CSCI-8856: DATABASE MANAGEMENT SYSTEMS

Spring 2007

WEB BASED HEALTH INFORMATION FORUM

Phase 1 – Data Definition

Galapita Mudiyanselage Dilanga

02592558

WEB BASED HEALTH INFORMATION FORUM

This web based application is health information based platform, which both medical specialists and patients can discuss and share information related to various health issues. Mainly, this platform is providing two main services. One of the services of this system is discussion forum. This forum facilitates patients to ask questions related to health issues where as medical specialists can provide information related to the questions. The other service is medical specialists can use this platform to share medical related articles to acknowledge patients.

The forum has both private and public status which can be used to ask question related to health issues privately or publicly. Using the private mode, patients can get advices through the forum which is not shared with any other, while the public mode, can be used to share patients experiences to openly discussed. The questions can be tagged with related health issues so that, specialists who are specialized on that area will be automatically notified. Article sharing module of this application has the facility of attaching various kind of document (ex: Video, audio, text documents, presentations, publications) related to health issues. Registered patients can join on the discussion regarding the articles with the uploader.

According to the above description, this system mainly store data related to users, forum questions and answers, article contents and related discussions, user involvement history. These data can be use in this application such that, retrieving questions and related answers according to the search queries such as health issue, date range, user based. Articles can be searched and downloaded by the registered users with respect to varies queries with conditions. User related data stores for registration and authentication purposes.

User: Mainly two users, Doctor (medical specialist) and patient. User ID of doctor is doctor ID whereas for patient, it's patient Id.

Specialization: attribute which is representing the specialization area of a doctor.

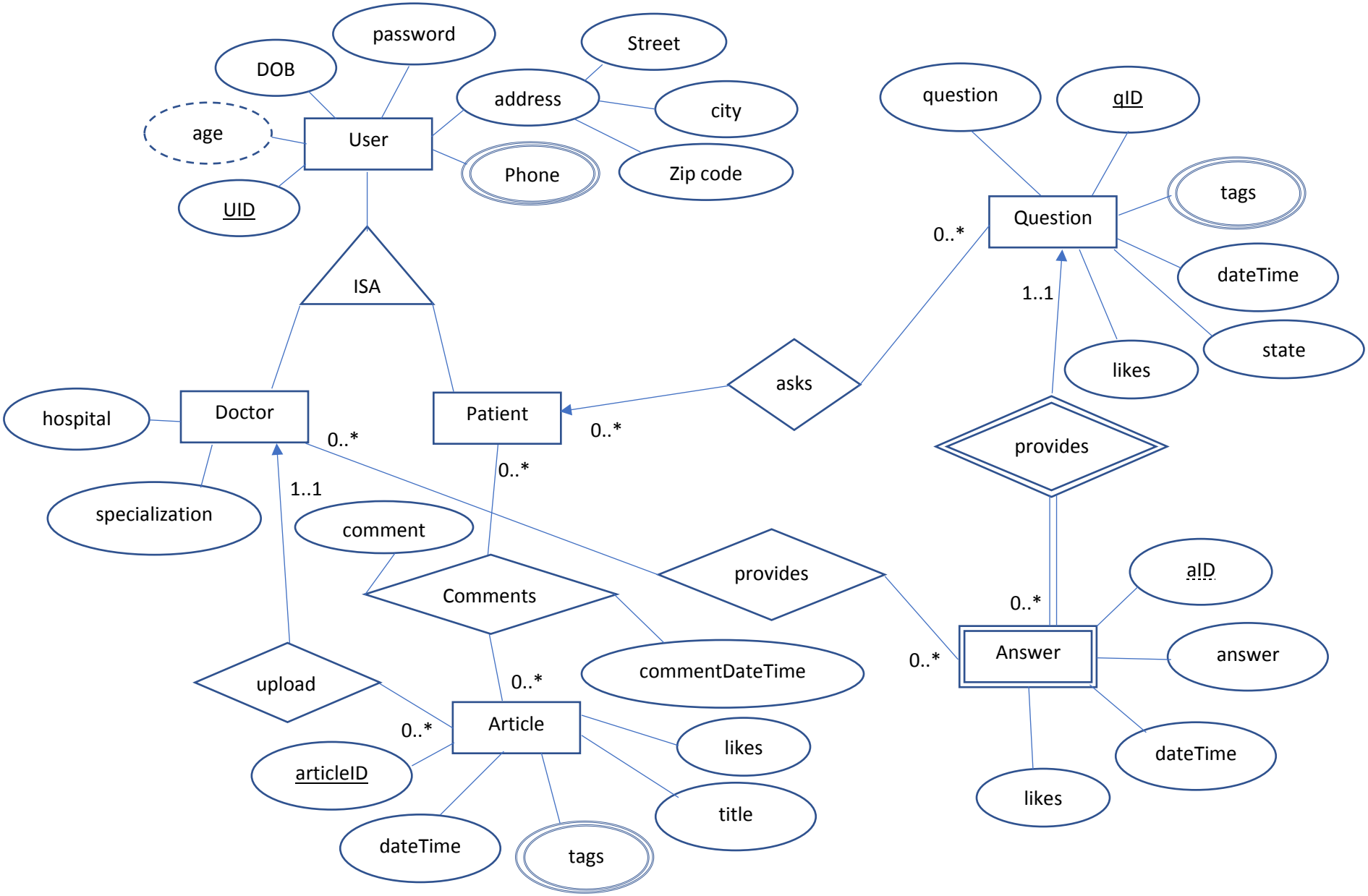
Article: One doctor can upload one or many articles.

