WEB-BASED MEDICAL INFORMATION SYSTEMS

Suresh Chalasani, University of Wisconsin-Parkside, chalasan@uwp.edu

ABSTRACT

Corporations and academia are currently exploring medical Information Systems. Medical information systems that contain information on the patients' health records, appointments, laboratory tests/results, billing and insurance in one comprehensive web site are almost non-existent. Some of the reasons for this include privacy concerns and reluctance to invest in new technology by health-care providers. In this paper, we provide architectures for delivering medical information on the Internet. It also presents a security architecture for confidential and secure access to medical data on the internet.

Keywords: Software architectures, Web-based systems, Medical information systems, Health care systems

INTRODUCTION

Medical practices today are faced with greater business problems than ever before. Reimbursements are reducing for medical practices (14) and stringent government regulations are forcing health care professionals to spend a lot of time on reporting. At the same time, managed care is adding another dimension of complexity to the health-care professional using capitation to limit reimbursements. In addition, the threat of malpractice lawsuits are forcing physicians to keep minute details of their interactions with patients (15). Because of these factors, direct costs of providing quality care to patients is increasing considerably, and hence health care providers are increasingly seeking efficiency in practicing medicine.

A demonstrated way to increase efficiency in the medical practice is to use technology and automate as many processes as possible without compromising privacy and security. Inefficient medical information systems often result in manual operations, which in turn require more employees, and reduce the profitability of medical practices. If processes are not made more efficient via automation, new growth opportunities are often bypassed, simply because the medical staff is overwhelmed just keeping up the daily manual processes. Instead of guiding their practices in the most profitable directions, physicians and their staffs often find themselves simply reacting to the strongest business pressures and struggling to deal with an ever-increasing amount of paperwork. In this paper, we propose a model for medical information systems on the Internet. An advantage of Internet medical information systems include the following:

- Easy accessibility for patients to their health records.
- Easy and direct accessibility of patient records among a group of physicians, all of whom are involved in providing care to the patient.
- Integration of patient appointments, schedules, billing, history & physical, and medication information in one comprehensive system.
- Easy accessibility of relevant information to the regulatory agencies such as the Medicare and Medicaid.

The real challenge facing the medical field in their quest to become web-enabled is not building separate information systems for patient schedules, appointments and billing and maintaining their records, but creating an efficient, integrated and comprehensive system which maintains all the different pieces of information in one system. Separate systems are available for maintaining patient billing information (1), patient medical records (2, 3). Most of these

systems are client-server systems and do not support accessibility for patients, physicians (or their staff) and the regulatory agencies via the Internet. In this paper, we provide architectural designs for medical information systems on the Internet.

ARCHITECTURES FOR MEDICAL INFORMATION SYSTEMS ON THE WEB

It is widely accepted that the thin-client model is the de facto standard when it comes to developing web applications. Typically, a thin-client application is most readily associated with a browser-hosted user interface (UI), which is dynamically generated and sent to the client, in the form of HTML, by the server. It is considered thin because clients of web applications are expected to have a browser pre-installed on their machine so the application need only focus on feeding the browser UI instructions it can understand and use to build a presentation to the end user. With this configuration, web clients are essentially dummy terminals that send HTTP requests to the server, where all the business logic and data source integration occurs. A typical thin-client web model is shown below:

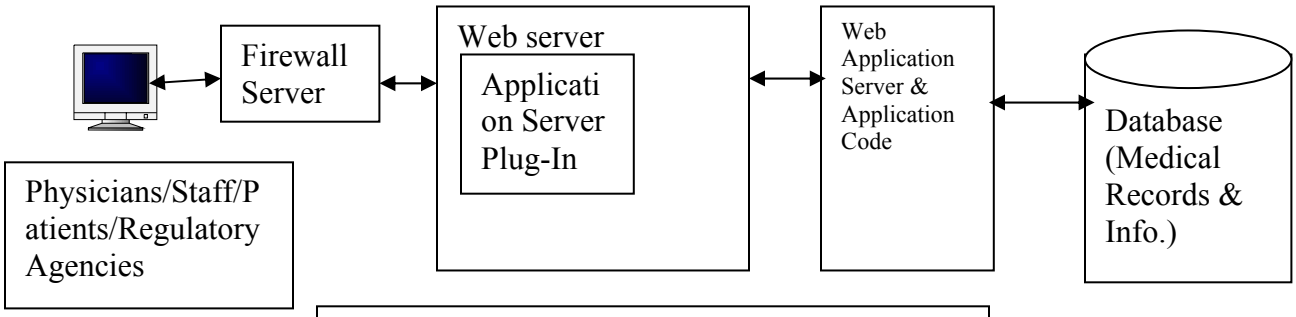

Figure 1: General thin-client architecture.

