cancel book if needed. User can also comment, rate the system service, user can also order food, request laundry, ironing service through internet. As a result, the system reduce time spent on administrative tasks, increase online presence that helps to build good relationship between guests.

1. **Conceptual diagram**

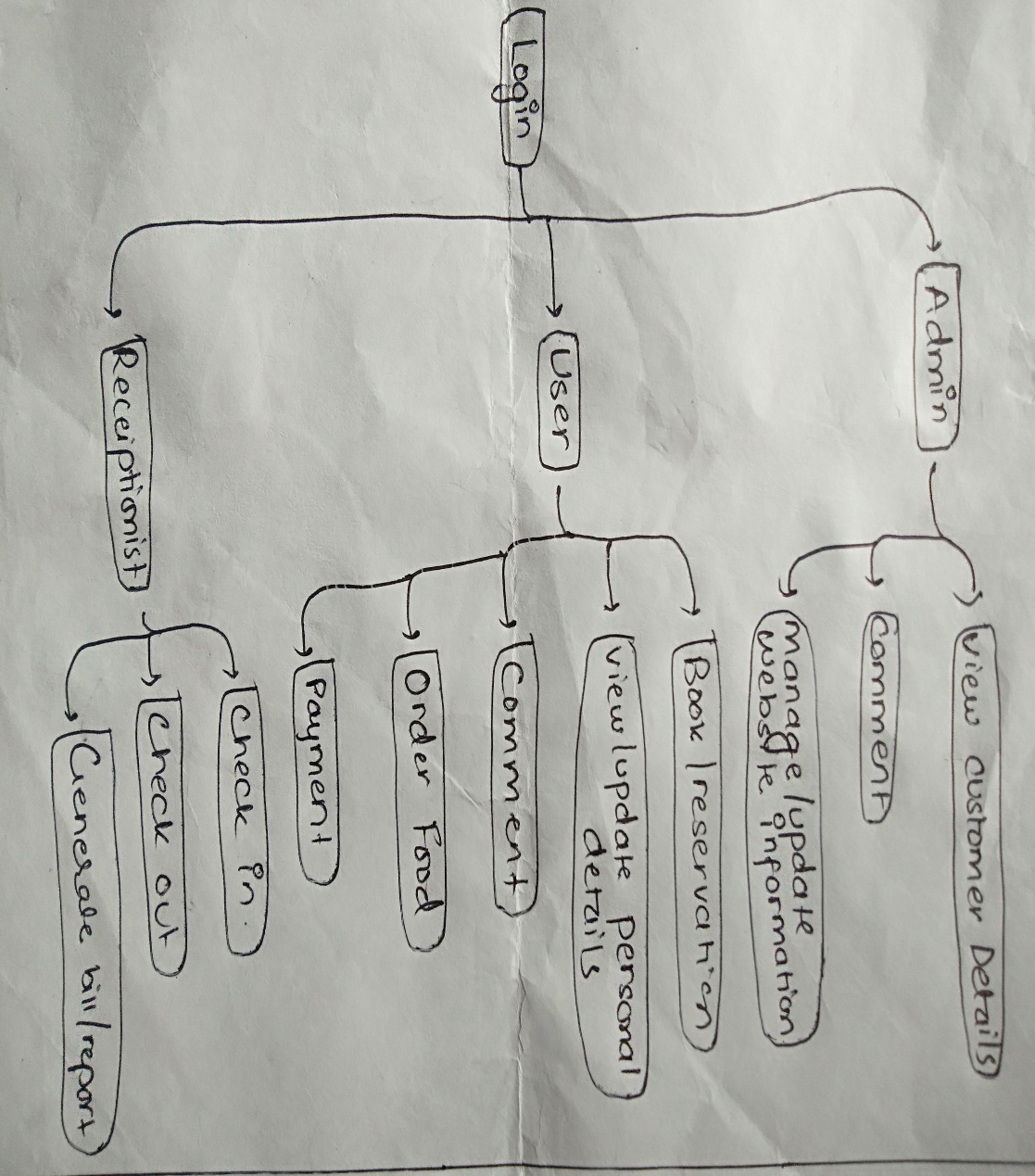
The illustration of representing the arrangements and relationships of key attributes within a system by using a variety of symbols that can be easily understandable. It is an effective way to communicate the complex message in a normal and informative manner. It provides better understanding to the system process via the portrayed. 

