Dynamic Content Report

## Task 1 - Setup and test the development environment

# Organise access to a web server.

The web server that I used was WAMP Server

# Create a screen dump of this page.

This is the home page on the WAMP server.

## Task 2 - write a report on the development environment

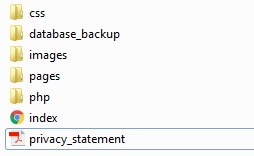
# A description of the hardware and software used in the development environment

The hardware that was used for this assignment was a PC at TAFE and my laptop for working at home. The software that I used was, WAMP server, notepad ++ and PHPmyAdmin.

# Describe who would be required to have access to the web server and why

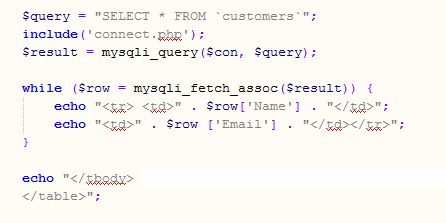
While coding the site, I was the only one who had access to the web server. After it was finished, the database administrator would have access to the web server so that not just anyone could alter the database.

# A proposed folder structure for your website

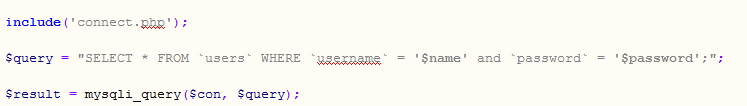
This is what the folder structure should look like.

## Task 3 - Develop dynamic website content

# Variables and arrays

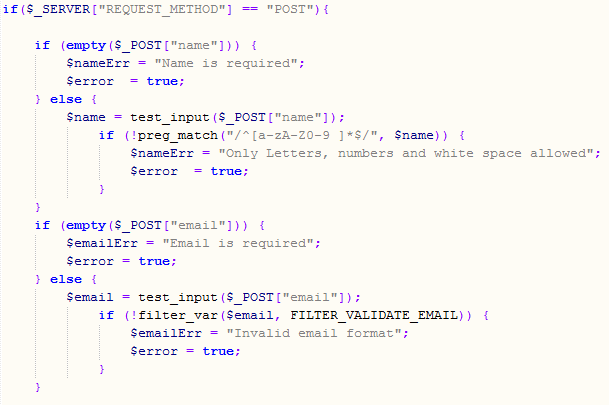
This is an example of an associative array. This is found on the login page. Where there is a ‘$’, this indicates a variable.

