Web-based Course Registration System using Component-Based Development

Submitted by:
Noor Mohiuddin 20362950
Nabiilah Rajabalee 20185995
Raees Hameed 20364393

Abstract

Web-based services offer users convenient access to and the ability to manipulate information that is of concern to them. Due to the high requirements in functionality and performance, these systems are often very large in terms of the size of the underlying software. The use of Component-Based Design (CBD) for web-based software comes into play for this very reason. Not only does Component-Based Software Engineering (CBSE) address the manageability issue of web based software, but it also ensures greater consistency and high re-usability of web-based components. These advantages in turn lead to better productivity and hence better quality of the overall design of the system.

In order to exemplify the effectiveness of CBD, the design and implementation of a component-based online course registration system is proposed for this project. The website will allow registered students and faculty members to view and make changes to course registration related issues for a specific semester. The system will be developed using a Service-Oriented Architecture (SOA), which involves grouping components into web services.

Problem Description

Many of the current course registration systems being used in educational institutions worldwide have lots of problems associated to usability and functionality issues.

The user interfaces for these systems are most often not well designed to provide its users with an experience that lives up to the standards of other web applications. Moreover, some of the systems have complex designs that make it difficult for users to navigate through the pages in order to use particular provided services. On the other hand, course registration systems usually have limited functionalities and only provide simple services. It would be advantageous to users if such systems could be extended to include more services, such as direct payment of tuition fees and requesting permission to register for courses.
Description of Proposed System

In this project, a web-based course registration system is to be designed and implemented using CBD. The main aim is to utilize the advantages of component-based development to solve the issues associated with most online course registration systems. Developing this online system using components will provide the following benefits:

- Quicker Development of System
- Better Services Provided to Users
- Friendlier User Interface
- Adaptable to Changes in Functionality
- Flexibility in Upgrade and Replacement of System Components
- Easier Maintenance
- Simpler Testing and Analysis

The proposed web-based course registration system is a system that allows students and faculty members to have access to different services. Figure 1 below is a high-level block diagram that shows the main components that will be designed, based on the functionalities that the final system will have.

![High-level System Block Diagram](image-url)

Figure 1 – High-level System Block Diagram
**Student Courses/Course Permissions:**
Student users will be able to manage his courses in terms of adding a course, dropping a course, searching for courses and viewing his time table for a specific semester. If they are unable to register for a course due to some course restrictions, they will be allowed to request permission for the course online.

**Student Applications:**
Students will be able to manage their application to university by viewing their admission status and uploading required documents (transcripts, references, etc). It will also be possible for them to use the system later on to apply for a change in major, apply to graduate from their degree and also apply for graduate studies.

**Student Accounts/Scholarships:**
The system will allow student users to manage details related to their tuition fees, scholarships/awards. Students will be able to view their tuition fees status and also have the option to pay for any outstanding fees using a credit card billing system. They will also be able to search for scholarships, apply for scholarships and accept/refuse scholarship offers.

**Student Transcript & Grades:**
Another functionality of the system is that students will be capable of viewing their transcripts and semester grades, and also request official transcripts online. In order to request official copies, students will have the option of paying for the transcript fees online using a credit card billing system.

**User Profile:**
Both the student and faculty users will have access to their profiles in order to view and modify their personal information, contact information and change their passwords.

**Faculty Courses:**
Faculty members will be able to view information related to the courses that they are teaching, such as the class schedule and the students registered for each course. They will also be able to view their weekly teaching schedule. Finally, faculty users will also be able to set student grades for each course.

In this system, the concept of reusability is being employed with the credit card billing component. This is because both the “Student Accounts” and “Request Official Transcripts” entities allow transactions to be done using credit card. Furthermore, the concept of substitutability can be applied to various internal components (not shown on diagram). For example, the component that will deal with searching of courses can later on be substituted by another component that allows a search to be done based on different search criteria.
Process:

This section describes the software lifecycle that we intend to follow for this project. An iterative approach will be used such that the requirements and design will be refined during the implementation and testing phases. Moreover, an incremental approach will be adopted to first create a basic system and then later on enhance it with new functionalities as the system is being developed.

Requirements and Analysis:
The functional requirements of the system will be gathered in this stage. This includes inception and consequent elaboration on all the web services that the system will provided to the students and faculty members.

Architectural Design:
The next step is to come up with the appropriate architectural model for the system. An appropriate Platform Independent Model (PIM) suitable for web-based development needs to be chosen. This will simply be an abstract view of the overall structure of the system, and will most probably consist of a heterogeneous style.

Domain-Specific & Component Design:
In this phase, we will concentrate on designing the Platform Specific Model (PSM). This will involve designing the system using various enterprise middleware styles based on the requirements defined in the initial phase. The system will most probably consist of a 3-tier or N-tier architecture. It will be designed in such a way so as to ensure proper separation of concerns between the different tiers as well as efficient communication between them. The result of this stage is a complete UML Component Diagram, where the functionalities of each component and their required and provided interfaces are properly defined.