Questions: If question is deleted from the forum, then all the answers should not be existing related to that question. So, answer entity is a weak entity.

State: If the state of the question is private, then only doctors can view those questions, if that is public, other patients can also share their experiences on that questions.

Comments: Patients can comment on articles uploaded by doctors.

Likes : Integer value of likes for particular content.


password
DOB
Street
address
city
age
User
Phone
Zip code
UID
ISA
question
qID
Question
tags
0..*
dateTime
1..1
likes
state
asks
hospital
Doctor
Patient
0..*
0..*
provides
0..*
1..1
specialization
comment
aID
provides
Comments
0..*
answer
upload
commentDateTime
0..*
Answer
0..*
Article
likes
articleID
dateTime
likes
title
dateTime
tags

Relational Schema

User(uid,type,street,city,zip_code,password,DOB)

Phone(uid,phone)

Doctor(doctor_id,type, hospital, specialization)

Patient(patient_id, type)

Question(q_id, title, question, question_date_time, state, likes)

Answer(q_id, a_id, answer, answer_date_time, likes)

Doctor_answer(doctor_id, q_id, a_id)

Question_tags(q_id, tag)

Article_tags(art_id,tag)

Article(art_id, title, upload_date, likes)

Patient_comment(art_id, patient_id, comment, comment_date)

