

**School of Computer Engineering**

**SCE11-0353**

**WEB APPLICATION FOR RADIOLOGY INFORMATION SYSTEM**

Submitted in Partial Fulfillment of the Requirements for the

Degree of Bachelor of Computer Engineering of the

Nanyang Technological University

Author: Desmond Poh Lik Meng (U0922923K)

Supervisor: Assoc Prof Lin Feng

2012

# Abstract

Telemedicine is emerging as a fast growing medical technology in the health care industry. It helps to eliminate distance barriers and can improve access to medical services that would often not be usually available in distant rural communities. Created from a convergence of telecommunication and information technologies, telemedicine permits communications between the patient and healthcare provider with fidelity and convenience.

The most popular use of telemedicine is via teleradiology, which is the ability to send medical images from one location to another. Traditionally, it requires a sending station, a receiving station, and a transmission network. However, with today’s high-speed broadband Internet, healthcare professionals and patients alike can view medical images stored on remote servers; only a standard Personal Computer and DSL/Cable Internet connection is needed.

In this project, we will be developing a web based Radiology Information System (RIS), in an attempt to migrate the information systems used in radiology departments online. It aims to let healthcare professionals access patient medical data at any time, while letting patients themselves monitor their health.

The development process, from its inspiration to implementation to wrap up, is documented in this report. Special attention is given to discussing the application design, difficulties encountered during design and development, the strengths and weaknesses of the application, and areas that can be improved upon in future.

// TODO: Resummarize this report after finish the entire project

# Acknowledgement

I would like to express my deepest gratitude to the following individuals who have provided me valuable insights, help and guidance for completing my Final Year Project.

* Dr. Lin Feng, my project supervisor. His guidance and encouragement has proved invaluable at all stages of this project. Were it not for him, this project would not have seen the light of day.
* The staff members at the Radiology Department of Raffles Medical Group. Their paid introductory tour of a typical workflow process of their department gave many insights into the potential challenges this project would be facing.
* The team at BitBucket, a source version control vendor. Their graciousness to upgrade the code repository at no cost meant that critical mistakes made during development can be rolled back without any down time.
* All my friends and family who have supported and helped me during the time which this project was going through a difficult development period.

# Table of Contents

[Abstract ii](#_Toc315181859)

[Acknowledgement iii](#_Toc315181860)

[Table of Contents iv](#_Toc315181861)

[List of Figures vi](#_Toc315181862)

[List of Tables vii](#_Toc315181863)

[Chapter 1 Introduction 1](#_Toc315181864)

[1.1 Background 1](#_Toc315181865)

[1.2 Objective 2](#_Toc315181866)

[1.3 Scope 3](#_Toc315181867)

[1.4 Project Planning and Schedule 4](#_Toc315181868)

[1.5 Resources 6](#_Toc315181869)

[1.6 Report Organization 7](#_Toc315181870)

[Chapter 2 Literature Review 8](#_Toc315181871)

[2.1 Radiology Business Model 8](#_Toc315181872)

[2.2 ASP.NET 11](#_Toc315181873)

[2.3 C# 11](#_Toc315181874)

[2.4 Microsoft SQL Server 2008 R2 11](#_Toc315181875)

[2.5 DICOM 12](#_Toc315181876)

[Chapter 3 Database System Design 14](#_Toc315181877)

[3.1 Overview 14](#_Toc315181878)

[3.2 ASP.NET Tables 16](#_Toc315181879)

[3.3 Radiology Information System (RIS) Tables 22](#_Toc315181880)

[Chapter 4 Web Application Modules 29](#_Toc315181881)

[Chapter 5 Experimental Systems and Results 30](#_Toc315181882)

[Chapter 6 Discussion and Conclusion 31](#_Toc315181883)

[6.1 Summary 31](#_Toc315181884)

[6.2 Discussion 31](#_Toc315181885)

[6.3 Future Work 31](#_Toc315181886)

[References 32](#_Toc315181887)

[Appendix 33](#_Toc315181888)

# List of Figures

[Figure 1: Integrating Medical Peripherals into the System 58](file:///E:\Temp\Projects\FYP\Reference\FYP%20Report\Amended%20FYP%20Final%20Report%20v2.doc#_Toc226119301)

# 

# List of Tables

[Table 7 : Web Links table field description 38](#_Toc226119313)

# Introduction

## 1.1 Background

Telemedicine refers to provision of medical information and services via telecommunication devices. Created from a convergence of telecommunication and information technologies, it not only permits long distance communication between healthcare providers, but also with patients, via today’s broadband Internet infrastructure.

