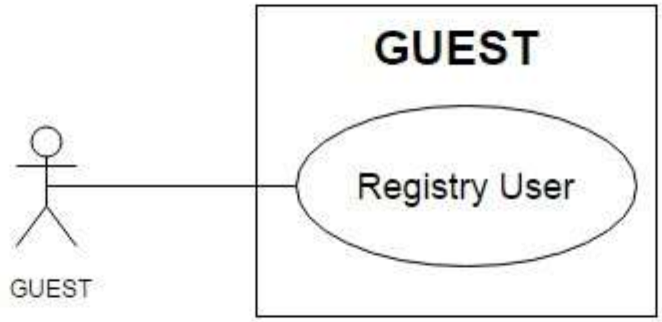
Table of Contents

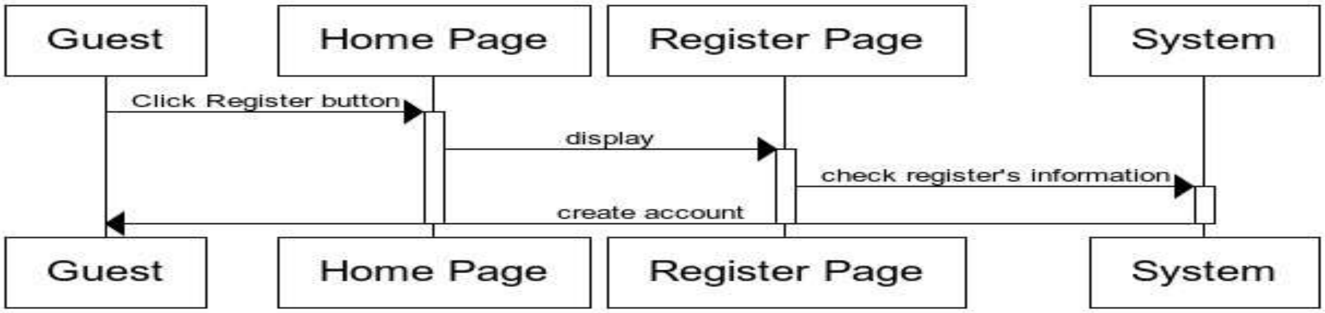
Case and Sequence Diagram Use

Guest:

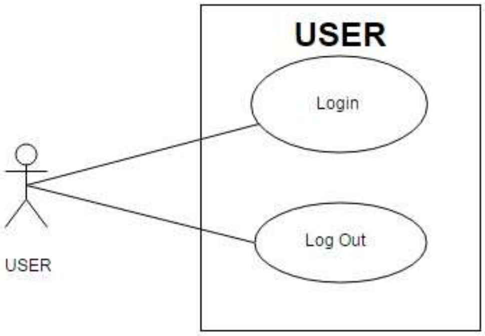


Use Case Name	Register user	
Actors	Guest	
Description	Guest wants to register with the website to become a customer.	
Requirements	Guest provides Guest’s information	
Pre-conditions		
Post-conditions	Success: Guest creates a account in this website Fail: Refill information	
Basic flow	Actor Actions: 1. Actor clicks ‘Register’ link in Home page	System Responses: 2. System redirects to Register page and displays Registration form with the following control : - ‘UserName’ text field - ‘Password’ text field - ‘Confirm Password’ text field - ‘Name’ text field - ‘Address’ text field - ‘Phone’ text field - ‘Email’ text field

Exception	Actor Actions: [Exception] blank. 1. Actor lefts ‘UserName’ text field ‘Email is invalid’ 2. UserName 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 ‘Address’ text field blank. 8. Actor lefts ‘Phone’ text field blank. 9. Input phone not in numeric or phone length not in range 10 – 11 character. 10. Actor lefts ‘Email’ text field blank or 11. input an invalid email address.	System Responses: System displays message to Actor : - ‘UserName cannot be blank’. - ‘This Username has already exist!’. - ‘Password cannot be blank’’. - ‘Confirm Password can’t be blank’. - ‘Confirm password not match password.’ - ‘Name cannot be blank’. - ‘Address cannot be blank’. - ‘Phone cannot be blank’. • - ‘The phone number should be numeric only and must be between 10 and 11 digits’.

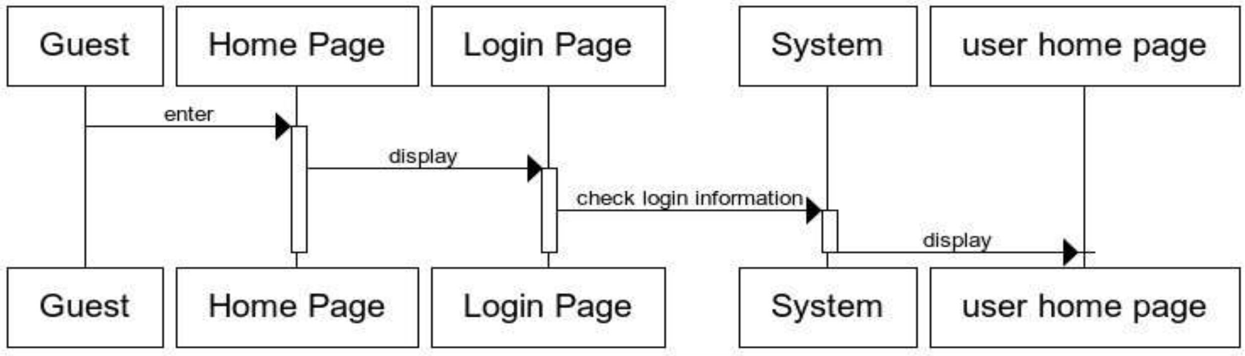


Login/Log Out:



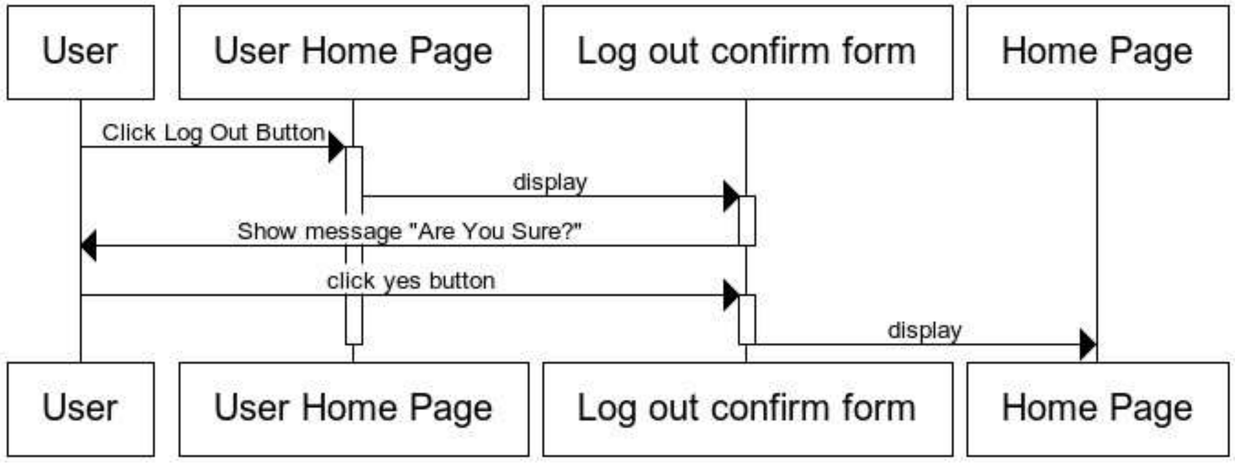
Login:

Use Case Name	Login	
Actors	Customer, Admin or Staff.	
Description	Customer, Admin or Staff wants to login to website.	
Requirements	Actor provides username and password	
Post-conditions	Success: User logged in to website. Fail: Refill information	
Basic flow	- Actor Actions: - Actor go to Home page of website.	- 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 - ‘Login’ button - ‘Cancel’ button - 4. System checks the information
Exception	- Actor Actions: - [Exception] - 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. - - ‘Cancel’ button. - System show a message box: “Invalid - Username and password”.

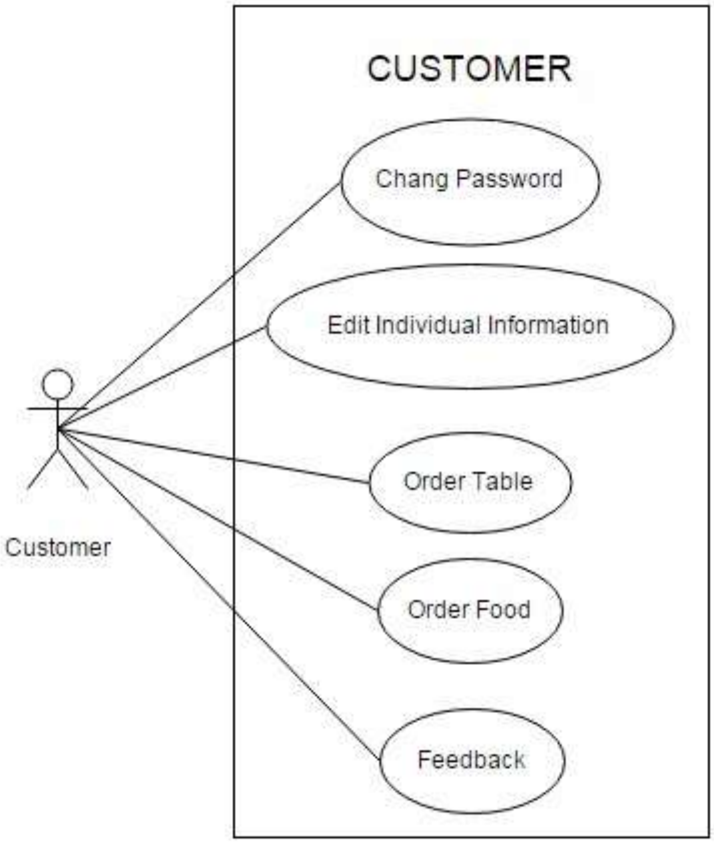


Log out:

Use Case Name	Log out	
Actors	Customer, Admin or Staff.	
Description	Customer, Admin or Staff wants to log out of the System.	
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



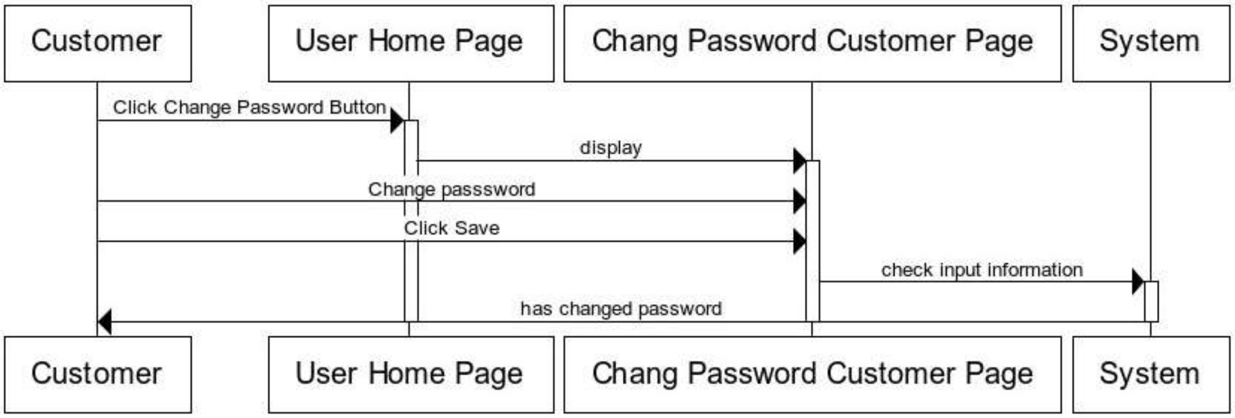
Customer:



Change Password:

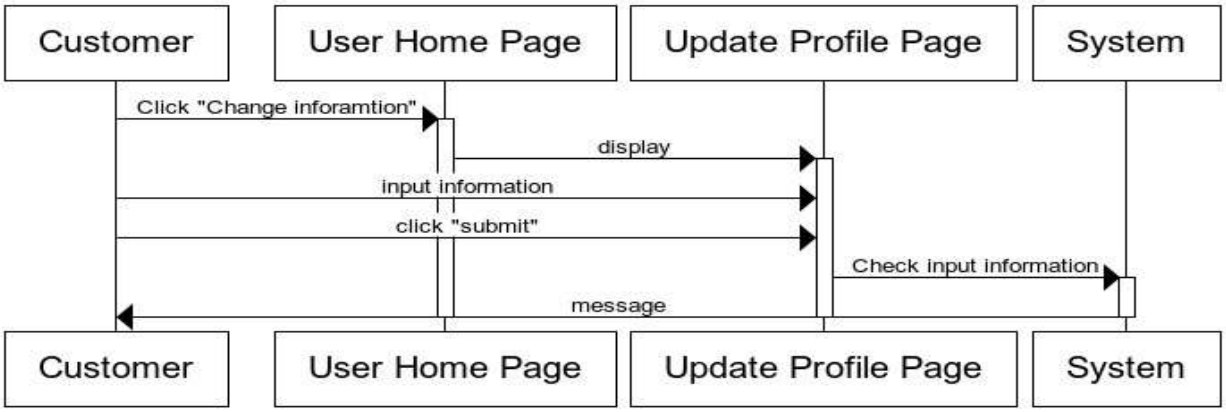
Use case name	Change Password
Actors	Customer
Description	Change Password
Requirements	Actor provides Username and Password
Pre-conditions	Actor has logged in as Customer
Post-conditions	Success: The password has changed.

Basic flow	- Actor actions: - From User Home page, actor clicks - on to the ‘Change Password’ button. - Actor clicks on to ‘Save’ button to - finish adding.	- 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] - 5. System adds new Administrator into the database..
Exception	- [Exception] - - 1. Actor lefts ‘Old Password’ textfield. - 2. Actor lefts ‘New Password’textfield. - 3. Actor lefts ‘Confirm NewPassword’ 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.”

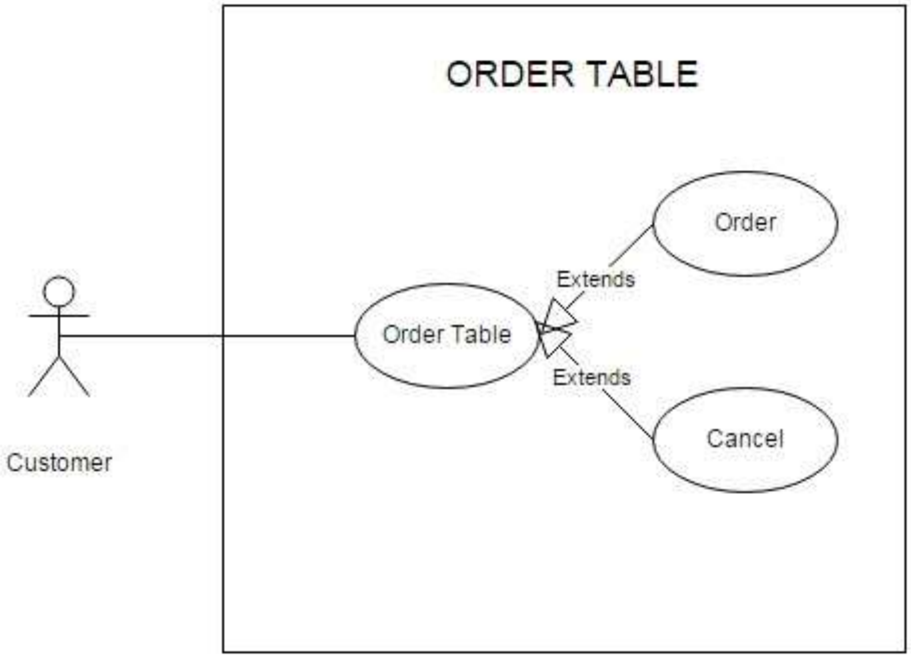


Use Case Name	Edit Individual Information	
Actors	Customer	
Description	Customer wants to Edit Individual Information.	
Requirements	Customer provides Customer's information	
Pre-conditions	Actor has logged in as Customer	
Post-conditions	Success: Change Individual Information success. Fail: Refill information.	
Basic flow	• Actor Actions: • Actor clicks 'Update Profile' link inUser Home page • 3. Actor inputs individual informationand click 'Submit' button.	1. System Responses: 2. 2. System redirects to update Profile page and displays Update profile form with the following control : 3. - 'UserName' text field, read only. 4. - 'Address' text field 5. - 'Phone' text field 6. - 'Email' text field 7. - 'Submit' button 8. - 'Cancel' button 9. 4. System checks the information 10. [Exception] 11. 5. System edit individual information into database, show message: 'Change Individual Information Success!' and then redirects to User Home page.

