

UNIVERSITY OF ALASKA ANCHORAGE

CSCE A470

CAPSTONE PROJECT

Operation Christmas Child

Website Front End

Author:

Cody McWilliams

Supervisor:

Prof. Kenrick Mock, PhD

Anchorage AK, April 2016



**Computer Science &
Engineering Department**
UNIVERSITY *of* ALASKA ANCHORAGE

© Copyright 2016

by

Cody McWilliams

crmcmwilliams@alaska.edu

Version 1.1

Abstract

The purpose of this project was to build a website for the local branch of the charitable organization known as Operation Christmas Child. The website was designed to ease the task of information, and, event management for volunteers, and, to make forms easier to fill out for donors. The website uses geolocation data so that certain forms may only be accessed when present at a donation drop off location, or, organized event.

The website front end had three major requirements, it must be easy to use, accessible through mobile devices, and, must only allow access to certain forms when the user is within a certain radial distance of a location allowed by the form. To make sure the first two requirements were met, the website was designed specifically to work on mobile devices using a combination of CSS, and, JavaScript. To fulfill the third requirement geolocation data is collected via JavaScript, and, compared against stored location data using a combination of PHP, MySQL, and JavaScript.

Acknowledgements

I would like to thank Dr. Adriano Cavalcanti, and, Dr. Kenrick Mock for supervising this project, and my partner David Bretz for giving me the opportunity to help him complete this project which he has been working on for semesters previous to my joining.

Contents

1 Introduction

- 1.1 Background.....1
- 1.2 Application.....2
- 1.3 Motivation.....4
- 1.4 Technology.....5

2 System/Methodology

- 2.1 System.....7
- 2.2 User Interface.....7
- 2.3 Client-Server-Database Communication.....10
- 2.4 Timeline.....12
- 2.5 Agile Workflow.....13

3 Design and Testing

- 3.1 Requirements.....14
- 3.2 Design.....15
- 3.3 Testing.....16
- 3.4 Browsers.....18
- 3.5 Agile Coding.....20

4 User Manual

- 4.1 Access.....21
- 4.2 Account.....21
- 4.3 Other.....22

5 Summary

- 5.1 Implications.....23
- 5.2 Advantages.....23
- 5.3 Conclusion.....24

Appendix A: Current Website Layout

Appendix B: Source Code

References

List of Figures

Fig 1.1: Operation Christmas Child Logo.....	2
Fig 1.2: Initial Website Layout.....	3
Fig 1.3: Web version of Donation Form.....	4
Fig 1.4: OCC Donation Drop-Off Form.....	5
Fig 1.5: HTML, JavaScript/JQuery, and, CSS working together in a single PHP file.....	6
Fig 2.1: Example of HTML showing head, title, link, and script tags.....	8
Fig 2.2: CSS inside HTML style tags.....	9
Fig 2.3: JavaScript and JQuery in HTML script tags.....	10
Fig 2.4: MySQL queries using PHP.....	11
Fig 2.5: Gantt Chart.....	13
Fig 3.1: Home Page.....	15
Fig 3.2: CSS Units.....	17
Fig 3.3: Relationship between viewport and Browser.....	18
Fig 3.4: Popular Browsers.....	19
Fig 3.5: Mobile Devices.....	19
Fig 5.1: Reduce Waste.....	24
Fig A.1: Website Layout Diagram.....	26

Chapter 1

Introduction

1.1 Background

This project was a collaboration with fellow student David Bretz, we were tasked with building a website for the local branch of the charitable organization, Operation Christmas Child(OCC).

The website is meant to make information management a simpler task for the volunteers and coordinators at OCC. Previously, whenever a donation was made, a paper form was filled out by the donor, and, a volunteer later input the information by hand. This created two major problems, first manual data entry is an incredibly time consuming task, and, second this task is difficult to accomplish when forms are filled out incorrectly or illegibly. Additionally, when information is input manually, and, by an intermediate party, it becomes a difficult task to retain the integrity of that information.



Fig 1.1 Operation Christmas Child Logo

To help alleviate a portion of this burden, our team set out to create a website that could be used to remove most of the manually data entry. The following sections detail the methods used to create this website, the difficulties faced, and, the results of our efforts.

1.2 Application

The website was broken into two parts, David created the data management system, while I created the user interface. The website was designed with three different types of users in mind, donors, volunteers, and coordinators.



Fig1.2 Initial Website Layout

The donors needed to be able to access the forms that needed to be filled out when making donations, however, we were asked to only make these forms available to the user when a donation center was located nearby. The volunteers needed to be able to manage donation events, and, manually input information in the case of internet failure. The coordinators needed to be able to create accounts for volunteers, create new events, and, directly communicate with the database at an administrative level.

The image shows a web interface for a 'Donor Drop-Off' form. At the top left is a logo for 'Operation Christmas Child' featuring a red and white airplane and the text 'Samaritan's Purse'. The top navigation bar is green and contains the text 'Event Check-In' and 'Short-Term Volunteers', with a 'Login' link in the top right corner. The main form area is dark grey and contains the following fields:


Donor Drop-Off	
First Name	Middle Name
Last Name	Suffix
Street/Apt.	City
State	Zip Code
Phone	Email
Church	Number of Boxes
<input type="submit" value="Submit"/>	

Fig 1.3 Web version of Donation Form

1.3 Motivation

Whenever a donation is made to OCC, the donor must fill out paperwork, later, a volunteer is tasked with manually inputting all of the information from the paperwork into a database. During the process of inputting all of the information contained in the paperwork, it is likely that the volunteer will come across a document that is filled out illegibly, or nearly. Because the volunteer does not know what every document is supposed to say, they may make errors in trying to interpret the content. While the website has additional features the primary motivation was to reduce the time taken up by manual data entry, and, to help maintain the integrity of the data going into their long-term records.

SAMARITAN'S PURSE® 243



Shoebox Drop-Off Log

LOCATION CODE:

DATE (MM/DD/YY):

NUMBER OF SHOEBOXES YOU ARE DROPPING OFF:

INDIVIDUAL NAME *OR* CHURCH/GROUP NAME

MY PERSONAL MAILING ADDRESS (AND PHONE) *OR* CHURCH/GROUP MAILING ADDRESS (AND PHONE)

CITY STATE ZIP

PHONE:

EMAIL ADDRESS:

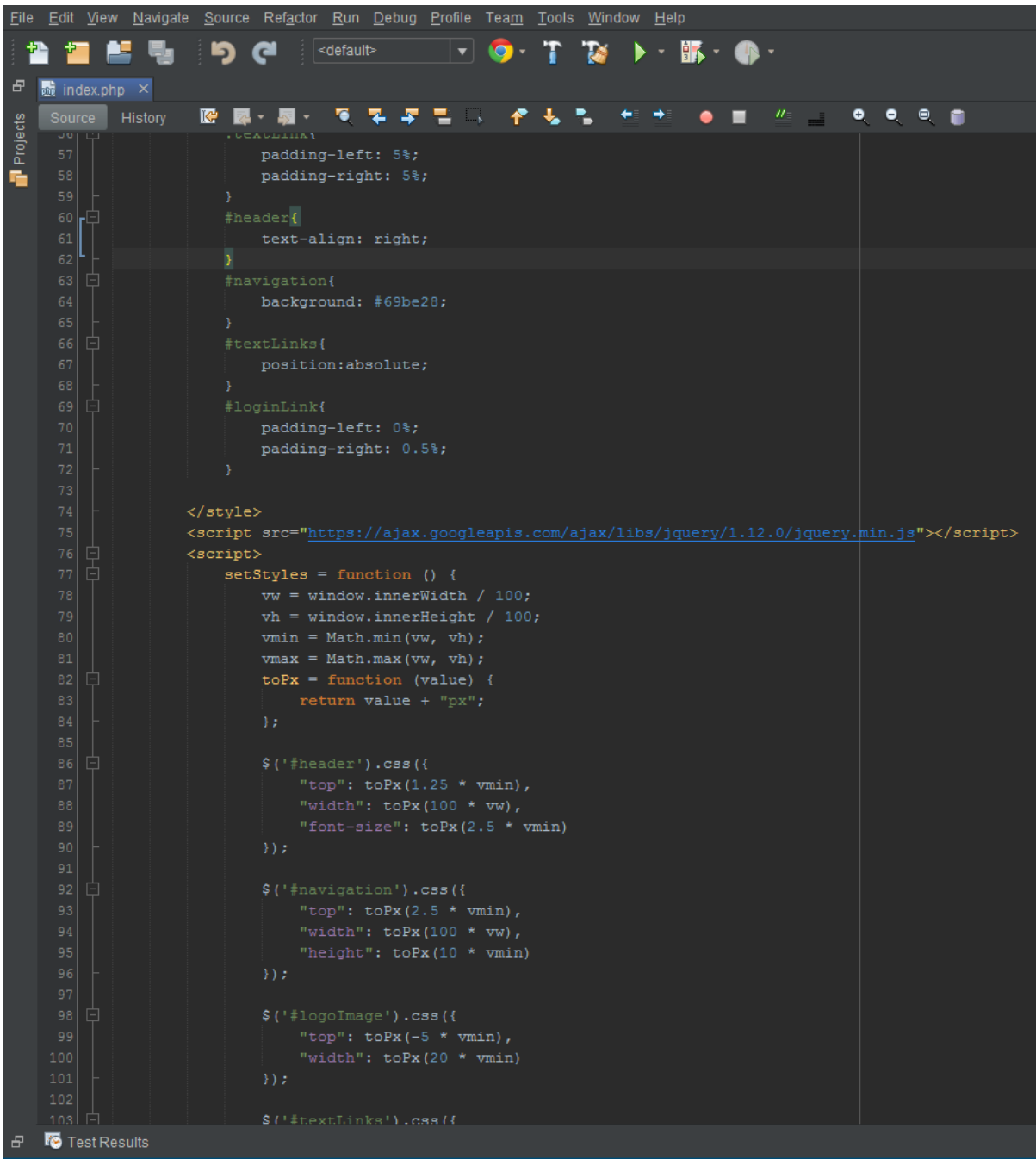
CONTACT PERSON (FOR CHURCH/GROUP):

CONTACT PERSON PHONE:

Fig 1.4 OCC Donation Drop-Off Form

1.4 Technology

The website was created using a combination of different programming and markup languages. To create the layout, and, change the appearance of the website, HTML, and, CSS were used. JavaScript, was used to retrieve geolocation data, perform calculations, and, to communicate with the server. The JavaScript library JQuery was used to change HTML, and, CSS values based on user input. Finally, PHP was used to generate repetitive HTML, and, to generate MySQL queries for use in sending and retrieving information from the database.



The screenshot shows an IDE window titled 'index.php' with a dark theme. The code is as follows:

```
57     padding-left: 5%;
58     padding-right: 5%;
59   }
60   #header{
61     text-align: right;
62   }
63   #navigation{
64     background: #69be28;
65   }
66   #textLinks{
67     position:absolute;
68   }
69   #loginLink{
70     padding-left: 0%;
71     padding-right: 0.5%;
72   }
73
74 </style>
75 <script src="https://ajax.googleapis.com/ajax/libs/jquery/1.12.0/jquery.min.js"></script>
76 <script>
77   setStyles = function () {
78     vw = window.innerWidth / 100;
79     vh = window.innerHeight / 100;
80     vmin = Math.min(vw, vh);
81     vmax = Math.max(vw, vh);
82     toPx = function (value) {
83       return value + "px";
84     };
85
86     $('#header').css({
87       "top": toPx(1.25 * vmin),
88       "width": toPx(100 * vw),
89       "font-size": toPx(2.5 * vmin)
90     });
91
92     $('#navigation').css({
93       "top": toPx(2.5 * vmin),
94       "width": toPx(100 * vw),
95       "height": toPx(10 * vmin)
96     });
97
98     $('#logoImage').css({
99       "top": toPx(-5 * vmin),
100      "width": toPx(20 * vmin)
101    });
102
103    $('#textLinks').css({
```

Fig 1.5 HTML, JavaScript/JQuery, and, CSS working together in a single PHP file

Chapter 2

System/Methodology

2.1 System

Our website can be broken into the different parts, the source code for the user interface, the client-server-database communication, and, the database. The following sections describe the technology used to create the user interface, and, to handle communication between the client, the server, and, the database, the database itself was created by David Bretz, and, will not be discussed in detail here.

2.2 User Interface

To begin, we will cover the technology used to create the user interface, which was created using HTML, CSS, and, JavaScript/JQuery. HTML, and, CSS are not programming languages, they are what is known as markup languages. Markup languages are used to describe aesthetic characteristics rather than perform calculations. HTML is used to model the physical layout of a page using keywords called tags.

```
<head>
  <title>Operation Christmas Child</title>
  <link rel="stylesheet" href="volunteer.css">
  <link rel="shortcut icon" href="SP_Icon.png" type="image/png">
  <script src="volunteerMethods.js"></script>
</head>
```

Fig 2.1 Example of HTML showing head, title, link, and script tags.

Some HTML tags such as `<script>`, and, `<link>` allow the programmer to include JavaScript, and, CSS from external files as shown in [Fig 6], although scripts and styles may be written between their respective tags as well, as in [Fig 7], and, [Fig 8].

```

<style>
  *{
    margin: 0px;
    padding: 0px;
    font-family: sans-serif;
    position:relative;
  }
  html{
    zoom:100%;
    width:fit-content;
    margin-left:auto;
    margin-right:auto;
  }
  a{
    text-decoration: none;
  }
  body{
    background-color: #333;
  }
  a , .caption{
    color:white;
  }
  .linkSet{
    text-align: center;
  }
  .textLink{
    padding-left: 5%;
    padding-right: 5%;
  }
  #header{
    text-align: right;
  }
  #navigation{
    background: #69be28;
  }
  #textLinks{
    position:absolute;
  }
  #loginLink{
    padding-left: 0%;
    padding-right: 0.5%;
  }
</style>

```

Fig 2.2 CSS inside HTML style tags

Styles, created using CSS, the style tag, or by adding a style attribute to a tag, allow the programmer to change the appearance of the webpages layout created using HTML. The style in [Fig 7] changes the size, color, positioning, and, alignment of different components in the layout. JavaScript also has a hand in the webpages aesthetics. The script shown in [Fig 8] is written using a JavaScript library called JQuery, this script inject styles into different HTML tags so that the website looks as it is supposed to across different web browsers, and, different sized screens. JQuery is specifically listed even though it is a JavaScript library because the syntax is quite different from vanilla JavaScript using a (\$) as a prefix to blocks of code. JQuery was not used

until very late in the development cycle, however, the addition of JQuery greatly reduced, the time needed to complete tasks, the complexity of the code, and, the amount of code overall.

```

<script>
  setStyles = function () {
    vw = window.innerWidth / 100;
    vh = window.innerHeight / 100;
    vmin = Math.min(vw, vh);
    vmax = Math.max(vw, vh);
    toPx = function (value) {
      return value + "px";
    };

    $('#header').css({
      "top": toPx(1.25 * vmin),
      "width": toPx(100 * vw),
      "font-size": toPx(2.5 * vmin)
    });

    $('#navigation').css({
      "top": toPx(2.5 * vmin),
      "width": toPx(100 * vw),
      "height": toPx(10 * vmin)
    });

    $('#logoImage').css({
      "top": toPx(-5 * vmin),
      "width": toPx(20 * vmin)
    });

    $('#textLinks').css({
      "left": toPx($('#logoImage').width()),
      "right": toPx(1),
      "bottom": toPx(3.25 * vmin),
      "font-size": toPx(2.5 * vmin)
    });

    $('#imageLinks').css({
      "top": toPx(10 * vmin),
      "width": toPx(100 * vw),
      "font-size": toPx(2.5 * vmin)
    });

    $('#dropoffImage, #eventImage, #volunteerImage').css({
      "width": toPx(25 * vmin),
      "height": toPx(25 * vmin)
    });

    $(document).ready(function () {
      setStyles();
      $(window).resize(function () {
        setStyles();
      });
    });
  };
</script>