Telemedicine is part of a larger movement towards electronic based healthcare (eHealth). Compared to traditional healthcare, the concept of eHealth is based upon the central idea of storing all medical information, records and images electronically. Although adoption of eHealth varies widely, both internationally and within Singapore itself, most medical institutions have already migrated much of their paper-based processes to Healthcare Information Systems (HIS).

Complementing a HIS is a specialized subsystem called Radiology Information System (RIS). As its name implies, a RIS is used almost exclusively by the radiology department of a medical institution. In layman’s terms, a RIS is a computerized database for storing, manipulating and distributing patient radiological data and imagery.

## 1.2 Objective

The objectives of this project are as follows:

* To allow radiologists to work using the RIS system, and its underlying database
* To allow radiologists to retrieve imaging orders for a patient referred to them
* To allow radiologists to store medical image(s) taken
* To allow radiologists to enter data relating to the medical image
* To access the Outpatient Information System (OPIS) database, which is being developed concurrently in another project, in future

## 1.3 Scope

The scope of this project is as follows:

* Visit a radiology department of any medical institution to gather as much information as possible on the workflow process
* Understand the commonly used technical jargons in radiology
* Analyze the requirements of the RIS based upon information gathered
* Learn and understand DICOM, the standard used in medical imaging
* Learn ASP.NET, the web technology upon which the RIS will be hosted on
* Learn C#, the primary programming language that will be used in this project
* Learn SQL Server 2008 R2, the database that will be used in this project
* Design and develop the backend database iteratively
* Design and develop the RIS web application iteratively
* Learn about technologies that might potentially shorten the development time of the RIS and utilize them, if possible
* Explore about the possibility of optimizing the RIS for mobile platforms, if time permits

The fully implemented RIS shall fulfill the objectives of this project, as stated in section 1.2.

## 1.4 Project Planning and Schedule

The initial proposed project schedule is indicated in Figure 1. As the project goes into the implementation stage, it is expected that difficulties will force a rescheduling of the development flow.

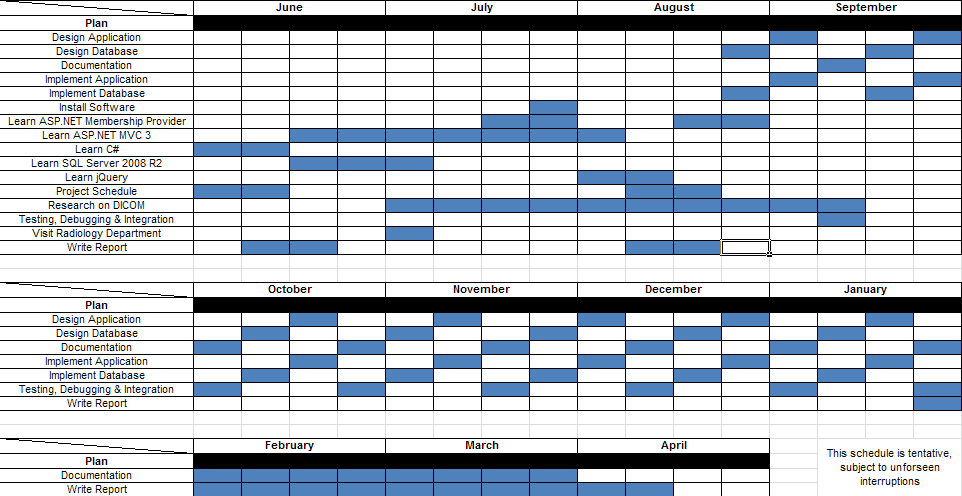


Figure 1 – Original Schedule

Figure 2 shows the actual development that took place, due to the expected difficulties that were predicted during initial planning.

Notice that development activity became unusually heavy since the second week of December. This was due to discovering that NHibernate, a critical software framework for abstracting interactions with the underlying Microsoft SQL Server 2008 R2, was deemed to be incompatible with a new feature of the database very late during development.

As such, all work that was done up to that point had to be abandoned, and the project essentially restarted from scratch. Moreover, the DICOM standard specifications are still posing a challenge at the time of writing, thereby necessitating more research activity on it until the last iteration of development.

Thankfully, despite the restarting of development work, the database schema was able to be reused due to a conscious effort from the beginning to separate the data persistence and application logic layers.

