4/25/2016

Scenario and Testing Result

KTF UI’s Website

By BigBang

Table of Contents

[1. Unit Testing 2](#_Toc449451664)

[2. Function Testing 22](#_Toc449451665)

[3. User Interface Testing 25](#_Toc449451666)

[4. Load and Stress Testing 29](#_Toc449451667)

[5. Configuration Testing 30](#_Toc449451668)

[6. Security and Access Testing 32](#_Toc449451669)

In this documentation, we will be describing the details of the tests that need to be performed for each target test item that was identified. These tests will be organized into the following sections:

1. Unit Testing
2. Function Testing
3. User Interface Testing
4. Load and Stress Testing
5. Configuration Testing
6. Security and Access Control Testing

Moreover, for each of these test motivators, test cases will be described in detail. For each test case, we will provide a description of the test case, the inputs (or steps to reproduce) of the test case, and the outputs (the expected results) of the test case.

## 1. Unit Testing

Unit testing will test individual components along with their functions in isolation. This low level form of testing will include black box testing and white box testing. In black box testing, the function’s boundaries will be tested to see if any errors occur there. White box testing will verify that all the paths in the function are correct through basis path testing.

#### **1.1 Function pureal\_init\_style**

Initiation of style.css file.

***1.1.1 Black Box Testing***

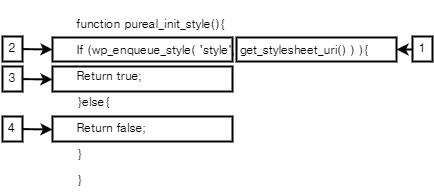
Test case 1 => style.css file existed

Test case 2 => style.css file not existed

|  |  |  |
| --- | --- | --- |
| Tester Name | Marvin Mitchell | |
| Test Date | April, 25th 2016 | |
| Test Case | 1 | 2 |
| Expected Results | The web styling will works | The web work will properly, but there is no styling |
| Actual Results | The web styling works | The web work properly, but there is no styling |
| Bug Found? | No | No |

***1.1.2 White Box Testing***

Basis Path Testing

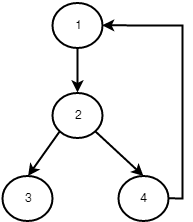


|  |  |
| --- | --- |
| Tester Name | Anissa Lintang |
| Path 1 | 1-2-3 |
| Path 2 | 1-2-4 |

|  |  |
| --- | --- |
| Path 1 | 1-2-3 |
| Variables | Style.css file existed |
| Results | Function will return true |

|  |  |
| --- | --- |
| Path 2 | 1-2-4 |
| Variables | Style.css not existed |
| Results | Function will return false |

Path diagram for function *pureal\_init\_style*



#### **1.2 Function pureal\_init\_wp\_setup**

Initiation of wordpress related file and functions through wordpress hooks

***1.2.1 Black Box Testing***

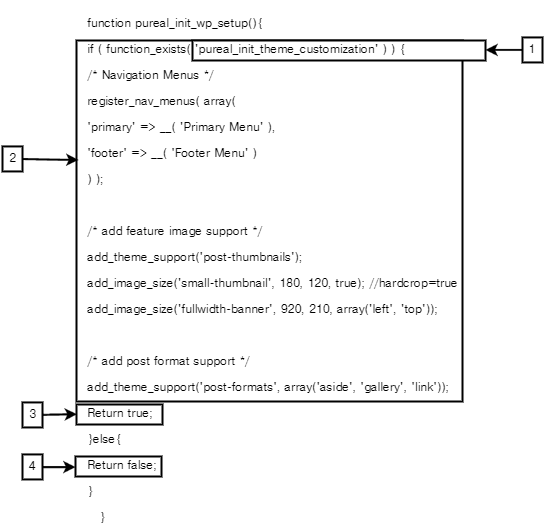
Test case 1 => correct function call

Test case 2 => wrong function call

|  |  |  |
| --- | --- | --- |
| Tester Name | Marvin Mitchell | |
| Test Date | April, 25th 2016 | |
| Test Case | 1 | 2 |
| Expected Results | The function will return true | The server will throw an error |
| Actual Results | The function return true | The server throw an error |
| Bug Found? | No | No |

***1.2.2 White Box Testing***

Basis Path Testing

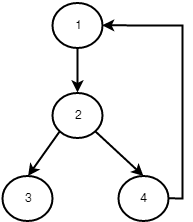


|  |  |
| --- | --- |
| Tester Name | Anissa Lintang |
| Path 1 | 1-2-3 |
| Path 2 | 1-2-4 |

|  |  |
| --- | --- |
| Path 1 | 1-2-3 |
| Variables | Function existed |
| Results | Function return correct |

|  |  |
| --- | --- |
| Path 2 | 1-2-4 |
| Variables | Function not existed |
| Results | Function return false |