```

Fig 2.3 JavaScript and JQuery in HTML script tags

2.3 Client-Server-Database Communication

Next we will cover the client-server-database communication, which was done using a combination of JavaScript, PHP, and MySQL. The database was created using MySQL, so to communicate with it MySQL queries must be sent by the website, this is accomplished using

PHP. PHP has built in methods that allow for MySQL queries to be passed in as strings, and, sent to that database, a response is then generated by the database, and, sent back to the website for processing.

```

<?php
function insertInto($table, $columns, $values) {
    $query = "INSERT INTO `{$table}` VALUES (";
    for ($i = 0; $i < count($columns); $i++) {
        if ($i == 0) {
            $query .= "`{$columns[$i]}`";
        } else {
            $query .= ",`{$columns[$i]}`";
        }
    }
    $query .= ") (";
    for ($i = 0; $i < count($values); $i++) {
        if ($i == 0) {
            $query .= "\"{$values[$i]}\"";
        } else {
            $query .= ",\"{$values[$i]}\"";
        }
    }
    $query .= ")";
    $result = mysql_query($query);
    print("Result of insert: $result"); // True if successful
    if (!$result) {
        die("Couldn't query database: " . mysql_error());
    }
}

function selectFrom($field, $table) {
    return "SELECT $field FROM `{$table}`";
}

function deleteFrom($table, $field, $value) {
    return "DELETE FROM $table WHERE $field='$value'";
}
?>

```

Fig 2.4 MySQL queries using PHP

One of the features of our website is forms that are only access able when the user is located near a donation drop-off location, or, and OCC event. To create this feature it is necessary to retrieve geolocation data from the user, this is done using JavaScript. Once the geolocation data is retrieved, there is one issue, the locations to compare against are stored in the database, and, the user's computer (the client) is unable to communicate directly with the database.

Using JQuery, the client requests geolocation data corresponding to OCC events, and, donor drop-off locations. Upon receiving the request the server queries the database for the geolocation data, and waits for a response. Once the query is processed by the database, the response is sent back to the client, where calculations can be performed to convert latitude and longitude into units of distances.

Other parts of the website also use a more condensed version of this communication system, logging into the administrator section, or submitting information to a webform requires querying the database, however, the data input by the client in these situations is handled directly by the server through an HTML form, so JavaScript is not necessary.

2.4 Timeline

I joined this project late in the development cycle, as such, I was not heavily involved in the planning and scheduling. The Gantt chart shown in [Fig. 10] was created by David Bretz, who has been working on this project from the start.

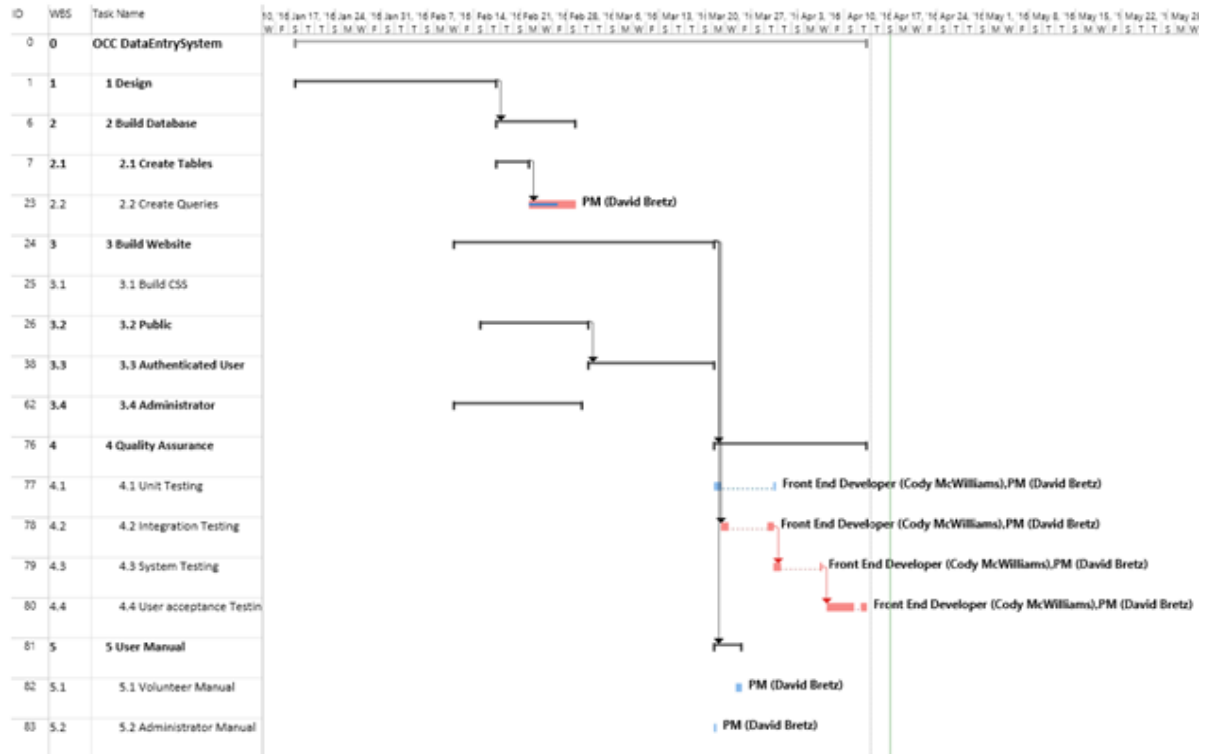


Fig 2.5 Gantt Chart

2.5 Agile Workflow

An Agile workflow is one which allows a team to work around chaotic schedules and unforeseen complications. This is accomplished by frequent group meetings, and, consistently completing small portions of the project called deliverables, over short periods of time called iterations. The Agile team meets frequently, so that if something is not going according to plan, the plan can be reevaluated. An agile workflow was the best option for our team because David, and, I had the ability to meet frequently in between classes, and, on weekends.

Chapter 3

Design and Testing

3.1 Requirements

- Mobile friendly
- Allow donors to input information
 - Number of shoe boxes donated
 - Name or Church/Organization
 - Contact Info
 - Under these conditions
 - It is a collection week
 - User is near drop-off location
- Volunteers must be able to
 - Review information
 - View statistics
 - Add project leaders
 - Create events, and, accounts
- User entries must be validated
- Donor information is not validated
- Login to access privileged information
- Restrict database access to system admins
- Passwords
 - 8+ characters
 - At least one letter, and, number
 - At least one capitol, and, one lowercase letter

3.2 Design

The website consists of a home page, and, four main sections, donations, events, volunteers and admin. The donations section allows donors to fill out the paperwork that goes along with their donation, this section is only accessible while located near a donation site, and, during a collection time. The events section is similar to the donations sections, it allows users to check into, and, check out of events that they are currently being held in their location. The volunteer section allows volunteers to view, and, sign up for events. The admin section is separated from the rest of the site by a login page.