Figure 2: Conceptual diagram of Hotel management system

# 2.2 Information Gathering Techniques

Requirements gathering are the essential part for a software project to be built. The process of gathering requirements is usually between user of the software and developer. The developer needs to access the requirement for the development of the software which means collection of requirements need to be done. However, Technique defines the process how tasks are to be performed within specific circumstances.

There are various techniques for the requirement gathering such as **Interview** (One-to-one Interview, Group Interview), **Questionnaires**, **Use Cases, Requirement Workshops, Observations, SWOT, Document Analysis, CATWOE** and many more.

Shortly, Requirement gathering is done to know about user perception about what kind of software they want to use. In order to retrieve user requirement, I have chosen Interviews and

* **Interview**

It can be done either one to one person or within a group. It is considered as easiest yet most powerful technique available for gathering requirements. The interview was done with the individual and focus group who are mostly travel to various place time to time. They were chosen because they would know about the difficulties while booking a hotel or problem faced while staying ta hotel.

Some of the questions during interview are mentioned below:

1. Have you ever booked the hotel room online?

|  |
| --- |
|  |

1. What are the most common difficulties you have faced while using hotel website?

|  |
| --- |
|  |

1. Is there any function you wish to have in hotel site?

|  |
| --- |
|  |

1. Do you wish to perform all the hotel transaction via internet or you feel secured in person to person transaction?

|  |
| --- |
|  |

Since it is easier and cheaper way to collect information from different number of people, the gather information was used for further process.

# 2.3 Feasibility Study

It is the initial step performed during project management which is done to indicate if a project is viable or not. It also indicates quantitative and qualitative assessments of other essential resources including identification of critical points and its timetable with cost estimation. This study ensures a project is technically, legally and economically feasible which helps to analysis either investing time, effort and cost on the project is profitable or not. Similarly, performing feasible study in this project is done considering in following sectors.

**Technical Feasibility**

The study that mostly consider the technology material for working in the project. In order to check the technical feasibility, we need to consider if the project can be completed with its present equipment, software, personnel or not? Here we are using PHP as a programming language we acquired the technical knowledge for development and implementation of proposed system.

**Economical Feasibility**

The estimation of cost and benefits associated are compared as it is tangible and intangible benefits outweigh the cost so it is considered acceptable. This system also decreases the technical staff to do various jobs that single software can do. Since the proposed system does not have any economic constraint to be considered, so it is proven as economically feasible.

**Legal Feasibility**

Does the proposed system conform the legal and ethical requirement such as social media laws, data protection act or not? Considering the raised question, the project does not conflict or contradict with any legal requirement, it is considered legally feasible.

**Scheduling Feasibility**

The measurement of how reasonable the project duration is done. The estimation of time is done according the stages so it is feasible in scheduling as well.

**Behavioral Feasibility**

Since many hotels have already used various software for managing their data and information and can be acceptable by all hotel administration or hotel staff, it is considered as behavior feasible.

# 2.4 SRS (System Requirement Specification)

The set of documentation that shows the behavior of the system application which includes a variety of element that describe the intended functionality required by the clients to satisfy their man users in a specific environment. It is done to define the objectives or scope of the software that helps the development team to build the software to avoid the failure in software project. Here, we have specified as functional and non-functional requirements.