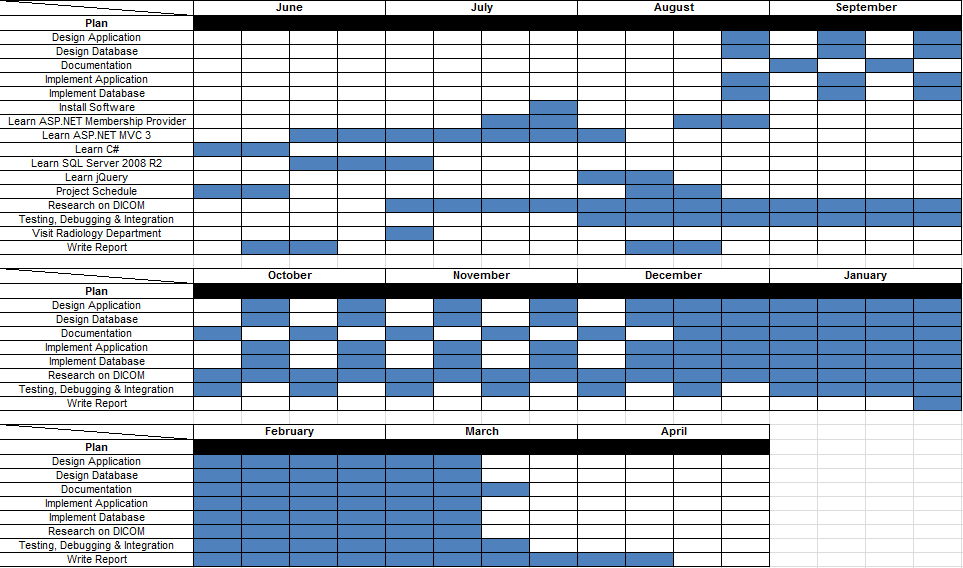


Figure 2 – Actual Schedule

## 1.5 Resources

Two specific software suites were utilized in the design and development of RIS:

* Microsoft SQL Server 2008 R2
* Microsoft Visual Studio 2010 Ultimate

In addition to design and development, this project also utilized other software as follows:

* AJAX Control Toolkit. An AJAX UI framework built on top of the default ASP.NET AJAX controls
* ASP.NET. A web application framework from Microsoft to build dynamic web applications
* dotPeek. A .NET decompiler
* Mercurial. A revision control managing the changes to program files, database files, documentation etc.
* Microsoft Office. A productivity suite for formally documenting the development process in the form of a report
* NuGet. A package manager for the .NET Framework
* ReSharper. A productivity extension to Visual Studio that helps to maintain and improve C# code (among many others) in Visual Studio projects.
* Telerik RadControls for ASP.NET AJAX. Another AJAX UI framework built on top of the default ASP.NET AJAX controls.

## 1.6 Report Organization

This report is organized in the following manner:

**Chapter 1** presents an introduction the reader regarding the project. It includes the background information to help readers understand the objectives of the project.

**Chapter 2** presents an explanation of the business and working model of a sample radiology department. It gives a sample radiology workflow and layman explanations of jargons. A literature review of DICOM, C#, Microsoft SQL Server 2008 R2 and aspects of ASP.NET that were utilized is also given.

**Chapter 3** presents the database system design that is used in the web application. It provides the specifics of the schema of the database and provides the rationale for the choices made, where applicable.

**Chapter 4** gets into the technical implementation of the web application. The server and client side designs are elaborated upon in this section. It provides the rationale and advantages of critical design decisions.

**Chapter 5** presents the experimental system and results. Details will be given on the rationale behind implementation. Where appropriate, screenshots will be provided.

**Chapter 6** concludes the project and includes discussions on possible future development and enhancements.

# Literature Review

## 2.1 Radiology Business Model

To better understand the workflow processes and jargons used within a radiology department, a paid, introductory tour was arranged with Raffles Medical Group. The tour occurred during the first week of July 2011.

According to the medical center, a typical process for outpatients would be as shown in Figure 3, whereas a typical process for inpatients would be as shown in Figure 4.

