

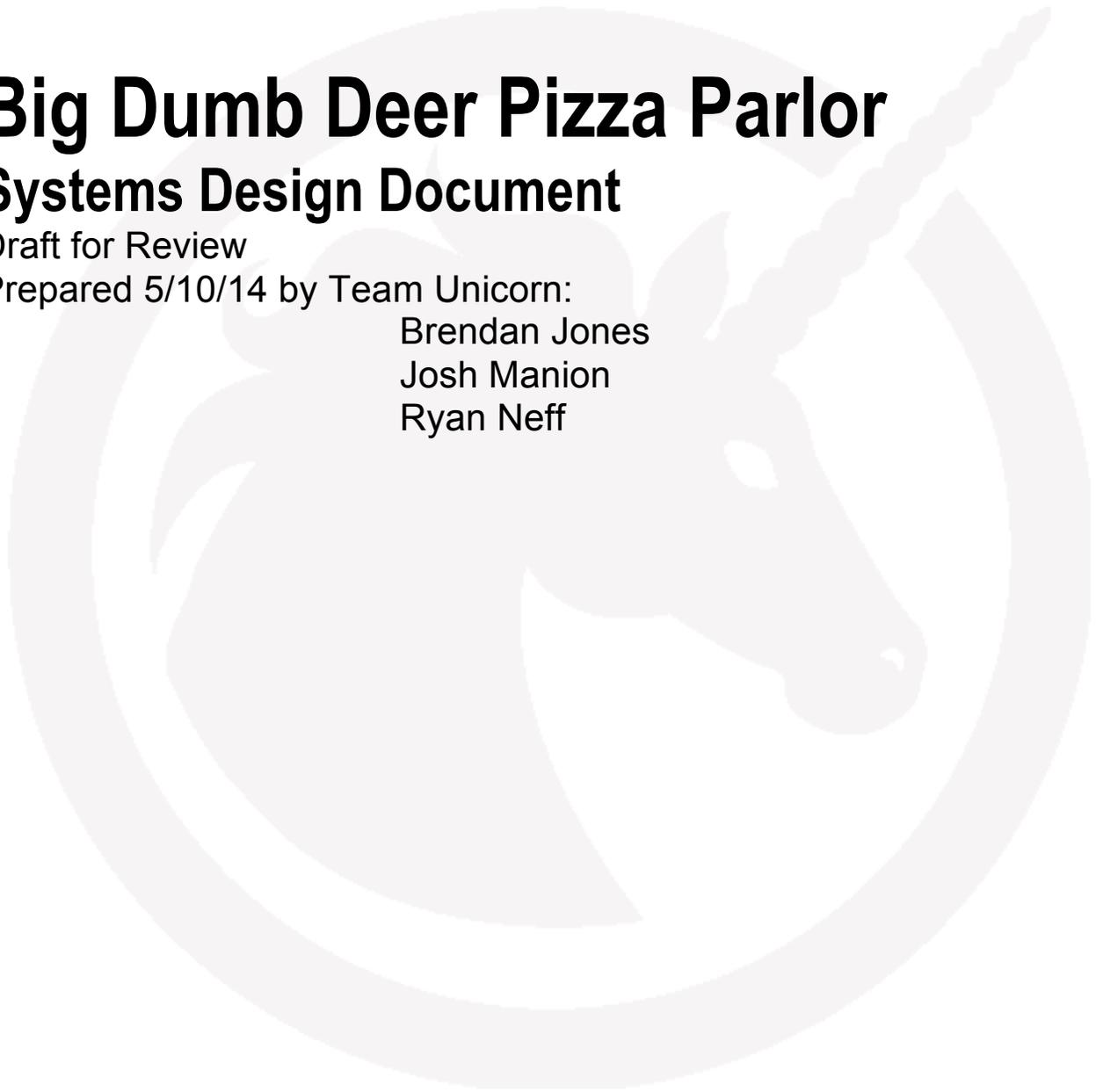
# **Big Dumb Deer Pizza Parlor**

## **Systems Design Document**

Draft for Review

Prepared 5/10/14 by Team Unicorn:

Brendan Jones  
Josh Manion  
Ryan Neff



# Table of Contents

<b>Introduction</b> .....	<b>2</b>
<b>Purpose and Scope</b> .....	<b>3</b>
Purpose.....	3
Scope.....	3
Out of Scope.....	3
<b>Project Plan</b> .....	<b>4</b>
Web Languages .....	4
Database Applications.....	4
Hardware and Software Requirements.....	4
Functional Requirements .....	5
Information Requirements .....	5
File Structure.....	6
<b>Test Plan</b> .....	<b>13</b>
<b>Testing Approach</b> .....	<b>13</b>
Glass Box Testing .....	13
Black Box Testing.....	13
User Acceptance Testing.....	14
<b>Test Environment</b> .....	<b>14</b>
Location.....	14
Testing Hours.....	14
How the tests will be conducted.....	14
Personnel.....	14
<b>Appendices</b> .....	<b>15</b>
<b>Appendix A: Project Schedule</b> .....	<b>16</b>
<b>Appendix B: Use Case Scenarios</b> .....	<b>17</b>
UC001 Order Pizza.....	17
UC002 Order Summary .....	18
UC003 Confirm Order.....	19
UC004 Send Customer Email .....	19
UC005 Admin Login .....	20
<b>Appendix C: Site Map</b> .....	<b>21</b>
<b>Wire Frame: Pizza Entry Page</b> .....	<b>22</b>
<b>Wire Frame: Customer Information Page</b> .....	<b>23</b>
<b>Wire Frame: Confirmation Page</b> .....	<b>24</b>
<b>File Structure:</b> .....	<b>25</b>
<b>Approvals</b> .....	<b>26</b>
<b>Change Log</b> .....	<b>27</b>

## Introduction

Big Dumb Deer Pizza Parlor (hereafter referred to as BDDPP) is a pizzeria chain, with corporate headquarters in Lynnwood Washington. The business began 3 years ago with John Smith selling pizza from the back of his van. The business has rapidly expanded from a hobby type business to a moderate brand-name enterprise with three locations and sixteen full time employees, as well as a cadre of part time staff. As the business has grown, their current website has not been able to meet their business needs. John Smith has commissioned Team Unicorn to provide a new website that better serves BDDPP business needs.

The purpose of this Systems Design Specification is to present the steps of the Systems Development Life Cycle (SDLC), which include design and implementation of the new BDDPP website. This document contains the following:

- **Purpose and Scope:** The basic functional requirements of the project are clearly defined.
- **Project Plan:** The phase of the project from design to implementation are reviewed.
- **Technical Specifications:** – This section discusses the type of hardware and network architecture to be used by the new system.
- **Functional Specifications:** – This includes each section of the site and it's functional requirements, as well as the visual layout.
- **Test Plan:** This outlines the testing methods and use case scenarios employed to ensure that quality standards are met.
- **Appendix:** The Appendices will contain all of the supporting documentation pertaining to this project.

## Purpose and Scope

### Purpose

Team Unicorn has been requested to provide an interactive website that will allow customers TBDDPP customers to enter their pizza orders, accept customer delivery information, accept customer payment information, provide the customer with an order confirmation page before submitting their final order, provide the customer with an estimated delivery time and order confirmation, and send an automated email to both the customer and TBDDPP confirming expected delivery time. Regarding the system administration capabilities, administrators will be able to log in, and then generate reports about existing customers and outstanding customer orders.

### Scope

The new system will include information related to the entities that were explicitly provided as requirements by TBDDPP. Support for the following information will be included in the TO-BE system:

- Customer contact/delivery information
- Customer Credit Card information
- Pizza order parameters
- Estimated delivery times
- Administrator report generation

### Out of Scope

The initial implementation of the website will only apply to one location. Any additional locations are outside the scope of this project. The current visual presentation has been determined to be acceptable, and any future revisions to site aesthetics that are not directly determined by the functionality listed above are considered beyond the scope of this project, and are subject to Team Unicorns hourly rate.

