REVIEW 1

**PROBLEM DEFINITION**

# Introduction

- Now days tremendous economic development, the software industry is constantly developing new web applications. The more practical is that the hotel business industry is growing as the number of visitors is increasing year by year due to increasing personal income.

- Now demand for hotel management is not only entry data and internal maintenance as before. There are many more professional management models than should be required.

- In order to meet the growing business, we developed a website to serve small and medium-sized hotels and motels :

+ Employees can manage room management activities and services. The service that the hotel offers, hotel guests. From there, the hotel management will also be able to obtain the necessary reports of hotel operations.

+ Guests and customers can visit website to see rooms, hotel’s furnitures, news and send feedback to hotel’s employees. And specially they can booking available rooms online with their favorite, and their chooise.

# Existing Scenario

The hotel has many rooms, we offer online booking service. Customers can choose their favorite room and make a reservation.  
Customers require the entry of personal information so the hotel can call the booking confirmation.  
We will have staff call customer to confirm order. Registered customers can manage their orders.

# Customer Requirement Specification

A site needs to be designed where guests or customers can booking rooms online.

If customers of hotels, they can booking rooms online, mangager their reservations and send feedback to hotel manager.

They can easily search, book available rooms and manager their reservations.

# Functional Requirement Specification

## Guest can register to customer

### Description

The system must provide a page to allow guest create their accounts with the following information:

* Password
* Full Name
* Person Identifier
* Birthday
* Gender
* Company
* Address
* Phone
* Email

### Functional Requirements

1. Guests must provide password, name, birthday, sex, address, phone, email to create a new account.
2. Email is unique, this is username for customer login.
3. Phone number is valid.

## Customers can login for managing their reservations, edit personal information

### Description

The system must provide a page to allow customers login.

+ They can add or remove their reservations.

+ They can edit their information except email.

### Functional Requirements

1. Customers must provide Email and Password to login into the website.
2. Customers can only access to pages they have permission.

## Searching, Booking, View rooms online for guest and customers

### Description

The system must provide a page to allow guests and customers search available rooms and reservate available rooms they choose.

They can choose room type, check-in date, check-out date and number of people for searching available room.

They can see room’s detail and room’s service and room’s furniture.

Then they can only book available rooms. When customers book a reservation online it’s status is “new”, if they don’t confirm with hotel’s staff,

Status of reservation become “cancel”.

### Functional Requirements

1. If guests, they must provider essential information which hotel’s staffs can contact for them.

## Customers send feedback

### Description

The system provide a page for customers send feedback to hotel.

Properties information:

* Question

### Functional Requirements

1. Question is not null, and login is require.

## Guests and customers can view news on website

### Description

System provide for guests and customers some pages for views hotel’s new, photos and new anoucements.

## Management of Rooms, Room Types

### Description

System must provide for Employees a page to manage hotel’s rooms and room type.

Employees can login into admin page and manager hotel’s rooms, employee can view, add, update, delete rooms and room types.

\* Information of Room Type:

1. Room type name
2. Room rate
3. Number of childs
4. Number of adults

\*Information of Rooms

1. Room name
2. Room status
3. Images

### Functional Requirements

1. Employee login into admin pages.
2. Employee have permission to edit, add, update, delete data.
3. If employees delete room type, all rooms of this room type will be deleted.
4. Check if rented rooms can not be delete.

## Employee can manager customers information

### Description

Employess can login to customers page and can add, update, delete, view customer information.

These are information employees can update:

* Password
* Full Name
* Company
* Person Identifier
* Birthday
* Gender
* Address
* Phone
* Status

### Functional Requirements

1. Employee login into admin pages.
2. Employee have permission to add, update data.

## Management of reservations

### Description

System must provide reservation manager page where employee can view, update, add new reservation.

* Check-in date
* Check-out date
* Number of people
* Status

### Functional Requirements

1. Employee login into admin pages.
2. Employee have permission to update, add reservation.

## Manager services

### Description

The system provides pages to manage hotel’s services, employees can add, update and delete services.

While customers are renting rooms, they can only offer hotel’s services.

And employees are reponsibility for add services to reservation for customers.

### Functional Requirements

1. Employee login into admin pages.
2. Employee have permission to update, add reservation.

## Manager and calculate bills

### Description

The system provides pages to calculate bills for customers

### Functional Requirements

1. Employee login into admin pages.
2. Employee have permission to update, add, view bill page.

## Manager news, feedback

### Description

The system provides pages to manage hotel’s services, employees can add, update and delete services.

### Functional Requirements

1. Employee login into admin pages.
2. Employee have permission to update, add, delete, view news, feedbacks.

# System Requirements

## Hardware

### Web Server

|  |  |
| --- | --- |
| Processor | Pentium IV or higher. |
| Memory | 512MB RAM or greater. |
| Modem/ADSL | Connecting to internet 24/24 |

### Client

|  |  |
| --- | --- |
| Processor | Pentium III or higher. |
| Memory | 1GB RAM or greater. |
| Monitor | Super VGA (800x600) or higher resolution with 256 colors. |
| Pointing Device | Microsoft Mouse, Microsoft IntelliMouse®, or compatible pointing device. |
| Modem/ADSL | Connecting to internet |

## Software

### Web Server

|  |  |
| --- | --- |
| Operation System | Windows 8 or higher |
| Browser | All except opera |
| Database | Microsoft SQL Server 2012 |
| Software | Java SDK 1.7 |
|  | Glassfish 4.0.1 |
|  | JSF Framework 2.2 |

### Client

|  |  |
| --- | --- |
| Operation System | Windows 8 or higher |
| Browser | All except opera. |

# Development Software

* Windows 8 or higher
* Netbean 7.4
* Java SDK 1.7
* Glassfish 4.0.1
* Microsoft SQL Server 2012
* All except opera

# Techonology

* JSF Framework
* J2EE
* JSF 2.2, Managed bean
* XHTML
* EJB

**Task Sheet Review 1**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Members Group** | | **Date Prepareation Of Activity Plan** | | | | |
| **No** | **Name** | **Description Task** | | **Start Date** | **End Date** | **Status** |
| 1 | Lại Văn Trách | Require | | 28/3/2017 | 28/3/2017 | Done |
| 2 | Lê Phạm Anh Vũ | Require | | 28/3/2017 | 28/3/2017 | Done |
| 3 | Nguyễn Văn Hưng | Require | | 28/3/2017 | 28/3/2017 | Done |
| 4 | Nguyễn Ngọc Nghĩa | Require | | 28/3/2017 | 28/3/2017 | Done |
|  | | | | | | |
| **Teacher** | | | **Team Leader** | | | |
|  | | |  | | | |
| **Ngô Phước Nguyên** | | | **Lại Văn Trách** | | | |

# REVIEW 2

# Use Case and Sequence Diagram

## Guest

View rooms/ read news

Register as customer

Searching/ Booking available rooms

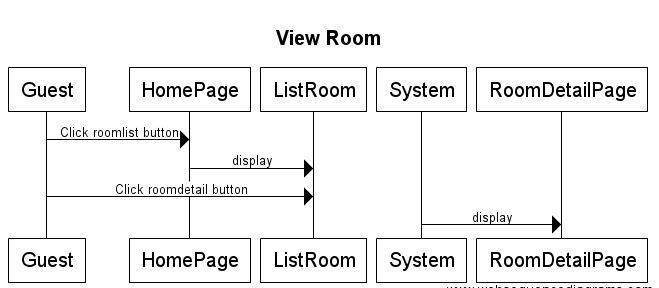
### Register as Customer

|  |  |  |
| --- | --- | --- |
| **Author** | Le Pham Anh Vu | |
| **Use Case Name** | Register as Customer | |
| **Actors** | Guest | |
| **Description** | Guest wants to register with the website to become a customer member. | |
| **Requirements** | Guest provides Guest’s information | |
| **Pre-conditions** |  | |
| **Post-conditions** | Success: Guest creates a Buyer account in this website  Fail: Refill information | |
| **Basic flow** | Actor Actions:  1. Actor clicks ‘Register’ link in Home page  3. Actor clicks on to Agree button. | System Responses:  2. System redirects to Registration Policy page.  4. System redirects to Register page and displays Registration form with the following control :  - ‘Email’ text field  - ‘Password’ text field  - ‘Confirm Password’ text field  - ‘Name’ text field  - ‘Birthday’ datebox  - ‘Identifier personal  - ‘Gender’ radio group  - ‘Address’ text area  - ‘Phone’ text field  - ‘Register’ button  - ‘Back to Home’ button  6. System checks the information  [Exception 1]  7. System inserts the account into database, show message: ‘Your account has been created!’ and then redirects to Login page. |
| **Alternative flow** | Actor Actions:  [Alternative 1]  1. Actor click ‘Back to Home’ button. | System Responses:  2. System returns to Home page. |
| **Exception** | Actor Actions:  [Exception 1]  1. Actor lefts ‘Email’ text field blank,  2. Email pattern format is invalid.  2. Email has already exists.  3. Actor lefts ‘Password’ text field blank.  4. Actor lefts ‘Confirm Password’ text field blank.  5. Password and confirm password don’t match.  6. Actor lefts ‘Name’ text field blank.  7. Actor lefts ‘Date Of Birthday’ datebox blank.  8. Input age to ‘Date Of Birthday’ datebox less than 18.  9. Actor lefts ‘Address’ text field blank.  10. Actor lefts ‘Phone’ text field blank.  11. Input phone not in numeric or phone length not in range 8 – 14 character.  12. Actor lefts ‘Email’ text field blank or input an invalid email address.  13. Actor left ‘Captcha’ text field blank or input an unmatch value. | System Responses:  System displays message to Actor :  - ‘Email cannot be blank’.  - ‘This Email has already exist!’.  - ‘Password cannot be blank’’.  - ‘Confirm Password can’t be blank’.  - ‘Confirm password not match password.’  - ‘Name cannot be blank’.  - ‘Birthday is invalid’.  - ‘Age must over 18’.  - ‘Address cannot be blank’.  - ‘Phone is invalid’.  - ‘The phone number should be numeric only and must be between 8 and 14 digits’.  - ‘Email is invalid’.  - ‘Not match the image’. |



### View rooms detail

|  |  |  |
| --- | --- | --- |
| **Author** | Nguyen Van Hung | |
| **Use Case Name** | View rooms detail | |
| **Actors** | Guests, Customers | |
| **Description** | Guest and Customers can see room details | |
| **Requirements** | Choose rooms on the website to view | |
| **Pre-conditions** |  | |
| **Post-conditions** | Success: Room detail page will show.  Fail: Home page back | |
| **Basic flow** | Actor Actions:  1. Actor clicks view room detail button on room he want to see | System Responses:  2. System redirects to Room detail page |
| **Alternative flow** |  |  |
| **Exception** | Actor Actions:  [Exception 1]  1. List room is empty | System Responses:  System displays message to Actor :  ‘Not found rooms on website’. |



## Customers

Login

Logout

Update personal information

Reading news

Manage reservations

Send feedback

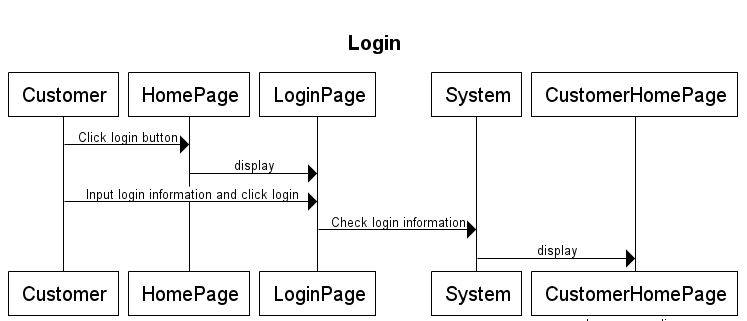
Change password

Searching available rooms

Booking available rooms

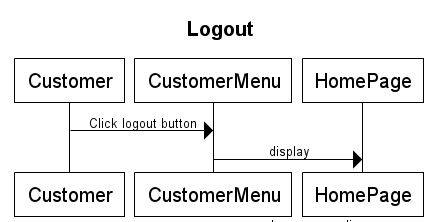
### Login

|  |  |  |
| --- | --- | --- |
| **Author** | Le Pham Anh Vu | |
| **Use Case Name** | Login | |
| **Actors** | Customer | |
| **Description** | Customer want to login to website to manage reservation or feedback to hotel. | |
| **Requirements** | Actor provides username and password | |
| **Pre-conditions** |  | |
| **Post-conditions** | Success: User logged in to website.  Fail: Refill information | |
| **Basic flow** | Actor Actions:  1. Actor go to Home page of website.  3. Actor inputs UserName and Password, then click the Login button. | System Responses:  2. System redirects to Home page and displays Login form on the left with the following control:  - ‘UserName’ text field  - ‘Password’ text field  - ‘Go’ button  4. System checks the information  [Exception 1]  5. System redirects to Actor’s Home page. |
| **Alternative flow** | Actor Actions:  [Alternative 2]  1. Actor click ‘Login’ button, inputs UserName and Password, then click the Login button.  [Alternative 1]  1. Actor click ‘Back to Home’ button. | System Responses:  2. System checks the information  [Exception 1]  3. System redirects to Actor’s Home page.  2. System returns to Home page. |
| **Exception** | Actor Actions:  [Exception 1]  1. Actor input invalid username and password. | System Responses:  - System redirect to Login page with the following controls:  - ‘UserName’ text field  - ‘Password’ text field  - ‘Login’ button.  - ‘Back to Home’ button.  System show a message box: “Invalid Username and password”.[Alternative 2] |



### Logout

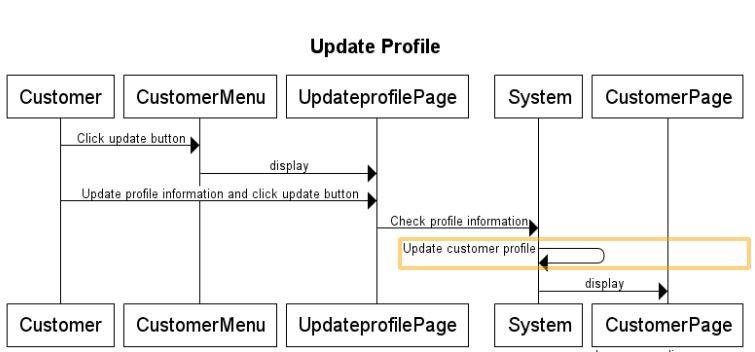
|  |  |  |
| --- | --- | --- |
| **Author** | Le Pham Anh Vu | |
| **Use Case Name** | Login | |
| **Actors** | Customer | |
| **Description** | Customer wants to log out of the System. | |
| **Requirements** |  | |
| **Pre-conditions** | Actor has logged in. | |
| **Post-conditions** | Success: Actor Logged out. | |
| **Basic flow** | Actor Actions:  1. Actor clicks ‘Log out’ button on the Actor’s Home menu. | System Responses:  2. System redirect to Logout page, remove session, and then redirect to Login page. |
| **Alternative flow** |  |  |
| **Exception** |  |  |



### Manage Account

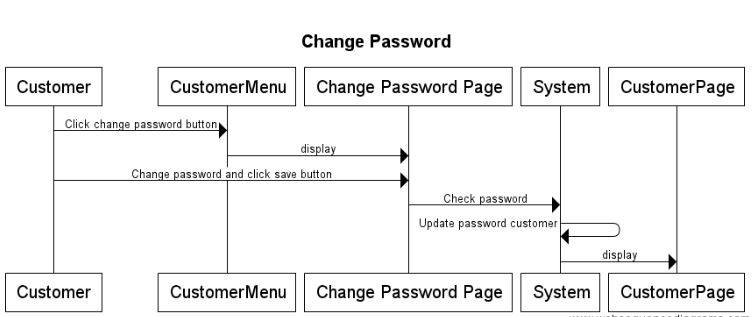
#### Update Profile

|  |  |  |
| --- | --- | --- |
| **Author** | Le Pham Anh Vu | |
| **Use Case Name** | Update Profile | |
| **Actors** | Customer | |
| **Description** | Customer wants to Update his/her profile’s information. | |
| **Requirements** | Actor provides profile information | |
| **Pre-conditions** | Actor has logged in. | |
| **Post-conditions** | Success: Profile’s information of this user has been updated.  Fail: Refill information | |
| **Basic flow** | Actor Actions:  1. Actor clicks ‘Profile’ button in Actor’s Customer’s menu.  3. Actor updates information and click ‘Save’ button. [Alternative 1] | System Responses:  2. System redirects to Update Profile page and displays Update Profile form with the following control :  - ‘UserName’ text field (read only)  - ‘Old Password’ text field  - ‘New Password’ text field  - ‘New Password Confirm’ text field  - ‘Name’ text field  - ‘Birthday’ text field  - ‘Gender’ checkbox group  - ‘Address’ text field  - ‘Phone’ text field  - ‘Email’ text field  - ‘Save’ button  - ‘Back to Home’ button  4. System checks the information  [Exception 1]  5. System update the account into database and then redirects to Actor Home page. |
| **Alternative flow** | Actor Actions:  [Alternative 1]  1. Actor click ‘Back to Home’ button. | System Responses:  2. System returns to Home page. |
| **Exception** | Actor Actions:  [Exception 1]  1. Actor lefts ‘Name’ text field blank.  2. Actor lefts ‘Birthday’ datebox blank.  3. Input age to ‘Birthday’ datebox less than 16.  4. Actor lefts ‘Address’ text field blank.  5. Actor lefts ‘Phone’ text field blank.  6. Input phone not in numeric or phone length not in range 8 – 14 character.  7. Actor lefts ‘Email’ text field blank or input an invalid email address. | System Responses:  System displays message to Actor :  - ‘Name cannot be blank’.  - ‘Birthday is invalid’.  - ‘Age must over 18’.  - ‘Address cannot be blank’.  - ‘Phone is invalid’.  - ‘The phone number should be numeric only and must be between 8 and 14 digits’.  - ‘Email is invalid’. |



