|  |  |
| --- | --- |
| Team Members | Email |
| Aolei Li | [z1756249@students.niu.edu](mailto:z1756249@students.niu.edu) |
| Phani Teja Nunna | [Z1784111@students.niu.edu](mailto:Z1784111@students.niu.edu) |
| Saranya Garimella | [Z1784115@students.niu.edu](mailto:Z1785115@students.niu.edu) |
| Sreeram Sunkara | [z1784112@students.niu.edu](mailto:z1784112@students.niu.edu) |

Presented to: Akshay Bhagwatwar

Prepared by: “Hands-On Team”

Date: 03/06/2016

**Project Submission**

**Team: “Hands On”**

**Lab 7**

Northern Illinois University

Department of Operations Management Information Systems

OMIS 675 – Internet Computing Applications

Table of Contents

Functional Requirements 3

Non Functional Requirements 4

Use Case Diagram 5

Wire Frames 6

Pizza Customization Screen 6

Customer Details and Consult Order 7

Development Activities: 7

Code Snippet: 8

Pizza Order Screen: 8

Aspx: 8

Cs: 10

Screenshot: 11

Pizza Confirmation Screen: 11

Aspx: 11

Cs: 12

Screenshot: 13

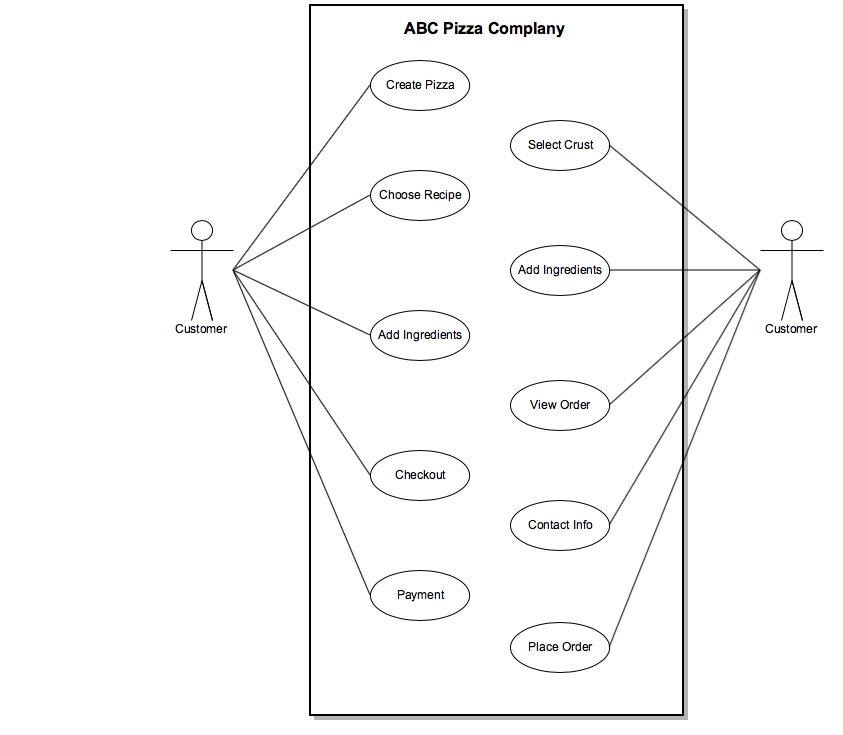
# Functional Requirements

|  |  |  |  |
| --- | --- | --- | --- |
| ID | Requirement | MVP | Example |
| FR-001 | Customer has to choose the near by location, so that the respective pizza store in that location treat the order. | Yes | Dekalb, Napperville |
| FR-002 | Customer has to start the pizza customization by selecting the Pizza size | Yes | Small, Medium |
| FR-003 | After selecting the pizza size, Customer need to select the bread crust that he want for the pizza. | Yes | Flat, Thin |
| FR-004 | Customer then select the kind of meet that he want on the top of Pizza | Yes | Chicken, Bacon |
| FR-005 | Customer then select various veggies based on the collection | Yes | Tomato, Mushroom |
| FR-006 | Customer need to select the cheese and sauce that he want to be added on the pizza | Yes | Cheddar, Garlic parmesan |
| FR-007 | Customer can view all the pizza order | Yes |  |
| FR-008 | Customer can add more than one pizza for delivery and can customize all the Pizzas | Yes |  |
| FR-009 | Customer need to confirm the order to proceed for making | Yes |  |
| FR-010 | Customer need to enter the delivery address | Yes |  |

