

# RESULT MANAGEMENT SYSTEM

SOFTWARE REQUIREMENTS SPECIFICATION

TOUKIR AHAMMED

ABA KOWSER

# **Software Requirements Specification Report**

## **Result Management System**

**Course: SE505 Software Project Lab – II**

Submitted by

<i>Toukir Ahammed</i>	<i>Roll: 0806</i>	<i>2015-2016</i>
<i>Aba Kowser</i>	<i>Roll: 0825</i>	<i>2015-2016</i>

Document Version 1.0

Submitted to

**SPL II Coordinators**



**Institute of Information Technology**

**University of Dhaka**

[20-03-2018]

Date: 20-03-2018

SPL II Coordinators

Institute of Information Technology(IIT)

University of Dhaka, Dhaka, 1000

Subject: Submission of Software Requirements Specification Report of Software Project Lab(SPL-2).

Dear Sir,

With due respect, we are pleased to submit the final Software Requirements Specification Report of 'Result Management System'. Although the report may have some lapses, we tried our best to submit a standard Software Requirements Specification Report.

We will be highly obliged if you overlook our mistakes and accept the efforts given in the documentation.

Yours sincerely,

Supervised by,

Toukir Ahammed

Md. Rayhanur Rahman,

Roll: BSSE0806

Lecturer, IIT,

University of Dhaka.

Aba Kowser

Roll: BSSE0825

# TABLE OF CONTENTS

CHAPTER 1: INTRODUCTION .....	1
1.1 PURPOSE .....	1
1.2 INTENDED AUDIENCE .....	1
1.3 CONCLUSION .....	2
CHAPTER 2: INCEPTION .....	3
2.1 INTRODUCTION .....	3
2.1.1 LISTING DOWN THE STAKEHOLDERS .....	3
2.1.2 RECOGNIZING MULTIPLE VIEWPOINTS .....	4
FACULTY MEMBER’S VIEWPOINTS .....	4
PROGRAM CHAIR AND RESULT COMMITTEE’S VIEWPOINTS .....	4
STUDENT’S VIEWPOINTS .....	4
2.1.3 WORKING TOWARDS COLLABORATION .....	4
COMMON REQUIREMENTS .....	5
CONFLICTING REQUIREMENTS .....	5
FINAL REQUIREMENTS .....	5
2.1.4 COMMUNICATION INITIATION .....	5
2.2 CONCLUSION .....	6
CHAPTER 3: ELICITATION .....	7
3.1 INTRODUCTION .....	7
3.2 ELICITING REQUIREMENTS .....	7
3.2.1 COLLABORATIVE REQUIREMENTS GATHERING .....	7
3.2.2 PROBLEM IN THE SCOPE .....	7
3.2.3 QUALITY FUNCTION DEPLOYMENT .....	8
3.2.3.1 NORMAL REQUIREMENTS .....	8
3.2.3.2 EXPECTED REQUIREMENTS .....	8
3.2.3.3 EXCITING REQUIREMENTS .....	9
3.3 USAGE SCENARIO .....	9
3.3.1 AUTHENTICATION .....	9
3.3.1.1 SIGN UP .....	9
3.3.1.2 SIGN IN .....	9

3.3.1.3 SIGN OUT .....	9
3.3.1.4 ACCOUNT RECOVERY .....	9
3.3.2 MARK MAINTENANCE .....	10
3.3.2.1 COURSE CREATION.....	10
3.3.2.2 MARK SUBMISSION .....	10
3.3.2.3 MARK COLLECTION .....	11
3.3.3 RESULT CALCULATION AND PUBLICATION .....	11
3.3.3.1 TABULATION SHEET GENERATION.....	11
3.3.3.2 RESULT CALCULATION .....	12
3.3.3.3 RESULT PUBLICATION .....	12
3.4 ELICITATION WORK PRODUCT .....	13
CHAPTER 4: SCENARIO-BASED MODEL.....	14
4.1 INTRODUCTION .....	14
4.2 DEFINITION OF USE CASE .....	14
PRIMARY ACTOR.....	14
SECONDARY ACTOR.....	14
4.3 USE CASE DIAGRAMS .....	15
4.3.1 LEVEL-0 USE CASE.....	15
4.3.2 LEVEL-1 USE CASE.....	15
4.3.3 LEVEL-1.1 USE CASE: AUTHENTICATION .....	16
4.3.4 LEVEL-1.2 USE CASE: MARK MAINTENANCE .....	17
4.3.5 LEVEL-1.3 USE CASE: RESULT CALCULATION AND PUBLICATION.....	19
4.4 ACTIVITY DIAGRAMS .....	21
4.4.1 LEVEL-1.1.1 ACTIVITY DIAGRAM: SIGNUP .....	21
4.4.2 LEVEL-1.1.2 ACTIVITY DIAGRAM: SIGN IN .....	22
4.4.3 LEVEL-1.1.3 ACTIVITY DIAGRAM: SIGN OUT .....	23
4.4.4 LEVEL-1.1.4 ACTIVITY DIAGRAM: ACCOUNT RECOVERY .....	24
4.4.5 LEVEL-1.2.1 ACTIVITY DIAGRAM: COURSE CREATION .....	25
4.4.6 LEVEL-1.2.2 ACTIVITY DIAGRAM: MARK SUBMISSION .....	26
4.4.7 LEVEL-1.2.3 ACTIVITY DIAGRAM: MARK COLLECTION .....	27

4.4.8 LEVEL-1.3.1, 1.3.2 ACTIVITY DIAGRAM: TABULATION SHEET GENERATION AND RESULT CALCULATION .....	28
4.4.9 LEVEL-1.3.3 ACTIVITY DIAGRAM: RESULT PUBLICATION AND RESULT SHOWING.....	29
4.5 SWIMLANE DIAGRAMS.....	30
4.5.1 LEVEL-1.1.1 SWIMLANE DIAGRAM: SIGNUP .....	30
4.5.2 LEVEL-1.1.2 SWIMLANE DIAGRAM: SIGN IN .....	31
4.5.3 LEVEL-1.1.3 SWIMLANE DIAGRAM: SIGN OUT .....	32
4.5.4 LEVEL-1.1.4 SWIMLANE DIAGRAM: ACCOUNT RECOVERY.....	33
4.5.5 LEVEL-1.2.1 SWIMLANE DIAGRAM: COURSE CREATION.....	34
4.5.6 LEVEL-1.2.2 SWIMLANE DIAGRAM: MARK SUBMISSION.....	35
4.5.7 LEVEL-1.2.3 SWIMLANE DIAGRAM: MARK COLLECTION.....	36
4.5.8 LEVEL-1.2.1, 1.2.2 SWIMLANE DIAGRAM: TABULATION SHEET GENERATION AND RESULT CALCULATION .....	37
4.5.9 LEVEL-1.2.1 SWIMLANE DIAGRAM: RESULT PUBLICATION AND RESULT SHOWING.....	38
CHAPTER 5: DATA-BASED MODEL .....	39
5.1 INTRODUCTION .....	39
5.2 DATA OBJECTS.....	39
5.2.1 NOUN IDENTIFICATION .....	39
5.2.2 POTENTIAL DATA OBJECTS.....	43
5.2.4 FINAL DATA OBJECTS.....	43
5.3 ENTITY RELATIONSHIP DIAGRAM .....	44
5.4 SCHEMA DIAGRAM.....	45
CHAPTER6: CLASS-BASED MODEL .....	49
6.1 INTRODUCTION .....	49
6.2 IDENTIFYING ANALYSIS CLASSES .....	49
6.2.1 GENERAL CLASSIFICATION .....	49
6.2.2 SELECTION CRITERIA .....	53
6.2.3 ASSOCIATING NOUNS WITH VERBS .....	55
6.2.4 FINAL CLASSES.....	56
6.3 CLASS RESPONSIBILITY COLLABORATION CARD .....	56

6.4 CLASS DIAGRAM .....	59
CHAPTER7: FLOW-ORIENTED MODEL .....	60
7.1 DATAFLOW DIAGRAMS.....	60
7.1.1 LEVEL-0 DATAFLOW DIAGRAM .....	60
7.1.2 LEVEL-0 DATFLOW DIAGRAM.....	61
CHAPTER8: BEHAVIORAL MODEL .....	62
8.1 STATE TRANSITION .....	62
8.1.1 EVENT IDENTIFICATION .....	62
8.1.2 STATE TRANSITION DIAGRAM .....	63
8.1.2.1 STATE TRANSITION DIAGRAM: ADMINISTRATOR.....	64
8.1.2.2 STATE TRANSITION DIAGRAM: FACULTYMEMBER .....	65
8.1.2.3 STATE TRANSITION DIAGRAM: STUDENT.....	65
8.1.2.4 STATE TRANSITION DIAGRAM: AUTHENTICATION.....	66
8.1.2.5 STATE TRANSITION DIAGRAM: SPREADSHEET .....	67
8.1.2.6 STATE TRANSITION DIAGRAM: TABULATIONSHEET .....	67
8.1.3 SEQUENCE DIAGRAM.....	68

## LIST OF TABLES

Table 1: Grading System .....	12
Table 2: Noun identification for data modeling.....	39
Table 3: Schema table of User .....	45
Table 4: Schema table of Administrator .....	45
Table 5: Schema table of Faculty Member .....	45
Table 6: Schema table of Student .....	46
Table 7: Schema table of Log .....	46
Table 8: Schema table of Course .....	46
Table 9: Schema table of TakeCourse .....	47
Table 10: Schema table of Maintain .....	47
Table 11: Schema table of UnderTakenCourse .....	47
Table 12: Schema table of Marksheet.....	47
Table 13: Schema table of Result .....	48
Table 14: Schema table of TakeCourse Has .....	48
Table 15: General Classification.....	50
Table 16: Selection Criteria .....	53
Table 17: Associate noun and verb identification.....	55
Table 18: CRC Card: User.....	56
Table 19: CRC Card: Administrator .....	56
Table 20: CRC Card: FacultyMember.....	57
Table 21: CRC Card: Authentication.....	57
Table 22: CRC Card: Course .....	57
Table 23: CRC Card: Student .....	58
Table 24: CRC Card: Log .....	58
Table 25: CRC Card: Spreadsheet .....	58
Table 26: CRC Card: TabulationSheet .....	58



## TABLE OF FIGURES

Figure 1: Level-0 Use Case Diagram.....	15
Figure 2: Level-1 Use Case Diagram.....	15
Figure 3: Level-1.1 Use Case Diagram.....	16
Figure 4: Level-1.2 Use Case Diagram.....	17
Figure 5: Level-1.3 Use Case Diagram.....	19
Figure 6: Level-1.1.1 Activity Diagram .....	21
Figure 7: Level-1.1.2 Activity Diagram .....	22
Figure 8: Level-1.1.3 Activity Diagram .....	23
Figure 9: Level-1.1.4 Activity Diagram .....	24
Figure 10: Level-1.2.1 Activity Diagram .....	25
Figure 11: Level-1.2.2 Activity Diagram .....	26
Figure 12: Level-1.2.3 Activity Diagram .....	27
Figure 13: Level-1.3.1, 1.3.2 Activity Diagram .....	28
Figure 14: Level-1.3.3 Activity Diagram .....	29
Figure 15: Level-1.3.3 Activity Diagram .....	29
Figure 16: Level-1.1.1 Swimlane Diagram.....	30
Figure 17 Level-1.1.2 Swimlane Diagram.....	31
Figure 18: Level-1.1.3 Swimlane Diagram.....	32
Figure 19: Level-1.1.4 Swimlane Diagram.....	33
Figure 20: Level-1.2.1 Swimlane Diagram.....	34
Figure 21: Level-1.2.2 Swimlane Diagram.....	35
Figure 22: Level-1.2.3 Swimlane Diagram.....	36
Figure 23: Level-1.3.1, 1.3.2 Swimlane Diagram.....	37
Figure 24: Level-1.3.3 Swimlane Diagram.....	38
Figure 25: Level-1.3.3 Swimlane Diagram.....	38
Figure 26: ER Diagram .....	44
Figure 27: Class Diagram .....	59
Figure 28: Level-0 Dataflow Diagram.....	60
Figure 29: Level-1 Dataflow Diagram.....	61
Figure 30: State Transition diagram: Administrator .....	64
Figure 31: State Transition diagram: Administrator: Faculty Member .....	65
Figure 32: State Transition diagram: Student .....	65
Figure 33: State Transition diagram: Authentication.....	66
Figure 34: State Transition diagram: Spreadsheet .....	67
Figure 35: State Transition diagram: Tabulation Sheet .....	67

# CHAPTER 1: INTRODUCTION

This chapter is a part of our software requirement specification and analysis for the project “Result Management System”. In this chapter, we focus on the intended audience for this project.

## 1.1 PURPOSE

This document briefly describes the Software Requirement Specification and Analysis of Result Management System. It contains functional, non-functional and supporting requirements and establishes a requirements baseline for the developing the system. The SRS holds the requirements are independent, uniquely numbered and organized by topic. The SRS serves as a platform to forward user requirements to the developer and provides a common reference point for both the developer team and the stakeholder community. The SRS will evolve over time as users and developers work together to validate, clarify and expand its contents.

## 1.2 INTENDED AUDIENCE

This SRS is intended for several audiences including the customers as well as the project managers, designers, developers, and testers.

- The customer will use this SRS to verify that the developer team has created a product that the customer finds acceptable.
- The project managers of the developer team will use this SRS to plan milestones and a delivery date and ensure that the developing team is on the right track when developing the system.
- The designers will use this SRS as a basis for creating the system’s design. The designers will continually refer back to this SRS to ensure that the system they are designing will fulfill the customer’s demands.
- The developers will use this SRS as a basis for developing the system’s functionality. The developers will link the requirements defined in this SRS to the software they create to ensure that they have created a software that will fulfill all of the customer’s documented requirements.
- The testers will use this SRS to derive test plans and test cases for each documented requirement. When portions of the software are complete, the testers will run their tests on that software to ensure that the software fulfills the requirements documented in this SRS. The testers will again run their tests on the entire system when it is complete and ensure that all requirements documented in this SRS have been fulfilled.

## **1.3 CONCLUSION**

This analysis of the audience helped us to focus on the users who will be using our analysis. This document will help each and every person related to this project to perceive the subject matter of the project.

# CHAPTER 2: INCEPTION

In this chapter, the Inception part of the SRS will be discussed briefly.

## 2.1 INTRODUCTION

The renowned genius Albert Einstein has said, “If I had an hour to solve a problem I’d spend 55 minutes thinking about the problem and 5 minutes thinking about the solution.” This means, it is more necessary to dig deep into the facts of the problem rather than jumping to providing a solution. Developing efficient software falls under the same jurisdiction.

Inception is the first phase of requirements engineering. It defines the scope and nature of the problem. The principal target of this stage is to create a basic understanding of the problem, identify the people involved and comprehend the nature of the solution via communication.

For a clear perception of the software requirements, a groundwork is established involving the following steps:

- Listing down the stakeholders
- Recognizing multiple viewpoints
- Working towards collaboration
- Breaking the ice and initiating communication.

### 2.1.1 LISTING DOWN THE STAKEHOLDERS

According to Sommerville and Sawyer [Som97], “Anyone who benefits in a direct or indirect way from the system which is being developed is a stakeholder.” This implies that stakeholders include the end users of the developed software as well as the people whose activities might be influenced by the tool. Towards the end of inception, the list of stakeholders is usually larger as every stakeholder is allowed to suggest one or more individuals who might be probable stakeholders for the given problem.

