Appendix 1: Client Consultations

Interview Transcript: Important Takeaways From Initial Meeting

February 2017

Could you please describe the current situation/problem that you are facing?

Our GC is having problems because it is difficult for us to make a profit when there are not enough events to sell the products that we have. We end up making a small profit at the end of every year, and we also have a lot of stock that we purchased from India left over for next year.

What is the system you have in place?

The system we currently have in place that is not really working is a Wix.com website.

What problems has this caused?

Wix.com has been hard to work and adjust to. It is not very suitable for what we wanted to do on the site because not all of us could use it at the same time. It has a limit for how many items we could display on the screen for sale, and it does not allow us to design it with similar colours and logos that match with Tiger Watch.

Therefore, what have you tried to solve this problem?

Not much at this point. You were the one who introduced this great idea of creating a website during our GC meeting, and I think will be a suitable way for us to solve this problem.

What requirements would you have for this website?

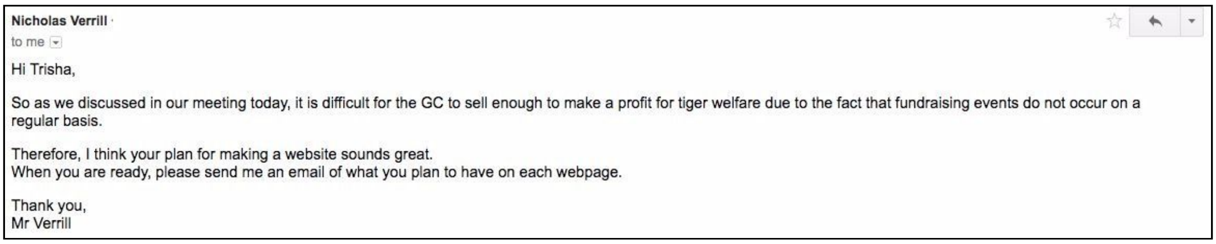
It would be great if it has a page where the products are, and maybe a page where information about members of the GC is provided. When the link is opened, it would be nice to see a screen with information about Dhonk and Ranthambore.

Is there any timescale that you would like to stick to?

It would be great if the website is created by January 2018. I think by then we could start selling some more products since there is a huge gap from January to March until our next event.

Email Exchange With Mr Nicholas Verrill

February 2017



Nicholas Verrill

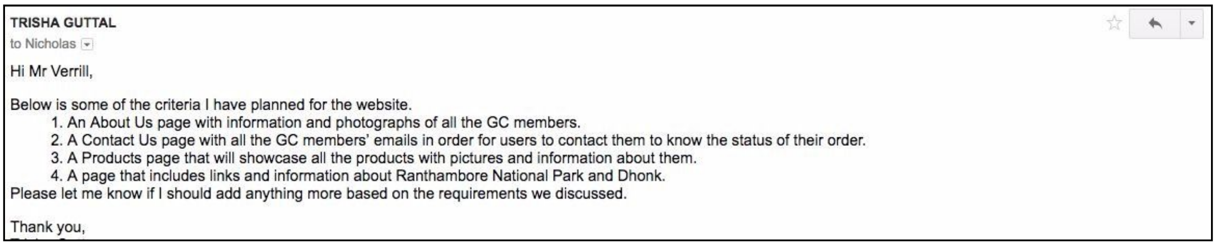
to me

Hi Trisha,

So as we discussed in our meeting today, it is difficult for the GC to sell enough to make a profit for tiger welfare due to the fact that fundraising events do not occur on a regular basis.

Therefore, I think your plan for making a website sounds great.
When you are ready, please send me an email of what you plan to have on each webpage.

Thank you,
Mr Verrill



TRISHA GUTTAL

to Nicholas

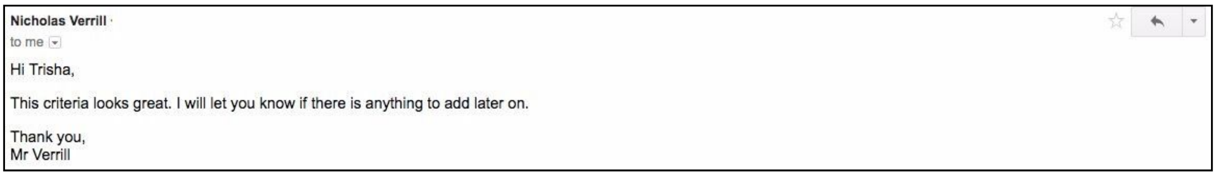
Hi Mr Verrill,

Below is some of the criteria I have planned for the website.

1. An About Us page with information and photographs of all the GC members.
2. A Contact Us page with all the GC members' emails in order for users to contact them to know the status of their order.
3. A Products page that will showcase all the products with pictures and information about them.
4. A page that includes links and information about Ranthambore National Park and Dhonk.

Please let me know if I should add anything more based on the requirements we discussed.

Thank you,



Nicholas Verrill

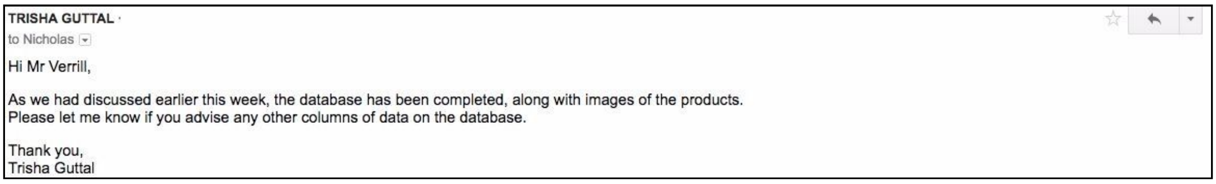
to me

Hi Trisha,

This criteria looks great. I will let you know if there is anything to add later on.

