CSC 261: Term Project

## Preamble

Last Modified: April 2nd, 2015

Team: TU06

Team Members

1. Yongke Yu {$IPI = $\_POST['PI'];yyu27}
2. Slyu Sun {ssun10}
3. Kiera Crist {kcrist}
4. Mohan Liu {mliu26}

Master Directory: /home/mliu26/public\_html/CS261/home

/home/mliu26/public\_html/CS261/kcrist

/home/mliu26/public\_html/CS261/mliu26

/home/mliu26/public\_html/CS261/yyu27

/home/mliu26/public\_html/CS261/ssun10

## Project Domain

The database will store information for a hospital management system. It will contain data specific to every staff member, unique positions, departments, medications, patients, visits, etc. The database can be used to find doctors, patients, review patient medical history, and review employment history. In contrast, the database will not store financial data or data that is not related to health care.

## Use Cases

* 1. A patient calls in to schedule an appointment. In the case he knows which doctor he wants, we can use the Doctor ID to schedule an appointment by adding to the Visits table. In addition we will also store the reason for the visit and the date of the visit. In the case the patient doesn’t know which doctor he wants, we can find one by querying for a doctor using the DepartmentID in the Staff object and a subquery in the Positions Table for a doctor. In the case this is a new patient, we can make a new tuple for him in the Patients table, if not we can reference the Patients table for his insurance information.
  2. A doctor might want to prescribe some medication for the patient. He can access the description of the medication from the medication table to see all medicines that would be appropriate. He can then access the Patient table to see if the patient is allergic and has coverage under his insurance company before selecting one that is appropriate. He can then finally get the properties of the medicine using the med\_id.

## Attributes (M) (Updated)

Author: Mohan Liu{mliu26}

Object: Staff

|  |  |  |  |
| --- | --- | --- | --- |
| ID | Type | Purpose | Example |
| Staff ID | INT | Unique identification number of each staff member | 123456789 |
| First Name | VARCHAR | An employee's first name. | John |
| Last Name | VARCHAR | An employee's last name | Smith |
| Gender | INT | An employee's gender. | 1 |
| SSN | BIGINT | An employee's social security number. | 123456789 |
| Home Phone | BIGINT | An employee's home phone number. | 6785674567 |
| Mobile Phone | BIGINT | An employee's mobile phone number. | 5858386543 |
| Email | VARCHAR | An employee's email address. | mith@hospital.com |
| DepartmentID | VARCHAR | An employee's department number | Cardiology |
| Position\_iD | VARCHAR | An employee's department number | 2 |
| Position Title | VARCHAR | Title given to a position | Surgeon |

Author: Yongke Yu, yyu27

Object: Patient

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| |  |  |  |  | | --- | --- | --- | --- | | Patient ID | INT | Identifying number for each Patient | 92323021 | | First\_Name | VARCHAR | First Name of Client | Jane | | Last\_Name | VARCHAR | Last Name of Client | Doe | | Gender | INT | Client’s Gender | 1 | | Home\_Phone | BIGINT | Home Phone of Client | 1234567777 | | Cell\_Phone | BIGINT | Cell Phone of Client | 1324354432 | | Emergency Contact | BIGINT | Emergency Contact Number | 9083219993 | | Birthday | date | Birthday of Patient | 01021970 | | Allergies | VARCHAR | Allergies of Patient | Penicillin | | Surgical\_History | VARCHAR | Surgical History of Patient | Tonsil Removal | | Staff\_Id | INT | The Last Doctor Seen | 13 | | Insurance | VARCHAR | Which Insurance Plan they are on | Aetna Student | | Visit\_Date | date | The date of last visit | 0125/2015 | | Med\_id | INT | medication number | 13 | | Reason | VARCHAR | Reasons for visiting a doctor | cold | | |
| Author: Siyu Sun(ssun10)  Objects: Diagnosis |  |
|  |  |

|  |  |  |  |
| --- | --- | --- | --- |
| ID | Type | Purpose | Example |
| Diag\_Id | int | identification of certain symptom | 1,2,3,4,etc |
| Diag\_Details | Varchar | Details about symptom | Fever |
| Severity | int | Severity of the symptom from 1 to 10 | 5(moderate) |
| Diag\_Date | DATE | Date for the diagnosis | 04/12/14 |
| Remark | varchar | Remark for special cases | genetic disease |
| Patient\_Id | int | Which Patient was seen using ID number | 12345678 |
| Patient\_FName | varchar | Patient’s first name by reference | John |
| Patient\_LName | varchar | Patient’s last name by reference | Smith |
| Staff\_Id | int | Which Doctor made the diagnosis using ID number | 123456 |
| Med\_id | int | medication for certain diagnosis, identifying by id | 34562 |
| Second\_Diag\_Date | DATE | Date for the second appointment for diagnosis | 05/02/15 |