To identify stakeholders, we consulted a number of faculty members of different departments of University of Dhaka and asked them the following questions:

- Who will be using the product?
- Whose work will this project affect?

We identified the following stakeholders for our project.

- Faculty Member
- Student

- Program Chair
- Result Committee

## **2.1.2 RECOGNIZING MULTIPLE VIEWPOINTS**

### **FACULTY MEMBER'S VIEWPOINTS**

- ❖ Authentication for users
- ❖ Creating courses
- ❖ Giving marks in online
- ❖ Customizable fields for giving marks in different criteria's
- ❖ Single examiner and dual examiner
- ❖ Freezing marks after a specific submission deadline

### **PROGRAM CHAIR AND RESULT COMMITTEE'S VIEWPOINTS**

- ❖ Authentication for users
- ❖ Creating courses
- ❖ Giving marks in online
- ❖ Generation of tabulation sheet
- ❖ Calculating result
- ❖ Nonflexible course outline
- ❖ Freezing marks immediately after given
- ❖ Authentication via digital signature
- ❖ Single examiner and dual examiner

### **STUDENT'S VIEWPOINTS**

- ❖ Getting result to through online

The list of stakeholders will contribute to the input when requirements are elicited. Every stakeholder has different views of the system and achieves different benefits when the system is developed.

## **2.1.3 WORKING TOWARDS COLLABORATION**

Each of the stakeholder constituencies (and non-stakeholder constituency) contributes to the requirement engineering process. The greater the numbers of interactions with multiple stakeholders, the higher is the probability of inconsistency, conflicts and clashes of viewpoints. In such circumstances, requirement engineers finalize the requirements following some steps, which are listed below.

- ❖ Finding out the commonality and the conflicting points of stakeholders
- ❖ Categorizing stakeholders
- ❖ Listing down the requirements based on the stakeholder's priority points

## **COMMON REQUIREMENTS**

- ❖ Authentication for users
- ❖ Creating courses
- ❖ Customizable fields for giving marks in different criteria's
- ❖ Giving marks in online
- ❖ Generation of tabulation sheet
- ❖ Calculating result

## **CONFLICTING REQUIREMENTS**

- ❖ Flexible course outline and nonflexible course outline
- ❖ Freezing marks immediately after given and freezing marks after a specific submission deadline
- ❖ Authentication via digital signature
- ❖ Single examiner and dual examiner

## **FINAL REQUIREMENTS**

- ❖ Creating Courses
- ❖ Giving marks in online
- ❖ Customizable fields for giving marks in different criteria's
- ❖ Generating tabulation sheet
- ❖ Calculating result
- ❖ Managing different modules for single examiner and dual examiner
- ❖ Error free system in terms of output
- ❖ Efficient than existing system
- ❖ Authentication
- ❖ Publishing result to student through online

## **2.1.4 COMMUNICATION INITIATION**

In requirements engineering, the involved individuals can be broadly divided into two clusters: the developers and the stakeholders. Coming from different backgrounds, it will be obvious that these two parties will have different points of views regarding the problem. The stakeholders have more knowledge on facing the problem. Meanwhile, the developers are experienced with providing computerized solutions. Thus, in order to obtain an efficient solution to the problem, it is important to 'loosen up' or 'break the ice' between the two groups.

Following the ideal guidelines of requirement engineering, some context free questions were asked. The context free questions help throwing light on the stakeholders of the project. The next set of questions includes the context itself so that a better understanding of the problem is obtained. The stakeholder is encouraged to voice out his/her opinions about an alternate solution and also provide recommendations to the developer's suggestions. The final set of questions focuses on the communication activity itself.

## 2.2 CONCLUSION

The intense hours of developing a software is fruitful only if the users are benefitted and satisfied. Jumping into coding, right after signing up for a project throws both the clients and the developers into the risks of failure. A successful project demands a better perception of the problem. The best and easiest way to sketch out the hints of a solution is to interact with those encountering the problem itself. This is where inception phase comes.

Inception phase has given us the opportunity to create a basic understanding of the problem and perceive an abstract idea of the nature of the solution. Direct interaction with the stakeholders made us come across core points of a solution and realize the effectiveness of communication between two parties. We believe that our groundwork will help us implement an efficient solution to the problem.

# CHAPTER 3: ELICITATION

After discussing on the Inception phase, we need to focus on the Elicitation phase. So, this chapter specifies the Elicitation phase.

## 3.1 INTRODUCTION

The second phase of requirements engineering is elicitation. The main task of elicitation is to combine elements of problem solving, elaboration, negotiation and specification. Gathering information from stakeholders regarding the problem was not sufficient to design the software. The problems that arose, were encountered following the principles of elicitation.

## 3.2 ELICITING REQUIREMENTS

Stakeholders work together to identify the problems, propose elements of the solutions, negotiate different approaches and specify an initial set of solution requirements. This approach is sometimes called Facilitated Application Specification Technique (FAST). Elicitation has some sub-phases which are:

- ❖ Collaborative Requirements Gathering
- ❖ Quality Function Deployment
- ❖ Usage Scenario
- ❖ Elicitation Work Products

### 3.2.1 COLLABORATIVE REQUIREMENTS GATHERING

During inception, basic questions and answers established the scope of the problem. However, some problems persisted about the scope as the boundary of the system was ill defined and the customers have stated some unnecessary confusing detail. Also, customers/stakeholders do not have a proper understanding about the abilities of the computing environment which results in further discussion regarding the problem domain and product requirements. The requirements were put under re-evaluation by doing following tasks.

- ❖ Meetings were conducted with stakeholders and we went into further investigation about their requirements and expectations
- ❖ They were inquired about the problems with the existing workflow
- ❖ The final requirement list was derived at the end of the meeting

### 3.2.2 PROBLEM IN THE SCOPE



A number of problems were encountered in the course of preparing the software requirement specification and analysis of Result Management System.

## **Scopes**

- ❖ Software was designed for a single department at a time
- ❖ Software was designed for web-based platform

## **Limitations**

- ❖ High level security of data was not ensured
- ❖ Faculty member are responsible for providing valid course related and student related information

### **3.2.3 QUALITY FUNCTION DEPLOYMENT**

Quality Function Deployment (QFD) is a quality management technique that translates the needs of the clients into technical requirements for the software. The prime concern of the QFD is customer satisfaction maximization. In order to ensure this, QFD enforces an understanding of what the customers describe as ‘valuable’ and then deploy these values throughout the engineering process.

QFD defines three types of requirements:

- ❖ Normal Requirements
- ❖ Expected Requirements
- ❖ Exciting Features

#### **3.2.3.1 NORMAL REQUIREMENTS**

Normal requirements refer to the objectives and the goals that are stated for the product during the meeting with the stakeholders. The presence of these requirements ensures the satisfaction of the customers. The normal requirements for the project are stated below.

- ❖ Creating Courses
- ❖ Giving marks in online
- ❖ Customizable fields for giving marks in different criteria's
- ❖ Generating tabulation sheet
- ❖ Calculating result
- ❖ Managing different modules for single examiner and dual examiner

#### **3.2.3.2 EXPECTED REQUIREMENTS**

- ❖ Error free system in terms of output

- ❖ Efficient then existing system
- ❖ Authentication and authorization

### **3.2.3.3 EXCITING REQUIREMENTS**

## **3.3 USAGE SCENARIO**

Result Management System is a web-based system consisting of following subsystems:

- ❖ Authentication
- ❖ Mark Maintenance
- ❖ Result Calculation and Publication

### **3.3.1 AUTHENTICATION**

#### **3.3.1.1 SIGN UP**

The faculty member can sign up into the system by providing the following information:

- ❖ Name
- ❖ Designation
- ❖ Faculty
- ❖ Department/Institution
- ❖ Email Address (must be unique and provided by University of Dhaka)
- ❖ Password (must be at least 8 characters)

After providing this information a verification code will be sent to the provided email address with five minutes expiry time. Then the system will ask for the verification code from the user. If the user can give the correct code, the account will be activated. The administrator's account will be predefined with a username and password.

#### **3.3.1.2 SIGN IN**

The users can sign in into the system with their username and password. The email address will be used as username. After signing in, the users can change their password and other information.

#### **3.3.1.3 SIGN OUT**

The users can sign out by selecting sign out option.

#### **3.3.1.4 ACCOUNT RECOVERY**

The user can recover her account if she forgets her password. She will be asked for the email address she used for her account. If the email address exists, a verification code will be sent to the provided email address with five minutes expiry time. Then the system will ask for the verification

code from the user. If the user can give the correct code, she can change her password and thus the account will be recovered and she will be signed in into the system.

## **3.3.2 MARK MAINTENANCE**

### **3.3.2.1 COURSE CREATION**

The faculty member can create her course by providing the following information:

- ❖ Course Title
- ❖ Course Code
- ❖ Course mode (e.g. single or dual examiner)
- ❖ Credit
- ❖ Semester/Year
- ❖ Academic Year
- ❖ Role (e.g. first examiner, second examiner)

The enrolled student information can be imported from a CSV file. The CSV file will contain following information of student:

- ❖ Name
- ❖ Registration Number
- ❖ Admission Session
- ❖ Class Roll
- ❖ Exam Roll
- ❖ Current Semester/Year

### **3.3.2.2 MARK SUBMISSION**

After creating a course and importing student information as a first examiner, the faculty member will get a spreadsheet with the student information and the following column:

- ❖ Attendance
- ❖ Assignment
- ❖ Class Test
- ❖ Midterm/In-Course
- ❖ Lab Examination
- ❖ Total Continuous Evaluation
- ❖ Final Examination
- ❖ Total Marks

The faculty member can also add additional column and additional field under a column. Then she will assign weight for every category of marks. She can input marks, edit them and save it. The total marks will be calculated according to the weights she has given.

The second and third examiner will get a spreadsheet only for submitting final examination marks. The activity of giving marks will be recorded in log. The administrator will define two different deadlines for submitting total marks of continuous evaluation and final examination. The faculty member must submit the marks within the deadline. She cannot make any changes after submission or after the deadline.

### **3.3.2.3 MARK COLLECTION**

The administrator will collect the marks for all courses of an individual semester/year after the submission deadline is over. If any courses mark is missing, the administrator can extend the deadline with respect to the approval of the departmental head or director's approval. The marks of final examination of dual examiners courses will be calculated as per the following rules:

- ❖ If the marks of two examiners differ by less than 20%, average of these two will be considered
- ❖ Else a third examiner will have to submit the marks evaluating the answer scripts again. The average marks of third examiner and the examiner whose mark is closer to the third examiner will be considered

## **3.3.3 RESULT CALCULATION AND PUBLICATION**

### **3.3.3.1 TABULATION SHEET GENERATION**

The administrator can generate the tabulation sheet for a specific year/semester after the marks of each course of that year/semester have been submitted by the courses' instructors. The administrator will provide year/semester and calendar year. Then a tabulation sheet of that specific year/semester will be generated with the following information of each student enrolled in that year/ semester:

- ❖ Name
- ❖ Class Roll
- ❖ Exam Roll
- ❖ Registration No.
- ❖ Session
- ❖ Course Title, Course Credit, Obtained Marks, Letter Grade, Grade point (for all courses individually)

The administrator can view, download or print the tabulation sheet.

Grades in each course will be assigned (in accordance with the uniform grading system of University of Dhaka) as mentioned in the table below.

Table 1: Grading System

Marks	Letter Grade	Grade Point
80% or above	A+	4.00
$\geq 75\%$ but $< 80\%$	A	3.75
$\geq 70\%$ but $< 75\%$	A-	3.50
$\geq 65\%$ but $< 70\%$	B+	3.25
$\geq 60\%$ but $< 65\%$	B	3.00
$\geq 55\%$ but $< 60\%$	B-	2.75
$\geq 50\%$ but $< 55\%$	C+	2.50
$\geq 45\%$ but $< 50\%$	C	2.25
$\geq 40\%$ but $< 45\%$	D	2.00
Less than 40%	F	0.00
	I	Incomplete
	W	Withdrawn

### 3.3.3.2 RESULT CALCULATION

The administrator can calculate the result of an individual year/semester by selecting the option. The system will calculate the result for every student enrolled in that semester or year. The GPA or CGPA will be calculated with the following formula:

$$GPA \text{ or } CGPA = \frac{\sum(\text{Grade point in a course} \times \text{credits for the course})}{\text{Total Credits Taken}}$$

The administrator can view and download the result sheet.

### 3.3.3.3 RESULT PUBLICATION

The administrator will provide a specific date for result publication. The result will be published on that date and will be available to students. The student can see their result by providing their registration number, admission session, year/semester, academic year. The result will contain the following information:

- ❖ Course wise grade point
- ❖ GPA of current semester

- ❖ CGPA up to previous semester
- ❖ CGPA up to current semester

The student can view download their result.

## **3.4 ELICITATION WORK PRODUCT**

At first, we have to know whether the output of the elicitation task may vary because of the dependency on the size of the system or the product to be built. Here, the elicitation work product includes the following steps.

- ❖ Making a statement of our requirements for the Result Management System
- ❖ Making a bounded statement of scope for our system
- ❖ Making a list of users and other stakeholders who participated in the requirements elicitation
- ❖ A set of usage scenarios that provide insight into the use of the system
- ❖ Description of the system's technical environment

# **CHAPTER 4: SCENARIO-BASED MODEL**

This chapter describes the Scenario-Based Model for the Result Management System.

## **4.1 INTRODUCTION**

When developing software, user satisfaction is given the highest priority. The effective method to identify the requirements to establish meaningful analysis and design model is by determining how end user and other actor wants to interact with the system. Thus, requirements modelling begins with scenario generation in the form of use cases, activity diagrams and swim lane diagrams.

## **4.2 DEFINITION OF USE CASE**

A Use Case captures a contract that describes the system behavior under various conditions as the system responds to a request from one of its stakeholders. In essence, a Use Case tells a stylized story about how an end user interacts with the system under a specific set of circumstances. A Use Case diagram simply describes a story using corresponding actors who perform important roles in the story and makes the story understandable for the users.

The first step in writing a Use Case is to define that set of “actors” that will be involved in the story. Actors are the different people that use the system or product within the context of the function and behavior that is to be described. Actors represent the roles that people play as the system operators. Every user has one or more goals when using the system.

### **PRIMARY ACTOR**

Primary actors interact directly to achieve required system function and derive the intended benefit from the system. They work directly and frequently with the software.

### **SECONDARY ACTOR**

Secondary actors support the system so that primary actors can do their work. They either produce or consume information.

