

Design and Implementation of a Mobile Shopping App Specializing in Regional Product

Kyubark Shim¹ and Jaegeol Yim²

¹*Department of Applied Statistics, Dongguk University at Gyeongju, Korea*

²*Dept. of Computer Engineering Dongguk University, Gyeongju, Gyeongbuk,
Korea*

{shim, yim}@dongguk.ac.kr

Abstract

This paper introduces our design and implementation of a mobile shopping app specializing in local products. One of the most distinguished features of this app is that it is closely cooperates with the local Internet TV. The Internet TV service is integrated into the app, and special events for sales promotion held by the M-commerce system can be dynamically directed to the local Internet TV. The M-commerce system is a client-server system. The server system supports both this M-commerce and the E-commerce that also specializes in regional product. This paper focuses on description of our implementation of the client part.

Keywords: *Mobile App; M-Commerce; Client-Server; Android; Internet TV; E-Commerce*

1. Introduction

As sales of mobile devices grow exponentially, the usage of mobile apps become more integrated into our lives. The authors of [1] introduced Careggi Smart Hospital that provides functions of search and outdoor navigation. With the search, user can find buildings, departments and people. The navigation helps users to find a path to the selected building and shows information about the accessibility of the destination building.

The authors of [2] introduced Android apps they have implemented including Expense and Bollywood Bingo. Expense is an Android App that allows users to track and manage expenses simply and intuitively. Bollywood Bingo is a bingo game to be played via mobile devices. They found a few challenges for the development of the Indian mobile app market including device fragmentation (Fine tuning and validating apps for a broad range of Android devices) and late changes (Customers can experiment and play with an app and come up with change requests that can include ideas to improve the look and feel of an app or optimize its accessibility or new features that developers had not thought of earlier.)

The authors of [3] introduced their mobile app attempting to re-imagine, in 3D, the Labyrinthe de Versailles (a garden) in its original 17th century state. The authors of [4] introduced the design and implementation of a smart meeting Android application called CyberOffice which is based on Wi-Fi Direct. This application enables a presenter to start meetings quickly anywhere at any time without Internet connection.

This paper introduces our design of mobile shopping app that sells regional products. One of the main features of this app is that it makes use of local Internet TV as much as it can. Firstly, the Internet TV service provided by the local government is integrated in the app. Secondly, the shopping and the Internet TV cooperate with each other. For example, if the Internet TV telecasts a regional emergency, it is automatically fed into the shopping app.

2. Related Works

Considering others' research results, we designed the app more efficient, safe, and attractive. The authors of [5] uncovered 12 types of user complaints. They found that functional errors, feature requests and app crashes are the most frequent complaints. Complaints about privacy and ethical issues, and hidden app costs have the most negative impact on the rating of an app. They also found that users attributed their complaint to a recent updates of the app in 11% of the reviews. Identified complaint types are: App crashing, compatibility, feature removal (complaint about a disliked feature like ads), feature request (app needs additional features to get a better rating), functional error, hidden cost, interface design (not intuitive design), network problem (slow response), privacy and ethical, uninteresting content (boring game), unresponsive app (slow response to input), not specific (Honestly the worst app ever!).

The authors of [6] identified common functionalities of existing mobile virtual labs and they exploited App Quality Alliance guidelines towards proposing a method for developing mobile virtual labs. The proposed guidelines are grouped into User interface, Performance, and Network Utilization.

The author of [7] designed a new multimedia push framework to integrate diverse existing push messaging gateways and to provide a store-and-forward function for mobile applications. By integrating the gateways, the server side developers do not have to implement many kinds of gateway functions. Also, by providing store-and-forward functionality, the mobile application developers do not have to worry about losing their notification message in the air area.

The authors of [8] defined the secure coding rules that reflect the characteristics of the mobile environments and applications. They designed and implemented the compiler that inspects vulnerabilities of the mobile applications based on the defined secure coding rules.

The authors of [9] investigated the most highly used game app in the Android open market and analyzed the normal system call event patterns. They also investigated 1260 malware samples distributed by Android MalGenome Project and analyzed the malicious system call event patterns.