Author: Kiera Crist (kcrist)

Object: Medications

|  |  |  |  |
| --- | --- | --- | --- |
| Med\_Id | int | unique identification | 13 |
| Name | VARCHAR | name of the medication | abc |
| Manf | VARCHAR | name of the manufacturer | abc |
| Contents | VARCHAR | the chemical name of the medicine | abc |
| Effects | VARCHAR | what the medication does to the body | abc |
| Form | VARCHAR | what form (pill, liquid ect) it’s taken in | abc |
| Dosage | VARCHAR | what dosages it comes in | abc |
| Side\_Effects | VARCHAR | what side effects the medication has | abc |
| Schedule | VARCHAR | when to take it | abc |
| Description | VARCHAR | a description of the medication | abc |
| Cost | INT | cost of medication | 30 |

## 

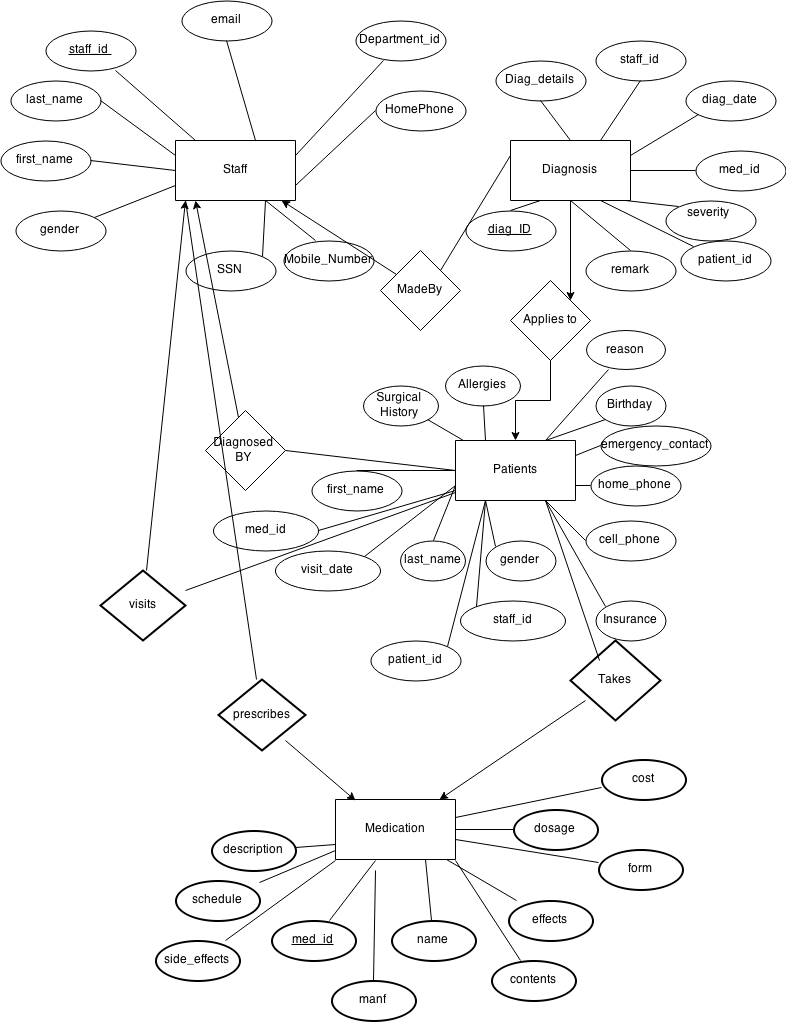
## 5.Functional Dependencies (M)

{staff\_id} → {First Name, Last Name, email, Home Phone, Mobile Phone}

{email} → {First Name, Last Name, SSN, Home Phone, Mobile Phone}

DepartmentID->Position Id

6. **ER Diagram (updated)**



**7. Relational Schemata (updated)**

Relations:

* Staff(First\_Name,Last\_Name,staff\_id,SSN,mobile\_number,homephone,department\_id,gender,email)
* Diagnosis(Diag\_Id,Staff\_Id,Severity,Remark,Details,Med\_Id)
* MadeBy(Staff\_Id,Diag\_Id)
* Patients(First\_Name, Last\_Name, Patient\_Id, Birthday, Emergency\_Contact,Last\_Visit\_Date, Last\_Visit\_Id, Last\_Doctor, Gender, Home\_Phone, Cell\_Phone, Insurance, Allergies, Med\_Id, Surgical\_History)
* Diagnosed\_By(Staff\_Id, Patient\_Id)
* Applies\_To(Patient\_Id, Diag\_Id)
* Takes(Patient\_Id, Med\_Id,Dosage,Schedule)
* Prescribes(Staff\_id, Med\_Id)
* Visits(Patient\_Id, Staff\_Id)

