Available online at www.sciencedirect.com

ScienceDirect

Procedia Computer Science 62 (2015) 203 – 212

Procedia Computer Science

The 2015 International Conference on Soft Computing and Software Engineering (SCSE 2015)

A Secure Healthcare System: From Design to Implementation

Ebru Celikel Cankaya[1] Than Kywe[2]

[1, 2]: University of Texas at Dallas Department of Computer Science, Richardson, TX USA
{exc067000, txk102220}@utdallas.edu

Abstract

We introduce the design and development of a comprehensive electronic health record system (EHR) that incorporates AES encryption to assure security. Our work adopts a didactic approach to introduce the formal design steps of an EHR with its underlying database from a software engineering perspective. For this, we adopt two formal development methodologies as software engineering perspective and database development approach and combine the two to present a guideline to design and develop similar projects in other domains. For informative purposes, the steps of the development process are formalized based on database ER-model, and the final design is normalized into 3NF. We provide insight on rationale for employing specific methodologies, and using particular material and tools.

Peer-review under responsibility of organizing committee of The 2015 International Conference on Soft Computing and Software Engineering (SCSE 2015)

Keywords: *Database design, ER diagram, normalization, healthcare system.*

1. Introduction

With advances in healthcare informatics, one can provide better means to process patient records and therefore speed up the treatment, which in turn reduces the overall cost. Many tools exist for facilitating patient record processing: from assisting data entry to manipulating records, from generating output in required form to transferring it to other physicians for further examination, or to save it digitally for future use. The significance of our work relies on the fact that we bring a software engineering oriented systematic approach to design and develop an electronic health record system (EHR). Our scheme is fundamentally a computer based patient record (CPR) system [1, 2, 3]. In particular, this work extends a CPR system by incorporating a software engineering approach during development, and incorporates many aspects from database design, web deployment, and security.

While developing our design, we follow formal patient privacy requirements for healthcare systems that are enforced by Hipaa [4, 5].

Peer-review under responsibility of organizing committee of The 2015 International Conference on Soft Computing and Software Engineering (SCSE 2015)
doi:10.1016/j.procs.2015.08.441

A healthcare system is much complicated and comprehensive as it involves many aspects from hospital management to staff tracking. We focus particularly on patient record manipulation, as even this concentration is sufficiently complex to provide a comprehensive analysis of data, database design, and implementation steps.

2. Related Work

The need for a comprehensive electronic health record (EHR)system is pervasive, explaining the reason why there are many implementations as mentioned in [6, 7, 8, 9, 10]. Yet, each implementation focuses mainly on certain aspects while disregarding the others due to such factors as time, performance, number of users, user acceptance, and state or nationwide policies [3, 11, 12, 13, 14, 15, 16]. Actually, considering an EHR w.r.t. each factor listed above could be a topic of research itself, and is beyond the scope of this paper.

In [6], Lien et al. introduce an EHR system similar to our work. Additionally, our design integrates security to the EHR system by incorporating cryptology. When a user logs into the system, our system first checks the username in the backend database. If found, then corresponding encrypted password is retrieved from database and is decrypted. Next the user entered password is compared with the decrypted password. If they match, the password is valid and the system proceeds by checking the user permission level to grant appropriate access. Otherwise, the access is denied. [7] presents another EHR system designed to use for several types of databases with mobile applications via j2se program. However, this work also is missing a cryptography feature. The main advantage of our design is that we apply java cryptography architecture throughout so as to store all security sensitive data in the form of a cipher. When we retrieve this sensitive data from database, we decrypt it based on the user access level.

Other similar EHR system applications utilize different underlying infrastructure for the operating systems, as well as the database. For example, [17] designs an EHR for web based and mobile platforms, which runs on a Linux (Ubuntu 10.04) operating system with MySql database and ApacheTomcat. Our design runs on a newer version of Linux (Ubuntu 13.10) with Oracle 10g database.

Observing patient privacy is the fundamental requirement that an EHR system should possess. And many EHR system designs practice privacy preservation through HIPPA policy that remains only at the database level [18]. While manipulating sensitive patient records within database level as other similar systems do, our system extends secure data handling to user interaction level as well.

