Prathna Sukheja | Seanglidet Yean | Nishant Chemburkar

SCRAM

cz2002 Assignment

# C:\Users\jibi\Dropbox\CZ2002 - OODP\Diagram\Class Diagram1.jpgUML Class Diagram

# Design Considerations

## Files list

Application Files

|  |  |
| --- | --- |
| Classes | Description |
| Application.java | The main function, to run the application. |
| Admin.java | Contains the functions that are called in Applicatrion.java |
| Course.java | Each course in the system is an object of this class. It stores all the course information. |
| Subcomponent.java | Each subcomponent of a course is an object of this class. |
| courseInfo.java | This class stores Student information about a particular course. |
| Person.java | Super Class of class Student and Instructor. |
| Instructor.java | Each instructor in the system is an object of this class. It stores particulars of an Instructor. |
| Student.java | Each student in the system is an object of this class. It stores particulars of a Student. |
| FileOp.java | To read/write to the files which store Student, Instructor and Course information respectively. |

Text Files

|  |  |
| --- | --- |
| * Text Files | * Description |
| * course.txt | * Stores course registered in the system. |
| * instructor.txt | * Stores instructors registered in the system. |
| * student.txt | * Stores students registered in the system. |

# Assumptions

* Registered Courses, Students and professors with SCRAME are stored in course.txt, student.txt, and instructor.txt respectively. Courses registered to a Student are stored in student.txt along with Student information.
* In Student class, it was assumed that the maximum number of course to be registered is 7 course. In other words, student cannot overload courses.
* Calculating GPA, simple calculation was used which does not include complex cases, in particular, S/U option.
* Marks have not been stored in the file because data is stored in a text file and storing marks for each student becomes very tedious. Thus, marks for a student’s registered course are valid for that session only.
* Size of all Tutorial/Lab groups is same. If a student chooses to be in Tutorial group 1 of a course and lab exists for that course, then he is assigned to Lab group 1.

|  |  |  |
| --- | --- | --- |
| Function requirements | Conditions | Relevant Methods in application class. |
| Add Student | New student must not exist in the system i.e. every student should have a unique ID. | + getValidSid() + checkExistSid() + getValidEmail()  + addStudent() |
| Add Course | New course must not exist in the system i.e. every course must have a unique ID. | + getValidCid() + checkExistCid()  + addCourse() |
| Register Course | The student must already be registered in the System before adding courses. | + registerCourse() |
| Print Vacancy | Courses must be registered in the system. | +getCourseId()  +printVacancy() |
| Print Student List | Courses must be registered in the system. Tutorial or Lab lists can be printed if it exists in the Course. | +printStudentList() |
| Enter Assessment Weightage  Sub component criteria | Course must be registered in the system. | +enterAssessment() |
| Enter Course Marks | Student must be registered in the course to enter course marks for the course. | +enterCourseMarks() |
| Enter Exam Marks | Student must be registered in the course to enter exam marks for the course. | +enterExamMarks() |
| Print Course Statistics | Course has to be registered in the system. The function then prints the average percentage of exam and Course work marks for a course. | +printCourseStatistics() |
| Print Transcript | Student must be registered in the system and marks for courses taken up by students should be entered. This function then prints results for all courses a student has registered for. | +printStudentTranscript() |

# Methods, Remarks and Approaches

Changes are unavoidable while implementing a program. Therefore, to avoid rigidity, fragility and immobility, the design principles, SOLID, were taken into account.

When designing an application, the design should follow certain guidelines to minimize impact of change. Keeping design guidelines in mind, we have tried to achieve low coupling and high cohesion.

The principles which are illustrated in our program are Single Responsibility Principle (SRP) and Don’t Repeat Yourself (DRY).

### SINGLE RESPONSIBILITY PRINCIPLE

If a class has more than one responsibility, then the responsibilities become coupled. Classes Course, Student, Instructors are solely responsible for storing information of each course, each student and each course respectively.

DON’T REPEAT YOURSELF  
In order to avoid duplicate codes, the repetitive parts of the codes have been implanted into functions. These functions are to validate user input such as getValidSid(), getValidCid() or to find the student/course in the system when user enters CourseID and StudentID. Besides this validation has been implemented to check if the course and student exists in the system.

# UML Sequence Diagram

### Print Student TranscriptC:\Users\jibi\Dropbox\CZ2002 - OODP\Diagram\oodp.Admin.printStudentTranscript().jpg

# Test Cases

This section of the report demonstrated the various test cases for all the functional requirements.

Welcome to SCRAME

-------------------------------------

STUDENT COURSE REGISTRATION AND MARK ENTRY Application

Please choose an option

1.Add Student

2.Add Course

3.Register Course

4.Check Vacancy for a Course

5.Print Student list of a course

6.Enter Course Assessment Components Weightage

7.Enter Coursework Marks

8.Enter Exam Marks

9.Print Course Statistics

10.Print Student Transcript

11.Exit

### Add Student

##### Test Case 1: Invalid Input

Your Selection

1

Enter student ID:

1

Student with this ID already exists.

Nishant Chemburkar 1

Please enter Student ID again:

5

Enter Student Name:

Pooja Sukheja

Enter Email ID:

pooja@.com

Invalid email! Please enter email ID again:

Pooja@gmail.com

Enter Student Gender:

f

Enter Study Year:

7

Study Year should be between 1 and 6

Enter Study Year again:

2

Enter Programme:

csc

Student successfully registered!

-------Students registered with SCRAME:-------

ID Name

1 Nishant Chemburkar

2 Prathna Sukheja

3 Saanya Lulla

4 Akriti Vij

5 Pooja Sukheja

### Add Course

##### Test Case 1: Invalid Input

**Test case 2:Invalid Inputs.**

Your Selection

2

Enter Course ID:

CZ2004

Course with this ID already exists.

