UNITED STATES PATENT AND TRADEMARK OFFICE

_____

BEFORE THE PATENT TRIAL AND APPEAL BOARD

_____

LIGHTSPEED COMMERCE INC.,

Petitioner,

v.

CLOUDOFCHANGE, LLC,

Patent Owner.

_____

Case IPR2022-00997

Patent 10,083,012

DECLARATION OF WAYNE BARATTA

DECLARATION

I, Wayne Baratta, declare as follows:

1. I am one of the named inventors of U.S. Patent No. 9,400,640 (the "'640 Patent"). I am a co-Founder and part owner of CloudofChange, LLC. I was previously the Founder and CEO of TYRAMM International Inc.

2. In 2006, I began collaborating with Mr. Quentin Olson on potential projects related to point of sale (POS) systems. On January 23, 2007, I formed the entity TYRAMM International Inc. in connection with our efforts to develop intellectual property related to web-based POS systems.

3. Prior to February 26, 2007, Mr. Olson and I agreed to an obligation to assign any patents that could issue from the above-listed patent applications to the same person or entity.

4. On behalf of Mr. Olson and myself, I engaged Mr. Stephen Ackerman of the law firm Saile Ackerman LLC to prepare and prosecute the following patent applications:

Patent App. No.	Filing Date	Inventor(s)
11/710,722	Feb. 26, 2007	Quentin Olson
11/710,723	Feb. 26, 2007	Quentin Olson Wayne Baratta
12/012,393	Feb. 1, 2008	Quentin Olson Wayne Baratta
12/012,666	Feb. 5, 2008	Quentin Olson Wayne Baratta

5. Attached as Exhibit A is a true and correct copy of an invoice I received from Mr. Ackerman, on or around March 6, 2007 in connection with U.S. Patent Application Nos. 11/710,722 and 11/710,723.

6. I informed Mr. Ackerman that any patents that may issue from the above-referenced patent applications would not be assigned to TYRAMM International Inc., but that they still would be assigned to the same entity, which was yet to be named.

7. On July 27, 2011, TYRAMM International Inc. was dissolved.

8. On November 18, 2016, Mr. Olson and I formed the entity CloudofChange, LLC in the State of New York.

9. After U.S. Patent Application No. 12/012,666 issued as the ’640 Patent on July 26, 2016, Mr. Olson and I received assignment documents from Mr. Ackerman to assign the ’640 Patent to CloudofChange, LLC.

10. On January 31, 2017, I executed an assignment document to assign the ’640 Patent to CloudofChange, LLC as the assignee.

11. Attached as Exhibit B is a true and correct copy of the Patent Assignment Cover Sheet and Assignment Documents recorded with the USPTO on March 2, 2017.

I understand that willful false statements and the like are punishable by fine or imprisonment, or both (18 U.S.C. § 1001) and may jeopardize the validity of the application or any patent issuing thereon. All statements made of my own knowledge are true and all statements made on information and belief are believed to be true. I declare under the penalty of perjury that the foregoing is true and correct to the best of my knowledge and that this declaration was executed on this 22nd day of July 2022 in Poughkeepsie, New York.

Wayne Baratta

EXHIBIT A1

TY06-001

Quentin Olson

To PTO 2/26/07

Ser #11/710,722 2/26/07 Ralph

** Appeal No: 2013-003322

Filing Receipt 3/22/07

Not to file Missing Parts 3/22/07

Info. Disc. Stmt. - 05/11/07

Not to file Missing Parts 5/22/07

Notice of publication 9/12/08

Supp. IDS - 10/29/08

Rejection 12/29/08

Amendment 3/30/09

Final Rejection 7/1/09

~~Assignee: TYRAMM International, Inc.~~

TY06-001C

RCE 1/4/10 (3mos. late)

Suppl. Amendment 3/29/10

Rejection 4/28/10

Amendment 7/28/10

Final Rejection 9/28/10

Amend AF 11/29/10

Advisory Action 12/21/10

RCE 3/28/11 3mos. late

Rejection 5/23/11

Amendment 8/23/11

Final Rejection 9/21/11

Amend AF 1/23/11

Advisory Action 2/9/12

TY06-001

EXHIBIT A2

US 20080208696A1

(19) United States

(12) Patent Application Publication

Olson

(10) Pub. No.: US 2008/0208696 A1

(43) Pub. Date: Aug. 28, 2008

(54) POINT OF SALE SYSTEM WITH WEB-BASED BACK-OFFICE

(76) Inventor: **Quentin Olson**, Poulsbo, WA (US)

Correspondence Address:
George O Saile and Assocs
28 Davis Ave
Poughkeepsie, NY 12603

(21) Appl. No.: **11/710,722**

(22) Filed: **Feb. 26, 2007**

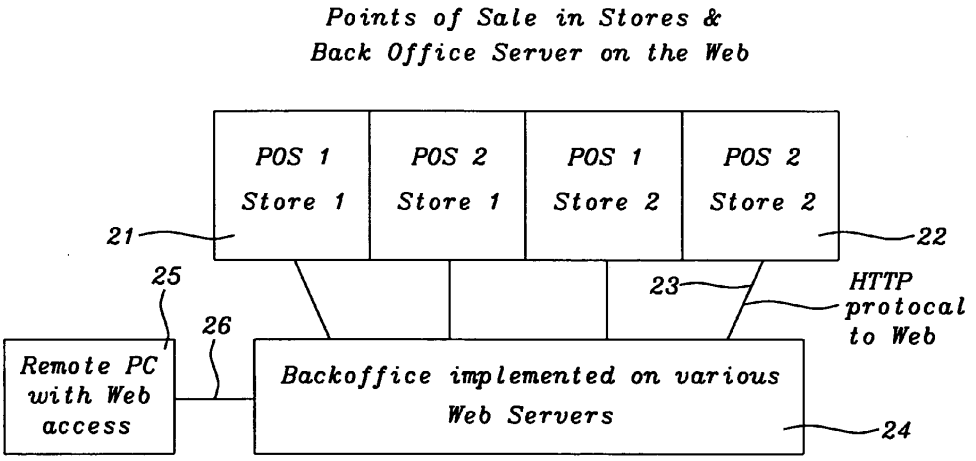
Publication Classification

(51) **Int. Cl.**
G06Q 10/00 (2006.01)

(52) **U.S. Cl.** **705/16**

(57) **ABSTRACT**

This invention provides a system and a method for managing a range of businesses. The business operations range from a single branch to a large chain of stores or branches. This invention relates to the managing of the points of sale (POS) with Web-based back-office systems. POS transaction engine terminals are implemented on personal computer, PC hardware using PC software with a relational database. The transaction engine architecture provides cost-effective application customization. Web-based back-office software is implemented on servers. POS and BO systems are monitored and supported via the web using protocol and open software components.



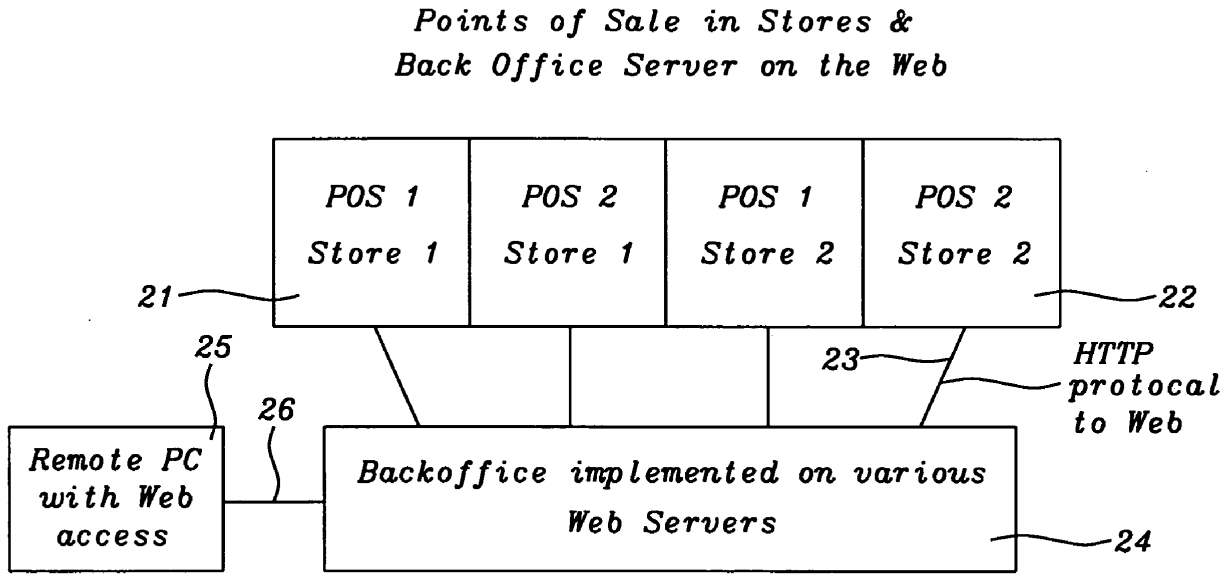

Points of Sale in Stores &
Back Office Server on the Web
POS 1
Store 1
POS 2
Store 1
POS 1
Store 2
POS 2
Store 2
21
22
23
HTTP
protocal
to Web
25
26
Remote PC
with Web
access
Backoffice implemented on various
Web Servers
24

FIG. 1

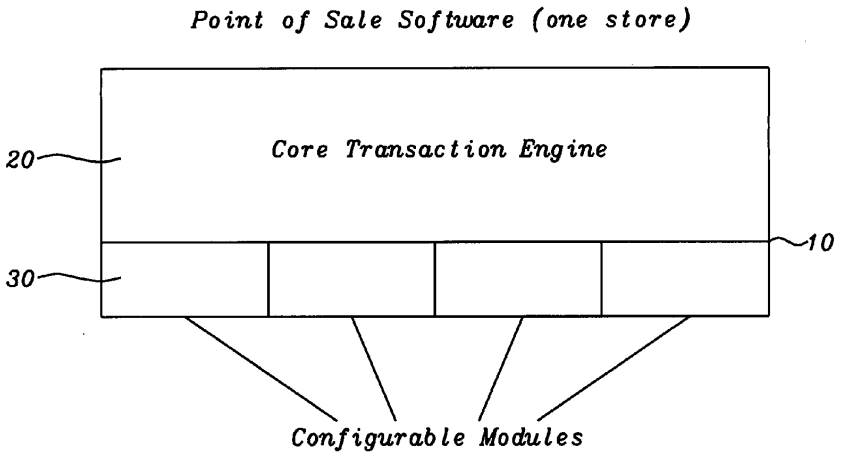

Point of Sale Software (one store)
Core Transaction Engine
20
10
30
Configurable Modules

FIG. 2

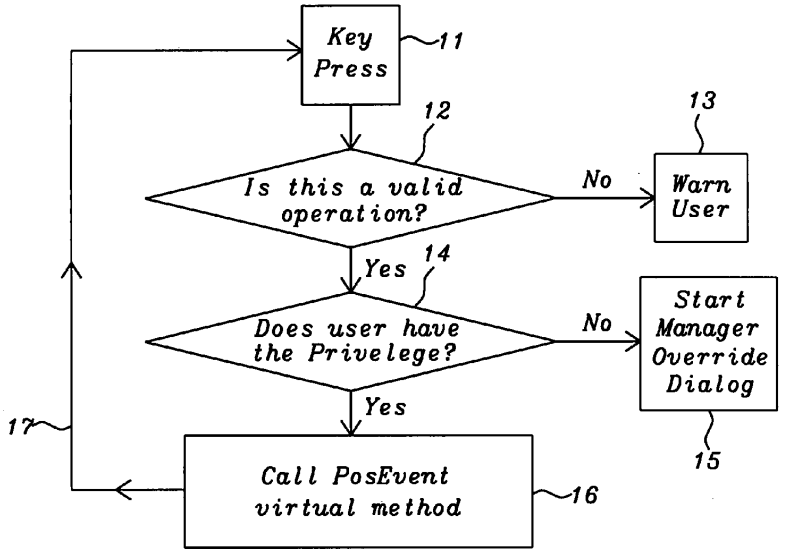

Key Press
11
12
Is this a valid operation?
No
13
Warn User
Yes
14
Does user have the Privelege?
No
Start Manager Override Dialog
15
Yes
17
Call PosEvent virtual method
16

FIG. 3

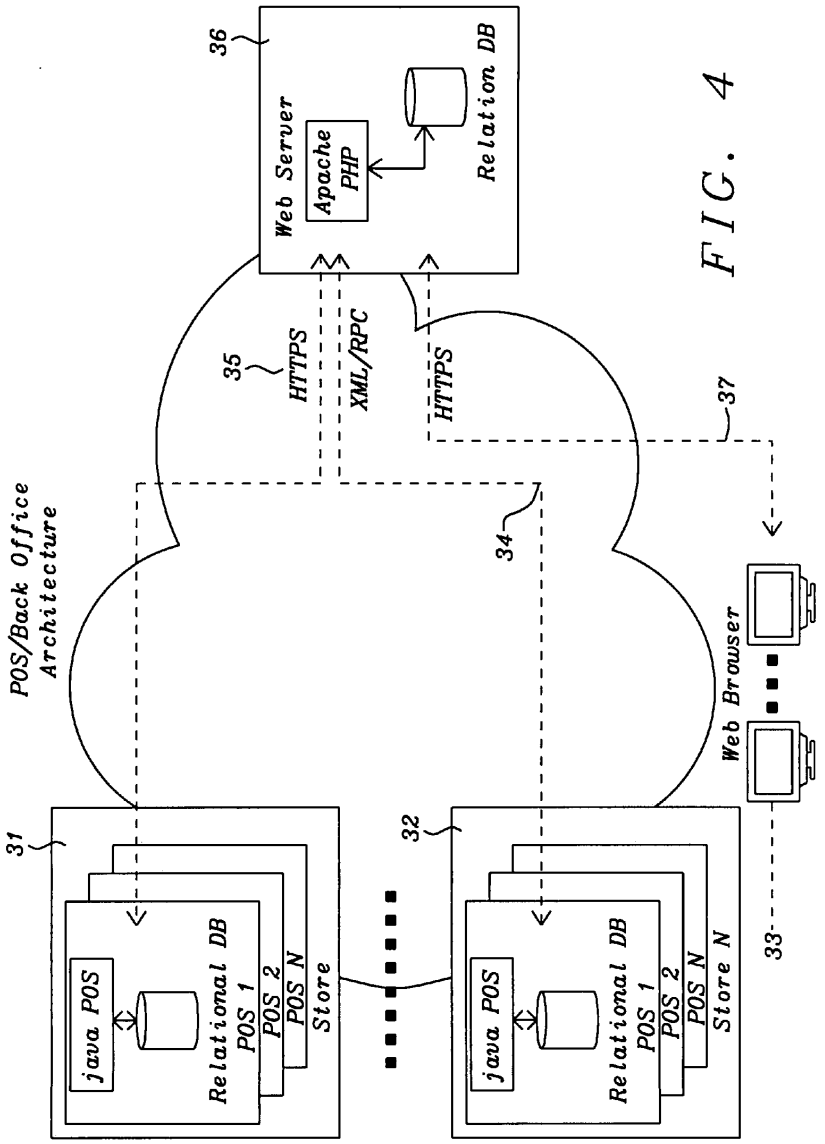

POS/Back Office Architecture
31
32
33
34
35
36
37
Web Server
Apache PHP
Relation DB
HTTPS
XML/RPC
HTTPS
Web Browser
java POS
Relational DB
POS 1
POS 2
POS N
Store
Store N

FIG. 4

POINT OF SALE SYSTEM WITH WEB-BASED BACK-OFFICE

RELATED PATENT APPLICATION

[0001] This application is related to docket number TY2006-002, filed on _____, Ser. No. _____, which is herein incorporated by reference in its entirety.

BACKGROUND OF THE INVENTION

[0002] 1. Field of the Invention

[0003] This invention relates to a system and an apparatus for managing a range of businesses. The business operations range from a single branch to a large chain of stores or branches.

[0004] More particularly this invention relates to the managing of the points of sale (POS) with Web-based back-office systems.

[0005] 2. Description of Related Art

[0006] Currently, point of sale (POS) functions are implemented with custom hardware and/or custom software in the stores or business branches. In addition, custom software is currently used for server functions. Today, servers are required to be located in each store or branch for normal mode operation, and for a backup mode of operation, which is needed if the servers go down. The disadvantages of current practice are that store and branch changes require time and money for software development and testing. In addition, current practice has weak scalability and requires significant technical skill to manage the in-store components. As POS terminals are added within a store or branch or as stores or branches are added, the performance of the overall system degrades. Another disadvantage of current practice is the use of non-standard protocols. Many of these non-standard protocols were implemented before the availability of PC hardware and software and before the wide spread use of universal web protocols. Consequently, current practice includes proprietary protocols which were implemented earlier, but their continued operation is required to enable enterprises to preserve the large investment in proprietary business operations and systems.

[0007] U.S. Pat. No. 4,679,154 (Blanford) describes a checkout system for processing purchased merchandise items. The system includes a checkout counter which contains a scanning assembly for scanning a coded label on the purchased item to produce data signals identifying the item purchased, a display member on the counter for displaying the price of the merchandise item purchased, an indicator light and a tone generator located in the display member, a back-office processor for obtaining the price of the merchandise item, and a data terminal device for printing the price of the scanned items on a record member.

[0008] U.S. Pat. No. 6,856,970 B1 (Cambell et al.) discloses a financial transaction system which allows clients of a financial institution to use a web-based workstation to interface with a plurality of back-office systems within one or more financial institutions. This system comprises a hub server, a plurality of financial institutions, at least one web server, and at least one database server.

[0009] U.S. Pat. No. 6,504,481 B2 (Teller) describes a system, method, and device for monitoring the dispensing of liquids, in particular, beverages dispensed at a bar or restaurant. The system includes an electronic sensor device, data receiver, personal computer, bottle opener/camera system, bar code scanner, point of sale, bar camera, network, Web server, and paging system. The method automatically detects and monitors dispensing events, including detecting the type and amount of liquid, reconciling dispensing events with register ring-ups, recording information, saving video specific to the dispensing event, and sending a page indicating anomalies.