## 4.3 USE CASE DIAGRAMS

### 4.3.1 LEVEL-0 USE CASE

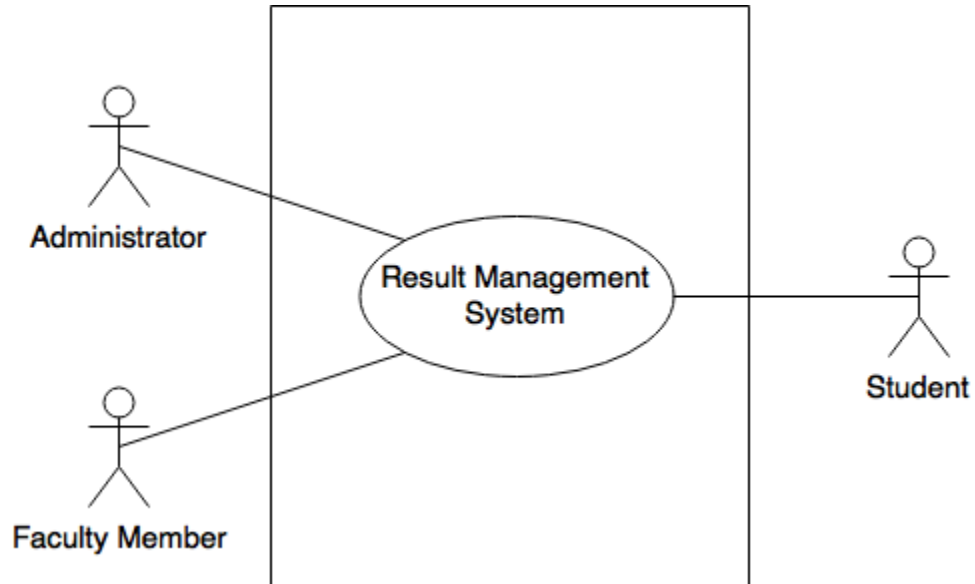


Figure 1: Level-0 Use Case Diagram

### 4.3.2 LEVEL-1 USE CASE

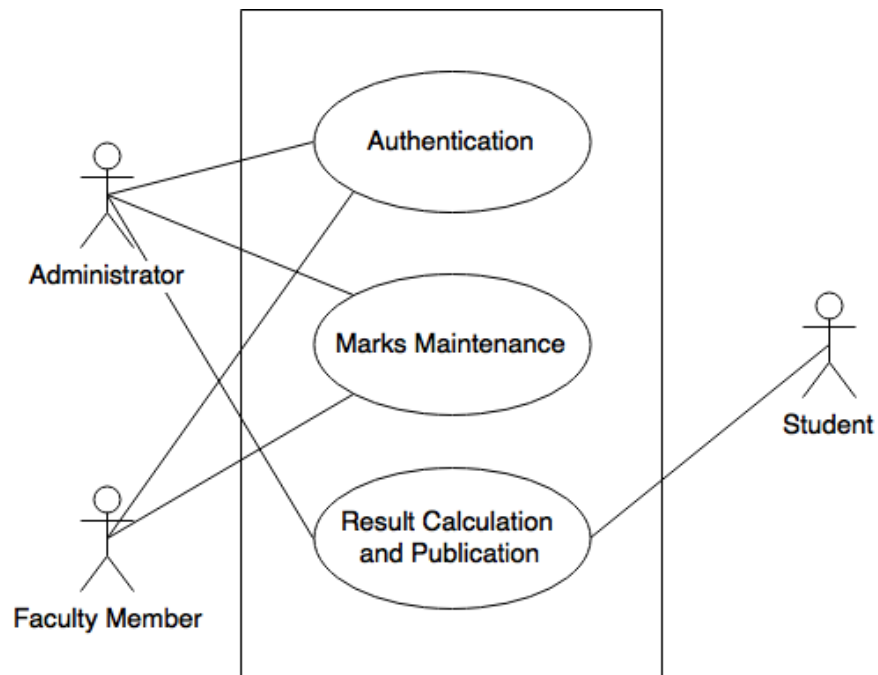


Figure 2: Level-1 Use Case Diagram



### 4.3.3 LEVEL-1.1 USE CASE: AUTHENTICATION

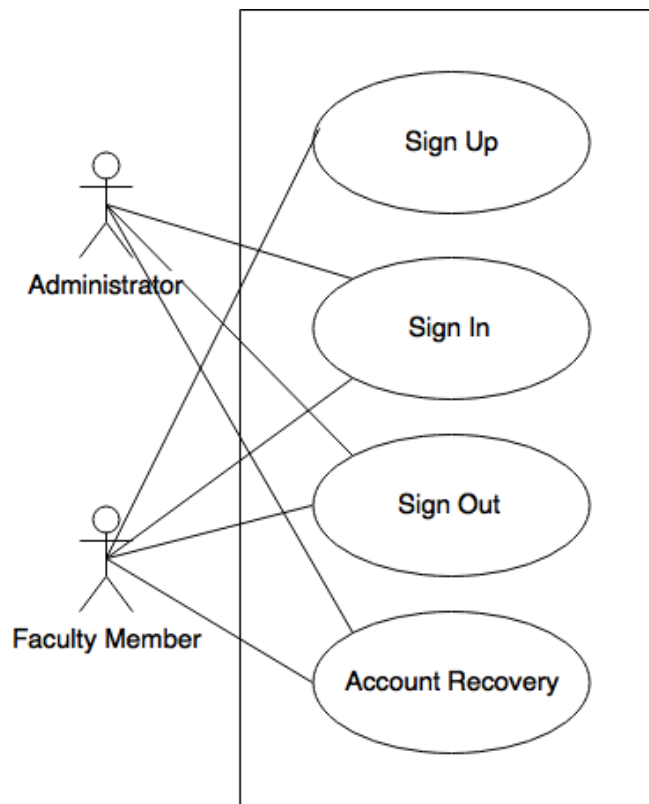


Figure 3: Level-1.1 Use Case Diagram

#### Sign Up:

- **Primary Actor:** Administrator, Faculty Member
- **Secondary Actor:** NA

**Action:** User prompt for sign up and provide related data.

**Reply:** System Check Data Validity. If provide data is valid then system show sign up successful.

#### Sign In/Sign Out:

- **Primary Actor:** Administrator, Faculty Member
- **Secondary Actor:** NA

**Action:** User will enter whether she wants to sign in or sign out. If she wants to sign then she should enter her username and password.

**Reply:** System will receive data show whether the entered data is valid or not Based On valid Data the system will permit the user to sign in.

## Account Recovery:

- **Primary Actor:** Administrator, Faculty Member
- **Secondary Actor:** NA

**Action:** The User will request for account recovery

**Reply:** The system will send verification code to user

**Action:** The user enters verification code

**Reply:** system check entered verification code is valid or not if then she will permit to change her account information

## 4.3.4 LEVEL-1.2 USE CASE: MARK MAINTENANCE

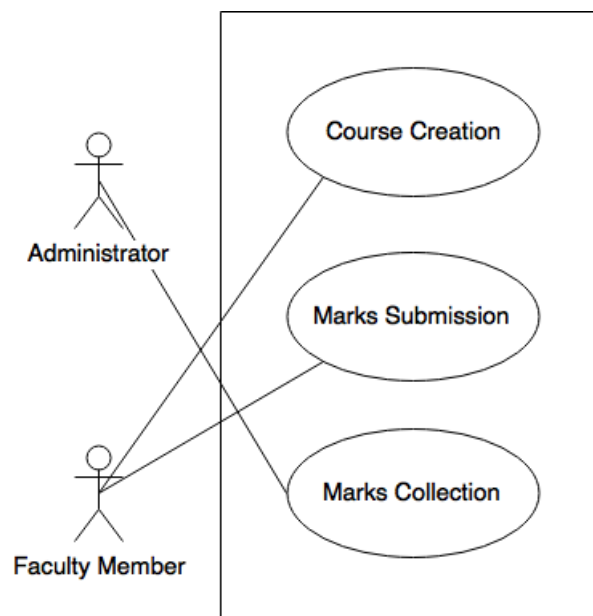


Figure 4: Level-1.2 Use Case Diagram

## Course creation:

- **Primary actor:** Faculty member
- **Secondary actor:** Administrator

**Action:** Faculty member will request to create course.

**Reply:** System will ask course related information and save to database.

### **Marks submission:**

- **Primary actor:** Faculty member
- **Secondary actor:** Administrator

**Action:** Faculty member will submit course mark

**Reply:** System will store that mark

### **Marks collection:**

- **Primary actor:** Faculty member
- **Secondary actor:** Administrator

**Action:** In single examiner system administrator will request for course mark after deadline is over.

**Reply:** System will provide requested mark.

**Action:** In dual examiner system administrator will request for course mark after deadline is over

**Reply:** If the marks of two examiners differ by less than 20%, average of these two will be provide. Else a third examiner will have to submit the marks evaluating the answer scripts again. The average marks of third examiner and the examiner whose mark is closer to the third examiner will be provide.

### 4.3.5 LEVEL-1.3 USE CASE: RESULT CALCULATION AND PUBLICATION

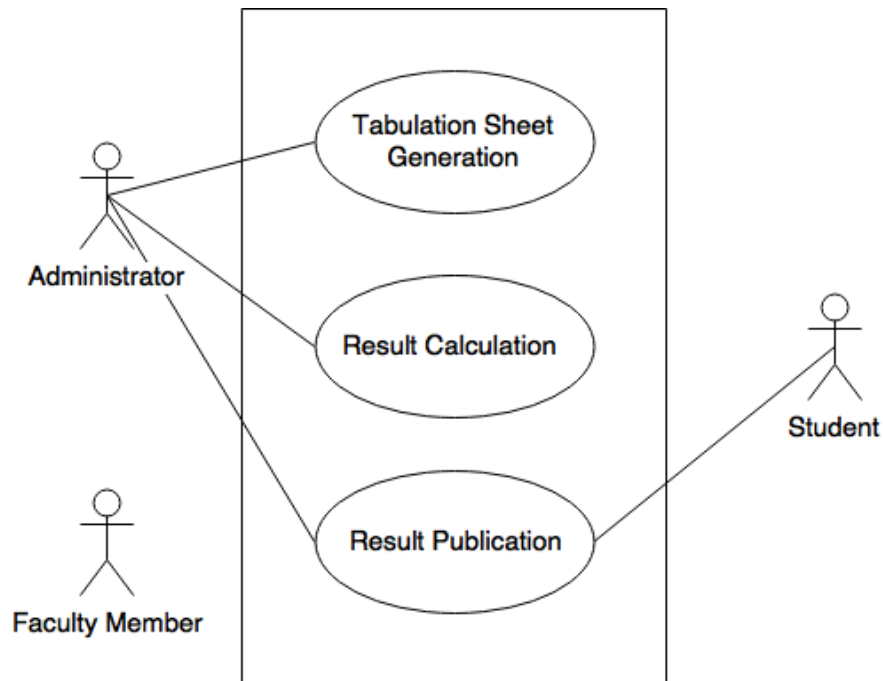


Figure 5: Level-1.3 Use Case Diagram

#### Tabulation Sheet Generation

- **Primary Actor:** Administrator
- **Secondary Actor:** Faculty Member

**Action:** Administrator will request to generate tabulation sheet.

**Reply:** System will generate Tabulation sheet according to provided information.

#### Result Calculation

- **Primary Actor:** Administrator
- **Secondary Actor:** NA

**Action:** The administrator will select individual year/semester to calculate the result.

**Reply:** The system will calculate the result for every student enrolled in that semester or year.

#### Result Publication

- **Primary Actor:** Administrator
- **Secondary Actor:** Faculty Member, Student