The author of [10] discussed the characteristics of automotive model-based development processes, the consequences for test development and the need to reconsider testing procedures in practice. The authors of [11] developed an easy-to-customize framework to enhance the reusability. The authors of [12] proposed a new view synthesis technique for coding of multi-view color and depth data in arbitrary camera arrangements.

3. Design of the App for M-commerce

Our mobile app is a hybrid app in that both M-commerce and live telecast are integrated in the app. This app can be considered hybrid in another view point. In other words, web applications are also integrated in the app. The main page of this app is designed as shown in Figure 1. The main menu, TV Player, and the list of recommended merchandise are the main components of the main page. The main menu consists of the Shopping TV, EPG, Food, Tourism, Smart product, and, Ventures buttons. The unique feature of this app is that the shopping TV is integrated in it. Whenever the Shopping TV button is clicked, the shopping TV is played on the screen. When the EPG button is clicked, the TV schedule of the shopping TV is displayed. The remaining menu buttons represent the categories of merchandise. When the Quick menu button is clicked, the screen slides to the right hand side and the new menu view consisting of the Shopping cart, view all orders, My page, Cancel/change/refund, category1,, and k-th category buttons appear as shown in Figure 2. If the user selects a subcategory, then three items picked by the manager and all the items belong to the subcategory displayed as shown in

Figure 3. Maximum ten items can be displayed in the view. We provide a scroll button to allow the user to browse another page [13].

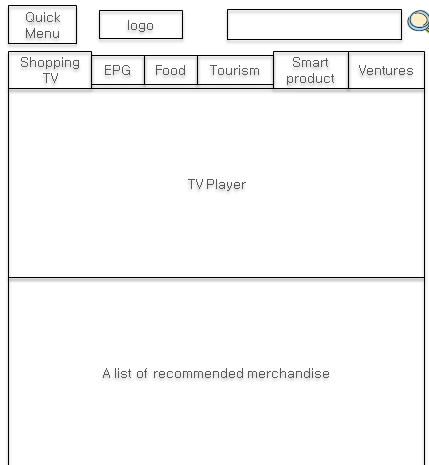


Figure 1. The Layout of the Main



Figure 2. Quick Menu Window

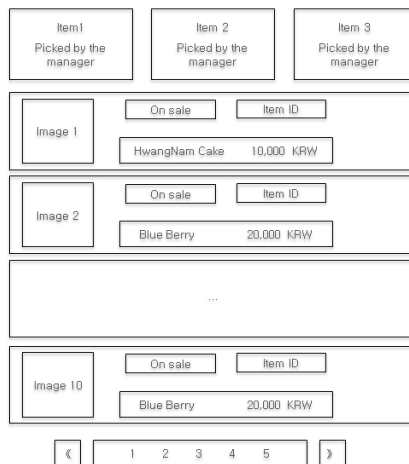


Figure 3. The Product List View for the Selected Category

The figure shows a product detail form with the following elements:

- Item ID
- Item name and a brief description of the item in natural language
- Image
- Type: - Select Option
- How many: - 1 +
- How much: 10,000 KRW
- Buy
- Cart
- Product detail
- Delivery/Refund/Change Policy
- Detail description of the product

Figure 4. The Product Detail View for the Selected Product

If the user selects a certain product from the product list view, then it moves to the product detail view shown in Figure 4. In this view, the user is allowed to select a type (color, for example) of the product, to determine the quantity of the product to purchase, to move to the payment page, and to move to the cart page. If the selected product is a packaged product that consists of two or more items then more than one type selection boxes, one per item, appear in the product detail view. When the selected product is out of stock, then we move to the sold out view instead of the product detail view [13].

In the cart page, a list of all items selected by the user is displayed. The user is allowed to select some items and delete them from the cart. The user is also allowed to adjust quantities. The total price of the products in the cart is changed as the user changes the quantity. From the cart page, the user can move to the shopping page after checking the items in the cart or move to the payment page in order to complete the transaction. By the way, if more than one items are sold by the same provider, then it will be reflected in the total delivery charge in this page. For example, a box of herb tea (delivery charge is 2 USD) and a box of green tea (delivery charge is 2USD) are in the cart and both of them are provided by the same merchant then total delivery charge will be 2, not 4, USD [13].