BRIEF SUMMARY OF THE INVENTION

[0010] It is the objective of this invention to provide a system and a method for a system and an apparatus for managing a range of businesses. The business operations range from a single branch to a large chain of stores or branches. This invention relates to the managing of the points of sale (POS) with Web-based back-office systems.

[0011] It is further an object of this invention to the manage the points of sale (POS) with Web-based back-office systems.

[0012] The objects of this invention are achieved by point of sale system with a web-based back-office system comprised of one or more point of sale (POS) terminals and one or more web servers. The POS terminals and web servers communicate via commonly used, standardized internet protocols. The POS terminals are implemented with personal computer hardware and software using one or more relational databases. The web server is implemented with web server hardware and software, also using one or more relational databases.

BRIEF DESCRIPTION OF THE DRAWINGS

[0013] FIG. **1** shows a high level diagram of this invention.

[0014] FIG. **2** gives a high level diagram of a point of sale (POS) terminal.

[0015] FIG. **3** gives a core transaction engine/configurable module flow chart.

[0016] FIG. **4** is a more detailed diagram of a main embodiment of this invention.

DETAILED DESCRIPTION OF THE INVENTION

[0017] FIG. **1** shows a high level diagram of this invention. Each point of sale (POS) terminal such as POS **21** represents a single cash register in a store, retail location, or business location. POS **21** is in STORE **1** and POS **2** (**22**) is in store **2**. Each POS includes personal computer hardware and software. Additional POS terminals beyond the two shown, as well as additional stores beyond the two shown, are within the scope of the invention. Each POS normally operates with a hardware/software connection **23** to the Internet or Web. However, if the web goes down, the POS terminal continues to operate. There is a loose coupling of the POS to the back office (BO). The POS to BO connection is not required for the basic business functions of the POS. All transaction data is stored in a relational database on the hard drive in the POS.

[0018] A relational database stores all of its data inside tables. All operations on data are done on the tables themselves. Some operation produce other tables as the result. A table is a set of rows and columns. Each row is a set of columns with only one value for each. All rows from the same table have the same set of columns, although some columns may have NULL values. A NULL value is an "unknown" value. The rows from a relational table are analogous to a record, and the columns are analogous to a field. Below is an example of a relational table.

NAME	COMPANY	E_MAIL
Jane A. Doe	ABC	jad@abc.com
Bill X. Smith	XYZ	bxs@xyz.com

There are two basic operations one can perform on a relational table. The first one is retrieving a subset of its columns. The second is retrieving a subset of its rows. The field names such as company describe the content of the columns of the relational table. The rows delineate the individual records stored in the relational tables.

[0019] As transactions are created at a POS a log entry for the newest transaction is also created, this log entry is used to flag if the transaction has been uploaded to the web server. Part of the POS application, the BO interface is continuously running in the background. This component reads the log of transactions. If a transaction needs to be sent, it tries to send it. If the send fails (for example, if the connection to, or the Internet itself, is down), it goes to sleep and tries again later. Additionally, the BO interface requests update from the BO such as new items, price changes, employees, etc. The POS terminals communicate via HTTP protocol (hypertext transfer protocol) **23** with Back-office BO software, which is implemented on web servers **24**, which can be located anywhere in the world. In addition, the BO software and data can be viewed from any store employee at any PC who has Internet access **26** and a password.

[0020] The POS such as **21** send transaction data to the BO in the form of an HTTP post or communication. The packet **23** sent from the POS to the BO consists of transactions, employee clock, customer add/update, item add/update, promotions and more. Promotions are configured in the back office and associated with items or customers or departments. For example, a promotion may be associated with a customer to implement customer loyalty points or a promotion may be associated with a certain item for a % discount. A client who is the store manager or owner selects a promotion type, associates it with an item, department, etc, then sets the parameters that control how that promotion works. These transaction transmissions between the POS and the BO can be encrypted to insure privacy and security. A typical encryption method is **128** bit SSL (secure sockets layer). A further element of security is that each BO client (individual POS, store or multi-store owner) gets their own instance of a database. When they log into the BO they are attached to their own relational database associated and validated via their user login and password.

[0021] FIG. **2** is a high-level diagram of a POS terminal **10**. There is a core transaction Engine **20** which is software which processes all input and output at the POS terminal and invokes the appropriate retail logic. The core transaction engine interfaces with configurable modules **30**, which handle the meaning & processing steps of terminal keys required. For example, one business or one store may require a different function performed by the transaction engine when the "Total" key on the POS terminal is pressed. A convenience store may have a different way of computing the summation of taxable items than a restaurant business. These configurable modules allow the POS terminal to be very flexible for use by many different types of stores and businesses.

[0022] An example of a promotion which requires some level of configuration information is "Buy 1 and get one free or buy 1 and get the second 50% off". Typically, these configurable modules are preferably implemented in a computer language such as JAVA which is a combined compile and interpretive language. The use of a JAVA-type object-oriented language allows the dynamic loading of executable configurable modules at run-time to handle things such as different operations of the "Total" key, without changing the other transaction engine modules. Other examples of transaction engine modules are Cash Tender module and Sub-Total module. This allows an efficient, low labor, low cost method of implementing different POS terminal functions.

[0023] FIG. **3** is a flowchart showing how core transaction software could process a given key press action **11**. This key press action **11** could be associated with one of the several keys on a point of sale terminal keyboard, touch screen or touch pad. The core transaction engine asks **12** if this key press action is a valid operation in the current state of the POS machine. If it is not **13**, the user is warned via an indicator or message on the POS screen. If it is valid, the core transaction asks if the user has privilege **14**. If the user does not have privilege, the transaction engine starts the manager override dialog **15**, which asks the user for more information in order to gain access. If the user does have privilege, the transaction engine calls a configurable module **16** called the PosEvent virtual method to handle the required meaning of the key press action. After handling the tasks associated with the key press, the transaction engine returns control **17** back to the key press **11**.

[0024] The POS transaction is engine designed to allow easy extension of retail business logic through Java dynamic class loading. For example, if an end user wants to modify the way the "Total Key" operates they only need to sub-class an abstract class (PosEvent) and make it available to the Java class loader. No other piece of code requires modification. The system accomplishes this design through a layered architecture. At the bottom is the POS context and electronic journal. These make up the core POS transaction engine by binding together the user context, system parameters and the transaction-processing engine. POS events interact with hardware and the core engine to accomplish the business logic. A hardware layer abstracts the POS events and the physical and virtual devices (key presses, peripheral I/O, input/output). Hardware abstraction classes manage receipt (or spooled) device I/O, interactive I/O (operator prompts) and peripherals that invoke business logic directly (such as cash drawer close). The design utilizes an event model familiar to most GUI application programmers. A listener is registered for events and these events are processed using classes derived from the base (virtual) PosEvent class. The POS event may be invoked or triggered in one of three ways: 1) Directly from a key press/GUI (graphical user interface) button press as shown in FIG. **3**, 2) from a device event (credit card swipe, barcode scan, etc.), direct I/O or status change from JavaPOS devices or 3) programmatically from within other PosEvent derived classes or a dialog. An example of a dialog is shown in FIG. **3**. A Start Manager override dialog box **15** is shown. This dialog box helps the system to determine if a user should be able to trigger the POS event **16** even though the user does not have "privilege" rights to do so.

[0025] FIG. **4** is a more detailed diagram of the main embodiment of this invention. Stores **31**, **32** illustrate several POS implemented with Java software using relational databases. The boxes **31**, **32** in FIG. **4** show each POS having its own relational database. This architecture offers an element

of security. Each BO client, which is an individual POS gets their own instance of a database. When they log into the BO they are attached to their own relational database associated and validated via their user login and password.

[0026] Web Server **36** illustrates a BO implemented with Apache PHP using relational databases. PHP (recursive acronym for "PHP: Hypertext Preprocessor") is a widely-used Open Source general-purpose scripting language that is especially suited for Web development and can be embedded into HTML (hypertext markup language). Apache is a web server currently used in over half of all web sites worldwide. PHP code is different from a script written in other languages like Perl or C. Instead of writing a program with lots of commands to output HTML, one can write an HTML script with some embedded code to do something (ie. to output some text). The PHP code is enclosed in special start and stop tags that allow you to jump into and out of "PHP mode". What distinguishes PHP from something like client-side JavaScript is that the code is executed on the server. If you were to have HTML script with PHP code imbedded in it, the client would receive the results of running that script, with no way of determining what the underlying code may be. You can even configure your web server to process all your HTML files with PHP. The advantages of using PHP are that it is extremely simple for a newcomer, free to use, but offers many advanced features for a professional programmer. A programmer can jump in and in a short time start writing simple scripts. Remote and local web browsers **33** illustrate a method used by store managers, corporate administrators, or system administers to access and view a company & its store's instantaneous performance from minute to minute. The POS, BO, and Web Browsers all communicate via the Web. **34**. The protocols between the POS and Web browsers and BO servers is HTTPs, XMURPC. **35**. XML-RPC is a remote procedure call protocol which uses XML to encode its calls and HTTP as a transport mechanism. It is a very simple protocol which defines only a handful of data types and commands.

[0027] The key advantages of this point of sale system with a web-based back-office system are as follows. The web and PC architecture both offer full-scalability which results in improved performance and lower cost. There is a lower cost associated with projects developed with the technology of this invention due to the flexibility of easy design changes and well-understood software. There is less training required for programmers and system testers. Projects can draw on the huge talent pool in the open source development community. The invention allows configurable modules for different types of businesses and sales promotions. The invention allows remote monitoring of both the POS and back-office systems from anywhere via the web. There is minimal time required for POS installation, since POS setup is as basic as a home PC setup. Another advantage is that the back-office system can be provided as a service or deployed within a corporation. For example, Software as a Service (SAAS) is a software distribution model in which applications are hosted by a vendor or service provider and made available to customers over a network, typically the Internet. Another advantage of this invention is that the back-office system is maintained in customer centric databases, making it impossible for customers to see other's data. Each BO client (individual POS, store or multi-store owner) gets their own instance of a database. When they log into the BO they are attached to their own relational database associated and validated via their user login and password.

[0028] While this invention has been particularly shown and described with Reference to the preferred embodiments thereof, it will be understood by those Skilled in the art that various changes in form and details may be made without Departing from the spirit and scope of this invention.

What is claimed is:

1. A point of sale system with a web-based back-office system, comprising:

one or more point of sale, POS, terminals and

one or more web servers,

wherein said POS terminals and said web servers communicate via internet protocols,

wherein said POS terminals are implemented with personal computer hardware and software using one or more relational databases,

wherein said web server is implemented with web server hardware and software, using one or more relational databases.

2. The point of sale system with a web-based back-office system of claim **1** wherein said POS terminals and said web servers are implemented with executable, configurable software modules, wherein said modules can be easily changed to implement different functions for different businesses.

3. The point of sale system with a web-based back-office system of claim **1**, wherein said POS terminals can operate standalone without communicating with said web server.

4. The point of sale system with a web-based back-office system of claim **1**, wherein said POS system and said web-based back-office servers are scalable allowing the addition of POS terminals and business locations without degrading system performance.

5. The point of sale system with a web-based back-office system of claim **1**, wherein a real-time status of said POS terminals and said back-office web servers is obtained at remote locations via personal computers with browser based web access.

6. A web-based back-office system comprising:

one or more web servers,

wherein said web servers are implemented with web server hardware and software, using one or more relational databases,

wherein said web servers can communicate with point of sale, POS, terminals via internet protocols.

7. The web-based back-office system of claim **6**, wherein said web servers are implemented with executable, configurable software modules, wherein said modules can be easily changed to implement different functions for different businesses.

8. The web-based back-office system of claim **6**, wherein said web servers allow said POS terminals to operate standalone without communicating with said web servers.

9. The web-based back-office system of claim **6**, wherein said web-based back office is scalable allowing the addition of POS terminals and business locations without degrading system performance.

10. The web-based back-office system of claim **6**, wherein a real-time status of said POS terminals and/or said back-office system is obtained at remote locations via personal computers with browser based web access.

11. A point of sale POS system comprising,

one or more point of sale, POS, terminals,

wherein said POS terminals can communicate with web servers via internet protocols,

wherein said point of sale terminals are implemented with personal computer hardware and software, using one or more relational databases.

12. The point of sale POS system of claim **11**, wherein said POS terminals are implemented with executable, configurable software modules, wherein said modules can be easily changed to implement different functions for different businesses.

13. The point of sale POS system of claim **11**, wherein said POS terminals can operate standalone without communicating with said web servers.

14. The point of sale POS system of claim **11**, wherein said POS system are scalable allowing the addition of POS terminals and business locations without degrading system performance.

15. The point of sale POS system of claim **11**, wherein a real-time status of said POS terminals and/or said back-office web servers is obtained at remote locations via personal computers with browser based web access.

16. A method of providing a point of sale system with a web-based back-office system, comprising the steps of:

providing one or more point of sale, POS, terminals and

providing one or more web servers,

wherein said POS terminals and said web servers communicate via internet protocols,

wherein said POS terminals are implemented with personal computer hardware and software using one or more relational databases,

wherein said web server are implemented with web server hardware and software, using one or more relational databases.

17. The method of providing a point of sale system with a web-based back-office system of claim **16** wherein said POS terminals and said web servers are implemented with executable, configurable software modules, wherein said modules can be easily changed to implement different functions for different businesses.

18. The method of providing a point of sale system with a web-based back-office system of claim **16**, wherein said POS terminals can operate standalone without communicating with said web server.

19. The method of providing a point of sale system with a web-based back-office system of claim **16**, wherein said POS system and said web-based back-office servers are scalable allowing the addition of POS terminals and business locations without degrading system performance.

20. The method of providing a point of sale system with a web-based back-office system of claim **16**, wherein a real-time status of said POS terminals and said back-office web servers is obtained at remote locations via personal computers with browser based web access.

21. A method of providing a web-based back-office system comprising the steps of:

providing one or more web servers,

wherein said web servers are implemented with web server hardware and software, using one or more relational databases,

wherein said web servers can communicate with point of sale, POS, terminals via internet protocols.

22. The method of providing a web-based back-office system of claim **21**, wherein said web servers are implemented with executable, configurable software modules, wherein said modules can be easily changed to implement different functions for different businesses.

23. The method of providing a web-based back-office system of claim **21**, wherein said web servers allow said POS terminals to operate standalone without communicating with said web servers.

24. The method of providing a web-based back-office system of claim **21**, wherein said web-based back-office is scalable allowing the addition of POS terminals and business locations without degrading system performance.

25. The method of providing a web-based back-office system of claim **21**, wherein a real-time status of said POS terminals and/or said back-office system is obtained at remote locations via personal computers with browser based web access.

26. A method of providing a point of sale POS system comprising the steps of:

providing one or more point of sale, POS, terminals, wherein said POS terminals can communicate with web servers via internet protocols,

wherein said point of sale terminals are implemented with personal computer hardware and software, using one or more relational databases.

27. The method of providing a point of sale POS system of claim **26**, wherein said POS terminals are implemented with executable, configurable software modules, wherein said modules can be easily changed to implement different functions for different businesses.

28. The method of providing a point of sale POS system of claim **26**, wherein said POS terminals can operate standalone without communicating with said web servers.

29. The method of providing a point of sale POS system of claim **26**, wherein said POS system are scalable allowing the addition of POS terminals and business locations without degrading system performance.

30. The method of providing a point of sale POS system of claim **26**, wherein a real-time status of said POS terminals and/or said back-office web servers is obtained at remote locations via personal computers with browser based web access.

* * * * *

EXHIBIT B1

TY06-002

Quentin Olson

Wayne Baratta

To PTO 2/26/07

Ser# 11/710,723 2/26/07 (Ralph)

Filing Receipt 4/18/07

Not to file Missing Parts 4/18/07

✓ Info. Disc. Stmt. - 05/11/07

Response 6/18/07

Rejection 7/22/08

~~Notice of Publication 8/28/08~~

Amendment 10/22/08

✓ Suppl. IDS - 12/11/08

Final Rejection 12/31/08

**Appeal No. 2013 - 003323

~~Assignee: Tyrapin International~~

TY06-002C

RCE 3/31/09

Rejection 5/8/09

Amendment 8/10/09

~~Final Rejection~~ 12/24/09

Amend AF 3/24/10

Advisory Action 4/23/10

TY06-002CC

RCE 5/24/10 (2mos. late)

Rejection 7/28/10

Amendment 10/28/10

Final Rejection 12/7/10

RCE w/amendment 5/9/11 2mos. late

TY06-002

EXHIBIT B2

US 20090198595A1

(19) United States

(12) Patent Application Publication

Olson et al.

