2008

OFFICE FOODY: Conceptual Restaurant Website Development

Chih-Wei Hsu
OFFICE FOODY: Conceptual Restaurant Website Development
by Chih-Wei Hsu

Feb 14, 2008
Approvals

**Chief Advisor:** Chris Jackson, Associate Professor, Computer Graphics Design

<table>
<thead>
<tr>
<th>Chris Jackson</th>
<th>Date</th>
</tr>
</thead>
<tbody>
<tr>
<td>Signature of Chief Advisor</td>
<td></td>
</tr>
</tbody>
</table>

**Associate Advisor:** Dan Deluna, Assistant Professor, Computer Graphics Design

<table>
<thead>
<tr>
<th>Dan Deluna</th>
<th>Date</th>
</tr>
</thead>
<tbody>
<tr>
<td>Signature of Associate Advisor</td>
<td></td>
</tr>
</tbody>
</table>

**Associate Advisor:** Jason Arena, Assistant Professor, New Media Design

<table>
<thead>
<tr>
<th>Jason Arena</th>
<th>Date</th>
</tr>
</thead>
<tbody>
<tr>
<td>Signature of Associate Advisor</td>
<td></td>
</tr>
</tbody>
</table>

**School of Design Chairperson:**

Patti Lachance, Associate Professor, School of Design

<table>
<thead>
<tr>
<th>Patti Lachance</th>
<th>Date</th>
</tr>
</thead>
<tbody>
<tr>
<td>Signature of Administrative Chairperson</td>
<td></td>
</tr>
</tbody>
</table>

**Reproduction Granted:**

I, Chih-Wei, Hsu, hereby grant/deny permission to Rochester Institute of Technology to reproduce my thesis documentation in whole or part. Any reproduction will not be for commercial use or profit.

<table>
<thead>
<tr>
<th>Chih-Wei, Hsu</th>
<th>Date</th>
</tr>
</thead>
<tbody>
<tr>
<td>Signature of Author</td>
<td></td>
</tr>
</tbody>
</table>

**Inclusion in the RIT Digital Media Library Electronic Thesis and Dissertation (ETD) Archive:**

I, Chih-Wei, Hsu, additionally grant to Rochester Institute of Technology Digital Media Library the non-exclusive license to archive and provide electronic access to my thesis in whole or in part in all forms of media in perpetuity. I understand that my work, in addition to its bibliographic record and abstract, will be available to the worldwide community of scholars and researchers through the RIT DML. I retain all other ownership rights to the copyright of the thesis. I also retain the right to use in future works (such as articles and books) all or part of this thesis. I am aware that Rochester Institute of Technology does not require registration of copyright for ETDs. I hereby certify that, if appropriate, I have obtained and attached written permission statements from owners of each third party copyrighted matter to be included in my thesis. I certify that the version I submit is the same as that approved by my committee.

<table>
<thead>
<tr>
<th>Chih-Wei, Hsu</th>
<th>Date</th>
</tr>
</thead>
<tbody>
<tr>
<td>Signature of Author</td>
<td></td>
</tr>
</tbody>
</table>
1. Abstract

The conceptual restaurant website “OFFICE FOODY” offers users a new experience of ordering their meal via the Internet.

Through this website, the restaurant is allowed to receive customer’s order and start to prepare their food immediately. Just like sometimes, we make an order when waiting in the long line during rush hour, this time the users can make an order simply in their office.

Besides, by designing the 2D visual environment based on a restaurant metaphor, the users can not only make their orders, but also customize their favorite setting. Furthermore, the reservation and E-mail interface in the website are designed to enhance the convenience for users when they go for a meal during rush hour.

The notion of creating OFFICE FOODY is “come to the restaurant and enjoy your meal in a second.” This website can help users to deal with all the bothersome process during lunch hour before they reach the restaurant.
2. Keywords

OFFICE FOODY
Restaurant
Visual environment
Metaphor
Customize
3. Project Description

3.1 Overview

The inspiration for developing an interactive restaurant website came two years ago when I worked in an advertising agency. At that time, I found the office employees, including myself, really liked to discuss which restaurants to eat at during lunch hour. Maybe talking with coworkers about where to eat energized them in the morning which was always occupied with bothersome phone calls and emails. Due to this reason, I started to sketch this idea in my mind.