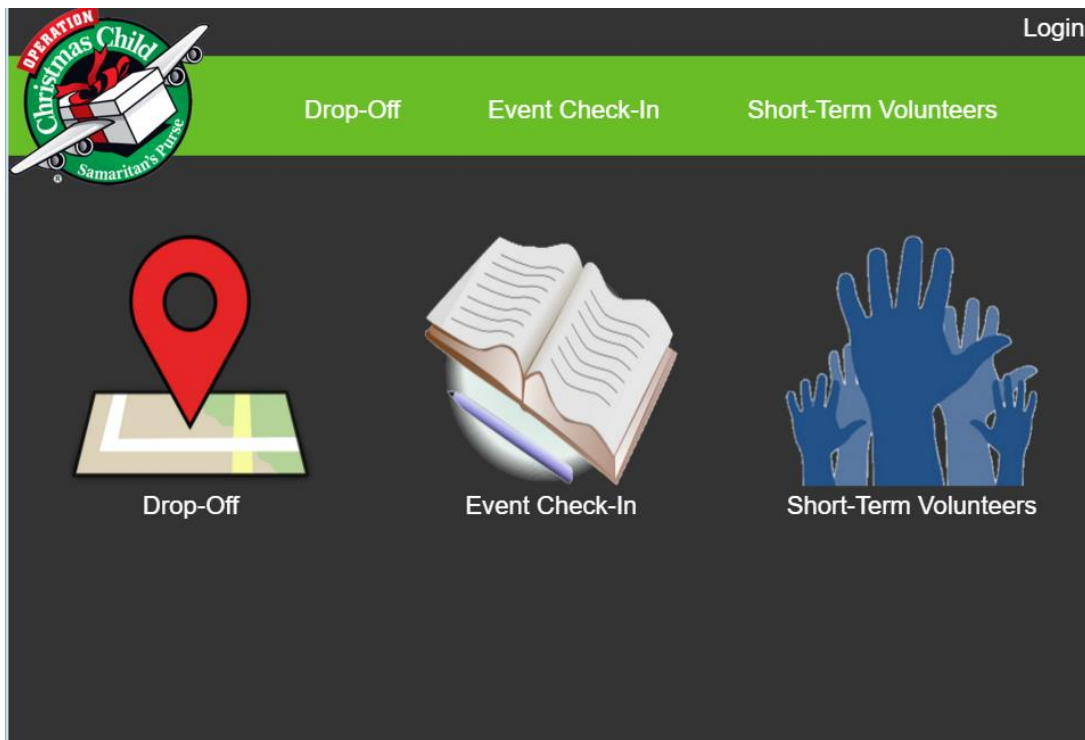


Fig 3.1 Home Page

Long term volunteers, and, system administrators can both login to the admin section, however, volunteers do not have the same access to the site that administrators have. Volunteers are able to manage events, review information, manually input form data, and, create accounts with the same level of access. Administrators are able to do anything that a logged in volunteer can do, in addition administrators have access to the database.

3.3 Testing

One of the most important requirements for our website is that it functions properly in all major web browsers, and, is mobile friendly. In a world where web browsers use the same standards, this would be quite easy, however, this is not the case. When designing a website, there are certain things we can do using CSS or HTML in one browser that will work differently, or will not work at all in another.

Name	Symbol	Available in		Relative to
		CSS2.1	CSS3	
Pixel	px	Yes	Yes	Display resolution
				Note: There is no fixed ratio of device pixels vs. CSS pixels in high-density displays – browsers provide <code>window.devicePixelRatio</code> so you can determine how many device pixels fit within a single CSS pixel at 1:1 zoom.
Percentage	%	Yes	Yes	Container when used for dimensions, or the container's font when used with font-size or line-height. Percentage has no effect when used with letter-spacing.
Em	em	Yes	Yes	Element's font size unless used in font-size, when it refers to the parent's font size
X-height	ex	Yes	Yes	Container's font (approximated as 0.5 em)
Viewport width	vw	No	Yes	Viewport
Viewport height	vh	No	Yes	Viewport
Smaller viewport dimension	min	No	Yes	Viewport
Width of 0	ch	No	Yes	Container's font
Root element m-height	rem	No	Yes	Root element's font
Grid	gd	No	Yes	Container's text grid (see CSS3 Text Module)

Fig 3.2 CSS Units

One specific instance of this caused us a great deal of trouble when testing the website across different browsers. Internet Explorer does not support CSS viewport units. CSS has a set of units that dynamically change value, without the need for JavaScript, the viewport units. Viewport units work in every other major browser that we tested, however, because Internet Explorer does not support viewport units we were forced to define them in JavaScript, and, convert values from viewport units to pixels. Once we began using JavaScript the problems became less difficult to solve.

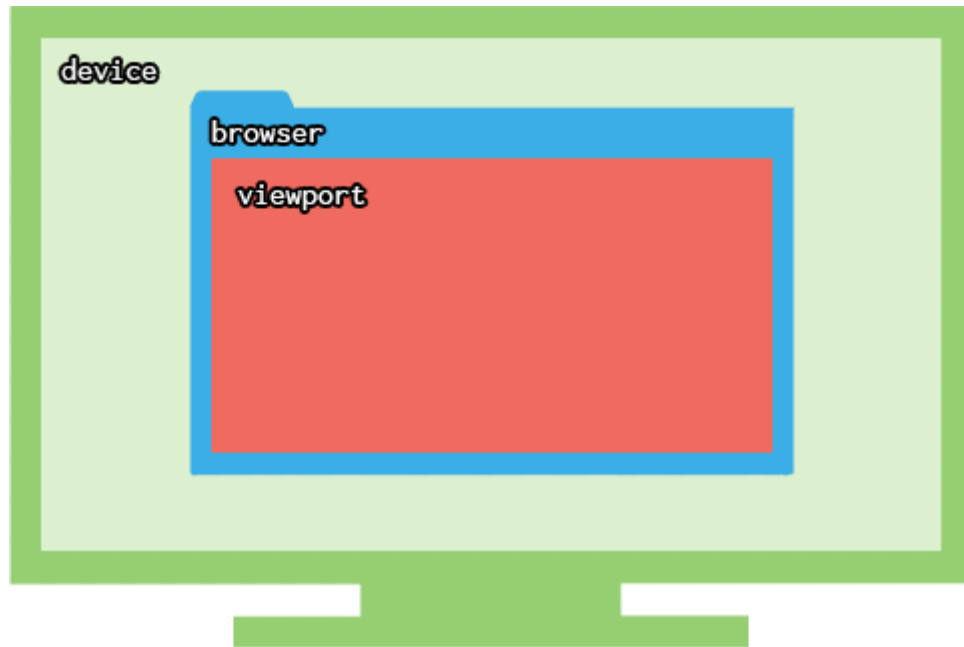


Fig 3.3 Relationship between viewport and Browser

3.4 Browsers

To make sure that our website functioned properly it was tested in the following browsers:

- Internet Explorer
- Mozilla Firefox
- Opera
- Microsoft Edge
- Google Chrome
- Safari
- Android Browser



Fig 3.4 Popular Browsers

And on the following devices:

- Windows Laptop
- Windows Desktop
- Apple Laptop
- Multiple Android Phones
- Multiple iPhones



Fig 3.5 Mobile Devices

3.5 Agile Coding

Agile coding is not the same as following the agile method, agile coding is a process for writing code that is both simple to understand, and, easy to maintain. When a team is spending more time trying to decipher each other's code then they are actually programming, they don't get much done. Writing agile code helps alleviate this problem by standardizing naming conventions and design patterns.

Chapter 4

User Manual

4.1 Access

The website will only be accessible to persons affiliated with OCC, the web address to access the site will be given only to those involved with OCC, or, attending an OCC event.

Certain areas of the site are only accessible during OCC events, or during collections week, regardless of location.

To access certain areas of the site, the browser will need access to geolocation data, this requires the user has enabled JavaScript, and, the user has allowed the browser to use their location.

If the user is unwilling or unable to enable JavaScript, or allow access to their location data, an OCC volunteer with the appropriate access will be able to manually input information.

4.2 Account

If an OCC volunteer or administrator wishes to access privileged areas of the website, they must login using the link in the top-right corner of the screen.

To login a user must have an account created by an administrator, or long term volunteer.

Each account has a username and password associated with it, the password must contain eight or more characters, at least one uppercase, and one lowercase letter, and, at least one number.

4.3 Other

To return to the homepage, at any time, click on the OCC logo in the top-left corner of the screen.

This website has been tested using the following major web browsers, Opera, Mozilla Firefox, Internet Explorer, Microsoft Edge, Safari, and, and Google Chrome.

Chapter 5

Summary

5.1 Implications

As previously stated this website will help alleviate some of the burden on the OCC volunteers who have been tasked with manually inputting the information from donation forms. The information will have a higher level of integrity because there is no chance of a volunteer misreading bad handwriting. Most importantly it will save time the volunteers tasked with data entry will now be able to do other things instead of spending hours trying to decipher thousands of forms.

5.2 Advantages

Aside from the increase in data integrity, and, the time saved, our website will also reduce paper waste since donors will no longer be filling out paper forms. Restrictions on when, and, where certain areas of the website can be accessed make it more difficult to commit fraud by submitting false donation records.



Fig 5.1 Reduce Waste

5.3 Conclusion

We built this website to aid the volunteers at OCC in the task of information management, the site was designed with the intention of removing the time consuming job of manual data input for the OCC volunteers. By having donors input their information digitally, the information has a higher level of integrity, than if a second party were to input the data later. We limited access to certain areas of the based on location, and, date, this makes it more difficult for someone to submit fraudulent donation information. We also gave administrators access to the database, and, long term volunteers the ability to manage events. This will in the long run save time, and, reduce paper waste.