(10) Pub. No.: US 2009/0198595 A1

(43) Pub. Date: Aug. 6, 2009

(54) HAND-HELD WEB BROWSER INVENTORY CHECKING SYSTEM

(76) Inventors: **Quentin Olson**, Poulsbo, WA (US); **Wayne Baratta**, Poughkeepsie, NY (US)

Correspondence Address:
SAILE ACKERMAN LLC
28 DAVIS AVENUE
POUGHKEEPSIE, NY 12603 (US)

(21) Appl. No.: 12/012,393

(22) Filed: Feb. 1, 2008

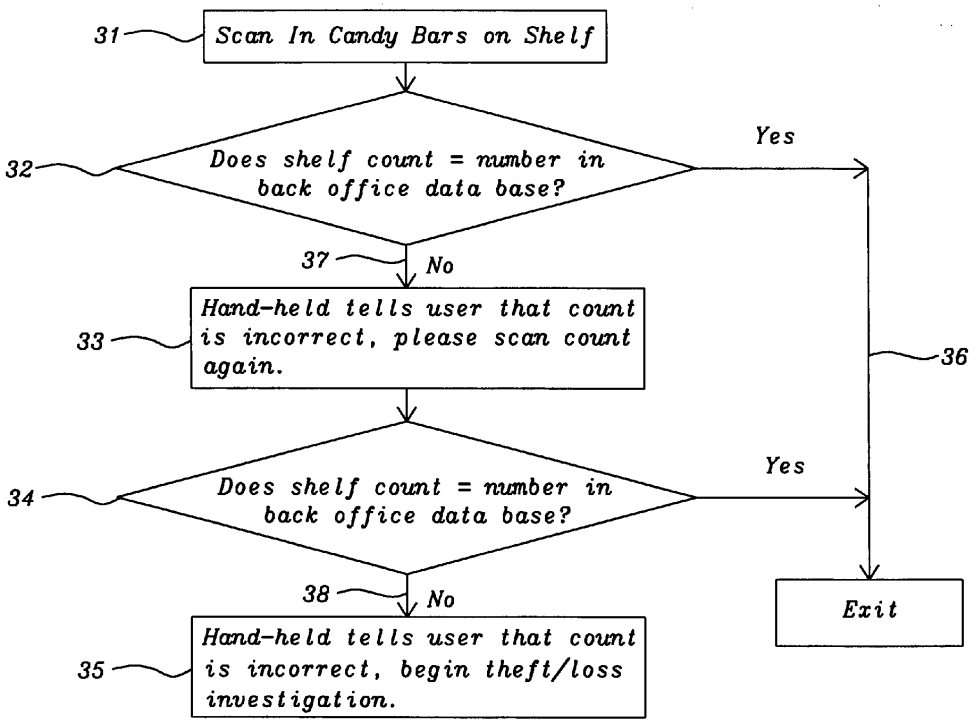
Publication Classification

(51) Int. Cl.
G06Q 10/00 (2006.01)

(52) U.S. Cl. 705/28

(57) ABSTRACT

This invention provides a system and a method for real-time checking and inputting business inventory. The business operations range from a single branch to a large chain of stores or branches. The application utilizes a hand-held web-browser based system which allows store and business operators to view and input inventory to a back-office server in real-time to assist in the management and security of businesses. The programming for the system resides in an out-of-store back office server. The hand-held device is simply a web browser with a bar code scanner. This system and method facilitates easy inventory checking for real-time theft and loss prevention.



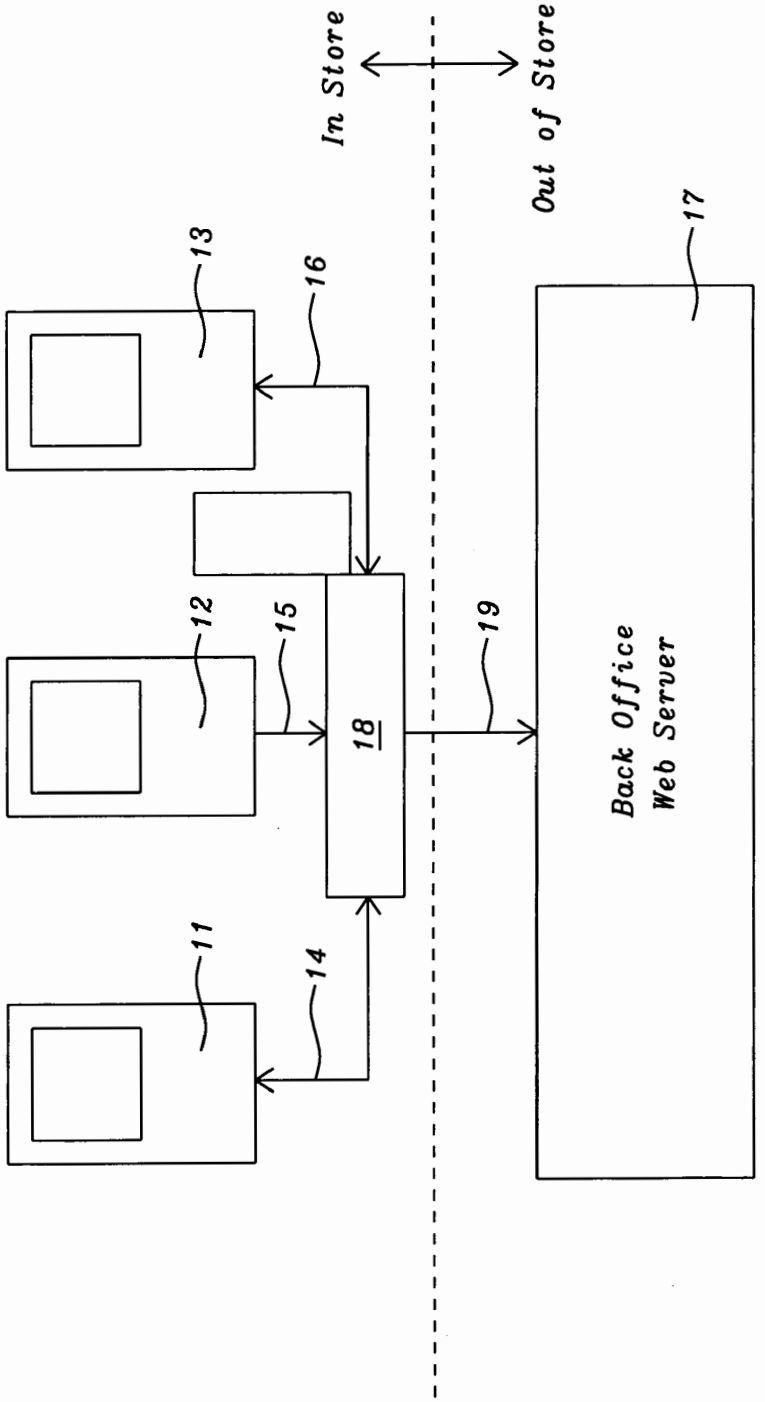

In Store
Out of Store
13
16
12
15
18
19
11
14
17
Back Office
Web Server

FIG. 1

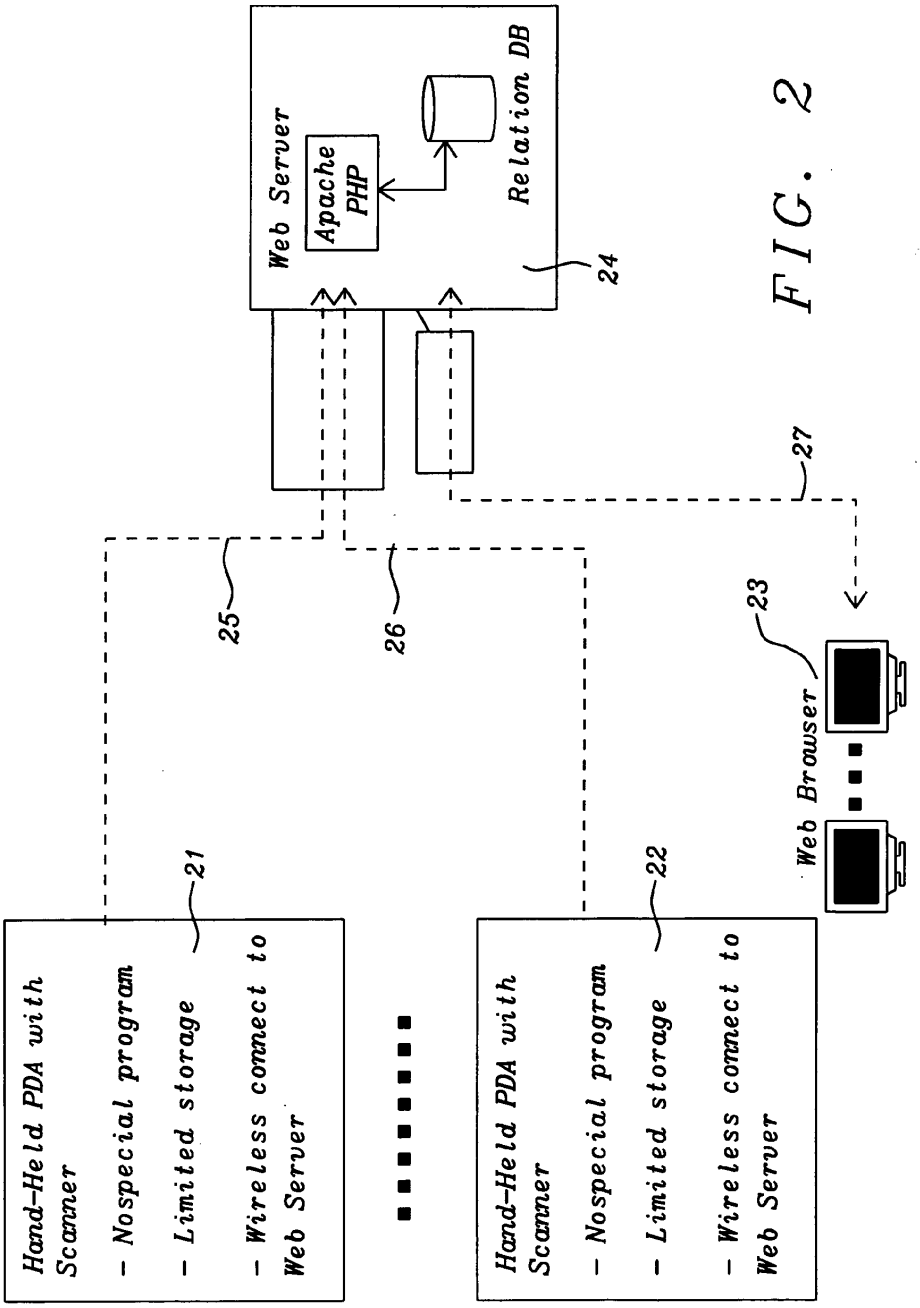

Web Server
Apache PHP
Relation DB
24
25
26
27
23
Web Browser
21
Hand-Held PDA with Scanner
- Nospecial program
- Limited storage
- Wireless connect to Web Server
22
Hand-Held PDA with Scanner
- Nospecial program
- Limited storage
- Wireless connect to Web Server

FIG. 2

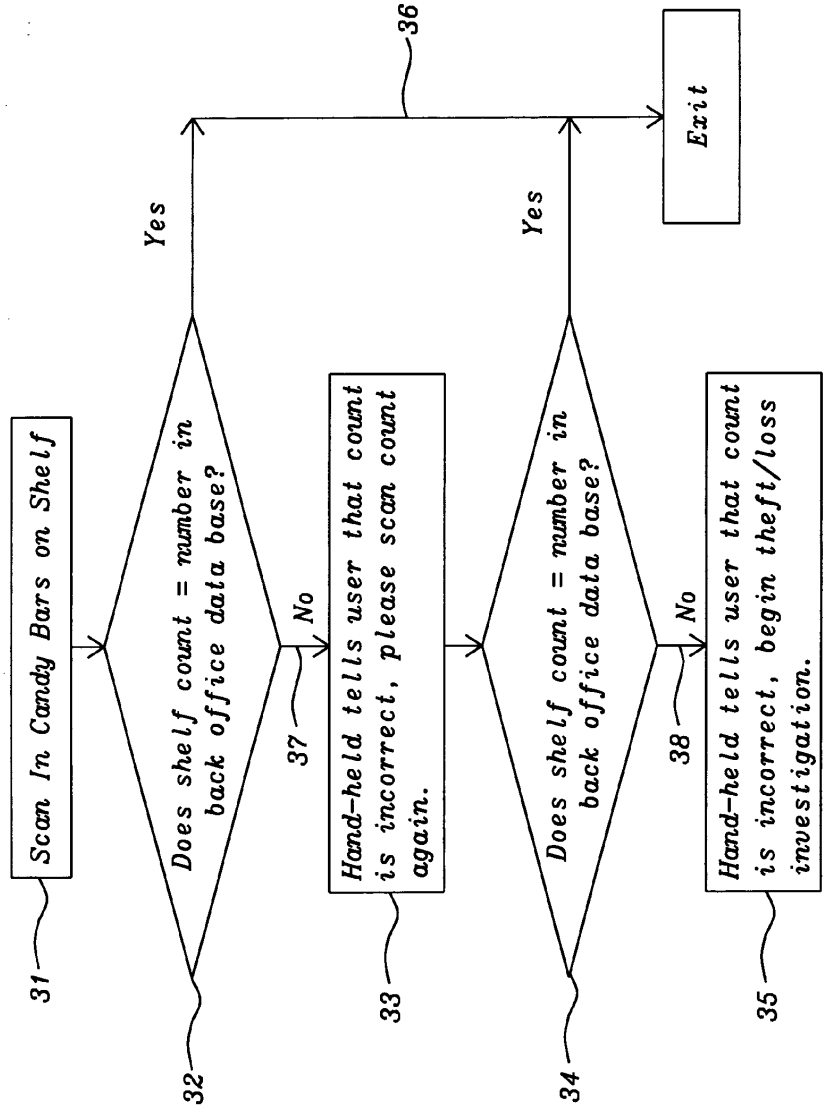

Scan In Candy Bars on Shelf
31
Does shelf count = number in back office data base?
32
Yes
No
37
Hand-held tells user that count is incorrect, please scan count again.
33
36
Exit
Does shelf count = number in back office data base?
34
Yes
No
38
Hand-held tells user that count is incorrect, begin theft/loss investigation.
35

FIG. 3

HAND-HELD WEB BROWSER INVENTORY CHECKING SYSTEM

RELATED PATENT APPLICATION

[0001] This application is related to docket number TY2006-001, filed on Feb. 26, 2007, Ser. No. 11/710,722, which is herein incorporated by reference in its entirety.

BACKGROUND OF THE INVENTION

[0002] 1. Field of the Invention

[0003] This invention relates to a system and a method for real-time checking and inputting business inventory. The business operations range from a single branch to a large chain of stores or branches.

[0004] More particularly this invention relates to a hand-held web-browser based system which allows store and business operators to view and input inventory to a back-office server in real-time to assist in the management and security of businesses.

[0005] 2. Description of Related Art

[0006] Current practice in the field of inventory management includes running an inventory application for retail stores, restaurants, warehouses, loading docks, and manufacturing operations on hand-held devices, then storing and uploading data to a computer server in a back room of the business operation or to a remote server in non-real time operation. During the time between storing and uploading data to the server, the inventory application is presenting inaccurate or not current information, because of the time delay to update the information at the server. Typically, hand-held devices currently used in inventory control systems need unique programs and substantial dedicated storage to operate. Therefore, the setup and testing of these hand-held devices is time-consuming and laborious.

[0007] U.S. Pat. No. 6,851,611 B1 (Shaw-Sinclair) describes a personal inventory system which uses a hand-held device which can read barcodes. The system allows users to organize, access, edit, and update information about items that they own.

[0008] U.S. Pat. No. 7,092,947 B2 (Zellner et al.) describes a portable telecommunication test set, such as a telephone line butt set, with a web browser included. The test equipment allows the technician to access the Internet and other remotely stored information to retrieve data and other useful technical information while in the field for network or line maintenance, trouble-shooting or repair.

[0009] U.S. Pat. No. 6,662,193 B1 (Christensen) discloses systems and methods for gathering and updating inventory data within a database through the use of hand-held devices.

[0010] U.S. Pat. No. 5,399,844 (Holland) describes an inspection prompting and reading recording system which includes a portable computer having a display, a keyboard, a bar scanner, a memory, a processor and a sound generator.

BRIEF SUMMARY OF THE INVENTION

[0011] It is the objective of this invention to provide a system and a method for real-time checking and inputting of business inventory.

[0012] It is further an object of this invention to provide a hand-held web-browser based system which allows store and business operators to view and input inventory to a back-office server in real-time to assist in the management and security of businesses.

[0013] The objects of this invention are achieved by a hand-held web browser inventory checking system comprising one or more hand-held terminals and one or more web servers. The hand-held terminals and web servers communicate via wireless Internet protocols. The hand-held terminals are implemented with personal digital assistants (PDAs). The web server is implemented with web server hardware and software, using one or more relational databases. The programming for the hand-held devices resides in and is executed from the web server.

BRIEF DESCRIPTION OF THE DRAWINGS

[0014] FIG. **1** shows a block diagram of the elements of this invention.

[0015] FIG. **2** is a more detailed diagram of a main embodiment of this invention.

[0016] FIG. **3** is a flowchart which illustrates the type of inventory checking procedures that are done with this hand-held system.

DETAILED DESCRIPTION OF THE INVENTION

[0017] FIG. **1** shows a high level diagram of this invention. An out-of-store back office (BO) web server **15**, as described in related patent application Ser. No. 11/710,722 filed on Feb. 26, 2007, is shown. This back office web server contains databases which include company inventory and sales transaction records. Hand-held (HH) units **11**, **12**, and **13** are also shown. These HH terminals have limited storage and minimal program processing, but are able to access the web via web browsing. These HH units have live wireless connections **14**, **15**, **16** to an in-store wireless router **18**. This wireless router **18** communicates with the out-of-store back office server **17** via a high-speed broadband link **19** such as DSL (digital subscriber line) as shown in FIG. **1**. In addition, the HH units have scanners such as bar code scanners or radio frequency identification (RFID) scanners for scanning, counting and inputting the store inventory to the BO web server.

[0018] Typically, the hand-held devices are personal digital assistants (PDAs). They contain microprocessors and about 64 megabytes of random access memory (RAM). These hand-held devices have the ability to perform as web browsers displaying web pages from the Internet. They communicate with wireless routers via wireless technologies such as the standard WiFi 802.11 b and g protocols. Typical hand-held devices are the Symbol PDA (Motorola, Holtsville, N.Y.), the Intermec CN2 PDA (Intermec Technologies, Everett, Wash.), and the Falcon PSC 4200 PDA (Datalogic (formerly PSC), Bologna, Italy).