## Project Plan

The BDDPP website is not considered to be an extensive standalone application, and so the complete systems development life cycle(SDLC) will be abbreviated. The abridged form of the SDLC we will be using includes the following phases:

Phase 1: Functional Requirements, as defined by this document

Phase 2: Site design, including wireframes and mockups

Phase 3: Database Creation. All entities and their attributes have been provided.

Phase 4: Testing as defined in the testing section of this document.

Phase 5: Deliver the final product for approval

## Technical Specifications

### Web Languages

The site structure will be written in HTML5. The site's visual presentation will be configured using external CSS files. Dynamic content will be created with PHP where applicable.

### Database Applications

MySQL is required

### Hardware and Software Requirements

A professionally hosting company, with a Linux based server & bandwidth allotment appropriate for a business class enterprise.

# Functional Specifications

## Functional Requirements

### **Customer experience:**

The customer should choose their pizza toppings on the home page. Submitting this information will take them to a page where they input their information - name, address, etc. Submitting the customer information form will take them to a page where they can confirm their customer information and their order. If both are correct the customer will confirm the information. When this happens the order will be submitted to the database and the customer will be taken to a summary page. If the information in the confirmation page is not correct the customer can choose to go back to fix it.

### **Admin experience:**

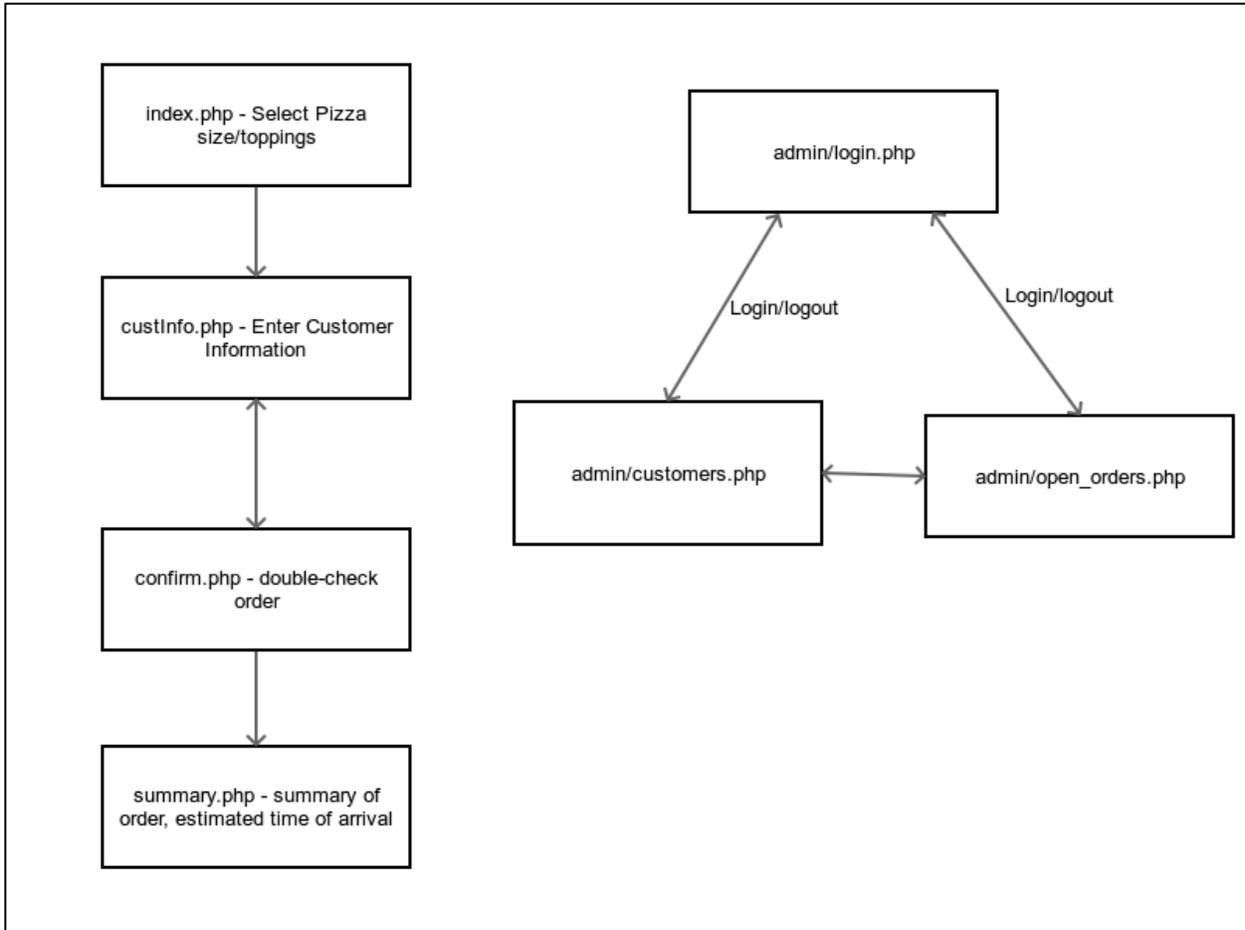
The admin page will be password-protected. Entering the correct username/password combination will take you to a page with information on the currently outstanding pizza orders. On this page the user can get rid of orders that have been completed. From there the user can go to another page that has the customer information for every customer in the database. On this page the user can manipulate customer information. When the user is done they can logout from either the customer information page or the current orders page.

## Information Requirements

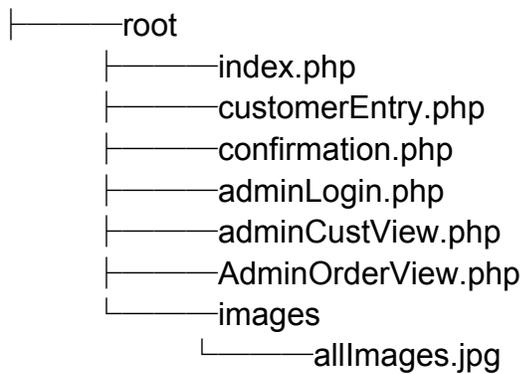
Information validation will be performed through RegEx validation where appropriate. The system will have data entry fields that support the following information:

- Pizza size
- Crust Type
- Toppings
- Customer First Name
- Customer Last Name
- Customer Email Address
- Customer Address
- Customer Apartment
- Customer City
- Customer State
- Customer Zip
- Customer Phone Number
- Admin username
- Admin password

