Web-based Admission System
for
Physical Therapy Graduate Program

A Manuscript
Submitted to
the Department of Computer Science
and the Faculty of the
University of Wisconsin-La Crosse
La Crosse, Wisconsin

by
Hien Nguyen
in Partial Fulfillment of the
Requirements for the Degree of
Master of Software Engineering
July, 2008
Web-based Admission System for
Physical Therapy Graduate Program

By Hien Nguyen

We recommend acceptance of this manuscript in partial fulfillment of this candidate’s requirements for the degree of Master of Software Engineering in Computer Science. The candidate has completed the oral examination requirement of the capstone project for the degree.

____________________________________
Dr. Thomas Gendreau
Examination Committee Chairperson

____________________________________
Dr. Kenny Hunt
Examination Committee Member

____________________________________
Dr. Kasi Periyasamy
Examination Committee Member
ABSTRACT

Advisors: Dr. Thomas Gendreau, Dr. Kenny Hunt.

Web-based applications are no doubt replacing a paper-based process in most of higher education activities. Web-based applications can help to improve the reliability and to reduce the processing time and cost. In graduate admissions, the applicants are able to apply and retrieve the application status online. All of the application forms, fees, resumes, transcripts, GRE scores, and letters of recommendation are submitted electronically.

This project was initiated by Doctoral Program in Physical Therapy (DPT) at University of Wisconsin-La Crosse (UWL), in order to assist the admission office to manage the applications. The project objective is to build a web-based system that allows the applicants to submit their applications online. The recommenders for the applicants are also able to submit their recommendation via this system. Furthermore, the admission officers can manage the applications and export the needed reports.

This manuscript is written as a summary of the development process of the web-based application mentioned above. It also includes the challenges and fun that I have encountered during this project. Limitations and possible improvements are also mentioned.
ACKNOWLEDGEMENTS

This capstone project cannot be done on my own. I’m indebted to a lot of people who have assisted and encouraged me during the project development.

I would like to show my gratitude to Dr. John Greany from Physical Therapy Program, UW-La Crosse, who initiated this project and selected me as the student developer.

I also wish to thank Dr. Thomas Gendreau and Dr. Kenny Hunt for their thoughtful advising and invaluable comments.

Michael Taylor from ITS, who is the database administrator, has provided me great help on technical problems. Thank you, Mike.

I’m also thankful to the faculties and staff at Computer Science Department, UW-La Crosse: Dr. Kasi Periyasamy, Dr. Steven Senger, Dr. Thomas Gendreau, Dr. Kenny Hunt, Dr. Mark Headington, Dr. David Riley, Mrs. Becky Yoshizumi, and many others for inspiring me and showing me new horizons. Thanks to them, I’m determined to pursue an educational career.

Another special thanks I would like to say to Vietnam Ministry of Education and Training for financially supporting my graduate study.

Finally, I want to thank my family and all my friends for their love and patience. Without them, I cannot become what I am today.
# Table of Contents

ABSTRACT .......................................................................................................................... iii
ACKNOWLEDGEMENTS .................................................................................................... iv
TABLE OF CONTENTS ....................................................................................................... v
LIST OF FIGURES ............................................................................................................. vi
GLOSSARY .......................................................................................................................... viii
1. INTRODUCTION .............................................................................................................. 1
2. DEVELOPMENT MODELS FOR DPT ADMISSION SYSTEM ....................................... 2
3. THE ADMISSION SYSTEM DEVELOPMENT .................................................................. 5
4. THE ADMISSION SYSTEM WEB PAGES ....................................................................... 23
5. LIMITATIONS & FUTURE WORK .................................................................................. 50
6. CONCLUSIONS .............................................................................................................. 51
7. BIBLIOGRAPHY ............................................................................................................. 52
APPENDIX A. ..................................................................................................................... 53
APPENDIX B. ..................................................................................................................... 53
APPENDIX C. ..................................................................................................................... 53
APPENDIX D. ..................................................................................................................... 53
LIST OF FIGURES

Figure 1. Rapid prototyping model ................................................................. 2
Figure 2. The incremental prototyping model .................................................. 3
Figure 3. The waterfall model ........................................................................ 4
Figure 4. Software lifecycle model that was used in this project ...................... 4
Figure 5. User groups ..................................................................................... 5
Figure 6. State transitions of admission applications ..................................... 6
Figure 7. Web skin using UW-L templates ..................................................... 6
Figure 8. Entity-Relationship Diagram for Applicants .................................. 8
Figure 9. MVC architecture .......................................................................... 9
Figure 10. System layers .............................................................................. 9
Figure 11. Diagram of core classes ............................................................... 11
Figure 12. Toplink Map Figure 13. Toplink classes ........................................ 11
Figure 14. PTPublicFacadeBean.java .......................................................... 12
Figure 15. PTPublicFacadeLocal.xml .......................................................... 13
Figure 16. Data control methods .................................................................. 13
Figure 17. JSPX pages Figure 18. Java beans Figure 19. Page definitions ........ 14
Figure 20. Page transitions for applicants .................................................. 14
Figure 21. Unit testing for an applicant’s operations .................................... 15
Figure 22. ERD for recommenders .............................................................. 16
Figure 23. Page transition for references ................................................... 16
Figure 24. Unit testing for a reference’s operations ..................................... 16
Figure 25. Ptuser table ................................................................................. 17
Figure 26. PTDELETEDATE table and its relationship with PTCANDIDATE ... 17
Figure 27. Pages transitions for administrators .......................................... 18
Figure 28. Entity Relationship Diagram ..................................................... 19
Figure 29. Database-driven Class Diagram ............................................... 20
Figure 30. Login form for admission staff ................................................... 23
Figure 31. Change password form .............................................................. 24
Figure 32. Managing applications (initially) .................................................. 24
Figure 33. Managing applications (filtered) ................................................ 25
Figure 34. Reset an applicant’s password ................................................... 25
Figure 35. Email to inform password reset ................................................ 26
Figure 36. Retrieve an application .............................................................. 26
Figure 37. Email to inform applicant’s updates .......................................... 27
Figure 38. Send email to a recommender ................................................... 27
Figure 39. The content of the email to the recommender ................................ 28
Figure 40. General Information Form ......................................................... 28
Figure 41. Academic History Form .............................................................. 29
Figure 42. Employment History Form .......................................................... 29
Figure 43. Standardized Tests Form .............................................................. 30
Figure 44. Personal Narratives and Disclosure Statements ........................... 30
GLOSSARY

Applicant
A prospective student who creates a profile on the PT admission website to apply for the admission to the PT Doctoral Program.

Application
The collection of all required materials for an applicant to be considered for admission.

ADF
Oracle Application Development Framework (ADF) is an innovative Java development framework for creating enterprise applications.

EAR
Enterprise Archive (EAR) is a file format for packaging application modules so that they are deployed consistently.

EJB
Enterprise JavaBeans (EJB) is a server-side component which encapsulates the business logic of the application.

GRE
Graduate Record Examinations (GRE) is a test supplied by Educational Testing Service. It is required by almost all graduate programs in the U.S.

HTML
Hypertext Markup Language (HTML) is a language for creating web pages.
ITS

Information Technology Services (ITS) is a department in UW-La Crosse. They are responsible for providing, maintaining the web server and database server for this project.

J2EE

Java Platform, Enterprise Edition (J2EE) is an environment for developing multi-tier Java-based applications.

JDeveloper

JDeveloper is a free Integrated Development Environment from Oracle.

JSF

JavaServer Faces (JSF) is a Java web development framework. It simplifies the user interfaces building in Java-based web applications.

JSP

JavaServer Pages (JSP) is a technology that allows HTML or XML code to be dynamically generated in response to web client requests.

JSPX

JSP document, which is an XML representation of a JSP page.

MSE

Master of Software Engineering (MSE) is a graduate program in Computer Science Department, UW-La Crosse

MVC

Model-View-Controller (MVC) is a widely used software design approach.
ORACLE AS

Oracle Application Server (Oracle AS) is a platform for deploying J2EE applications.

POJO

Plain Old Java Objects (POJO) is a model technology that creates java objects from database tables

PT

Physical Therapy (PT), Department of Health Professions, UW-La Crosse

TOEFL

Test of English as a Foreign Language (TOEFL) is a test supplied by Educational Testing Service. It is required for international students.