**Action:** The administrator will provide a specific date for result publication.

**Replay:** The result will be published on that date and will be available to students.

**Action:** Student will provide required data to view result.

**Replay:** If provided data is valid then student can view her result.

## 4.4 ACTIVITY DIAGRAMS

### 4.4.1 LEVEL-1.1.1 ACTIVITY DIAGRAM: SIGNUP

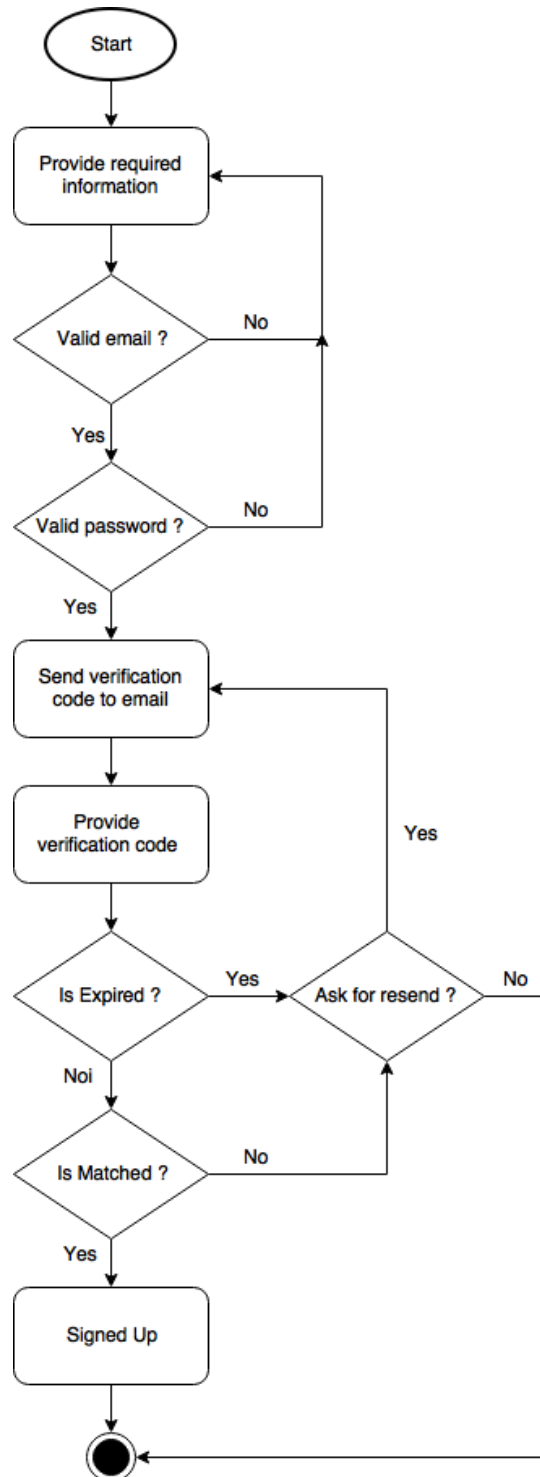


Figure 6: Level-1.1.1 Activity Diagram

#### 4.4.2 LEVEL-1.1.2 ACTIVITY DIAGRAM: SIGN IN

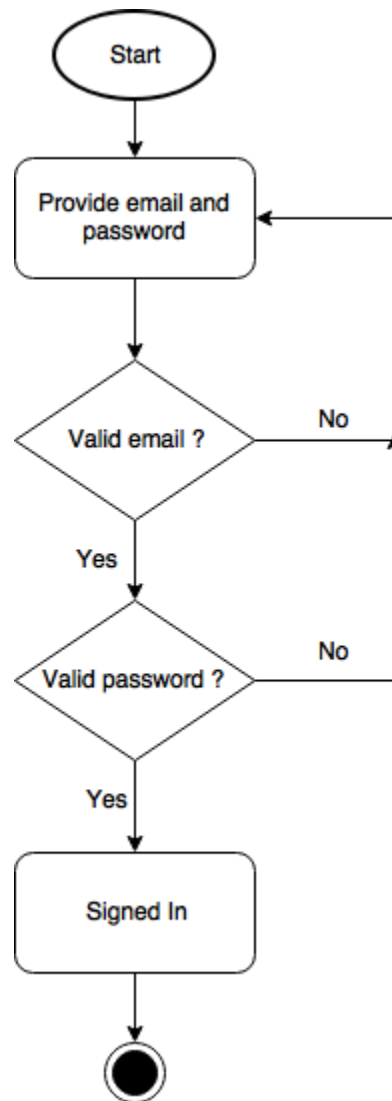


Figure 7: Level-1.1.2 Activity Diagram

### 4.4.3 LEVEL-1.1.3 ACTIVITY DIAGRAM: SIGN OUT

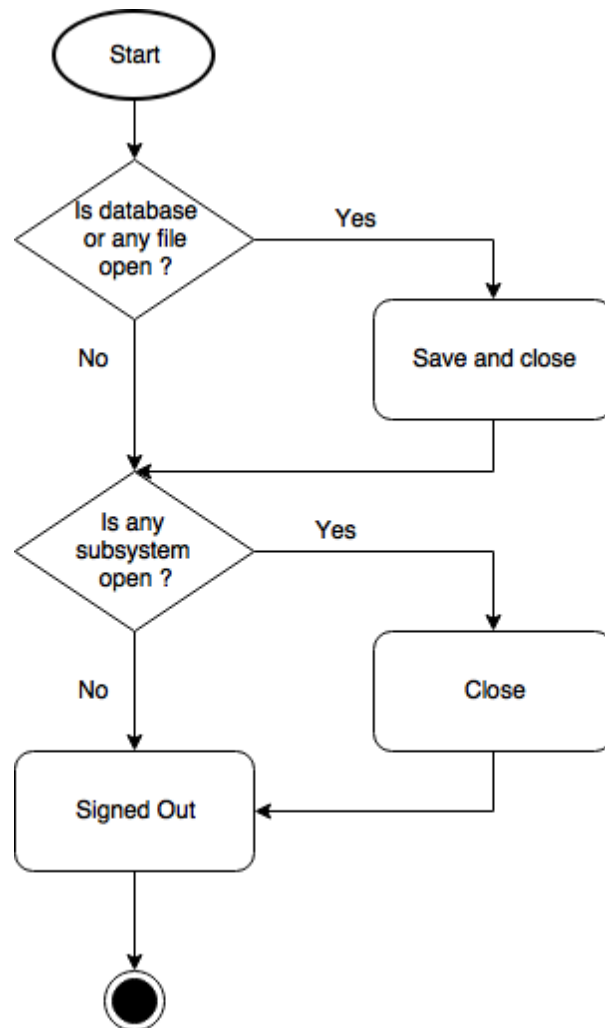


Figure 8: Level-1.1.3 Activity Diagram



#### 4.4.4 LEVEL-1.1.4 ACTIVITY DIAGRAM: ACCOUNT RECOVERY

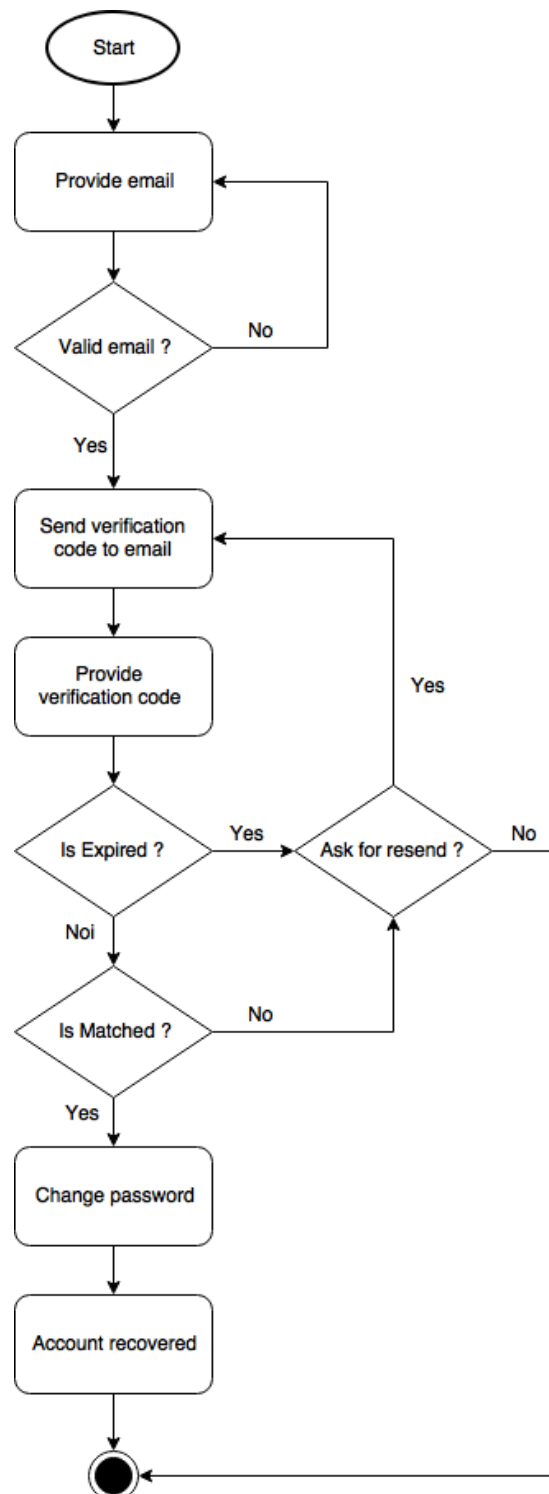


Figure 9: Level-1.1.4 Activity Diagram

#### 4.4.5 LEVEL-1.2.1 ACTIVITY DIAGRAM: COURSE CREATION

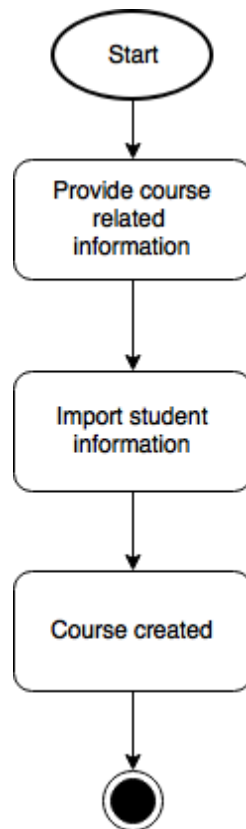


Figure 10: Level-1.2.1 Activity Diagram

#### 4.4.6 LEVEL-1.2.2 ACTIVITY DIAGRAM: MARK SUBMISSION

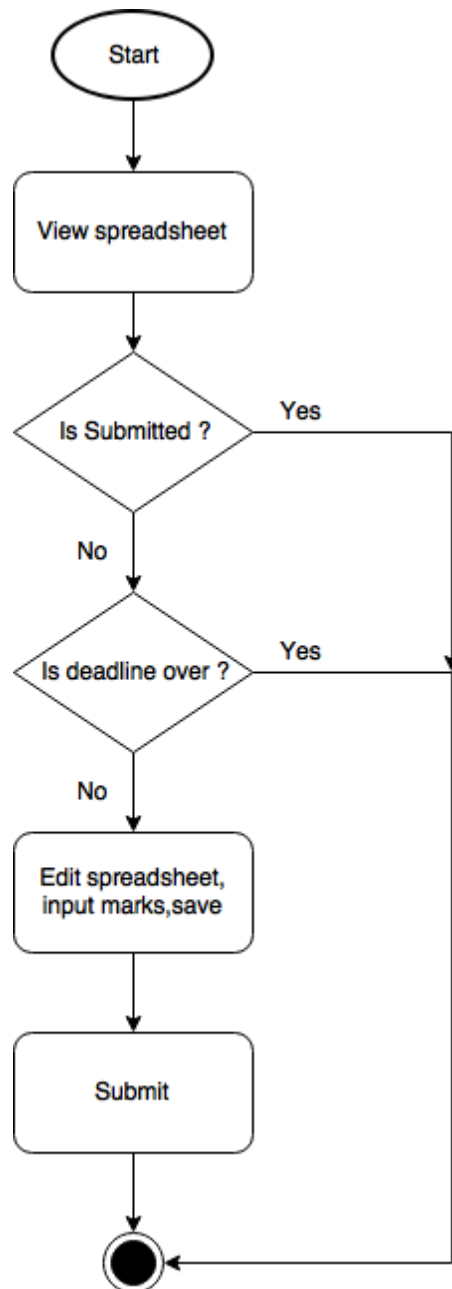


Figure 11: Level-1.2.2 Activity Diagram

#### 4.4.7 LEVEL-1.2.3 ACTIVITY DIAGRAM:MARK COLLECTION

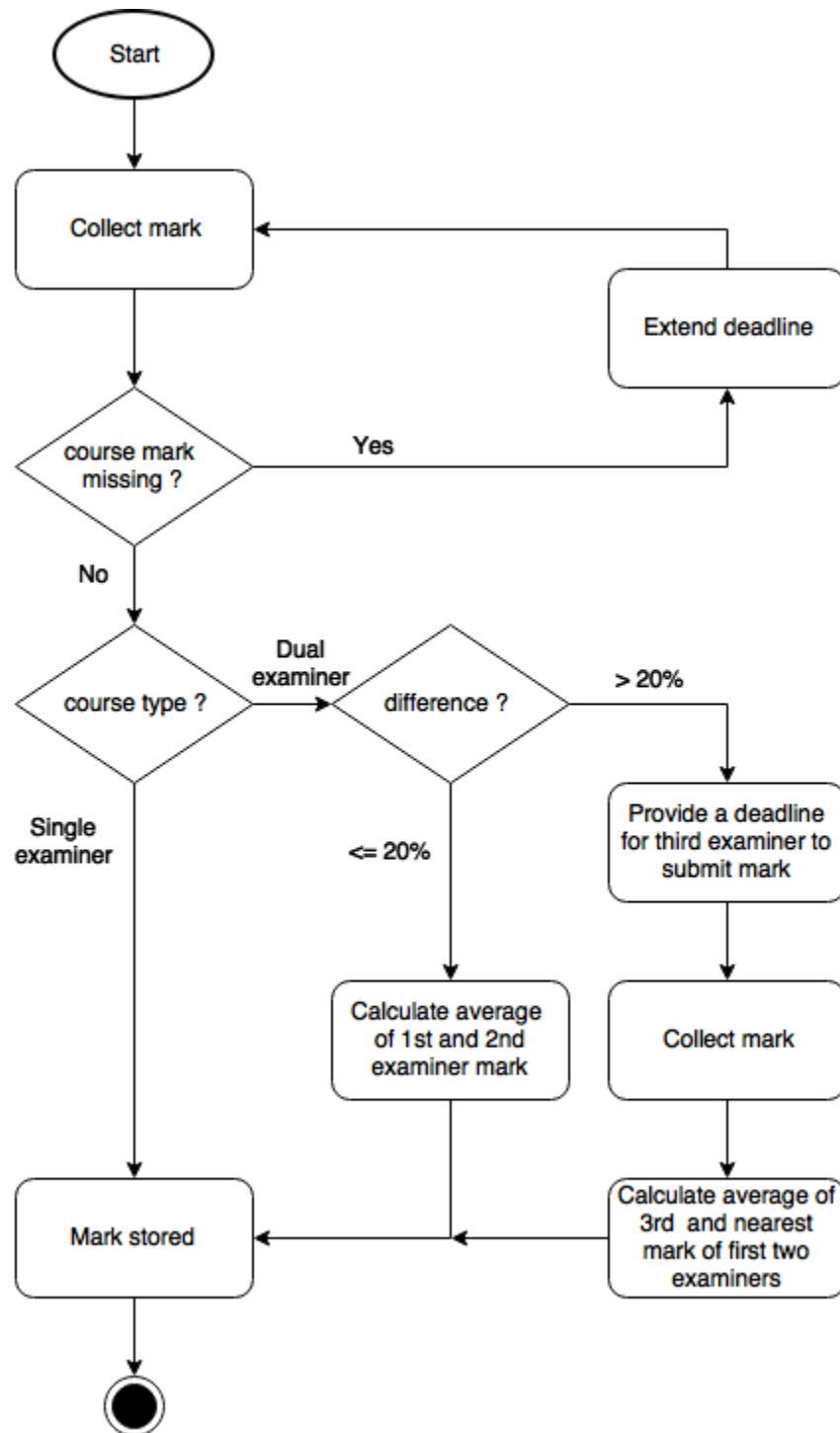


Figure 12: Level-1.2.3 Activity Diagram

#### 4.4.8 LEVEL-1.3.1, 1.3.2 ACTIVITY DIAGRAM: TABULATION SHEET GENERATION AND RESULT CALCULATION

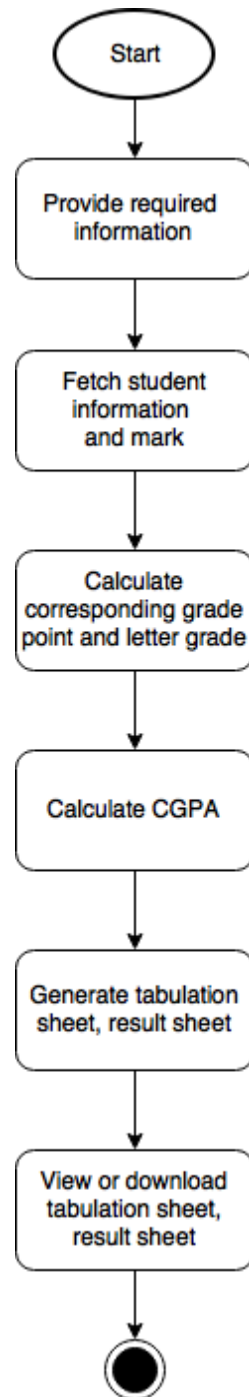