Appendix A: Current Website Layout

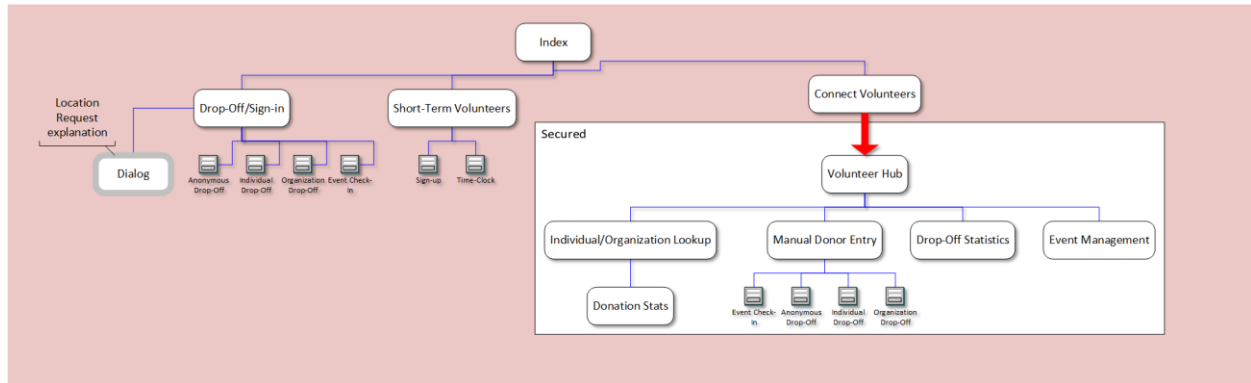


Fig A.1 Website Layout Diagram

Appendix B: Source Code

INDEX.PHP

```
<!DOCTYPE html >
<?php
session_start();
ini_set('error_reporting', E_ALL);
ini_set('display_errors', 1);

function insertInto($table, $columns, $values) {
    $query = "INSERT INTO `{$table}` VALUES (";
    for ($i = 0; $i < count($columns); $i++) {
        if ($i == 0) {
            $query .= "`{$columns[$i]}`";
        } else {
            $query .= ",`{$columns[$i]}`";
        }
    }
    $query .= ")(";
    for ($i = 0; $i < count($values); $i++) {
        if ($i == 0) {
            $query .= "\"{$values[$i]}\"";
        } else {
            $query .= ",\"{$values[$i]}\"";
        }
    }
    $query .= "));";
    return $query;
}
```



```
}

```

```
function selectFrom($field, $table) {
    return "SELECT $field FROM ` $table `";
}

```

```
function deleteFrom($table, $field, $value) {
    return "DELETE FROM $table WHERE $field='$value'";
}

```

```
if (isset($_POST['f8DeleteButton'])) {
    $connection = mysqli_connect("127.0.0.1", "occ", "occ470", "occ")
        OR die('Could not connect to MySQL ' . mysqli_connect_error());

    mysqli_select_db($connection, "occ");

    $query = 'delete from `Event` where eventID=' . $_POST['f8DeleteID'] . ' ';
    $result = mysqli_query($connection, $query);
    mysqli_close($connection);
    unset($_POST['f8DeleteButton']);
    unset($_POST['f8DeleteID']);
}

```

```
if (isset($_POST['loginSubmitButton'])) {
    $connection = mysqli_connect("127.0.0.1", "occ", "occ470", "occ")
        OR die('Could not connect to MySQL ' . mysqli_connect_error());

    mysqli_select_db($connection, "occ");

    $query = selectFrom('*', 'Volunteer');
    $result = mysqli_query($connection, $query);
}

```

```

while ($row = mysqli_fetch_array($result)) {
    if ($_POST['username'] == $row['userName']) {
        if ($_POST['password'] == $row['password']) {
            $_SESSION['username'] = $_POST['username'];
        }
    }
}

mysqli_close($connection);
}
if (isset($_POST['logoutSubmitButton'])) {
    session_start();
    session_unset();
    session_destroy();
    header("Location:index.php");
}
if (isset($_POST['logoValue'])) {
    header("Refresh:0");
    unset($_POST['logoValue']);
}
if (isset($_POST['anonSubmitButton'])) {
    $connection = mysqli_connect("localhost", "occ", "occ470", "occ")
        OR die('Could not connect to MySQL ' . mysqli_connect_error());

    mysqli_select_db($connection, 'occ');
    $query = "call anonymousDropOff('C99504'," . $_POST['boxes'] . ")";
    mysqli_query($connection, $query);
    mysqli_close($connection);
    unset($_POST['anonSubmitButton']);
}

```

```

}
if (isset($_POST['donorSubmitButton'])) {
    $connection = mysqli_connect("localhost", "occ", "occ470", "occ")
        OR die('Could not connect to MySQL ' . mysqli_connect_error());

    mysqli_select_db($connection, 'occ');
    $query = "call personDropOff("
        . "'' . $_POST['firstName'] . "','"
        . "'' . $_POST['middleName'] . "','"
        . "'' . $_POST['lastName'] . "','"
        . "'' . $_POST['suffix'] . "','"
        . "'' . $_POST['street'] . "','"
        . "'' . $_POST['city'] . "','"
        . "'' . $_POST['zip'] . "','"
        . "'' . $_POST['state'] . "','"
        . "'' . $_POST['phone'] . "','"
        . "'' . $_POST['email'] . "','"
        . "'C99504',"
        . "'' . $_POST['boxes'] . '');"

    mysqli_query($connection, $query);
    mysqli_close($connection);
    unset($_POST['donorSubmitButton']);
    unset($_POST['firstName']);
    unset($_POST['middleName']);
    unset($_POST['lastName']);
    unset($_POST['suffix']);
    unset($_POST['street']);
    unset($_POST['city']);

```

```

unset($_POST['zip']);
unset($_POST['state']);
unset($_POST['phone']);
unset($_POST['email']);
unset($_POST['boxes']);
}
if (isset($_POST['eventSubmitButton'])) {
    $connection = mysqli_connect("localhost", "occ", "occ470", "occ")
        OR die('Could not connect to MySQL ' . mysqli_connect_error());

    mysqli_select_db($connection, 'occ');
    $query = "call insertAttendee("
        . "" . $_POST['firstName'] . ","
        . "" . $_POST['middleName'] . ","
        . "" . $_POST['lastName'] . ","
        . "" . $_POST['suffix'] . ","
        . "" . $_POST['street'] . ","
        . "" . $_POST['city'] . ","
        . "" . $_POST['zip'] . ","
        . "" . $_POST['state'] . ","
        . "" . $_POST['phone'] . ","
        . "" . $_POST['email'] . ","
        . "1");";

    mysqli_query($connection, $query);
    mysqli_close($connection);
    unset($_POST['eventSubmitButton']);
    unset($_POST['firstName']);
    unset($_POST['middleName']);

```

```

unset($_POST['lastName']);
unset($_POST['suffix']);
unset($_POST['street']);
unset($_POST['city']);
unset($_POST['zip']);
unset($_POST['state']);
unset($_POST['phone']);
unset($_POST['email']);
}
if (isset($_POST['orgSubmitButton'])) {
    $connection = mysqli_connect("localhost", "occ", "occ470", "occ")
        OR die('Could not connect to MySQL ' . mysqli_connect_error());

    mysqli_select_db($connection, 'occ');
    $query = "call organizationDropOff("
        . "" . $_POST['orgName'] . ","
        . "" . $_POST['street'] . ","
        . "" . $_POST['city'] . ","
        . "" . $_POST['zip'] . ","
        . "" . $_POST['state'] . ","
        . "" . $_POST['phone'] . ","
        . "" . $_POST['email'] . ","
        . "C99504',"
        . "" . $_POST['boxes'] . ",";

    if ($_POST['churchBool'] == 'yes') {
        $query.= "1";
    } else {
        $query.= "0";
    }
}