#### Change password

|  |  |  |
| --- | --- | --- |
| **Author** | Le Pham Anh Vu | |
| **Use Case Name** | Change password | |
| **Actors** | Customer | |
| **Description** | Customer wants to Update his/her profile’s information. | |
| **Requirements** | Actor provides profile information | |
| **Pre-conditions** | Actor has logged in. | |
| **Post-conditions** | Success: Password of this user has been updated.  Fail: Refill password | |
| **Basic flow** | Actor Actions:  1. Actor clicks ‘Profile’ button in Actor’s Customer’s menu.  3. Actor updates information and click ‘Save’ button. [Alternative 1] | System Responses:  2. System redirects to Update Profile page and displays Update Profile form with the following control :  - ‘Old Password’ text field  - ‘New Password’ text field  - ‘New Password Confirm’ text field  - ‘Save’ button  - ‘Back to Home’ button  4. System checks the information  [Exception 1]  5. System update the account into database and then redirects to Actor Home page. |
| **Alternative flow** | Actor Actions:  [Alternative 1]  1. Actor click ‘Back to Home’ button. | System Responses:  2. System returns to Home page. |
| **Exception** | Actor Actions:  [Exception 1]  1. Actor lefts ‘New Password’ text field blank.  2. Actor lefts ‘New Confirm Password’ text field blank.  3. New Password and New Password Confirm don’t match. | System Responses:  System displays message to Actor :  - ‘Password cannot be blank’’.  - ‘Confirm Password can’t be blank’.  - ‘Confirm password not match Password.’ |

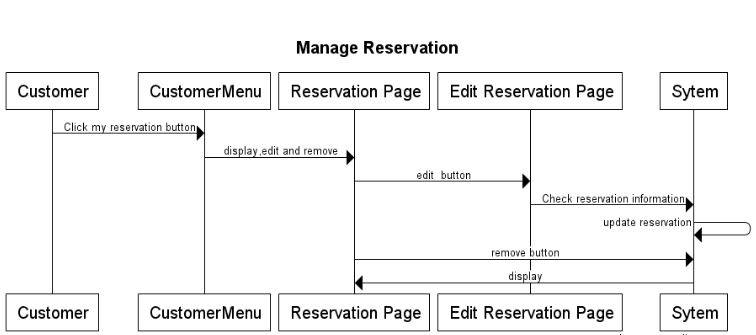


### Manage reservations

|  |  |  |
| --- | --- | --- |
| **Author** | Lai Van Trach | |
| **Use Case Name** | Manage reservation | |
| **Actors** | Customer | |
| **Description** | Customer wants to Update his/her reservations. | |
| **Requirements** | Actor view/ remove/update her/his reservation | |
| **Pre-conditions** | Actor has logged in. | |
| **Post-conditions** | Success: Refresh reservation manage page.  Fail: Show error message | |
| **Basic flow** | Actor Actions:  1. Actor clicks ‘My Resersation’ button in Actor’s Customer’s menu.  3. Actor remove or update and click ‘Save’ button. [Alternative 1] | System Responses:  2. System redirects to Reservation manage page and displays customer’s reservations.  Checkin datetime  Checkout datetime  Room name  Room type  4. System checks the information  [Exception 1]  \* Rooms have been rented  \* This perior is not availabale for renting  5. System update the account into database and then redirects to customer’s reservations. |
| **Alternative flow** | Actor Actions:  [Alternative 1]  1. Actor click ‘Back to Home’ button. | System Responses:  2. System returns to customer’s reservations and show error message. |
| **Exception** | Actor Actions:  [Exception 1]   1. Can not update perior of reservation | System Responses:  System displays message to Actor :  Can not update perior of reservation |

### Searching available room

|  |  |  |
| --- | --- | --- |
| **Author** | Lai Van Trach | |
| **Use Case Name** | Searching available | |
| **Actors** | Customer | |
| **Description** | Customer wants to search available rooms for reservation. | |
| **Requirements** | Actor provides required information for searching. | |
| **Pre-conditions** |  | |
| **Post-conditions** | Success: List available rooms  Fail: Refill Information. | |
| **Basic flow** | Actor Actions:  1. Actor fill in information for searching rooms on home page  3. Actor inputs roomtype, checkin, checkout, number of child,number of adults, and click ‘Search’ button. [Alternative 1] | System Responses:  2. Search form with the following controls:  - ‘Room type’ combobox field.  - ‘Check-in’ datetime field.  - Check-out’ datetime field.  - ‘Number of child’ combobox.  - ‘Number of adults’ combobox.  - ‘Search’ button.  - ‘Back to Home’ button.  4. System checks the information  [Exception 1].  5. System reponse list available rooms to show for actors. |
| **Alternative flow** | Actor Actions:  [Alternative 1]  1. Actor click ‘Back to Home’ button. | System Responses:  2. System returns to Customer’s Reservation page. |
| **Exception** | [Exception 1]  -Wrong input datetime input | System Responses:  System returns to Customer’s Reservation page. |



### Reading news

|  |  |  |
| --- | --- | --- |
| **Author** | Lai Van Trach | |
| **Use Case Name** | Reading news, anoucements, sales from hotel | |
| **Actors** | Customer | |
| **Description** | Read news | |
| **Requirements** |  | |
| **Pre-conditions** | Actor has logged in as Customer | |
| **Post-conditions** | Success: show news page  Fail: Refill information | |
| **Basic flow** | Actor Actions:  1. Actor clicks ‘News’ button on Menu Home page  3. Actor inputs information and submit:  - Provide subject in ‘Subject’ text field  - Provide content of the message in ‘Content’ text area  - Actor clicks ‘Send Message’ button.[Alternative 1] | System Responses:  2. System redirects to SendMessage page and displays form for Actor to list of news.  - ‘Subject’ text field  - ‘Content’ text area  - ‘Send Message’ button  - ‘Cancel’ button  4. System checks the information  [Exception 1]  5. System send a message to Seller’s email and displays a message ‘Your message have been sent to Seller successful’ |
| **Alternative flow** | Actor Actions:  [Alternative 1]  1. Actor click ‘Cancel’ button | System Responses:  - System returns previous page |
| **Exceptions** | Actor Actions:  [Exception 1]  1. Actor lefts ‘Subject’ text field blank  2. Actor lefts ‘Content’ text field blank | System Responses:  System displays message to Actor :  - ‘Subject cannot be blank. Please input subject’  - ‘Content cannot be blank. Please input content’ |

### Send Feedback to Hotel

|  |  |  |
| --- | --- | --- |
| **Author** | Lai Van Trach | |
| **Use Case Name** | Send feedback to hotel | |
| **Actors** | Customer | |
| **Description** | Customer wish to send message to hotel | |
| **Requirements** | Customer should provide subject , content of the message | |
| **Pre-conditions** | Actor has logged in as Customer | |
| **Post-conditions** | Success: send message to hotel  Fail: Refill information | |
| **Basic flow** | Actor Actions:  1. Actor clicks ‘Feedback’ button on Menu Home page  3. Actor inputs information and submit:  - Provide subject in ‘Subject’ text field  - Provide content of the message in ‘Content’ text area  - Actor clicks ‘Send Message’ button.[Alternative 1] | System Responses:  2. System redirects to SendFeedback page and displays form for Actor to input information with the following control :  - ‘Subject’ text field  - ‘Content’ text area  - ‘Send Message’ button  - ‘Cancel’ button  4. System checks the information  [Exception 1]  5. System send a message to Customer and displays a message ‘Your message have been sent to hotel successful’ |
| **Alternative flow** | Actor Actions:  [Alternative 1]  1. Actor click ‘Cancel’ button | System Responses:  - System returns previous page |
| **Exceptions** | Actor Actions:  [Exception 1]  1. Actor lefts ‘Subject’ text field blank  2. Actor lefts ‘Content’ text field blank | System Responses:  System displays message to Actor :  - ‘Subject cannot be blank. Please input subject’  - ‘Content cannot be blank. Please input content’ |

## Administrators

### Login

|  |  |  |
| --- | --- | --- |
| **Author** | Le Pham Anh Vu | |
| **Use Case Name** | Login | |
| **Actors** | Employee | |
| **Description** | Employee wants to login to Admin page. | |
| **Requirements** | Actor provides username and password | |
| **Pre-conditions** |  | |
| **Post-conditions** | Success: Employee logged in to website.  Fail: Refill information | |
| **Basic flow** | Actor Actions:  1. Actor inputs Admin page URL in Address bar of Browser.  3. Actor inputs UserName and Password, then click the Login button. [Alternative 1] | System Responses:  2. System redirects to Login page and displays Login form with the following control:  - ‘UserName’ text field  - ‘Password’ text field  - ‘Login’ button  4. System checks the information  [Exception 1]  5. System redirects to Admin Home page. |
| **Alternative flow** |  |  |
| **Exception** | Actor Actions:  [Exception 1]  1. Actor lefts ‘UserName’ text field blank. 2. Actor lefts ‘Password’ text field blank. | System Responses:  System displays message to Guest :  - ‘UserName’ cannot be blank’.  - ‘Password cannot be blank’. |

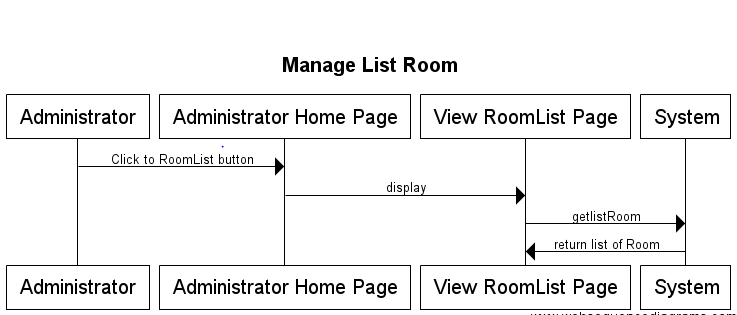
### Change Password

|  |  |  |
| --- | --- | --- |
| **Author** | Le Pham Anh Vu | |
| **Use case name** | Change Password | |
| **Actors** | Employee | |
| **Description** | Change Password | |
| **Requirements** | Actor provides Username and Password | |
| **Pre-conditions** | Actor has logged in as Employee | |
| **Post-conditions** | Success: The password of Employee has changed. | |
| **Basic flow** | Actor actions:  1. From ‘Admin’ page, actor clicks on to the ‘Change Password’ button.  3. Actor clicks on to ‘Save’ button to finish adding. [Alternative 1] | System response:  2. System redirects to ‘Change Password’ page and display form with the following controls:  ‘Old Password’ textfield.  ‘New Password’ textfield.  ‘Confirm New Password’ textfield.  ‘Save’ button.  4. System checks input information.  [Exception 1]  5. System adds new Administrator into the database. |
| **Alternative flow** | [Alternative 1]  1. Actor clicks on to ‘Admin Home Page’ button. | 2. System redirects to ‘Admin’ page. |
| **Exception** | [Exception 1]  1. Actor lefts ‘Old Password’ textfield.  2. Actor lefts ‘New Password’ textfield.  3. Actor lefts ‘Confirm New Password’ textfield.  4. ‘New Password’ textfield is not same as ‘Confirm New Password’ textfield. | System displays a message to notice actor:  “Old Password cannot be blank. Please input Old Password.”  “New Password cannot be blank. Please input New Password.”  “Confirm New Password cannot be blank. Please input Confirm Password.”  “New Password and Confirm New Password must be same.” |

### Manage List of Rooms

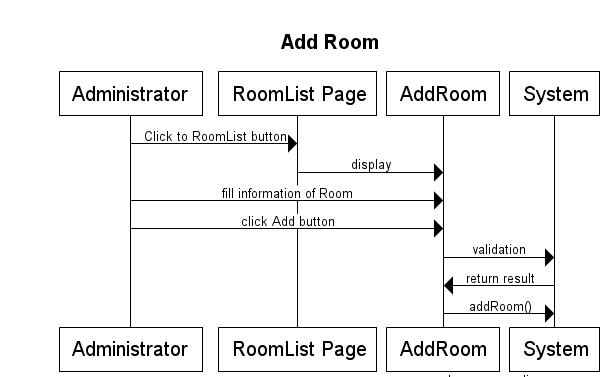
#### View List of Rooms

|  |  |  |
| --- | --- | --- |
| **Author** | Nguyen Van Hung | |
| **Use Case Name** | View List of Rooms | |
| **Actors** | Customers | |
| **Description** | Customers wish to view list of rooms | |
| **Requirements** |  | |
| **Pre-conditions** | Actor has logged in as Customer | |
| **Post-conditions** | Success: List of rooms show successful | |
| **Basic flow** | Actor Actions:  1. Actor clicks ‘Managing Rooms’ button on Admin page | System Responses:  2. System redirects to View List of Service page and displays form for Actor :  -‘Room Id’ label  -‘Room Name’ label  -‘Room type label  -‘Room status’ label  -‘Add Date’ label  -‘Image1’ label  -‘Image2’ label  -‘Image3’ label  -‘Add’ button  -‘Modify’ button  -‘Delete’ button  [Alternative 1] [Alternative 2] [Alternative 3] [Alternative 4] |
| **Alternative flow** | Actor Actions:  [Alternative1]  Actor clicks ‘Add’ button on View List of Rooms page  [Alternative 2]  Actor clicks ‘Modify’ button on View List of Service page  [Alternative 3]  Actor clicks Activate button on View List of Service page  [Alternative 4]  Actor clicks Delete button on View List of Service page | System Responses:  System redirects to Add Service page  System redirects to Modify Service page  System change status of bank to deactivate and change label of button to Deactivate  View in Delete Banks usecase |
| **Exceptions** |  |  |



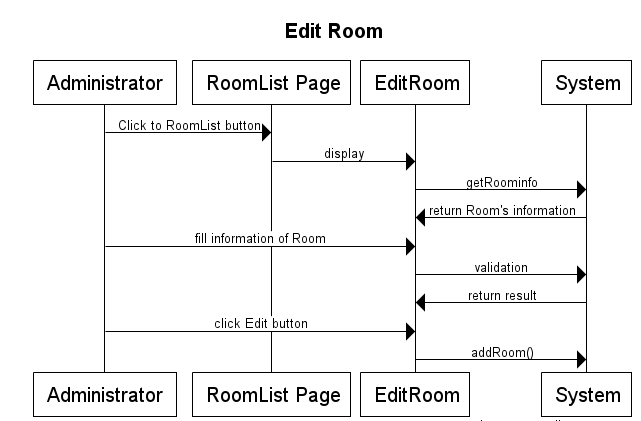
#### Add Room

|  |  |  |
| --- | --- | --- |
| **Author** | Nguyen Van Hung | |
| **Use Case Name** | Add Room | |
| **Actors** | Employee | |
| **Description** | Employee wish to add new room | |
| **Requirements** | Employee must be provide name of room and room type | |
| **Pre-conditions** | Actor has logged in as Administrator | |
| **Post-conditions** | Success: Add bank successful  Fail: Refill information | |
| **Basic flow** | Actor Actions:  1. Actor clicks ‘Add’ button on View list of service page  3. Actor inputs property type information :  - Provide the bank’s name in ‘Name’ text field.  -Choose type of rooms in listbox  5. Actor click ‘Add’ button.  [Alternative 1]  [Alternative 2] | System Responses:  2. System redirects to this page and displays form for Actor to input information with the following control :  -‘Room name’ text field  -‘Status’ list box (Available/Unavailable)  - ‘RoomType’ combobox  -‘Add Date’ textfield (auto fill with current date)  -‘Add’ button  -‘Cancel’ button  -‘Clear’ button  4. System checks information input [Exception 1]  6. System add bank into database then display message ‘Your room has been add successful’. |
| **Alternative flow** | Actor Actions:  [Alternative 1]  Actor click ‘Clear’ button  [Alternative 2]  Actor click ‘Cancel’ button | System Response:  All field will blank  System redirects to View List of Bank page. |
| **Exceptions** | Actor Actions:  [Exception 1]  1. Employee left ‘Name’ text field blank  2. Length of name <10 char  3. Value in ‘Name’ text field has been in database | System Response:  1. System displays message to notice seller: ‘Name of room can’t be blank. Please enter name for room.’  2. System displays message to notice customer: ‘Length of bank’s name can’t be large than 10 char.’  3. System displays message to notice customer: ‘This name of room has been in database. Please check and enter another name.’ |