Figure 13: Level-1.3.1, 1.3.2 Activity Diagram

#### 4.4.9 LEVEL-1.3.3 ACTIVITY DIAGRAM: RESULT PUBLICATION AND RESULT SHOWING

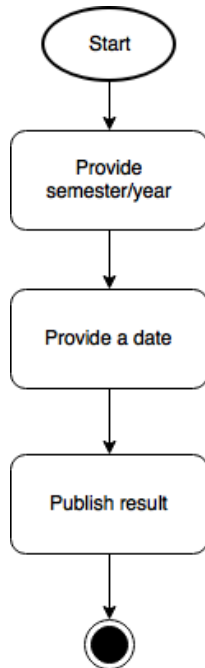


Figure 14: Level-1.3.3 Activity Diagram

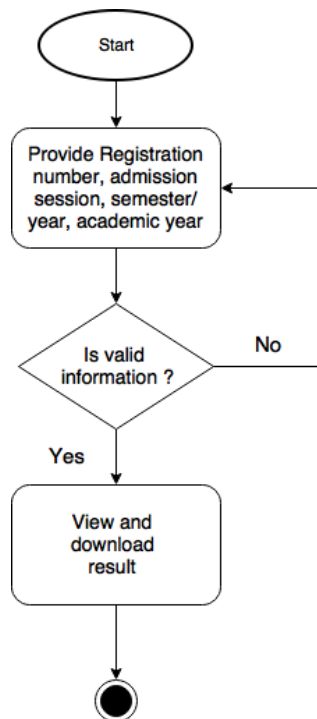


Figure 15: Level-1.3.3 Activity Diagram

## 4.5 SWIMLANE DIAGRAMS

### 4.5.1 LEVEL-1.1.1 SWIMLANE DIAGRAM: SIGNUP

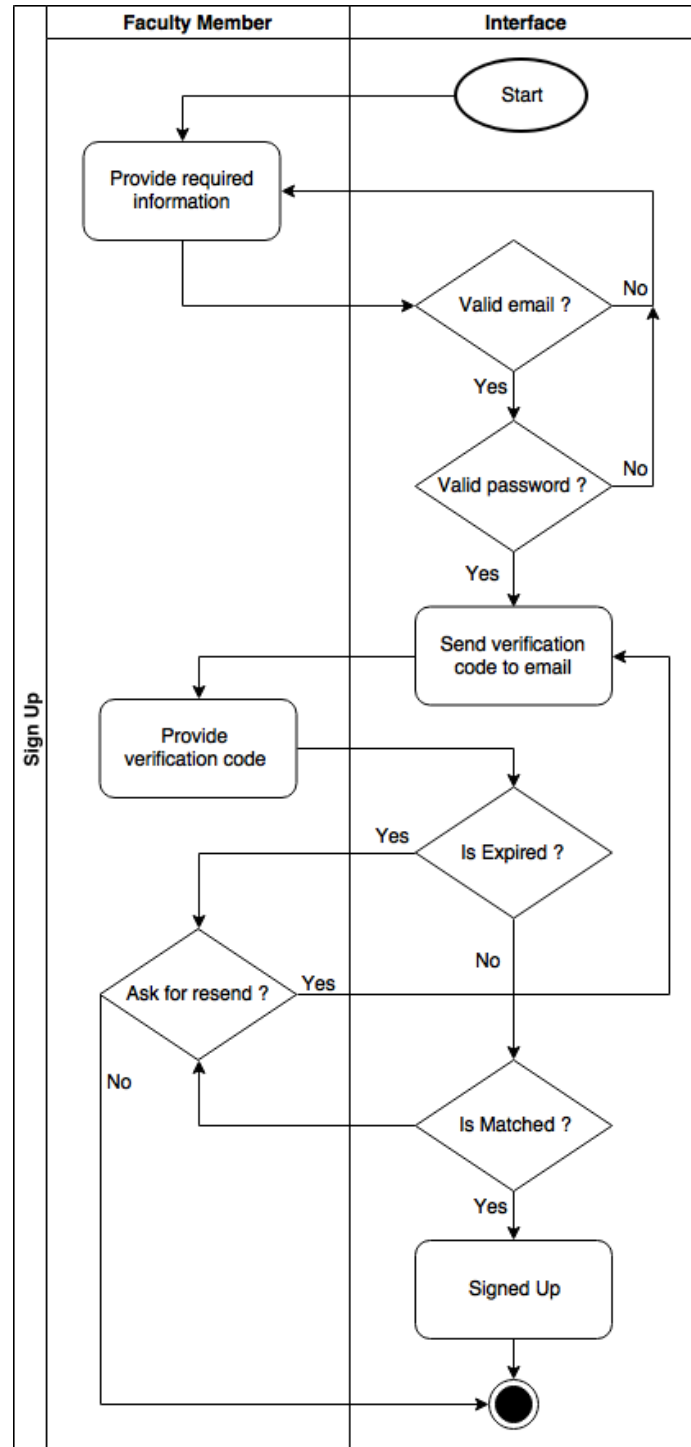


Figure 16: Level-1.1.1 Swimlane Diagram

## 4.5.2 LEVEL-1.1.2 SWIMLANE DIAGRAM: SIGN IN

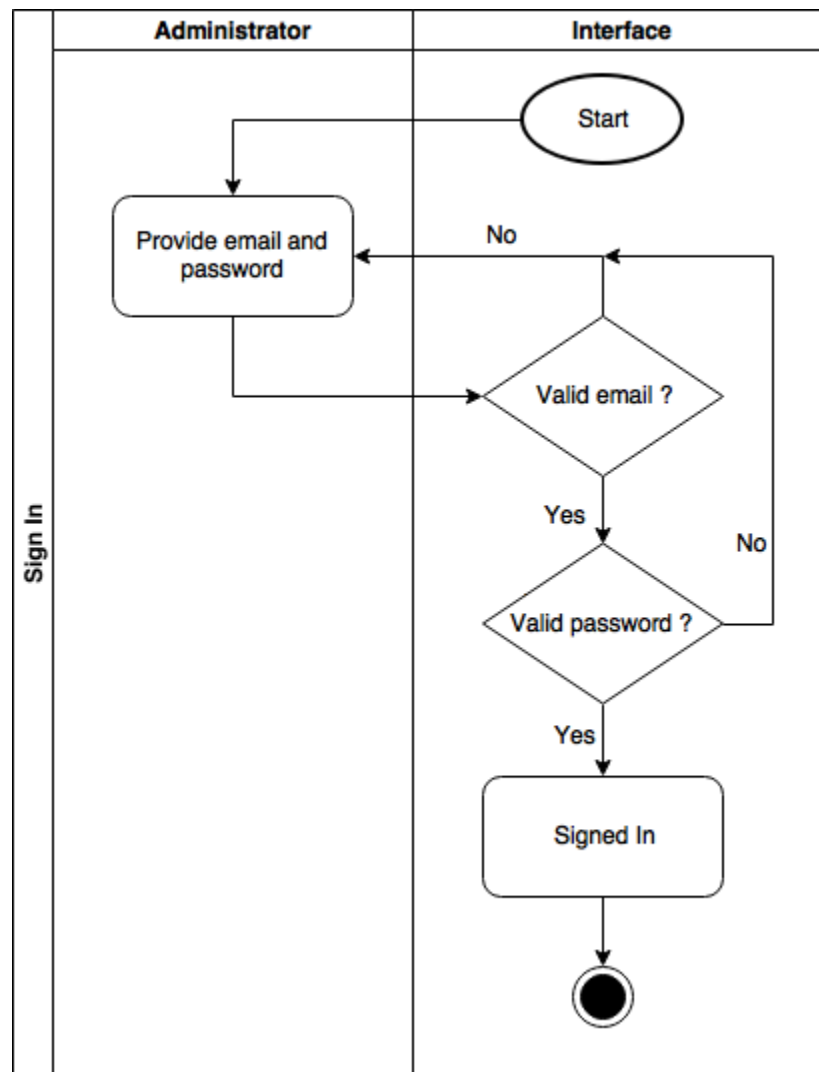


Figure 17 Level-1.1.2 Swimlane Diagram



### 4.5.3 LEVEL-1.1.3 SWIMLANE DIAGRAM: SIGN OUT

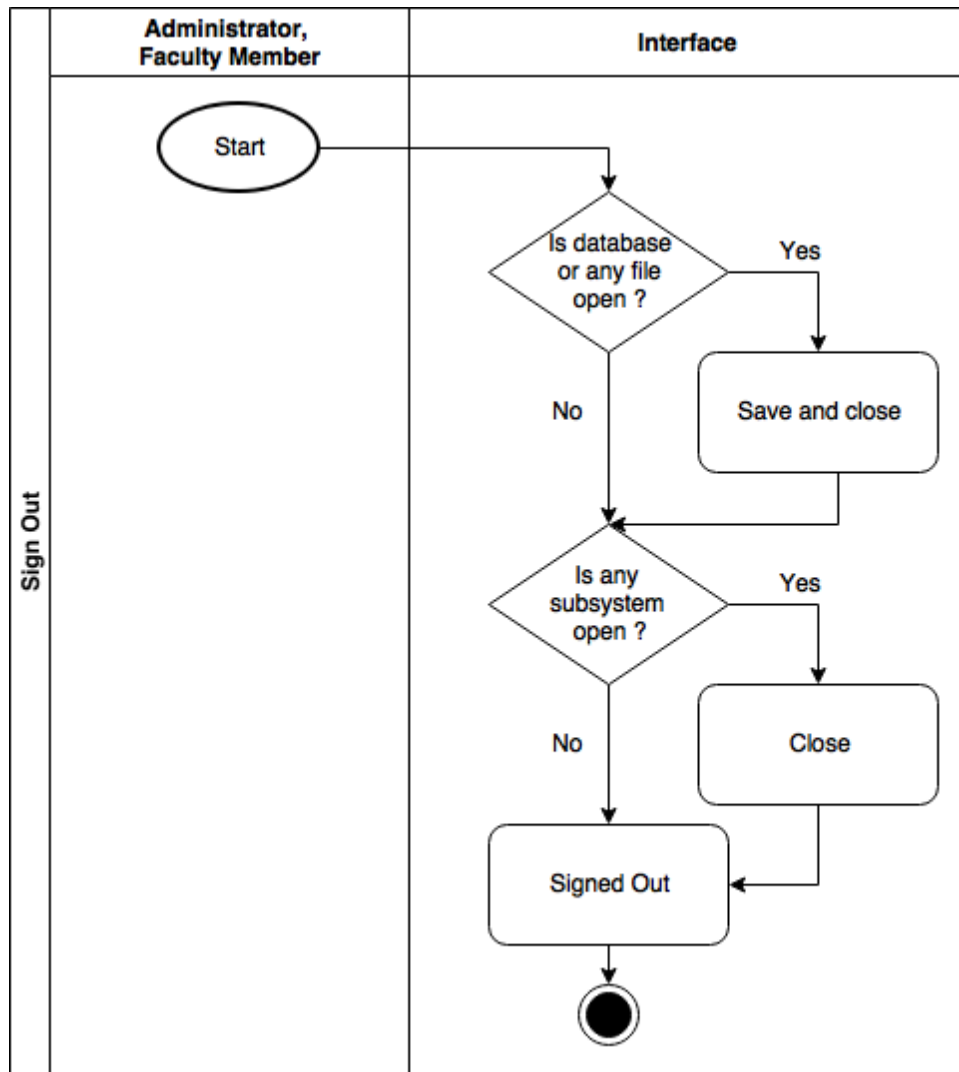


Figure 18: Level-1.1.3 Swimlane Diagram

#### 4.5.4 LEVEL-1.1.4 SWIMLANE DIAGRAM: ACCOUNT RECOVERY

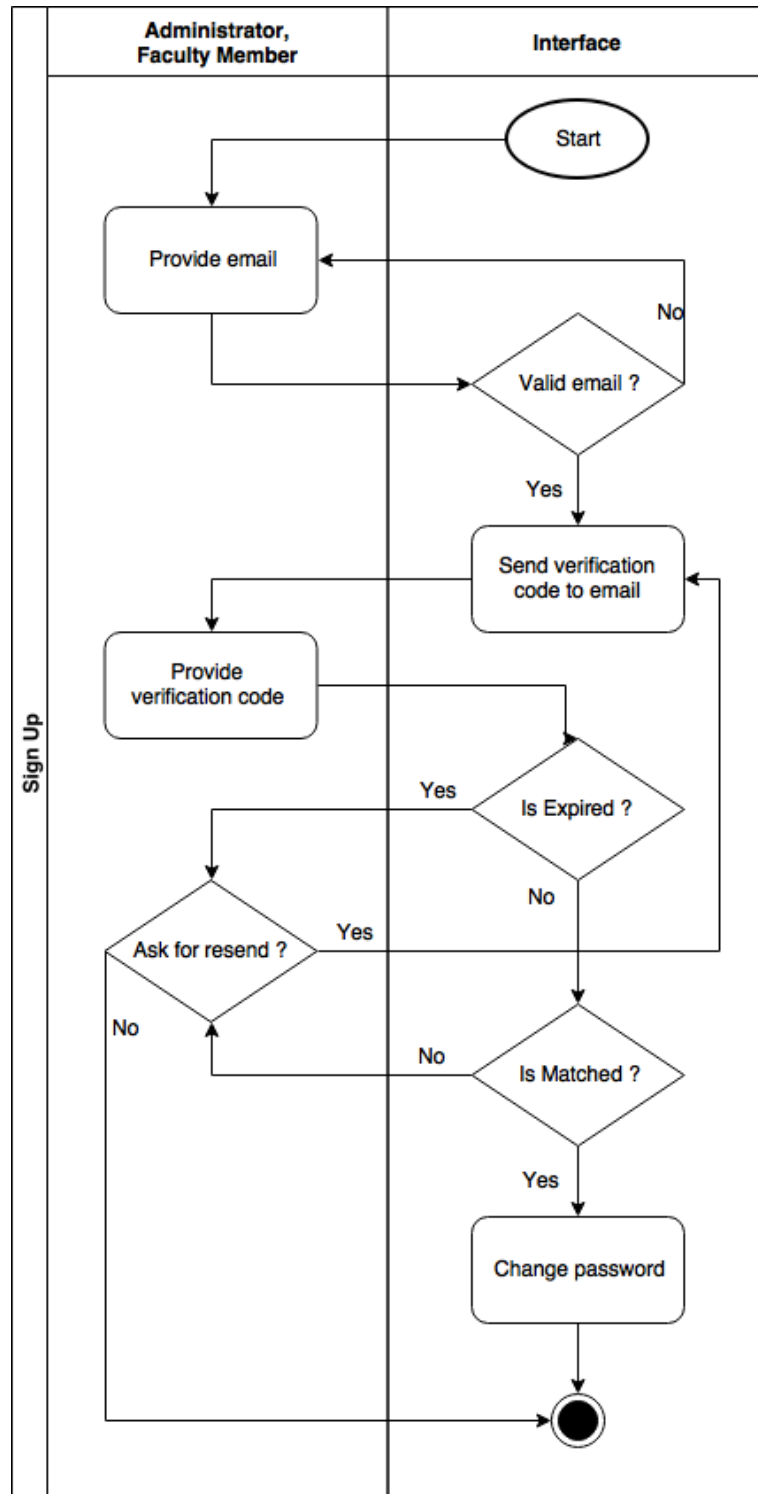


Figure 19: Level-1.1.4 Swimlane Diagram

#### 4.5.5 LEVEL-1.2.1 SWIMLANE DIAGRAM: COURSE CREATION

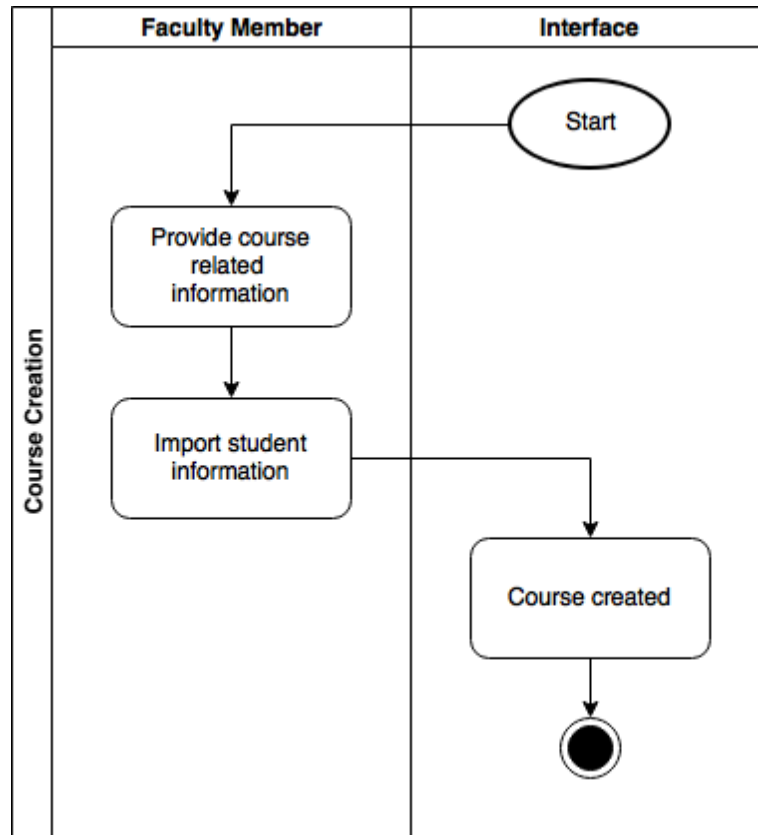


Figure 20: Level-1.2.1 Swimlane Diagram

#### 4.5.6 LEVEL-1.2.2 SWIMLANE DIAGRAM: MARK SUBMISSION

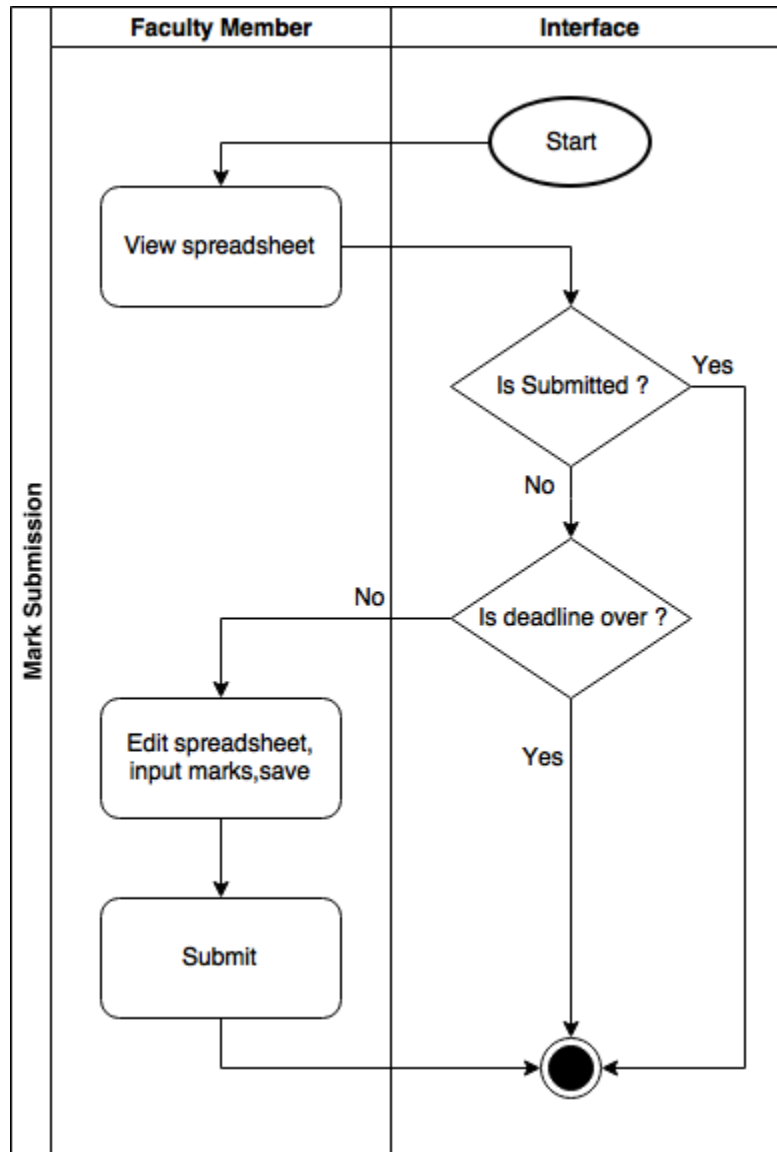


Figure 21: Level-1.2.2 Swimlane Diagram

## 4.5.7 LEVEL-1.2.3 SWIMLANE DIAGRAM: MARK COLLECTION

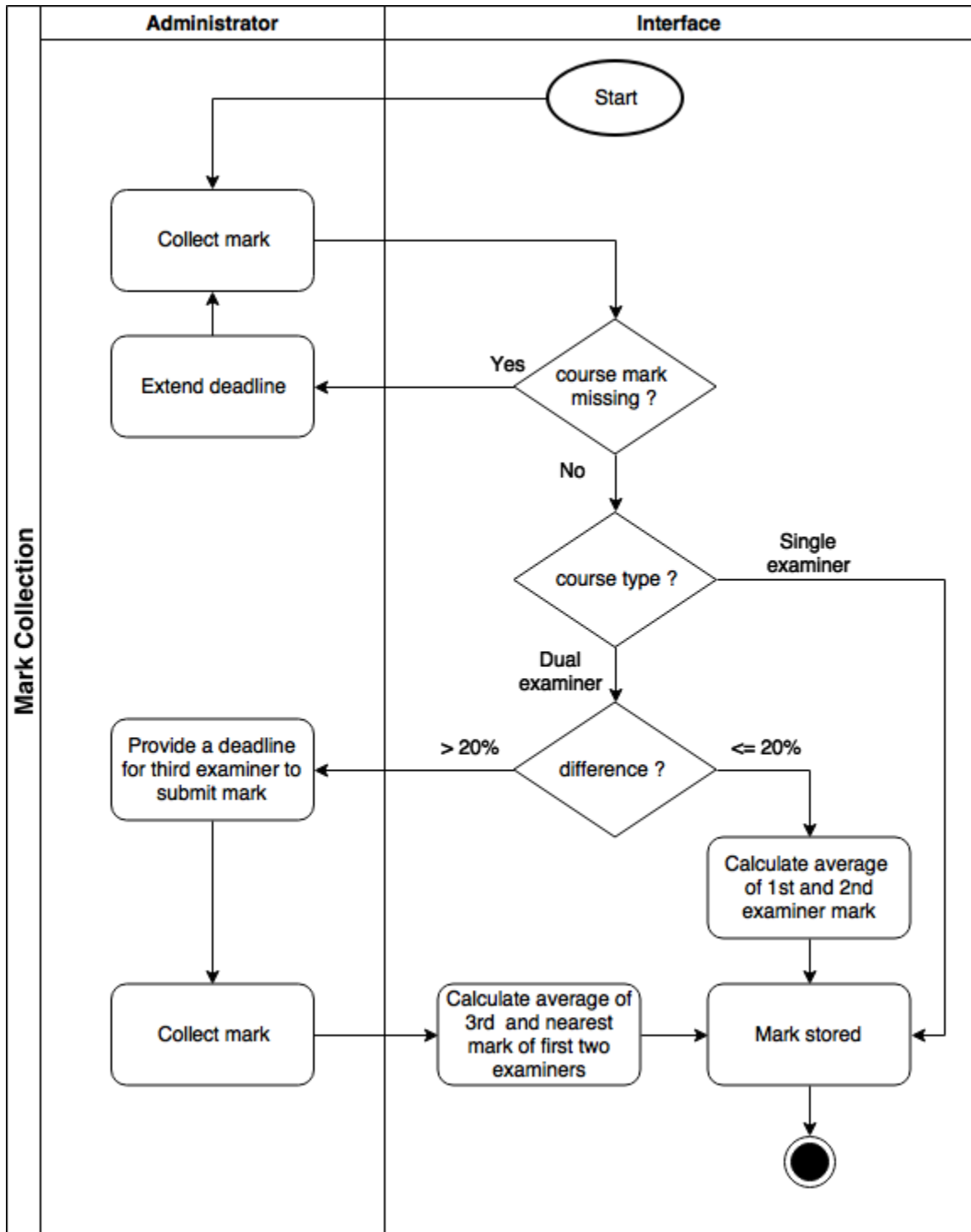


Figure 22: Level-1.2.3 Swimlane Diagram

#### 4.5.8 LEVEL-1.2.1, 1.2.2 SWIMLANE DIAGRAM: TABULATION SHEET GENERATION AND RESULT CALCULATION

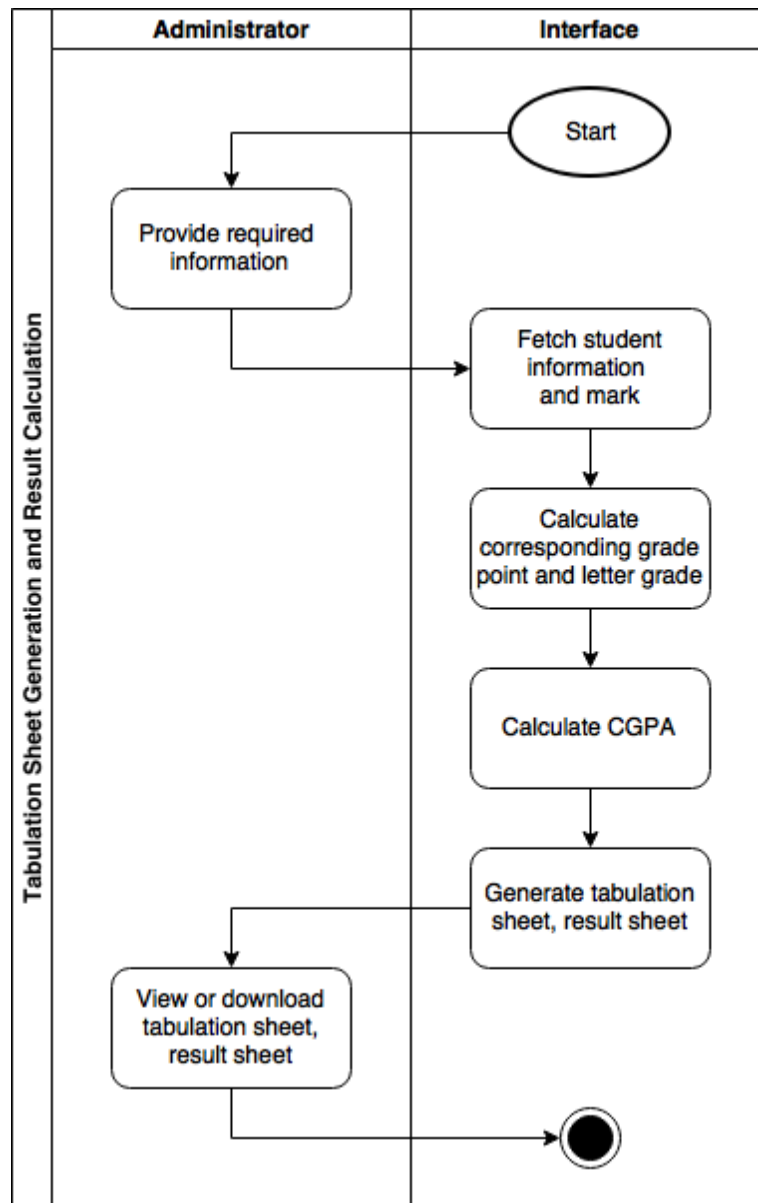