The system shown in Figure 1 can handle queries from physicians, medical staff, patients or staff of regulatory agencies such as Medicare/Medicaid. Requests from physicians, patients etc. first pass through a firewall server. Simple requests (such as requests for static HTML pages) are handled by the web server, while more complicated requests for the data are handled by the application server. Application server passes the complex requests to the application code. Application code may interact with the database to retrieve specific data requested by the user and formats them appropriately to return to the user. For example, a request for specific history and physical record (H&P record) for a patient on a specific date is fulfilled by reading from the appropriate database tables and sending the information back to the user. Application code is typically written in Java. Typically Java server pages, which intermix Java code with HTML, can be used to design appropriate interfaces for the patients to retrieve their medical, billing and appointment records, for physicians to access patient health records, and for government agencies to run reports on Medicare and Medicaid patients.

DATABASE DESIGN FOR MEDICAL INFORMATION SYSTEMS

The key to designing databases for the medical information systems consists of separating the functionality between patient medical records, patient schedules and billing and patient insurance information, and security/authentication. Below, we provide a description of the tables for each of these three areas.

Database Tables for Patient Medical Records

A block diagram of the database tables for holding the data contained in patient medical records is shown in Figure 2. Patient table holds information on the patient records. Similarly, physician

table holds information on the physicians and insurance table holds information on the insurance companies that insure the patients. For simplicity, only a couple of attributes of each of these tables are shown in the Figure 2. The table for phone numbers takes an id and a phone number (both of these could form the primary key or a unique primary key can be automatically generated by the system). This table is used to hold any number of phone/fax/pager numbers for the patients, physicians and/or insurance companies. Patient Insurance Xref is a cross-reference table that holds information on one or more insurances that the patient may hold. For each patient and insurance he/she carries, this table stores the primary care physician information and the patients insurance number with that insurance company. To store patient medical records, we extend the database by adding more tables as shown in Figure 3. This new set of tables contains

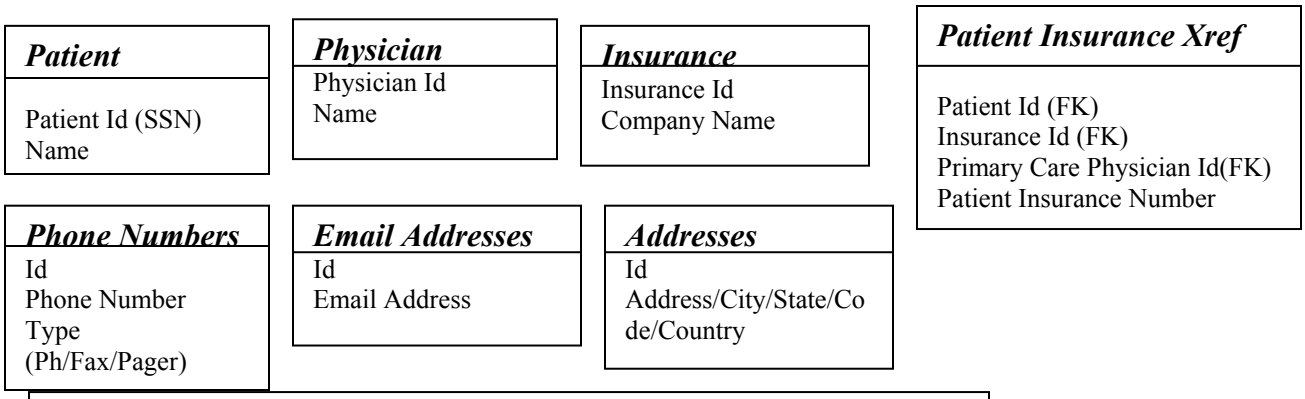

Figure 2: Basic database tables for the Medical Information System.

information on hospitals and clinics. These hold information on the hospitals and clinics that the patient may visit. Information on each patient visit is stored in the patient visit table, which uses the visit id as the primary key. In addition, this tables saves the date/time of the patient visit and the patient id as well as the hospital id the patient visited. During a visit, a patient can be seen by one or more physicians; patient-visit-physician-xref table saves information on who attended to the patient during his/her visit. For the physician to summarize his/her observations on the patient, we designed the observation table. Observation table consists of several entries; for example, temperature can be an entry with its corresponding normal high and normal low values. Patient-visit-physician-observation-xref table is used to store all the notes made by the physician on the patient on a given day of any visit. If a patient is hospitalized for more than one day, the date-time field in this table is useful for distinguishing between different observations on different dates by the physician. A similar table exists for saving the laboratory test results on patients.