Path diagram for function *pureal\_init\_wp\_setup*



#### **1.3 Function pureal\_sanitize\_image**

*Filter images to be included in the post*

***1.3.1 Black Box Testing***

Test case 1 => the file is not image

Test case 2 => the file is an unknown image

Test case 3 => the file is an image

|  |  |  |  |
| --- | --- | --- | --- |
| Tester Name | Marvin Mitchell | | |
| Test Date | April, 25th 2016 | | |
| Test Case | 1 | 2 | 3 |
| Expected Results | The function will return false | The function will return false | The function will return true |
| Actual Results | The function return false | The function return false | The function return true |
| Bug Found? | No | No | No |

***1.3.2 White Box Testing***

Basis Path Testing

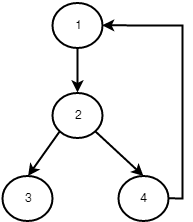


|  |  |
| --- | --- |
| Tester Name | Anissa Lintang |
| Path 1 | 1-2-3 |
| Path 2 | 1-2-4 |

|  |  |
| --- | --- |
| Path 1 | 1-2-3 |
| Variables | File is an image with valid mime type |
| Results | Function return correct |

|  |  |
| --- | --- |
| Path 2 | 1-2-4 |
| Variables | File is an image without valid mime type or not an image |
| Results | Function return false and back to default |

Path diagram for function *pureal\_sanitize\_image*



#### **1.4 Function pureal\_init\_widget**

Initialize widget location for wordpress

***1.4.1 Black Box Testing***

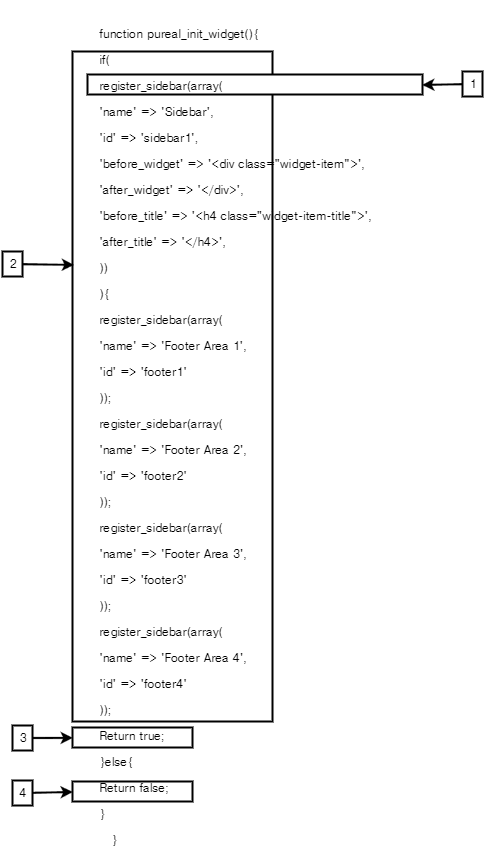
Test case 1 => wrong widget location

Test case 2 => wrong wordpress function call

|  |  |  |
| --- | --- | --- |
| Tester Name | Marvin Mitchell | |
| Test Date | April, 25th 2016 | |
| Test Case | 1 | 2 |
| Expected Results | The function will return false | The server will throws an error |
| Actual Results | The function return false | The server throws an error |
| Bug Found? | No | No |

***1.4.2 White Box Testing***

Basis Path Testing

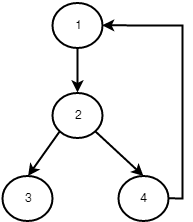


|  |  |
| --- | --- |
| Path 1 | 1-2-3 |
| Path 2 | 1-2-4 |

|  |  |
| --- | --- |
| Path 1 | 1-2-3 |
| Variables | Name and id are registered |
| Results | Function return correct |

|  |  |
| --- | --- |
| Path 2 | 1-2-4 |
| Variables | Name and id are not registered |
| Results | Function return false |

