CartRegister

A Grocery Shopping Web Site, Database, and In-Store Customer Assistance Device

Heath Hensley, Oscar Salas, Michael Sieben

University of Central Florida, Department of Computer and Electrical Engineering, Orlando, Florida, 32816-2450

***Abstract* — This paper presents a system designed to facilitate the grocery shopping experience by providing a website, product database, and an in-store device that presents the shopping list and aisle locations of the selected items to the customer at time of purchase. The device is solar powered with a rechargeable battery back-up, has an LCD display, and communicates with a store server via a wireless interface, making the device completely portable. The store server contains the product inventory, pricing, and aisle location data, and interfaces over the Internet to download the customers' shopping lists based on club card identification barcodes scanned by the barcode reader attached to the Smart Cart device. Shopping without a club card I.D. is also facilitated.**

***Index Terms* — Bar codes, client-server systems, indoor radio communication, inventory control.**

I. Introduction

CartRegister is a system designed for the convenience of the supermarket customer. It will allow the customer to interact with an Internet based application to produce a shopping list based on the customer's shopping needs, plus recipes available on the website. The customer will be able to search for recipes using keywords such as the type of dish or ingredients. The website will facilitate the creation of the shopping list by adding the items from selected recipes, plus allow the customer to add other items as needed. The customer will login using an identifier from the store club card. This will link the shopping list to the customer in the web server database. At the store, the customer will be provided with a device called the Smart Cart, which is a microcontroller based box attached to the shopping cart. The customer's club card can be scanned with a bar code reader that is attached to the Smart Cart, and if the customer has a shopping list in the database, the information will be loaded onto the database server in that store. The server will communicate wirelessly with the Smart Cart to display each item and its aisle location to the customer, and guide the customer through the store via the shortest route. The customer will use the bar code reader to scan each item as it is put into the cart, and price information will be displayed. The information can be used by the server to keep track of inventory and to automatically check out the customer at the end of the shopping experience. Although these last two items are beyond the scope of this project, they could easily be added with additional time and resources.

II. Smart Cart Device

The hardware portion of the CartRegister system consists of the Smart Cart device, which displays the user's shopping list information, and has an attached bar code reader to allow the user to scan each item as it is put into the shopping cart. The Smart Cart displays information including item description, size, price, items to go, number of items scanned, and total price on an LCD display. There are also three pushbuttons for scrolling up and down through the shopping list, and for deleting an item. The Smart Cart functionality is centered on a microcontroller, the Microchip PIC18F2455.

To enable complete portability, the Smart Cart device is powered by a solar panel, backed up by rechargeable batteries. The power budget shows that the batteries need to provide a maximum current of 250 mA at 3.6 volts. While it is anticipated that the solar panel on the Smart Cart will be almost constantly illuminated, an allowance will be made for the light to the solar panel to be blocked for a maximum of one hour. Were this to occur, the batteries would need to provide 250 mA-hrs of back-up capacity. Several types of rechargeable batteries were researched, and Nickel-Cadmium was chosen for its ability to withstand overcharging, making it preferred over Lithium and Nickel Metal Hydride types. This is because the batteries are on a constant trickle charge when the device is powered by the solar panel. A battery pack consisting of three NiCad batteries with a total of 3.6 volts and 1200 mA-hrs capacity is used.

The Smart Cart uses a wireless device to communicate with the store server. The device chosen is the XBee module, produced by Digi International of Utah. This was chosen for its low power consumption, low cost, and plug-and-play capability. The XBee module is packaged on a small circuit board. This includes a transceiver and an antenna. At the Smart Cart end, this circuit board plugs into the main PCB, and interfaces to the microcontroller via the UART. On the server end, an interface board purchased from Digi interfaces the XBee module with the store server's USB port. The supply voltage is 2.8 to 3.4 volts, and an LM3940 3.3 volt regulator is used on the Smart Cart PCB for this purpose. The indoor transmission range is specified at up to 30 meters. Peak current for the

XBee is 40mA during transmit or receive. Two XBee modules are shown in Fig. 1 below. The XBee Pro shown has a wire antenna option, which is the antenna used on the XBee modules user for CartRegister.

