HEALTHCARE MANAGEMENT SYSTEM

DATABASE PROJECT

Of

CIS 9340
Database Management Systems

Fall 2000

By

JMTZ Bee Healthcare, Inc.

Business Computer Information Systems
Baruch College / CUNY

12/6/2000

Healthcare Management System

By

JMTZ Bee Healthcare, Inc.

INTRODUCTION

A healthcare center is looking to develop a state of patient portfolio management system which is able to track their patients' medical history. This system is to facilitate the center to retrieve, update, and report the patient information efficiently, in turn helping the doctors make timely, effective diagnoses. At the same time, the center can utilize this system to monitor their medical and financial management.

Currently, different departments in the healthcare center have their own separated systems leading to the lack of communications and the inefficient data sharing. For example, the finance department uses simple EXCEL spreadsheets to record the paycheck information of the employees which is inconvenient to retrieve and update employees' information; in the clinic department, the doctors have to write down the prescriptions for the patients and keep paper documents, and also do not have any information about the patients' insurance plans; the medicine department has to keep the prescription and inventory records on their own computer system. While each system serves a distinctive purpose, there is no coordinating, assimilating and representing of data. The systems may have duplicate data which is a waste of space. The different systems also may have different application programs which cause incompatible files.

Due to these disadvantages of the current system, a healthcare management system is proposed. Healthcare management system is a database management system (DBMS), which is based on computer networks, using the advanced database technology to construct, maintain, and manipulate various kinds of data in a database system (DBS). The DBMS can track and update all the information of recorded patients in the healthcare center during a particular time span. The major advantages of the DBMS are easy to retrieve and update information, efficient data sharing and communication, and reliable backup and security.

IMPLEMENTATION

E-R Diagram

After requirements gathering and system needs analysis for the healthcare center, an Entity Relationship (E-R) Diagram was designed, which is shown in Figure 1.
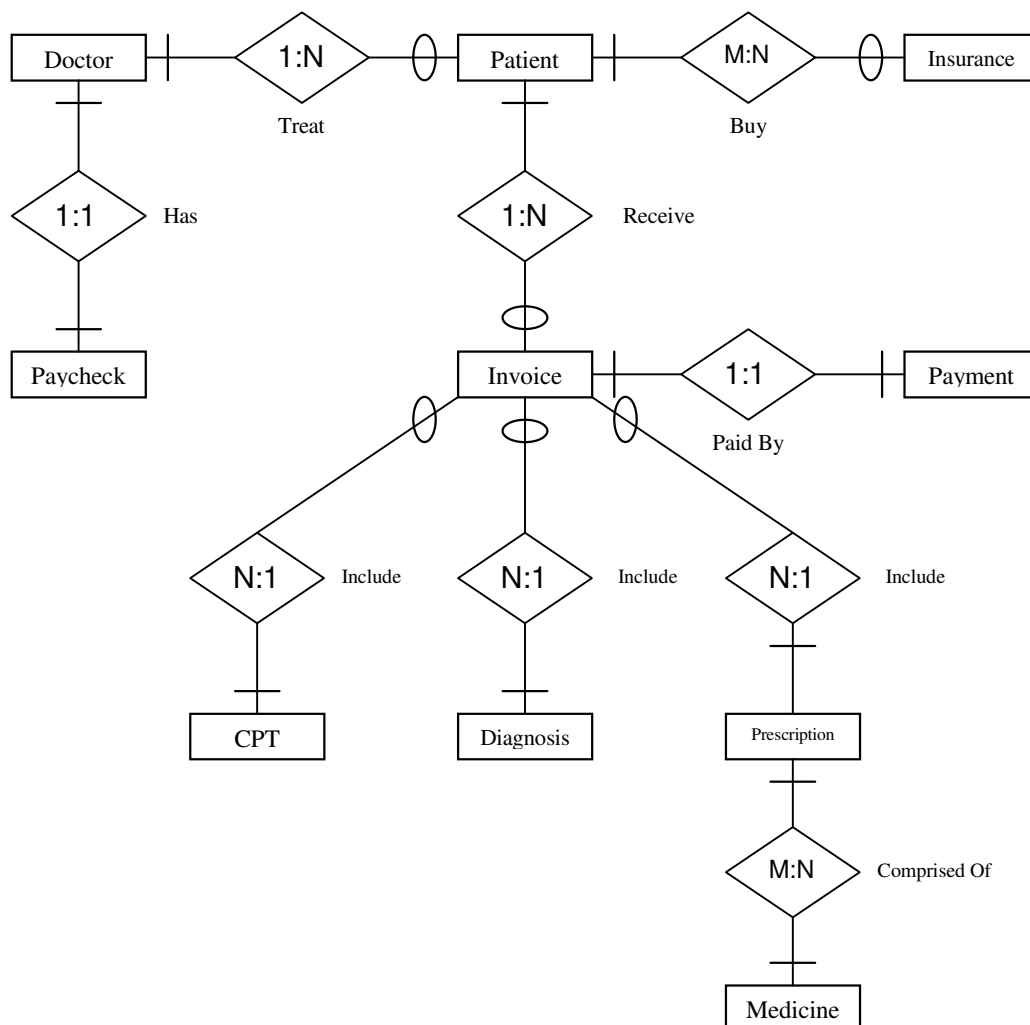