Functional Dependencies:

* in Staff: Staff\_Id->First\_Name, Last\_Name,Mobile\_Number,HomePhone,address,SSN,department\_id,gender,email)
* in Diagnosis: diag\_id->staff\_id,severity, remark,details,med\_id
* in Patient: patient\_id ->first\_name,last\_name,birthday,emergency\_contact,last\_visit date, last\_visit\_id, last\_doctor, gender, home\_phone, cell\_phone, insurance, allergies, med\_id, surgical\_history)
* in Medications med\_id ->name, manf, contents,effect, form,dosage, side\_effects, schedule, description, cost

**8. Proof of Normal Form (need to be updated )**

This is in 3NF, because each table is determined only by the candidate key(staff\_id,diag\_id,patient\_id). As you can see in each table, there is a candidate key that all attributes hold relations to (it uniquely identifies a tuple). In addition, it is already in 2NF since all non-prime attributes are dependent on a candidate key and each attribute contains a single value in the domain. So with the combination of candidate key determining all attributes, no redundancies non prime attributes and that it is in 2NF since there are no non prime attributes dependent on any subset of a candidate key, it is in 3NF.

9.

db-create-team-TU06.sql:

source /home/yyu27/db-create-yyu27.sql

source /home/ssun10/db-create-ssun10.sql

source /home/kcrist/db-create-kcrist.sql

source /home/mliu26/db-create-mliu26.sql

db-create-*TU06*-output.txt:

-bash-4.1$ vim db-create-TU06-out.txt

--------------

CREATE TABLE patients(

patient\_id INT(10) primary key,

first\_name VARCHAR(100),

last\_name VARCHAR(100) ,

gender VARCHAR(10),

home\_phone INT(20),

cell\_phone INT(20),

emerg\_cont INT(20),

dob INT(20),

allergies VARCHAR(20) ,

surgical\_history VARCHAR(20),

visit\_date INT(20) ,

visit\_id VARCHAR(20),

lastdoctor\_id VARCHAR(20),

insurance VARCHAR(20),

medications VARCHAR(20)

)

--------------

--------------

SET SQL\_MODE = "NO\_AUTO\_VALUE\_ON\_ZERO"

--------------

--------------

SET time\_zone = "+00:00"

--------------

--------------

/\*!40101 SET @OLD\_CHARACTER\_SET\_CLIENT=@@CHARACTER\_SET\_CLIENT \*/

--------------

--------------

/\*!40101 SET @OLD\_CHARACTER\_SET\_RESULTS=@@CHARACTER\_SET\_RESULTS \*/

--------------

--------------

/\*!40101 SET @OLD\_COLLATION\_CONNECTION=@@COLLATION\_CONNECTION \*/

--------------

--------------

/\*!40101 SET NAMES utf8 \*/

--------------

--------------

1,1 Top

9 (updated)

or you can check at the captain’s mater directory

/home/mliu26/phase3/createtable.sql

**10. Constraints**

we specify all the the reasonably constraints on our tables

**11. Data Entry**

For each table, we have 20 tuples, so 20\*4 = 80 tuples, and everyone of us is responsible for 20 tuples in total.

**12. Show and List (finished)**

**13 Home page**

http://betaweb.csug.rochester.edu/~mliu26/CS261/home/csc261homepage.html

14 **Web Site DB Query 3 - Complex (M1)**

In our list page, we took our key attribute, PatientId, MedId, DiagId,StaffId and passed it into the url of our show page if the show button is clicked. This allows us to link to that patient’s show page. In that show page, we query each of the database’s table using the KeyId, and get each tuple using the Key. The resulting queries are displayed.

e.g.

