**Earring Holler – ITECH3108 Lab Assignment 2**

**Teddy J. Noyau – 30320407**

**Statement of Completion**

Abstract

This document discusses the development process of Earring Holler, a discussion forum for Earring enthusiasts. This document includes a statement of completion, a list of all third-party libraries used throughout the development process with details on other references and materials used to complete this assignment. This report also provides some important technical insights about the approach taken both in the design and execution process of the application and project setup.

Earring Holler – ITECH3108 Lab Assignment 2

Earring Holler is a single page web application where earring enthusiasts can post their opinions and thoughts about earrings and comment to other posts. The application is developed using a RESTful Backend API and a JavaScript front-end using jQuery and AJAX methods for Asynchronous calls.

# Back-End Design and Structure

Let’s begin, the back end was designed following a RESTful API with a project folder structure as follow (Dalisay, 2015):

**├─ api/  
├─── config/  
├────── core.php - project core configuration  
├────── database.php - database connection.  
├─── objects/  
├────── earringpost.php - database queries.   
├─── product/  
├────── create.php – create new posts.  
├────── delete.php – delete posts.  
├────── read.php – read posts (all).  
├────── read\_paging.php - output "earring posts" JSON data with pagination.  
├────── read\_one.php - accept posts ID to read one record from the database.  
├────── update.php - accept posts ID to update a database record.  
├────── search.php - accept keywords parameter to search for posts.**

**├────── comment\_paging.php – output "earring posts comments " JSON data with pagination.**

**├────── like\_unlike.php - accept a posts ID to increment the likes records.  
├────── post\_reply.php – allow members to comment on other posts  
├─── JSON/  
├────── earringpost.json – markup of complete data**

**├─── shared/  
├────── utilities.php - pagination array mechanism.**

The above configuration allows our front-end script to communicate with the website without refreshing the page.

## Front-End and Folder Structure

The front-end structure is similar in many ways to the back end.

**├─ app/  
├─── assets/  
├────── css/  
├───────── style.css  
├────── js/  
├───────── bootbox.min.js  
├───────── jquery.js**

**├───────── script.js  
├─── earringposts/  
├────── create-earringpost.js  
├────── delete-earringpost.js  
├────── read-one-earringpost.js  
├────── read-earringpost.js  
├────── update-earringpost.js**

**├────── earringpost.js  
├────── read-earringpost-comments.js**

**├────── like-unlike-earringpost.js  
├────── earringpost-comments.js**

**├────── search-earringpost.js  
├─── app.js   
├─ index.html**

### The Web App.

Using AJAX prevent the application from re-loading the whole page every time a button is clicked which on the other hand provide a much better user experience when browsing the webpage while also making our web application faster. Ajax is a group of technologies which can include HTML, CSS, JavaScript and server-side scripting languages like PHP. In this project, we used jQuery, JSON and PHP. jQuery is used to assist with our AJAX Asynchronous calls on JSON data that will be handled by the REST API built using PHP.

#### External Libraries.

Libraries used in this project are the Bootstrap 4 library with access to quickly design and customise responsive mobile-first sites with Bootstrap. The library is one of the world’s most popular front-end open source toolkit, featuring Sass variables and mixins, responsive grid system, extensive prebuilt components, and powerful JavaScript plugins (4, n.d.). Another library used is the **Bootbox.js** library which is a small JavaScript library that allows you to swiftly create programmatic dialog boxes using Bootstrap modals, without having to worry about creating, managing, or removing any of the required DOM elements or JavaScript event handlers (Payne., 2011 - 2019 ).

##### Content Gathering and Research.

Research and Content on the subject was done via the internet and most information used in the web application template comes from the associated forum (Forum, 2007). The website provided valuable insight on the Earring Forum theme which lead to the design and layout of the content of Earring Holler.

### Statement of Contribution.

The development of this assignment has been completed by me. I made use of valuable online tutorials and examples and worked my way towards what I needed to achieve. The project was very challenging but fun once things started making sense while learning new tools and techniques. Other assistance received is from the subject’s tutor whom many times have gone out of his way to assist with coding issues and troubleshooting.

References

4, B. (n.d.). Retrieved from Bootstrap : https://getbootstrap.com/

Dalisay, M. (2015, June 2). *AJAX CRUD Tutorial Using JavaScript, JSON and PHP – Step by Step Guide!* . Retrieved from CODE NINJA! : https://www.codeofaninja.com/2015/06/php-crud-with-ajax-and-oop.html

Forum, C.-D. (2007). *City-Data*. Retrieved from https://www.city-data.com/forum/fashion-beauty/191115-ladies-guys-earrings-opinions.html

Payne., N. (2011 - 2019 ). *Examples*. Retrieved from bootbox: http://bootboxjs.com/examples.html