## Site Plan

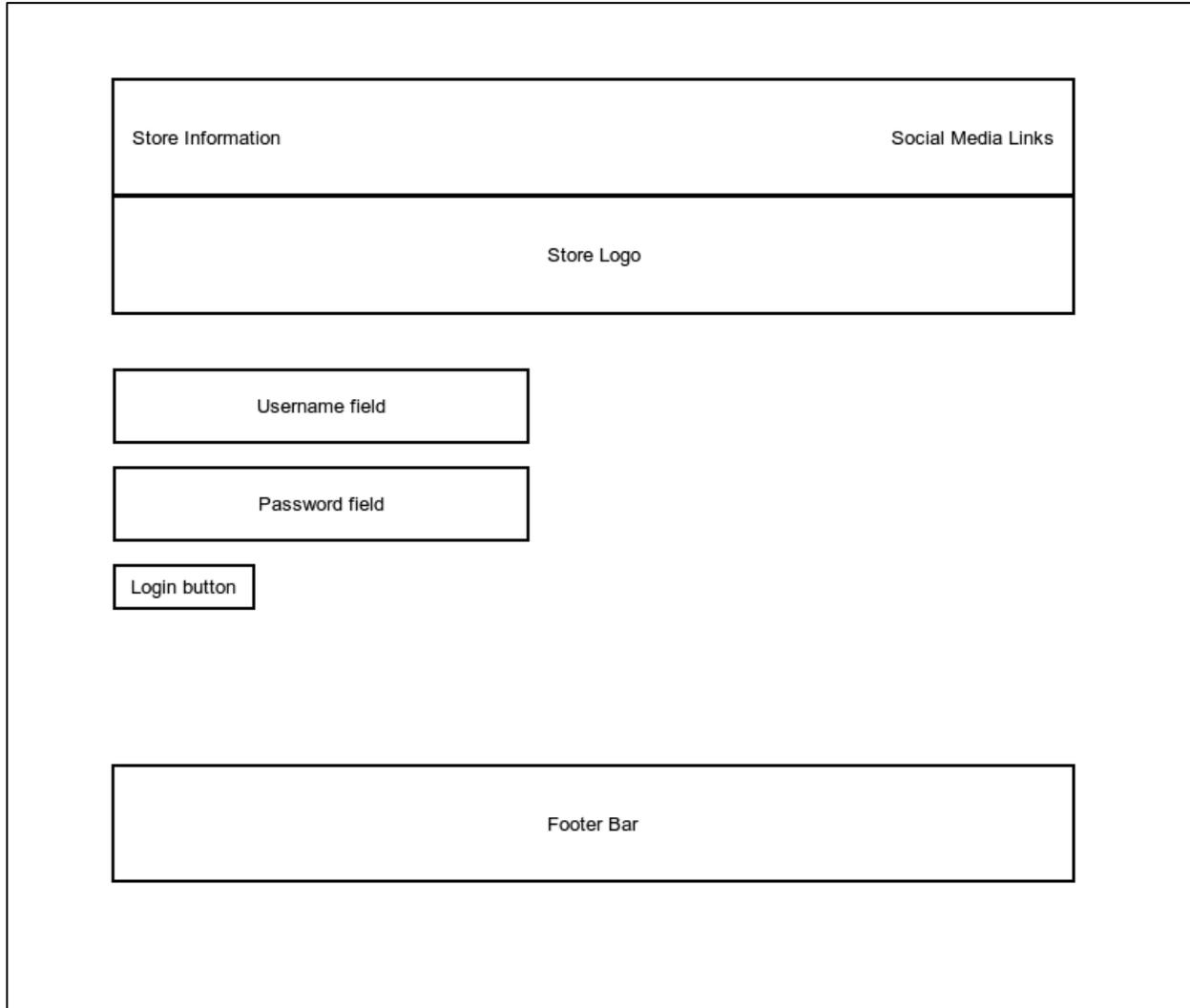


## File Structure



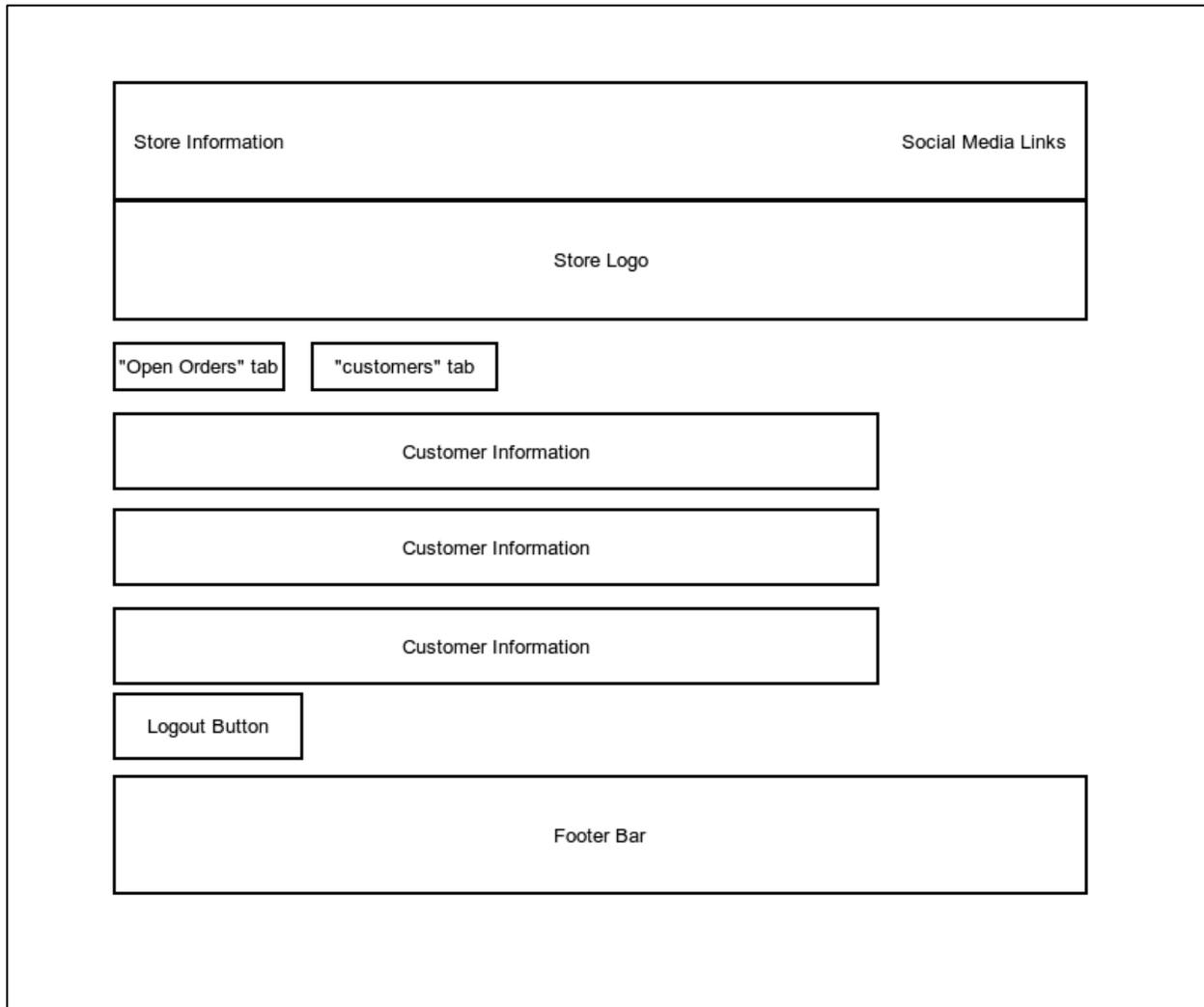
## Wire Frames

### Administrator Login

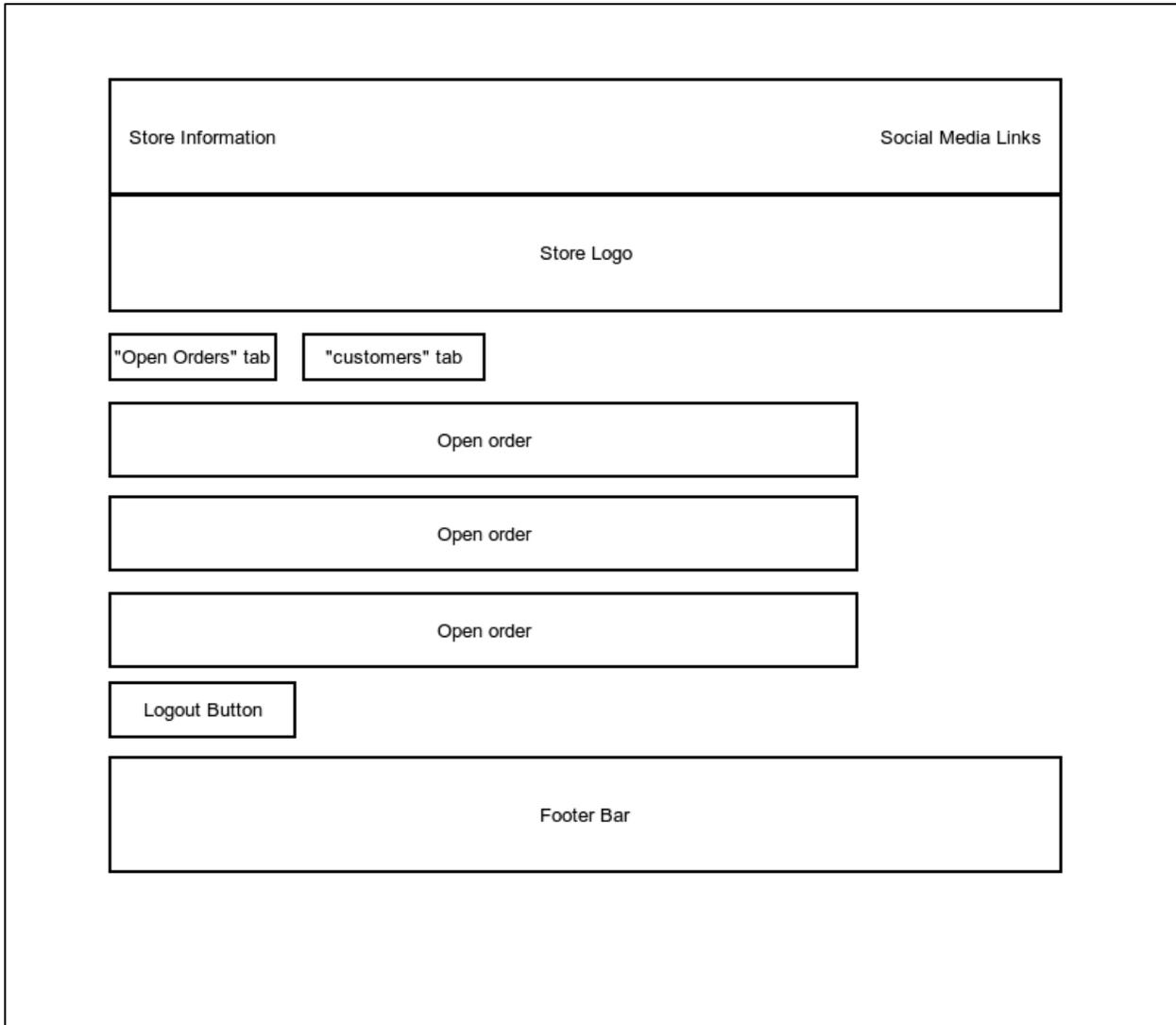


The wireframe illustrates the layout of the Administrator Login page. It is enclosed in a large rectangular border. At the top, there is a wide horizontal bar divided into two sections: 'Store Information' on the left and 'Social Media Links' on the right. Below this bar is a large rectangular area labeled 'Store Logo'. Underneath the logo area, there are three vertically stacked input fields: 'Username field', 'Password field', and 'Login button'. At the bottom of the page, there is a wide horizontal bar labeled 'Footer Bar'.

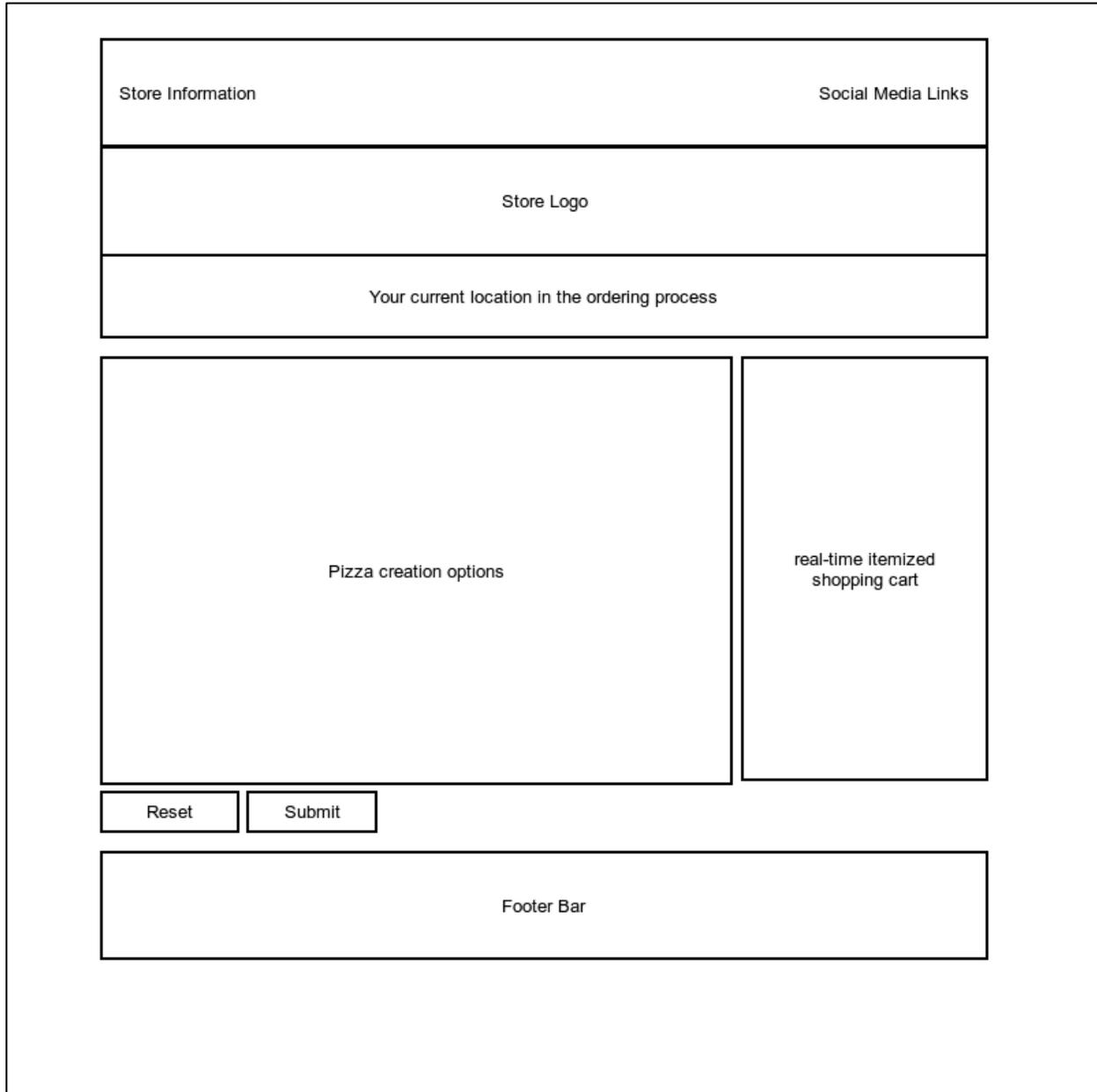
## Administrators Customer View



## Administrator Open Orders



## Customer Order Entry



## Customer Information Entry

Store Information	Social Media Links
Store Logo	
Your current location in the ordering process	
Fields to enter customer information	Itemized shopping cart
<input type="button" value="Reset"/>	<input type="button" value="Submit"/>
Footer Bar	

## Customer Order Summary

Store Information	Social Media Links
Store Logo	
Your current location in the ordering process	
Customer information	Itemized Order: items and price  Subtotal Tax Total
Message to customer:	
Footer Bar	

# Test Plan

## Testing Approach

The three testing methods we will use to test the site are; black box testing, glass box testing, and user acceptance testing. Each testing method has advantages and disadvantages that contribute to the final assurance of a quality system.

### Glass Box Testing

The Team Unicorn developers will perform all Glass box testing, because test data is derived from direct examination of the code. Both static and dynamic analysis will be used, where deemed appropriate by the Team Unicorn developers.

#### **Benefits**

Because the developers have internal knowledge of the software, they can walk through the code line by line, and reason carefully through the implementation of the code to reveal errors.

#### **Risks**

Typically this is a very expensive method of testing, because larger projects can require a large amount of labor to manually review the code. Because of the limited scope of this project, we are able to implement glass box testing with the developers already employed by Team Unicorn.

### Black Box Testing

Team Unicorn members that are not directly responsible for software development will be performing black box testing. The black box testing will focus on the functional requirements, rather than examination of the code. The use cases found in Appendix B will be used to direct the black box testing.