# 2.4.1 Functional Requirement

Some of the functional requirement of Hotel management system is mentioned below

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| ID | Title | Description | Rational | Dependency |
| FR01 | Registration | Personal details and email are created for further transactions and registered themselves. | To sign up new user | - |
| FR02 | Login | Valid Email and password are written and dashboard site is opened. | To access to further facilities of system. | FR01 |
| FR03 | Reservation/ Booking | The user can book their preferred room as wanted after login. | To reserve the room. | FR01, FR02 |
| FR04 | Cancel booking | The user can cancel their booking as wanted. | To cancel their reservation. | FR03 |
| FR05 | Comments | The user can write their feedback and admin can reply respectively. | To mentioned their feedback. | FR02 |
| FR06 | Update Customer Dashboard | User can edit their personal details if any change is required. | To modify the data if given information is wrong. | FR02 |
| FR07 | Admin Control of System (edit, update customers, staff) | Admin can view the customer, staff details and update if required. | To ensure the right information is stored in database. | FR01, FR02 |
| FR08 | Google Map Integration | Customer to prefer to come to the hotel can view map integration to reach destination. | To make ensure customer can reach hotel without any difficulty to search. | -- |
| FR09 | Notification | Customer can make sure their reservation or booking is successful. | To ensure the work is done. | FR03, FR04, FR05 |
| FR10 | Search Food Menu | Customer can search and order their food as wanted. | To make customer to have less difficulty in ordering meal. | FR01, FR02 |
| FR11 | Employee details | Any users can see the staff details and can know their background. | To make good communication and trustworthy to the users. |  |
| FR12 | Billing system | Customer can get the expenses details through the bill system. | To avoid any misunderstanding in money transaction. | FR03 |
| FR13 | Gallery | Any users can see the photo gallery of that system to view the hotel before booking. | To attract the user through proper gallery of the hotel. | FR01, FR02 |
| FR14 | Online Transportation booking | Any customer willing to come to the hotel, can book the transportation to reach the destined hotel. | To make customer easy to reach hotel. | FR02, FR03 |

# 2.4.2 Non-functional Requirements

Some of the non-functional requirement of Hotel management system is mentioned below:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| ID | Title | Description | Rational | Dependency |
| NFR1 | Security | Authorization access scenario and definition should be maintained. | To make the system secure | ---- |
| NFR2 | Performance | Speed and capability of the system should be managed so that the system can provide better performance | To make the system robust | ----- |
| NFR3 | Availability | The system should be accessible anytime whenever needed. | To make the system available | NFR1 |
| NFR4 | Maintainability | Maintaining update and upgrade facilities should be easily adaptable. | To make the system easy to repair | NFR2, NFR3 |
| NFR5 | Reliability | The monitoring system performance should be well managed. | To make the system reliable | NFR1, NFR2, NFR3, NFR4 |
| NFR6 | Integrity | The data and information of one user should not be accessible to other which prevents from data integrity. | To assure the consistent of data | NFR1 |
| NFR7 | Usability | Prioritizing the important functions of the system based on usage pattern should be done. | To minimize the criticism system | NFR2, NFR3 |
| NFR8 | Scalability | Every number of people can access use to the system if increased no of users if needed. | To make the system scalable according to the no of users. | NFR2 |
| NFR9 | Testable | It must be started with objective, measurable as well as testable as if we can’t test it, we can’t ship it. | To make the system function as designed. | NRF2, NFR5 |

# 2.4.3 Prioritization

The process of defining or evaluating the urgency of a things. Arranging the items on the order of the relative importance of the function should be done so that the important requirement in the system development is not excluded. We can divide the list by analyzing or group discussion or evaluating from requirement gathering technique. The main objectives behind performing prioritization is It determine what is important thing to do on first stage which helps to reduce high risk of system failure and increase the system to become more reliable, efficient and productive one.

There are numerous methods on prioritizing requirement in which I have decided to use MoSCoW Technique in system development. MoSCoW prioritization is an effective method that allows us to determine how much effort goes in each category which ensure that good variety of initiatives is delivered. MoSCoW have 4 different categories of initiatives including Must-have, Should-have, Could-have and wont-have. The word is described as below:

**M**- Must have initiatives: This category represents non-negotiable needs for the project Anything in this initiative is considered as mandatory that decide the success of the project.

**S**- Should have initiatives: This category is important to the product but not represent as a vital. This initiative adds significant value to in the project if included.

**C**- Could have initiatives: This category is not so necessary to the core part of the products but might have slightly impact on the outcome if left out. As a result, this initiative may be deprioritized if needed.

**W**- Won’t have initiatives: This category is mentioned in non-priority for specific time frame. This initiative is most not likely to be happen at all that does not affect behind the failure or success of the project.