SQL Queries

```
mysql> CREATE TABLE user(
    ->  u_id CHAR(10) NOT NULL,
    ->  type ENUM('doctor','patient'),
    ->  street VARCHAR(255),
    ->  city VARCHAR(50),
    ->  zip_code VARCHAR(5),
    ->  password VARCHAR(250),
    ->  dob DATE,
    ->  PRIMARY KEY(u_id)
    -> );
Query OK, 0 rows affected (0.01 sec)

mysql> CREATE TABLE doctor(
    ->  doctor_id CHAR(10),
    ->  type ENUM('doctor'),
    ->  hospital VARCHAR(100),
    ->  specialization VARCHAR(100),
    ->  PRIMARY KEY (doctor_id),
    ->  CONSTRAINT fk_doctor_user FOREIGN KEY (doctor_id,type)
    ->          REFERENCES user(u_id,type)
    ->                  ON DELETE CASCADE
    ->                  ON UPDATE CASCADE
    -> );
Query OK, 0 rows affected (0.00 sec)

mysql> CREATE TABLE patient(
    ->  patient_id CHAR(10),
    ->  type ENUM('patient'),
    ->  PRIMARY KEY (patient_id),
    ->  CONSTRAINT fk_patient_user FOREIGN KEY (patient_id,type)
    ->          REFERENCES user(u_id,type)
    ->                  ON DELETE CASCADE
    ->                  ON UPDATE CASCADE
    -> );
Query OK, 0 rows affected (0.01 sec)

mysql> CREATE TABLE phone(
    ->  u_id CHAR(10) NOT NULL,
    ->  phone CHAR(10) NOT NULL,
    ->  PRIMARY KEY(u_id),
    ->          CONSTRAINT fk_phone_user FOREIGN KEY (u_id)
    ->                  REFERENCES user(u_id)
    ->                          ON DELETE CASCADE
    ->                          ON UPDATE CASCADE
    -> );
Query OK, 0 rows affected (0.00 sec)
```

```
mysql> CREATE TABLE question(
    ->  q_id CHAR(8) NOT NULL,
    ->  patient_id CHAR(10),
    ->  title VARCHAR(100),
    ->  question VARCHAR(255),
    ->  question_time DATETIME,
    ->  state ENUM('private','public'),
    ->  likes INT,
    ->  PRIMARY KEY(q_id),
    ->  CONSTRAINT fk_patient_question FOREIGN KEY (patient_id)
    ->          REFERENCES patient(patient_id)
    ->                  ON DELETE CASCADE
    ->                  ON UPDATE CASCADE
    -> );
Query OK, 0 rows affected (0.01 sec)

mysql> CREATE TABLE answer(
    ->  q_id CHAR(8) NOT NULL,
    ->  ans_id CHAR(8) NOT NULL,
    ->  answer VARCHAR(255),
    ->  answer_time DATETIME,
    ->  likes INT,
    ->  PRIMARY KEY(ans_id,q_id),
    ->  CONSTRAINT fk_question_answer FOREIGN KEY (q_id)
    ->          REFERENCES question(q_id)
    ->                  ON DELETE CASCADE
    ->                  ON UPDATE CASCADE
    -> );
Query OK, 0 rows affected (0.01 sec)

mysql> CREATE TABLE doctor_answer(
    ->  doctor_id CHAR(10) NOT NULL,
    ->  q_id CHAR(8) NOT NULL,
    ->  ans_id CHAR(8) NOT NULL,
    ->  PRIMARY KEY(doctor_id,q_id,ans_id),
    ->  CONSTRAINT fk_doctor_answer FOREIGN KEY (doctor_id)
    ->          REFERENCES doctor(doctor_id)
    ->                  ON DELETE CASCADE
    ->                  ON UPDATE CASCADE,
    ->
    ->  CONSTRAINT fk_answer_d_ans FOREIGN KEY (q_id,ans_id)
    ->          REFERENCES answer(q_id,ans_id)
    ->                  ON DELETE CASCADE
    ->                  ON UPDATE CASCADE
    -> );
Query OK, 0 rows affected (0.01 sec)
```

```
mysql> CREATE TABLE article(
    ->  art_id CHAR(8) NOT NULL,
    ->  doctor_id CHAR(10) NOT NULL,
    ->  title VARCHAR(100),
    ->  description VARCHAR(255),
    ->  article_link VARCHAR(255),
    ->  upload_date DATE,
    ->  likes INT,
    ->  PRIMARY KEY(art_id),
    ->  CONSTRAINT fk_doctor_article FOREIGN KEY (doctor_id)
    ->          REFERENCES doctor(doctor_id)
    ->                  ON DELETE CASCADE
    ->                  ON UPDATE CASCADE
    -> );
Query OK, 0 rows affected (0.01 sec)

mysql> CREATE TABLE patient_comment(
    ->  art_id CHAR(8) NOT NULL,
    ->  patient_id CHAR(10) NOT NULL,
    ->  comment VARCHAR(255),
    ->  commentDate DATE,
    ->  PRIMARY KEY(art_id,patient_id),
    ->  CONSTRAINT fk_patent_article FOREIGN KEY (patient_id)
    ->          REFERENCES patient(patient_id)
    ->                  ON DELETE CASCADE
    ->                  ON UPDATE CASCADE,
    ->
    ->  CONSTRAINT fk_article_patient FOREIGN KEY (art_id)
    ->          REFERENCES article(art_id)
    ->                  ON DELETE CASCADE
    ->                  ON UPDATE CASCADE
    -> );
Query OK, 0 rows affected (0.01 sec)


mysql> CREATE TABLE question_tags(
    ->  q_id CHAR(10) NOT NULL,
    ->  tag VARCHAR(50),
    ->  PRIMARY KEY(q_id,tag),
    ->  CONSTRAINT fk_question_tags FOREIGN KEY (q_id)
    ->          REFERENCES question(q_id)
    ->                 ON DELETE CASCADE
    ->                 ON UPDATE CASCADE
    -> );
Query OK, 0 rows affected (0.00 sec)

mysql> CREATE TABLE article_tags(
    ->  art_id CHAR(10) NOT NULL,
    ->  tag VARCHAR(50),
    ->  PRIMARY KEY(art_id,tag),
