TEESSIDE UNIVERSITY

SCHOOL OF COMPUTING

FEED CLOUD REFLECTIVE ESSAY

AND

FEED CLOUD DEPLOYMENT DOCS

CONTENTS

CRITICAL REFLECTIVE ESSAY

Feed cloud and Yii version 1.1.16

Welcome to the world of Yii, what is Yii? Well it stands for "Yes it is" Is it fast? ... Is it secure? ... Is it professional? ... Is it right for my next project? ... Yes, it is! Is it easy to use for a beginner? No it isn't.

As a beginner I find myself most of the time feeling very confused, mostly as to where the mainly used folders and files live, such as the Controllers folder, the Models folder and the Views folder, although after a while and a lot of trial and error, it does become less confusing, unless you want to use "Themes" within Yii which is the route that I took, where there will be another Views folder and it is here that the whole thing can get rather confusing for a Newbie.

The overall approach for this project was to use the agile methodology, as this was thought to be the best solution and best way ahead because as a newcomer to using Yii, it was easy to break the application and did so on many occasions, but because of the methodology used, being iterative, it meant that keeping my last iteration, I could in fact go back one step to my previous version and look for any differences and to correct them.

Going back to the very beginning of my Yii application, it was necessary to design and develop a relational database which would be used within my application and it is the source of the Gii tool. This procedure is very important and probably the most difficult, as the relationships between entities have to be correct from the beginning or the application will not respond and will throw up exceptions or fatal errors. It is clear that I have made some errors within my ERD and am now paying for those errors, as I have discovered that I do not have the correct relationship between the Users table and the Modules table and thus am unable to insert and view the correct data within my creation of a new module.

After stating that the mistakes have been made within the database table relationships, it is possible and very likely a normal procedure whilst creating an

application within Yii that one would correct the mistakes and then run CRUD using the Gii tool again, but in this case time restraints prevent me from doing so as I do not want to take the risk of breaking the application altogether, although I feel confident enough that it would be a procedure that I could undertake in normal circumstances.

I felt it necessary to concatenate the users Fname and Lname, so they could be used to show the logged in user, instead of showing their User_ID and so spent a lot of time researching how to. I found a way of achieving that by creating a function called FullName and placing it in the file called WebUser.php within the Components folder and the using an echo statement calling that function.

In defense of Yii, I must admit that it has a lot going for it in terms of security, with it's built in layers of defense already in place and with the use of the MVC framework which is created automatically after running the Gii tool within the application.

The MVC folders that are created after running Gii contain all the forms for the use of CRUD within the application in the Views folder. The tool also creates the Controllers folder which contains the necessary actions needed for the forms within the views folder, such as index, view, create, update and delete.

It is true that Gii will produce about 80% of the code needed to run an application that will actually be worthwhile, but it also means that the developer needs at least to be able to create a small function, an if statement and a for loop to continue to develop and progress to enhance the applications development.

Incorporating EWURF into the equation was a worthwhile effort as it enforced the needs for the functional and non-functional functions as below.

List of Functions

- Allow user to register – The registration form was created using the user-form generated by the Gii tool, it was edited to show only the needed fields User_ID, Fname, Lname, Email, Password and a dropdown box showing contents of the Roles field which consisted of admin, staff and students. The admin user was excluded from the drop down as we are only wanting

to register staff or students. The password field has been encrypted using SHA512 and salted enabling a more secure way of storing passwords.

- Allow user to login – Once registered the user will login to the app and be directed via the dropdown in the menu bar to the function that is made available, depending on user type.
- Allow user to create new academic year – The user will be able to create new academic year by clicking on the dropdown menu and choosing "Create Module"
- Allow user to create new module – The user will use the create form from the views folder and upload a .PDF file, I have not quite implemented this procedure but have accomplished the ability to upload an image file and place it into the database in the Feedback table.
- Allow user to view feedback – Once created, the user has the ability to view his/her feedback file.
- Allow user to view swot feed analysis - Once created, the user has the ability to view, edit and delete his/her swot analysis.
- Allow user to view feedback action plan – Once created the user has the ability to view.

List of Non Functions

- Allow user to search new academic year
- Allow user to search module – The user will use the create form from the views folder and upload a .PDF file, I have not quite implemented this procedure but have accomplished the ability to upload an image file and place it into the database in the Feedback table.
- Allow user to contact the administrators
- Allow user to view the about page.

DEPLOYMENT DOCUMENTATION

Feed Cloud Deployment

After downloading Yii version 1.1.16 (December 21, 2014) it was unzipped where the folder and its contents were uploaded to the OS webserver at Teesside University which is a secure version of HTTP, it has SSL installed (secure socket layer) meaning all data transfers are encrypted. The version of Yii installed requires at least version 5.1 of PHP to run correctly, it has also been tested on APACHE HTTP webserver on both Linux and windows.