The MoSCoW prioritization of Functional requirement of Hotel Management System

|  |  |  |
| --- | --- | --- |
| ID | Functional Requirements | MoSCoW |
| FR101 | Login | Must have |
| FR102 | Registration | Must have |
| FR103 | Reservation/ Booking | Must have |
| FR104 | Cancel booking | Must have |
| FR105 | Comments | Must have |
| FR106 | Update Customer Dashboard | Should have |
| FR107 | Admin Control of System | Must have |
| FR108 | Google Map Integration | Should have |
| FR109 | Notification | Should have |
| FR110 | Search Food Menu | Should have |
| FR111 | Employee details | Could have |
| FR112 | Billing | Must have |
| FR113 | Gallery | Could have |
| FR114 | Online Transportation booking | Could have |

The MoSCoW prioritization of Functional requirement of Hotel Management System

|  |  |  |
| --- | --- | --- |
| ID | Non-Functional Requirements | MoSCoW |
| NFR1 | Security | Must have |
| NFR2 | Availability | Must have |
| NFR3 | Reliability | Should have |
| NFR4 | Integrity | Must have |
| NFR5 | Usability | Could have |
| NFR6 | Scalability | Should have |
| NFR7 | performance | Must have |
| NFR8 | Maintainability | Must have |

# 2.4.3 Hardware and software specification

**Hardware specification**

* For installation minimum 4 GB or more RAM.
* Processor i3 or more.
* Devices like computers or laptops is needed.
* Hard disk for backup.

**Software specification**

* The operating system (Windows and Linux) is compatible.
* Browsers like Mozilla Firefox, Google chrome is suitable.
* XAMPP as server.
* MYSQL as database.

Integrated Development Environment: Php HTML, CSS

# 2.5 Use Case Diagram

The dynamic or behavior diagram that model the functionality of a system b sing actors and use cases. It is more valuable for analyzing the functional requirements of a system that results in translating into future design choices and development priorities. Here I have drawn a use case diagram showing how the system interacts with actors about the functionality implemented.

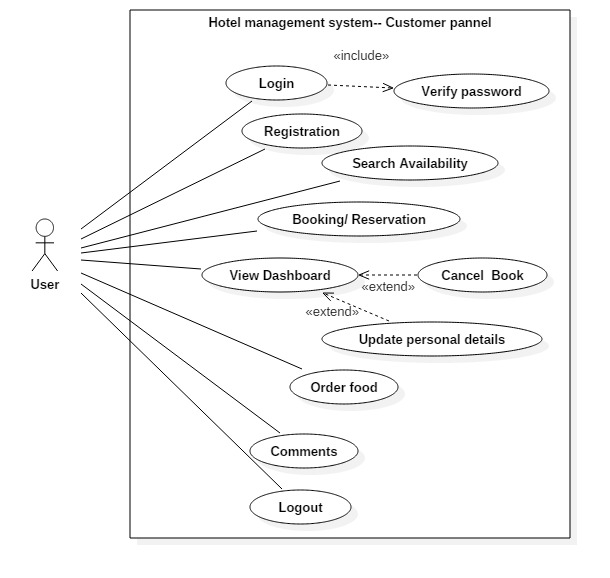


Figure 3:Use case diagram of user panel

**Registration**

|  |  |
| --- | --- |
| Name of case | Registration |
| Actor | User |
| Flow of event | User should provide valid data to register themselves to facilitated further transaction. |

**Login**

|  |  |
| --- | --- |
| Name of case | Login |
| Actor | User |
| Flow of event | User should login before reservation and to gain further facilities. |

**Search room availability**

|  |  |
| --- | --- |
| Name of case | Search room availability |
| Actor | User |
| Flow of event | User can search for the type of room they wanted after login. |

**Booking**

|  |  |
| --- | --- |
| Name of case | Booking |
| Actor | User |
| Flow of event | After successfully login user can book their desired room. |

**View Dashboard**

|  |  |
| --- | --- |
| Name of case | View Dashboard |
| Actor | User |
| Flow of event | User can view their dashboard and can update their details if wanted. |

**Order food online**

|  |  |
| --- | --- |
| Name of case | Order food online |
| Actor | User |
| Flow of event | User can order their meal from anywhere via internet access. |

**Comments**

|  |  |
| --- | --- |
| Name of case | Comments |
| Actor | User |
| Flow of event | User can keep their point of view through comments. It can also be said as feedback. |

**Logout**

|  |  |
| --- | --- |
| Name of case | Logout |
| Actor | User |
| Flow of event | After their use of system through online facilities’ user can dismiss their action by doing logout. |