OFFICE FOODY is a website that offers users an interface to make an order through the Internet. In this way, chefs can prepare the food they need before customers reach the restaurant. The advantages of this idea are customer can save valuable time during lunch hour by reducing the time they spend on waiting for the meal and standing in a long line. Also, to take a short break during office hours to think about what they want to eat for lunch could be one of the biggest enjoyments of the workday.

For the enjoyment sake, OFFICE FOODY combines some game elements like character setting and community functions in it. The stage was divided into four parts, the first one is the façade of the restaurant, which offers the general information about OFFICE FOODY. The second one is the lobby, which let users set their personal information and initiate the environment of the website. After finishing these essential settings, the users are invited to the third part – the interior of restaurant, in which they can order their dishes through visualization interface. When users finishing their orders, the scene will transition to the final section - service desk, where users can make a reservation, print their checklist, or send an e-card to invite friends.
Besides, users are allowed to save their favorite dishes into a personal account if they start a membership. Moreover, the snap shot gallery offers a platform for users to share their happy moments in OFFICE FOODY.

The target customers of OFFICE FOODY are white-collar employees who are always accompanied by a computer during office hours and are fascinated by novelty and enjoy pop cultures. Ordering online is definitely not the only way that users can pick what they want. They can just browse what they prefer first and come to the restaurant if they like to deal with people. This is just an option for users to experience an express way in their busy modern days.
After double clicking the order in section one, the stage will transition to SECTION 2 automatically.
3.3 Functional Specification

3.3.1 Homepage

The homepage is mainly composed by the appearance of restaurant and featured by following links and functions:

- **Weather forecast**

  Weather forecast is supported by *Yahoo! Web Service™*. The background can present four different kinds of conditions, which are sunny, rainy, cloudy and snow, depending on the current weather condition.

- **About OFFICE FOODY**

  The link allows users to download the documentation about this site in PDF format.
· **Promotion / Discount**

A window shows discounts and events currently offered in OFFICE FOODY.

· **Map**

A direction guide powers by *Yahoo! Web Service™*.

· **Entrance**

A direction to register / log in page.
3.3.2 Register / Log in

The register / log in page was designed as a restaurant vestibule featured by the following options:

- **Visitor**

Visitors can access the website by inputting the name and selecting a portrait.

- **Membership**

Users have the option to register with OFFICE FOODY. The benefits of registering are the ability to create a personal menu in the account number and to have special discounts.

Required fields for registration in OFFICE FOODY:
3.3.3 Instruction

Before users enter into the main stage, the instructions shows some important features of this website.
3.3.4 Main Stage

The main stage is composed by two parts: one is the navigation area, which users can hang around in and place their order. The other section is the information panel, which is used to record users’ data.

- **Information panel (a.)**

  This area is used to record users information and update their orders.

- **Global navigation (b.)**

  The global navigation was designed as flags hang on the ceiling, which are
  
a. Beverage

  b. Soup
c. Entree

d. Snack

e. Message board

f. Service desk

· **Quick Link (c.)**

For convenience sake, users can pick their orders by using quick links, which can slide among different parts of selection immediately.

· **Reset button (d.)**

 Resets the users' order.

· **Check out button (e.)**

 After the users finish their order, they click this button to shift the stage to service desk.

· **Load customer's favorite items (f.)**

 OFFICE FOODY allows members to save a set of their favorite dishes in the database. By clicking this button, users can reload the items they have saved.

· **Sound Controller (g.)**

 To turn on or turn off the background sound.
3.3.5 Menu

The menu window pops up when users click on any flag on the main stage.

- **Image holder (a.)**
  Displays the picture of item that the user is currently clicked on.

- **List (b.)**
  Displays the dishes OFFICE FOODY is serving today.

- **Select button (c.)**
  To select the desired dishes, the selected items will update on the information panel.
· **Add selection to favorite (d.)**

This allows members to save a set of their favorite dishes in the database. By clicking this button, users can save the selected items in the database.
3.3.6 Message Board

Just like sometimes we can see the restaurant has a billboard for customers to write down the message and attach the snap shots they take in the restaurant, this message board is designed for users to leave their message and share their happy moments in OFFICE FOODY.

- **Message Board Area (a.)**
  
The snap shot on the message board is designed as Polaroid film that indicates different content in it.

- **Upload button (b.)**
  
Users click this button if they have any pictures or messages they want to upload to the message board.
3.3.7 Service Center

After a user finishes their order and makes sure they want to check out, the stage will slide to the service center, which features the following options:

- **Send to friend (a.)**
  This function allows users to leave messages and select the background they prefer to their friends.