Fig. 1. XBee and XBee Pro RF Modules
(Reprinted with permission from Digi website [1])

The CartRegister system includes a bar code reader to allow the customer to identify each item that is placed into the shopping cart, which causes the items to be checked off the shopping list and added to the shopping cart inventory. Bar codes are already used to identify almost all products in the supermarket. The device used is the Econoscan from ID Tech. This can read all standard code types up to a maximum distance of 4.7". Reading is accomplished by pulling a trigger on the handle. Operating current is only 85mA at 5 volts, and the cost is only $89. The Econoscan is shown in Fig. 2 below.

Fig. 2. ID Tech Econoscan Bar Code Reader
(Reprinted with permission from ID Tech website [2])

The Smart Cart has a 20 character by 4 line LCD display on the top panel. There are four main screens to provide information to the user. The first is the Welcome Screen, shown in Fig. 3.

W	E	L	C	O	M	E		T	O		P	U	B	W	I	N	X	I	E
			S	C	A	N		C	L	U	B		C	A	R	D			
	O	R		S	C	A	N		F	I	R	S	T		I	T	E	M	
						T	O		B	E	G	I	N						

Fig. 3. LCD Welcome Screen

The next screen is the Normal Mode screen. This provides information for the next item on the shopping list. This is shown in Fig. 4.

	I	T	E	M		9						2	4		T	O		G	O
G	R	E	E	N		B	E	A	N	S		1	6	O	Z		3	X	
A	I	S	L	E		5													
8		I	T	E	M	S		T	O	T	A	L		$	4	5	.	2	3

Fig. 4. LCD Normal Mode Screen

The third screen is the Three Second Display screen. This is displayed for 3 seconds after the user scans a product, to provide the product information before displaying the next item. See Fig. 5.

G	R	E	E	N		B	E	A	N	S									
D	E	L		M	O	N	T	E		1	6	O	Z						
F	R	E	N	C	H		C	U	T										
$	1	.	4	9			E	A	C	H									

Fig. 5. LCD Three Second Display Screen

The final main screen is the Delete screen, displayed when the user presses the Delete button, shown in Fig. 6 below.

S	C	A	N		I	T	E	M		T	O		D	E	L	E	T	E	
									O	R									
P	R	E	S	S		D	E	L	E	T	E		B	U	T	T	O	N	
			A	G	A	I	N		T	O		E	X	I	T				

Fig. 6. LCD Delete Screen

The power system for the Smart Cart device consists of a solar panel powering the circuitry and components, and also keeping a charge on a battery pack as a backup power source for times that the solar panel is blocked or not enough light is available. The charging circuit for the battery pack consists of a diode and a resistor. The selected battery pack is the CD-4/5Sc3iWT, which has a capacity of 1200 mAH. To be able to keep a near constant trickle charge on the battery pack, all of the output of the solar panel is input to a MAX1674 device [3]. It takes an input of between 0.7 and 5.5 volts, and converts it to a regulated output of 5.0 volts at up to 1 A output current. The 5.0 volt output from this device is used to trickle charge the battery pack through the charging circuit. The 5.0 volt output of the MAX1674 also powers the bar code reader through pin 9 of the RS-232 port, the LCD, and the other PCB components. The MAX1674 chip has a Low Battery Input on pin 2 (LBI). This is connected to the output of the solar panel. When this input falls below 1.3 volts, an internal reference comparator puts a low output on pin 3 (/LBO) Low Battery Output. LBO is an open collector output, and is pulled up to the output voltage (+5V) through a 10K resistor. This output is used to drive a small DC relay to switch the input of the MAX1674 from the solar panel to the battery pack. Since the LBI input remains connected to the solar panel, the MAX1674 will run on power from the battery pack until the solar panel output goes back above 1.3 volts, at which time the relay will switch off input from the battery pack and over to input from the solar panel. See Fig. 7.