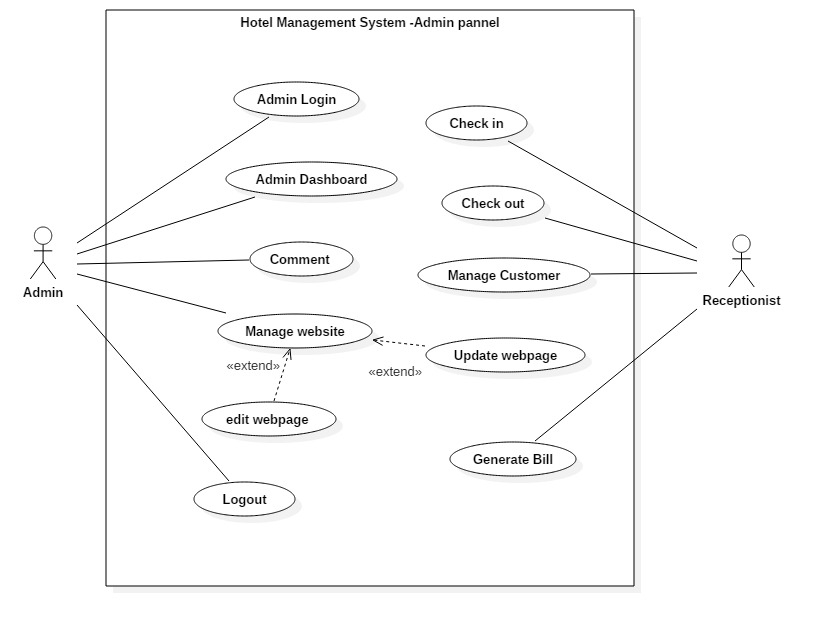


Figure 4: Use case diagram of admin panel

**Admin login**

|  |  |
| --- | --- |
| Name of case | Admin login |
| Actor | Admin |
| Flow of event | The login can only be done by the admin user and can have authorized facilities. |

**Admin Dashboard**

|  |  |
| --- | --- |
| Name of case | Admin Dashboard |
| Actor | Admin |
| Flow of event | Admin can view the comments, messaged done by the users. |

**Comments**

|  |  |
| --- | --- |
| Name of case | Comments |
| Actor | Admin |
| Flow of event | Admin can reply to the comments post by the users. |

**Manage website**

|  |  |
| --- | --- |
| Name of case | Manage website |
| Actor | Admin |
| Flow of event | Admin can update the site content by deleting or adding further information. |

**Logout**

|  |  |
| --- | --- |
| Name of case | Logout |
| Actor | Admin |
| Flow of event | Admin can dismiss the action by doing logout so that other unauthorized people cannot access to the data of the users. |

**Check in**

|  |  |
| --- | --- |
| Name of case | Check in |
| Actor | Receptionist |
| Flow of event | After customer come to the hotel, receptionist accept and provide the key. |

**Check out**

|  |  |
| --- | --- |
| Name of case | Check out |
| Actor | Receptionist |
| Flow of event | After certain time spent in hotel, customer who want to leave hotel, receptionist check out to them. |

**Generate Bill**

|  |  |
| --- | --- |
| Name of case | Generate Bill |
| Actor | Receptionist |
| Flow of event | At the moment of check out, the bill is generated that display whole expenses spent all over the hotel transaction. |

As we can see there are two different panel, one is of customer and another is of Admin. Taking about customer panel, customer can login to the system and search for room availability and book for desired room type and cancel the book if required. Not only that customer can see their personal details and activity details in their own dashboard. The customer can comment, order food online which helps customer to be more internet friendly and reliable to have such kind of system.

Looking after admin panel, admin can manage the website and have own dashboard. Admin can also reply to the customer comments on the forum. However, receptionist can have the details of customer like checking in and checking out. Generate bill for them and handle them thoroughly.