TOPLINK

Oracle Toplink is a technology that provides Java object-to-relational persistence, enabling developers to access and persist relational data.

TSE

Test of Spoken English (TSE) is a test supplied by Educational Testing Service. It is required for international students.

XML

Extensible Markup Language (XML) is a specification language for creating markup languages.
1. INTRODUCTION

The current admission process of Doctoral Program in Physical Therapy (DPT) requires the applicants to download the forms from the DPT website, complete them by hand and return them via mail.

In Spring semester 2007, Dr. John Greany from DPT came to see Dr. Kasi Periyasamy, who is the MSE Director, to initiate a project to provide DPT admission office with a web-based admission system. I was, fortunately, selected to be the student developer for this project.

The admission system was supposed to perform the following core functionalities:

The applicants are able to complete the forms, submit the application, and retrieve the application status online. The forms can be modified several times before the applicant submits them.

Each applicant has to provide at least 02 supervising physical therapists as recommenders and at least 02 other personal references. The recommenders are able to complete the recommendation forms in this system.

All of the applications, recommendations are managed by the administrators. The administrators are able to retrieve the forms in pre-defined formats.

Beside these functional requirements, the system also needs to be user-friendly, reliable and secure.
2. DEVELOPMENT MODELS FOR DPT ADMISSION SYSTEM

Software development involves a great deal of activities to guarantee that the software product is delivered on time and is high-quality. The set of those activities and the strategy of how to perform them form the development model. According to R.S. Pressman in [4], there are a number of different models widely used: the linear sequential model, the prototyping model, the component assembly model, the formal method model, etc. Each model has its own advantages and challenges, depending on the nature of the application domain, time, cost, and resources. A particular project may apply more than one model to take their advantages. In this project, a few models have been selected to cope with the system development.

At the beginning, in order to gather functional requirements, I used rapid prototyping model. The admission system is supposed to be built from scratch; and the customer does not have a similar system in use. Therefore, they were not specific and clear what work needed to be completed. The rapid prototype was able to help both parties have a look and feel about the final product.

![Figure 1. Rapid prototyping model](image)

This prototyping model has completed its role of requirements gathering more than expectation. I learned more about the application domain functionalities, while the
customer learned more about web-related operations such as user authentication and authorization. Furthermore, we both agreed upon the user interface design used in this prototype.

When core functionalities became clear and specific with the first prototype, I started to build the second one. Then I faced another challenge: requirement changes. Functionalities are likely to be modified and added more and more on the way. To cope with the changes, I applied a different model that was more helpful: incremental prototyping.

![Diagram](image.png)

Figure 2. The incremental prototyping model
Courtesy of Dr. Kasi Periyasamy in [5].

With incremental prototyping model, the prototype would not be thrown away but reused with added or modified functionalities. During the development of a prototype, I integrated some phases from the waterfall model.

According to I. Sommerville in [3], the water model consists of 5 phases:
- requirements analysis and definition
- system and software design
- implementation and unit testing
- integration and system testing
- operation and maintenance.

The name waterfall originated from the idea that the phases are processed one after another.
Theoretically speaking, each phase has to wait until its preceding phase is completed. When any problems occur, the developers may return to the preceding phase to work out the solution. At one time, only one phase is active while the others are locked, e.g. staying unchanged.

But in this project, I integrated the waterfall model with the incremental prototyping model. To be specific, in each prototype, I applied four steps: requirements analysis, design, implementation and testing. Not every prototype used all four steps. Some of them needed only implementation and testing steps.

In brief, I used rapid prototyping model to gather functional requirements, then I used a hybrid model of incremental prototyping and waterfall to develop the system.
3. THE ADMISSION SYSTEM DEVELOPMENT

As stated above, the development models that I used was prototyping models. Therefore, I am going to reveal the prototypes one after another. Some prototypes may be more specific than others, especially the ones that run through the phases of waterfall model.

3.1 Prototype #1: rapid prototype

This prototype aimed to gather the functional requirements. The admission system was supposed to be built from scratch. There were no functional requirements specifications beforehand. Therefore, the system functionalities were learned through application domain, which is graduate admission procedure in this project. After analyzing a variety of forms and application materials, as well as discussing with the customer, there came the core functionalities as below.

There are 3 groups of users who will interact with the system: (i) the applicants, (ii) the references of the applicants, and (iii) the staff of admission office (also called administrators). There are two times of consideration for admission, one in November for early admissions and one in January for general admissions. The admission applications which fail the first time will be open for updating until the second deadline in January.

Figure 5. User groups
An applicant is able to create a new admission application, edit it as many times as he wants until he submits it. The staff of the admission office is able to write comments on the applications, and able to open them for the applicants to update.

![State transitions of admission applications](image)

**Figure 6. State transitions of admission applications**

Each applicant must have at least two physical therapists who agree to be his references. He must also provide two other people as personal references. Those references form a group of users. They are provided a web page to complete the respective recommendation forms.

From these core functionalities, I developed the rapid prototype with the graphical user interface (GUI) only. The prototype consisted of web pages using UW-L web templates; but having no interactions with the database. Actually, no database was designed by this time.

The color and style of the pages are in accordance to UW-La Crosse template: maroon header, dark green sidebar, and dark blue footer (Figure 7).

![Web skin using UW-L templates](image)

**Figure 7. Web skin using UW-L templates**
3.2 Prototype #2

Gaining the customer’s agreement on GUI specifications via the rapid prototype, I started to build another prototype using incremental prototyping. Taking the feature that the system is supposed to interact with 3 groups of users, I planned to develop the prototypes in a respective routine to user groups. The first prototype of the incremental model was to cover the applicants’ operations.

3.2.1 Database design

According to the application materials, an applicant has to provide a lot of information including personal data, courses grades and employment history. Initially, all personal data such as address, phone number, etc are put into a table named PTCANDIDATE which uses CANDID field as the primary key.

Each applicant may have attended many institutions. As a result, the institutions’ information is put into a separated table named PTINSTITUTION with a foreign key which references CANDID in the PTCANDIDATE table (Figure 8).

This idea was repeated for similar entities such as employments (PTEMPLOYMENT), courses (PTCOURSE), references (PTREFERENCE), and GRE scores (PTGRE). As for TOEFL and TSE scores, domestic applicants don’t need these score, while each international applicant is supposed to submit only one score. For the sake of space capacity, TOEFL and TSE scores are stored in a different table (PTTOEFL). The primary key of this new table is also the foreign key, which references the primary key of the PTCANDIDATE table.
3.2.2 MVC architecture

Model-View-Controller (MVC) is an architectural pattern that divides software into three components: the model, the view, and the controller, according to Christensen and Thayer in [2]. The model represents the working data and states of the system; the view represents the user interface; the controller receives the requests from the view and gives response, it also directs the model to change the application state. One of the greatest advantages of this pattern is that the user interface (the view) and the business logic (the model) can be modified independently.
In this particular admission system, I used the MVC architecture with a slight modification. The three components together with the database are depending on each other as layers (Figure 10).

![Figure 10. System layers](image)

The data model is database-driven. It consists of classes generated from database tables. Upon these classes, several functional methods are developed to form the data control layer. The data control receives the events from the user interface and makes the data model operate. The user interface layer is the set of web pages that users directly work with.

3.2.3. Implementation

The system was planned to deploy on a server maintained by UW-L Information Technology Services (ITS). Therefore, in this project, the choice of programming languages and technologies was affected by ITS’s policy. After meetings with ITS server administrator, we came to a few options for MVC pattern as in Table 1.
<table>
<thead>
<tr>
<th>Model</th>
<th>Controller</th>
<th>View</th>
</tr>
</thead>
<tbody>
<tr>
<td>ADF BC</td>
<td>ADF BC</td>
<td>ADF Faces</td>
</tr>
<tr>
<td>Toplink</td>
<td>EJB</td>
<td>Struts</td>
</tr>
<tr>
<td></td>
<td></td>
<td>JSF, JSP</td>
</tr>
</tbody>
</table>

Table 1. Model View Controller implementation options

ADF: Oracle Application Development Framework
BC: Business Components
JSF: JavaServer Faces
EJB: Enterprise JavaBean
JSP: JavaServer Pages

Oracle Application Development Framework (ADF) is a development framework created by Oracle for Java applications. It was designed to highly support MVC pattern with its components (ADF BC and ADF Faces). It’s also integrated into JDeveloper.