Figure 1. E-R Diagram

Ten (10) entities/tables (including the information needed) associated with nine (9) relations are proposed based on the E-R Diagram. The attributes of entities/tables are listed in Table 1.

Entity	*Attributes*
Doctor	D_Name, D_Ssn, Gender, Age, Position, Phone, Address, Office, City, Zip
Paycheck	Chk_Num, Salary, Bonus, Pay_Date
Patient	P_Name, P_Ssn, Gender, Age, Position, Phone, Address, Office, City, Zip
Insurance	InsCo_Id, InsCo_Name, Category, Phone, Address, City, Zip
Invoice	Invoice_Num, Amount, Invoice_Date, Due_Date
Payment	Invoice_Num, PayTrans_Num, Pay_Method, Pay_Status, Paid_Date
CPT	CPT_Id, Category
Diagnosis	Diagnosis_Id, Category
Prescription	Prescription_Id, Medicine_Quantity
Medicine	MInventory_Id, M_Name, Manufacturer, Price, Quantity, Exp_Date

Table 1. Attributes of Entities/Tables

Relational Model

Based on the functional dependencies of the relations listed in Table 2, we normalized all relations and converted the E-R model into a relational model which is illustrated as follows:

Doctor *(D_Ssn, D_Name, Gender, Age, Position, Office, Phone, Address, City, Zip)*

Paycheck *(D_Ssn, Chk_Num, Salary, Bonus, Pay_Date)*

Patient *(D_Ssn, P_Ssn, P_Name, Gender, Age, Position, Phone, Address, City, Zip)*

Insurance *(InsCo_Id, InsCo_Name, Category, Phone, Address, City, Zip)*

PatientInsurance *(P_Ssn, InsCo_Id)*

Invoice *(P_Ssn, Invoice_Num, CPT_Id, Diagnosis_Id, Prescription_Id, Amount,*

Invoice_Date, Due_Date)

Payment *(Invoice_Num, PayTrans_Num, Pay_Method, Pay_Status, Paid_Date)*

CPT *(CPT_Id, Category)*

Diagnosis *(Diagnosis_Id, Category)*

Prescription *(Prescription_Id, Medicine_Quantity)*

Medicine *(MInventory_Id, M_Name, Manufacturer, Price, Quantity, Exp_Date)*

PrescriptionMedicine *(Prescription_Id, Minventory_Id)*

Relation	*Functional Dependencies*
Doctor	D_Ssn → D_Name, Gender, Age, Position, Office, Phone, Address, City, Zip
Paycheck	Chk_Num → Salary, Bonus, Pay_Date D_Ssn, Chk_Num → Salary, Bonus, Pay_Date
Patient	P_Ssn → P_Name, Gender, Age, Position, Phone, Address, City, Zip D_Ssn, P_Ssn → P_Name, Gender, Age, Position, Phone, Address, City, Zip
Insurance	InsCo_Id → InsCo_Name, Category, Phone, Address, City, Zip
Invoice	Invoice_Num → Amount, Invoice_Date, Due_Date P_Ssn, Invoice_Num, CPT_Id, Diagnosis_Id, Prescription_Id → Amount, Invoice_Date, Due_Date
Payment	PayTrans_Num → Pay_Method, Pay_Status, Paid_Date Invoice_Num, PayTrans_Num → Pay_Method, Pay_Status, Paid_Date
Diagnosis	Diagnosis_Id → Category
CPT	CPT_Id → Category
Prescription	Prescription_Id → Medicine Quantity
Medicine	MInventory_Id → M_Name, Manufacturer, Price, Exp_Date Prescription_Id, MInventory_Id → M_Name, Manufacturer, Price, Exp_Date

Table 2. Functional Dependencies of Relations

Applications

In this project, Microsoft Access, Structured Query Language (SQL), and Microsoft Visual Basic (VB) were used to implement the applications. Tables, forms, queries, reports, and menus were created.

