**Mini Project Four**

**Alain Edwards**

**(aedwar50)**

**12/1/13**

**Developers Nest**

**Site:** <http://lamp.cse.fau.edu/~aedwar50/MP4/index.php>

**Purpose:**

The purpose of this project was to reconfigure and existing site called “Robins Nest” by adding more functionality and graphical effects. The site is to be implemented using a combination of PHP, CSS, HTML, AJAX, & JavaScript. The site will also allow the user to preform user based task like, submitting code, editing user profile, and creating links between friends.

**The Design:**

You will notice that the design of the website is completely reworked and re-skinned. It no long has those old gray and white boxes and limited color schemes. We have renamed the site “Developers Nest”, because the purpose of the site is somewhat like that of [Stack Overflow](http://stackoverflow.com/) website, which allows people to post code and receive help on said code.

This project was created by:

* + - Name: **Alain Edwards**

Role: All Functionality and Structural Design

**Functionality**

Almost every single feature from the original “Robin Nest” package and because of this, I will list the changes we have made below:

**Registration Page:**

* **New Fields have been added**
  + These fields include a first name, last name, email, retype password, website, & introduction. All of these fields are also new MySQL fields in the database. Most of this information is taken from the user to help decorate their personal profile page. For an example [Alain’s Profile](http://lamp.cse.fau.edu/~aedwar50/MP4/users/profile.php?user=admin).
* **You can upload your avatar from the registration page**
  + This function allows the user to upload there avatar from the start instead of have to register, login, and then edit your profile. Unfortunately, FAU lamp servers prohibit students from uploading files to the server through a webpage. This means the feature is not fully operational on the lamp servers but does have the source code available (behind the scenes) for localhost/offsite computing.
  + A solution to this problem was to provide newly registered users with a default avatar. All new users will be given the same default avatar, which can be found in images/avatars/user.png.
  + This function also insures that a user is only uploading an image of format jpeg, jpg, png, and gif.
* **There is a new graphical representation of validating the email and password fields.**
  + By modifying the previously provided username availability checker, we were able to confirm valid emails and enforce rules for passwords.
  + The first password filed now checks the length of the entered password to confirm that it is between 5 and 30 characters long.
  + The second password field checks the first password filed against the second password field to insure they match.
* **Human Verification**
  + A user now has to go through a human verification process called reCaptcha. We utilized the google reCaptcha API to prevent bots from spamming the MySQL database and receiving unwanted queries.

* **IP Address Logging**
  + IP addresses will now be logged in the MySQL database every time a new user registers or when a user logs in. This is a security measure in place for verification processes and other related security measures.
* **MD5 Encryption**
  + All passwords are now MD5 encrypted, to insure an extra layer of security.

**Functions.php**

* **Added two new methods, addfriend() & follow() (Better Way)**
  + These two functions are in charge of creating the friends and followers links between users.
  + This was implemented before in “Robins Nest” but it was implemented very poorly. By implementing it using JavaScript and the functions.php file, we were able to create a separation of concern and reduce the chances of error or duplicate queries to the server.

**Profile Page**

* **The New Profile page**
  + The new profile page displays your full name, handle (username), status, date registered, last time you were online, your website, introduction, image, friends, and all the post you have made.
  + Posts, is really cool, it allows all the users to see how you contribute to the community and which projects you may be working on.

**Edit Profile Page**

* **Inputs populate the data on the server automatically**
  + One thing we didn’t like about “Robins Nest” is how you edit your profile. With the original “Robins Nest” you had no idea what you entered before for things, like your intro, and it was too limited. With the new user editor page, the test fields, such as the first name, last name, intro, etc., will populate with what is currently on your profile page. Making it easier to update and change information.
* **If empty leave unchanged**
  + Another problem with the original “Robins Nest” is once you change one thing you had to change the others as well, in other words you can’t just change one thing (like name), and one thing alone, and expect all the previous data to be persistent for the other text fields; but now you can.

This also includes your avatar image.

**Submit Code page**

* **New Submit Code page**
  + Allows users and only users to submit source code to the site.