[http://betaweb.csug.rochester.edu/~mliu26/CS261/yyu27/list.php](http://betaweb.csug.rochester.edu/~mliu26/CS261/yyu27/show.php)

[http://betaweb.csug.rochester.edu/~mliu26/CS261/mliu26/list.php](http://betaweb.csug.rochester.edu/~mliu26/CS261/yyu27/list.php)

[http://betaweb.csug.rochester.edu/~mliu26/CS261/ssun10/list.php](http://betaweb.csug.rochester.edu/~mliu26/CS261/yyu27/list.php)

http://betaweb.csug.rochester.edu/~mliu26/CS261/kcrist/list.php

15.**Web Site DB Query 4 - Search (M2)**

We used “<INPUT TYPE = "Submit" Name = "Submit1" VALUE = "search">, to submit the form, with the search value. So that only the tuple with the search value would appear. We also handled the exception of when an invalid value is searched by loading the entire form. The value is passed to this segment of code

// sql query formation echo "Query Generated:"; $sql = "select \* from $table where $attribute = $value";

**e.g.**

[http://betaweb.csug.rochester.edu/~mliu26/CS261/yyu27/list.php](http://betaweb.csug.rochester.edu/~mliu26/CS261/yyu27/show.php)

[http://betaweb.csug.rochester.edu/~mliu26/CS261/mliu26/list.php](http://betaweb.csug.rochester.edu/~mliu26/CS261/yyu27/list.php)

[http://betaweb.csug.rochester.edu/~mliu26/CS261/ssun10/list.php](http://betaweb.csug.rochester.edu/~mliu26/CS261/yyu27/list.php)

http://betaweb.csug.rochester.edu/~mliu26/CS261/kcrist/list.php

**16. Web Site DB Insert (M1)**

We created a link to our insertion page using a button. In our insertion pages, we assign a variable to each attribute. Then we use Post to ask the user to define the attribute. E.g $IPI = $\_POST['PI']; We then use insert, to put the attribute changes into the table. We then reload the table. Using isset we put in the new tuple and reload the table.

e.g [http://betaweb.csug.rochester.edu/~mliu26/CS261/mliu26/insertion.php](http://betaweb.csug.rochester.edu/~mliu26/CS261/yyu27/list.php)

[http://betaweb.csug.rochester.edu/~mliu26/CS261/yyu27/insertion.php](http://betaweb.csug.rochester.edu/~mliu26/CS261/yyu27/list.php)

[http://betaweb.csug.rochester.edu/~mliu26/CS261/kcrist/insertion.php](http://betaweb.csug.rochester.edu/~mliu26/CS261/yyu27/list.php)

[http://betaweb.csug.rochester.edu/~mliu26/CS261/ssun10/insertion.php](http://betaweb.csug.rochester.edu/~mliu26/CS261/yyu27/list.php)

**17.Web Site DB Update (M2)**

We created a link to our update page using a button from the show page. In our update pages, we assign a variable to each attribute. Then we use Post to ask the user to define the attribute. E.g $IPI = $\_POST['PI']; We then use isset to update the tuples in the table. We then reload the table with the updated tuple data. We used <INPUT TYPE="button" onClick="history.go(0)” value = ‘cancel’. To cancel and refresh the page if the input wasn’t.

**e.g.**

[http://betaweb.csug.rochester.edu/~mliu26/CS261/mliu26/update.php](http://betaweb.csug.rochester.edu/~mliu26/CS261/yyu27/list.php)

[http://betaweb.csug.rochester.edu/~mliu26/CS261/yyu27/update.php](http://betaweb.csug.rochester.edu/~mliu26/CS261/yyu27/list.php)

[http://betaweb.csug.rochester.edu/~mliu26/CS261/kcrist/update.php](http://betaweb.csug.rochester.edu/~mliu26/CS261/yyu27/list.php)

[http://betaweb.csug.rochester.edu/~mliu26/CS261/ssun10/update.php](http://betaweb.csug.rochester.edu/~mliu26/CS261/yyu27/list.php)

**18.Web Site DB Delete**

We created a link to our deletion page using a button. In our deletion pages, we give a keyid value, by using post. <INPUT TYPE = "Submit" Name = "Delete" VALUE = "Delete">, we have the keyid, we then search for the keyid that was defined using \_Post and we delete the tuple from the table. using if($IDI != 'Patient\_Id') // change to deletion {$sql2 = "delete from $table where Patient\_Id = $IDI"; . So we ask for an id, check for validity then delete the tuple using the id.

[http://betaweb.csug.rochester.edu/~mliu26/CS261/kcrist/deletion.ph](http://betaweb.csug.rochester.edu/~mliu26/CS261/yyu27/list.php)p

[http://betaweb.csug.rochester.edu/~mliu26/CS261/yyu27/deletion.ph](http://betaweb.csug.rochester.edu/~mliu26/CS261/yyu27/list.php)p

[http://betaweb.csug.rochester.edu/~mliu26/CS261/mliu26/deletion.ph](http://betaweb.csug.rochester.edu/~mliu26/CS261/yyu27/list.php)p

[http://betaweb.csug.rochester.edu/~mliu26/CS261/ssun10/deletion.php](http://betaweb.csug.rochester.edu/~mliu26/CS261/yyu27/list.php)