Thank you,
Mr Verrill

April 2017



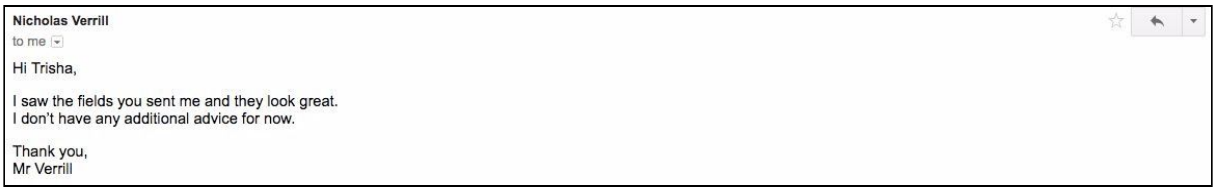
TRISHA GUTTAL

to Nicholas

Hi Mr Verrill,

As we had discussed earlier this week, the database has been completed, along with images of the products.
Please let me know if you advise any other columns of data on the database.

Thank you,
Trisha Guttal



Nicholas Verrill

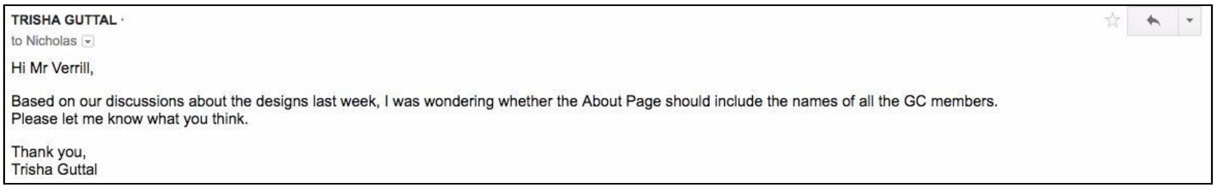
to me

Hi Trisha,

I saw the fields you sent me and they look great.
I don't have any additional advice for now.

Thank you,
Mr Verrill

May 2017



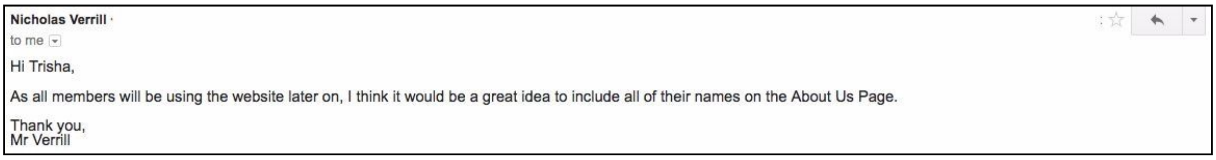
TRISHA GUTTAL

to Nicholas

Hi Mr Verrill,

Based on our discussions about the designs last week, I was wondering whether the About Page should include the names of all the GC members.
Please let me know what you think.

Thank you,
Trisha Guttal



Nicholas Verrill

to me

Hi Trisha,

As all members will be using the website later on, I think it would be a great idea to include all of their names on the About Us Page.

Thank you,
Mr Verrill

Appendix 2: Final Feedback Questionnaire

Client Feedback (September 2017)

Number	Question	Comments
Q1	Are you pleased with final product?	Yes. I feel like this product satisfies the needs of the user and myself. It is visually attractive and well programmed.
Q2	What do you believe are the strengths of this program?	The visual aspects and the easy to use interface are both impressive aspects of this product.
Q3	Where the program should be more developed?	Perhaps there can be more development in terms of the product database. It would be great to have a short sentence description of the materials involved in the production of the products being sold.
Q4	Have you been satisfied with the amount of communication?	Yes, communication has been at a good level.
Q5	Would you use my services again?	Yes, I am willing to consult you further to perhaps develop a few more aspects of this website.
Starting Success Criteria	1. A *Home* page with information about Ranthambore National Park and Dhonk. 2. An *About Us* page with information and photographs of all the GC members. 3. A *Contact Us* page with all the GC members' emails in order for users to contact them to know the status of their order. 4. A *Products* page that will showcase all the products with pictures and information about them. 5. Images displaying each product alongside its name on the *Products* page. 6. A form at the end of the purchasing process that confirms the user's order. 7. A function that emails the user about the products purchased.	1. **Achieved:** I love the design, especially the fact that we can read more about Ranthambore National Park and Dhonk. 2. **Achieved:** The interface is neat and well-designed. 3. **Achieved:** This looks great. I liked the page, and you could perhaps also add emails of all the Global Concern members so the user can choose who to contact. 4. **Achieved**: The products are listed accurately and I loved how the page functions so seamlessly. 5. **Achieved:** The products are well organised and the images are accurate. 6. **Achieved:** the form seems very well programmed and allows me to navigate very easily when they want to purchase something. 7. **Achieved:** The email function works well too and gives the user a detailed digital receipt.