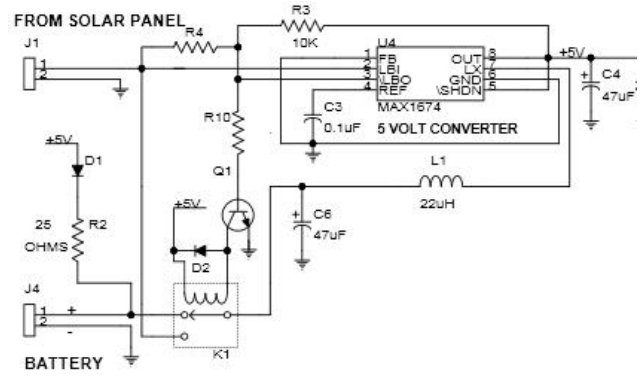

Fig. 7. Smart Cart Power Circuit

The Smart Cart device is housed in a plastic enclosure, to allow RF transmission without protruding the XBee antenna outside of the box. The enclosure is just large enough to accommodate the solar panel, the LCD display, and the three pushbutton keypad on the top panel. The connector for the bar code reader is located on the side of the box. See Fig. 8 below for a rough layout of the top panel.

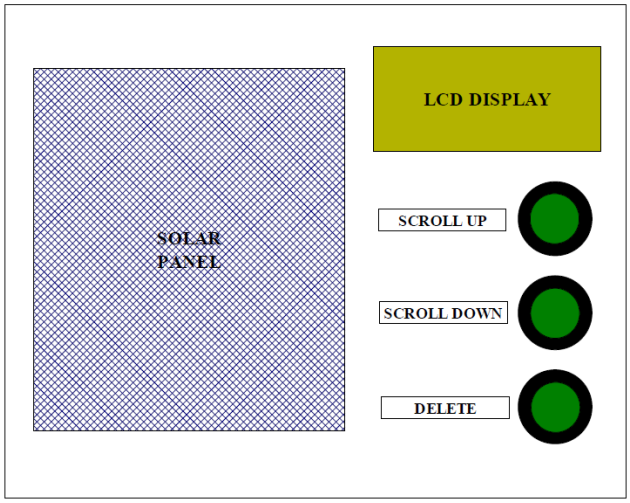

Fig. 8. Smart Cart Top Panel Layout

The Smart Cart device uses a microcontroller as the center of all functionality, including communication with the store server via the XBee device, displaying information on the LCD, accepting inputs from the three pushbuttons, and accepting data from the bar code reader. The microcontroller is a Microchip PIC18F2455. This was chosen for its low power consumption, ease of programming and development tools, and high speed capability. Development and programming were accomplished using the Microchip MPLAB IDE integrated development environment, the Microchip MCC18 compiler, and the PICkit2 programmer. The program was written in the C language. The MCU used is a 28 pin DIP package, and the pin-out is shown below in Fig. 9.

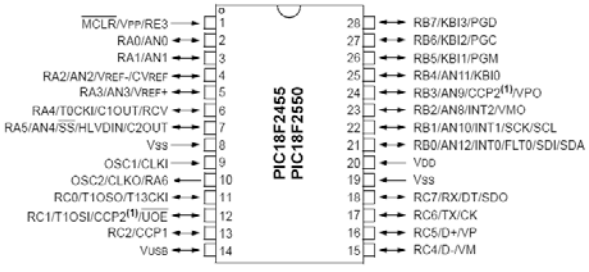

Fig. 9. PIC18F2455 Pin-Out [4]

III. Web Application

Scalability was a focus point of the team when entering into the design phase of the web application of this project. Knowing that the web application portion of this project could take on any number of feature enhancements very easily and also knowing the time limits that this project was under, the team decided that it would be best to focus on the core functionality first with scalability in mind. The team also decided to value quality and completion over a wide variety of features.

A Database Diagram is used to show the relational database structure of the needed tables. Fig. 10 shows a mock up of the CartRegister database diagram. In this diagram are the essential tables that the CartRegister web application uses to store its data.

