**Cake Website**

**Development’s Guide**

**Faculty: Duong Trong Phu Son**

**Batch No: F3-13-1312-T0**

**Group No: 1**

**Student:**

|  |  |  |
| --- | --- | --- |
| **No** | **Full Name** | **Roll No** |
| **1** | **Tran Quang Tung** | **Student449703** |
| 2 | Dang Tran Quoc Thinh | 25.001/200510/0450 |
| 3 | Nguyen Vo Thanh Huy | Student399304 |

Review 1

[1. Introduction 4](#_Toc428012342)

[2. Existing Scenario 4](#_Toc428012343)

[3. Customer Requirement Specification 4](#_Toc428012344)

[Home page 4](#_Toc428012345)

[About Us 4](#_Toc428012346)

[All Product 4](#_Toc428012347)

[Order Page 4](#_Toc428012348)

[Product by event 5](#_Toc428012349)

[Store location 5](#_Toc428012350)

[Site map 5](#_Toc428012351)

[Manager product 5](#_Toc428012352)

[Manager user 5](#_Toc428012353)

[Manager order 5](#_Toc428012354)

[4. Functional Requirement Specification 5](#_Toc428012355)

[Home page 5](#_Toc428012357)

[All Product 6](#_Toc428012358)

[Guest, User can view product detail 6](#_Toc428012359)

[4.4 Admin can manager event 7](#_Toc428012360)

[4.5 Admin can manager order 7](#_Toc428012361)

[4.6 Admin can view system statitics 7](#_Toc428012362)

[5. System Requirements 8](#_Toc428012363)

[5.1 Hardware 8](#_Toc428012364)

[5.2 Software 8](#_Toc428012365)

[6. Development Software 9](#_Toc428012366)

[7. Techonology 9](#_Toc428012367)

[8. Scope of work 9](#_Toc428012368)

[9. Task Sheet 10](#_Toc428012369)

Review 2

[1. Use Case and Sequence Diagram 12](#_Toc428012370)

[1.1. Guest: 12](#_Toc428012371)

[1.2. ADMIN 26](#_Toc428012372)

[2. Entity Relationship 45](#_Toc428012373)

[2.1. ERD 45](#_Toc428012375)

[2.2. Entity 45](#_Toc428012376)

[3. Task Sheet 50](#_Toc428012378)

Review 3

[1. Database Relationship Diagram 52](#_Toc428012379)

[2. Table Definition 53](#_Toc428012380)

[2.1. Admins 53](#_Toc428012381)

[2.2. Cakes 53](#_Toc428012382)

[2.3. Customers 54](#_Toc428012384)

[2.4. Events 54](#_Toc428012386)

[2.5. Feedback 55](#_Toc428012388)

[2.6. Order Detail 56](#_Toc428012390)

[2.7. Order 56](#_Toc428012392)

[3. Sitemap 58](#_Toc428012394)

[3.1. Admin 58](#_Toc428012395)

[4.2. User 59](#_Toc428012399)

[4. GUI Design 60](#_Toc428012400)

[4.1. Home Site 60](#_Toc428012401)

[4.1.1. Home page 60](#_Toc428012402)

[4.1.2. Login Page 62](#_Toc428012403)

[4.1.3. Register page 64](#_Toc428012404)

[4.1.4. Event page 65](#_Toc428012405)

[4.1.5. Search Page 67](#_Toc428012406)

[4.1.6. Send Feedback 67](#_Toc428012407)

[4.1.7. Change Password 68](#_Toc428012408)

[4.1.8. Cart Page 70](#_Toc428012409)

[4.1.9. Checkout 71](#_Toc428012410)

[4.2. Admin Site 73](#_Toc428012411)

[4.2.1. Login 73](#_Toc428012412)

[4.2.2. Home Page 73](#_Toc428012413)

[4.2.3. Cake Manager 74](#_Toc428012414)

[4.2.4. Add Cake 76](#_Toc428012415)

[4.2.5. Update Cake 77](#_Toc428012416)

[4.2.6. User manager 78](#_Toc428012417)

[4.2.7. Event manager 78](#_Toc428012418)

[4.2.8. Update Event 79](#_Toc428012419)

[4.2.9. Add New Event 80](#_Toc428012420)

[4.2.10. Feedback Manager 80](#_Toc428012421)

[5. Task Sheet 82](#_Toc428012422)

**REVIEW 1**

# Introduction

* To create an informative, interactive and business website for Cake manufacturers. The brand name is “The Cake” and the shop is located all over the country. The website has to be informative enough for the end users / viewers to know brief of the various products and its prices and the availability.

# Existing Scenario

* For customer :
* Customer is being given a list of newest products in the market to choose.
* Each product has all details customer needs to know to have a suitable choice.
* Helping customer easily classify and search the products they are looking for.
* Helping customer orders online, save on money, time.
* For admin :
* Easily adding new brands also updating information on old products.
* Convenient to manage customer.
* Convenient to manage product.
* Convenient to manage customer.
* Convenient to manage turnover, profits.

# Customer Requirement Specification

## Home page

* The page should be designed with a banner at the top of the page. The home page should cover a brief description of the entire website / presentation as well as a brief history of cake.

## About Us

* Divide the page into three sections vertically. Provide the foundation, mission and vision on these sections.

## All Product

* The product page should be designed with all the products / thumbnails along with brief details of the individual products, such as : item code, price, egg less, order button ....

## Order Page

* On click on the order button a purchase form page should appear on a new browser window. The purchase form page should have all the form fields to capture the entire transaction and the users' details like : name, address, contact number, item code, quantity, Delivery address, Delivery Date

## Product by event

* On click on the zoom button an enlarged image of the same product should appear on a new browser window

## Store location

* Provide a categorized sub-navigation on this page such as : Birthday, Anniversary, Engagement, Marriage and so on

## Site map

* The site map page should contain the entire navigational structure / diagram in a visual form and all the links (page names) of the diagram should be programmed for hyperlink to the respective pages

## Manager product

* The administrator should be able to add/update/delete the various products

## Manager user

* The administrator should be able to add/update/delete the user details

## Manager order

* The administrator should be able to view/delete the order

# Functional Requirement Specification



## Home page

* + 1. Description
* The homepage displays products list and categories, events, searching function for user to choose.
  + 1. Functional Requirements

REQ01 : Users access to view newest products list.

REQ02 : Displaying categories, events for use to choose.

REQ03 : Searching products by name.

REQ04 : In searching case, user must put in name of searching product. System will filter data to show products that have name like the name user put in.

## All Product

* + 1. Description
* Home page will show all product with some information :
* Image
* Cake title
* Cake price
* Order button

## Guest, User can view product detail

* + 1. Description
* User can show product detail by click a link show in index or category page
* The detail page must provide information of product. (name, thumb image, description, price, comment and relation product)
* The user can show the Buy Now button for add product to cart
  + 1. Function Requirements

REQ05 : User must click a link in index

* 1. Product by event.
     1. Description
* System provides, classifies products by event. System provides some default events like: birthday, Christmas event...
* Event can be added in the managing page.
  + 1. Function Requirements

REQ06 : Admin must create events in the managing page.

REQ07 : When adding new product, admin must choose event for that product.

REQ08 : Event is able to be used to search products for user.

* 1. Admin can manager user
     1. Description
* Admin can manage customer after logging in managing page. At the managing page, admin can view customers list, change customer information, block or delete customer account.
  + 1. Function Requirements

REQ09 : The system must provide the managing page, there will provide customers list with managing functions.

REQ10 : In blocking case, account of customer will still be existing on system but customer can’t login and buy products...

REQ11 : In deleting case, account of customer will be deleted completely from system include personal information, trading history.

REQ12 : In changing information case, admin must put in all personal information of user and can’t change login name or email of customer.

REQ13 : Admin can’t change password of user.

* 1. Admin can manager event
     1. Description
* Admin can manage event with functions: adding, deleting and changing event .
* Only after logging in the managing page, admin is able to use those functions.
  + 1. Function Requirements

REQ14 : The system must provide the products category with functions: adding, deleting and changing event.

REQ15 : In deleting case, there will be a dialog box appearing for admin to confirm. In confirming deleting case, the deleted event will be gone with all the products belong to it.

REQ16 : In adding or changing case, admin must put in all fields.

* 1. Admin can manager order
     1. Description
* Admin can view, delete or change the status of orders.
* Only after logging in the managing page, admin is able to use those functions.
  + 1. Function Requirements

REQ17 : The system must provide the product orders list in newest arrangement and provide functions to manage orders.

REQ18 : In deleting case, there will be a dialog box appearing for admin to confirm. In confirming deleting case, the deleted order will be gone.

REQ19 : In changing status case, there will be those statuses

* 1. Admin can view system statitics
     1. Description
* Admin can view system statistics with details on the quantity of custermer buying in month, year; the quantity of new customer registration; the profits in month, year...
* Only after logging in the managing page, admin is able to use those functions.
  + 1. Function Requirements

REQ20: The system must provide system statistics page with those details:

* The quantity of new customer registration in month, year.
* The quantity of order in month, year.
* The trading money and profits in month, year.

# System Requirements

* 1. Hardware
     1. Web Server

|  |  |
| --- | --- |
| Processor | Intel I3, I5, I7 |
| Memory | 4GB RAM or greater. |
| Modem/ADSL | Connecting to internet 24/24 |

* + 1. Client

|  |  |
| --- | --- |
| Processor | Intel I3, I5, I7 |
| Memory | 4GB RAM or greater. |
| Monitor | Resolution : 1308x780 |
| Pointing Device | Microsoft Mouse, Microsoft IntelliMouse®, or compatible pointing device. |
| Modem/ADSL | Connecting to internet |

* 1. Software
     1. Web Server

|  |  |
| --- | --- |
| Operation System | Windows 8. Windows 8.1 |
| Browser | Internet Explorer 9.0 or higher. |
| Database | Microsoft SQL Server 2012 |
| Software | Java SDK 1.7 |
| Server | GlassFish 4 |
| Framework | JSF 2.0 |

* + 1. Client

|  |  |
| --- | --- |
| Operation System | Windows 8. Windows 8.1 |
| Browser | Internet Explorer 9.0 or higher. |

# Development Software

* Microsoft Windows 8, Window 8.1
* Netbean 7.4
* Java SDK 1.7
* GlassFish 4
* Microsoft SQL Server 2012
* Internet Explorer 9.0 or higher

# Techonology

* J2EE
* JSF Framework
* HTML 5
* CSS 3

# Scope of work

* Not managerment employee.
* Not support online payment.

# Task Sheet

|  |  |  |  |
| --- | --- | --- | --- |
| **No** | **Full Name** | **Task** | **Status** |
| **1** | **Tran Quang Tung** | Joint working | Complete |
| 2 | Dang Tran Quoc Thinh | Joint working | Complete |
| 3 | Nguyen Vo Thanh Huy | Joint working | Complete |

|  |  |
| --- | --- |
| Leader | Faculty |
|  |  |
|  |  |
| Tran Quang Tung | Duong Trong Phu Son |

**REVIEW 2**

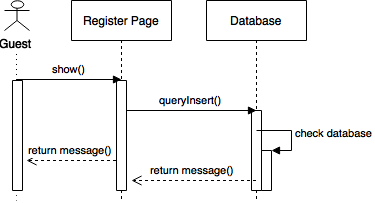
# Use Case and Sequence Diagram

## Guest:

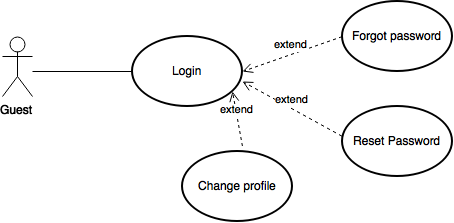
* + 1. Register:



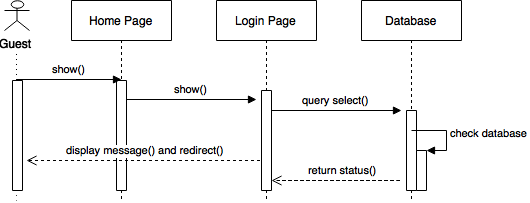
|  |  |  |
| --- | --- | --- |
| **Author** | **Tran Quang Tung** | |
| **Use Case Name** | Register | |
| **Actors** | Guest | |
| **Description** | Guest wants to register with the website to become a member | |
| **Requirements** | Guest provides Guest’s information | |
| **Pre-conditions** |  | |
| **Post-conditions** | Success: Guest creates a customer 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.  5. Actor inputs register’s information, click ‘Register’ button. [Alternative 1] | System Responses:  2. System redirects to Registration Policy page.  4. 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  - ‘Birthday’ datebox  - ‘Address’ text field  - ‘Phone’ text field  - ‘Email’ 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 ‘UserName’ text field blank.  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 ‘Birthday’ datebox blank.  8. Actor lefts ‘Address’ text field blank.  9. Actor lefts ‘Phone’ text field blank.  10. 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. | 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’.  - ‘Birthday is invalid’.  - ‘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’. |



* + 1. Login:

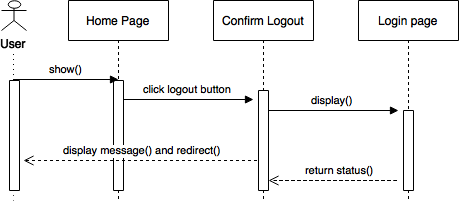


|  |  |  |
| --- | --- | --- |
| **Author** | **Tran Quang Tung** | |
| **Use Case Name** | Login | |
| **Actors** | Customer | |
| **Description** | Customer wants to login to website | |
| **Requirements** | Actor provides username and password | |
| **Pre-conditions** |  | |
| **Post-conditions** | 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 ‘OK’ 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] |

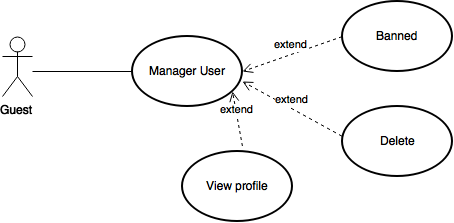


* + 1. Logout:

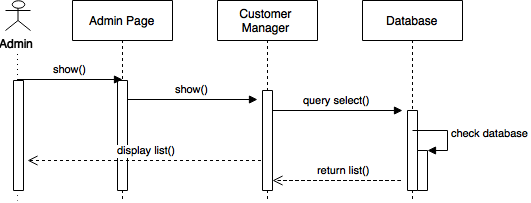
|  |  |  |
| --- | --- | --- |
| **Author** | Tran Quang Tung | |
| **Use Case Name** | Log out | |
| **Actors** | Customers | |
| **Description** | Customers wants to log out of the System. | |
| **Requirements** |  | |
| **Pre-conditions** | Actor has logged in. | |
| **Post-conditions** | Success: 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** |  |  |



* + 1. Manage Account:

