

# The Open Forum: Minimalistic Approach

by

Michael Moriarty

Submitted to the Board of Math and Computer Science  
School of Natural and Social Science  
in partial fulfillment of the requirements  
for the degree of Bachelor of Arts

Purchase College  
State University of New York

Dec 2018

Sponsor: Irina Shablinsky

Second Reader: Lee Tusman

Table of Contents

<b>Abstract</b>	<b>2</b>
<b>Introduction</b>	<b>3</b>
<b>Web Design</b>	<b>4</b>
<b>Local Network</b>	<b>5</b>
<b>Software</b>	<b>5</b>
<b>Apache HTTP Server Project</b>	<b>5</b>
Atom	6
Browsers	6
MariaDB	6
XAMPP	7
<b>Implementation</b>	<b>7</b>
<b>Issues/ troubleshooting</b>	<b>9</b>
<b>Bibliography</b>	<b>10</b>
<b>The Code</b>	<b>11</b>

## Abstract

In today's social media everyone is looking to see "who" you know and "what" you have done. In doing so we ultimately prejudge others and determine those we favor and dislike before we even exchange words. The goal of this project is to create a place(s) for individuals to socialize, network, and connect within a localized network of individuals without the preconceived notions of the other individuals before starting a conversation. Rather than taking the path of using the numerous of tools which help you create such web spaces, utilizing templates and pre-designed themes the project will be a learning experience into what it takes to create and design one myself.

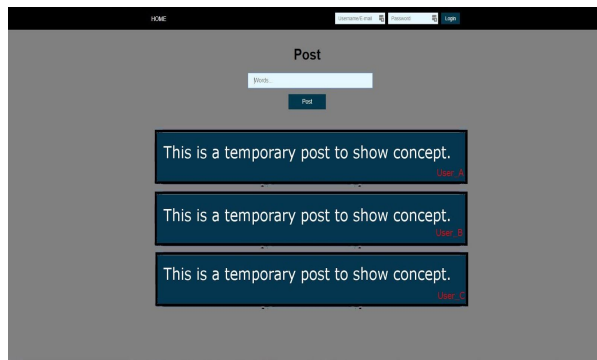
## Introduction

The focus of the Open Forum is to create a space where students may socialize and be expressive without being limited by their real-world person. Individuals will be required to sign up and log in before participating in the forum. The idea of utilizing profiles will be limited to just as a means to manage poster and commenter identifiers. Once logged into the page the individuals will be free to post and comment on posts. Currently, the student body of Purchase College uses A group page on Facebook to relay community events as well as ask and answer questions which individuals may have. This page as useful as it may be, also allow individuals to look to the poster's and commentator's info that they have added to their profiles and others have submitted about them through photos and tags.

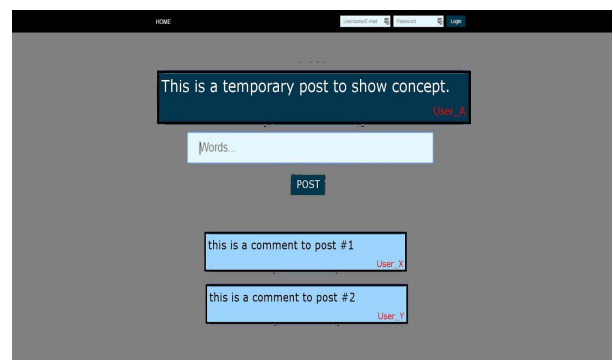
# Web Design

Initial designs of the pages were created with the idea of a simple forum look. It was always the intention to restrict user base to those in a small “local network”. To do so signing up would require an email which would be checked to limit profiles outside of the Purchase College community. Profiles are to remain vague and reinforce the idea of anonymity rather than who is who. The overall design is to be a simple structured forum where a single post will have a single branching thread where comments will be displayed and can be made.

## Concept MainForum Page



## Concept Topic Thread Page



## Local Network

The local network during the process of the project would be purposely start small and increase with time. Starting at a small group would help in a couple of ways. First, with the hopes to host the webpage files and their data on a personal system, it would make managing traffic and strain on server possible if needed. The other benefit with starting small would mean in early stages bugs and feedback would be more meaningful in development. Moving up to a class-sized user group on the said server was the next step to take.