	8. A well organised database of all the products, their prices, availabilities and sizes if applicable. a. A database makes it much easier to add and delete new products.	8. **Achieved.**

Appendix 3: Code Listing

Home Page

```
<%@ page language="java" contentType="text/html; charset=UTF-8"

    pageEncoding="UTF-8"%>

    <%@ taglib prefix="c" uri="http://java.sun.com/jsp/jstl/core"%>

<!DOCTYPE html PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN"
"http://www.w3.org/TR/html4/loose.dtd">

<html lang="en-US">

<head>

<meta charset="UTF-8">

<meta http-equiv="X-UA-Compatible" content="IE=edge">

<meta name="viewport" content="width=device-width, initial-scale=1">

<title>Tiger Watch | Home</title>

<link href="singlePageTemplate.css" rel="stylesheet" type="text/css">

<!--Script tag downloads a font from the Adobe Edge Web Fonts server for use within the web
page.-->

<script>var __adobewebfontsappname__="dreamweaver"</script>

<script src="http://use.edgefonts.net/source-sans-pro:n2:default.js" type="text/javascript"></script>

      <script src="https://oss.maxcdn.com/html5shiv/3.7.2/html5shiv.min.js"></script>

      <script src="https://oss.maxcdn.com/respond/1.4.2/respond.min.js"></script>

    <![endif]-->

</head>

<body>

<!-- Main Container -->

<div class="container">

  <!-- Navigation -->

  <header> <a href="">

    <h3 class="logo">TIGER WATCH</h3>

    </a>

    <nav>
```

```
      <ul>
        <li><a href="HomePage2.jsp">HOME</a></li>
        <li><a href="AboutUsPage2.jsp">ABOUT</a></li>
                          <li> <a href="ContactUsPage2.jsp">CONTACT</a></li>
        <li> <a href="Database.jsp">PRODUCTS</a></li>
      </ul>
    </nav>
  </header>
 <!-- Hero Section -->
  <section class="hero" id="hero">
    <img src="threetigers.JPG" align= "center">
  </section>
  <!-- About Section -->
  <section class="about" id="about">
    <h2 class="hidden">About</h2>
    <p class="text_column"> Tiger Watch is a Global Concern that works to
                                     fundraise for Ranthambore National Park in Rajasthan, India. Tiger
                                     Watch aims to provide support in these areas: &nbsp; <br> &nbsp;
                                     <br> <b><i>Poacher rehabilitation program:</i></b> which
                                     includes, education for Moghiya children, Training of women,
                                     providing an alternative livelihood to ex-poachers, medical help for
                                     these poachers etc. &nbsp; <br> &nbsp; <br> <b><i>Village
                                                                Wildlife Watchers:</i></b> These people are equipped with
nothing but
                                     mobile phones and they use them to warn of imminent poaching dangers.
                                     They send photos of what they have seen via WhatsApp and this helps
                                     the flying squad to apprehend wildlife criminals. The program is
                                     extremely successful. Camera traps are also bought frequently for the
                         national park in order to further protect the tigers. &nbsp; <br> </p>
  </section>
```

```
<!-- Stats Gallery Section -->
<div class="gallery">
  <div class="thumbnail">
    <h1 class="stats">3890</h1>
    <h4>TIGERS WORLDWIDE</h4>
  </div>
  <div class="thumbnail">
    <h1 class="stats">2226</h1>
    <h4>IN INDIA</h4>
  </div>
  <div class="thumbnail">
    <h1 class="stats">3 OUT OF 9 </h1>
    <h4>TIGER SPECIES ARE EXTINCT</h4>
  </div>
  <div class="thumbnail">
    <h1 class="stats">93%</h1>
    <h4>OF HABITAT LOST</h4>
  </div>
</div>
<!-- Parallax Section -->
<section class="banner">
  <h2 class="parallax">DHONK CRAFTS</h2>
  <p class="parallax_description">Dhonk aims to provide
                    sustainable living in Ranthambore by making and selling handicrafts.
                    Tiger Watch purchases majority of products from Dhonk which are then
                    resold on this website. All the funds go to Ranthambore National
                    Park.</p>
</section>
<!-- More Info Section -->
<footer>
```

```
    <article class="footer_column">
      <h3>RANTHAMBORE NATIONAL PARK</h3>
      <img src="funtiger.jpg" alt="" width="400" class="cards">
      <p>Ranthambore National Park is the home of the Tigers that Tiger Watch focuses on. All funds gained from the selling of these products go to Ranthambore National Park.</p>
    </article>
    <article class="footer_column">
      <h3>LOCATION</h3>
      <img src="map.png" alt="" width="400" class="cards">
      <p>The Ranthambore National Park wildlife reserve is located in Prempura, Rajasthan 322001, India </p>
    </article>
  </footer>
 <!-- Footer Section -->
</div>
<!-- Main Container Ends -->
</body>
</html>
```