# Non Functional Requirements

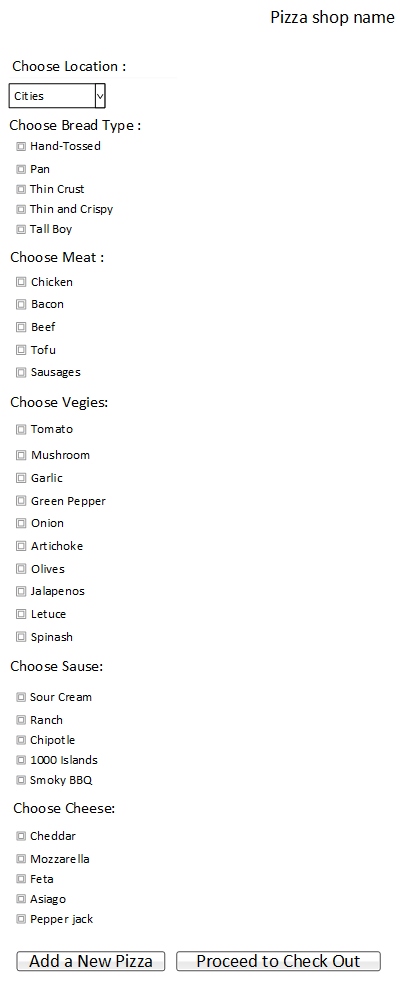
|  |  |  |
| --- | --- | --- |
| ID | Requirement | MVP |
| NFR-001 | Web-Page rendering should not take more than 5 seconds. | Yes |
| NFR-002 | Cookies need to be cached and user need to access offline data to view various options available. | Yes |
| NFR-003 | Topping list need to be automatically updated to the new items getting added in the kitchen. | Yes |
| NFR-004 | Every functionality development needs to be having 50% of comment lines to have the knowledge retention for long. | Yes |
| NFR-005 | Web page development needs to be responsive based on the screen size. | Yes |
| NFR-006 | The web page should have the support to allow 100 users to place order online at once. | Yes |
| NFR-007 | Any failure in the service or the database, need to be recovered in 2 hours of time. | Yes |
| NFR-008 | Web page should not be accessed on an unsecured Wi-Fi network. (Accessibility only in secure network) | Yes |
| NFR-009 | Content on the web-page need to be approved by the Legal team before publishing. | Yes |
| NFR-010 | Food items listed in the web-page need to be items approved from FSMA (FDA food safety modernization Act). | Yes |

# Use Case Diagram

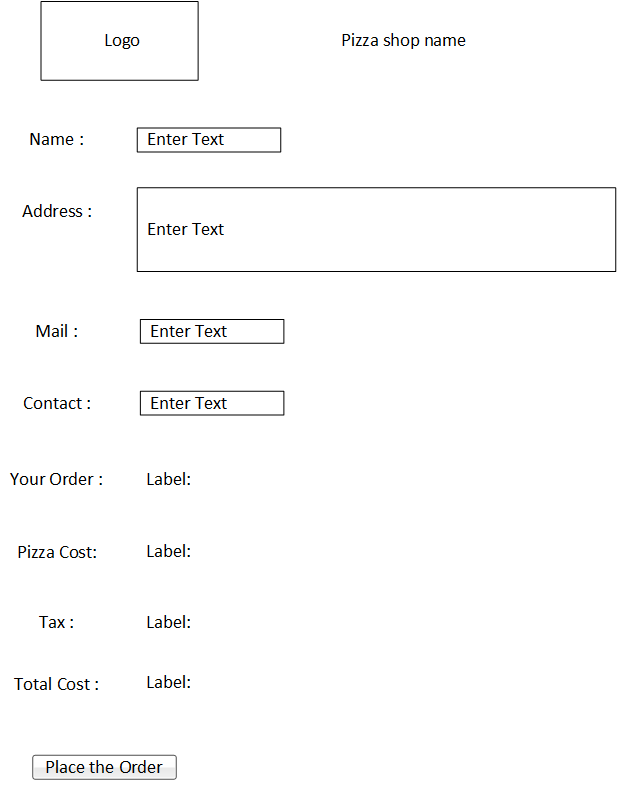


# Wire Frames

## Pizza Customization Screen



## Customer Details and Consult Order



# Development Activities:

* Connect to a database for the Customer to select his/her location
* Customer to create his/her pizza based on the vast options on bread, veggies, meet, cheese.,
* Customer to fill his delivery address
* Customer to consult his pizza order and the cost of the order