Component Implementation & Testing
In this phase, each component, provided and required interfaces will be coded and tested separately. The system will be developed using the .NET framework. The presentation tier will be implemented using ASP .NET, while the business tier will be created using C#. Finally a portion of the database system will also be created for testing and demo purposes using Microsoft SQL Server.

System Integration & Testing
This phase involves integrating all the components together. Each component created will be combined together into web services. Then, each web services will be separately tested to ensure that they meet the required functionalities. Once this has been achieved, all the web services will be combined together and the system will be tested as a whole.
Required Resources

See “Implementation & Testing”.

Evaluation

A key element in this project is to analyze how well this course registration system performs alongside other systems. The developed system will be compared to other existing systems based on functional and non-functional analysis.

**Functional Analysis:**
The services provided by our developed system will be compared with other similar existing systems in terms of features available.

**Non-Functional Analysis:**
Tests related to non-functional criteria such as performance, code complexity, data flow and user interface friendliness. Performance analysis can be done by introducing different loads in the system using any of the open course web testing tools available and analyzing the output. The complexity of the system can be analyzed in terms of the number of times certain functions are called and the number of times certain components and interfaces are used during data access. Again, this will be done using web testing tools. The data flow analysis will consist of “measuring” the user’s navigation through pages when accessing particular services. The metrics that will be considered here consist of the number of clicks it takes to reach a service, the number of pages it takes to reach a service and how easy it is to go from one service to another. Finally, our system will be compared with other systems in terms of how user-friendly their interfaces are. This form of analysis might seem a little subjective at first. However there are several available standards that can perform evaluation based on this criterion.
## Schedule:

<table>
<thead>
<tr>
<th>Task</th>
<th>Start</th>
<th>End</th>
<th>Person</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Requirements and Analysis</strong></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Define System Functionalities</td>
<td>2nd June</td>
<td>10th June</td>
<td>Nabiilah, Noor, Raees</td>
</tr>
<tr>
<td>Research on Enterprise Styles</td>
<td>2nd June</td>
<td>6th June</td>
<td>Noor</td>
</tr>
<tr>
<td>Research on .NET and Java technologies</td>
<td>6th June</td>
<td>6th June</td>
<td>Nabiilah</td>
</tr>
<tr>
<td>Research on how to evaluate software systems</td>
<td>7th June</td>
<td>7th June</td>
<td>Noor</td>
</tr>
<tr>
<td><strong>Design</strong></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Select the System Architecture (PIM)</td>
<td>10th June</td>
<td>10th June</td>
<td>Nabiilah, Noor, Raees</td>
</tr>
<tr>
<td>Create Component Diagram (PSM)</td>
<td>11th June</td>
<td>14th June</td>
<td>Nabiilah, Noor, Raees</td>
</tr>
<tr>
<td>Delegation of tasks</td>
<td>15th June</td>
<td>15th June</td>
<td>Nabiilah, Noor, Raees</td>
</tr>
<tr>
<td>Research on ASP.NET and C#</td>
<td>16th June</td>
<td>21st June</td>
<td>Nabiilah, Noor</td>
</tr>
<tr>
<td>Research on OCL</td>
<td>21st June</td>
<td>23rd June</td>
<td>Raees</td>
</tr>
<tr>
<td><strong>Component Implementation &amp; Testing</strong></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Implementation of components</td>
<td>24th June</td>
<td>1st August</td>
<td>Nabiilah, Noor, Raees</td>
</tr>
<tr>
<td>Refinement of design</td>
<td>24th June</td>
<td>1st August</td>
<td>Nabiilah, Noor, Raees</td>
</tr>
<tr>
<td>Testing of Individual Components</td>
<td>8th July</td>
<td>26th July</td>
<td>Nabiilah, Noor, Raees</td>
</tr>
<tr>
<td><strong>System Integration &amp; Testing</strong></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Integration of Components</td>
<td>12th July</td>
<td>5th August</td>
<td>Nabiilah, Noor, Raees</td>
</tr>
<tr>
<td>System Testing</td>
<td>15th July</td>
<td>5th August</td>
<td>Nabiilah, Noor, Raees</td>
</tr>
<tr>
<td><strong>Evaluation</strong></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Functional Analysis</td>
<td>20th July</td>
<td>28th July</td>
<td>Noor</td>
</tr>
<tr>
<td>Non-Functional Analysis</td>
<td>25th July</td>
<td>6th August</td>
<td>Nabiilah, Raees</td>
</tr>
<tr>
<td><strong>Documentation</strong></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>First Presentation</td>
<td>1st July</td>
<td>7th July</td>
<td>Nabiilah, Noor, Raees</td>
</tr>
<tr>
<td>Second Presentation</td>
<td>14th July</td>
<td>21st July</td>
<td>Nabiilah, Noor, Raees</td>
</tr>
<tr>
<td>Final Report</td>
<td>1st August</td>
<td>13th August</td>
<td>Nabiilah, Noor, Raees</td>
</tr>
</tbody>
</table>
References