Path diagram for function *pureal\_init\_widget*



#### **1.5 Function pureal\_set\_excerpt\_length**

Edit paragraph excerpt length

***1.5.1 Black Box Testing***

Test case 1 => set the excerpt to be higher than 0

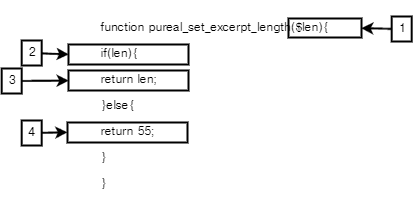
Test case 2 => set the excerpt to 0

Test case 3 => set the excerpt to be lower than 0

|  |  |  |  |
| --- | --- | --- | --- |
| Tester Name | Marvin Mitchell | | |
| Test Date | April, 25th 2016 | | |
| Test Case | 1 | 2 | 3 |
| Expected Results | The paragraph will be trimmed to the specified excerpt length | The paragraph will return false and input the default length | The paragraph will return false and input the default length |
| Actual Results | The paragraph trimmed to the specified excerpt length | The paragraph return false and input the default length | The paragraph return false and input the default length |
| Bug Found? | No | No | No |

***1.5.2 White Box Testing***

Basis Path Testing

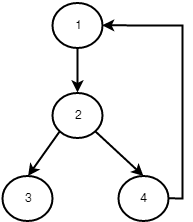


|  |  |
| --- | --- |
| Tester Name | Anissa Lintang |
| Path 1 | 1-2-3 |
| Path 2 | 1-2-4 |

|  |  |
| --- | --- |
| Path 2 | 1-2-3 |
| Variables | Length input higher than 0 |
| Results | Function return true and input the length |

|  |  |
| --- | --- |
| Path 2 | 1-2-4 |
| Variables | Length input is 0 or less |
| Results | Function return false and input the default |

Path diagram for function *pureal\_set\_excerpt\_length*



#### **1.6 Function pureal\_init\_theme\_customization**

Initialization of theme customizer API. Customizer API responsible for dynamically change the theme styling.

***1.6.1 Black Box Testing***

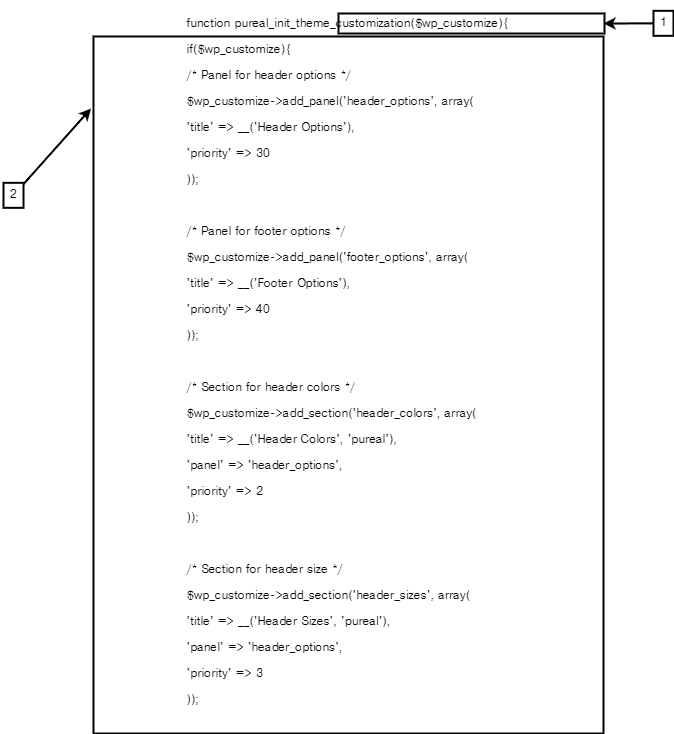
Test case 1 => wrong Customizer function call

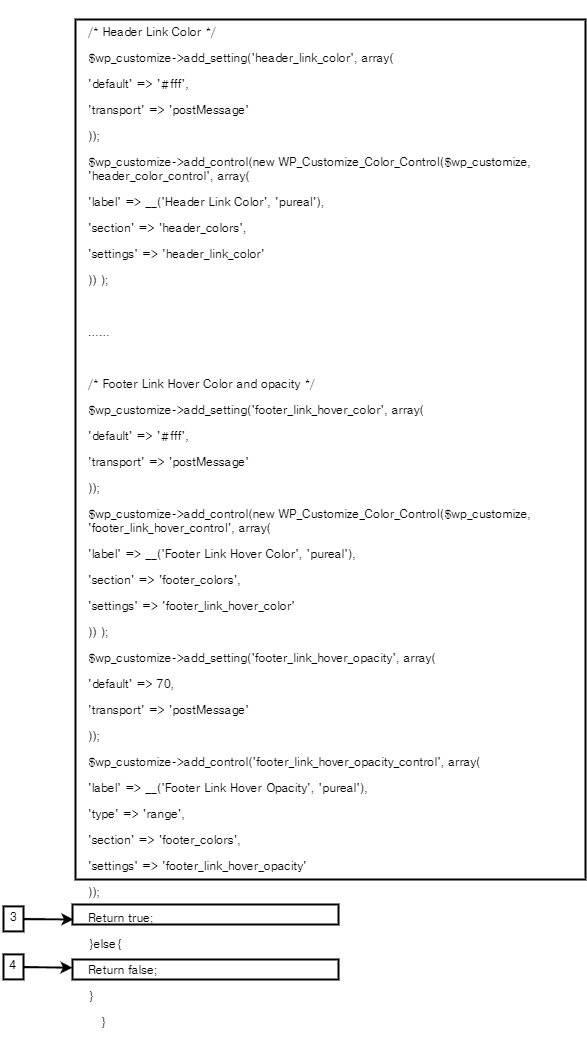
Test case 2 =>call the right Customizer function call