The concern to provide and maintain security of sensitive patient data is the main concentration of our system. For this, we adopt a multi-tiered architecture where the main ideology is to protect data anywhere at any time. This simply means that data must be protected whether it is stored in a database server or in the cloud; or is interacting with users in any means—from client computer, or from the web browser, or from mobile devices.

3. System Design And Development

The overall diagram of our design is illustrated in Figure 1 below.

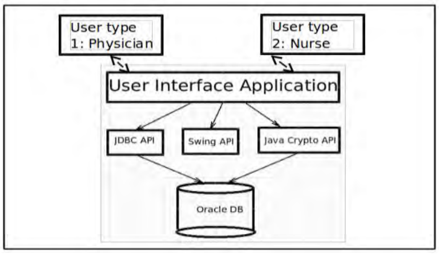


Figure 1.Overall system schema for the EHR design.

As can be seen from Figure 1, there are five essential components in our design: The User interface provides a GUI based interaction with the end user who is either a physician, or a nurse. The JDBC API offers database connection services, while Swing API supports the GUI services. The Java Crypto API is employed to complement the application with AES symmetric encryption. The last component is the Oracle database that lies at the bottom of the abstraction hierarchy to store patient data records, as well as encrypted content, such as user login data, disease categories, physician information, and patient records for visits and treatment.

The formal categorization and definition of the entities in our design are listed as follows:

- Patient: Subject of medical records (or agent).
- Patient health data: Data about patient's health or treatment enabling identification of patient.
- Healthcare provider: Mostly the physician, who has access to personal health information. The other type of a provider is nurse.
- Medical Office Staff: The staff of the healthcare center that help manage and maintain information regarding healthcare providers and patients.

For implementation, we use Netbeans IDE 8.0, JDK 1.7, and Oracle 10g. The reasons behind these choices are concerns on cost effectiveness, as well as performance. We see that these choices prove to be well made, as we get responses from our system with no major problems in a timely manner.

3.1. Requirements Gathering and analysis

We start designing our EHR system by gathering the facts and requirements about a generic patient entity first. From a software engineering perspective, a patient record should store information about patient’s identity differentiated by a unique patient id (for this, SSN is used), address, contact phone number, date of birth, gender, blood type, and a special field to add remarks about the patient.

As for the manipulation of a patient health data, three roles are defined: a physician, a nurse, and a staff member. A physician can add, modify, or delete his treatment records for a patient. He/she can view the complete treatment records of the patients assigned to him/her. The patient health data should also be manipulated by a nurse, though with lesser rights granted this time. Namely, a nurse should be able to add health related treatment record for a patient. A nurse should also be able to browse the visit history of a patient in a restricted way, which means that a nurse cannot view the detailed treatment records of physician(s) about a patient. The third role owner, i.e. a staff member, can view the patient’s basic information related to his identity, as well as his visit date and time, and the cost of visit. The staff should also be able to track patient wait time, and assign an appropriate and available healthcare provider to the patient.

In addition to the three roles defined above, the system also introduces an admin role, who is responsible for defining roles, assigning users to these roles and managing role granting and revocation on a dynamic basis (as physicians/nurses/staff can be hired, or may leave the job, etc.).

3.2. Design

After initial gathering of facts and requirements completed in Section 3.1., the collected data is further analyzed to formally define the database tables, fields of each table, primary and foreign key assignments, relationship and functional dependency definitions.

To exercise the design phase of the formal software engineering approach, we combine it with the formal database design step that involves the steps described in subsections 3.2.1 through 3.2.4:

3.2.1. ER Diagram

The Entity-relationship (ER) diagram first formally defines and constructs the entities and relationships in the system, then brings them together with relationships specified.