Exception	1. Actor Actions: 2. [Exception] 3. Actor lefts ‘Address’ text field blank. 4. Actor lefts ‘Phone’ text field blank. 5. Input phone not in numeric or phone length not in range 10 – 11 character. 4. Actor lefts ‘Email’ text field blank or input an invalid email address.	System Responses: System displays message to Actor : - ‘Address cannot be blank’. - ‘Phone cannot be blank’. - ‘The phone number should be numeric only and must be between 10 and 11 digits’. - ‘Email is invalid’.

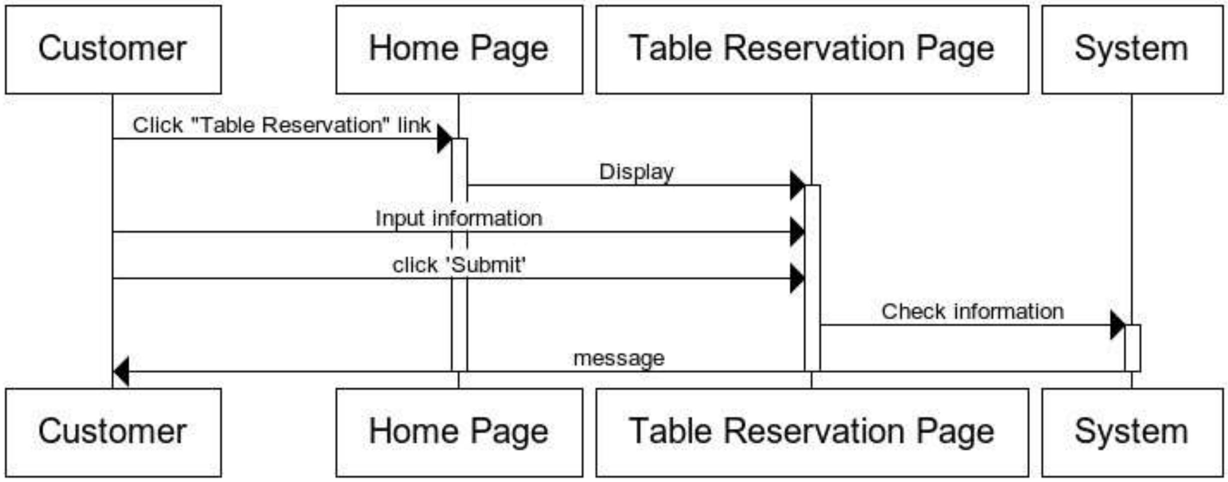


Order Table:



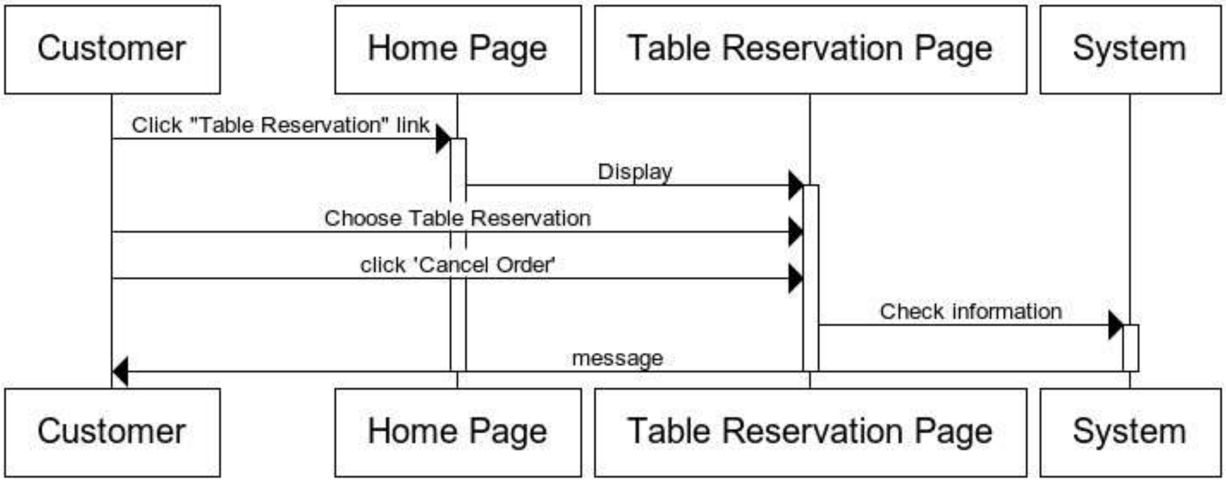
Orderr table:

Use Case Name	Order Table	
Actors	Customer	
Description	Customer wants to Order Table.	
Requirements	Customer provides ordering table's information	
Pre-conditions	Actor has logged in as Customer	
Post-conditions	Success: order table success Fail: Refill information.	
Basic flow	Actor Actions: 1. Actor clicks 'table reservation' link in Home page 3. Actor inputs order information and click 'Submit' button.	System Responses: 2. System redirects to table reservation page and displays table reservation form with the following control : - 'branch' drop down list field - 'table' drop down list field -'banquet date' text field -'banquet day' drop down list field - 'Submit' button - 'Cancel' button 4. System checks the information [Exception] 5. System ordering table information into database, show message: 'order reservation Success!' and then redirects to Home page.
Exception	Actor Actions: [Exception] 1. Input banquet date and banquet day near (order date time).	System Responses: System displays message to Actor : - 'With dinner banquet, you have order before 11am of banquet date! With lunch banquet, you have order before 16pm the day before'. - 'Over table!'.



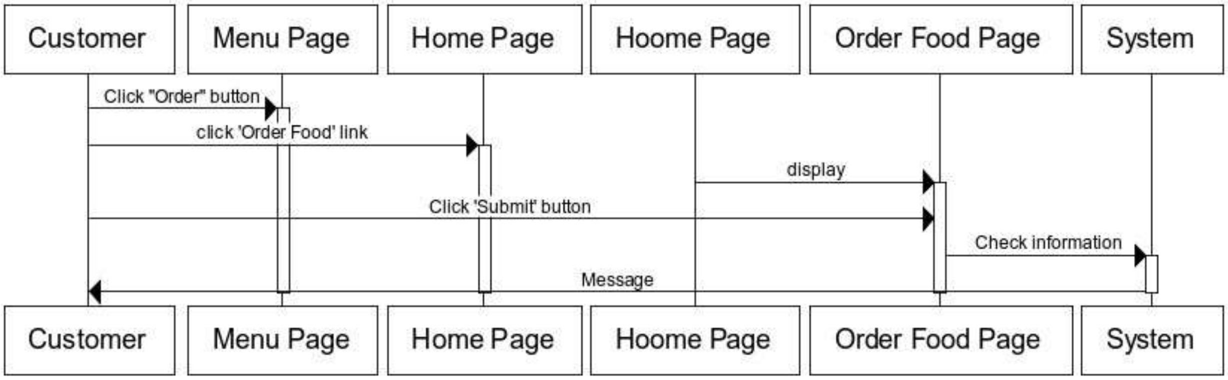
Cancel Ordering Table:

Use Case Name	Cancel Ordering Table	
Actors	Customer	
Description	Customer wants to Cancel Ordering Table.	
Requirements		
Pre-conditions	Actor has logged in as Customer	
Post-conditions	Success: Cancel ordering table success	
	Fail: Cancel ordering table faild.	
Basic flow	Actor Actions: 1. Actor clicks 'table reservation' link in Home page 3. Actor choose ordering tableand click 'Cancel Order' button.	System Responses: 2. System redirects to table reservation page and displays table reservation information form with the following control : - 'List Ordering table of customer. - 'Cancel Order' button 4. System checks the information [Exception] 5. System edit individual information into database, show message: 'order reservation Success!' and then redirects to Home page.
Exception	Actor Actions: [Exception] 1. Actor choose Ordering table to cancel with conditional not valid.	System Responses: System displays message to Actor : - 'With diner banquet, you have cancel before 11am of banquet date! With lunch banquet, you have cancel before 16pm the day before'. - 'you have confirmed this ordering table! PLZ… Contact with our staff!'.



Order Food:

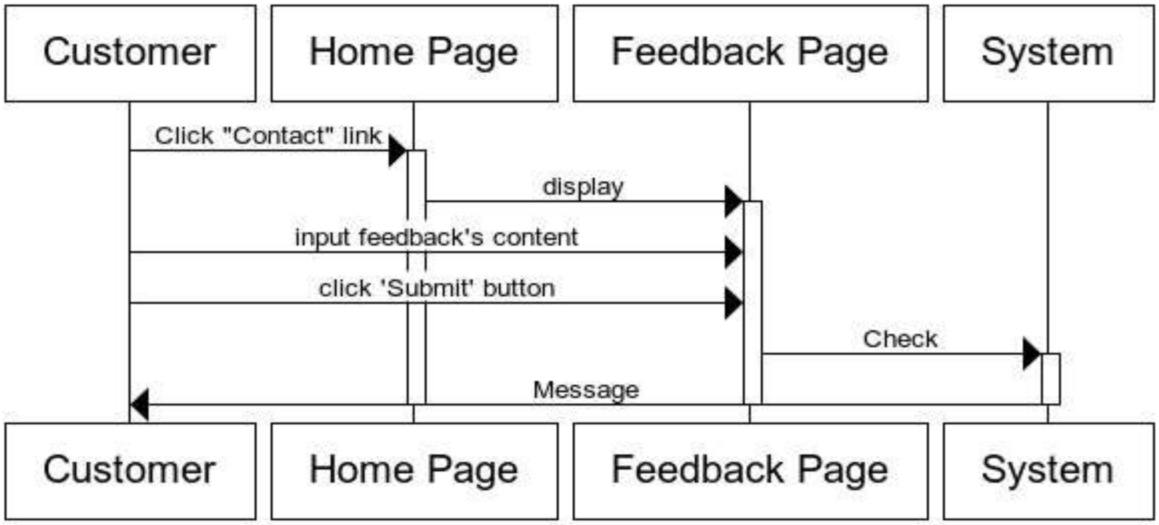
Use Case	Order Food	
Actors	Customer	
Description	Customer wants to Order Food.	
Requirements	Customer provides ordering food's information	
Pre-conditions	Actor has logged in as Customer	
Post-conditions	Success: order food success Fail: Refill information.	
Basic flow	Actor Actions: 1. Actor clicks 'Order' button in Menu page. 2. Actor click 'Order Food' link in Home Page 4. Actor inputs order information and click 'Submit' button.	System Responses: 3. System redirects to Order Food page and displays Order Food form with the following control : - List Of Food, which customer order. - 'Submit' button - 'Cancel' button 5. System checks the information [Exception] 6. System ordering food information into database,



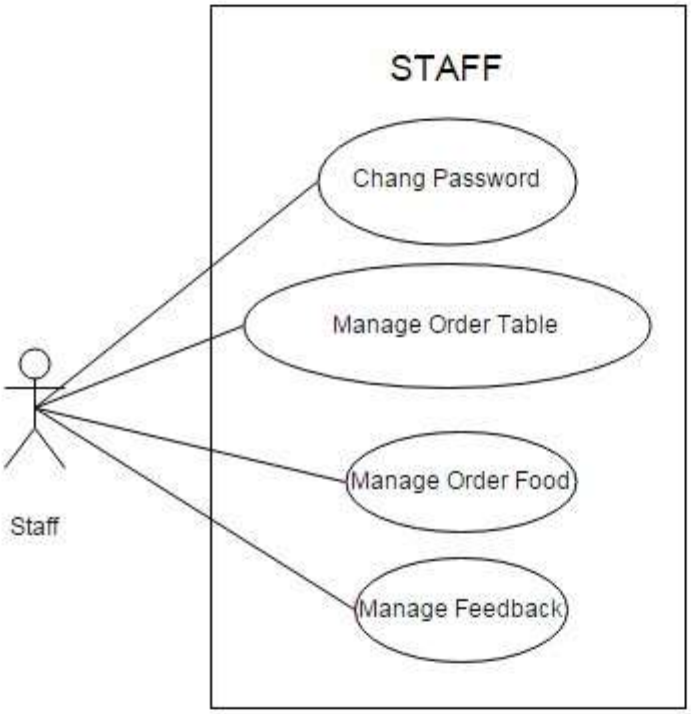
Feedback

Use Case Name	Feedback	
Actors	Customer	
Description	Customer wants to Feedback.	
Requirements	Customer provides deedback's content.	
Pre-conditions	Actor has logged in as Customer	
Post-conditions	Success: feedback success. Fail: Refill information.	
Basic flow	Actor Actions: 1. Actor clicks 'contact' link in Home page 3. Actor inputs feedback's content and click 'Submit' button.	- System Responses: - 2. System redirects to feedback page and displays feedback form with the following control : - -'content' text field - 'Submit' button - 'Cancel' button - 4. System checks the information - [Exception] - 5. System ordering table information into database, show message: 'Send feedback Success!' and then redirects to Home page.

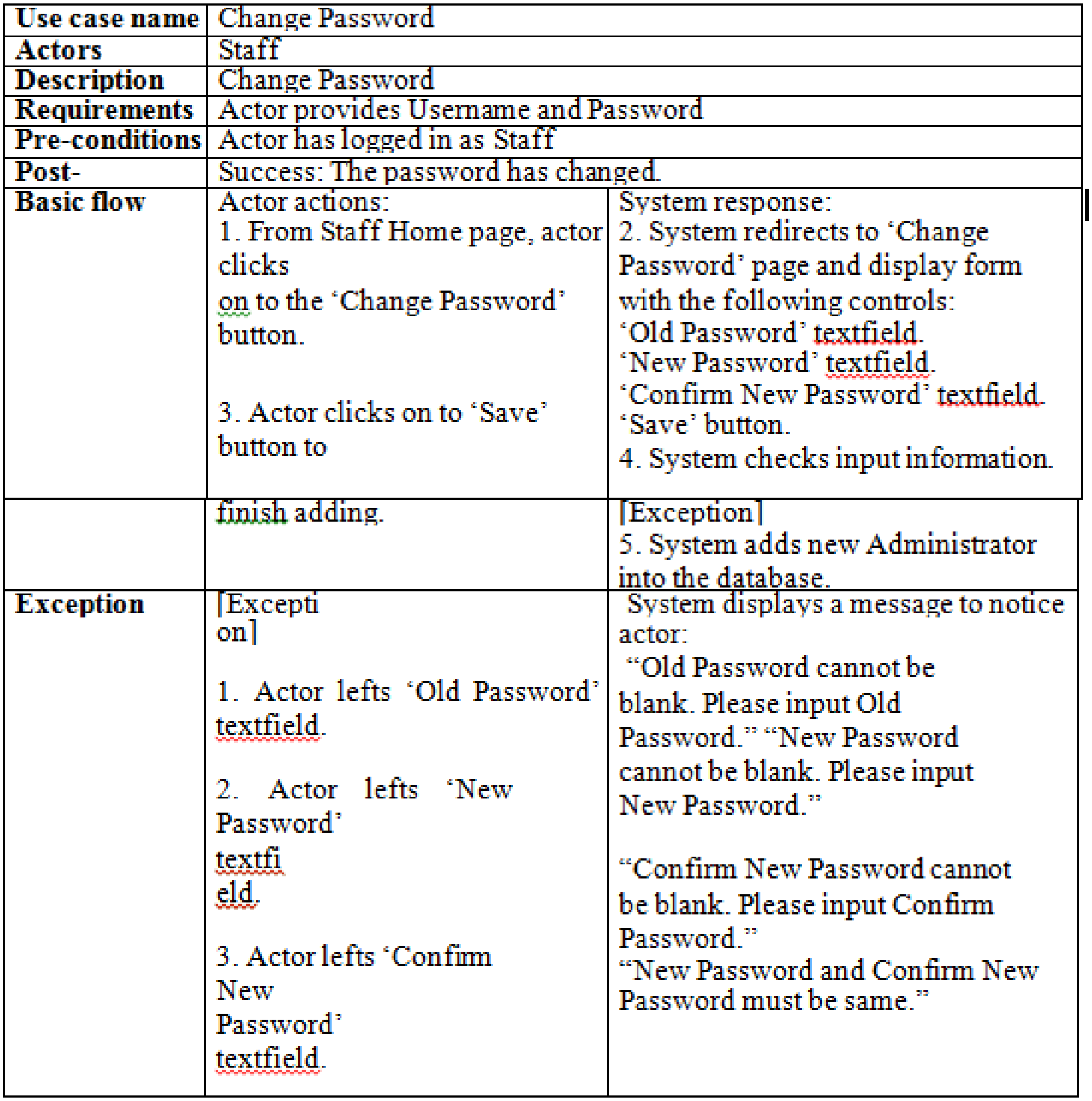
Exception	Actor Actions: [Exception] 1. Actor lefts ‘content’ text field blank.	System Responses: System displays message to Actor : - ‘content cann’t be blank! ’



