Test Record

System Test Record

|  |  |
| --- | --- |
| **Test Case ID** | STC-01 |
| **Test Case Name** | View Index page |
| **Test Case Description** | Test for check that Index page can be viewed |
| **Test Responsibility** | Tester |
| **Tested by** | Chanyanuch Chaithongsri |
| **Test Date** | 18/05/2015 |
| **Prerequisites** | No Sign-in |
| **Procedure steps** | 1. Guest Click on “Home” |

|  |  |  |  |
| --- | --- | --- | --- |
| **Result** | | | |
| **Input** | **Expected Output** | **Actual Output** | **Result** |
| Click on “Home” link button | Home page is shown | Home page is shown | Pass |

|  |  |
| --- | --- |
| **Test Case ID** | STC-02 |
| **Test Case Name** | View Temple Background page |
| **Test Case Description** | Test for check that Temple Background page can be viewed |
| **Test Responsibility** | Tester |
| **Tested by** | Chanyanuch Chaithongsri |
| **Test Date** | 18/05/2015 |
| **Prerequisites** | No Sign-in |
| **Procedure steps** | 1. Guest Click on “Temple Background” |

|  |  |  |  |
| --- | --- | --- | --- |
| **Result** | | | |
| **Input** | **Expected Output** | **Actual Output** | **Result** |
| Click on “Temple Background” link button | Temple Background page is shown | Temple Background page is shown | Pass |

|  |  |
| --- | --- |
| **Test Case ID** | STC-03 |
| **Test Case Name** | View News and Activities page |
| **Test Case Description** | Test for check that News and Activities page can be viewed |
| **Test Responsibility** | Tester |
| **Tested by** | Chanyanuch Chaithongsri |
| **Test Date** | 18/05/2015 |
| **Prerequisites** | No Sign-in |
| **Procedure steps** | 1. Guest Click on “News and Activities” |

|  |  |  |  |
| --- | --- | --- | --- |
| **Result** | | | |
| **Input** | **Expected Output** | **Actual Output** | **Result** |
| Click on “News and Activities” link button | News and Activities is shown | News and Activities page is shown | Pass |

|  |  |
| --- | --- |
| **Test Case ID** | STC-04 |
| **Test Case Name** | View more details in selected News and Activities topic |
| **Test Case Description** | Test for check that more details of selected News and Activities topic page can be viewed |
| **Test Responsibility** | Tester |
| **Tested by** | Chanyanuch Chaithongsri |
| **Test Date** | 18/05/2015 |
| **Prerequisites** | No Sign-in |
| **Procedure steps** | 1. Guest Click on “News and Activities” 2. Guest click on “More details” |

|  |  |  |  |
| --- | --- | --- | --- |
| **Result** | | | |
| **Input** | **Expected Output** | **Actual Output** | **Result** |
| Click on “More details” link button | More details of selected News and Activities is shown | More details of selected News and Activities is shown | Pass |

|  |  |
| --- | --- |
| **Test Case ID** | STC-05 |
| **Test Case Name** | View Dhamma Content page |
| **Test Case Description** | Test for check that Dhamma Content page can be viewed |
| **Test Responsibility** | Tester |
| **Tested by** | Chanyanuch Chaithongsri |
| **Test Date** | 18/05/2015 |
| **Prerequisites** | No Sign-in |
| **Procedure steps** | 1. Guest Click on “Dhamma Content” |

|  |  |  |  |
| --- | --- | --- | --- |
| **Result** | | | |
| **Input** | **Expected Output** | **Actual Output** | **Result** |
| Click on “Dhamma Content” link button | Dhamma Content is shown | Dhamma Content is shown | Pass |

|  |  |
| --- | --- |
| **Test Case ID** | STC-06 |
| **Test Case Name** | View more details in selected Dhamma Content topic |
| **Test Case Description** | Test for check that more details of selected Dhamma Content topic page can be viewed |
| **Test Responsibility** | Tester |
| **Tested by** | Chanyanuch Chaithongsri |
| **Test Date** | 18/05/2015 |
| **Prerequisites** | No Sign-in |
| **Procedure steps** | 1. Guest Click on “Dhamma Content” 2. Guest click on “More details” |

|  |  |  |  |
| --- | --- | --- | --- |
| **Result** | | | |
| **Input** | **Expected Output** | **Actual Output** | **Result** |
| Click on “More details” link button | More details of selected Dhamma Content is shown | More details of selected Dhamma Content is shown | Pass |

|  |  |
| --- | --- |
| **Test Case ID** | STC-07 |
| **Test Case Name** | View register page |
| **Test Case Description** | Test for check that register page can be views |
| **Test Responsibility** | Tester |
| **Tested by** | Chanyanuch Chaithongsri |
| **Test Date** | 18/05/2015 |
| **Prerequisites** | No Sign-in |
| **Procedure steps** | 1. Guest Click on “Register” |

|  |  |  |  |
| --- | --- | --- | --- |
| **Result** | | | |
| **Input** | **Expected Output** | **Actual Output** | **Result** |
| Click on “Register” link button | Register page is shown | Register page is shown | Pass |

|  |  |
| --- | --- |
| **Test Case ID** | STC-08 |
| **Test Case Name** | Register to be member |
| **Test Case Description** | Test for check that can be register to be member |
| **Test Responsibility** | Tester |
| **Tested by** | Chanyanuch Chaithongsri |
| **Test Date** | 18/05/2015 |
| **Prerequisites** | No Sign-in |
| **Procedure steps** | 1. Guest Click on “Register” 2. Guest input their register information in register form 3. Guest click “Submit” button |

|  |  |  |  |
| --- | --- | --- | --- |
| **Result** | | | |
| **Input** | **Expected Output** | **Actual Output** | **Result** |
| **Username** = “member1”  **Password** = “member1”  **Re-Password** = “member1”  **Email** = “chanyanuch.ch.025@gmail.com” | Sign in and show Hello,member1 | Sign in and show Hello,member1 | Pass |
| **Username** = “member2”  **Password** = “member2”  **Re-Password** = “member2”  **Email** = “chanyanuch.ch.025@gmail.com” | Sign in and show Hello,member2 | Sign in and show Hello,member2 | Pass |

|  |  |
| --- | --- |
| **Test Case ID** | STC-09 |
| **Test Case Name** | View Sign in page |
| **Test Case Description** | Test for check that sign in page can be viewed |
| **Test Responsibility** | Tester |
| **Tested by** | Chanyanuch Chaithongsri |
| **Test Date** | 18/05/2015 |
| **Prerequisites** | No Sign-in |
| **Procedure steps** | 1. Guest Click on “Sign In” |

|  |  |  |  |
| --- | --- | --- | --- |
| **Result** | | | |
| **Input** | **Expected Output** | **Actual Output** | **Result** |
| Click on “Sign in” link button | Sign in page is shown | Sign in page is shown | Pass |

|  |  |
| --- | --- |
| **Test Case ID** | STC-10 |
| **Test Case Name** | View Temple Map page |
| **Test Case Description** | Test for check that Temple Map page can be viewed |
| **Test Responsibility** | Tester |
| **Tested by** | Chanyanuch Chaithongsri |
| **Test Date** | 18/05/2015 |
| **Prerequisites** | No Sign-in |
| **Procedure steps** | 1. Guest Click on “Temple Map” |

|  |  |  |  |
| --- | --- | --- | --- |
| **Result** | | | |
| **Input** | **Expected Output** | **Actual Output** | **Result** |
| Click on “Temple Map” link button | Temple Map page is shown | Temple Map page is shown | Pass |

|  |  |
| --- | --- |
| **Test Case ID** | STC-11 |
| **Test Case Name** | Get directions to temple from Google Maps |
| **Test Case Description** | Test for check that can be use Google Maps to get direction to temple |
| **Test Responsibility** | Tester |
| **Tested by** | Chanyanuch Chaithongsri |
| **Test Date** | 18/05/2015 |
| **Prerequisites** | No Sign-in |
| **Procedure steps** | 1. Guest Click on “Temple Map” 2. Guest view Google Maps |

|  |  |  |  |
| --- | --- | --- | --- |
| **Result** | | | |
| **Input** | **Expected Output** | **Actual Output** | **Result** |
| View on Google Maps | Google Maps show the direction to temple | Google Maps show the direction to temple | Pass |

|  |  |
| --- | --- |
| **Test Case ID** | STC-12 |
| **Test Case Name** | View Contact page |
| **Test Case Description** | Test for check that Contact page can be viewed |
| **Test Responsibility** | Tester |
| **Tested by** | Chanyanuch Chaithongsri |
| **Test Date** | 18/05/2015 |
| **Prerequisites** | No Sign-in |
| **Procedure steps** | 1. Guest Click on “Contact” |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Result** | | | | |
| **Input** | **Expected Output** | | **Actual Output** | **Result** |
| Click on “Contact” link button | Contact page is shown | | Contact page is shown | Pass |
| **Test Case ID** | | STC-13 | | |
| **Test Case Name** | | Contact Temple by email | | |
| **Test Case Description** | | Test for check that can be send email to contact temple by using contact form | | |
| **Test Responsibility** | | Tester | | |
| **Tested by** | | Chanyanuch Chaithongsri | | |
| **Test Date** | | 18/05/2015 | | |
| **Prerequisites** | | No Sign-in | | |
| **Procedure steps** | | 1. Guest Click on “Contact” 2. Guest input contact information in contact form 3. Guest click on “Submit” | | |

|  |  |  |  |
| --- | --- | --- | --- |
| **Result** | | | |
| **Input** | **Expected Output** | **Actual Output** | **Result** |
| **Contact Name** = “Chanyanuch”  **Contact Email** = “chanyanuch.ch.025@gmail.com”  **Contact Topic** = “Contact Temple”  **Contact Detail** = “I am contact you” | Email has been send to temple | Email has been send to temple | Pass |
| **Contact Name** = “Chanyanuch”  **Contact Email** = “se552115015@vr.camt.info”  **Contact Topic** = “Contact Temple Again”  **Contact Detail** = “I am contact you again” | Email has been send to temple | Email has been send to temple | Pass |

|  |  |
| --- | --- |
| **Test Case ID** | STC-14 |
| **Test Case Name** | Sign In |
| **Test Case Description** | Test for check that can be sign in to the system as a member |
| **Test Responsibility** | Tester |
| **Tested by** | Chanyanuch Chaithongsri |
| **Test Date** | 18/05/2015 |
| **Prerequisites** | No Sign-in |
| **Procedure steps** | 1. Guest Click on “Sign In” 2. Guest input username and password 3. Guest click on “Submit” button |

|  |  |  |  |
| --- | --- | --- | --- |
| **Result** | | | |
| **Input** | **Expected Output** | **Actual Output** | **Result** |
| **User Name** = “member1”  **Password** = “member1” | Hello , member1 is shown | Hello , member1 is shown | Pass |
| **User Name** = “member2”  **Password** = “member2” | Hello , member2 is shown | Hello , member2 is shown | Pass |

|  |  |
| --- | --- |
| **Test Case ID** | STC-15 |
| **Test Case Name** | Sign out |
| **Test Case Description** | Test for check that member can be sign out |
| **Test Responsibility** | Tester |
| **Tested by** | Chanyanuch Chaithongsri |
| **Test Date** | 18/05/2015 |
| **Prerequisites** | Sign In as a member |
| **Procedure steps** | 1. Click on “Sign Out” |

|  |  |  |  |
| --- | --- | --- | --- |
| **Result** | | | |
| **Input** | **Expected Output** | **Actual Output** | **Result** |
| member1 clicks Sign out | member1 Sign out and redirect to Index page | member1 Sign out and redirect to Index page | Pass |
| member2 clicks Sign out | member2 Sign out and redirect to Index page | member2 Sign out and redirect to Index page | Pass |

|  |  |
| --- | --- |
| **Test Case ID** | STC-16 |
| **Test Case Name** | View Index page |
| **Test Case Description** | Test for check that Index page can be viewed if sign in as a member |
| **Test Responsibility** | Tester |
| **Tested by** | Chanyanuch Chaithongsri |
| **Test Date** | 18/05/2015 |
| **Prerequisites** | Sign In as a member |
| **Procedure steps** | 1. Sign In as a member 2. Click on “Home” |

|  |  |  |  |
| --- | --- | --- | --- |
| **Result** | | | |
| **Input** | **Expected Output** | **Actual Output** | **Result** |
| Click on “Home” link button | Home page is shown | Home page is shown | Pass |

|  |  |
| --- | --- |
| **Test Case ID** | STC-17 |
| **Test Case Name** | View Temple Background page |
| **Test Case Description** | Test for check that Temple Background page can be viewed if sign in as a member |
| **Test Responsibility** | Tester |
| **Tested by** | Chanyanuch Chaithongsri |
| **Test Date** | 18/05/2015 |
| **Prerequisites** | Sign In as a member |
| **Procedure steps** | 1. Sign In as a member 2. Click on “Temple Background” |

|  |  |  |  |
| --- | --- | --- | --- |
| **Result** | | | |
| **Input** | **Expected Output** | **Actual Output** | **Result** |
| Click on “Temple Background” link button | Temple Background page is shown | Temple Background page is shown | Pass |

|  |  |
| --- | --- |
| **Test Case ID** | STC-18 |
| **Test Case Name** | View News and Activities page |
| **Test Case Description** | Test for check that News and Activities page can be viewed if sign in as a member |
| **Test Responsibility** | Tester |
| **Tested by** | Chanyanuch Chaithongsri |
| **Test Date** | 18/05/2015 |
| **Prerequisites** | Sign In as a member |
| **Procedure steps** | 1. Sign In as a member 2. Click on “News and Activities” |

|  |  |  |  |
| --- | --- | --- | --- |
| **Result** | | | |
| **Input** | **Expected Output** | **Actual Output** | **Result** |
| Click on “News and Activities” link button | News and Activities page is shown | News and Activities page is shown | Pass |

|  |  |
| --- | --- |
| **Test Case ID** | STC-19 |
| **Test Case Name** | View more details of selected News and Activities topic |
| **Test Case Description** | Test for check that selected News and Activities topic can be view for get more details if sign in as a member |
| **Test Responsibility** | Tester |
| **Tested by** | Chanyanuch Chaithongsri |
| **Test Date** | 18/05/2015 |
| **Prerequisites** | Sign In as a member |
| **Procedure steps** | 1. Sign In as a member 2. Click on “News and Activities” 3. Click on “More details” of selected News and Activities topic |

|  |  |  |  |
| --- | --- | --- | --- |
| **Result** | | | |
| **Input** | **Expected Output** | **Actual Output** | **Result** |
| Click on “More details” of selected News and Activities topic | More details of selected News and Activities topic is shown | More details of selected News and Activities topic is shown | Pass |

|  |  |
| --- | --- |
| **Test Case ID** | STC-20 |
| **Test Case Name** | Receive Email |
| **Test Case Description** | Test for check that member can be receive email from admin |
| **Test Responsibility** | Tester |
| **Tested by** | Chanyanuch Chaithongsri |
| **Test Date** | 18/05/2015 |
| **Prerequisites** | Register to be member |
| **Procedure steps** | 1. Open their email |

|  |  |  |  |
| --- | --- | --- | --- |
| **Result** | | | |
| **Input** | **Expected Output** | **Actual Output** | **Result** |
| Open their email | Get email from temple in their inbox | Get email from temple in their inbox | Pass |

|  |  |
| --- | --- |
| **Test Case ID** | STC-21 |
| **Test Case Name** | View Dhamma Content page |
| **Test Case Description** | Test for check that Dhamma Content page can be viewed if sign in as a member |
| **Test Responsibility** | Tester |
| **Tested by** | Chanyanuch Chaithongsri |
| **Test Date** | 18/05/2015 |
| **Prerequisites** | Sign In as a member |
| **Procedure steps** | 1. Sign In as a member 2. Click on “Dhamma Content” |

|  |  |  |  |
| --- | --- | --- | --- |
| **Result** | | | |
| **Input** | **Expected Output** | **Actual Output** | **Result** |
| Click on “Dhamma Content” link button | Dhamma Content page is shown | Dhamma Content page is shown | Pass |

|  |  |
| --- | --- |
| **Test Case ID** | STC-22 |
| **Test Case Name** | View more details of selected Dhamma Content topic |
| **Test Case Description** | Test for check that the selected of Dhamma Content page can be viewed if sign in as a member |
| **Test Responsibility** | Tester |
| **Tested by** | Chanyanuch Chaithongsri |
| **Test Date** | 18/05/2015 |
| **Prerequisites** | Sign In as a member |
| **Procedure steps** | 1. Sign In as a member 2. Click on “Dhamma Content” 3. Click on “More details” of selected Dhamma Content topic |