The approach taken in the deployment of this development was to use a sFTP client, WINSCP was used, as it is the easiest method, the alternative being the use of a console to use command prompt.

Once the **Yii** folder was uploaded and in place on the root of the server, (Software, 2016) which contained 3 folders (demos, requirements, and framework) and some instructions in plain text files, the demos and requirements folders needed to be placed into a Web-accessible directory, which was called **webApp** so that they could be accessed in the browser. The requirements folder is used for the purpose of testing the environment for Yii, using a checker script, it was removed along with the demo folder, as they were not needed.

The **framework** folder can be placed securely elsewhere, so long as the application's entry script php file contains the correct path to **yii.php**, although it is wise to keep to the original structure and hierarchy as it could cause problems further down the line with issues concerning the use of BASEURL and possibly THEME BASEURL.

There were two choices given as to how to implement the application, one was to develop entirely in Yii, utilizing the default theme or the second choice was to develop the application utilizing an appropriate HTML5 and CSS3 responsive framework. It was decided at an early stage that this project was to be developed in a responsive framework using a starter bootstrap3 theme which came with its

own JQuery slider, although the slider was not used, because it was decided to use a carousel widget within an extension to be used, which will be discussed further along.

The theme also had style sheets needed for bootstrap3 including, bootstrap.min.css and a style for the theme. The JS files included were bootstrap.min.js which allows the use of its component elements such as dialog popups, accordions and panels. The theme uses semantic HTML5 and CSS3, therefor using elements such as H1, header, section, labels within the forms and the aside element was imperative for search engine optimization. The head of the index page also contains some Meta tags with relevant content such as Feed Cloud, Portal, Swot, and Feedback, all words that are relevant.

Due to the fact the application was intended to reach as many users as possible, including people with disabilities, a lot of thought has gone into the process of accessibility, through the use of the Role within an element.

Although the front end of the application was a bootstrap3 starter theme or skeleton theme, the back end uses a completely different theme called "Abound" it was chosen because of its admin panel look and feel, but did cause some problems with the application because it came with its own style sheets and JavaScript's that tended to conflict with others being used, that situation was remedied after they were systematically disabled one at a time and then viewed to see the changes if any. After some reconfiguration with the layout in the admin section, the problem was solved and the site gelled nicely together with one set of JavaScript and css files.

The functional requirements for the application were enhanced and made a little easier to comprehend by the use of EWURF (*eWURF 3.2*, 2016) "The aim of this web application is to help you Capture, Analyze and Document requirements for your web app project".

The use of extensions: is a big plus for Yii as it allows updating the look and feel of the application, from the beginning during our initial iterations, the class was introduced to the extension called **YiiBooster**, in which was uploaded and added to the extensions folder, allowing the user to see its potential and benefits. This

extension was found to be very complicated in its initial installation, although it was up and running within minutes, but was not too impressed with its capabilities. Another extension was uploaded and installed, called **YiiStrap**, which is a library of files composed of the latest bootstrap3 css and js files.

Within YiiStrap's extension are widgets that can be used to replace most widgets used by Yii, and on most occasion's look and feel cutting edge and up to date as they were used in the ActiveForm and GridView widgets throughout the application. The widget used for the Carousel was from YiiBootstrap widget collection.

The creation of Functions: It was necessary to create functions as the application progressed to advance and enhance the look and feel, one such being the concatenation of the user first and surname called FullName. The function was created within the file called **WebUser.php** which is situated in the folder called Components and is used by the config file **main.php** inside the config folder. Another function used was called DefaultScope which is a part of Yii and used with a condition to allow the logged in user to view only his/her related records.

Installation of SEO extension: An SEO extension was installed and set up, although it has not been utilized, simply because it needed to be run through CRUD using the Gii tool and it was thought that maybe it was not worth the risk of breaking the application by running the tool again.

THE ERD

Figure 1 showing the data model for Feed - Cloud

REFERENCES

eWURF 3.2 (no date) Available at: https://www.wurf.co.uk/#admin (Accessed: 9 May 2016).

Fowler, M. and Sadalage, P. (2003) *Evolutionary database design.* Available at: http://martinfowler.com/articles/evodb.html (Accessed: 6 March 2016).

Hadlock, K. (2007) *Ajax for web application developers.* Indianapolis, IN: Sams Publishing

Letkowski, Jerzy, 2012. "Doing database design with MySQL", ISSN: 2327-5359*,* *Journal of technology research*, Date: 01.01.2015, Volume: Page: 1.

Software, Y. (2016) *Getting started: Installation.* Available at: http://www.yiiframework.com/doc/guide/1.1/en/quickstart.installation (Accessed: 9 May 2016)