Contact Us Page

```
<!DOCTYPE html PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN"
"http://www.w3.org/TR/html4/loose.dtd">
<%@ page language="java" contentType="text/html; charset=UTF-8"
    pageEncoding="UTF-8"%>
    <%@ taglib prefix="c" uri="http://java.sun.com/jsp/jstl/core"%>
<html lang="en-US">
<head>
<meta charset="UTF-8">
<meta http-equiv="X-UA-Compatible" content="IE=edge">
<meta name="viewport" content="width=device-width, initial-scale=1">
<title>Tiger Watch | Contact</title>
<link href="contactusstyle.css" rel="stylesheet" type="text/css">
<script>var __adobewebfontsappname__="dreamweaver"</script>
<script src="http://use.edgefonts.net/source-sans-pro:n2:default.js" type="text/javascript"></script>
      <script src="https://oss.maxcdn.com/html5shiv/3.7.2/html5shiv.min.js"></script>
      <script src="https://oss.maxcdn.com/respond/1.4.2/respond.min.js"></script>
</head>

<body>
<!-- Main Container -->
<div class="container">
  <!-- Navigation -->
  <header> <a href="">
    <h3 class="logo">TIGER WATCH</h3>
    </a>
    <nav>
      <ul>
        <li><a href="HomePage2.jsp">HOME</a></li>
        <li><a href="AboutUsPage2.jsp">ABOUT</a></li>
```

```
                    <li> <a href="ContactUsPage2.jsp">CONTACT</a></li>
          <li> <a href="Database.jsp">PRODUCTS</a></li>
        </ul>
      </nav>
    </header>
    <section class="hero" id="hero">
      <img src="threetigers.JPG" align= "center">
    </section>
    <section>
    <div class="gallery">
            <div class="thumbnail"> <a href="#"><img src="call.jpg" alt="" width="2000" height="800"
class="cards"/></a>
        <h4>CALL US AT</h4>
        <p>&nbsp;</p>
        <p class="phonenumber"> +65 00000000</p>
        <p class="phonenumber">&nbsp;</p>
      </div>
      <div class="thumbnail"> <a href="#"><img src="mail.jpg" alt="" width="2000"
class="cards"/></a>
        <h4>EMAIL THE HEAD AT</h4>
        <p>&nbsp;</p>
        <p class="phonenumber"> email@gmail.com</p>
        <p class="phonenumber">&nbsp;</p>
      </div>
       <div class="thumbnail"> <a href="#"><img src="mail.jpg" alt="" width="2000"
class="cards"/></a>
        <h4>EMAIL THE GC AT</h4>
        <p>&nbsp;</p>
        <p class="phonenumber"> email@gmail.com</p>
        <p class="phonenumber">&nbsp;</p>
       </div>
```

```
         </div>
 </section>
 <!-- Hero Section -->
</div>
</body>
</html>
```

About Us Page

```
<%@ page language="java" contentType="text/html; charset=UTF-8"
    pageEncoding="UTF-8"%>
    <%@ taglib prefix="c" uri="http://java.sun.com/jsp/jstl/core"%>

<!doctype html>
<html>
<head>
<meta charset="UTF-8">
<meta http-equiv="X-UA-Compatible" content="IE=edge">
<meta name="viewport" content="width=device-width, initial-scale=1">
<title>Tiger Watch | About Us</title>
<link href="multiColumnTemplate.css" rel="stylesheet" type="text/css">
      <script src="https://oss.maxcdn.com/html5shiv/3.7.2/html5shiv.min.js"></script>
      <script src="https://oss.maxcdn.com/respond/1.4.2/respond.min.js"></script>
    <![endif]-->
</head>
<body>
<div class="container">
  <header> <a href="">
    <h3 class="logo">TIGER WATCH</h4>
    </a>
    <nav>
      <ul>
        <li><a href="HomePage2.jsp">HOME</a></li>
        <li><a href="AboutUsPage2.jsp">ABOUT</a></li>
                        <li> <a href="ContactUsPage2.jsp">CONTACT</a></li>
        <li> <a href="Database.jsp">PRODUCTS</a></li>
      </ul>
    </nav>
  </header>
  <!-- Hero Section -->
  <section class="hero" id="hero">
    <img src="threetigers.JPG" align= "center">
  </section>
<!-- Names and Student Information Section -->
  <div class="row">
    <div class="columns">
      <p class="thumbnail_align"> <img src="unknown2.png" alt="" width="200" height="200" class="thumbnail"/> </p>
      <h4>NAME SURNAME</h4>
      <p>Name is an 11th grader who  believes that Tiger Watch is a platform that allows her to indirectly help increase the dwindling Tiger population to something much larger by raising funds and awareness around the school. When she has free time, she loves to make videos, sing and travel.</p>
    </div>
    <div class="columns">
      <p class="thumbnail_align"> <img src="unknown2.png" alt="" width="200" class="thumbnail"/> </p>
      <h4>NAME SURNAME</h4>
      <p>Name is an 11th grader who has lived in India for 13 years and would like to make a difference and raise awareness about the issues of poaching of endangered tigers in Ranthambore, India. She is passionate about animals and believes that all animals should be treated with respect. </p>
    </div>
    <div class="columns">
      <p class="thumbnail_align"> <img src="unknown2.png" alt="" width="200" height="200" class="thumbnail"/> </p>
      <h4>NAME SURNAME</h4>
      <p>Name is an 11th grader that is part of the Tiger Watch GC at UWCSEA East Campus. He is part of Tiger Watch because he cares about Tigers and is linked to India as he lived there for three years. He is passionate about traveling and the environment.</p>
    </div>
    <div class="columns">
```

```
      <p class="thumbnail_align"> <img src="unknown2.png" alt="" width="200" height="200" class="thumbnail"/> </p>

      <h4>NAME SURNAME</h4>

      <p>Name is an 11th grader passionate about animals. She has always been concerned about environmental conservation. She joined tiger watch because she knew about the Tiger treasure hunt activities that they run every year and wanted to get more involved.  </p>

    </div>

  <div class="row2">

    <div class="columns2">

      <p class="thumbnail_align"> <img src="unknown2.png" alt="" width="200" height="200" class="thumbnail"/> </p>

      <h4>NAME SURNAME</h4>

      <p>Name is an 11th grader, part of tiger watch to help spread awareness and give a voice to the tigers with no voice in human society. She is passionate about equality for animals and travelling.</p>

    </div>

    <div class="columns2">

      <p class="thumbnail_align"> <img src="unknown2.png" alt="" width="200" height="200" class="thumbnail"/> </p>

      <h4>NAME SURNAME</h4>

      <p>Name is an 12th Grader who is part of Tiger Watch GC and thinks that it is important to bring awareness about problems such as poaching and the low number of tigers around the globe. She is passionate about travelling and new experiences/people that it comes with.</p>

    </div>

    <div class="columns2">

      <p class="thumbnail_align"> <img src="unknown2.png" alt="" width="200" height="200" class="thumbnail"/> </p>

      <h4>NAME SURNAME</h4>

      <p>Name is an 11th grader who is a part of Tiger Watch GC. He lived in India for 12 years, so he is well aware of the issue and wantS to take action to help.</p>

    </div>

    <div class="columns2">

      <p class="thumbnail_align"> <img src="unknown2.png" alt="" width="200" height="200" class="thumbnail"/> </p>

      <h4>NAME SURNAME</h4>
    <p>Name is an 11th grader who is part of the Tiger Watch GC. She believes that issues relating to
animal poaching must be eradicated and believes that she can make a change by being a part of this
Global Concern. </p>

</div>

  <div class="social">

  </div>

</div>

</body>

</html>
```