- **Make reservation (b.)**
  Makes a reservation by assigning the seat and time section.

- **Review checklist (c.)**
It allows the user to send a recipe to their e-mail account as a copy.
4. Process

4.1 Visual identity

The first step of the design process was VI (Visual Identity) development. In the very beginning, I designed four logos that represent different styles as following:

<table>
<thead>
<tr>
<th>relaxed, simplified</th>
<th>succinct, urban style</th>
</tr>
</thead>
<tbody>
<tr>
<td>OFFICE FOODY</td>
<td>OFFICE FOODY</td>
</tr>
<tr>
<td>playable, happiness</td>
<td>fun, childish</td>
</tr>
<tr>
<td>OFFICE FOODY</td>
<td>FOODY</td>
</tr>
</tbody>
</table>

After I discussed each logo concept with my thesis committee members, I decided to use the bottom left logo as the foundation. This logo made a better connection to the notion of this website. In addition, the shape is also much easier to apply on web-based environment.

In the next step, I tried to integrate graphic with texts. After several attempts on typography, I decided to take the following one on next page as final VI.
Final visual identity  Color: R253 1G85 B19  Font: Engravers MT Bold
4.2 Appearance of restaurant

After I finished the visual identity, I started to illustrate the facade of the restaurant. I gathered some images of restaurants from digital resources and magazines and tried to get some inspiration from them. The following one is the first edition of the restaurant:

I took this one and discussed it with my committee members. We concluded that this was this one is just a nice illustration but the characteristic did not match to VI. I also ignored the consistency between VI and the appearance of restaurant.

Then I started to work on the second edition. I decided to add more imagination and cartoon elements in it, since I want to develop a website for people to enjoy. I decided it should look playable and vivid as the VI does. Based on this concept, I tried to represent utensils and foods we can easily see in the restaurant as design elements, for example. I used straw as pipe, coffee cup as building and cheesecake as roof. In order to make it look more colorful, I applied a lot of patterns to this edition instead of soft gradient as I did in the last edition.

The final appearance of restaurant can be found in 3.3.1.

4.3 Character design
The following process was used in character design. This process went without a hitch because the design style was clearly represented in the last couple steps.

Following are characters I designed for users to select as identity in OFFICE FOODY:

Character portrait in OFFICE FOODY
4.4 Menu

The menu contains three elements in it: title, image and buttons. In the beginning, I displayed all dishes’ pictures on the window. Users could navigate them through moving the cursor and then buttons and titles came out when users click on any of them.

This design looked straightforward to me, but I received some feedback from users. The feedback indicated the following facts:

- Users had no idea what they were ordering through only images. They would like to know the title before they clicked on any of them.
- Users would like to see a larger image instead a lot of images at a time, too many images at the same time bother them.

Based on these points, I made the improvements that can be found in 3.3.5.
4.5 Interior

Because the navigation of the main stage is based on a parallax scrolling image, it is very important to work on creating the illusion of the depth for the interior or it would look very unnatural. When I started to design this part, I simply thought what I needed to do was illustrate different counters and put them on the stage then everything will be fine.

However, it didn’t work well because it made the stage look very flat, which means the depth created by the parallax image was not as apparent on the screen. In order to solve this problem, I tried to find a way to create the depth besides merely scaling down or scaling up the objects. After I placed the beverage counter up on the ceiling, stuck the soup counter to the wall, embedded the snack counter in the wall and put the entrée counter into the concavity, the parallax image could finally be shown naturally on the stage.

![Interior design – main stage](image)
4.6 Programming

After completing the graphical elements of the website I began writing the necessary programming code. The programs utilized in developing this website include Flash Actionscript 2.0, XML, HTML, mySQL and PHP. A coding example can be found in appendix.

The navigation of the main stage is based on a parallax scroll image by using Actionscript 2.0, which offers a very intuitive way for users to browse through the website. The menu information including image, price and name are all loaded from external data through XML and Actionscript 2.0. It is more efficient for a designer to maintain the site in the future by importing the external data.

The log in / register function was created by using Actionscript 2.0, PHP and mySQL database. By combining these codes with Flash-based graphical element, it can create more stylized interface than HTML format.

A message board allows users to upload their personal image and leave messages; it was created by using Actionscript 2.0 file reference class and PHP. The “send to a friend” function was also created by using Actionscript 2.0 and PHP. Both of these functions are based on the concept of sharing in order to achieve the goal of promoting the website through users themselves; word of mouth.
5. Usability

Following are the points I modified after user test when the project was nearly complete.