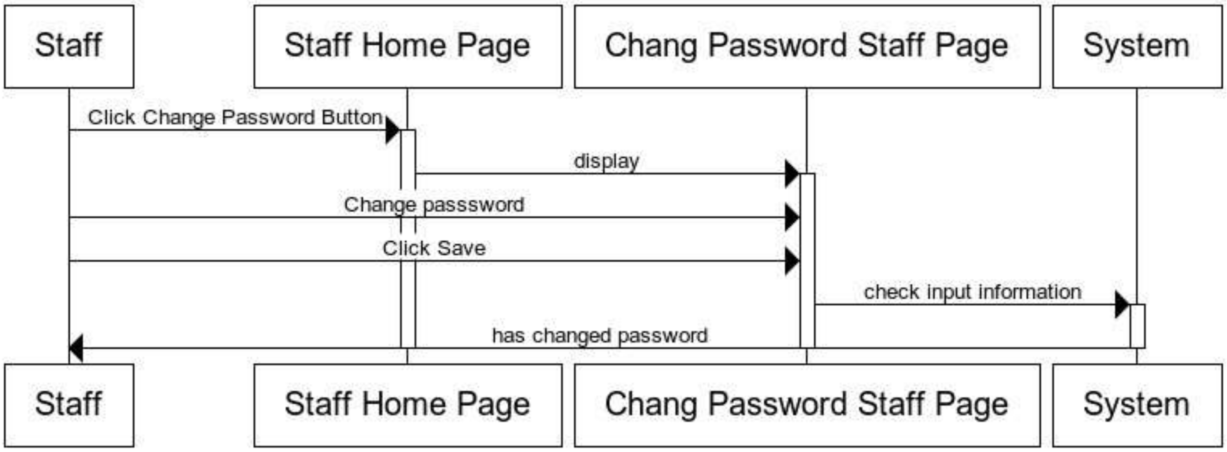
Staff:



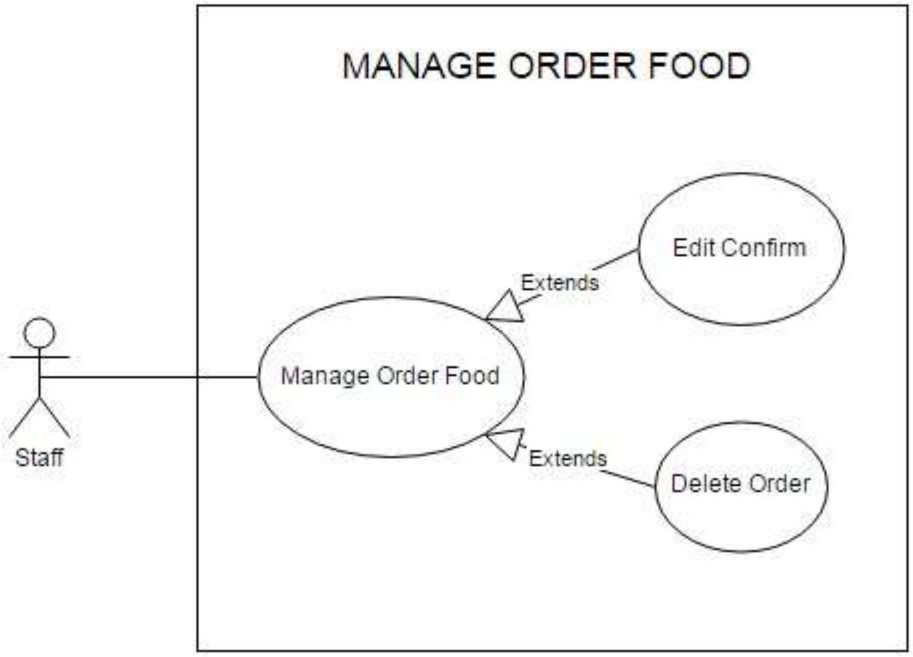
Change Password:



Use case name	Change Password	
Actors	Staff	
Description	Change Password	
Requirements	Actor provides Username and Password	
Pre-conditions	Actor has logged in as Staff	
Post-	Success: The password has changed.	
Basic flow	Actor actions: 1. From Staff Home page, actor clicks on to the 'Change Password' button. 3. Actor clicks on to 'Save' button to	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.
	finish adding.	[Exception] 5. System adds new Administrator into the database.
Exception	[Exception] 1. Actor lefts 'Old Password' textfield. 2. Actor lefts 'New Password' textfield. 3. Actor lefts '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."

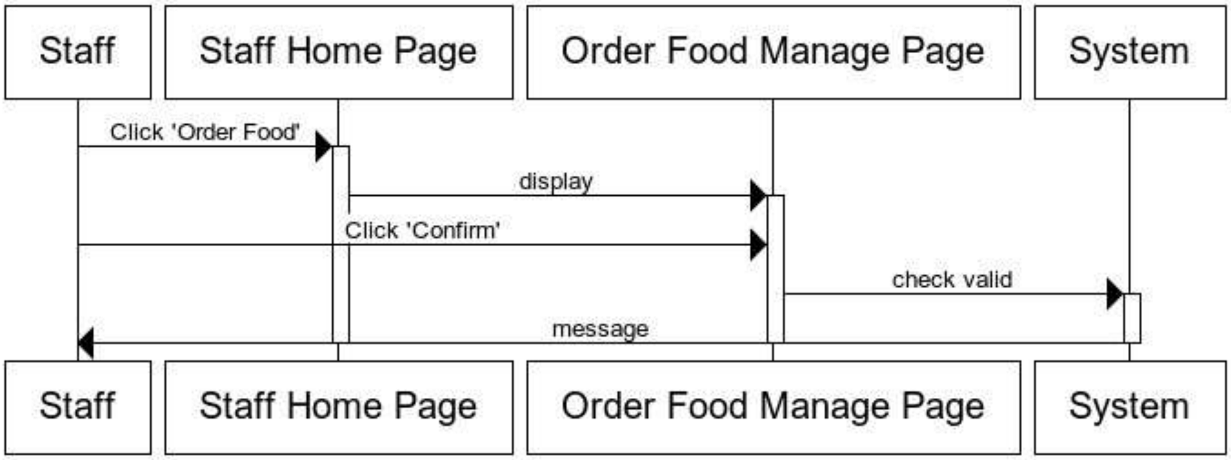


Manage Order Food:



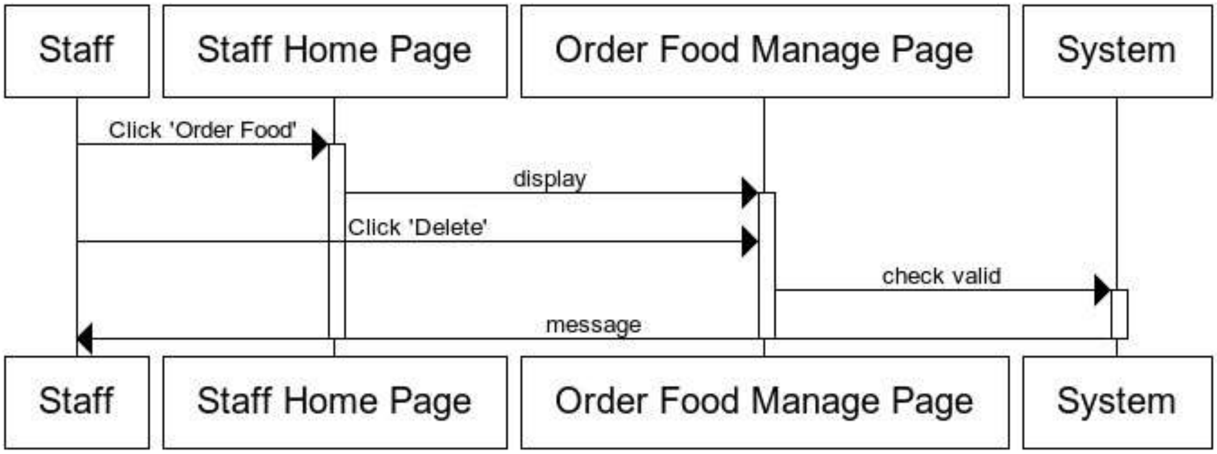
Edit Confirm:

Use Case Name	Edit Confirm	
Actors	Staff	
Description	Staff wants to Edit Confirm.	
Requirements	Staff choose Ordering food.	
Pre-conditions	Actor has logged in as Staff	
Post-conditions	Success: has confirm. Fail: confirm fail.	
Basic flow	Actor Actions: 1. Actor clicks 'Order Food' link in Staff Home page 3. Actor click 'Confirm' button.	System Responses: 2. System redirects to Order Food Management page and displays list of ordering food and following control : - 'Confirm' button 4. System checks the information [Exception] 5. System confirm order into database, show message: 'Order has just confirm!' and then redirects to Staff Home page.
Exception	Actor Actions: [Exception] 1. Actor choose ordering food, which have banquet date before current date	System Responses: System displays message to Actor : - 'You cannot confirm this ordering food!'.

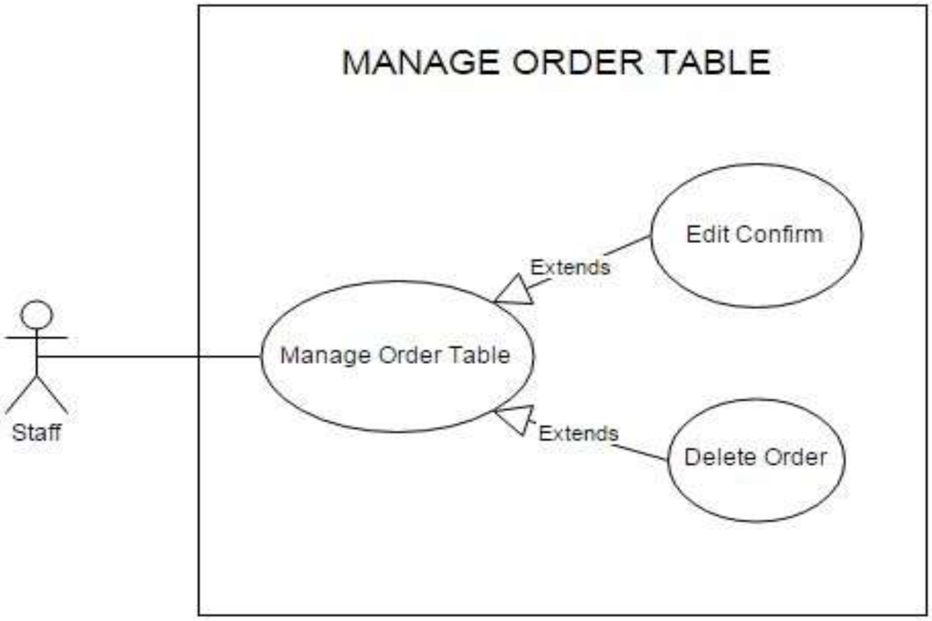


Delete Ordering Food:

Use Case	Delete Ordering Food	
Actors	Staff	
Description	Staff wants to Delete Ordering Food	
Requirement	Staff choose Ordering food.	
Pre-conditions	Actor has logged in as Staff	
Post-conditions	Success: delete success. Fail: delete fail.	
Basic flow	Actor Actions: 1. Actor clicks ‘Order Food’ link in Staff Home page 3. Actor click ‘Delete’ button.	System Responses: 2. System redirects to Order Food Manage page and displays list of ordering food and following control : - ‘Delete’ button 4. System checks the information [Exception] 5. System confirm order into database, show message: ‘Order has just confirm!’ and then redirects to Staff Home page.
Exception	Actor Actions: [Exception] 1. Actor choose ordering food, which have banquet date before current date. 2.Actor choose ordering food, which has confirm.	System Responses: System displays message to Actor : - ‘You cannot delete this ordering food!’.

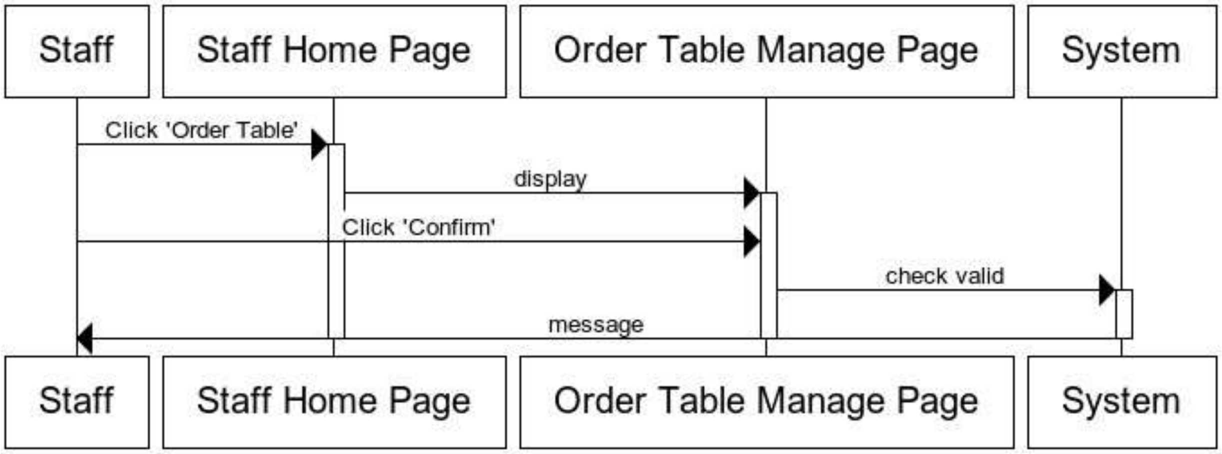


Manage Order Table:



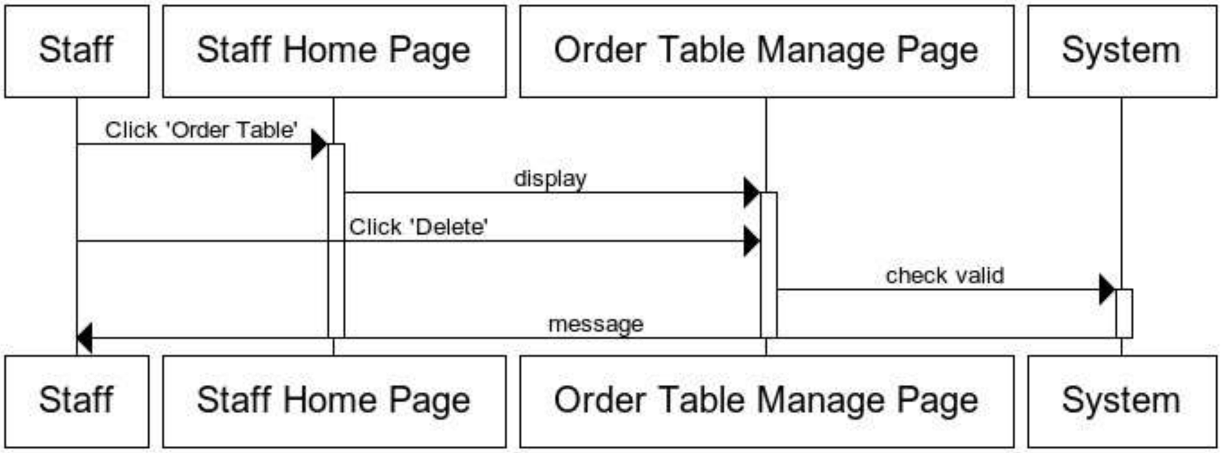
Edit Confirm:

Use Case Name	Edit Confirm	
Actors	Staff	
Description	Staff wants to Edit Confirm.	
Requirements	Staff choose Ordering Table.	
Pre-conditions	Actor has logged in as Staff	
Post-conditions	Success: has confirm. Fail: confirm fail	
Basic flow	- Actor Actions: - Actor clicks ‘Order Table’ link in Staff - Home page - 3. Actor click ‘Confirm’ button.	- System Responses: - 2. System redirects to Order Table Manage page and displays list of ordering Table and following control : - ‘Confirm’ button - 4. System checks the information - [Exception] - 5. System confirm order into database, show message: ‘Order has just confirm!’ and then redirects to Staff Home page.
Exception	Actor Actions: [Exception] 1. Actor choose ordering table, which have banquet date before current date.	System Responses: System displays message to Actor : - ‘You cannot confirm this ordering food!’.