Figure 2 shows the overall ER diagram for our EHR system:

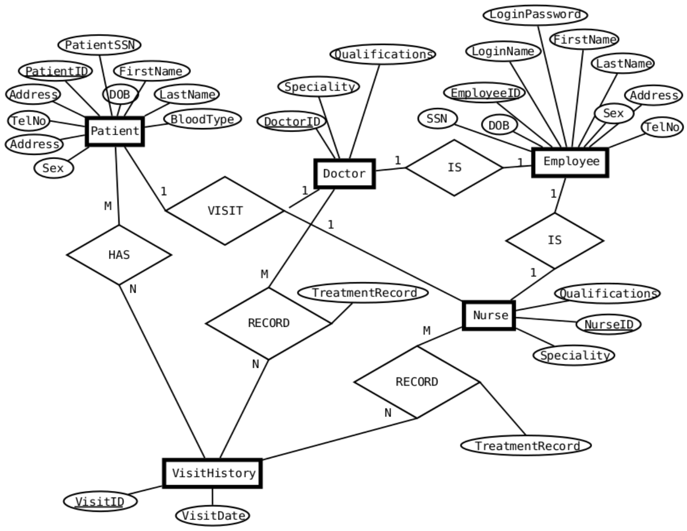


Figure 2.ER-Diagram for the EHR design.

As seen in Figure 2, the entities in our EHR design consists of fundamental components, rather than complex entities; and the relationships are simple accordingly. This is a deliberate design choice to make the process less complicated and easy to follow for educational purposes.

3.2.2. RELATIONAL DATABASE SCHEMA

The relational database schema given in Figure 3 displays a schematic view of the entities and the cardinalities associated with each entity.

As we see from Figure 3 the relational database schema illustrates the data type and domain information pertaining to each entity and relation, where the legend of symbols are listed as follows:

As we see from Figure 3 the relational database schema illustrates the data type and domain information pertaining to each entity and relation, where the legend of symbols are listed as ◆to indicate primary key, ●to indicate a non-nullable field, and ○ to indicate a nullable field.

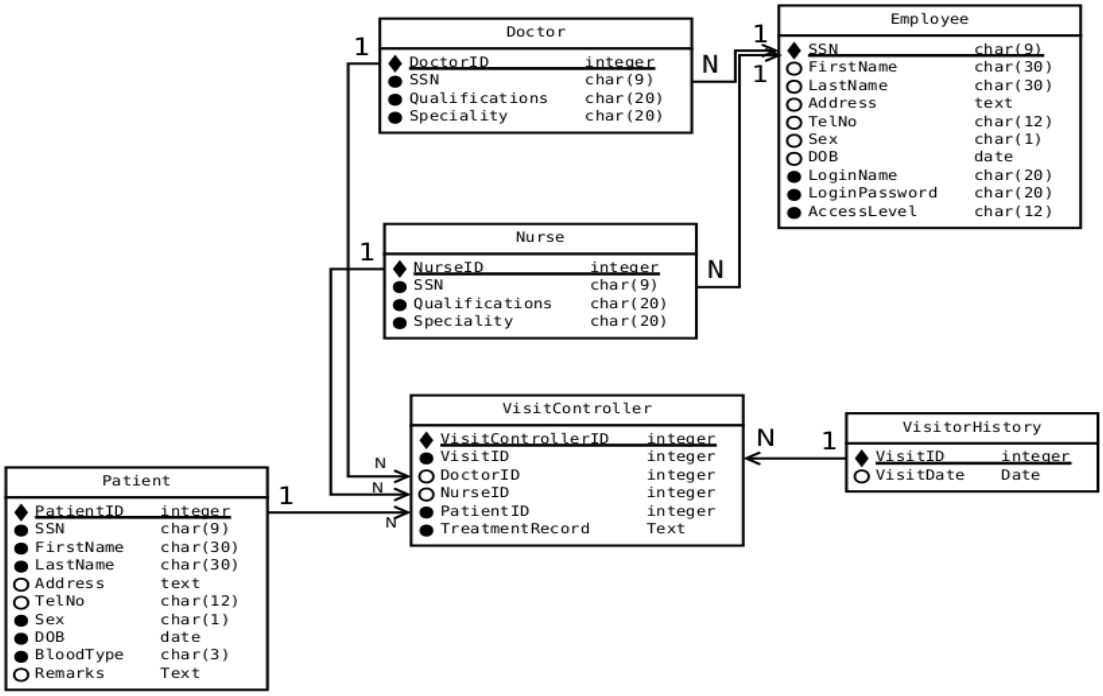


Figure 3.Relational database schema.

