AJAX-BASED BUSINESS SOLUTION : CASE STUDY POS (POINT OF SALE) SYSTEM

Dodi Kuswono, Edy Satriyanto, S. Si
Department of Information Technology
Politeknik Elektronika Negeri Surabaya - ITS
Campus PENS-ITS Surabaya 60111 Sukolilo Keputih
Tel (+62) 31-5947280, 5946114, Fax. (+62) 31-5946114
Email: dodikuswono@gmail.com

Abstract

Retailers find themselves up against fierce competition. They must fight for every sale, and work hard to build customer loyalty and protect already slim margins. Today, an increasing number of smaller retailers understand the urgent need to catch up to larger players to remain competitive. They also recognize the important role that IT investments play in organizations' strategic decision-making and operational efficiency in all areas of the business, including point of sale, supply chain management, and inventory. The savvy retailer knows that POS (Point Of Sale) data and functionality has quickly become critical to business rather than a mere convenience.

In recent time Ajax based applications have become very popular. Ajax is a new model for web applications to provide more responsive and faster user interfaces resembling more closely to dekstop applications. Typical usage areas are user input validation without page submission, integrating small elements from several servers on a single page, and simulating push-services. Especially the latter are promising for enhancing web applications and for realizing them directly in browsers without plug-ins or additional software. Many frameworks and libraries (open source or comercial) are available which support Ajax development.

In this final project, we will integrate some open-source Ajax framework to build low-cost, interactive and integrate POS (Point Of Sale) systems which is accessible to a wide retailer through the Internet. This is we call as iPOS. We hope it's will become a solution for retailers to run their business more efective and effisien.

***Keyword* : *Web*, Ajax.**

1. INTRODUCTION

In a field of any retail business, data collection is very important articles. Neatness in the preparation and collection of goods in the warehouse will determine the neatness of the system is running at retail. Goods out of the barn and will be sold must be recorded properly organized and how many items are out so that the store did not have vacuum supply (stock). Similarly, the entry of goods into the warehouse must be properly recorded so as not to cause the accumulation of too much inventory and impact expire on goods (if taken) is not sold it.

Manual data recording allows recording errors. Mistakes - mistakes that have occurred among other unrecorded transactions, recording error, the number of transactions that have a repetition of recording, the evidence is missing transaction. By using the manual information systems also require the use of paper so much, the use of books as a record of transactions will become more and accumulate and require large storage space. In addition to the manual information systems, to perform the necessary search data, take a long time to find the transaction that u drunk so much.

To overcome this - something that allows the various errors and irregularities in the handling of inventory in the warehouse, there must be a system of managing traffic information and data. The system will be made should ensure that the management should run without errors. One system that allows the realization of this is a computerized information system as a system called POS (Point Of Sale).

With a computerized POS system, diverse data and large amounts can be processed quickly and precisely in a digital database that can be done saving time. Because with the support of a structured database, the search process, processing and calculation can be done quickly. With this computerized system is also saving on paper usage

can also be done and paper waste can dikurangi.banyak benefits derived by means of this sophisticated technology. Each transaction can we make a final report for evaluation and business development.

For retail stores that have some branches that spread in a number of areas required means of communication that connects them all. To build the necessary means of communication was a big fund, the other alternative is to use the Internet as a means of communication that connects all the existing POS system.

Lately, a new trend emerged in the development of web applications on the Internet known as Ajax. Ajax is a new model for inter-face web applications more responsive and faster like a desktop application.

Because of the things above, then the End Task is selected to implement Ajax in the manufacture of POS systems are functional and interactive.

2. SUPPORTING THEORY

2.1 Ajax

Ajax *(Asynchronous JavaScript and XML)* is a technique and way of thinking in the development of web applications for creating interactive web applications, responsive, fast and efficient. Ajax makes h a web page does not have to always be *loaded* again each time a user requested changes. The technology used is *(X) HTML & CSS, Document Object Model,* XML & XSLT, e XMLHttpRequest, and JavaScript. Thus, Ajax is not the technology itself, but a term that refers to the number of standard technologies that already exist

One of the most common aspects are discussed from Ajax is XMLHttpRequest object, which is a component of the browser to perform the asynchronous connection. Called asynchronous because the user can communicate with the server without reloading the page to capture what is usually required. With Ajax, the data retrieved from the server that is only really needed it, not the entire page.

More detail, the traditional web works like this: Action from the user triggers an HTTP Request to a web server. Servers perform a number of processes that involve Web Server Script - retrieve data from the database, perform calculations, to check access rights - and then returned to the client web page. However, how this work is not suitable for an application.

Above approach is not satisfactory from the user side. While servers perform various processes, what's users do? Just wait. And the more steps are needed more, more waiting time from the user.

Ajax application eliminates the unfavorable interaction of the above by presenting the user of Ajax Engine intermediary between users and servers. That's like adding a layer in the application.

While the web page is *loaded,* at the beginning of the session, the browser had an *Ajax engine* written in JavaScript and stored in a hidden frame. This machine is responsible for rendering the user interface and communicate what users do with the server. Ajax engine allows the user interaction occurs in asynchronous applications - independent of communication with the server. So the user does not have to start with a blank browser window, waiting for the server to do something.