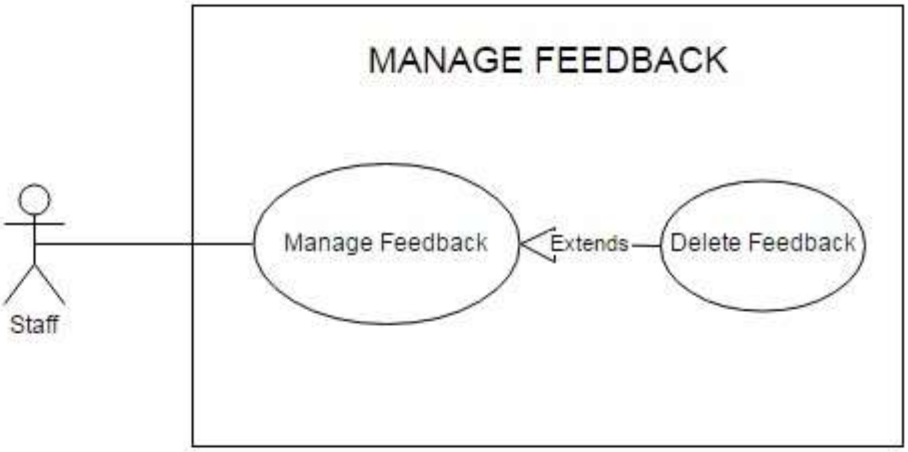


Delete Ordering Table:

Use Case Name	Delete Ordering Table	
Actors	Staff	
Description	Staff wants to Delete Ordering Table	
Requirements	Staff choose Ordering table.	
Pre-conditions	Actor has logged in as Staff	
Post-conditions	Success: delete success. Fail: delete fail.	
Basic flow	Actor Actions: 1. Actor clicks ‘Order Table’ link in Staff Home page 3. Actor click ‘Delete’ button.	System Responses: 2. System redirects to Order Table Manage page and displays list of ordering table and following control: - ‘Delete’ button 4. System checks the information [Exception] 5. System confirm order into database, show message: ‘Order has just confirm!’ and then redirects to Staff Home page.
Exception	Actor Actions: [Exception] 1. Actor choose ordering table, which have banquet date before current date. 2.Actor choose ordering table, which has confirm.	System Responses: System displays message to Actor : - ‘You cannot delete this ordering table!’.



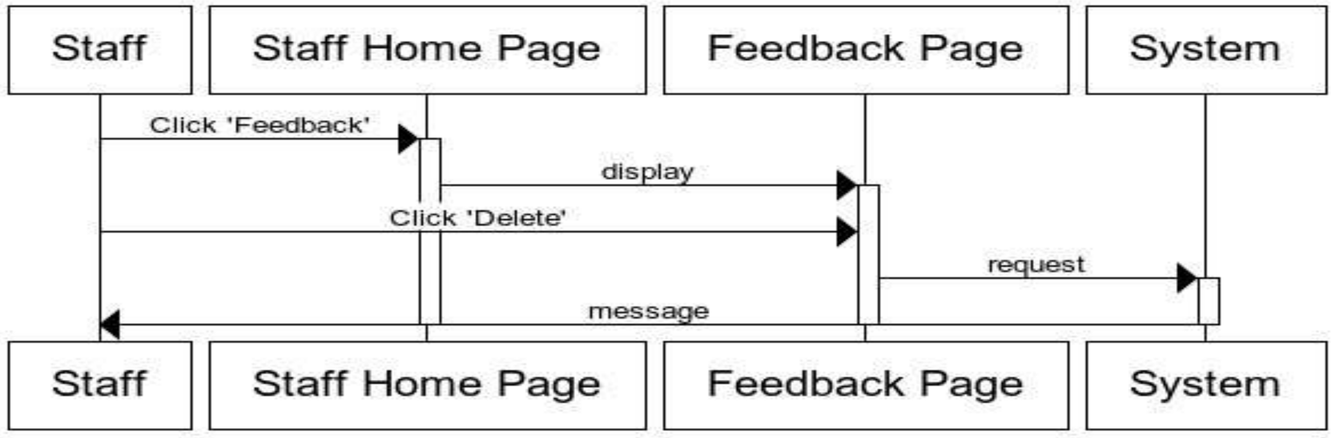
Manage Feedback:

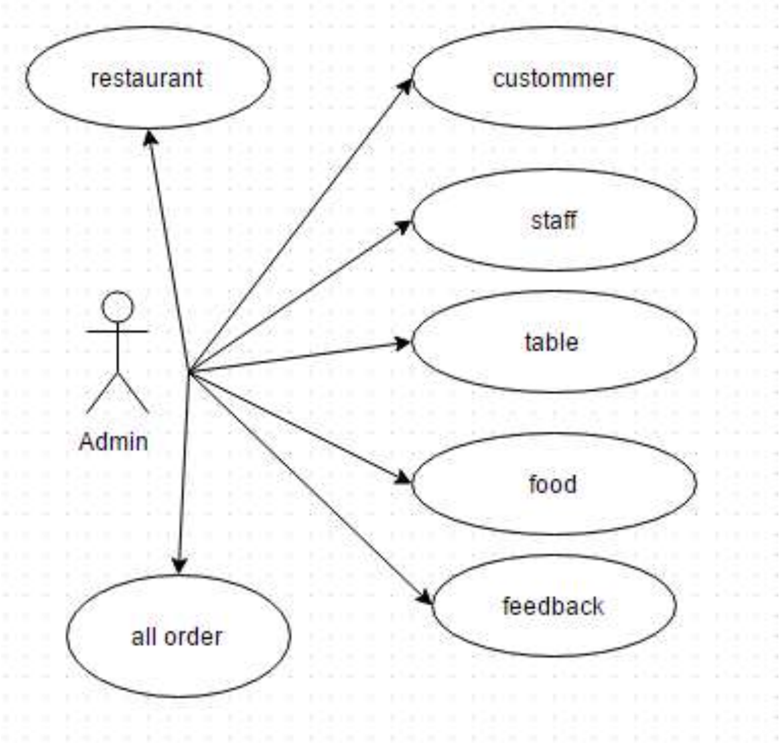


Use Case Name	Delete Feedback	
Actors	Staff	
Description	Staff wants to Delete Feedback	
Requirements	Staff choose Feedback.	
Pre-conditions	Actor has logged in as Staff	
Post-conditions	Success: delete success.	
Basic flow	Actor Actions: 1. Actor clicks ‘Feedback’ link in Staff Home page	System Responses: 2. System redirects to Feedback page and displays list of feedback and following control : - ‘Delete’ button 4. System checks the information

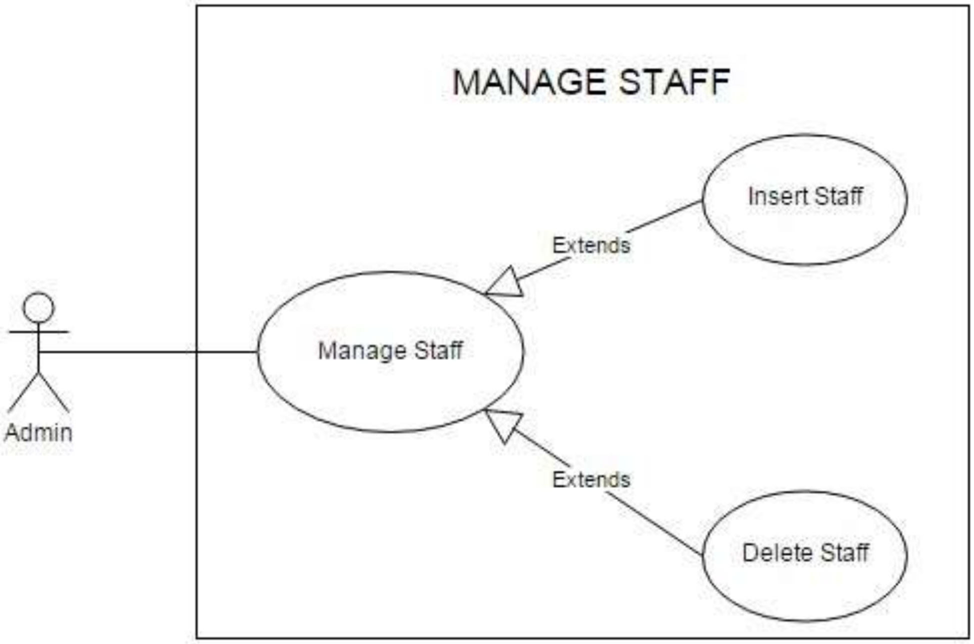
Admin:

(Note: Management of admin include: all management of staff and private management of admin).



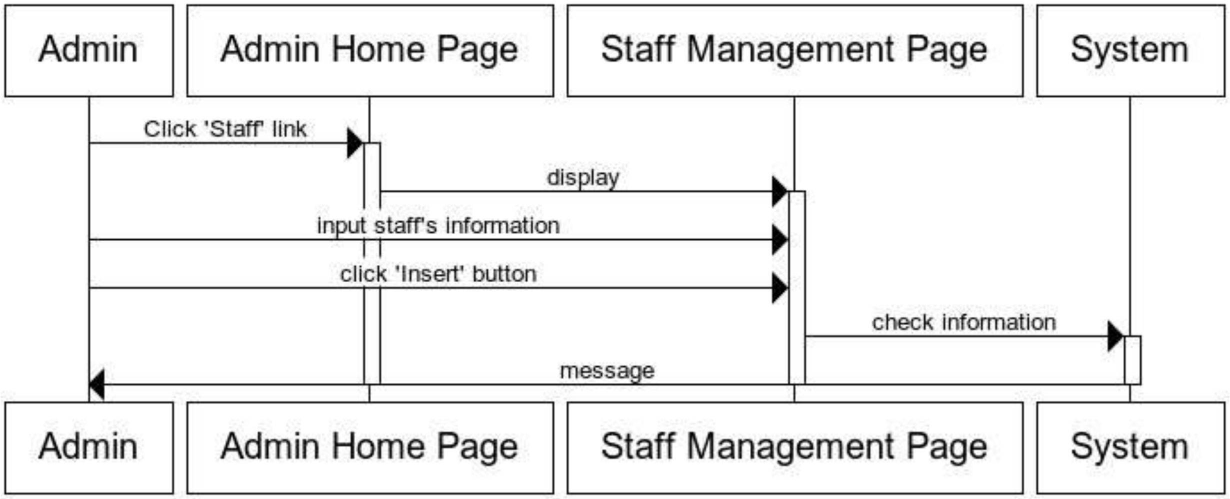


Manage Staff:



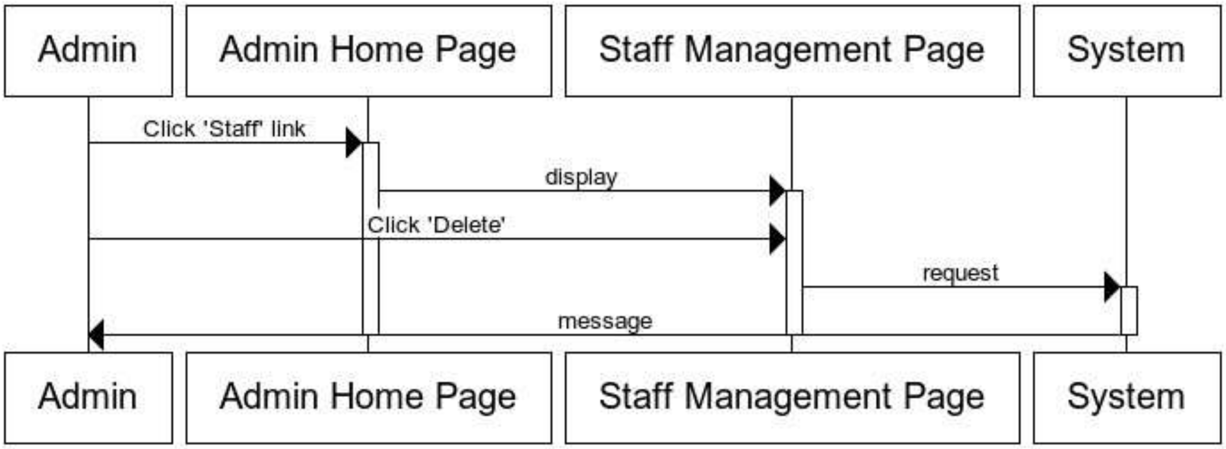
Insert Staff:

Use Case Name	Insert Staff	
Actors	Admin	
Description	Admin wants to insert a staff.	
Requirements	Admin provides Staff's information	
Pre-conditions	Actor has logged in as Admin.	
Post-conditions	Success: Insert Staff Success. Fail: Refill informati	
Basic flow	Actor Actions: 1. Actor clicks 'Staff' link in Admin Home page 3. Actor inputs staff's information and click 'Insert' button.	- System Responses: - 2. System redirects to Staff Management page and Insert Staff form with the following control : - 'UserName' text field - 'Password' text field - 'Insert' button - 'Cancel' button - 4. System checks the information - [Exception] - 5. System inserts the Staff into database, show message: 'Insert Staff Success!' and then redirects to Admin Home page.
Exception	Actor Actions: [Exception] 1. Actor lefts 'UserName' text field blank. 2. UserName has already exists. 3. Actor lefts 'Password' text field blank.	System Responses: System displays message to Actor : - 'UserName cannot be blank'. - 'This Username has already exist!'. - 'Password cannot be blank''.