#### **Benefits**

Multiple modules can be tested simultaneously, so black box testing is more effective for larger units of code. Because testing is performed from a users point of view, any functional requirement that is not met can be identified.

#### **Risks**

Black box testing may have some overlap with glass box testing, so there may be some unnecessary repetition of tests. There is also the risk of back-end code that is not accessible to the user will not get tested. Combining glass box testing with black box testing greatly reduces this risk.

## User Acceptance Testing

A BDDPP representative will perform user acceptance testing on June 17<sup>th</sup> during a live system walkthrough.

### **Benefits**

The user acceptance testing ensures that all necessary features have been included in the final product.

### **Risks**

There is a risk that missing functional requirements will be identified during user acceptance testing. This risk has been greatly reduced by Team Unicorn's strict adherence to standard practices and procedures outlined in the system development lifecycle.

## Test Environment

Several contributing factors are critical to a successful testing environment.

### **Location**

Because of the limited scope of the BDDPP website, all black box and glass box testing will be conducted at the team unicorn offices, and the user acceptance testing will be performed at the BDDPP corporate headquarters. These locations have been selected to minimize disruption to BDDPP's daily operations.

### **Testing Hours**

Black box and glass box testing will be conducted during normal business hours, because the testing will be performed at the Team Unicorn offices. The user acceptance test will be scheduled at the discretion of BDDPP management.

### **How the tests will be conducted**

The black, glass box, and user acceptance testing will be conducted on Team Unicorn's hardware, on a security-encrypted network. The testing setup used by Team Unicorn will simulate a live web-hosting environment.

### **Personnel**

Team Unicorn will provide the personnel required for both the black box testing and the glass box testing. A specified BDDPP employee must perform the final acceptance testing. The designated employee must be authorized to approve acceptance of the final product.

# Appendices

## Appendices Table of Contents

<b>Appendix A: Project Schedule</b> .....	16
<b>Appendix B: Use Case Scenarios</b> .....	17
UC001 Order Pizza.....	17
UC002 Order Summary .....	18
UC003 Confirm Order.....	19
UC004 Send Customer Email .....	19
UC005 Admin Login .....	20
<b>Appendix C: Site Map</b> .....	21
<b>Wire Frame: Pizza Entry Page</b> .....	22
<b>Wire Frame: Customer Information Page</b> .....	23
<b>Wire Frame: Confirmation Page</b> .....	24
<b>File Structure:</b> .....	25
<b>Approvals</b> .....	26
<b>Change Log</b> .....	27

## Appendix A: Project Schedule

Deliverable	Due Date	Person Responsible
Design Document Sections: Cover page, Toc, Scope, Project plan, technical specs, site map, wire frame, approvals, change log.	5/20/14	Ryan Neff
Design Document Sections: Functional Specifications	5/20/14	Josh Manion
Design Document Sections: Project Schedule, Use Case Scenarios.	5/20/14	Brendan Jones
Site Mockup including Administration Site	5/27/14	Team Unicorn
Database Creation and Connectivity	6/3/14	Team Unicorn
Testing	6/10/14	Team Unicorn
Final Project Delivery	6/17/14	Team Unicorn

## Appendix B: Use Case Scenarios

### UC001 Order Pizza

<b>Use Case Name:</b>	<b>Order Pizza</b>	ID: UC001
<b>Primary Actor:</b>	Customer	
<b>Brief Description:</b>	This use case is where the customer visit's the site and begins ordering a pizza by selecting the size or a pre-made option and ends with submitting the order details to continue the check out process.	
<b>Trigger:</b>	Customer begins this process by selecting pizza options such as the size or a pre-made option.	
<b>Related Use Cases:</b>	Order Summary, Confirm Order, Send Customer an Admin Email and Admin Login.	
<b>Normal flow of events:</b>	<ol style="list-style-type: none"> <li>1) This case begins when the customer begins to order a pizza by deciding on a build your own option or a pre-made option.</li> <li>2) The customer will then either submit his order and go to the next page/part of the order process or will pick the pizza size options desired.</li> <li>3) The customer will then pick the kind of crust desired for their pizza.</li> <li>4) The customer will then pick the toppings desired for the pizza.</li> <li>5) The customer will then submit the order choices and continue with the order process.</li> <li>6)</li> </ol>	
<b>Exception(s):</b>	<ol style="list-style-type: none"> <li>1) If any required information is missing or invalid, an exception message is created instructing the customer to resolve the issue.</li> </ol>	
<b>Pre-condition(s):</b>	The customer must be on the website.	
<b>Post-conditions(s)</b>	Customer will continue to the Order Summary screen to continue the order.	
<b>Information Requirements:</b>	Customer will need to know which pizza options are desired.	
<b>Assumptions:</b>	The customer is interested in buying a pizza.	
<b>Business Rules:</b>	Data validation i.e. one pizza at a time, one kind of pizza at a time.	

## UC002 Order Summary

<b>Use Case Name:</b>	<b>Order Summary</b>	ID: UC002
Primary Actor:	Customer	
Brief Description:	This case describes the step in the pizza ordering process called the Order Summary. The customer has previously entered their order details and are prompted to inter their personal delivery information. This case ends with the customer submitting their information to the website.	
Trigger:	Submitting the customer information.	
Related Use Cases:	Send Customer an Admin Email and Admin Login.	
Normal flow of events:	The customer will be prompted	
Exception(s):	1) If any required information is missing or invalid, an exception message is created instructing the customer to resolve the issue.	
Pre-condition(s):	The customer has gone through the order pizza process on the first page of the website.	
Post-conditions(s)	The customer submits their information.	
Information Requirements:	Customer name, address, and billing info.	
Assumptions:	The customer has an address.	
Business Rules:	Data validation i.e. one pizza at a time, one kind of pizza at a time.	

### UC003 Confirm Order

<b>Use Case Name:</b>	<b>Confirm order</b>	ID: UC003
Primary Actor:	customer	
Brief Description:	This use case describes the process of the customer confirming the order.	
Trigger:	An order has to be submitted.	
Related Use Cases:	Order Pizza, Order Summary	
Normal flow of events:	This use case happens after the customer has ordered a pizza. The customer is shown a screen where they can confirm their order.	
Exception(s):	none	
Precondition(s):	Customer has ordered a pizza from the website.	
Postconditions(s)	The customer confirms that their order is correct.	
Information Requirements:	Customer double checks the details of the order being placed.	
Assumptions:	The customer will correct any mistakes made during the order process.	
Business Rules:	Data validation i.e. one pizza at a time, one kind of pizza at a time.	