**MySQL Database**

* **New table designs**
  + We have added new fields to the members and friends tables. These new fields accommodate the new registered form as well as hold information like date registered, IP address of the user, and avatar image location.
  + We noticed that some of the tables from “Robins Nest” were improperly created by not providing a primary key field. We went ahead and fixed this problem.
  + Some of the original fields have also been changed from varchar to text or longtext to accommodate long strings.
* **Members Table**
  + In addition to logging user information every time a user registers or logins, we also created a field called status for each user that tells us when a user is currently online or offline. This is later used when displaying the currently online users in the developers section.

**Code Post Table**

* **New Table code\_post.sql**
  + This table is in charge of managing all the source code submissions along with all their respective fields: like developer (author), programming language, poster, code of the day (COD), title, etc.

**Overall**

* **The GUI was updated using Bootstrap.**
* **There is a “code of the day” feature implemented on the code page.**
* **There is a date sorting feature on the code page (i.e. recent & old).**
* **Home page always displays the 5 most recent code submissions, login or no login.**
* **Developers page displays all the developers, as well as who’s currently online.**
* **Almost everything has been rewritten for the better.**

**Time:**

The most time consuming part of building this site was PHP, JavaScript, and MySQL manipulation. When designing the JavaScript and MySQL side of the website, we had to make sure that we wasn’t over complicating the site and making sure we did everything in the most efficient way. Like the old famous say “there’s more than one way to skin a cat”, there is more than one way to code a site. A lot of the time was also spent insuring an error free site without lacking in functionality. For an example, a lot of the functionality on the site is based on whether or not a user is logged in or not. Due to this fact, I had to create a way to query the SQL server based on the user.

**Tools:**

The tools we used to complete this project were Text Wrangler, Adobe Dreamweaver CS5, Gimp, Photoshop, FileZilla and an assortment of browsers. Text Wrangler and Dreamweaver were used only as an IME. We did not use the drag and drop part of Dream weaver, the only reason we like to use Dreamweaver is because of the auto text filler. In other words it’s easier and faster to type in Dreamweaver; also you don’t have to search for things you forgot it’s all there. Gimp and Photoshop were used to create the top banner and some other logos. FileZilla was used to upload all the files to the FAU server through SSH. In the end every single tool had its unique capabilities and was all essential to the completion of this project.

**Sitemap:**

**Site Map:**

* Home: index.php
* Code
* Developers
* Member
  + Login
  + Register
  + Logout
  + Edit Profile
  + Submit Code
  + My profile

**Browser Compatibility/Testing:**

**Google Chrome: (Fully Operational)**

**Firefox: (Fully Operational)**

**Safari: (Fully Operational)**

**Internet Explorer 10: (Fully Operational)**

**Others (Not Tested)**

After good amount of testing we can conclude that the site is fully operation on these to browsers. This is not to say that there are no problems with Chrome, IE, Safari and Firefox; but simply to say it works.

**Mobile Website: (N/A)**

Unfortunately, we were not able to put together a mobile compatible site before the dead line. Seeing that this was not in mini projects one criteria, our attempt to get extra credit was cut short.

**Cookies (Not Used)**

We have chosen not to use cookies throughout this project. The reason for this is cookies can be easily manipulated from the clients side. What this means is, users could pose as other users or have access to something they shouldn’t have.

**MySQL User Database**

**Registration (Fully Operational)**

How to Register:

1. Click the members’ button on the top navigation’s bar.
2. Click Register
3. Fill out all fields
4. You’re Done! You can now login.

**Login (Fully Operational)**

We have created a grader login username and password on the MySQL database for you (the professor or grader) as well as Alain’s login information.

Alain’s Login Information:

First Name: Alain

Last Name: Edwards

Email: hmack1141@gmail.com

IP: 127.0.0.1

Date: 11/30/13

Username: admin

Password: monkey

Grader Login Information:

First Name: Grader

Last Name: Grades

Email: aplus@gmail.com

IP: 127.0.0.1

Date: 12/01/13

Username: Grader

Password: monkey

Instructions - How to Login:

1. Click the members’ button on the top navigation’s bar.
2. Click Login.
3. Username: admin
4. Password: monkey
5. Click the Login button
6. You’re Done!