## Software

### Apache HTTP Server Project

Apache is an open source web server software which is maintained and initially developed by a group of individuals from around the world to create the Apache group which later became the Apache Software Foundation. Initial work on the project was started in 1995 after the head of the national center for supercomputing appliances (NCSA), Rob McCool left. This left the development of Http daemon, the most popular web server software of the time, at a standstill. Development goals were to create a

replacement as well as a better piece of software that would be accessible to everyone. <sup>( 2 ) ( 6 )</sup>

## Atom

Atom is an open source text editor allow ease of cross-platform use. Like many other editors, Atom has many convenient features such as autocompletion, auto-formatting, and diagnostic warnings and errors. Atom started out as a project called Electron by MIT. and has grown to a versatile text editor which can be customized visually as well as supports add-ons which allow for local. <sup>( 1 )</sup>

## Browsers

Chrome, Mozilla Firefox, and Safari is the preferred browser at the time to use. Ideally, the pages would be made in such a way where format/design would allow for responsive design and multiple device support.

## MariaDB

MariaDB is a popular open source database server, whose development forks from MySQL. When Oracle acquired MySQL in 2009 the creators, as well as a number of core developers of MySQL, choose to start a new company by the name of Monty Program Ab. The new project would be called MariaDB and was intended to be open source and available to everyone. Overtime between the community and the company

the software has grown and surpassed MySQL even with efforts to keep them as interchangeable as possible. ( 6 ) ( 5 ) ( 7 )

## XAMPP

XAMPP is an open source web server solution stack that is available across multiple platforms created by a group named, Apache friends. XAMPP is a collection of components usable across multiple platforms whose names make up the acronym, Apache HTTP server, MariaDB databases, PHP, and Perl. ( 6 )

## Implementation

At the start of the project, most of the knowledge of web design and web pages were limited to fundamental coding in HTML and CSS. Preliminary research revealed some of the tools and languages that would be necessary to create and host the web pages envisioned. To do so would take a host server from which the pages would be accessed at. With user accounts and management of the posts and comments would best be done with a database. With this information a decision to use PHP as the primary language as well as using XAMPP because it contained many of the necessary packages that would be needed to start. Using XAMPP the packages for Apache and MariaDB are unpacked and settings for each as well as admin login info is set and can be changed the config and setup files. Server setup required the changing login from the default as well as setting an IP address for which the pages would be located at. In



the database, tables needed to be created and set up how they are structured and the type of info rows and columns contain. Development of the web pages started with setting up the connection to databases. Next, came the index page which would become the main page. During the creation of the index and the various page elements, physical looks and positioning would be managed in the CSS file. During the time it took to structure and build the signup and login pages it became easier to host the files on the school's student server for a place to have access to the pages at all times. This transition was simple even though the school's server used slightly different software for databases management which was phpMyAdmin. phpMyAdmin is a web-based management software and allows all the same functions as MariaDB did locally. Once signup and login were both fully working the next set of pages that would be necessary would be the main forum page where users could post a topic and also see what topics other users posted. This is where the progress of the project so far is at. Currently, the forum page allows logged in users to make submissions for topics, however, the display for the most recent topics is not fully implemented. Future work on The Open Forum will include finishing the forum page to get topics to correctly display as well as creating the page where comments can be made on each topic. Once the main elements of all pages are working work on appearance as well as building a more responsive design to the page to work better on or off mobile devices.

## Issues/ troubleshooting

During the process, there were a number of bugs and issues that arose where I needed to either fix parts of code or restructure the files to work properly. The first major issue that needed to be addressed was figuring out how to get information from the Post/Topic form to be correctly entered into the database. Originally text that was submitted in the forum was being sent correctly to the database, however, the user who was submitting it was not. The user ID that I was using was stored inside SESSION variables and at the time were not working correctly. With some work and review of the code, it was discovered that the issue was due to the fact that in the Post page's code session variable were not accessible because a session was not active. The next issue that followed came up when the second group of individuals tried to access the web page and were given a 404 error stating that the pages were unable to be found. It was unclear and still is unclear what was causing the issue as it was not present with the original handful of people who tried to connect. The issue, in the end, seemed to be resolved on its own and has not been present since the initial increase in the size of the group.