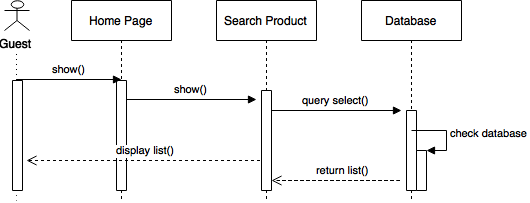


|  |  |  |
| --- | --- | --- |
| **Author** | **Tran Quang Tung** | |
| **Use Case Name** | Update Profile | |
| **Actors** | Customer | |
| **Description** | Customer wants to Update his/her profile’s information. | |
| **Requirements** | Actor provides profile information | |
| **Pre-conditions** | 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 Home page 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  - ‘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 ‘New Password’ text field blank.  2. Actor lefts ‘New Confirm Password’ text field blank.  3. New Password and New Password Confirm don’t match.  4. Actor lefts ‘Name’ text field blank.  5. Actor lefts ‘Birthday’ datebox 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 8 – 14 character.  10. Actor lefts ‘Email’ text field blank or input an invalid email address. | System Responses:  System displays message to Actor :  - ‘Password cannot be blank’’.  - ‘Confirm Password can’t be blank’.  - ‘Confirm password not match Password.’  - ‘Name cannot be blank’.  - ‘Birthday is invalid’.  - ‘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’. |



* + 1. Search Products:

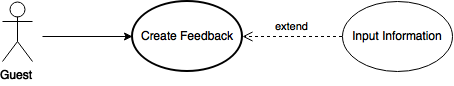
|  |  |  |
| --- | --- | --- |
| **Author** | **Tran Quang Tung** | |
| **Use Case Name** | Customer Searching | |
| **Actors** | Customer | |
| **Description** | Customer to perform a Property for Sale Searching. | |
| **Requirements** | Actor provides search information | |
| **Pre-conditions** | Actor has logged in as customer | |
| **Post-conditions** | Success: Properties match the condition show up.  Fail: Do not have any property match the condition. | |
| **Basic flow** | Actor Actions:  1. Actor clicks ‘Search’ button on the Actor Home page menu.  3. Actor inputs searching’s information and click ‘Search’ button’. [Alternative 1] | System Responses:  2. System show the Search contains the following controls:  -‘name’ textbox  -‘search for price’ dropbox  -‘search for events’ dropbox  4. System select the products in database that match searching condition and display the result in the Searching Result list below the products for Searching form. |
| **Alternative flow** | Actor Actions:  [Alternative 1]  1. Actor click ‘Back to Home’ button. | System Responses:  2. System returns to Actor Home page. |
| **Exception** |  |  |



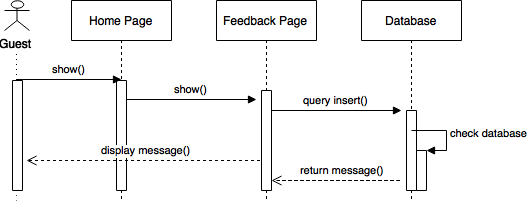
* + 1. Order products:

|  |  |  |
| --- | --- | --- |
| **Author** | Dang Tran Quoc Thinh | |
| **Use Case Name** | Order products | |
| **Actors** | Customer | |
| **Description** | Customer wants to order products on site | |
| **Requirements** | Actor provides Credit Card information. | |
| **Pre-conditions** | Actor has logged in. | |
| **Post-conditions** | Success: Customer order products sucessfully  Fail: Refill Information. | |
| **Basic flow** | Actor Actions:  1. Actor clicks ‘Order’ button on the Actor Home page menu.  3. Actor inputs Recipient name, recipient address, contact number, note, delivery date, recipient DOB and click ‘Order’ button. [Alternative 1]  6. Actor inputs Credit No, Credit Expired Month and Year, and click ‘Pay’ button. [Alternative 1] | System Responses:  2. System redirect to Buy Products and show the Order products form with the following controls:  - ‘Recipient name’ text field.  - ‘Recipient address’ text field.  - ‘Contact number’ text field.  -‘Note’ text field.  -‘Delivery date’ datebox.  -‘Recipient DOB’ datebox.  - ‘Buy’ button.  - ‘Back to Home’ button.  4. System checks the information  [Exception 1].  5. System create Orders and show Payment form.  - ‘Customer name’ text field.  - ‘Credit No’ text field.  - ‘Credit Expired Month’ combobox.  - ‘Credit Expired Year’ combobox.  - ‘Pay’ button.  - ‘Back to Home’ button.  7. System checks the information  [Exception 2].  8. System checks payment information and show Success form. |
| **Alternative flow** | Actor Actions:  [Alternative 1]  1. Actor click ‘Back to Home’ button. | System Responses:  2. System returns to Actor Home page. |
| **Exception** | Actor Actions:  [Exception 1]  1. Actor lefts ‘Recipient name’ text field blank.  2. Actor lefts ‘Recipient Address’ text field blank.  3. Actor lefts ‘Contact Number’ text field blank.  4. Actor lefts ‘Note’ text field blank.  5. Actor lefts ‘Delivery Date’ datebox blank.  6. Actor lefts ‘Recipient DOB’ datebox blank.  [Exception 2]  1. Credit No is invalid.  2. Input credit expired month less than current month. | System Responses:  System displays message to Actor :  - ‘Recipient name cannot be blank’’.  - ‘Recipient Address can’t be blank’.  - ‘Contact Number’ can’t be blank’.  - ‘Note cannot be blank’.  - ‘Delivery Date is invalid’.  - ‘Repicient DOB is invalid’.  - System shows the message: ‘Invalid Credit No’.  - System shows the message: ’Credit Expired Date is invalid’. |

* + 1. Create Feedback:

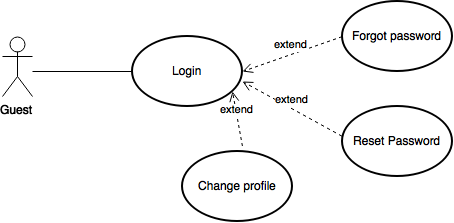


|  |  |  |
| --- | --- | --- |
| **Author** | Tran Quang tung | |
| **Use Case Name** | Customer Feedback Creating | |
| **Actors** | Customer | |
| **Description** | Customer to create a feedback | |
| **Requirements** | Actor provides feedback content | |
| **Pre-conditions** | Actor has logged in as customer | |
| **Post-conditions** | Success: Customer fill the textbox with content  Fail: Customer leave textbox blank | |
| **Basic flow** | Actor Actions:  1. Actor clicks ‘Feedback’ button on the Actor Home page menu.  3. Actor inputs feeback’s information and click ‘Send’ button’. [Alternative 1] | System Responses:  2. System show the feedback contains the following controls:  -‘content’ textbox  4. System checks information  [Exception 1] |
| **Alternative flow** | Actor Actions:  [Alternative 1]  1. Actor click ‘Back to Home’ button. | System Responses:  2. System returns to Actor Home page. |
| **Exception** | Actor Actions:  [Exception 1]  1. Actor lefts ‘content’ text field blank. | System Responses:  System displays message to Actor :  - ‘Content cannot be blank’’. |

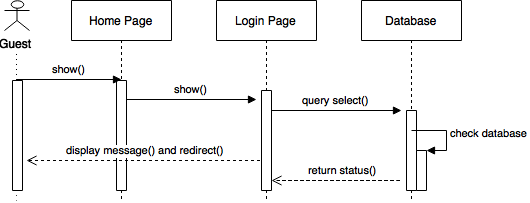


## ADMIN

* + 1. Login

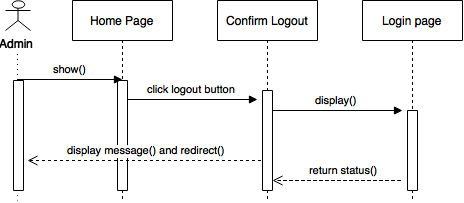


|  |  |  |
| --- | --- | --- |
| **Author** | Tran Quang Tung | |
| **Use Case Name** | Login | |
| **Actors** | ADMIN | |
| **Description** | ADMIN wants to login to website | |
| **Requirements** | Actor provides username and password | |
| **Pre-conditions** |  | |
| **Post-conditions** | Success: 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 ‘OK’ 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] |

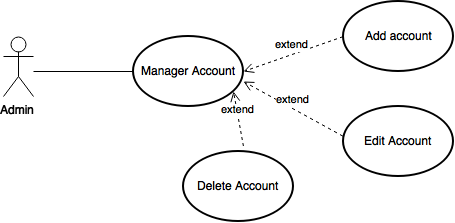


* + 1. Logout:

|  |  |  |
| --- | --- | --- |
| **Author** | Tran Quang Tung | |
| **Use Case Name** | Log out | |
| **Actors** | ADMIIN | |
| **Description** | ADMIN wants to log out of the System. | |
| **Requirements** |  | |
| **Pre-conditions** | Actor has logged in as ADMIN | |
| **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** |  |  |

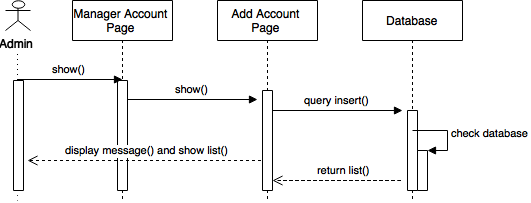


* + 1. Manage Accounts:



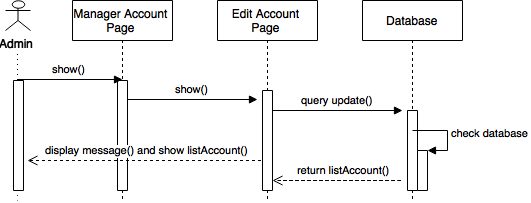
* + - 1. Add Accounts:

|  |  |  |
| --- | --- | --- |
| **Author** | Tran Quang Tung | |
| **Use Case Name** | Adding account | |
| **Actors** | ADMIN | |
| **Description** | ADMIN wants to add account | |
| **Requirements** | Actor provides profile information | |
| **Pre-conditions** | Actor has logged in as ADMIN | |
| **Post-conditions** | Success: a new account has been added!  Fail: Refill information | |
| **Basic flow** | Actor Actions:  1. Actor clicks ‘manage accounts’ button in Actor’s Home page menu.  3. Actor clicks on ‘Add new account’ button.  3. Actor updates information and click ‘Create account’ button. [Alternative 1] | System Responses:  2.System redirects to Manage Accounts page and show the list of customer with add, delete, update or order history button.  4. System redirects to Add new account page and displays add new account form with the following control :  - ‘UserName’ text field  - ‘Name’ text field  - ‘Birthday’ text field  - ‘Address’ text field  - ‘Phone’ text field  - ‘Email’ text field  - ‘Create account’ button  - ‘Back to Home’ button  4. System checks the information  [Exception 1]  5. System create 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.  4. Actor lefts ‘Name’ text field blank.  5. Actor lefts ‘Birthday’ datebox 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 8 – 14 character.  10. Actor lefts ‘Email’ text field blank or input an invalid email address. | System Responses:  System displays message to Actor :  - ‘Password cannot be blank’’.  - ‘Confirm Password can’t be blank’.  - ‘Confirm password not match Password.’  - ‘Name cannot be blank’.  - ‘Birthday is invalid’.  - ‘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’. |



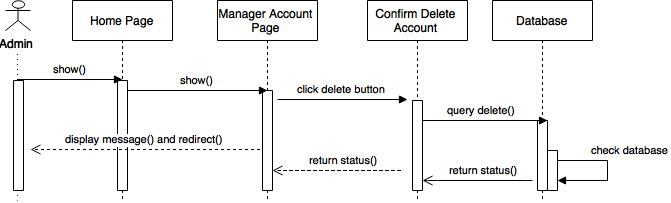
* + - 1. Update Accounts:

|  |  |  |
| --- | --- | --- |
| **Author** | Tran Quang Tung | |
| **Use Case Name** | Update account | |
| **Actors** | ADMIN | |
| **Description** | ADMIN wants to update accounts | |
| **Requirements** | Actor provides profile information | |
| **Pre-conditions** | Actor has logged in as ADMIN | |
| **Post-conditions** | Success: account’s information has been updated!  Fail: Refill information | |
| **Basic flow** | Actor Actions:  1. Actor clicks ‘manage accounts’ button in Actor’s Home page menu.  3. Actor clicks on ‘Update’ button.  3. Actor updates information and click ‘Save’ button. [Alternative 1] | System Responses:  2.System redirects to Manage Accounts page and show the list of customer with add, delete, update or order history button.  4. System redirects to Update account and displays Account Details form with the following control :  - ‘UserName’ text field (read only)  - ‘Name’ text field  - ‘Birthday’ text field  - ‘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 ‘New Password’ text field blank.  2. Actor lefts ‘New Confirm Password’ text field blank.  3. New Password and New Password Confirm don’t match.  4. Actor lefts ‘Name’ text field blank.  5. Actor lefts ‘Birthday’ datebox 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 8 – 14 character.  10. Actor lefts ‘Email’ text field blank or input an invalid email address. | System Responses:  System displays message to Actor :  - ‘Password cannot be blank’’.  - ‘Confirm Password can’t be blank’.  - ‘Confirm password not match Password.’  - ‘Name cannot be blank’.  - ‘Birthday is invalid’.  - ‘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’. |



* + - 1. Delete Accounts:

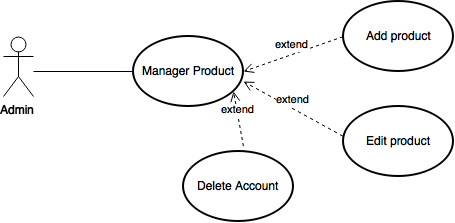
|  |  |  |
| --- | --- | --- |
| **Author** | Tran Quang Tung | |
| **Use Case Name** | Delete Account | |
| **Actors** | Admin | |
| **Description** | Admin wants to delete accounts | |
| **Requirements** |  | |
| **Pre-conditions** | Actor has logged in as Admin | |
| **Post-conditions** | Success: Account has been deleted | |
| **Basic flow** | Actor Actions:  1. Actor clicks ‘manage accounts’ button in Actor’s Home page menu.  3. Actor clicks on ‘Delete’ button on the shortcut he/she wish to delete.  5. Actor click ‘Yes’ button. [Alternative 1] | System Responses:  2.System redirects to Manage Accounts page and show the list of customer with add, delete, update or order history button.  4. System show the Delete Shortcut Confirm form with the following control :  - ‘Do you want to delete this account?’ label.  - ‘Yes’ button.  - ‘No’ button.  6. System deletes the selected account from database and return to Account list page. |
| **Alternative flow** | Actor Actions:  [Alternative 1]  1. Actor click ‘No’ button. | System Responses:  2. System returns to Account List page. |
| **Exception** |  |  |



* + - 1. Tracking Purchasing History:

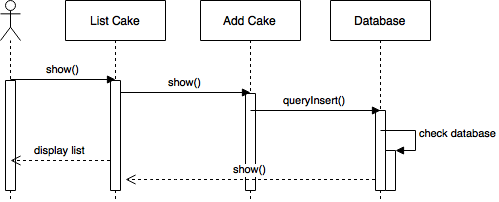
|  |  |  |
| --- | --- | --- |
| **Author** | Tran Quang Tung | |
| **Use Case Name** | Tracking Purchasing History | |
| **Actors** | Admin | |
| **Description** | Admin wants to track purchasing history of customer. | |
| **Requirements** |  | |
| **Pre-conditions** | Actor has logged in as Admin | |
| **Post-conditions** |  | |
| **Basic flow** | Actor Actions:  1. Actor clicks ‘manage accounts’ button in Actor’s Home page menu.  3. Actor clicks on ‘Order history’ button on the shortcut he/she wish to delete.  5. Actor click on ‘Order ID’ textbox. | System Responses:  2.System redirects to Manage Accounts page and show the list of customer with add, delete, update or order history button.  4. System show the Order list of customer with Order ID and Status.  6. System show the Order details of that product |
| **Alternative flow** |  |  |
| **Exception** |  |  |

* + 1. Manage product:



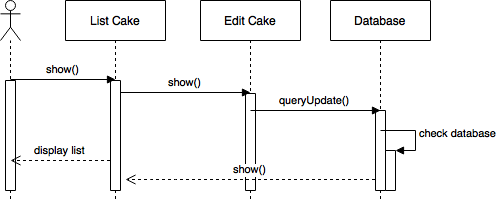
* + - 1. Add Product:

|  |  |  |
| --- | --- | --- |
| **Author** |  | |
| **Use Case Name** | Adding product | |
| **Actors** | ADMIN | |
| **Description** | ADMIN wants to add product | |
| **Requirements** | Actor provides profile information | |
| **Pre-conditions** | Actor has logged in as ADMIN | |
| **Post-conditions** | Success: a new product has been added!  Fail: Refill information | |
| **Basic flow** | Actor Actions:  1. Actor clicks ‘manage products’ button in Actor’s Home page menu.  3. Actor clicks on ‘Add new product’ button.  3. Actor updates information and click ‘Add’ button. [Alternative 1] | System Responses:  2.System redirects to Manage Products page and show the list of product with add, delete, update or order history button.  4. System redirects to Add new product page and displays add new product form with the following control :  - ‘CakeName’ text field  - ‘CakeDetails’ text field  - ‘CakePrice’ number  - ‘CakeImage’ image  - ‘Category’ dropbox  - ‘IsEgg’ int  - ‘Status’ textfield.  - ‘EventID’ dropbox.  - ‘Add’ button.  4. System checks the information  [Exception 1]  5. System create the product 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 ‘CakeName’ text field blank.  2. Actor lefts ‘CakeDetails’ text field blank.  3. Actor lefts ‘CakePrice’ number blank.  4. Actor lefts ‘CakeImage’ image blank. | System Responses:  System displays message to Actor :  - ‘CakeName’ cannot be blank’’.  - ‘CakeDetails can’t be blank’.  - ‘CakePrice cannot be blank’.  - ‘CakeImage is missing’. |



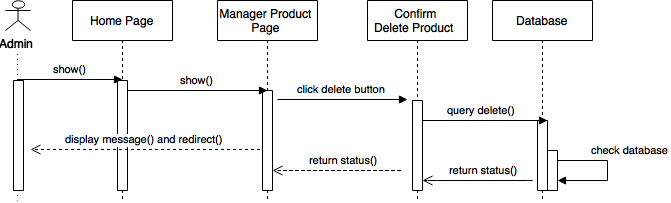
* + - 1. Update Product:

|  |  |  |
| --- | --- | --- |
| **Author** | Nguyen Vo Thanh Huy | |
| **Use Case Name** | Update Product | |
| **Actors** | ADMIN | |
| **Description** | ADMIN wants to update Product | |
| **Requirements** | Actor provides profile information | |
| **Pre-conditions** | Actor has logged in as ADMIN | |
| **Post-conditions** | Success: product’s information has been updated!  Fail: Refill information | |
| **Basic flow** | Actor Actions:  1. Actor clicks ‘manage products’ button in Actor’s Home page menu.  3. Actor clicks on ‘Update’ button.  3. Actor updates information and click ‘Save’ button. [Alternative 1] | System Responses:  2.System redirects to Manage Products page and show the list of customer with add, delete, update or order history button.  4. System redirects to Update product and displays Product Details form with the following control :  - ‘CakeName’ text field  - ‘CakeDetails’ text field  - ‘CakePrice’ number  - ‘CakeImage’ image  - ‘Category’ dropbox  - ‘IsEgg’ int  - ‘Status’ textfield.  - ‘EventID’ dropbox.  - ‘Save’ button  - ‘Back to Home’ button  4. System checks the information  [Exception 1]  5. System update the product 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]  [Exception 1]  1. Actor lefts ‘CakeName’ text field blank.  2. Actor lefts ‘CakeDetails’ text field blank.  3. Actor lefts ‘CakePrice’ number blank.  4. Actor lefts ‘CakeImage’ image blank. | System Responses:  System displays message to Actor :  - ‘CakeName’ cannot be blank’’.  - ‘CakeDetails can’t be blank’.  - ‘CakePrice cannot be blank’.  - ‘CakeImage is missing’. |



* + - 1. Delete Products:

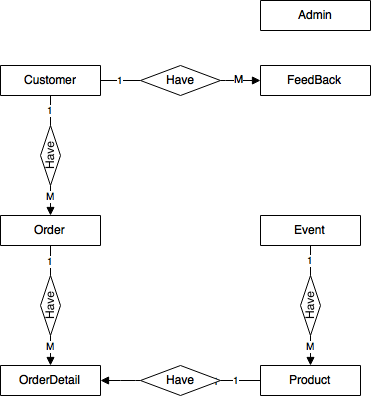
|  |  |  |
| --- | --- | --- |
| **Author** | Nguyen Vo Thanh Huy | |
| **Use Case Name** | Delete Product | |
| **Actors** | Admin | |
| **Description** | Admin wants to delete products | |
| **Requirements** |  | |
| **Pre-conditions** | Actor has logged in as Admin | |
| **Post-conditions** | Success: Product has been deleted | |
| **Basic flow** | Actor Actions:  1. Actor clicks ‘manage products’ button in Actor’s Home page menu.  3. Actor clicks on ‘Delete’ button on the shortcut he/she wish to delete.  5. Actor click ‘Yes’ button. [Alternative 1] | System Responses:  2.System redirects to Manage Products page and show the list of customer with add, delete, update or order history button.  4. System show the Delete Shortcut Confirm form with the following control :  - ‘Do you want to delete this product?’ label.  - ‘Yes’ button.  - ‘No’ button.  6. System deletes the selected product from database and return to Product list page. |
| **Alternative flow** | Actor Actions:  [Alternative 1]  1. Actor click ‘No’ button. | System Responses:  2. System returns to Product List page. |
| **Exception** |  |  |



# Entity Relationship

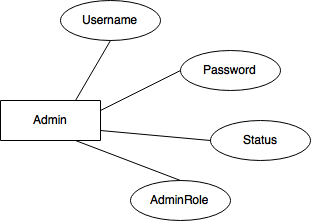


## ERD

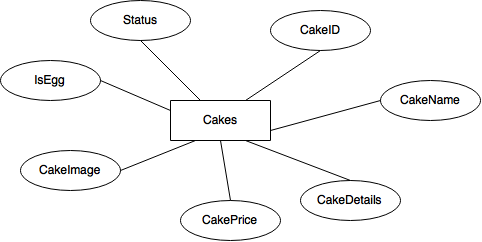


## Entity

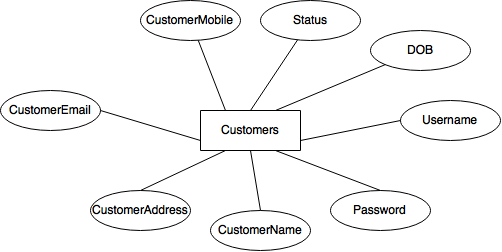
* + 1. Admin



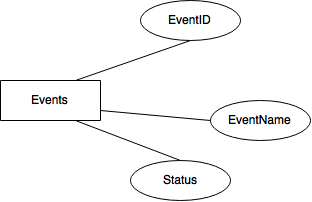
* + 1. Cakes



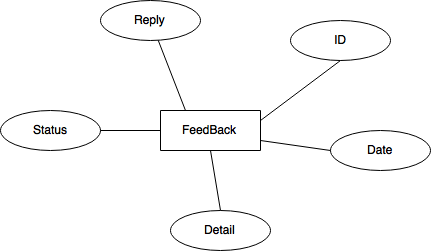
* + 1. Customers



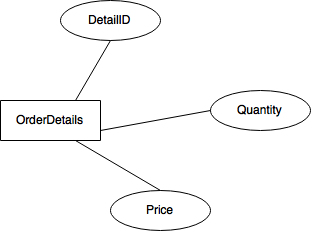
* + 1. Events



* + 1. FeedBack



* + 1. OrderDetails



* + 1. Orders

# C:\Users\nguyencanh\Desktop\EntityDiagram\Tung 2\Orders.png

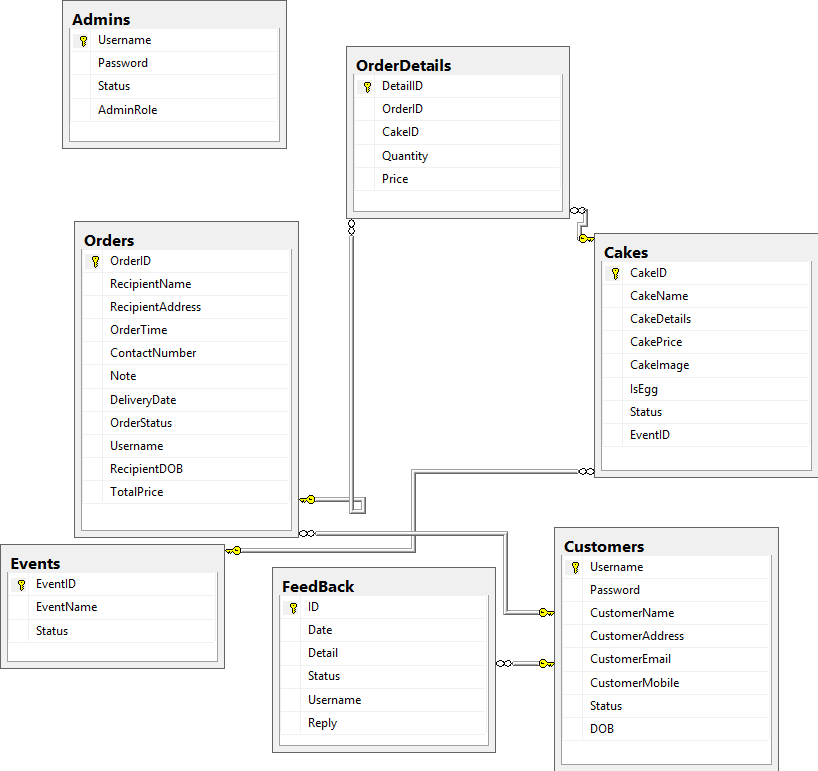
# Task Sheet

|  |  |  |  |
| --- | --- | --- | --- |
| **No** | **Full Name** | **Task** | **Status** |
| **1** | **Tran Quang Tung** | Joint working | Complete |
| 2 | Dang Tran Quoc Thinh | Joint working | Complete |
| 3 | Nguyen Vo Thanh Huy | Joint working | Complete |

|  |  |
| --- | --- |
| Leader | Faculty |
|  |  |
|  |  |
| Tran Quang Tung | Duong Trong Phu Son |

**REVIEW 3**

# Database Relationship Diagram



# Table Definition

## Admins

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Key** | **Field Name** | **Data Type** | **Length** | **Allow Null** | **Default value** | **Table-Field-Ref** | **Description** |
| PK | **Username** | nvarchar | 20 | No |  |  | User Name |
|  | Password | nvarchar | 50 | No |  |  | Password of Admin |
|  | Status | smallint |  | No |  |  | status of Admin |
|  | AdminRole | smallint |  | No |  |  | Role of Admin |

* Contrainst:

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

## Cakes

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Key** | **Field Name** | **Data Type** | **Length** | **Allow Null** | **Default value** | **Table-Field-Ref** | **Description** |
| PK | **CakeID** | int |  | No |  |  | Cake ID, FK of OrderDetail table |
|  | CakeName | nvarchar | 200 | No |  |  | Name of cake |
|  | CakeDetails | ntext |  | No |  |  | Cake Details |
|  | CakePrice | money |  | No |  |  | Cake Price |
|  | CakeImage | nvarchar | 100 | No |  |  | Cake Image |
|  | IsEgg | bit |  | no |  |  | Has Egg |
|  | Status | int |  | no |  |  | Status of cake |
| FK | EventID | int | 10 | no |  | Events.EventID | Event attach with cake |

### Contrainst:

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

## Customers

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Key** | **Field Name** | **Data Type** | **Length** | **Allow Null** | **Default value** | **Table-Field-Ref** | **Description** |
| PK | **Username** | nvarchar | 50 | No |  |  | Username of customer, FK of Order,Feedback |
|  | Password | nvarchar | 50 | No |  |  | Password of customer |
|  | CustomerName | nvarchar |  | No |  |  | Customer Name of customer |
|  | CustomerAddress | nvarchar |  | No |  |  | Customer Address of customer |
|  | CustomerEmail | nvarchar | 100 | No |  |  | Customer Email of customer |
|  | CustomerMobile | nvarchar |  | No |  |  | Customer Mobile of customer |
|  | Status | smallint |  | no |  |  | Status of customer |
|  | DOB | datetime | 10 | no |  |  | Birthdate of customer |

### Contrainst:

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

## Events

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Key** | **Field Name** | **Data Type** | **Length** | **Allow Null** | **Default value** | **Table-Field-Ref** | **Description** |
| PK | **EventID** | int |  | No |  |  | EventID of events, FK of cake |
|  | EventName | nvarchar | 50 | No |  |  | Event Name |
|  | Status | smallint |  | No |  |  | Status of event |

**Contrainst:**

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

## Feedback

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Key** | **Field Name** | **Data Type** | **Length** | **Allow Null** | **Default value** | **Table-Field-Ref** | **Description** |
| PK | **ID** | int |  | No |  |  | ID of Feeedback |
|  | Date | datetime |  | No |  |  | Date of Feeedback |
|  | Detail | ntext |  | No |  |  | Detail of Feeedback |
|  | Status | smallint |  | No |  |  | Status of Feeedback |
|  | Username | nvarchar | 50 | No |  | Customers. Username | Username Feeedback |
|  | Reply | ntext |  | No |  |  | Reply of Feeedback |

**Contrainst:**

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

## Order Detail

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Key** | **Field Name** | **Data Type** | **Length** | **Allow Null** | **Default value** | **Table-Field-Ref** | **Description** |
| PK | **DetailID** | int |  | No |  |  | ID of OrderDetail |
|  | OrderID | int |  | No |  | Order.OrderID | Order of OrderDetail |
|  | CakeID | int |  | No |  | Cake.CakeID | Cake of OrderDetail |
|  | Quantity | int |  | No |  |  | Quantity of OrderDetail |
|  | Price | int |  | No |  |  | Price of OrderDetail |