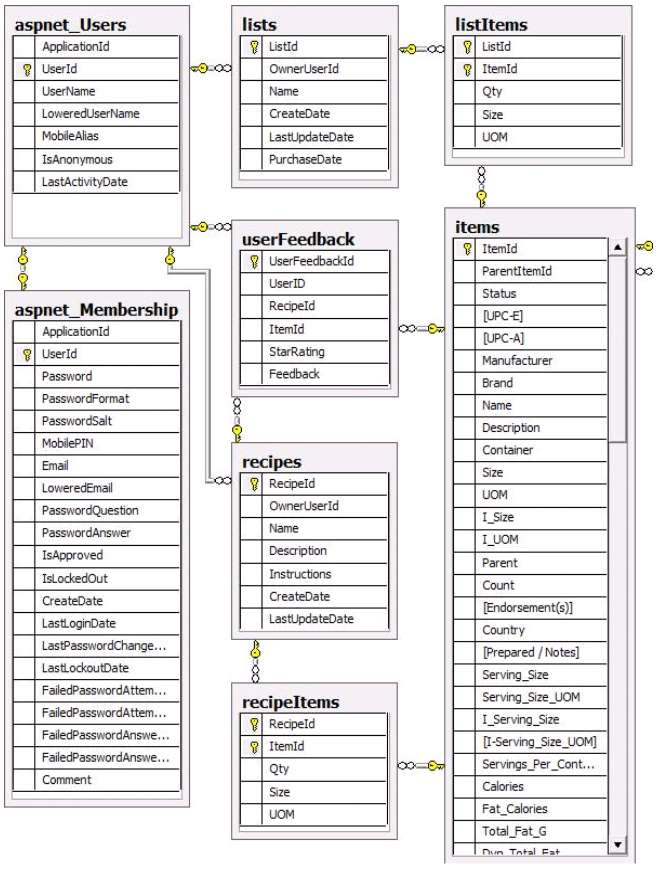

Fig. 10. Database Diagram

Some elements of the database diagram should be discussed in more detail starting with the two tables prefixed "aspnet_". These tables have a naming convention slightly different than the others due to the fact that they are apart of a packaged feature in the ASP.NET Framework called Membership. TheASP.NET membership feature is basically a way to validate and store user credentials. This feature main purpose it to manage user authentication to the web application but it also has functionality for things such as creating new users, managing passwords, and role management. A few highlights of this data structure that are worth mentioning is first that it is extremely scalable. With the membership structure and robust items, lists, and recipes tables this database can easily grow in the future because it has a solid foundation. One of the things that makes it a solid foundation is it is all build on unique identifiers that are not defined by the end user. With the use of auto incremented columns it simplifies the data update process tremendously. Finally, this data structure takes advantage of a new data type that SQL introduced in SQL 2008 and that is hierarchicalId. This is a built in data type that simplifies many common issues designers face when dealing with tree type structures.

A conceptual web site map is commonly used to graphically communicate the flow of one page to the next in a website. In Fig. 11 one can see the proposed web site map for the CartRegister web application. Objects with a dashed outline signify pages and features of the system that are planned but might not necessarily make it into the initial release of the project.