|  |  |  |
| --- | --- | --- |
| Tester Name | Marvin Mitchell | |
| Test Date | April, 25th 2016 | |
| Test Case | 1 | 2 |
| Expected Results | The server will throw an error. | The styling of the theme will be able to change dynamically according to the user needs |
| Actual Results | The server throw an error. | The styling of the theme able to change dynamically according to the user needs |
| Bug Found? | No | No |

***1.6.2 White Box Testing***

Basis Path Testing



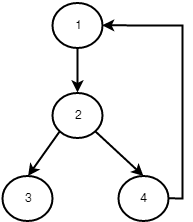


|  |  |
| --- | --- |
| Tester Name | Anissa Lintang |
| Path 1 | 1-2-3 |
| Path 2 | 1-2-4 |

|  |  |
| --- | --- |
| Path 1 | 1-2-3 |
| Variables | Function customizer existed |
| Results | Function return correct |

|  |  |
| --- | --- |
| Path 2 | 1-2-4 |
| Variables | Function customizer not existed |
| Results | Function return false |

Path diagram for function *pureal\_init\_theme\_customization*



#### ***1.7 Function* pureal\_update\_custom\_css**

The function needed to update style.css dynamically

***1.7.1 Black Box Testing***

Test case 1 => wrong function call

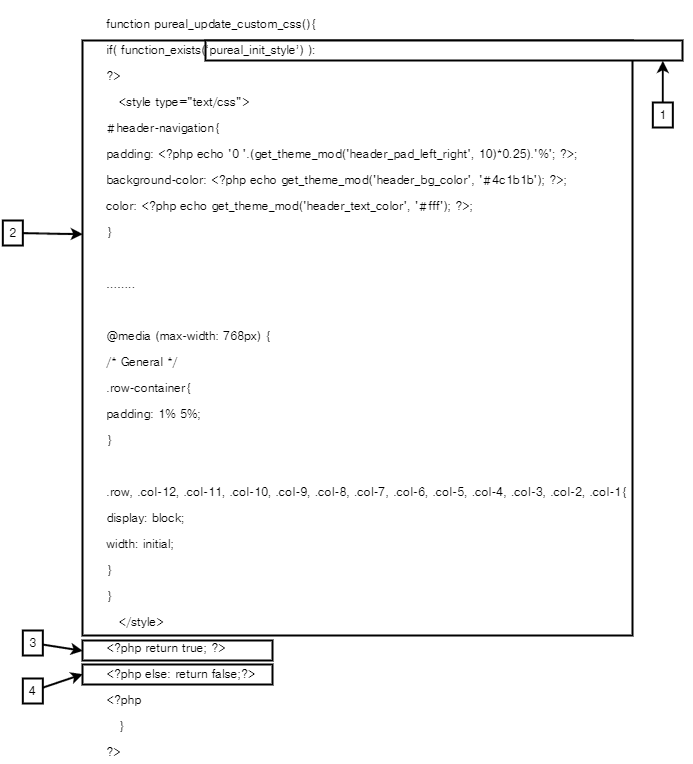
Test case 2 => Wrong css property

Test case 3 => right function call

|  |  |  |  |
| --- | --- | --- | --- |
| Tester Name | Marvin Mitchell | | |
| Test Date | April, 25th 2016 | | |
| Test Case | 1 | 2 | 3 |
| Expected Results | The server will throw an error | The web styling will behave weird. | The style.css file will be updated accordingly. |
| Actual Results | The server throw an error | The web styling behave weird. | The style.css file updated accordingly. |
| Bug Found? | No | No | No |