**Contrainst:**

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

## Order

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Key** | | **Field Name** | | **Data Type** | | **Length** | | **Allow Null** | | **Default value** | | **Table-Field-Ref** | | **Description** | |
| PK | | **OrderID** | | int | |  | | No | |  | |  | | ID of OrderDetail, IDENTITY(1,1) ,FK to OrderDetails Table | |
|  | | RecipientName | | nvarchar | | 50 | | No | |  | |  | | Recipient Name | |
|  | | RecipientAddress | | nvarchar | | 200 | | No | |  | |  | | Recipient Address | |
|  | | OrderTime | | datetime | |  | | No | |  | |  | | Order Time | |
|  | | ContactNumber | | varchar | | 11 | | No | |  | |  | | Contact Number | |
|  | | Note | | ntext | |  | | No | |  | |  | | Note of Order | |
|  | | DeliveryDate | | datetime | |  | | No | |  | |  | | Delivery Date of Order | |
|  | | OrderStatus | | smallint | |  | | No | |  | |  | | Status of Order | |
|  | | Username | | Nvarchar | | 50 | | No | |  | | Customers.Username | | Username Created Order | |
|  | | RecipientDOB | | datetime | |  | | No | |  | |  | | Birthday of Reciever | |
|  | | TotalPrice | | money | |  | | No | |  | |  | | Total Price of Order | |

**Contrainst:**

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

# Sitemap

## Admin

Login

Admin Home

Delete Cake

Add Cake

Update Cake

List Cake

Delete User

User List

List Order

Order Detail

Order Delete

List FeedBack

FeedBack Detail

FeedBack Delete

List Event

Add Event

Update Event

Delete Event



## User

Home Page

Login

Register

Cake Detail

My Account

Profile

Change Profile

Change Pass

Search

Event

FeedBack

About us

Checkout

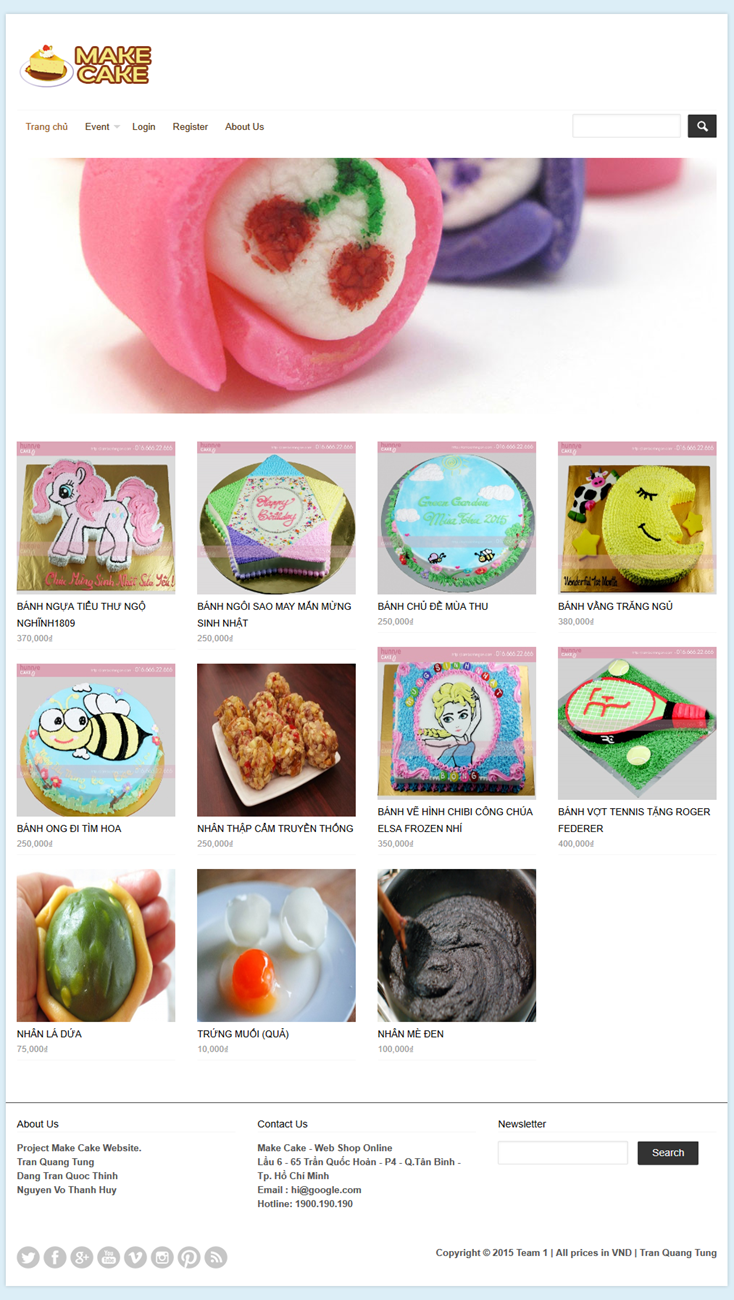
Cart

Logout

# GUI Design

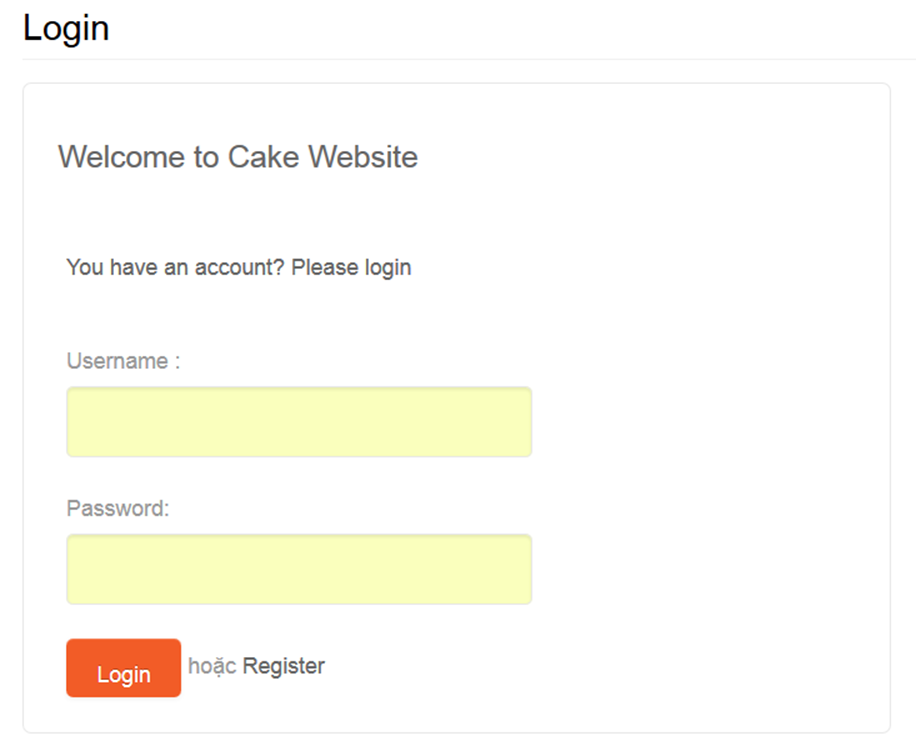
## Home Site

### Home page



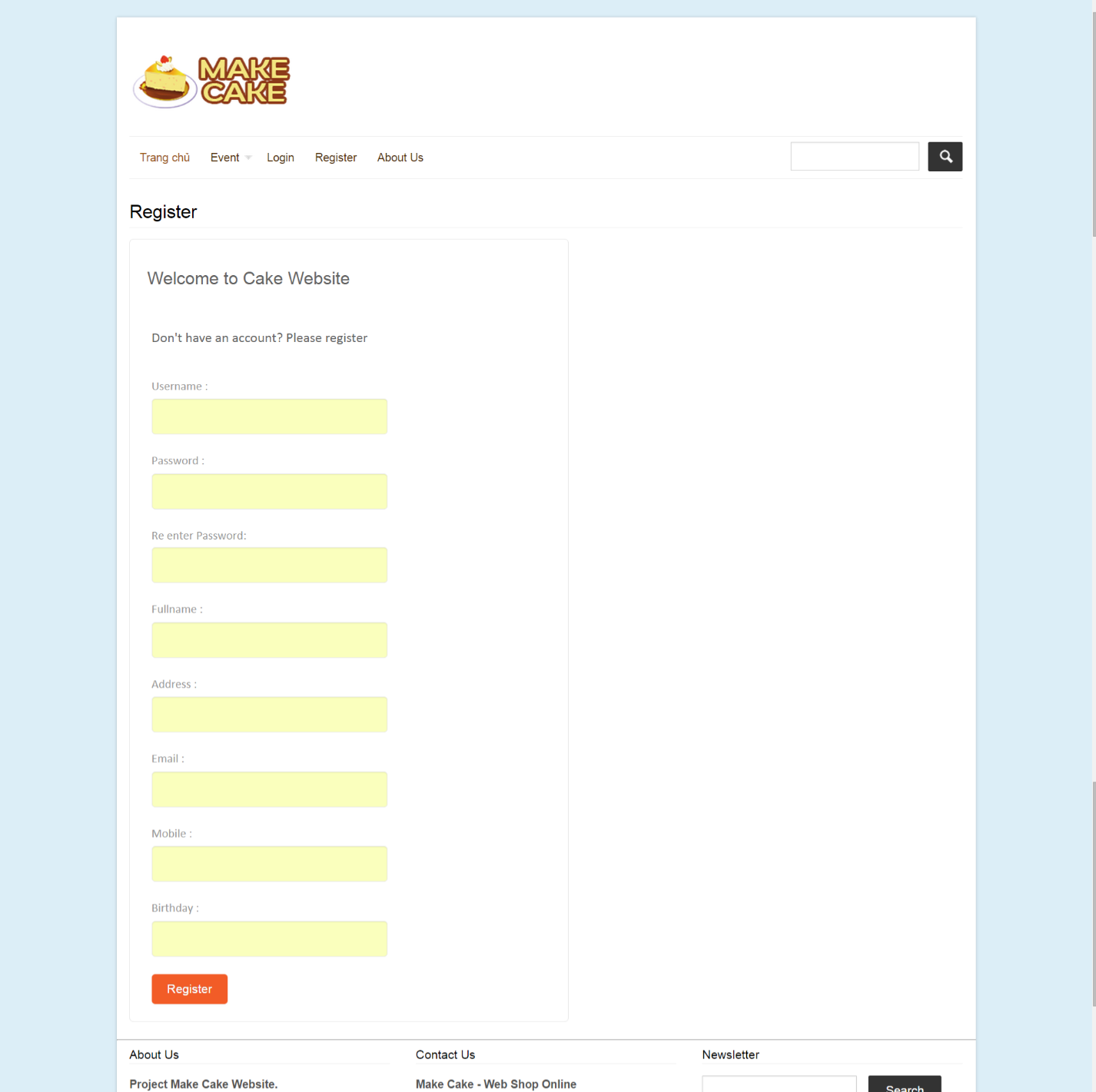
|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **No.** | **Name** | **Type** | **Validation** | **Event** | **Description** | **Status** |
| 1 | btnHome | Button |  | Click | Redirect to Home page |  |
| 2 | slEvent | DropDown |  | Click | Redirect to Event Site |  |
| 3 | login | Link |  | Click | Redirect to Login page |  |
| 4 | Register | Link |  | Click | Redirect to Register page |  |
| 5 | About Us | Link |  | Click | Redirect to About Us page |  |
| 6 | btnSearch | Button |  | Click | Search Product by Name |  |
| 7 | txtName | Text field |  |  | Input Product Name to search |  |

### Login Page



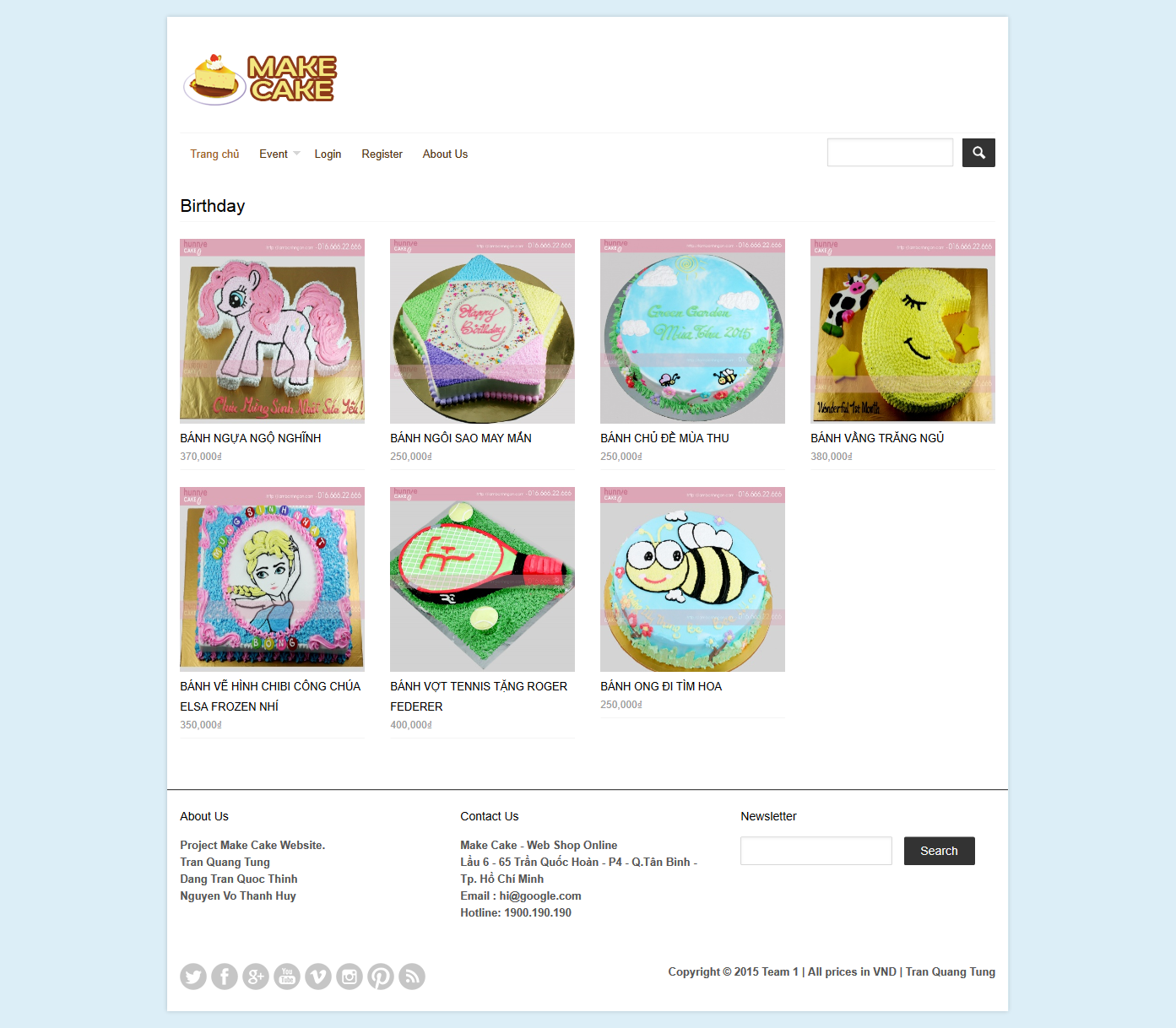
|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **No.** | **Name** | **Type** | **Validation** | **Event** | **Description** | **Status** |
| 1 | txtUsername | Text Field | Not Null |  | User name |  |
| 2 | txtPassword | Text Field | Not Null |  | Password |  |
| 3 | btnLogin | Button |  | Click | Login to website and redirect to home page |  |
| 4 | Register | Link |  | Click | Redirect to Register page |  |