[0019] FIG. **2** shows a more detailed block diagram of the device and the store system where it is used. Two hand-held units or PDAs (personal data assistants) with bar code scanners are shown **21**, **22**. These are shown communicating with an out of store BO web server **24** via wireless connections **25**, **26** respectively. FIG. **2** also shows both local and remote web browsers **23** accessing the BO web server, via a high-speed connection such as DSL **27**, in order to view store and company status and to input store data. In addition, the HH devices **21**, **22** are also web browsers. They also have the ability to view store and company status and to input store data such as inventory and inventory checking results. Since the HH devices **21**,**22** have no special program and limited storage, the HH programming is located in the BO server **24**.

[0020] Web Server **24** illustrates a BO implemented with Apache PHP using relational databases. PHP (recursive acronym for "PHP: Hypertext Preprocessor") is a widely-used Open Source general-purpose scripting language that is especially suited for Web development and can be embedded into HTML (hypertext markup language). Apache is a web server currently used in over half of all web sites worldwide. PHP code is different from a script written in other languages like Perl or C. Instead of writing a program with lots of commands to output HTML, one can write an HTML script with some embedded code to do something (i.e. to output some text). The PHP code is enclosed in special start and stop tags that allow a jump into and out of "PHP mode". What distinguishes PHP from something like client-side JavaScript is that the code is executed on the server. If there is HTML script with PHP code imbedded in it, the client would receive the results of running that script, with no way of determining what the underlying code may be. A web server can even be configured to process all HTML files with PHP. The advantages of using PHP are that it is extremely simple for a newcomer, free to use, but offers many advanced features for a professional programmer. A programmer can jump in and in a short time start writing simple scripts. Remote and local web browsers **23** illustrate a method used by store managers, corporate administrators, or system administers to access and view a company & its store's instantaneous performance from minute to minute. The POS, BO, and Web Browsers all communicate via the Web **27**. The protocols between the PDAs **21**, **22**, the web browsers **23** and BO server **24** are HTTPs, XMLURPC. **35**. XML-RPC is a remote procedure call protocol which uses XML to encode its calls and HTTP as a transport mechanism. It is a very simple protocol which defines only a handful of data types and commands.

[0021] The system in FIG. **2** is used for 1) real time scanning for inputting inventory and 2) real time scanning for outputting inventory for checking and comparing counted inventory on the store shelves. This system is very useful as a check against theft and faulty store/company operations.

[0022] In FIG. **2**, all inventory and transaction data is stored in relational databases in the BO server **24**. A relational database stores all of its data inside tables. All operations on data are done on the tables themselves. Some operation produce other tables as the result. A table is a set of rows and columns. Each row is a set of columns with only one value for each. All rows from the same table have the same set of columns, although some columns may have NULL values. A NULL value is an "unknown" value. The rows from a relational table are analogous to a record, and the columns are analogous to a field. Below is an example of a relational table.

NAME	COMPANY	E_MAIL
Jane A. Doe	ABC	jad@abc.com
Bill X. Smith	XYZ	bxs@xyz.com

There are two basic operations one can perform on a relational table. The first one is retrieving a subset of its columns. The second is retrieving a subset of its rows. The field names such as company describe the content of the columns of the relational table. The rows delineate the individual records stored in the relational tables.

[0023] FIG. **3** shows an example flowchart that is implemented in programming in the BO web server **24** shown in FIG. **2**. The purpose of the flowchart in FIG. **3** is to prompt the store worker using the HH device to check the store inventory and to report to the store worker if the in-store count is correct. The first step **31** is to use the hand-held PDA to scan in the candy bar packages on the shelf. Next, the candy bar shelf count is compared **32** to the number of candy bars in the back office database. If the back office database candy bar count does not equal the shelf count **37**, the flowchart directs the worker to count again **33**. If that count comparison fails again **38**, the flowchart tells the store worker to begin theft/loss investigations **35**. If the comparisons are equal **36**, the program flow exits normally with acknowledgement of a successful count comparison. Typically, the flowchart of FIG. **3** would be implemented in the PHP web server programming language discussed above.

[0024] The key advantages of this hand-held web browser inventory checking system are as follows. The hand-held devices have live wireless web connections to a back office (BO) server outside of the store. Also, there is no special programming required in the hand-held device. All operations are reflected real-time into the BO server's inventory database. For example, all inventory scanned or inputted from the hand-held device appears instantly in the BO database, which is instantly viewable anywhere in the world via web browsers. The hand-held devices can check in store inventory against the expectations of the BO server in order to check real time against theft and loss. Another big advantage is that all hand-held computer programs are located and executed in the BO server. In fact, the hand-held device can do all back office operations. For example, from the hand-held device, products can be added, deleted, and their prices can be changed. Since these changes are immediately visible from any manager's PC at their home or at headquarters, there is always management oversight of these changes. This hand-held device allows for local in-store flexibility, but also provides for corporate control and standardization. It also eliminates the need for any server to be located in the store. Another advantage of this system is the use of standard PC and web architecture which offers both full-scalability without degrading system performance. This results in improved performance and lower cost of implementing these business systems. There is a lower cost associated with projects developed with the technology of this invention due to the flexibility of easy design changes and well-understood software. There is less training required for programmers and system testers. Projects can draw on the huge talent pool in the open source development community. The invention allows configurable software modules for different types of businesses and sales promotions. The invention allows remote monitoring of both the hand-held devices and back-office systems from anywhere via the web. There is minimal time required for hand-held device installation, since hand-held device setup is as basic as a home PC setup. Another advantage is that the hand-held inventory checking system can be provided as a service or deployed within a corporation. For example, Software as a Service (SMS) is a software distribution model in which applications are hosted by a vendor or service provider and made available to customers over a network, typically the Internet. Another advantage of this invention is that the hand-held system is maintained in customer centric databases, making it impossible for customers to see other's data. Each hand-held inventory system client gets their own instance of a database. When they log into the BO they are

attached to their own relational database associated and validated via their user login and password.

[0025] While this invention has been particularly shown and described with Reference to the preferred embodiments thereof, it will be understood by those Skilled in the art that various changes in form and details may be made without Departing from the spirit and scope of this invention.

What is claimed is:

1. A hand-held web browser inventory checking system, comprising:

one or more hand-held terminals and

one or more web servers,

wherein said hand-held terminals and said web servers communicate via wireless Internet protocols,

wherein said hand-held terminals are implemented with personal digital assistants (PDAs),

wherein said web server is implemented with web server hardware and software, using one or more relational databases,

wherein programming for said hand-held devices resides in and is executed from said web server.

2. The hand-held web browser inventory checking system of claim **1** wherein said hand-held terminals and said web servers are implemented with executable, configurable software modules, wherein said modules can be easily changed to implement different functions for different businesses.

3. The hand-held web browser inventory checking system of claim **1**, wherein said hand-held terminals can scan codes which identify individual inventory items.

4. The hand-held web browser inventory checking system of claim **1**, wherein said hand-held system and said web servers are scalable allowing the addition of hand-held terminals and business locations without degrading system performance.

5. The hand-held web browser inventory checking system of claim **1**, wherein a real-time status of said hand-held terminals and said web servers is obtained at remote locations via personal computers with browser based web access.

6. A hand-held web browser system comprising:

one or more web servers,

wherein said web servers are implemented with web server hardware and software, using one or more relational databases,

wherein said web servers can communicate with hand-held terminals via Internet protocols,

wherein programming for said hand-held devices resides in and is executed from said web servers.

7. The hand-held web browser of claim **6**, wherein said web servers are implemented with executable, configurable software modules, wherein said modules can be easily changed to implement different functions for different businesses.

8. The hand-held web browser of claim **6**, wherein said web servers issue commands to said hand-held terminals instructing store owner how to check inventory.

9. The hand-held web browser system of claim **6**, wherein said web servers are scalable allowing the addition of hand-held terminals and business locations without degrading system performance.

10. The hand-held web browser system of claim **6**, wherein a real-time status of said hand-held terminals and/or said back-office system is obtained at remote locations via personal computers with browser based web access.

11. A hand-held web browser system comprising,

one or more hand-held terminals,

wherein said hand-held terminals can communicate with web servers via Internet protocols,

wherein said hand-held terminals are implemented with personal computer hardware and software, using one or more relational databases,

wherein programming for said hand-held devices resides in and is executed from said web server.

12. The hand-held web browser system of claim **11**, wherein said hand-held terminals are implemented with executable, configurable software modules, wherein said modules can be easily changed to implement different functions for different businesses.

13. The hand-held web browser system of claim **11**, wherein said hand-held terminals can scan codes which identify individual inventory items.

14. The hand-held web browser system of claim **11**, wherein said hand-held web browser system is scalable allowing the addition of said hand-held terminals and business locations without degrading system performance.

15. The hand-held web browser system of claim **11**, wherein a real-time status of said hand-held terminals and/or said web servers is obtained at remote locations via personal computers with browser based web access.

16. A method of providing a hand-held web browser inventory checking system, comprising the steps of:

providing one or more hand-held terminals and

providing one or more web servers,

wherein said hand-held terminals and said web servers communicate via wireless Internet protocols,

wherein said hand-held terminals are implemented with personal digital assistants (PDAs),

wherein said web server is implemented with web server hardware and software, using one or more relational databases,

wherein programming for said hand-held devices resides in and is executed from said web server.

17. The method of providing a hand-held web browser inventory checking system of claim **16** wherein said hand-held terminals and said web servers are implemented with executable, configurable software modules, wherein said modules can be easily changed to implement different functions for different businesses.

18. The method of providing a hand-held web browser inventory checking system of claim **16**, wherein said hand-held terminals can scan codes which identify individual inventory items.

19. The method of providing a hand-held web browser inventory checking system of claim **16**, wherein said hand-held system and said web servers are scalable allowing the addition of hand-held terminals and business locations without degrading system performance.

20. The method of providing a hand-held web browser inventory checking system of claim **16**, wherein a real-time status of said hand-held terminals and said web servers is obtained at remote locations via personal computers with browser based web access.

21. A method of providing a hand-held web browser system comprising the steps of:

providing one or more web servers,

wherein said web servers are implemented with web server hardware and software, using one or more relational databases,

wherein said web servers can communicate with hand-held terminals via Internet protocols,

wherein programming for said hand-held devices resides in and is executed from said web servers.

22. The method of providing a hand-held web browser of claim **21**, wherein said web servers are implemented with executable, configurable software modules, wherein said modules can be easily changed to implement different functions for different businesses.

23. The method of providing a hand-held web browser of claim **21**, wherein said web servers issue commands to said hand-held terminals instructing store owner how to check inventory.

24. The method of providing a hand-held web browser system of claim **21**, wherein said web servers are scalable allowing the addition of hand-held terminals and business locations without degrading system performance.

25. The method of providing a hand-held web browser system of claim **21**, wherein a real-time status of said hand-held terminals and/or said web servers is obtained at remote locations via personal computers with browser based web access.

26. A method of providing a hand-held web browser system comprising the steps of:

providing one or more hand-held terminals,

wherein said hand-held terminals can communicate with web servers via Internet protocols,

wherein said hand-held terminals are implemented with personal computer hardware and software, using one or more relational databases,

wherein programming for said hand-held devices resides in and is executed from said web server.

27. The method of providing a hand-held web browser system of claim **26**, wherein said hand-held terminals are implemented with executable, configurable software modules, wherein said modules can be easily changed to implement different functions for different businesses.

28. The method of providing a hand-held web browser system of claim **26**, wherein said hand-held terminals can scan codes which identify individual inventory items.

29. The method of providing a hand-held web browser system of claim **26**, wherein said hand-held web browser system is scalable allowing the addition of said hand-held terminals and business locations without degrading system performance.

30. The method of providing a hand-held web browser system of claim **26**, wherein a real-time status of said hand-held terminals and/or said web servers is obtained at remote locations via personal computers with browser based web access.

* * * * *

EXHIBIT C1

TY06-003

Quentin Olson
Wayne Baratta

*sm.entity

Conf. # 2323

To PTO 2/1/08
w/o declaration

Ser # 12/012,393 2/01/08

Ralph

Filing Receipt 3/12/08
Not to file Missing Parts 3/12/08
Response 7/14/08 (2 mos. late)
Info. Disc. Stmt. - 08/01/08
Rejection 1/2/09
Amendment 4/2/09
Not of Abandonment 10/2/09
Not. of Pub. 8/6/09
Petition 12/11/09
Decision on Petition (Granted) 5/13/10
Final Rejection 7/2/10
Amend AF 9/2/10
Advisory Action 9/28/10
RCE w/ amend 11/2/10
Rejection 11/23/10
Amendment 4/25/11
Final Rejection 6/22/11
Amend AF 8/22/11
Advisory Action 9/22/11
RCE W/ Amendment 11/22/11
Final Rejection 12/27/11
Amend AF 2/27/12
Advisory Action 3/13/12
Notice of Appeal 3/27/12

No Assignment per SBA

~~Assignee: TYRAMM International Inc.~~

Appeal Number
2013-001954

TY06-003

EXHIBIT C2

US 20080208698A1

(19) United States

(12) Patent Application Publication

(10) Pub. No.: US 2008/0208698 A1

(43) Pub. Date: Aug. 28, 2008

Olson et al.

(54) POINT OF SALE VIDEO SERVER PROCESSING SYSTEM

(76) Inventors: **Quentin Olson**, Poulsbo, WA (US); **Wayne Baratta**, Poughkeepsie, NY (US)

Correspondence Address:
George O Saile and Assocs
28 Davis Ave
Poughkeepsie, NY 12603

(21) Appl. No.: **11/710,723**

(22) Filed: **Feb. 26, 2007**

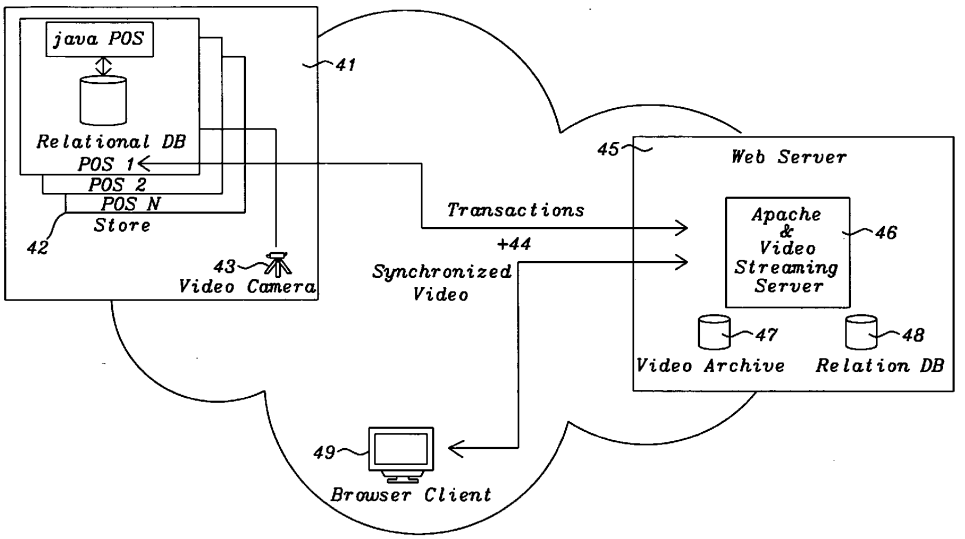
Publication Classification

(51) **Int. Cl.**
G06Q 20/00 (2006.01)

(52) **U.S. Cl.** **705/21**

(57) **ABSTRACT**

This invention provides a system and a method for low cost video surveillance and archiving which is synchronized with point of sale (POS) transactions. There is no local storage of video at the POS. There is high speed transmission of the video out of the stores to central servers. All back office applications are browser based. The system provides for inherent video compression due to hardware and software involved with the video capture. Management can watch present or past store activity anywhere anytime via web browsers. Video records are indexed and retrieved from server memory along with corresponding transaction information.