    ->  CONSTRAINT fk_question_tags FOREIGN KEY (art_id)
    ->          REFERENCES article(art_id)
    ->                  ON DELETE CASCADE
    ->                  ON UPDATE CASCADE
    -> );
Query OK, 0 rows affected (0.01 sec)
```

```
mysql> SELECT * FROM user;
+------------+---------+----------------+-------+----------+----------+------------+
| u_id       | type    | street         | city  | zip_code | password | dob        |
+------------+---------+----------------+-------+----------+----------+------------+
| D001212121 | doctor  | 23 Cass St.    | Omaha | 68114    | pass     | 1988-12-02 |
| D001212122 | doctor  | 1212 Dodge St. | Omaha | 68144    | pass     | 1982-01-02 |
| P001212121 | doctor  | 23 Cass St.    | Omaha | 68114    | pass     | 1981-12-02 |
| P001212122 | patient | 1212 Dodge St. | Omaha | 68144    | pass     | 1984-01-02 |
+------------+---------+----------------+-------+----------+----------+------------+
4 rows in set (0.00 sec)

mysql> SELECT * FROM doctor;
+------------+--------+----------------+------------------+
| doctor_id  | type   | hospital       | specialization   |
+------------+--------+----------------+------------------+
| D001212121 | doctor | Omaha hospital | Cardiologist     |
| D001212122 | doctor | Omaha hospital | Anesthesiologist |
+------------+--------+----------------+------------------+
2 rows in set (0.00 sec)

mysql> SELECT * FROM patient;
+------------+---------+
| patient_id | type    |
+------------+---------+
| P001212121 | patient |
| P001212122 | patient |
+------------+---------+
2 rows in set (0.00 sec)

mysql> SELECT * FROM article_tags;
+----------+-----+
| art_id   | tag |
+----------+-----+
| Art00121 | CT  |
| Art00121 | MRI |
+----------+-----+
2 rows in set (0.00 sec)

mysql> SELECT * FROM question_tags;
+----------+-----+
| q_id     | tag |
+----------+-----+
| Q0000001 | CT  |
| Q0000001 | MRI |
+----------+-----+
2 rows in set (0.00 sec)

mysql> SELECT * FROM question;
+----------+------------+------------------+------------+---------------------+---------+-------+
| q_id     | patient_id | title            | question   | question_time       | state   | likes |
+----------+------------+------------------+------------+---------------------+---------+-------+
| Q0000001 | P001212121 | Question title 1 | Question 1 | 1999-12-31 12:59:59 | public  |    12 |
| Q0000002 | P001212122 | Question title 2 | Question 2 | 1999-12-31 12:59:59 | private |     1 |
+----------+------------+------------------+------------+---------------------+---------+-------+
2 rows in set (0.00 sec)
```

```
mysql> SELECT * FROM answer;
+----------+----------+----------+---------------------+-------+
| q_id     | ans_id   | answer   | answer_time         | likes |
+----------+----------+----------+---------------------+-------+
| Q0000001 | A0000001 | Answer 1 | 1999-12-31 23:59:59 |    12 |
| Q0000002 | A0000001 | Answer 2 | 1999-12-31 12:59:59 |     1 |
+----------+----------+----------+---------------------+-------+
2 rows in set (0.00 sec)
```

```
mysql> SELECT * FROM phone;
+------------+------------+
| u_id       | phone      |
+------------+------------+
| P001212121 | 4025559666 |
| D001212122 | 4027894562 |
+------------+------------+
2 rows in set (0.00 sec)
```

```
mysql> SELECT * FROM doctor_answer;
+------------+----------+----------+
| doctor_id  | q_id     | ans_id   |
+------------+----------+----------+
| D001212121 | Q0000001 | A0000001 |
| D001212121 | Q0000002 | A0000001 |
+------------+----------+----------+
2 rows in set (0.00 sec)
```

```
mysql> SELECT * FROM patient_comment;
+----------+------------+-------------------+-------------+
| art_id   | patient_id | comment           | commentDate |
+----------+------------+-------------------+-------------+
| Art00121 | P001212121 | Article comment 1 | 2012-12-02  |
| Art00122 | P001212121 | Article comment 2 | 2013-12-02  |
+----------+------------+-------------------+-------------+
2 rows in set (0.00 sec)
```

```
mysql> SELECT * FROM article;
+----------+------------+-----------------+----------------+----------------------+-------------+-------+
| art_id   | doctor_id  | title           | description    | article_link         | upload_date | likes |
+----------+------------+-----------------+----------------+----------------------+-------------+-------+
| Art00121 | D001212121 | Article title 1 | Article desc 1 | path\upload\article | 2012-12-02  |    13 |
| Art00122 | D001212122 | Article title 2 | Article desc 2 | path\upload\article | 2010-12-02  |     1 |
+----------+------------+-----------------+----------------+----------------------+-------------+-------+
2 rows in set (0.00 sec)
```