In the first payment page, a list of all ordered product items, total price plus delivery charge, and the basic information of the user will be displayed. If the user flips the screen, then the second payment page appears. In this page, the user can specify the address to which the merchandise is delivered. This address can be the same as the address of the user. In this case, the page will be automatically filled up with the basic information of the user. Payment can be done by either a credit card or the user's phone. After selecting payment method, the user can see the confirmation page. In the confirmation page, the user must make a check mark on the "I understand the e-commerce law and I am confirming the total price and the list of products I am purchasing" sentence before clicking the payment button. The user can click the cancel button instead of the payment button. After payment, the user can take a look at a list of the purchased products.

If the "My page" menu in the quick menu is clicked, then a list of the transactions the user made is displayed. The status of a transaction can be one of "Ordered", "Payment done", "Before delivery", "Under delivery", "Delivery done", or "Canceled". From the "My page", we can move to the 'trace the delivery' page or the 'detail information of the order' page. In the 'detail information of the order' page, we can see the description of the ordered product, the day of purchase, the total price, the receiver, and the payment information.

4. Design and Implementation of the System

We design a database consisting of many tables. A few of them are shown in Figure 5. We have the Menu, SystemCode and ErrorLog tables to record information of menus, system codes, and error logs. Meaning of the remaining tables is obvious from the table names.

In the web.config file for mobile app development, we define URLs for all web services as shown in Figure 6. When the URL of the web server is changed, we have to change the values in the figure. Once we change the values, the program runs properly without changing any program code. LGD_MERTKEY defines the key for the LG payment gate.

A part of the program to open the main page of the mobile app is shown in Figure 7. The class has the Page_Load method as other classes do. The BindMainBannerProduct() method binds the main product to the main banner. What the remaining methods do can be guessed from the names of the methods.

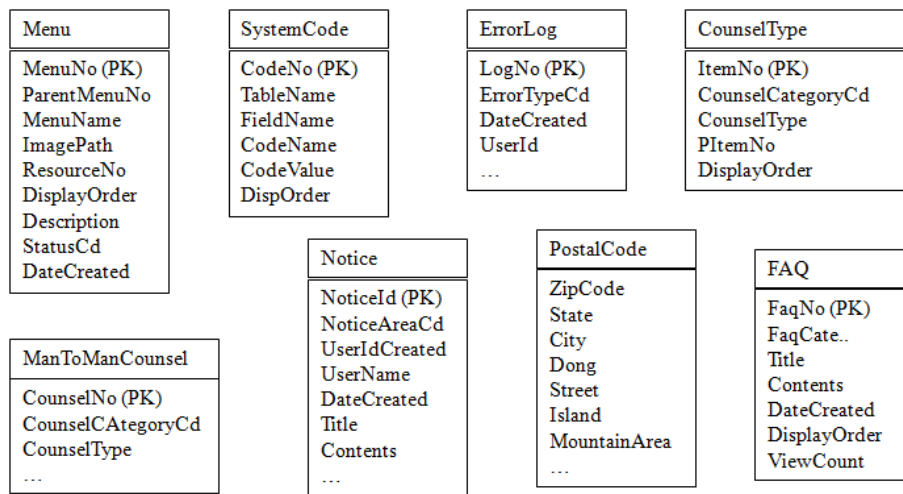


Figure 5. Several Sample Tables from the Database

```

<setting name="...MOBILE_WEB...Catalog" serializeAs="String">
  <value>http://112.....:3100/DisplayCatalog.aspx</value>
</setting>
<setting name="...MOBILE_WEB...DisplayCategory" serializeAs="String">
  <value>http://112.....:3100/DisplayCategory.aspx</value>
</setting>
<setting name="...MOBILE_WEB...Product" serializeAs="String">
  <value>http://112.....:3100/Product.aspx</value>
... name="...MOBILE_WEB...DisplayCatalogContents" serializeAs="String">
<value>http://112.....:3100/DisplayCatalogContents.aspx</value>
</setting>
...
<add key="LGD_MERTKEY" value="1...3...0"/>
    
```

```
<add key="CST_PLATFORM" value="platform"/>
<add key="CST_MID" value="merchantID"/>
<add key="LGD_CONFIG_PATH" value="C:\\..."/>
<add key="LGD_CASNOTEURL" value="http://...gjkorea.com/....aspx"/>
<add key="LGD_RETURNURL" value="....aspx"/>
...
```