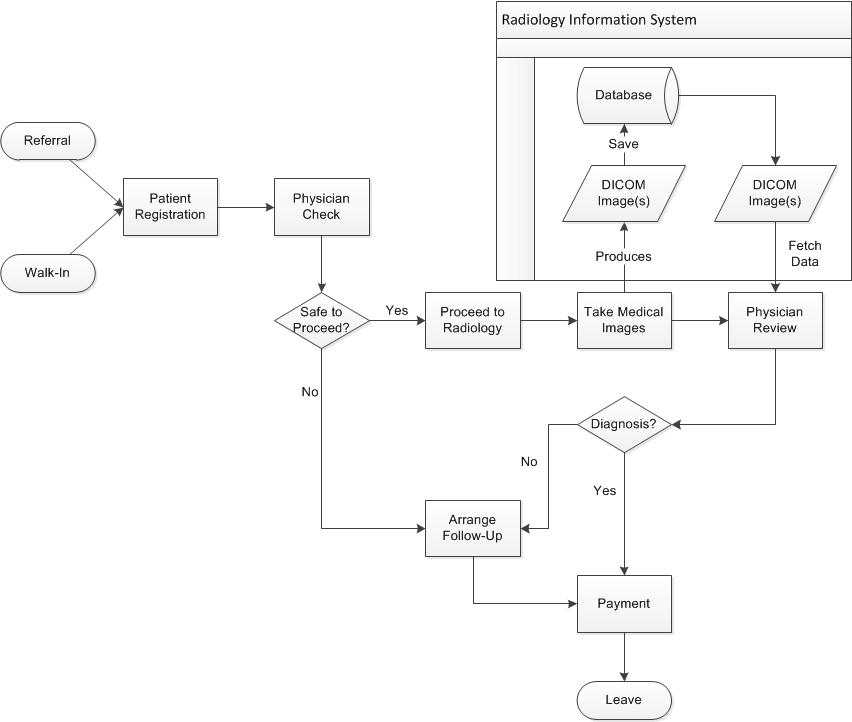


Figure 3 – Outpatient Process

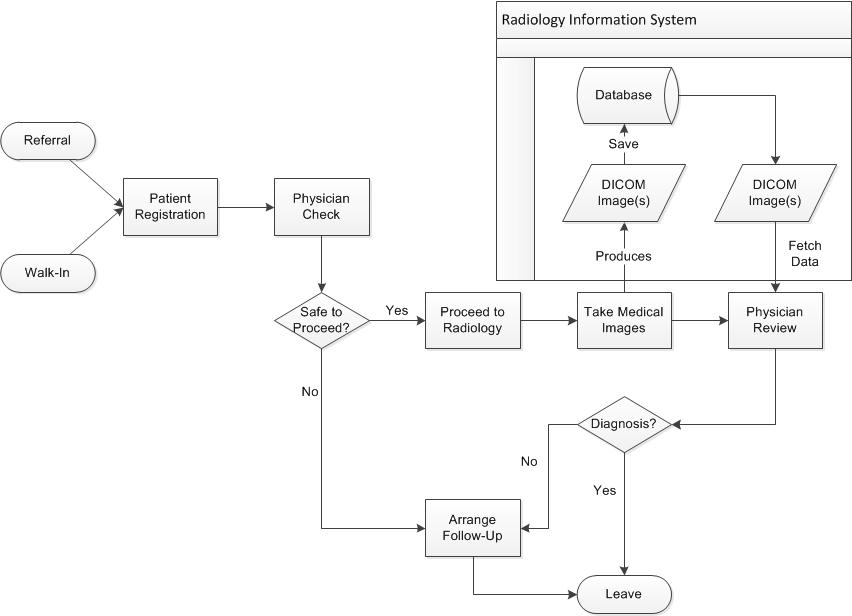


Figure 4 – Inpatient Process

For both processes, we see that the RIS is only used during the saving and retrieving of medical images. However upon further analysis, it was deemed necessary for the RIS to be able to track the patient’s progress within the workflow, so that staff members would know where next to direct him/her.

When a patient is referred to the radiology department, s/he has started a study. In a study, different types of imaging equipment (series) might be used. Each series represents a different modality, and can have several images. This information is also encapsulated in DICOM (explained later), as seen in the image below.

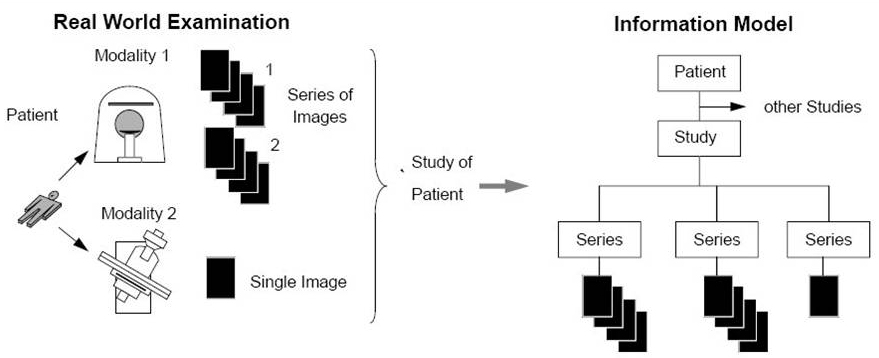


Figure 5 - Radiology to DICOM

## 2.2 ASP.NET

ASP.NET is a web application framework by Microsoft. It is built on top the Common Language Runtime, allowing .NET supported languages to be used. The main building block of ASP.NET is its web pages, officially known as Web Forms.