# Code Snippet:

## Pizza Order Screen:

### Aspx:

<%@ Page Title="Home Page" Language="C#" MasterPageFile="~/Site.Master" AutoEventWireup="true" CodeFile="Default.aspx.cs" Inherits="\_Default" %>

<asp:Content ID="BodyContent" ContentPlaceHolderID="MainContent" runat="server">

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

<p class="text-center" style="font-size: x-large; text-decoration: underline;">

<strong><em>TASTY PIZZA</em></strong>

</p>

<br />

<p>

<strong><em>Choose Location :

<asp:SqlDataSource ID="SqlDataSource1" runat="server" ConnectionString="<%$ ConnectionStrings:z1784115ConnectionString %>" SelectCommand="SELECT [Campus] FROM [Courses]"></asp:SqlDataSource>

</em></strong>

</p>

<p>

<asp:DropDownList ID="DropDownList1" runat="server" DataSourceID="SqlDataSource1" DataTextField="Campus" DataValueField="Campus">

</asp:DropDownList>

</p>

<p>

<strong>Choose Pizza Size:</strong></p>

<p>

<asp:DropDownList ID="DropDownList2" runat="server">

<asp:ListItem>Small</asp:ListItem>

<asp:ListItem>Medium</asp:ListItem>

<asp:ListItem>Large</asp:ListItem>

</asp:DropDownList>

</p>

<p>

<strong><em>Choose Bread Type :

</em></strong>

</p>

<p>

<asp:CheckBoxList ID="CheckBoxList2" runat="server" OnSelectedIndexChanged="CheckBoxList2\_SelectedIndexChanged1" >

<asp:ListItem>Hand-Tossed</asp:ListItem>

<asp:ListItem>Pan</asp:ListItem>

<asp:ListItem>Thin Crust</asp:ListItem>

<asp:ListItem>Thin and Crispy</asp:ListItem>

<asp:ListItem>Tall Boy</asp:ListItem>

</asp:CheckBoxList>

<p>

<strong><em>Choose Meet :</em></strong>

</p>

<p>

<asp:CheckBoxList ID="CheckBoxList1" runat="server" OnSelectedIndexChanged="CheckBoxList1\_SelectedIndexChanged1" >

<asp:ListItem>Chicken</asp:ListItem>

<asp:ListItem>Bacon</asp:ListItem>

<asp:ListItem>Beef</asp:ListItem>

<asp:ListItem>Tofu</asp:ListItem>

<asp:ListItem>Sausages</asp:ListItem>

</asp:CheckBoxList>

</p>

<p><strong><em>Choose Vegies :</em></strong></p>

<p>

<asp:CheckBoxList ID="CheckBoxList3" runat="server" OnSelectedIndexChanged="CheckBoxList3\_SelectedIndexChanged1" >

<asp:ListItem>Tomato</asp:ListItem>

<asp:ListItem>Mushroom</asp:ListItem>

<asp:ListItem>Garlic</asp:ListItem>

<asp:ListItem>Green Pepper</asp:ListItem>

<asp:ListItem>Onion</asp:ListItem>

<asp:ListItem>Artichoke</asp:ListItem>

<asp:ListItem>Olives</asp:ListItem>

<asp:ListItem>Jalapenos</asp:ListItem>

<asp:ListItem>Letuce</asp:ListItem>

<asp:ListItem>Spinach</asp:ListItem>

</asp:CheckBoxList>

</p>

<p>

<strong><em>Choose Sause :&nbsp;&nbsp;</em></strong>

</p>

<p>

<asp:CheckBoxList ID="CheckBoxList4" runat="server" OnSelectedIndexChanged="CheckBoxList4\_SelectedIndexChanged1" >