Figure 6. A Part of the Web Configuration File

A part of the Detail class is shown in Figure 8. Those methods to display detail information of a product, to handle an order request, and to handle putting a product item into the cart are defined in the class. The BindSKU and BindSetSKU methods invoke the web services to fetch options for the product and bind it to the window. The DoNonMemberCart and DoCart methods invoke the web services to put the product into the basket.

```
...
protected void Page_Load(object sender, EventArgs e)
private void BindMainBannerProduct()
private void BindRecommendProduct()
private void BindSetProduct()
private void BindBestProduct()
...
```

Figure 7. A Part of the Index Class

```
public partial class Detail : System.Web.UI.Page
...
protected void Page_Load(object sender, EventArgs e)
protected void OrderButton_Click(object sender, EventArgs e)
protected void CartButton_Click(object sender, EventArgs e)
private void BindProduct()
private void BindSKU(string strProductCode)
private void BindSetSKU()
private int DoNonMemberCart(string basketTypeCd)
private int DoCart(string basketTypeCd)
...
```

Figure 8. A Part of the Methods Defined in the Detail Class

5. Experiments

We tested the process shown in Figure 2 with various radius (note that the process assumes that the bus is at the bus stop j if the distance between $A[i]$ and $B[j]$ is less than

the radius): 25, 50, 75, and 100 meters. Our test results showed that the process correctly recognizes the bus route when the radius is set to 50 meters. An example test result is shown in Figure 5. When the radius is 25, there are many false-negatives whereas there are many false-positives when the radius is greater than 50.

We have performed experiments to test our implementation. A screenshot of the main page we can obtain by accessing m.gjmart.net is shown in Figure 9. The main menu, the best item, recommended items, and others are shown in the main page. Home Shopping, Electronic Program Guide (EPG), Food, and regional product are submenus of the main menu. When the EPG submenu is clicked, a page similar to Figure 10 appears. We can read TV schedules on this page. If we click the food submenu, then a page similar to Figure 11 appears.