|  |  |  |  |
| --- | --- | --- | --- |
| **Result** | | | |
| **Input** | **Expected Output** | **Actual Output** | **Result** |
| Click on “More details” of selected Dhamma Content topic | More details of selected Dhamma Content topic is shown | More details of selected Dhamma Content topic is shown | Pass |

|  |  |
| --- | --- |
| **Test Case ID** | STC-23 |
| **Test Case Name** | View Gallery page |
| **Test Case Description** | Test for check that Gallery page can be viewed if sign in as a member |
| **Test Responsibility** | Tester |
| **Tested by** | Chanyanuch Chaithongsri |
| **Test Date** | 18/05/2015 |
| **Prerequisites** | Sign In as a member |
| **Procedure steps** | 1. Sign In as a member 2. Click on “Gallery” |

|  |  |  |  |
| --- | --- | --- | --- |
| **Result** | | | |
| **Input** | **Expected Output** | **Actual Output** | **Result** |
| Click on “Gallery” link button | Gallery page is shown | Gallery page is shown | Pass |

|  |  |
| --- | --- |
| **Test Case ID** | STC-24 |
| **Test Case Name** | View selected image in full size on Gallery page |
| **Test Case Description** | Test for check that can view image in full size on Gallery page |
| **Test Responsibility** | Tester |
| **Tested by** | Chanyanuch Chaithongsri |
| **Test Date** | 18/05/2015 |
| **Prerequisites** | Sign In as a member |
| **Procedure steps** | 1. Sign In as a member 2. Click on “Gallery” 3. Put mouse on selected image |

|  |  |  |  |
| --- | --- | --- | --- |
| **Result** | | | |
| **Input** | **Expected Output** | **Actual Output** | **Result** |
| Put mouse on selected image | Selected Image is shown in full size | Selected Image is shown in full size | Pass |

|  |  |
| --- | --- |
| **Test Case ID** | STC-25 |
| **Test Case Name** | View Activities Video Clips page |
| **Test Case Description** | Test for check that Activities Video Clips page can be viewed if sign in as a member |
| **Test Responsibility** | Tester |
| **Tested by** | Chanyanuch Chaithongsri |
| **Test Date** | 18/05/2015 |
| **Prerequisites** | Sign In as a member |
| **Procedure steps** | 1. Sign In as a member 2. Click on “Activities Video Clips” |

|  |  |  |  |
| --- | --- | --- | --- |
| **Result** | | | |
| **Input** | **Expected Output** | **Actual Output** | **Result** |
| Click on “Activities Video Clips” link button | Activities Video Clips page is shown | Activities Video Clips page is shown | Pass |

|  |  |
| --- | --- |
| **Test Case ID** | STC-26 |
| **Test Case Name** | View selected video on Activities Video Clips page |
| **Test Case Description** | Test for check that video on Activities Video Clips page can be viewed |
| **Test Responsibility** | Tester |
| **Tested by** | Chanyanuch Chaithongsri |
| **Test Date** | 18/05/2015 |
| **Prerequisites** | Sign In as a member |
| **Procedure steps** | 1. Sign In as a member 2. Click on “Activities Video Clips” 3. Click on selected video to be viewed |

|  |  |  |  |
| --- | --- | --- | --- |
| **Result** | | | |
| **Input** | **Expected Output** | **Actual Output** | **Result** |
| Click on selected video to be viewed | Selected video is played | Selected video is played | Pass |

|  |  |
| --- | --- |
| **Test Case ID** | STC-27 |
| **Test Case Name** | View Internet TV page |
| **Test Case Description** | Test for check that Internet TV page can be viewed if sign in as a member |
| **Test Responsibility** | Tester |
| **Tested by** | Chanyanuch Chaithongsri |
| **Test Date** | 18/05/2015 |
| **Prerequisites** | Sign In as a member |
| **Procedure steps** | 1. Sign In as a member 2. Click on “Internet TV” |

|  |  |  |  |
| --- | --- | --- | --- |
| **Result** | | | |
| **Input** | **Expected Output** | **Actual Output** | **Result** |
| Click on “Internet TV” link button | Internet TV page is shown | Internet TV page is shown | Pass |

|  |  |
| --- | --- |
| **Test Case ID** | STC-28 |
| **Test Case Name** | View Internet TV |
| **Test Case Description** | Test for check that Internet TV on Internet TV page can be viewed if sign in as a member |
| **Test Responsibility** | Tester |
| **Tested by** | Chanyanuch Chaithongsri |
| **Test Date** | 18/05/2015 |
| **Prerequisites** | Sign In as a member |
| **Procedure steps** | 1. Sign In as a member 2. Click on “Internet TV” 3. Click play button |

|  |  |  |  |
| --- | --- | --- | --- |
| **Result** | | | |
| **Input** | **Expected Output** | **Actual Output** | **Result** |
| Click on Play button | Internet TV is played | Internet TV is played | Pass |

|  |  |
| --- | --- |
| **Test Case ID** | STC-29 |
| **Test Case Name** | View Temple Map page |
| **Test Case Description** | Test for check that Temple Map page can be viewed if sign in as a member |
| **Test Responsibility** | Tester |
| **Tested by** | Chanyanuch Chaithongsri |
| **Test Date** | 18/05/2015 |
| **Prerequisites** | Sign In as a member |
| **Procedure steps** | 1. Sign In as a member 2. Click on “Temple Map” |

|  |  |  |  |
| --- | --- | --- | --- |
| **Result** | | | |
| **Input** | **Expected Output** | **Actual Output** | **Result** |
| Click on “Temple Map” link button | Temple Map page is shown | Temple Map page is shown | Pass |

|  |  |
| --- | --- |
| **Test Case ID** | STC-30 |
| **Test Case Name** | Get directions to temple from Google Maps |
| **Test Case Description** | Test for check that can be use Google Maps to get direction to temple if sign in as a member |
| **Test Responsibility** | Tester |
| **Tested by** | Chanyanuch Chaithongsri |
| **Test Date** | 18/05/2015 |
| **Prerequisites** | Sign in as a member |
| **Procedure steps** | 1. Sign In as a member 2. Click on “Temple Map” 3. View the direction on Google Maps |

|  |  |  |  |
| --- | --- | --- | --- |
| **Result** | | | |
| **Input** | **Expected Output** | **Actual Output** | **Result** |
| View the direction on Google Maps | Google Maps show the direction to temple | Google Maps show the direction to temple | Pass |

|  |  |
| --- | --- |
| **Test Case ID** | STC-31 |
| **Test Case Name** | View Contact page |
| **Test Case Description** | Test for check that Contact page can be viewed if sign in as a member |
| **Test Responsibility** | Tester |
| **Tested by** | Chanyanuch Chaithongsri |
| **Test Date** | 18/05/2015 |
| **Prerequisites** | Sign In as a member |
| **Procedure steps** | 1. Sign In as a member 2. Click on “Contact” |

|  |  |  |  |
| --- | --- | --- | --- |
| **Result** | | | |
| **Input** | **Expected Output** | **Actual Output** | **Result** |
| Click on “Contact” link button | Contact page is shown | Contact page is shown | Pass |

|  |  |
| --- | --- |
| **Test Case ID** | STC-32 |
| **Test Case Name** | Contact Temple by email |
| **Test Case Description** | Test for check that can be send email to contact temple by using contact form if sign in as a member |
| **Test Responsibility** | Tester |
| **Tested by** | Chanyanuch Chaithongsri |
| **Test Date** | 18/05/2015 |
| **Prerequisites** | Sign-in as a member |
| **Procedure steps** | 1. Sign in as a member 2. Click on “Contact” 3. Input contact information in contact form 4. Click on “Submit” |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Result** | | | | |
| **Input** | **Expected Output** | | **Actual Output** | **Result** |
| **Contact Name** = “Chanyanuch”  **Contact Email** = “chanyanuch.ch.025@gmail.com”  **Contact Topic** = “Contact Temple”  **Contact Detail** = “I am contact you” | Email has been send to temple | | Email has been send to temple | Pass |
| **Contact Name** = “Chanyanuch”  **Contact Email** = “se552115015@vr.camt.info”  **Contact Topic** = “Contact Temple Again”  **Contact Detail** = “I am contact you again” | Email has been send to temple | | Email has been send to temple | Pass |
| **Test Case ID** | | STC-33 | | |
| **Test Case Name** | | View Dhamma Board | | |
| **Test Case Description** | | Test for check that Dhamma Board can be viewed if sign in as a member | | |
| **Test Responsibility** | | Tester | | |
| **Tested by** | | Chanyanuch Chaithongsri | | |
| **Test Date** | | 18/05/2015 | | |
| **Prerequisites** | | Sign In as a member | | |
| **Procedure steps** | | 1. Sign In as a member 2. Click on “Dhamma Board” | | |

|  |  |  |  |
| --- | --- | --- | --- |
| **Result** | | | |
| **Input** | **Expected Output** | **Actual Output** | **Result** |
| Click on “Dhamma Board” link button | Dhamma Board page is shown | Dhamma Board page is shown | Pass |

|  |  |
| --- | --- |
| **Test Case ID** | STC-34 |
| **Test Case Name** | Post question on Dhamma Board |
| **Test Case Description** | Test for check that can post question on Dhamma Board if sign in as a member |
| **Test Responsibility** | Tester |
| **Tested by** | Chanyanuch Chaithongsri |
| **Test Date** | 18/05/2015 |
| **Prerequisites** | Sign In as a member |
| **Procedure steps** | 1. Sign In as a member 2. Click on “Dhamma Baord” 3. Click on “Post Question” 4. Input question 5. Click submit |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Result** | | | | |
| **Input** | **Expected Output** | | **Actual Output** | **Result** |
| **Question** = “How can I go to temple ?” | Question has been send to temple | | Question has been send to temple | Pass |
| **Test Case ID** | | STC-35 | | |
| **Test Case Name** | | Edit Personal Detail | | |
| **Test Case Description** | | Test for check that member can edit their personal detail | | |
| **Test Responsibility** | | Tester | | |
| **Tested by** | | Chanyanuch Chaithongsri | | |
| **Test Date** | | 18/05/2015 | | |
| **Prerequisites** | | Sign In as a member | | |
| **Procedure steps** | | 1. Sign In as a member 2. Click on their username 3. Input new information 4. Click submit | | |

|  |  |  |  |
| --- | --- | --- | --- |
| **Result** | | | |
| **Input** | **Expected Output** | **Actual Output** | **Result** |
| **Old Password** = “member1”  **New Password** = “member11”  **Re-New** = “member11” | New password has been save | New password has been save | Pass |

|  |  |
| --- | --- |
| **Test Case ID** | STC-36 |
| **Test Case Name** | Sign in as Admin |
| **Test Case Description** | Test for check that can sign in as Admin |
| **Test Responsibility** | Tester |
| **Tested by** | Chanyanuch Chaithongsri |
| **Test Date** | 18/05/2015 |
| **Prerequisites** | No sign in |
| **Procedure steps** | 1. Browse for Admin sign in page 2. Input Admin username, password 3. Click submit |

|  |  |  |  |
| --- | --- | --- | --- |
| **Result** | | | |
| **Input** | **Expected Output** | **Actual Output** | **Result** |
| **User Name** = “admin”  **Password** = “admin1234” | Index managing page is shown | Index managing page is shown | Pass |

|  |  |
| --- | --- |
| **Test Case ID** | STC-37 |
| **Test Case Name** | Admin Sign out |
| **Test Case Description** | Test for check that Admin can sign out |
| **Test Responsibility** | Tester |
| **Tested by** | Chanyanuch Chaithongsri |
| **Test Date** | 18/05/2015 |
| **Prerequisites** | Sign in as Admin |
| **Procedure steps** | 1. Sign in as Admin 2. Click “Sign out” button |

|  |  |  |  |
| --- | --- | --- | --- |
| **Result** | | | |
| **Input** | **Expected Output** | **Actual Output** | **Result** |
| Click on “Sign out” link button | Sign out and redirect to Admin Sign in page | Sign out and redirect to Admin Sign in page | Pass |

|  |  |
| --- | --- |
| **Test Case ID** | STC-38 |
| **Test Case Name** | View Index managing page |
| **Test Case Description** | Test for check that Admin can view Index managing page |
| **Test Responsibility** | Tester |
| **Tested by** | Chanyanuch Chaithongsri |
| **Test Date** | 18/05/2015 |
| **Prerequisites** | Sign in as Admin |
| **Procedure steps** | 1. Sign in as Admin 2. Click on “Index managing” button |

|  |  |  |  |
| --- | --- | --- | --- |
| **Result** | | | |
| **Input** | **Expected Output** | **Actual Output** | **Result** |
| Click on “Index managing” link button | Index managing page is shown | Index managing page is shown | Pass |

|  |  |
| --- | --- |
| **Test Case ID** | STC-39 |
| **Test Case Name** | View Temple Background managing page |
| **Test Case Description** | Test for check that Admin can view Temple Background managing page |
| **Test Responsibility** | Tester |
| **Tested by** | Chanyanuch Chaithongsri |
| **Test Date** | 18/05/2015 |
| **Prerequisites** | Sign in as Admin |
| **Procedure steps** | 1. Sign in as Admin 2. Click on “Temple Background managing” button |

|  |  |  |  |
| --- | --- | --- | --- |
| **Result** | | | |
| **Input** | **Expected Output** | **Actual Output** | **Result** |
| Click on “Temple Background managing” link button | Temple Background managing page is shown | Temple Background managing page is shown | Pass |

|  |  |
| --- | --- |
| **Test Case ID** | STC-40 |
| **Test Case Name** | View News and Activities managing page |
| **Test Case Description** | Test for check that Admin can view News and Activities managing page |
| **Test Responsibility** | Tester |
| **Tested by** | Chanyanuch Chaithongsri |
| **Test Date** | 18/05/2015 |
| **Prerequisites** | Sign in as Admin |
| **Procedure steps** | 1. Sign in as Admin 2. Click on “News and Actitvities managing” button |

|  |  |  |  |
| --- | --- | --- | --- |
| **Result** | | | |
| **Input** | **Expected Output** | **Actual Output** | **Result** |
| Click on “News and Activities managing” link button | News and Activities managing page is shown | News and Activities managing page is shown | Pass |

|  |  |
| --- | --- |
| **Test Case ID** | STC-41 |
| **Test Case Name** | View Dhamma Board managing page |
| **Test Case Description** | Test for check that Admin can view Dhamma Board managing page |
| **Test Responsibility** | Tester |
| **Tested by** | Chanyanuch Chaithongsri |
| **Test Date** | 18/05/2015 |
| **Prerequisites** | Sign in as Admin |
| **Procedure steps** | 1. Sign in as Admin 2. Click on “Dhamma Board managing” button |

|  |  |  |  |
| --- | --- | --- | --- |
| **Result** | | | |
| **Input** | **Expected Output** | **Actual Output** | **Result** |
| Click on “Dhamma Board managing” link button | Dhamma Board managing page is shown | Dhamma Board managing page is shown | Pass |

|  |  |
| --- | --- |
| **Test Case ID** | STC-42 |
| **Test Case Name** | View Dhamma Content managing page |
| **Test Case Description** | Test for check that Admin can view Dhamma Content managing page |
| **Test Responsibility** | Tester |
| **Tested by** | Chanyanuch Chaithongsri |
| **Test Date** | 18/05/2015 |
| **Prerequisites** | Sign in as Admin |
| **Procedure steps** | 1. Sign in as Admin 2. Click on “Dhamma Content managing” button |

|  |  |  |  |
| --- | --- | --- | --- |
| **Result** | | | |
| **Input** | **Expected Output** | **Actual Output** | **Result** |
| Click on “Dhamma Content managing” link button | Dhamma Content managing page is shown | Dhamma Content managing page is shown | Pass |

|  |  |
| --- | --- |
| **Test Case ID** | STC-43 |
| **Test Case Name** | View Admin Sign in page |
| **Test Case Description** | Test for check that Admin can view Admin Sign in page |
| **Test Responsibility** | Tester |
| **Tested by** | Chanyanuch Chaithongsri |
| **Test Date** | 18/05/2015 |
| **Prerequisites** | No Sign in |
| **Procedure steps** | 1. Browse for Admin Sign in page |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Result** | | | | |
| **Input** | **Expected Output** | | **Actual Output** | **Result** |
| Browse for “http://pathammapirom.  azurewebsites.net/Admin/Login” | Admin sign in page is shown | | Admin sign in page is shown | Pass |
| **Test Case ID** | | STC-44 | | |
| **Test Case Name** | | View Gallery managing page | | |
| **Test Case Description** | | Test for check that Admin can view Gallery managing page | | |
| **Test Responsibility** | | Tester | | |
| **Tested by** | | Chanyanuch Chaithongsri | | |
| **Test Date** | | 18/05/2015 | | |
| **Prerequisites** | | Sign in as Admin | | |
| **Procedure steps** | | 1. Sign in as Admin 2. Click on “Gallery managing” button | | |

