Smart Client Application for Jewel Market Condition

Aye Nyein Chit, Myint Myint Maw
Universities of Computer Studies, Yangon, Myanmar
ayenyeinchit.11@gmail.com, myintmyintmawt2009@gmail.com

Abstract

In today's world of distributed computing and the emerging business requirements, the demand for smart client application is growing rapidly. They can provide users with a rich and responsive user interface, the ability to work offline and a way to take advantage of local hardware and software resources. And then they can be designed to run on a broad spectrum of client devices including desktop PC. Table PC and hand held mobile devices such as Pocket PC and smart phone. Smart Client give users assessing information remote services with powerful and intuitive client environment and are affection and for increasing user productivity. For that reason, this system is to apply a window form smart client application to support trade organization.

1. Introduction

In the business world trading organization want to know current market condition and features organizations are appeared and support market information to trade organizations. They use information technology to provide customers with accessing to information whenever, wherever they need to getting connection to be online is vital in this case. But most organizations have limited or narrow bandwidth connection. And then they want to use service both online and offline. The possible answer to solve this condition is using smart client architecture.

The term client/server was first used in 1980s in reference to personal computers (PC) on a network. The actual client/server model started gaining acceptance in the late 1980s. The client/server software architecture is a versatile message-based and modular infrastructure that is intended to improve usability flexibility interoperability and scalability as compared to centralized mainframe time sharing computing. A client defined as a requester of services and a server is defined as the provider at services.

A single machine can be both a client and a server depending on the software configuration. The client server model is a form of distributed computing where one program (the client) communicates with another program (the server) for the purpose of exchanging information.

2. Related work

There are many approaches for information retrieval system to provide the Smart Client application to end user. They can provide users with a rich and responsive user interface, the ability to work offline, and a way to take advantage of local hardware and software resources.

Mark Boulter proposed a smart client architecture and design guide for client/server. Smart clients can consume and use different services and data over the network. They are an effective way to retrieve data from many different sources and can be designed to analyze or aggregate the data, allowing the user to make more efficient and better informed decisions. For example, a smart client could use a mapping service to provide details on location and driving directions.

Smart client applications should be as connected as possible and should make use of the resources and services that are available to them over the network. They should not be stand-alone applications and should always form part of a larger distributed solution. At a minimum, a smart client application should use centralized services that help maintain the application and provide deployment and update services.

The connected nature of smart client applications allows them to provide valuable data aggregation, analysis, and transformation services. They can allow users to collaborate on tasks in real time or over a period of time. In many cases, a smart client application can provide portal-like capabilities to the user, allowing disparate data and services to be coordinated and integrated into an overall solution.

Smart clients can be designed to adapt to the host environment, providing appropriate functionality for the device on which they are running.

3. Background Theory

3.1. Client/Server technologies

The term Client/Server was first used in 1980s in reference to personal computers (PCs) on a network. The actual Client/Server model started gaining acceptance in the late 1980s. The Client/Server software architecture is a versatile, massage-based and modular infrastructure that is intended to improve usability, flexibility, interoperability and scalability as compared to centralized, main frame time sharing computing.

A Client is defined as a requester of service and a Server is defined as the provider of services. A single machine can be both a Client and Server depending on the software configuration. The Client/Server model is a form of distributed computing where one program (he Client) communicates with another program (the Server) for the purpose of exchanging information.

The client's responsibility is usually to

1. Handle the user interface
2. Translate the user's request into the desired protocol
3. Sent the request to the server
4. Wait for the server's refuse
5. Translate the response into 'human readable' results
6. Present the results to the user

The server's functions includes

1. Listen for a client's query
2. Process that query
3. Return the result back to the client

3.2. Three tier architectures

The three tier architectures emerged to overcome the limitations of the two tier architectures. In two tier architectures, a middle tier was added between the user system interface client environment and the database management server environment.