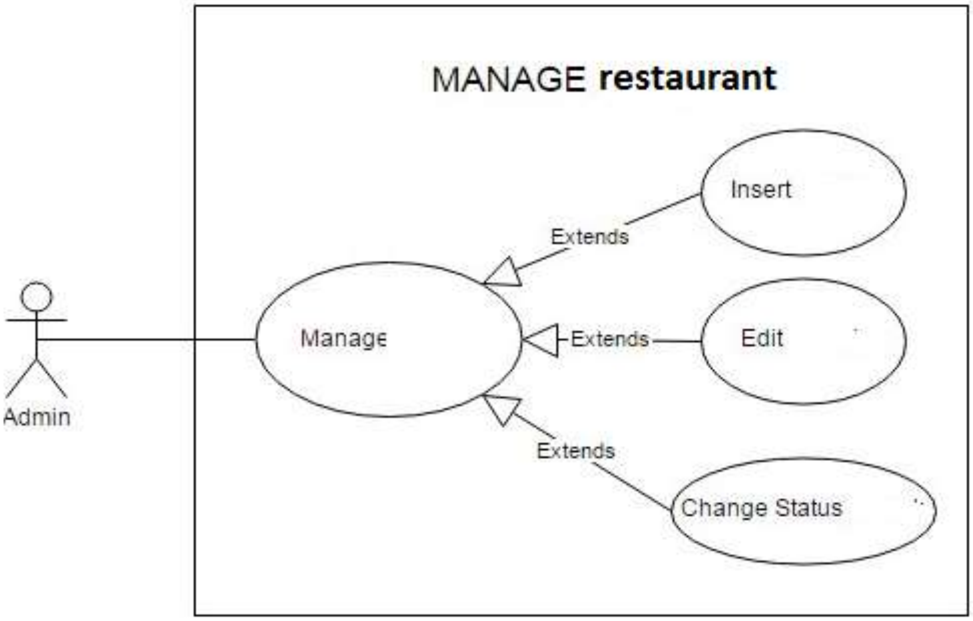


Delete Staff:

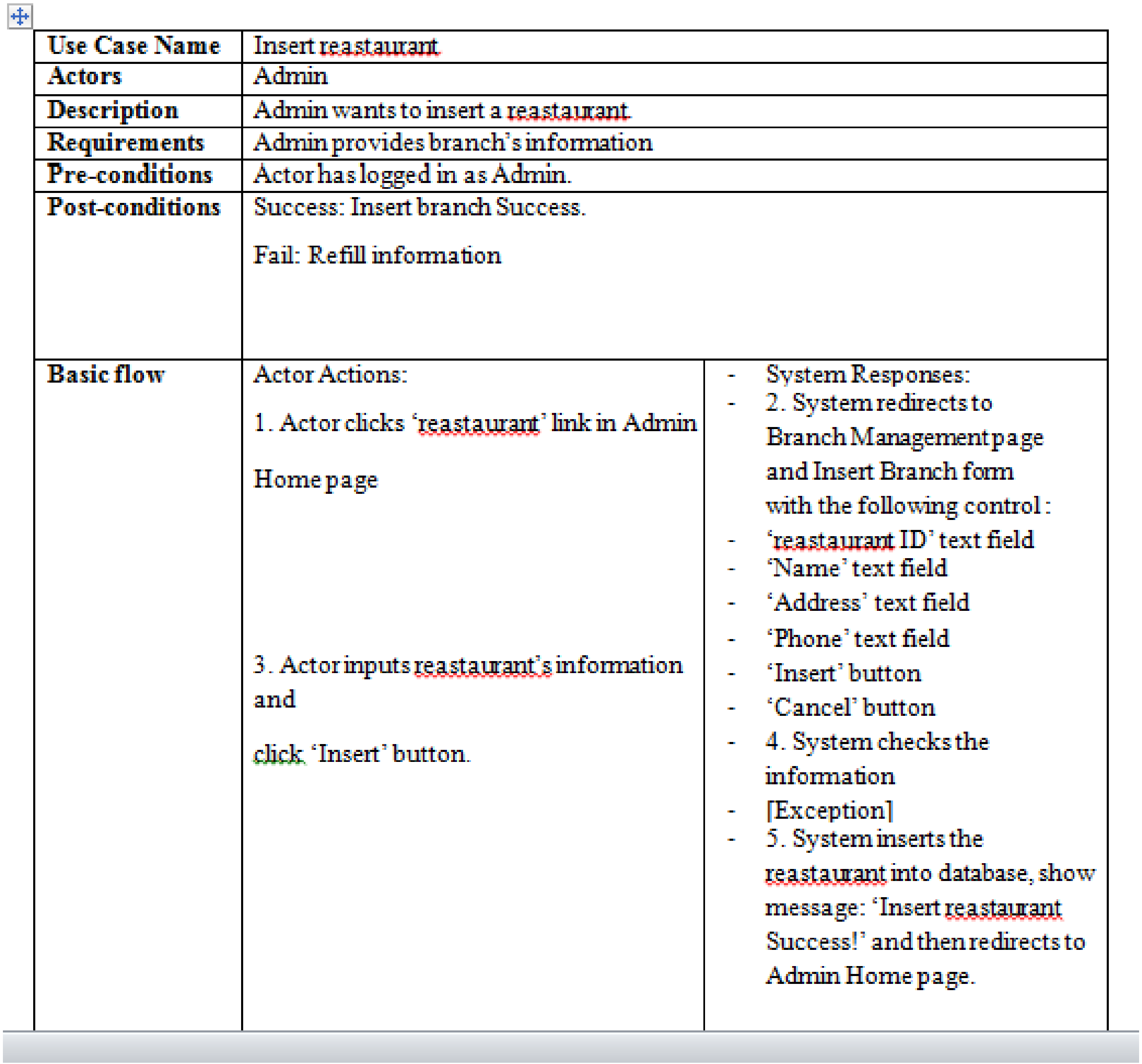
Use Case Name	Delete Staff	
Actors	Admin	
Description	Admin wants to Delete Staff	
Requirements	Admin choose Staff.	
Pre-conditions	Actor has logged in as Admin	
Post-conditions	Success: delete success.	
Basic flow	Actor Actions: 1. Actor clicks ‘Staff’ link in Admin Home page 3. Actor click ‘Delete’ button.	System Responses: 2. System redirects to Staff Management page and displays list of Staff and following control : ‘Delete’ button 4. System checks the information System save into database, show message: ‘Delete Success!’ and then redirects to Admin Home page.



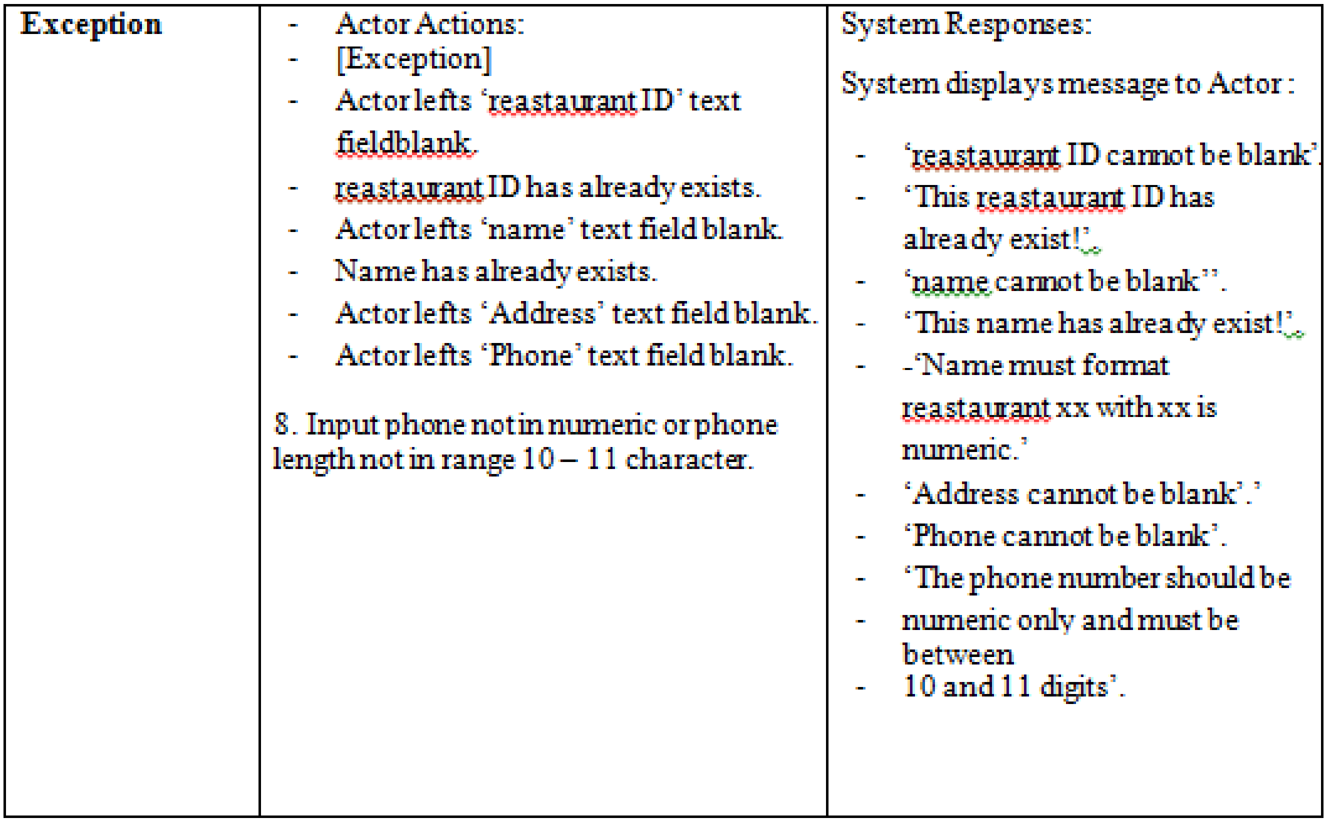
Manage restaurant:



Insert restaurant:

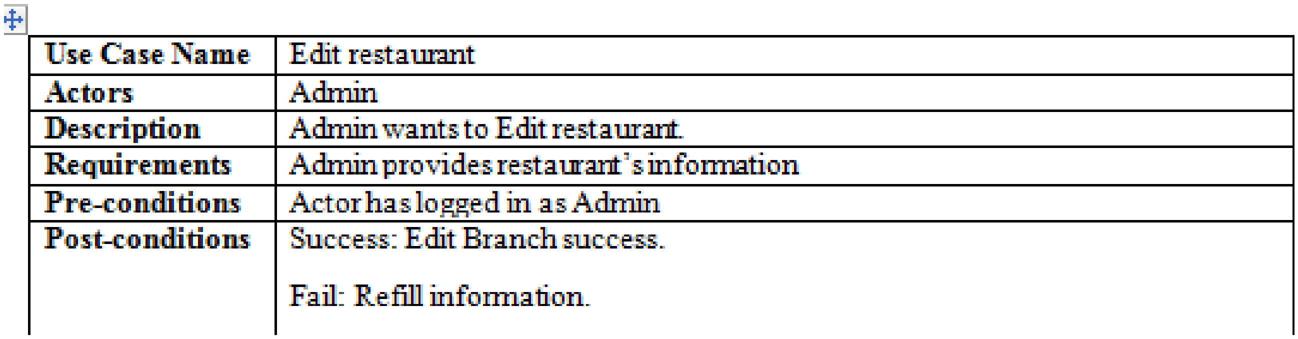


Use Case Name	Insert reastaurant	
Actors	Admin	
Description	Admin wants to insert a reastaurant	
Requirements	Admin provides branch's information	
Pre-conditions	Actor has logged in as Admin.	
Post-conditions	Success: Insert branch Success. Fail: Refill information	
Basic flow	Actor Actions: 1. Actor clicks ‘reastaurant’ link in Admin Home page 3. Actor inputs reastaurant’s information and click ‘Insert’ button.	- System Responses: - 2. System redirects to Branch Management page and Insert Branch form with the following control : - ‘reastaurant ID’ text field - ‘Name’ text field - ‘Address’ text field - ‘Phone’ text field - ‘Insert’ button - ‘Cancel’ button - 4. System checks the information - [Exception] - 5. System inserts the reastaurant into database, show message: ‘Insert reastaurant Success!’ and then redirects to Admin Home page.



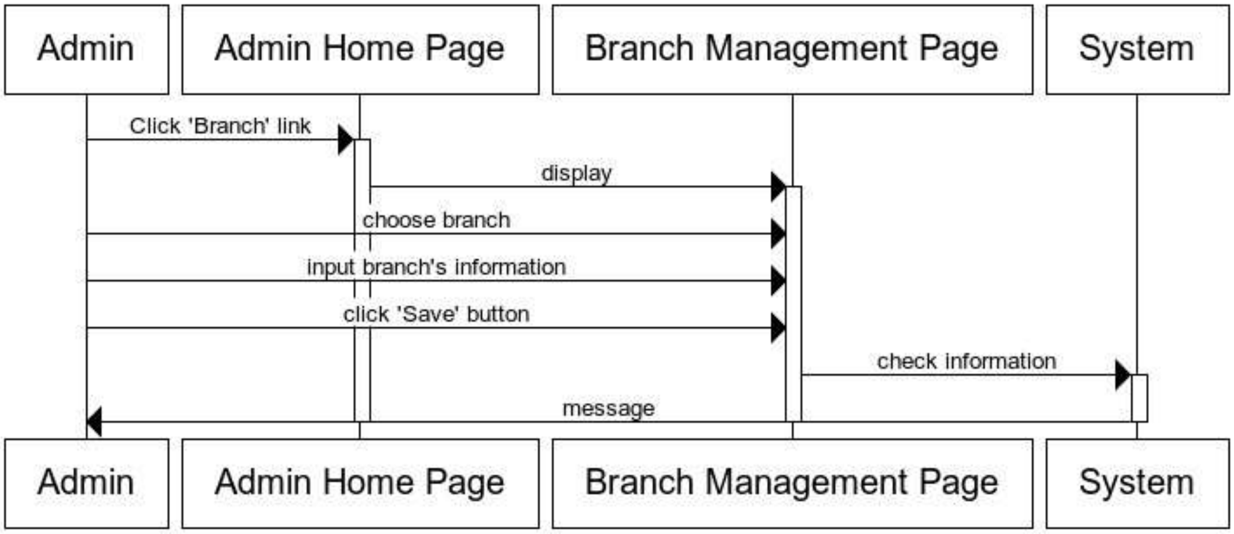
Exception	- Actor Actions: - [Exception] - Actor lefts 'reastaurant ID' text fieldblank. - reastaurant ID has already exists. - Actor lefts 'name' text field blank. - Name has already exists. - Actor lefts 'Address' text field blank. - Actor lefts 'Phone' text field blank. 8. Input phone not in numeric or phone length not in range 10 – 11 character.	System Responses: System displays message to Actor : - 'reastaurant ID cannot be blank'. - 'This reastaurant ID has already exist!'. - 'name cannot be blank''. - 'This name has already exist!'. - -'Name must format reastaurant xx with xx is numeric.' - 'Address cannot be blank'.' - 'Phone cannot be blank'. - 'The phone number should be - numeric only and must be between - 10 and 11 digits'.

Edit Restaurant:



Use Case Name	Edit restaurant
Actors	Admin
Description	Admin wants to Edit restaurant.
Requirements	Admin provides restaurant's information
Pre-conditions	Actor has logged in as Admin
Post-conditions	Success: Edit Branch success. Fail: Refill information.

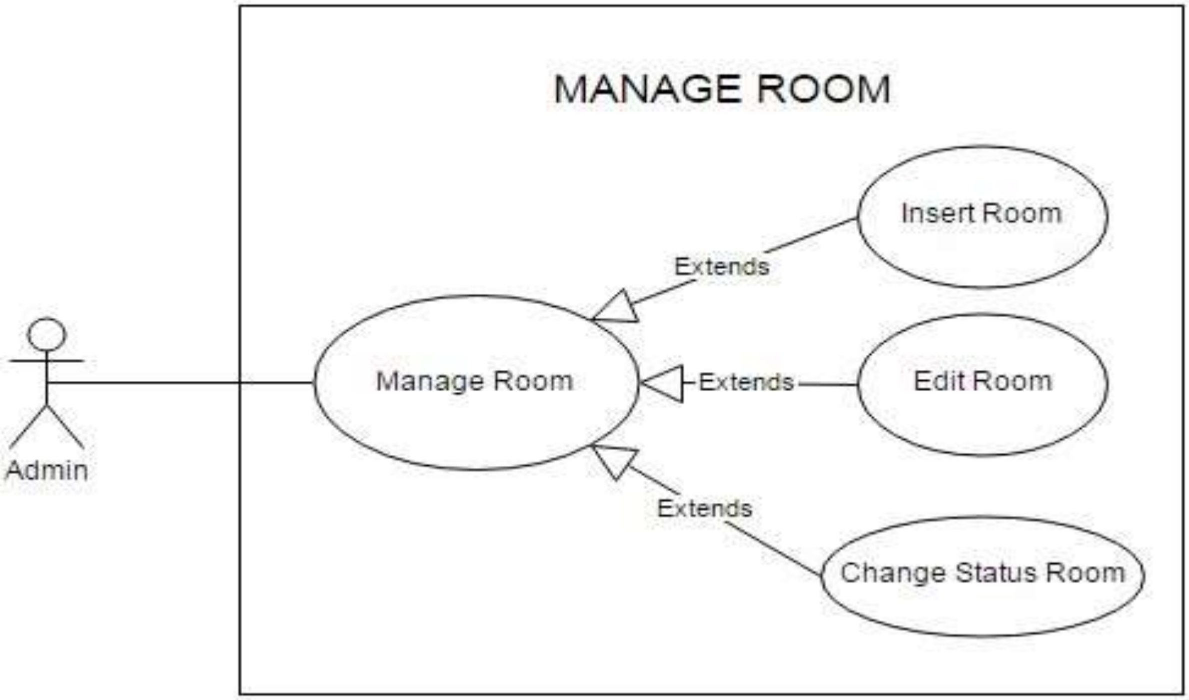
		5. System save information into database, show message: ‘Edit restaurant Success!’ and then redirects to Admin Home page.
Exception	Actor Actions: [Exception] 1. Actor lefts ‘Address’ text field blank. 2. Actor lefts ‘Phone’ text field blank. 3. Input phone not in numeric or phone length not in range 10 – 11 character.	System Responses: System displays message to Actor : - ‘Address cannot be blank’.’ - ‘Phone cannot be blank’. - ‘The phone number should be numeric only and must be between 10 and 11 digits’.