|  |  |  |  |
| --- | --- | --- | --- |
| **Result** | | | |
| **Input** | **Expected Output** | **Actual Output** | **Result** |
| Click on “Gallery managing” link button | Gallery managing page is shown | Gallery managing page is shown | Pass |

|  |  |
| --- | --- |
| **Test Case ID** | STC-45 |
| **Test Case Name** | View Activities Video Clips managing page |
| **Test Case Description** | Test for check that Admin can view Activities Video Clips managing page |
| **Test Responsibility** | Tester |
| **Tested by** | Chanyanuch Chaithongsri |
| **Test Date** | 18/05/2015 |
| **Prerequisites** | Sign in as Admin |
| **Procedure steps** | 1. Sign in as Admin 2. Click on “Activities Video Clips managing” button |

|  |  |  |  |
| --- | --- | --- | --- |
| **Result** | | | |
| **Input** | **Expected Output** | **Actual Output** | **Result** |
| Click on “Activities Video Clips managing” link button | Activities Video Clips managing page is shown | Activities Video Clips managing page is shown | Pass |

|  |  |
| --- | --- |
| **Test Case ID** | STC-46 |
| **Test Case Name** | Add new image on Index managing page |
| **Test Case Description** | Test for check that Admin can add new image to Image Presentation in Index page |
| **Test Responsibility** | Tester |
| **Tested by** | Chanyanuch Chaithongsri |
| **Test Date** | 18/05/2015 |
| **Prerequisites** | Sign in as Admin |
| **Procedure steps** | 1. Sign in as Admin 2. Click on “Index managing” button 3. Click “Add new image to slide” 4. Input image link 5. Click submit |

|  |  |  |  |
| --- | --- | --- | --- |
| **Result** | | | |
| **Input** | **Expected Output** | **Actual Output** | **Result** |
| **Image link = “**http://i.imgur.com/WzuNluz.jpg**”** | New image has been added to image slide and show | New image has been added to image slide and show | Pass |

|  |  |
| --- | --- |
| **Test Case ID** | STC-47 |
| **Test Case Name** | Delete selected image on Index managing page |
| **Test Case Description** | Test for check that Admin can delete selected image on Image Presentation in Index page |
| **Test Responsibility** | Tester |
| **Tested by** | Chanyanuch Chaithongsri |
| **Test Date** | 18/05/2015 |
| **Prerequisites** | Sign in as Admin |
| **Procedure steps** | 1. Sign in as Admin 2. Click on “Index managing” button 3. Select image to be deleted 4. Click delete |

|  |  |  |  |
| --- | --- | --- | --- |
| **Result** | | | |
| **Input** | **Expected Output** | **Actual Output** | **Result** |
| Click “delete” on selected image | Selected image has been deleted | Selected image has been deleted | Pass |

|  |  |
| --- | --- |
| **Test Case ID** | STC-48 |
| **Test Case Name** | Edit Temple Background details |
| **Test Case Description** | Test for check that Admin edit Temple Background details in Temple Background managing page |
| **Test Responsibility** | Tester |
| **Tested by** | Chanyanuch Chaithongsri |
| **Test Date** | 18/05/2015 |
| **Prerequisites** | Sign in as Admin |
| **Procedure steps** | 1. Sign in as Admin 2. Click on “Temple Background managing” button 3. Click “Edit” 4. Edit Temple Background details 5. Click submit |

|  |  |  |  |
| --- | --- | --- | --- |
| **Result** | | | |
| **Input** | **Expected Output** | **Actual Output** | **Result** |
| Edit Temple Background details | Temple Background details has been change | Temple Background details has been change | Pass |

|  |  |
| --- | --- |
| **Test Case ID** | STC-49 |
| **Test Case Name** | Add recent news and activities |
| **Test Case Description** | Test for check that Admin can add recent news and activities in News and Activities managing page |
| **Test Responsibility** | Tester |
| **Tested by** | Chanyanuch Chaithongsri |
| **Test Date** | 18/05/2015 |
| **Prerequisites** | Sign in as Admin |
| **Procedure steps** | 1. Sign in as Admin 2. Click on “News and Activities managing” button 3. Click “Add News and Activities” 4. Input News and Activities information 5. Click submit |

|  |  |  |  |
| --- | --- | --- | --- |
| **Result** | | | |
| **Input** | **Expected Output** | **Actual Output** | **Result** |
| **News and Activities Name** = “งานแห่พระยกขาว”  **News and Activities details** = “เป็นงานที่จัดขึ้นเป็นประจำทุกปี”  **News and Activities Date** = “30/05/2558”  **News and Activities Time** = “10.00-14.00”  **News and Activities Place** = “วัดป่าธรรมาภิรมย์” | Recent News and Activities has been added | Recent News and Activities has been added | Pass |

|  |  |
| --- | --- |
| **Test Case ID** | STC-50 |
| **Test Case Name** | Delete selected news and activities |
| **Test Case Description** | Test for check that Admin can delete selected news and activities in News and Activities managing page |
| **Test Responsibility** | Tester |
| **Tested by** | Chanyanuch Chaithongsri |
| **Test Date** | 18/05/2015 |
| **Prerequisites** | Sign in as Admin |
| **Procedure steps** | 1. Sign in as Admin 2. Click on “News and Activities managing” button 3. Selected news and activities topic to be deleted 4. Click “delete” |

|  |  |  |  |
| --- | --- | --- | --- |
| **Result** | | | |
| **Input** | **Expected Output** | **Actual Output** | **Result** |
| Click “delete” on selected News and Activities topic | Selected News and Activities topic has been deleted | Selected News and Activities topic has been deleted | Pass |

|  |  |
| --- | --- |
| **Test Case ID** | STC-51 |
| **Test Case Name** | Edit detail of selected news and activities topic |
| **Test Case Description** | Test for check that Admin can edit details of selected news and activities topic in News and Activities managing page |
| **Test Responsibility** | Tester |
| **Tested by** | Chanyanuch Chaithongsri |
| **Test Date** | 18/05/2015 |
| **Prerequisites** | Sign in as Admin |
| **Procedure steps** | 1. Sign in as Admin 2. Click on “News and Activities managing” button 3. Selected news and activities topic to be edited 4. Edit details of selected news and activities topic 5. Click submit |

|  |  |  |  |
| --- | --- | --- | --- |
| **Result** | | | |
| **Input** | **Expected Output** | **Actual Output** | **Result** |
| Edit details of selected news and activities topic | Details of selected news and activities topic has been edited | Details of selected news and activities topic has been edited | Pass |

|  |  |
| --- | --- |
| **Test Case ID** | STC-52 |
| **Test Case Name** | Reply member’s question |
| **Test Case Description** | Test for check that Admin can reply member’s question that member post on Dhamma Board |
| **Test Responsibility** | Tester |
| **Tested by** | Chanyanuch Chaithongsri |
| **Test Date** | 18/05/2015 |
| **Prerequisites** | Sign in as Admin |
| **Procedure steps** | 1. Sign in as Admin 2. Click on “Dhamma Baord managing” button 3. Select question to answer 4. Click “Answer” 5. Input Answer 6. Click submit |

|  |  |  |  |
| --- | --- | --- | --- |
| **Result** | | | |
| **Input** | **Expected Output** | **Actual Output** | **Result** |
| **Answer** = “Go to Temple Map” | The answer is show in Dhamma Board page | The answer is show in Dhamma Board page | Pass |

|  |  |
| --- | --- |
| **Test Case ID** | STC-53 |
| **Test Case Name** | Add new image to Gallery page |
| **Test Case Description** | Test for check that Admin can add new image to Gallery page |
| **Test Responsibility** | Tester |
| **Tested by** | Chanyanuch Chaithongsri |
| **Test Date** | 18/05/2015 |
| **Prerequisites** | Sign in as Admin |
| **Procedure steps** | 1. Sign in as Admin 2. Click on “Gallery managing” button 3. Click “Add new image” 4. Input image link 5. Click submit |

|  |  |  |  |
| --- | --- | --- | --- |
| **Result** | | | |
| **Input** | **Expected Output** | **Actual Output** | **Result** |
| **Image link** = “http://i.imgur.com/WzuNluz.jpg” | Image has been added to Gallery | Image has been added to Gallery | Pass |

|  |  |
| --- | --- |
| **Test Case ID** | STC-54 |
| **Test Case Name** | Delete selected image in Gallery page |
| **Test Case Description** | Test for check that Admin can delete selected image in Gallery page |
| **Test Responsibility** | Tester |
| **Tested by** | Chanyanuch Chaithongsri |
| **Test Date** | 18/05/2015 |
| **Prerequisites** | Sign in as Admin |
| **Procedure steps** | 1. Sign in as Admin 2. Click on “Gallery managing” button 3. Select image to delete 4. Click “delete” |

|  |  |  |  |
| --- | --- | --- | --- |
| **Result** | | | |
| **Input** | **Expected Output** | **Actual Output** | **Result** |
| Click “delete” on selected image | Selected image has been deleted | Selected image has been deleted | Pass |

|  |  |
| --- | --- |
| **Test Case ID** | STC-55 |
| **Test Case Name** | Add new video to Activities Video Clips page |
| **Test Case Description** | Test for check that Admin can add new video to Activities Video Clips page |
| **Test Responsibility** | Tester |
| **Tested by** | Chanyanuch Chaithongsri |
| **Test Date** | 18/05/2015 |
| **Prerequisites** | Sign in as Admin |
| **Procedure steps** | 1. Sign in as Admin 2. Click on “Activities Video Clips managing” button 3. Click “Add New Video” 4. Input the video name and video embed link 5. Click submit |

|  |  |  |  |
| --- | --- | --- | --- |
| **Result** | | | |
| **Input** | **Expected Output** | **Actual Output** | **Result** |
| **Video Name** = “พิธีทอดกฐินสามัคคี”  **Video Embed link** = “https://www.youtube.com/  embed/watch?v=o953FlKnNW0” | Video has been added | Video has been added | Pass |

|  |  |
| --- | --- |
| **Test Case ID** | STC-62 |
| **Test Case Name** | Delete selected video in Activities Video Clips page |
| **Test Case Description** | Test for check that Admin can delete selected video in Activities Video Clips page |
| **Test Responsibility** | Tester |
| **Tested by** | Chanyanuch Chaithongsri |
| **Test Date** | 18/05/2015 |
| **Prerequisites** | Sign in as Admin |
| **Procedure steps** | 1. Sign in as Admin 2. Click on “Activities Video Clips managing” button 3. Select video to delete 4. Click “delete” |

|  |  |  |  |
| --- | --- | --- | --- |
| **Result** | | | |
| **Input** | **Expected Output** | **Actual Output** | **Result** |
| Click on “Index managing” link button | Index managing page is shown | Index managing page is shown | Pass |

|  |  |
| --- | --- |
| **Test Case ID** | STC-57 |
| **Test Case Name** | Send email to member |
| **Test Case Description** | Test for check that Admin can send email to member |
| **Test Responsibility** | Tester |
| **Tested by** | Chanyanuch Chaithongsri |
| **Test Date** | 18/05/2015 |
| **Prerequisites** | - |
| **Procedure steps** | 1. Open temple email address 2. Send email to member 3. Click submit |

|  |  |  |  |
| --- | --- | --- | --- |
| **Result** | | | |
| **Input** | **Expected Output** | **Actual Output** | **Result** |
| Send email to member | Email has been send to member | Email has been send to member | Pass |

**Non Functional Testing**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Performance** | | | | |
| **Response Time** | **Criteria** | **Result** | **Accepted** | **Rejected** |
| Application loading within 5 second. |  |  |  |
| Refresh Time within 5 second. |  |  |  |
| **Process Time** | **Criteria** | **Result** | **Accepted** | **Rejected** |
| Update information loading within 5 second. |  |  |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Capacity** | | | | |
| **Storage** | **Criteria** | **Result** | **Accepted** | **Rejected** |
| At least 1 Gb be able to store |  |  |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Recover** | | | | |
| **Recovery Process** | **Criteria** | **Result** | **Accepted** | **Rejected** |
| Recovery refresh time within 10 second |  |  |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Compatibility** | | | | |
| **Compatibility with shared application** | **Criteria** | **Result** | **Accepted** | **Rejected** |
| Google Maps can be called to use |  |  |  |
| **Compatibility on different operating systems** | **Criteria** | **Result** | **Accepted** | **Rejected** |
| Able to be running on Windows and Mac operating system |  |  |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Usability** | | | | |
| **Look and feel standards** | **Criteria** | **Result** | **Accepted** | **Rejected** |
| Website layout and flow be able to understand within 3 minutes |  |  |  |
| Website color appropriate with the concept of website |  |  |  |
| Website user interface is feel easy to use it |  |  |  |
| **Internationalization / localization requirement** | **Criteria** | **Result** | **Accepted** | **Rejected** |
| Thai people can be understand in every page of website |  |  |  |

**User Acceptance Test**

If average score is more than 3.5 means that user accept that page.

**Index Page**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Criteria** | **1** | **2** | **3** | **4** | **5** | **Remarks** |
| 1. Index page shall work properly as defined requirement specification. |  |  |  |  |  |  |
| 1. Image presentation show images that it was set to be shown. |  |  |  |  |  |  |

|  |  |
| --- | --- |
| **Comment** |  |

**Temple Background Page**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Criteria** | **1** | **2** | **3** | **4** | **5** | **Remarks** |
| 1. Temple Background page shall work properly as defined requirement specification. |  |  |  |  |  |  |
| 1. Show temple background detail. |  |  |  |  |  |  |

|  |  |
| --- | --- |
| **Comment** |  |

**News and Activities Page**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Criteria** | **1** | **2** | **3** | **4** | **5** | **Remarks** |
| 1. News and Activities page shall work properly as defined requirement specification. |  |  |  |  |  |  |
| 1. Show more detail of selected News and Activities topic. |  |  |  |  |  |  |

|  |  |
| --- | --- |
| **Comment** |  |

**Dhamma Content Page**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Criteria** | **1** | **2** | **3** | **4** | **5** | **Remarks** |
| 1. Dhamma Content page shall work properly as defined requirement specification. |  |  |  |  |  |  |
| 1. Show more detail of selected Dhamma content topic. |  |  |  |  |  |  |

|  |  |
| --- | --- |
| **Comment** |  |

**Temple Map Page**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Criteria** | **1** | **2** | **3** | **4** | **5** | **Remarks** |
| 1. Temple Map page shall work properly as defined requirement specification. |  |  |  |  |  |  |
| 1. Google Maps on Temple Map shall get the directions to Temple |  |  |  |  |  |  |

|  |  |
| --- | --- |
| **Comment** |  |

**Contact Page**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Criteria** | **1** | **2** | **3** | **4** | **5** | **Remarks** |
| 1. Contact page shall work properly as defined requirement specification. |  |  |  |  |  |  |
| 1. Show contact detail to contact Temple. |  |  |  |  |  |  |
| 1. Send email to contact temple by input information into Contact Form on Contact page. |  |  |  |  |  |  |

|  |  |
| --- | --- |
| **Comment** |  |

**Register Page**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Criteria** | **1** | **2** | **3** | **4** | **5** | **Remarks** |
| 1. Register page shall work properly as defined requirement specification. |  |  |  |  |  |  |
| 1. Provide Register Form to register. |  |  |  |  |  |  |
| 1. Register to be Member. |  |  |  |  |  |  |

|  |  |
| --- | --- |
| **Comment** |  |

**Sign In Page**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Criteria** | **1** | **2** | **3** | **4** | **5** | **Remarks** |
| 1. Sign In page shall work properly as defined requirement specification. |  |  |  |  |  |  |
| 1. Provide Sign In Form to sign in. |  |  |  |  |  |  |
| 1. Sign In to the system. |  |  |  |  |  |  |
| 1. Sign out. |  |  |  |  |  |  |

|  |  |
| --- | --- |
| **Comment** |  |

**Gallery Page**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Criteria** | **1** | **2** | **3** | **4** | **5** | **Remarks** |
| 1. Gallery page shall work properly as defined requirement specification. |  |  |  |  |  |  |
| 1. Image on Gallery page can viewed in full-size. |  |  |  |  |  |  |

|  |  |
| --- | --- |
| **Comment** |  |

**Activities Video Clips Page**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Criteria** | **1** | **2** | **3** | **4** | **5** | **Remarks** |
| 1. Activities Video Clips page shall work properly as defined requirement specification. |  |  |  |  |  |  |
| 1. Video on Activities Video Clips page can played. |  |  |  |  |  |  |