Database/Products Page

```
<%@ page language="java" contentType="text/html; charset=UTF-8"
    pageEncoding="UTF-8"%>
    <%@ taglib prefix="c" uri="http://java.sun.com/jsp/jstl/core"%>
<%@page import="java.sql.DriverManager"%>
<%@page import="java.sql.ResultSet"%>
<%@page import="java.sql.Statement"%>
<%@page import="java.sql.Connection"%>

<%
<!--Database connection-->
String driverName = "com.mysql.jdbc.Driver";
String connectionUrl = "jdbc:mysql://localhost:3306/";
String dbName = "Products";
String dbTable="products";
String userId="root";
String password="peace123";

try {
Class.forName(driverName);
} catch (ClassNotFoundException e) {
e.printStackTrace();
}

Connection connection = null;
Statement statement = null;
ResultSet resultSet = null;
%>
<html lang="en-US">
```

```
<head>
<meta charset="UTF-8">
<meta http-equiv="X-UA-Compatible" content="IE=edge">
<meta name="viewport" content="width=device-width, initial-scale=1">
<title>Tiger Watch | Products</title>
<link href="databasedesign.css" rel="stylesheet" type="text/css">
<!--Script tag downloads a font from the Adobe Edge Web Fonts server for use within web page.-->
<script>var __adobewebfontsappname__="dreamweaver"</script>
<script src="http://use.edgefonts.net/source-sans-pro:n2:default.js" type="text/javascript"></script>
      <script src="https://oss.maxcdn.com/html5shiv/3.7.2/html5shiv.min.js"></script>
      <script src="https://oss.maxcdn.com/respond/1.4.2/respond.min.js"></script>
<title>Tiger Watch | Products</title>
</head>
<div class="container">
  <!-- Navigation -->
  <header> <a href="">
    <h3 class="logo">TIGER WATCH</h3>
    </a>
    <nav>
      <ul>
                    <li><a href="HomePage2.jsp">HOME</a></li>
                    <li><a href="AboutUsPage2.jsp">ABOUT</a></li>
                    <li> <a href="ContactUsPage2.jsp">CONTACT</a></li>
                    <li> <a href="Database.jsp">PRODUCTS</a></li>
      </ul>
    </nav>
  </header>
<!-- Hero section -->
<section class="hero" id="hero">
    <img src="threetigers.JPG" align= "center">
  </section>

  <br>

  <br>

<table class= "dbtable" id="dbtable" style="width:100%">

<tr>


</tr>

<!--Titles of table, first row-->

<tr>

    <td bgcolor="#C67643"><font color="white"><b>id</b></font></td>

                <td bgcolor="#C67643"><font color="white"><b>Product Name</b></font></td>

                <td bgcolor="#C67643"><font color="white"><b>Product Size</b></font></td>

                <td bgcolor="#C67643"><font color="white"><b>Product Price</b></font></td>

                <td bgcolor="#C67643"><font color="white"><b>Image</b></font></td>

                <td bgcolor="#C67643"><font color="white"><b>Availability</b></font></td>

                <td bgcolor="#C67643"><font color="white"><b>Order Quantity</b></font></td>

  </tr>


<%

try{


//connection = DriverManager.getConnection(connectionUrl+"/"+dbName,userId,password);


connection =
DriverManager.getConnection("jdbc:mysql://localhost:3306/Products",userId,password);


            statement=connection.createStatement();

String sql ="SELECT * FROM Products.products";


resultSet = statement.executeQuery(sql);
while(resultSet.next()){
%>
<!-- Getting products from database -->
<tr bgcolor="#ffffff" align="center" style = width:100%, border: 1px black, font: "Times New Roman">
<form action="/ComputerScienceIA/Controller?action=purchasing" method="post">
<td><%=resultSet.getString("id") %></td>
<td><%=resultSet.getString("Product Name") %></td>
<td><%=resultSet.getString("Product Size") %></td>
<td><%=resultSet.getString("Product Price") %></td>
<td><img src="http://localhost:8080/ComputerScienceIA/Controller?action=getImage&id=<%=resultSet.getString("id")%>" width=60/></td>
<td><%=resultSet.getString("Avaliability") %></td>
<td><input type="text" name="buyProductId<%=resultSet.getString("id")%>" id="buyProductId<%=resultSet.getString("id")%>"/></td>
</tr>


<%
}
} catch (Exception e) {
e.printStackTrace();
}
%>
</table>
<!-- purchase button -->
<input id="submit" class="submit" type="submit" value="Purchase"></input>
</form>
</div>
</body>
```