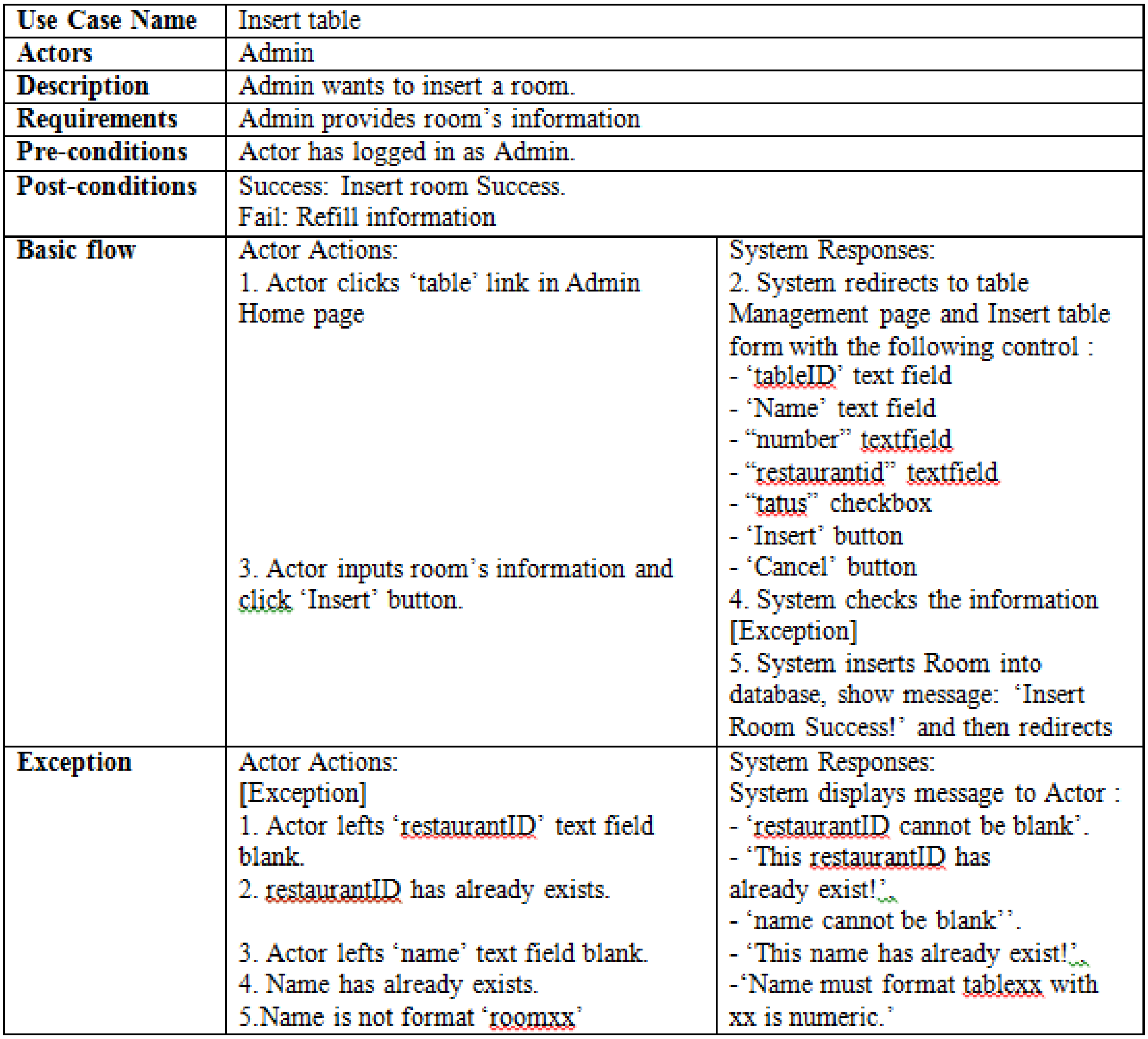
Change Status restaurant:

Use Case Name	Change Status Branch	
Actors	Admin	
Description	Admin wants to change status of restaurant.	
Requirements	Admin choose restaurant.	
Pre-conditions	Actor has logged in as Admin	
Post-conditions	Success: change status branch success.	
Basic flow	Actor Actions: 1. Actor clicks ‘restaurant’ link in Admin Home page Actor click ‘Active’ button or ‘Not Active’ button	System Responses: 2. System redirects to restaurant Management Page and change status branch form with the following control : if restaurant is active, ‘not active’ button display. - if restaurant is not active, ‘active’ button display. 4. System checks the information 5. System save information into database, show message: ‘Change Status Success!’ and then redirects to Admin Home page

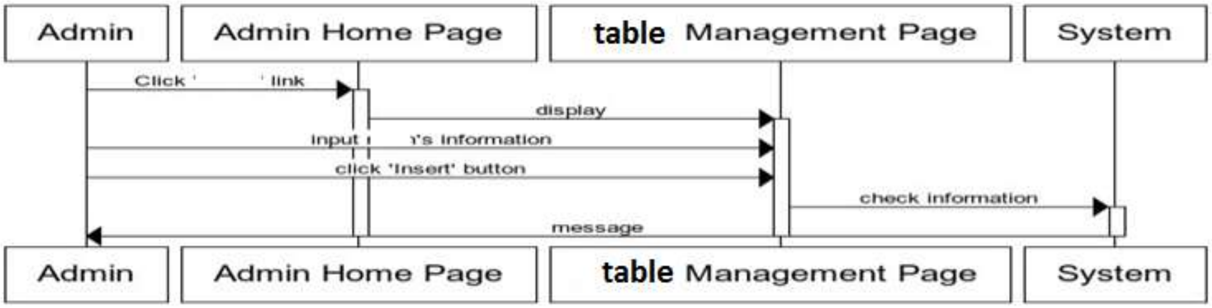
Manage Room:



Insert table:

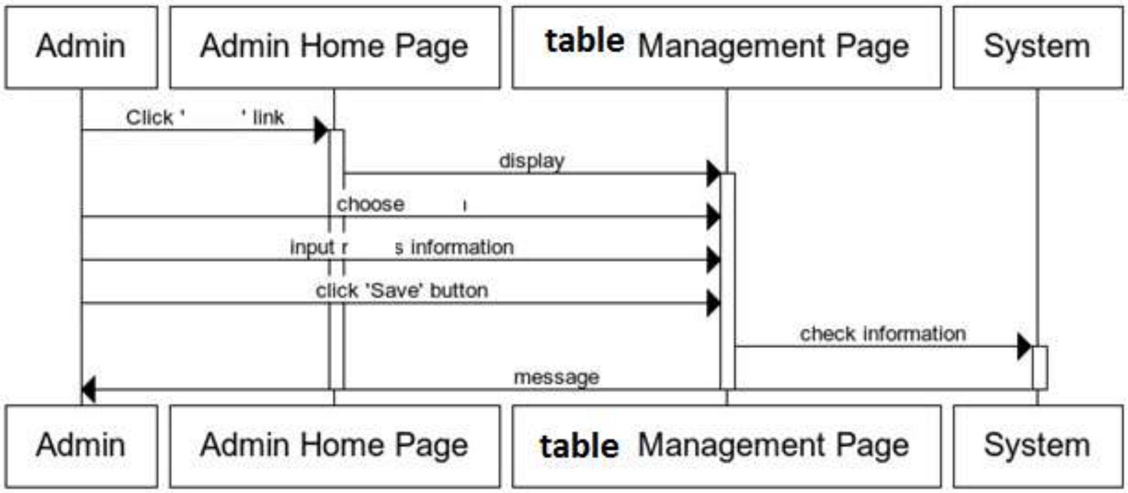


Use Case Name	Insert table	
Actors	Admin	
Description	Admin wants to insert a room.	
Requirements	Admin provides room's information	
Pre-conditions	Actor has logged in as Admin.	
Post-conditions	Success: Insert room Success. Fail: Refill information	
Basic flow	Actor Actions: 1. Actor clicks 'table' link in Admin Home page 3. Actor inputs room's information and click 'Insert' button.	System Responses: 2. System redirects to table Management page and Insert table form with the following control : - 'tableID' text field - 'Name' text field - "number" textfield - "restaurantid" textfield - "tatus" checkbox - 'Insert' button - 'Cancel' button 4. System checks the information [Exception] 5. System inserts Room into database, show message: 'Insert Room Success!' and then redirects
Exception	Actor Actions: [Exception] 1. Actor lefts 'restaurantID' text field blank. 2. restaurantID has already exists. 3. Actor lefts 'name' text field blank. 4. Name has already exists. 5.Name is not format 'roomxx'	System Responses: System displays message to Actor : - 'restaurantID cannot be blank'. - 'This restaurantID has already exist!'. - 'name cannot be blank''. - 'This name has already exist!'. -'Name must format tablexx with xx is numeric.'



Edit table:

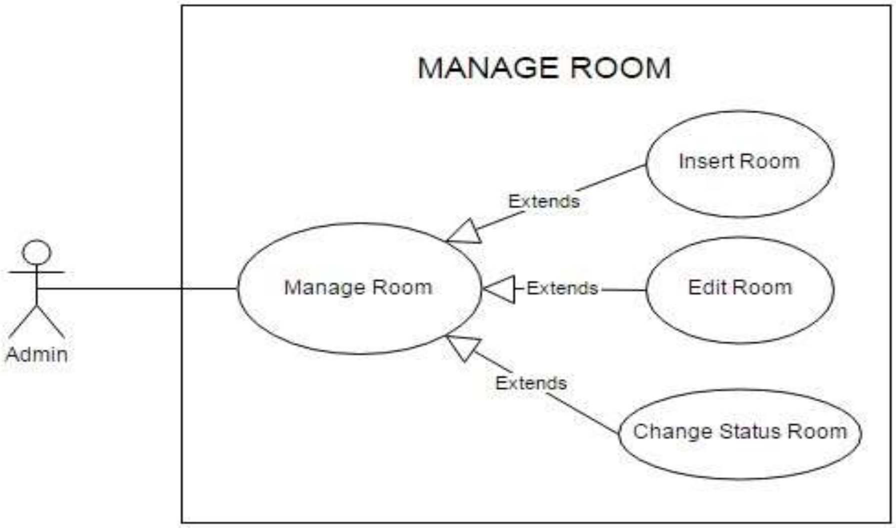
Use Case Name	Edit table	
Actors	Admin	
Description	Admin wants to Edit table.	
Requirements	Admin provides Room’s information	
Pre-conditions	Actor has logged in as Admin	
Post-conditions	Success: Edit Room success. Fail: Refill information.	
Basic flow	Actor Actions: 1. Actor clicks ‘table’ link in Admin Home page 3. Actor inputs Rooml information and click ‘Save’ button.	System Responses: 2. System redirects to Room Management Page and displays edit room form with the following control : - ‘table ID’ text field, read only. - ‘name’ text field,read only. - “number” textfield - “restaurantid” textfield - “tatus” checkbox - ‘Save’ button - ‘Cancel’ button 4. System checks the information 5. System save information into database, show message: ‘Edit Room Success!’ and then redirects to Admin Home page.



Change Status table:

Use Case Name	Change Status table	
Actors	Admin	
Description	Admin wants to change status of table.	
Requirements	Admin choose room.	
Pre-conditions	Actor has logged in as Admin	
Post-conditions	Success: change status room success.	
Basic flow	Actor Actions: 1. Actor clicks 'table' link in Admin Home page	System Responses: 2. System redirects to table Management Page and change status room form with the following control : - if table is active, 'not active' button display. - if table is not active, 'active' button display.
	3. Actor click 'Active' button or 'Not Active' button.	4. System checks the information 5. System save information into database, show message: 'Change Status Success!' and then redirects to Admin Home page.

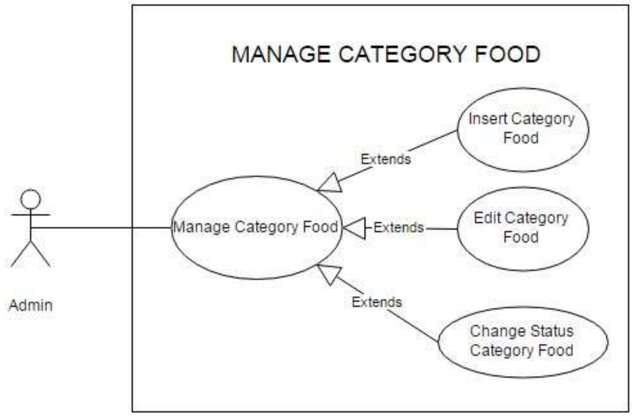
Manage Room:



Change Status table:

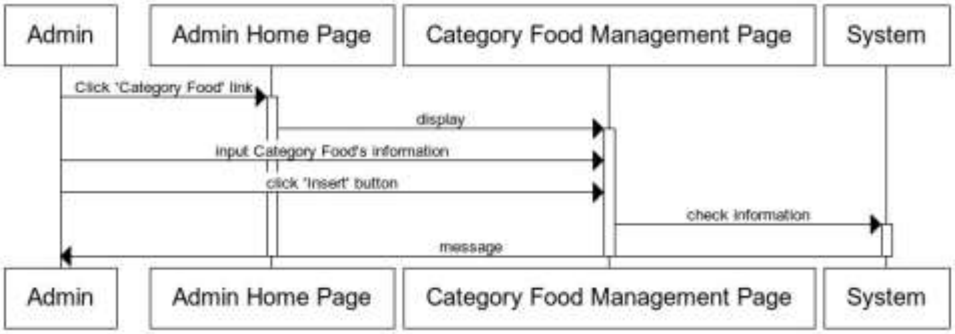
Use Case Name	Change Status table	
Actors	Admin	
Description	Admin wants to change status of table.	
Requirements	Admin choose room.	
Pre-conditions	Actor has logged in as Admin	
Post-conditions	Success: change status room success.	
Basic flow	Actor Actions: 1. Actor clicks 'table' link in Admin Home page 3. Actor click 'Active' button or 'Not Active' button.	System Responses: 2. System redirects to table Management Page and change status room form with the following control : - if table is active, 'not active' button display. - if table is not active, 'active' button display. 4. System checks the information 5. System save information into database, show message: 'Change Status Success!' and then redirects to Admin Home page.

Manage Category Food:

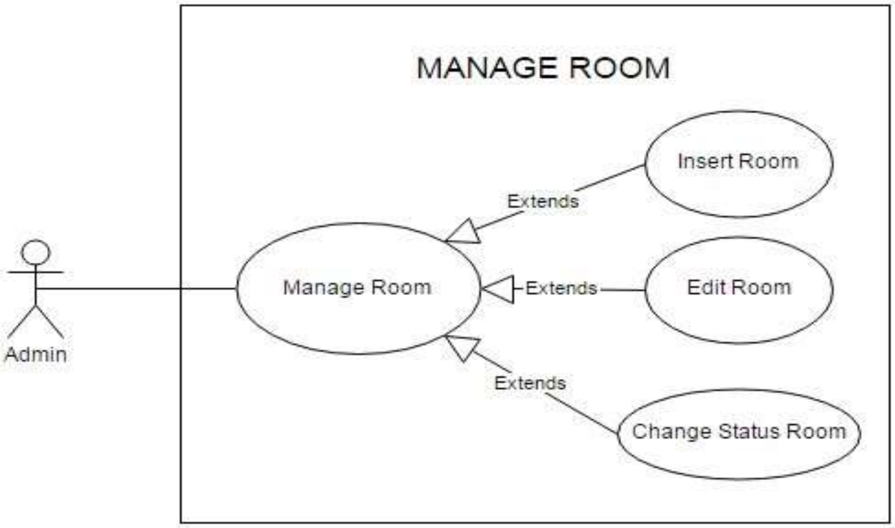


Insert Category Food:

Use Case Name	Insert Category Food	
Actors	Admin	
Description	Admin wants to insert a category Food.	
Requirements	Admin provides category Food ‘s information	
Pre-conditions	Actor has logged in as Admin.	
Post-conditions	Success: Insert category Food Success. Fail: Refill information	
Basic flow	Actor Actions: 1. Actor clicks ‘Food’ link in Admin Home page 3. Actor inputs category Food’s information and click ‘Insert’ button.	System Responses: 2. System redirects to category Food Management page and Insert category Food form with the following control : - ‘categorID’ text field - ‘Name’ text field -“status” checkbox - ‘Insert’ button - ‘Cancel’ button 4. System checks the information [Exception] 5. System inserts category Food into database, show message: ‘Insert category Food Success!’ and then redirects to Admin Home
Exception	Actor Actions: [Exception] 3. Actor lefts ‘name’ text field blank. 4. Name has already exists.	System Responses: System displays message to Actor : - ‘name cannot be blank’’. - ‘This name has already exist!’.