Human Computer InteractionCZ2004

Course Already Exists! Please enter Course ID again:

CZ2005

Enter course name:

OS

Enter maximum number of students allowed in the Course

30

Enter Course Credits

7

Credits have to be between 1-5.Enter Course Credits again!

3

Select Professor :

1) Name: Anwitaman Datta

2) Name: Michael Harold Lees

3) Name: David Dyer

3

Choose type of Course:

1.Lec

2.Lec/Tut

3.Lec/Tut/Lab

2

No. of tutgroups:

2

Course successfully added

-------Courses registered with SCRAME:-------

ID Name

CZ2001 Algorithms by Anwitaman Datta

CZ2002 Object Oriented Programming by Michael Harold Lees

CZ2003 Graphics by David Dyer

CZ2004 Human Computer Interaction by Michael Harold Lees

CZ2005 OS by David Dyer

### Register Course

##### Test case 1: Course Already Registered

Your Selection

3

Enter Student ID

4

1) Algorithms

2) Object Oriented Programming

3) Graphics

4) Human Computer Interaction

5) OS

6) Software Engineering

Input Course No:4

You have registered already!

##### Test case 2: No Vacancies

Your Selection

3

Enter Student ID

4

1) Algorithms

2) Object Oriented Programming

3) Graphics

4) Human Computer Interaction

5) OS

6) Software Engineering

Input Course No:6

Sorry No Vacancy for Lecture. Sorry No Vacancy for Tutorial/Lab.

##### Test case3: Invalid Student ID

Your Selection

3

Enter Student ID

8

Student not found registered in SCRAME.

##### Test case 4: Successful Registration.

Your Selection

3

Enter Student ID

4

1) Algorithms

2) Object Oriented Programming

3) Graphics

4) Human Computer Interaction

5) OS

6) Software Engineering

Input Course No:3

Lecture Vacancy : 30

Choose TUT group

Tutorial 1 Vacancy:15

Tutorial 2 Vacancy:15

Enter Group number:

1

You have successfully registered

### Check Vacancy for a Course

##### Test Case 1: Check Vacancy

Your Selection

4

Enter course ID:

CZ3002

Lecture Vacancy:10/10

Check Vacancy for:

1.Tutorials

2.Labs

1

Tutorial 1

Tutorial 2

Enter Tutorial group:

1

Vacancy for Tutorial 1

No students have been registered to this tutorial group!

##### Test Case 2: Invalid Course ID

Your Selection

4

Enter course ID:

CZ3001

No such course exists

### Print Student List

##### Test case1: Print Student List

Your Selection

5

Enter course ID:

CZ2002

Print Student List:

1.Lecture

2.Tutorials

4.Stop Printing

Enter your selection:

1

---------------Lecture List-----------------:

ID Name

2 Prathna Sukheja

3 Saanya Lulla

Print Student List:

1.Lecture

2.Tutorials

4.Stop Printing

Enter your selection:

2

Tutorials in this course:

Tutorial 1

Tutorial 2

Enter Tutorial group:

2

---------------Tutorial List-----------------:

ID Name

No students registered to this Tutorial

Print Student List:

1.Lecture

2.Tutorials

4.Stop Printing

Enter your selection:

2

Tutorials in this course:

Tutorial 1

Tutorial 2

Enter Tutorial group:

1

---------------Tutorial List-----------------:

ID Name

2 Prathna Sukheja

3 Saanya Lulla

### Enter Weightage for Courses along with Sub components

Your Selection

6

Enter Course ID

CZ0002

How many subcomponents do you wish to have?

0

Enter Exam Weightage:

90

Therefore, Course Work weightage: 10

### Enter Course Work Marks

Your Selection

7

Enter Student ID

1

Enter course Id : CZ3001

Enter Marks for Assignment1 (Out of 100)

80

Enter Marks for Assignment2 (Out of 100)

70

### Enter Exam Marks

Your Selection

8

Enter Student ID

1

Enter course Id : CZ3001

Enter Exam marks for CZ3001:

85

### Print Course Statistics

Your Selection

9

Enter course Id : CZ3001

Average Percentage Exam Marks : 90.0

Average Percentage Coursework Marks : 81.0

### Print Student Transcript

Your Selection

10

Enter Student ID

1

-----CZ3001-----

Exam marks:85.0 Exam Weightage:80

---Course Work---

Assignment1 Marks:80.0 Weightage:10

Assignment2 Marks:80.0 Weightage:10

TOTAL MARKS : 84.0

-----CZ0002-----

Exam marks:95.0 Exam Weightage:90

Course Work marks:90.0 Course Work Weightage10

TOTAL MARKS : 94.5

Nishant Chemburkar

=======================================

GPA: 4.41/5

# Work Contributions:

### **Declaration of Original Work for CE/CZ2002 Assignment**

We hereby declare that the attached group assignment has been researched, undertaken, completed and submitted as a collective effort by the group members listed below.

We have honored the principles of academic integrity and have upheld Student Code of Academic Conduct in the completion of this work.

We understand that if plagiarism is found in the assignment, then lower marks or no marks will be awarded for the assessed work.

|  |  |  |  |
| --- | --- | --- | --- |
| **Name** | **Course**  **(CZ2002)** | **Lab Group** | **Signature /Date** |
| YEAN SEANGLIDET | CZ2002 | SS2 | Yean Seanglidet/14-11-12 |
| PRATHNA | CZ2002 | SS2 | Sukheja Prathna Beju/14-11-12 |
| NISHANT | CZ2002 | SS2 | Chemburkar Nishant Rajeev/14-11-12 |