Figure 23: Level-1.3.1, 1.3.2 Swimlane Diagram

#### 4.5.9 LEVEL-1.2.1 SWIMLANE DIAGRAM: RESULT PUBLICATION AND RESULT SHOWING

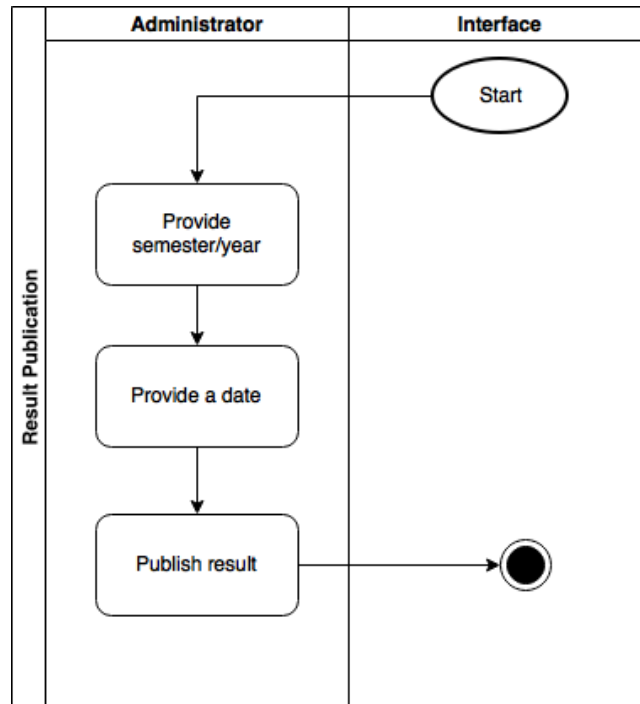


Figure 24: Level-1.3.3 Swimlane Diagram

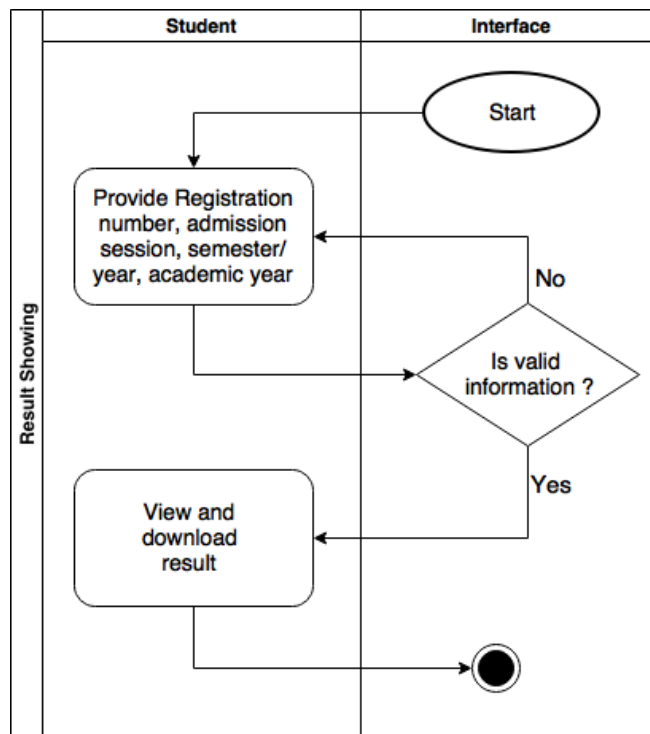


Figure 25: Level-1.3.3 Swimlane Diagram

# CHAPTER 5: DATA-BASED MODEL

This chapter describes the Scenario Based Model for the Result Management System.

## 5.1 INTRODUCTION

Sometimes software requirements include the necessity to create, extend or interact with a database or complex data structures need to be constructed and manipulated. The software team chooses to create data models as a part of overall requirements modelling. The entity-relationship diagram (ERD) defines all data objects that are processed within the system, the relationships between the data objects and the information about how the data objects are entered, stored, transformed and produced within the system.

## 5.2 DATA OBJECTS

A data object is a representation of composite information that must be understood by the software. Here, composite information means an information that has a number of different properties or attributes. A data object can be an external entity, a thing, an occurrence, a role, an organizational unit, a place or a structure.

### 5.2.1 NOUN IDENTIFICATION

We identified all the nouns whether they are in problem space or in solution space from our usage scenario.