ASP.NET web forms contain static HTML markup, as well as markup defining server-side web controls and user controls for both static and dynamic content on the page. Static content is placed in the client facing .aspx file, while dynamic content server are placed within a <% %> block in a separate .aspx.cs (otherwise known as a code-behind file) linked that the .aspx file.

An ASP.NET project has three other file types that aid in reducing development time – the web.config, master and sitemap files. Web.config is an XML document containing the main settings and configuration file for an ASP.NET application. Specifically, the web.config file contains information that controls module loading, security configuration, session state, application language, compilation settings and database connection strings. The master file is the file from which all other .aspx files in the application derives their layout from, and the sitemap file simplifies site management by consolidating all web pages into one centralized location.

Recent versions of ASP.NET added in a Membership provider feature, and AJAX framework. Membership handles common security tasks such as logins, registration, authentication and role management, while AJAX provides asynchronous controls for the client. The RIS has two extensions of the AJAX framework installed – AJAX Control Toolkit and Telerik RadControls. Both controls further simplify AJAX based controls in the application itself.

## 2.3 C#

C# is a programming language designed for the Common Language Runtime. The most recent version, 4.0, was released on April 2010. C# is most commonly used for application and business logic in the code-behind files for the ASP web pages.

## 2.4 Microsoft SQL Server 2008 R2

Microsoft SQL Server 2008 R2 is an updated release of Microsoft SQL Server 2008. The key feature of this release is a feature named FileStream, which integrates the database engine with NTFS by storing varbinary(max) binary large object (BLOB) data as files on the file system itself. There are three key reasons why FileStream is used for RIS:

1. DICOM files are on average larger than 1MB in size
2. Fast read access is considered an important requirement
3. RIS uses a middle tier for application and business logic

The major disadvantage of FileStream is its incompatibility with NHibernate, an open source object-relational mapper. As such, a significant portion of relational data persistence related programming tasks that could have been abstracted now needs to be done manually.

Like other major commercial database management systems, SQL Server has its own management software called SQL Server Management Studio. It is through this software that most of the initial interactions with SQL Server take place, before it is accessed programmatically by RIS.

## 2.5 DICOM

DICOM is a standard for the creation, transmission, and storage of digital medical image and report data. It defines a data dictionary, data structures, file formats, client and server services, workflow, and compression, among other things. The most important aspects of DICOM are that it contains a file header portion, a File Meta Information portion, and a single SOP (Service-Object Pair) instance [1].

The header is made up of a 128 byte preamble, followed by the characters ‘D’, ‘I’, ‘C’, ‘M’. The preamble is usually not used, and thus contains all zeroes, although applications may use it for proprietary data [2].

After the header is the File Meta Information. This portion follows a tagged file format, and contains information about the file, the series and study it belongs to, and the patient that it belongs to [1]. This information is frequently parsed and used in the RIS itself.

A visual reference of a DICOM file is provided in the following page.

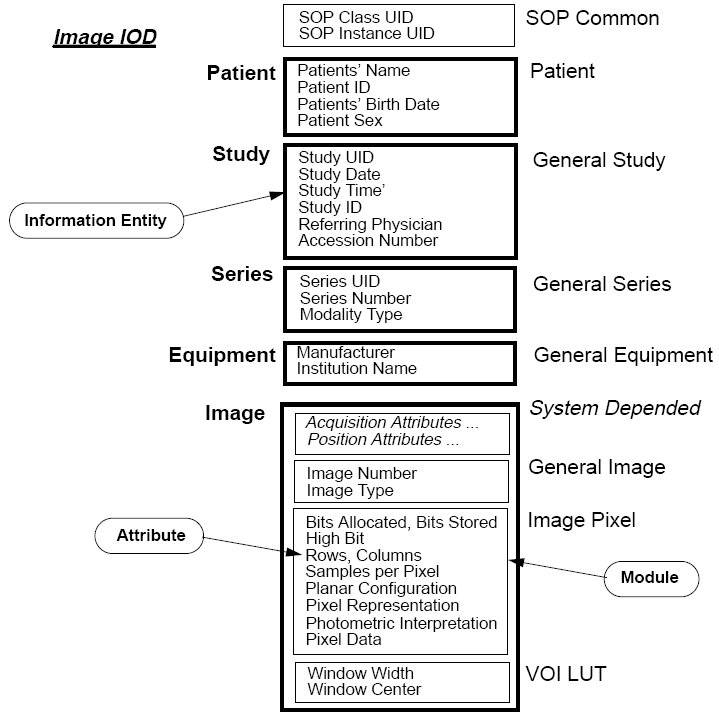


Figure 6 - Sample DICOM File

# **Database System Design**

## 3.1 Overview

The RIS web application database is made up of two logically distinct collections of SQL tables. The first collection of tables is used to model domain entities in an object relational environment, and the other collection of tables is used to handle security issues common to web applications.

