Tutorial No.3

Title: Case Study: Prepare problem statement, supplementary specification and use case specification for Student/Course Registration System.

I. Problem Statement : Student Registration System

Assign the task to develop a new student registration system. The college would like a new client-server system to replace its much older system developed around mainframe technology. The new system will allow students to register for courses and view report cards from personal computers attached to the campus LAN as well as over the Internet. Professors will be able to access the system to sign up to teach courses as well as record grades.

Due to a decrease in federal funding, the college cannot afford to replace the entire system at once. The college will keep the existing course catalog database where all course information is maintained. The legacy system performance is rather poor, so the new system must ensure that access to the data on the legacy system occurs in a timely manner. The new system will access course information from the legacy database but will not update it. The registrar's office will continue to maintain course information through another system.

At the beginning of each semester, students may request a course catalogue containing a list of course offerings for the semester. Information about each course, such as professor, department, and prerequisites, will be included to help students make informed decisions.

The new system will allow students to select four course offerings for the coming semester. In addition, each student will indicate two alternative choices in case the student cannot be assigned to a primary selection. Course offerings will have a maximum of ten students and a minimum of three students. A course offering with fewer than three students will be cancelled. For each semester, there is a period of time that students can change their schedule. Students must be able to access the system during this time to add or drop courses. Once the registration process is completed for a student, the registration system sends information to the billing system so the student can be billed for the semester. If a course fills up during the actual registration process, the student must be notified of the change before submitting the schedule for processing.

At the end of the semester, the student will be able to access the system to view an electronic report card. Since student grades are sensitive information, the system must employ extra security measures to prevent unauthorized access.

Professors must be able to access the on-line system to indicate which courses they will be teaching. They will also need to see which students signed up for their course offerings. In addition, the professors will be able to record the grades for the students in each class.

II. Student Registration System Supplementary Specification :

1. Objectives

The purpose of this document is to define requirements of the Course Registration System. This Supplementary Specification lists the requirements that are not readily captured in the use cases of the use-case model. The Supplementary Specifications and the use-case model together capture a complete set of requirements on the system.

2. Scope

Supplementary Specification applies to the Course Registration System, which will be developed by the OOA and OOD students. This specification defines the non-functional requirements of the system; such as reliability, usability, performance, and supportability, as well as functional requirements that are common across a number of use cases. (The functional requirements are defined in the Use Case Specifications.).

3. References: None.

4. Functionality

- Multiple users must be able to perform their work concurrently.
- If a course offering becomes full while a student is building a schedule including that offering, the student must be notified.

5. Usability

The desktop user-interface shall be Windows NT and Windows2000 compliant.

6. Reliability

The system shall be available 24 hours a day 7 days a week, with no more than 10% down time.

7. Performance

- The system shall support up to 2000 simultaneous users against the central database at any given time, and up to 500 simultaneous users against the local servers at any one time.
- The system shall provide access to the legacy course catalog database with no more than a 10 second latency.
  Note: Risk-based prototypes have found that the legacy course catalog database cannot meet our performance needs without some creative use of mid-tier processing power

- The system must be able to complete 80% of all transactions within 2 minutes.

8. Supportability: None.

9. Security

- The system must prevent students from changing any schedules other than their own, and professors from modifying assigned course offerings for other professors.
- Only Professors can enter grades for students.
- Only the Registrar is allowed to change any student information.

10. Design Constraints

The system shall integrate with an existing legacy system, the Course Catalog System, which is an RDBMS database.

III. Use Case Specification:

1. Login:

Brief Description
This use case allows an actor to login the system.
Flow of Events
Basic Flow
This use case starts when the actor wishes to log into the Course Registration System.
1. The system requests that the actor enter his/her name and password.
2. The actor enters his/her name and password.
3. The system validates the entered name and password and logs the actor into the system.

Alternative Flow
If in the Basic Flow, the actor enters an invalid name and/or password, the system displays an error message. The actor can choose to either return to the beginning of the
Basic Flow or cancel the login, at which point the use case ends.
Pre-conditions
None
Post-Conditions
1. If the use case was successful, the user can access the system.

2. Sign up to Teach Course:

Brief Description
This use case allows a Professor to select the course offering from the course catalog for the course that he/she is eligible for and wishes to teach in the upcoming semester.
Flow of Events
Basic Flow
This use case starts when a Professor wishes to sign up to teach some course offering for the upcoming semester
1. The system retrieves and displays the list of course offering that the professor is eligible to teach. The system also retrieves and displays the list of course the professor has previously selected to teach.
2. The professor selects the course offerings that he /she wishes to teach
Alternative Flow
1. If in the Basic Flow, the professors not eligible to teach any course offerings, the system will display an error message. The professor acknowledges the message and the use case ends.
2. If the system finds a schedule conflict when trying to establish the course offering the Professor should take, the system will display an error message indicating that a schedule conflict occurs.
Pre-conditions
The professor must be logged onto the system before this use case begins.
Post-Conditions
If the use case was successful, the course offering (that a Professor schedules) has been updated.