Finally, the technologies I chose to use are TopLink, EJB together with ADF Faces.; and the IDE (Integrated Development Environment) was JDeveloper 10g. TopLink is a very powerful mapping package to converts relational databases to classes and vice versa. TopLink in this project is in charge of creating the data model from the database as well as processing queries. It maintains a mapping between the database and the data model which physically consists of java and xml files. Each table in the database is mapped to a Toplink descriptor and a class in the data model (physically a java file). Two classes are associated only if two respective tables have relationship between.

---

3.2.3.1 The data model layer

The data model was created by using Toplink technology. Toplink creates a map that maps database tables to Toplink descriptors (Figure 12). Based on these descriptors, we can write several needed queries by creating Toplink Named Queries.

Oracle Toplink also creates several java classes and specification files for all descriptors (Figure 13). The properties of a class are the fields of the respective descriptor.
3.2.3.3 The data control layer

This layer was implemented by using EJB Session Bean. The Session Bean encapsulates the model logic and provides the interface for the data control to access without concern about the model complexity. The Session Bean includes two files:

- PTPublicFacadeBean.java: containing the bean code
- PTPublicFacadeLocal.xml: containing the local interface.

The following figures partly reveal these two files.

```java
package oracle.ptedmission.model;

import ...

@Stateless(name="PTPublicFacade")
public class PTPublicFacadeBean implements PTPublicFacade,
    PTPublicFacadeLocal {

    private SessionFactory sessionFactory;

    public PTPublicFacadeBean() {
        this.sessionFactory =
            new SessionFactory("META-INF/sessions.xml", "PTAdmissionSession");
    }

    public Object mergeEntity(Object entity) {
        UnitOfWork uow = getSessionFactory().acquireUnitOfWork();
        Object workingCopy = uow.readObject(entity);
        if (workingCopy == null)
            throw new RuntimeException("Could not find entity to update");
        uow.deepMergeClone(entity);
        uow.commit();

        return workingCopy;
    }
}
```

Figure 14. PTPublicFacadeBean.java

---

JDeveloper supplies a function which allows the developers to create data control from the session bean. All the needed queries were converted to Java code in PTPublicFacadeBean.java, and to access methods stored in PTPublicFacadeLocal.xml.
3.2.3.4 The user interface layer

This layer was implemented based on Oracle ADF Faces [7] with JSF components. It consists of three categories of files. The first category is the JSP documents, which contain web pages elements such as tables, texts, combo boxes, etc (Figure 18). The second category is the java beans, which contain the java code (Figure 19). The last category is page definitions in XML representation, which contain the connectors between the data controls and the web pages (Figure 20).

When an event was activated in a JSPX page, the data controls may tell the HTTP server to redirect to another JSPX page. Those transitions are described in Figure 21.

![Diagram](image)
An applicant must register an account at /cand/register.jspx. After his account is created, he can log in the system and view his admission application status at /cand/appstatus.jspx. The page /cand/forms.jspx contains the forms that he needs to fill out. When the applicant submits the application, the server will redirect to /cand/submit.jspx to process.

3.2.3.5 Testing
In this prototype, I performed two kinds of testing: unit testing and functional testing. For unit testing, I integrated JUnit to JDeveloper and put testing code into a package. For functional testing, the black box testing technique was used based upon the functional requirements specification. And of course, only applicants’ operations were tested.

3.3 Prototype #3
This prototype focused on the operations of recommenders. There are two different kinds of recommenders: physical therapists and personal references. Each type of recommender uses a different recommendation form. Personal references have to give comments about certain characteristics such as maturity, emotional stability, communication skills, critical thinking and leadership skills. Meanwhile, physical therapists have to rate the applicant’s qualifications by numerical values. The admission office needs the average value of these rates. Therefore, I have to redesign some of the database tables and relationships.

3.3.1 Database design
I splitted the PTREFERENCE table to two. One table still held the same name to store personal references data, the other one named PTTHERAPISTS and stored the therapists data. Besides, two additional tables were created to support the rating of physical therapists: PTEVALUATIONFORMQUEST to store the questions in the recommendation form; and PTEVALUATION to store the rates. (Figure)
3.3.2 Implementation

The Toplink worked smoothly with the incremental prototyping model. At this phase, I only needed to extend the data model, then created data controls for the extended part. Amazingly, I can do it in a few minutes without writing any lines of code. The only portion that I had to write codes for was the user interface layer.

The recommenders do not have much to do with the system. They can only log in, fill the recommendation and log out (Figure 23). The account information was sent via email by one of the operations of the administrator.

3.3.3 Testing

This prototype also needed unit testing. Similar to the previous prototype, I used JUnit to create the testing code. Besides, I did the functional testing for a recommender’s operations with black box testing upon the requirements specification. The functional
testing for an applicant’s operations was also performed again to make sure the added portion of this prototype had not affected the implemented operations.

3.4 Prototype #4
This prototype came with the functionalities that an administrator can do. Basically, an administrator is able to retrieve all the information of applicants and recommenders. Fortunately, no additional relational tables are needed for this retrieval operations. But there are a few more operations that a root administrator has to take care of. A root administrator account is an user account that can not be deleted. He or she can manage the deadlines of the applications and to mange other administrator accounts.

<table>
<thead>
<tr>
<th>PTUSER</th>
</tr>
</thead>
<tbody>
<tr>
<td>USERD : NUMBER</td>
</tr>
<tr>
<td>EMAIL : VARCHAR2(50 CHAR)</td>
</tr>
<tr>
<td>PASS/WORD : VARCHAR2(50 CHAR)</td>
</tr>
<tr>
<td>NAME : VARCHAR2(50 CHAR)</td>
</tr>
<tr>
<td>TYPE : VARCHAR2(20 CHAR)</td>
</tr>
</tbody>
</table>

Figure 25. Ptuser table

The administrator accounts are stored in a separate table named PTUSER (Figure 25).

As for the deadlines, they come with the semester and year of entry. The table PTDUEDATE stores these information. Each applicant has to select which semester and year that he wants to get into the PT Program. For each semester (and year), there may be hundreds of applicants. That’s why the table PTDUEDATE came into being together with a relationship with table PTCANDIDATE (Figure 26).

<table>
<thead>
<tr>
<th>PTDUEDATE</th>
</tr>
</thead>
<tbody>
<tr>
<td>DUEID : NUMBER</td>
</tr>
<tr>
<td>SEMESTER : VARCHAR2(20 CHAR)</td>
</tr>
<tr>
<td>YEAR : VARCHAR2(4 CHAR)</td>
</tr>
<tr>
<td>DUEDATE : VARCHAR2(20)</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>PTCANDIDATE</th>
</tr>
</thead>
<tbody>
<tr>
<td>CANDID : NUMBER</td>
</tr>
<tr>
<td>EMAIL : VARCHAR2(50 CHAR)</td>
</tr>
<tr>
<td>PASSWORD : VARCHAR2(200 CHAR)</td>
</tr>
<tr>
<td>FIRSTNAME : VARCHAR2(50 CHAR)</td>
</tr>
<tr>
<td>LASTNAME : VARCHAR2(50 CHAR)</td>
</tr>
</tbody>
</table>

Figure 26. PTDUEDATE table and its relationship with PTCANDIDATE
An administrator has to log in the system to perform his privileged operations. Figure 29 shows the transitions between the pages that the administrator can access to perform his operations. The Main page contains a list of all applications in the database and enables the administrator to filter the data based on semester and year, application status, and a few more. When he selects to process an individual application, the server will redirect to Candidates’ applications page. The Deadline page help the administrator to manage the entrance semesters and years.

![Diagram of pages transitions for administrators]

Figure 27. Pages transitions for administrators

Up to this point, the database design has been completed, taking all types of users into account. Figure 28 shows the entities and relationships, then Figure 29 shows the class diagram driven by the database design.
Based on the database, the Toplink will create the data model in Java classes and make associations between them based on the relationships between the tables.
Figure 29. Database-driven Class Diagram
Prototype #4 (including administrators’ tasks) witnessed a huge effort of testing, especially integrated testing for the tasks that affect other types of users. Those tasks are:

◊ Initiating the recommendation forms for recommenders and sending them emails to inform how to log in the system.
◊ Changing the application status and giving feedback to applicants.

The integrated testing aims at making sure the applicants and recommenders are able to interact with the system correctly when the administrators performs these tasks.