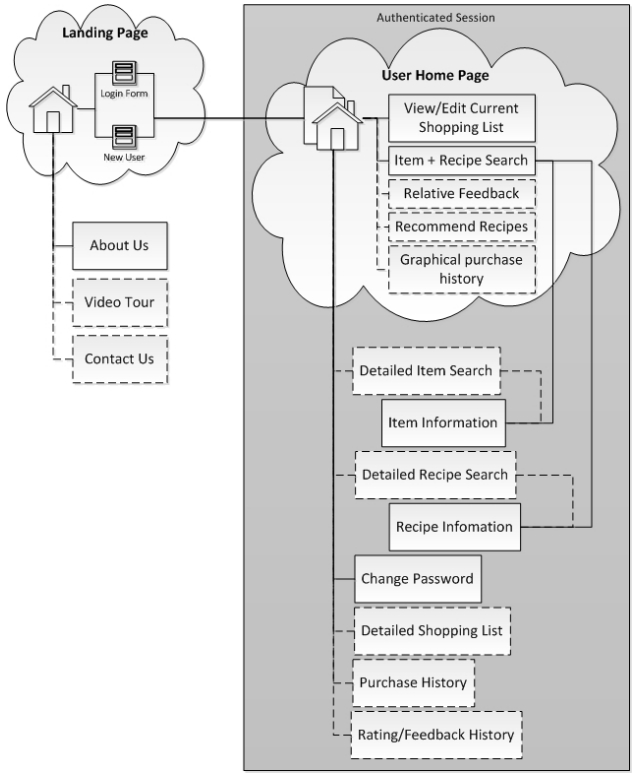

Fig. 11. Website Map

In the upper left of Fig. 11 one can see a cloud object labeled landing page. This has become a very popular practice in web applications to have a landing page that not only has access avenues to the authenticated portion of the application but it also promotes the web application for those that do not know what it is. Therefore when an individual goes to CartRegister.com they will have the opportunity to learn a little about the application before they sign up. The landing page also serves the purpose of having key phrases about the application so that web crawlers can index the web page for search engine purposes. This is very important in the world of websites. Marketing a website lives and dies by the search engines. The more key words and phrases that can be picked up on the landing page the better ones chances are of getting to the top of the organic search (meaning not pay per click). The other purpose of the landing page that one can see from the figure is to house both the log-in form and the first time sign up. This has also become a very common practice in web applications simply because it is easier for both the current users and potential users to navigate. There is no need to click through a series of pages just to sign up or log-in. Those extra tedious steps can easily drive potential users away. A prototype of the landing page can be found in the following figure (Fig. 12).

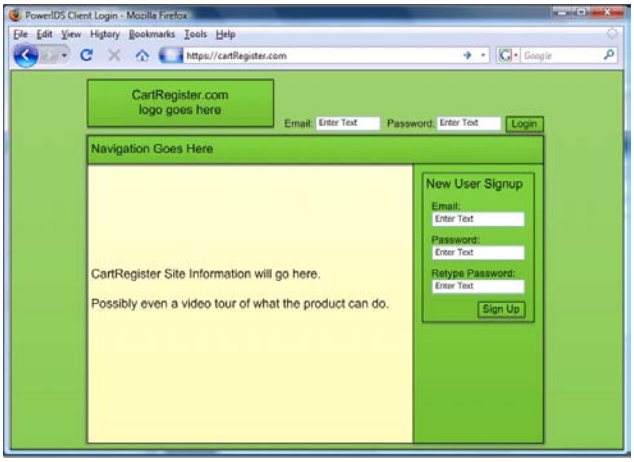

Fig. 12. Landing Page Prototype

One can see in the upper right of Fig. 12 that there are two text boxes and a button that is used to log-in to the web application. To the left of that there is a place holder for the logo. There is also a navigation menu bar that will be used to get to other areas of the site, for example video tour, contact us, and about us. One can also see that these are the pages that are listed under the landing page in Fig. 11. Other pages could be linked to from this menu bar in the future, pages such as support for the product and frequently asked questions. The main light colored area in Fig. 12 will be used for the explanation and promotion of the product and website. This area will contain all the information a potential user would want to know about the application so they can decide if they want to sign up. Once the user has made their decision they can simply use the form on the right side of the page to sign up and get started using the application.

Now back to Fig. 11 one can see that the left side of the figure has now been covered. The user can log-in from the main landing page and now enter into the authenticated portion of the application signified by the big gray box. This area can only be accessed by users that have an account with the CartRegister website. The first page that the user will see once logged in is their home page. This is symbolized by the large cloud labeled "User Home Page". This cloud contains several components that will simplify the usability of the application and hopefully not clutter the user interface. Experimentation will need to be done in order to insure that this home page is not too overwhelming. As one can see from the figure the user's home page will contain their latest shopping list, search functionality for both items and recipes, and time permitting many other useful and interesting information that is pertinent to the user. These things would include recommended recipes based on their purchase and search history. The system would be able to detect a pattern of items that the user frequently consumes and from that grab recipes that contain those items in their ingredients list. The home page could also contain feedback that would be relative to that users purchase history. This could be something like if the user posted their own recipe and someone else gave it a star rating or wrote something about it this would be displayed on the user's home page to make them aware of it. The final piece that could be added to the user's home page would be some sort of graphical representation of their purchase history. This could look something like a month to month bar graph showing the users purchased amounts and total item counts. It could also bring in nutritional facts showing an average daily calorie, fat, or sugar intake based on purchased items from month to month. Fig. 13 is a prototype of what the user's home page could potentially look like. There are a few items worth mentioning, and the first is the user's current grocery list that is displayed on the right side of the screen. Being that the primary goal of the application is to easily create grocery lists the team has taken a very simple approach to this process. The user's active grocery list will be displayed on every page in the authenticated portion of the application. That way at any point when the users thinks of an item that they need to add to their list they can easily add it without having to navigate away from the page they are viewing. Another feature needing explanation is the process of adding an item to the list.