***1.7.2 White Box Testing***

Basis Path Testing

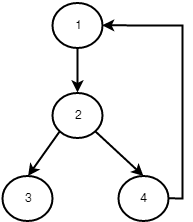


|  |  |
| --- | --- |
| Tester Name | Anissa Lintang |
| Path 1 | 1-2-3 |
| Path 2 | 1-2-4 |

|  |  |
| --- | --- |
| Path 1 | 1-2-3 |
| Variables | Function *pureal\_init\_style* existed |
| Results | Php return correct |

|  |  |
| --- | --- |
| Path 2 | 1-2-4 |
| Variables | Function *pureal\_init\_style* not existed |
| Results | Php return false |

Path diagram for function *pureal\_update\_custom\_css*



#### **1.8 Function pureal\_preview\_js**

The javascript needed for Customizer API to dynamically change the theme styling

***1.8.1 Black Box Testing***

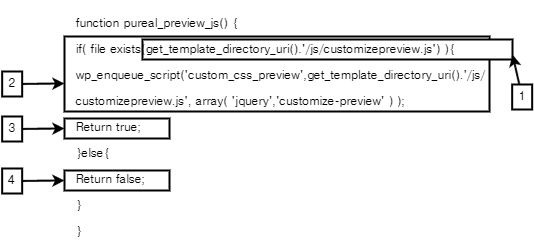
Test case 1 => wrong function call

Test case 2 => include the right javascript file

|  |  |  |
| --- | --- | --- |
| Tester Name | Marvin Mitchell | |
| Test Date | April, 25th 2016 | |
| Test Case | 1 | 2 |
| Expected Results | The server will throw an error | The web styling will change dynamically according to the user needs. |
| Actual Results | The server throw an error | The web styling change dynamically according to the user needs. |
| Bug Found? | No | No |

***1.8.2 White Box Testing***

Basis Path Testing

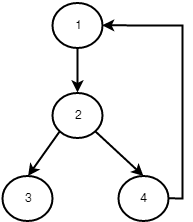


|  |  |
| --- | --- |
| Tester Name | Anissa Lintang |
| Path 1 | 1-2-3 |
| Path 2 | 1-2-4 |

|  |  |
| --- | --- |
| Path 1 | 1-2-3 |
| Variables | Template directory correct and include javascript file |
| Results | Function return correct |

|  |  |
| --- | --- |
| Path 2 | 1-2-4 |
| Variables | Template directory wrong and exclude javascript file |
| Results | Function return false |

Path diagram for function *pureal\_preview\_js*



#### **1.9 Function yScrollHandler**

The javascript function to trigger css3 animation when scrolling occur.

***1.9.1 Black Box Testing***

Test case 1 => animations array is empty

Test case 2 => animations array length is greater than 0

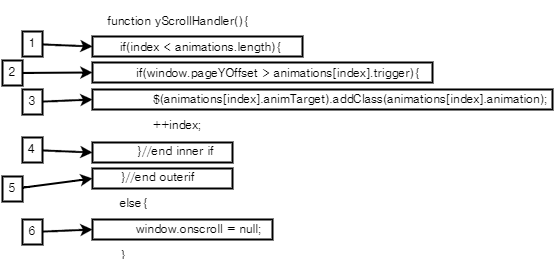
Test case 2.1 => window scroll offset is bigger than the animation height trigger

Test case 2.2 => window scroll offset is smaller than the animation height trigger

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Tester Name | Marvin Mitchell | | | |
| Test Date | April, 25th 2016 | | | |
| Test Case | 1 | 2 | 2.1 | 2.2 |
| Expected Results | The window scroll event will disabled | The scroll offset will be checked against all animation height trigger | Animation will occur | Animation will not occur |
| Actual Results | The window scroll event disabled | The scroll offset checked against all animation height trigger | Animation occured | Animation not occured |
| Bug Found? | No | No | No | No |

***1.9.2 White Box Testing***

Basis Path Testing



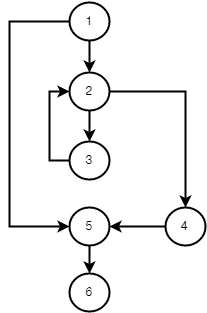
|  |  |
| --- | --- |
| Tester Name | Anissa Lintang |
| Path 1 | 1-2-3 |
| Path 2 | 1-5-6 |
| Path 3 | 1-2-4-5-6 |

|  |  |
| --- | --- |
| Path 1 | 1-2-3 |
| Variables | Animations array length is greater than 0 and window scroll offset is bigger than the animation height trigger |
| Results | Window scroll enabled and animation occured |

|  |  |
| --- | --- |
| Path 2 | 1-5-6 |
| Variables | Animations array is empty |
| Results | The window scroll event disabled and animation not occured |

|  |  |
| --- | --- |
| Path 3 | 1-2-4-5-6 |
| Variables | Animations array length is greater than 0 but window scroll offset is lesser than the animation height trigger |
| Results | The window scroll event disabled and animation not occured |

