



Daffodil
International
University

Course Project
University Registration System

Course Code: SWE331

Course Title: Object Oriented Software Development

Jahin Tasnim Ratri	151-35-1114
Section	B
Supervisor Name	Md. Alamgir Kabir
Department	Software Engineering

Date: 06-12-17

Table of Contents

Chapter 1: Introduction	6
1 Introduction	6
1.1 About the system	6
1.2 Purpose	6
1.3 Scope	6
1.4 Vision	7
1.5 Why This System Necessary??	7
1.6 Proposed Solution	7
Chapter 2: System Analysis	8
2 System Analysis	8
2.1 Actor Goal List	8
2.2 Use Case Model	9
2.3 Use Case Description (Brief)	10
2.3.1 Process Information	10
2.3.2 Access Capability	11
2.3.3 Process Registration	12
2.3.4 Course Processing Capability	13
2.4 Use Case Description (Detailed)	13
2.4.1 Process Information	14
2.4.2 Access Capability	15
2.4.3 Process Registration	16
2.4.4 Process Course	17
2.5 System Sequence Diagram	18
2.5.1 Process Information Success Scenario	18
2.5.2 Access Capability Success Scenario	18
2.5.3 Process Registration Success Scenario	19
2.5.4 Process Course Success Scenario	19
2.6 Domain Model	20
2.7 Activity Diagram	21

Chapter 3: System Design	22
3 System Design	22
3.1 Sequence Diagram	22
3.1.1 Process Information for success scenario	22
3.1.2 Access Capability for success scenario	23
3.1.3 Process Registration for success scenario	23
3.1.4 Process Course for success scenario	24
3.2 Class Diagram	25
3.3 Entity Relationship Diagram.....	26
Chapter 4: Implementation.....	27
4 Implementation	27
4.1 Tools and technologies	27

List of Table

Table 1: Actor Goal List	8
Table 2: Use Case Description for process information	14
Table 3: Use case Description for Access Capability.....	15
Table 4: Use Case Description for Process Registration.....	16
Table 5: Use Case Description for Process Course.....	17

List of Figure

Figure 1: Use Case Diagram	9
Figure 2 : Use case diagram to process information	10
Figure 3: Use case diagram to access capability	11
Figure 4: Use case diagram to process registration	12
Figure 5: Use case diagram to Process course	13
Figure 6: System sequence diagram for process information	18
Figure 7: System sequence diagram for access capability	18
Figure 8: system sequence diagram for process registration	19
Figure 9: System sequence diagram for process course	19
Figure 10: Domain Model diagram	20
Figure 11: Activity Diagram.....	21
Figure 12: Sequence diagram to process information	22
Figure 13: Sequence diagram to access capability	23
Figure 14: Sequence diagram to process registration	23
Figure 15: Sequence diagram to process course	24
Figure 16: Class Diagram	25
Figure 17: Entity Relationship Diagram	26

Chapter 1: Introduction

1 Introduction

The name of my system is University Registration System (URS). Where register maintain information, create course and delete course, Professors must be able to access the online system to indicate which courses they will be teaching, and to see which students signed up for their course offerings and student will see the course catalog, select four courses for next semester.

1.1 About the system

This system is mainly for university student to do online course registration also for professors and register in a university. This system will have the following features:

- View All Course
- Select Course
- View Course Fee
- View All Register Student
- Select Taken Course
- Create Course
- Delete Course
- Maintain Information

1.2 Purpose

This system will be made to allow you to reap the benefit of a massive cost reduction in your annual enrollment and registration process. No more paper to lose, no more trying to figure out What is the name written in that document. Instead you end up with clear validated data that is ready for your University Registration System.

In short, you are in control in your process and your way.

1.3 Scope

University Registration System is a System which will be developed to ensure that you gets the best digital system for university.

- Student--someone who is registered to take courses at the University.
- Professor--someone who is licensed to teach at the University.
- Registrar--someone who is responsible for the maintenance of the Registration System.
- Classes and schedule systematization.
- Implement changes easy as one, two, and three.
- 100 percent workability and safety.

1.4 Vision

The vision of this system would be used by professors to indicate the courses they will teach, by students to select courses, and by the registrar to complete the registration process by creating courses.

1.5 Why This System Necessary??

Now a day's course registration is currently done by hand. Students fill out forms that contain their course selections and return the forms to the adviser. Adviser then enters the selections into a database and a process is executed to create student schedules. The registration process takes from one to two weeks to complete. For saving time this system is necessary.

1.6 Proposed Solution

- ✓ Online Course Catalog
- ✓ Online Database
- ✓ Backup or restore

Chapter 2: System Analysis

2 System Analysis

2.1 Actor Goal List

Actor	Goal
Register	Maintain all information about course, professor, and student. Retrieve registration system. Create course catalog.
Professor	View Student information. Select Course to teach. View course information.
Student	View Schedule Check Course availability See course Credit View all Course Add course Drop Course

Table 1: Actor Goal List

2.2 Use Case Model

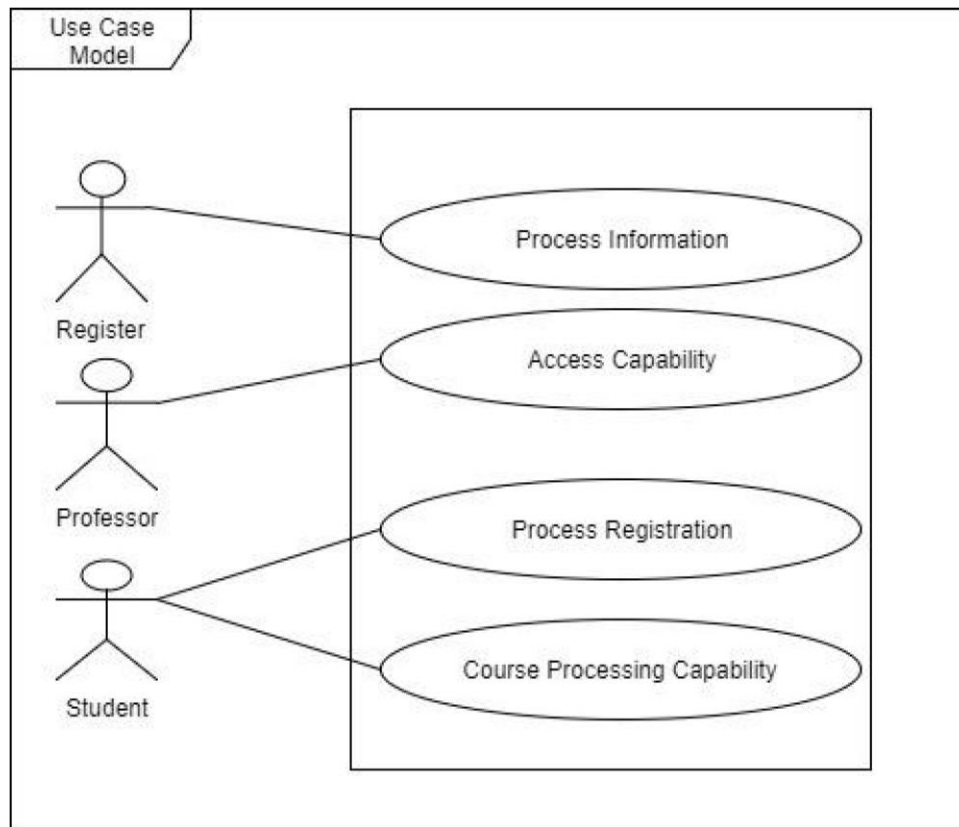


Figure 1: Use Case Diagram

2.3 Use Case Description (Brief)

2.3.1 Process Information

Register will be collect all information and maintain those information.