3.5 Prototype #5

This last prototype added the reporting and printing functionalities. In brief, each type of users is able to view applications and/or recommendations in PDF format and to print it out. There was not any modifications to the database design or architectural design. And fortunately, there are some free reporting tools for Java applications. The most popular one thus far is JasperReport from Jasper Soft. I integrated JasperReport to JDeveloper to make PDF files. Generally, it reads an XML file, compile it to a Jasper file, then generate a PDF file from the jasper file. Therefore, all the reports are design in XML formats. Jasper Soft also provides a tool to design reports in XML files. That’s iReport. In fact, I used iReport 3.0.0 to design the PDF format for applications and recommendations and stored them in XML files. The JasperReport would be in charge of generating the PDF when users trigger report buttons.

This prototype also came with an addition testing: installation testing. I tried to simulate the real web server by installing Oracle Application Server 10g to a PC in MSE Lab. After that, I created the deployment profile for this prototype and deployed it to the installed simulating web server.

3.6 System security

It is impossible to make a system absolutely secure. In this system, I employed a few security strategies to make it acceptably secure.
Firstly, every user must be authenticated by log-in mechanism. The user has to provide the username (email) and the password to log in. The password was encrypted in the database, so even database administrator can not reveal the password. Actually, the password encoding algorithm was SHA (Secure Hash Algorithm). It is a built-in algorithm in java.security.MessageDigest class. In addition, a session timeout is assigned to a working session. If the user forgets to log out, the system will log him out when the timeout expires.

Secondly, each user is authorised before granted access to a page. An applicant can not see a recommender’s pages, for instance. Another example is that only the root administrator can create and modify the administrator accounts.

Thirdly, in an attempt to prevent SQL injection, most of the input texts are validated before going to the server. Using email as the username is one step to avoid SQL injection in login pages.

### 3.7 Some metrics

The data model includes 15 classes which are generated from the database tables. There are around 100 lines of code (LOC) in average in each class.

The data controller contains only 1 class (EJB session bean). Its size is over 950 LOC.

The view component includes 21 JSF pages for all types of users, 33 backing beans (15 beans for 15 core classes plus supporting beans). Each bean is over 600 LOC in average.

The number of test cases generated by JUnit is around 70.
4. THE ADMISSION SYSTEM WEB PAGES

This section discusses the user interface. There are a few common features that users need to consider before perform any specific tasks:

✓ All web pages except for login pages are required logging in to access. If they are directly visited without logging in, the system will redirect to respective login pages.
✓ The system uses the UW-La Crosse mail server to send emails. Sometimes, there may be a few hours delay in sending emails.
✓ In any forms (login, search…) the users are supposed to click the buttons rather than to press Enter.
✓ All tables are able to be sorted by columns (either ascending or descending). Just click the columns’ headers.
✓ In any cases that navigational links are available, users should use them rather than using browser’s back button.
✓ * symbol means required fields
✓ In searching forms, use % symbol as the wildcard. For instance, to search for names that begin with letter J, type J%

4.1 Operations of the admission staff

4.1.1 Logging in

URL : /faces/staff/login.jspx

The admission officers are supposed to provide username (email) and password to log in the system. After providing email and password, the Login button must be clicked. The
login forms for applicants and recommenders look and operate in the same way with this one.

### 4.1.2. Changing password
URL: /faces/staff/changepass.jspx

![Change Password Form](image)

Figure 31. Change password form

In this change password form, there are a number of rules that the texts must be followed:

- The current password must match the password currently stored in the database.
- The new password must be at least 6 characters. It may or may not be the same with the current one.
- The confirm password must be the same with the new password. It is to make sure there is no typo in the new password.

For applicants and recommenders, the change password forms are the same.

### 4.1.3 Managing applications
URL: /faces/staff/main.jspx

![Managing Applications](image)

Figure 32. Managing applications (initially)
This page is loaded when the user successfully logs in. Initially, no applications are displayed. The user has to select the semester of entry, then click Filter button. There are more criteria for filtering, including email, name, application status.

After clicking Filter, several rows of applicants will be displayed. The user then can do two operations with a selected applicant: view/edit his application and reset his password.

**4.1.4 Reset an applicant’s password**

Step 1: retrieve that applicant in the managing applications page: /faces/staff/main.jsp

Step 2: select the applicant from the table

Step 3: click Reset Password button. A textbox and a Save button will appear.

Step 4: type the new password in the textbox. The minimum number of characters is 6.

Step 5: click Save

![Figure 33. Managing applications (filtered)](image1)

![Figure 34. Reset an applicant’s password](image2)
After resetting the password, the system will automatically send an email to the applicant to inform him.

![Email to inform password reset](image)

**Figure 35. Email to inform password reset**

### 4.1.5. Process an application

Repeat step 1 and step 2 in previous section of Resetting password. Then click View Detail / Edit button.

![Retrieve an application](image)

**Figure 36. Retrieve an application**
From this page, beside the capability of retrieving all of the applicant’s information, the admission staff is also able to perform the following operations:

- View the application in PDF Format: simply click the PDF Format button.
- Change the application status and write some comments about it. Click Save to finish. The system will send an email to the applicant to inform about the update.

<table>
<thead>
<tr>
<th>Subject: Notification from Admission Office, Physical Therapy Program in UW-La Crosse</th>
</tr>
</thead>
<tbody>
<tr>
<td>Dear Applicant</td>
</tr>
<tr>
<td>Your application into Physical Therapy Program, UW-La Crosse has some updates. Please login to our system to see those updates. (Copy and paste into your web browser) Application updates link: <a href="http://138.19.196.201:8888/pt/faces/cand/main.jspx">http://138.19.196.201:8888/pt/faces/cand/main.jspx</a></td>
</tr>
<tr>
<td>Regards</td>
</tr>
<tr>
<td>Physical Therapy Program - University of Wisconsin La Crosse</td>
</tr>
</tbody>
</table>

Figure 37. Email to inform application's updates

- Send an email to the respective supervising PT or personal recommenders. The list of recommenders may be changed by the applicant if he has not submitted the application yet. Therefore, the admission staff should check the status Submitted before sending email. By clicking Send Email button, the user triggers the system to run the following tasks:

- Dynamically create a password for the recommender.
- Initialize the respective recommendation form.
- Send an email to the recommender to inform him/her about the applicant and his/her account.

Figure 38. Send email to a recommender
You have been asked to give an evaluation/recommendation for an applicant to our Physical Therapy Program. 
Applicant Name: Hien Nguyen
Semester of Entry: Fall 2009. Deadline: 01/15/2009

Please visit the link below, use the following username and password to log in.
(Copy and paste into your web browser)
Recommendation Form Link: http://138.49.196.201:8888/pt/faces/personalreflogin.jspx

Username: kasi@geemail.com
Password: VrezBuKISUTxB

Regards
Physical Therapy Program - University of Wisconsin La Crosse

Figure 39. The content of the email to the recommender

After sending email, the system will disable the Send Email button, to make sure the password and the form won’t be replaced. At the same time, the Recommendation button will be enabled. This button will lead to a PDF Format of the recommendation form.

Also on this page, the admission staff is able to view all other parts of the application.