Path diagram for function *yScrollHandler*



## 2. Function Testing

This section is concerned with testing the functions (or requirements) of the software. This is a critical aspect of the testing effort, as it ensures that the software meets the requirements, and thus ensures acceptance by the users. Despite time restrictions, we have included all the major product functions as well as test cases for each one.The following sections are devoted to the major functions that were selected as testing targets.

|  |  |
| --- | --- |
| Test Case 1 | Request profile |
| Tester Name | Dion Edo |
| Test Date | April, 25th 2016 |
| Test Case Description | User can request and access (see) profile from the website |
| Input (Steps) | 1. Open the website 2. Select and Open the profile page |
| Expected Results | The website will shows the profile that the user access |
| Actual Results | The website shows the profile that the user access |

|  |  |
| --- | --- |
| Test Case 2 | Request information |
| Tester Name | Dion Edo |
| Test Date | April, 25th 2016 |
| Test Case Description | User can request and access (see) information from the website |
| Input (Steps) | 1. Open the website 2. Select and Open the page contained information the user want to know |
| Expected Results | The website will shows the information that the user access |
| Actual Results | The website shows the information that the user access |

|  |  |
| --- | --- |
| Test Case 3 | Request multimedia |
| Tester Name | Dion Edo |
| Test Date | April, 25th 2016 |
| Test Case Description | User can request and access (see) multimedia from the website |
| Input (Steps) | 1. Open the website 2. Select and Open the page contained multimedia |
| Expected Results | The website will shows and play (if video) the multimedia that the user access |
| Actual Results | The website shows and play the slideshow multimedia that the user access |

|  |  |
| --- | --- |
| Test Case 4 | Insert content |
| Tester Name | Dion Edo |
| Test Date | April, 25th 2016 |
| Test Case Description | Admin can insert new content to the website |
| Input (Steps) | 1. Open the website 2. Admin login to dashboard 3. Admin select new content type 4. Insert the content |
| Expected Results | The website will have a new content that corresponding to the admin content inserted |
| Actual Results | The website have a new content that corresponding to the admin content inserted |

|  |  |
| --- | --- |
| Test Case 5 | Edit and Update content |
| Tester Name | Dion Edo |
| Test Date | April, 25th 2016 |
| Test Case Description | Admin can edit and update existing content to the website |
| Input (Steps) | 1. Open the website 2. Admin login to dashboard 3. Admin select existing content that want to edit and update 4. Edit and update the content |
| Expected Results | The website will have a content that edited and updated corresponding to the admin content |
| Actual Results | The website have a content that edited and updated corresponding to the admin content |

## 3. User Interface Testing

To test the User Interface, each functionality described in the design document will be verified to see if it has been implemented correctly. A schema will be used to test (unit), what is the purpose of the test (what is tested), what are the inputs (from the user for instance), what is the expected result and also what is the effective (real result).

|  |  |
| --- | --- |
| Test Case 1 | Test Slideshow Image Profile |
| Tester Name | Dion Edo |
| Test Date | April, 25th 2016 |
| Test Case Description | User can click three bullets in order to change the big profile picture |
| Input (Steps) | 1. Open the website 2. Click Bullets Button in slideshow |
| Expected Results | The website will shows the picture multimedia that the user clicked |
| Actual Results | Picture slide to the left change to another picture without clicking the bullet button or not. |

|  |  |
| --- | --- |
| Test Case 2 | Check Achievement Page |
| Tester Name | Dion Edo |
| Test Date | April, 25th 2016 |
| Test Case Description | User can request and access (see) the whole Achievement description in Achievement Page |
| Input (Steps) | 1. Open the website 2. Click menu page Achievement Page |
| Expected Results | The website will change directly into Achievement Page |
| Actual Results | The website go to Achievement page directly |

|  |  |
| --- | --- |
| Test Case 3 | Check Project Page |
| Tester Name | Dion Edo |
| Test Date | April, 25th 2016 |
| Test Case Description | User can request and access (see) the whole Project info and description in Project Page |
| Input (Steps) | 1. Open the website 2. Click menu page Project Page |
| Expected Results | The website will change directly into Project Page |
| Actual Results | The website go to Project page directly |

|  |  |
| --- | --- |
| Test Case 4 | Check Gallery Page |
| Tester Name | Dion Edo |
| Test Date | April, 25th 2016 |
| Test Case Description | User can request and access (see) the whole Gallery info and description in Gallery Page |
| Input (Steps) | 1. Open the website 2. Click menu page Gallery Page |
| Expected Results | The website will change directly into Gallery Page |
| Actual Results | The website go to Gallery page directly |