## Bibliography

- <sup>1</sup>“A Hackable Text Editor for the 21st Century.” *Atom*, atom.io/.
- <sup>2</sup>Group, Documentation. “About the Apache HTTP Server Project.” *About the Apache HTTP Server Project - The Apache HTTP Server Project*, [http://httpd.apache.org/ABOUT\\_APACHE.html](http://httpd.apache.org/ABOUT_APACHE.html).
- <sup>3</sup>*Browser Statistics*, [www.w3schools.com/php/default.asp](http://www.w3schools.com/php/default.asp). Use of this source was to find help with specific code usage and implementation.
- <sup>4</sup>*Browser Statistics*, [www.w3schools.com/sql/default.asp](http://www.w3schools.com/sql/default.asp). Use of this source was to find help with specific code usage and implementation.
- <sup>5</sup>“About MariaDB.” *MariaDB.org*, [mariadb.org/about/](http://mariadb.org/about/).
- <sup>6</sup>“Apache Friends.” *Apache Friends RSS*, [www.apachefriends.org/about.html](http://www.apachefriends.org/about.html).
- <sup>7</sup>Pearce, Rohan. “Dead Database Walking: MySQL's Creator on Why the Future Belongs to MariaDB.” *Computerworld*, 28 Mar. 2013, [www.computerworld.com.au/article/457551/dead\\_database\\_walking\\_mysql\\_creator\\_why\\_future\\_belongs\\_mariadb/](http://www.computerworld.com.au/article/457551/dead_database_walking_mysql_creator_why_future_belongs_mariadb/).
- <sup>8</sup>“Where Developers Learn, Share, & Build Careers.” *Stack Overflow*, [stackoverflow.com/](http://stackoverflow.com/). Use of this source was to find help with specific code usage and implementation.

## The Code

```
//Index
<?php
include_once 'header.php';
?>
<section class="main-container">
  <div class="main-wrap">
    <h2> HOME</h2>
    <?php
      if (isset($_SESSION['u_id'])) {
        header("Location: forum.php");

      } else {
        header("Location: signup.php");

      }

    ?>
  </div>
</section>
<?php
include_once 'footer.php';
?>

<!--
https://code.tutsplus.com/tutorials/how-to-create-a-phpmysql-powered-forum-from-scratch--net-10188
https://www.w3schools.com/php/default.asp
https://www.w3schools.com/sql/default.asp/
-->
-----
-
//Header

<?php
session_start();
?>
<!DOCTYPE html>
<html>
  <head>
```

```

</title></title>
<link rel="stylesheet" type="text/css" href="style.css">
</head>
<body>

<header>
<nav>
<div class="main-wrap">
<ul>
<li><a href="index.php">HOME</a></li>
</ul>
<div class="nav-login">
<?php
if (isset($_SESSION['u_id'])) {
echo ' <form action="includes/logout.inc.php" method="POST">
<button type="submit" name="submit">Logout</button>
</form>';
} else {
echo ' <form action = "includes/login.inc.php" method = "POST">
<input type="text" name="uid" placeholder="Username/E-mail">
<input type="password" name="pass" placeholder="Password">
<button type="submit" name="submit">Login</button>
</form>';
}
?>
</div>
</div>
</nav>
</header>
-----
-
//footer

<footer>
<p>
<small>
m@m<br>
For all issues and comments please submit on the GitHub page or by email:<br>
<a href= "https://github.com/m-moriarty/Web-Forum/issues"target="_blank">GitHub</a><br>
</small>
</p>
</footer>
</body>

```

```
</html>
```

---

```
-
```

```
//Signup Form
```

```
<?php
include_once 'header.php';
?>
<section class="main-container">
  <div class="main-wrap">
    <h2> Signup</h2>
    <form class="signup-form" action="includes/signup.inc.php"method="POST">
      <input type="text" name="first" placeholder="First Name">
      <input type="text" name="last" placeholder="Last Name">
      <input type="text" name="email" placeholder="E-mail">
      <input type="text" name="uid" placeholder="Username">
      <input type="text" name="pass" placeholder="Password">
      <button type="submit" name="submit">Sign Up</button>
    </form>
  </div>
</section>
<?php
include_once 'footer.php';
?>
```