```

```

$query.= ');';
mysqli_query($connection, $query);
mysqli_close($connection);
unset($_POST['orgSubmitButton']);
unset($_POST['orgName']);
unset($_POST['street']);
unset($_POST['city']);
unset($_POST['zip']);
unset($_POST['state']);
unset($_POST['phone']);
unset($_POST['email']);
unset($_POST['boxes']);
unset($_POST['churchBool']);
}
if (isset($_POST['f7SubmitButton'])) {
    $connection = mysqli_connect("localhost", "occ", "occ470", "occ")
        OR die('Could not connect to MySQL ' . mysqli_connect_error());

    mysqli_select_db($connection, 'occ');
    $query = "call createEvent("
        . "" . $_POST['f7EventName'] . ", "
        . "" . $_POST['f7EventDate'] . ", "
        . "" . $_POST['f7EventStartTime'] . ", "
        . "" . $_POST['f7EventEndTime'] . ", "
        . "" . $_POST['f7EventOrg'] . ", "
        . "" . $_POST['f7EventStreet'] . ", "
        . "" . $_POST['f7EventCity'] . ", "
        . "" . $_POST['f7EventZipCode'] . ", "
        . "" . $_POST['f7EventState'] . ", "
        . "" . $_POST['f7EventPhone'] . ", "

```

```

        . "" . $_POST['f7EventEmail'] . """;
if ($_POST['f7EventChurchBool'] == 'yes') {
    $query.= ""1"";
} else {
    $query.= ""0"";
}
$query.= ",NULL,"
        . "" . $_POST['f7EventLat'] . "","
        . "" . $_POST['f7EventLon'] . "","
        . ""1"";

$query.= ");";
mysqli_query($connection, $query);
mysqli_close($connection);

unset($_POST['f7SubmitButton']);
unset($_POST['f7EventName']);
unset($_POST['f7EventDate']);
unset($_POST['f7EventStartTime']);
unset($_POST['f7EventEndTime']);
unset($_POST['f7EventOrg']);
unset($_POST['f7EventStreet']);
unset($_POST['f7EventCity']);
unset($_POST['f7EventZipCode']);
unset($_POST['f7EventState']);
unset($_POST['f7EventPhone']);
unset($_POST['f7EventEmail']);
unset($_POST['f7EventChurchBool']);
unset($_POST['f7EventLat']);

```

```

unset($_POST['f7EventLon']);
}
?>
<html>
<head>
<title>Operation Christmas Child</title>
<link rel="shortcut icon" href="SP_Icon.png" type="image/png">
<style>
*{
margin: 0px;
padding: 0px;
font-family: sans-serif;
position:relative;
}
html{
zoom:100%;
width:fit-content;
margin-left:auto;
margin-right:auto;
}
body{
background-color: #333;
}
a , .caption,label,#loginLink,#bannerText,#eventTable{
color:white;
}
.text:hover{
background-color: #ee9;
}

```



```
.link:hover,.f7Link:hover{  
    background-color: #69be28;  
    color: white;  
  
}  
  
#bannerText{  
    position:absolute;  
    text-align: center;  
  
}  
  
.form{  
    text-align: center;  
    width: auto;  
    margin-left: auto;  
    margin-right:auto;  
}  
  
table{  
    margin-left: auto;  
    margin-right: auto;  
    text-align: center;  
}  
  
#submitButton{  
    background: #69be28;  
    color:white;  
    width:fit-content;  
}  
  
#header{  
    text-align: right;  
}  
  
#navigation{
```

```
    background: #69be28;
}
#textLinks{
    position:absolute;
}
#loginLink{
    padding-left: 0%;
    padding-right: 0.5%;
}
.formSubmitButton{

    background: #69be28;
    color:white;
}
#main{
    text-align: center;
    margin-left: auto;
    margin-right:auto;
}
#eventTable,#form7Table,#form8Table{

    text-align: center;
    border:solid;
    border-color: #69be28;
}

input{
    border:solid;
    border-radius: 5px;
    border-width: 3px;
```

```

    color: #333;
    border-color: #69be28;
    width:100%;
}
</style>
<script
src="https://ajax.googleapis.com/ajax/libs/jquery/2.2.2/jquery.min.js"></script>
<script>
    function haversine(theta) {
        return Math.pow(Math.sin(theta / 2), 2);
    }
    function latLongToKm(lat1, long1, lat2, long2) {
        dlat = Math.abs(lat1 - lat2);
        dlong = Math.abs(long1 - long2);
        radius = 6371;
        HDR = haversine(dlat * Math.PI / 180) +
            Math.cos(lat1 * Math.PI / 180) *
            Math.cos(lat2 * Math.PI / 180) *
            haversine(dlong * Math.PI / 180);
        return 2 * radius * Math.asin(Math.sqrt(HDR));
    }
    function getLocation() {
        if (navigator.geolocation) {
            navigator.geolocation.getCurrentPosition(getCoords);
        }
    }
    function getCoords(position) {
        lat = parseFloat(position.coords.latitude);
        lon = parseFloat(position.coords.longitude);
    }

```

```

$.get('dropoffCallBack.php', {lat: lat, lon: lon}, function (data) {
    var arr = String(data).split(",");
    var result = [];
    for (i = 0; i < arr.length - 1; i++) {
        result[i] = parseFloat(arr[i]);
    }
    for (i = 0; i < result.length - 1; i = i + 2) {
        var value = latLongToKm(lat, lon, result[i], result[i + 1]);

        if (value < .3) {
            $('#dropoff').show();
        }
    }
});

$.get('eventCallBack.php', {lat: lat, lon: lon}, function (data) {
    var arr = String(data).split(",");
    var result = [];
    for (i = 0; i < arr.length - 1; i++) {
        result[i] = parseFloat(arr[i]);
    }
    for (i = 0; i < result.length - 1; i = i + 2) {
        var value = latLongToKm(lat, lon, result[i], result[i + 1]);
        if (value < .3) {
            $('#event').show();
        }
    }
});
}

setStyles = function () {

```

```
vw = window.innerWidth / 100;  
vh = window.innerHeight / 100;  
vmin = Math.min(vw, vh);  
vmax = Math.max(vw, vh);  
toPx = function (value) {  
    return value + "px";  
};  
$('#header').css({  
    "top": toPx(1.25 * vmin),  
    "width": toPx(100 * vw),  
    "font-size": toPx(2.5 * vmin)  
});  
$('#navigation').css({  
    "top": toPx(2.5 * vmin),  
    "width": toPx(100 * vw),  
    "height": toPx(10 * vmin)  
});  
$('#logoImage').css({  
    "top": toPx(-5 * vmin),  
    "width": toPx(20 * vmin)  
});  
$('#bannerText').css({  
    "width": toPx(100 * vw),  
    "bottom": toPx(3.25 * vmin),  
    "font-size": toPx(2.5 * vmin)  
});  
$('.link').css({  
    "top": toPx(5 * vmin),  
    "width": toPx(60 * vmin),  
    "height": toPx(20 * vmin),
```

```
"font-size": toPx(6 * vmin),
"margin": toPx(2.5 * vmin)
});
$('.f7Link').css({
  "font-size": toPx(2.5 * vmin)

});
$('.text').css({
  "font-size": toPx(5 * vmin),
  "width": toPx(40 * vmin)
});
$('.form').css({
  "top": toPx(8 * vmin),
  "width": toPx(100 * vmin)

});
$('.formSubmitButton').css({
  "font-size": toPx(5 * vmin),
  "width": toPx(22 * vmin)
});
$('#eventImage').css({
  "width": toPx(20 * vmin)
});
$('#eventTable,#form7Table,#form8Table').css({
  "width": toPx(100 * vw),
  "top": toPx(10 * vmin),
  "font-size": toPx(1.5 * vw)
});
$('.f7,.f8').css({
  "font-size": toPx(1.5 * vw),
```

```

        "border": "none"
    });
};

login = function (str1, str2) {
    $('#loginLink').click(function () {
        $('.admin,.dropoff,.index,#form1,#form2,#form3,#form4,.eventTable," +
str1).fadeOut(750,
            function () {
                $(str2).show(2000);
            });
    });
};

init = function () {

    setStyles();
    $('#dropoff').hide();
    $('#event').hide();
    $('.dropoff').hide();
    $('.admin').hide();
    $('.form').hide();
    $('.eventTable').hide();
    $('.locationTable').hide();
};

onClick = function () {
    $('#dropoff').click(function () {
        $('.index').fadeOut(750, function () {
            $('.dropoff').fadeIn(750);
        });
    });
};