#### Modify room

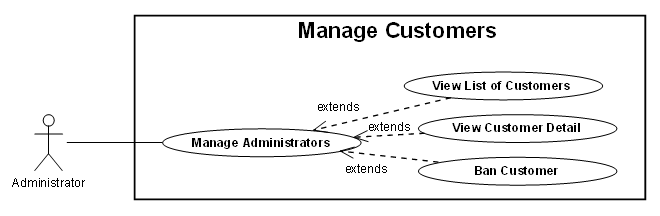
|  |  |  |
| --- | --- | --- |
| **Author** | Nguyen Van Hung | |
| **Use Case Name** | Modify Room | |
| **Actors** | Employee | |
| **Description** | Employee wish to modify room | |
| **Requirements** | Employee must be provide name of room and correct roomtype | |
| **Pre-conditions** | Actor has logged in as Customer | |
| **Post-conditions** | Success: Modify room successful  Fail: Refill information | |
| **Basic flow** | Actor Actions:  1. Actor clicks Modify button on View list of service page  3. Actor inputs property type information :  - Provide the bank’s name in ‘Name’ text field.  - Provide the bank’s address in ‘Address’ text field.  - Provide the Url of bank’s webservice in ‘URL’ text field.  5. Actor click ‘Modify’ button.  [Alternative 1]  [Alternative 2] | System Responses:  2. System redirects to this page and displays form for Actor to input information with the following control :  -‘Name’ text field  -‘Status’ list box  -‘Add Date’ textfield (auto fill with value in table Banks)  -‘Modify’ button  -‘Cancel’ button  -‘Refill Data’ button  System fill name , url, status for bank  4. System checks information input [Exception 1]  6. System add bank into database then display message ‘Your bank has been add successful’. |
| **Alternative flow** | Actor Actions:  [Alternative 1]  Actor click ‘Refill Data’ button  [Alternative 2]  Actor click ‘Cancel’ button | System Response:  All field wil reload data  System redirects to View List of Bank page. |
| **Exceptions** | Actor Actions:  [Exception 1]  1. Employee left ‘Name’ text field blank  2. Length of name <10 char  3. Value in ‘Name’ text field has been in database | System Response:  1. System displays message to notice seller: ‘Name of room can’t be blank. Please enter name for room.’  2. System displays message to notice customer: ‘Length of bank’s name can’t be large than 10 char.’  3. System displays message to notice customer: ‘This name of room has been in database. Please check and enter another name.’ |



#### Delete Room

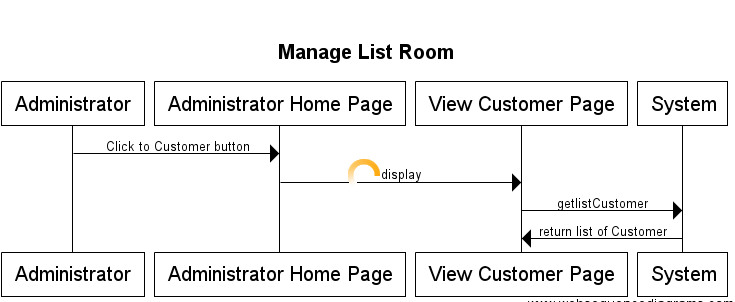
|  |  |  |
| --- | --- | --- |
| **Author** | Nguyen Van Hung | |
| **Use Case Name** | Delete Room | |
| **Actors** | Customer | |
| **Description** | Customer wish to delete room | |
| **Requirements** | Customer must be choose room to delete | |
| **Pre-conditions** | Actor has logged in as Customer | |
| **Post-conditions** | Success: Delete room successful  Fail: Error Page show | |
| **Basic flow** | Actor Actions:  1. Actor clicks Delete button on View list of service page  3. Actor clicks Yes button on popup message  [Alternative 1] | System Responses:  2. System show popup message :”Do you want to remove this bank : ” + bank’s name. With Yes and No button  4. System delete bank from database then display message ‘Your bank has been delete successful’. |
| **Alternative flow** | Actor Actions:  [Alternative 1]  Actor clicks No button on popup message | System Responses:  System returns to current page. |
| **Exceptions** |  |  |

### Manage Customers



#### View List of Customers

|  |  |  |
| --- | --- | --- |
| **Author** | Le Pham Anh Vu | |
| **Use case name** | View List of Customers | |
| **Actors** | Administrator | |
| **Description** | View List of Customers | |
| **Requirements** |  | |
| **Pre-conditions** | Actor has already logged into the System as Administrator | |
| **Post-conditions** | Success: View List of Customers | |
| **Basic flow** | Actor actions:  1. From ‘Admin’ page, actor clicks on to ‘Managing Customers’ button. | System response:  2. System redirects to ‘Managing Customers’ page and display list of Customers in table form:   * CustomerID * Email * Phone * Gender * FullName * Status (Active/Deactivate button) |
| **Alternative flow** |  |  |
| **Exception** |  |  |



#### View Customer Detail

|  |  |  |
| --- | --- | --- |
| **Author** | Le Pham Anh Vu | |
| **Use case name** | View Customer Detail | |
| **Actors** | Administrator | |
| **Description** | View Customer Detail | |
| **Requirements** |  | |
| **Pre-conditions** | Actor has already logged into the System as Administrator | |
| **Post-conditions** | Success: View Customer Detail | |
| **Basic flow** | Actor actions:  1. From ‘View List of Customers’ page, actor clicks on to a ‘Customer Name’ hyperlink. | System response:  2. System redirects to ‘Customer Detail’ page and display Customer information:   * Email * Person Indentity * Date of Birth * FullName * Gender * Company * Address * Phone * Email * Status |
| **Alternative flow** |  |  |
| **Exception** |  |  |

#### Lock Customer

|  |  |  |
| --- | --- | --- |
| **Author** | Le Pham Anh Vu | |
| **Use case name** | Lock Customer | |
| **Actors** | Employee | |
| **Description** | Lock Customer | |
| **Requirements** |  | |
| **Pre-conditions** | Actor has already logged into the System as Employee | |
| **Post-conditions** | Success: A Customer is locked | |
| **Basic flow** | Actor actions:  1. From ‘View List of Customers’ page, actor clicks on to the ‘Status’ button of the Customer whom actor wants to ban (symbol of Status button is activate).[Alternative 1]  3. Actor clicks ‘Yes’ button.[Alternative 2] | System response:  2. A confirms message box appears:”Are you sure that you want to ban this Customer?”  4. The symbol of ‘Status’ button will change to Activate symbol.  5. System saves Customers information. |
| **Alternative flow** | [Alternative 1]  1. The symbol of Status button is deactivate, Actor clicks on to Status button.  3. Actor clicks ‘Yes’ button.[Alternative 2]  [Alternative 2]  1. Actor clicks ‘No’ button. | 2. A confirms message box appears:”Are you sure that you want to activate this Customer?”  4. The symbol of ‘Status’ button will change to Deactivate symbol.  5. System saves Customers information.  2. System redirects to List of Customers page. |
| **Exception** |  |  |

### Manage Employee

#### View List of Employee

|  |  |  |
| --- | --- | --- |
| **Author** | Le Pham Anh Vu | |
| **Use case name** | View List of Employee | |
| **Actors** | Administrator | |
| **Description** | View List of Employee | |
| **Requirements** |  | |
| **Pre-conditions** | Actor has already logged into the System as Employee and Role property is true. | |
| **Post-conditions** | Success: View List of Administrators | |
| **Basic flow** | Actor actions:  1. From ‘Admin’ page, actor clicks on to the ‘Employee’ button. | System response:  2. System redirects to ‘Employee’ page and display list of all of Employee in table form:   * EmployeeID * UserName * Gender * Email * Delete button   The current Employee record will not appear in this table. |
| **Alternative flow** |  |  |
| **Exception** |  |  |

#### Add Employee

|  |  |  |
| --- | --- | --- |
| **Author** | Le Pham Anh Vu | |
| **Use case name** | Add Employee | |
| **Actors** | Employee | |
| **Description** | Add a Employee | |
| **Requirements** | Actor provides Username, Password, DOB, Gender, Email | |
| **Pre-conditions** | Actor has already logged into the System as Employee and Role property is false. | |
| **Post-conditions** | Success: An Employee is added into the database.  Failed: Refill information | |
| **Basic flow** | Actor actions:  1. From ‘View List of Administrators’ page, actor clicks on to the ‘Add ‘ button.  3. Actor clicks on to ‘Add’ button to finish adding. [Alternative 1] | System response:  2. System redirects to ‘Add Employee’ page and display form with the following controls:  ‘Username’ textfield.  ‘Password’ textfield.  ‘Confirm Password’ textfield.  ‘DOB’ datepicker  ‘Gender’ radio button  ‘Email’ textfield  ‘Add’ button.  4. System checks input information.  [Exception 1]  5. System adds new Employee into the database. |
| **Alternative flow** | [Alternative 1]  1. Actor clicks on to ‘Back’ button. | 2. System redirects to ‘Login Admin’ page. |
| **Exception** | [Exception 1]  1. Actor lefts ‘Username’ textfield.  2. Actor lefts ‘Password’ textfield.  3. Actor lefts ‘Confirm Password’ textfield.  4. Password textfield is not same as Confirm Password textfield.  5. Username has already existed in the database. | System displays a message to notice actor:  “Username cannot be blank. Please input Username.”  “Password cannot be blank. Please input Password.”  “Confirm Password cannot be blank. Please input Confirm Password.”  “Password and Confirm Password must be same.”  “The username has already existed. Please input another username.”  “Email must pattern format” |

#### Delete Employee

|  |  |  |
| --- | --- | --- |
| **Author** | Le Pham Anh Vu | |
| **Use case name** | Delete Employee | |
| **Actors** | Employee | |
| **Description** | Delete an Employee | |
| **Requirements** |  | |
| **Pre-conditions** | Actor has already logged into the System as Employee and Role property is true. | |
| **Post-conditions** | Success: Deleted an Employee | |
| **Basic flow** | Actor actions:  1. From ‘View List of Employee page, actor clicks on to the ‘Delete’ button of Employee which actor wants to delete.  3. Actor clicks ‘Yes’ button.[Alternative 1] | System response:  2. A confirms message box appears:”Are you sure that you want to delete this Employee?”  4. System deletes the employee in the database. |
| **Alternative flow** | [Alternative 1]  1. Actor clicks on to ‘No’ button | 2. System redirects to ‘View List of Employee page |
| **Exception** |  |  |

### Manage Service

#### View List of Service

|  |  |  |
| --- | --- | --- |
| **Author** | Nguyen Ngoc Nghia | |
| **Use Case Name** | View List of Service | |
| **Actors** | Employee | |
| **Description** | Employee wish to view list of banks | |
| **Requirements** |  | |
| **Pre-conditions** | Actor has logged in as Employee | |
| **Post-conditions** | Success: List of Service show successful | |
| **Basic flow** | Actor Actions:  1. Actor clicks ‘Managing Service button on Admin page | System Responses:  2. System redirects to View List of Service page and displays form for Actor :  -‘SerID’ label  -‘Name’ label  -‘Price’ label  -‘Add’ button  -‘Update’ button  -‘Delete’ button  [Alternative 1] [Alternative 2] [Alternative 3] [Alternative 4] |
| **Alternative flow** | Actor Actions:  [Alternative1]  Actor clicks ‘Add’ button on View List of Service page  [Alternative 2]  Actor clicks ‘Update’ button on View List of Service page  [Alternative 3]  Actor clicks Delete button on View List of Service page | System Responses:  System redirects to Add Service page  System redirects to Update Service page  View in Delete Banks usecase |
| **Exceptions** |  |  |

#### Add service

|  |  |  |
| --- | --- | --- |
| **Author** | Nguyen Ngoc Nghia | |
| **Use Case Name** | Add Service | |
| **Actors** | Employee | |
| **Description** | Employee wish to add new Service | |
| **Requirements** | Add new Service | |
| **Pre-conditions** | Actor has logged in as Employee | |
| **Post-conditions** | Success: Add service successful  Fail: Refill information | |
| **Basic flow** | Actor Actions:  1. Actor clicks ‘Add’ button on View list of service page  3. Actor inputs property type information :  - Provide the service’s name in ‘ServiceName’ text field.  - Provide the service’s price in ‘Price text field.  5. Actor click ‘Add’ button.  [Alternative 1]  [Alternative 2] | System Responses:  2. System redirects to this page and displays form for Actor to input information with the following control :  -‘ServiceName text field  -‘Price textfield  -‘Add’ button  -‘Cancel’ button  -‘Clear’ button  4. System checks information input [Exception 1]  6. System add service into database then display message ‘Your service has been add successful’. |
| **Alternative flow** | Actor Actions:  [Alternative 1]  Actor click ‘Clear’ button  [Alternative 2]  Actor click ‘Cancel’ button | System Response:  All field will blank  System redirects to View List of service page. |
| **Exceptions** | Actor Actions:  [Exception 1]  1. Service left ‘ServiceName’ text field blank  2. Value in ‘Name’ text field has been in database  4. Service lefts ‘Price’ text field blank | System Response:  1. System displays message to notice seller: ‘ServiceName of Service can’t be blank. Please enter name for bank.’  2. System displays message to notice seller: ‘This Service of Service has been in database. Please check and enter another name.’  3. System displays message to notice seller: ‘Price of Service can’t be blank. Please enter Price of Service’s Hotel.’ |

#### Update Service

|  |  |  |
| --- | --- | --- |
| **Author** | Nguyen Ngoc Nghia | |
| **Use Case Name** | Update Bank | |
| **Actors** | Employee | |
| **Description** | Employee wish to update service | |
| **Requirements** | Update Service | |
| **Pre-conditions** | Actor has logged in as Employee | |
| **Post-conditions** | Success: Update bank successful  Fail: Refill information | |
| **Basic flow** | Actor Actions:  1. Actor clicks Update button on View list of service page  3. Actor inputs property type information :  - Service left ‘ServiceName’ text field blank  - Value in ‘ServiceName’ text field has been in database  - Service lefts ‘Price’ text field blank 5. Actor click ‘Modify’ button.  [Alternative 1]  [Alternative 2] | System Responses:  2. System redirects to this page and displays form for Actor to input information with the following control :  -‘ServiceName text field  -‘Price textfield  -‘Update button  -‘Cancel’ button  -‘Clear’ button  4. System checks information input [Exception 1]  6. System add service into database then display message ‘Your service has been update successful’. |
| **Alternative flow** | Actor Actions:  [Alternative 1]  Actor click ‘Refill Data’ button  [Alternative 2]  Actor click ‘Cancel’ button | System Response:  All field wil reload data  System redirects to View List of Service page. |
| **Exceptions** | Actor Actions:  [Exception 1]  1. Buyer left ‘Name’ text field blank  2. Length of name >50 char  3. Value in ‘Name’ text field has been in database  4. Buyer lefts ‘Url’ text field blank | System Response:  1. System displays message to notice seller: ‘ServiceName of Service can’t be blank. Please enter name for bank.’  2. System displays message to notice seller: ‘This Service of Service has been in database. Please check and enter another name.’  3. System displays message to notice seller: ‘Price of Service can’t be blank. Please enter Price of Service’s Hotel.’ |