<asp:ListItem>Sour Cream</asp:ListItem>

<asp:ListItem>Ranch</asp:ListItem>

<asp:ListItem>Chipotle</asp:ListItem>

<asp:ListItem>1000 Islands</asp:ListItem>

<asp:ListItem>Smoky BBQ</asp:ListItem>

</asp:CheckBoxList>

</p>

<p>

Choose Cheese :

</p>

<p>

<asp:CheckBoxList ID="CheckBoxList5" runat="server" OnSelectedIndexChanged="CheckBoxList5\_SelectedIndexChanged1" >

<asp:ListItem>Cheddar</asp:ListItem>

<asp:ListItem>Mozzarella</asp:ListItem>

<asp:ListItem>Feta</asp:ListItem>

<asp:ListItem>Asiago</asp:ListItem>

<asp:ListItem>Pepper jack</asp:ListItem>

</asp:CheckBoxList>

</p>

<p>

<asp:Button ID="Button1" runat="server" Text="Add a New Pizza" OnClick="Button1\_Click1" PostBackUrl="~/Default.aspx"/>

<asp:Button ID="Button2" runat="server" Text="Proceed to Check Out" OnClick="Button2\_Click1" />

</p>

</asp:Content>

### Cs:

using System;

using System.Collections.Generic;

using System.Linq;

using System.Web;

using System.Web.UI;

using System.Web.UI.WebControls;

public partial class \_Default : Page

{

public static String pizzaorder = "Pizza: ";

public static int numberOfPizza = 1;

protected void Page\_Load(object sender, EventArgs e)

{

}

protected void CheckBoxList2\_SelectedIndexChanged1(object sender, EventArgs e)

{

pizzaorder = pizzaorder + " Bread Type : " + CheckBoxList2.SelectedValue;

}

protected void CheckBoxList1\_SelectedIndexChanged1(object sender, EventArgs e)

{

pizzaorder= pizzaorder + " Meet : " + CheckBoxList1.SelectedValue;

}

protected void CheckBoxList3\_SelectedIndexChanged1(object sender, EventArgs e)

{

pizzaorder = pizzaorder + " Veggies : " + CheckBoxList3.SelectedValue;

}

protected void CheckBoxList4\_SelectedIndexChanged1(object sender, EventArgs e)

{

pizzaorder = pizzaorder + " Sause : " + CheckBoxList4.SelectedValue;

}

protected void CheckBoxList5\_SelectedIndexChanged1(object sender, EventArgs e)

{

pizzaorder = pizzaorder + " Cheese : " + CheckBoxList5.SelectedValue;

}

protected void Button1\_Click1(object sender, EventArgs e)

{

pizzaorder = pizzaorder + "\n Pizza:";

numberOfPizza = numberOfPizza + 1;

}

protected void Button2\_Click1(object sender, EventArgs e)