Table 2: Noun identification for data modeling

No.	Noun	Problem/Solution	Attributes
1	faculty member	S	3,4,5,6,8,9
2	sign up	S	
3	Faculty Name	S	
4	Designation	S	
5	Faculty	S	
6	Department	S	



7	Institution	P	
8	Email Address	S	
9	Password	S	
10	verification code	S	
11	five minutes	P	
12	user	S	
13	administrator	S	15,9
14	sign in	S	
15	username	S	
16	sign out	S	
17	Account recovery	S	
18	account	P	
19	creation	P	
20	Course Title	S	
21	Course Code	S	
22	Course mode	S	
23	Credit	S	
24	Semester	S	
25	Year	S	
26	Academic Year	P	
27	Role	S	
28	first examiner	S	
29	Student Name	S	
30	second examiner	S	
31	Registration Number	S	
32	Admission Session	S	

33	Class Roll	S	
34	Exam Roll	S	
35	Current Semester/Year	S	
36	submission	P	
37	Attendance	S	
38	Assignment	S	
39	Class Test	S	
40	Midterm/In-Course	S	
41	Lab Examination	S	
42	Total Continuous Evaluation	S	
43	Final Examination	S	
44	Total Marks	S	
45	column	P	
46	category	P	
47	marks	S	
48	log	S	77,78
49	spreadsheet	S	37,38,39,40,41,42,43,44
50	deadline	S	
51	course	S	20,21,22,23,24,25
52	departmental head	P	
53	director	P	
54	third examiner	S	
55	answer script	P	
56	tabulation sheet	S	20,21,24,25,29,33,34,57

57	Grade point	S	
58	Cumulative Grade Point	S	
59	student	S	29,31,32,33,34,35,57,58
60	expiry time	S	
61	correct code	P	
62	information	P	
63	option	S	
64	system	P	
65	CSV file	S	
66	field	P	
67	evaluation	P	
68	average mark	S	
69	calculation	S	
70	current semester	P	
71	previous semester	P	
72	authentication	S	
73	individual semester/year	P	
75	academic year	S	
76	result	P	
77	Log ID	S	
78	Log Content	S	
79	input	S	
80	weight	S	
81	approval	P	

## **5.2.2 POTENTIAL DATA OBJECTS**

1. Course: 20-25
2. Administrator: 9,15
3. Student: 29-35,57,58
4. Faculty Member: 3-9
5. Spreadsheet: 37-44
6. Tabulation Sheet: 20,21,24,25,29,33,34,57
7. Log: 77,78

## **5.2.4 FINAL DATA OBJECTS**

1. Course: 20-25
2. Administrator: 9,15
3. Student: 29-35,57,58
4. Faculty Member: 3-9
5. Log: 77,78

## 5.3 ENTITY RELATIONSHIP DIAGRAM

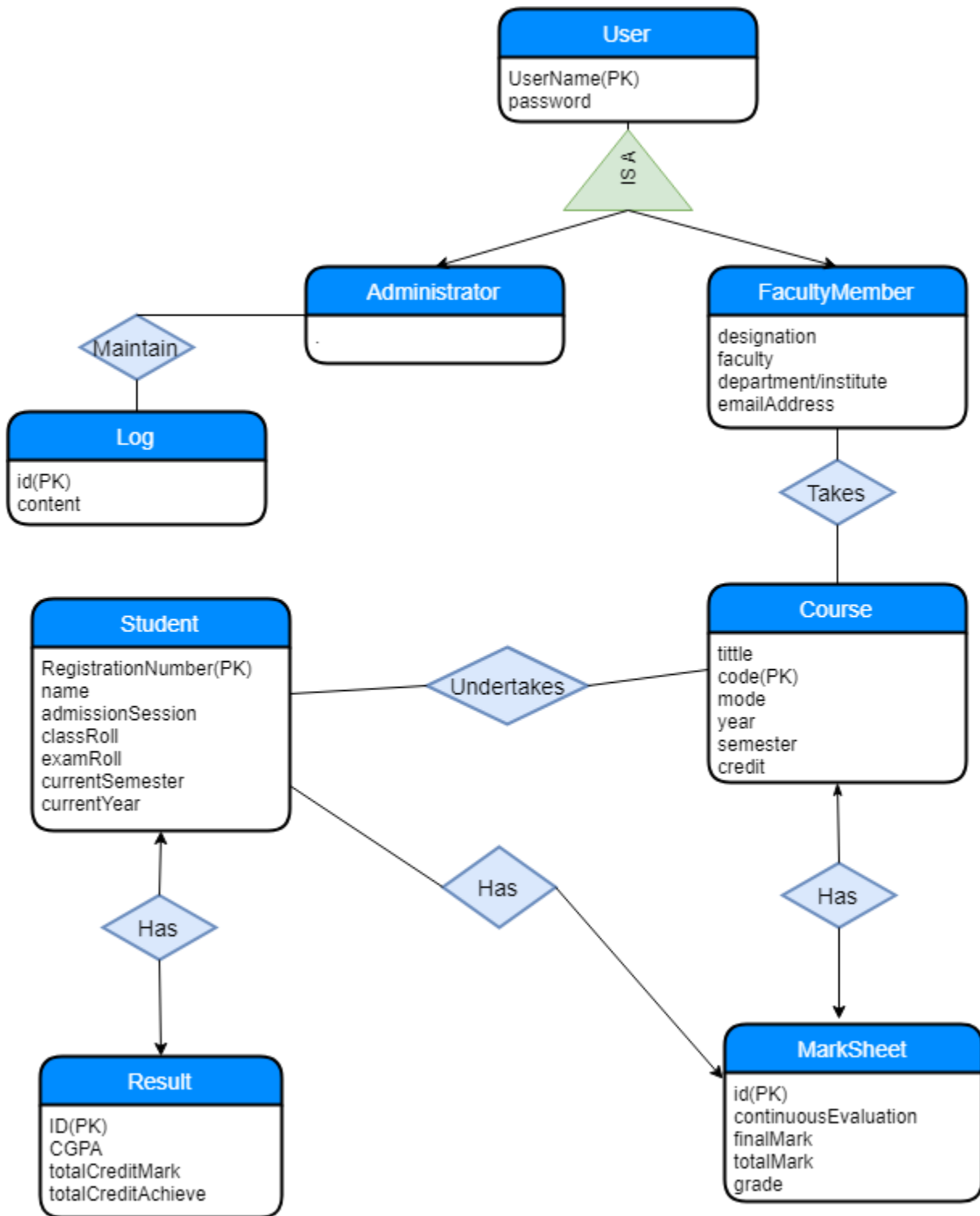


Figure 26: ER Diagram

## 5.4 SCHEMA DIAGRAM

Table 3: Schema table of User

User		
Attributes	Type	Size
Username	Varchar	80
Password	Varchar	17

Table 4: Schema table of Administrator

Administrator		
Attributes	Type	Size
Username	Varchar	80
Password	Varchar	17

Table 5: Schema table of Faculty Member

Faculty member		
Attributes	Type	Size
Name	Varchar	80
Designation	Varchar	20
Faculty		
Department/institute	Varchar	30
Emailaddress	Varchar	40
Password	Varchar	17

Table 6: Schema table of Student

<b>Student</b>		
<b>Attributes</b>	<b>Type</b>	<b>Size</b>
Name	Varchar	80
Registrationnumber	Varchar	20
Admissionsession	Varchar	15
Classroll	Varchar	15
Examroll	Varchar	15
Currentsemester	Number	
Currentyear	Number	

Table 7: Schema table of Log

<b>Log</b>		
<b>Attributes</b>	<b>Type</b>	<b>Size</b>
Id	Number	20
Content	Varchar	150

Table 8: Schema table of Course

<b>Course</b>		
<b>Attributes</b>	<b>Type</b>	<b>Size</b>
Title	Varchar	80
Code	Varchar	20
Mode	Boolean	
Year	Number	
Semester	Number	
Credit	Number	

Table 9: Schema table of TakeCourse

Takecourse		
Attributes	Type	Size
Username	Varchar	80
Courseid	Varchar	20

Table 10: Schema table of Maintain

Maintain		
Attributes	Type	Size
Logid	Number	
Username	Varchar	80

Table 11: Schema table of UnderTakenCourse

Undertakencourse		
Attributes	Type	Size
Coursecode	Varchar	20
Registrationnumber	Varchar	20

Table 12: Schema table of Marksheet

Marksheet		
Attributes	Type	Size
Courseid	Varchar	20
Marksheetid	Varchar	20
Continuousevaluation	Real	
Finalmark	Real	



Totalmark	Real	
Grade	Real	

Table 13: Schema table of Result

Result		
Attributes	Type	Size
Registrationnumber	Varchar	20
Id	Number	
Totalcredittaken	Real	
Totalcreditachieve	Real	
CGPA	Real	

Table 14: Schema table of TakeCourse Has

Has		
Attributes	Type	Size
Marksheet id	Varchar	20
Registrationnumber	Varchar	20

# CHAPTER6: CLASS-BASED MODEL

This chapter describes the Class Based Model for the Result Management System.

## 6.1 INTRODUCTION

Class-based methods for requirements modelling use common concepts of object oriented programming to craft an impression of an application that can be understood by nontechnical stakeholders. As the requirements model is refined and expanded, it evolves into a specification that can be used by software engineers in the creation of the software design. Class-based modelling represents:

1. The objects the system will manipulate
2. The operations (methods or services) that will be applied for effective manipulation
3. The relationships between the objects
4. The collaborations that occur between the classes

## 6.2 IDENTIFYING ANALYSIS CLASSES

Classes are identified by underlining each noun or noun phrase and plotting it into a simple table. If the class (noun) is required to implement a solution, then it becomes a part of the solution space. Otherwise if the noun is used only to describe a solution, it is regarded as a part of the problem space. Once all the nouns have been isolated, General classification and Selection is done.

### 6.2.1 GENERAL CLASSIFICATION

Nouns belonging to the solution space should exhibit any of the following criteria to be considered as a class. The 7 general characteristics are stated below.

1. **External entities:** Other systems, devices, people that produce or consume information to be used by a computer-based system
2. **Things:** Reports, displays, letters, signals that are a part of the information domain for the problem.
3. **Events:** Actions or transfers (a property transfer or the completion of a series of robot movements) that occur within the context of system operation.
4. **Roles:** Responsibilities played by the people who interact with the system.
5. **Organizational units:** Divisions, groups, teams that are relevant to an application.
6. **Places:** Platform that establishes the context of the problem and overall function of the system.
7. **Structures:** Something that defines a class of objects or related classes of objects.

Table 15: General Classification

No.	Noun	GC
1	Faculty member	2,4,5
2	Sign up	3
3	Faculty name	
4	Designation	
5	Faculty	
6	Department	
7	Institution	
8	Email address	
9	Password	
10	Verification code	
11	Five minutes	
12	User	2,4,5
13	Administrator	2,4,5
14	Sign in	3
15	Username	
16	Sign out	3
17	Account recovery	3
18	Account	
19	Creation	3
20	Course title	
21	Course code	
22	Course mode	
23	Credit	
24	Semester	

25	Year	
26	Academic year	
27	Role	
28	First examiner	2,4,5
29	Student name	
30	Second examiner	2,4,5
31	Registration number	
32	Admission session	
33	Class roll	
34	Exam roll	
35	Current semester/year	
36	Submission	3
37	Attendance	2
38	Assignment	2
39	Class test	2
40	Midterm/in-course	2
41	Lab examination	2
42	Total continuous evaluation	2
43	Final examination	2
44	Total marks	
45	Column	
46	Category	
47	Marks	
48	Log	2,7
49	Spreadsheet	2,6

50	Deadline	
51	Course	2,7
52	Departmental head	
53	Director	
54	Third examiner	2,4,5
55	Answer script	
56	Tabulation sheet	2,6
57	Grade point	
58	Cumulative grade point	
59	Student	1,7
60	Expiry time	
61	Correct code	
62	Information	
63	Option	2
64	System	
65	CSV file	1
66	Field	
67	Evaluation	3
68	Average mark	
69	Calculation	3
70	Current semester	
71	Previous semester	
72	Authentication	3
73	Individual semester/year	
75	Academic year	
76	Result	

77	Log ID	
78	Log content	

## 6.2.2 SELECTION CRITERIA

Classes that fulfilled at least 3 characteristics of general classification are again reconsidered by six Selection Criteria. The six characteristics for the selection criteria are:

1. **Retained information:** The potential class will be useful during analysis only if information about it must be remembered so that the system can function.
2. **Needed services:** The potential class must have a set of identifiable operations that can change the value of its attributes in some way.
3. **Multiple attributes:** During requirement analysis, the focus should be on “major” information; a class with a single attribute may, in fact, be useful during design, but is probably better represented as an attribute of another class during the analysis activity.
4. **Common attributes:** A set of attributes can be defined for the potential class and these attributes apply to all instances of the class.
5. **Common operations:** A set of operations can be defined for the potential class and these operations apply to all instances of the class.
6. **Essential requirements:** External entities that appear in the problem space and produce or consume information essential to the operation of any solution for the system will almost always be defined as classes in the requirements model.

To be considered a legitimate class for inclusion in the requirements model, a potential object should satisfy all (or almost all) of these characteristics. The decision for inclusion of potential classes in the analysis model is somewhat subjective, and later evaluation may cause an object to be discarded or reinstated.

Table 16: Selection Criteria

Serial	Noun	Selection Criterion
1	Faculty member	1,2,3,4,5
2	Sign up	
3	User	1,2,3,4,5
4	Administrator	1,2,3,4,5

5	Sign in	
6	Sign out	
7	Account recovery	
8	Creation	
9	First examiner	1,3,4,5
10	Second examiner	1,3,4,5
11	Third examiner	1,3,4,5
12	Submission	
13	Attendance	
14	Assignment	
15	Class test	
16	Midterm/ in-course	
17	Lab examination	
18	Total continuous evaluation	
19	Final examination	
20	Log	3,4,5
21	Spreadsheet	1,2
22	Course	1,2,3,4,5
23	Tabulation sheet	1,2
24	Student	1,3,4,5,6
25	Option	
26	Csv file	6
27	Evaluation	
28	Calculation	
29	Authentication	2,3,4,5

### 6.2.3 ASSOCIATING NOUNS WITH VERBS

Table 17: Associate noun and verb identification

No.	Class Name	Noun	Verb
1	Faculty Member	Name, Designation, Faculty, Department/Institution, Email Address, Password	sign up, sign in, sign out, Account recovery, Role, submit, Course Creation, Evaluation
2	User	User Name, Password	Sign in, sing out
3	Administrator	User Name, Password	Sign in, sign out, Account recovery, Result Calculation, Log Maintain, Deadline set, Result Publication
4	Log	ID, Content	N/A
5	Student	Name, Registration Number, Admission Session, Class Roll, Exam Roll, Current Semester/Year	Calculate GPA, View Result
6	Authentication		Sign in, Account recovery
7	Course	Title, Code, Mode, Credit, Semester, Year	Calculate Grade Point
8	Spreadsheet	Attendance, Assignment, Class Test, Midterm / In-Course, Lab Examination, Total Continuous Evaluation, Final Examination, Total Marks	Calculate Total Mark
9	Tabulation Sheet	Course Title, Course Code, Semester, Year, Student Name, Class Roll, Exam Roll, Grade point	N/A



## 6.2.4 FINAL CLASSES

1. User
2. Faculty Member
3. Administrator
4. Student
5. Course
6. Authentication
7. Tabulation Sheet
8. Spreadsheet
9. Log

## 6.3 CLASS RESPONSIBILITY COLLABORATION CARD

Table 18: CRC Card: User

<b>User</b>	
<b>Responsibilities</b>	<b>Collaborative Class</b>
Sign In	Authentication
Account Recovery	Authentication

Table 19: CRC Card: Administrator

<b>Administrator</b>	
<b>Responsibilities</b>	<b>Collaborative Class</b>
Set Deadline	
Result Calculation	Spreadsheet, Course, Student, FacultyMember
Generate Tabulation Sheet	Course, Student, FacultyMember, TabulationSheet
Result Publication	Student, Course
Log Maintain	Log

Table 20: CRC Card: FacultyMember

<b>FacultyMember</b>	
<b>Responsibilities</b>	<b>Collaborative Class</b>
Course Creation	Course
Sign Up	Authentication
Sign In	Authentication
Recover Account	Authentication
Provide Student Information	Student, Spreadsheet
Provide Course Related Information	Course, Spreadsheet
Course Mark submission	Course, Spreadsheet