#### Delete Service

|  |  |  |
| --- | --- | --- |
| **Author** | Nguyen Ngoc Nghia | |
| **Use Case Name** | Delete Bank | |
| **Actors** | Employee | |
| **Description** | Employee wish to update service | |
| **Requirements** | Delete Service | |
| **Pre-conditions** | Actor has logged in as Employee | |
| **Post-conditions** | Success: Delete bank successful  Fail: Error Page show | |
| **Basic flow** | Actor Actions:  1. Actor clicks Delete button on View list of service page  3. Actor clicks Yes button on popup message  [Alternative 1] | System Responses:  2. System show popup message :”Do you want to remove this bank : ” + service’s name. With Yes and No button  4. System delete bank from database then display message ‘Your service has been delete successful’. |
| **Alternative flow** | Actor Actions:  [Alternative 1]  Actor clicks No button on popup message | System Responses:  System returns to current page. |
| **Exceptions** |  |  |

### Manage Reservation

#### View List of Reservation

|  |  |  |
| --- | --- | --- |
| **Author** | Lại Văn Trách | |
| **Use Case Name** | View List of Reservation | |
| **Actors** | Customers | |
| **Description** | Customers wish to view list of Reservation | |
| **Requirements** |  | |
| **Pre-conditions** | Actor has logged in as Customer | |
| **Post-conditions** | Success: List of Reservation show successful | |
| **Basic flow** | Actor Actions:  1. Actor clicks ‘Managing Reservation ‘ button on Admin page | System Responses:  2. System redirects to View List of Service page and displays form for Actor :  -‘ReservationID’ label  -‘StartDate’ label  -‘EndDate label  -‘NumberofPeople’ label  -‘Paid’ label  -‘Status’ label  -‘Add’ button  -‘Modify’ button  -‘Delete’ button  [Alternative 1] [Alternative 2] [Alternative 3] [Alternative 4] |
| **Alternative flow** | Actor Actions:  [Alternative1]  Actor clicks ‘Add’ button on View List of Reservation page  [Alternative 2]  Actor clicks ‘Modify’ button on View List of Reservation page  [Alternative 3]  Actor clicks Activate button on View List of Reservation page  [Alternative 4]  Actor clicks Delete button on View List of Reservation page | System Responses:  System redirects to Add Reservation page  System redirects to Modify Reservation page  System change status of Reservation to deactivate and change label of button to Deactivate  View in Delete Reservation usecase |
| **Exceptions** |  |  |

#### Add Reservation

|  |  |  |
| --- | --- | --- |
| **Author** | Lại Văn Trách | |
| **Use Case Name** | View List of Reservation | |
| **Actors** | Employee | |
| **Description** | Employee wish to add Reservation | |
| **Requirements** |  | |
| **Pre-conditions** | Actor has logged in as Employee | |
| **Post-conditions** | Success: Add Reservation successful | |
| **Basic flow** | Actor Actions:  1. Actor clicks ‘Add’ button on View list of reservation page  3. Actor inputs property type information :  - Provide the reservation’s id in ‘Name’ text field.  -Choose StartDate to datepicker  -Choose EndDate to datepicker  5. Actor click ‘Add’ button.  [Alternative 1]  [Alternative 2] | System Responses:  2. System redirects to this page and displays form for Actor to input information with the following control :  -‘Room name’ text field  -‘StartDate textfield  - EndDate textfield  - ‘NumberofPeople textfield  - Paid textfield  -‘Add’ button  -‘Cancel’ button  -‘Clear’ button  4. System checks information input [Exception 1]  6. System add bank into database then display message ‘Your reservation has been add successful’. |
| **Alternative flow** | Actor Actions:  [Alternative 1]  Actor click ‘Clear’ button  [Alternative 2]  Actor click ‘Cancel’ button | System Response:  All field will blank  System redirects to View List of reservation page. |
| **Exceptions** | Actor Actions:  [Exception 1]  1. Employee left ‘Name’ text field blank  2. Length of name <10 char  3. Value in ‘Name’ text field has been in database | System Response:  1. System displays message to notice seller: ‘Name of room can’t be blank. Please enter name for room.’  2. System displays message to notice customer: ‘Length of bank’s name can’t be large than 10 char.’  3. System displays message to notice customer: ‘This name of room has been in database. Please check and enter another name.’ |

#### Update Reservation

|  |  |  |
| --- | --- | --- |
| **Author** | Lại Văn Trách | |
| **Use Case Name** | Update of Reservation | |
| **Actors** | Employee | |
| **Description** | Employee wish to view list of Reservation | |
| **Requirements** |  | |
| **Pre-conditions** | Actor has logged in as Employee | |
| **Post-conditions** | Success: Modify reservation successful  Fail: Refill information | |
| **Basic flow** | Actor Actions:  1. Actor clicks ‘Add’ button on View list of reservation page  3. Actor inputs property type information :  - Provide the reservation’s id in ‘Name’ text field.  -Choose StartDate to datepicker  -Choose EndDate to datepicker  5. Actor click ‘Add’ button.  [Alternative 1]  [Alternative 2] | System Responses:  2. System redirects to this page and displays form for Actor to input information with the following control :  -‘Room name’ text field  -‘StartDate textfield  - EndDate textfield  - ‘NumberofPeople textfield  - Paid textfield  -‘Add’ button  -‘Cancel’ button  -‘Clear’ button  4. System checks information input [Exception 1]  6. System add bank into database then display message ‘Your reservation has been add successful’. |
| **Alternative flow** | Actor Actions:  [Alternative 1]  Actor click ‘Clear’ button  [Alternative 2]  Actor click ‘Cancel’ button | System Response:  All field will blank  System redirects to View List of reservation page. |
| **Exceptions** | Actor Actions:  [Exception 1]  1. Employee left ‘Name’ text field blank  2. Length of name <10 char  3. Value in ‘Name’ text field has been in database | System Response:  1. System displays message to notice seller: ‘Name of room can’t be blank. Please enter name for room.’  2. System displays message to notice customer: ‘Length of bank’s name can’t be large than 10 char.’  3. System displays message to notice customer: ‘This name of room has been in database. Please check and enter another name.’ |

#### Delete Reservation

|  |  |  |
| --- | --- | --- |
| **Author** | Lại Văn Trách | |
| **Use Case Name** | Delete Reservation | |
| **Actors** | Employee | |
| **Description** | Customers wish to view list of Reservation | |
| **Requirements** |  | |
| **Pre-conditions** | Actor has logged in as Employee | |
| **Post-conditions** | Success: Delete reservation successful  Fail: Error Page show | |
| **Basic flow** | Actor Actions:  1. Actor clicks Delete button on View list of reservation page  3. Actor clicks Yes button on popup message  [Alternative 1] | System Responses:  2. System show popup message :”Do you want to remove this bank : ” + bank’s name. With Yes and No button  4. System delete bank from database then display message ‘Your reservation has been delete successful’. |
| **Alternative flow** | Actor Actions:  [Alternative 1]  Actor clicks No button on popup message | System Responses:  System returns to current page. |
| **Exceptions** |  |  |

### Manage FeedBack

#### View List of Questions

|  |  |  |
| --- | --- | --- |
| **Author** | Nguyen Ngoc Nghia | |
| **Use case name** | View List of Questions | |
| **Actors** | Employee | |
| **Description** | View List of Questions | |
| **Requirements** |  | |
| **Pre-conditions** | Already login into the System as Employee | |
| **Post-conditions** | Success: View List of Questions | |
| **Basic flow** | Actor actions:  1. From ‘Employee’ page, actor clicks on to the ‘Managing Questions’ button. | System response:  2. System redirects to ‘Managing Questions’ page and display list of all of Questions in table form with control :  ‘FeedBackID’ label  ‘DateReceive’ label  ‘DateFeedBack’ label  ‘Question’ label  ‘FeedBack’ button  ‘Delete’ button |
| **Alternative flow** |  |  |
| **Exception** |  |  |

#### 8.3.10.1 FeedBack

|  |  |  |
| --- | --- | --- |
| **Author** | Nguyen Hai Dang | |
| **Use case name** | Modify FeedBack | |
| **Actors** | Employee | |
| **Description** | FeedBack to Customer | |
| **Requirements** |  | |
| **Pre-conditions** | Already login into the System as Employee | |
| **Post-conditions** | Success: A FeedBack is updated in the database. | |
| **Basic flow** | Actor actions:  1. From ‘View List of FeedBacks’ page, actor clicks on to the ‘Feedback button  3. Actor:   * Inputs FeedBack into ‘FeedBack’ textfield. * Inputs Answer into ‘Feedback textfield. * Clicks on to ‘Save’ button to finish adding. [Alternative 1] [Alternative 2] | System response:  2. System redirects to ‘Modify FeedBack’ page and display form with the following controls:  ‘FeedBack’ textfield.  ‘DateReceive’ textfield.  DateFeedBack’ textfield  ‘Save’ button.  ‘Refill Data’ button  ‘Cancel’ button  4. System checks information.  [Exception 1] |
| **Alternative flow** | [Alternative 1]  1. Actor clicks on to ‘Cancel’ button  [Alternative 2]  1. Actor clicks on to ‘Refill Data’ button | 1. System redirects to ‘View List of FeedBacks’ page  2. Reload data to all field |
| **Exception** | [Exception 1]  1. Actor lefts ‘FeedBack’ textfield. | System displays a message to notice actor:  “FeedBack cannot be blank. Please input FeedBack.” |

#### Delete FeedBack

# 

# Entity Relationship Diagram (ERD)

\*Entity Relationship Diagram (ERD)

Feedback

Room Type

Has

n

1

Room

Reserver

Customer

Book

Reservation

Has

Has

Bill

Has

Cus-Service

Service

Has

Employee

n

n

1

n

n

1

n

n

1

1

n

1

News

Feedback

Post

1

n

1

n

1

n

\*Entity and Properties

* **RoomType:**

|  |
| --- |
| * **Room:**   Rooms |

**Employee:**

Employee

* C**ustomer**

Customer

* **Reservation:**

Reservation

* **Service:**

Service

* **Bill:**

Bill

**V.Algorithms**

* Empty room search

Begin

End Date - Start date

Empty Period >= Period

End

Get in database

True

I = i+1

Room

False

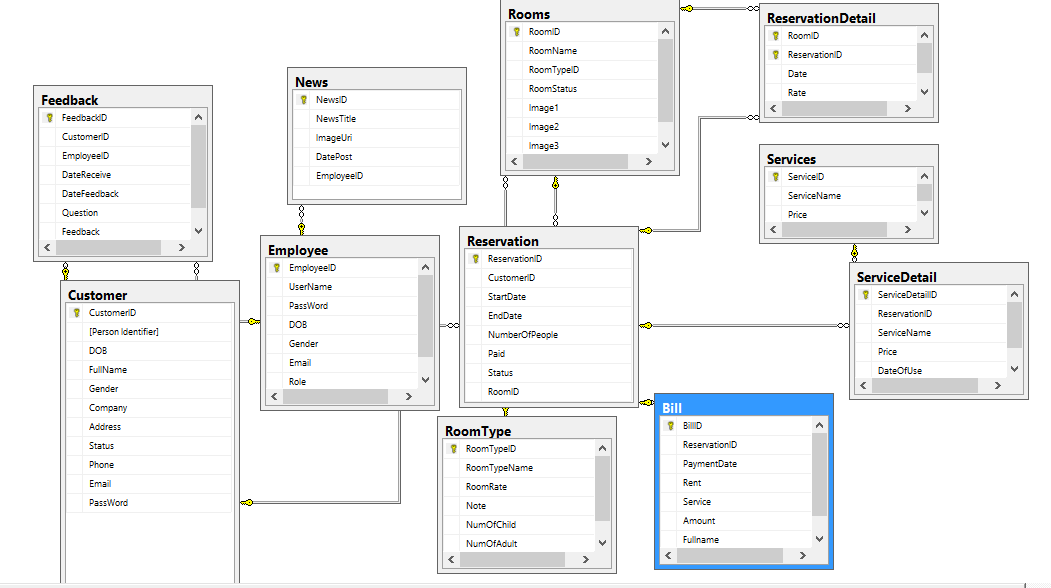
1. **TaskSheet**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Project Ref.No: 3** | | **Project Title:**  **Hotel Management** | **Date of Preparation of Activity Plan** | | | |
| **Sr. No.** | **Task** | | **Actual Start Date** | **Actual Days** | **Team Member Names** | **Status** |
| 01 | Data Flow Diagram | | Mar, 29, 2017 | 5 | All Members | Completed |
| 02 | Entity Relationship Diagram (ERD) | | All Members | Completed |
| 03 | Entity and Properties | | All Members | Completed |
| 04 | Database Relationship Diagram (DRD) | | All Members | Completed |
| 05 | Database Structure | | All Members | Completed |
| 06 | Algorithms | | All Members | Completed |
| 07 | Tasksheet | | All Members | Completed |

|  |  |  |
| --- | --- | --- |
|  | Prepare By: Group 3 | Approved By: Faculty |
| Date: Mar, 2017 | Team Leader  **Lại Văn Trách** | **Ngo Phuoc Nguyen** |

REVIEW 3

# Database Relationship Diagram



# Table Definition

## Room type

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **RoomType** | | | | | | |
| **Column Name** | **Data Type** | **Allow Null** | **Default Value** | **Key** | **Validation** | **Description** |
| RoomTypeID | Int | No |  | Primary |  | ID of **RoomType** |
| RoomTypeName | varchar(50) | No |  |  |  | Name of **RoomType** |
| RoomRate | money | No |  |  |  | Rate of **RoomType** |
| Note | varchar(100) |  |  |  |  | Note of **RoomType** |

### Contrainst:

|  |  |  |  |
| --- | --- | --- | --- |
| **Name** | **Add** | **Delete** | **Update** |
| **RoomType** | + | +  (can not delete if have reference to room table) | + |

## Room

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Room** | | | | | | |
| **Column Name** | **Data Type** | **Allow Null** | **Default Value** | **Key** | **Validation** | **Description** |
| RoomID | Int | No |  | Primary |  | ID of **Room** |
| RoomName | varchar(50) | No |  |  | No duplication | Name of **Room** |
| RoomTypeID | Int | No |  | Foreign key |  | RoomTypeID of **Room** |
| StatusID | int | No |  | Foreign key |  | StatusID of **Room** |

### Contrainst:

|  |  |  |  |
| --- | --- | --- | --- |
| **Name** | **Add** | **Delete** | **Update** |
| **Room** | + | + | + |

## Employee

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Employee** | | | | | | |
| **Column Name** | **Data Type** | **Allow Null** | **Default Value** | **Key** | **Validation** | **Description** |
| EmployeeID | int | No |  | Primary |  | ID of **Employee** |
| EmpName | varchar(30) | No |  |  |  | Name of **Employee** |
| DOB | Datetime | No |  |  |  | DOB to **Employee** |
| Gender | varchar(10) | No |  |  |  | Gender of **Employee** |
| Address | varchar(10) |  |  |  |  | Address of **Employee** |
| Phone | Varchar(20) |  |  |  |  | Phone of **Employee** |
| Job | Varchar(50) |  |  |  |  | Job of **Employee** |
| Status | Varchar(10) | no |  |  |  | Status of **Employee** |
| Password | Varchar(50) | no |  |  |  | Password to Login |

### Contrainst:

|  |  |  |  |
| --- | --- | --- | --- |
| **Name** | **Add** | **Delete** | **Update** |
| **Employee** | + | - | + |

## Customer

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Customer** | | | | | | |
| **Column Name** | **Data Type** | **Allow Null** | **Default Value** | **Key** | **Validation** | **Description** |
| CusID | int | No |  | Primary |  | ID of **Customer** |
| FullName | nvarchar(30) | No |  |  |  | Name of **Customer** |
| DOB | Datetime | No |  |  |  | DOB to **Customer** |
| Gender | varchar(10) | No |  |  |  | Gender of **Customer** |
| Address | nvarchar(100) | No |  |  |  | Address of **Customer** |
| Phone | Varchar(12) |  |  |  |  | Phone of **Customer** |
| Company | Varchar(50) |  |  |  |  | Company of **Customer** |
| Email | Varchar(100) |  |  |  |  | Email of **Customer** |
| Person Identifier | Varchar(50) | no |  |  |  | Passport of **Customer** |
| Status | varchar(10) | No |  |  |  | Status of **Customer** |

### Contrainst:

|  |  |  |  |
| --- | --- | --- | --- |
| **Name** | **Add** | **Delete** | **Update** |
| **Customer** | + | - | + |