5.1 Introduction Page (see figure 3.3.3)

In the beginning, I made introduction as a 90 seconds Flash movie, but I found it is painful to users to go through it. To improve this problem, I took some important features which OFFICE FOODY has and designed them as an iconic style just like the way that electronic devices print their features on the package or manual (see figure below). In this way, users can spend less time and it is also clearer than the previous design.

![Iconic style of features introduction](image)

5.2 Buttons

The buttons I designed for food counters and service desk were flags hung up on the ceiling. For some users, this is not a very intuitive design, some of them got lost when they entered into the main stage.

To solve this problem, I tried to assign each flag to a different color. I picked some bright colors so that the flags would stand out on the stage. In this way, most users could tell the flags were buttons that they can click on.
5.3 Global navigation

The committee reminded me that some users might not have too much time or patients to play around with website, I need to design another navigator for these users.

The global navigation offers a faster, easier way for users to pick their desired dishes. I designed it as traditional buttons with the same color scheme as the flags.

5.4 Fonts

Some users felt that the fonts I used on the button were not very clear. To solve this problem, I turned off the anti-alias function and then the texts could be shown very clearly and still kept the fonts in the same size. (see figure 5.3)
6. Conclusion

After I visited the homepage of Kinetic design studio
(http://www.kinetic.com.sg/k20035/launch.htm) three years ago, to develop an
environmental based website like that became my biggest ambition in my graduate study.
Including my thesis project, I had done three environmental based websites in this two years
and I think I finally got everything I pursued integrated into this thesis project.

However, this website can still expand a lot especially in community functionalities, which is a
feature that the most popular trend in today's website development. For example, allows
users to write down their comments about OFFICE FOODY is a necessary and important one.
Besides, although OFFICE FOODY can process small amounts of data from users, it will
absolutely need more solid server side skills to deal with the database if I want to publish it.

Programming is always the toughest work to me, I need to thank my committee members who
gave me a lot of help in this part and push me to another level which I never thought I can
reach. I am glad that I can finally integral the graphic skills I learned in my undergraduate
study and programming skills I learned here in this interactive thesis project.
7. Appendix

7.1

Following codes demonstrate how to get RSS feeds from Yahoo! Web Service™ and display the information on the stage:

```javascript
var weather_xml = new XMLHttpRequest();
weather_xml.onreadystatechange = function() {
    if (weather_xml.readyState == 4) {
        if (weather_xml.status == 200) {
            var weather_xml = new XML();
            weather_xml.ignoreWhitespace = true;
            weather_xml.onload = function() {
                var conditionCode = this.firstChild.firstChild.childNodes[12].childNodes[7].attributes.code;
                var date_txt = this.firstChild.firstChild.childNodes[12].childNodes[7].attributes.date;
                var weatherCondition_txt = this.firstChild.firstChild.childNodes[12].childNodes[7].attributes.text;
                // trace(this.firstChild.firstChild.childNodes[12].childNodes[7].attributes.text);
                if (conditionCode == 31 || conditionCode == 32 || conditionCode == 33 || conditionCode == 34) {
                    _root.weatherChange_mc.gotoAndStop(1);
                } else if (conditionCode == 26 || conditionCode == 27 || conditionCode == 28 || conditionCode == 29) {
                    _root.weatherChange_mc.gotoAndStop(1);
                } else if (conditionCode == 35 || conditionCode == 36 || conditionCode == 37 || conditionCode == 38 || conditionCode == 40) {
                    _root.weatherChange_mc.gotoAndStop(1);
                } else if (conditionCode == 5 || conditionCode == 8 || conditionCode == 10 || conditionCode == 13 || conditionCode == 14 || conditionCode == 15 || conditionCode == 16 || conditionCode == 17 || conditionCode == 41) {
                    _root.weatherChange_mc.gotoAndStop(1);
                } else if (conditionCode == 42) {
                    _root.weatherChange_mc.gotoAndStop(1);
                } else {
                    _root.weatherChange_mc.gotoAndStop(1);
                }
            }
        }
    }
};
weather_xml.open('GET', 'http://weather.yahooapis.com/forecast/rss?p=1462');
weather_xml.send();
```
7.2