```

```
$( '#event' ).click( function () {  
    $( '.index' ).fadeOut( 750, function () {  
        $( '#form2' ).fadeIn( 750 );  
    } );  
});  
  
$( '#individual' ).click( function () {  
    $( '.dropoff' ).fadeOut( 750, function () {  
        $( '#form1' ).fadeIn( 750 );  
    } );  
});  
  
$( '#organization' ).click( function () {  
    $( '.dropoff' ).fadeOut( 750, function () {  
        $( '#form3' ).fadeIn( 750 );  
    } );  
});  
  
$( '#anon' ).click( function () {  
    $( '.dropoff' ).fadeOut( 750, function () {  
        $( '#form4' ).fadeIn( 750 );  
    } );  
});  
  
$( '#admin' ).click( function () {  
    $( '.index' ).fadeOut( 750, function () {  
        $( '.admin' ).fadeIn( 750 );  
    } );  
});  
  
$( '#manageEvents' ).click( function () {  
    $( '.admin' ).fadeOut( 750, function () {  
        $( '.eventTable, #form7, #form8' ).fadeIn( 750 );  
    } );  
});
```



```

$(#logoImage').click(function () {
    $('#hiddenForm').submit();
});
login();
};

```

```

$(document).ready(function () {
    init();
    getLocation();
    $(window).resize(function () {
        setStyles();
    });
    onClick();
});

```

```
</script>
```

```
</head>
```

```
<body>
```

```
<div id="header" >
```

```
<div id="loginLink">
```

```
<?php
```

```

if (isset($_SESSION['username'])) {
    echo "" . $_SESSION['username'];
    echo '<script>';
    echo'login("#form5", "#form6");';
    echo'</script>';
}

```

```
    } else {
        echo 'Login';
        echo '<script>';
        echo'login("#form6","#form5");';
        echo'</script>';
    }
?>

</div>
</div>

<div id="navigation">

    
    <form action="" id="hiddenForm" method="post">
        <input type="hidden" name="logoValue" value="1">
    </form>
    <div id="bannerText">
        <?php
            echo 'Operation Christmas Child';
        ?>
    </div>

</div>

<div id="main">

    <input type="submit" class="index link" id="dropoff" value="Drop-Off">
```

```

<input type="submit" class="index link" id="event" value="Event Check-In">
<?php
if (isset($_SESSION['username'])) {
    echo '<input type = "submit" class = "index link" id = "admin" value =
"Admin">';
    }
?>
<input type="submit" class="dropoff link" id="individual" value="Individual">
<input type="submit" class="dropoff link" id="organization"
value="Organization">
<input type="submit" class="dropoff link" id="anon" value="Anonymous">
<input type="submit" class="admin link" id="manageEvents" value="Manage
Events">
<!--<input type="submit" class="admin link" id="manageLocations"
value="Manage Locations">
<input type="submit" class="admin link" id="dataLookup" value="Data
Lookup">
<input type="submit" class="admin link" id="dataEntry" value="Data Entry">--
>

```

```

<form id="form1" class="form" action="" method="post">
<label for="nameTable1">Donor Drop-Off</label>
<table id="nameTable1">
<tr>
<td><input type="text" class="text" placeholder="First Name"
name="firstName"></td>
<td><input type="text" class="text" placeholder="Middle Name"
name="middleName"></td>
</tr>
<tr>

```

```

        <td><input type="text" class="text" placeholder="Last Name"
name="lastName"></td>
        <td><input type="text" class="text" placeholder="Suffix"
name="suffix"></td>
    </tr>
</table>
<br />
<table id="addressTable1">
    <tr>
        <td><input type="text" class="text" placeholder="Mailing Address"
name="street"></td>
        <td><input type="text" class="text" placeholder="City"
name="city"></td>
    </tr>
    <tr>
        <td><input type="text" class="text" placeholder="State"
name="state"></td>
        <td><input type="text" class="text" placeholder="Zip Code"
name="zip"></td>
    </tr>
</table>
<br />
<table id="otherTable1">
    <tr>
        <td><input type="tel" class="text" placeholder="Phone"
name="phone"></td>
        <td><input type="email" class="text" placeholder="Email"
name="email"></td>
    </tr>
    <tr>
        <td><input type="text" class="text" placeholder="Church Name"
name="church"></td>

```

```

        <td><input type="text" class="text" placeholder="Number of Boxes"
name="boxes"></td>

```

```

    </tr>

```

```

</table>

```

```

    <input type="submit" id="form1Button" class="formSubmitButton"
value="Submit" name="donorSubmitButton">

```

```

</form>

```

```

<form id="form2" class="form" action="" method="post">

```

```

    <!--  -->

```

```

    <br />

```

```

    <label>Event Check-In</label>

```

```

    <table id="nameTable2">

```

```

        <tr>

```

```

            <td><input type="text" class="text" placeholder="First Name"
name="firstName"></td>

```

```

            <td><input type="text" class="text" placeholder="Middle Name"
name="middleName"></td>

```

```

        </tr>

```

```

        <tr>

```

```

            <td><input type="text" class="text" placeholder="Last Name"
name="lastName"></td>

```

```

            <td><input type="text" class="text" placeholder="Suffix"
name="suffix"></td>

```

```

        </tr>

```

```

    </table>

```

```

    <br />

```

```

    <table id="addressTable2">

```

```

        <tr>
            <td><input type="text" class="text" placeholder="Mailing Address"
name="street"></td>
            <td><input type="text" class="text" placeholder="City"
name="city"></td>
        </tr>
        <tr>
            <td><input type="text" class="text" placeholder="State"
name="state"></td>
            <td><input type="text" class="text" placeholder="Zip Code"
name="zip"></td>
        </tr>
    </table>
    <br />
    <table id="otherTable2">
        <tr>
            <td><input type="tel" class="text" placeholder="Phone"
name="phone"></td>
            <td><input type="email" class="text" placeholder="Email"
name="email"></td>
        </tr>
    </table>

    <input type="submit" id="form2Button" class="formSubmitButton"
value="Submit" name="eventSubmitButton">
</form>

```

```

<form id="form3" class="form" action="" method="post">
    <label>Organization Drop-Off</label>
    <table id="boxTable3">

```

```

        <tr>
            <td><input type="text" class="text" placeholder="Number of Boxes"
name="boxes"></td>
        </tr>
    </table>
    <br/>
    <table id="nameTable3">
        <tr>
            <td><input type="text" class="text" placeholder="Organization"
name="orgName"></td>
            <td>
                <input list="churchSelect" class="text" placeholder="Church?"
name="churchBool">
                <datalist id="churchSelect">
                    <option value="yes">
                    <option value="no">
                </datalist>
            </td>
        </tr>
    </table>
    <br />
    <table id="addressTable3">
        <tr>
            <td><input type="text" class="text" placeholder="Mailing Address"
name="street"></td>
            <td><input type="text" class="text" placeholder="City"
name="city"></td>
        </tr>
        <tr>

```

```

        <td><input type="text" class="text" placeholder="State"
name="state"></td>
        <td><input type="text" class="text" placeholder="Zip Code" name
="zip"></td>
    </tr>
</table>
<br />
<table id="otherTable3">
    <tr>
        <td><input type="tel" class="text" placeholder="Phone"
name="phone"></td>
        <td><input type="email" class="text" placeholder="Email"
name="email"></td>
    </tr>
</table>

<input type="submit" id="form3Button" class="formSubmitButton"
value="Submit" name="orgSubmitButton">

</form>

<form id="form4" class="form" action="" method="post">
    <label >Anonymous Drop-Off</label>

    <table id="otherTable4">
        <tr>
            <td><input type="text" class="text" placeholder="Number of Boxes"
name="boxes"></td>
        </tr>