---

```
-
```

```
//Forum Main Page with Post Form
```

```
<?php
session_start();
include_once 'header.php';

?>
<section class="main-container">
  <div class="main-wrap">
    <h2>Open Forum</h2>
    <form class="Post-form" action="includes/post.inc.php"method="POST">
      <input type="text" name="context" placeholder="Words...">
      <button type="submit" name="submit">Post</button>
    </form>
    <?php
    //display 30 most recent posts.
    include 'includes/dbm.inc.php';
    if(mysqli_connect_errno()){echo "Failed to connect to database";}
```

```

$sql="SELECT p_id, p_user, p_context FROM posts ORDER BY p_id DESC LIMIT 30";
$result = mysqli_query($conn, $sql);
$rows = mysqli_fetch_assoc($result);

foreach($rows as $row){
    echo $row[p_context];
    // echo '<div class="topic">';
    // echo '<a href = "topic.php">' . $row['p_context'] . '</a>';
    // echo '</div>';
}
// $rows = mysqli_num_rows($result);
// if($rows < 1){
//     echo 'sorry no dice. something went wrong';
// }
// else{
//     echo 'array initiated: posts will be coming soon.';
// }
?>
</div>
</section>
<?php
    include_once 'footer.php';
?>
-----
-
//Database connection

<?php
$db_Host = "localhost";
$db_User = "***** "; // Login info replaced
$db_Password = "***** "; // login info replaced
$db_Name = "mmoriarty_db";

$conn = mysqli_connect($db_Host, $db_User, $db_Password, $db_Name);
-----
-
//Login Post Processing

<?php

session_start();

if (isset($_POST['submit'])) {

```

```

include 'dbm.inc.php';

$uid = mysqli_real_escape_string($conn, $_POST['uid']);
$pass = mysqli_real_escape_string($conn, $_POST['pass']);

//Errorrrrs
//empty input fileds
if (empty($uid) || empty($pass)) {
    header("Location: ../index.php?login=empty");
    exit();
} else {
    $sql="SELECT * FROM users WHERE user_uid='$uid' OR user_email ='$uid'";
    $result = mysqli_query($conn, $sql);
    $check = mysqli_num_rows($result);
    if ($check < 1) {
        header("Location: ../index.php?login=error1");
        exit();
    } else {
        if ($row = mysqli_fetch_assoc($result)) {
            //decrypt password
            $encryptPass = password_verify($pass, $row['user_pass']);
            if ($encryptPass == false) {
                header("Location: ../index.php?login=error2");
                exit();
            } elseif ($encryptPass == true) {
                //user info logged include
                $_SESSION['u_id'] = $row['user_id'];
                $_SESSION['u_first'] = $row['user_first'];
                $_SESSION['u_last'] = $row['user_last'];
                $_SESSION['u_email'] = $row['user_email'];
                $_SESSION['u_uid'] = $row['user_uid'];
                header("Location: ../index.php?login=sucess");
                exit();
            }
        }
    }
} else {
    header("Location: ../index.php?login=error");
    exit();
}