3.2.3. DATABASE CONSTRAINTS

Cardinality Constraints

Employee-Doctor 1:M

Employee-Nurse 1:M

1:M relationship between Employee and Doctor and Employee and Nurse as a there could be many doctors as employees, and one doctor can only register once as an employee. The same cardinality applies to the relationship between Employee and Nurse entities.

Patient-Doctor (M:N)

M:N relationship between Patient and visiting a doctor, as a patient can visit many doctors, and the same doctor can be visited by many patients.

Doctor manipulates visit history (M:N)

M:N relationship between Doctor and manipulating visit history, as a doctor can manipulate many visit histories, and the same visit history can be associated with more than one doctor.

Patient has visit history (1:M)

1:M relationship between Patient and having a visit history, as a patient have many visit histories, and the same visit history can only belong to one patient.

Key Constraints – Primary Key (PK)

SSN is the PK in Employee table, , as it uniquely identifies each employee.

DoctorID is the PK in Doctor table, as it uniquely identifies each doctor.

NurseID is the PK in Nurse table, as it uniquely identifies each nurse.

PatientID is the PK in Patient table, as it uniquely identifies each patient.

VisitID is the PK in VisitHistory table, as it uniquely identifies each visit history.

VisitControllerID is the PK in VisitController table, as it uniquely identifies each visit controller.

Referential integrity constraints - Foreign Key (FK)

DoctorID is FK in VisitController.

NurseID is FK in VisitController.

PatientID is FK in VisitController.

VisitID is FK in VisitController.

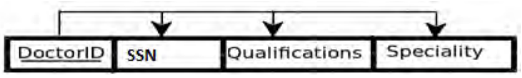
SSN is FK in Doctor.

SSN is FK in Nurse.

3.2.4. Dependency Diagrams

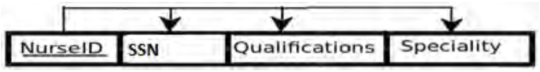
Doctor Table: (DoctorID, SSN, Qualifications, Speciality)

Corresponding Relational Schema for the Doctor table in 3NF



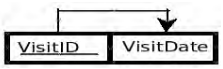
Nurse Table: :(NurseID, Qualifications, Speciality)

Corresponding Relational Schema for the Nurse table in 3NF



VisitHistory Table: :(VisitID, VisitDate)

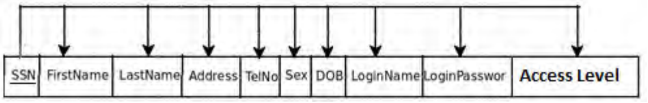
Corresponding Relational Schema for the VisitHistory table in 3NF



Employee Table:

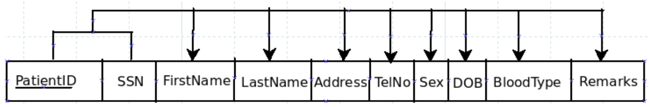
(SSN, FirstName, LastName, Address, TelNo, Sex, DOB, LoginName, LoginPassword, AccessLevel).

Corresponding Relational Schema for the Employee table in 3NF



Patient Table: (PatientID, SSN,FirstName,LastName, Address, TelNo, Sex, DOB, BloodType, Remarks).

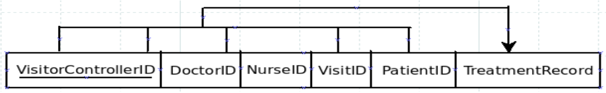
Corresponding Relational Schema for the Patient table in 3NF:



VisitorController Table:

(VisitorControllerID, DoctorID,NurseID, VisitID, PatientID, TreatmentRecord).

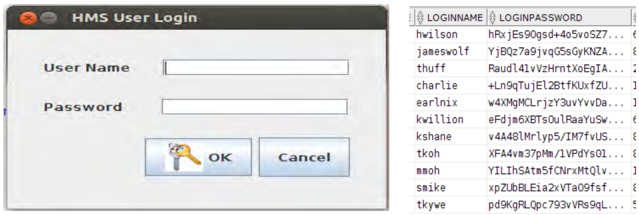
Corresponding Relational Schema for the VisitorController table in 3NF:



3.3. Implementation

The implementation section starts with designing GUI for the end users.