The latter is offloaded to ASP.NET’s Membership framework, which is automatically installed together with the .NET framework. It handles logins, registrations, authentication and role management; thereby reducing development time, and thus freeing up more time to solve problems in the business domain.

Both collections of tables are linked together with several referential integrities, and located within the same database system, so as not to make the system overly complicated. A point to take note of is that although security has been offloaded to ASP.NET, code still needs to be written to call the Membership framework functions.

Figure 5 shows an overview of all the SQL tables used in the RIS web application. For an overall view of all the tables and their relationships, please refer to the appendix.

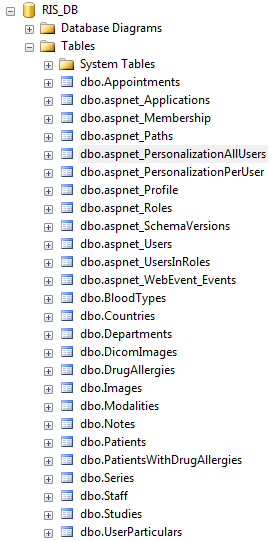


Figure 7 – All Database Tables

## 3.2 ASP.NET Tables

As described in the previous section, the ASP.NET Membership framework handles user registration, login, authentication and role management features. Detailed schemas for the ASP.NET Membership tables are included below.

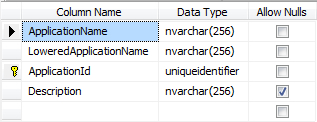


Figure 8 - aspnet\_Applications

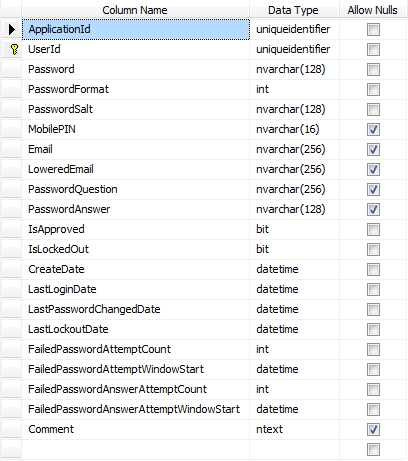


Figure 9 - aspnet\_Membership

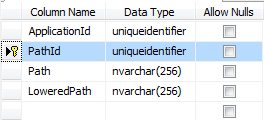


Figure 10 - aspnet\_Paths

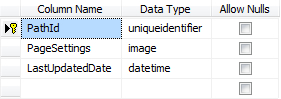


Figure 11 - aspnet\_PersonalizationAllUsers

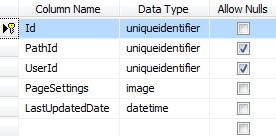


Figure 12 - aspnet\_PersonalizationPerUser

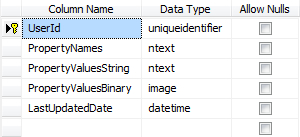


Figure 13 - aspnet\_Profile

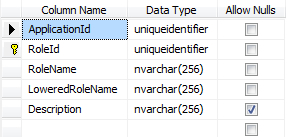


Figure 14 - aspnet\_Roles

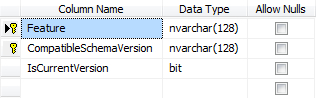


Figure 15 - aspnet\_SchemaVersions

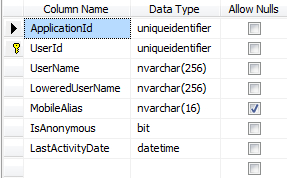


Figure 16 - aspnet\_Users

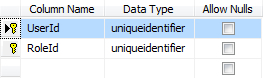


Figure 17 - aspnet\_UsersInRoles

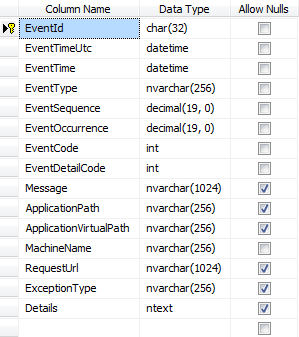


Figure 18 - aspnet\_WebEvent\_Events

To create the above tables in the database, we navigate to C:\%windir%\Microsoft.NET\Framework\version and run the aspnet\_regsql executable tool.