Database Tables for Patient Appointments and Billing Information

Patient appointments can be stored in the patient appointment table as shown below in Figure 4. If a patient keeps his/her appointment, a corresponding entry will be made in the patient-visit and patient-visit-physician-xref tables. Patient billing information is saved in the patient-visit-bill table.

Database Tables for Authentication and Security

In our design, we assign security/authentication levels to each user of the medical information system. Recall that the users of the system can be patients, physicians, medical staff and government agencies. In Table 2, we outline the privileges for each user. The above privilege structure can be implemented completely in the application code. But implementing the privileges in the application may lead to frequent changes to the application code, if the rules on

who needs to access which information changes frequently. Hence, we use a database table for saving the privilege information for each user. This set of database tables is shown below.

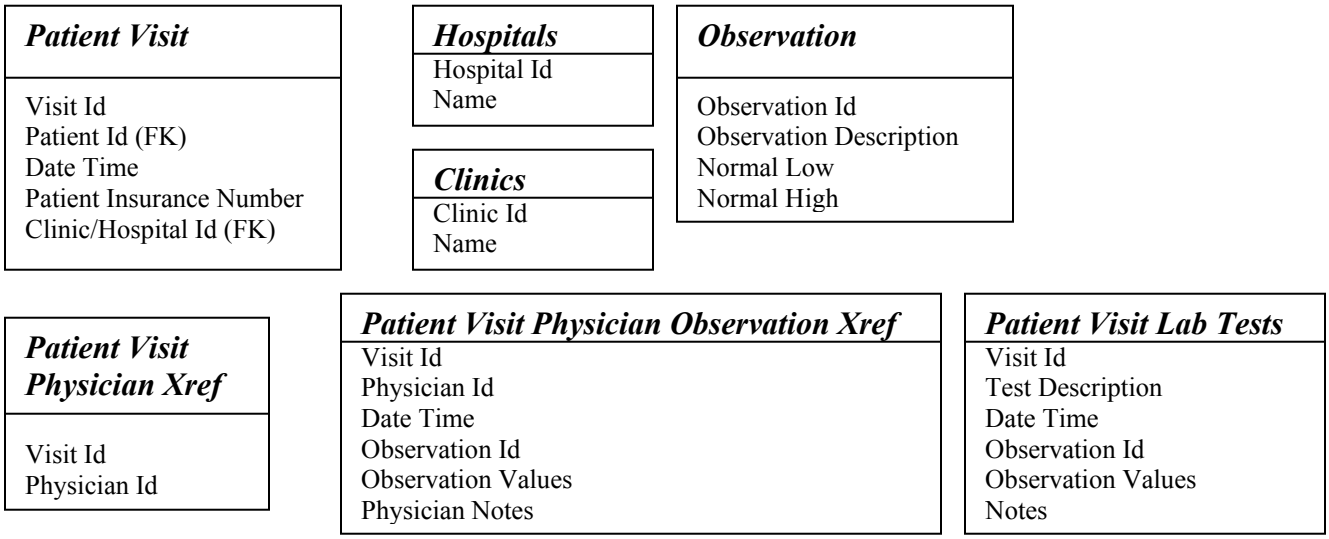

Figure 3: Database tables needed to hold information on patient's history and physical examination as well as physician's observations.