Admin
Admin Home Page
Category Food Management Page
System
Click 'Category Food' link
display
input Category Food's information
click 'Insert' button
check information
message
Admin
Admin Home Page
Category Food Management Page
System

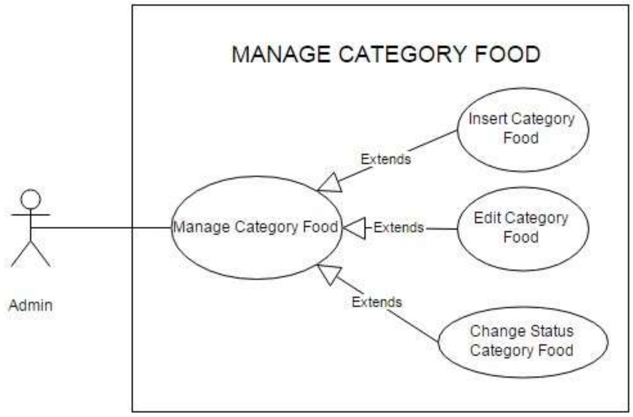
Manage Room:


MANAGE ROOM
Insert Room
Extends
Manage Room
Extends
Edit Room
Admin
Extends
Change Status Room

Change Status table:

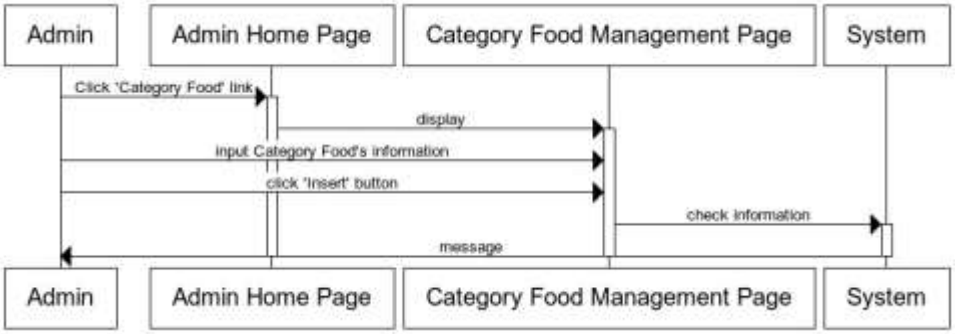
Use Case Name	Change Status table	
Actors	Admin	
Description	Admin wants to change status of table.	
Requirements	Admin choose room.	
Pre-conditions	Actor has logged in as Admin	
Post-conditions	Success: change status room success.	
Basic flow	Actor Actions: 1. Actor clicks 'table' link in Admin Home page 3. Actor click 'Active' button or 'Not Active' button.	System Responses: 2. System redirects to table Management Page and change status room form with the following control : - if table is active, 'not active' button display. - if table is not active, 'active' button display. 4. System checks the information 5. System save information into database, show message: 'Change Status Success!' and then redirects to Admin Home page.

Manage Category Food:



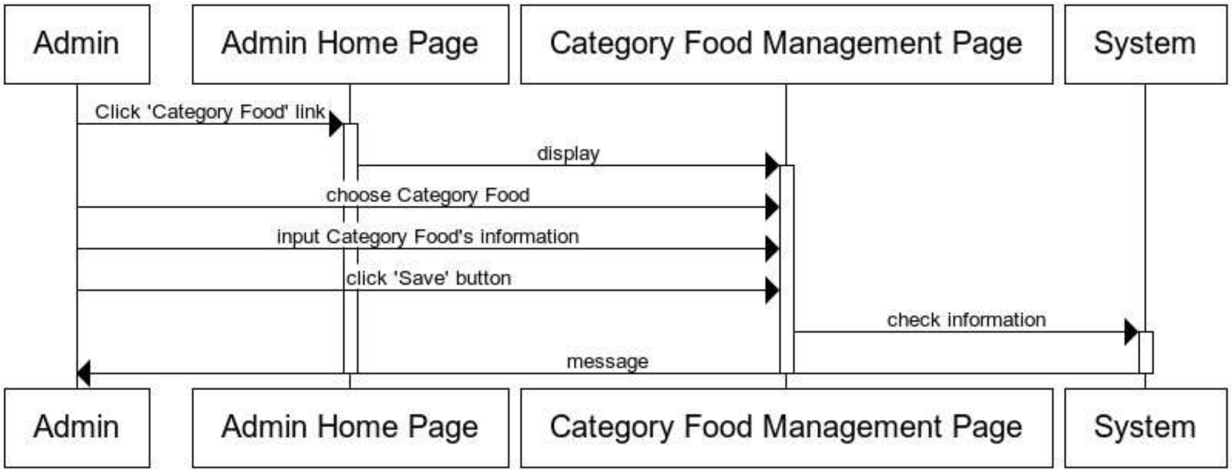
Insert Category Food:

Use Case Name	Insert Category Food	
Actors	Admin	
Description	Admin wants to insert a category Food.	
Requirements	Admin provides category Food ‘s information	
Pre-conditions	Actor has logged in as Admin.	
Post-conditions	Success: Insert category Food Success. Fail: Refill information	
Basic flow	Actor Actions: 1. Actor clicks ‘Food’ link in Admin Home page 3. Actor inputs category Food’s information and click ‘Insert’ button.	System Responses: 2. System redirects to category Food Management page and Insert category Food form with the following control : - ‘categorID’ text field - ‘Name’ text field -“status” checkbox - ‘Insert’ button - ‘Cancel’ button 4. System checks the information [Exception] 5. System inserts category Food into database, show message: ‘Insert category Food Success!’ and then redirects to Admin Home
Exception	Actor Actions: [Exception] 3. Actor lefts ‘name’ text field blank. 4. Name has already exists.	System Responses: System displays message to Actor : - ‘name cannot be blank’”. - ‘This name has already exist!’.


Admin
Admin Home Page
Category Food Management Page
System
Click 'Category Food' link
display
input Category Food's information
click 'Insert' button
check information
message
Admin
Admin Home Page
Category Food Management Page
System

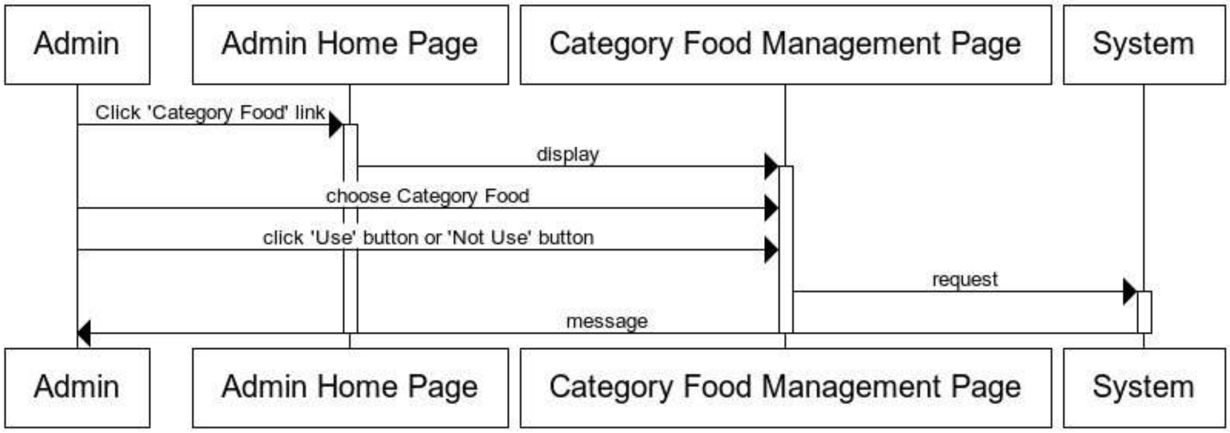
Edit Category Food:

Use Case Name	Edit Category Food	
Actors	Admin	
Description	Admin wants to Edit Category Food.	
Requirements	Admin provides Category Food’s information	
Pre-conditions	Actor has logged in as Admin	
Post-conditions	Success: Edit Category Food success. Fail: Refill information.	
Basic flow	Actor Actions: 1. Actor clicks ‘Category Food’ link in Admin Home page 3. Actor inputs Category Food information and click ‘Save’ button.	System Responses: 2. System redirects to Category Food Management Page and displays edit Category Food form with the following control : - ‘categorID’ text field - ‘Name’ text field -“status” checkbox - ‘Save’ button - ‘Cancel’ button 4. System checks the information 5. System save information into database, show message: ‘Edit Category Food Success!’ and then redirects to Admin Home page.

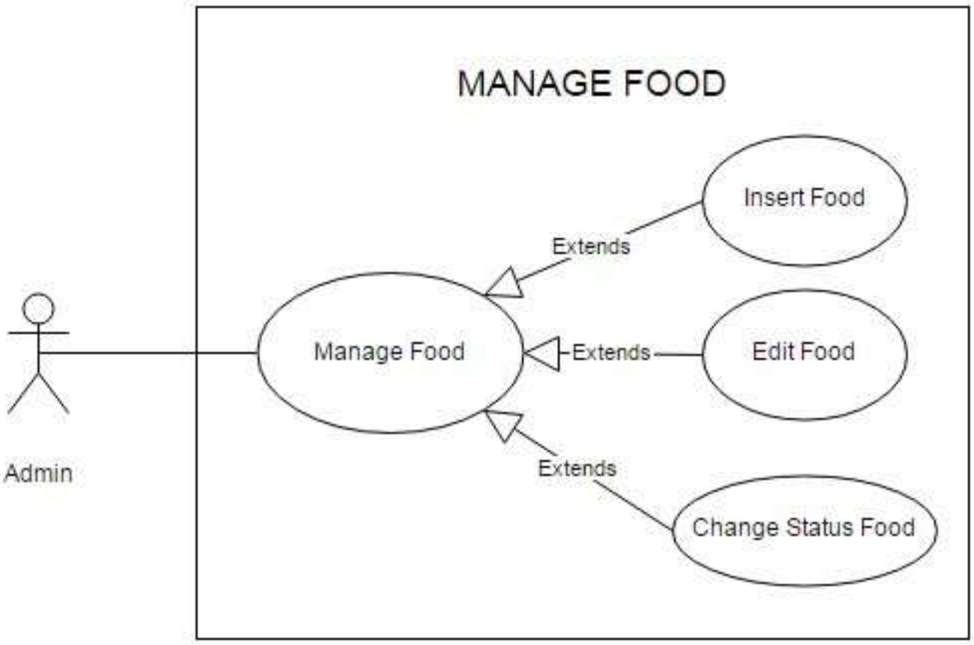


Change Status Category Table:

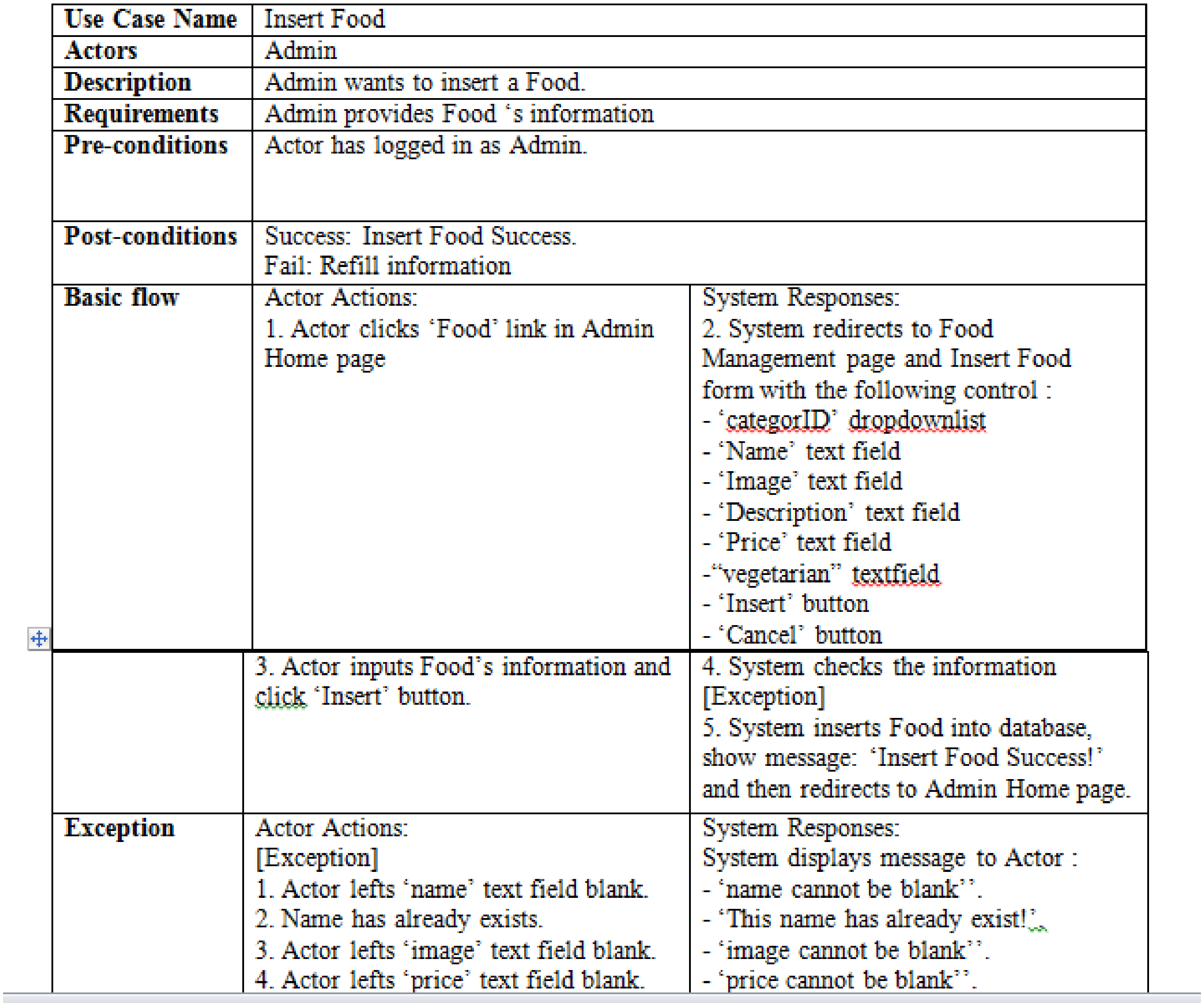
Use Case Name	Change Status Category Food	
Actors	Admin	
Description	Admin wants to change status of Category Food.	
Requirements	Admin choose Category Food.	
Pre-conditions	Actor has logged in as Admin	
Post-conditions	Success: change status Category Food success.	
Basic flow	Actor Actions: 1. Actor clicks 'Category Food' link in Admin Home page 3. Actor click 'Use' button or 'Not Use' button.	System Responses: 2. System redirects to Category Food Management Page and change status Category Food form with the following control : - if Category Food is use, 'Not Use' button display. - if Category Food is not use, 'Use' button display. 4. System checks the information 5. System save information into database, show message: 'Change Status Success!' and then redirects to Admin Home page.