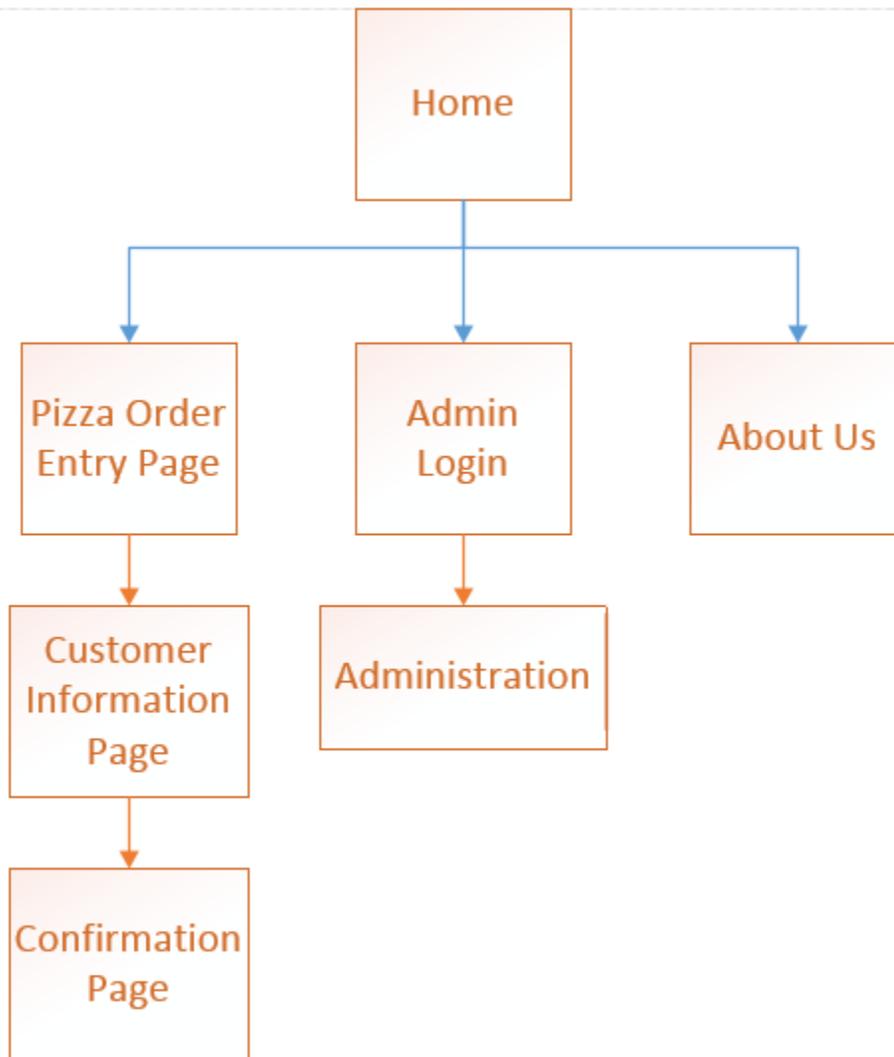
### UC004 Send Customer Email

<b>Use Case Name:</b>	<b>Send Customer An Admin Email</b>	ID: UC004
Primary Actor:	Customer	
Brief Description:	This use case scenario describes the next step of the order process, the customer receives an admin email with login information.	
Trigger:	Customer submits an order.	
Related Use Cases:	Order Pizza, Order Summary	
Normal flow of events:	After the customer submits their order they will be sent an admin email with details about their admin account.	
Exception(s):	Email address provided by the customer is incorrect.	
Precondition(s):	An order has been submitted by the customer.	
Postconditions(s)	Customer logs in to the admin account and can make orders using the admin account without entering their information twice.	
Information Requirements:	Customers email address.	
Assumptions:	The customer provided the correct email address.	
Business Rules:	The customer must not share their admin account.	

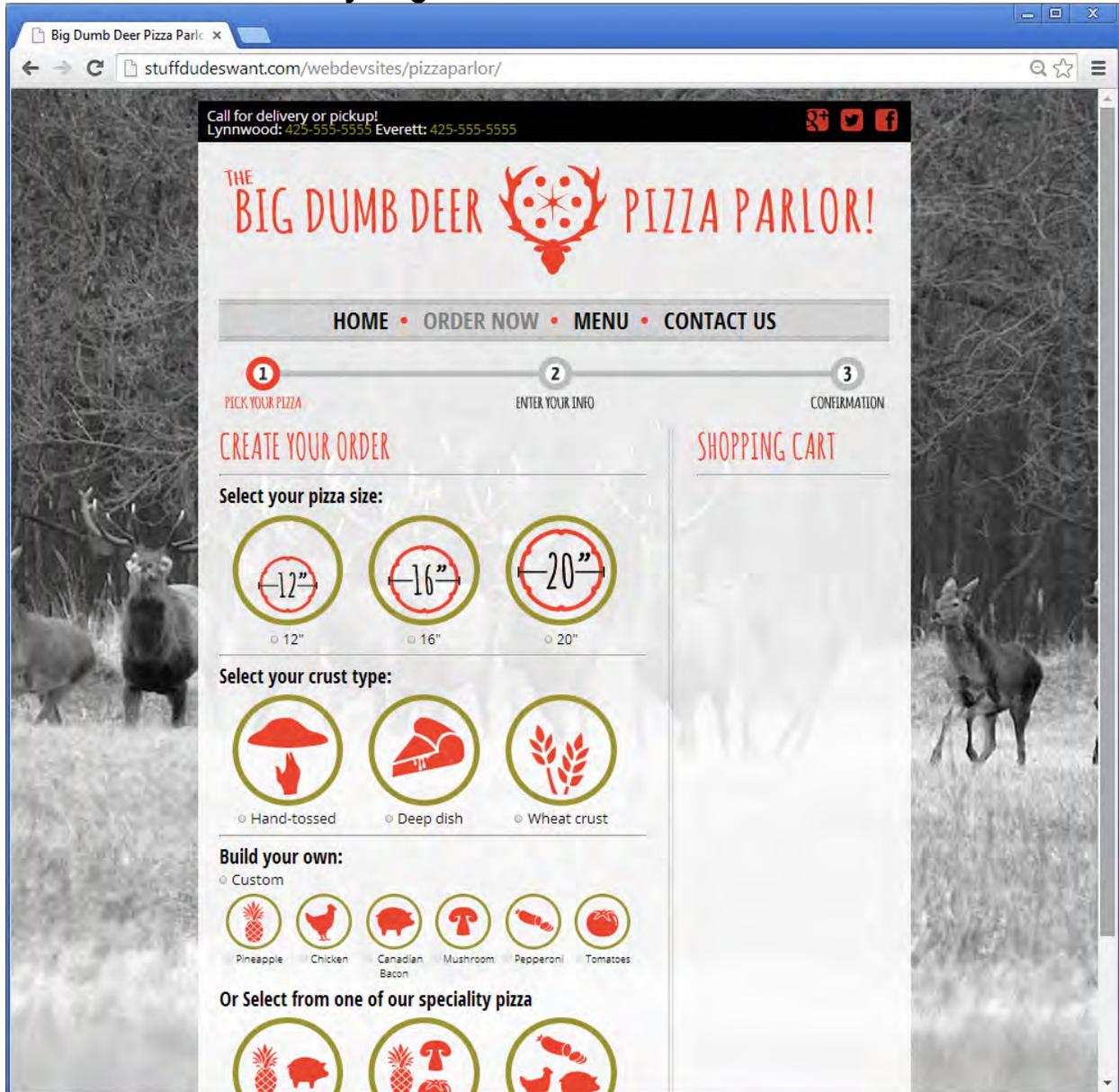
## UC005 Admin Login

<b>Use Case Name:</b>	<b>Admin Login</b>	ID: UC005
Primary Actor:	Customer	
Brief Description:	This use case describes the admin login process that is to be used for future orders.	
Trigger:	Customer enters their admin account information.	
Related Use Cases:	Order Pizza, Order Summary	
Normal flow of events:	Customer can order after logging in to their admin account with a few clicks, without entering their payment and delivery information again.	
Exception(s):	none	
Precondition(s):	Admin email with login details has been sent to the customer.	
Postconditions(s)	Order is delivered.	
Information Requirements:	Admin information that was sent to customer.	
Assumptions:	Customer will order again.	
Business Rules:	None.	

