

UNIVERSITY OF ALBERTA

TERM PROJECT

RaySyS

Steven MESHMEYER
Camden NARZT
Oscar RAMIREZ

Professor:
Dr. Li-Yan YUAN

April 4, 2011

RaySys

RaySys consists of several modules that allow different types of users to manage and interact a Radiological Database System. With this application users are divided into four different categories: administrators, patients, doctors, and radiologists each having different privileges. In order to summarize the system architecture and the major modules used in the system, along with their relationships and main interactions with the database this document will first describe those sections that are common to all users, and then it will be divided into specific user sections. Each section will contain a listing of the relevant **.jsp* and **.java* files for the module, along with the initial *URL* which can be used to access it.

Login

The login module is the first place the user will interact with RaySys. The user will first encounter this when accessing the system for the first time. At this point a login screen requesting a **username** and a **password** is shown. If the user enters information for a valid account a *session variable* **userName** will be set be a String representing the user's account type. Also as expected if the information is invalid the user will be notified appropriately.

The action of logging in utilizes a the servlet **LoginManager.java** which is located in the **loginModule** package. Through this server a prepared statement is created with the form **select password, class from users where user_name = ?** and the name is set to equal the username provided. The result set that is generated from executing this query is then inspected and if a password was returned it is compared to the provided one by the user.

URL: /radiologydb/login.jsp

**.jsp:* login.jsp

**.java:* LoginManager.java

Profile

Another part of the login module is the profile. This provides an interface for the user to update most of his information. All users are allowed to change: *first name*, *last name*, *password*, *address*, *email*, and *phone*. This section of the login module takes advantage of a servlet that is also used for updating user information by admins so the specifications of the servlet used can be found in the *User Update* section.

URL: /radiologydb/profile.jsp

**.jsp:* profile.jsp

**.java:* UserUpdate.java

Note: Further information found in *User Update* section

Connections

At this point it is worth mentioning how the connections to the database are managed in this application. #TODO

**.java:* ConnectionManager.java

Search

Search is available to all users, however the results are filtered depending on the user class. #TODO

URL: /radiologydb/index.jsp

**.jsp:* index.jsp

Adding Users

Adding users is only available to admins and the interface is very simple. Admins can add users by only setting the *username*, *password*, and *class*. This will add an entry to the **users** table in the database by utilizing the prepared statement **insert into users values (?, ?, ?, ?)** where the values correspond to the ones listed before hand, and the fourth value is set to the registration time which is generated when the user is added.

If the administrator decides to fill the remaining fields *first name*, *last name*, *address*, *email*, *phone* an entry will be created in the persons entry with the supplied information.

```
URL: /radiologydb/new-uesr.jsp
*.jsp: new-user.jsp
*.java: NewUser.java
```

User Update

The user update module's main purpose is to be utilized by administrators. This module provides an interface for administrators to update all information for any user type. The interface for this begins by providing the administrator a search for users where he can search by username, name, or last name and can then select what user to edit from the displayed results.

The interface for changing basic information is very similar to the one used to add users so no explanation should be needed. If the user to be modified is a patient or a doctor this module will display a list of all the doctors or patients available for choosing and a second list with the ones that are currently the patient/doctor's patients/doctors.

In order to edit user information two prepared statements, similar to those for adding new users, are used. The first prepared statement used is `update users set password=?, class = cast(? as char(1)) where user_name =?` and `update persons set first_name=?, last_name=?, address=?, email=?, phone =? where user_name=?`.

In order to display the available doctors the statement `select UNIQUE patient_name from family_doctor where doctor_name = ? + username + ?` is used. In order to make a change to the *family_doctor* table we will either add a new row with `insert into family_doctor values(? + username + ?, ? + uname + ?)` or remove a row with `delete from family_doctor where doctor_name = ? + username + ? and patient_name = ? + uname + ?`. Very similar queries are used for getting, adding and deleting patient rows.

```
URL: /radiologydb/update/index.jsp
*.jsp: /update/index.jsp /index/user-search.jsp /index/update.jsp
*.java: UserUpdate.java
```

Report Generation

Record Creation

Image Upload

Conclusion