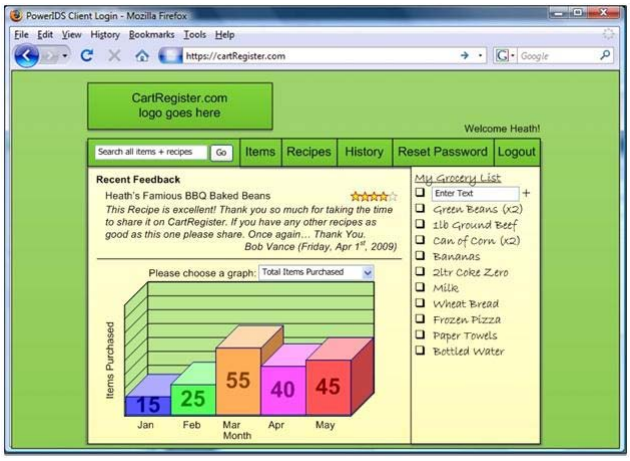

Fig. 13. User Home Page Prototype

At the very top of the list one can see that there is a text box and an add "+" button. Users will be able to start typing the item that they would like to add to their list and the application will use AJAX (Asynchronous Javascript + XML) technology to query the database and bring back a list of the most relevant items based on what they typed. The list of items that are brought back will be displayed directly below the text box but over top of the list so that the user can easily use either the tab key or up and down arrows to pick the item that they were intending to add to the list. This list will contain approximately the top ten most relevant items and as the user continues to type the search will be narrowed down to the exact item they are looking for. Once the user hits enter to add the item AJAX is used again to submit the add request to the server without having to post back the entire page. At this point a row has been added to the listItems table in the database so that when the whole grocery list on the right side is repopulated (once again using AJAX technology) this new item will be displayed. The item could possibly even flash for a second to make it obvious to the user that this item has been added to their current shopping list.

Another feature worth mentioning in Fig. 13 is the search text box in the upper left of the page. With this feature the user can type in any search whether it be for an item or recipe and the system will handle this very similarly to that of a Google search. After the user clicks the "Go" button it will take them to a search results page where they will be able to see a list of the most relevant items/recipes at the top. From there the user can click on the item/recipe and it will take them to the respective form contain all the details of that item/recipe. Since the one search feature is contained in the menu bar this feature will also be available on every page therefore at any moment when the user thinks of something they need to search for they have the option up in the menu bar. This is very similar to most modern day browsers that contain a search feature typically in the upper right side of the browser window.

A final note on Fig. 13 is that the navigation bar itself that contains many options for the user. Starting on the left is items and recipes. This option will take the user to an advanced search page for items or recipes. Contained on these search pages will be the standard paged table of all the items (or recipes) in the system. In this table there will be sortable columns such as name, star rating, and nutrition facts like calories, sugar, and fat. The primary purpose of these tables will be to display large quantities of data in a way that is easy to interpret and navigate and it will also provide advanced search functionality. The item or recipe name in these tables will be a hyperlink that takes the user to the respective form. The detailed item form will contain all the available information about the item. An example item form is shown in Fig. 14. The detailed recipe form will look very similar to the item form except it will contain instructions for making the recipe and a list of all the ingredients with the option of adding the items to the user's current list on the right side.

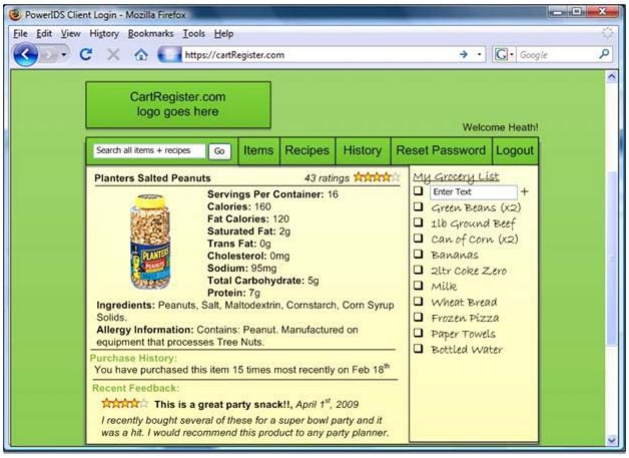

Fig. 14. Detailed Item Form

Getting back to the menu bar the next available menu option is History. History will take the user to a page listing out the users purchase and list history. The information on this page will become very valuable and informative to the user over time. By searching and viewing purchase and list history the user will be able to see trends in areas of spending and dietary habits. For example if the user decides to start clipping coupons out of the Sunday paper to save money they will be able to see a trend that they are still getting the same amount of items but at a lower price. They could also see things like average calorie intake per week based on the products they purchase. The next menu option is Reset Password. This

one is very self explanatory. It will take the user to a page where they will have the option of putting there old password in along with a new one and the system will reset the password. And finally, the last option on the menu is logout. No explanation is needed here.