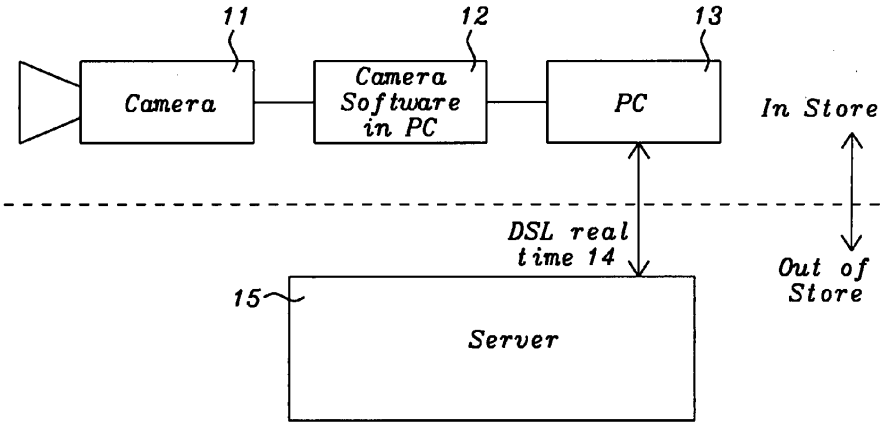


FIG. 1

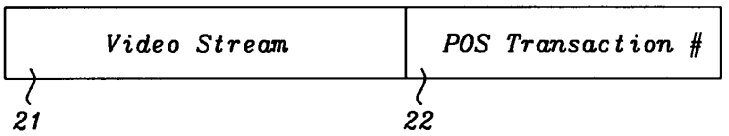


FIG. 2

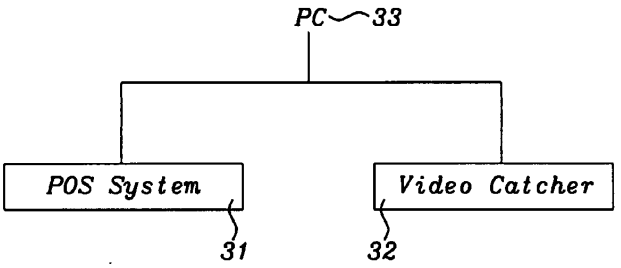


FIG. 3

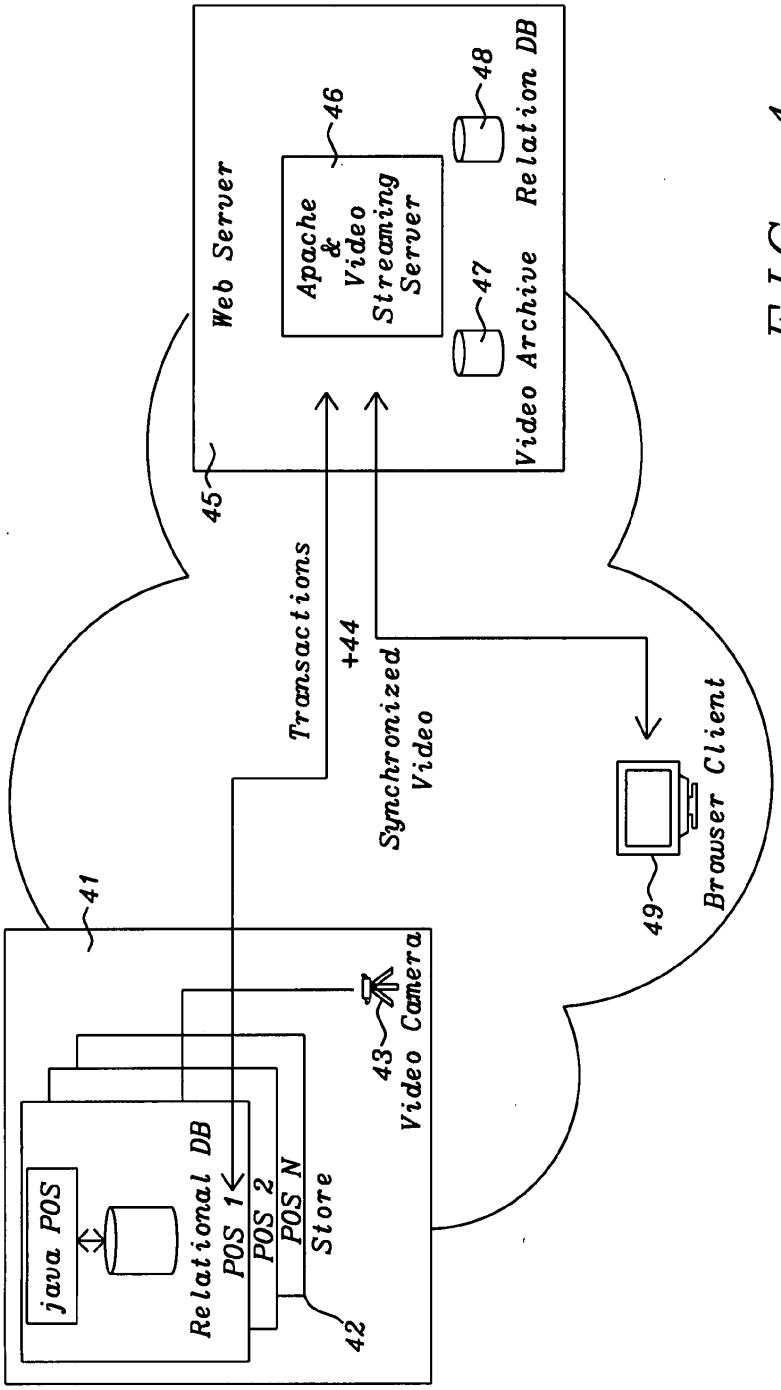

Web Server
45
Apache & Video Streaming Server
46
Video Archive
47
Relation DB
48
Transactions
+44
Synchronized Video
Browser Client
49
41
Video Camera
43
Relational DB
POS 1
POS 2
POS N
Store
42
java POS

FIG. 4

POINT OF SALE VIDEO SERVER PROCESSING SYSTEM

RELATED PATENT APPLICATION

[0001] This application is related to docket number TY2006-001, filed on _____, Ser. No. _____, which is herein incorporated by reference in its entirety.

BACKGROUND OF THE INVENTION

[0002] 1. Field of the Invention

[0003] This invention relates to a system and a method for collecting and retrieving video and business transactions. The business operations range from a single branch to a large chain of stores or branches.

[0004] More particularly this invention relates to the web-based back-office server processing of video of point of sale (POS) business transactions to assist in the management and security of businesses.

[0005] 2. Description of Related Art

[0006] Current practice includes large, expensive digital video recording systems at each point of sale terminal. It includes special software to be installed on point of sale client (POS) computers. Most current surveillance systems only provide current/real-time views without the ability to retrieve previous video. Systems that do record are expensive and hard to manage. They need to maintain a library of tapes or CD. Typically, video based archival is a resource intensive operation due to the size of video files. Current systems provide text on video, which is not indexed for retrieval based on a specific POS transaction. The disadvantages of current practice are that large, expensive networks are required. The problem to be solved is the desire for a video system that is low cost, easily retrievable transaction videos for store security including motion detection and salesperson integrity.

[0007] U.S. Pat. No. 6,504,481 B2 (Teller) describes a system, method, and device for monitoring the dispensing of liquids, in particular, beverages dispensed at a bar or restaurant. The system includes an electronic sensor device, data receiver, personal computer, bottle opener/camera system, bar code scanner, point of sale, bar camera, network, Web server, and paging system. The method automatically detects and monitors dispensing events, including detecting the type and amount of liquid, reconciling dispensing events with register ring-ups, recording information, saving video specific to the dispensing event, and sending a page indicating anomalies.

[0008] U.S. Pat. No. 6,847,393 B2 (Ashe et al.) describes a method and system for monitoring point of sale exceptions including the ability to automatically analyze POS transactional data for certain patterns or occurrences that typically indicate specific behavior at or near a POS terminal in connection with a sales related operation.

[0009] U.S. Pat. No. 6,726,094 B1 (Rantze et al.) discloses a customer self-service terminal for use in conducting retail transactions. The terminal includes a camera having multi-mode image capability. The camera is used to scan barcodes, check signing, credit card transactions, cash tendering, and general store video capture.

[0010] U.S. Pat. No. 6,343,739 B1 (Lippert) describes a method of providing security to operation of a self-service checkout terminal having a video camera and a printing device associated with the camera. The method captures video of the customer and determines if the customer has paid properly.

[0011] U.S. Pat. No. 5,895,453 (Cook) discloses a method and system for the detection of suspect sales transactions in a sales environment having point of sale devices and sales locations. The method and system further involve the use of surveillance cameras in which the output of the camera is keyed to the particular transactions of interest.

BRIEF SUMMARY OF THE INVENTION

[0012] It is the objective of this invention to provide a system and a method for collecting and retrieving video and business transactions. The business operations range from a single branch to a large chain of stores or branches.

[0013] It is further an object of this invention to process video of point of sale (POS) business transactions to assist in the management and security of businesses.

[0014] The objects of this invention are achieved by a point of sale video server processing system, comprised of one or more point of sale (POS) terminals, one or more cameras, and one or more web servers. The POS terminals and web servers communicate via internet protocols. The POS terminals are implemented with point of sale personal computer hardware and software using one or more relational databases. The video captured by said cameras is processed and organized by video software residing on the point of sale personal computer hardware, video clips are captured and sent to the web server for retrieval. The web server is implemented with web server hardware and software, using one or more relational databases.

BRIEF DESCRIPTION OF THE DRAWINGS

[0015] FIG. **1** shows a block diagram of the elements of this invention.

[0016] FIG. **2** gives a data structure for correlating transaction data and transaction videos.

[0017] FIG. **3** illustrates a point of sale system PC terminal with software to support a POS transaction system and the video catcher system of this invention.

[0018] FIG. **4** is a more detailed diagram of a main embodiment of this invention.

DETAILED DESCRIPTION OF THE INVENTION

[0019] FIG. **1** shows a high level diagram of this invention. The dotted line **10** delineates the in store and out of store elements. The camera **11** is usually placed at or near each point of sale (POS) terminal. POS terminals would typically be implemented by standard personal computers such as PC **13** in FIG. **1**. The camera can capture each customer transaction as well as be active at all times. The camera interfaces with camera software **12** which is resident in a personal computer, PC, **13**. The camera connects to the PC via universal interfaces. The camera software has inherent video compression. The PC is connected to the internet via a broadband connection such as DSL. The video captured by the camera **11** and indexed by the camera software in the PC is stored in real-time at the server **15** outside of the store. The PC **13** in FIG. **1** also can perform the POS transactions. The PC software **12** would correlate POS transaction data and the video capture data from the camera **11**.

[0020] Each POS normally operates with a hardware/software connection **14** to the Internet or Web. However, if the

web goes down, the POS terminal continues to operate. There is a loose coupling of the POS to the back office (BO) server **15**. The POS to BO connection is not required for the basic business functions of the POS. All transaction data is stored in a relational database on the hard drive in the POS.

[0021] A relational database stores all of its data inside tables. All operations on data are done on the tables themselves. Some operation produce other tables as the result. A table is a set of rows and columns. Each row is a set of columns with only one value for each. All rows from the same table have the same set of columns, although some columns may have NULL values. A NULL value is an "unknown" value. The rows from a relational table are analogous to a record, and the columns are analogous to a field. Below is an example of a relational table.

NAME	COMPANY	E_MAIL
Jane A. Doe	ABC	jad@abc.com
Bill X. Smith	XYZ	bxs@xyz.com

There are two basic operations one can perform on a relational table. The first one is retrieving a subset of its columns. The second is retrieving a subset of its rows. The field names such as company describe the content of the columns of the relational table. The rows delineate the individual records stored in the relational tables.

[0022] As transactions are created at a POS a log entry for the newest transaction is also created, this log entry is used to flag if the transaction has been uploaded to the web server. Part of the POS application, a transaction engine, is continuously running in the background. This component reads the log of transactions. If a transaction needs to be sent, it tries to send it. If the send fails (for example, if the connection to, or the Internet itself, is down), it goes to sleep and tries again later. The POS terminals communicate via HTTP protocol (hypertext transfer protocol) **14** with Back-office BO software, which is implemented on web servers **15**, which can be located anywhere in the world. In addition, the BO software and data can be viewed from any store employee at any PC who has Internet access and a password.

[0023] A typical data structure for video storage at the server **15** is illustrated by the following example. The video is captured at the POS in five minute clips, indexed by the same clock as the POS application then sent to the BO for archival and later retrieval. Since the video indexing is synchronized at the POS and those same POS transactions are available at the server through the standard BO functionality specific retrieval of video correlated to any POS event is possible.

[0024] FIG. **2** shows a data structure where the video stream **21** captured by the camera. In addition, FIG. **2** shows the corresponding POS transaction data **22**. Both the video **21** and the transaction **22** are captured at the point of sale terminal in the store. They both are indexed by the same POS clock. Then, both the video **21** and the transaction data **22** are sent to the back office at the server outside of the store for archiving and later retrieval.

[0025] FIG. **3** shows how both the POS system **31** and the video catcher or capture system **32** co-exist in each POS PC **33** terminal in the stores or business branches. A given store or business branch could have several PC POS terminals, each of which would have the system elements shown in FIG. **3**.

[0026] As transactions are created at a POS **31**, a log entry for the newest transaction is also created, this log entry is used to flag if the transaction and its corresponding video has been uploaded to the web server. Part of the POS application **31**, the BO interface, is continuously running in the background. This component reads the log of transactions. If a transaction and its corresponding video needs to be sent, it tries to send it. If the send fails (for example, if the connection to, or the Internet itself, is down), it goes to sleep and tries again later. Additionally, the BO interface requests update from the BO such as new items, price changes employees, etc.

[0027] FIG. **4** is a more detailed diagram of the main embodiment of this invention. Store **41** has several POS PC terminals **42** implemented with Java software using relational databases. The video cameras **43** are also shown. Typically, there would be one camera per POS terminal. The POS transaction plus the synchronized video **44** are sent to the web server **45** via a broadband connection such as DSL. The web server consists of a back-office transaction server plus a video streaming server **46**. The web server **45** also includes a video archive data storage **47** and a relational database **48**. Remote and local client web browsers **49** illustrate a method for store managers, corporate administrators, or system administrators to access and view a company's and its store's instantaneous performance along with individual POS terminal transactions and videos.

[0028] The web server **46** illustrates a back-office server implemented with Apache PHP using relational databases. PHP (recursive acronym for "PHP: Hypertext Preprocessor") is a widely-used Open Source general-purpose scripting language that is especially suited for Web development and can be embedded into HTML (hypertext markup language). Apache PHP is one offering of the PHP language. PHP code is different from a script written in other languages like Perl or C. Instead of writing a program with lots of commands to output HTML, one can write an HTML script with some embedded code to do something (ie. to output some text). The PHP code is enclosed in special start and stop tags that allow you to jump into and out of "PHP mode". What distinguishes PHP from something like client-side JavaScript is that the code is executed on the server. If you were to have HTML script with PHP code imbedded in it, the client would receive the results of running that script, with no way of determining what the underlying code may be. You can even configure your web server to process all your HTML files with PHP. The advantages of using PHP are that it is extremely simple for a newcomer, but offers many advanced features for a professional programmer. A programmer can jump in and in a short time start writing simple scripts. Remote and local web browsers **33** illustrate a method used by store managers, corporate administrators, or system administers to access and view a company & its store's instantaneous performance from minute to minute. The POS, BO, and Web Browsers all communicate via the Web **44**.

[0029] The key advantages of this point of sale video server processing system. This system and method offers a low cost web archiving of video which is indexed and retrievable with its corresponding POS event/transaction data. Most current surveillance systems only provide current/real-time views without the ability to retrieve previous video. Systems that do record are expensive and hard to manage. Additionally, since this system stores the video on the web server management of the video by the client is vastly simplified, they no longer need

to maintain a library of tapes or CD and all video is played back through a browser based web interface so no special application is required.

[0030] Typically video based archival is a resource intensive operation due to the size of video files, this application is further enabled through the use of the H264/AVC codec which provides up to 50% better compression than other video formats currently in use.

[0031] The web and PC architecture both offer full-scalability which results in improved performance and lower cost. There is a lower cost associated with projects developed with the technology of this invention due to the flexibility of easy design changes and well-understood software. There is less training required for programmers and system testers. Projects can draw on the huge talent pool in the open source development community. The invention allows configurable modules for different types of businesses and sales promotions. The invention allows remote monitoring of the POS terminals, the video software and back-office systems from anywhere via the web. There is minimal time required for installation of the POS terminals and the video software, since the POS terminal setup is as basic as a home PC setup. Another advantage is that the back-office web servers and video streaming servers can be provided as a service or deployed within a corporation. For example, Software as a Service (SAAS) is a software distribution model in which applications are hosted by a vendor or service provider and made available to customers over a network, typically the Internet. Another advantage of this invention is that the back-office system and the video streaming system is maintained in customer centric databases, making it impossible for customers to see other's data. Each store client (individual POS, store or multi-store owner) which includes a store's individual video capture system gets their own instance of a database. When they log into the back-office and video streaming servers, they are attached to their own relational database associated and validated via their user login and password. This method provides easily retrievable transaction videos and data for store security including motion detection and salesperson integrity.

[0032] While this invention has been particularly shown and described with Reference to the preferred embodiments thereof, it will be understood by those Skilled in the art that various changes in form and details may be made without Departing from the spirit and scope of this invention.

What is claimed is:

1. A point of sale video server processing system, comprising:

one or more point of sale (POS) terminals,

one or more cameras, and

one or more web servers,

wherein said POS terminals and said web servers communicate via internet protocols,

wherein said POS terminals are implemented with point of sale personal computer hardware and software using one or more relational databases,

wherein video captured by said cameras is processed and organized by video software residing on said point of sale personal computer hardware,

wherein said web server is implemented with web server hardware and software, using one or more relational databases.

2. The point of sale video server processing system of claim **1**, wherein said POS terminals, said video software, and said web servers are implemented with executable, configurable software modules, wherein said modules can be changed to implement different functions for different businesses.

3. The point of sale video server processing system of claim **1**, wherein said POS terminals and said video software can operate standalone without communicating with said web server.

4. The point of sale video server processing system of claim **1**, wherein said POS terminals, said video software, and said web-based back-office servers are scalable allowing the addition of POS terminals and business locations without degrading system performance.

5. The point of sale video server processing system of claim **1**, wherein a real-time status of said POS terminals, said video software, and said back-office web servers is obtained at remote locations via personal computers with web access, wherein said status includes ability to review captured video, which may or may not be correlated to customer transactions.

6. A web-based back-office system comprising:

one or more web servers, and

one or more video streaming servers,

wherein said web servers and said video streaming servers are implemented with web server hardware and software and video streaming hardware and software, using one or more relational databases,

wherein said web servers and said video streaming servers have communication with point of sale (POS) terminals and with video software via internet protocols,

wherein said communication includes movement of POS transactions and synchronized video to said web servers and to said video streaming servers respectively.

7. The web-based back-office system of claim **6**, wherein said web servers and said video streaming servers are implemented with executable, configurable software modules, wherein said modules can be changed to implement different functions for different businesses.

8. The web-based back-office system of claim **6**, wherein said web servers and said video streaming servers allow said POS terminals and said video software to operate standalone without communicating with said web servers and said video streaming servers.

9. The web-based back-office system of claim **6**, wherein said web-based back office is scalable allowing the addition of POS terminals, cameras, and business locations without degrading system performance.

10. The web-based back-office system of claim **6**, wherein a real-time status of said POS terminals, said video software, and said back-office system is obtained at remote locations via personal computers with web access, wherein said status includes ability to review captured video, which may or may not be correlated to customer transactions.

11. A point of sale POS system comprising,

one or more point of sale (POS) terminals, and

one or more cameras,

wherein said POS terminals can communicate with web servers via internet protocols,

wherein said POS are implemented with point of sale personal computer hardware and software, using one or more relational databases,

wherein video captured by said cameras is processed and organized by video software residing on said point of sale personal computer hardware.