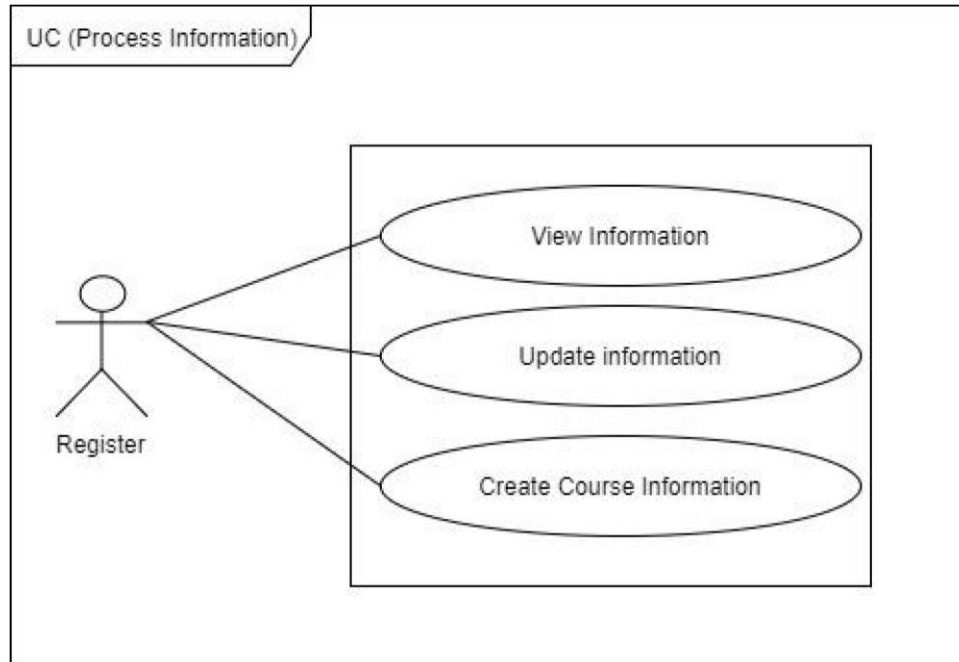


Figure 2 : Use case diagram to process information

2.3.2 Access Capability

Professors have to option to choose their course for teaching and judge those students who are register that course.

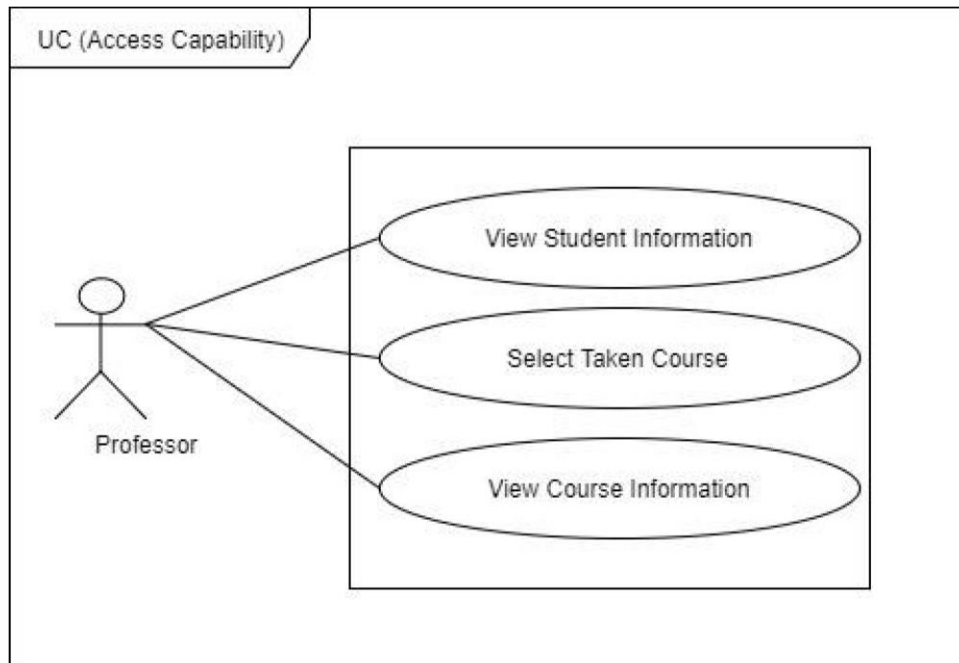


Figure 3: Use case diagram to access capability

2.3.3 Process Registration

Student must borrow their clearance to complete their registration.

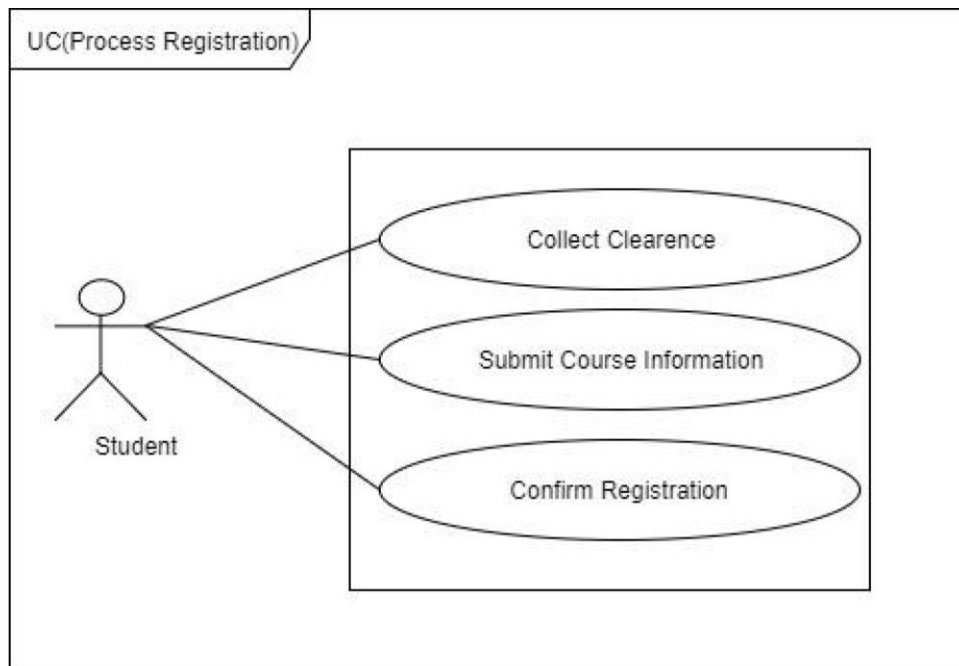


Figure 4: Use case diagram to process registration

2.3.4 Course Processing Capability

Students are able to look up their course availability with course code, course fee and course schedule.