IV. Web server, Store Server, and Device Interaction

As explained in previous sections, the Smart Cart device uses the XBee wireless communication device to send and receive data. It would be extremely inefficient for the device to send and receive all its data over the Internet. There must be a middle man that can bring down the user's list information once from the web server over the Internet and store it locally on the store server in order to provide an efficient medium for the device to access its needed data. Fig. 15 gives a graphical representation of how the device and the two servers are going to interact.

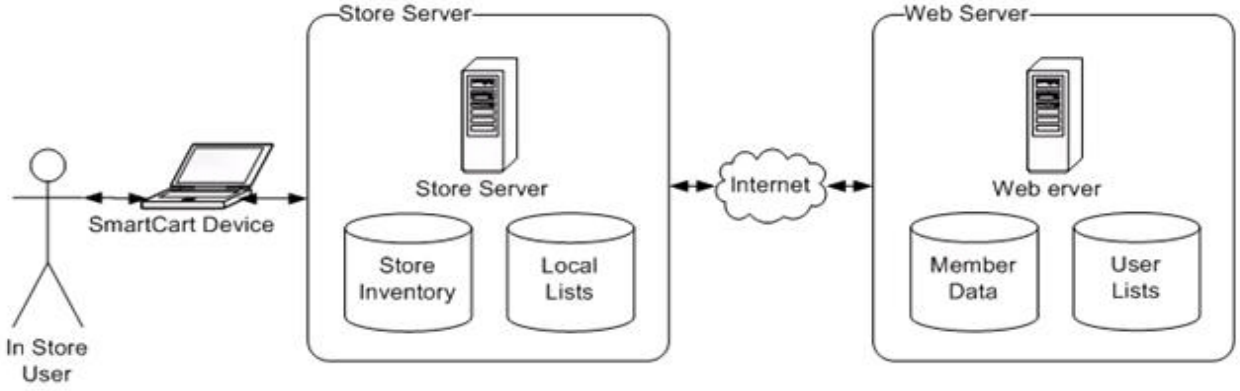

Fig. 15. Smart Cart and Servers Interaction

In Fig. 15 one can see there are five components on the x axis. Starting on the left there is the In Store User that will be using the Smart Cart device that is attached to the cart. This will be the only human interaction in the system. The next component is the Smart Cart device. Contained in this device is a wireless transceiver that has the ability to send and receive data with the next component in line which is the Store Server. The Store Server acts as the middle man in this system between the device that the user uses to view their shopping list and the data for that shopping list which is stored on the Web Server. The Store Server also contains the store inventory database with current prices which was another deciding factor in this design. The Web Server could not possibly keep track of all the different stores inventory and pricing information. It would simply be too much information and too cumbersome to keep up to date. With the likelihood of frequent inventory and price changes it only makes sense to keep that information local for the store and attempt to integrate with it with the CartRegister system. This leads into the other database shown under the Store Server, the Local Lists.

When the user first scans their user identification card, the system uses that ID to retrieve their active shopping list from the CartRegister web site. This retrieval is accomplished with a technology called web services. Once a connection has been made with the web server using the users ID, the users list is be brought down to the local Store Server. At this point there is no need for any other communication over the Internet until the user goes to check out. Now that the users shopping list is on the store's server and is uniquely identified with the user and therefore the users Smart Cart device all communication and processing of the list can be done locally. So when a user scans a item it will not only retrieve valuable information about the product and display that to the user, but it will also check and see if the item can be marked off their shopping list. With this design there will be two lists maintained for each user in the store. One list will contain the items that the user currently has in their shopping cart and the other will keep track of their current shopping list. By pragmatically comparing the two lists to each other the system will be able to communicate to the user how many and which items still need to be purchased on their shopping list. This concept is further explained by Fig. 16.