|  |  |
| --- | --- |
| **Comment** |  |

**Dhamma Board**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Criteria** | **1** | **2** | **3** | **4** | **5** | **Remarks** |
| 1. Dhamma Board shall work properly as defined requirement specification. |  |  |  |  |  |  |
| 1. Provide Question form. |  |  |  |  |  |  |
| 1. Post question on Dhamma Board. |  |  |  |  |  |  |

|  |  |
| --- | --- |
| **Comment** |  |

**Internet TV Page**

\*\* Have only the content layout to be added in the future

**Overview**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Criteria** | **1** | **2** | **3** | **4** | **5** | **Remarks** |
| 1. User Interface Design of web application meet user needs. |  |  |  |  |  |  |
| 1. Web application easy to understand |  |  |  |  |  |  |
| 1. The function of web application is cover all requirement. |  |  |  |  |  |  |

|  |  |
| --- | --- |
| **Comment** |  |

**Only Admin Part**

**Index Managing Page**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Criteria** | **1** | **2** | **3** | **4** | **5** | **Remarks** |
| 1. Index managing page shall work properly as defined requirement specification. |  |  |  |  |  |  |
| 1. Adding new image to Image Presentation on Index page. |  |  |  |  |  |  |
| 1. Deleting selected image in Image Presentation on Index page. |  |  |  |  |  |  |

|  |  |
| --- | --- |
| **Comment** |  |

**Temple Background Managing Page**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Criteria** | **1** | **2** | **3** | **4** | **5** | **Remarks** |
| 1. Temple Background managing page shall work properly as defined requirement specification. |  |  |  |  |  |  |
| 1. Editing Temple Background Detail. |  |  |  |  |  |  |

|  |  |
| --- | --- |
| **Comment** |  |

**News and Activities Managing Page**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Criteria** | **1** | **2** | **3** | **4** | **5** | **Remarks** |
| 1. News and Activities managing page shall work properly as defined requirement specification. |  |  |  |  |  |  |
| 1. Adding recent news and activities on News and Activities managing page. |  |  |  |  |  |  |
| 1. Deleting selected news and activities on News and Activities managing page. |  |  |  |  |  |  |
| 1. Editing detail of selected news and activities topic. |  |  |  |  |  |  |

|  |  |
| --- | --- |
| **Comment** |  |

**Dhamma Content Managing Page**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Criteria** | **1** | **2** | **3** | **4** | **5** | **Remarks** |
| 1. Dhamma Content managing page shall work properly as defined requirement specification. |  |  |  |  |  |  |
| 1. Adding new content on Dhamma Content managing page. |  |  |  |  |  |  |
| 1. Editing selected Dhamma content on Dhamma Content managing page. |  |  |  |  |  |  |
| 1. Deleting selected Dhamma Content on Dhamma Content managing page. |  |  |  |  |  |  |

|  |  |
| --- | --- |
| **Comment** |  |

**Temple Map Page**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Criteria** | **1** | **2** | **3** | **4** | **5** | **Remarks** |
| 1. Dhamma Content page shall work properly as defined requirement specification. |  |  |  |  |  |  |
| 1. Show more detail of selected Dhamma content topic. |  |  |  |  |  |  |

|  |  |
| --- | --- |
| **Comment** |  |

**Admin Sign in Page**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Criteria** | **1** | **2** | **3** | **4** | **5** | **Remarks** |
| 1. Admin Sign In page shall work properly as defined requirement specification. |  |  |  |  |  |  |
| 1. Admin can Sign In. |  |  |  |  |  |  |
| 1. Admin can Sign out. |  |  |  |  |  |  |

|  |  |
| --- | --- |
| **Comment** |  |

**Gallery Managing Page**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Criteria** | **1** | **2** | **3** | **4** | **5** | **Remarks** |
| 1. Gallery manaing page shall work properly as defined requirement specification. |  |  |  |  |  |  |
| 1. Adding new image to Gallery page. |  |  |  |  |  |  |
| 1. Deleting selected image on Gallery page. |  |  |  |  |  |  |

|  |  |
| --- | --- |
| **Comment** |  |

**Activities Video Clips Managing Page**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Criteria** | **1** | **2** | **3** | **4** | **5** | **Remarks** |
| 1. Activities Video Clips Managing page shall work properly as defined requirement specification. |  |  |  |  |  |  |
| 1. Adding new video on Activities Video Clips managing page. |  |  |  |  |  |  |
| 1. Deleting selected video on Activities Video Clips page. |  |  |  |  |  |  |

|  |  |
| --- | --- |
| **Comment** |  |

**Dhamma Board Managing Page**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Criteria** | **1** | **2** | **3** | **4** | **5** | **Remarks** |
| 1. Dhamma Board managing page shall work properly as defined requirement specification. |  |  |  |  |  |  |
| 1. Reply member’s question on Dhamma Board managing page. |  |  |  |  |  |  |

|  |  |
| --- | --- |
| **Comment** |  |

**Contact Managing Page**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Criteria** | **1** | **2** | **3** | **4** | **5** | **Remarks** |
| 1. Contact managing page shall work properly as defined requirement specification. |  |  |  |  |  |  |
| 1. Editing Contact detail on Contact page. |  |  |  |  |  |  |
| 1. Sending email to member. |  |  |  |  |  |  |

|  |  |
| --- | --- |
| **Comment** |  |

**Admin Part Overview**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Criteria** | **1** | **2** | **3** | **4** | **5** | **Remarks** |
| 1. User Interface Design of Admin Management part meet user needs. |  |  |  |  |  |  |
| 1. Admin management part easy to use and understand. |  |  |  |  |  |  |
| 1. The function of Admin management part is cover all requirement. |  |  |  |  |  |  |

|  |  |
| --- | --- |
| **Comment** |  |

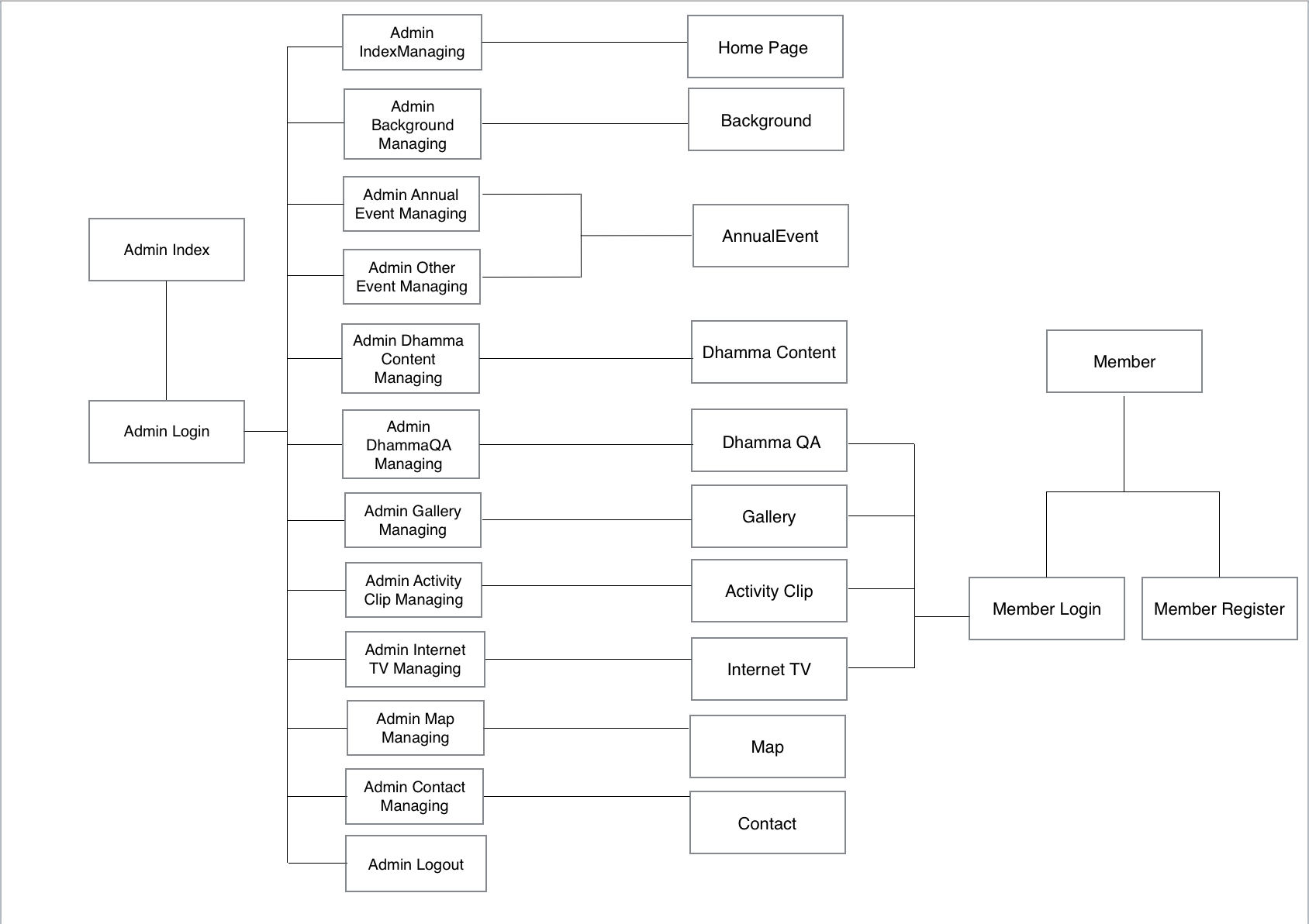
INTEGRATION TEST

Related Classes

|  |  |  |
| --- | --- | --- |
| **ID** | **Class** | **Class** |
| IT-01 | Admin Index | Admin Login |
| IT-02 | Admin Login | Admin IndexManaging |
| IT-03 | Admin Login | Admin Background Managing |
| IT-04 | Admin Login | Admin AnnualEvent Managing |
| IT-05 | Admin Login | Admin OtherEvent Managing |
| IT-06 | Admin Login | Admin DhammaQA Managing |
| IT-07 | Admin Login | Admin Gallery Managing |
| IT-08 | Admin Login | Admin ActivityClip Managing |
| IT-09 | Admin Login | Admin Logout |
| IT-10 | Admin IndexManaging | Home Pgae |
| IT-11 | Admin Background Managing | Background |
| IT-12 | Admin AnnualEvent Managing | Annual Event |
| IT-13 | Admin OtherEvent Managing | Annual Event |
| IT-14 | Admin DhammaQA Managing | Dhamma QA |
| IT-15 | Admin Gallery Managing | Gallery |
| IT-16 | Admin ActivityClip Managing | Activity Clip |
| IT-17 | Admin InternetTV Managing | Internet TV |
| IT-18 | Admin Map Managing | Map |
| IT-19 | Admin Contact Managing | Contact |
| IT-20 | Member | Member Register |
| IT-21 | Member | Member Login |
| IT-22 | Member Login | Dhamma QA |
| IT-23 | Member Login | Gallery |

|  |  |  |
| --- | --- | --- |
| **ID** | **Class** | **Class** |
| IT-24 | Member Login | Activity Clip |
| IT-25 | Member Login | Internet TV |

**[INTEGRATION TEST]**

Architecture

**[INTEGRATION TEST]**

**Integration Test** IT-01 (Admin Index, Admin Login)

**Test Date** 17 May 2015

**Description**

|  |  |  |  |
| --- | --- | --- | --- |
| **ID** | **Input Specification** | **Output Specification** | **Environment Needs** |
| 1 | Username : admin | Check if the correct username and password, the website will direct to Index Page |  |
| 2 | Password : admin1234 | Check if the correct username and password, the website will direct to Index Page |  |

**Integration Test** IT-02 (Admin Login, Admin Index Managing)

**Test Date** 17 May 2015

**Description**

|  |  |  |  |
| --- | --- | --- | --- |
| **ID** | **Input Specification** | **Output Specification** | **Environment Needs** |
| 1 | Insert new image | New image on screen | Admin Login |
| 2 | Delete image | Image was deleted | Admin Login |

**Integration Test** IT-03 (Admin Login, Admin Background Managing)

**Test Date** 17 May 2015

**Description**

|  |  |  |  |
| --- | --- | --- | --- |
| **ID** | **Input Specification** | **Output Specification** | **Environment Needs** |
| 1 | Insert new background | New background on screen | Admin Login |
| 2 | Edit background | Background was edited | Admin Login |

**Integration Test** IT-04 (Admin Login, Admin AnnualEvent Managing)

**Test Date** 17 May 2015

**Description**

|  |  |  |  |
| --- | --- | --- | --- |
| **ID** | **Input Specification** | **Output Specification** | **Environment Needs** |
| 1 | Insert new annual event | New annual event on screen | Admin Login |
| 2 | Edit annual event | Annual event was edited | Admin Login |
| 3 | Delete annual event | Annual event was deleted | Admin Login |

**Integration Test** IT-05 (Admin Login, Admin OtherEvent Managing)

**Test Date** 17 May 2015

**Description**

|  |  |  |  |
| --- | --- | --- | --- |
| **ID** | **Input Specification** | **Output Specification** | **Environment Needs** |
| 1 | Insert new other event | New other event on screen | Admin Login |
| 2 | Edit other event | Other event was edited | Admin Login |
| 3 | Delete other event | Other event was deleted | Admin Login |

**Integration Test** IT-06 (Admin Login, Admin DhammaQA Managing)

**Test Date** 17 May 2015

**Description**

|  |  |  |  |
| --- | --- | --- | --- |
| **ID** | **Input Specification** | **Output Specification** | **Environment Needs** |
| 1 | Answer the question | Provide answer | Admin Login |
| 2 | Delete the question | Question was deleted | Admin Login |

**Integration Test** IT-07 (Admin Login, Admin Gallery Managing)

**Test Date** 17 May 2015

**Description**

|  |  |  |  |
| --- | --- | --- | --- |
| **ID** | **Input Specification** | **Output Specification** | **Environment Needs** |
| 1 | Insert image | New image in gallery | Admin Login |
| 2 | Delete image | image was deleted | Admin Login |

**Integration Test** IT-08 (Admin Login, Admin ActivityClip Managing)

**Test Date** 17 May 2015

**Description**

|  |  |  |  |
| --- | --- | --- | --- |
| **ID** | **Input Specification** | **Output Specification** | **Environment Needs** |
| 1 | Insert activity clip | New activity clip on screen | Admin Login |
| 2 | Delete activity clip | Activity Clip was deleted | Admin Login |

**Integration Test** IT-09 (Admin Login, Admin Logout)

**Test Date** 17 May 2015

**Description**

|  |  |  |  |
| --- | --- | --- | --- |
| **ID** | **Input Specification** | **Output Specification** | **Environment Needs** |
| 1 | Click “ออกจากระบบ” button | Log out of Admin and redirect to Index Page | Admin Login |

**Integration Test** IT-10 (Admin Index Managing, Home Page)

**Test Date** 17 May 2015

**Description**

|  |  |  |  |
| --- | --- | --- | --- |
| **ID** | **Input Specification** | **Output Specification** | **Environment Needs** |
| 1 | Managing home page | Home page was managed by Admin Index Managing | Admin Login |
| 2 | Image slide show | Slide show Wat Pa Thammapirom images |  |
| 3 | Click to other page | Enable to click |  |

**Integration Test** IT-11 (Admin Background Managing, Background)

**Test Date** 17 May 2015

**Description**

|  |  |  |  |
| --- | --- | --- | --- |
| **ID** | **Input Specification** | **Output Specification** | **Environment Needs** |
| 1 | Managing background page | Background page was managed by Admin Background Managing | Admin Login |
| 2 | Showing background detail | Show background of Wat Pa Thammapirom |  |
| 3 | Click to other page | Enable to click |  |

**Integration Test** IT-12 (Admin AnnualEvent Managing, Annual Event)

**Test Date** 17 May 2015

**Description**

|  |  |  |  |
| --- | --- | --- | --- |
| **ID** | **Input Specification** | **Output Specification** | **Environment Needs** |
| 1 | Managing annual event page | Annual Event page was managed by Admin AnnualEvent Managing | Admin Login |
| 2 | Showing each annual event | Show annual event on screen |  |
| 3 | Click to showing annual event detail | Enable and showing annual event detail |  |
| 4 | Click to other page | Enable to click |  |

**Integration Test** IT-13 (Admin Other Managing, Annual Event)

**Test Date** 17 May 2015

**Description**

|  |  |  |  |
| --- | --- | --- | --- |
| **ID** | **Input Specification** | **Output Specification** | **Environment Needs** |
| 1 | Managing other event page | Annual Event page was managed by Admin OtherEvent Managing | Admin Login |
| 2 | Showing each other event | Show other event on screen |  |
| 3 | Click to showing other event detail | Enable and showing other event detail |  |
| 4 | Click to other page | Enable to click |  |