|  |  |
| --- | --- |
| Test Case 5 | Check Contact Page |
| Tester Name | Dion Edo |
| Test Date | April, 25th 2016 |
| Test Case Description | User can request and access (see) the whole Contact info and description in Contact Page |
| Input (Steps) | 1. Open the website 2. Click menu page Contact Page |
| Expected Results | The website will change directly into Contact Page |
| Actual Results | The website go to Contact page directly |

|  |  |
| --- | --- |
| Test Case 6 | Check Social Media Access Link |
| Tester Name | Dion Edo |
| Test Date | April, 25th 2016 |
| Test Case Description | User can request and access (see) Social Media of KTF UI |
| Input (Steps) | 1. Open the website 2. Click each social media button link (Youtube,Instagram, Facebook and Twitter) |
| Expected Results | The user bring into social media which button clicked (Twitter, FB, Youtube and Instagram) |
| Actual Results | The website brings user to the social media which clicked |

|  |  |
| --- | --- |
| Test Case 7 | Check Integrity Data each Page |
| Tester Name | Dion Edo F |
| Test Date | April, 25th 2016 |
| Test Case Description | Test conducted in order to see the data matching with the data given. |
| Input (Steps) | 1. Open the website 2. See every page and matching the data(title, description, information, icon ) with the data given by client. |
| Expected Results | Data matching. |
| Actual Results | Because of client don’t give data about all of description yet. So actual results is there is no data yet. |

|  |  |
| --- | --- |
| Test Case 8 | Check Font each page |
| Tester Name | Dion Edo F |
| Test Date | April, 25th 2016 |
| Test Case Description | Client want the website font using Din regular, century schoolbook, stackyard personal, din light alternate, din medium |
| Input (Steps) | 1. Open the website 2. Checking matching, consistency every font in website with client expected |
| Expected Results | Font has consistency and using font style suitable with client expectancy |
| Actual Results | Font has consistency and using font style suitable with client expectancy |

|  |  |
| --- | --- |
| Test Case 9 | Check color in element each page |
| Tester Name | Dion Edo F |
| Test Date | April, 25th 2016 |
| Test Case Description | Client want the color of the website is white, orange gold and grey |
| Input (Steps) | 1. Open the website 2. Checking consistency color in header, footer, and link |
| Expected Results | Color has consistency and suitable with client expectancy |
| Actual Results | Header has white color, footer has grey color and link has orange gold color. All of it has consistency in every page. |

|  |  |
| --- | --- |
| Test Case 10 | Check Website Responsiveness |
| Tester Name | Dion Edo F |
| Test Date | April, 25th 2016 |
| Test Case Description | User can see the website through mobile gadget(Handphone and tablets) |
| Input (Steps) | 1. Open the website 2. Press Ctrl + Shift + M (Toggle Device Mode) 3. Choose device model device mobile S(320px), M(375px), L(425px), Tablet(768px), Laptop(1024px), Laptop L (1440px). |
| Expected Results | Website has responsiveness and be able to suitable of every model device. |
| Actual Results | Website has responsiveness and be able to suitable of every model device. The content responsively matching to every model device so that user still can see it clear. |

## 4. Load and Stress Testing

Load testing is normally concerned with testing the system beyond the limits it was designed for. However, due to restrictions we have placed on the number of visitor (unknown number ) that can access in one time, we cannot, for instance, test to see if the website works with maximum visitor (undefinitely yet). Therefore, we will be testing the website as close as possible to the limits it was designed for. In this scenario, we will re-evaluate the website’s functionalities and response times. The tools used for the test is Jmeter.

|  |  |
| --- | --- |
| Test Case 1 | Continues user visit |
| Tester Name | Kega Kurniawan |
| Test Date | April, 25th 2016 |
| Test Description | On this test the website will be tested with the continues visit from multiple user in every 5 second for 30 minutes, the test conducted with jmeter application |
| Input (Step) | 1. Open Jmeter Application 2. Make a “Thread Group” 3. In “Thread Group” Input “Number of Thread” set to 4, “Ramp-up Period” set to 2 and set “Loop” to forever 4. Add “HTTP request Defaults” to “Thread Group”, and fill “Server Name or IP” to ktfui.hol.es 5. Add “HTTP Cookie Manager” to “Thread Group” 6. Add “HTTP request” to “Thread Group”, and fill “Server Name or IP” to ktfui.hol.es 7. Add “Summary Report”, “View Result Tree”, “View Result in Table”, and “Graph Results” to “Thread Group” for better view result testing 8. Start the test along 30 minute 9. After 30 minute stop the test, and view the result |
| Expected Results | Error result from the website can be minimize. Standard deviation and latency from result test is not too high |
| Actual Results | From the testing already conduct error result from the test is 24,52% its mean from 100% request already made only 24,52% request with error. And Standard deviation from the test is 403,66 with is quite good because it’s still under the number of average time response 529 |