</html>

CartForm

```
<%@ page language="java" contentType="text/html; charset=UTF-8"
            pageEncoding="UTF-8"%>
<!-- Importing JSTL -->
<%@ taglib prefix="c" uri="http://java.sun.com/jsp/jstl/core"%>
<!DOCTYPE html PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN"
"http://www.w3.org/TR/html4/loose.dtd">

<meta charset="UTF-8">
<meta http-equiv="X-UA-Compatible" content="IE=edge">
<meta name="viewport" content="width=device-width, initial-scale=1">
<title>Tiger Watch | Forms</title>
<link href="cartformdesign.css" rel="stylesheet" type="text/css">
<!--Script tag downloads a font from the Adobe Edge Web Fonts server for use within web page.-->
<script>var __adobewebfontsappname__="dreamweaver"</script>
<script src="http://use.edgefonts.net/source-sans-pro:n2:default.js" type="text/javascript"></script>
<div class="container">
  <!-- Navigation -->
  <header> <a href="">
    <h3 class="logo">TIGER WATCH</h3>
    </a>
    <nav>
      <ul>
        <li><a href="HomePage2.jsp">HOME</a></li>
        <li><a href="AboutUsPage2.jsp">ABOUT</a></li>
                            <li> <a href="ContactUsPage2.jsp">CONTACT</a></li>
        <li> <a href="Database.jsp">PRODUCTS</a></li>
      </ul>
    </nav>
  </header>
  <section class="hero" id="hero">
    <img src="threetigers.JPG" align= "center">
  </section>
<!--Title-->
  <h2 id="title" style="text-align: center">
            <font face="Times New Roman" color="#C67643"> Shopping Cart Form </font>
</h2>
<!-- Form for inputting email, name, last name, mentor group -->
<form action="/ComputerScienceIA/Controller?action=ConfirmOrder" method="post">
            Email:<br> <input id="textfield" type="text" name="email"
pattern="/^[a-zA-Z0-9.!#$%&’*+/=?^_`{|}~-]+@[a-zA-Z0-9-]+(?:\.[a-zA-Z0-9-]+)*$/"
    required><br> <br>
            First Name:<br> <input id="textfield" type="text" name="firstname" required><br>
            <br> Last Name:<br> <input id="textfield" type="text" name="lastname" required><br>
            <br> Student's Mentor Group:<br> <input id="textfield" type="text"
                        name="mentor" required><br> <br>
<!-- Shows products selected from products page -->
            <h2 id="h4" style="text-align: left">
                        <font face="Times New Roman" color="black"> Product Amounts </font>
            </h2>


            <table>


                        <tr>
                                    <td><b><font face="Times New Roman" color="black">id</font></b></td>
                                    <td><b><font face="Times New Roman" color="black">
                                                            Image</font></b></td>
                                    <td><b><font face="Times New Roman" color="black">Product
                                                            Name</font></b></td>
                                    <td><b><font face="Times New Roman" color="black">
                                                Price</font></b></td>
                                                <td><b><font face="Times New Roman" color="black">Quantity</font></b></td>
                        </tr>
<!-- JSTL -->
                        <c:forEach var="entry" items="${sessionScope.shoppingCart}">
                        <tr>
                        <td><c:out value="${entry.key}" /></td>
                        <td><img src="http://localhost:8080/ComputerScienceIA/Controller?action=getImage&id=${entry.value.id}" width=60/></td>
                        <td><c:out value="${entry.value.name}" /></td>
                        <td><c:out value="${entry.value.price}" /></td>
                        <td><c:out value="${entry.value.quantity}" /></td>
                        </tr>
                        </c:forEach>
<!-- Total price -->
<tr>
<td>totalPrice=${requestScope.totalPrice}</td>
</tr>
            </table>
<!-- Submit button -->
                        <input id="submit" type="submit"
                                    value="Confirm Order">
</form>
</div>
</body>
</html>
```