The final two components of Fig. 15 are the Internet and the Web Server. The Web Server is what is hosting the CartRegister web application that has been discussed in detail in previous sections. This server also hosts the web service that communicates with the Store Server to send and receive information about the users and their lists of items. It was mentioned earlier in this section that the Store Server would communicate once to get the users current shopping list when the user enters the store, and communicate again over the Internet when the user checks out, but this final step was not explained. When the user

goes to check out of the store there is a lot of valuable information that is only on the Store Server. Therefore at the point of checkout, when the final item quantity and price values are calculated, the Store Server will once again communicate with the web server via the web service and send a list of all the items that the user purchased with their price value. This information will then be used in the user's online account to keep track of purchase history and shopping trends.

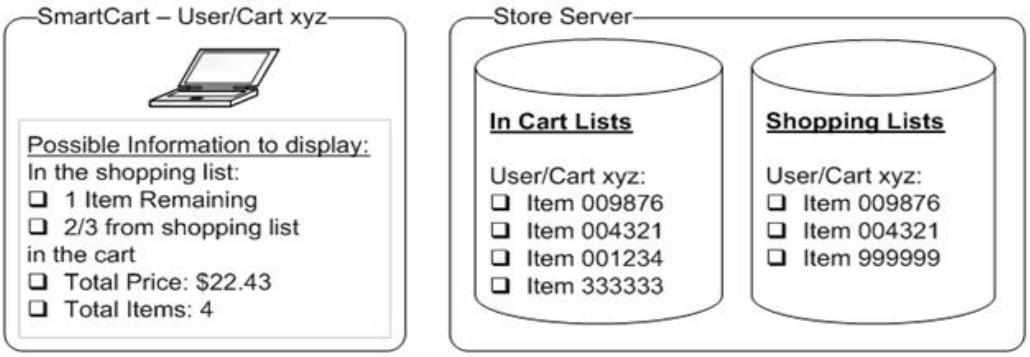

Fig. 16. Smart Cart and Store Server

ACKNOWLEDGEMENTS

The project team would like to thank Dr. Samuel Richie for his guidance, and would also like to thank Advanced Circuits (4PCB.com) for providing their $33 student pricing for a printed circuit board with no minimum quantity, as well as their very quick delivery time. The team would also like to thank Mr. Gregg London for providing a sample UPC database at no charge.

THE PROJECT TEAM

Heath Hensley, a senior computer engineering student at the University of Central Florida scheduled to graduate in December 2009. He is currently the Director of Software Development for Innovative Data Solutions, Inc., a software company based in Orlando, Florida. He is passionate about software development and specifically data driven web applications.

Michael Sieben, a senior electrical engineering student at the University of Central Florida, will graduate in August 2009. He previously worked in the engineering field, helping design and test eye tracking systems for laser vision correction surgery devices for Alcon Laboratories in Orlando, Florida. He will continue to pursue a career in electrical engineering.

Oscar J. Salas, a senior Electrical Engineering student at the University of Central Florida, will graduate in August of 2009. He also holds BA in Marketing from the University of Puerto Rico, Aguadilla. Currently works as a Co-op Student at Progress Energy Florida, and will pursue a career in the utility industry. He is a member of SHPE, Society of Hispanic Professional Engineers.

REFERENCES

[1] Digi International, "2.4 GHz and 900 MHz RF Products Overview," [Online]. Available: http://www.digi.com/technology/wireless/products.jsp[Accessed: April 16, 2009]

[2] ID Tech, "Econoscan Bar Code Scanner," [Online]. Available:http://www.idtechproducts.com/pdut_detail.asp?F_PdutID=35&F_CateID=4&F_SubCateID=6 [Accessed: April 18, 2009]

[3] Maxim, "MAX1674/MAX1675/MAX1676," [Online]. Available:http://datasheets.maxim-ic.com/en/ds/MAX1674-MAX1676.pdf [Accessed: July 23, 2009]

[4] Microchip, "PIC18F2455," [Online]. Available: http://www.microchip.com/wwwproducts/Devices.aspx?dDocName=en010273 [Accessed: April 24, 2009]