<table>
<thead>
<tr>
<th>General Information</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Email</strong></td>
</tr>
<tr>
<td><strong>First Name</strong></td>
</tr>
<tr>
<td><strong>Middle Initial</strong></td>
</tr>
<tr>
<td><strong>Last Name</strong></td>
</tr>
<tr>
<td><strong>Address</strong></td>
</tr>
<tr>
<td><strong>City</strong></td>
</tr>
<tr>
<td><strong>State</strong></td>
</tr>
<tr>
<td><strong>Zip</strong></td>
</tr>
<tr>
<td><strong>Phone - Daytime</strong></td>
</tr>
<tr>
<td><strong>Country</strong></td>
</tr>
<tr>
<td><strong>Phone - Evening</strong></td>
</tr>
<tr>
<td><strong>State Of Residence</strong></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Permanent Mailing Address</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Address</strong></td>
</tr>
<tr>
<td><strong>City</strong></td>
</tr>
<tr>
<td><strong>State</strong></td>
</tr>
<tr>
<td><strong>Zip</strong></td>
</tr>
<tr>
<td><strong>Country</strong></td>
</tr>
<tr>
<td><strong>Phone</strong></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>International</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Yes</strong></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Date of Birth</th>
</tr>
</thead>
<tbody>
<tr>
<td>09/17/1979</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Ethnic / Racial Heritage</th>
</tr>
</thead>
<tbody>
<tr>
<td>Vietnamese</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Gender</th>
</tr>
</thead>
<tbody>
<tr>
<td>Male</td>
</tr>
</tbody>
</table>

Figure 40. General Information Form
### Academic History

#### Institutions

<table>
<thead>
<tr>
<th>Institution</th>
<th>Location</th>
<th>Degree</th>
<th>Major</th>
<th>From (mm/yyyy)</th>
<th>To (mm/yyyy)</th>
<th>GPA</th>
<th>Credits earned</th>
</tr>
</thead>
<tbody>
<tr>
<td>Univ of Wisconsin-La Crosse</td>
<td>La Crosse, WI</td>
<td>Master</td>
<td>Software Engineering</td>
<td>09/2006</td>
<td>08/2008</td>
<td>3.6</td>
<td>48</td>
</tr>
<tr>
<td>Univ of Technology</td>
<td>Saigon, Vietnam</td>
<td>Bachelor</td>
<td>Computer Engineering</td>
<td>09/1997</td>
<td>05/2002</td>
<td>3</td>
<td>120</td>
</tr>
</tbody>
</table>

#### Prerequisite courses

<table>
<thead>
<tr>
<th>Course Name</th>
<th>Course Title &amp; Department</th>
<th>Semester</th>
<th>Year</th>
<th>Credit Hours</th>
<th>Grade</th>
<th>Indicate UWL or other institution</th>
</tr>
</thead>
<tbody>
<tr>
<td>General Biology</td>
<td></td>
<td>Spring</td>
<td>2007</td>
<td>4</td>
<td>4</td>
<td>UWL</td>
</tr>
<tr>
<td>Chemistry I</td>
<td></td>
<td>Fall</td>
<td>2007</td>
<td>4</td>
<td>3.5</td>
<td>UWL</td>
</tr>
<tr>
<td>Chemistry II</td>
<td></td>
<td>Spring</td>
<td>2008</td>
<td>4</td>
<td>3</td>
<td>UWL</td>
</tr>
<tr>
<td>Human Anatomy / Physiology I</td>
<td></td>
<td>Spring</td>
<td>2008</td>
<td>4</td>
<td>4</td>
<td>UWL</td>
</tr>
<tr>
<td>Human Anatomy / Physiology II</td>
<td></td>
<td>Fall</td>
<td>2007</td>
<td>3</td>
<td>4</td>
<td>UWL</td>
</tr>
<tr>
<td>Physics I</td>
<td></td>
<td>Spring</td>
<td>2007</td>
<td>3</td>
<td>3</td>
<td>UWL</td>
</tr>
<tr>
<td>Physics II</td>
<td></td>
<td>Spring</td>
<td>2008</td>
<td>4</td>
<td>3</td>
<td>UWL</td>
</tr>
<tr>
<td>Statistics</td>
<td></td>
<td>Fall</td>
<td>2006</td>
<td>3</td>
<td>3</td>
<td>UWL</td>
</tr>
<tr>
<td>Psychology</td>
<td></td>
<td>Spring</td>
<td>2007</td>
<td>3</td>
<td>3.5</td>
<td>UWL</td>
</tr>
<tr>
<td>Sociology</td>
<td></td>
<td></td>
<td></td>
<td>0</td>
<td>0</td>
<td></td>
</tr>
</tbody>
</table>

GPA of prerequisite courses: 3.45

#### Courses in last 2 semesters

<table>
<thead>
<tr>
<th>Course Name</th>
<th>Course Title &amp; Department</th>
<th>Year</th>
<th>Semester</th>
<th>Credit Hours</th>
<th>Grade</th>
<th>Indicate UWL or other institution</th>
</tr>
</thead>
<tbody>
<tr>
<td>Chemistry</td>
<td></td>
<td>2008</td>
<td>Spring</td>
<td>4</td>
<td>4</td>
<td>UWL</td>
</tr>
<tr>
<td>Physics I</td>
<td></td>
<td>2008</td>
<td>Spring</td>
<td>4</td>
<td>3</td>
<td>UWL</td>
</tr>
</tbody>
</table>

GPA of last 2 semesters courses: 3.5

#### In-progress courses

<table>
<thead>
<tr>
<th>Course Name</th>
<th>Course Title &amp; Department</th>
<th>Semester</th>
<th>Year</th>
<th>Credit Hours</th>
<th>Indicate UWL or other Institution</th>
</tr>
</thead>
<tbody>
<tr>
<td>Psychology</td>
<td></td>
<td>Fall</td>
<td>2008</td>
<td>4</td>
<td>UWL</td>
</tr>
<tr>
<td>Sociology</td>
<td></td>
<td>Fall</td>
<td>2008</td>
<td>4</td>
<td>UWL</td>
</tr>
<tr>
<td>Sample course</td>
<td></td>
<td>Spring</td>
<td>2008</td>
<td>4</td>
<td>UWL</td>
</tr>
</tbody>
</table>

Figure 41. Academic History Form

### Employment History

<table>
<thead>
<tr>
<th>Occupation</th>
<th>Start Date</th>
<th>End Date</th>
</tr>
</thead>
<tbody>
<tr>
<td>UWL</td>
<td>05/2007</td>
<td>06/2008</td>
</tr>
<tr>
<td>Educational Technologies</td>
<td>07/2007</td>
<td>09/2008</td>
</tr>
</tbody>
</table>

Figure 42. Employment History Form
GRE and TOEFL

**GRE**

<table>
<thead>
<tr>
<th>Date Taken</th>
<th>Verbal</th>
<th>Quantitative</th>
<th>Analytical</th>
</tr>
</thead>
<tbody>
<tr>
<td>10/2005</td>
<td>500</td>
<td>800</td>
<td>5</td>
</tr>
<tr>
<td>04/2007</td>
<td>550</td>
<td>790</td>
<td>6</td>
</tr>
</tbody>
</table>

**TOEFL (for international applicants)**

- Date: 10/2005
- Score: 600
- Type: PBT

**TSE (for international applicants)**

- Date: 
- Score: 

Figure 43. Standardized Tests Form

---

**Personal Narratives and Disclosure Statement**

<table>
<thead>
<tr>
<th>#Questions</th>
<th>Answer</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. Please consider the characteristics of your background including your</td>
<td>answer for personal narrative #1</td>
</tr>
<tr>
<td>previous experiences and describe how these characteristics might allow</td>
<td></td>
</tr>
<tr>
<td>you to contribute to a diverse learning environment or a diverse Physical</td>
<td></td>
</tr>
<tr>
<td>Therapy program. (Characteristics of your background might include, for</td>
<td></td>
</tr>
<tr>
<td>example, socioeconomic background, personal international travel, family</td>
<td></td>
</tr>
<tr>
<td>make up or size, cultural background. Please note that the characteristics</td>
<td></td>
</tr>
<tr>
<td>of your narrative background are not limited to the above examples but,</td>
<td></td>
</tr>
<tr>
<td>rather, should include any characteristics and past experiences that you</td>
<td></td>
</tr>
<tr>
<td>feel may influence you as a Physical Therapy or Physical Therapy student.</td>
<td></td>
</tr>
<tr>
<td>2. Please tell us about a time in your life when you experienced adversity.</td>
<td></td>
</tr>
<tr>
<td>Was this situation stressful? Why or Why not? How did you handle the</td>
<td></td>
</tr>
<tr>
<td>situation and what did you learn about yourself from this experience?</td>
<td></td>
</tr>
<tr>
<td>3. Is there any additional information that you would like the admissions</td>
<td></td>
</tr>
<tr>
<td>committee to know about you that has not already been shared in this</td>
<td></td>
</tr>
<tr>
<td>application?</td>
<td></td>
</tr>
<tr>
<td>4. Have you ever been admitted to, then withdrawn from, or been asked to</td>
<td>No</td>
</tr>
<tr>
<td>withdraw from, or been dropped from, a clinical program for other than</td>
<td></td>
</tr>
<tr>
<td>academic reasons?</td>
<td></td>
</tr>
<tr>
<td>5. Has licensure been denied to you, or been revoked from you, in any state</td>
<td>No</td>
</tr>
<tr>
<td>in the United States for reasons other than insufficient credits or</td>
<td></td>
</tr>
<tr>
<td>courses?</td>
<td></td>
</tr>
<tr>
<td>6. Have you ever been charged with, convicted of, pleaded guilty or no</td>
<td>No</td>
</tr>
<tr>
<td>contest to, or forfeited bail for any criminal conduct under law or</td>
<td></td>
</tr>
<tr>
<td>ordinance, excluding only minor traffic violations? PLEASE NOTE: The</td>
<td></td>
</tr>
<tr>
<td>existence of criminal charges or a criminal record or denial of</td>
<td></td>
</tr>
<tr>
<td>revocation of a license does not constitute an automatic bar to</td>
<td></td>
</tr>
<tr>
<td>admission and will be considered only as they substantially relate to</td>
<td></td>
</tr>
<tr>
<td>the duties and responsibilities of the program and eventual</td>
<td></td>
</tr>
<tr>
<td>licensure.</td>
<td></td>
</tr>
<tr>
<td>7. Have you ever been suspended, expelled, placed on probation, or</td>
<td>No</td>
</tr>
<tr>
<td>otherwise disciplined by any college or university, or from any</td>
<td></td>
</tr>
<tr>
<td>program of a college or university other than for academic reasons?</td>
<td></td>
</tr>
<tr>
<td>8. Is there anything that would prohibit you from performing essential</td>
<td></td>
</tr>
<tr>
<td>job-related functions, or functions related to No</td>
<td></td>
</tr>
<tr>
<td>your responsibilities as a student in the program to which you are</td>
<td></td>
</tr>
<tr>
<td>applying? Please refer to Part III of application packet entitled</td>
<td></td>
</tr>
<tr>
<td>Technical Standards of Performance for Applicants.</td>
<td></td>
</tr>
<tr>
<td>9. Additional explanation</td>
<td></td>
</tr>
</tbody>
</table>