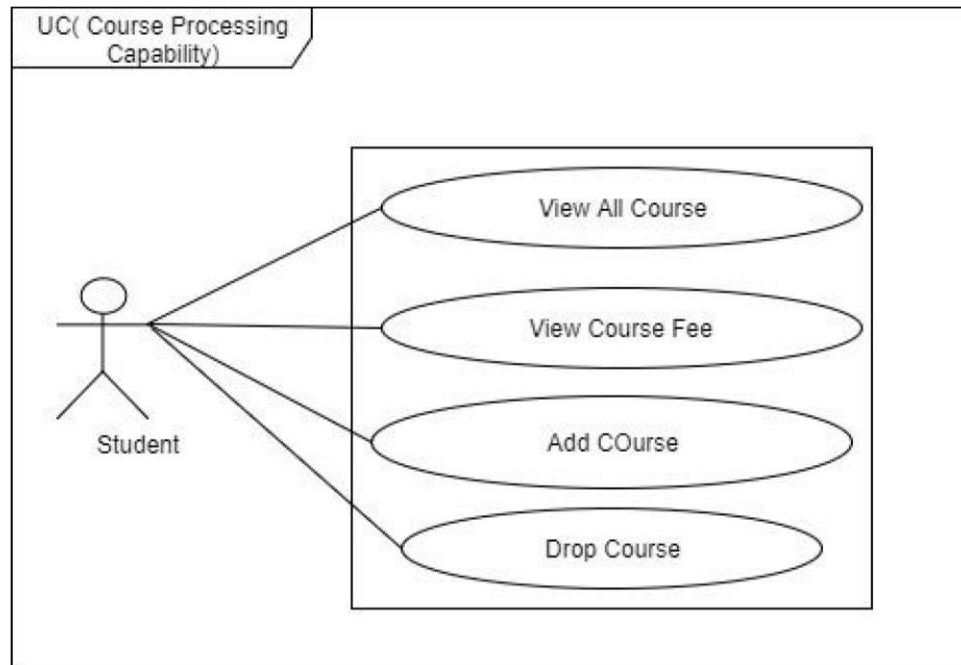


Figure 5: Use case diagram to Process course

2.4 Use Case Description (Detailed)

Here is detailed given for every single use cases. Because of that brief of use cases everyone can understand what that use case has been used for, for which actor this use case has been used.

2.4.1 Process Information

Use Case Name:	Process Information	
Scenario:	All information maintain and course catalog create	
Brief Description:	Register collects all information about course, professor and student. Register can easily check down all information. Register are capable to offering courses. If student are fewer than 3 students will be cancelled by register.	
Actor:	Register	
Preconditions:	He must be authenticated and selected by university authority and think about the course impotence.	
Post conditions:	Carefully check all information and can edit update and delete information when done processing also manage perfect teacher and set course fee.	
Flow of Events:	Actor	System
	1. Manage Information 2. Create New Course	1.1 Edit, delete and Update Course, Student and professor Information by using database where primary key is S-id, P-id, C-id. 2.1 Set C-id, C-type, C-name and Set course fee
Exception Conditions:	1.1 Without admitted student, joining professor register cannot maintain university information. 2.1 Without creating better course university would not be getting qualify student.	

Table 2: Use Case Description for process information

2.4.2 Access Capability

Use Case Name:	Access Capability	
Scenario:	Course teaching	
Brief Description:	Professor can select their course to teach student and view student who will be register under his selected course.	
Actors:	Professor	
Preconditions:	Must be capable for knowledge about selected course and authenticate for that course.	
Post conditions:	Maintain the course schedule and checking student activity and marking them.	
Flow of Events:	Actor	System
	1. Professors have to right to choose his course. 2. Teachers have the power to judge all register students in his course.	1.1 Sending request in database using C_id. 2.1 Searching database by S_id.
Exception Conditions:	1.1 If student quantity low then choosing course will be stop. 2.1 No access to view register students which course will be not under him.	

Table 3: Use case Description for Access Capability

2.4.3 Process Registration

Use Case Name:	Process Registration	
Scenario:	Course Registration	
Brief Description:	Collecting course clearance student will be complete their registration by adding or dropping courses.	
Actors:	Students	
Preconditions:	Pass all courses in previous semester and must borrow the clearance.	
Post conditions:	Ready for study and achieve some new lesson and maintain course schedule.	
Flow of Events:	Actor	System
	1. Student will be view all courses and add or drop courses by seeing fee and complete registration by getting clearance copy.	1.1 Sending request by using C-id, C-fee.
Exception Conditions:	1. 1 When any student adds course if student quantity low then course will be cancelled. 1.2 When net down student will be late. 1.3 Without full payment students are not able to doing registration.	

Table 4: Use Case Description for Process Registration

2.4.4 Process Course

Use Case Name:	Process Course	
Scenario:	About course	
Brief Description:	Student can view all the courses with schedule, availability and credit and easily add and drop their courses after paid.	
Actors:	Students	
Preconditions:	Think about the course schedule and fee for maintain that's course and must check previous result.	
Post conditions:	If all will be ok then must be done registration for learning some new lesson. Collect clearance properly.	
Flow of Events:	Actor	System
	1. Student are knows about course credit for collect their clearance copy. 2. Students see course availability and course schedule	1.1 Sending request in database by using C-id, C-fee. 2.1 Sending request in database with Cid, C-schedule, C-availability.
Exception Conditions:	1.1 Unless know about the course fee and date they will be late to doing registration. 2.1 If systems have some problem then students are not able to see course availability or course credit.	