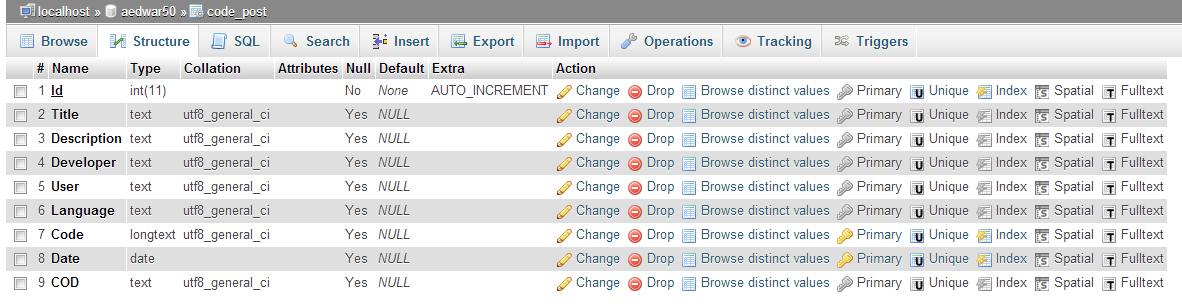
If you login you will noticed that it starts a new session for that user. It also un-hides the hidden piece of source code in the drop down menu. This piece of source code creates a logout drop down option, which allows the logged in user to logout successfully and end the session.

**SQL Injection Prevention (Fully Operational)**

We have taken proper information sanitization steps to insure a SQL injection free zone. By checking for GET method variables, by using the isset() function, we tried ensure an error free environment; the same goes for JavaScript. We have minimalized the event of an SQL injection-taking place. This is not to say, that an injection cannot happen but this will minimize the risk, because one of the key components of MySQL injection is to cause a PHP error.

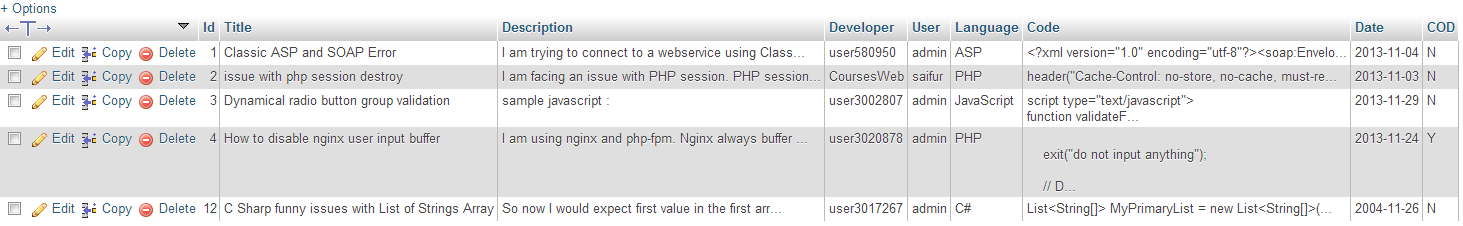
**I ALAIN EDWRDS DID ALL THE WORK FOR THIS PROJECT**

**Code\_post Architecture:**



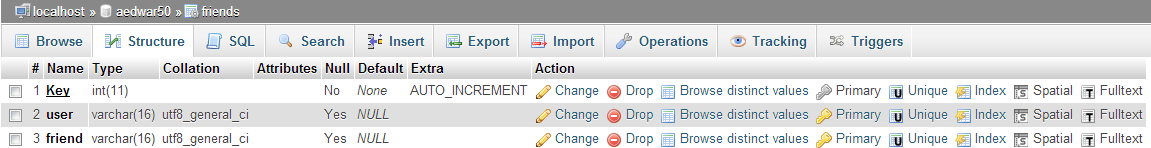
**(Full image in same directory – code\_post database structure.JPG)**

**Code\_post Database:**



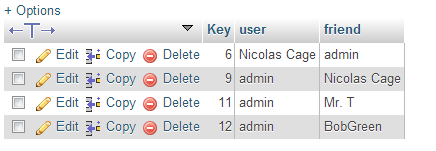
**(Full image in same directory – code\_post database.JPG)**