Figure 44. Personal Narratives and Disclosure Statements
4.1.6 Managing Supervising Physical Therapists

URL: /faces/staff/therapists.jspx

Managing Supervising Physical Therapists

PT Name

PT Email

Filter

Select and (Send Email) Recommendation Reset Password

<table>
<thead>
<tr>
<th>Select</th>
<th>PT Name</th>
<th>PT Email</th>
<th>Licensure</th>
<th>Facility Name</th>
<th>Facility Location</th>
<th>Applicant Name</th>
<th>Applicant Email</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Sample PT</td>
<td><a href="mailto:hoanghien@gmail.com">hoanghien@gmail.com</a></td>
<td></td>
<td></td>
<td></td>
<td>Hien Nguyen</td>
<td><a href="mailto:nguyen.hien@students.wlax.edu">nguyen.hien@students.wlax.edu</a></td>
</tr>
<tr>
<td></td>
<td>Dr. Kenny Hunt</td>
<td><a href="mailto:hh@gmail.com">hh@gmail.com</a></td>
<td>123</td>
<td></td>
<td></td>
<td>Hien Nguyen</td>
<td><a href="mailto:nguyen.hien@students.wlax.edu">nguyen.hien@students.wlax.edu</a></td>
</tr>
</tbody>
</table>

Figure 45. Managing Supervising PT (filtered)

In this page, the user can filter the PT according to name or email. With a selected PT, the user is able to perform the following operations:

✓ Reset password for the recommender:
  - Step 1: click the Reset Password button. A textbox and a Save button will appear.
  - Step 2: type the new password in the text box.
  - Step 3: click Save. The text box and Save button will disappear.

Managing Supervising Physical Therapists

PT Name

PT Email

Filter

Select and (Send Email) Recommendation Reset Password

<table>
<thead>
<tr>
<th>Select</th>
<th>PT Name</th>
<th>PT Email</th>
<th>Licensure</th>
<th>Facility Name</th>
<th>Facility Location</th>
<th>Applicant Name</th>
<th>Applicant Email</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Sample PT</td>
<td><a href="mailto:hoanghien@gmail.com">hoanghien@gmail.com</a></td>
<td></td>
<td></td>
<td></td>
<td>Hien Nguyen</td>
<td><a href="mailto:nguyen.hien@students.wlax.edu">nguyen.hien@students.wlax.edu</a></td>
</tr>
<tr>
<td></td>
<td>Dr. Kenny Hunt</td>
<td><a href="mailto:hh@gmail.com">hh@gmail.com</a></td>
<td>123</td>
<td></td>
<td></td>
<td>Hien Nguyen</td>
<td><a href="mailto:nguyen.hien@students.wlax.edu">nguyen.hien@students.wlax.edu</a></td>
</tr>
</tbody>
</table>

Figure 46. Reset a recommender's password

The system will automatically send an email to the recommender to inform that the password has been reset.
Subject: Notification from Admission Office, Physical Therapy Program in UW-La Crosse

Your password to login our system has been reset

Please visit the link below, use the following username and password to log in.
(Copy and paste into your web browser)
Recommendation Form Link:
http://190.49.196.201:8989/pt/faces/ref/login.jspx

Username: hoangnien@gmail.com
Password: asdfgh123

Regards
Physical Therapy Program – University of Wisconsin La Crosse

Figure 47. Email to inform reset password

- View the PDF format of the recommendation: just click Recommendation button
- Send Email to the recommender. These two operations are the same as described in previous section.

4.1.7. Managing Personal References

URL: /faces/staff/personalRef.jspx

The operations are similar to those with supervising PTs: sending email, retrieving recommendation and resetting password.

![Managing Personal References](image)

Figure 48. Managing Personal References (filtered)

4.1.8 Managing semester of entry

URL: /faces/staff/setdate.jspx

![Managing semester of entry](image)

Figure 49. Managing semester of entry
How to add a new deadline:

- **Step 1:** click *Add a new deadline* button. A blank row will be displayed.
- **Step 2:** fill in the blank text boxes.
- **Step 3:** click Save.

### Manage entrance semester, year and application deadline

![Table](image)

Figure 50. Add a new deadline

### 4.1.9. Managing Staff Accounts

**URL:** /faces/staff/staff.jspx

![Table](image)

Figure 51. Administrator account

There is an administrator account of Dr. John Greany. This user can add, edit or delete a staff account. He is the only one who can get access to this page. Moreover, this administrator account cannot be deleted. That’s why when this user is selected; the Edit and Delete button are disabled.

![Table](image)

Figure 52. Staff accounts
How to edit a staff account

- Step 1: select the account to be edited
- Step 2: click edit. The email and name text boxes will become editable. A text box for new password is also rendered.
- Step 3: type new name and/or new email and/or new password.
- Step 4: click Save.

![Staff Accounts](image)

Figure 53. Edit a staff account

How to add a new staff account

- Step 1: click Create. A new blank row will be rendered.
- Step 2: fill in the name and email of this staff.
- Step 3: click Save
- Step 4: click Edit
- Step 5: type a password
- Step 6: click Save

![Staff Accounts](image)

Figure 54. Add a new staff account
4.2 Operations of the supervising physical therapists

4.2.1 Logging in
URL: /faces/ref/login.jspx

![Login page for PT](image)

Figure 55. Login page for PT

4.2.2 Changing password
URL: /faces/ref/changepass.jspx

![Changing password](image)

Figure 56. Changing password

Basically, the logging in and changing password operations are the same with other users. The account information (email and password) must be sent to the recommenders via email beforehand.
4.2.3 Fill the recommendation form
URL: /faces/ref/comment.jspx

Evaluation Form - Filled by the Supervising Physical Therapist

The applicant personal information:
- Email: nguyen.hien@students.uwlax.edu
- First Name: Hien
- Last Name: Nguyen
- Gender: Male
- Phone Number: 3475933505
- Semester of entry: Fall 2009, Deadline: 01/15/2009

The following data are provided by the applicant:
- Name of Supervising Physical Therapist: Sample PT
- Facility Name
- Facility Location (City, State)
- Type of Setting
  - Number of paid hours: 0
  - Number of volunteer hours: 0
- Start date (mm/dd/yyyy)
- End date (mm/dd/yyyy)

Note: The applicant refuses to waive his/her right to review your comments on this page.

Listed below are qualities that are crucial in physical therapy students and practicing physical therapists. For each category below, please rate the applicant on the 1 to 5 scale (1 = poor, 5 = excellent) and make comments of cite examples. If you wish to go into greater detail than this space allows, feel free to attach a letter. Thank you for this valuable contribution to the application process for future physical therapists. The information you provide is significant in the evaluation of this applicant for entry into the UW-La Crosse Physical Therapy Program.

