**HOTEL BOOKING METASEARCH ENGINE**

### Project Vision

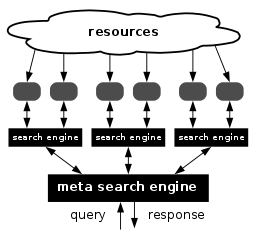
A **metasearch engine** is a [search](https://en.wikipedia.org/wiki/Information_retrieval) tool that uses another [search engine's](https://en.wikipedia.org/wiki/Web_search_engine) data to produce its own results from the [Internet](https://en.wikipedia.org/wiki/Internet). Metasearch engines take input from a user and simultaneously send out queries to third party search engines for results. Sufficient [data](https://en.wikipedia.org/wiki/Data) is gathered, formatted by their ranks and presented to the users.

Metasearch engines have their own sets of unique problems. All of the websites stored on search engines are different, which draws irrelevant content. Problems such as [spamming](https://en.wikipedia.org/wiki/Spamming) reduces result [accuracy](https://en.wikipedia.org/wiki/Accuracy_and_precision). The process of fusion aims to tackle this issue and improve the engineering of a metasearch engine.

There are many types of metasearch engines available to allow users to access specialized [information](https://en.wikipedia.org/wiki/Information) in a particular field.

Main Features:

* Allow visitors to book, reserve room with just easy -way of accessing through a totally automatic account system.
* Allow customers to check, modify the booking, send requests, feedbacks and verify the bill of staying.
* Give permission to do the admin tasks such as check-in, check-out, searching rooms, customers, replying requests and so on.



### Project Scope

The scope of this project is to provide a single platform to rerach out multiple platforms to book hotels for a desired location, handled by metasearch engine which provides the best deal available as per ratings,location,cost,basic amenities etc.

## 

## 2. Functional Requirements

#### 2.1.1 Searching:

* Offers the best price available by comparing different parameters on different websites .
* Redirecting the users to the selected product over to client website.
* Users can pay online via E-banking provided by client website.
* After successful booking, users will be informed a confirmation code, and all the information of the users will be saved at database for further purposes.

#### 2.1.2 Checking in, Checking out:

* Admin log into system to check the information of customers when they check in (out). Each Admin will be provided with a unique account. When they log into the system, a friendly user interface will display and a connection with database will be established.
* Allows users to check in (out) when somebody else has booked rooms for them.

#### 

#### 2.1.3 Allows users to post their feedbacks on a webpage.

### 2.2 Data management

#### 2.2.1 The system works with 2 account types:

Admin, Users.

#### 2.2.2 Allow staff or admin to search and check every account

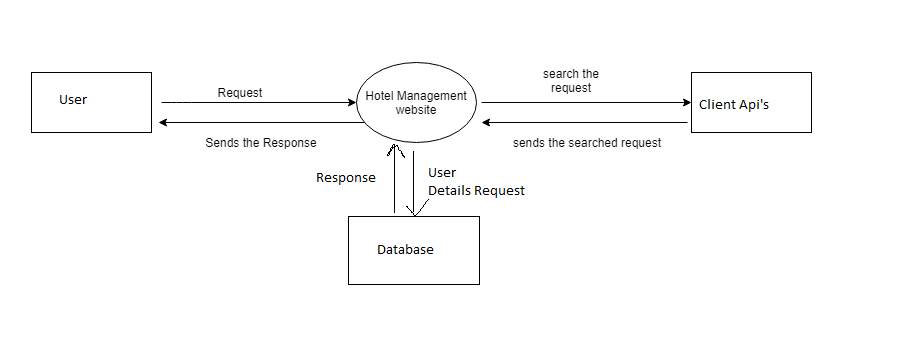
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Task to do | Admin |  |  | User |
| Create accounts | X |  |  | X |
| Modify accounts | X |  |  | X |
| Delete accounts | X |  |  | X |
| Search accounts | X |  |  |  |

### 

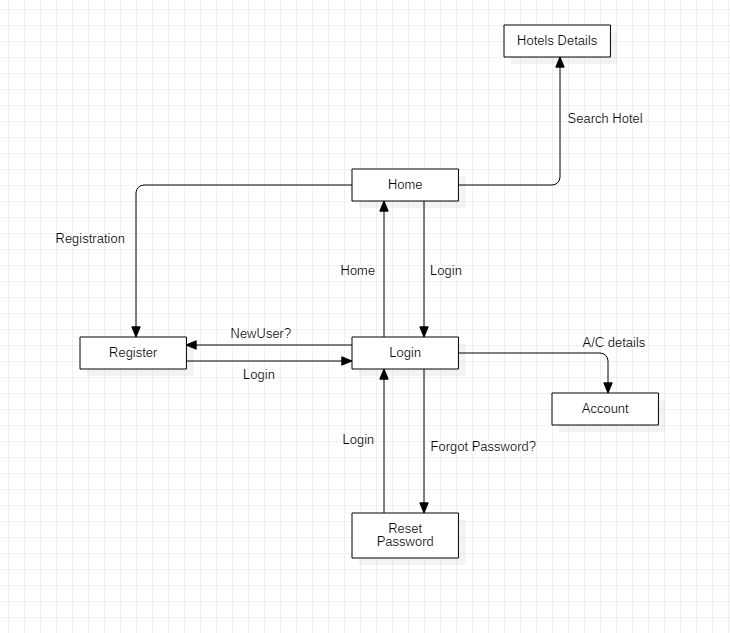
#### 2.4.1 Display following information:

* Recently viewed details of user.
* Login logout details .
* Booking Details.

#### 2.4.2 Modify and update information.



Level-0 DFD



Level-1 DFD

## 4.Non-Functional Requirements

### 4.1. Execution qualities:

#### 4.1.1. Portability:

The web page should display correctly within any Java-enabled browsers.

#### 4.1.2. Usability:

The web page should provide friendly user interface, thus allowing users to operate with ease.

#### 4.1.3: Availability/Reliability:

System down time should be less than 1 minute.

#### 4.1.4. Security:

 Unauthorized parties should be prevented from accessing to the system.

The access permission to the database, user account can only be changed by the System Administrator.

#### 4.1.5. Performance:

 System should be able to process 100 orders per minute.

All system data are backed up once a day and stored.

System automatically saves data inside the database when power fails

### 4.2 Evolution qualities:

#### 4.2.1. Localizability:

System should support multicurrency

#### 4.2.2. Maintainability:

Web page code should be well commented and indented, allowing easy maintenance.

## 5. Interface Requirements

This is a web – based application which provides functionality for hotel booking by comparing 2-3 websites.

It constructs a friendly user interface for normal users even for those who begin to have first attachment with networks. It has to be easy for them to book a room and send request for special needs as well as their feedbacks. Professional and well – organized structure website will impress users at their very first look.

Administrator will be authenticated with highest priority in terms of accessing rights. Software must provide them some tools to do their job.

### 5.1. Hardware interface

This application supports PC with Windows operating system like XP, Vista … Notably, the system provides high compatibility for different hardware, and that is, the system can be installed and accessed through major commercial PCs since it is programmed with high level languages.

### 5.2. Software interface

Our application can run on popular browsers, for instances, Internet Explorer, Firefox, Opera, Safari, Google Chrome … Client should use latest version of these browsers to take advance all of newest features.