### Register page



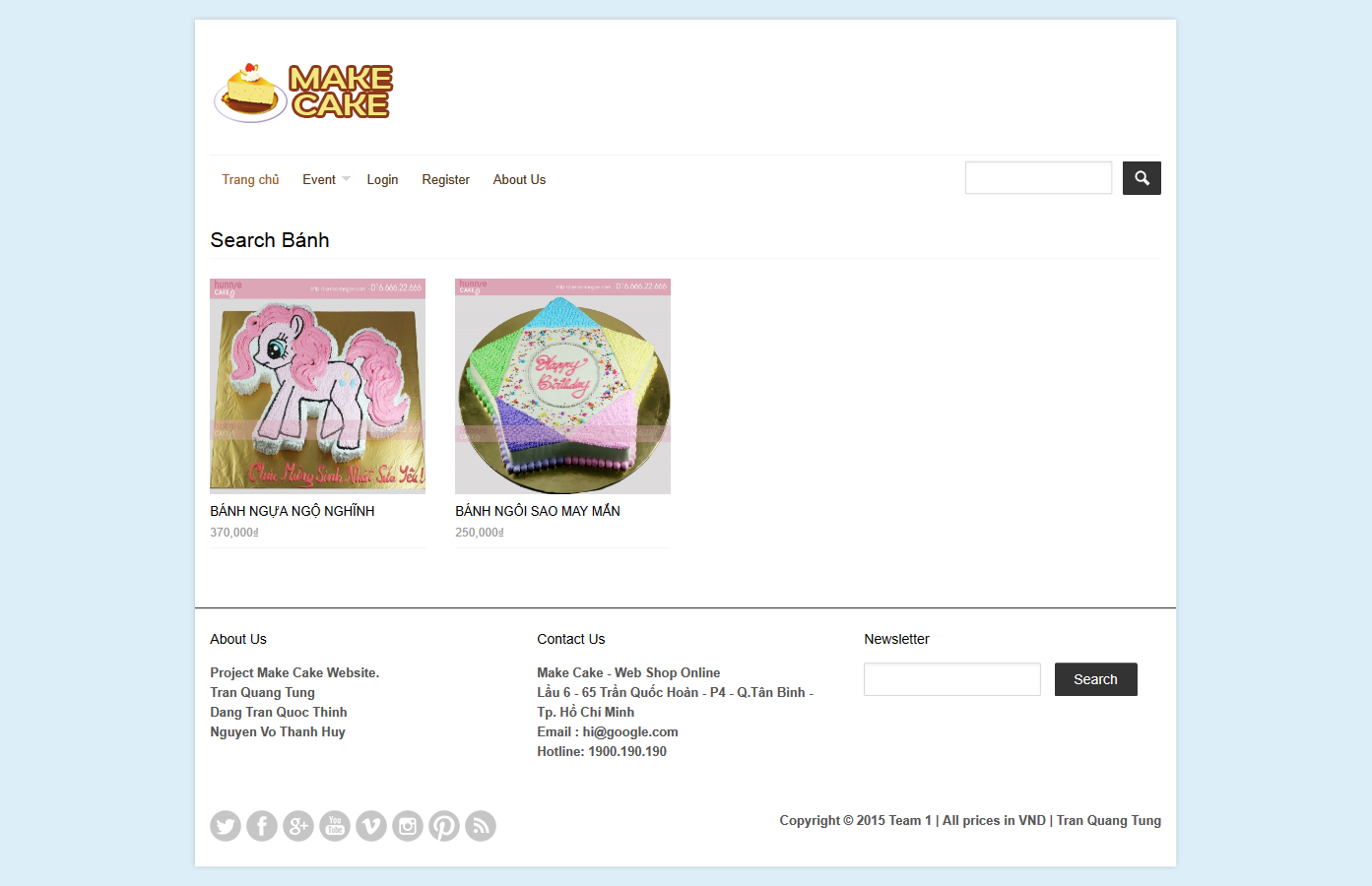
|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **No.** | **Name** | **Type** | **Validation** | **Event** | **Description** | **Status** |
| 1 | txtUsername | Text Field | Not Null |  | User name |  |
| 2 | txtPassword | Text Field | Not Null |  | Password |  |
| 3 | txtRePass | Text Field | Not Null, equal txtPassword |  | Re enter password |  |
| 4 | txtFullname | Text Field | Not Null |  | Fullname |  |
| 5 | txtAddress | Text Field | Not Null |  | Address |  |
| 6 | txtMail | Text Field | Not Null and Email type |  | Email |  |
| 6 | txtMobile | Text Field | Not Null, 10 – 11 Number |  | Phone |  |
| 7 | txtBirthDay | Text Field | Not Null, Date Type |  | Birth Day |  |
| 8 | btnRegister | Button |  | Click | Register to system and redirect to index page |  |

### Event page



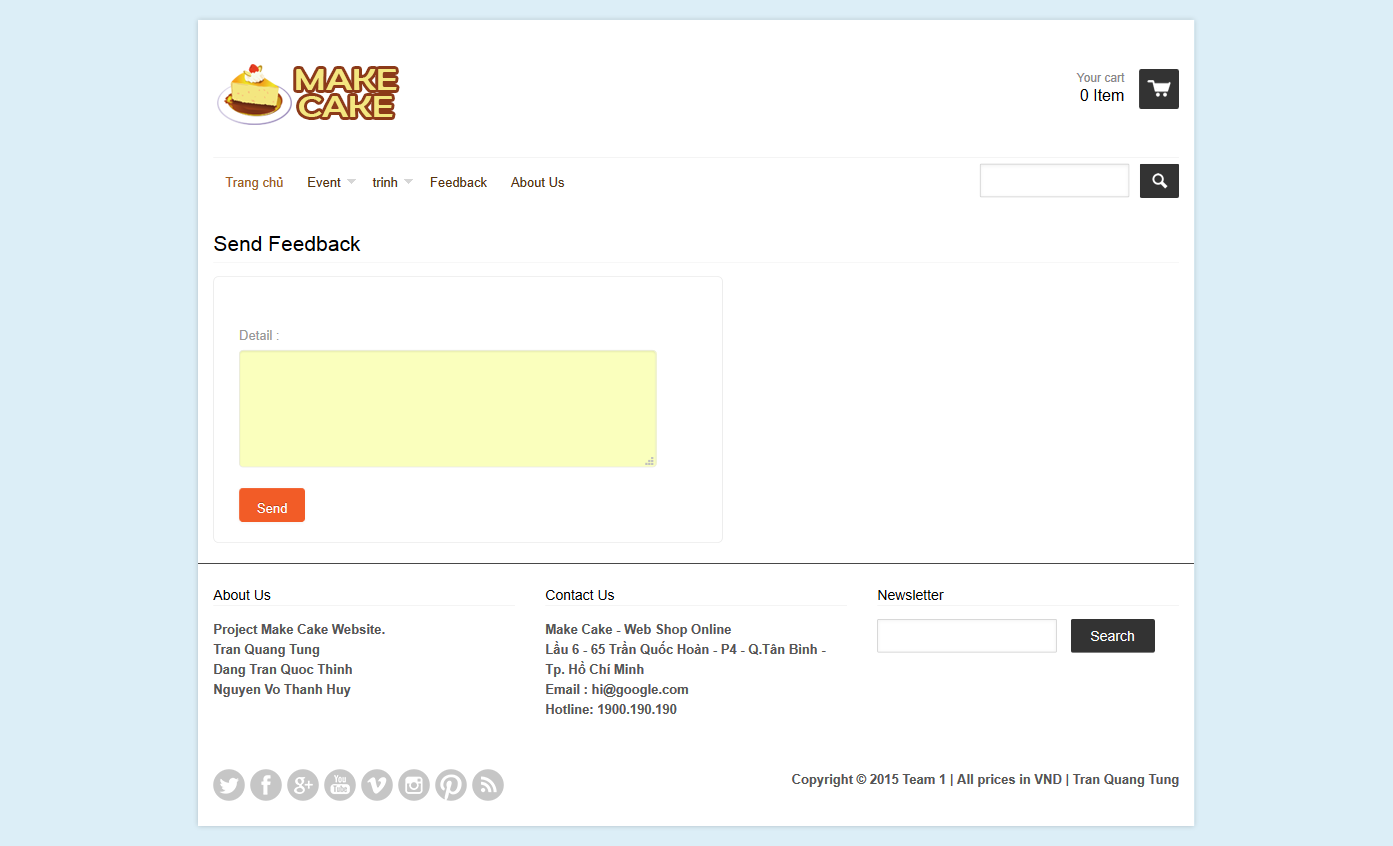
|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **No.** | **Name** | **Type** | **Validation** | **Event** | **Description** | **Status** |
| 1 | lbEvent | Label |  |  | Event name |  |
| 2 | Detail | Link |  |  | Link to detail product page |  |
| 3 | lstProduct | List |  |  | List product in event |  |

### Search Page



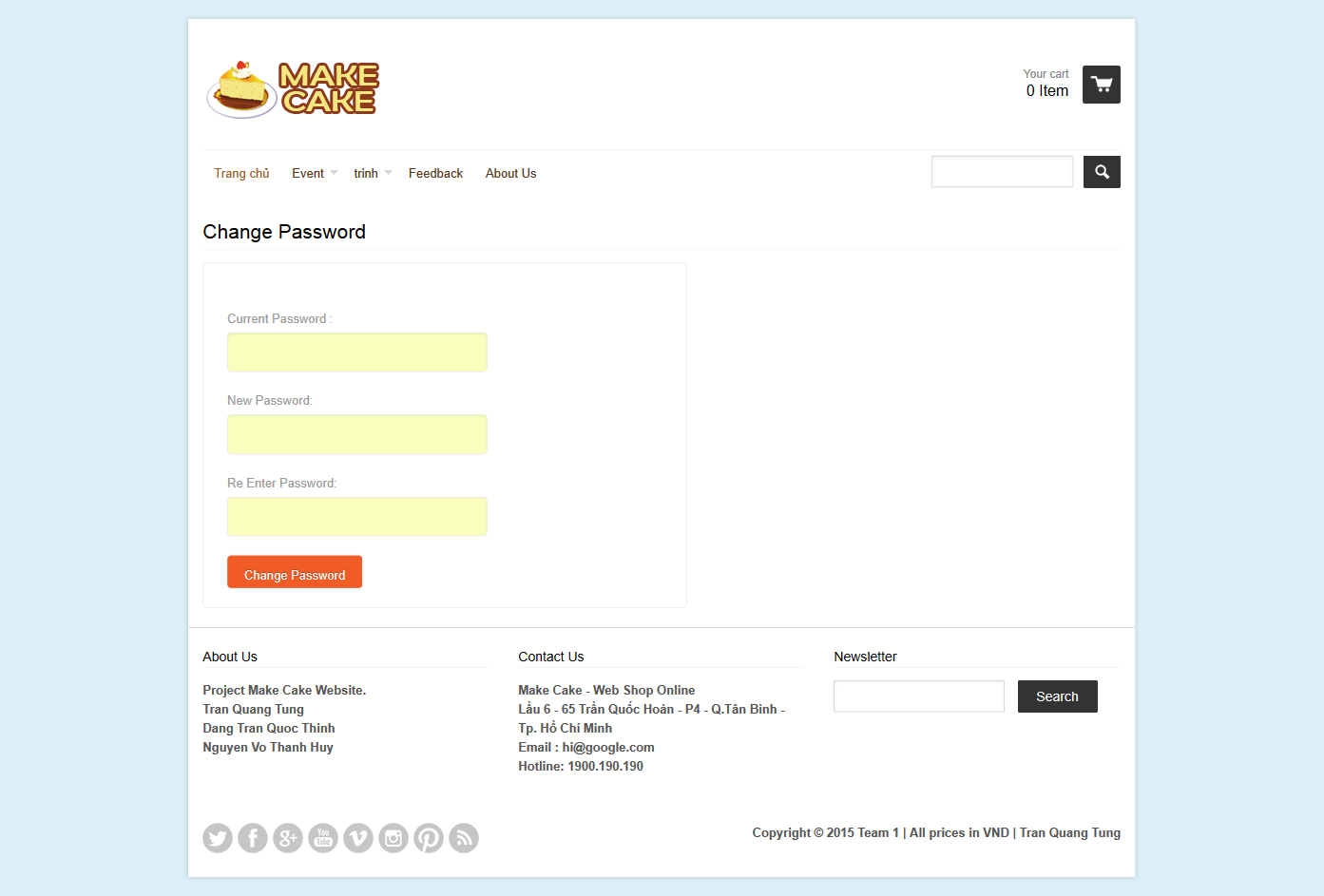
|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **No.** | **Name** | **Type** | **Validation** | **Event** | **Description** | **Status** |
| 1 | lbSearch | Label |  |  | Search name |  |
| 2 | Detail | Link |  |  | Link to detail product page |  |
| 3 | lstProduct | List |  |  | List product in search |  |