## Reservation

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Reservation** | | | | | | |
| **Column Name** | **Data Type** | **Allow Null** | **Default Value** | **Key** | **Validation** | **Description** |
| CusID | Int | No |  | Foreign key |  | **Reservation of CusID** |
| ReservationID | int |  |  | Primary key |  | Reservation of **Reservation** |
| NumberPeople | Int | No |  |  |  | NumberPeople of **Reservation** |
| StartDate | Datetime | No | Current Day |  |  | StartDate of **Reservation** |
| EndDate | Datetime | No |  |  |  | EndDate of **Reservation** |
| Status | varchar | No |  |  |  | Status of **Reservation** |
| Paid | Money | No |  |  |  | Deposit of **Reservation** |

### Contrainst:

|  |  |  |  |
| --- | --- | --- | --- |
| **Name** | **Add** | **Delete** | **Update** |
| **Reservation** | + | +  (can not delete if exitst reference in reservation detail) | + |

## **Bill**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Bill** | | | | | | |
| **Column Name** | **Data Type** | **Allow Null** | **Default Value** | **Key** | **Validation** | **Description** |
| BillID | int | No |  | Primary |  | ID of Bill |
| PaymentDate | datetime | No |  |  |  | Payment of Bill |
| CustomerID | int | No |  |  |  | ID of Customer |
| UserName | Varchar(50) | No |  |  |  | Name of Employee |
| ReservationID | Int | No |  |  |  | ID of Occurpancy |

### Contrainst:

|  |  |  |  |
| --- | --- | --- | --- |
| **Name** | **Add** | **Delete** | **Update** |
| Bill | + | + | + |

## **ReservationDetail**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **ReservationDetail** | | | | | | |
| **Column Name** | **Data Type** | **Allow Null** | **Default Value** | **Key** | **Validation** | **Description** |
| RoomID | int | No |  | Primary |  | ID of Occurpancy |
| ReservationID | Int | No |  | Primary |  | Payment of Occurpancy |
| RentDate | Datetime | No |  |  |  | Rent date |
| UserID | int | No |  |  |  | userID of Occurpancy |

### Contrainst:

|  |  |  |  |
| --- | --- | --- | --- |
| **Name** | **Add** | **Delete** | **Update** |
| **ReservationDetail** | + | + | + |

## **Service**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Service** | | | | | | |
| **Column Name** | **Data Type** | **Allow Null** | **Default Value** | **Key** | **Validation** | **Description** |
| ServiceID | int | No |  | Primary |  | ID of Service |
| ReservationID | int | No |  |  |  | ID of reservation |
| ServiceName | Varchar(100) | No |  |  |  | Name of Service |

### Contrainst:

|  |  |  |  |
| --- | --- | --- | --- |
| **Name** | **Add** | **Delete** | **Update** |
| **Service** | + | + | + |

## **News**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **News** | | | | | | |
| **Column Name** | **Data Type** | **Allow Null** | **Default Value** | **Key** | **Validation** | **Description** |
| NewsID | int | No |  | Primary |  | ID of Service |
| Title | Varchar(100) | No |  |  |  | ID of Occurpancy |
| ImageUri | Varchar(100) | No |  |  |  | Name of Service |
| DatePost | DateTime | No |  |  |  | Date post news |
| EmployeeID | Int | No |  |  |  | Employee who post news |

### Contrainst:

|  |  |  |  |
| --- | --- | --- | --- |
| **Name** | **Add** | **Delete** | **Update** |
| **News** | + | + | + |

## **Feedback**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Feedback** | | | | | | |
| **Column Name** | **Data Type** | **Allow Null** | **Default Value** | **Key** | **Validation** | **Description** |
| FeedbackID | int | No |  | Primary |  | ID of Service |
| CustomerID | int | No |  |  |  | ID of Occurpancy |
| EmployeeID | int | No |  |  |  | Name of Service |
| DateReceive | DateTime |  |  |  |  | Date post FeedBack |
| DateFeedback | DateTime |  |  |  |  | Date employee feedback |
| FeedBack | Varchar(200) |  |  |  |  | Customer’s FeedBack |
| Response | Varchar(200) |  |  |  |  | Employee’s feedback |

### Contrainst:

|  |  |  |  |
| --- | --- | --- | --- |
| **Name** | **Add** | **Delete** | **Update** |
| **Feedback** | + | + | + |

# Site map

## Home Side

register.xhtml

login.xhtml

Index.xhtml

myaccount.xhtml

changepassword.xhtml

information.xhtml

feedback.xhtml

My reservation.xhtml

logout.xhtml

contact.xhtml

gallery.xhtml

search.xhtml

random rooms.xhtml

about.xhtml

## Admin Side

changepassword.xhtml

myaccount.xhtml

login.xhtml

Admin Index.xhtml

logout.xhtml

Manage room.xhtml

Add room.xhtml

Update room.xhtml

Delete room.xhtml

Delete room type.xhtml

Add room type.xhtml

Manage room type.xhtml

Manage reservation.xhtml

Add reservation.xhtml

Update reservation.xhtml

Delete reservation.xhtml

Update room type.xhtml

Update emp.xhtml

Add emp.xhtml

Manage employee.xhtml

Manage reservation detail.xhtml

Delete res detail.xhtml

Manage customer.xhtml

Add cus.xhtml

Update cus.xhtml

Manage service.xhtml

Add service.xhtml

Update service.xhtml

Delete service.xhtml

Manage feedback.xhtml

Add feedback.xhtml

Update feedback.xhtml

Delete feedback.xhtml

Manage news.xhtml

Add news.xhtml

Update news.xhtml

Delete news.xhtml

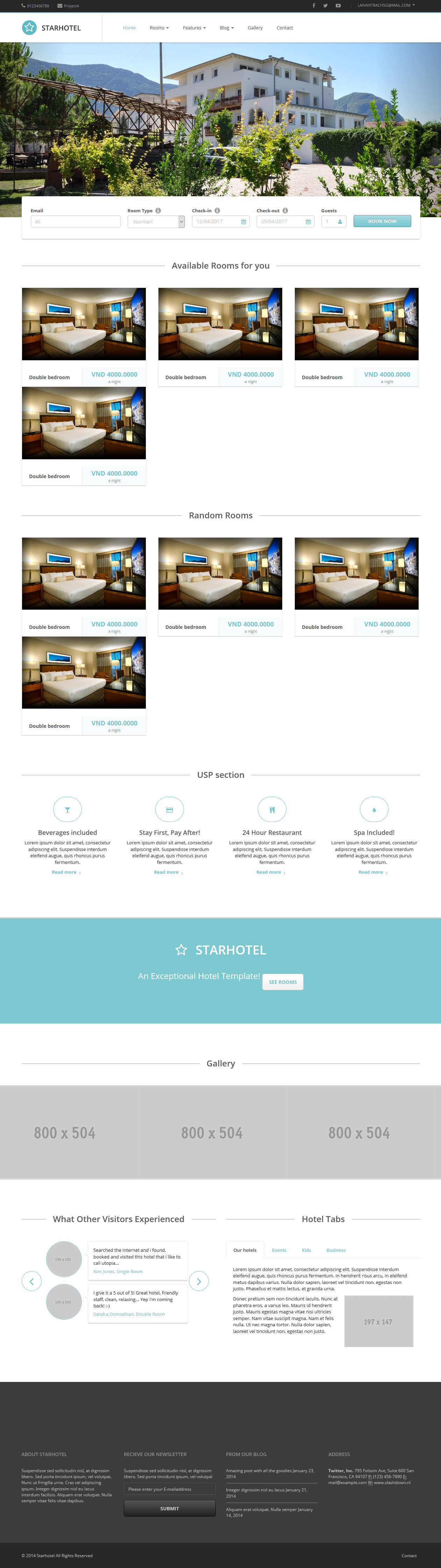
# GUI Design



## Home side

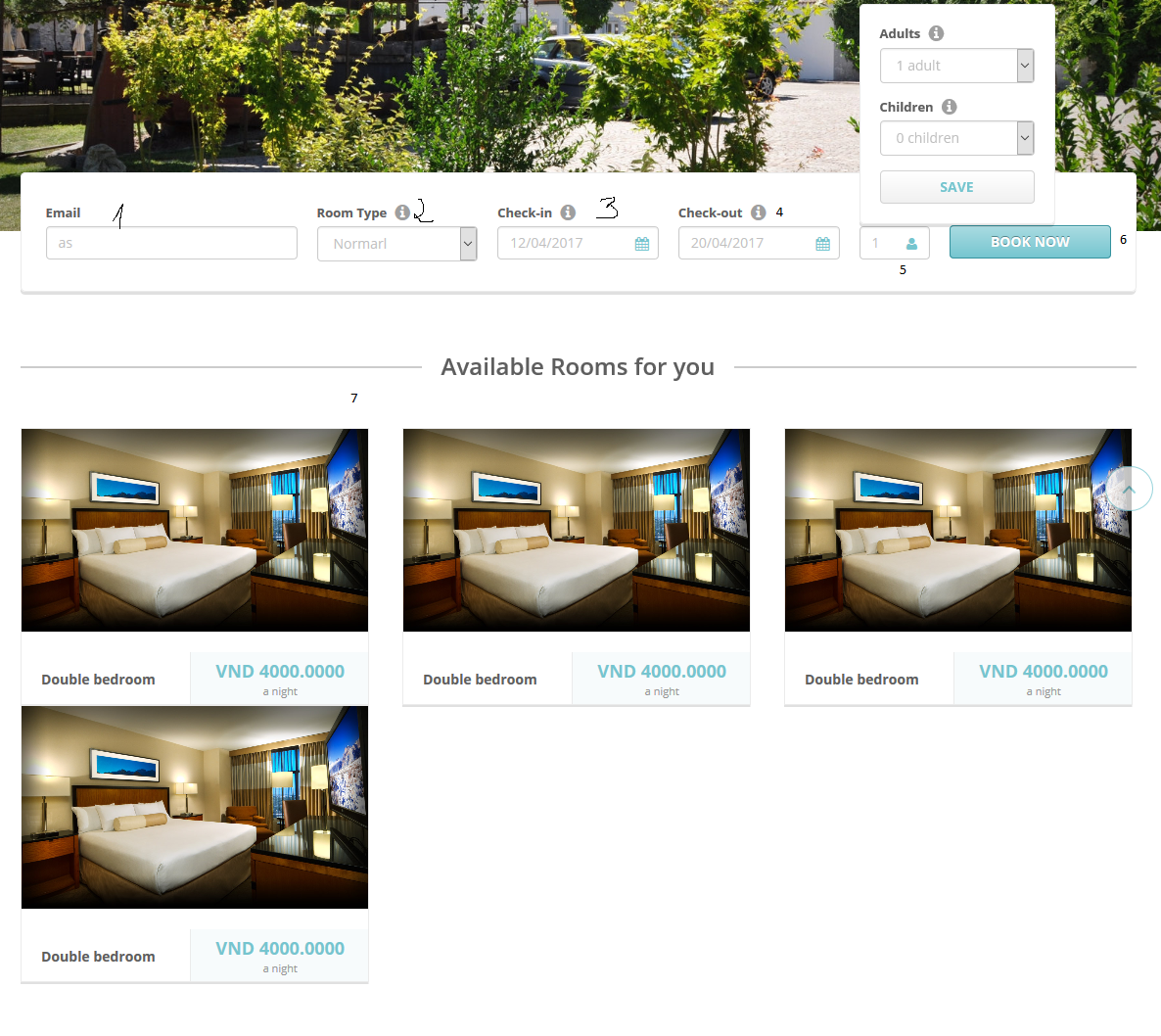
### Home page:

This is Home page of the website, the customers will go to this page first.



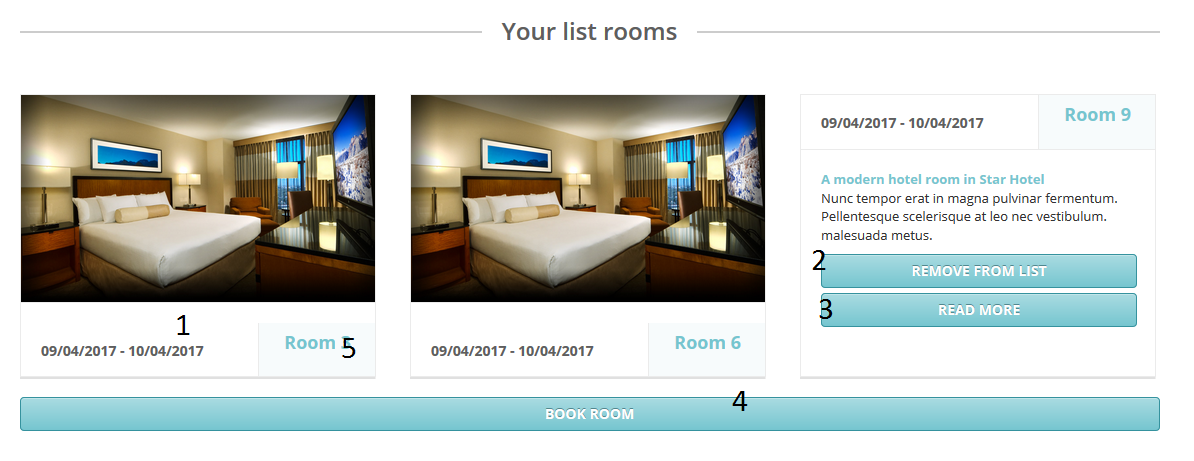
|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **No.** | **Name** | **Type** | **Validation** | **Event** | **Description** | **Status** |
| 1 | btnHome | Button |  | Click | Redirect to Home page |  |
| 2 | btnRoom | Button |  | Click | Redirect to Site List Rooms |  |
| 3 | btnFeature | Button |  | Click | Redirect to Feature page |  |
| 4 | btnBlog | Button |  | Click | Redirect to news page |  |
| 5 | btnGalerry | Button |  | Click | Redirect to Gallery page |  |
| 6 | btnContact | Button |  | Click | Redirect to Contact page |  |
| 7 | Link Login | HyperLink |  | Click | Redirect to login page |  |
| 8 | Link Acount | Hyperlink |  | Click | Redirect to my account page |  |
| 9 | btnSocial | Button |  | Click | Redirect to hotel social page |  |
| 10 | btnBookNow | Button |  | Click | Show available rooms |  |
| 11 | cmbRoomType | comboBox |  |  | Show room types |  |
| 12 | datePickerCheckin | DatePicker |  |  | Date check in |  |
| 13 | datePickerCheckout | DatePicker |  | Click | Date check out | Must be greater than check in |
| 14 | inputboxGuest | ComboBox |  | Click | Number of people |  |
| 15 | Section Available Rooms for you | Section |  |  | Available room for booking |  |
| 16 | Section Random Rooms | Section |  |  | Random rooms |  |

### Search page



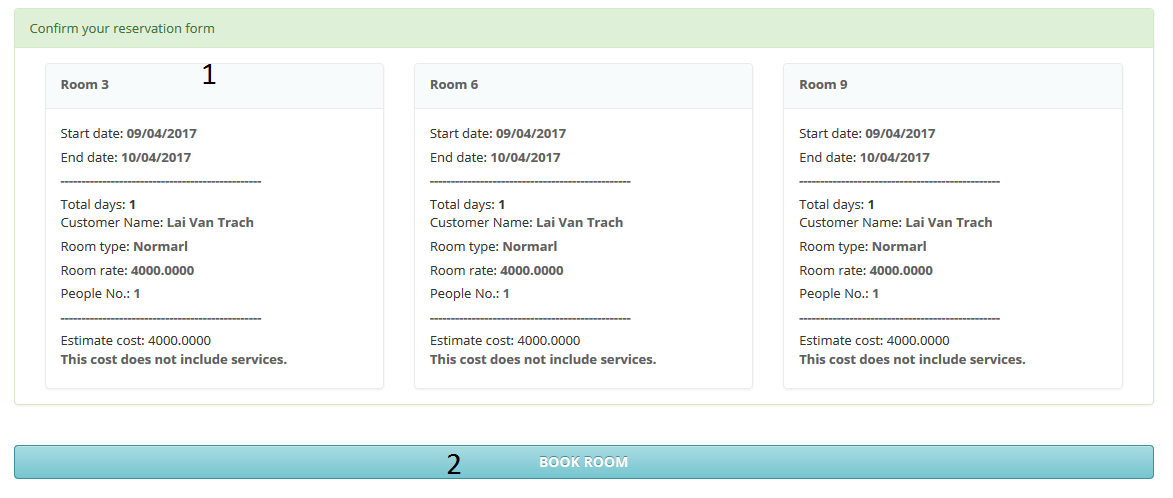
|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **No.** | **Name** | **Type** | **Validation** | **Event** | **Description** | **Status** |
| 1 | txtEmail | Text | Email pattern |  | Email of guest for booking room |  |
| 2 | cmbRoomType | Combobox |  | Click | Room type of hotel |  |
| 3 | dateCheckin | DatePicker | >= current day |  | Checkin date |  |
| 4 | dateCheckout | DatePicker | >=checkin day |  | Checkout date |  |
| 5 | numOfPeople | Combobox | >=0 and <=3 |  | Number of people |  |
| 6 | btnBookNow | Button |  |  | Click for search available room |  |
| 7 | Setion of available room |  |  |  | Avaible rooms show after searching |  |