|  |  |
| --- | --- |
| Test Case 2 | Many user visit the website in 1 time |
| Tester Name | Kega Kurniawan |
| Test Date | April, 25th 2016 |
| Test Description | Access the websites with many user (the number of user will be described below) in 1 time |
| Input (Step) | 1. Open Jmeter Application 2. Make a “Thread Group” 3. In “Thread Group” Input “Number of Thread” set to 1000, “Ramp-up Period” set to 1 and set “Loop” to 1 4. Add “HTTP request Defaults” to “Thread Group”, and fill “Server Name or IP” to ktfui.hol.es 5. Add “HTTP Cookie Manager” to “Thread Group” 6. Add “HTTP request” to “Thread Group”, and fill “Server Name or IP” to ktfui.hol.es 7. Add “Summary Report”, “View Result Tree”, “View Result in Table”, and “Graph Results” to “Thread Group” for better view result testing 8. Start the test 9. View the result |
| Expected Results | Error result from the website can be maximize and standard deviation is high |
| Actual Results | From the testing already conduct error result from the test is 100% its mean all request to website is error. And Standard deviation from the test is too 1000 with is really bad because it’s still over the number of average time response 10. This problem happen maybe because we are using free hosting website, and free hosting website have very limited bandwidth. |

## 5. Configuration Testing

Configuration testing is concerned with testing the application under different environment configurations the users may have. For the KTF UI Website, we will be focusing on testing the website under different browser. As per the requirements document, this includes Mozilla Firefox, Google Chrome, IE and Safari. In order to simulate these different client environments, we will be using well-known different browser directly. In order to test the KTF UI Website under several different browser.

|  |  |
| --- | --- |
| Test Case 1 | Access website via Mozilla Firefox |
| Tester Name | Anissa Lintang |
| Test Date | April, 25th 2016 |
| Test Description | Test and see if the website can be accessed and work properly via browser Mozilla Firefox |
| Expected Results | Website will be successfully accessed and work properly |
| Actual Results | Website successfully accessed and work properly |

|  |  |
| --- | --- |
| Test Case 2 | Access website via Google Chrome |
| Tester Name | Anissa Lintang |
| Test Date | April, 25th 2016 |
| Test Description | Test and see if the website can be accessed and work properly via browser Google Chrome |
| Expected Results | Website will be successfully accessed and work properly |
| Actual Results | Website successfully accessed and work properly |

|  |  |
| --- | --- |
| Test Case 3 | Access website via Safari |
| Tester Name | Anissa Lintang |
| Test Date | April, 25th 2016 |
| Test Description | Test and see if the website can be accessed and work properly via browser Safari |
| Expected Results | Website will be successfully accessed and work properly |
| Actual Results | Website successfully accessed and work properly |

## 6. Security and Access Testing

Security and access control testing have to be conducted to ensure security system especially security for the admin system, it is necessary because this testing will be test the website how well can handle and prevent unauthorized user to access the admin system. In this testing we will conduct security system test for admin login, and see how well security system from admin login to prevent unauthorized user to enter the admin system. Because security and access system in this website has been already setup by CMS Wordpress, then we will only do black box testing for this testing.

|  |  |
| --- | --- |
| Test Case 1 | Login admin system using wrong username and wrong password |
| Tester Name | Kega Kurniawan |
| Test Date | April, 25th 2016 |
| Expected Results | Login will failed |
| Actual Results | Login failed |

|  |  |
| --- | --- |
| Test Case 2 | Login admin system using correct username and wrong password |
| Tester Name | Kega Kurniawan |
| Test Date | April, 25th 2016 |
| Expected Results | Login will failed |
| Actual Results | Login Failed |

|  |  |
| --- | --- |
| Test Case 3 | Login admin system using wrong username and correct password |
| Tester Name | Kega Kurniawan |
| Test Date | April, 25th 2016 |
| Expected Results | Login will failed |
| Actual Results | Login failed |

|  |  |
| --- | --- |
| Test Case 4 | Login admin system using correct username and correct password |
| Tester Name | Kega Kurniawan |
| Test Date | April, 25th 2016 |
| Expected Results | Login will succeed |
| Actual Results | Login success |

-END OF REPORT-