## Appendix C: Site Map

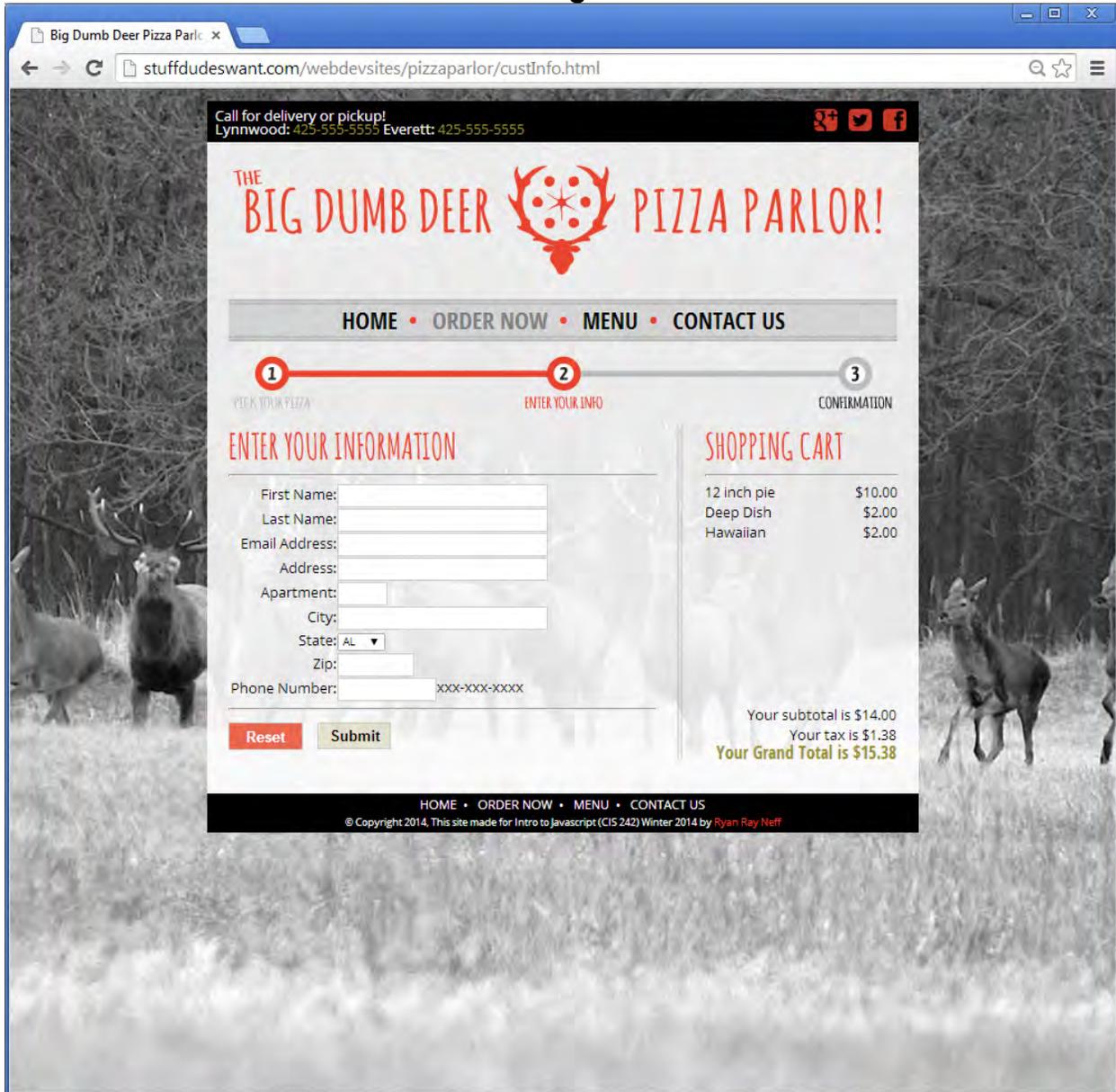


## Wire Frame: Pizza Entry Page



The wireframe shows a web browser window with the URL `stuffdudeswant.com/webdevsites/pizzaparlor/`. The page features a dark header with contact information: "Call for delivery or pickup! Lynnwood: 425-555-5555 Everett: 425-555-5555" and social media icons for YouTube, Twitter, and Facebook. The main heading is "THE BIG DUMB DEER PIZZA PARLOR!" with a logo of a deer head. A navigation bar includes "HOME • ORDER NOW • MENU • CONTACT US". The page is divided into three numbered steps: 1. PICK YOUR PIZZA, 2. ENTER YOUR INFO, and 3. CONFIRMATION. The "CREATE YOUR ORDER" section includes "Select your pizza size:" with options for 12", 16", and 20" (the 12" option is selected); "Select your crust type:" with options for Hand-tossed, Deep dish, and Wheat crust (the Hand-tossed option is selected); "Build your own:" with a "Custom" option and icons for Pineapple, Chicken, Canadian Bacon, Mushroom, Pepperoni, and Tomatoes; and "Or Select from one of our speciality pizza" with three specialty pizza icons. A "SHOPPING CART" section is visible on the right side of the page.

## Wire Frame: Customer Information Page



Call for delivery or pickup!  
Lynnwood: 425-555-5555 Everett: 425-555-5555

THE BIG DUMB DEER PIZZA PARLOR!

HOME • ORDER NOW • MENU • CONTACT US

1 PICK YOUR PIZZA 2 ENTER YOUR INFO 3 CONFIRMATION

ENTER YOUR INFORMATION

First Name:

Last Name:

Email Address:

Address:

Apartment:

City:

State: AL

Zip:

Phone Number:  XXX-XXX-XXXX

Reset Submit

SHOPPING CART

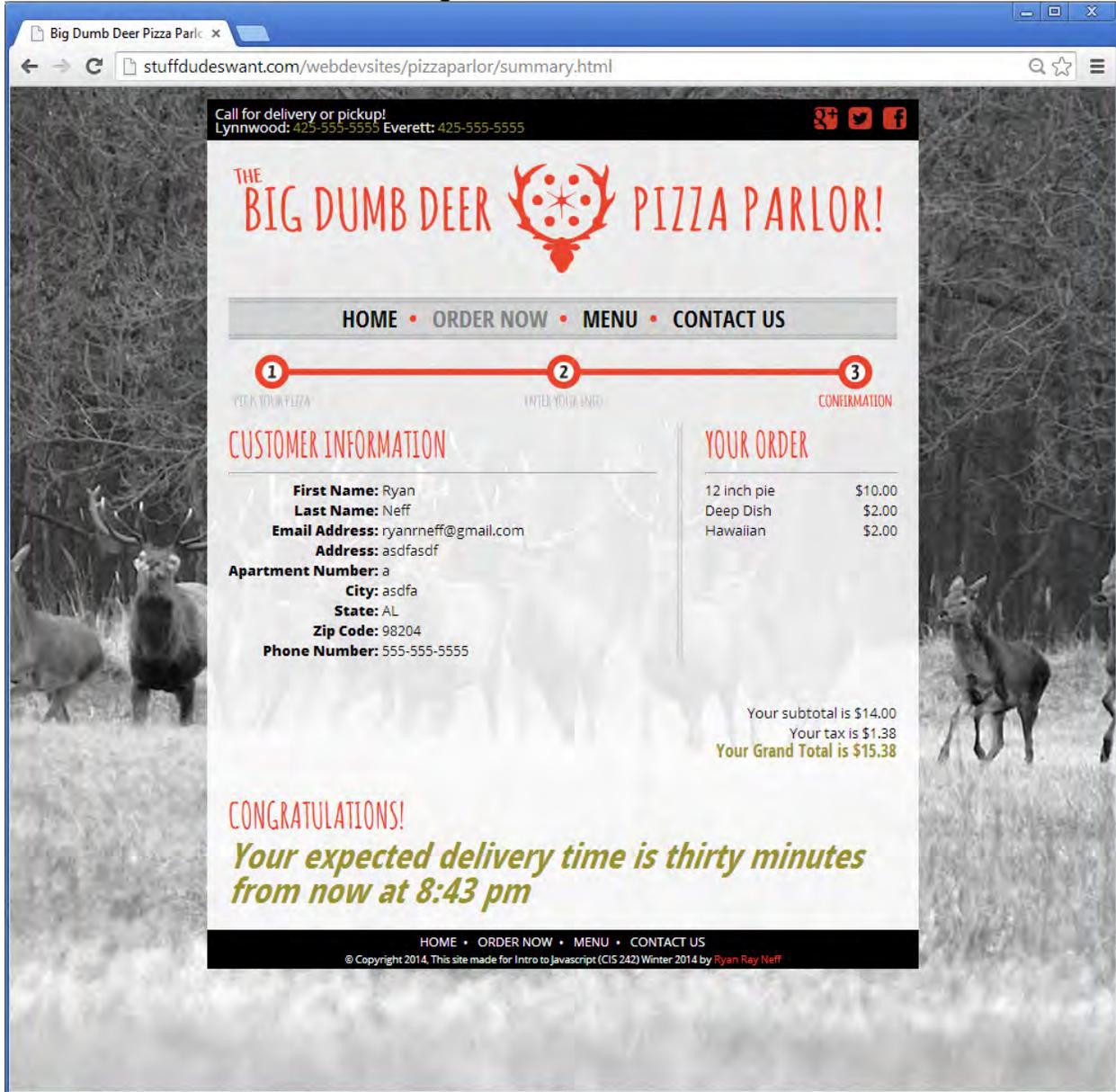
12 inch pie	\$10.00
Deep Dish	\$2.00
Hawaiian	\$2.00

Your subtotal is \$14.00  
Your tax is \$1.38  
**Your Grand Total is \$15.38**

HOME • ORDER NOW • MENU • CONTACT US

© Copyright 2014. This site made for Intro to Javascript (CIS 242) Winter 2014 by Ryan Ray Neff

## Wire Frame: Confirmation Page



Call for delivery or pickup!  
Lynnwood: 425-555-5555 Everett: 425-555-5555

THE BIG DUMB DEER PIZZA PARLOR!