Figure 9. A Part of the Main Page

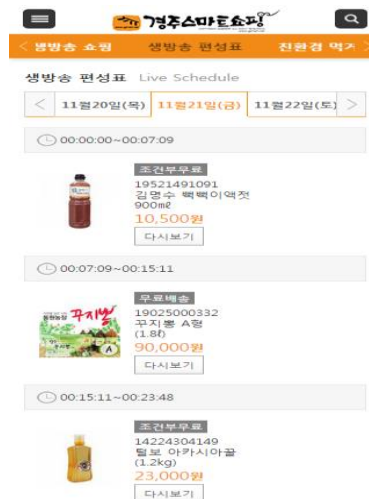


Figure 10. A Part of the EPG Page

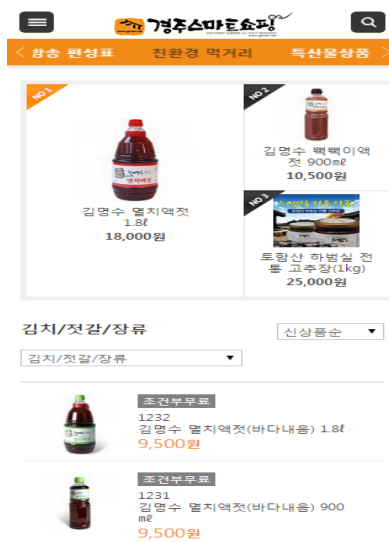


Figure 11. A Part of the Food-Category Page

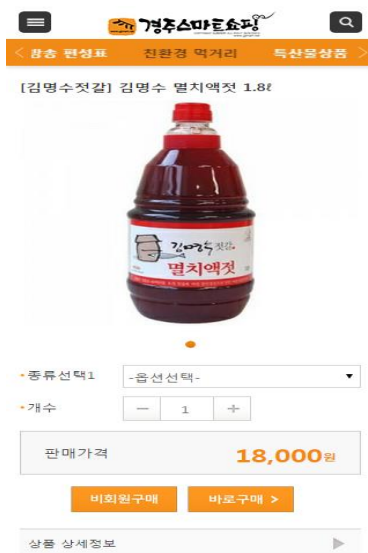


Figure 12. A Part of the "Product Detail" Page

If we click any image of product, we move to the "product detail" page similar to Figure 12. This page allows us to choose an option (size, color, quality, and type), increase or decrease quantity, move to the "description" page, and move to the order page. A part of the "description" page is shown in Figure 13 and a part of the "order" page is shown in Figure 14.

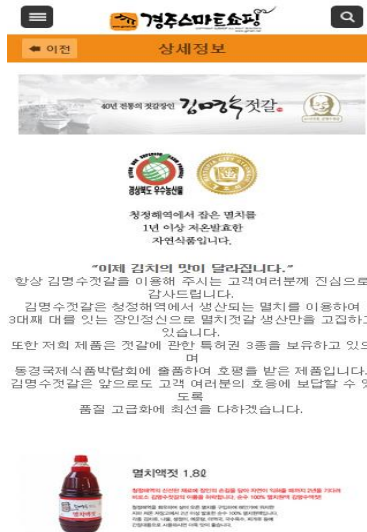


Figure 13. A Part of the "Description" Page

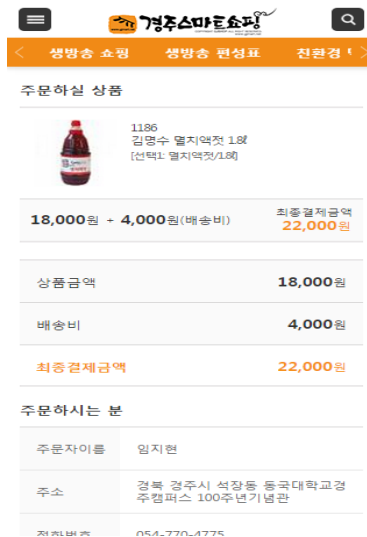


Figure 14. A Part of the "Order" Page

The option menu button is always enabled. The submenus of the main menu are all included in the list of the submenus of the option menu as shown in Figure 15. If we choose the "cart" button, then we can move to the "cart" page shown in Figure 16 whereas we can move to the "My page" page shown in Figure 17 by clicking the "My page" button.

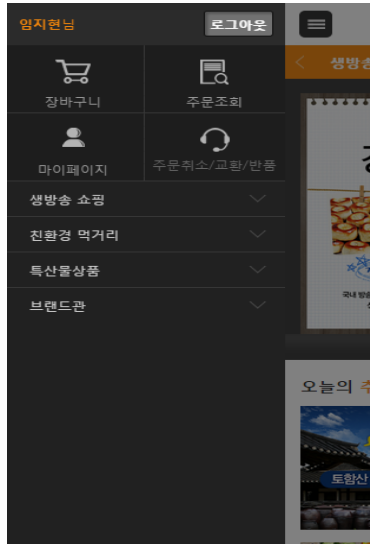


Figure 15. A Screenshot of the Option Menu

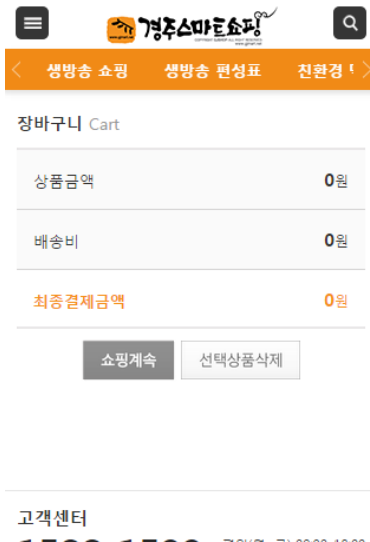


Figure 16. A Part of the "Cart" Page



Figure 17. A Part of the "My Page" Page

6. Conclusions

We introduced our design and implementation of a mobile app that specializes in regional products. This system is open to content providers and to the local Internet TV system. A content provider is a person or an organization that is authorized to upload content to the system. A producer, a manufacturer and a distributor can be a content provider. They take videos and pictures of their products edit them and upload them to the system. Leveraging the infrastructure of the local Internet TV system, this system also streams out a sequence of video clips advertising product items. The system is flexible to system reconfiguration. If any one of the servers is replaced by a new one, then we have to update the configuration file. However, we do not have to change any program code.