Following codes demonstrate how to register new users by using array, and utilize the if statement to check all incoming date of the user input:

```javascript
function registerUser() {
    var validated = true;
    var textFieldArray = new Array("firstName", "lastName", "username", "useremail");
    //error txt text = "";
    // Check if empty fields
    for (var i = 0; i < textFieldArray.length; i++) {
        var theField = _root.scene2interface_mc.regisInfo.mc.textField[i].text;
        if (theField.length < 2) {
            trace("theField^
            validated = false;
            _root._alkPop_mc.feedback_txt.text = "Sorry, all fields must contain at least 2 characters!");
            break;
        }
    }
    // IF valid, check password and send to PHP script
    if (validated) {
        menuItemSound.start();
        userDets.firstName = _root.scene2interface_mc.regisInfo.mc.firstName_txt.text;
        userDets.lastName = _root.scene2interface_mc.regisInfo.mc.lastName_txt.text;
        userDets.username = _root.scene2interface_mc.regisInfo.mc.username_txt.text;
        userDets.userEmail = _root.scene2interface_mc.regisInfo.mc.useremail_txt.text;
        userDets.password = _root.scene2interface_mc.regisInfo.mc.password_txt.text;
        userDets.firstName = _root.scene2interface_mc.regisInfo.mc.firstName_txt.text;
        userDets.lastName = _root.scene2interface_mc.regisInfo.mc.lastName_txt.text;
        userDets.username = _root.scene2interface_mc.regisInfo.mc.username_txt.text;
        userDets.password = _root.scene2interface_mc.regisInfo.mc.password_txt.text;
        userDets.firstName = _root.scene2interface_mc.regisInfo.mc.firstName_txt.text;
        userDets.lastName = _root.scene2interface_mc.regisInfo.mc.lastName_txt.text;
        userDets.username = _root.scene2interface_mc.regisInfo.mc.username_txt.text;
        userDets.password = _root.scene2interface_mc.regisInfo.mc.password_txt.text;
        userDets.firstName = _root.scene2interface_mc.regisInfo.mc.firstName_txt.text;
        userDets.lastName = _root.scene2interface_mc.regisInfo.mc.lastName_txt.text;
        userDets.username = _root.scene2interface_mc.regisInfo.mc.username_txt.text;
        userDets.password = _root.scene2interface_mc.regisInfo.mc.password_txt.text;
        userDets.firstName = _root.scene2interface_mc.regisInfo.mc.firstName_txt.text;
        userDets.lastName = _root.scene2interface_mc.regisInfo.mc.lastName_txt.text;
        userDets.username = _root.scene2interface_mc.regisInfo.mc.username_txt.text;
        userDets.password = _root.scene2interface_mc.regisInfo.mc.password_txt.text;
    } else {
        password_txt.text = "Please enter password!
        _root._alkPop_mc.feedback_txt.text = "Sorry, all fields must contain at least 2 characters!");
        break;
    }
}
```
7.3

Following codes demonstrate how to allow users to upload their files to a specific folder on the sever by using Flash file reference class and PHP:

```javascript
7.3