Thereafter, we have to enable the Membership feature in the global web.config file by adding the following lines of code:



Figure 19 - Membership Configuration

## 3.3 Radiology Information System (RIS) Tables

In addition to the ASP.NET default tables, the RIS web application also has its own tables that capture the domain entities of a radiology department in object relational tables as shown below. All primary key columns with integer data type are configured to be auto-incrementing identity values.

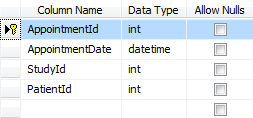


Figure 20 – Appointments

The Appointments table has:

* One-to-many relationship with Studies table
* Many-to-one relationship with Patients table

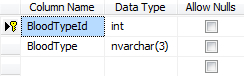


Figure 21 - BloodTypes

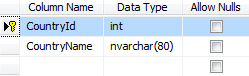


Figure 22 - Countries

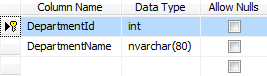


Figure 23 - Departments



Figure 24 – DicomImages

The DicomImages table uses a Globally Unique Identifier (GUID) as its data type for the primary key column. This number is automatically generated by SQL Server, and guaranteed to be a unique value based upon a time seed.

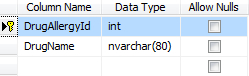


Figure 25 – DrugAllergies

The DrugAllergies table has:

* Many-to-many relationship with Patients table

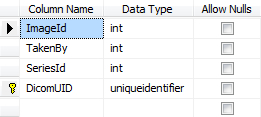


Figure 26 – Images

The Images table has:

* Many-to-one relationship with Staff table
* Many-to-one relationship with Series table
* One-to-one relationship with DicomImages table

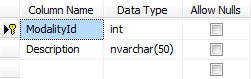


Figure 27 - Modalities

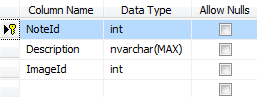


Figure 28 – Notes

The Notes table has:

* Many-to-one relationship with Image table

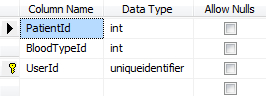


Figure 29 – Patients

The Patients table has:

* Many-to-one relationship with BloodType table
* One-to-one relationship with aspnet\_Users table
* Many-to-many relationship with DrugAllergies table

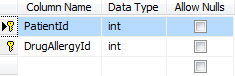


Figure 30 – PatientsWithDrugAllergies

The PatientsWithDrugAllergies is a join table between Patients and DrugAllergies.

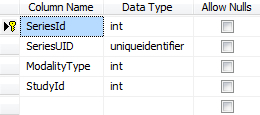


Figure 31 – Series

The Series table has:

* Many-to-one relationship with Modalities table
* Many-to-one relationship with Studies table

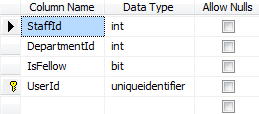


Figure 32 – Staff

The Staff table has:

* Many-to-one relationship with Department table
* One-to-one relationship with aspnet\_Users table

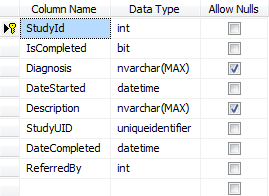


Figure 33 – Studies

The Studies table has

* Many-to-one relationship with Staff table

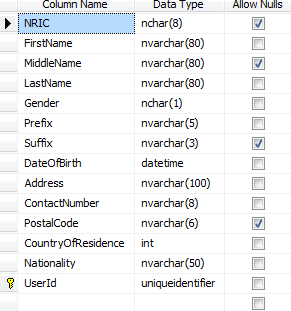


Figure 34 – UserParticulars

The UserParticulars table has:

* One-to-one relationship with aspnet\_Users table

This table also has the following characteristics:

* Due to the probability that some patients are tourists, the NRIC and PostalCode columns are nullable
* The ContactNumber uses the regular expression, ^[3689]\d{7}$ , as its check constraint
* The PostalCode uses the regular expression, \d{6}$, as it check constraint

# Web Application Modules

// TODO: Write this portion after code freeze

# Experimental Systems and Results

// TODO: Write this portion after code freeze and unit testing

# Discussion and Conclusion

## 6.1 Summary

// TODO: Summarize what was done, how long it took, challenges

## 6.2 Discussion

// TODO: Talk about the expected benefits of RIS

## 6.3 Future Work

// TODO: Discuss about MVC, Spring etc.