Controller Servlet

```
import java.io.IOException;

//libraries imported

//import javax.servlet.jsp.jstl.sql.ResultSupport

import java.io.OutputStream;

import java.sql.Blob;

import java.sql.Connection;

import java.sql.DriverManager;

import java.sql.PreparedStatement;

import java.sql.ResultSet;

import java.sql.Statement;

import java.util.ArrayList;

import java.util.Enumeration;

import java.util.HashMap;

import java.util.Iterator;

import java.util.List;

import java.util.Map;

import java.util.Properties;

import java.util.Set;


import javax.servlet.http.Cookie;

import javax.servlet.ServletException;

import javax.servlet.annotation.WebServlet;

import javax.servlet.http.HttpServlet;

import javax.servlet.http.HttpServletRequest;

import javax.servlet.http.HttpServletResponse;

import javax.servlet.http.HttpSession;


import dataclasses.Products;
import javax.servlet.ServletException;
import javax.servlet.annotation.WebServlet;
import javax.servlet.http.HttpServlet;
import javax.servlet.http.HttpServletRequest;
import javax.servlet.http.HttpServletResponse;
import java.util.Properties;
import javax.mail.Message; //Models an email message
import javax.mail.MessagingException; //Base for all exceptions thrown by the class
import javax.mail.PasswordAuthentication; //Repository for a username and password
import javax.mail.Session; //A multithreaded object that represents the connection factory
import javax.mail.Transport; //Models a message transport mechanism for sending an email message
import javax.mail.internet.InternetAddress; //Represents Internet email address
import javax.mail.internet.MimeMessage; //Message in an abstract class
import javax.swing.JOptionPane;

/**
 * Servlet implementation class Controller
 */
@WebServlet("/Controller")
public class Controller extends HttpServlet {
        private static final long serialVersionUID = 1L;

        /**
         * @see HttpServlet#HttpServlet()
         */
        public Controller() {
                super();
                // TODO Auto-generated constructor stub
        }
```

```
/**
 * @see HttpServlet#doGet(HttpServletRequest request, HttpServletResponse
 *      response)
 */
protected void doGet(HttpServletRequest request, HttpServletResponse response)
        throws ServletException, IOException {
    // TODO Auto-generated method stubprotected void
    // doGet(HttpServletRequest request, HttpServletResponse response)
    // Get the action parameter form the url,
    // eg. www.google.com?action=login
    // will return "login"

    String action = request.getParameter("action");

    // get the session and create one if it doesn't already exist
    HttpSession session = request.getSession();

    // try to open a database connection
    Connection connection = getDatabaseConnection();

    if (action.equals("ConfirmOrder")) {
        String email = request.getParameter("email");
        String firstname = request.getParameter("firstname");
        String lastname = request.getParameter("lastname");
        String mentor = request.getParameter("mentor");
        StringBuffer emailBody=new StringBuffer();

        emailBody.append("Thank you for your order:"+ "\n");
```

```
  ShoppingCartItem sci;
            Map shoppingCart = (HashMap) session.getAttribute("shoppingCart");
            float totalPrice = 0;
            for (Object xxx : shoppingCart.keySet()) {

                sci = ((ShoppingCartItem) shoppingCart.get(xxx));
                totalPrice = totalPrice + (sci.getPrice() * sci.getQuantity());
                emailBody.append("Your Product is: "+sci.name+ "\n" + "The quantity of this product is: " +sci.quantity+ "\n" + "The price of this product is: " +sci.price+ "\n" + "\n");
            }
//sending the email draft
            emailBody.append("\n" + "\n" + "The Total Price will be:"+totalPrice);
            sendMail("email@gmail.com", "passwordhiddenduetoIBrestrictions", email, "Your Tiger Watch Order", emailBody.toString());
            getServletContext().getRequestDispatcher("/" + "HomePage2.jsp").forward(request, response);
        }

        if (action.equals("purchasing"))

        {
            if (session.getAttribute("shoppingCart") == null) {
                session.setAttribute("shoppingCart", new HashMap());
            }
            Map shoppingCart = (HashMap) session.getAttribute("shoppingCart");

            Enumeration e = request.getParameterNames();
            // ArrayList<TigerWatchProduct> productsToBuy=new
            // ArrayList<TigerWatchProduct>();
            ShoppingCartItem sci;
            while (e.hasMoreElements()) {
                Object obj = e.nextElement();
                String fieldName = (String) obj;
                String fieldValue = request.getParameter(fieldName);
                if (fieldName.startsWith("buyProductId") && fieldName.length() >
12 && !fieldValue.equals("")) {
                    String id = fieldName.substring(12);
                    System.out.println("buying " + fieldValue + " of " + id);
                    if (shoppingCart.containsKey(id)) {
                        try {
                            sci = (ShoppingCartItem)
(shoppingCart.get(id));
                            int current_quantity = sci.getQuantity();
                            int additional_quantity =
Integer.parseInt(fieldValue);
                            int new_quantity = current_quantity +
additional_quantity;
                            sci.setQuantity(new_quantity);
                            shoppingCart.put(id, sci);

                        } catch (Exception ex) {
                            ex.printStackTrace();
                        }
                        // Then add the new quantity to existing quantity
                        // Integer p=(Product)shoppingCart.get(id);
                        //
p.setOrderQuantity(String.parseInt(fieldValue))+p.getOrderQuantity();
                    } else {
                        sci = getShoppingCartItem(id);
                        sci.setQuantity(Integer.parseInt(fieldValue));
                        shoppingCart.put(id, sci);
                    }
                }

                System.out.println(fieldName + " : " + fieldValue);
            }
            System.out.println("Contents of session shopping cart");
            float totalPrice = 0;
            for (Object xxx : shoppingCart.keySet()) {
                System.out.println(xxx);
                System.out.println(shoppingCart.get(xxx));
                sci = ((ShoppingCartItem) shoppingCart.get(xxx));
                totalPrice = totalPrice + (sci.getPrice() * sci.getQuantity());
            }
            request.setAttribute("totalPrice", totalPrice);
            getServletContext().getRequestDispatcher("/" +
"CartForm.jsp").forward(request, response);
            System.out.println(totalPrice);
            return;
        }

        if (action.equals("getImage")) {
            System.out.println("getting an image in doGet, id=" +
request.getParameter("id"));
            byte[] image;
            String id = request.getParameter("id");
            image = getImage(id, connection);

            if (image != null) {
                response.setHeader("expires", "0");
                response.setContentType("image/png");
                OutputStream os = response.getOutputStream();
```

```
                        os.write(image);