### Your reservation page



|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **No.** | **Name** | **Type** | **Validation** | **Event** | **Description** | **Status** |
| 1 | labelDate | Label |  |  | Checkin and checkout date |  |
| 2 | btnRemove | Button |  | Click | Remove room out of list rooms |  |
| 3 | btnReadMore | Button |  | Click | Read more detail about room |  |
| 4 | btnBookRoom | Button |  | Click | Goto reservation form |  |
| 5 | labRoom | Label |  |  | Room name |  |

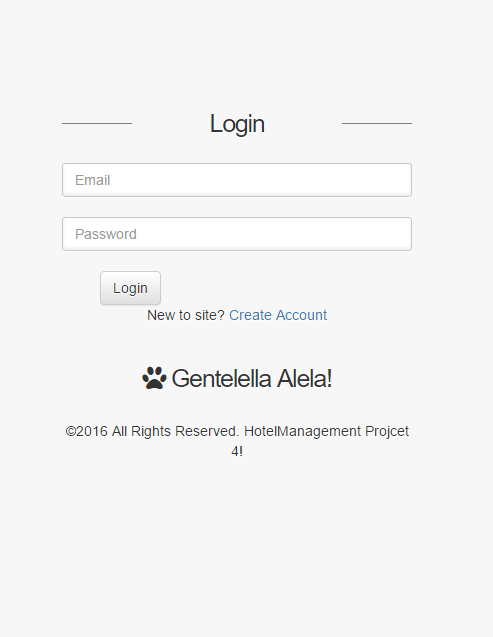
* + 1. Reservation form



|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **No.** | **Name** | **Type** | **Validation** | **Event** | **Description** | **Status** |
| 1 | Panel | Panel |  |  | Detail of reservation for a room include start date, end date, room name, room type, customer, price, estimate cost |  |
| 2 | btnBookRoom | Button |  | Click | Add rooms to databse and redirect to my reservation |  |

* + 1. Login page

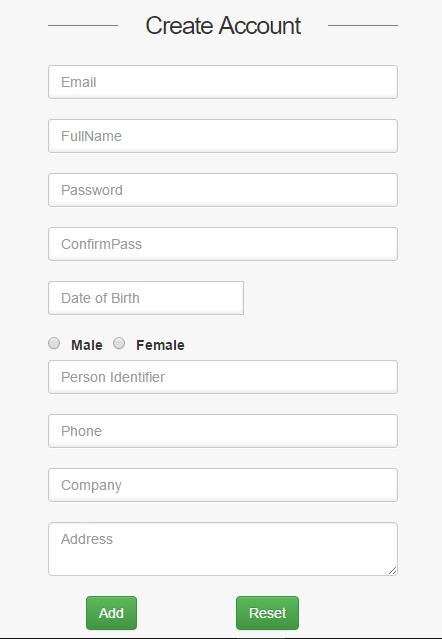
The Customers can login into the system by this page.



|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **No.** | **Name** | **Type** | **Validation** | **Event** | **Description** | **Status** |
| 1 | txtEmail | Text field | Not null |  | Email |  |
| 2 | txtPassword | Text field | Not null |  | Password |  |
| 3 | btnLogin | Button |  | Click | Login to the system and redirect to User’s Home page |  |

* + 1. Register page

The Customers can register an account by this page.

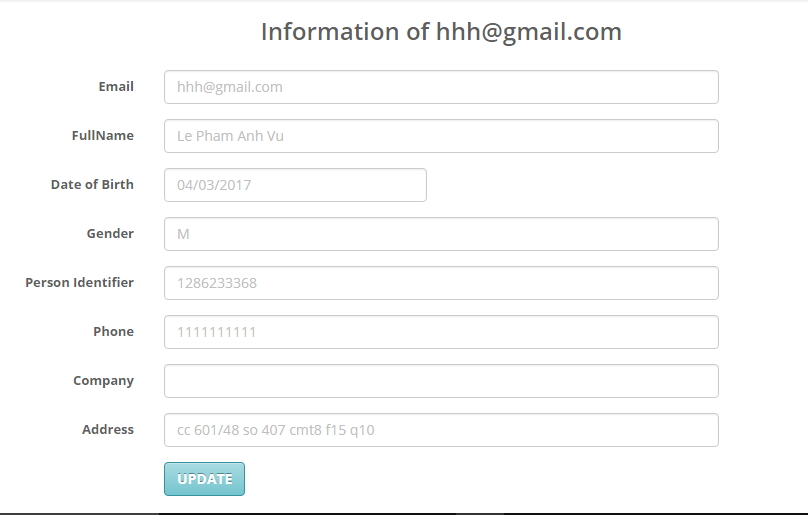


|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **No.** | **Name** | **Type** | **Validation** | **Event** | **Description** | **Status** |
| 1 | txtEmail | Text field | Email format, not null, duplicate |  | Email |  |
| 2 | txtPassword | Text field | Length 6-30 |  | User’s Password |  |
| 3 | txtConfirm Password | Text field | same as txtPassword |  | Confirm Password |  |
| 4 | txtName | Text field | 0<length<=30 |  | User’s Name |  |
| 5 | dbBirthday | Datepicker | Not null  Date<=current year - 16 | Click | User’s Birthday |  |
| 6 | rbMale | Radio button |  | Click | Gender selection |  |
| 7 | rbFemale | Radio button |  | Click | Gender selection |  |
| 8 | ibIdentifier | Text field | 9<length<=10,not null  [0-9] |  | User’s Identifier |  |
| 9 | ibPhone | Text field | 10<length<=11,not null  [0-9] |  | User’s Phone |  |
| 10 | txtCompany | Text field | 0<length<=100 |  | User’s Company |  |
| 11 | txtAddress | Text field | 0<length<=50 |  | User’s Address |  |
| 12 | btnRegister | Button |  | Click | Register user with the system and redirect to home page |  |
| 13 | btnBack | Button |  | Click | Back to Home page |  |

* + 1. My Account

#### Profile page

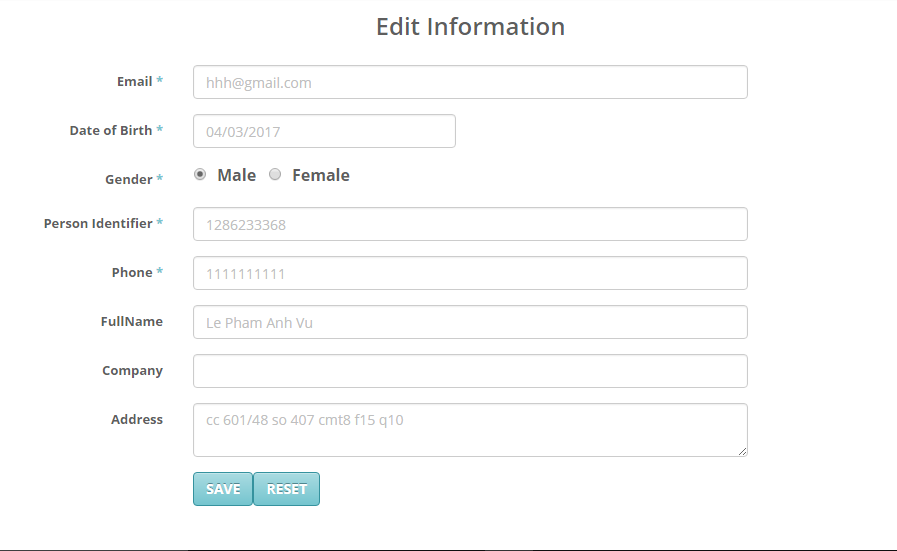
User can update their profile in this page.



|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **No.** | **Name** | **Type** | **Validation** | **Event** | **Description** | **Status** |
| 1 | txtEmail | Text field | Email format, not null, duplicate |  | Email |  |
| 2 | txtPassword | Text field | Length 6-30 |  | User’s Password |  |
| 3 | txtConfirm Password | Text field | same as txtPassword |  | Confirm Password |  |
| 4 | txtName | Text field | 0<length<=30 |  | User’s Name |  |
| 5 | dbBirthday | Datepicker | Not null  Date<=current year - 16 | Click | User’s Birthday |  |
| 6 | rbMale | Radio button |  | Click | Gender selection |  |
| 7 | rbFemale | Radio button |  | Click | Gender selection |  |
| 8 | ibIdentifier | Text field | 9<length<=10,not null  [0-9] |  | User’s Identifier |  |
| 9 | ibPhone | Text field | 10<length<=11,not null  [0-9] |  | User’s Phone |  |
| 10 | txtCompany | Text field | 0<length<=100 |  | User’s Company |  |
| 11 | txtAddress | Text field | 0<length<=50 |  | User’s Address |  |
| 12 | btnRegister | Button |  | Click | Register user with the system and redirect to home page |  |
| 13 | btnBack | Button |  | Click | Back to Home page |  |

#### Update Profile

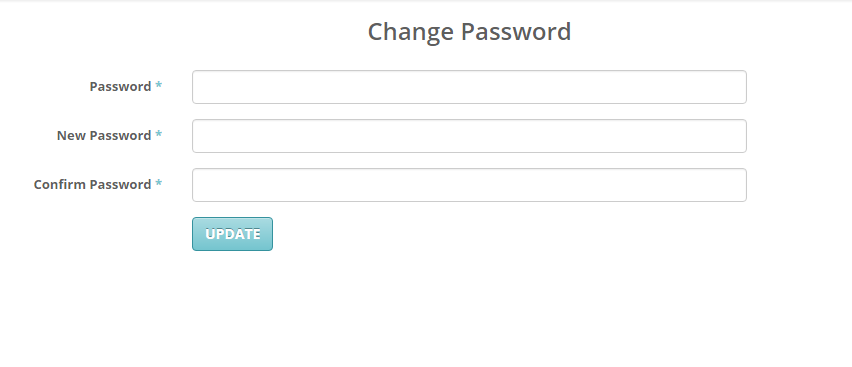
The Buyer can buy a GoldBuyer package to become a GoldBuyer in this page



|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **No.** | **Name** | **Type** | **Validation** | **Event** | **Description** | **Status** |
| 1 | txtEmail | Text field | Email format, not null, duplicate |  | Email |  |
| 2 | txtPassword | Text field | Length 6-30 |  | User’s Password |  |
| 3 | txtConfirm Password | Text field | same as txtPassword |  | Confirm Password |  |
| 4 | txtName | Text field | 0<length<=30 |  | User’s Name |  |
| 5 | dbBirthday | Datepicker | Not null  Date<=current year - 16 | Click | User’s Birthday |  |
| 6 | rbMale | Radio button |  | Click | Gender selection |  |
| 7 | rbFemale | Radio button |  | Click | Gender selection |  |
| 8 | ibIdentifier | Text field | 9<length<=10,not null  [0-9] |  | User’s Identifier |  |
| 9 | ibPhone | Text field | 10<length<=11,not null  [0-9] |  | User’s Phone |  |
| 10 | txtCompany | Text field | 0<length<=100 |  | User’s Company |  |
| 11 | txtAddress | Text field | 0<length<=50 |  | User’s Address |  |
| 12 | btnRegister | Button |  | Click | Register user with the system and redirect to home page |  |
| 13 | btnBack | Button |  | Click | Back to Home page |  |

#### Change Password User

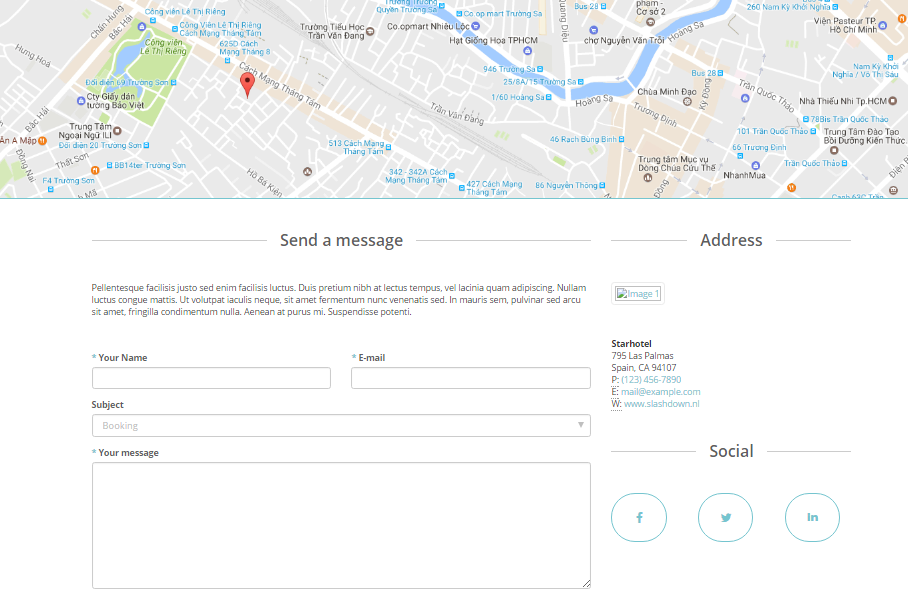
User can change password



|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **No.** | **Name** | **Type** | **Validation** | **Event** | **Description** | **Status** |
| 1 | txtPassword | Text field | Old password from database |  | User’s old password |  |
| 2 | txtNewPassword | Text field | Length 6-30 |  | User’s New Password |  |
| 3 | txtConfirm Password | Text field | same as txtPassword |  | Confirm Password |  |
| 4 | btnRegister | Button |  | Click | Register user with the system and redirect to home page |  |

* + 1. Contact

Customers can contact with us

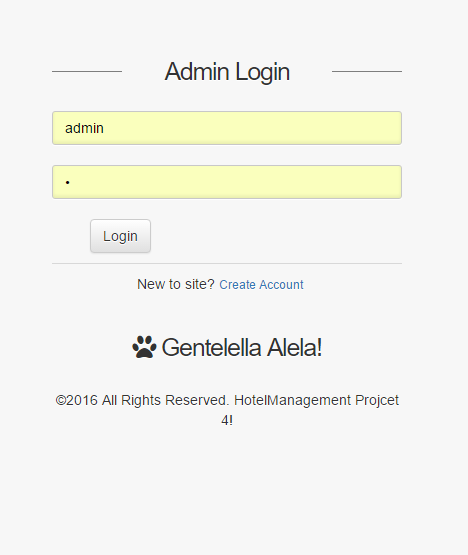


## Admin side

Administrators can use this page to manage the website.