**Integration Test** IT-14 (Admin DhammaQA Managing, Dhamma QA)

**Test Date** 17 May 2015

**Description**

|  |  |  |  |
| --- | --- | --- | --- |
| **ID** | **Input Specification** | **Output Specification** | **Environment Needs** |
| 1 | Managing dhamma qa page | Dhamma QA page was managed by Admin DhammaQA | Admin Login |
| 2 | Showing dharma question and answer | Show dharma questions and answers on screen | Member Login |
| 3 | Click to asking dhamma question | Enable to click | Member Login |
| 4 | Fill the question | Enable to Fill question | Member Login |
| 5 | Send the question | Enable to sending the question | Member Login |
| 6 | Click to other page | Enable to click |  |

**Integration Test** IT-15 (Admin Gallery Managing, Gallery)

**Test Date** 17 May 2015

**Description**

|  |  |  |  |
| --- | --- | --- | --- |
| **ID** | **Input Specification** | **Output Specification** | **Environment Needs** |
| 1 | Managing gallery page | Gallery was managed by Admin Gallery Managing | Admin Login |
| 2 | Showing images in gallery | Show images in gallery | Member Login |
| 3 | Showing the bigger size of image that padding by mouse icon | Show the bigger size of image when padding | Member Login |
| 4 | Click to other page | Enable to click |  |

**Integration Test** IT-16 (Admin ActivityClip Managing, Activity Clip)

**Test Date** 17 May 2015

**Description**

|  |  |  |  |
| --- | --- | --- | --- |
| **ID** | **Input Specification** | **Output Specification** | **Environment Needs** |
| 1 | Managing activity clip page | Activity Clip page was managed by Admin ActivityClip Managing | Admin Login |
| 2 | Showing activity clips | Show Clips of Wat Pa Thammapirom activity | Member Login |
| 3 | Play activity clip | Enable to play activity clips | Member Login |
| 4 | Click to other page | Enable to click |  |

**Integration Test** IT-17 (Admin InternetTV Managing, Internet TV)

**Test Date** 17 May 2015

**Description**

|  |  |  |  |
| --- | --- | --- | --- |
| **ID** | **Input Specification** | **Output Specification** | **Environment Needs** |
| 1 | Managing internet TV page | Internet TV page was managed by Admin Internet TV | Admin Login |
| 2 | Showing internet TV page | Show internet TV page | Member Login |
| 3 | Click to other page | Enable to click |  |

**Integration Test** IT-18 (Admin Map Managing, Map)

**Test Date** 17 May 2015

**Description**

|  |  |  |  |
| --- | --- | --- | --- |
| **ID** | **Input Specification** | **Output Specification** | **Environment Needs** |
| 1 | Managing map page | Map page was managed by Admin Map Managing | Admin Login |
| 2 | Showing map pin to Wat Pa Thammapirom | Show Map was pin the location of Wat Pa Thammapirom on screen |  |
| 3 | Click to other page | Enable to click |  |

**Integration Test** IT-19 (Admin Contact Managing, Contact)

**Test Date** 17 May 2015

**Description**

|  |  |  |  |
| --- | --- | --- | --- |
| **ID** | **Input Specification** | **Output Specification** | **Environment Needs** |
| 1 | Managing contact page | Contact page was managed by Admin Contact Managing | Admin Login |
| 2 | Showing contact detail to Wat Pa Thammapirom | Show  โทรศัพท์: 0-53-57-8514  โทรสาร: 0-53-57-8514  ติดต่อเพิ่มเติม  พระครูสังฆกิจจารักษ์: 08-1784-8323  คุณณัฐพชร์ เมธีภัคดิภูมิ (คุณนก): 08-1884-2392  ที่อยู่  เลขที่ ๙๒ หมู่ ๑๑ ปากทางม่วงโตน ตำบลเหล่ายาว  อำเภอบ้านโฮ่ง จังหวัดลำพูน ๕๑๑๓๐ |  |
| 3 | Fill contact information | Show text box to fill contact information   * Contact Name (ผู้ติดต่อ) * Email of Contact (อีเมลล์) * Contact Subject (หัวเรื่องติดต่อ) * Message (ข้อความ) |  |
| 4 | Send contact information | Enable to send contact information |  |
| 5 | Click to other page | Enable to click |  |

**Integration Test** IT-20 (Member, Member Register)

**Test Date** 17 May 2015

**Description**

|  |  |  |  |
| --- | --- | --- | --- |
| **ID** | **Input Specification** | **Output Specification** | **Environment Needs** |
| 1 | Showing dropdown list, what member can do | Show dropdown list   * Member Login (ลงชื่อเข้าใช้) * Register (สมัครสมาชิก) * Dhamma QA (ถาม-ตอบธรรมะ) * Gallery (แกลลอรี่) * Activity Clip (วิดีโอกิจกรรม) * Internet TV (อินเตอร์เน็ตทีวี) |  |
| 2 | Click “สมัครสมาชิก”button | Enable to click, then website will redirect to Member register page |  |
| 3 | Fill member information   * username * password * confirm password * email | Enable to fill information   * username * password * confirm password * email |  |
| 4 | Click register button | Enable to click for register to be new member |  |
| 5 | Click to other page | Enable to click |  |

**Integration Test** IT-21 (Member, Member Login)

**Test Date** 17 May 2015

**Description**

|  |  |  |  |
| --- | --- | --- | --- |
| **ID** | **Input Specification** | **Output Specification** | **Environment Needs** |
| 1 | Showing dropdown list, what member can do | Show dropdown list   * Member Login (ลงชื่อเข้าใช้) * Register (สมัครสมาชิก) * Dhamma QA (ถาม-ตอบธรรมะ) * Gallery (แกลลอรี่) * Activity Clip (วิดีโอกิจกรรม) * Internet TV (อินเตอร์เน็ตทีวี) |  |
| 2 | Click “ลงชื่อเข้าใช้”button | Enable to click, then website will redirect to Member home page |  |
| 3 | Fill member information   * username * password | Enable to fill information   * username * password |  |
| 4 | Click “ลงชื่อเข้าใช้” button | Enable to click for login of member |  |
| 5 | Click to other page | Enable to click |  |

**Integration Test** IT-22 (Member Login, Dhamma QA)

**Test Date** 17 May 2015

**Description**

|  |  |  |  |
| --- | --- | --- | --- |
| **ID** | **Input Specification** | **Output Specification** | **Environment Needs** |
| 1 | Showing dropdown list, what member can do | Show dropdown list   * Member Login (ลงชื่อเข้าใช้) * Register (สมัครสมาชิก) * Dhamma QA (ถาม-ตอบธรรมะ) * Gallery (แกลลอรี่) * Activity Clip (วิดีโอกิจกรรม) * Internet TV (อินเตอร์เน็ตทีวี) | Member Login |
| 2 | Click “ถามตอบธรรมะ” button | Enable to click, then website will redirect to DhammaQA page | Member Login |
| 3 | Showing questions and answers | Show questions and answers | Member Login |
| 4 | Click “คลิ๊กที่นี่เพื่อตั้งคำถาม” link | Enable to click for create new question and redirect to Dhamma QA page | Member Login |
| 5 | Fill the question | Enable to fill the question | Member Login |
| 6 | Send the question | Enable to send the question | Member Login |
| 5 | Click to other page | Enable to click |  |

**Integration Test** IT-23 (Member Login, Gallery)

**Test Date** 17 May 2015

**Description**

|  |  |  |  |
| --- | --- | --- | --- |
| **ID** | **Input Specification** | **Output Specification** | **Environment Needs** |
| 1 | Showing dropdown list, what member can do | Show dropdown list   * Member Login (ลงชื่อเข้าใช้) * Register (สมัครสมาชิก) * Dhamma QA (ถาม-ตอบธรรมะ) * Gallery (แกลลอรี่) * Activity Clip (วิดีโอกิจกรรม) * Internet TV (อินเตอร์เน็ตทีวี) | Member Login |
| 2 | Click “แกลลอรี่” button | Enable to click, then website will redirect to Gallery page | Member Login |
| 3 | Showing images | Show images | Member Login |
| 4 | Padding to bigger size image | Enable to pad bigger size image from gallery | Member Login |
| 5 | Click to other page | Enable to click |  |

**Integration Test** IT-24 (Member Login, Activity Clip)

**Test Date** 17 May 2015

**Description**

|  |  |  |  |
| --- | --- | --- | --- |
| **ID** | **Input Specification** | **Output Specification** | **Environment Needs** |
| 1 | Showing dropdown list, what member can do | Show dropdown list   * Member Login (ลงชื่อเข้าใช้) * Register (สมัครสมาชิก) * Dhamma QA (ถาม-ตอบธรรมะ) * Gallery (แกลลอรี่) * Activity Clip (วิดีโอกิจกรรม) * Internet TV (อินเตอร์เน็ตทีวี) | Member Login |
| 2 | Click “วิดีโอกิจกรรม” button | Enable to click, then website will redirect to ActivityClip page | Member Login |
| 3 | Showing activity clips | Show Activity Clips | Member Login |
| 4 | Click to other page | Enable to click |  |

**Integration Test** IT-25 (Member Login,Internet TV)

**Test Date** 17 May 2015

**Description**

|  |  |  |  |
| --- | --- | --- | --- |
| **ID** | **Input Specification** | **Output Specification** | **Environment Needs** |
| 1 | Showing dropdown list, what member can do | Show dropdown list   * Member Login (ลงชื่อเข้าใช้) * Register (สมัครสมาชิก) * Dhamma QA (ถาม-ตอบธรรมะ) * Gallery (แกลลอรี่) * Activity Clip (วิดีโอกิจกรรม) * Internet TV (อินเตอร์เน็ตทีวี) | Member Login |
| 2 | Click “อินเตอร์เน็ตทีวี” button | Enable to click, then website will redirect to Internet TV page | Member Login |
| 3 | Showing internet TV page | Show internet TV page | Member Login |
| 4 | Click to other page | Enable to click |  |

UNIT TEST

Table of Unit Test

|  |  |
| --- | --- |
| **Unit Test** | **Test Method** |
| UT-01 | Class AccountController : ActionResult Login(string returnUrl) |
| UT-02 | Class AccountController : ActionResult Login(LoginModel model, string returnUrl) |
| UT-03 | Class AccountController : ActionResult LogOff() |
| UT-04 | Class AccountController : ActionResult Register() |
| UT-05 | Class AccountController : ActionResult Register(RegisterModel model) |
| UT-06 | Class AccountController : ActionResult Disassociate(string provider, string providerUserId) |
| UT-07 | Class AccountController : ActionResult Manage(ManageMessageId? message) |
| UT-08 | Class AccountController : ActionResult Manage(LocalPasswordModel model) |
| UT-09 | Class AccountController : ActionResult ExternalLogin(string provider, string returnUrl) |
| UT-10 | Class AccountController : ActionResult ExternalLoginCallback(string returnUrl) |
| UT-11 | Class AccountController : ActionResult ExternalLoginConfirmation(RegisterExternalLoginModel model, string returnUrl) |
| UT-12 | Class AccountController : ActionResult ExternalLoginFailure() |
| UT-13 | Class AccountController : ActionResult ExternalLoginsList(string returnUrl) |
| UT-14 | Class AccountController : ActionResult RemoveExternalLogins() |
| UT-15 | Class AccountController : ActionResult Redirect(string returnUrl) |
| UT-16 | Class AccountController : enum ManageMessageID |
| UT-17 | Class AccountController : internal ExternalLoginResult : ActionResult  ExternalLoginResult(string provider, string returnUrl),  Provider {get; private set;}, ReturnUrl {get; private set;},  override void ExecuteResult(ControllerContext context) |
| UT-18 | Class AccountController : ErrorCodeToString(MembershipCreateStatus createStatus) |
| UT-19 | Class AdminController : ActionResult Login(string returnUrl) |
| UT-20 | Class AdminController : ActionResult Login(AdminAccount model,string returnUrl) |
| UT-21 | Class AdminController : ActionResult Logout() |
| UT-22 | Class AdminController : ViewResult Index() |

|  |  |
| --- | --- |
| **Unit Test** | **Test Method** |
| UT-23 | Class AdminController : ViewResult IndexManaging() |
| UT-24 | Class AdminController : ViewResult AddWelcomeImage() |
| UT-25 | Class AdminController : ViewResult AddGalleryImage() |
| UT-26 | Class AdminController : ActionResult DeleteWelcomeImage(int ImageID) |
| UT-27 | Class AdminController : ViewResult GalleryManaging() |
| UT-28 | Class AdminController : ViewResult EditBackgroundInfo(int backgroundId) |
| UT-29 | Class AdminController : ViewResult BackgroundManaging() |
| UT-30 | Class AdminController : ViewResult AddActivityClip() |
| UT-31 | Class AdminController : ViewResult AddAnnualEvent() |
| UT-32 | Class AdminController : ViewResult AddOtherEvent() |
| UT-33 | Class AdminController : ViewResult ReplyDhammaQA() |
| UT-34 | Class AdminController : ViewResult ActivityClipManaging() |
| UT-35 | Class AdminController : ViewResult DhammaQAManaging() |
| UT-36 | Class AdminController : ViewResult AnnualEventManaging() |
| UT-37 | Class AdminController : ViewResult OtherEventManaging() |
| UT-38 | Class AdminController : ActionResult DeleteGalleryImage(int ImageID) |
| UT-39 | Class AdminController : ActionResult DeleteActivityClip(int clipID) |
| UT-40 | Class AdminController : ActionResult DeleteDhammaQA(int QAID) |
| UT-41 | Class AdminController : ActionResult DeleteAnnualEvent(int EventID) |
| UT-42 | Class AdminController : ActionResult DeleteOtherEvent(int EventID) |
| UT-43 | Class AdminController : ActionResult IndexManaging(WelcomeImage welcomeImage, HttpPostedFileBase image) |
| UT-44 | Class AdminController : ActionResult Index(GalleryImage galleryImage, HttpPostedFileBase image) |
| UT-45 | Class AdminController : ActionResult EditBackground(Background background) |
| UT-46 | Class AdminController : ActionResult AddAnnualEvent(AnnualEvent annualEvent) |
| UT-47 | Class AdminController : ActionResult AddOtherEvent(OtherEvent otherEvent) |
| UT-48 | Class AdminController : ActionResult AddActivityClip(ActivityClip activityClip) |
| UT-49 | Class AdminController : ActionResult ReplyDhammaQuestion(DhammaQA dhammaQA) |

|  |  |
| --- | --- |
| **Unit Test** | **Test Method** |
| UT-50 | Class HomeController : FileContentResult GetWelcomeImage(int imageId) |
| UT-51 | Class HomeController : FileContentResult GetGalleryImage(int imageId) |
| UT-52 | Class HomeController : ActionResult Index() |
| UT-53 | Class HomeController : ActionResult Background() |
| UT-54 | Class HomeController : ActionResult AnnualEvent() |
| UT-55 | Class HomeController : ActionResult AnnualEventDetail(int aEventID) |
| UT-56 | Class HomeController : ActionResult DhammaContent() |
| UT-57 | Class HomeController : ActionResult DhammaQA() |
| UT-58 | Class HomeController : ViewResult SendDhammaQ() |
| UT-59 | Class HomeController : ActionResult SendDhammaQ(DhammaQA dhammaQA) |
| UT-60 | Class HomeController : ActionResult Gallery() |
| UT-61 | Class HomeController : ActionResult ActivityClip() |
| UT-62 | Class HomeController : ActionResult InternetTV() |
| UT-63 | Class HomeController : ActionResult Map() |
| UT-64 | Class HomeController : ActionResult Contact() |
| UT-65 | Class HomeController : ActionResult Contact(Contact \_objModelMail) |

**UNIT TEST**

**UT-01** Class AccountController : ActionResult Login(string returnUrl)

**Test Date** 16 May 2015

**Description** Test for view the Login page

|  |  |  |  |
| --- | --- | --- | --- |
| **ID** | **Expected Result** | **Actual Result** | **Pass/Fail** |
| 1 | ViewBag.ReturnUrl = returnUrl  return View(); | ViewBag.ReturnUrl = returnUrl  return View(); | PASS |