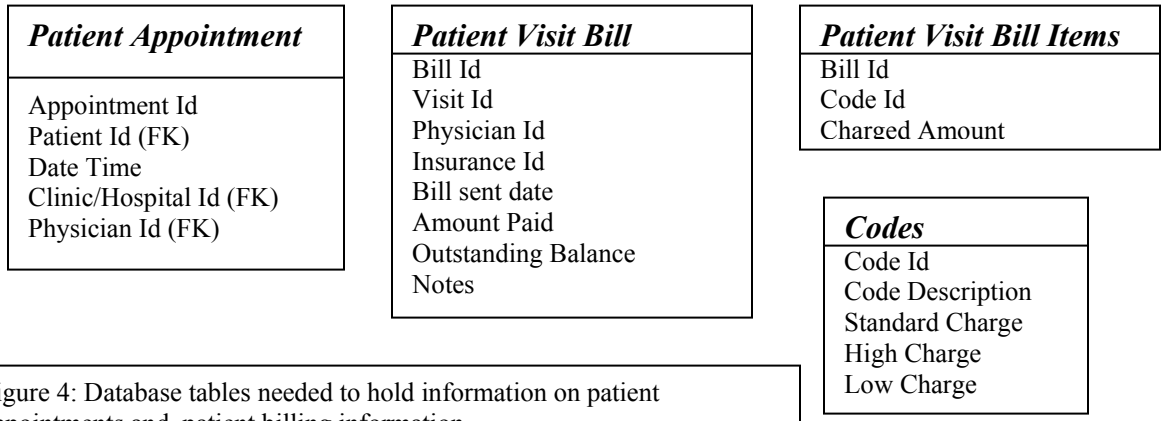

Figure 4: Database tables needed to hold information on patient appointments and patient billing information.

Application modules for the medical information system under discussion are shown in Figure 6. These application modules are being designed using the following principles:

- Entering information should be made as easy as possible. For example, whenever possible, allow the user to pick from a list of choices (codes when entering billing information).
- Allow users to view information in a logical, streamlined manner.
- Observations should be able to be recorded quickly and easily at the point of care. Internet medical information systems using thin-client architectures allow the users to accomplish this.
- Patient encounter documentation incorporating detailed observations should be available to the entire care team at all times.

Table 2: Privileges to view/modify medical records.

Person Type	View Only Privileges	Modify Privileges
Patient	Information on Appointments, Visits, Medical Records, and Bills. **Can only view his/her information.**	Patient's personal information such as address, phone numbers etc.
Physicians/Nurses/Physician Assistants	Patient appointments and visit information and medical records information for the patients he/she had seen or scheduled to see.	Observations on patient that he/she had provided care for.
Medical staff	Information on patient appointments, visits and bills only. **No access to medical records.**	Information on patient appointments, visits and bills only. **No access to medical records.**
Medical Technologists	Information on patient lab tests that they entered.	Information on patient lab tests.
Representatives of Government Agencies	Information on patients who use Medicare or Medicaid.	N/A

User Information
Id
Password

User Privileges
Id
Privilege

Figure 5: Database tables needed to hold information on privileges to access medical information.

APPLICATION ARCHITECTURE DESIGN

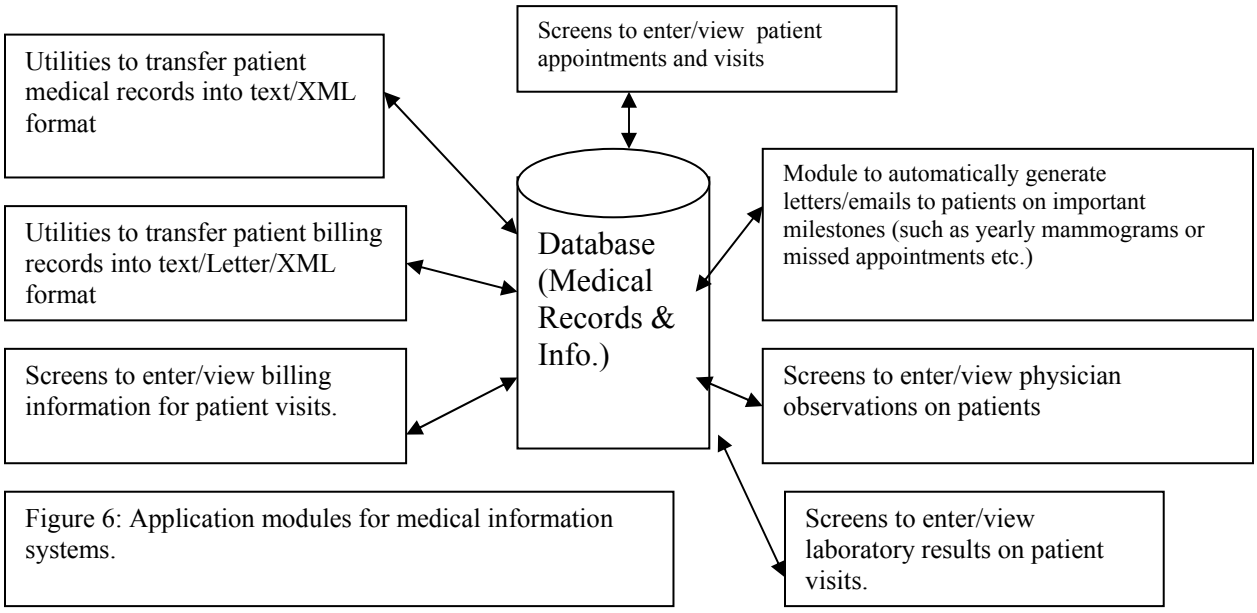

Figure 6: Application modules for medical information systems.