Table 5: Use Case Description for Process Course

2.5 System Sequence Diagram

2.5.1 Process Information Success Scenario

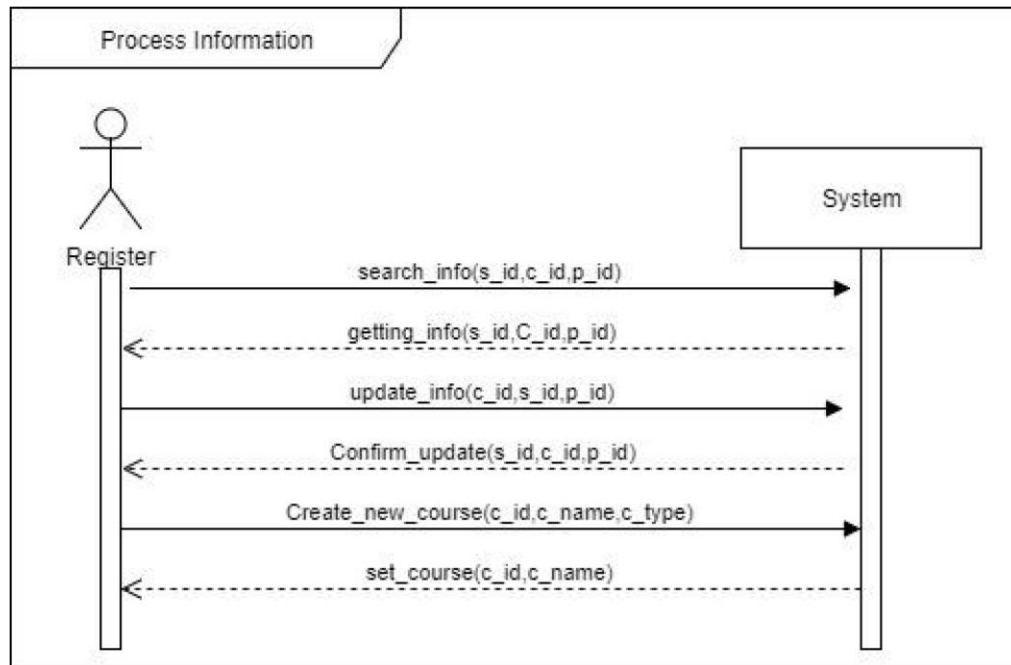


Figure 6: System sequence diagram for process information

2.5.2 Access Capability Success Scenario

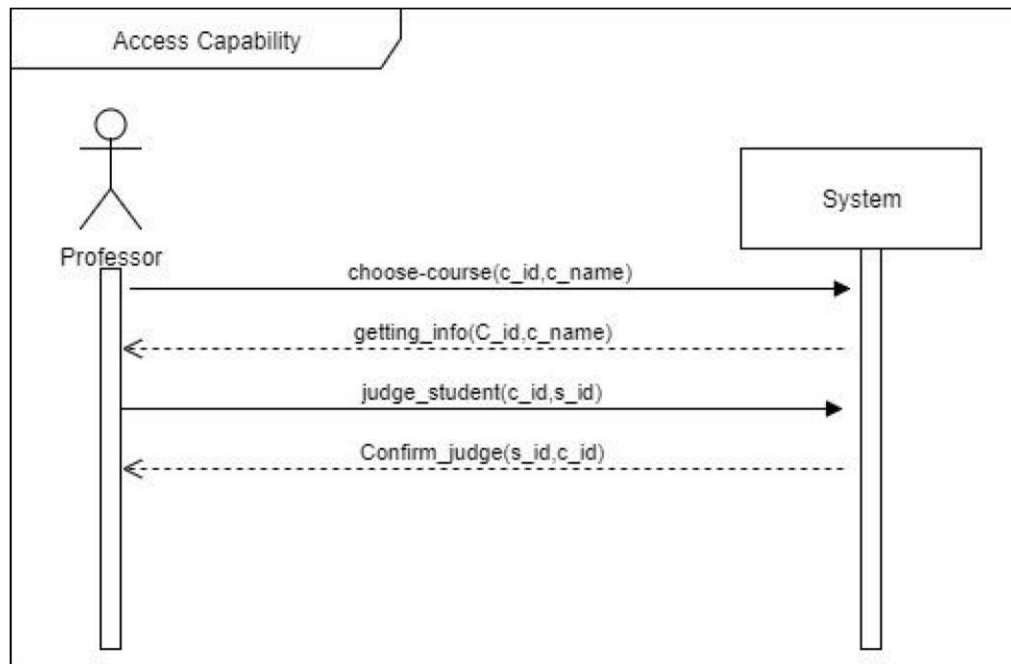


Figure 7: System sequence diagram for access capability

2.5.3 Process Registration Success Scenario

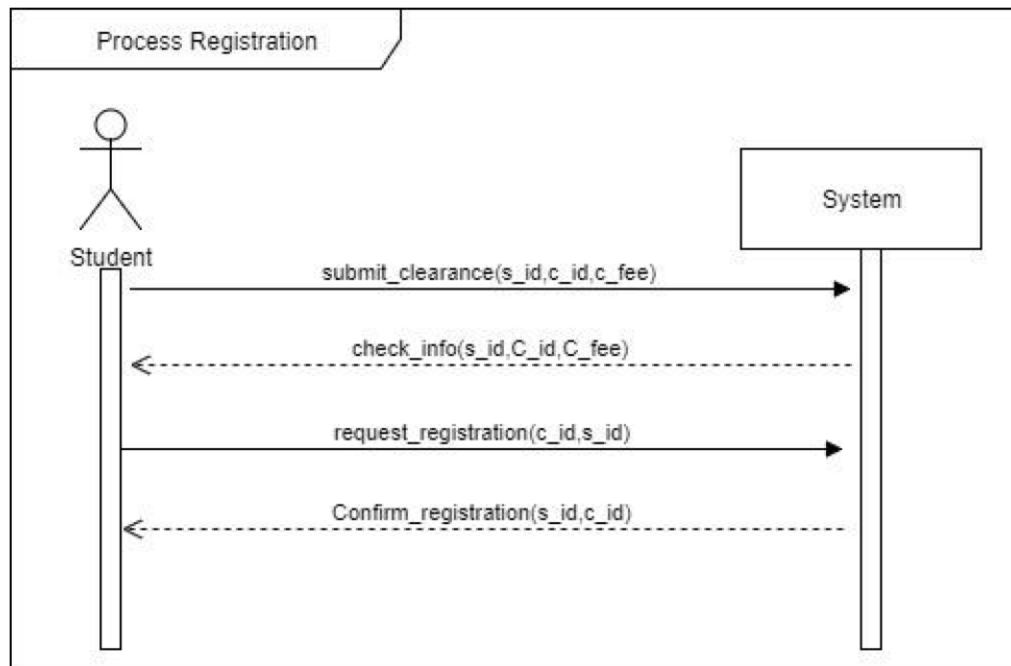