HOME • ORDER NOW • MENU • CONTACT US

1 PICK YOUR PIZZA 2 ENTER YOUR INFO 3 CONFIRMATION

**CUSTOMER INFORMATION**

**First Name:** Ryan  
**Last Name:** Neff  
**Email Address:** ryanneff@gmail.com  
**Address:** asdfasdf  
**Apartment Number:** a  
**City:** asdfa  
**State:** AL  
**Zip Code:** 98204  
**Phone Number:** 555-555-5555

**YOUR ORDER**

12 inch pie	\$10.00
Deep Dish	\$2.00
Hawaiian	\$2.00

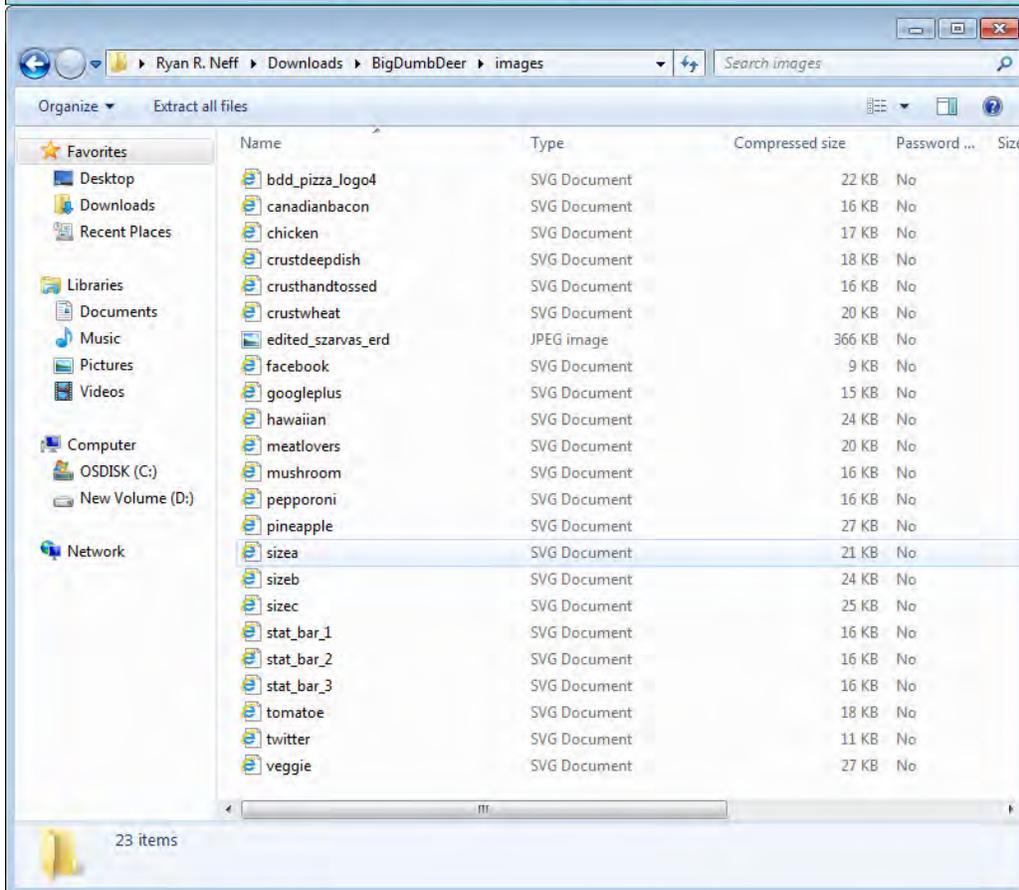
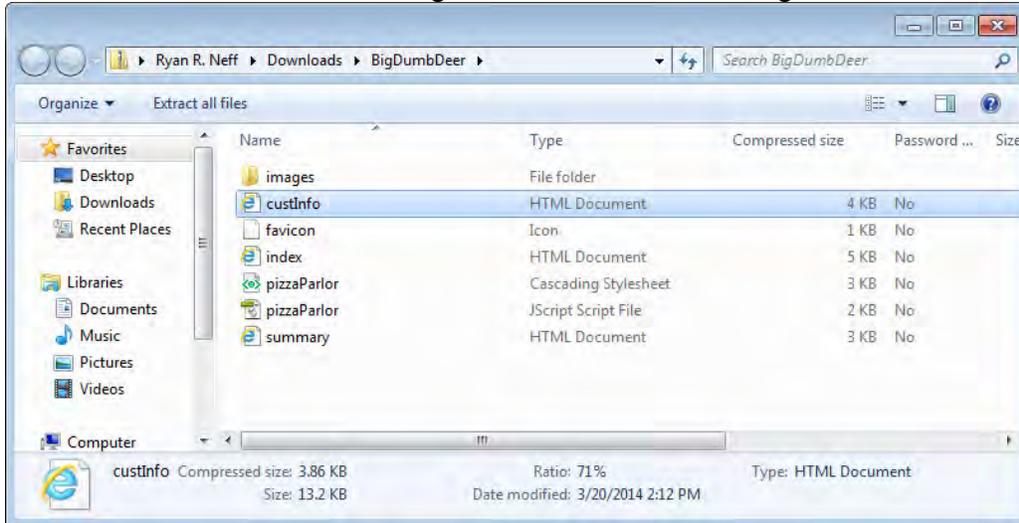
Your subtotal is \$14.00  
Your tax is \$1.38  
**Your Grand Total is \$15.38**

**CONGRATULATIONS!**  
*Your expected delivery time is thirty minutes from now at 8:43 pm*

HOME • ORDER NOW • MENU • CONTACT US  
© Copyright 2014, This site made for Intro to Javascript (CIS 242) Winter 2014 by Ryan Ray Neff

## File Structure:

The php pages for each of the 4 pages will reside in the root folder, along with the style sheet and the favicon. All images will reside in the images folder.



# Approvals

This document constitutes an agreement between BDDPP and Team Unicorn. This document clearly all of the specifications and design plans for the new website currently being designed and built for the BDDPP by Team Unicorn. Team Unicorn will preform the work mentioned in the attached document. BDDPP will provide one half of the dollar amount agreed upon before work begins, and the remaining payment is due within. 30 days of project acceptance. Project acceptance will be determined during the final acceptance test by a designated BDDPP employee.

\_\_\_\_\_  
John Smith  
BDDPP CIO

\_\_\_\_\_ Date \_\_\_\_\_

\_\_\_\_\_  
Brendan Jones  
Team Unicorn CEO

\_\_\_\_\_ Date \_\_\_\_\_

\_\_\_\_\_  
Josh Manion  
Team Unicorn Lead Developer

\_\_\_\_\_ Date \_\_\_\_\_

\_\_\_\_\_  
Ryan Neff  
Team Unicorn Designer

\_\_\_\_\_ Date \_\_\_\_\_

## Change Log

<b>Date</b>	<b>Name</b>	<b>Change Description</b>	<b>Version</b>