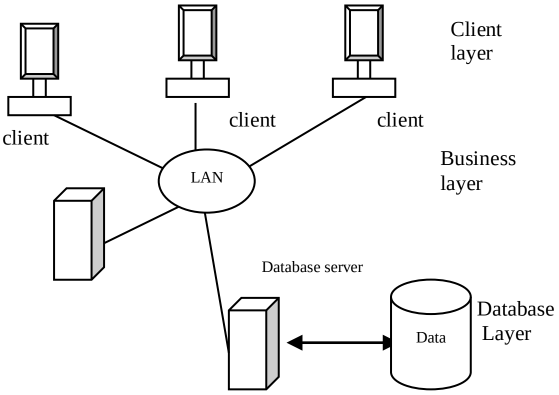


Figure 1. Three tier architectures

There are a variety of ways of implementing this middle tier, such as transaction processing monitors, message servers, or application servers. The middle tier can perform queuing, application execution, and database staging. The three tier architectures have been shown to improve performance for groups with a large number of users and improve flexibility when compared to the two tier approach. Flexibility in partitioning cans a simple as "dragging and dropping" application code modules onto different computers in some three tier architectures. A limitation with three tier architectures is that the development environment is reportedly more difficult to use than the visually-oriented development of two tier application.

3.3. Smart client

Smart Clients are easily deployed and managed client application that provides an adaptive responsive and rich interactive experience by leveraging local resources and intelligently connecting to distributed data sources.

-Rich UI (non-browser based)
-Connects to back end services
-Runs securely on the client
-Supports offline scenarios
-Automatic deployment and updating

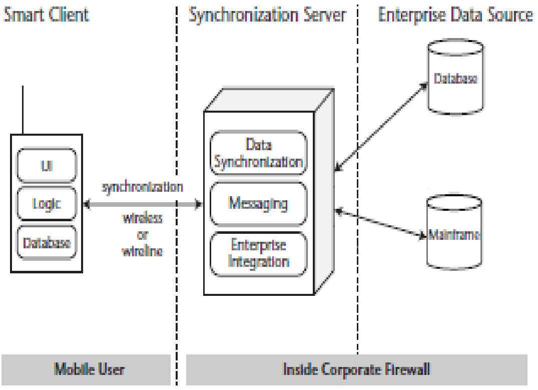


Figure 2. Smart Client architectures

The smart client architecture is illustrated in Figure 2. On the client, you have the user interface, the business logic, as well as a persistent data store. This application communicates with a back-end data source, often through an intermediate synchronization server. The communication stream itself can run either wirelessly or over a wire line connection. Depending on the technology being used, the connection may require an IP-based network or an additional communication layer for the synchronization process.

When this type of solution is implemented, it quickly becomes clear why it is called a smart client solution. By deploying an application to the device itself, you have the ability to give the client application some "smarts," or logic. This logic dictates many aspects of the application. It determines where the application gets it data from (either locally or from a round-trip to the server), how the data is presented and stored, as well as the set of data that needs to be communicated back to the enterprise systems via a synchronization process. In the wireless space, the impact of having business logic on the device is often overlooked as many vendors and developers focus on low-level technical features of a solution rather than satisfying the requirements of the mobile user.

3.4. Types of smart clients

Smart clients vary greatly in design and implementation, both in application requirements and in the number of scenarios and environments in which they can be used. Smart clients therefore can take many different forms and styles. These forms can be divided into three broad categories according to the platform that the smart client application is targeting:

- Windows smart client applications
- Office smart client applications
- Mobile smart client applications

It is common for a smart client application to target one or more of these platforms, depending on the role of the user and the functionality required. Such flexibility is one of the key strengths of smart client applications.

The remainder of this guide concentrates on issues that are common to all three types of smart client applications, rather than providing a detailed explanation of issues that affect each individual category. However, it is useful to briefly examine each type in turn so that you can determine which style of application might be best for your situation.

3.4.1. Windows smart client applications

Smart client applications can be designed to combine the benefits of a rich client application with the deployment and manageability strengths of a thin client application, although the precise nature of the balance between the two approaches depends on the exact scenario.

Smart client applications often have very diverse requirements, and so vary greatly in design and implementation. However, all smart clients share some or all of the following characteristics:

- Make use of local resources
- Make use of network resources
- Support occasionally connected users
- Provide intelligent installation and update
- Provide client device flexibility

Many applications do not need all of these characteristics. As you design your smart clients, you will need to carefully consider your application scenario and decide which of these characteristics your smart client application requires. Incorporating all of these characteristics into your application will require very careful planning and design, and in many cases you will need significant implementation resources.

Smart Client implements multiple layers of services and components based upon those standards. These rich capabilities are then optimized and tested on every popular web browser and operating system. Essentially, Smart Client treats browser Document Object Model (DOM) calls as the new "assembly language", and HTML as the new "pixels". So Smart Client applications run without software download, installation or configuration, on standard web browsers including Internet Explorer, Firefox, Mozilla, Netscape, Safari and Opera, on Windows, Linux, Solaris, and MacOS operating systems.