Figure 8: system sequence diagram for process registration

2.5.4 Process Course Success Scenario

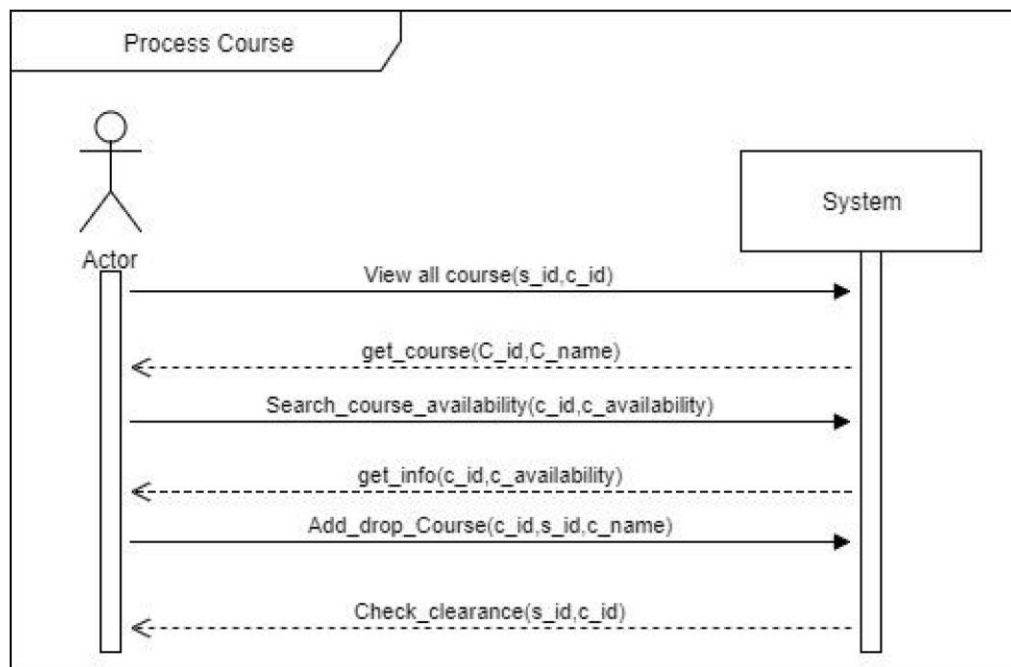


Figure 9: System sequence diagram for process course

2.6 Domain Model

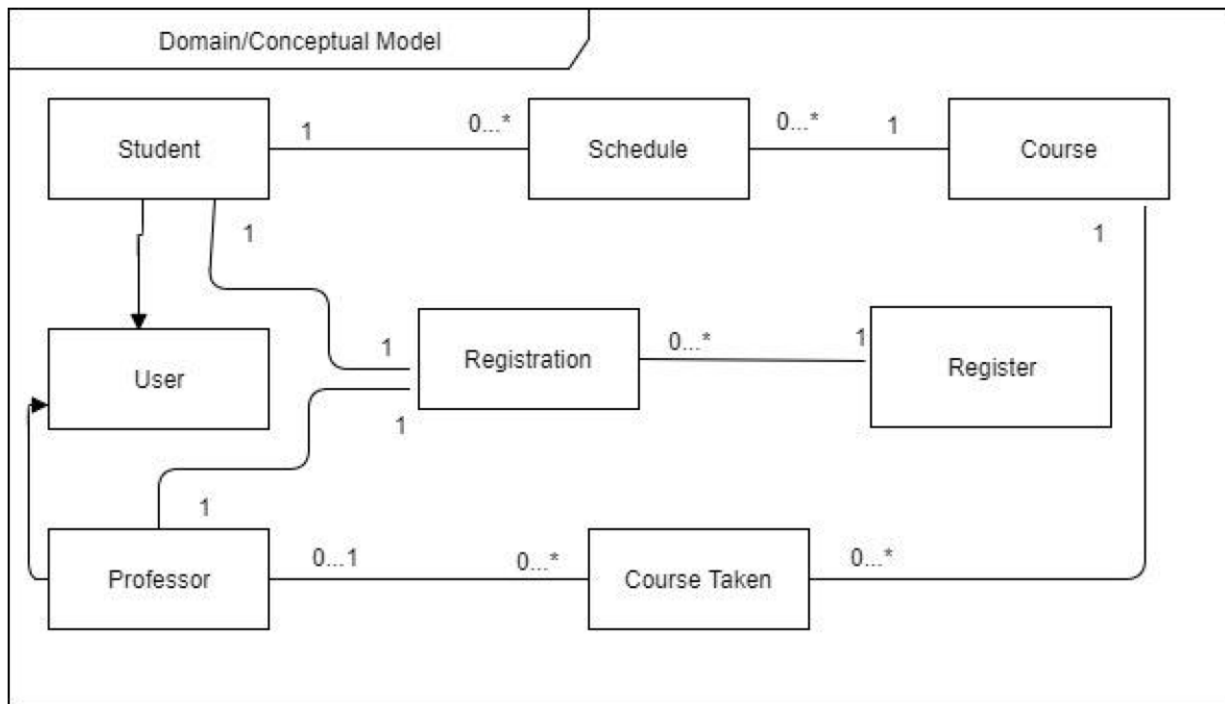


Figure 10: Domain Model diagram

2.7 Activity Diagram

Activity diagrams are graphical representations of workflows of stepwise activities and actions with support for choice, iteration and concurrency.