12. The point of sale system of claim **11**, wherein said POS terminals and said video software are implemented with

executable, configurable software modules, wherein said modules can be changed to implement different functions for different businesses.

13. The point of sale system of claim **11**, wherein said POS terminals and said video software can operate standalone without communicating with said web servers.

14. The point of sale system of claim **11**, wherein said POS terminals and said video software are scalable allowing the addition of POS terminals and business locations without degrading system performance.

15. The point of sale system of claim **11**, wherein a real-time status of said POS terminals, said video software, and said back-office web servers is obtained at remote locations via personal computers with web access, wherein said status includes ability to review captured video, which may or may not be correlated to customer transactions.

16. A method of providing a point of sale video server processing system, comprising the steps of:

providing one or more point of sale (POS) terminals,

providing one or more cameras, and

providing one or more web servers,

wherein said POS terminals and said web servers communicate via internet protocols,

wherein said POS terminals are implemented with point of sale personal computer hardware and software using one or more relational databases,

wherein video captured by said cameras is processed and organized by video software residing on said point of sale personal computer hardware,

wherein said web server is implemented with web server hardware and software, using one or more relational databases.

17. The method of providing a point of sale video server processing system of claim **16**, wherein said POS terminals, said video software, and said web servers are implemented with executable, configurable software modules, wherein said modules can be changed to implement different functions for different businesses.

18. The method of providing a point of sale video server processing system of claim **16**, wherein said POS terminals and said video software can operate standalone without communicating with said web server.

19. The method of providing a point of sale video server processing system of claim **16**, wherein said POS terminals, said video software, and said web-based back-office servers are scalable allowing the addition of POS terminals and business locations without degrading system performance.

20. The method of providing a point of sale video server processing system of claim **16**, wherein a real-time status of said POS terminals, said video software, and said back-office web servers is obtained at remote locations via personal computers with web access, wherein said status includes ability to review captured video, which may or may not be correlated to customer transactions.

21. A method of providing a web-based back-office system comprising the steps of:

providing one or more web servers, and

providing one or more video streaming servers,

wherein said web servers and said video streaming servers are implemented with web server hardware and software and video streaming hardware and software, using one or more relational databases,

wherein said web servers and said video streaming servers have communication with point of sale (POS) terminals and with video software via internet protocols,

wherein said communication includes movement of POS transactions and synchronized video to said web servers and to said video streaming servers respectively.

22. The method of providing a web-based back-office system of claim **21**, wherein said web servers and said video streaming servers are implemented with executable, configurable software modules, wherein said modules can be changed to implement different functions for different businesses.

23. The method of providing a web-based back-office system of claim **21**,

wherein

said web servers and said video streaming servers allow said POS terminals and said video software to operate standalone without communicating with said web servers and said video streaming servers.

24. The method of providing a web-based back-office system of claim **21**, wherein said web-based back office is scalable allowing the addition of POS terminals, cameras, and business locations without degrading system performance.

25. The method of providing a web-based back-office system of claim **21**, wherein a real-time status of said POS terminals, said video software, and said back-office system is obtained at remote locations via personal computers with web access, wherein said status includes ability to review captured video, which may or may not be correlated to customer transactions.

26. A method of providing a point of sale POS system comprising the steps of:

providing one or more point of sale (POS) terminals, and

providing one or more cameras,

wherein said POS terminals can communicate with web servers via internet protocols,

wherein said POS are implemented with point of sale personal computer hardware and software, using one or more relational databases,

wherein video captured by said cameras is processed and organized by video software residing on said point of sale personal computer hardware.

27. The method of providing a point of sale POS system of claim **26**, wherein said POS terminals and said video software are implemented with executable, configurable software modules, wherein said modules can be changed to implement different functions for different businesses.

28. The method of providing a point of sale POS system of claim **26**, wherein said POS terminals and said video software can operate standalone without communicating with said web servers.

29. The method of providing a point of sale POS system of claim **26**, wherein said POS terminals and said video software are scalable allowing the addition of POS terminals and business locations without degrading system performance.

30. The method of providing a point of sale POS system of claim **26**, wherein a real-time status of said POS terminals, said video software, and said back-office web servers is obtained at remote locations via personal computers with web access, wherein said status includes ability to review captured video, which may, or may not be correlated to customer transactions.

* * * * *

EXHIBIT D1

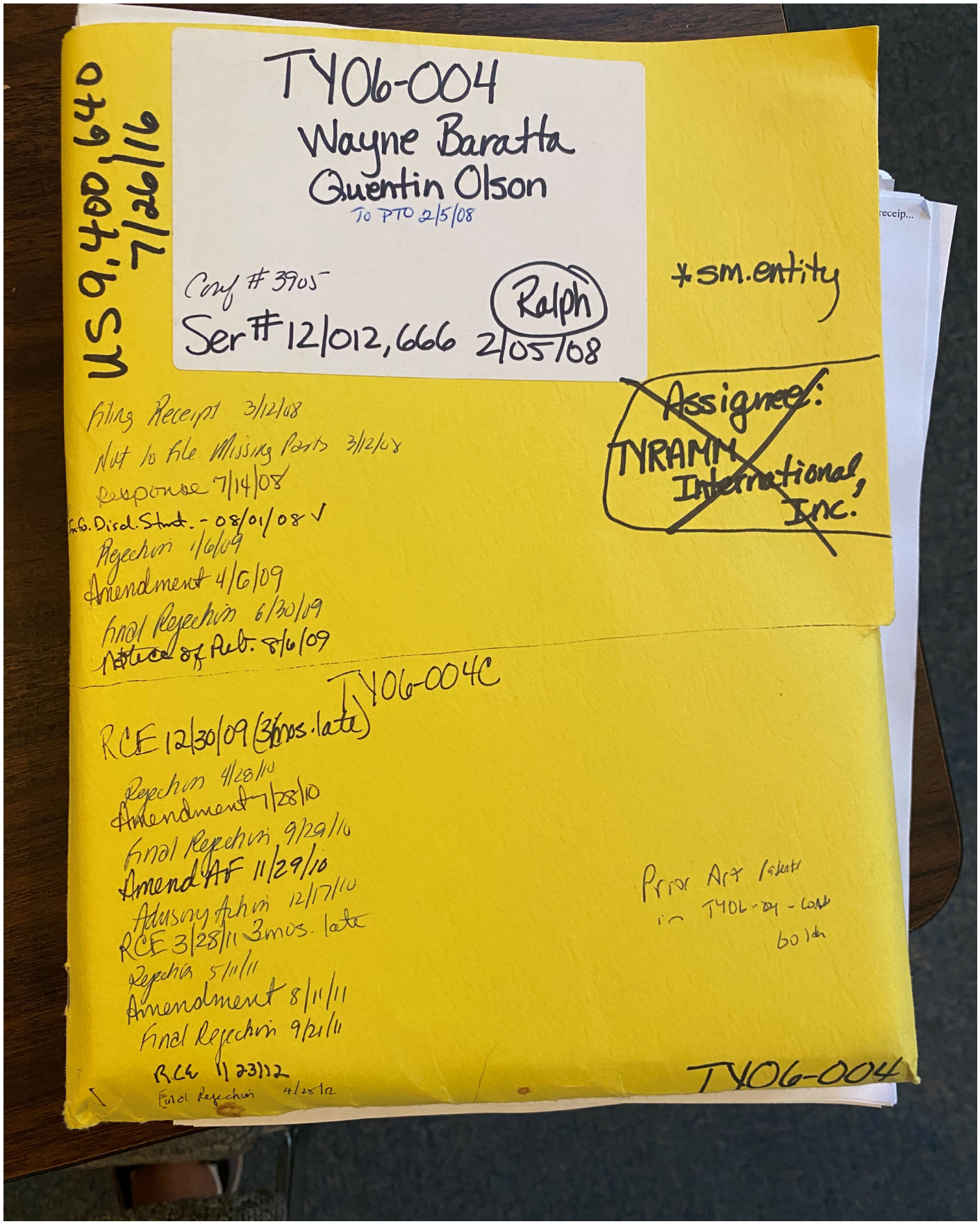

US 9,400,640
7/26/16
TY06-004
Wayne Baratta
Quentin Olson
To PTO 2/5/08
Conf # 3905
Ser # 12/012,666 2/05/08
Ralph
*sm.entity
Assignee:
TYRAMM International, Inc.
Filing Receipt 3/12/08
Not to file Missing Parts 3/12/08
Response 7/14/08
Info. Disd. Stmt. - 08/01/08 ✓
Rejection 1/6/09
Amendment 4/6/09
Final Rejection 6/30/09
Notice of Pub. 8/6/09
TY06-004C
RCE 12/30/09 (3mos. late)
Rejection 4/28/10
Amendment 7/28/10
Final Rejection 9/29/10
Amend AF 11/29/10
Advisory Action 12/17/10
RCE 3/28/11 3mos. late
Rejection 5/11/11
Amendment 8/11/11
Final Rejection 9/21/11
RCE 1/23/12
Final Rejection 4/25/12
Prior Art Patents
in TY06-04 - Cont
Golden
TY06-004

EXHIBIT D2

US009400640B2

(12) **United States Patent**
Baratta et al.

(10) **Patent No.: US 9,400,640 B2**
(45) **Date of Patent: Jul. 26, 2016**

(54) **WEB-BASED POINT OF SALE BUILDER**

(76) Inventors: **Wayne Baratta**, Poughkeepsie, NY (US); **Quentin Olson**, Poulsbo, WA (US)

(*) Notice: Subject to any disclaimer, the term of this patent is extended or adjusted under 35 U.S.C. 154(b) by 897 days.

(21) Appl. No.: **12/012,666**

(22) Filed: **Feb. 5, 2008**

(65) **Prior Publication Data**

US 2009/0199154 A1 Aug. 6, 2009

(51) **Int. Cl.**
G06Q 10/00 (2012.01)
G06F 9/44 (2006.01)
G06Q 30/06 (2012.01)

(52) **U.S. Cl.**
CPC . ***G06F 8/34*** (2013.01); ***G06Q 30/06*** (2013.01)

(58) **Field of Classification Search**
CPC G06Q 30/00; G06Q 20/00
USPC 235/375, 383, 385, 379; 348/61, 143, 348/148, 150, 12, 161; 705/1, 16, 17, 21, 705/22, 28, 54, 35
See application file for complete search history.

(56) **References Cited**

U.S. PATENT DOCUMENTS

5,818,428 A 10/1998 Eisenbrandt et al.
5,956,716 A * 9/1999 Kenner et al.
5,987,426 A 11/1999 Goodwin, III
6,629,080 B1 9/2003 Kolls
7,015,945 B1 * 3/2006 Sullivan 348/150
7,051,091 B1 5/2006 Cohen et al.
2004/0128199 A1 * 7/2004 Cusack G06Q 30/06 705/16
2004/0181454 A1 * 9/2004 Manno G06Q 20/20 705/21
2005/0021409 A1 * 1/2005 Michaud G06Q 20/20 705/16
2008/0077872 A1 * 3/2008 Zeevi G06F 3/0481 715/762

OTHER PUBLICATIONS

IBM 4690 ASTRA—Graphical User Interface (GUI)—Specifications, found Feb. 8, 2008, htp://www-03.ibm.com/roducts/retail/product/sofware/gui/astra/specs.html, 1 page.
Soft32Download, db2k e-Store Builder 3.00 details, Localsoft Tech, found Feb. 8, 2008, http://www.soft32download.com/software/db2k-e-Store-Builder-download-details.html, pp. 1-3.
Co-pending U.S. Appl. No. 12/012,393, filed Feb. 1, 2008, "Hand-Held Web Browser Inventory Checking System," assigned to the same assignee as the present invention.
Co-pending U.S. Appl. No. 11/710,722, filed Feb. 26, 2007, "Point of Sale System with Web-Based Back-Office," assigned to the same assignee as the present invention.
Co-pending U.S. Appl. No. 11/710,723, filed Feb. 26, 2007, Point of Sale Video Server Processing System, assigned to the same assignee as the present invention.

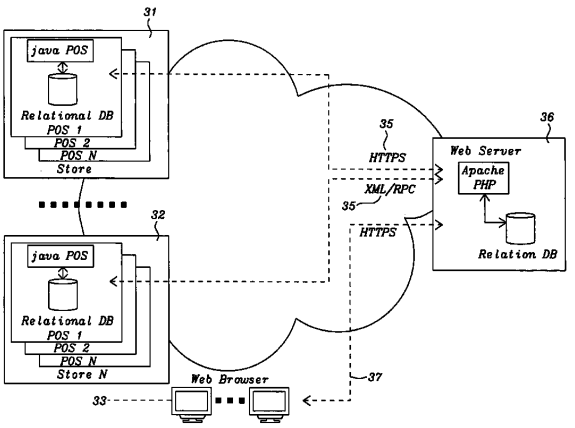
* cited by examiner

Primary Examiner — Rokib Masud
(74) *Attorney, Agent, or Firm* — Saile Ackerman LLC; Stephen B. Ackerman

(57) **ABSTRACT**

This invention provides a system and a method for online, web-based point of sale (POS) building and configuration, which can assist non-expert business operators in building, editing and testing a point of sale system to manage their businesses. The business operations range from a single branch to a large chain of stores or branches. The key advantages of the Web-based POS builder are that it is completely built on the foundation of the Web. The POS builder is accessible anywhere in the world. It can be used by a person of any skill level. The POS builder builds, edits, and tests new POS terminals in real time.

28 Claims, 5 Drawing Sheets



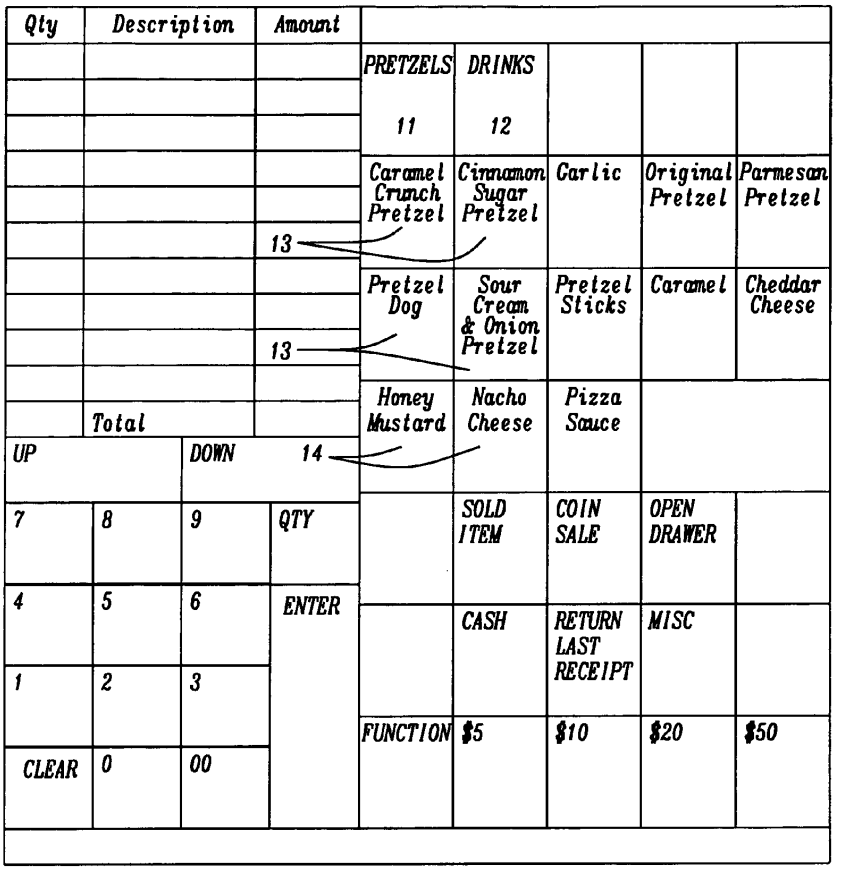

Qty
Description
Amount
Total
UP
DOWN
14
7
8
9
QTY
4
5
6
ENTER
1
2
3
CLEAR
0
00
PRETZELS
DRINKS
11
12
Caramel Crunch Pretzel
Cinnamon Sugar Pretzel
Garlic
Original Pretzel
Parmesan Pretzel
13
Pretzel Dog
Sour Cream & Onion Pretzel
Pretzel Sticks
Caramel
Cheddar Cheese
13
Honey Mustard
Nacho Cheese
Pizza Sauce
SOLD ITEM
COIN SALE
OPEN DRAWER
CASH
RETURN LAST RECEIPT
MISC
FUNCTION
$5
$10
$20
$50

FIG. 1

Qty	Description	Amount
	garlic	2.29
	pizza sauce	0.59
	cranberry grape	1.49
	large sprite	1.50
	Total	

UP		DOWN	
7	8	9	QTY
4	5	6	ENTER
1	2	3	
CLEAR	0	00	

HOME	DRINKS	SMALL	MEDIUM	LARGE
COKE	DIETCOKE	PUNCH		SPRITE
APPLE JUICE	CRANBERRY APPLE RASPBERRY	CRANBERRY GRAPE	DANNON SPRING WATER	MINUTE MAID ORANGE JUICE
COLUMBIAN		VANILLA	MUT	MACADAMIA NUT
	CASH	RETURN LAST RECEIPT	MISC	
FUNCTION	$5	$10	$20	$50