1. Commitment to Learning: The ability to self-assess, self-correct and self-direct, identify needs and sources of learning, continually seek new knowledge and understanding.
   - 1 2 3 4 5 _ _ _ _ _ Not Assessed

2. Interpersonal Skills: The ability to interact effectively with patients, families, colleagues, other health care professionals and the community; deal effectively with cultural/ethnic diversity issues.
   - 1 2 3 4 5 _ _ _ _ _ Not Assessed
The recommender is able to get the form in PDF format to store or print. Just click PDF Format button.

4.3 Operations of a personal reference

4.3.1 Loging in

URL: /faces/personalref/login.jspx
4.3.2 Changing password
URL: /faces/personalref/changepass.jspx

4.3.3 Fill the recommendation form
URL: /faces/personalref/recform.jspx

Figure 58. Recommendation form of a personal reference
The recommender is able to get the form in PDF format to store or print. Just click PDF Format button.

4.4 Operations of an applicant

4.4.1 Register an applicant account
URL: /faces/cand/register.jspx

Figure 59. Register an applicant account

When creating a new applicant account successfully, the system will automatically send him/her a email to confirm the registration.

Subject: Notification from Admission Office, Physical Therapy Program in UW-La Crosse

Dear Applicant

This email is to confirm that you have created an account in our Admission System. Username/Email: nguyen.hien@students.uwlax.edu
Password: 123456

You can review your application at the following address (copy and paste into your web browser)
http://130.49.196.201:8080/pt/faces/cand/login.jspx

Regards

Physical Therapy Program - University of Wisconsin La Crosse

Figure 60. Email sent to the new applicant to confirm the registration

4.4.2 Logging in
URL: /faces/cand/login.jspx

At the first time logging in, the applicant will be asked to initialize his/her application.
Once the application is initialized, the applicant will be able to view the application status, the comments from the admission office. On the other hand, he will be asked to select the semester of entry as well as the deadline. He can change this information anytime before submitting the application.
### 4.4.3 Fill in the forms

<table>
<thead>
<tr>
<th>Form 1: General Information</th>
</tr>
</thead>
<tbody>
<tr>
<td>First Name</td>
</tr>
<tr>
<td>Middle Initial</td>
</tr>
<tr>
<td>Last Name</td>
</tr>
<tr>
<td>Address</td>
</tr>
<tr>
<td>City</td>
</tr>
<tr>
<td>State</td>
</tr>
<tr>
<td>Zip</td>
</tr>
<tr>
<td>Country</td>
</tr>
<tr>
<td>Phone - Daytime</td>
</tr>
<tr>
<td>Phone - Evening</td>
</tr>
<tr>
<td>Current State of Legal Residence</td>
</tr>
<tr>
<td>Permanent Mailing Address (if different from above)</td>
</tr>
<tr>
<td>Address</td>
</tr>
<tr>
<td>City</td>
</tr>
<tr>
<td>State</td>
</tr>
<tr>
<td>Zip</td>
</tr>
<tr>
<td>Country</td>
</tr>
<tr>
<td>Phone</td>
</tr>
<tr>
<td>Are you an international student?</td>
</tr>
</tbody>
</table>

The items in this section are for data collection purposes and completion is voluntary

| Date of Birth (mm/dd/yyyy) | 08/17/1979                     |
| Gender                     | Male                           |
| Ethnic/Racial Heritage     | Vietnamese                     |

Figure 63. General Information Form
## Form 2: Academic History

Please list in chronological order ALL colleges, universities and technical schools you have attended.

<table>
<thead>
<tr>
<th>Institution</th>
<th>Location (city, state)</th>
<th>From (mm/yyyy)</th>
<th>To (mm/yyyy)</th>
<th>Degree</th>
<th>Major</th>
<th>GPA</th>
<th>Credits earned</th>
</tr>
</thead>
<tbody>
<tr>
<td>Univ of Wisconsin- L</td>
<td>La Crosse, WI</td>
<td>09/2006</td>
<td>08/2008</td>
<td>Master</td>
<td>Software</td>
<td>3.5</td>
<td>48</td>
</tr>
<tr>
<td>Univ of Technology</td>
<td>Saigon, Vietnam</td>
<td>09/1997</td>
<td>05/2002</td>
<td>Bachelor</td>
<td>Computer</td>
<td>3</td>
<td>120</td>
</tr>
</tbody>
</table>

### Pre-requisite course work completed

<table>
<thead>
<tr>
<th>Course Name</th>
<th>Year</th>
<th>Semester</th>
<th>Grade</th>
<th>Credit Hours</th>
<th>Indicate UWL or other institution</th>
</tr>
</thead>
<tbody>
<tr>
<td>General Biology</td>
<td>2007</td>
<td>Spring</td>
<td>4</td>
<td>4</td>
<td>UWL</td>
</tr>
<tr>
<td>Chemistry I</td>
<td>2007</td>
<td>Fall</td>
<td>3.5</td>
<td>4</td>
<td>UWL</td>
</tr>
<tr>
<td>Chemistry II</td>
<td>2008</td>
<td>Spring</td>
<td>3</td>
<td>4</td>
<td>UWL</td>
</tr>
<tr>
<td>Human Anatomy/Physiology I</td>
<td>2008</td>
<td>Spring</td>
<td>4</td>
<td>4</td>
<td>UWL</td>
</tr>
<tr>
<td>Human Anatomy/Physiology II</td>
<td>2007</td>
<td>Fall</td>
<td>4</td>
<td>3</td>
<td>UWL</td>
</tr>
<tr>
<td>Physics I</td>
<td>2007</td>
<td>Spring</td>
<td>3</td>
<td>3</td>
<td>UWL</td>
</tr>
<tr>
<td>Physics II</td>
<td>2008</td>
<td>Spring</td>
<td>3</td>
<td>4</td>
<td>UWL</td>
</tr>
<tr>
<td>Statistics</td>
<td>2006</td>
<td>Fall</td>
<td>3</td>
<td>3</td>
<td>UWL</td>
</tr>
<tr>
<td>Psychology</td>
<td>2007</td>
<td>Spring</td>
<td>3.5</td>
<td>3</td>
<td>UWL</td>
</tr>
<tr>
<td>Sociology</td>
<td>2007</td>
<td>Spring</td>
<td>0</td>
<td>0</td>
<td></td>
</tr>
</tbody>
</table>

### Courses in last 2 semesters

<table>
<thead>
<tr>
<th>Select</th>
<th>Course Name</th>
<th>Course Title &amp; Department</th>
<th>Year</th>
<th>Semester</th>
<th>Grade</th>
<th>Credit Hours</th>
<th>Indicate UWL or other institution</th>
</tr>
</thead>
<tbody>
<tr>
<td>☑</td>
<td>Physiology I</td>
<td></td>
<td>2008</td>
<td>Spring</td>
<td>4</td>
<td>4</td>
<td>UWL</td>
</tr>
<tr>
<td>☑</td>
<td>Chemistry II</td>
<td></td>
<td>2008</td>
<td>Spring</td>
<td>3</td>
<td>4</td>
<td>UWL</td>
</tr>
</tbody>
</table>
Required courses in progress

<table>
<thead>
<tr>
<th>Select</th>
<th>Course Name</th>
<th>Course Title &amp; Department</th>
<th>Year</th>
<th>Semester</th>
<th>Credit Hours</th>
<th>Indicate UWL or other institution</th>
</tr>
</thead>
<tbody>
<tr>
<td>o</td>
<td>Psychology</td>
<td></td>
<td>2008</td>
<td>Fall</td>
<td>4</td>
<td>UWL</td>
</tr>
<tr>
<td>o</td>
<td>Sociology</td>
<td></td>
<td>2008</td>
<td>Fall</td>
<td>4</td>
<td>UWL</td>
</tr>
<tr>
<td>o</td>
<td>Sample course</td>
<td></td>
<td>2008</td>
<td>Spring</td>
<td>4</td>
<td>UWL</td>
</tr>
</tbody>
</table>