### Send Feedback



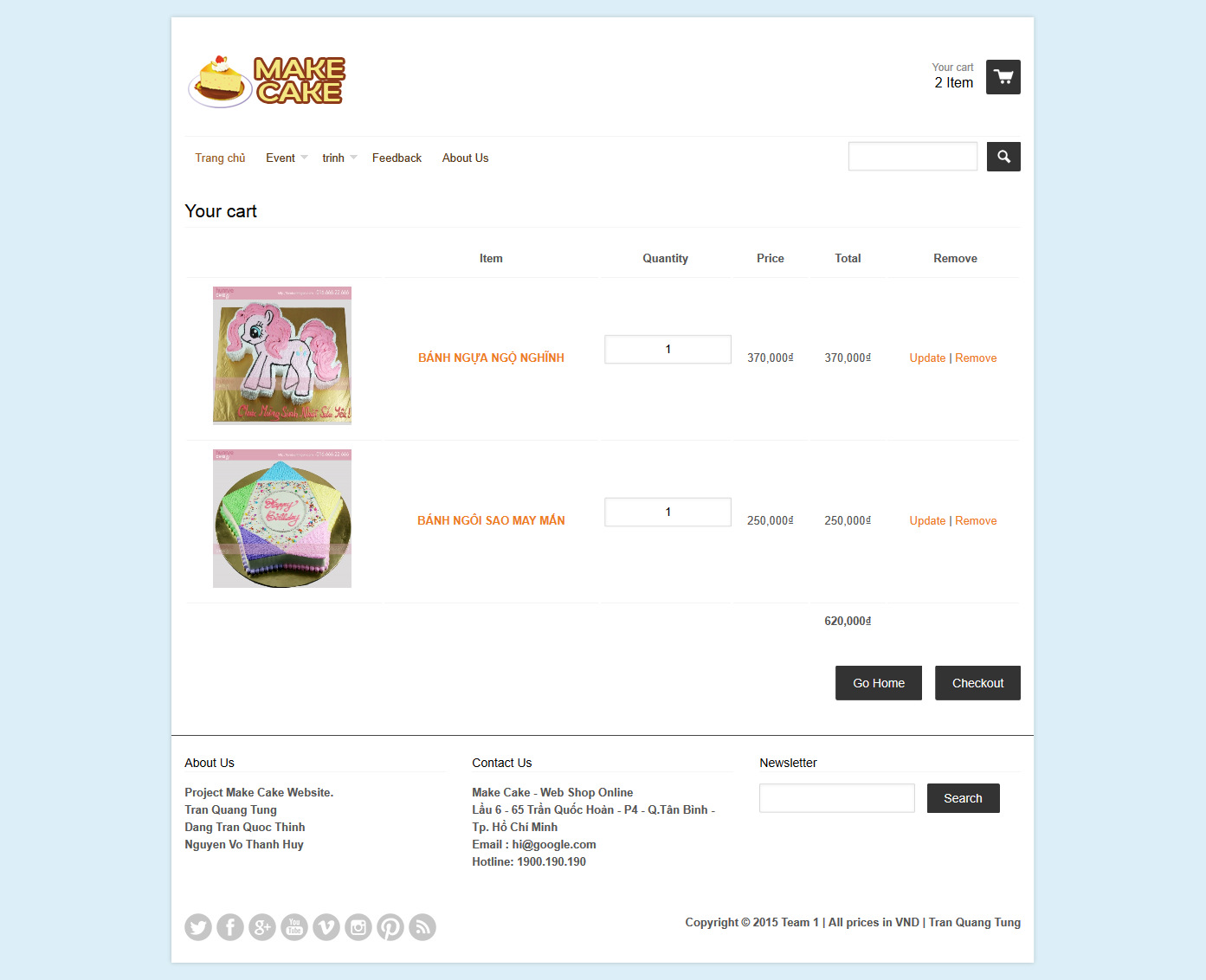
|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **No.** | **Name** | **Type** | **Validation** | **Event** | **Description** | **Status** |
| 1 | txtDetail | Text Field | Not Null |  | Detail |  |
| 2 | btnSend | Button |  | Click | Send feedback to admin and display message |  |

### Change Password



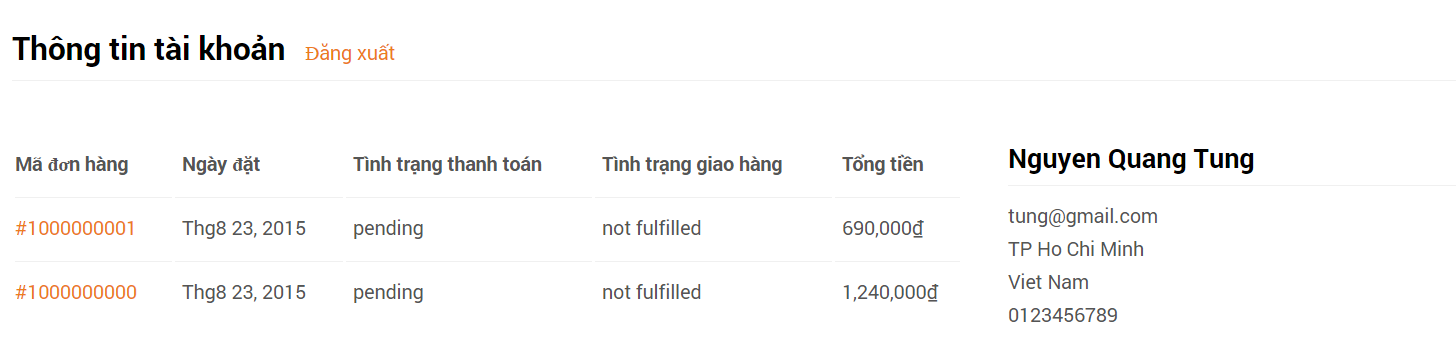
|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **No.** | **Name** | **Type** | **Validation** | **Event** | **Description** | **Status** |
| 1 | lbTitle | Label |  |  | Label of page |  |
| 2 | txtCurrent | Text Field | Not Null |  | Current Password |  |
| 3 | txtNewPass | Text Field | Not Null |  | New Password |  |
| 4 | txtRePass | Text Field | Not Null And Equal New Password |  | Re enter password |  |
| 5 | btnChangePass | Button |  | Click | Change password and display message |  |

### Cart Page



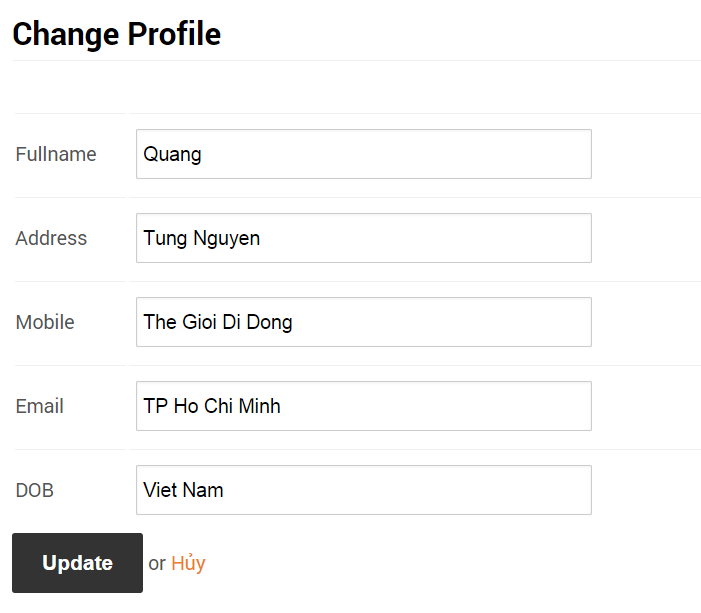
|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **No.** | **Name** | **Type** | **Validation** | **Event** | **Description** | **Status** |
| 1 | lbTitle | Label |  |  | Display Title |  |
| 2 | imgThumb | Image |  |  | Display image of product |  |
| 3 | Product | Link |  | Click | Link to product detail page |  |
| 4 | txtQuantity | Text Field | Not Null, Number |  | Input Quantity |  |
| 5 | lbPrice | Label |  |  | Display price of product |  |
| 6 | lbTotal | Label |  |  | Display total price of product |  |
| 7 | Update | Link |  | Click | Update quantity of product |  |
| 8 | Delete | Link |  | Click | Click to remove product from cart |  |
| 9 | lbTotalCart | Label |  |  | Total price of cart |  |
| 10 | btnHome | Button |  | Click | Go to home |  |
| 11 | btnCheckOut | Button |  | Click | Checkout |  |

### UserProfile



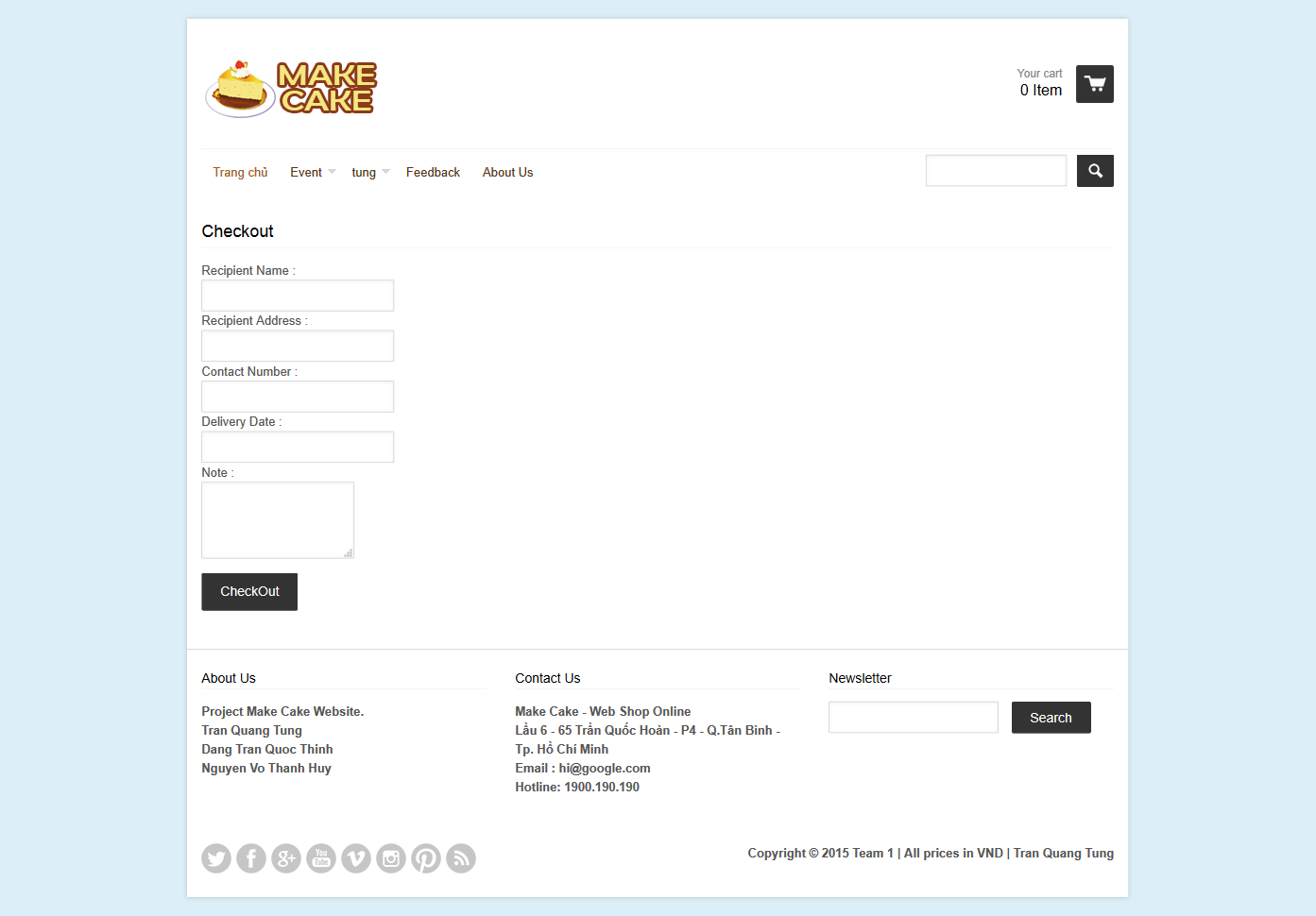
|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **No.** | **Name** | **Type** | **Validation** | **Event** | **Description** | **Status** |
| 1 | btnLogout | Link |  | Click | Logout and redirect to home |  |
| 2 | lnkOrder | Link |  | Click | Link to detail Order |  |

### Change Profile



|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **No.** | **Name** | **Type** | **Validation** | **Event** | **Description** | **Status** |
| 1 | txtName | Text Field | Not Null |  | Fullname |  |
| 2 | txtAddress | Text Field | Not Null |  | Address |  |
| 3 | txtPhone | Text Field | Not Null |  | Mobile |  |
| 4 | txtEmail | Text Field | Not Null, Email type |  | Email |  |
| 5 | txtDOB | Text Field | Not Null |  | Birthday |  |
| 6 | btnUpdate | Button |  | Click | Update profile |  |
| 7 | lnkCancel | Link |  | Click | Redirect to home |  |

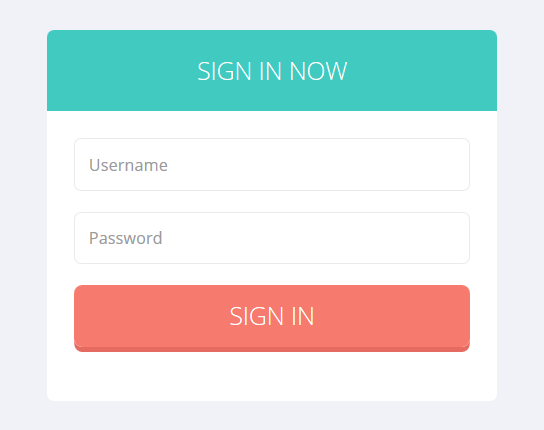
### Checkout



|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **No.** | **Name** | **Type** | **Validation** | **Event** | **Description** | **Status** |
| 1 | txtName | Text Field | Not Null |  | Recipient Name |  |
| 2 | txtAddress | Text Field | Not Null |  | Recipient Address |  |
| 3 | txtPhone | Text Field | Not Null |  | Recipient Phone |  |
| 4 | txtDate | Text Field | Not Null |  | Delivery Date |  |
| 5 | txtNote | Textarea |  |  | Note |  |
| 6 | btnCheckout | Button |  | Click | Checkout and redirect to index |  |