FIG. 2

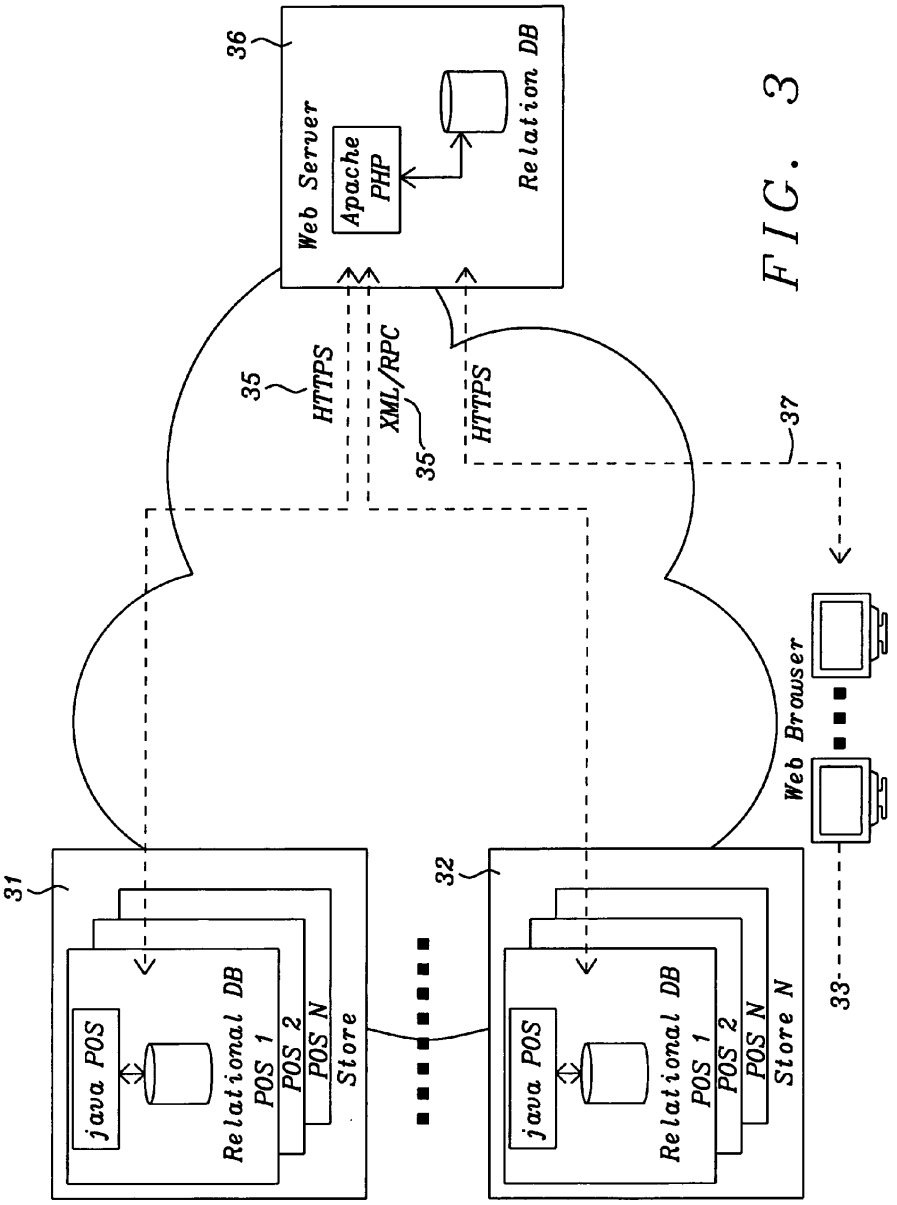

36
Web Server
Apache PHP
Relation DB
35
HTTPS
XML/RPC
35
HTTPS
37
Web Browser
31
java POS
Relational DB
POS 1
POS 2
POS N
Store
32
java POS
Relational DB
POS 1
POS 2
POS N
Store N
33

FIG. 3

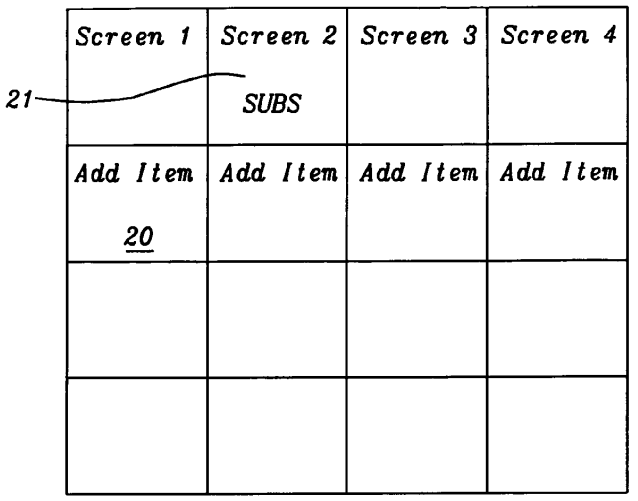

Screen 1
Screen 2
Screen 3
Screen 4
21
SUBS
Add Item
Add Item
Add Item
Add Item
20

FIG. 4a

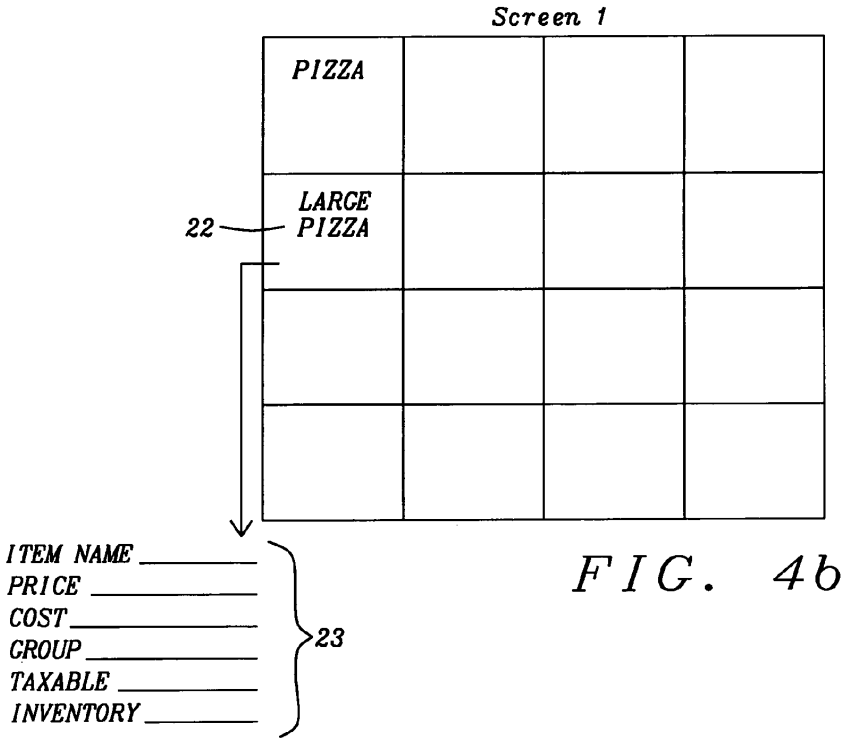

Screen 1
PIZZA
LARGE PIZZA
22
ITEM NAME
PRICE
COST
GROUP
TAXABLE
INVENTORY
23

FIG. 4b

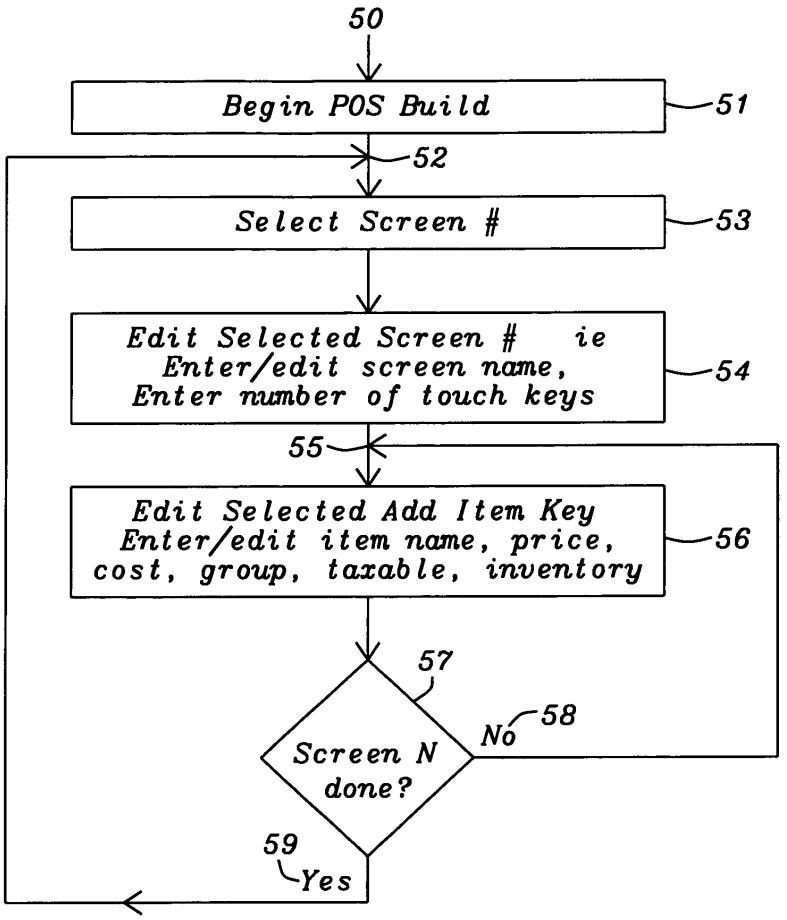

50
Begin POS Build
51
52
Select Screen #
53
Edit Selected Screen # ie
Enter/edit screen name,
Enter number of touch keys
54
55
Edit Selected Add Item Key
Enter/edit item name, price,
cost, group, taxable, inventory
56
57
Screen N
done?
No
58
59
Yes

FIG. 5

WEB-BASED POINT OF SALE BUILDER

RELATED PATENT APPLICATIONS

This application is related to, filed on Feb. 26, 2007, Ser. No. 11,710,722 and, filed on Feb. 26, 2007, Ser. No. 11,710, 723, which are herein incorporated by reference in its entirety.

BACKGROUND OF THE INVENTION

1. Field of the Invention

This invention relates to a system and a method for building a point of sale (POS) system to manage business operations. The business operations range from a single branch to a large chain of stores or branches.

More particularly this invention relates to an online, web-based point of sale builder method, which can assist non-expert business operators in assembling a point of sale system to manage their businesses.

2. Description of Related Art

Current practice in the field of assembling point of sale systems includes manually coding front-of-screen information. Typically, this front-of-screen information contains menu selections, page selections, and general answers to business questions. This front-of-screen menu is typically manually coded by a business expert with the help of a programmer or data expert. Also, currently the entry of this front-of-screen information requires intimate knowledge of a complex interface to a front-of-screen programming language. In summary, current practice includes the manual building of a point of sale (POS) screen. This manual process requires defining the position and operation of touch screen keys and their database correspondence. Currently, only specially trained people can build or change POS screens. This manual POS building and editing is prone to mistakes and is time-consuming. Since POS screen changes are difficult and prone to error, store owners tend to retain older, inaccurate, out-of-date POS screens in order to avoid the POS screen editing process. Also, current POS screen editing occurs off-line with the testing of the screens occurring at a later date, at a remote store location. The following references represent prior art in the field of screen configuration building.

U.S. Pat. No. 5,818,428 (Eisenbrandt et al.) describes a control system with a user configurable interface, particularly suitable for use in connection with appliances. Users can configure display screens either at a point of sale location or at home with a personal computer.

U.S. Pat. No. 6,629,080 B1 (Kolls) describes a universal advertising and payment system and method for networking, monitoring and advancing electronic commerce and controlling vending equipment.

U.S. Pat. No. 7,051,091 B1 (Cohen et al.) discloses a configuration builder useful in configuring software containing hardware units which are serviced by a center which services a multiplicity of similar units having a plurality of different configurations.

U.S. Pat. No. 5,987,426 (Goodwin) describes a system and method of transferring information between a first software application and a second software application which employ an isolation layer. The system includes a client computer system provided by a first seller of computer systems, including a client software application, and a server computer system provided by a second seller of computer system.

BRIEF SUMMARY OF THE INVENTION

It is the objective of this invention to provide a system and a method for building a point of sale (POS) system to manage

business operations. The business operations range from a single branch to a large chain of stores or branches.

It is further an object of this invention to provide an online, web-based point of sale builder system and method, which can assist non-expert or expert business operators in assembling a point of sale system to manage their businesses. This point of sale building operation can be done in real time from anywhere in the world.

The objects of this invention are achieved by a web-based point of sale (POS) builder comprising one or more point of sale terminals, which display POS, screens, an Internet connection to a web server, one or more local or remote PC workstations, and point of sale builder software which runs on said web server. Local or remote workstations can be utilized to build or edit said POS terminals in real time, from anywhere in the world and over the world-wide web.

BRIEF DESCRIPTION OF THE DRAWINGS

FIG. **1** shows a typical point of sale touch screen for a pretzel store, as an example only.

FIG. **2** shows a typical touch screen for the drinks panel of a pizza restaurant, as an example only.

FIG. **3** is a system diagram for web-based back office which supports point of sale terminals.

FIG. **4***a* is a sample screen builder panel before the screen building process begins.

FIG. **4***b* is a sample screen builder panel after the screen building process is under way.

FIG. **5** is a high level flowchart which illustrates the main embodiment of the screen building process.

DETAILED DESCRIPTION OF THE INVENTION

FIG. **1** shows a typical point of sale (POS) touch screen for, as an example only, a pretzel store. There are touch keys for pretzels **11** and for drinks **12**. Selecting these keys would typically bring up secondary screens displaying specific product keys for ordering different types of pretzels and drinks respectively. In addition, the screen in FIG. **1** has some specific pretzel product keys **13** and specific pretzel topping keys **14**. Currently in the prior art, a touch screen as shown in FIG. **1** is manually configured by a programmer who knows the specific proprietary point of sale system used by a store or business. The FIG. **1** screen design involves the specific key layout and size of keys. In addition, the FIG. **1** screen keys must have corresponding hooks or references to product data such as item name, price, cost, group, taxable, and inventory as shown in FIG. **4**. In this invention, this product data and the touch key structure is stored in relational databases in the back office which is stored on the web servers **36** shown in FIG. **3**.

As an example only, FIG. **2** shows a touch screen for the drinks page of a pizza restaurant. Again in the prior art, a specialized programmer had to design the layout and data for these POS touch keys. Typically, the programmer is located remotely from the store or business. He or she must learn about the store's POS requirements via phone calls, emails, and meetings with store operators. In addition, the programmer would need to iterate several passes of the touch screen design and allow the store operator to test the screens. With this invention, the store operator will be able to build his POS screens online over the Internet. With input from the store operator, the POS builder can specify and display the number, shape and arrangement of selection keys or buttons on said POS screens. The store operator, who does not have to be technically trained, will be able to edit and test his screens

until he is satisfied with the end results. The testing of said POS screens can be done iteratively by the store operator in real time while said POS terminals are simultaneously in use during store and business operation hours or after store hours. Alternatively, the testing of said POS screens can be done iteratively by a remotely located person such as a store manager or director in real time while said POS terminals are simultaneously in use during store hours or after store hours. All backoffice changes which include screen changes, price changes, employee validation changes are submitted to a batch bucket or queue. These changes have to be submitted for final posting at a scheduled time. For example, the phasing in of new screens and/or new data such as prices and employee validation can be scheduled. The time schedule for uploading or posting these screen changes and/or new data can be specified as follows. Only as examples, the changes can take place after the present transactions are completed. Alternatively, the changes can take place at the end of the business day, during the night, at the start of the next day or at the next application restart for example. Typically, screen changes will take place at the next application start at the beginning of a business day.

This automatic online POS builder will reduce the development time for POS screens by weeks. In addition, the store operator will be able to edit the POS screens and its relational databases any time as often as desired. In addition, the store operator will be able to edit, change and test the screens within minutes in real time. The store operator can iterate these changes instantly until he gets the desired screen appearance. This real-time testing and iteration of screen designs is an important feature of this invention. This feature motivates the store operator to keep his screens up to date and accurate. Previously, the store operator would avoid updating screens, since it involved the time and expense of working with programmers off line.

FIG. **3** shows a high level diagram of this invention. There are N POS terminals (POS **1**, POS **2** . . . POS N) in "Store" **31** and in "Store" **32**. POS **31** is in Store **1** and POS **2** (**32**) is in Store **2**. Each POS includes personal computer hardware and software. Additional POS terminals beyond those shown, as well as additional stores beyond the two shown, are within the scope of the invention. Each POS normally operates with a hardware/software connection **35** to the Internet or Web. However, if the web goes down, the POS terminal continues to operate. There is a "loose coupling" of the POS to the back office (BO): the POS to BO connection is not required for the basic business functions of the POS. All transaction data is stored in a relational database on the hard drive in the POS.

A relational database stores all of its data inside tables. All operations on data are done on the tables themselves. Some operation produce other tables as the result. A table is a set of rows and columns. Each row is a set of columns with only one value for each. All rows from the same table have the same set of columns, although some columns may have NULL values. A NULL value is an "unknown" value. The rows from a relational table are analogous to a record, and the columns are analogous to a field. Below is an example of a relational table.

NAME	COMPANY	E_MAIL
Jane A. Doe	ABC	jad@abc.com
Bill X. Smith	XYX	bxs@xyz.com

There are two basic operations one can perform on a relational table. The first one is retrieving a subset of its columns. The second is retrieving a subset of its rows. The field names such as company describe the content of the columns of the relational table. The rows delineate the individual records stored in the relational tables.

As transactions are created at a POS a log entry for the newest transaction is also created, this log entry is used to flag if the transaction has been uploaded to the web server. Part of the POS application, the BO interface is continuously running in the background. This component reads the log of transactions. If a transaction needs to be sent, it tries to send it. If the send fails (for example, if the connection to, or the Internet itself, is down), it goes to sleep and tries again later. Additionally, the BO interface requests update from the BO such as new items, price changes, employees, etc. The POS terminals communicate via HTTP protocol (hypertext transfer protocol) **35** with Back-office BO software, which is implemented on web servers **36**, which can be located anywhere in the world. In addition, the BO software and data can be viewed from any store employee at any PC **33** who has Internet access **37** and a password.