One key component of our design is that our system will be able to handle different data formats. For example, if patient medical records come as a text file or in XML format, our system will be able to import that data into the database. Similarly, our system will be able to export medical information into predefined XML format and text file formats. We are also exploring the possibility of exporting this information into Microsoft Excel and Word formats. We believe this flexibility is essential, especially since many physicians are moving towards the

use of hand-held computing devices such as PDAs, and we are likely to witness the need for accommodating a multitude of technologies in future.

SECURITY ARCHITECTURE FOR PASSING MEDICAL DATA AMONG PARTNERS

Medical industry consists of a conglomeration of corporate entities such as physician groups, HMOs, and insurance companies. A patients medical records at different times need to be seen by representatives of each one of these corporate entities. For example, a physician in one physician group may need to access records of a patient whose primary care physician belongs to another group. Similarly, an insurance company representative may need to look at a patient's medical record before accepting or denying a medical claim. As multiple corporate bodies access patient records, secure and confidential access to patients' medical records becomes a very valid concern. In fact, one of the reasons why web-based medical information systems have not been too popular in the United States is because of the concern over the security of data. In this section, we propose a security architecture for different partners (such as Physician Groups and HMOs) to share patient information in a secure manner.

Our proposal involves the use of security software/hardware products such as Netegrity's SiteMinder (6). The detailed interaction between a user at corporate entity A (referred as the user in the following discussion) and the system at corporate entity B (referred to as the system in the following discussion) while the user accesses medical information is depicted via a series of steps in Figure 8. User's request for information from systems at Corporate entity B is intercepted by security software agents such as SiteMinder. This software forces the user to login, if he/she is not already logged into the system at Corporate entity B. If the user logs in successfully, user's request is processed; otherwise an error page is displayed. Products such as SiteMinder intergrate well with Microsoft's Active Directory and also Lightweight Directory Access Protocol (LDAP).

CONCLUDING REMARKS

In this paper, we discussed architectures for delivering medical information on the Internet. Sub-second response times are crucial for the users to accept web-based medical information systems. For web-based medical information systems to be widely accepted, secure access to confidential medical data is a critical pre-requisite. This paper provides an insight into security architecture for web-based medical information systems.

REFERENCES

(1) Avanta Medical Information Systems, http://www.avanta.com/features/fandb/billing.html.
(2) Ergo EMRitus: Electronic Medical Record, http://www.ergopartners.com/index.html.
(3) DOCS Inc., Makers of SOAPWare (Electronic Medical Record System), http://www.docs.com/.
(4) Peter Burrows et al., "Where is the Upside: HP-Compaq will lead in PCs but not in costs or technology," *Business Week*, September 17, 2001, pp. 40-43.
(5) "Shaping your profile on the web", *Medical Economics*, May 7 2001.
(6) Netegrity's SiteMinder, http://www.netegrity.com/products/index.cfm?leveltwo=SiteMinder
(7) S. Allamaraju et al., Professional Java E-Commerce, WROX Press, February 2001.
(8) S. Allamaraju et al., Professional Java Server programming J2EE 1.3 Edition, WROX Press, September 2001.
(9) G. Reese, Database Programming with JDBC and Java, O'Reilly, 2000.
(10) R.V. Boppana and S. Chalasani, "Fault-Tolerant Communication with Partitioned Dimension-Order Routers," *IEEE Transactions on Parallel and Distributed Systems,* vol. 10, no. 10, pp. 1026-1039, October 1999.

(11) S. Chalasani and R.V. Boppana, "Communication in Multicomputers with Nonconvex Faults," IEEE Transactions on Computers, vol. 46, pp. 612-622, no. 5, May 1997.

(12) S. Chalasani and M. Wafa, "Reliable Technologies for B2B E-Commerce Based Supply Chain Management Systems," Proceedings of the 38th Annual MBAA Meeting, Chicago, IL, February 2002.

(13) S. Chalasani & R. Boppana, Software Architectures for E-Commerce Computing Systems with External Hosting, *Proceedings of the International Conference on Information Technology: Coding and Computing,* April 2002, Las Vegas, NV.