3.4.2. Smart client applications advantages

Smart client applications overcome the tradeoff between It needs for centralized control and security, and end user's needs for power and flexibility- by merging the usability, performance and integration of desktop applications with a centrally-managed, secure, easy to deploy and low maintenance cost solution. The results are robust business solutions that use the internet as a worldwide LAN with the desktop functionality and performance. Smart client capabilities appeal to the needs of both IT administrators and end users. Because Smart client applications are build on the standard Microsoft business platform, and cost far less to develop and maintain than browser-based applications, the Smart client approach makes good economic sense as well.

3.4.3. Characteristics of smart client applications

3.4.3.1. Local resource utilization

A smart client application always has code artifacts on the client that enable local resources to be utilized.

3.4.3.2. Connected

Smart client applications are never standalone and always form part of a larger distributed solution. This could mean that the application interacts with a number of Web services that provide access to data.

3.4.3.3. Offline capable

Because they are running on the local machine, one of the key benefits that smart client applications offer is that they can be made to work even when the user is not connected. For applications running in occasional or intermittent connectivity situations such as those used by traveling workers or even those running on laptops, tablets, PDA’s and so on, where connectivity cannot be guaranteed at all times being able to work while disconnected is essential. Even when the client is connected, the smart client application can improve performance and usability by caching data and managing the connection in an intelligent way.

4. Process Flow Diagram for Smart Client System

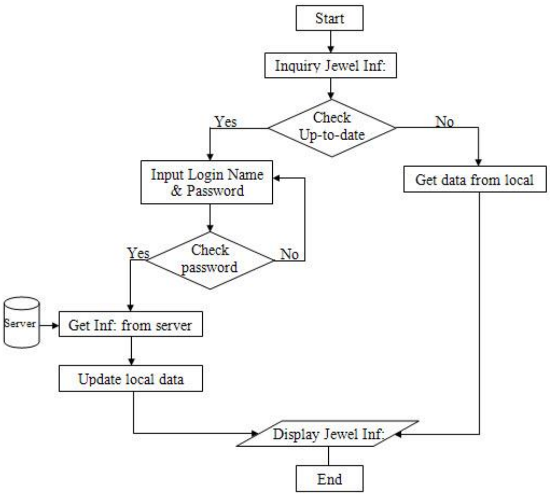


Figure 3. Process flow diagram

When user inquiries jewel information, user can get required jewel information. The desired data user need must be provided by smart client. Moreover, smart client information must be up-to-date. If jewel information data is not up-to-date data, smart client check up-to-date and password for jewel information with server and update the smart client. Finally, desired valuable jewel information can be displayed to user.

5. Proposed System Design

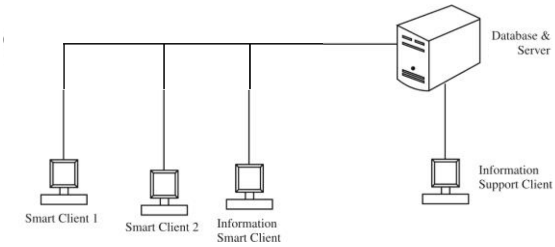


Figure 4. Proposed System Design

The proposed system has designed smart client/server architecture to implement a flexible, usable system using a commodity hardware and software products.

Smart clients allow data and logic to be distributed to the client computer, whereas thin clients tend to keep the data and logic centralized on the window server and other back-end services. Although the smart client approach allows you to make the application more efficient, with no round trips to the server to determine next steps, you need to consider that the application and its data are now more widely distributed than with thin client applications, and modify your design accordingly.

If you are implementing business rules on the client, you will need to update those rules as required, without updating the entire application. This may mean that you use differing mechanisms for updating the application and updating business rules within the application.

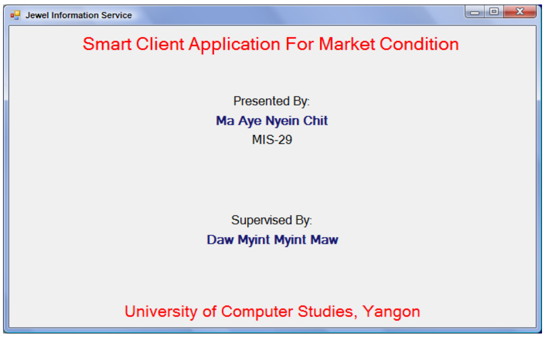
By caching data on the client, you can significantly improve the performance and usability of an application, but you must ensure that the data is refreshed appropriately and that stale data is not used. Because many users can access and use the same data, you must also consider the effects of data concurrency. Your application must be able to handle data conflicts or reconciliation issues that arise because the application is now more widely distributed and can operate while offline.