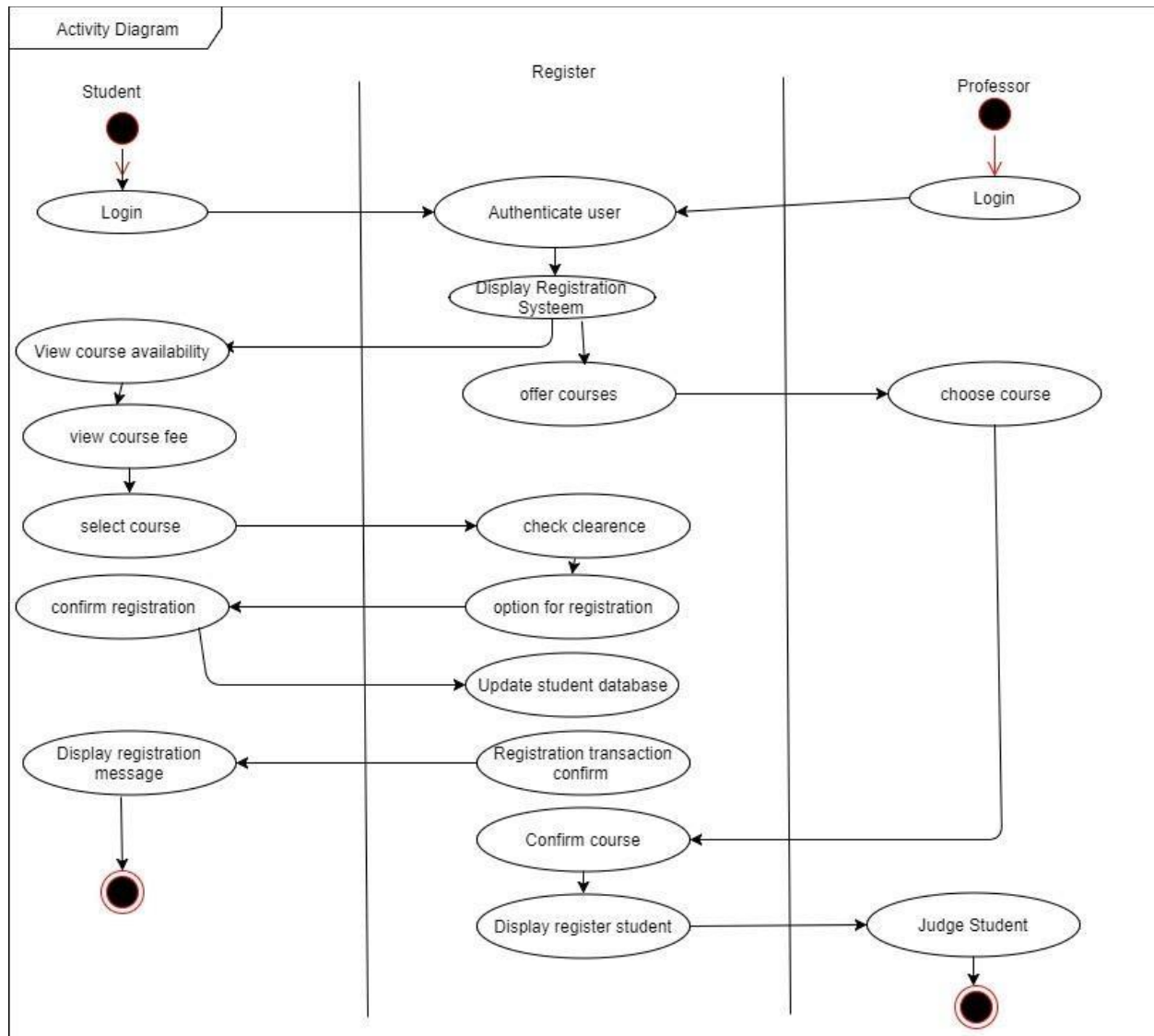


Figure 11: Activity Diagram

Chapter 3: System Design

3 System Design

Design is a process that uses the product of analysis to produce a specification for implementing a system. Design is the logical description of how a system will work.

Design emphasizes a conceptual solution that fulfills the requirements, rather than its implementation. For example, a description of a database schema and software objects.

3.1 Sequence Diagram

The UML includes interaction diagrams to illustrate how objects interact via messages. They are used for dynamic object modeling.