### Login page

Administrator can login into the system in this page.



|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **No.** | **Name** | **Type** | **Validation** | **Event** | **Description** | **Status** |
| 1 | txtEmail | Text field | Not null |  | Email |  |
| 2 | txtPassword | Text field | Not null |  | Password |  |
| 3 | btnLogin | Button |  | Click | Login to the system and redirect to User’s Home page |  |

### Admin Home page

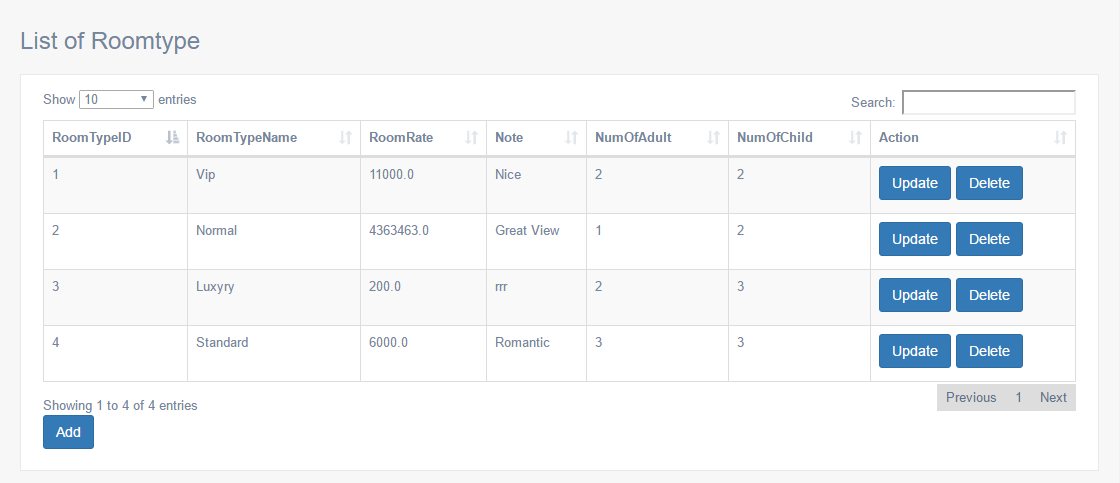
When the Administrators has logged in,

### 

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **No.** | **Name** | **Type** | **Validation** | **Event** | **Description** | **Status** |
| 1 | Room | Dropdown Menu |  |  | List of RoomType,Room |  |
| 2 | Reservation | Dropdown Menu |  |  | List of Reservation,Reservation Detail |  |
| 3 | Employee | Menu |  |  | List of Employee |  |
| 4 | Customer | Menu |  |  | List of Customer | 1 |
| 5 | News | Menu |  |  | List of News | 1 |
| 6 | Service | Menu |  |  | List of Service | 1 |
| 7 | FeedBack | Menu |  |  | List of FeedBack | 1 |
| 8 | btnProfile | Button |  |  | Information of Admin | 1 |
| 9 | btnChangePassWord | Button |  |  | ChangePassWord Admin | 1 |
| 10 | btnLogout | Button |  |  | Logout account | 1 |

### List of RoomType

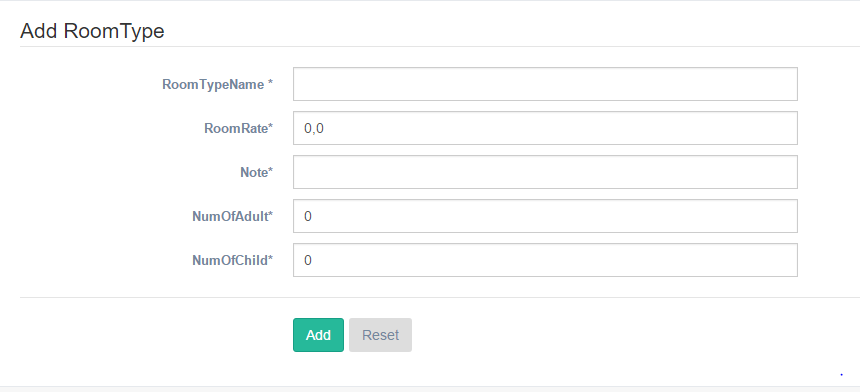
This is list of RoomType.



|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **No.** | **Name** | **Type** | **Validation** | **Event** | **Description** | **Status** |
| 1 | cbbRoomtype | Combobox |  |  | Combobox RoomType Page |  |
| 2 | txtSearch | Txttextfield |  |  | Search Roomtype |  |
| 3 | lblRoomtypeID | Label |  |  | RoomTypeID |  |
| 4 | lblRoomTypeName | Label |  |  | RoomTypeName |  |
| 5 | lblRoomRate | Label |  |  | RoomRate |  |
| 6 | lblNote | Label |  |  | Note |  |
| 7 | LblNumofAdult | Label |  |  | NumofChild |  |
| 8 | lblNumofChild | Label |  |  | NumofChild |  |
| 8 | lblAction | Label |  |  | Action update,delete |  |
| 9 | btnUpdate | Button |  |  | Update Roomtype |  |
| 10 | BtnDelete | Button |  |  | Delete Roomtype |  |
| 11 | btnAdd | Button |  |  | Add RoomType |  |
| 12 | Paging | Button |  |  | Paging Roomtype |  |

### Add Room Type

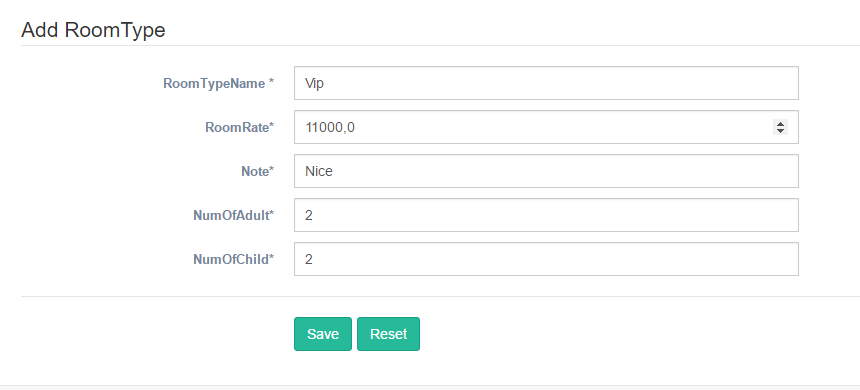
Admin can add roomType



|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **No.** | **Name** | **Type** | **Validation** | **Event** | **Description** | **Status** |
| 1 | txyRoomTypeName | txttextfield | Not null, duplicate |  | RoomTypeName |  |
| 2 | txtRoomRate | txtnumber | Not null |  | RoomRate |  |
| 3 | txtNote | txttextfield | Not null |  | Note |  |
| 4 | txtNumofAdult | txtnumber | Number<3 |  | NumofChild |  |
| 5 | txtNumofChild | txtnumber | Number<3 |  | NumofChild |  |
| 6 | btnAdd | Button |  |  | Add Roomtype |  |
| 7 | btnReset | Button |  |  | Reset Roomtype |  |

### Edit Roomtype

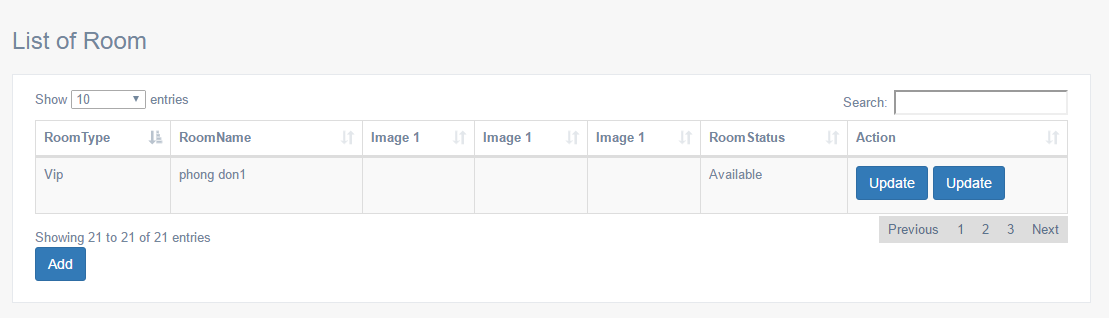
Administrator can edit Roomtype



|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **No.** | **Name** | **Type** | **Validation** | **Event** | **Description** | **Status** |
| 1 | txyRoomTypeName | txttextfield | Not null, duplicate |  | RoomTypeName |  |
| 2 | txtRoomRate | txtnumber | Not null |  | RoomRate |  |
| 3 | txtNote | txttextfield | Not null |  | Note |  |
| 4 | txtNumofAdult | txtnumber | Number<3 |  | NumofChild |  |
| 5 | txtNumofChild | txtnumber | Number<3 |  | NumofChild |  |
| 6 | btnSave | Button |  |  | Edit Roomtype |  |
| 7 | btnReset | Button |  |  | Reset Roomtype |  |

### List of Room

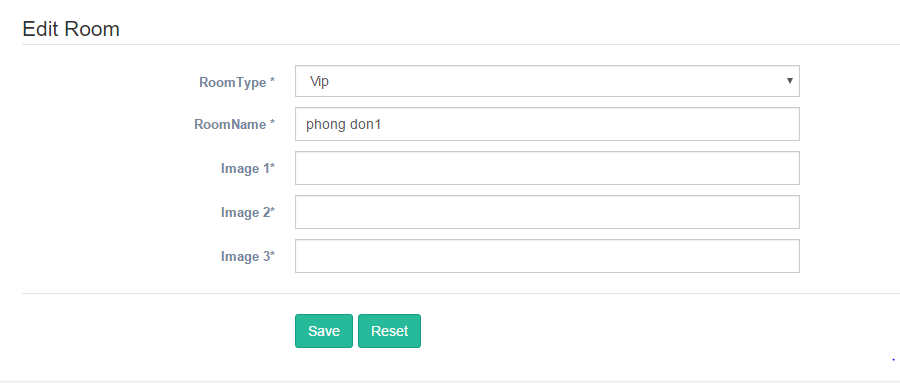
Administrator can see list of room or update ,delete, add



|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **No.** | **Name** | **Type** | **Validation** | **Event** | **Description** | **Status** |
| **No.** | **Name** | **Type** | **Validation** | **Event** | **Description** | **Status** |
| 1 | cbbRoomtype | Combobox |  |  | Combobox Room Page |  |
| 2 | txtSearch | Txttextfield |  |  | Search Room |  |
| 3 | lblRoomtype | Label |  |  | RoomTypeName |  |
| 4 | lblRoomName | Label |  |  | RoomName |  |
| 5 | lblImage1 | Label |  |  | Image 1 |  |
| 6 | lblImage2 | Label |  |  | Image 2 |  |
| 7 | lblImage3 | Label |  |  | Image 3 |  |
| 8 | lblRoomStatus | Label |  |  | Roomstatus |  |
| 8 | lblAction | Label |  |  | Action update,delete |  |
| 9 | btnUpdate | Button |  |  | Update Roomtype |  |
| 10 | BtnDelete | Button |  |  | Delete Room |  |
| 11 | btnAdd | Button |  |  | Add Room |  |
| 12 | Paging | Button |  |  | Paging Room |  |

### Edit Room

Admin can edit Room.



|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **No.** | **Name** | **Type** | **Validation** | **Event** | **Description** | **Status** |
| 1 | cbbRoomtype | Combobox |  |  | Combobox RoomType Page |  |
| 2 | lblRoomName | Label | Not null,duplicate |  | RoomName |  |
| 3 | lblImage1 | Label |  |  | Image 1 |  |
| 4 | lblImage2 | Label |  |  | Image 2 |  |
| 5 | lblImage3 | Label |  |  | Image 3 |  |
| 6 | btnSave | Button |  |  | Edit Room |  |
| 7 | btnReset | Button |  |  | Reset Room |  |

### Add Room

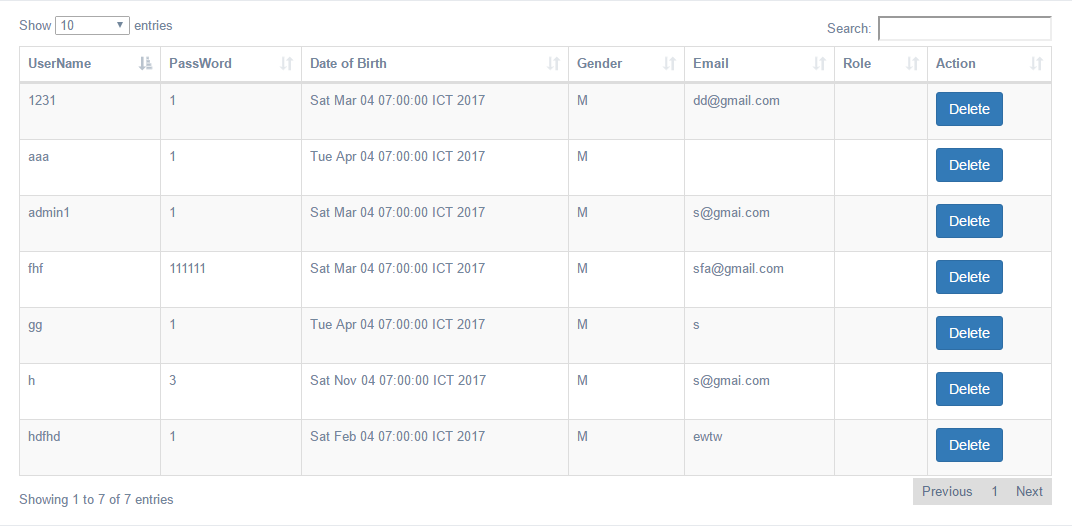
Admin can add Room.



|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **No.** | **Name** | **Type** | **Validation** | **Event** | **Description** | **Status** |
| 1 | cbbRoomtype | Combobox |  |  | Combobox RoomType Page |  |
| 2 | lblRoomName | Label | Not null,duplicate |  | RoomName |  |
| 3 | lblImage1 | Label |  |  | Image 1 |  |
| 4 | lblImage2 | Label |  |  | Image 2 |  |
| 5 | lblImage3 | Label |  |  | Image 3 |  |
| 6 | btnSave | Button |  |  | Add Room |  |
| 7 | btnReset | Button |  |  | Reset Room |  |

### List of employee

Admin can see list of employee



|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **No.** | **Name** | **Type** | **Validation** | **Event** | **Description** | **Status** |
| 1 | cbbRoomtype | Combobox |  |  | Combobox Employee Page |  |
| 2 | txtSearch | Txttextfield |  |  | Search Room |  |
| 3 | lblUserName | Label |  |  | Username employee |  |
| 4 | lblPassword | Label |  |  | Password employee |  |
| 5 | lblIDateOFBirth | Label |  |  | DOB employee |  |
| 6 | lblGender | Label |  |  | Gender employee |  |
| 7 | lblEmail | Label |  |  | Email employee |  |
| 8 | BtnDelete | Button |  |  | Delete Employee |  |

### List of Customer

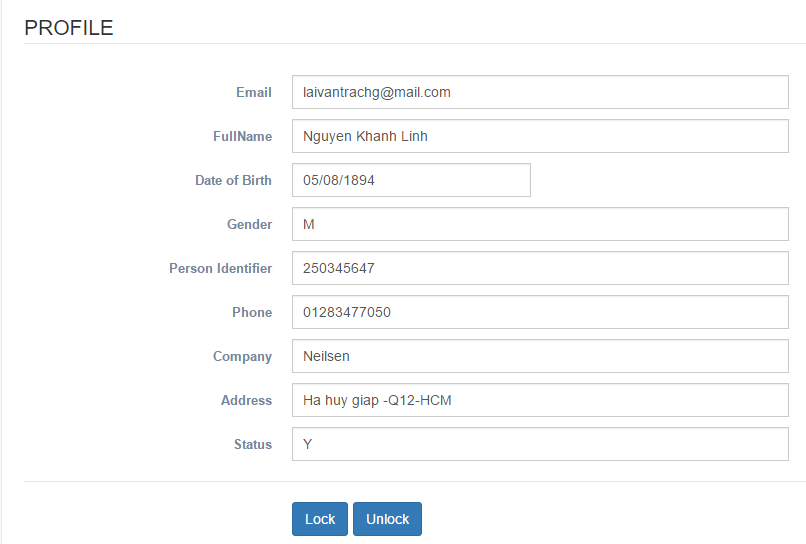
Admin can see list of customer.



|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **No.** | **Name** | **Type** | **Validation** | **Event** | **Description** | **Status** |
| 1 | cbbRoomtype | Combobox |  |  | Combobox Employee Page |  |
| 2 | txtSearch | Txttextfield |  |  | Search Room |  |
| 3 | lblEmail | Label |  |  | Email Customer |  |
| 4 | lblPhone | Label |  |  | Phone Customer |  |
| 5 | lblIPerson | Label |  |  | Person Customer |  |
| 6 | lblDOB | Label |  |  | Dob Customer |  |
| 7 | lblGender | Label |  |  | Gender Customer |  |
| 8 | lblStatus | Label |  |  | Status Customer |  |
| 9 | BtnDelete | Button |  |  | Delete Employee |  |

### Detail Customer

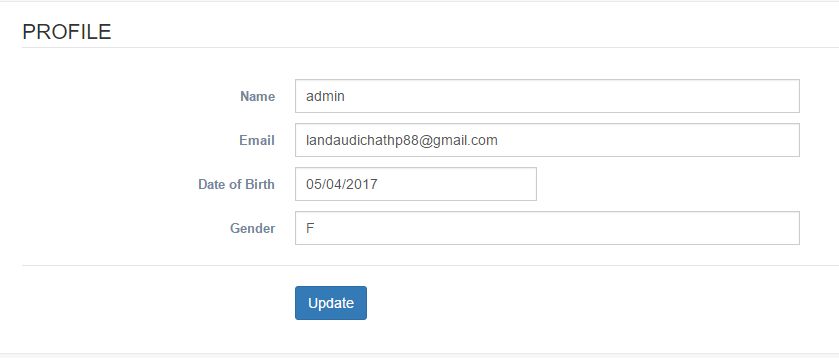
Admin can see detail of customer or lock and unlock customer.