Acknowledgment

This work was supported by Basic Science Research Program through the National Research Foundation of Korea(NRF) funded by the Ministry of Education (NRF-2011-0006942), by ‘Development of Global Culture and Tourism IPTV Broadcasting Station’ Project through the Industrial Infrastructure Program for Fundamental Technologies funded by the Ministry of Knowledge Economy (10037393) and the Dongguk University Research Fund of 2015.

References

- [1] L. A. Belardinelli, L. Marzi, F. Frosini, R. Miniati and E. Iadanza, “Careggi Smart hospital: A mobile app for patients, citizens and healthcare staff”, IEEE-EMBS International Conference on Biomedical and Health Informatics (BHI), (2014), pp.125-128.
- [2] R. Sadafule, “Mobile app development for the Indian market”, IEEE Software, vol. 31, no. 3, (2014), pp.17-20.
- [3] C. Giloth and J. Tanant, “Reconstitution of the Labyrinthe of Versailles as a Mobile App”, 18th International Conference on Information Visualization, (2014), pp.269-274.
- [4] D. Kim and J. Jung, “CyberOffice: A Smart Mobile Application for Instant Meetings”, IJSEIA, vol. 8, no. 1, (2014), pp. 43-52.
- [5] H. Khalid, E. Shihab, M. Nagappan and A. Hassan, “What Do Mobile App Users Complain About?” IEEE Software, vol. 32, no. 3, (2014), pp. 70-77.
- [6] P. Zervas, I. Kalimeris and D. Sampson, “A Method for Developing Mobile Virtual Laboratories”, IEEE 14th International Conference on Advanced Learning Technologies, (2014), pp.8-10.
- [7] D. Lee, “Designing the Multimedia Push Framework for Mobile Applications”, IJAST, vol. 32, (2011), pp. 117-124.

- [8] Y. Son and S. Oh, "Design and Implementation of a Compiler with Secure Coding Rules for Secure Mobile Applications", IJSIA, vol. 6, no. 4, (2012), pp. 201-206.
- [9] Y. Ham, D. Moon, H. Lee, J. Lim and J. Kim, "Android Mobile Application System Call Event Pattern Analysis for Determination of Malicious Attack", IJSIA, vol. 8, no. 1, (2014), pp. 231-246.
- [10] H. Kim, "Hybrid Model Based Testing for Mobile Applications", IJSEIA, vol. 7, no. 3, (2013), pp. 223-238.
- [11] H. Kim, "Designing of Framework for Mobile Applications Assets Management", IJSEIA, vol. 7, no. 3, (2013), pp.387-398.
- [12] S. Yoon and Y. Ho, "View Synthesis and Coding of Multi-view Data in Arbitrary Camera Arrangements Using Multiple Layered Depth Images", Journal of Multimedia and Information System, vol. 1, no.1, (2014), pp. 1-10.
- [13] Yim J., "Design of a Mobile Shopping App for Regional Products", Advanced Science and Technology Letters, vol. 66, (2014), pp. 53-56.

Authors



Kyubark Shim, He received the M.S. and the Ph.D. degrees in Statistics from the Dongguk University in 1986 and 1993, respectively. He is a Professor in the Department of Information and Statistics at Dongguk University at Gyeongju, Korea. His current research interests include Computational Statistics, Reliability Test, and Analysis of Statistical Data.



Jaegeol Yim, He received the M.S. and Ph.D. degrees in Computer Science from the University of Illinois at Chicago, in 1987 and 1990, respectively. He is a Professor in the Department of Computer Science at Dongguk University at Gyeongju Korea. His current research interests include Petri net theory and its applications, Location Based Service, AI systems, and multimedia systems. He has published more than 50 journal papers, 100 conference papers (mostly written in Korean Language), and several undergraduate textbooks.