Figure 64. Academic History Form

Form 3: GRE and TOEFL

GRE

<table>
<thead>
<tr>
<th>Select</th>
<th>Test Date (mm/yyyy)</th>
<th>Verbal</th>
<th>Quantitative</th>
<th>Analytical</th>
</tr>
</thead>
<tbody>
<tr>
<td>o</td>
<td>10/2006</td>
<td>500</td>
<td>800</td>
<td>5</td>
</tr>
<tr>
<td>o</td>
<td>04/2007</td>
<td>550</td>
<td>790</td>
<td>6</td>
</tr>
</tbody>
</table>

TOEFL (for international applicants)

Test Date (mm/yyyy) 10/2005
Score 600
Type PBT

TSE (for international applicants)

Test Date (mm/yyyy)
Score

Figure 65. Standardized Tests Form
When the applicant click *Add a physical therapist* button, another web page will be opened.
Forty (40) hours of either paid or volunteer patient care experience in **two distinctly different** physical therapy patient care settings (e.g., Inpatient Rehabilitation, Acute Care, Subacute Care, Outpatient Orthopedics, Pediatrics, Skilled Nursing Facility, Schools, Home Health) under the supervision of a licensed physical therapist are required for admission. The two distinctly different experiences may occur within one institution; however, the patient populations must not overlap. It is vital that this directive is followed as it insures that each applicant has appropriate exposure to diverse patient populations. There must be a minimum of 20 hours in each setting. This experience may be either observational or participatory but **MUST INVOLVE DIRECT PATIENT CARE**.

When the application is submitted, an evaluation form will be sent to the supervising physical therapist by email.

Name of Supervising Physical Therapist

* Email address for Physical Therapist

Facility Name

Facility Location (City, State)

Type of Setting

Number of paid hours

Number of volunteer hours

Start date (mm/dd/yyyy)

End date (mm/dd/yyyy)

Patient-related activities which I observed/Performed

Valuable insights into the physical therapy profession that I acquired from this experience

Under the Family Education Rights and Privacy Act of 1974 (Buckley Amendment), you are entitled to review this form or to waive your right to access. If you do not waive this right, and request to review the form, you will be provided a copy. Please select the appropriate option below.

- [ ] I waive my right to review the letter of recommendation from this reference.
- [ ] I refuse to waive my right to review the letter of recommendation from this reference.
Figure 68. Adding a new Supervising PT

**Form 6: Personal References**

List 2 persons who are submitting letters of reference on your behalf. They must be different individuals from those who are submitting clinical evaluations.

<table>
<thead>
<tr>
<th>Select</th>
<th>Name</th>
<th>Relationship to Applicant</th>
<th>Address</th>
<th>Phone</th>
<th>Email of the reference</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Dr Thomas Gen</td>
<td>Professor</td>
<td>1725 State St</td>
<td>608-785-1234</td>
<td><a href="mailto:nhhien@yahoo.com">nhhien@yahoo.com</a></td>
</tr>
<tr>
<td></td>
<td>Dr Penyammy</td>
<td>Professor</td>
<td>1725 State St</td>
<td></td>
<td><a href="mailto:kasi@geemail.com">kasi@geemail.com</a></td>
</tr>
</tbody>
</table>

**Figure 69. Personal Reference Form**

**Form 6: Personal References**

List 2 persons who are submitting letters of reference on your behalf. They must be different individuals from those who are submitting clinical evaluations.

<table>
<thead>
<tr>
<th>Select</th>
<th>Name</th>
<th>Relationship to Applicant</th>
<th>Address</th>
<th>Phone</th>
<th>Email of the reference</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Dr Thomas Gen</td>
<td>Professor</td>
<td>1725 State St</td>
<td>608-785-1234</td>
<td><a href="mailto:nhhien@yahoo.com">nhhien@yahoo.com</a></td>
</tr>
<tr>
<td></td>
<td>Dr Penyammy</td>
<td>Professor</td>
<td>1725 State St</td>
<td></td>
<td><a href="mailto:kasi@geemail.com">kasi@geemail.com</a></td>
</tr>
</tbody>
</table>

**Figure 70. Adding a new personal reference**
Form 7: Personal Narratives

1. Please consider the characteristics of your background including your previous experiences and describe how these characteristics might allow you to contribute to a diverse learning environment or a diverse Physical Therapy workforce. (Characteristics of your background might include, for example, socioeconomic background, international travel, family makeup or size, cultural background. Please note that the characteristics of your background are not limited to the above examples but, rather, should include any characteristics and past experiences that you feel may influence you as a Physical Therapist or Physical Therapy student.)

   this is the answer for personal narrative #1

2. Please tell us about a time in your life when you experienced adversity. Was this situation stressful? Why or Why not? How did you handle the situation and what did you learn about yourself from this experience?

   this is the answer for personal narrative #2

3. Is there any additional information that you would like the admissions committee to know about you that has not already been shared in this application?

   this is the answer for personal narrative #3

Figure 71. Personal Narratives Form
Form 8: Disclosure Statements

1. Have you ever been admitted to, then withdrawn from, or been asked to withdraw from, or been dropped from, a clinical program for other than academic reasons?
   ○ Yes ○ No

2. Has licensure been denied to you, or been revoked from you, in any state in the United States for reasons other than insufficient credits or courses?
   ○ Yes ○ No

3. Have you ever been suspended, expelled, placed on probation, or otherwise disciplined by any college or university, or from any program of a college or university other than for academic reasons?
   ○ Yes ○ No

4. Have you ever been charged with, convicted of, pleaded guilty or no contest to, or forfeited bail for any criminal conduct under law or ordinance, excluding only minor traffic violations?
   ○ Yes ○ No

PLEASE NOTE: The existence of criminal charges or a criminal record or denial of revocation of a license does not constitute an automatic bar to admission and will be considered only as they substantially relate to the duties and responsibilities of the program and eventual licensure.

5. Is there anything that would prohibit you from performing essential "job-related" functions, or functions related to your responsibilities as a student in the program to which you are applying? Please refer to Part III of application packet entitled Technical Standards of Performance for Applicants.
   ○ Yes ○ No

PLEASE NOTE: The existence of a physical or mental condition or impairment does not constitute an automatic bar to admission to the student/intern program and will be considered only as it/they relate(s) to an ability to perform the duties or responsibilities of a student or intern. Further, the University will make reasonable accommodations to a physical or mental disability.

If you say 'Yes' to any of the above. Please explain

Figure 72. Disclosure Statements Form
4.4.4 Submit the application
URL: /faces/cand/submit.jspx

Submit your application

Are you sure you want to submit your application. After submission, you won't be able to modify your application.

Submit  Cancel

Figure 73. Submit the application

Once the applicant submits the application, all of the information will become read-only. He cannot modify anything except for the case that the admission staff opens it for late consideration (the second deadline). In order to do that, the admission staff just needs to change the application status from ‘Submitted’ to ‘Incomplete’. The system will automatically send an email to the applicant to notify this update.
5. LIMITATIONS & FUTURE WORK

The final web-based admission system satisfies the core functional requirements as stated in the SRS (system requirements specification). However, there exists some limitations that can be improved in future work.

The most serious limitation of this system is that all the data are transmitted over the Internet in plain text. That would cause the system vulnerable to eavesdropping attack. To overcome this threat, we can use SSL (secure socket layer) protocol to encrypt the transmitted data.

The second thing is the reliability of the system. There is no projects where all possible test cases are run through because of the combinatorial explosion. But the more test cases are performed, the more reliable the system becomes. So I suggest to perform more testings in the future work.

As for functional improvements, it is more usable if the system adds the upload functionality. This functionality would help applicants as well as recommenders to upload the documents, forms to the system.

And if possible, a payment module should be added to the system to support the admission fee payment.
6. CONCLUSIONS

The web-based admission system is a web application that allows a graduate program to collect the admission applications electronically. It reduces the time of sending application materials and time of typing data to nearly zero.

The technologies used for user interface, business logic and database are Oracle technologies. Therefore, they are working together consistently and coherently. Oracle technologies are fully supported by ITS, the maintenance team of this system.

In brief, this web-based admission system is a user-friendly, cost effective web application which resolves rather complicated tasks of a graduate admission office.
7. BIBLIOGRAPHY


APPENDIX A.
The application printed by the applicant (without average calculation).

APPENDIX B.
The printed recommendation form of a supervising physical therapist.

APPENDIX C.
The printed recommendation form of a personal reference.

APPENDIX D.
The application printed by the administrator (with average calculation).