6. Implementation of the System

This implementation of the proposed windows based application system is constructed for admin and anonymous user. Admin user side is the part of online updateable service and anonymous user side is to provide online information service in 24 hours.

The processes flow of the proposed system is as follows:

Step1: User can access jewel information service on online.



Jewel Information Service

Smart Client Application For Market Condition

Presented By:
Ma Aye Nyein Chit
MIS-29

Supervised By:
Daw Myint Myint Maw

University of Computer Studies, Yangon

Figure 5. Jewel information service page

Step2: If you search on online, access user input to Login, you can write admin name and password in figure 6.Then jewel information will get on server and then screen is shows input jewel information and description of file menu in figure 6. If the customers want to buy the jewel, find and click on that category. After accepting the user query, it is post back to window server and retrieves the efficient data from backing storage of database server. The view information displays details information of the entire jewel in database. If customer finds the jewel by description, then all results are displayed with show information button.

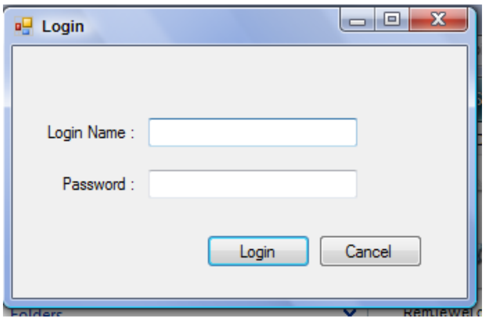


Figure 6. Login entry page

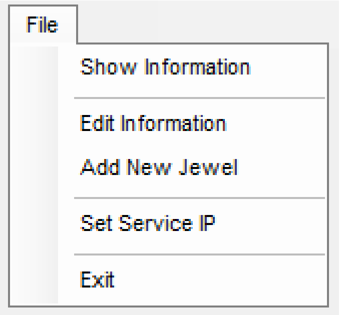
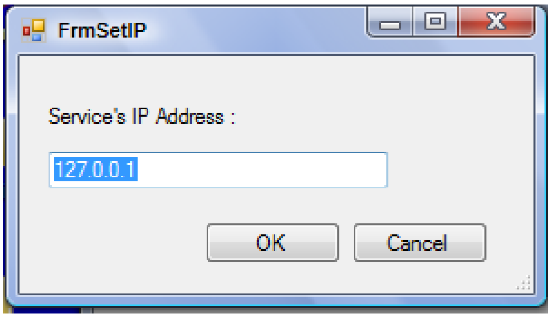


Figure 7. Show the File menu form



Service's IP Address : 127.0.0.1

Figure 8. Service IP address form

Step3: The admin uses the data described in figure 8 as you can link server write the IP address. And then click on the edit button, If the change to about the jewel information.

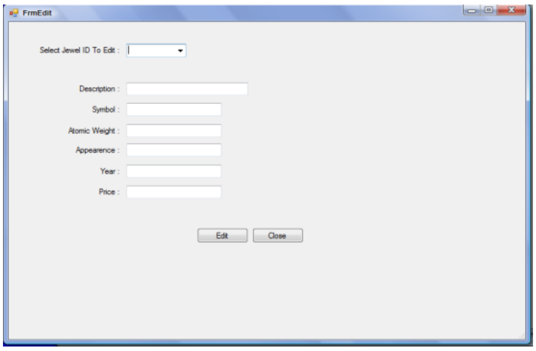


Figure 9. Edit jewel information form

Step4: The administrators can edit jewel information. If there is the change to about the jewel information, click on the edit button. Then click save button.

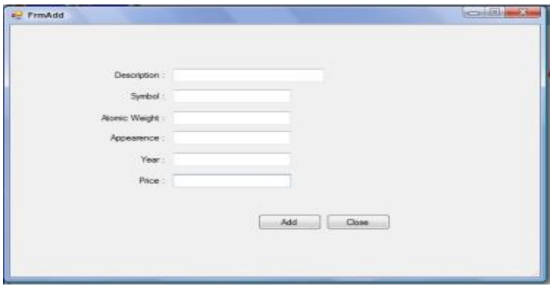
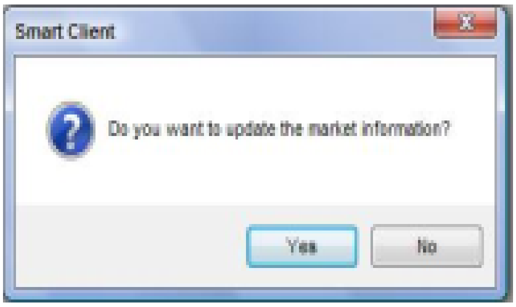