```



```
</table>
```

```
<input type="submit" id="form4Button" class="formSubmitButton"
value="Submit" name="anonSubmitButton">
```

```
</form>
```

```
<form id="form5" class="form" action="" method="post">
```

```
<label >Connect Volunteer Login</label>
```

```
<table id="otherTable5">
```

```
<tr>
```

```
<td><input type="text" class="text" placeholder="Username"
name="username"></td>
```

```
<td><input type="text" class="text" placeholder="Password"
name="password"></td>
```

```
</tr>
```

```
</table>
```

```
<input type="submit" id="form5Button" class="formSubmitButton"
value="Submit" name="loginSubmitButton">
```

```
</form>
```

```
<form id="form6" class="form" action="" method="post">
```

```
<input type="submit" id="form6Button" class="link" value="Logout"
name="logoutSubmitButton">
```

```
</form>
```

```

<form id="form7" class="eventTable" action="" method="post">
  <table id="eventTable" class="eventTable">
    <?php
      $connection = mysqli_connect("127.0.0.1", "occ", "occ470", "occ")
        OR die('Could not connect to MySQL ' . mysqli_connect_error());

      mysqli_select_db($connection, "occ");

      $query = selectFrom('*', 'EventData');
      $result = mysqli_query($connection, $query);
      $str = '<tr class="eventTable">';
      $str.= '<td class="eventTable">ID</td>';
      $str.= '<td class="eventTable">Event</td>';
      $str.= '<td class="eventTable">Date</td>';
      $str.= '<td class="eventTable">Starts</td>';
      $str.= '<td class="eventTable">Ends</td>';
      $str.= '<td class="eventTable">Organization</td>';
      $str.= '<td class="eventTable">Street</td>';
      $str.= '<td class="eventTable">City</td>';
      $str.= '<td class="eventTable">State</td>';
      $str.= '<td class="eventTable">Zip</td>';
      $str.= '</tr>';
      echo $str;
      while ($row = mysqli_fetch_array($result)) {

        $str = '<tr class="eventTable">';

```

```

$str.= '<td class="eventTable">' . $row['eventID'] . '</td>';
$str.= '<td class="eventTable">' . $row['eventName'] . '</td>';
$str.= '<td class="eventTable">' . $row['date'] . '</td>';
$str.= '<td class="eventTable">' . $row['startTime'] . '</td>';
$str.= '<td class="eventTable">' . $row['endTime'] . '</td>';
$str.= '<td class="eventTable">' . $row['officialName'] . '</td>';
$str.= '<td class="eventTable">' . $row['street'] . '</td>';
$str.= '<td class="eventTable">' . $row['city'] . '</td>';
$str.= '<td class="eventTable">' . $row['state'] . '</td>';
$str.= '<td class="eventTable">' . $row['zip'] . '</td>';
$str.= '</tr>';
echo $str;
}

```

```

mysqli_close($connection);
?>

```

```
</table>
```

```
<table id="form7Table" class="eventTable">
```

```

<tr class="eventTable form7Table">
  <td class="eventTable form7Table">
    <input type="text" class="f7" placeholder="Event Name"
name="f7EventName">
  </td>
  <td class="eventTable form7Table">

```

```

        <input type="text" class="f7" placeholder="Event Date"
name="f7EventDate">
        </td>

</tr>
<tr class="eventTable form7Table">
    <td class="eventTable form7Table">
        <input type="text" class="f7" placeholder="Start Time"
name="f7EventStartTime">
        </td>
        <td class="eventTable form7Table">
            <input type="text" class="f7" placeholder="End Time"
name="f7EventEndTime">
        </td>
    </tr>
<tr class="eventTable form7Table">
    <td class="eventTable form7Table">
        <input type="text" class="f7" placeholder="Street"
name="f7EventStreet">
        </td>
        <td class="eventTable form7Table">
            <input type="text" class="f7" placeholder="City" name="f7EventCity">
        </td>
    </tr>
<tr class="eventTable form7Table">
    <td class="eventTable form7Table">
        <input type="text" class="f7" placeholder="State"
name="f7EventState">
        </td>

```

```

        <td class="eventTable form7Table">
            <input type="text" class="f7" placeholder="Zip"
name="f7EventZipCode">
        </td>
    </tr>
    <tr class="eventTable form7Table">

        <td class="eventTable form7Table">
            <input type="text" class="f7" placeholder="Organization"
name="f7EventOrg">
        </td>

        <td class="eventTable form7Table">
            <input type="text" class="f7" placeholder="Phone"
name="f7EventPhone">
        </td>

    </tr>
    <tr class="eventTable form7Table">
        <td class="eventTable form7Table">
            <input type="text" class="f7" placeholder="Email"
name="f7EventEmail">
        </td>
        <td>
            <input list="f7churchSelect" class="f7" placeholder="Church?"
name="f7EventChurchBool">
            <datalist id="f7churchSelect">
                <option value="yes">
                <option value="no">
            </datalist>
        </td>
    </tr>

```

```

<tr class="eventTable form7Table">

    <td class="eventTable form7Table">
        <input type="text" class="f7" placeholder="Latitude"
name="f7EventLat">
    </td>
    <td class="eventTable form7Table">
        <input type="text" class="f7" placeholder="Longitude"
name="f7EventLon">
    </td>

</tr>
<tr class="eventTable form7Table">
    <td class="eventTable form7Table">
        <input type="submit" id="form7CreateButton" class="f7Link"
value="Create Event" name="f7SubmitButton">
    </td>
</tr>

</table>

</form>

<br/>
<br/>
<br/>
<form id="form8" class="form8Table" action="" method="post">
    <table id="form8Table" class="eventTable " >
        <tr class="eventTable ">
            <td class="eventTable form8Table">

```

```

        <input type="submit" id="form8DeleteButton" class="f8Link"
value="Delete Event" name="f8DeleteButton">
    </td>
    <td class="eventTable form8Table">
        <input type="text" class="f8" placeholder="Event ID"
name="f8DeleteID">
    </td>
</tr>
</table>
</form>
</div>
</body>
</html>

```

DropoffCallBack.php

```
<?php
```

```

$connection = mysqli_connect("127.0.0.1", "occ", "occ470", "occ")
    OR die('Could not connect to MySQL ' . mysqli_connect_error());

```

```
mysqli_select_db($connection, "occ");
```

```
$query = 'Select * from `DropOffLocationData`';
```

```
$result = mysqli_query($connection, $query);
```

```
while ($row = mysqli_fetch_array($result)) {
```

```
    echo $row['latitude'];
```

```
    echo ',';
```

```
    echo $row['longitude'];  
    echo ',';  
}  
mysqli_close($connection);  
?>
```

eventCallBack.php

```
<?php
```

```
$connection = mysqli_connect("127.0.0.1", "occ", "occ470", "occ")  
    OR die('Could not connect to MySQL ' . mysqli_connect_error());
```

```
mysqli_select_db($connection, "occ");
```

```
$query = 'Select * from `EventData`';  
$result = mysqli_query($connection, $query);
```

```
while ($row = mysqli_fetch_array($result)) {
```

```
    echo $row['latitude'];  
    echo ',';  
    echo $row['longitude'];  
    echo ',';  
}
```

```
mysqli_close($connection);
```

```
?>
```


References

[Fig 1] <https://www.samaritans-purse.org.uk/wp-content/themes/samaritans-purse/images/newocc/logo.png>

[Fig 2] Image courtesy of David Bretz

[Fig 3] Screen shot of website

[Fig 4] Image courtesy of David Bretz

[Fig 5] Screen shot of Source Code

[Fig 6] Screen shot of Source Code

[Fig 7] Screen shot of Source Code

[Fig 8] Screen shot of Source Code

[Fig 9] Screen shot of Source Code

[Fig 10] Image courtesy of David Bretz

[Fig 11] Screen shot of website

[Fig 12] <http://maxcdn.webappers.com/img/2013/04/diagram2.png>

[Fig 13] <http://marksheet.io/images/device-browser-viewport.png>

[Fig 14] <http://vestavialibrary.org/wp-content/uploads/2016/02/web-browsers.jpg>

[Fig 15] <http://watchtowergeeks.com/wp-content/uploads/2015/03/android-iphone.jpg>

[Fig 16] <https://www.comrise.com/wp-content/themes/comrise/images/paper.png>

[Fig 17] Courtesy of David Bretz