```



---

-

//Signup Post Processing

```

<?php
if (isset($_POST['submit'])) {
    include_once'dbm.inc.php';

    $first = mysqli_real_escape_string($conn, $_POST['first']);
    $last = mysqli_real_escape_string($conn, $_POST['last']);
    $email = mysqli_real_escape_string($conn, $_POST['email']);
    $uid = mysqli_real_escape_string($conn, $_POST['uid']);
    $pass = mysqli_real_escape_string($conn, $_POST['pass']);

    //ERRORRRS
    //Check empty field
    if (empty($first) || empty($last) || empty($email) || empty($uid) || empty($pass)) {
        header("Location: ../signup.php?signup=empty");
        exit();
    }else {
        //for valid characters
        if (!preg_match("/^[a-zA-Z]*$/", $first) || !preg_match("/^[a-zA-Z]*$/", $last)) {
            header("Location: ../signup.php?signup=invalid");
            exit();
        }else{
            //check email format
            if (!filter_var($email, FILTER_VALIDATE_EMAIL)) {
                header("Location: ../signup.php?signup=email");
                exit();
            }
            //check if username already exists
            } else {
                $sql = "SELECT * FROM users WHERE user_uid='$uid'";
                $result = mysqli_query($conn, $sql);
                $Check = mysqli_num_rows($result);

                if ($Check > 0) {
                    header("Location: ../signup.php?signup=taken");
                    //find way to add subtext to/after "form" stating email already used
                    //add check or add system var that checks if email has already been verified (toggle on and
                    off if an account is closed/terminated)
                    exit();
                }else {
                    // Pass encrypt

```

```

    $encryptPass = password_hash($pass, PASSWORD_DEFAULT);
    //adding users to Database
    $sql = "INSERT INTO users (user_first, user_last, user_email, user_uid, user_pass)
        VALUES('$first','$last','$email','$uid','$encryptPass')";
    mysqli_query($conn, $sql);
    header("Location: ../signup.php?signup=sucess");
    exit();
}
}
}
}
} else {
    header("Location: ../signup.php");
    exit();
}

```

---

-

//Logout Post Processing

```

<?php
//check if logout was submitted
if (isset($_POST['submit'])) {
//makes sure session is active then kills and know atributs associated with session
    session_start();
    session_unset();
    session_destroy();
    header("Location: ../index.php");
    exit();
}
?>

```

---

-

//Topic Submission Post Processing

```

<?php
session_start();
if (isset($_POST['submit'])) {
    include_once'dbm.inc.php';

    $context = mysqli_real_escape_string($conn, $_POST['context']);
    $date    = date("d/m/y h:i:s");

```

```

$user = $_SESSION['u_uid'];

//errrrrrrrs
//check for empty dialogue
if (empty($context)) {
    header("Location: ../forum.php?form=empty");
    exit();
// when dialog box contains characters
}else {
    $sql = "INSERT INTO posts (p_context, p_date, p_user)
    VALUES('$context','$date','$user')";
    mysqli_query($conn, $sql);
    header("Location: ../post.php?post=sucess");
    exit();
}

} else {
    header("Location: ../post.php");
    exit();
}