The initial login screen seen in Figure 4 below provides a means to collect user credentials for further verification by checking the already stored encrypted user passwords in the system



LOGINNAME	LOGINPASSWORD
hwilson	hRxjEs9Ogsd+4o5voSZ7...
jameswolf	YjBQz7a9jvqG5sGyKNZA...
thuff	Raudl41vVzHrntXoEgIA...
charlie	+Ln9qTujEl2BtfKUxfZU...
earlnix	w4XMgMCLrjzY3uvYvvDa...
kwillion	eFdjm6XBTsOulRaaYuSw...
kshane	v4A48lMrlyp5/IM7fvUS...
tkoh	XFA4vm37pMm/1VPdYs01...
mmoh	YILIhSAtm5fCNrxMtQlv...
smike	xpZUbBLEia2xVTa09fsf...
tkywe	pd9KgRLQpc793vVRs9qL...

Figure 4. Initial login screen and encrypted credentials.

After a successful login, the user is displayed the main screen of operations in Figure 5 that provides the menu of options he can do, based on his role in the system:

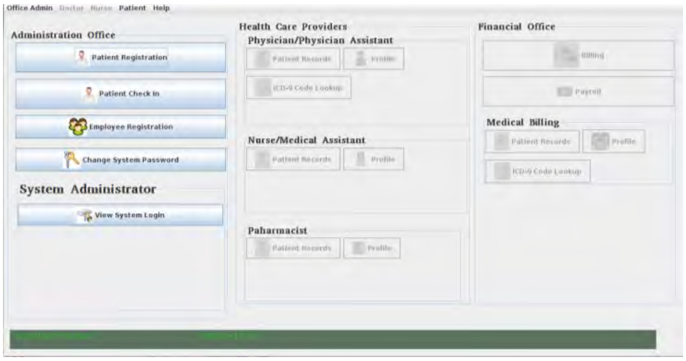


Figure 5.Admin screen of operations.

The screen displayed in Figure 5 captures the current user name and his role, and displays it together with the access time in status bar, for log purposes. Based on the role of the user, the functions that can be accessed are different: If the current user role is physician (Figure 6), then he can access the doctor and patient options from the main menu. If the current user is a nurse, he can access the nurse and patient options from the main menu. Similarly, if the current user is a staff, he can access the office admin menu. When the current user is an admin, he can view the log data regarding who accesses the system at what time, for what purposes.

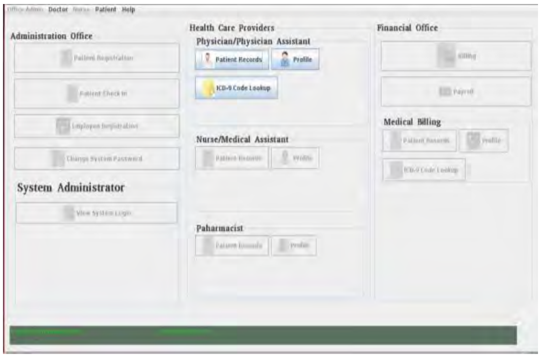
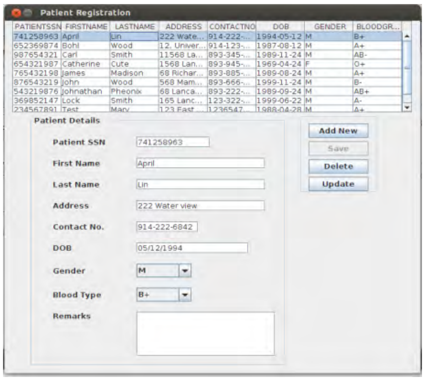


Figure 6.Physician screen of operations.

The process of patient registration is accomplished by a staff member via following screens in Figure 7:



PATIENTSSN	FIRSTNAME	LASTNAME	ADDRESS	CONTACTNO	DOB	GENDER	BLOODGR...
741258963	April	Lin	222 Wate...	914-222-...	1994-05-12	M	B+
652369874	Bohl	Wood	12. Univer...	914-123-...	1987-08-12	M	A+
987654321	Carl	Smith	11568 La...	893-345-...	1989-11-24	M	AB-
654321987	Catherine	Cute	1568 Lan...	893-945-...	1969-04-24	F	O+
765432198	James	Madison	68 Richar...	893-885-...	1989-08-24	M	A+
876543219	John	Wood	568 Mam...	893-666-...	1999-11-24	M	B-
543219876	Johnathan	Pheonix	68 Lanca...	893-222-...	1989-09-24	M	AB+
369852147	Lock	Smith	165 Lanc...	123-322-...	1999-06-22	M	A-
234567891	Test	Mary	123 East ...	1236547...	1988-04-28	M	A+

