# IT 315 Final Project Part III Solution Submission Template

This template is a guide for you to organize your information. To complete it, **replace the bracketed text with the relevant information.** Some areas may be too large or too small for the information you are inserting. Adjust the size of the areas, as necessary.

**Name:** Carlos Ramirez

**Date:** 4/17/2021

1. Generate your student information system (SIS) sequence diagram for the Register a Student for Classes use case.

A picture containing diagram

Description automatically generated

Generate your SIS communication diagram for the Register a Student for Classes use case.

Diagram

Description automatically generated

**SIS Method Contract 1 template** (refer to textbook pages 306–314):

|  |  |  |
| --- | --- | --- |
| Method Name:  ValidateRegistration | Class Name:  RegisterStudent | ID:  001 |
| Clients (Consumers):  Staff and Students | | |
| Associated Use Cases:  Register Student for Class | | |
| Description of Responsibilities:  Takes arguments to verify that all requirements are meet before completing class registration. | | |
| Arguments Received:  Int – Student ID  String - Class ID | | |
| Type of Value Returned:  Boolean – True if all | | |
| Pre-Conditions:  Actor enters student ID and select class to register | | |
| Post-Conditions:  Actor received notification message of registration success | | |

**SIS Method Contract 2 template:**

|  |  |  |
| --- | --- | --- |
| Method Name:  GetClassToRegister | Class Name:  RegisterStudent | ID:  002 |
| Clients (Consumers):  Staff and Students | | |
| Associated Use Cases:  Register Student for Class | | |
| Description of Responsibilities:  Display a full list of class based on the course entered. | | |
| Arguments Received:  String - Course ID | | |
| Type of Value Returned:  Array – List of all classes associated to a course | | |
| Pre-Conditions:  Actor enters course ID to search | | |
| Post-Conditions:  Actors select a class from the list to register to | | |

**SIS Method Specification 1 template** (refer to textbook pages 314–318):

|  |  |  |
| --- | --- | --- |
| **Method Name:**  Validate Registration | **Class Name:**  RegisterStudent | **ID:**  001 |
| **Contract ID:**  Staff and Students | **Programmer:**  Carlos Ramirez | **Date Due:**  6/01/2021 |
| **Programming Language:**  Java | | |
| **Triggers/Events:**  Student will select class to register, and system will use student ID to do all validation before confirming registration. | | |

| **Arguments Received:**  **Data Type:** | **Notes:** |
| --- | --- |
| Student ID String  Class ID String | Student ID enter by the staff or student.  Staff or student select class to register. |

| **Messages Sent & Arguments Passed:**  **ClassName.MethodName:** | **Argument Data Type:** | **Notes:** |
| --- | --- | --- |
| RegisterStudent.getClassToRegister.ValidateID()  RegisterStudent.getClassToRegister.RegisterStudent() | String | Student ID is used to check validations. |
| String | Student is register to class |
|  |  |

| **Argument Returned:**  **Data Type:** | **Notes:** |
| --- | --- |
| Boolean | Return true if all validation passed. |
| **Algorithm Specification:**  Student or staff enter Id and select class to register.  System will use student ID to search student records.  System will get class ID and ensure to duplicate registration exist.  System will check student GPA and registration limits.  If online class an Verify student has all requirement equipment  Register student to class and provide confirmation message. | |
| **Misc. Notes:**  None | |

**SIS Method Specification 2 template:**

|  |  |  |
| --- | --- | --- |
| **Method Name:**  getClassToRegister | **Class Name:**  RegisterStudent | **ID:**  002 |
| **Contract ID:**  Staff and Students | **Programmer:**  Carlos Ramirez | **Date Due:**  6/01/2021 |
| **Programming Language:**  Java | | |
| **Triggers/Events:**  Student or staff will enter a course ID, and the system will display a fill list of all classes associated to that course. | | |

| **Arguments Received:**  **Data Type:** | **Notes:** |
| --- | --- |
| Course ID string | Student ID enter by the staff or student.  Staff or student select class to register. |

| **Messages Sent & Arguments Passed:**  **ClassName.MethodName:** | **Argument Data Type:** | **Notes:** |
| --- | --- | --- |
| RegisterStudent.getClassToRegister.getList() | String | Student ID is used to check validations. |
|  |  |
|  |  |

| **Argument Returned:**  **Data Type:** | **Notes:** |
| --- | --- |
| Boolean | Return true if all validation passed. |
| **Algorithm Specification:**  Student or staff enter course ID.  System will use the entered course ID.  Display classes associated to course.  Student will Select class to register. | |
| **Misc. Notes:**  None | |

1. **Verify and validate your sequence diagram and communication diagram against your SIS functional model and structural model.**

For the verification and validation of the sequence diagram I made sure that I displayed the proper sequence that the system will follow to register a student for a class. The system will start by getting the login information from the staff or student, which the system will than proceed to validate if the entered information is valid. Once the user has been validated, than the system will enter into a loop that will allow the user the register a student for a class, the system will display a menu that will prompt the user to select. After selection, the user will enter a student ID that will be used to register to a class and a validation process will occur. Once the student has been validated and course was entered, all the classes associate to that class will be displayed. Next, the user will select which class to register to, if the class is online the user will have to acknowledge that the proper equipment is own. Later, the system will proceed to check for all requirement such as student GPA, duplicate registration and display a confirmation message once registration has been validated.

1. **Explain your approach to the problem, the decisions you made to arrive at your solution, and how you completed it.**

When implementing my sequence diagram, my main goal was to implement a diagram that captured the essence of how the program will function. When I looked at all the information and work that I have completed during this course, I wanted to make sure that you can still see that in my diagrams. After identifying my actors, I proceeded to display my steps as if I were building an pseudocode or an algorithm. With this method I was able to place the steps in the proper sequence which made it easy for me to translate it into my sequence diagram. Then, I used both my pseudocode and my sequence diagram as a reference to create my communication diagram since I felt they both accomplish the same task.

1. **Reflect on this experience and the lessons you learned from it.**

During this week work I was able to learn yet another important part of the design process of a software and how the interact with each other. I did have a hard time with this assignment as I was not able to completely capture all the necessary information that I needed to complete the work. I did learn more about sequence diagrams and how helpful they can be during the process of software creating. Additionally, the introduction of a communication diagram as confusing as I found them also help me connect the dot between everything that has been covered during this course. I know for a fact that there is a lot more that I need to learn to fully understand the impact and importance of all these different diagrams, but I feel this has been a great start and introduction and has opened the door for me to continue my studies on the subject.