**UT-02** Class AccountController : ActionResult Login(LoginModel model, string returnUrl)

**Test Date** 16 May 2015

**Description** Test for checking login username and password

|  |  |  |  |
| --- | --- | --- | --- |
| **ID** | **Expected Result** | **Actual Result** | **Pass/Fail** |
| 1 | if (ModelState.IsValid && WebSecurity.Login(model.UserName, model.Password, persistCookie: model.RememberMe))  {  return RedirectToLocal(returnUrl);  }  // If we got this far, something failed, redisplay form  ModelState.AddModelError("", "The user name or password provided is incorrect.");  return View(model); | if (ModelState.IsValid && WebSecurity.Login(model.UserName, model.Password, persistCookie: model.RememberMe))  {  return RedirectToLocal(returnUrl);  }  // If we got this far, something failed, redisplay form  ModelState.AddModelError("", "The user name or password provided is incorrect.");  return View(model); | PASS |

**UT-03** Class AccountController : ActionResult LogOff()

**Test Date** 16 May 2015

**Description** Test for Log off

|  |  |  |  |
| --- | --- | --- | --- |
| **ID** | **Expected Result** | **Actual Result** | **Pass/Fail** |
| 1 | WebSecurity.Logout();  return RedirectToAction("Index", "Home"); | WebSecurity.Logout();  return RedirectToAction("Index", "Home"); | PASS |

**UT-04** Class AccountController : ActionResult Register()

**Test Date** 16 May 2015

**Description** Test for Register view

|  |  |  |  |
| --- | --- | --- | --- |
| **ID** | **Expected Result** | **Actual Result** | **Pass/Fail** |
| 1 | return View(); | return View(); | PASS |

**UT-05** Class AccountController : ActionResult Register(RegisterModel model)

|  |  |  |  |
| --- | --- | --- | --- |
| **ID** | **Expected Result** | **Actual Result** | **Pass/Fail** |
| 1 | string ownerAccount = OAuthWebSecurity.GetUserName(provider, providerUserId);  ManageMessageId? message = null;  if (ownerAccount == User.Identity.Name){  using (var scope = new TransactionScope(TransactionScopeOption.Required, new TransactionOptions { IsolationLevel = IsolationLevel.Serializable })){  bool hasLocalAccount = OAuthWebSecurity.HasLocalAccount(WebSecurity.GetUserId(User.Identity.Name));  if (hasLocalAccount || OAuthWebSecurity.GetAccountsFromUserName(User.Identity.Name).Count > 1){  OAuthWebSecurity.DeleteAccount(provider, providerUserId);  scope.Complete();  message = ManageMessageId.RemoveLoginSuccess; }}}  return RedirectToAction("Manage", new { Message = message });} | string ownerAccount = OAuthWebSecurity.GetUserName(provider, providerUserId);  ManageMessageId? message = null;  if (ownerAccount == User.Identity.Name){  using (var scope = new TransactionScope(TransactionScopeOption.Required, new TransactionOptions { IsolationLevel = IsolationLevel.Serializable })){  bool hasLocalAccount = OAuthWebSecurity.HasLocalAccount(WebSecurity.GetUserId(User.Identity.Name));  if (hasLocalAccount || OAuthWebSecurity.GetAccountsFromUserName(User.Identity.Name).Count > 1){  OAuthWebSecurity.DeleteAccount(provider, providerUserId);  scope.Complete();  message = ManageMessageId.RemoveLoginSuccess; }}}  return RedirectToAction("Manage", new { Message = message });} | PASS |

**Test Date** 16 May 2015

**Description** Test for Model Register view

**UT-06** Class AccountController : ActionResult Disassociate(string provider, string providerUserId)

**Test Date** 16 May 2015

**Description** Test for redirect to home page of member

|  |  |  |  |
| --- | --- | --- | --- |
| **ID** | **Expected Result** | **Actual Result** | **Pass/Fail** |
| 1 | if (ModelState.IsValid) {  try {  WebSecurity.CreateUserAndAccount(model.UserName, model.Password); WebSecurity.Login(model.UserName, model.Password);  return RedirectToAction("Index", “Home"); }  catch (MembershipCreateUserException e) {  ModelState.AddModelError("", ErrorCodeToString(e.StatusCode)); } }  return View(model); | if (ModelState.IsValid) {  try {  WebSecurity.CreateUserAndAccount(model.UserName, model.Password); WebSecurity.Login(model.UserName, model.Password);  return RedirectToAction("Index", “Home"); }  catch (MembershipCreateUserException e) {  ModelState.AddModelError("", ErrorCodeToString(e.StatusCode)); } }  return View(model); | PASS |

**UT-07** Class AccountController : ActionResult Manage(ManageMessageId? message)

**Test Date** 16 May 2015

**Description** Test for showing the massage

|  |  |  |  |
| --- | --- | --- | --- |
| **ID** | **Expected Result** | **Actual Result** | **Pass/Fail** |
| 1 | ViewBag.StatusMessage =  message == ManageMessageId.ChangePasswordSuccess ? "Your password has been changed."  : message == ManageMessageId.SetPasswordSuccess ? "Your password has been set."  : message == ManageMessageId.RemoveLoginSuccess ? "The external login was removed."  : "";  ViewBag.HasLocalPassword = OAuthWebSecurity.HasLocalAccount(WebSecurity.GetUserId(User.Identity.Name));  ViewBag.ReturnUrl = Url.Action("Manage");  return View(); | ViewBag.StatusMessage =  message == ManageMessageId.ChangePasswordSuccess ? "Your password has been changed."  : message == ManageMessageId.SetPasswordSuccess ? "Your password has been set."  : message == ManageMessageId.RemoveLoginSuccess ? "The external login was removed."  : "";  ViewBag.HasLocalPassword = OAuthWebSecurity.HasLocalAccount(WebSecurity.GetUserId(User.Identity.Name));  ViewBag.ReturnUrl = Url.Action("Manage");  return View(); | PASS |

**UT-08** Class AccountController : ActionResult Manage(LocalPasswordModel model)

**Test Date** 16 May 2015

**Description** Test for verify password

|  |  |  |  |
| --- | --- | --- | --- |
| **ID** | **Expected Result** | **Actual Result** | **Pass/Fail** |
| 1 | bool hasLocalAccount = OAuthWebSecurity.HasLocalAccount(WebSecurity.GetUserId(User.Identity.Name));  ViewBag.HasLocalPassword = hasLocalAccount;  ViewBag.ReturnUrl = Url.Action("Manage");  if (hasLocalAccount){  if (ModelState.IsValid){  bool changePasswordSucceeded;  try{ changePasswordSucceeded = WebSecurity.ChangePassword(User.Identity.Name, model.OldPassword, model.NewPassword);  } catch (Exception e) { changePasswordSucceeded = false; }  if (changePasswordSucceeded){  return RedirectToAction("Manage", new { Message = ManageMessageId.ChangePasswordSuccess });} else{  ModelState.AddModelError("", "The current password is incorrect or the new password is invalid.”);}}} else{  ModelState state = ModelState[“OldPassword"];  if (state != null) {state.Errors.Clear(); }  if (ModelState.IsValid) { try { WebSecurity.CreateAccount(User.Identity.Name, model.NewPassword);  return RedirectToAction("Manage", new { Message = ManageMessageId.SetPasswordSuccess });} catch (Exception e){ModelState.AddModelError("", e);  return View(model); | bool hasLocalAccount = OAuthWebSecurity.HasLocalAccount(WebSecurity.GetUserId(User.Identity.Name));  ViewBag.HasLocalPassword = hasLocalAccount;  ViewBag.ReturnUrl = Url.Action("Manage");  if (hasLocalAccount){  if (ModelState.IsValid){  bool changePasswordSucceeded;  try{ changePasswordSucceeded = WebSecurity.ChangePassword(User.Identity.Name, model.OldPassword, model.NewPassword);  } catch (Exception e) { changePasswordSucceeded = false; }  if (changePasswordSucceeded){  return RedirectToAction("Manage", new { Message = ManageMessageId.ChangePasswordSuccess });} else{  ModelState.AddModelError("", "The current password is incorrect or the new password is invalid.”);}}} else{  ModelState state = ModelState[“OldPassword"];  if (state != null) {state.Errors.Clear(); }  if (ModelState.IsValid) { try { WebSecurity.CreateAccount(User.Identity.Name, model.NewPassword);  return RedirectToAction("Manage", new { Message = ManageMessageId.SetPasswordSuccess });} catch (Exception e){ModelState.AddModelError("", e);  return View(model); | PASS |

**UT-09** Class AccountController : ActionResult ExternalLogin(string provider, string returnUrl)

**Test Date** 16 May 2015

**Description** Test for external login

|  |  |  |  |
| --- | --- | --- | --- |
| **ID** | **Expected Result** | **Actual Result** | **Pass/Fail** |
| 1 | return new ExternalLoginResult(provider, Url.Action("ExternalLoginCallback", new { ReturnUrl = returnUrl })); | return new ExternalLoginResult(provider, Url.Action("ExternalLoginCallback", new { ReturnUrl = returnUrl })); | PASS |

**UT-10** Class AccountController : ActionResult ExternalLoginCallback(string returnUrl)

**Test Date** 16 May 2015

**Description** Test for external login

|  |  |  |  |
| --- | --- | --- | --- |
| **ID** | **Expected Result** | **Actual Result** | **Pass/Fail** |
| 1 | AuthenticationResult result = OAuthWebSecurity.VerifyAuthentication(Url.Action("ExternalLoginCallback", new { ReturnUrl = returnUrl }));  if (!result.IsSuccessful){  return RedirectToAction("ExternalLoginFailure");}if (OAuthWebSecurity.Login(result.Provider, result.ProviderUserId, createPersistentCookie: false)){  return RedirectToLocal(returnUrl);}  if (User.Identity.IsAuthenticated){ OAuthWebSecurity.CreateOrUpdateAccount(result.Provider, result.ProviderUserId, User.Identity.Name);  return RedirectToLocal(returnUrl);} else{  string loginData = OAuthWebSecurity.SerializeProviderUserId(result.Provider, result.ProviderUserId);  ViewBag.ProviderDisplayName = OAuthWebSecurity.GetOAuthClientData(result.Provider).DisplayName;  ViewBag.ReturnUrl = returnUrl;  return View("ExternalLoginConfirmation", new RegisterExternalLoginModel { UserName = result.UserName, ExternalLoginData = loginData }); | AuthenticationResult result = OAuthWebSecurity.VerifyAuthentication(Url.Action("ExternalLoginCallback", new { ReturnUrl = returnUrl }));  if (!result.IsSuccessful){  return RedirectToAction("ExternalLoginFailure");}if (OAuthWebSecurity.Login(result.Provider, result.ProviderUserId, createPersistentCookie: false)){  return RedirectToLocal(returnUrl);}  if (User.Identity.IsAuthenticated){ OAuthWebSecurity.CreateOrUpdateAccount(result.Provider, result.ProviderUserId, User.Identity.Name);  return RedirectToLocal(returnUrl);} else{  string loginData = OAuthWebSecurity.SerializeProviderUserId(result.Provider, result.ProviderUserId);  ViewBag.ProviderDisplayName = OAuthWebSecurity.GetOAuthClientData(result.Provider).DisplayName;  ViewBag.ReturnUrl = returnUrl;  return View("ExternalLoginConfirmation", new RegisterExternalLoginModel { UserName = result.UserName, ExternalLoginData = loginData }); | PASS |

**UT-11** Class AccountController : ActionResult ExternalLoginConfirmation(RegisterExternalLoginModel model, string returnUrl)

**Test Date** 16 May 2015

**Description** Test for confirm register of new member

|  |  |  |  |
| --- | --- | --- | --- |
| **ID** | **Expected Result** | **Actual Result** | **Pass/Fail** |
| 1 | string provider = null; string providerUserId = null;  if (User.Identity.IsAuthenticated || !OAuthWebSecurity.TryDeserializeProviderUserId(model.ExternalLoginData, out provider, out providerUserId)) {  return RedirectToAction("Manage");}  if (ModelState.IsValid) {  using (UsersContext db = new UsersContext()) {  UserProfile user = db.UserProfiles.FirstOrDefault(u => u.UserName.ToLower() == model.UserName.ToLower());  if (user == null) {  db.UserProfiles.Add(new UserProfile { UserName = model.UserName });  db.SaveChanges();  OAuthWebSecurity.CreateOrUpdateAccount(provider, providerUserId, model.UserName); OAuthWebSecurity.Login(provider, providerUserId, createPersistentCookie: false);  return RedirectToLocal(returnUrl); } else{ ModelState.AddModelError("UserName", "User name already exists. Please enter a different user name.”); }}}  ViewBag.ProviderDisplayName = OAuthWebSecurity.GetOAuthClientData(provider).DisplayName;  ViewBag.ReturnUrl = returnUrl;  return View(model); | string provider = null; string providerUserId = null;  if (User.Identity.IsAuthenticated || !OAuthWebSecurity.TryDeserializeProviderUserId(model.ExternalLoginData, out provider, out providerUserId)) {  return RedirectToAction("Manage");}  if (ModelState.IsValid) {  using (UsersContext db = new UsersContext()) {  UserProfile user = db.UserProfiles.FirstOrDefault(u => u.UserName.ToLower() == model.UserName.ToLower());  if (user == null) {  db.UserProfiles.Add(new UserProfile { UserName = model.UserName });  db.SaveChanges();  OAuthWebSecurity.CreateOrUpdateAccount(provider, providerUserId, model.UserName); OAuthWebSecurity.Login(provider, providerUserId, createPersistentCookie: false);  return RedirectToLocal(returnUrl); } else{ ModelState.AddModelError("UserName", "User name already exists. Please enter a different user name.”); }}}  ViewBag.ProviderDisplayName = OAuthWebSecurity.GetOAuthClientData(provider).DisplayName;  ViewBag.ReturnUrl = returnUrl;  return View(model); | PASS |

**UT-12** Class AccountController : ActionResult ExternalLoginFailure()

**Test Date** 16 May 2015

**Description** Test for provide failure page if cannot login

|  |  |  |  |
| --- | --- | --- | --- |
| **ID** | **Expected Result** | **Actual Result** | **Pass/Fail** |
| 1 | return View(); | return View(); | PASS |

**UT-13** Class AccountController : ActionResult ExternalLoginsList(string returnUrl)

**Test Date** 16 May 2015

**Description** Test for show list of external login

|  |  |  |  |
| --- | --- | --- | --- |
| **ID** | **Expected Result** | **Actual Result** | **Pass/Fail** |
| 1 | ViewBag.ReturnUrl = returnUrl;  return PartialView("\_ExternalLoginsListPartial", OAuthWebSecurity.RegisteredClientData); | ViewBag.ReturnUrl = returnUrl;  return PartialView("\_ExternalLoginsListPartial", OAuthWebSecurity.RegisteredClientData); | PASS |

**UT-14** Class AccountController : ActionResult RemoveExternalLogins()

**Test Date** 16 May 2015

**Description** Test for remove external login

|  |  |  |  |
| --- | --- | --- | --- |
| **ID** | **Expected Result** | **Actual Result** | **Pass/Fail** |
| 1 | ICollection<OAuthAccount> accounts = OAuthWebSecurity.GetAccountsFromUserName(User.Identity.Name);  List<ExternalLogin> externalLogins = new List<ExternalLogin>();  foreach (OAuthAccount account in accounts){  AuthenticationClientData clientData = OAuthWebSecurity.GetOAuthClientData(account.Provider);  externalLogins.Add(new ExternalLogin {  Provider = account.Provider, ProviderDisplayName = clientData.DisplayName, ProviderUserId = account.ProviderUserId,}); }  ViewBag.ShowRemoveButton = externalLogins.Count > 1 || OAuthWebSecurity.HasLocalAccount(WebSecurity.GetUserId(User.Identity.Name));  return PartialView("\_RemoveExternalLoginsPartial", externalLogins); | ICollection<OAuthAccount> accounts = OAuthWebSecurity.GetAccountsFromUserName(User.Identity.Name);  List<ExternalLogin> externalLogins = new List<ExternalLogin>();  foreach (OAuthAccount account in accounts){  AuthenticationClientData clientData = OAuthWebSecurity.GetOAuthClientData(account.Provider);  externalLogins.Add(new ExternalLogin {  Provider = account.Provider, ProviderDisplayName = clientData.DisplayName, ProviderUserId = account.ProviderUserId,}); }  ViewBag.ShowRemoveButton = externalLogins.Count > 1 || OAuthWebSecurity.HasLocalAccount(WebSecurity.GetUserId(User.Identity.Name));  return PartialView("\_RemoveExternalLoginsPartial", externalLogins); | PASS |

**UT-15** Class AccountController : ActionResult RedirectToLocal(string returnUrl)

**Test Date** 16 May 2015

**Description** Test for redirect to Index page and home page

|  |  |  |  |
| --- | --- | --- | --- |
| **ID** | **Expected Result** | **Actual Result** | **Pass/Fail** |
| 1 | if (Url.IsLocalUrl(returnUrl)) {  return Redirect(returnUrl);  } else {  return RedirectToAction("Index", “Home"); } | if (Url.IsLocalUrl(returnUrl)) {  return Redirect(returnUrl);  } else {  return RedirectToAction("Index", “Home"); } | PASS |