(14) W. J. Guglielmo, "Medicare Cuts: Will Patients Pay the Price?", *Medical Economics*, pp. 81-96, January 2003.

(15) D. Borfitz, "Malpractice: Is Going Bare the Only Option?", *Medical Economics*, pp. 81-96, March 2003.

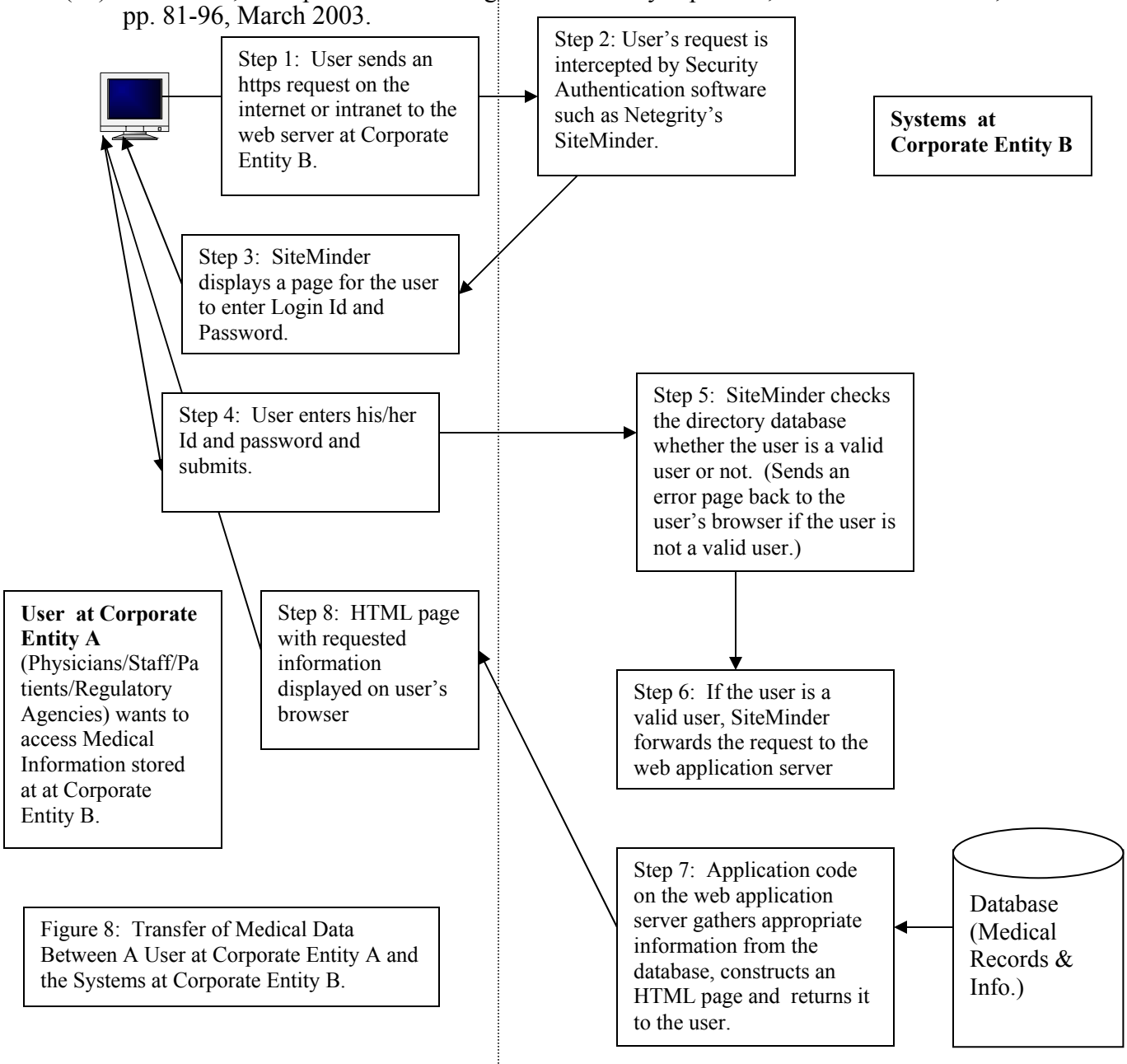

Figure 8: Transfer of Medical Data Between A User at Corporate Entity A and the Systems at Corporate Entity B.

Figure 8: Detailed Steps for Secure Data Access Between Two Corporate Partners