3. Register for Course:

Brief Description
This use case allows a student to select the course offering from the course catalog for the course that he/she wishes to study in the upcoming semester.
Flow of Events
Basic Flow
This use case starts when a student wishes to register course offering that he/she wants to study for the upcoming semester.
1. The system retrieves and displays the list of course offering that the student is eligible to study.
2. Firstly, the student selects the course offerings that he /she wish to study for primary selection. Then he\she selects secondary.
Alternative Flow
1. select over 4 courses
If the student selects over four course offerings, the system will display an error message indicating that the maximum selection is four.
2. more than 10 students. If the student selects the course offering of the student number that has been selected by 10 students, the system will display a message indicating that the quota of students is full.

Pre-conditions
The student must be logged onto the system before this use case begins and register or modify course selection before period of time.
Post-Conditions
If the use case successful,

4. Close Registration

Brief Description
This use case allows a registrar to set the period of date and close the registration of student to send information to the billing system.
Flow of Events
Basic Flow
This use case starts when a registrar wishes to set the period of date or close the registration of student.
1. The register set period of date that students register for course.
2. The system retrieves and displays the list of course offering.
3. the registrar sends the completed registration to the billing system.
Alternative Flow
The registrar removes the course when the course less than 3 students.
Pre-conditions
The registrar must be logged onto the system before this use case begins
Post-Conditions
None.

5. Validation

Brief Description
This use case allows a student to validate his/her before his/her view his report card.
1. The System will request again his/her name and password
2. The actor enters his/her name and password.
3. The system validates again the entered name and password and logs the actor into the system.
Alternative Flow
If in the Basic Flow, the student enters an invalid name and/or password, the system displays an error message. The student can choose to either return to the beginning of the Basic Flow or cancel the validation, at which point the use case ends.
Pre-conditions
None
Post-Conditions
If the use case was successful, the user can view his/her report card.

6. Maintain Professor Information

Brief Description

This use case allows the Registrar to maintain all professor information.
Flow of Events
Basic Flow
This use case start when the registrar wishes to maintain professor information
1. The system retrieves and displays which the list of professor.
2. The registrar selects the student to maintain professor information.
Alternative Flow
None.
Pre-conditions
The registrar must be logged onto the system before this use case begins
Post-Conditions
If the use case was successful, the professor information has been updated

7. Maintain Student Information

Brief Description
This use case allows the registrar to maintain all student information
Flow of Events
Basic Flow
This use case start when the registrar wishes to maintain student information
1. The system retrieves and displays the list of students.
2. The registrar selects the student to maintain student information.
Alternative Flow
None.
Pre-conditions
The registrar must be logged onto the system before this use case begins
Post-Conditions
If the use case was successful, the student information has been updated

8. Submit Grades

Brief Description
This use case allows a Professor to record the grades for the students
Flow of Events
Basic Flow
This use case start when a professor to record the grades for students in each class
Alternative Flow
1. The system retrieves and displays which the list of students sign up for their course offering.
2. The professor records the grades of student.
Special Requirements
None
Pre-conditions

The professor must be logged onto the system before this use case begins.
Post-Conditions
If the use case was successful, the grades of student have been saved.

9. View Report Card

Brief Description
The use case allows student to view their report card at the end of the semester
Flow of Events
Basic Flow
The use case start when a student wishes to view his report card
1. The system retrieves the grade of courses that the student had studied in the semester
2. The student can view the report card
Alternative Flow
None.
Pre-conditions
1. The student must be logged onto the system before this use case begins
2. The student must be pass through the addition validation
Post-Conditions
None

CONCLUSIONS:

From Problem Statement: The problem statement explain about developing a new student registration system in which-

- The legacy database will be used to access course information as there is low federal funding to replace the entire system.
- Students can view Information about each course, such as professor, department, and prerequisites and select four course offerings with two alternative choices and complete the billing to complete the registration process.At certain period of time they can change their schedule whether to add or drop courses. Also they can view electronic report cards to check for the grades.
- Professors can see which courses they will be teaching and the students signed up for their course.they will also record the grades of the students for the respective course.

- The registrar's office will maintain course information, student information and the professors information.

From Supplementary Specification: We have all the non-functional requirement such as reliability, usability, performance, security and supportability are defined here in the supplementary specifications. In the course registration system-

- Multiple users must be able to work with the system at a time and also there should be no delay in accessing the legacy course catalog database and completing the transactions.
- Students should be notified if enrollment of courses reaches a maximum of ten students and course offering with fewer than three students should be cancelled.
- The system should be available 24/7, with no more than 10% down time.
- Also the security constraints must be implemented in the system to prevent unauthorized access and also students, professor or Registrar should be able to do their assigned task only

From Use Case Specification: All the functional requirements are defined in the Use Case Specifications.Here in the student registration system the use case identified are-

1. Login
2. Select course to register
3. Pay fees
4. Select course to teach
5. Submit Grades
6. View report card
7. Maintain Student Information
8. Maintain Course Information
9. Maintain Professor Information