|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **No.** | **Name** | **Type** | **Validation** | **Event** | **Description** | **Status** |
| 1 | txtEmail | Text field |  |  | Email |  |
| 2 | txtName | Text field |  |  | User’s Name |  |
| 3 | dbBirthday | Datepicker |  |  | User’s Birthday |  |
| 4 | rbgender | Radio button |  |  | Gender customer |  |
| 5 | ibIdentifier | Text field |  |  | User’s Identifier |  |
| 6 | ibPhone | Text field |  |  | User’s Phone |  |
| 7 | txtCompany | Text field |  |  | User’s Company |  |
| 8 | txtAddress | Text field |  |  | User’s Address |  |
| 9 | txtStatus | Text field |  |  |  |  |
| 12 | btnLock | Button |  | Click | Lock customer |  |
| 13 | btnUnlock | Button |  | Click | Unlock customer |  |

### Detail of Employee

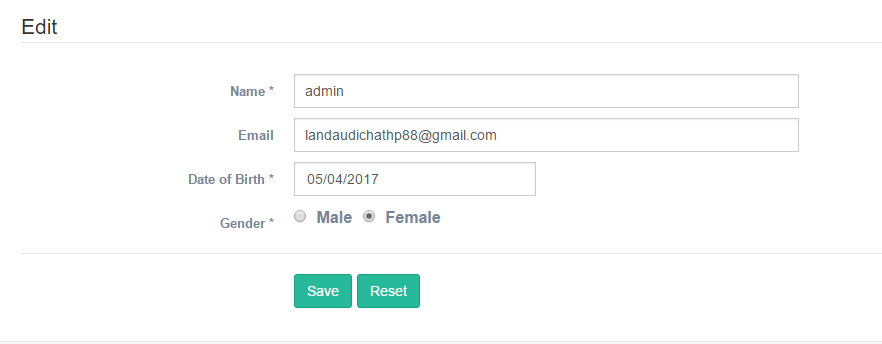
Admin can see detail



|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **No.** | **Name** | **Type** | **Validation** | **Event** | **Description** | | **Status** |
| 1 | txtName | Text field |  |  | Name Employee | |  |
| 2 | txtEmail | Text field |  |  | Employee’s Name | |  |
| 3 | dbBirthday | Datepicker |  |  | | Employee’s Birthday |  |
| 4 | rbgender | Textfield |  |  | Gender customer | |  |
| 5 | btnUpdate | Button |  | Click | Update information | |  |

### Edit Information admin

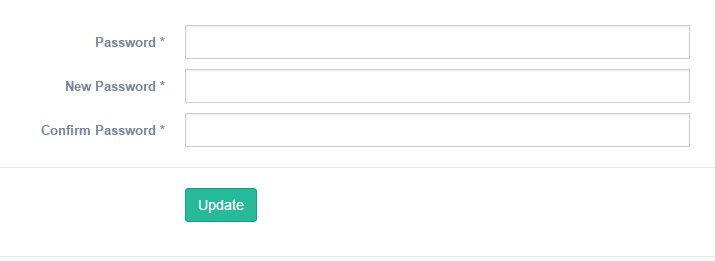
Admin can update information



|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **No.** | **Name** | **Type** | **Validation** | **Event** | **Description** | **Status** |
| 1 | txtName | Text field | Not null, duplicate |  | Name Employee |  |
| 2 | txtEmail | Text field | Email format |  | Employee’s Name |  |
| 3 | dbBirthday | Datepicker | Not null |  | Employee’s Birthday |  |
| 4 | rbgender | Radio  button | Not null |  | Gender customer |  |
| 5 | btnSave | Button |  | Click | Update information |  |
| 1 | btnReset | Button |  | Click | Reset field |  |

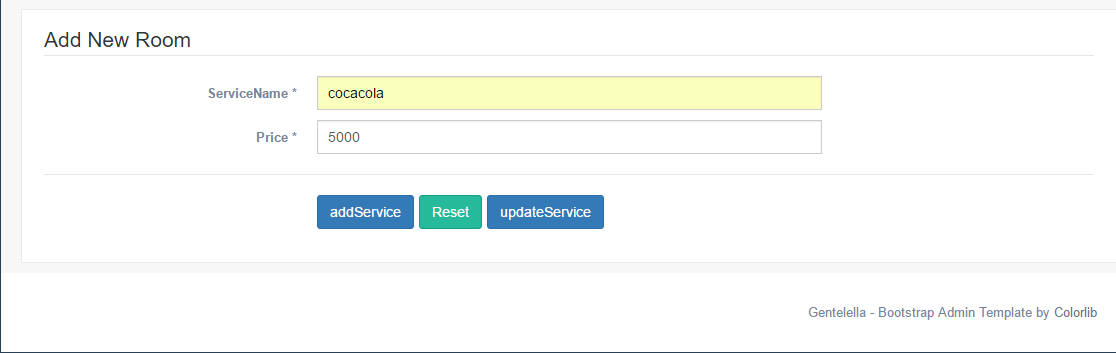
### Change password

Admin can change password



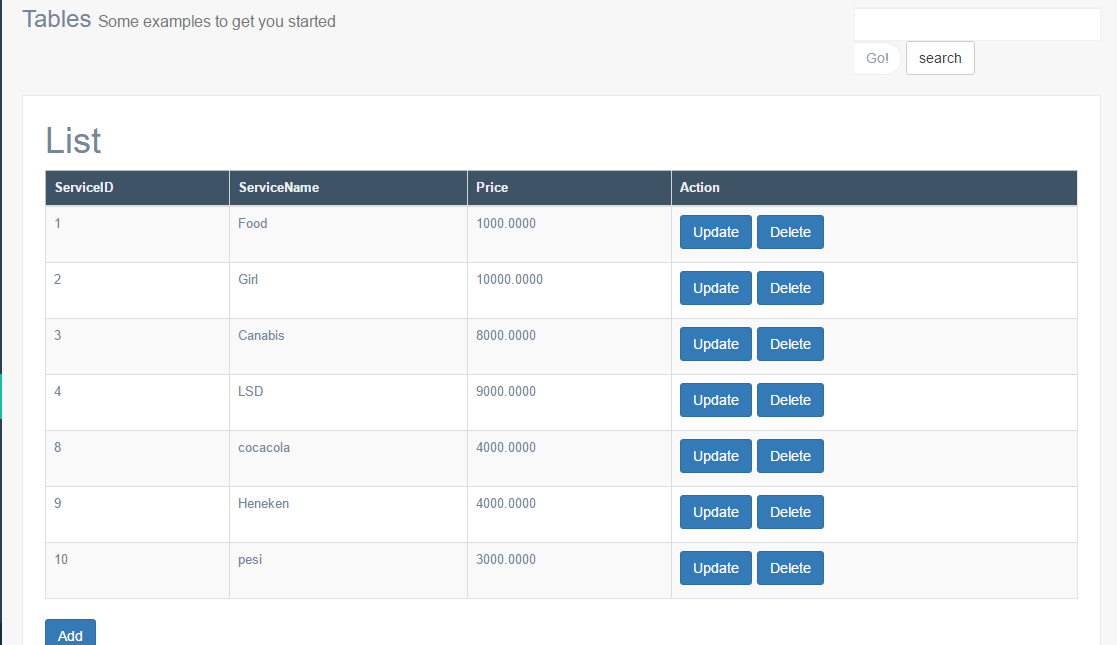
|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **No.** | **Name** | **Type** | **Validation** | **Event** | **Description** | **Status** |
| 1 | txtPassword | Text field | Old password from database |  | employee’s old password |  |
| 2 | txtNewPassword | Text field | Length 6-30 |  | Employee’s New Password |  |
| 3 | txtConfirm Password | Text field | same as txtPassword |  | Confirm Password |  |
| 4 | btnRegister | Button |  | Click | Register user with the system and redirect to home page |  |

### Add Service



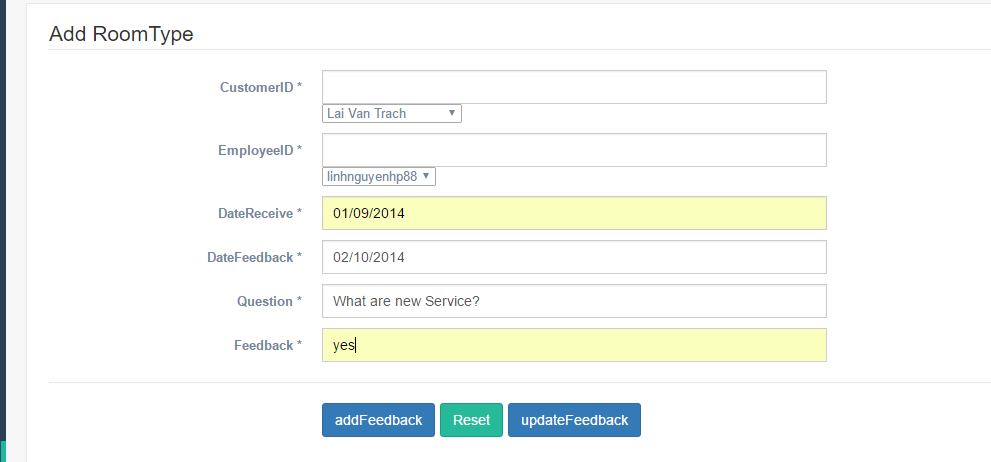
|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Name** | **Type** | **Event** | **Validation** | **Description** | **Status** |
| serviceName | inputText |  | Not Blank |  |  |
| Price | inputText |  | Not Blank |  |  |
| addService | commandButton | Click |  | Insert Command |  |
| Reset | commandButton | Click |  | Reset Command |  |
| UpdateService | commandButton | Click |  | Update Command |  |

### Show Service

****

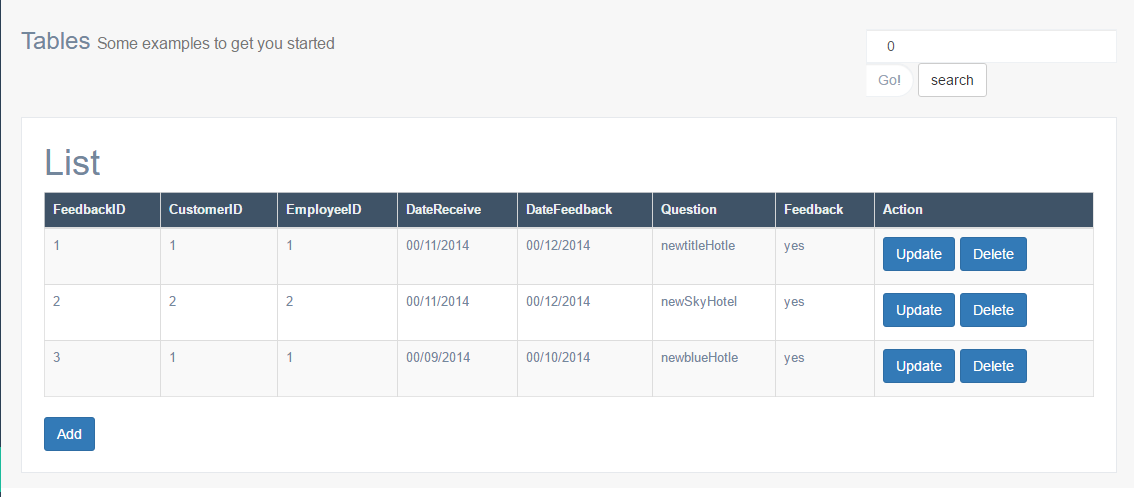
|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Name** | **Type** | **Event** | **Validation** | **Description** | **Status** |
| Add | commandButton | Click |  | Go to Add Page |  |
| Update | commandButton | Click |  | Go to Update Page |  |
| Search | commandButton | Click |  | Select Command |  |
| Delete | commandButton | Click |  | Delete Command |  |
| DataTable |  |  |  | Show list |  |

### Add Feedback



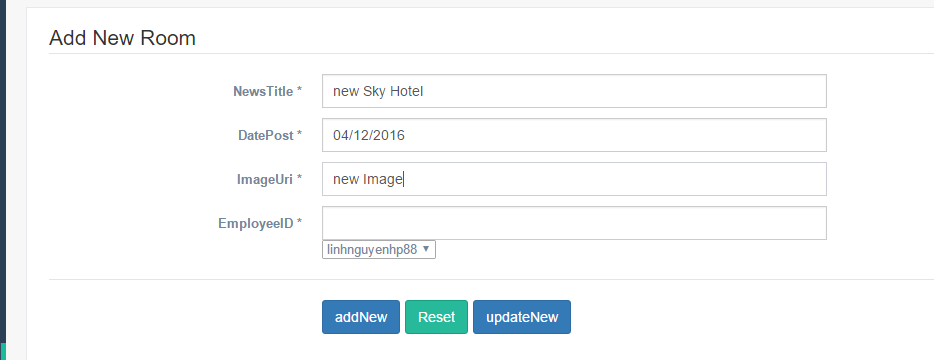
|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Name** | **Type** | **Event** | **Validation** | **Description** | **Status** |
| CustomerID | SelectOneMenu |  |  |  |  |
| EmployeeID | SelectOneMenu |  |  |  |  |
| DateReceive | inputText |  | Not Blank |  |  |
| DateFeedback | inputText |  | Not Blank |  |  |
| Question | nputText |  | Not Blank |  |  |
| Feedback | inputText |  | Not Blank |  |  |
| addFeedback | commandButton | Click |  | Insert Command |  |
| Reset | commandButton | Click |  | Reset Command |  |
| UpdateFeedback | commandButton | Click |  | Update Command |  |

### Show Feedback



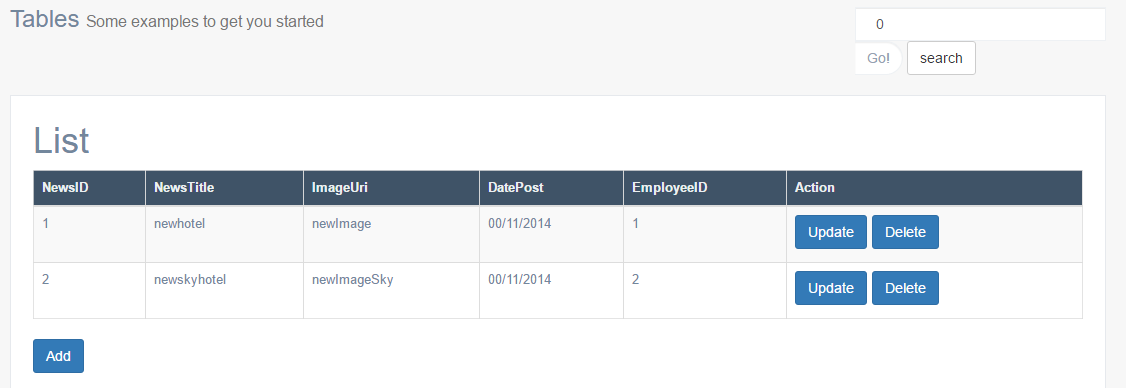
|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Name** | **Type** | **Event** | **Validation** | **Description** | **Status** |
| Add | commandButton | Click |  | Go to Add Page |  |
| Update | commandButton | Click |  | Go to Update Page |  |
| Search | commandButton | Click |  | Select Command |  |
| Delete | commandButton | Click |  | Delete Command |  |
| DataTable |  |  |  | Show list |  |

### Add News



|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Name** | **Type** | **Event** | **Validation** | **Description** | **Status** |
| NewsTtile | inputText |  | Not Blank |  |  |
| DatePost | nputText |  | Not Blank |  |  |
| ImageUri | inputText |  |  |  |  |
| EmployeeID | SelectOneMenu |  |  |  |  |
| addNews | commandButton | Click |  | Insert Command |  |
| Reset | commandButton | Click |  | Reset Command |  |
| UpdateNew | commandButton | Click |  | Update Command |  |

### Show News



|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Name** | **Type** | **Event** | **Validation** | **Description** | **Status** |
| Add | commandButton | Click |  | Go to Add Page |  |
| Update | commandButton | Click |  | Go to Update Page |  |
| Search | commandButton | Click |  | Select Command |  |
| Delete | commandButton | Click |  | Delete Command |  |
| DataTable |  |  |  | Show list |  |