2.2 Ajax Programming

Ajax programming involves programming on the client and server side. For the client side using JavaScript, HTML or XHTML and CSS or XSLT. While any server side scripting server can be used such as PHP, ASP, JSP, Perl, or ColdFusion.

Client side, hold the XMLHttpRequest object to perform an important role in asynchronous connection. The following will explain the implementation using JavaScript.

Creating instance of XMLHttpRequest object requires a different syntax code for different browsers. For Safari and Mozilla, the calling object constructor can be done by following simple code:

```
var req = new XMLHttpRequest();
```

2.3 Ajax Development

AJAX is the current 'epidemic' on the internet, therefore the adoption of Ajax by the developer for application development has made very quickly. Adoption Ajax for application development can be divided into four levels of adoption are:

1. Snipet Level.

   Is an individual-based Ajax development, where developers find and insert the code snippet into a p Likasi made. Adoption is vulnerable to intellectual property issues.

2. Widget Level.

   Developers usually stay put the code for the user element i n terface (Widget) on the application that created the widget either alone or open-source code from the other side.

3. Framework Level (Client-side framework).

   Developers using a framework that is accompanied by his IDE from another vendor who usually do not open-source.

4. Full Framework Level(Clien – Server Side Framework).

   As the level of integration framework but added with a back-end. L e more easily in the development but is highly dependent on a particular vendor.

Lots of framework-framework, which has been available either pencil or kome nonkomersil r can be selected to develop applications with Ajax. Some things to consider in choosing an appropriate framework are:

1. Documentation of quality. Usually commercial framework i HAVE to better documentation of the open-source.
2. Independency of a particular *back-end.*
3. Compatibility of the browser.
4. Easy to use.

Some frameworks that are available include:

- Dojo Toolkit
- Echo 2
- JavaScript/Ajax Toolbox
- jQuery
- Moo.fx
- Prototype
- Rico
- Sardalya
- Script.aculo.us
- Tacos
- TurboWidgets
- TwinHelix
- Extjs
- Yahoo! User Interface Library
- Atlas.

Among the above framework, a framework that is interesting to discuss are:

- Dojo Toolkit is a toolkit that provides a lot of functionality to create rich Internet applications. Some effects and many controls are available as well as various *advanced features.* Licenses are free to academic under BSD license and is supported by IBM and Sun Microsistems.
- Echo 2 is a framework that licensed Mozilla Public L i cense. Echo 2 uses the Java language to create web applications.
- Prototype. . This framework has the advantage of his kompatibilita s with many browsers and easily integrated with lin g an existing environment. Framework is licensed MIT License.
- Atlas is a set of technologies on the client side and server for ASP applications. NET 2.0. Some components of ATLAS allows to improve the ability of web applications without having to learn the specifics AJAX technology. ATLAS designed to run in the environment. NET is very easy from the side of use. Several other components make use use of JavaScript and the Document Object Model (DOM) which is standard in most browsers.

2.4 Security Issues

Excessive use of Ajax cause security problems are new, including:

1. **Cross-Site Scripting (XSS).** Yahoo's new email using Ajax and has no named Yamanner worm that runs on the browser and able to call the function m e Ajax on Yahoo, the same thing done by Samy worm is able to run scripts on Myxpace.
2. **XML Poisoning.** Accessing the XML document using Ajax also cause problems, because the attacker can use the script to change the structure of XML received by the server.
3. **Malicious Ajax Code.** Ajax work behind when you're working so the user does not know the activities undertaken by the Ajax behind the scenes. This can be exploited by attackers to run malicious code without being seen because the browser display, no problem.
4. **RSS / Atom Injection.** Be careful when taking RSS / Atom from an unknown website, because if in RSS te r Javascript script calls there is evil, then your website can d a night problem.
5. **SOAP injection.** WSDL and SOAP, both are part of the Web Services and using XML documents. Therefore they can be attacked even if pengaksesannya using Ajax.

3. APPLICATION DESCRIPTION

The system will be built is a system POS (Point Of Sale) using a web based Ajax. Here will be a case study for Indomaret POS system.

Indomaret is a company that specialized in retail sales for a variety of daily needs. The company has three parts. The first part is the central office that serves as the controller of the company's activities. The second is the warehouse that serves as a place of purchase, storage, and delivery of goods. Then, minimarket served as retail sales. In every minimarket that there was a small inventory, which is used to store goods that have been sent by the warehouse, then sold to the minimarket.

The system is built to involve the client and server components. Server side of web server and database server on the client side while a web browser. To access the user application must use a browser as a client. Users will access a particular web site where the application is installed and should iPOS authenticate login first before you can access the menus on the system.

There are three types of users: warehouse clerk, cashier clerk, and the Admin Center.