Patient Details

Patient SSN: 741258963

First Name: April

Last Name: Lin

Address: 222 Water view

Contact No.: 914-222-6842

DOB: 05/12/1994

Gender: M

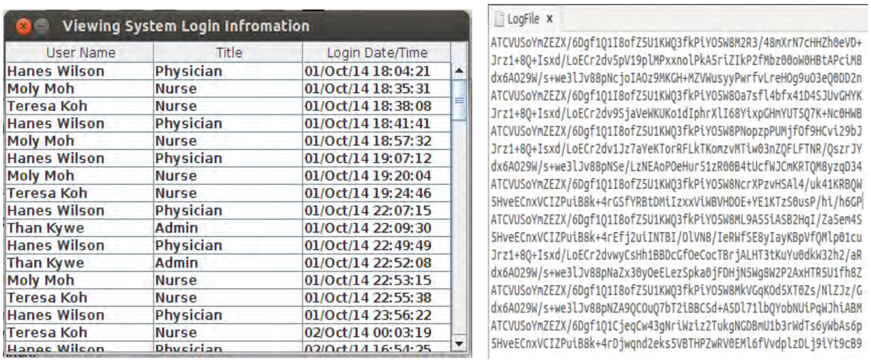
Blood Type: B+

Remarks:

Figure 7.Main screen of operations.

The staff member can add, modify, or delete a patient record in the system. Figure 7 shows a partial view of the patient registration records listed in ascending order of FirstNamecolumn.

The admin can view and check the system login information for error detection purposes. The login information (Figure 8) is directly associated with the encrypted log file displayed next to it. A symmetric encryption algorithm AES [19] along with 128 bits is used to encrypt the log file so as to provide the confidentiality of the log file from unintended third parties. The rationale behind using AES is twofold: Once it is fast (as compared to an asymmetric encryption algorithm), and also computationally hard to break due to its comparatively large key size (128 bits).



Viewing System Login Infromation

User Name	Title	Login Date/Time
Hanes Wilson	Physician	01/Oct/14 18:04:21
Moly Moh	Nurse	01/Oct/14 18:35:31
Teresa Koh	Nurse	01/Oct/14 18:38:08
Hanes Wilson	Physician	01/Oct/14 18:41:41
Moly Moh	Nurse	01/Oct/14 18:57:32
Hanes Wilson	Physician	01/Oct/14 19:07:12
Moly Moh	Nurse	01/Oct/14 19:20:04
Teresa Koh	Nurse	01/Oct/14 19:24:46
Hanes Wilson	Physician	01/Oct/14 22:07:15
Than Kywe	Admin	01/Oct/14 22:09:30
Hanes Wilson	Physician	01/Oct/14 22:49:49
Than Kywe	Admin	01/Oct/14 22:52:08
Moly Moh	Nurse	01/Oct/14 22:53:15
Teresa Koh	Nurse	01/Oct/14 22:55:38
Hanes Wilson	Physician	01/Oct/14 23:56:22
Teresa Koh	Nurse	02/Oct/14 00:03:19
Hanes Wilson	Physician	02/Oct/14 16:54:25