**UT-16** Class AccountController : enum ManageMessageID

**Test Date** 16 May 2015

**Description** Test for manage message ID

|  |  |  |  |
| --- | --- | --- | --- |
| **ID** | **Expected Result** | **Actual Result** | **Pass/Fail** |
| 1 | ChangePasswordSuccess,  SetPasswordSuccess,  RemoveLoginSuccess, | ChangePasswordSuccess,  SetPasswordSuccess,  RemoveLoginSuccess, | PASS |

**UT-17** Class AccountController : internal ExternalLoginResult : ActionResult

ExternalLoginResult(string provider, string returnUrl),

Provider {get; private set;}, ReturnUrl {get; private set;},

override void ExecuteResult(ControllerContext context)

**Test Date** 16 May 2015

|  |  |  |  |
| --- | --- | --- | --- |
| **ID** | **Expected Result** | **Actual Result** | **Pass/Fail** |
| 1 | public ExternalLoginResult(string provider, string returnUrl) {  Provider = provider;  ReturnUrl = returnUrl; }  public string Provider { get; private set; }  public string ReturnUrl { get; private set; }  public override void ExecuteResult(ControllerContext context) {  OAuthWebSecurity.RequestAuthentication(Provider, ReturnUrl);} | public ExternalLoginResult(string provider, string returnUrl) {  Provider = provider;  ReturnUrl = returnUrl; }  public string Provider { get; private set; }  public string ReturnUrl { get; private set; }  public override void ExecuteResult(ControllerContext context) {  OAuthWebSecurity.RequestAuthentication(Provider, ReturnUrl);} | PASS |

**Description** Test for result of external login

**UT-18** Class AccountController : ErrorCodeToString(MembershipCreateStatus createStatus)

**Test Date** 16 May 2015

**Description** Test for status of membership

|  |  |  |  |
| --- | --- | --- | --- |
| **ID** | **Expected Result** | **Actual Result** | **Pass/Fail** |
| 1 | switch (createStatus) {  case MembershipCreateStatus.DuplicateUserName: return "User name already exists. Please enter a different user name.”;  case MembershipCreateStatus.DuplicateEmail: return "A user name for that e-mail address already exists. Please enter a different e-mail address.”;  case MembershipCreateStatus.InvalidPassword: return "The password provided is invalid. Please enter a valid password value.";  case MembershipCreateStatus.InvalidEmail: return "The e-mail address provided is invalid. Please check the value and try again.";  case MembershipCreateStatus.InvalidAnswer: return "The password retrieval answer provided is invalid. Please check the value and try again.";  case MembershipCreateStatus.InvalidQuestion:return "The password retrieval question provided is invalid. Please check the value and try again.";  case MembershipCreateStatus.InvalidUserName: return "The user name provided is invalid. Please check the value and try again.”;  case MembershipCreateStatus.ProviderError: return "The authentication provider returned an error. Please verify your entry and try again. If the problem persists, please contact your system administrator.";case MembershipCreateStatus.UserRejected:return "The user creation request has been canceled. Please verify your entry and try again. If the problem persists, please contact your system administrator.”;  default: return "An unknown error occurred. Please verify your entry and try again. If the problem persists, please contact your system administrator.”; | switch (createStatus) {  case MembershipCreateStatus.DuplicateUserName: return "User name already exists. Please enter a different user name.”;  case MembershipCreateStatus.DuplicateEmail: return "A user name for that e-mail address already exists. Please enter a different e-mail address.”;  case MembershipCreateStatus.InvalidPassword: return "The password provided is invalid. Please enter a valid password value.";  case MembershipCreateStatus.InvalidEmail: return "The e-mail address provided is invalid. Please check the value and try again.";  case MembershipCreateStatus.InvalidAnswer: return "The password retrieval answer provided is invalid. Please check the value and try again.";  case MembershipCreateStatus.InvalidQuestion:return "The password retrieval question provided is invalid. Please check the value and try again.";  case MembershipCreateStatus.InvalidUserName: return "The user name provided is invalid. Please check the value and try again.”;  case MembershipCreateStatus.ProviderError: return "The authentication provider returned an error. Please verify your entry and try again. If the problem persists, please contact your system administrator.";case MembershipCreateStatus.UserRejected:return "The user creation request has been canceled. Please verify your entry and try again. If the problem persists, please contact your system administrator.”;  default: return "An unknown error occurred. Please verify your entry and try again. If the problem persists, please contact your system administrator.”; | PASS |

**UT-19** Class AdminController : ActionResult Login(string returnUrl)

**Test Date** 16 May 2015

**Description** Test for Login page of administer

|  |  |  |  |
| --- | --- | --- | --- |
| **ID** | **Expected Result** | **Actual Result** | **Pass/Fail** |
| 1 | ViewBag.ReturnUrl = returnUrl;  return View(); | ViewBag.ReturnUrl = returnUrl;  return View(); | PASS |

**UT-20** Class AdminController : ActionResult Login(AdminAccount model,string returnUrl)

**Test Date** 16 May 2015

**Description** Test for Login of administer

|  |  |  |  |
| --- | --- | --- | --- |
| **ID** | **Expected Result** | **Actual Result** | **Pass/Fail** |
| 1 | if (model.UserName == "admin" && model.Password == “admin1234") {  Session["LoginStatus"] = “True";  return RedirectToAction(“IndexManaging");  } else {  System.Windows.Forms.MessageBox.Show("ชื่อผู้ใช้หรือรหัสผ่านไม่ถูกต้อง กรุณากรอกใหม่อีกครั้ง”);  return View(); } | if (model.UserName == "admin" && model.Password == “admin1234") {  Session["LoginStatus"] = “True";  return RedirectToAction(“IndexManaging");  } else {  System.Windows.Forms.MessageBox.Show("ชื่อผู้ใช้หรือรหัสผ่านไม่ถูกต้อง กรุณากรอกใหม่อีกครั้ง”);  return View(); } | PASS |

**UT-21** Class AdminController : ActionResult Logout()

**Test Date** 16 May 2015

**Description** Test for Logout of Administer

|  |  |  |  |
| --- | --- | --- | --- |
| **ID** | **Expected Result** | **Actual Result** | **Pass/Fail** |
| 1 | Session.Remove(“LoginStatus");  return RedirectToAction(“Index"); } | Session.Remove(“LoginStatus”);  return RedirectToAction(“Index"); } | PASS |

**UT-22** Class AdminController : ViewResult Index()

**Test Date** 16 May 2015

**Description** Test for show index page of Administer

|  |  |  |  |
| --- | --- | --- | --- |
| **ID** | **Expected Result** | **Actual Result** | **Pass/Fail** |
| 1 | return View(); | return View(); | PASS |

**UT-23** Class AdminController : ViewResult IndexManaging()

**Test Date** 16 May 2015

**Description** Test for manage images of home page

|  |  |  |  |
| --- | --- | --- | --- |
| **ID** | **Expected Result** | **Actual Result** | **Pass/Fail** |
| 1 | return View(repository.WelcomeImages); | return View(repository.WelcomeImages); | PASS |

**UT-24** Class AdminController : ViewResult AddWelcomeImage()

**Test Date** 16 May 2015

**Description** Test for adding images

|  |  |  |  |
| --- | --- | --- | --- |
| **ID** | **Expected Result** | **Actual Result** | **Pass/Fail** |
| 1 | return View(“AddWelcomeImage”, new WelcomeImage); | return View(“AddWelcomeImage”, new WelcomeImage); | PASS |

**UT-25** Class AdminController : ViewResult AddGalleryImage()

**Test Date** 16 May 2015

**Description** Test for adding images to gallery

|  |  |  |  |
| --- | --- | --- | --- |
| **ID** | **Expected Result** | **Actual Result** | **Pass/Fail** |
| 1 | return View(“AddGalleryImage”, new GalleryImage); | return View(“AddGalleryImage”, new GalleryImage); | PASS |

**UT-26** Class AdminController : ActionResult DeleteWelcomeImage(int ImageID)

**Test Date** 16 May 2015

**Description** Test for delete images of home page

|  |  |  |  |
| --- | --- | --- | --- |
| **ID** | **Expected Result** | **Actual Result** | **Pass/Fail** |
| 1 | WelcomeImage deletedWelcomeImage = repository.DeleteWelcomeImage(ImageID)  if (deletedWelcomeImage != null{  TempData["message"] = "Delete success”; }  return RedirectToAction("IndexManaging"); | WelcomeImage deletedWelcomeImage = repository.DeleteWelcomeImage(ImageID)  if (deletedWelcomeImage != null{  TempData["message"] = "Delete success”; }  return RedirectToAction("IndexManaging"); | PASS |

**UT-27** Class AdminController : ViewResult GalleryManaging()

**Test Date** 16 May 2015

**Description** Test for view images into gallery

|  |  |  |  |
| --- | --- | --- | --- |
| **ID** | **Expected Result** | **Actual Result** | **Pass/Fail** |
| 1 | return View(repository2.GalleryImages); | return View(repository2.GalleryImages); | PASS |

**UT-28** Class AdminController : ViewResult EditBackgroundInfo(int backgroundId)

**Test Date** 16 May 2015

**Description** Test for edit background of Wat Pa Thammapirom

|  |  |  |  |
| --- | --- | --- | --- |
| **ID** | **Expected Result** | **Actual Result** | **Pass/Fail** |
| 1 | Background background = repository3.Backgrounds.FirstOrDefault(p => p.BackgroundID == backgroundId);  return View(background); | Background background = repository3.Backgrounds.FirstOrDefault(p => p.BackgroundID == backgroundId);  return View(background); | PASS |

**UT-29** Class AdminController : ViewResult BackgroundManaging()

**Test Date** 16 May 2015

**Description** Test for view background of Wat Pa Thammapirom

|  |  |  |  |
| --- | --- | --- | --- |
| **ID** | **Expected Result** | **Actual Result** | **Pass/Fail** |
| 1 | return View(repository3.Backgrounds); | return View(repository3.Backgrounds); | PASS |

**UT-30** Class AdminController : ViewResult AddActivityClip()

**Test Date** 16 May 2015

**Description** Test for adding activity clips

|  |  |  |  |
| --- | --- | --- | --- |
| **ID** | **Expected Result** | **Actual Result** | **Pass/Fail** |
| 1 | return View("AddActivityClip", new ActivityClip()); | return View("AddAnnualEvent", new AnnualEvent()); | PASS |

**UT-31** Class AdminController : ViewResult AddAnnualEvent()

**Test Date** 16 May 2015

**Description** Test for adding annual event

|  |  |  |  |
| --- | --- | --- | --- |
| **ID** | **Expected Result** | **Actual Result** | **Pass/Fail** |
| 1 | return View("AddAnnualEvent", new AnnualEvent()); | return View("AddAnnualEvent", new AnnualEvent()); | PASS |

**UT-32** Class AdminController : ViewResult AddOtherEvent()

**Test Date** 16 May 2015

**Description** Test for adding other event

|  |  |  |  |
| --- | --- | --- | --- |
| **ID** | **Expected Result** | **Actual Result** | **Pass/Fail** |
| 1 | return View("AddOtherEvent", new OtherEvent()); | return View("AddOtherEvent", new OtherEvent()); | PASS |

**UT-33** Class AdminController : ViewResult ReplyDhammaQA(int qaID)

**Test Date** 16 May 2015

**Description** Test for reply dharma question and answer

|  |  |  |  |
| --- | --- | --- | --- |
| **ID** | **Expected Result** | **Actual Result** | **Pass/Fail** |
| 1 | DhammaQA dhammaQA = repository5.DhammaQAs.FirstOrDefault(p => p.QAID == qaID);  return View(dhammaQA); | DhammaQA dhammaQA = repository5.DhammaQAs.FirstOrDefault(p => p.QAID == qaID);  return View(dhammaQA); | PASS |

**UT-34** Class AdminController : ViewResult ActivityClipManaging()

**Test Date** 16 May 2015

**Description** Test for view activity clips

|  |  |  |  |
| --- | --- | --- | --- |
| **ID** | **Expected Result** | **Actual Result** | **Pass/Fail** |
| 1 | return View(repository4.ActivityClips); | return View(repository4.ActivityClips); | PASS |

**UT-35** Class AdminController : ViewResult DhammaQAManaging()

**Test Date** 16 May 2015

**Description** Test for view dharma question and answer

|  |  |  |  |
| --- | --- | --- | --- |
| **ID** | **Expected Result** | **Actual Result** | **Pass/Fail** |
| 1 | return View(repository5.DhammaQAs); | return View(repository5.DhammaQAs); | PASS |

**UT-36** Class AdminController : ViewResult AnnualEventManaging()

**Test Date** 16 May 2015

**Description** Test for view annual event

|  |  |  |  |
| --- | --- | --- | --- |
| **ID** | **Expected Result** | **Actual Result** | **Pass/Fail** |
| 1 | return View(repository6.AnnualEvents); | return View(repository6.AnnualEvents); | PASS |

**UT-37** Class AdminController : ViewResult OtherEventManaging()

**Test Date** 16 May 2015

**Description** Test for view other event

|  |  |  |  |
| --- | --- | --- | --- |
| **ID** | **Expected Result** | **Actual Result** | **Pass/Fail** |
| 1 | return View(repository7.OtherEvents); | return View(repository7.OtherEvents); | PASS |

**UT-38** Class AdminController : ActionResult DeleteGalleryImage(int ImageID)

**Test Date** 16 May 2015

**Description** Test for delete images in gallery

|  |  |  |  |
| --- | --- | --- | --- |
| **ID** | **Expected Result** | **Actual Result** | **Pass/Fail** |
| 1 | GalleryImage deletedGalleryImage = repository2.DeleteGalleryImage(ImageID);  if (deletedGalleryImage != null){ TempData["message"] = "Delete success”; }  return RedirectToAction(“Index"); | GalleryImage deletedGalleryImage = repository2.DeleteGalleryImage(ImageID);  if (deletedGalleryImage != null){ TempData["message"] = "Delete success”; }  return RedirectToAction(“Index"); | PASS |

**UT-39** Class AdminController : ActionResult DeleteActivityClip(int clipID)

**Test Date** 16 May 2015

**Description** Test for delete activity clip

|  |  |  |  |
| --- | --- | --- | --- |
| **ID** | **Expected Result** | **Actual Result** | **Pass/Fail** |
| 1 | ActivityClip deletedActivityClip = repository4.DeleteActivityClip(clipID);  if (deletedActivityClip != null) { TempData["message"] = "Delete success”;}  return RedirectToAction("ActivityClipManaging"); | ActivityClip deletedActivityClip = repository4.DeleteActivityClip(clipID);  if (deletedActivityClip != null) { TempData["message"] = "Delete success”;}  return RedirectToAction("ActivityClipManaging"); | PASS |

**UT-40** Class AdminController : ActionResult DeleteDhammaQA()

**Test Date** 16 May 2015

**Description** Test for delete dhamma question and answer

|  |  |  |  |
| --- | --- | --- | --- |
| **ID** | **Expected Result** | **Actual Result** | **Pass/Fail** |
| 1 | DhammaQA deletedDhammaQA =repository5.DeleteDhammaQA(QAID);  if (deletedDhammaQA != null) {  TempData["message"] = "Delete success”; }  return RedirectToAction("DhammaQAManaging"); | DhammaQA deletedDhammaQA =repository5.DeleteDhammaQA(QAID);  if (deletedDhammaQA != null) {  TempData["message"] = "Delete success”; }  return RedirectToAction("DhammaQAManaging"); | PASS |

**UT-41** Class AdminController : ActionResult DeleteAnnualEvent(int EventID)

**Test Date** 16 May 2015

**Description** Test for delete annual event

|  |  |  |  |
| --- | --- | --- | --- |
| **ID** | **Expected Result** | **Actual Result** | **Pass/Fail** |
| 1 | AnnualEvent deletedAnnualEvent =repository6.DeleteAnnualEvent(EventID);  if (deletedAnnualEvent != null) {  TempData["message"] = "Delete success”; }  return RedirectToAction(“AnnualEventManaging"); } | AnnualEvent deletedAnnualEvent =repository6.DeleteAnnualEvent(EventID);  if (deletedAnnualEvent != null) {  TempData["message"] = "Delete success”; }  return RedirectToAction(“AnnualEventManaging"); } | PASS |