```
 Following codes demonstrate how to load the information from database and visualize the data on the stage:

```javascript
var today_date = new Date();
var sendDate = String((today_date.getFullYear()) + "-(today_date.getMonth() + 1)" + today_date.getDate() + " " + today_date.getHours() + ":" + today_date.getMinutes() + ":" + today_date.getSeconds());
tableID = "upload";
baseUrl = "http://cias.rit.edu/~cxh2617/upload08/";
function showResult() {
    var imageInfo = imgeXML.firstChild;
    for (var i = 0; i < imageInfo.length; i++) {
        clip.id = i;
        clip.x = 45 + Math.random() * 700;
        clip.y = 65 + Math.random() * 320;
        clip.rotation = Math.random() * 120;
        clip.onRelease = function() { slideSound.start();
            showHolder = true;
            for (var j = 0; j < imageInfo.length; j++) {
                x = j;
            }
            if (j < imageInfo.length) {
                imageHolder = new XML();
                imageXML.ignoreWhite = true;
                imageXML.onLoad = showResult;
                sendImage.sendXML = new Image();
                sendImage.sendXML.onLoad = uploadImage;
                sendImage.sendXML = sendXML;
            } else {
                sendImage.sendXML = new Image();
                sendImage.sendXML.onLoad = uploadImage;
            }
        }
        clip._y = 65 + Math.random() * 320;
        clip._x = 45 + Math.random() * 700;
        clip._rotation = Math.random() * 120;
        clip._id_txt = imageInfo[clip.id].attributes.imagelD;
        clip.onRelease = function() { slideSound.start();
            showHolder = true;
            for (var j = 0; j < imageInfo.length; j++) {
                x = j;
            }
            if (j < imageInfo.length) {
                imageHolder = new XML();
                imageXML.ignoreWhite = true;
                imageXML.onLoad = showResult;
                sendImage.sendXML = new Image();
                sendImage.sendXML.onLoad = uploadImage;
            } else {
                sendImage.sendXML = new Image();
                sendImage.sendXML.onLoad = uploadImage;
            }
        }
    }
    showHolder = false;
    for (var i = 0; i < imageInfo.length; i++) {
        imageInfo[i].attributes.imagelD = tableID;
        imageInfo[i].attributes.imagelD = tableID;
        sendImage.sendMessage = new XML();
        sendImage.sendMessage.onLoad = uploadImage;
        sendImage.sendMessage = sendXML;
        sendImage.sendMessage = new XML();
        sendImage.sendMessage.onLoad = uploadImage;
        sendImage.sendMessage = sendXML;
        sendImage.sendMessage = new XML();
        sendImage.sendMessage.onLoad = uploadImage;
        sendImage.sendMessage = sendXML;
    }
}
function createPic() {
    for (i = 0; i < imageInfo.length; i++) {
        imageInfo[i].attributes.imagelD = tableID;
        imageInfo[i].attributes.imagelD = tableID;
        sendImage.sendMessage = new XML();
        sendImage.sendMessage.onLoad = uploadImage;
        sendImage.sendMessage = sendXML;
        sendImage.sendMessage = new XML();
        sendImage.sendMessage.onLoad = uploadImage;
        sendImage.sendMessage = sendXML;
    }
}
function createPic() {
    for (i = 0; i < imageInfo.length; i++) {
        imageInfo[i].attributes.imagelD = tableID;
        imageInfo[i].attributes.imagelD = tableID;
        sendImage.sendMessage = new XML();
        sendImage.sendMessage.onLoad = uploadImage;
        sendImage.sendMessage = sendXML;
        sendImage.sendMessage = new XML();
        sendImage.sendMessage.onLoad = uploadImage;
        sendImage.sendMessage = sendXML;
    }
}
function showResult() {
    var imageInfo = imgeXML.firstChild;
    for (var i = 0; i < imageInfo.length; i++) {
        clip.id = i;
        clip.x = 45 + Math.random() * 700;
        clip.y = 65 + Math.random() * 320;
        clip.rotation = Math.random() * 120;
        clip.id_txt = imageInfo[clip.id].attributes.imagelD;
        clip.onRelease = function() { slideSound.start();
            showHolder = true;
            for (var j = 0; j < imageInfo.length; j++) {
                x = j;
            }
            if (j < imageInfo.length) {
                imageHolder = new XML();
                imageXML.ignoreWhite = true;
                imageXML.onLoad = showResult;
                sendImage.sendXML = new Image();
                sendImage.sendXML.onLoad = uploadImage;
            } else {
                sendImage.sendXML = new Image();
                sendImage.sendXML.onLoad = uploadImage;
            }
        }
        clip._y = 65 + Math.random() * 320;
        clip._x = 45 + Math.random() * 700;
        clip._rotation = Math.random() * 120;
        clip._id_txt = imageInfo[clip.id].attributes.imagelD;
        clip.onRelease = function() { slideSound.start();