Warehouse Officer, interaction with the system as follows:

- Enter the purchase of goods data, such as: code, name, price (purchase, discount, percentage of profits), type, sub type, quantity, supplier, purchase date, purchase type, no. invoices, minimum stock.
- Changing data warehouse goods
- Deleting data warehouse goods
- Doing calculations price of goods
- Displays information storage items, such as: code, name, price, type, sub type, number of rest, suppliers, and minimum stock
- Displays information requests as required goods minimarket
- Displays inventory reports, such as: information goods, purchasing, shipping, returns (suppliers and minimarket), payables / receivables
- Submitting the report to the center
- Displays information from the center, such as: buy, reject, and info etc.

Cashier clerk, interaction with the system as follows:

- Enter the sales of goods data
- Changing of selling goods data
- Removing selling goods data
- Displays information goods sold
- Doing barcoding process prices
- Conducting process cashier transactions
- Doing demand for the goods to the warehouse
- Displays sales reports, such as: daily, periodical.
- Submitting the report to the center
- Displays information from the center

Admin Center, the interaction with the system as follows:

- Displaying warehouse inventory reports
- Displaying reports Minimarket
- Provide information to the warehouse
- Provide information to the minimarket
- Showing financial statements, such as: profit and loss.

4. IMPLEMENTATION

4.1 Server Implementasi.

Implementation of server-side includes two parts, the SQL script to generate a database in MySQL and PHP scripting. PHP Scripting consists of the following modules:

Configuration Module

This module contains the basic configurations required by the other PHP scripts. Among others contains the configuration for database name, database user name, password etc.

Database Connection Module.

This module contains the global variables needed by the other PHP scripts to access the database. This module provides $con variable that can be used to execute database query. This module depends on adodb library.

User Authentication module.

This module is used for user authentication. This module depends on the connection module. The script will query the database to retrieve usernames, passwords, role. If the username and password match, then the script would genarate session for the user in question.

Module Item / Goods / Inventory

This module is divided into modules to process the items (add, update and delete) and modules for managing stock items (re-order, search, outofstock)

Sales Module

This module is to process and manage the new sales that have occurred.

Inventory Module

The Inventory module is provided with facilities to handle receipts, transfers, returns, sales and issues of stock with full stocktake and stock adjustment functionality, providing management control over the quantity and value of stock on hand.

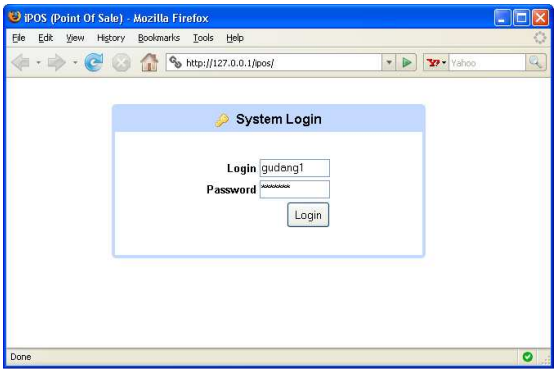
4.2 Implementasi UI Client

Implementation on the client side is used to build a user-interface web interface using HTML as the mark-up language (semantics), CSS for the presentation (display) and JavaScript to help you manage the events and data communication with the server in asynchronous (Ajax).

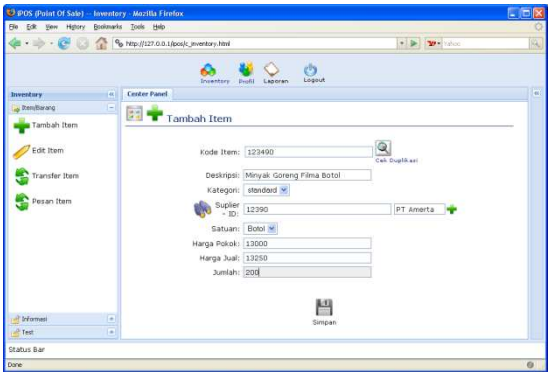
To display user-interface of this application will use the libraries of open-source YUI (Yahoo User Interface) and ExtJS.

5. TESTING

Trial 1, login as user gudan g 1 with the warehouse staff privileges.



Then Gudang1 user then adds a new item data.



6. CONCLUSION

1. Ajax is properly to create Web applications that are responsive and interactive is appropriate.
2. *Drawback* of the development of Ajax-based web applications is the increasing number of files that must be downloaded by the client and the increasingly complex web application development.

7. REFERENCES

1. Ross, Harmes and Dustin Diaz. Pro JavaScript Design Patterns. Apress, 2008.
2. Larry, Ullman. Building a Web Site with Ajax. CyberGarage, 2008.
3. Shawn, M. Lauriat. Advanced Ajax Architecture and Best Practices. Prentice Hall, 2007
4. John, Resig. Pro Javascript Techniques. Apress, 2006.
5. Jeremy, Keith. Bulletproof Ajax. New Riders, 2007.
6. Thomas, A. Powell, Ajax The Complete Reference, McGraw-Hill, 2008.
7. Indrajit, Richardus. Kumpulan Artikel Bisnis di Internet. Jakarta, Indonesia :Renaissance, 2000 .
8. Yank, Kevin. Simply Javascript. SitePoint, 2007.
9. Hoffman, Billy. Ajax Security. Pearson Education, 2008.
10. Andrew, Rachel. HTML Utopia Designing Without Tables Using CSS. SitePoint, 2008.