Admin
Admin Home Page
Category Food Management Page
System
Click 'Category Food' link
display
choose Category Food
click 'Use' button or 'Not Use' button
request
message
Admin
Admin Home Page
Category Food Management Page
System

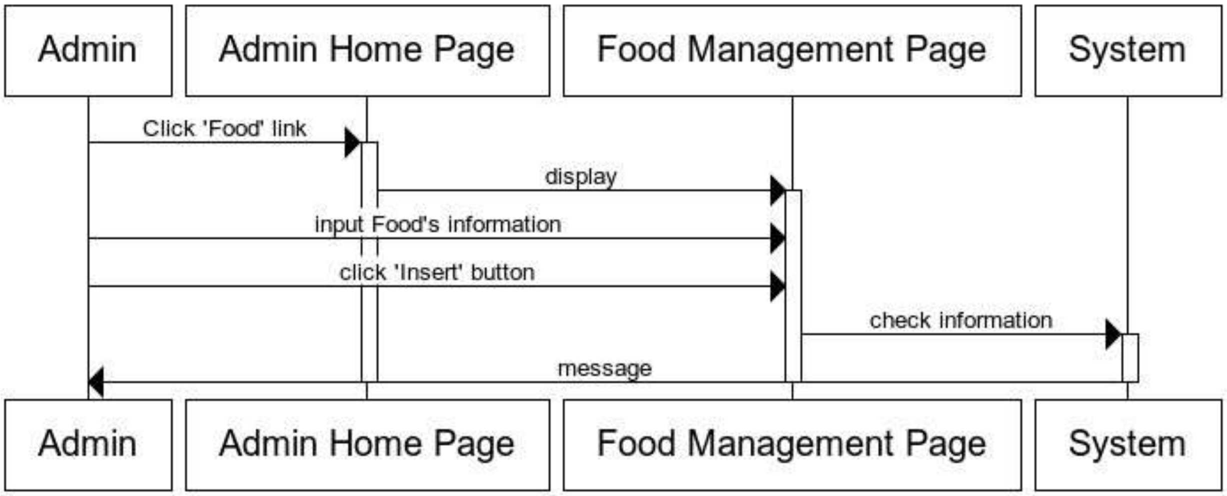
Manage Food:


MANAGE FOOD
Admin
Manage Food
Extends
Insert Food
Extends
Edit Food
Extends
Change Status Food

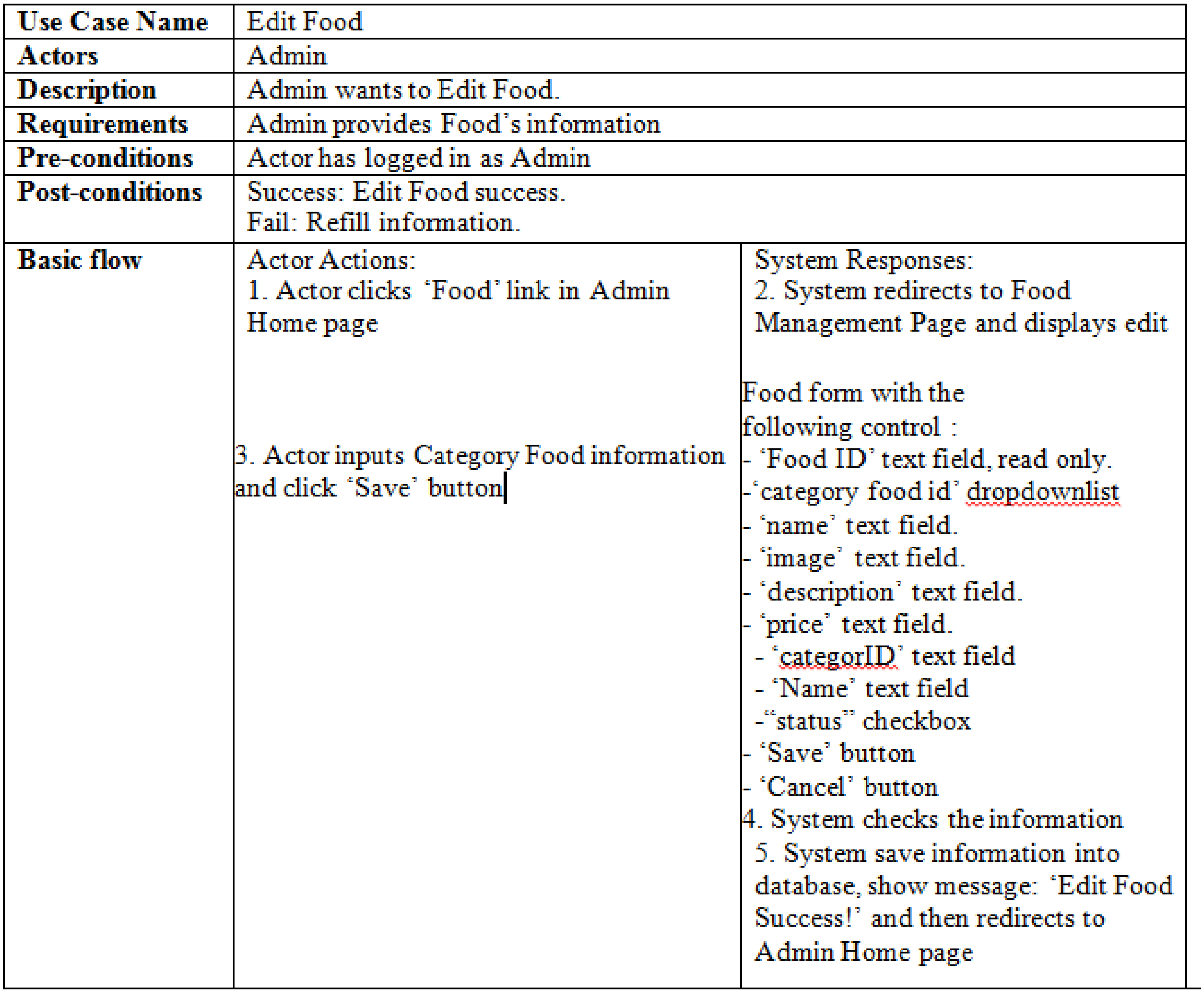
Insert Food:



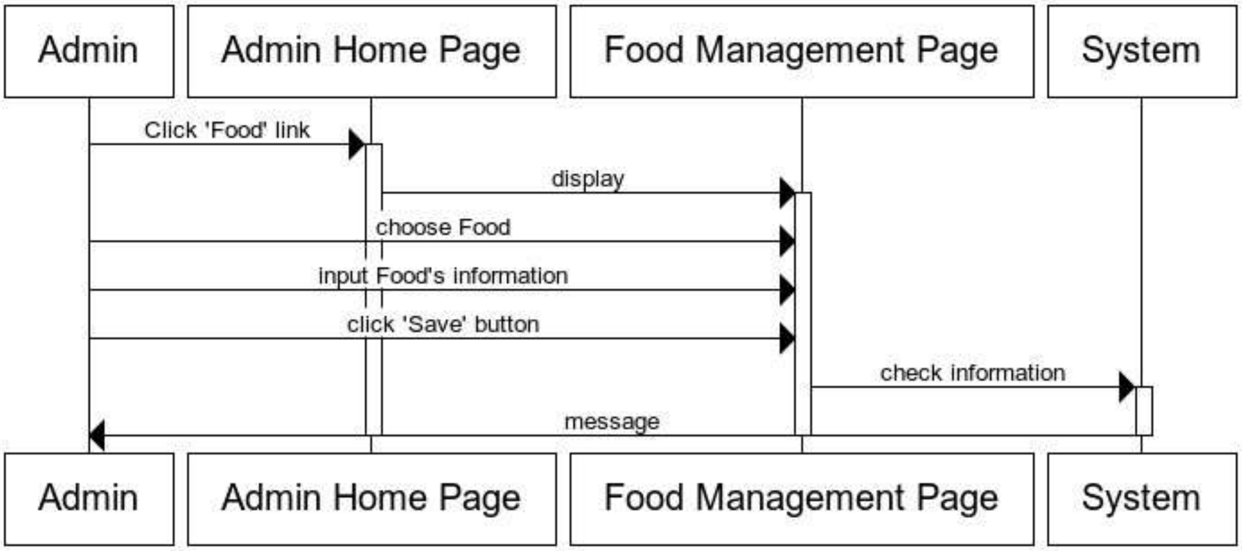
Use Case Name	Insert Food	
Actors	Admin	
Description	Admin wants to insert a Food.	
Requirements	Admin provides Food ‘s information	
Pre-conditions	Actor has logged in as Admin.	
Post-conditions	Success: Insert Food Success. Fail: Refill information	
Basic flow	Actor Actions: 1. Actor clicks ‘Food’ link in Admin Home page	System Responses: 2. System redirects to Food Management page and Insert Food form with the following control : - ‘categorID’ dropdownlist - ‘Name’ text field - ‘Image’ text field - ‘Description’ text field - ‘Price’ text field -“vegetarian” textfield - ‘Insert’ button - ‘Cancel’ button
	3. Actor inputs Food’s information and click ‘Insert’ button.	4. System checks the information [Exception] 5. System inserts Food into database, show message: ‘Insert Food Success!’ and then redirects to Admin Home page.
Exception	Actor Actions: [Exception] 1. Actor lefts ‘name’ text field blank. 2. Name has already exists. 3. Actor lefts ‘image’ text field blank. 4. Actor lefts ‘price’ text field blank.	System Responses: System displays message to Actor : - ‘name cannot be blank’’. - ‘This name has already exist!’. - ‘image cannot be blank’’. - ‘price cannot be blank’’.



Edit Category Food:

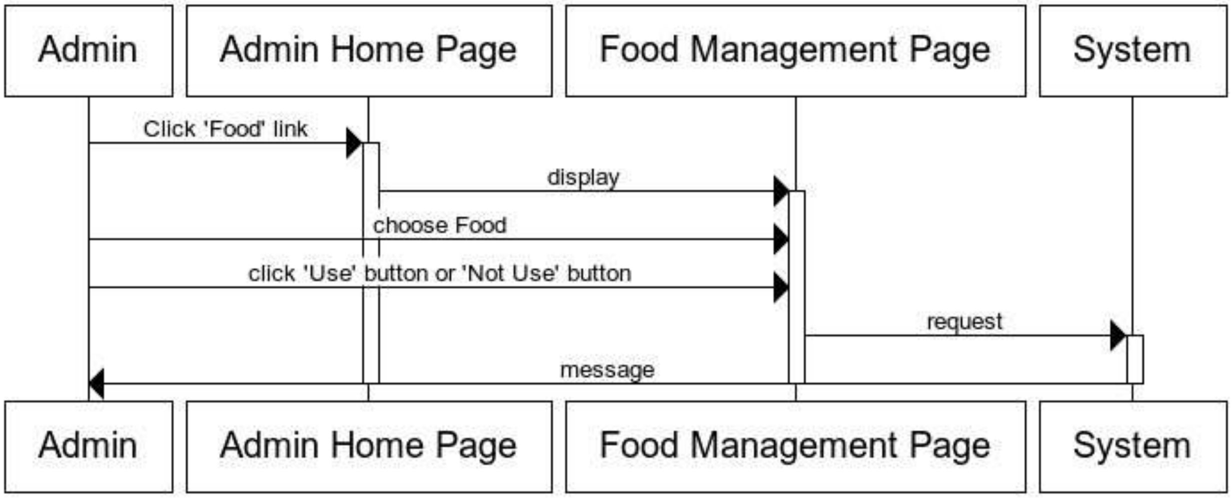


Use Case Name	Edit Food	
Actors	Admin	
Description	Admin wants to Edit Food.	
Requirements	Admin provides Food's information	
Pre-conditions	Actor has logged in as Admin	
Post-conditions	Success: Edit Food success. Fail: Refill information.	
Basic flow	Actor Actions: 1. Actor clicks 'Food' link in Admin Home page 3. Actor inputs Category Food information and click 'Save' button	System Responses: 2. System redirects to Food Management Page and displays edit Food form with the following control : - 'Food ID' text field, read only. -'category food id' dropdownlist - 'name' text field. - 'image' text field. - 'description' text field. - 'price' text field. - 'categorID' text field - 'Name' text field -"status" checkbox - 'Save' button - 'Cancel' button 4. System checks the information 5. System save information into database, show message: 'Edit Food Success!' and then redirects to Admin Home page

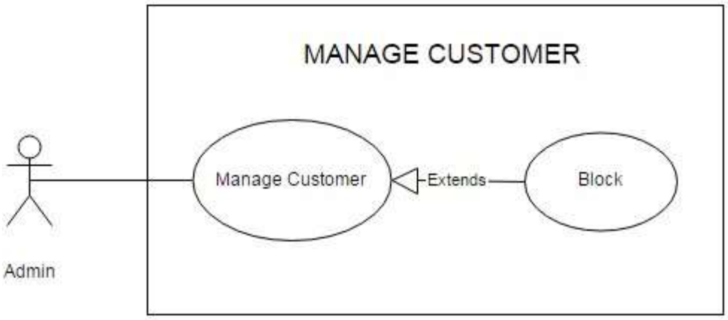


Change Status Food:

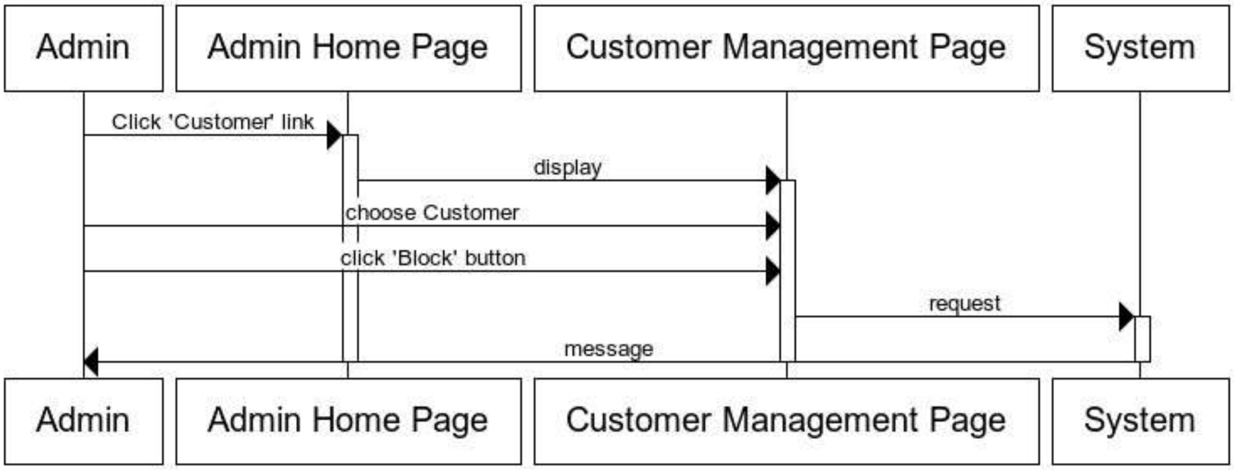
Use Case Name	Change Status Food	
Actors	Admin	
Description	Admin wants to change status of Food.	
Requirements	Admin choose Food.	
Pre-conditions	Actor has logged in as Admin	
Post-conditions	Success: change status Food success.	
Basic flow	Actor Actions: 1. Actor clicks 'Food' link in Admin Home page 3. Actor click 'Use' button or 'Not Use' button.	System Responses: 2. System redirects to Food Management Page and change status Food form with the following control : - if Food is use, 'Not Use' button display. - if Food is not use, 'Use' button display. 4. System checks the information 5. System save information into database, show message: 'Change Status Success!' and then redirects to Admin Home page.



Manage Customer:



Use Case Name	Block Customer	
Actors	Admin	
Description	Admin wants to Block Customer.	
Requirements	Admin choose Customer.	
Pre-conditions	Actor has logged in as Admin	
Post-conditions	Success: block customer success.	
Basic flow	Actor Actions: 1. Actor clicks ‘Customer’ link in Admin Home page 3. Actor click ‘Block’ button	System Responses: 2. System redirects to Customer Management Page and block customer form with the following control : -‘Block’ button. 4. System checks the information 5. System save information into database, show message: ‘Block Customer Success!’ and then redirects to Admin Home page.


Admin
Admin Home Page
Customer Management Page
System
Click 'Customer' link
display
choose Customer
click 'Block' button
request
message
Admin
Admin Home Page
Customer Management Page
System