# Functions which use arguments



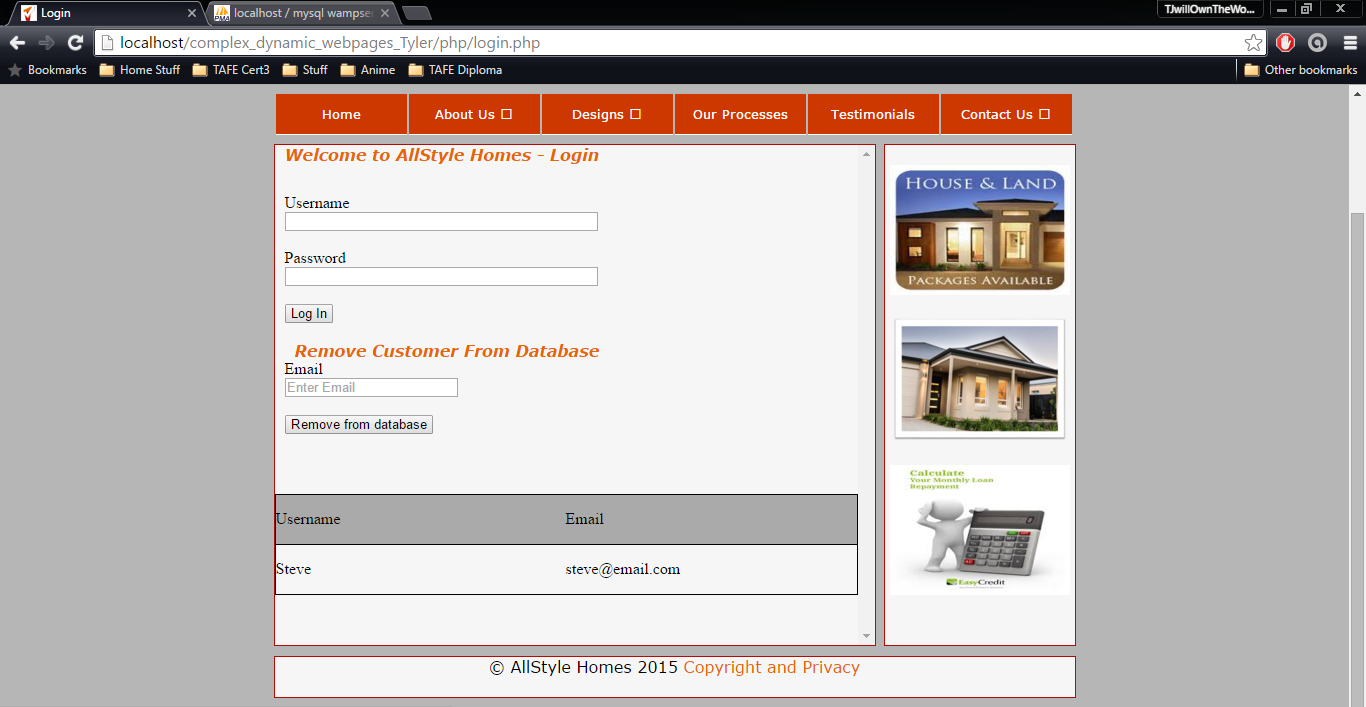
This is a function that uses arguments. This connects the login page to the database, and then it selects what is specified in the function to show. Then it posts or echoes the result of the function to the page.

# Conditionals (if/else statements)



This is the conditional (if/else) statement that I have used in the contacts page

# Facility for administrator to login and retrieve mailing list for follow up and to delete users.

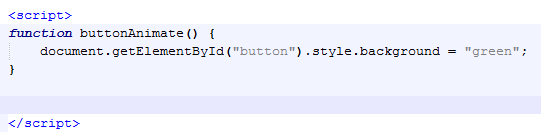


# You must include two (2) dynamic contents in your website. This can include any of the features that you have learnt. For example scrolling text or animation.

The following images are the dynamic content.

The first one is located in the testimonials page, and it makes what the clients say about AllStyle Homes appear by clicking on the image. This was done using JQuery

The second one is located on the login page and turns the button’s background colour change to green after clicking on it, unfortunately it only works for a second as the page reloads after clicking on the button. This was done using JavaScript.



# Develop the site (using JavaScript or some other method) to determine the time of day and change the css to display a night time viewing style.



This is the code that I have used to change the css based on time of day.

## Task 4 - Test the website and write a test report

# Re-validate your pages

All pages have been re-validated and all came up with no errors. I used <http://www.piliapp.com/php-syntax-check/> to check my php code that was used in the website.

# Re test to ensure Test the website to ensure it meets accessibility standards including visual impairment

Alt tags were used in images to allow screen readers to read out what the image is to the visually impaired. Tables were not used for text as this would have made it difficult for the visually impaired to understand what the page is on. I have not disabled screen resizing (scrolling) so that people can zoom in on the page.

# Load your pages to the development web server

See the screenshot at the end of the report.

# Write a report describing:

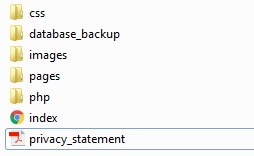
# The test procedures used

* I tested the site on Google Chrome and Firefox, ensuring that all aspects of the site were working as they should be on all screen sizes.

# Results of usability testing

* All works good. For the navigation, to get to the pages where the links drop down, hold your finger down on the main page for the subpages to keep them showing on the screen for tablet and mobile.

# The folder structure used



# Include the screen dump of the home page