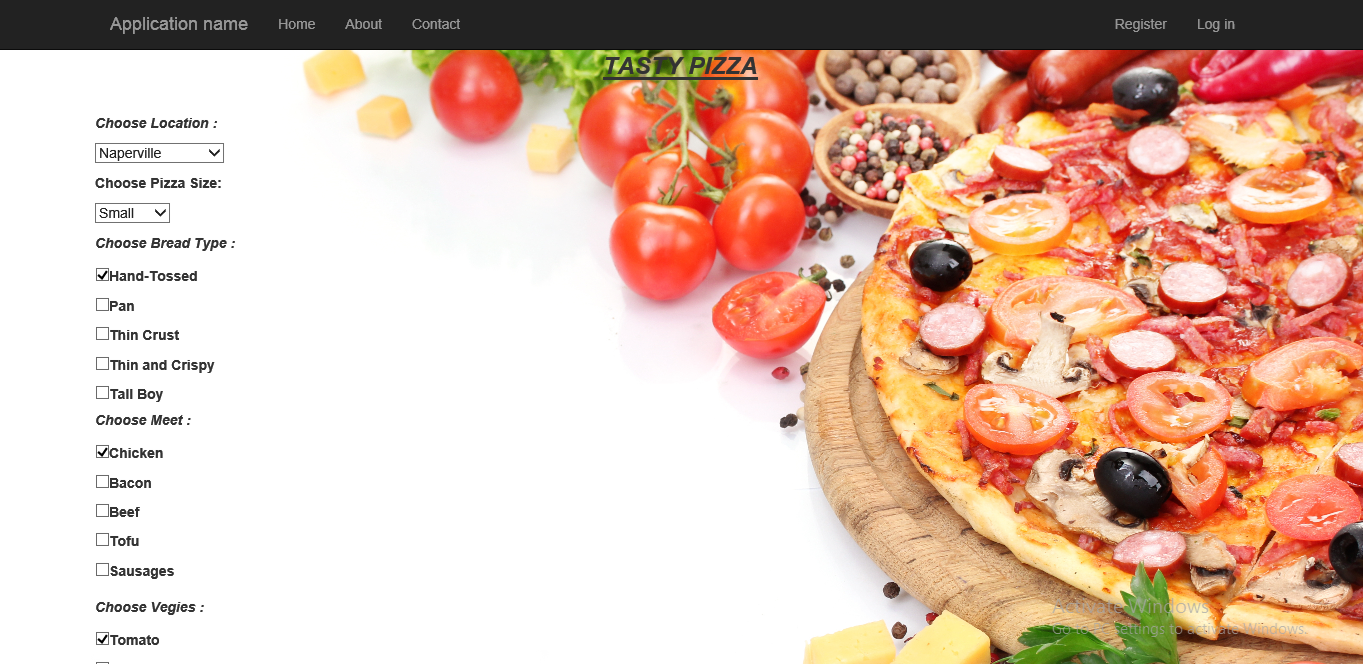
{

Response.Redirect("Default2.aspx?Name1=" + pizzaorder + "&Name2=" + numberOfPizza);

}

}

### Screenshot:

****

## Pizza Confirmation Screen:

### Aspx:

<%@ Page Title="" Language="C#" MasterPageFile="~/Site.master" AutoEventWireup="true" CodeFile="Default2.aspx.cs" Inherits="Default2" %>

<asp:Content ID="Content1" ContentPlaceHolderID="MainContent" Runat="Server">

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

<p>

<br />

Name :<asp:TextBox ID="TextBox1" runat="server"></asp:TextBox>

<asp:RequiredFieldValidator ID="RequiredFieldValidator1" runat="server" ControlToValidate="TextBox1" ErrorMessage="Name is required" Font-Bold="True" ForeColor="Red"></asp:RequiredFieldValidator>

</p>

<p>

Address :

<asp:TextBox ID="TextBox2" runat="server" Height="84px" TextMode="MultiLine" Width="295px"></asp:TextBox>

<asp:RequiredFieldValidator ID="RequiredFieldValidator2" runat="server" ControlToValidate="TextBox2" ErrorMessage="Address is required" Font-Bold="True" ForeColor="Red"></asp:RequiredFieldValidator>

</p>

<p>

Mail :<asp:TextBox ID="TextBox3" runat="server"></asp:TextBox>

<asp:RequiredFieldValidator ID="RequiredFieldValidator3" runat="server" ControlToValidate="TextBox3" ErrorMessage="Mail is required" Font-Bold="True" ForeColor="Red"></asp:RequiredFieldValidator>

<asp:RegularExpressionValidator ID="RegularExpressionValidator1" runat="server" ControlToValidate="TextBox3" ErrorMessage="Incorrect mail address" Font-Bold="True" ForeColor="Red" ValidationExpression="\w+([-+.']\w+)\*@\w+([-.]\w+)\*\.\w+([-.]\w+)\*"></asp:RegularExpressionValidator>

</p>

<p>

Contact :<asp:TextBox ID="TextBox4" runat="server"></asp:TextBox>

<asp:RequiredFieldValidator ID="RequiredFieldValidator4" runat="server" ControlToValidate="TextBox4" ErrorMessage="Contact is required" Font-Bold="True" ForeColor="Red"></asp:RequiredFieldValidator>

</p>

<p>

Your Order :

<asp:Label ID="Label2" runat="server"></asp:Label>

</p>

<p>

Pizza Cost :&nbsp;

<asp:Label ID="Label1" runat="server"></asp:Label>

</p>

<p>

Tax :