# 2.6 System Architecture

The set of rules in a computer system’s technical framework as well as customer requirements and specification that follows in designing. As we proposed three tier is suitable to our system. The given below diagram shows the details description architecture of the Hotel management system which is based on three-tier architecture. The three Tier are named as

* Presentation Tier
* Business Tier
* Data Access Tier

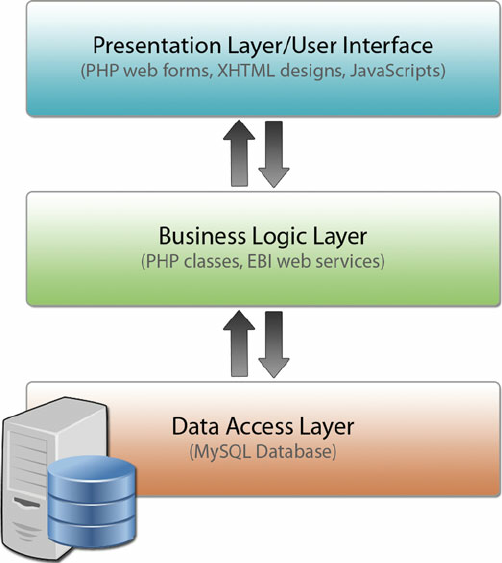


Figure 5: Three tier Architecture

From the above given Three-tier architecture, the presentation tier display information into people legible form which is related to services such as searching room availability, booking rooms, cancel booking, Managing Dashboard etc. The presentation tier communicates with the other tier by displaying results to the client tier and other tiers. After that, the business logic tier is responsible for exchanging the information between the user interface and the database of the relative project. Lastly, the final tier that consists of the server of the database stores all the related information of Hotel management system and can be retrieved from this tier.

## 2.6.1 Natural language analysis

Natural Language Analysis is the process of identifying the words like nouns, adjectives and verbs in the system which are interrelated to various components of the system. In which nouns is used as a potential class, objects and fields whereas verbs as potential methods or responsibilities of a class and adjectives as their attributes.

The following scenario is said by one of the people from the interview. I am a researcher and a traveler. I move to different place time to time. It’s my hobby and part of my job to travel. I need to stay at hotel while researching. I have faced a lot of problem like-no more room available, lack of proper management service, sitting at a queue for checking in and out. I seek for fast and reliable service in hotel. Implementing this hotel management system to hotel have saved my time so much. In fact, other users like me can also get this kind of facilities which means I can book my room from where ever I am present. I can see my dashboard for report. The comments and rating facilities has helped me to choose best hotel among list. Including ordering the food through internet, staff identification, requesting for room services, laundry, ironing via online. These all are the facilities that has helped lot to be internet friendly. Travelling time to time, booking has been out of my problem list recently.

In order to get the candidate noun, verbs and adjective following steps were adapted

Step 1: Listing out all the noun, verb, adjectives.

Step 2: After that filtration is done by

* Removing duplication.
* Removing synonyms.
* Irrelevance noun were also dismissed.
* The noun, verb, adjective which are not within a scope was also removed and which can be possibly part of future implementation was also removed.

Step 3: After certain filtration made from candidate class the below mentioned actual class were separated.

|  |  |
| --- | --- |
| S. N | Actual class |
| 1 | User |
| 2 | Book |
| 3 | Room |
| 4 | Food order |
| 5 | Laundry |
| 6 | Billing system |

## 2.6.2 Initial Class Diagram

Class diagram is the type of static structure diagram which explain the structure of a system by showing their class, attributes, operations (methods) and their relationship between objects. From the above given scenario following initial class diagram can be made.

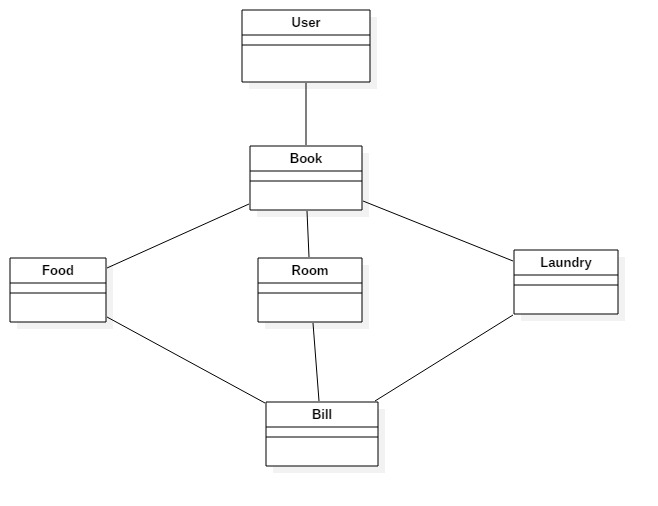


Figure 6: Initial class diagram

Since it is the initial diagram, in future implementation we will be drawing final diagram with its method and attributes.