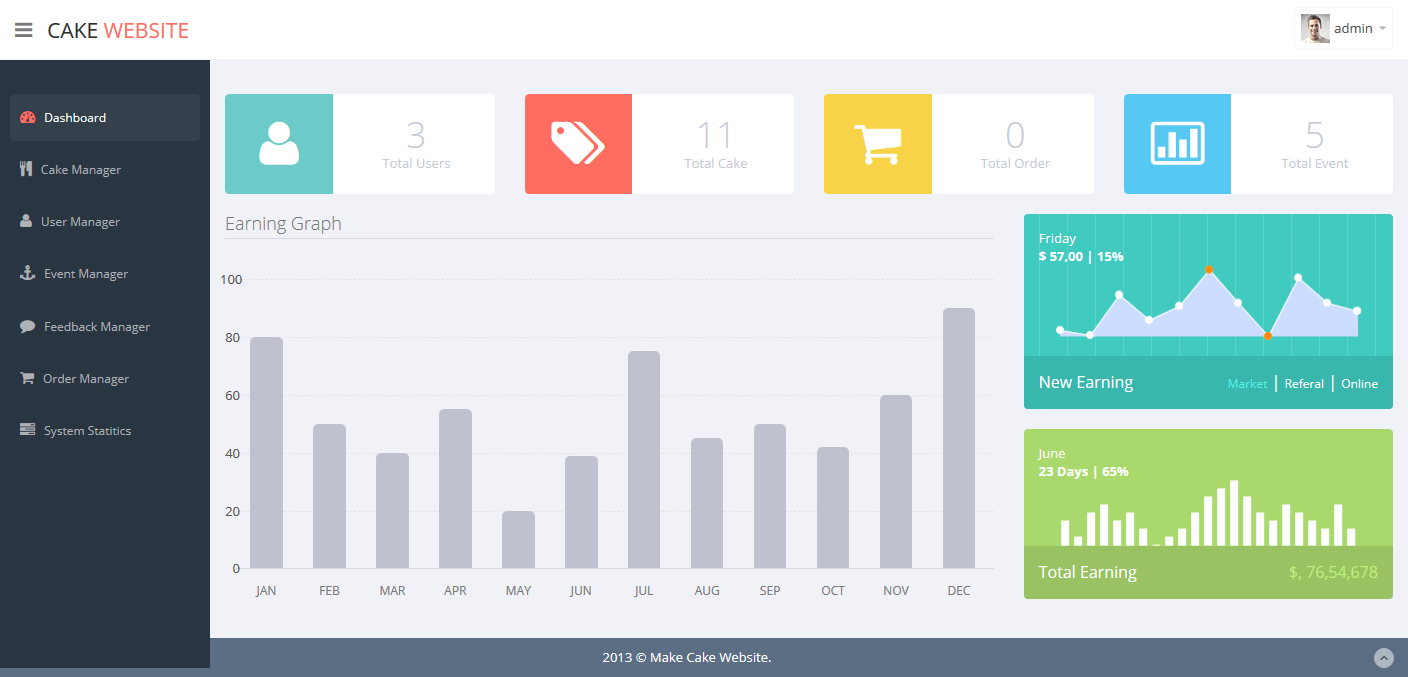
## Admin Site

### Login



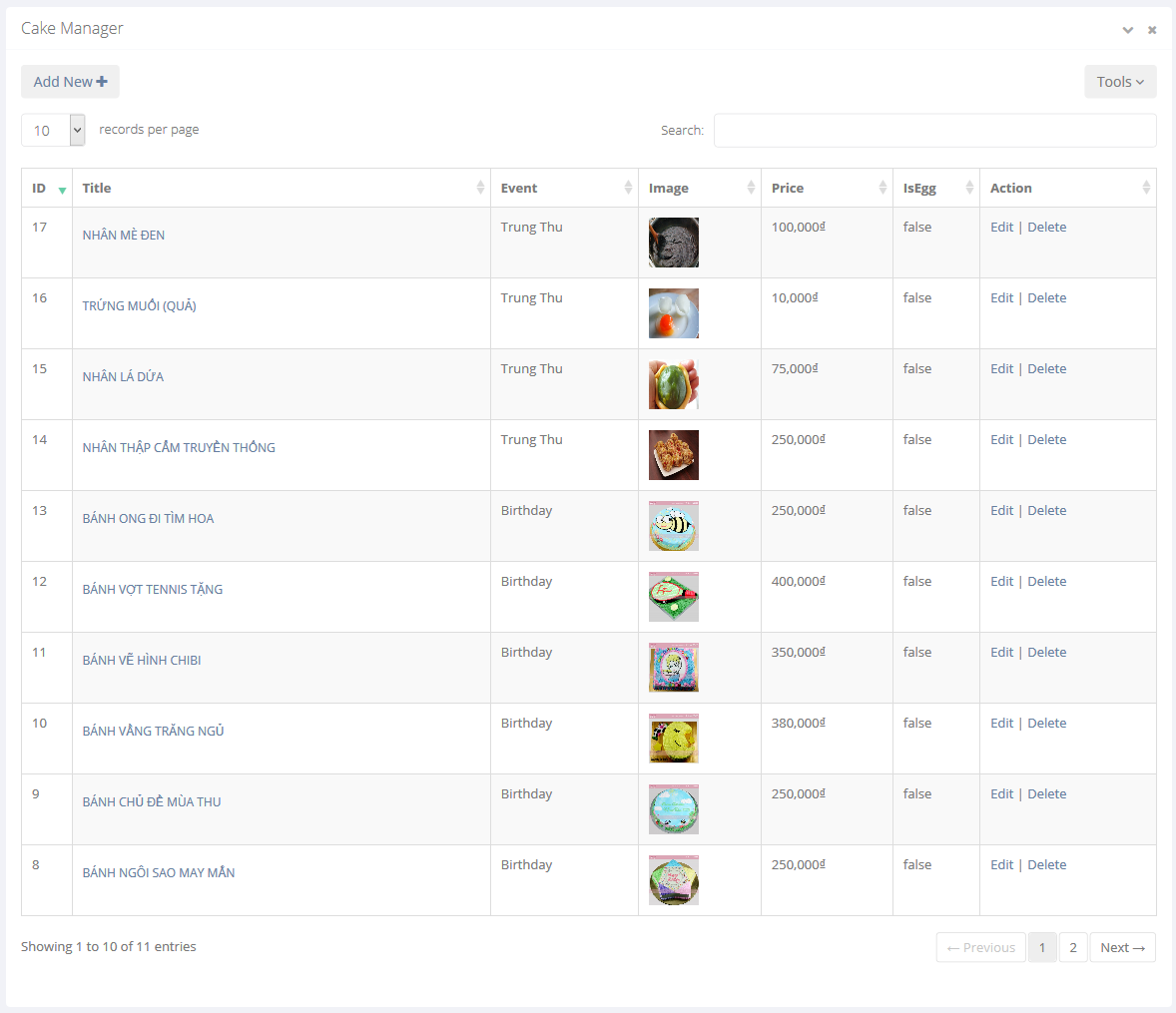
|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **No.** | **Name** | **Type** | **Validation** | **Event** | **Description** | **Status** |
| 1 | txtUsername | Text Field | Not Null |  | User name |  |
| 2 | txtPassword | Text Field | Not Null |  | Password |  |
| 3 | btnLogin | Button |  | Click | Login to website and redirect to home page |  |

### Home Page



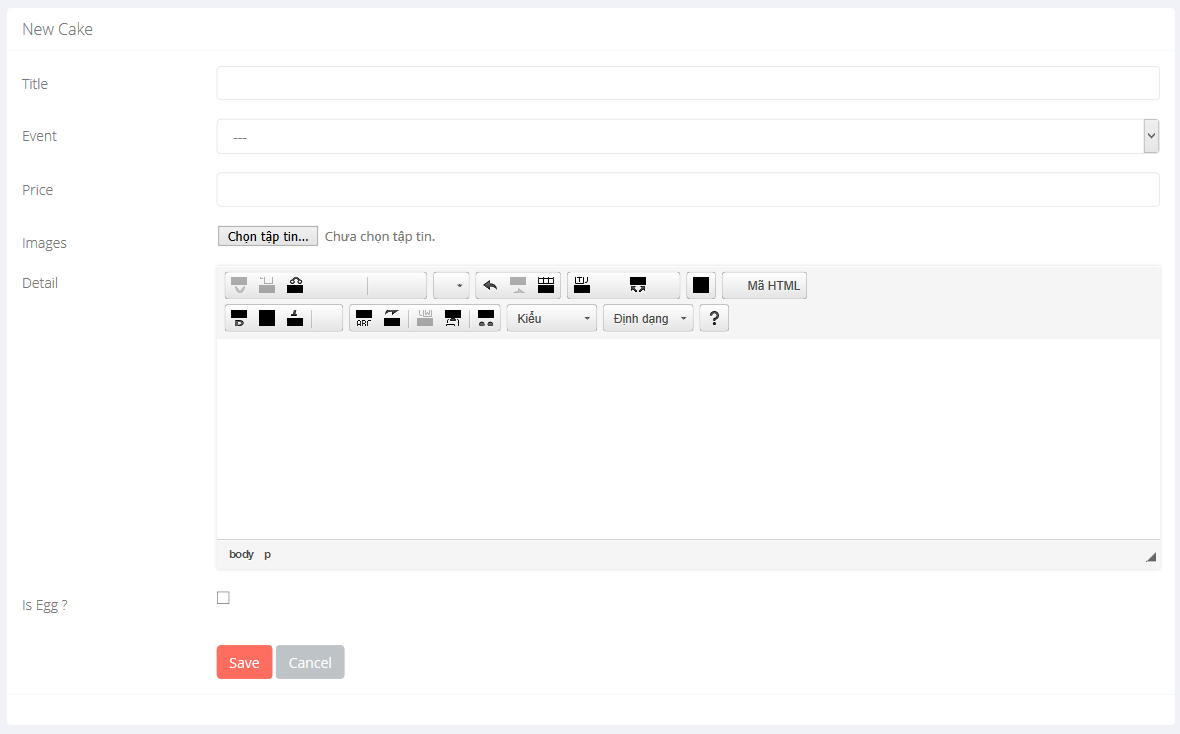
|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **No.** | **Name** | **Type** | **Validation** | **Event** | **Description** | **Status** |
| 1 | Dashboard | Link |  | Click | Link To Homepage |  |
| 2 | CakeManager | Link |  | Click | Link to cake manager page |  |
| 3 | UserManager | Link |  | Click | Link to user manager page |  |
| 4 | EventManager | Link |  | Click | Link to event manager page |  |
| 5 | Feedback Manager | Link |  | Click | Link to feedback manager page |  |
| 6 | Order Manager | Link |  | Cick | Link to order manager page |  |
| 7 | System Statitics | Link |  | Click | Link to system statitic |  |

### Cake Manager



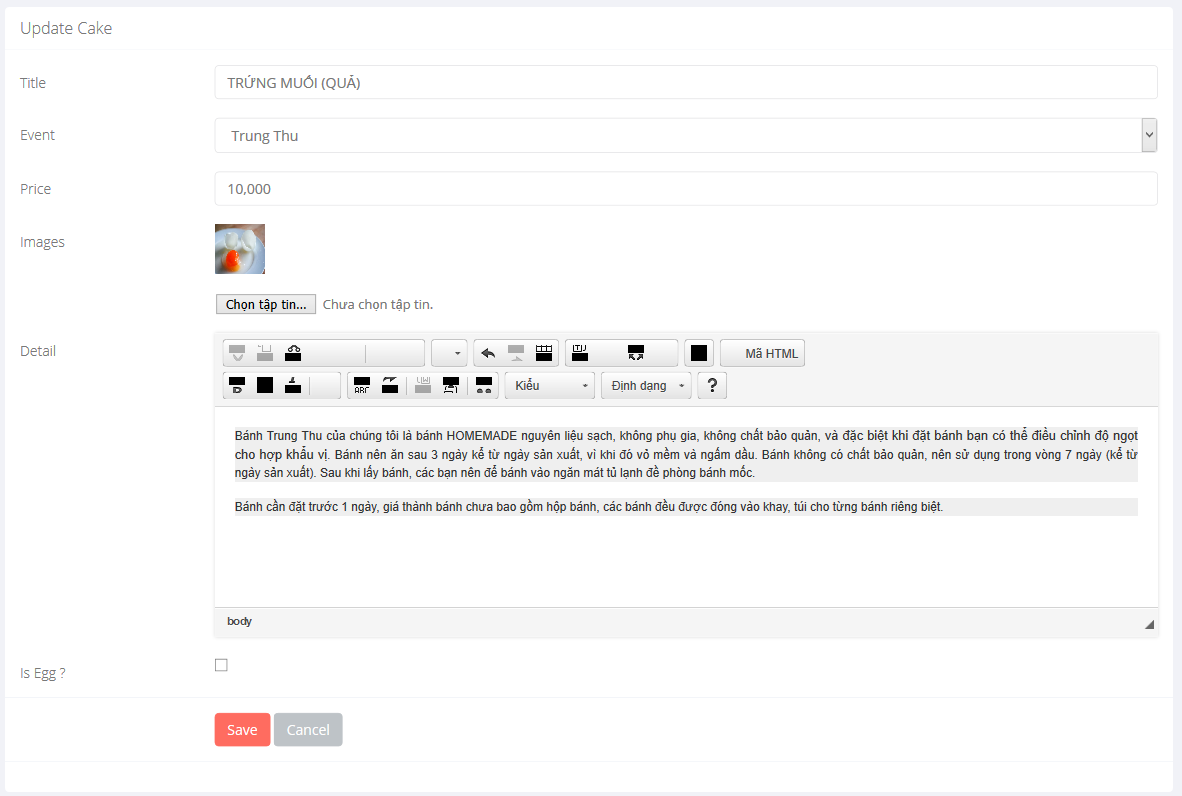
|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **No.** | **Name** | **Type** | **Validation** | **Event** | **Description** | **Status** |
| 1 | lbTitle | Label |  |  | Title of page |  |
| 2 | btnAddnew | Button |  | Click | Add New Cake |  |
| 3 | Detail | Link |  | Click | Link to product detail page |  |
| 4 | Thumb | Image |  |  | Image of product |  |
| 5 | Edit | Link |  | Click | Click to edit cake information |  |
| 6 | Delete | Link |  | Click | Click to delete cake |  |

### Add Cake



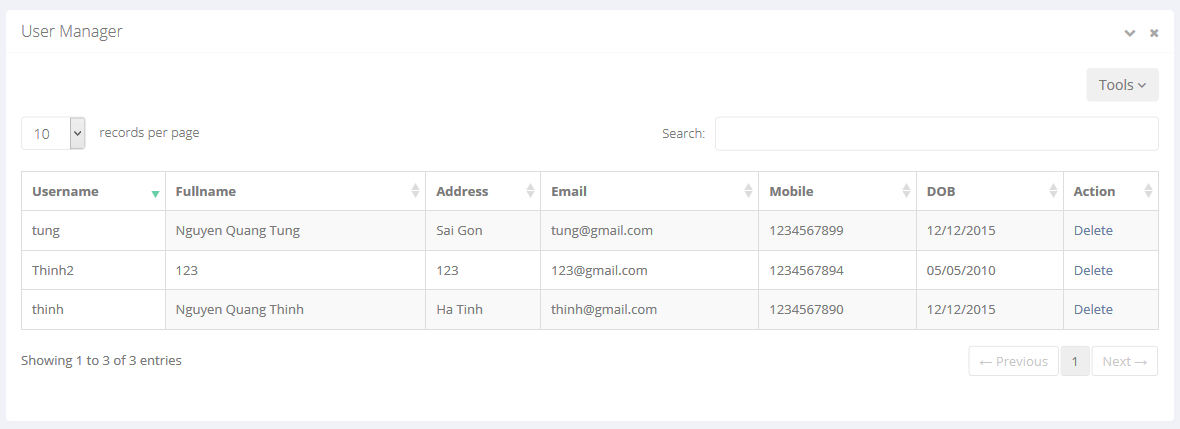
|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **No.** | **Name** | **Type** | **Validation** | **Event** | **Description** | **Status** |
| 1 | txtName | Text Field | Not Null |  | Cake Title |  |
| 2 | cboEvent | ComboBox | Not Null |  | Cake Event |  |
| 3 | txtPrice | Text Field | Not Null, Number |  | Cake Price |  |
| 4 | txtImage | File | Not Null |  | Thumb image of cake |  |
| 5 | txtDetail | Textarea | Not Null |  | Detail of cake |  |
| 6 | cbEgg | Check Box |  |  | IS Egg |  |
| 7 | btnSave | Button |  | Click | Save data to database and redirect to CakeManager page |  |
| 8 | btnCancel | Button |  | Click | Redirect to Cake Manager page |  |