3.1.1 Process Information for success scenario

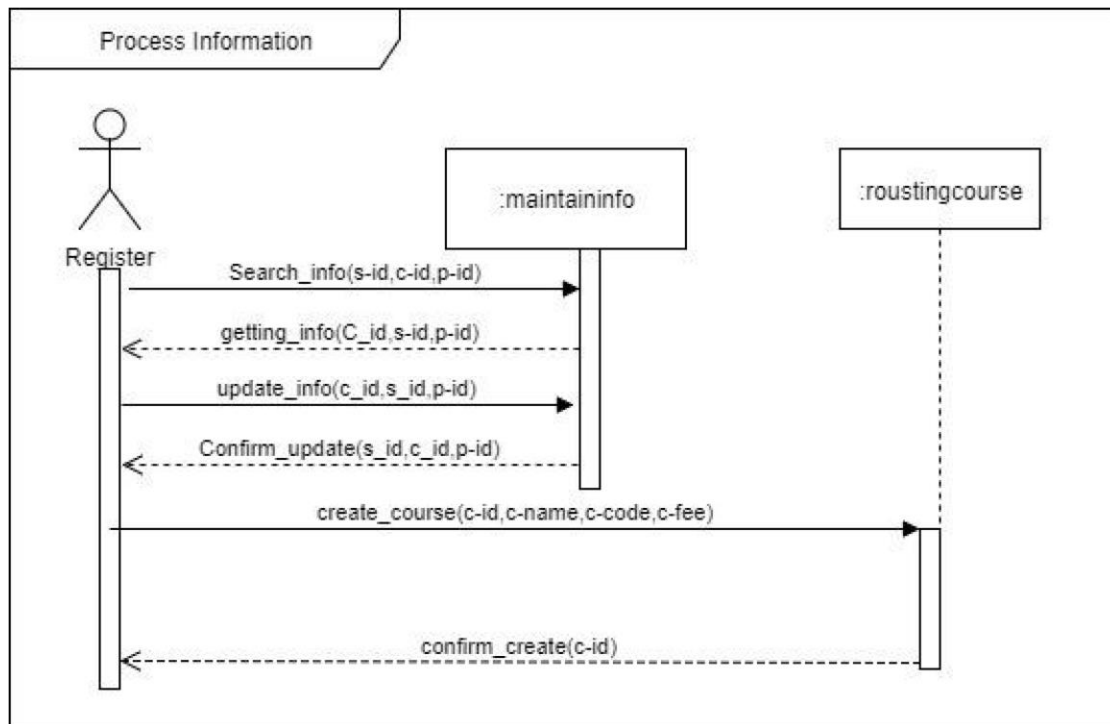


Figure 12: Sequence diagram to process information

3.1.2 Access Capability for success scenario

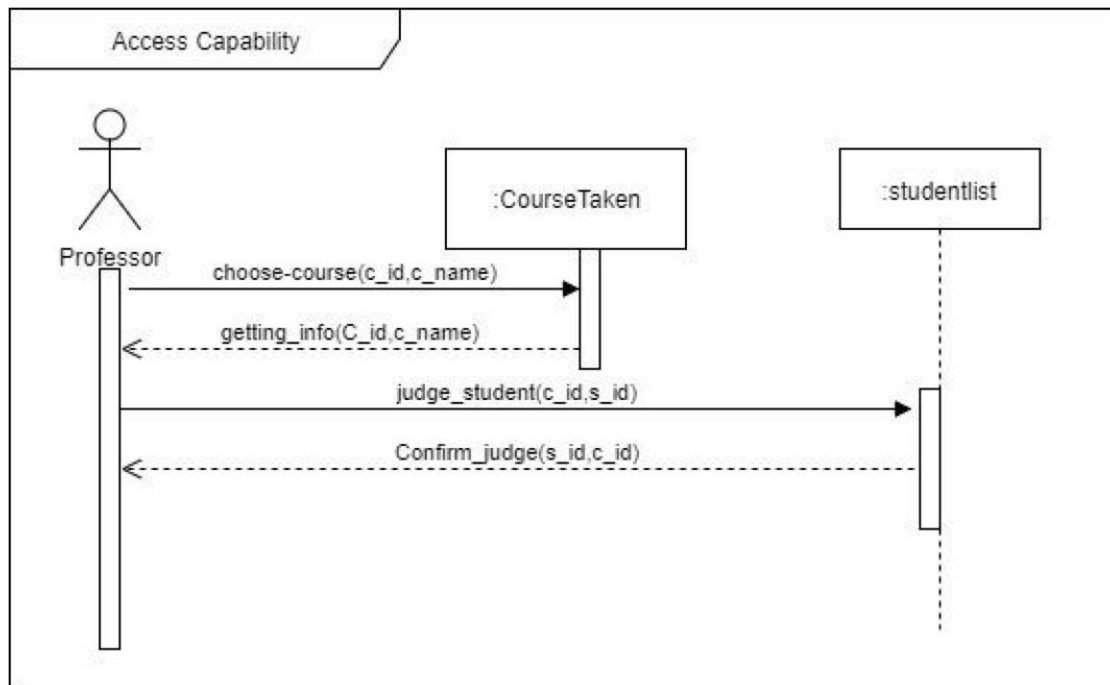


Figure 13: Sequence diagram to access capability

3.1.3 Process Registration for success scenario

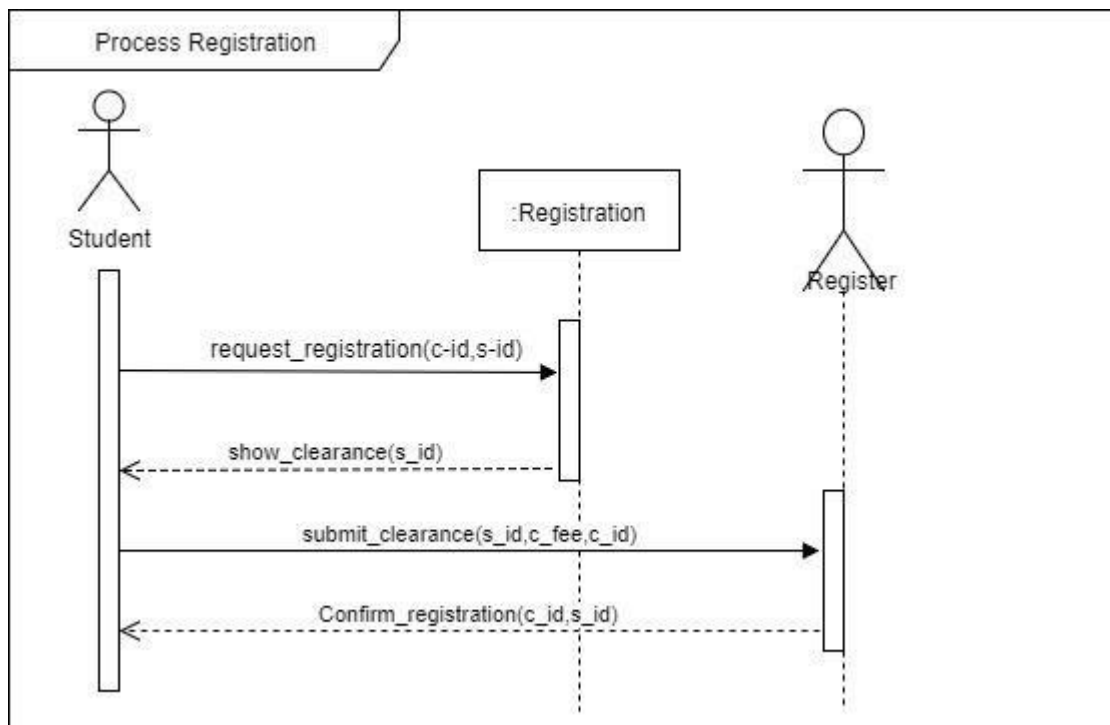


Figure 14: Sequence diagram to process registration

3.1.4 Process Course for success scenario

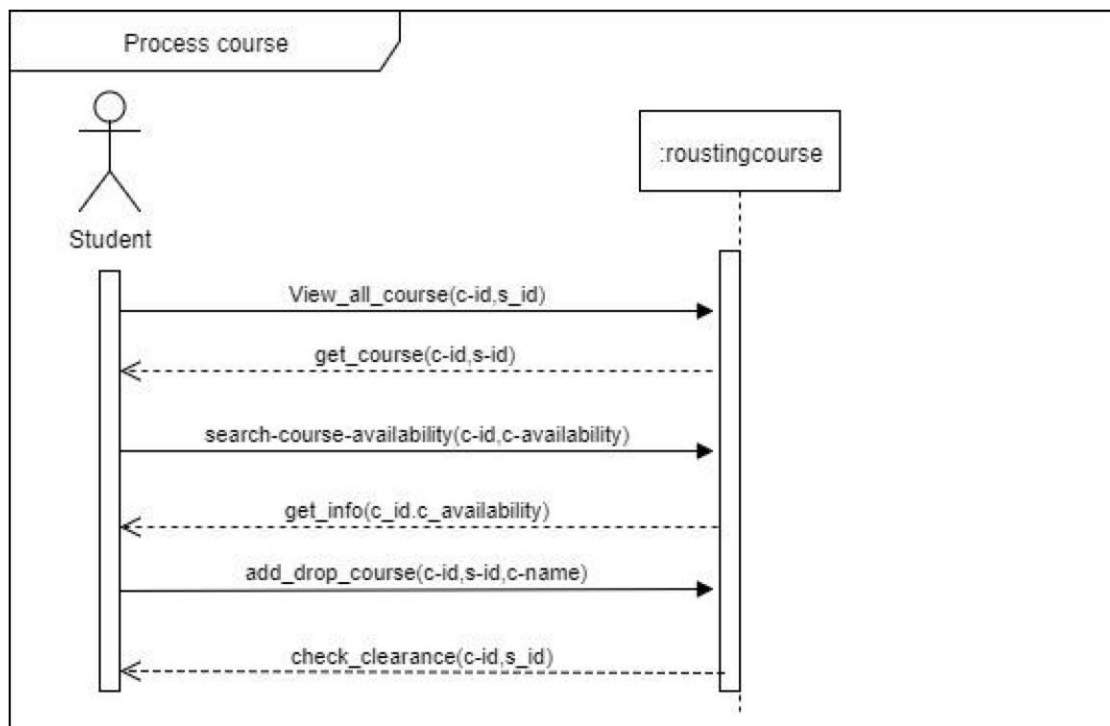


Figure 15: Sequence diagram to process course

3.2 Class Diagram

Class or structural diagrams define the basic building blocks of a model. They are used for static object modeling, describing what attributes and behavior it has rather than detailing the methods for achieving operations.