# References

|  |  |
| --- | --- |
| [1] | D. Evans, “A Very Basic DICOM Introduction,” dcm4che.org, 21 August 2006. [Online]. Available: http://www.dcm4che.org/confluence/display/d2/A+Very+Basic+DICOM+Introduction. [Accessed 16 October 2011]. |
| [2] | C. Rorden, "The DICOM Standard," [Online]. Available: http://www.cabiatl.com/mricro/dicom/index.html. [Accessed 10 December 2011]. |

# Appendix

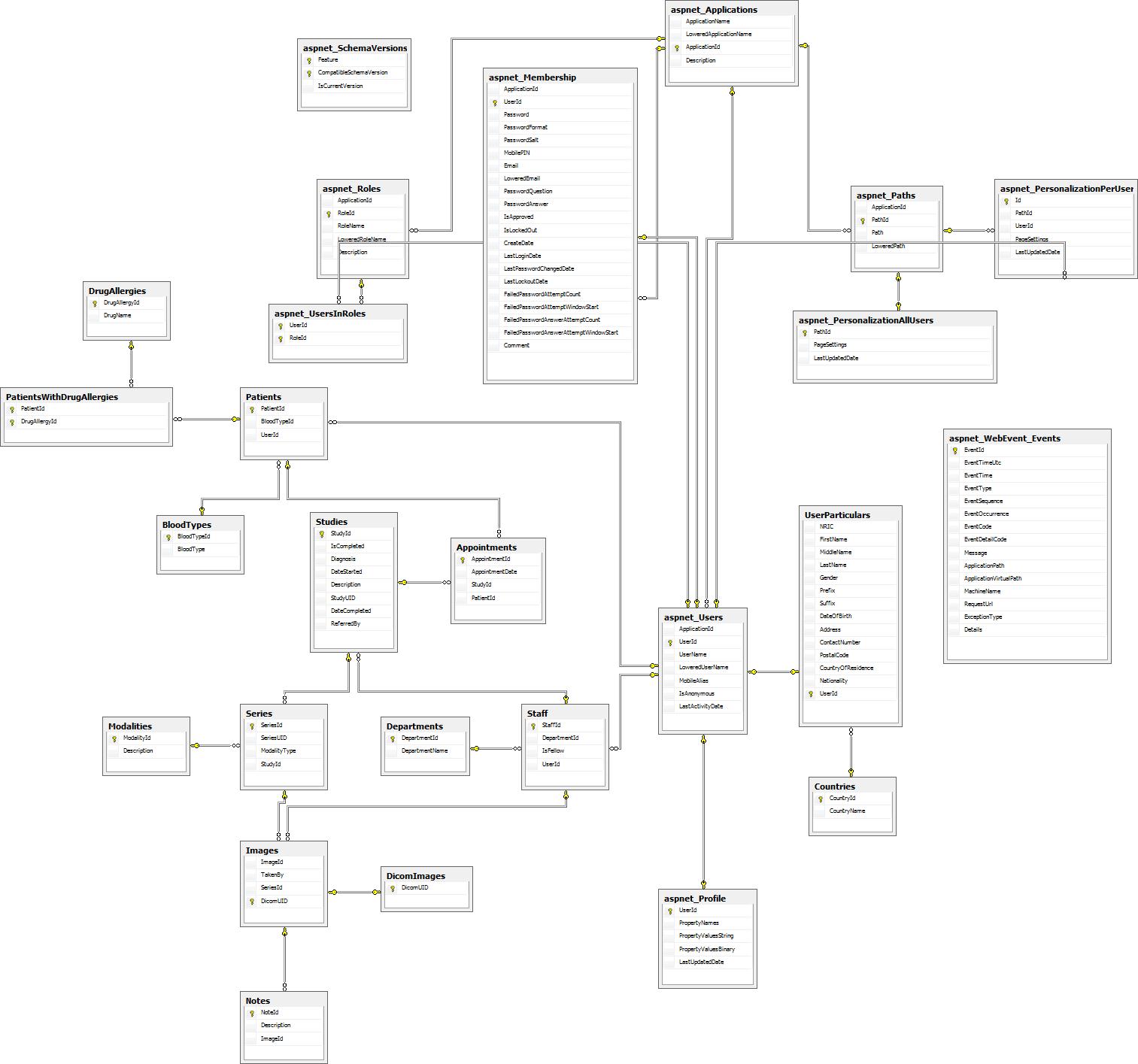


Figure 35 - Database Schema