LogFile

```
ATCVUSoYmZEZX/6Dgf1Q1I8ofZ5U1KWQ3fkPiYO5W8M2R3/48mXrN7cHHZh0eVD+
Jrz1+8Q+Isxd/LoECr2dv5pV19plMPxxnolPkA5riZIkP2fMbz00oW0HBtAPciM8
dx6AO29W/s+we3lJv88pNcjoIAOz9MKGH+MZVWusyyPwrfvLreHOg9uO3eQ0DD2n
ATCVUSoYmZEZX/6Dgf1Q1I8ofZ5U1KWQ3fkPiYO5W8Oa7sfl4bfx41D4SJUvGHYK
Jrz1+8Q+Isxd/LoECr2dv9SjaVeWKUKo1dIphrXlI68YixpGHmYUT5Q7K+Nc0HWB
ATCVUSoYmZEZX/6Dgf1Q1I8ofZ5U1KWQ3fkPiYO5W8PNopzpPUMjfOf9HCvi29bJ
Jrz1+8Q+Isxd/LoECr2dv1Jz7aYeKTorRFLkTKomzvMTiw03nZQFLFTNR/QszrJY
dx6AO29W/s+we3lJv88pNSe/LzNEAoPOeHur51zR00B4tUcfWJCmKRTQM8yzqD34
ATCVUSoYmZEZX/6Dgf1Q1I8ofZ5U1KWQ3fkPiYO5W8NcrXPzvHSAl4/uk41KRBQW
5HveECnxVCIZPuiB8k+4rGSfYRBtDMiIzxxViWBVHDOE+YE1KTzS0usP/hi/h6GP
ATCVUSoYmZEZX/6Dgf1Q1I8ofZ5U1KWQ3fkPiYO5W8ML9A55iASB2HqI/Za5em4S
5HveECnxVCIZPuiB8k+4rEfj2uiINTBI/OlVN8/IeRWfSE8yIayKBpVfQMlp01cu
Jrz1+8Q+Isxd/LoECr2dvwyCsHh1BBDcGfOeCocTBrjALHT3tKuYu0dkW32h2/aR
dx6AO29W/s+we3lJv88pNaZx30yOeELezSpka0jFDHjN5Wg8W2P2AxHTR5U1fh8Z
ATCVUSoYmZEZX/6Dgf1Q1I8ofZ5U1KWQ3fkPiYO5W8MkVGqKOd5XT0Zs/NlZJz/G
dx6AO29W/s+we3lJv88pNZA9QCOuQ7bT2iBBCSd+A5Dl71lbQYobNUiPqWJhiABM
ATCVUSoYmZEZX/6Dgf1Q1CjeqCw43gNriWziz2TukgNGDBmU1b3rWdTs6yWbAs6p
5HveECnxVCIZPuiB8k+4rDjwqnd2eks5VBTHPZwRV0EMl6fVvdplzDLj9iYt9cB9
```

Figure 8.System log data and its encrypted content.

Addressing the patient together with patient health data entries, we create a patient visit records screen as seen in Figure 9:

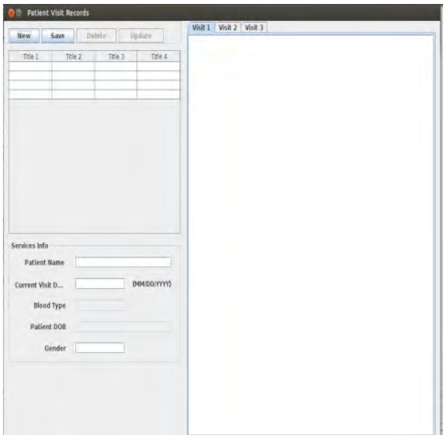


Figure 9.Patient visit records.

This screen allows a nurse to add treatment record(s) for a patient, as well as allows him to view the visit history of a patient in a limited manner (most probably, not to the degree that a physician is granted). The same screen allows a physician to add treatment record(s) for a patient, and also view the complete treatment records of the patients assigned to him. A staff member can also view patient basic information together with visit date and time for tracking purposes. The staff can also assign an available and a matching (meaning a specialist for a particular disease) health care provider to each patient.

4. CONCLUSION AND FUTURE WORK

We present the design and development of a novel EHR system incorporating two formal development methodologies as software engineering perspective, together with database development approach. This work provides a concrete example that combines these two formal methodologies to design and develop software, and can be used as an educational guideline to model after for designing similar projects in other domains. As part of future work, we plan on extending our design to provide an interactive tool that will help user get involved actively in the design process and self teachits steps.