                        os.flush();
                        os.close();
                    } else {

                    }
                    return;
                }
            }

            public String sendMail(String Email, String Password, String ToEmail, String Subject, String
Text) {

                String Msg;

                final String username = Email;
                final String password = Password;

                Properties props = new Properties();
                props.put("mail.smtp.auth", true);
                props.put("mail.smtp.starttls.enable", true);
                props.put("mail.smtp.host", "smtp.gmail.com");
                props.put("mail.smtp.port", "587");
                props.setProperty("mail.smtp.ssl.trust", "smtp.gmail.com");

                Session session = Session.getInstance(props, new javax.mail.Authenticator() {
                    @Override
                    protected PasswordAuthentication getPasswordAuthentication() {
                        return new PasswordAuthentication(username, password);
                    } // Return any saved Password Authentication for this (store or
                        // transport) URLName
                });


                try {


                    Message message = new MimeMessage(session); // Create a default

                // MimeMessage object
                    message.setFrom(new InternetAddress(Email));
                    message.setRecipients(Message.RecipientType.TO,
InternetAddress.parse(ToEmail));
                    message.setSubject(Subject);
                    message.setText(Text);


                    Transport.send(message); // Sending the message
                    Msg = "Thank you for ordering from Tiger Watch! Your order is:" + ". We
hope you order from us once again!";
                    return Msg;


                } catch (MessagingException e) {
                    return e.toString();
                }
            }


            // close the active database connection
            void closeDatabaseConnection(Connection connection) {


                try {
                    connection.close();
                } catch (Exception e) {
                    e.printStackTrace();
                }
            }

            /**
             * @see HttpServlet#doPost(HttpServletRequest request, HttpServletResponse
             *      response)
             */
            protected void doPost(HttpServletRequest request, HttpServletResponse response)
                    throws ServletException, IOException {
                // TODO Auto-generated method stub
                doGet(request, response);
            }

            void insertOrder(String firstname, String lastname, String studentfirstname, String
studentlastname,
                    String studentgrade, String studentmentor, String email) {

                Connection connection;

                try {
                    connection = getDatabaseConnection();
                    PreparedStatement ps = connection.prepareStatement("insert into Orders
values (?,?,?,?,?,?,?)");
                    ps.setString(1, firstname);
                    ps.setString(2, lastname);
                    ps.setString(3, studentfirstname);
                    ps.setString(4, studentlastname);
                    ps.setString(5, studentgrade);
                    ps.setString(6, studentmentor);
                    ps.setString(7, email);
                    ps.executeUpdate();
                    connection.close();
                } catch (Exception e) {
                    e.printStackTrace();
                }
            }

            byte[] getImage(String id, Connection connection) {

                System.out.println("in getImage, id=" + id);
                byte[] image = null;
                try {
                    Statement stmt = (Statement) connection.createStatement();
                    ResultSet rs;
                    rs = stmt.executeQuery("select * from Products.products where id=" + id);
                    if (rs.next()) {
                        Blob blob = rs.getBlob("Image");
                        if (blob != null) {
                            int blobLength = (int) blob.length();

                            image = blob.getBytes(1, blobLength);
                            blob.free();
                            System.out.println("Got picture for " + rs.getString("id") + "
of length " + blobLength);
                        }
                    }
                } catch (Exception e) {
                    e.printStackTrace();
                }
```

```
		return image;
	}
	/*
	 * List getAllProducts(Connection connection) {
	 * System.out.println("in getAllProducts"); List<Products> toReturn = new
	 * ArrayList<>(); try { Statement stmt = (Statement)
	 * connection.createStatement(); ResultSet rs; rs =
	 * stmt.executeQuery("select * from products"); Products p; while
	 * (rs.next()) { p = convertDatabaseRecordToProductsObject(rs);
	 * toReturn.add(p); System.out.println("Added product with id " +
	 * rs.getInt("id") + " to the list"); } } catch (Exception e) {
	 * e.printStackTrace(); } return toReturn; }
	 */

	Products convertDatabaseRecordToProductObject(ResultSet rs) throws Exception {
		Products p = new Products();
		p.setId(rs.getInt("id"));
		p.setName(rs.getString("name"));
		p.setColour(rs.getString("colour"));
		p.setSize(rs.getString("size"));
		p.setPrice(rs.getFloat("price"));
		p.setDescription(rs.getString("description"));
		p.setPromotion(rs.getBoolean("promotion"));
		p.setStocklevel(rs.getInt("stocklevel"));
		return p;
	}

	boolean tryToLogin(String email, String password, Connection con) {
		return true;
	}
```

```
String getCookie(HttpServletRequest req, String cookieName) {
    Cookie[] cookies = req.getCookies();
    if (cookies != null) {
        for (Cookie retrievedCookie : cookies) {
            if (cookieName.equals(retrievedCookie.getName()))
                return retrievedCookie.getValue();
        }
    }
    return null;
}

Connection getDatabaseConnection() {

    String mySqlUrl = "jdbc:mysql://localhost:3306/Products";
    Properties userInfo = new Properties();
    userInfo.put("user", "root");
    userInfo.put("password", "peace123");
    Connection con = null;

    // try to connect to the database and get a list of products
    try {
        // Load the database connector drive
        Class.forName("com.mysql.jdbc.Driver");
        con = DriverManager.getConnection(mySqlUrl, userInfo);
        return con;
    } catch (Exception e) {
        e.printStackTrace();
    }
```

```
        return con;
    }

    ShoppingCartItem getShoppingCartItem(String id) {
        ShoppingCartItem sc = new ShoppingCartItem();
        try {
            Connection connection = getDatabaseConnection();
            Statement statement = connection.createStatement();
            String sql = "SELECT * FROM Products.products where id=" + id;

            ResultSet resultSet = statement.executeQuery(sql);

            while (resultSet.next()) {

                sc.setId(resultSet.getString("id"));
                sc.setName(resultSet.getString("Product Name"));
                sc.setPrice(Float.parseFloat(resultSet.getString("Product Price")));
            }
        } catch (Exception e) {
            e.printStackTrace();
        }

        return sc;
    }
}
```