The POS such as **31** send transaction data to the BO in the form of an HTTP post or communication. The packet **35** sent from the POS to the BO consists of transactions, employee clock, customer add/update, item add/update, promotions and more. Promotions are configured in the back office and associated with items or customers or departments. For example, a promotion may be associated with a customer to implement customer loyalty points or a promotion may be associated with a certain item for a % discount. A client who is the store manager or owner selects a promotion type, associates it with an item, department, etc, then sets the parameters that control how that promotion works. These transaction transmissions between the POS and the BO can be Encrypted to insure privacy and security. A typical encryption method is 128 bit SSL (secure sockets layer). A further element of security is that each BO client (individual POS, store or multi-store owner) gets their own instance of a database. When they log into the BO they are attached to their own relational database associated and validated via their user login and password.

FIGS. **4***a* and **4***b* show a typical web-based POS builder interface. FIG. **4***a* shows a grid of boxes labeled with screen numbers **1**-**4**. Typically, screens will have screen names such as in **21**, "Subs". Under each screen box column are boxes labeled "Add Item". These boxes allow the addition of different products such as small pizza, large pizza, etc. as shown in FIG. **4***b*. FIG. **4***b* shows the data interface which would appear when selecting the large pizza box. The store operator would be able to enter and/or modify item name, price, cost, group, taxable and inventory. The above illustrates the ease of building POS screens by store operators via the Web.

FIG. **5** shows a flowchart of the point of sale builder methodology. The flow in FIG. **5** also refers to FIGS. **4***a* and **4***b*. The Begin POS Build block **51** is entered when the Builder Program is initiated **50** from a Web page action.

When creating a new POS, Block **51** brings up a screen such as that shown in FIG. **4***a*. The screens in FIG. **4***a* need to be defined. Block **53** allows the store operator to select which screen number to define. FIG. **4***b* shows what appears on the Web screen after the store operator selects screen #**1** (**53**) to work on. In FIG. **5**, Block **54** allows the store operator to enter/edit the screen name being worked on, such as pizza, as an example only, in FIG. **4***b*. In FIG. **5**, block **54** allows the store operator to enter the number of touch keys planned for the pizza screen, as an example only.

FIG. **4***b* shows the screen after a few touch screen buttons have been defined. Screen **1** has been labeled Pizza. The pizza screen in FIG. **4***b* currently has 1 touch screen button item defined on the screen, Large pizza **22**. The Large Pizza item button was entered by hitting ADD Item **20** in FIG. **4***a*. After hitting add item, FIG. **4***b* appears with the template **23** to be filled in. This step is shown in block **56** of FIG. **5**. The template includes Item Name, Price, Cost, Group, Taxable, Inventory. Item Name is Large Pizza. Price is easily changeable, Cost is the cost of making materials. Group is the Pizza Group, Taxable is as yes or no selection. Inventory can be used to monitor the number of Large Pizza's makeable with the dough, cheese and sauce on hand. Other Template items can be added to the template **23** in FIG. **4***b*.

In FIG. **5**, block **57** asks whether the screen being worked on i.e.) Pizza Screen is done. If the store operator answers yes **59**, the flowchart flows to Node **52** in FIG. **5**. This allows the store operator to select another screen # as shown in FIG. **4***a*. If the store operator answers no **58**, the flowchart flows to Node **55** in FIG. **5**. This allows the store operator to select, add, or edit another item on the pizza screen.

The key advantages of the Web-based POS builder are that it is completely built on the foundation of the Web. The POS builder is accessible anywhere in the world. It can be used by a person of any skill level. The POS builder builds, edits, and tests new POS terminals in real time. In addition, all screen designs and changes are reflected real-time into the back office (BO) server's screen database. For example, all screen designs inputted from any PC in the world appear instantly in the BO screen database, which is instantly viewable anywhere in the world via web browsers. Another big advantage is that all screen design software is located and executed in the BO server. Since all screen designs and changes are immediately visible from any manager's PC at their home or at headquarters, there is always management oversight of these changes. Therefore, this screen builder allows for local in-store flexibility by the individual store operator or manager, but also provides for corporate visibility of screens instantly for control and standardization. Also, this screen builder does not require the need for any server to be located in the store. Another advantage of this system is the use of standard PC and web architecture which offers both full-scalability without degrading system performance. This results in improved performance and lower cost of implementing these business systems. There is a lower cost associated with projects developed with the technology of this invention due to the flexibility of easy design changes and well-understood software. There is less training required for programmers and system testers. Projects can draw on the huge talent pool in the open source development community. The invention allows configurable software modules for different types of businesses and sales promotions. The invention allows remote monitoring of screen designs from anywhere via the web. There is minimal time required for the implementation and installation of the POS builder system, since the POS builder setup is as basic as a home PC setup. Another advantage is that the POS builder system can be provided as a service or deployed within a corporation. For example, Software as a Service (SMS) is a software distribution model in which applications are hosted by a vendor or service provider and made available to customers over a network, typically the Internet. Another advantage of this invention is that the POS builder system is maintained in customer centric databases, making it impossible for customers to see other's data. Each POS builder system client gets their own instance of a database. When they log into the BO they are attached to their own relational database associated and validated via their user login and password.

While this invention has been particularly shown and described with Reference to the preferred embodiments thereof, it will be understood by those Skilled in the art that various changes in form and details may be made without Departing; from the spirit and scope of this invention.

What is claimed is:

1. A web-based point of sale (POS) builder system comprising:

one or more point of sale terminals, that display POS screens,

an internet connection from said one or more point of sale terminals to a web server,

one or more local or remote PC workstations, and

point of sale builder software that runs on said web server, wherein said local or remote workstations are utilized to build or edit said POS terminals in real time, from anywhere in the world and over the worldwide web,

wherein said web servers are provided as a vendor subscription service wherein web server software resides and is hosted on said vendor's remote servers and wherein subscriber company's POS terminals access and repeatedly interact with said web server software from said vendor's remote servers, in order to perform the subscriber's desired terminal function, over a network, wherein the network comprises the Internet.

2. The web-based point of sale (POS) builder system of claim **1** wherein said POS terminals are implemented with personal computer (PC) architecture.

3. The web-based point of sale (POS) builder system of claim **1** wherein use of said builder requires no special training.

4. The web-based point of sale (POS) builder system of claim **1** wherein the testing of said POS screens is done iteratively in real time while said POS terminals are simultaneously in use during store and business operation hours.

5. The web-based point of sale (POS) builder system of claim **1** wherein no specialized hardware or software is required to implement said POS builder, and wherein no on site, in-store server is required.

6. The web-based point of sale (POS) builder system of claim **1** wherein said POS builder is useable to deliver POS application solutions using software as a subscription (SAAS).

7. The web-based point of sale (POS) builder system of claim **1** wherein said web server is a standard internet web server implemented with standard web server hardware and software, using one or more relational databases, wherein software for said POS builder resides in and is executed from said web server.

8. The web-based point of sale (POS) builder system of claim **1** wherein said POS builder adds new POS screens.

9. The web-based point of sale (POS) builder system of claim **1** wherein said POS builder edits existing POS screens.

10. The web-based point of sale (POS) builder system of claim **1** wherein said POS builder specifies and displays the number, shape and arrangement of selection keys or buttons on said POS screens.

11. The web-based point of sale (POS) builder system of claim **1** wherein said POS builder adds items to be associated with said keys or buttons.

12. The web-based point of sale (POS) builder system of claim **11** wherein said POS builder assigns attributes such as item name, price, cost, group, taxable and inventory status to said items.

13. The web-based point of sale (POS) builder system of claim **12** wherein said attributes are stored, retrievable and changeable from a back office database stored on said web server.

14. The web-based point of sale (POS) builder system of claim **1**, wherein said POS screens are viewable in real-time at remote locations anywhere in the world via personal computers with browser based web access.

15. A method of providing a web-based point of sale (POS) builder system comprising the steps of:

providing one or more point of sale terminals, which display POS screens,

providing an internet connection from said one or more point of sale terminals to a web server,

providing one or more local or remote PC workstations, and

providing point of sale builder software which runs on said web server, wherein said local or remote workstations is utilized to build or edit said POS terminals in real time, from anywhere in the world and over the worldwide web,

wherein said web servers is provided as a vendor subscription service wherein web server software resides and is hosted on said vendor's remote servers and wherein subscriber company's POS terminals access and repeatedly interact with said web server software from said vendor's remote servers, in order to perform the subscriber's desired terminal function, over a network, wherein the network comprises the Internet.

16. The method of providing a web-based point of sale (POS) builder system of claim **15** wherein said POS terminals are implemented with personal computer (PC) architecture.

17. The method of providing a web-based point of sale (POS) builder system of claim **15** wherein use of said builder requires no special training.

18. The method of providing a web-based point of sale (POS) builder system of claim **15** wherein the testing of said POS screens is done iteratively in real time while said POS terminals are simultaneously in use during store and business operation hours.

19. The method of providing a web-based point of sale (POS) builder system of claim **15** wherein no specialized hardware or software is required to implement said POS builder, and wherein no on site, in-store server is required.

20. The method of providing a web-based point of sale (POS) builder system of claim **15** wherein said POS builder is useable to deliver POS application solutions using software as a subscription (SAAS).

21. The method of providing a web-based point of sale (POS) builder system of claim **15** wherein said web server is a standard internet web server, implemented with standard web server hardware and software, using one or more relational databases, wherein programming for said POS builder resides in and is executed from said web server.

22. The method of providing a web-based point of sale (POS) builder system of claim **15** wherein said POS builder adds new POS screens.

23. The method of providing a web-based point of sale (POS) builder system of claim **15** wherein said POS builder edits existing POS screens.

24. The method of providing a web-based point of sale (POS) builder system of claim **15** wherein said POS builder specifies and displays the number, shape and arrangement of selection keys or buttons on said POS screens.

25. The method of providing a web-based point of sale (POS) builder system of claim **15** wherein said POS builder adds items to be associated with said keys or buttons.

26. The method of providing a web-based point of sale (POS) builder system of claim **25** wherein said POS builder assign attributes such as item name, price, cost, group, taxable and inventory status to said items.

27. The method of providing a web-based point of sale (POS) builder system of claim **26** wherein said attributes are stored, retrievable and changeable from a back office database stored on said web server.

28. The method of providing web-based point of sale (POS) builder system of claim **15**, wherein said POS screens are viewable in real-time at remote locations anywhere in the world via personal computers with browser based web access.

* * * * *

EXHIBIT E

PATENT ASSIGNMENT COVER SHEET

Electronic Version v1.1
Stylesheet Version v1.2

EPAS ID: PAT4299518

SUBMISSION TYPE:	NEW ASSIGNMENT
NATURE OF CONVEYANCE:	ASSIGNMENT

CONVEYING PARTY DATA

Name	Execution Date
WAYNE BARATTA	01/31/2017
QUENTIN OLSON	02/01/2017

RECEIVING PARTY DATA

Name:	CLOUDOFCHANGE, LLC
Street Address:	7 MANSION STREET
Internal Address:	SUITE 215
City:	POUGHKEEPSIE
State/Country:	NEW YORK
Postal Code:	12601

PROPERTY NUMBERS Total: 1

Property Type	Number
Patent Number:	9400640

CORRESPONDENCE DATA

Fax Number: (845)471-2064

Correspondence will be sent to the e-mail address first; if that is unsuccessful, it will be sent using a fax number, if provided; if that is unsuccessful, it will be sent via US Mail.

Phone: 845-452-5863
Email: debbi@saileackerman.com
Correspondent Name: SAILE ACKERMAN LLC
Address Line 1: 28 DAVIS AVENUE
Address Line 4: POUGHKEEPSIE, NEW YORK 12603

ATTORNEY DOCKET NUMBER:	TY06-004
NAME OF SUBMITTER:	STEPHEN ACKERMAN
SIGNATURE:	/Stephen Ackerman/
DATE SIGNED:	03/02/2017

Total Attachments: 3
source=TY06-004_Assignment#page1.tif
source=TY06-004_Assignment#page2.tif
source=TY06-004_Assignment#page3.tif

TY06-004

For [x] U.S. and/or [x] Foreign Rights

For [ ] Application or [x] U.S. Patent

By [x] Inventor(s) or [ ] Present Owner

ASSIGNMENT OF INVENTION

In consideration of the payment by ASSIGNEE to ASSIGNOR of the sum of One Dollar ($1.00), the receipt of which is hereby acknowledged, and for other good and valuable consideration,

ASSIGNOR:

QUENTIN OLSON
Type or print name of ASSIGNOR

1434 Grant Street
Street Address

Port Townsend, WA 98368
City, State, Country, Zip

USA
Nationality

WAYNE BARATTA
Type or print name of ASSIGNOR

89 S. Randolph Avenue
Street Address

Poughkeepsie, NY 12601
City, State, Country, Zip

USA
Nationality

hereby sells, assigns and transfers to

ASSIGNEE:

Cloudofchange, LLC
7 Mansion Street
Suite 215
Poughkeepsie, NY 12601
USA
Nationality

and the successors, assigns and legal representatives of the ASSIGNEE

(complete one of the following)

[x] the entire right, title and interest

[ ] an undivided _____ percent (___%) interest for the United States and its territorial possessions

(check the following box/f foreign rights are also to be assigned)

[x] and in all foreign countries

in and to any and all improvements which are disclosed in the invention entitled:

Web-Based Point of Sale Builder

and which is found in

☐ (a) U.S. patent application executed on even date herewith

☐ (b) U.S. patent application executed on

☐ (c) U.S. application serial no. _____ filed on _____

☒ (d) U.S. patent no. 9,400,640 issued July 26, 2016
(also check (e) if foreign application(s) is also being assigned)

☒ (e) and any legal equivalent thereof in a foreign country, including the right to claim priority

and, in and to, all Letters Patent to be obtained for said invention by the above application or any continuation, division, renewal, or substitute, thereof, and as to letters patent any re-issue or re-examination thereof.

ASSIGNOR hereby covenants that no assignment, sale, agreement or encumbrance has been or will be made or entered into which would conflict with this assignment.

ASSIGNOR further covenants that ASSIGNEE will, upon its request, be provided promptly with all pertinent facts and documents relating to said invention and said Letters Patent and legal equivalents as may be known and accessible to ASSIGNOR and will testify as to the same in any interference, litigation or proceeding related thereto and will promptly execute and deliver to ASSIGNEE or its legal representatives any and all papers, instruments or affidavits required to apply for, obtain, maintain, issue and enforce said application, said invention and said Letters Patent and said equivalents thereof which may be necessary or desirable to carry out the purposes thereof.

IN WITNESS WHEREOF, I / We have signed

WARNING Date of signing must be the same as the date of execution of the application if Item (a) was checked above.

_____	Feb 1, 2017
Signature of ASSIGNOR	Date of Signing
_____	_____
Signature of ASSIGNOR	Date of Signing

and which is found in

☐ (a) U.S. patent application executed on even date herewith

☐ (b) U.S. patent application executed on

☐ (c) U.S. application serial no. _____ filed on _____

☒ (d) U.S. patent no. 9,400,640 issued July 26, 2016
(also check (e) if foreign application(s) is also being assigned)

☒ (e) and any legal equivalent thereof in a foreign country, including the right to claim priority

and, in and to, all Letters Patent to be obtained for said invention by the above application or any continuation, division, renewal, or substitute, thereof, and as to letters patent any re-issue or re-examination thereof.

ASSIGNOR hereby covenants that no assignment, sale, agreement or encumbrance has been or will be made or entered into which would conflict with this assignment.

ASSIGNOR further covenants that ASSIGNEE will, upon its request, be provided promptly with all pertinent facts and documents relating to said invention and said Letters Patent and legal equivalents as may be known and accessible to ASSIGNOR and will testify as to the same in any interference, litigation or proceeding related thereto and will promptly execute and deliver to ASSIGNEE or its legal representatives any and all papers, instruments or affidavits required to apply for, obtain, maintain, issue and enforce said application, said invention and said Letters Patent and said equivalents thereof which may be necessary or desirable to carry out the purposes thereof.

IN WITNESS WHEREOF, I / We have signed

WARNING Date of signing must be the same as the date of execution of the application if item (a) was checked above.

Signature of ASSIGNOR	1/31/17 Date of Signing
Signature of ASSIGNOR	Date of Signing

EXHIBIT F

SAILE ACKERMAN LLC
28 DAVIS AVENUE
POUGHKEEPSIE, NY 12603

March 6, 2007

To: TYRAMM INTERNATIONAL
33 Sharon Drive
Fishkill, NY 12524

Attn: Todd E. Steinard
Wayne Baratta

From: Stephen B. Ackerman

INVOICE

FOR ATTORNEY WORK DONE IN U.S.

1) For Preparation and Filing of Patents

Application TY06-001 QUENTIN OLSON 30 CLAIMS			
Attorney	PREP. INVENTION DISCLOSURE & APPLICATION	$	5,000.00
Draftsman 3 sheets		$	300.00
Filing Fees (no assignment)		$	1,050.00
Copy costs		$	12.00
Mailing to PTO - EXPRESS		$	35.00
Application TY06-002 QUENTIN OLSON 30 CLAIMS			
Attorney	PREP. INVENTION DISCLOSURE & APPLICATION	$	5,000.00
Draftsman 2 sheets		$	200.00
Filing Fees (no assignment)		$	1,050.00
Copy costs		$	12.00
Mailing to PTO - EXPRESS		$	35.00

2) For Preparation of Responses to Patent Office Actions or Requirements

TY06-001	OLSON	NOVELTY SEARCH/REPORT	$	525.00
TY06-002	OLSON	NOVELTY SEARCH/REPORT	$	525.00
US PATENTS		5 PATENTS @ $8	$	40.00

3) Miscellaneous

TOTAL $ 13,784.00

You may altneratively send payment to me directly or electronically by sending payment to my Bank Account's Transit Routing Number and Account Number which is 021000322 004831928959.
My wire transfer number is 026009593 and Account Number is 004831928959.
My Bank address is Bank of America (New York), Poughkeepsie, NY 12603

With regards,

Stephen B. Ackerman