```

---

-

//CSS Style Sheet

/\*

html5doctor.com Reset Stylesheet

v1.6.1

Last Updated: 2010-09-17

Author: Richard Clark - <http://richclarkdesign.com>

Twitter: @rich\_clark

\*/

html, body, div, span, object, iframe,  
h1, h2, h3, h4, h5, h6, p, blockquote, pre,  
abbr, address, cite, code,  
del, dfn, em, img, ins, kbd, q, samp,  
small, strong, sub, sup, var,  
b, i,  
dl, dt, dd, ol, ul, li,  
fieldset, form, label, legend,  
table, caption, tbody, tfoot, thead, tr, th, td,  
article, aside, canvas, details, figcaption, figure,  
footer, header, hgroup, menu, nav, section, summary,

```
time, mark, audio, video {
    margin:0;
    padding:0;
    border:0;
    outline:0;
    font-size:100%;
    vertical-align:baseline;
    background:transparent;
}

body {
    line-height:1;
}

article,aside,details,figcaption,figure,
footer,header,hgroup,menu,nav,section {
    display:block;
}

nav ul {
    list-style:none;
}

blockquote, q {
    quotes:none;
}

blockquote:before, blockquote:after,
q:before, q:after {
    content:"";
    content:none;
}

a {
    margin:0;
    padding:0;
    font-size:100%;
    vertical-align:baseline;
    background:transparent;
}

/* change colours to suit your needs */
ins {
```

```
background-color:#ff9;
color:#000;
text-decoration:none;
}

/* change colours to suit your needs */
mark {
background-color:#ff9;
color:#000;
font-style:italic;
font-weight:bold;
}

del {
text-decoration: line-through;
}

abbr[title], dfn[title] {
border-bottom:1px dotted;
cursor:help;
}

table {
border-collapse:collapse;
border-spacing:0;
}

/* change border colour to suit your needs */
hr {
display:block;
height:1px;
border:0;
border-top:1px solid #cccccc;
margin:1em 0;
padding:0;
}

input, select {
vertical-align:middle;
}
```

```
/*MY SYTLE */
/*-----*/
.main-wrap{
  margin: auto;
  width: 1000px;
  background-color: gray;
}

a {
  text-decoration: none;
}

body {
  background-color: gray;
  padding-bottom: 60px;
}

header nav {
  width: 100%;
  height: 60px;
  background-color: Black;
}

header nav ul{
  float: left;
}

header nav ul li{
  float: left;
  list-style: none;
}

header nav ul li a{
  background-color: Black;
  font-family: sans-serif;
  font-size: 16px;
  color: white;
  line-height: 63px;
}

header .nav-login {
  float: right;
}
```

```
header .nav-login form {  
  float: left;  
  padding-top: 15px;  
}
```

```
header .nav-login form input {  
  float: left;  
  width: 140px;  
  height: 30px;  
  padding: 0px 10px;  
  margin-right: 10px;  
  border: none;  
  background-color: #e6f8ff;  
  font-family: sans-serif;  
  font-size: 14px;  
  color: #111;  
  line-height: 30px;  
}
```

```
header .nav-login form button {  
  float: left;  
  width: 60px;  
  height: 30px;  
  margin-right: 10px;  
  border: none;  
  background-color: #00394d;  
  font-family: sans-serif;  
  font-size: 14px;  
  color: white;  
  cursor: pointer;  
}
```

```
header .nav-login form button:hover {  
  background-color: #0083b3;  
}
```

```
header .nav-login a {  
  display: block;  
  width: 60px;  
  height: 60px;  
  border: none;  
  float: left;
```

```
background-color: #fff;  
font-family: sans-serif;  
font-size: 16px;  
color: Black;  
line-height: 63px;  
cursor: pointer;  
}
```

```
footer{  
  position: absolute;  
    bottom: 0;  
  width: 100%;  
  height: 60px;  
  border: none;  
  background-color: #fff;  
  clear: left;  
  text-align: center;  
  
}
```

```
.main-container{  
  padding-top: 40px;  
}
```

```
.main-container h2 {  
  text-align: center;  
  color: Black;  
  font-family: sans-serif;  
  font-size: 40px;  
  line-height: 50px;  
}
```

```
/*style signup.php*/  
.signup-form{  
  width: 400px;  
  margin: 0 auto;  
  padding-top: 30px;  
}
```

```
.signup-form input {  
  width: 90%;  
  height: 40px;  
  padding: 0px 5%;
```



```
margin-bottom: 4px;
border: none;
background-color: #e6f8ff;
font-family: sans-serif;
font-size: 16px;
color: #111;
line-height: 40px;
}
```

```
.signup-form button {
display: block;
margin: 0 auto;
width: 30%;
height: 40px;
border: none;
background-color: #00394d;
font-family: sans-serif;
font-size: 16px;
color: #fff;
cursor: pointer;
}
```

```
.signup-form button:hover {
background-color: #0083b3;
color: white;
}
```

```
h1 {
text-align: center;
color: #fff;
font-weight: 300;
margin: 0 0 40px;
}
```

```
.Post-form{
width: 400px;
margin: 0 auto;
padding-top: 30px;
}
```

```
.Post-form input {
width: 90%;
height: 40px;
```

```
padding: 0px 5%;  
margin-bottom: 4px;  
border: none;  
background-color: #e6f8ff;  
font-family: sans-serif;  
font-size: 16px;  
color: #111;  
line-height: 40px;  
}
```

```
.Post-form button {  
display: block;  
margin: 0 auto;  
width: 30%;  
height: 40px;  
border: none;  
background-color: #00394d;  
font-family: sans-serif;  
font-size: 16px;  
color: #fff;  
cursor: pointer;  
}
```

```
.Post-form button:hover {  
background-color: #0083b3;  
color: white;  
}
```

---