**Friends Architecture:**



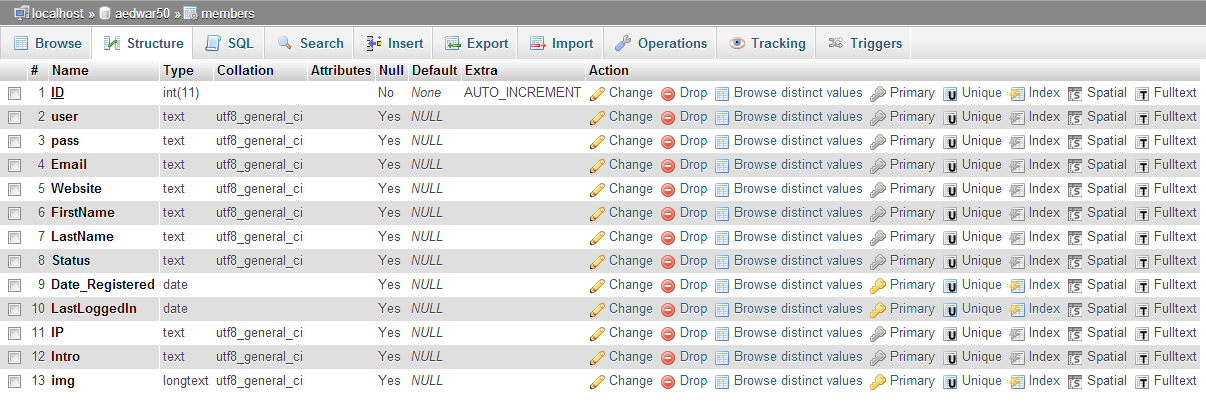
**(Full image in same directory – friends database structure.JPG)**

**Friends Database:**



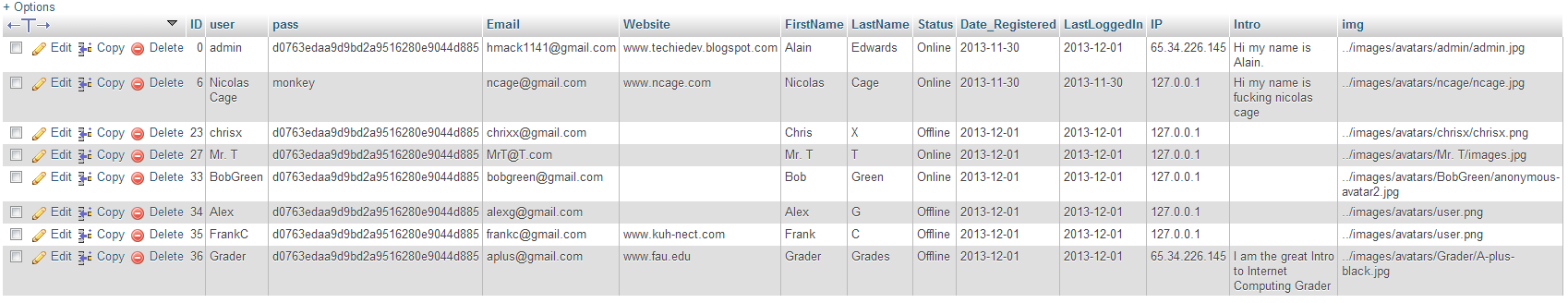
**(Full image in same directory – friends database.JPG)**

**Members Architecture:**



**(Full image in same directory – members database structure.JPG)**

**Members Database:**



**(Full image in same directory – members database.JPG)**

Any database not mentioned above was not edited or used in the reconfiguration of this site.

**Side Not:** The lamp server is a little glitchy with Apache and MySQL, this may have something to do with their current PHP version and or server setup. We have spoken to the professor about some of the issues and he is fully aware of these glitches.

**Conclusion:**

In conclusion this project was very rewarding, yet very long. We found that there is a lot of work that goes into the base structure of user based sites. This opportunity we were given has help us understand some of the inner workings of sites like Facebook and Twitter.