<asp:Label ID="Label3" runat="server"></asp:Label>

</p>

<p>

Total Cost :

<asp:Label ID="Label4" runat="server"></asp:Label>

</p>

<p>

<asp:Button ID="Button1" runat="server" Text="Place the Order" PostBackUrl="~/Default.aspx" />

</p>

</asp:Content>

### Cs:

using System;

using System.Collections.Generic;

using System.Linq;

using System.Web;

using System.Web.UI;

using System.Web.UI.WebControls;

public partial class Default2 : System.Web.UI.Page

{

protected void Page\_Load(object sender, EventArgs e)

{

if (Request.QueryString["Name1"] != null && Request.QueryString["Name2"] != null)

{

Label2.Text = Request.QueryString["Name1"];

Label1.Text = Convert.ToString(Convert.ToInt32(10) \* Convert.ToInt32(Request.QueryString["Name2"]));

Label3.Text = Convert.ToString("12");

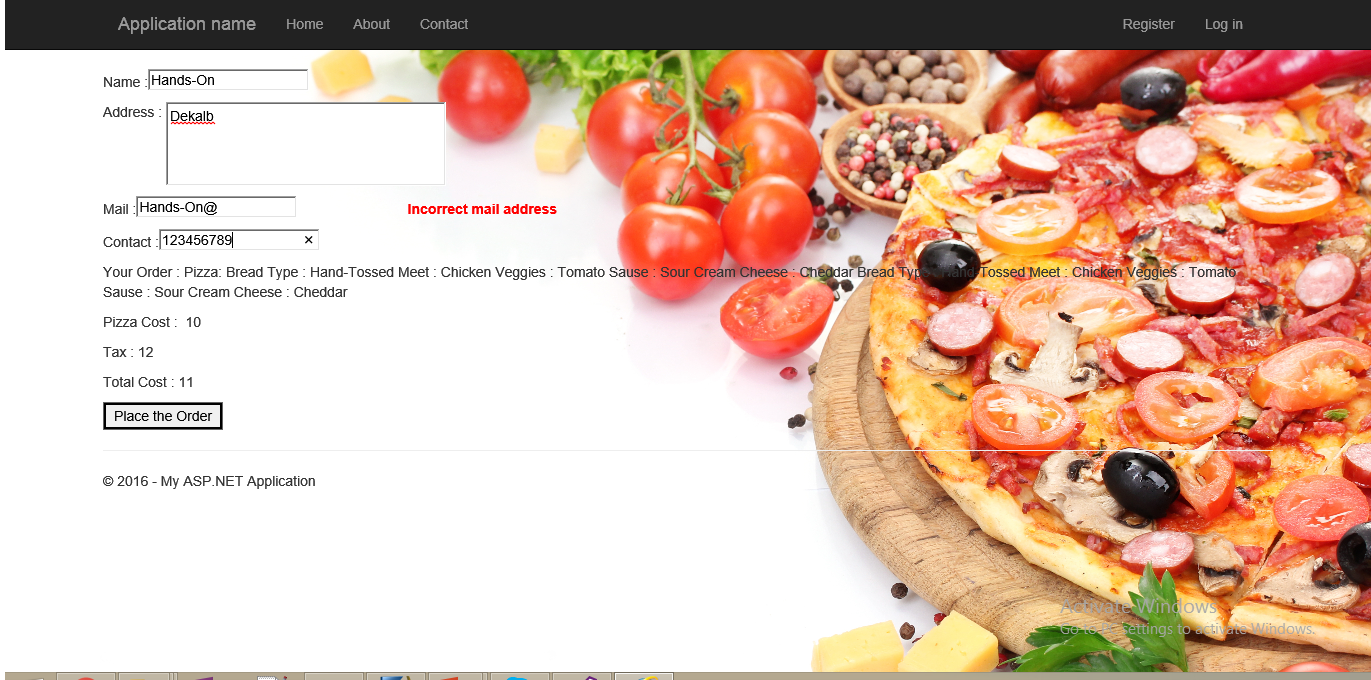
Label4.Text = Convert.ToString(Convert.ToInt32(Label1.Text) + (Convert.ToInt32(Label1.Text) \* Convert.ToInt32(Label3.Text)/100));

}

}

}

### Screenshot:

****