REFERENCES

1. Rahimi B., Moberg A., Timpka T., Vimarlund V. *Implementing an integrated computerized patient record system: Towards an evidence-based information system implementation practice in healthcare*, AMIA 2008.
2. Fitzpatrick G., Ellingsen G. A Review of 25 Years of CSCW Research in Healthcare: Contributions, Challenges and Future Agendas, Computer Supported Cooperative Work (CSCW), 22(4-6), pp 609-665, August 2013.
3. Middleton B., Bloomrosen M., Dente M. A, Hashmat B., Koppel R., Overhage J. M., Payne T. H., Rosenbloom S. T., Weaver C., Zhang J., Enhancing patient safety and quality of care by improving the usability of electronic health record systems: recommendations from AMIA, *Journal of the American Medical Informatics Assoication*, Vol. 20, Issue e1, 2013.
4. http://www.hhs.gov/ocr/privacy/ [Accessed on: 09/05/2014].
5. Hu J., Chen H., Hou T. A hybrid public key infrastructure solution (HPKI) for HIPAA privacy/security regulations. *Computer Standards & Interfaces* 32.5 (2010): 274-280.
6. Lien C., Liu C., Chen Y. *Integrating Security Considerations in Client Server Architectures of Health Information Systems Development*. Innovative Mobile and Internet Services in Ubiquitous Computing (IMIS), 2011.
7. Harish, U., Ganesan, R., *Design and development of secured m-healthcare system*, Advances in Engineering, Science and Management (ICAESM), pp.470-473, 30-31 March 2012.
8. Greenhalgh T., Hinder S., Stramer K., Bratan T., Russell J. Adoption, non-adoption, and abandonment of a personal electronic health record: case study of HealthSpace, *British Medical Journal*, 2010; 341: c5814.
9. Wright A., Sittig D. F., Ash J. S., Feblowitz J., Meltzer S., McMullen C., Guappone K., Carpenter J., Richardson J., Simonaitis L., Evans R. S., Nichol W. P., Middleton B. Development and evaluation of a comprehensive clinical decision support taxonomy: comparison of front-end tools in commercial and internally developed electronic health record systems, *Journal of the American Medical Informatics Association*, March 2011.
10. Parsons A., McCullough C., Wang J., Shih S. Validity of electronic health record-derived quality measurement for performance monitoring, *Journal of the American Medical Informatics Association*, January 2012.
11. Menachemi N., Collum T. H. Benefits and drawbacks of electronic health record systems, *Journal of Risk Management and Healthcare Policy*, 4: 47-55, May 2011.
12. Charles D., Gabriel M., Furukawa M. F. Adoption of Electronic Health Record Systems among U.S. Non-federal Acute Care Hospitals: 2008-2013 , *ONC Data Brief*, No. 16, May 2014, USA.
13. Hsiao C., Hing E. Use and Characteristics of Electronic Health Record Systems Among Office-based Physician Practices: United States, 2001–2013, *NCHS Data Brief*, No. 143, January 2014.
14. Pendergrass J. C., Heart K., Ranganathan C.,Venkatakrishnan V. N. A. *Threat Table Based Approach to Telemedicine Security*, Transactions of the International Conference on Health Information Technology Advancement, Vol.2 No. 1, 2013.
15. Sunyaev A., Dmitry C. Supporting chronic disease care quality: design and implementation of a health service and its integration with electronic health records. *Journal of Data and Information Quality (JDIQ)* 3.2 (2012): 3.
16. Thiong'o K. K., *Framework for the Implementation of a Patient Electronic Referral System: Case Study of Nairobi Province,* University of Nairobi, Scientific Conference Proceedings. 2013.
17. Persson M., Tobian A., Johansson P., Goode E., Kruzela I., Johansson O. *A new improved distributed e-healthcare system based on open standards for depression treatment*. Proceedings of the 3rd International Conference on Information and Communication Systems, ACM, 2012.
18. Liu L. S., Shih P. C., Hayes G. R. *Barriers to the adoption and use of personal health record systems*, Proceedings of the 2011 iConference, ACM, 2011.
19. Daemen J., Rijmen V. The Design of RijndaeL: AES - The Advanced Encryption Standard (Information Security and Cryptography), Springer, 2002.

A Secure Healthcare System: From Design to Implementation

Citation:

Cankaya, Ebru Celikel, and Than Kywe. 2015. "A Secure Healthcare System: From Design to Implementation." Procedia Computer Science 62, doi:10.1016/j.procs.2015.08.441.