            showHolder = true;
            for (var j = 0; j < imageInfo.length; j++) {
                x = j;
            }
            if (j < imageInfo.length) {
                imageHolder = new XML();
                imageXML.ignoreWhite = true;
                imageXML.onLoad = showResult;
                sendImage.sendXML = new Image();
                sendImage.sendXML.onLoad = uploadImage;
            } else {
                sendImage.sendXML = new Image();
                sendImage.sendXML.onLoad = uploadImage;
            }
        }
    }
}
function createPic() {
    for (i = 0; i < imageInfo.length; i++) {
        imageInfo[i].attributes.imagelD = tableID;
        imageInfo[i].attributes.imagelD = tableID;
        sendImage.sendMessage = new XML();
        sendImage.sendMessage.onLoad = uploadImage;
        sendImage.sendMessage = sendXML;
        sendImage.sendMessage = new XML();
        sendImage.sendMessage.onLoad = uploadImage;
        sendImage.sendMessage = sendXML;
    }
}
function showResult() {
    var imageInfo = imgeXML.firstChild;
    for (var i = 0; i < imageInfo.length; i++) {
        clip.id = i;
        clip.x = 45 + Math.random() * 700;
        clip.y = 65 + Math.random() * 320;
        clip.rotation = Math.random() * 120;
        clip.id_txt = imageInfo[clip.id].attributes.imagelD;
        clip.onRelease = function() { slideSound.start();
            showHolder = true;
            for (var j = 0; j < imageInfo.length; j++) {
                x = j;
            }
            if (j < imageInfo.length) {
                imageHolder = new XML();
                imageXML.ignoreWhite = true;
                imageXML.onLoad = showResult;
                sendImage.sendXML = new Image();
                sendImage.sendXML.onLoad = uploadImage;
            } else {
                sendImage.sendXML = new Image();
                sendImage.sendXML.onLoad = uploadImage;
            }
        }
    }
}
```
7.5
Following codes demonstrate the construction of XML files for OFFICE FOODY's menu, and illustrate how all data is being pulled to display for specific dishes:

```xml
<menu name="beverage">
  <project title="Tea" thumb="assets/beverage/thumb1.jpg" image="assets/beverage/image1.jpg" price="1.75"/>
  <project title="Kiwi Smoothie" thumb="assets/beverage/thumb2.jpg" image="assets/beverage/image2.jpg" price="1.75"/>
  <project title="Coffee Latte" thumb="assets/beverage/thumb3.jpg" image="assets/beverage/image3.jpg" price="2.25"/>
  <project title="Cake" thumb="assets/beverage/thumb4.jpg" image="assets/beverage/image4.jpg" price="1.25"/>
</menu>

<menu name="soup">
  <project title="Creamy Tomato Soup" thumb="assets/soup/thumb1.jpg" image="assets/soup/image1.jpg" price="1.75"/>
  <project title="Pumpkin Soup" thumb="assets/soup/thumb2.jpg" image="assets/soup/image2.jpg" price="1.25"/>
  <project title="Mushroom Soup" thumb="assets/soup/thumb3.jpg" image="assets/soup/image3.jpg" price="1.05"/>
  <project title="Red Borsch" thumb="assets/soup/thumb4.jpg" image="assets/soup/image4.jpg" price="4.05"/>
</menu>

<menu name="entree">
  <project title="Grilled Sea Bass" thumb="assets/entree/thumb1.jpg" image="assets/entree/image1.jpg" price="5.25"/>
  <project title="Chicken Fillet" thumb="assets/entree/thumb2.jpg" image="assets/entree/image2.jpg" price="4.95"/>
  <project title="Grilled Steak" thumb="assets/entree/thumb3.jpg" image="assets/entree/image3.jpg" price="4.75"/>
  <project title="Pasta Salad" thumb="assets/entree/thumb4.jpg" image="assets/entree/image4.jpg" price="4.05"/>
</menu>

<menu name="snack">
  <project title="Raspberry Chocolate" thumb="assets/snack/thumb1.jpg" image="assets/snack/image1.jpg" price="1.25"/>
  <project title="Assorted Cookies" thumb="assets/snack/thumb2.jpg" image="assets/snack/image2.jpg" price="1.95"/>
  <project title="Apple Pie" thumb="assets/snack/thumb3.jpg" image="assets/snack/image3.jpg" price="1.00"/>
  <project title="Biscuit" thumb="assets/snack/thumb4.jpg" image="assets/snack/image4.jpg" price="0.95"/>
</menu>
```

7.6
Following codes take XML nodes above and display the information on the stage by using for loops:

```javascript
function generateImage(id:Number){
  for(var i:Number = 0; i < menuTypes.length; i++){
    var xpos:Number = 0;
    var menu_MC:bigContainer_MC.container_MC.firstChild.childNodes[i].attributes.image, currentImage;
    var currentImage = currentImage + 238 + (270 * i);
    myXML.loadClip(menu_MC.firstChild.childNodes[i].childNodes[1].atts.image, currentImage);
    for(var j:Number = 0; j < menuTypes[i].childNodes.length; j++){
      var xpos:Number = 0;
      var currentImage = currentImage + 238 + (270 * i);...
```
8. Bibliography

Books


Websites

http://demo.fb.se/e/ikea/dreamkitchen/site/default.html


http://www.thelinear.com

http://www.heineken.com.sg/innovators