- **Example of Table** *(Medicine)*

MInventory_Id	M_Name	Manufacturer	Price	Quantity	Exp_Date
101	Biaxin	Pfizers	$10.99	1	10/5/02
102	Amoxil	Mercky	$6.99	1	1/11/02
103	Atenolol	Pfizers	$20.99	1	2/2/02
104	Nitroglycerin	Pfizers	$12.99	1	7/3/02
105	Coated ASA	Pfizers	$16.99	1	4/11/02
106	Metoprolol	Mercky	$11.99	1	2/12/02
107	Glipizide	Hulet	$25.90	1	4/5/02
108	Flonasl	Hulet	$7.50	1	5/9/03
109	Livostin	Mercky	$6.30	1	4/10/03
110	Cortisporin	Hulet	$5.70	1	3/3/03
111	Motrin	Mercky	$12.40	1	12/2/02
112	Prilosec	Hulet	$23.90	1	4/12/03

- **Example of Form** *(Patient)*

Function Description:
Patient Form is to retrieve and update patient information.

- **Example of Query** *(Doctor_Paycheck)*

D_Ssn	D_Name	Position	Chk_Num	Salary	Bonus	Pay_Date
999990001	Bill Adams	Pediatrist	1008	$8,000.00	$2,000.00	11/1/00
999990002	Henry Jackson	Physician	1002	$5,500.00	$1,650.00	11/1/00
999990003	Patricia Jones	Physician	1006	$9,000.00	$3,200.00	11/1/00
999990004	Paul Williams	Physician	1004	$6,000.00	$1,000.00	11/1/00
999990005	Sally Peg	Physician	1005	$7,000.00	$2,500.00	11/1/00
999990006	Virginia Clare	Ophthalmologist	1010	$6,000.00	$1,800.00	11/1/00
999990007	Tod Kenneth	Dentist	1009	$8,500.00	$4,000.00	11/1/00
999990008	Erica Anderson	Physician	1001	$5,500.00	$1,200.00	11/1/00
999990009	Jim Robinson	Surgeon	1003	$6,500.00	$1,500.00	11/1/00

SQL Statement:

> **SELECT** Doctor.D_Ssn, Doctor.D_Name, Doctor.Position,
> Paycheck.Chk_Num, Paycheck.Salary, Paycheck.Bonus,
> Paycheck.Pay_Date
> **FROM** Doctor, Paycheck
> **INNER JOIN** Doctor
> **ON** Paycheck.D_Ssn = Doctor.D_Ssn;

Function Description:

> The Doctor_Paycheck Query helps the payroll department to retrieve, update, and report the paycheck of each doctor.

- **Example of Report** *(Invoice_Payment)*

Function Description:

> Payment Report is to show invoice payment information in print view for financial need or monthly report.

- **Example of Information Retrieval** *(Patient)* **–** Achieved by Visual Basic

Function Description:
Patient_Retrieve Form is to search and retrieve patient information.

- **Example of System Menu –** Achieved by Switchboard in Microsoft Access

Function Description:
System Menu is to provide a user-friendly interface so that the management system could be easily accepted by users.

8

System Main Menu:

System Sub-Menu:

CONCLUSIONS

The Database Management System (DBMS) is a powerful tool in current and future business. This real world Healthcare DBMS involves huge amount of research and collection of information and its implementation is possible, profitable, and realistic. Through this system, the healthcare center can retrieve, update, and report information efficiently and greatly improve the data sharing and communication.

However, the future improvement is necessary and continuous modification is needed to approach the final success:

1) The security problem should be solved;
2) Visual Basic and/or other language should be more used to implement different functions;
3) A more user-friendly interface should be provided;
4) The system should be web-based.