Figure 10. Showing the Add Form

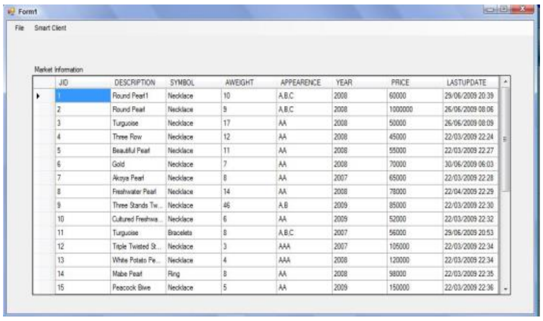
Step5: If the user clicks the 'No' button, can you see not update information. Then, this menu has two options. These are File and Smart client.



Do you want to update the market information?

Figure 11. Showing the Update Information Form

Step6: To create a File, the user first must create the View market information and Exit menu. Then the View market information is generated, show View market information table



Market Information

JID	DESCRIPTION	SYMBOL	AWEIGHT	APPEARENCE	YEAR	PRICE	LASTUPDATE
1	Round Pearl1	Necklace	10	A,B,C	2008	60000	29/06/2009 20:39
2	Round Pearl	Necklace	9	A,B,C	2008	1000000	26/06/2009 08:06
3	Turguoise	Necklace	17	AA	2008	50000	26/06/2009 08:09
4	Three Row	Necklace	12	AA	2008	45000	22/03/2009 22:24
5	Beautiful Pearl	Necklace	11	AA	2008	55000	22/03/2009 22:27
6	Gold	Necklace	7	AA	2008	70000	30/06/2009 06:03
7	Akoya Pearl	Necklace	8	AA	2007	65000	22/03/2009 22:28
8	Freshwater Pearl	Necklace	14	AA	2008	78000	22/04/2009 22:29
9	Three Stands Tw...	Necklace	46	A,B	2009	85000	22/03/2009 22:30
10	Cultured Freshwa...	Necklace	6	AA	2009	52000	22/03/2009 22:32
11	Turguoise	Bracelets	8	A,B,C	2007	56000	29/06/2009 20:53
12	Triple Twisted St...	Necklace	3	AAA	2007	105000	22/03/2009 22:34
13	White Potato Pe...	Necklace	4	AAA	2008	120000	22/03/2009 22:34
14	Mabe Pearl	Ring	8	AA	2008	98000	22/03/2009 22:35
15	Peacock Biwe	Necklace	5	AA	2009	150000	22/03/2009 22:36

Figure 12. Showing the Market Information

7. Conclusion

Smart client exploit the power of today computing hardware, work online or offline and access information in a standard way through the use of web services, helping users and business respond rapidly changing business needs.

With smart client users don't have to worry about where the data resides and how to get the data they just have the data when they need it. The combination of fast deployment and high functionality made smart clients more viable than the alternatives.

The results were more efficient business processes, improved user satisfaction, and cost savings.

References

[1] A.hoffer Jeffder, B.Prescott Mary, R.Macfadden Fred, "Modern Database Management" Eight Edition, 2007.

[2] Boulter, Mark, "Smart Client Architecture and Design Guide" Microsoft, 2004.

[3] URL: http://msdn.microsoft.com/perf

[4] URL: http://msdn.microsoft.com/smartclient/

[5] URL: http://www.microsoft.com/net/ smartclient/default.mspx

[6] URL:http://msdn.microsoft.com/library/en-us/vsent7/html/vxoriDesignConsiderationsForDistributedApplications.asp

[7] URL:http://msdn.microsoft.com/library/en-us/vsent7/html/vxoriplanningdistributedapplications.asp

[8]□□URL:http://www.windowsforms.net/ Default.aspx

[9] URL:htttp://msdn.microsoft.com/vstudio /using/understand/perf/

[10] URL:http://msdn.microsoft.com/library /default.asp

[11] URL:http://msdn.microsoft.com/library /default.asp

[12] URL:http://www.isomorphic.com/ " SmartClient™ Quick Start Guide", SmartClient v 6.5 ,April 2008.