### Update Cake



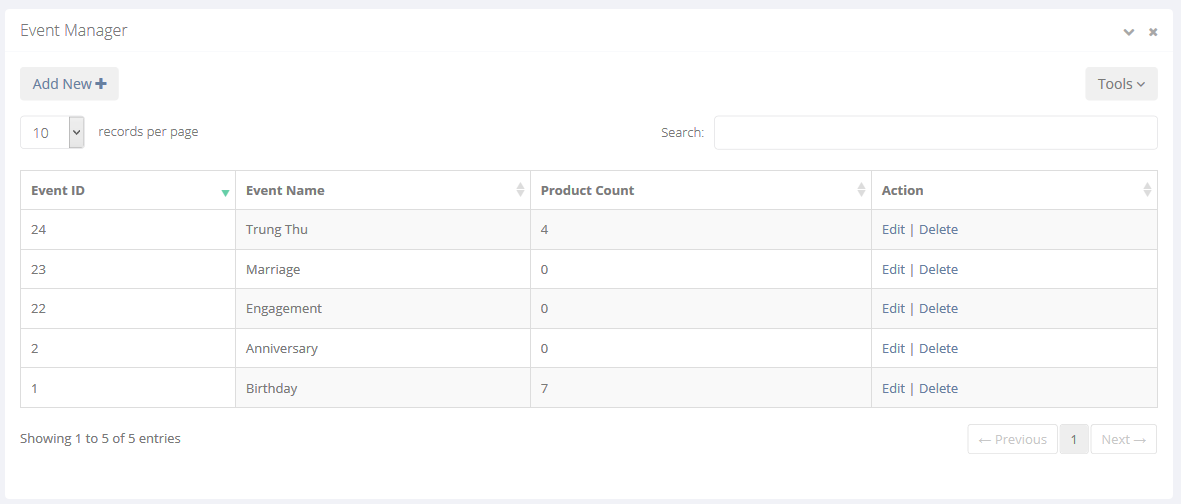
|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **No.** | **Name** | **Type** | **Validation** | **Event** | **Description** | **Status** |
| 1 | txtName | Text Field | Not Null |  | Cake Title |  |
| 2 | cboEvent | ComboBox | Not Null |  | Cake Event |  |
| 3 | txtPrice | Text Field | Not Null, Number |  | Cake Price |  |
| 4 | txtImage | File | Not Null |  | Thumb image of cake |  |
| 5 | txtDetail | Textarea | Not Null |  | Detail of cake |  |
| 6 | cbEgg | Check Box |  |  | IS Egg |  |
| 7 | btnSave | Button |  | Click | Save data to database and redirect to CakeManager page |  |
| 8 | btnCancel | Button |  | Click | Redirect to Cake Manager page |  |

### User manager



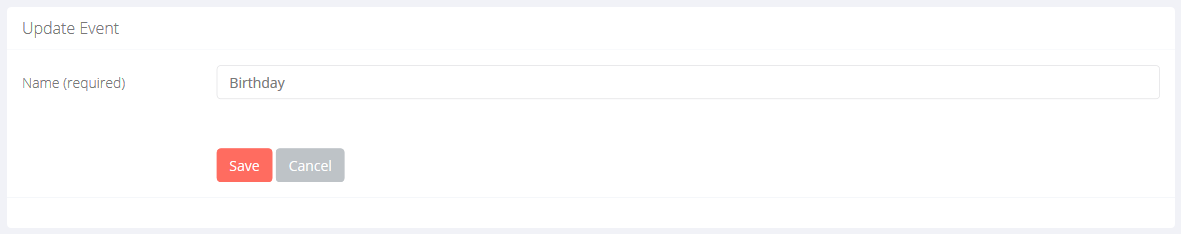
|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **No.** | **Name** | **Type** | **Validation** | **Event** | **Description** | **Status** |
| 1 | lbTitle | Label |  |  | Title of page |  |
| 6 | Delete | Link |  | Click | Click to delete user |  |

### Event manager



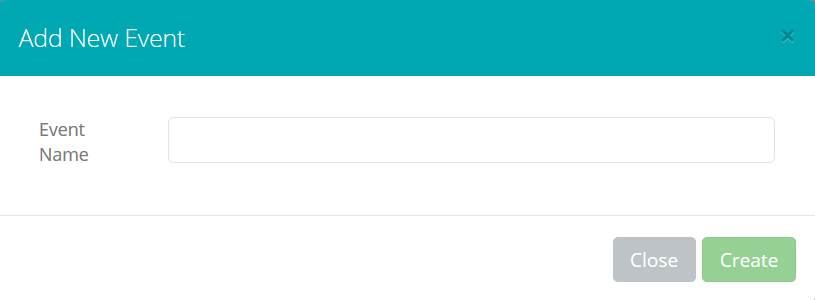
|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **No.** | **Name** | **Type** | **Validation** | **Event** | **Description** | **Status** |
| 1 | lbTitle | Label |  |  | Title of page |  |
| 2 | btnAddnew | Button |  | Click | Add New Event |  |
| 3 | Edit | Link |  | Click | Click to edit event information |  |
| 4 | Delete | Link |  | Click | Click to delete event |  |

### Update Event



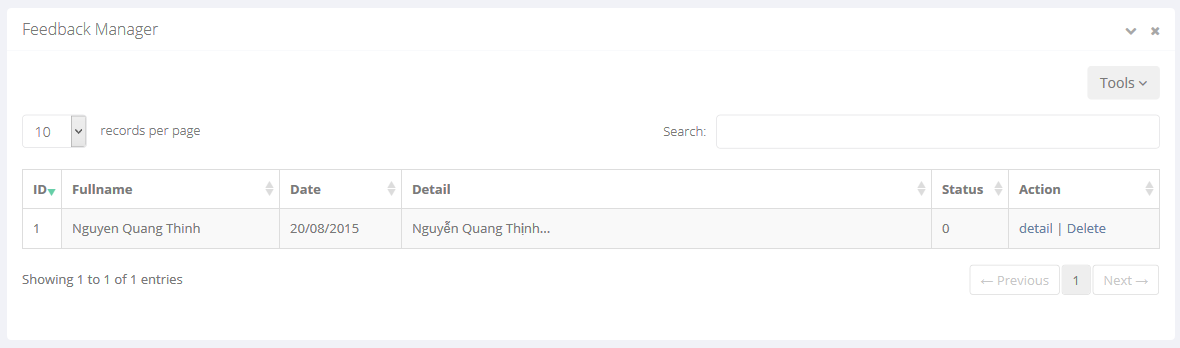
|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **No.** | **Name** | **Type** | **Validation** | **Event** | **Description** | **Status** |
| 1 | txtName | Text Field | Not Null |  | Event Name |  |
| 2 | btnSave | Button |  | Click | Save data to database and redirect to EventManager page |  |
| 3 | btnCancel | Button |  | Click | Redirect to Event Manager page |  |

### Add New Event



|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **No.** | **Name** | **Type** | **Validation** | **Event** | **Description** | **Status** |
| 1 | txtName | Text Field | Not Null |  | Event Name |  |
| 2 | btnSave | Button |  | Click | Save data to database and redirect to EventManager page |  |
| 3 | btnCancel | Button |  | Click | Redirect to Event Manager page |  |

### Feedback Manager



|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **No.** | **Name** | **Type** | **Validation** | **Event** | **Description** | **Status** |
| 1 | lbTitle | Label |  |  | Title of page |  |
| 2 | Detail | Link |  | Click | Link to view detail feedback |  |
| 3 | Delete | Link |  | Click | Click to delete feedback |  |

# Task Sheet

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Project Ref. No 1** | | **Project Title** | **Activity Plan Prepared By** | **Date of Preparation of Activity Plan** | | | | |
| **Sr. No** | **Task** | **Start Date** | **End Date** | **Actual Days** | **Member’s name** | **Status** |
| **Document** | | | | | | | | |
| 1 | Introduction | The Cake | Tran Quang Tung | 03/08/2015 | 06/08/2015 | 03 | All of Team | FINISH |
| 2 | Existing Scenario | The Cake | Tran Quang Tung | 03/08/2015 | 06/08/2015 | 03 | All of Team | FINISH |
| 3 | Customer Requirement Specification | The Cake | Tran Quang Tung | 03/08/2015 | 06/08/2015 | 03 | All of Team | FINISH |
| 4 | Functional Requirement Specification | The Cake | Tran Quang Tung | 03/08/2015 | 06/08/2015 | 03 | All of Team | FINISH |
| 5 | System Requirements | The Cake | Tran Quang Tung | 03/08/2015 | 06/08/2015 | 03 | All of Team | FINISH |
| 6 | Development Software | The Cake | Tran Quang Tung | 03/08/2015 | 06/08/2015 | 03 | All of Team | FINISH |
| 7 | Technology | The Cake | Tran Quang Tung | 03/08/2015 | 06/08/2015 | 03 | All of Team | FINISH |
| 8 | Scope of work | The Cake | Tran Quang Tung | 03/08/2015 | 06/08/2015 | 03 | All of Team | FINISH |
| 9 | Use case and sequence diagram | The Cake | Tran Quang Tung | 07/08/2015 | 13/08/2015 | 07 | All of Team | FINISH |
| 10 | Entity relationship diagram | The Cake | Tran Quang Tung | 07/08/2015 | 13/08/2015 | 07 | All of Team | FINISH |
| 11 | Database relationship | The Cake | Tran Quang Tung | 14/08/2015 | 23/08/2015 | 10 | All of Team | FINISH |
| 12 | Table definition | The Cake | Tran Quang Tung | 14/08/2015 | 23/08/2015 | 10 | All of Team | FINISH |
| 13 | Other constrains | The Cake | Tran Quang Tung | 14/08/2015 | 23/08/2015 | 10 | All of Team | FINISH |
| 14 | Sitemap | The Cake | Tran Quang Tung | 14/08/2015 | 23/08/2015 | 10 | All of Team | FINISH |
| 15 | GUI design | The Cake | Tran Quang Tung | 14/08/2015 | 23/08/2015 | 10 | All of Team | FINISH |
| **Functional** | | | | | | | | |
| 16 | UserViewCake | The Cake | Tran Quang Tung | 14/08/2015 | 23/08/2015 | 10 |  | FINISH |
| 17 | UserSearchCake | The Cake | Tran Quang Tung | 14/08/2015 | 23/08/2015 | 10 |  | FINISH |
| 18 | UserRegister as member | The Cake | Tran Quang Tung | 14/08/2015 | 23/08/2015 | 10 |  | FINISH |
| 19 | UserLogin | The Cake | Tran Quang Tung | 14/08/2015 | 23/08/2015 | 10 |  | FINISH |
| 20 | UserLogout | The Cake | Tran Quang Tung | 14/08/2015 | 23/08/2015 | 10 |  | FINISH |
| 21 | UserViewProfile | The Cake | Tran Quang Tung | 14/08/2015 | 23/08/2015 | 10 |  | FINISH |
| 22 | UserChangeProfile | The Cake | Tran Quang Tung | 14/08/2015 | 23/08/2015 | 10 |  | FINISH |
| 23 | UserChangePassword | The Cake | Tran Quang Tung | 14/08/2015 | 23/08/2015 | 10 |  | FINISH |
| 24 | UserFeedback | The Cake | Tran Quang Tung | 14/08/2015 | 23/08/2015 | 10 |  | FINISH |
| 25 | UserAddToCart | The Cake | Tran Quang Tung | 14/08/2015 | 23/08/2015 | 10 |  | FINISH |
| 26 | UserCheckout | The Cake | Tran Quang Tung | 14/08/2015 | 23/08/2015 | 10 |  | FINISH |
| 27 | UserViewAboutUs | The Cake | Tran Quang Tung | 14/08/2015 | 23/08/2015 | 10 |  | FINISH |
| 28 | UserViewEvent | The Cake | Tran Quang Tung | 14/08/2015 | 23/08/2015 | 10 |  | FINISH |
| 29 | AdminLogin | The Cake | Tran Quang Tung | 14/08/2015 | 23/08/2015 | 10 |  | FINISH |
| 30 | AdminLogout | The Cake | Tran Quang Tung | 14/08/2015 | 23/08/2015 | 10 |  | FINISH |
| 31 | AdminChangePassword | The Cake | Tran Quang Tung | 14/08/2015 | 23/08/2015 | 10 |  | FINISH |
| 32 | AdminViewFeedbacks | The Cake | Tran Quang Tung | 14/08/2015 | 23/08/2015 | 10 |  | FINISH |
| 33 | AdminDeleteFeedbacks | The Cake | Tran Quang Tung | 14/08/2015 | 23/08/2015 | 10 |  | FINISH |
| 34 | AdminDetailFeedack | The Cake | Tran Quang Tung | 14/08/2015 | 23/08/2015 | 10 |  | FINISH |
| 35 | AdminViewCustomers | The Cake | Tran Quang Tung | 14/08/2015 | 23/08/2015 | 10 |  | FINISH |
| 36 | AdminDeleteCustomer | The Cake | Tran Quang Tung | 14/08/2015 | 23/08/2015 | 10 |  | FINISH |
| 37 | AdminViewOrders | The Cake | Tran Quang Tung | 14/08/2015 | 23/08/2015 | 10 |  | FINISH |
| 38 | AdminProcessOrder | The Cake | Tran Quang Tung | 14/08/2015 | 23/08/2015 | 10 |  | FINISH |
| 39 | AdminDeleteOrder | The Cake | Tran Quang Tung | 14/08/2015 | 23/08/2015 | 10 |  | FINISH |
| 40 | AdminViewCake | The Cake | Tran Quang Tung | 14/08/2015 | 23/08/2015 | 10 |  | FINISH |
| 41 | AdminAddNewCake | The Cake | Tran Quang Tung | 14/08/2015 | 23/08/2015 | 10 |  | FINISH |
| 42 | AdminModifyCake | The Cake | Tran Quang Tung | 14/08/2015 | 23/08/2015 | 10 |  | FINISH |
| 43 | AdminDeleteCake | The Cake | Tran Quang Tung | 14/08/2015 | 23/08/2015 | 10 |  | FINISH |
| 44 | AdminViewEvent | The Cake | Tran Quang Tung | 14/08/2015 | 23/08/2015 | 10 |  | FINISH |
| 45 | AdminAddNewEvent | The Cake | Tran Quang Tung | 14/08/2015 | 23/08/2015 | 10 |  | FINISH |
| 46 | AdminModifyEvent | The Cake | Tran Quang Tung | 14/08/2015 | 23/08/2015 | 10 |  | FINISH |
| 47 | AdminDeleteEvent | The Cake | Tran Quang Tung | 14/08/2015 | 23/08/2015 | 10 |  | FINISH |
| 48 | AdminViewSystemStatitic | The Cake | Tran Quang Tung | 14/08/2015 | 23/08/2015 | 10 |  | FINISH |
| **Task sheet** | | | | | | | | |
| 49 | Task sheet | The Cake | Tran Quang Tung | 03/08/2015 | 23/08/2015 | 30 | Tran Quang Tung | FINISH |

|  |  |
| --- | --- |
| Leader | Faculty |
|  |  |
|  |  |
| Tran Quang Tung | Duong Trong Phu Son |