**UT-42** Class AdminController : ActionResult DeleteOtherEvent(int EventID)

**Test Date** 16 May 2015

**Description** Test for delete other event

|  |  |  |  |
| --- | --- | --- | --- |
| **ID** | **Expected Result** | **Actual Result** | **Pass/Fail** |
| 1 | OtherEvent deletedOtherEvent = repository7.DeleteOtherEvent(EventID);  if (deletedOtherEvent != null) {  TempData["message"] = "Delete success”; }  return RedirectToAction("OtherEventManaging"); | OtherEvent deletedOtherEvent = repository7.DeleteOtherEvent(EventID);  if (deletedOtherEvent != null) {  TempData["message"] = "Delete success”; }  return RedirectToAction("OtherEventManaging"); | PASS |

**UT-43** Class AdminController : ActionResult IndexManaging(WelcomeImage welcomeImage, HttpPostedFileBase image)

**Test Date** 16 May 2015

|  |  |  |  |
| --- | --- | --- | --- |
| **ID** | **Expected Result** | **Actual Result** | **Pass/Fail** |
| 1 | if (ModelState.IsValid{  if (image != null) {  welcomeImage.ImageMimeType = image.ContentType;  welcomeImage.ImageData = new byte[image.ContentLength];  image.InputStream.Read(welcomeImage.ImageData, 0, image.ContentLength); }  repository.SaveWelcomeImage(welcomeImage);  TempData["message"] = string.Format("{0} has been saved", welcomeImage.ImageID);  return RedirectToAction(“IndexManaging”); } else {  return View(welcomeImage); | if (ModelState.IsValid{  if (image != null) {  welcomeImage.ImageMimeType = image.ContentType;  welcomeImage.ImageData = new byte[image.ContentLength];  image.InputStream.Read(welcomeImage.ImageData, 0, image.ContentLength); }  repository.SaveWelcomeImage(welcomeImage);  TempData["message"] = string.Format("{0} has been saved", welcomeImage.ImageID);  return RedirectToAction(“IndexManaging”); } else {  return View(welcomeImage); | PASS |

**Description** Test for managing of index page of Administer

**UT-44** Class AdminController : ActionResult Index(GalleryImage galleryImage, HttpPostedFileBase image)

**Test Date** 16 May 2015

**Description** Test for saving images by managing of Administer

|  |  |  |  |
| --- | --- | --- | --- |
| **ID** | **Expected Result** | **Actual Result** | **Pass/Fail** |
| 1 | if (ModelState.IsValid){  if (image != null){  galleryImage.ImageMimeType = image.ContentType; galleryImage.ImageData = new byte[image.ContentLength]; image.InputStream.Read(galleryImage.ImageData, 0, image.ContentLength); }  repository2.SaveGalleryImage(galleryImage); TempData["message"] = string.Format("{0} has been saved", galleryImage.ImageID);  return RedirectToAction(“Index"); } else {  return View(galleryImage); | if (ModelState.IsValid){  if (image != null){  galleryImage.ImageMimeType = image.ContentType; galleryImage.ImageData = new byte[image.ContentLength]; image.InputStream.Read(galleryImage.ImageData, 0, image.ContentLength); }  repository2.SaveGalleryImage(galleryImage); TempData["message"] = string.Format("{0} has been saved", galleryImage.ImageID);  return RedirectToAction(“Index"); } else {  return View(galleryImage); | PASS |

**UT-45** Class AdminController : ActionResult EditBackground(Background background)

**Test Date** 16 May 2015

**Description** Test for edit background by managing of Administer

|  |  |  |  |
| --- | --- | --- | --- |
| **ID** | **Expected Result** | **Actual Result** | **Pass/Fail** |
| 1 | if (ModelState.IsValid) {  repository3.SaveBackgroundInfo(background);  return RedirectToAction(“BackgroundManaging"); } else {  return View(background); | if (ModelState.IsValid) {  repository3.SaveBackgroundInfo(background);  return RedirectToAction(“BackgroundManaging"); } else {  return View(background); | PASS |

**UT-46** Class AdminController : ActionResult AddAnnualEvent(AnnualEvent annualEvent)

**Test Date** 16 May 2015

**Description** Test for adding annual event by managing of Administer

|  |  |  |  |
| --- | --- | --- | --- |
| **ID** | **Expected Result** | **Actual Result** | **Pass/Fail** |
| 1 | if (ModelState.IsValid) {  repository6.SaveAnnualEvent(annualEvent);  return RedirectToAction(“AnnualEventManaging"); }  else {  return View(annualEvent); } | if (ModelState.IsValid) {  repository6.SaveAnnualEvent(annualEvent);  return RedirectToAction(“AnnualEventManaging"); }  else {  return View(annualEvent); } | PASS |

**UT-47** Class AdminController : ActionResult AddOtherEvent(OtherEvent otherEvent)

**Test Date** 16 May 2015

**Description** Test for adding other event by managing of Administer

|  |  |  |  |
| --- | --- | --- | --- |
| **ID** | **Expected Result** | **Actual Result** | **Pass/Fail** |
| 1 | if (ModelState.IsValid) {  repository7.SaveOtherEvent(otherEvent);  return RedirectToAction(“AnnualEventManaging”); }  else {  return View(otherEvent); } | if (ModelState.IsValid) {  repository7.SaveOtherEvent(otherEvent);  return RedirectToAction(“AnnualEventManaging”); }  else {  return View(otherEvent); } | PASS |

**UT-48** Class AdminController : ActionResult AddActivityClip(ActivityClip activityClip)

**Test Date** 16 May 2015

**Description** Test for adding activity clips by managing of Administer

|  |  |  |  |
| --- | --- | --- | --- |
| **ID** | **Expected Result** | **Actual Result** | **Pass/Fail** |
| 1 | if (ModelState.IsValid) {  repository4.SaveActivityClip(activityClip);  return RedirectToAction(“ActivityClipManaging"); }  else {  return View(activityClip); } | if (ModelState.IsValid) {  repository4.SaveActivityClip(activityClip);  return RedirectToAction(“ActivityClipManaging"); }  else {  return View(activityClip); } | PASS |

**UT-49** Class AdminController : ActionResult ReplyDhammaQuestion(DhammaQA dhammaQA)

**Test Date** 16 May 2015

**Description** Test for reply dharma question and answer by managing of Administer

|  |  |  |  |
| --- | --- | --- | --- |
| **ID** | **Expected Result** | **Actual Result** | **Pass/Fail** |
| 1 | if (ModelState.IsValid) {  repository5.SaveDhammaQA(dhammaQA);  return RedirectToAction(“DhammaQAManaging");  } else {  return View(dhammaQA); } | if (ModelState.IsValid) {  repository5.SaveDhammaQA(dhammaQA);  return RedirectToAction(“DhammaQAManaging");  } else {  return View(dhammaQA); } | PASS |

**UT-50** Class HomeController : FileContentResult GetWelcomeImage(int imageId)

**Test Date** 16 May 2015

**Description** Test for show slide images of home page by managing of Administer

|  |  |  |  |
| --- | --- | --- | --- |
| **ID** | **Expected Result** | **Actual Result** | **Pass/Fail** |
| 1 | WelcomeImage welimg = repository.WelcomeImages.FirstOrDefault(p => p.ImageID == imageId);  if (welimg != null) {  return File(welimg.ImageData, welimg.ImageMimeType);  } else {  return null;}} | WelcomeImage welimg = repository.WelcomeImages.FirstOrDefault(p => p.ImageID == imageId);  if (welimg != null) {  return File(welimg.ImageData, welimg.ImageMimeType);  } else {  return null;}} | Pass |

**UT-51** Class HomeController : FileContentResult GetGalleryImage(int imageId)

**Test Date** 16 May 2015

**Description** Test for get image of gallery by managing of Administer

|  |  |  |  |
| --- | --- | --- | --- |
| **ID** | **Expected Result** | **Actual Result** | **Pass/Fail** |
| 1 | GalleryImage galimg = repository2.GalleryImages.FirstOrDefault(p => p.ImageID == imageId);  if (galimg != null) {  return File(galimg.ImageData, galimg.ImageMimeType);  } else {  return null; } } | GalleryImage galimg = repository2.GalleryImages.FirstOrDefault(p => p.ImageID == imageId);  if (galimg != null) {  return File(galimg.ImageData, galimg.ImageMimeType);  } else {  return null; } } | PASS |

**UT-52** Class HomeController : ActionResult Index()

**Test Date** 16 May 2015

**Description** Test for view home page

|  |  |  |  |
| --- | --- | --- | --- |
| **ID** | **Expected Result** | **Actual Result** | **Pass/Fail** |
| 1 | ViewBag.Message = “หน้าแรก”;  return View(repository.WelcomeImages); | ViewBag.Message = “หน้าแรก”;  return View(repository.WelcomeImages); | PASS |

**UT-53** Class HomeController : ActionResult Background()

**Test Date** 16 May 2015

**Description** Test for view background page

|  |  |  |  |
| --- | --- | --- | --- |
| **ID** | **Expected Result** | **Actual Result** | **Pass/Fail** |
| 1 | ViewBag.Message = “ประวัติวัด”;  return View(repository3.Backgrounds); | ViewBag.Message = “ประวัติวัด”;  return View(repository3.Backgrounds); | PASS |

**UT-54** Class HomeController : ActionResult AnnualEvent()

**Test Date** 16 May 2015

**Description** Test for view annual event page

|  |  |  |  |
| --- | --- | --- | --- |
| **ID** | **Expected Result** | **Actual Result** | **Pass/Fail** |
| 1 | ViewBag.Message = “ข่าวสาร-กิจกรรม”;  return View(repository7.AnnualEvents); | ViewBag.Message = “ข่าวสาร-กิจกรรม”;  return View(repository7.AnnualEvents); | PASS |

**UT-55** Class HomeController : ActionResult AnnualEventDetail(int aEventID)

**Test Date** 16 May 2015

**Description** Test for view annual event detail page

|  |  |  |  |
| --- | --- | --- | --- |
| **ID** | **Expected Result** | **Actual Result** | **Pass/Fail** |
| 1 | AnnualEvent annualEvent = repository7.AnnualEvents.FirstOrDefault(p => p.EventID == aEventID);  return View(annualEvent); | AnnualEvent annualEvent = repository7.AnnualEvents.FirstOrDefault(p => p.EventID == aEventID);  return View(annualEvent); | PASS |

**UT-56** Class HomeController : ActionResult DhammaContent()

**Test Date** 16 May 2015

**Description** Test for view dharma content page

|  |  |  |  |
| --- | --- | --- | --- |
| **ID** | **Expected Result** | **Actual Result** | **Pass/Fail** |
| 1 | ViewBag.Message = “สาระธรรม”;  return View(); | ViewBag.Message = “สาระธรรม”;  return View(); | PASS |

**UT-57** Class HomeController : ActionResult DhammaQA()

**Test Date** 16 May 2015

**Description** Test for view dhamma question and answer page

|  |  |  |  |
| --- | --- | --- | --- |
| **ID** | **Expected Result** | **Actual Result** | **Pass/Fail** |
| 1 | ViewBag.Message = “ถาม-ตอบธรรม”;  return View(repository5.DhammaQAs); | ViewBag.Message = “ถาม-ตอบธรรม”;  return View(repository5.DhammaQAs); | PASS |

**UT-58** Class HomeController : ViewResult SendDhammaQ()

**Test Date** 16 May 2015

**Description** Test for view sending dhamma question and answer page

|  |  |  |  |
| --- | --- | --- | --- |
| **ID** | **Expected Result** | **Actual Result** | **Pass/Fail** |
| 1 | return View("SendDhammaQ", new DhammaQA()); | return View("SendDhammaQ", new DhammaQA()); | PASS |

**UT-59** Class HomeController : ActionResult SendDhammaQ(DhammaQA dhammaQA)

**Test Date** 16 May 2015

**Description** Test for send dhamma question and answer

|  |  |  |  |
| --- | --- | --- | --- |
| **ID** | **Expected Result** | **Actual Result** | **Pass/Fail** |
| 1 | if (ModelState.IsValid) {  repository5.SaveDhammaQA(dhammaQA);  return RedirectToAction(“DhammaQA”); } else {  return View(dhammaQA); } | if (ModelState.IsValid) {  repository5.SaveDhammaQA(dhammaQA);  return RedirectToAction(“DhammaQA”); } else {  return View(dhammaQA); } | PASS |

**UT-60** Class HomeController : ActionResult Gallery()

**Test Date** 16 May 2015

**Description** Test for view gallery page

|  |  |  |  |
| --- | --- | --- | --- |
| **ID** | **Expected Result** | **Actual Result** | **Pass/Fail** |
| 1 | ViewBag.Message = “แกลอรี่";  return View(repository2.GalleryImages); | ViewBag.Message = “แกลอรี่";  return View(repository2.GalleryImages); | PASS |

**UT-61** Class HomeController : ActionResult ActivityClip()

**Test Date** 16 May 2015

**Description** Test for view activity clip page

|  |  |  |  |
| --- | --- | --- | --- |
| **ID** | **Expected Result** | **Actual Result** | **Pass/Fail** |
| 1 | ViewBag.Message = “วิดีโอกิจกรรม";  return View(repository4.ActivityClips); | ViewBag.Message = “วิดีโอกิจกรรม";  return View(repository4.ActivityClips); | PASS |

**UT-62** Class HomeController : ActionResult InternetTV()

**Test Date** 16 May 2015

**Description** Test for view internet TV page

|  |  |  |  |
| --- | --- | --- | --- |
| **ID** | **Expected Result** | **Actual Result** | **Pass/Fail** |
| 1 | ViewBag.Message = “อินเตอร์เน็ตทีวี";  return View(); | ViewBag.Message = “อินเตอร์เน็ตทีวี";  return View(); | PASS |

**UT-63** Class HomeController : ActionResult Map()

**Test Date** 16 May 2015

**Description** Test for map page

|  |  |  |  |
| --- | --- | --- | --- |
| **ID** | **Expected Result** | **Actual Result** | **Pass/Fail** |
| 1 | ViewBag.Message = “แผนที่วัด”;  return View(); | ViewBag.Message = “แผนที่วัด”;  return View(); | PASS |

**UT-64** Class HomeController : ActionResult Contact()

**Test Date** 16 May 2015

**Description** Test for view contact page

|  |  |  |  |
| --- | --- | --- | --- |
| **ID** | **Expected Result** | **Actual Result** | **Pass/Fail** |
| 1 | ViewBag.Message = “ติดต่อวัด”;  return View(“Contact”, new Contact()); | ViewBag.Message = “ติดต่อวัด”;  return View(“Contact”, new Contact()); | PASS |

**UT-65** Class HomeController : ActionResult Contact(Contact -objModelMail)

**Test Date** 16 May 2015

**Description** Test for sending email for contact

|  |  |  |  |
| --- | --- | --- | --- |
| **ID** | **Expected Result** | **Actual Result** | **Pass/Fail** |
| 1 | if (ModelState.IsValid) {  repository6.SaveContact(\_objModelMail);  MailMessage mail = new MailMessage(); mail.To.Add(“se552115005@vr.camt.info"); mail.From = new MailAddress(\_objModelMail.SenderEmail);  mail.Subject = \_objModelMail.ContactTitle;  string Body = "โปรดติตต่อกลับ: "+\_objModelMail.Sender+"\n อีเมลล์: "+\_objModelMail.SenderEmail+"\n เรื่อง: “+\_objModelMail.ContactMessage;  mail.Body = Body;  mail.IsBodyHtml = true;  SmtpClient smtp = new SmtpClient();  smtp.Host = "smtp.gmail.com";  smtp.Port = 587;  smtp.UseDefaultCredentials = true;  smtp.Credentials = new System.Net.NetworkCredential("se552115005@vr.camt.info","552115005");// Enter seders User name and password  smtp.EnableSsl = true;  smtp.Send(mail);  return View("Contact", \_objModelMail);} else {  return View(); } | if (ModelState.IsValid) {  repository6.SaveContact(\_objModelMail);  MailMessage mail = new MailMessage(); mail.To.Add(“se552115005@vr.camt.info"); mail.From = new MailAddress(\_objModelMail.SenderEmail);  mail.Subject = \_objModelMail.ContactTitle;  string Body = "โปรดติตต่อกลับ: "+\_objModelMail.Sender+"\n อีเมลล์: "+\_objModelMail.SenderEmail+"\n เรื่อง: “+\_objModelMail.ContactMessage;  mail.Body = Body;  mail.IsBodyHtml = true;  SmtpClient smtp = new SmtpClient();  smtp.Host = "smtp.gmail.com";  smtp.Port = 587;  smtp.UseDefaultCredentials = true;  smtp.Credentials = new System.Net.NetworkCredential("se552115005@vr.camt.info","552115005");// Enter seders User name and password  smtp.EnableSsl = true;  smtp.Send(mail);  return View("Contact", \_objModelMail);} else {  return View(); } | PASS |