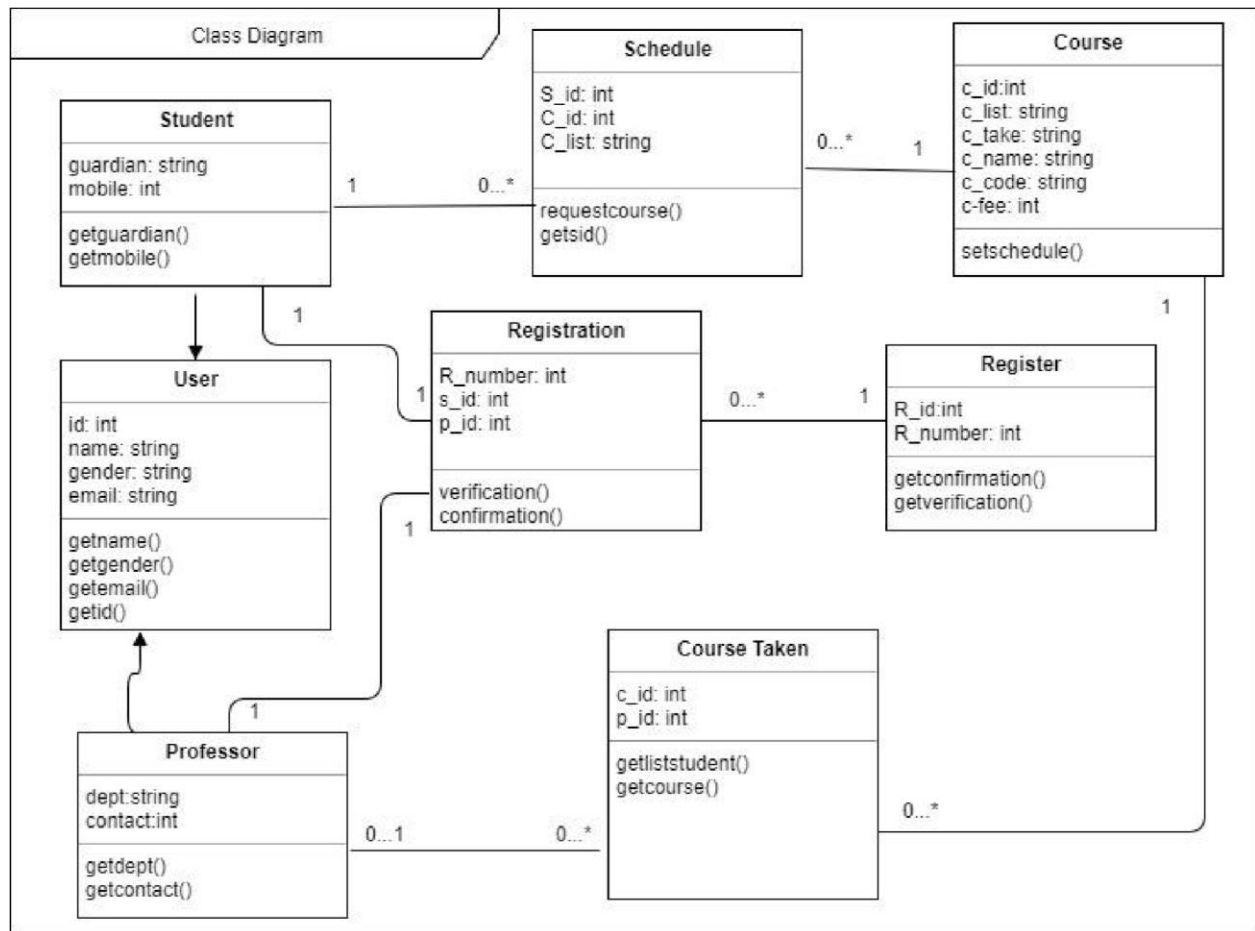


Figure 16: Class Diagram

3.3 Entity Relationship Diagram

An entity-relationship model is an abstract and conceptual representation of data. Entity relationship modeling is a database modeling method, used to produce a type of conceptual schema or semantic data model of a system, often a relational database, and its requirements in a top-down fashion. Diagrams created by this process are called Entity-Relationship Diagrams.

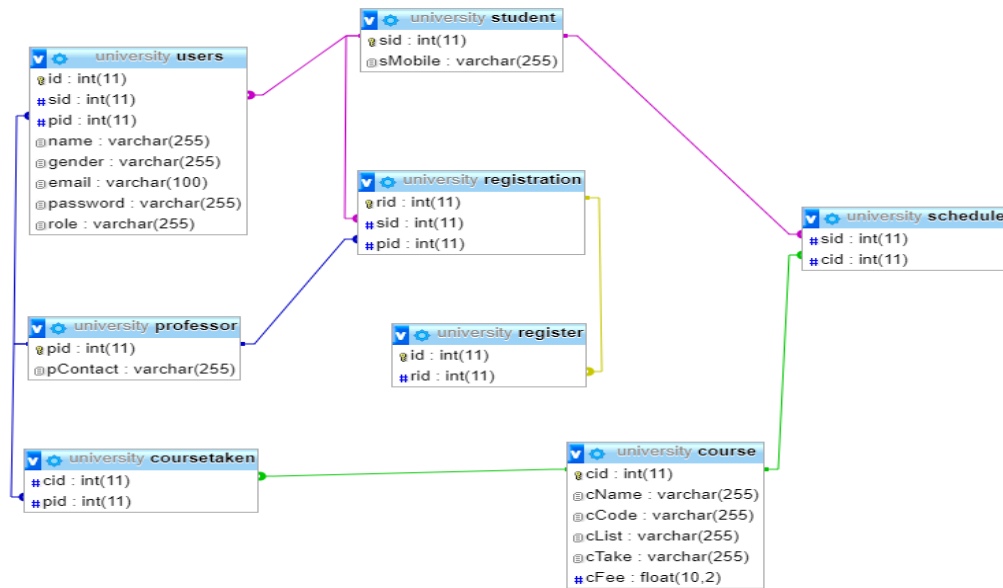


Figure 17: Entity Relationship Diagram

Chapter 4: Implementation

4 Implementation

Implementation (software) perspective describes software implementations in a particular technology (such as PHP). Implementation means programming and building the system, not deploying it. In the implementation phase, the developer builds the components either from scratch or by composition given the architecture document from the design phase and the requirement document from the analysis phase. The architecture document should give guidance. Sometimes, this guidance is found in the requirement document. The implementation phase deals with issues of quality, performance and debugging. The end deliverable of implementation phase is the product itself.

4.1 Tools and technologies

For developing this system what tools and technologies will be used are given below:

- ❖ Web Browser: Microsoft internet explorer, Mozilla Firefox, Chrome.
- ❖ Operating system as like windows 7,8.1
- ❖ Operate database software mysql for backend.
- ❖ Front in code will be written Bootstrap.
- ❖ Script will be done by java script
- ❖ Developed by PHP
- ❖ Diagram will be draw using draw.io

4.2 Code Links

<https://github.com/HmcRatri/University-Registration System>