Table 21: CRC Card: Authentication

<b>Authentication</b>	
<b>Responsibilities</b>	<b>Collaborative Class</b>
Authenticate	Administrator, FacultyMember
Authorize	Administrator, FacultyMember
Account Recovery	User

Table 22: CRC Card: Course

<b>Course</b>	
<b>Responsibilities</b>	<b>Collaborative Class</b>
Calculate Course Result	
Update Course Information	

Table 23: CRC Card: Student

<b>Student</b>	
<b>Responsibilities</b>	<b>Collaborative Class</b>
Contain Result	Spreadsheet, TabulationSheet
Update Student Information	

Table 24: CRC Card: Log

<b>Log</b>	
<b>Responsibilities</b>	<b>Collaborative Class</b>
Contain Change Information	Spreadsheet

Table 25: CRC Card: Spreadsheet

<b>Spreadsheet</b>	
<b>Responsibilities</b>	<b>Collaborative Class</b>
Contain course marks	Course
Calculate Course Total According to weight	

Table 26: CRC Card: TabulationSheet

<b>TabulationSheet</b>	
<b>Responsibilities</b>	<b>Collaborative Class</b>
Generate Tabulation Sheet	

## 6.4 CLASS DIAGRAM

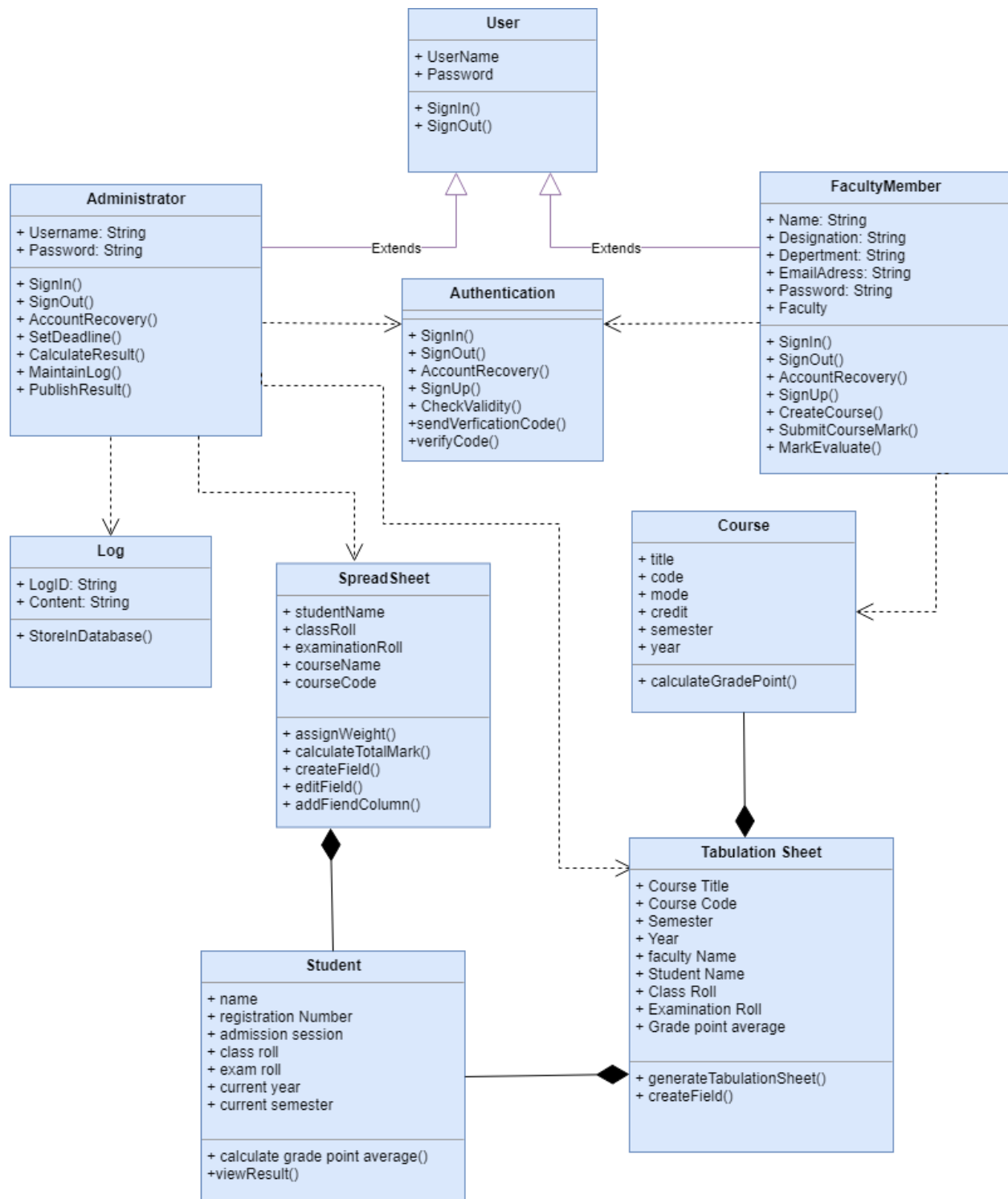


Figure 27: Class Diagram

# CHAPTER7: FLOW-ORIENTED MODEL

## 7.1 DATAFLOW DIAGRAMS

### 7.1.1 LEVEL-0 DATAFLOW DIAGRAM

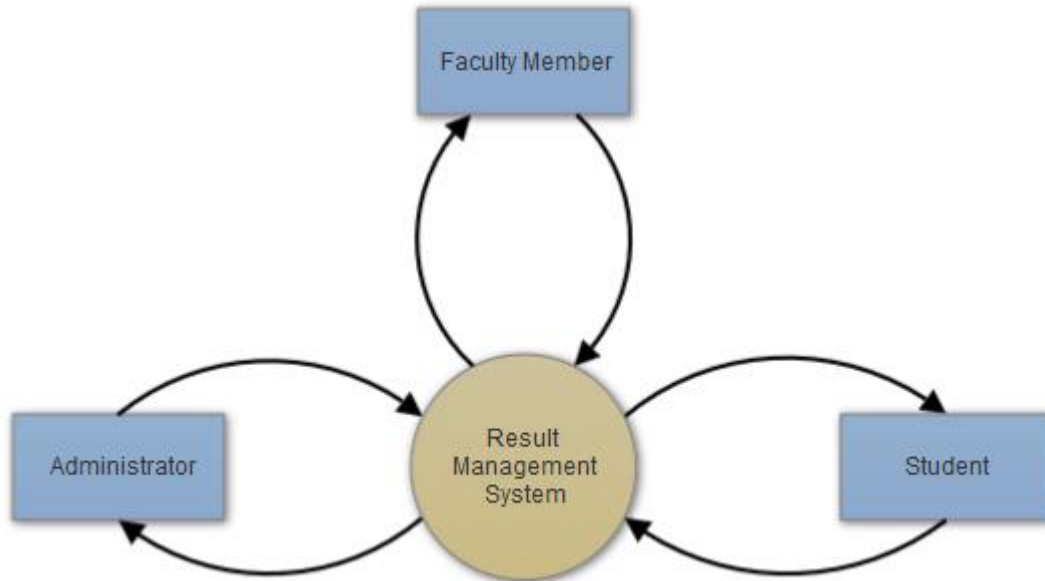


Figure 28: Level-0 Dataflow Diagram

### 7.1.2 LEVEL-0 DATFLOW DIAGRAM

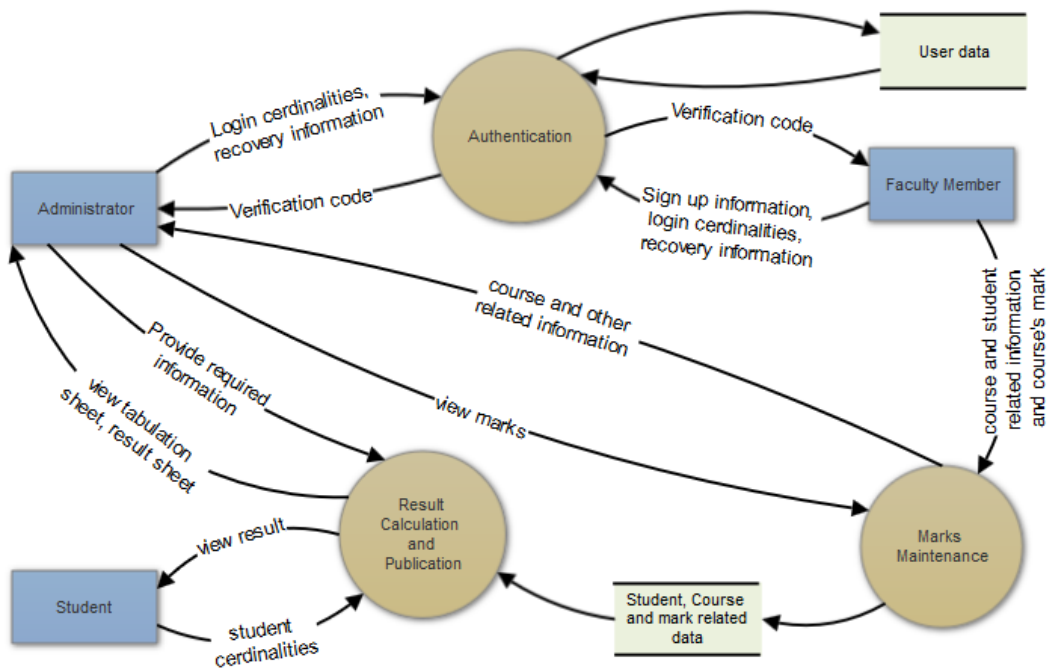


Figure 29: Level-1 Dataflow Diagram

# CHAPTER 8: BEHAVIORAL MODEL

The behavioral model indicates how software will respond to external events or stimuli. This chapter throws light on the ways RMS interacts.

## 8.1 STATE TRANSITION

In the context of behavioral modelling to different characterization of states must be considered and these are:

- The state of each class as the system performs its functions.
- The state of the system observed from the outside as the system performs its functions.

### 8.1.1 EVENT IDENTIFICATION

No.	Events	Primary Object	Collaborator	Invoked Method
1	Sign Up	Authentication	Faculty Member	signUp(), Checkvalidity()
2	Sign In	Authentication	Faculty Member, Administrator	signIn(), checkValidity()
3	Sign Out	User	Faculty Member, Administrator	signOut()
4	Recover Account	Authentication	Faculty Member, Administrator	accountRecovery(), checkValidity()
5	Verify User	Authentication		sendVerificationCode(), verifyCode()
6	Create Course	Faculty Member	Course, Student	createCourse()
7	Edit Mark	Faculty Member	Course, SpreadSheet, Student	editMark()
8	Submit Mark	Faculty Member	Course, SpreadSheet, Student	submitMarks()

9	Set Deadline	Administrator		setDeadline()
10	Maintain Log	Administrator		maintainLog()
11	Calculate Result	Administrator	Course, Student, SpreadSheet	calculateResult()
12	Publish Result	Administrator	Course,Student	publishResult()
13	Generate Tabulation Sheet	Tabulation Sheet	Administrator, Course, Student, Faculty Member	generateTabulationSheet()
14	Calculate Grade Point Average	Student	Tabulation Sheet	calculateCGPA()
15	View Result	Student		viewResult()
16	Calculate Grade Point	Course	SpreadSheet	calculateGradePoint()
17	Calculate Course Total Mark	SpreadSheet	Course	calculateCourseTotalMark()
18	Import Student List	Administrator	Student	importStudent()
19	Set Role	Faculty Member	Course, SpreadSheet	setRole()
20	Store Change	Log	Administrator	storeChangeLog()
21	Assign Weight	SpreadSheet	Faculty Memeber	assignWeight()

### 8.1.2 STATE TRANSITION DIAGRAM



### 8.1.2.1 STATE TRANSITION DIAGRAM: ADMINISTRATOR

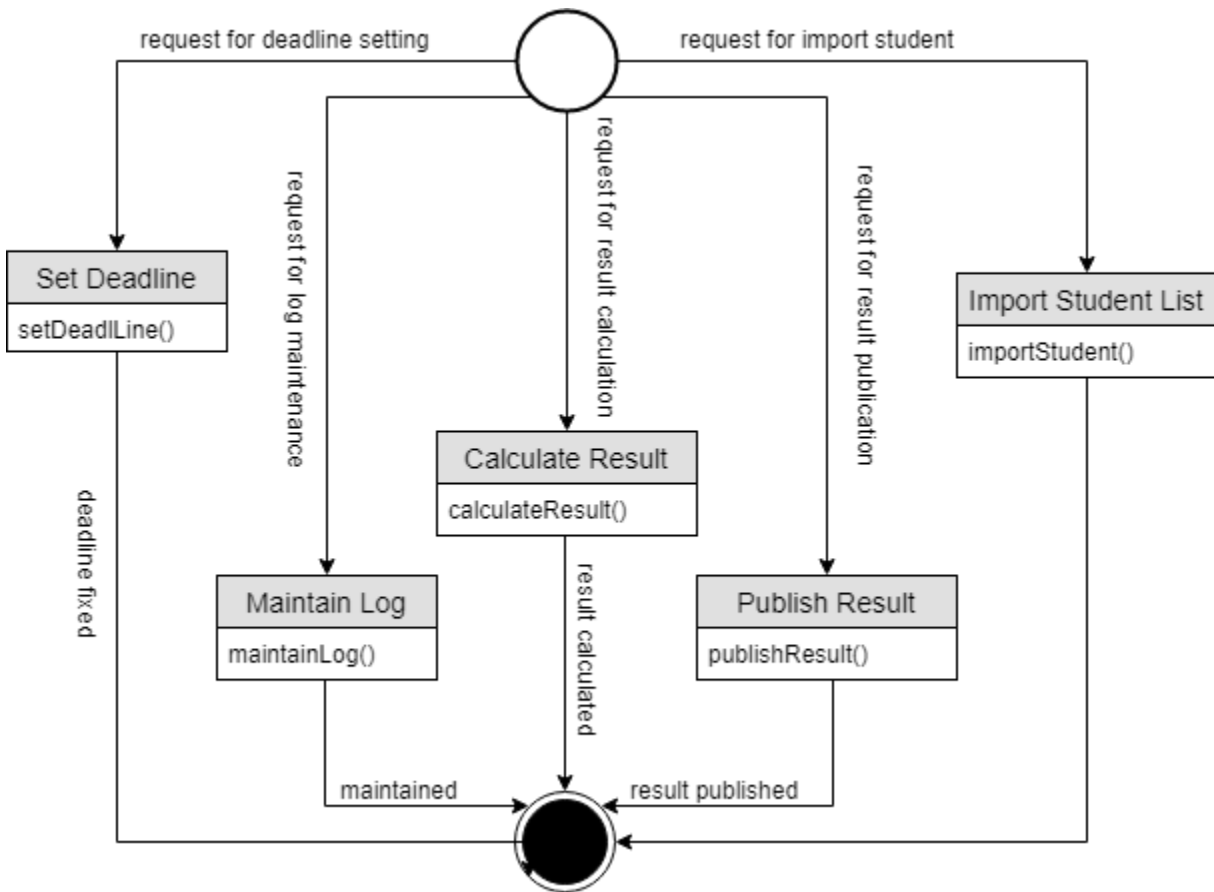


Figure 30: State Transition diagram: Administrator

### 8.1.2.2 STATE TRANSITION DIAGRAM: FACULTYMEMBER

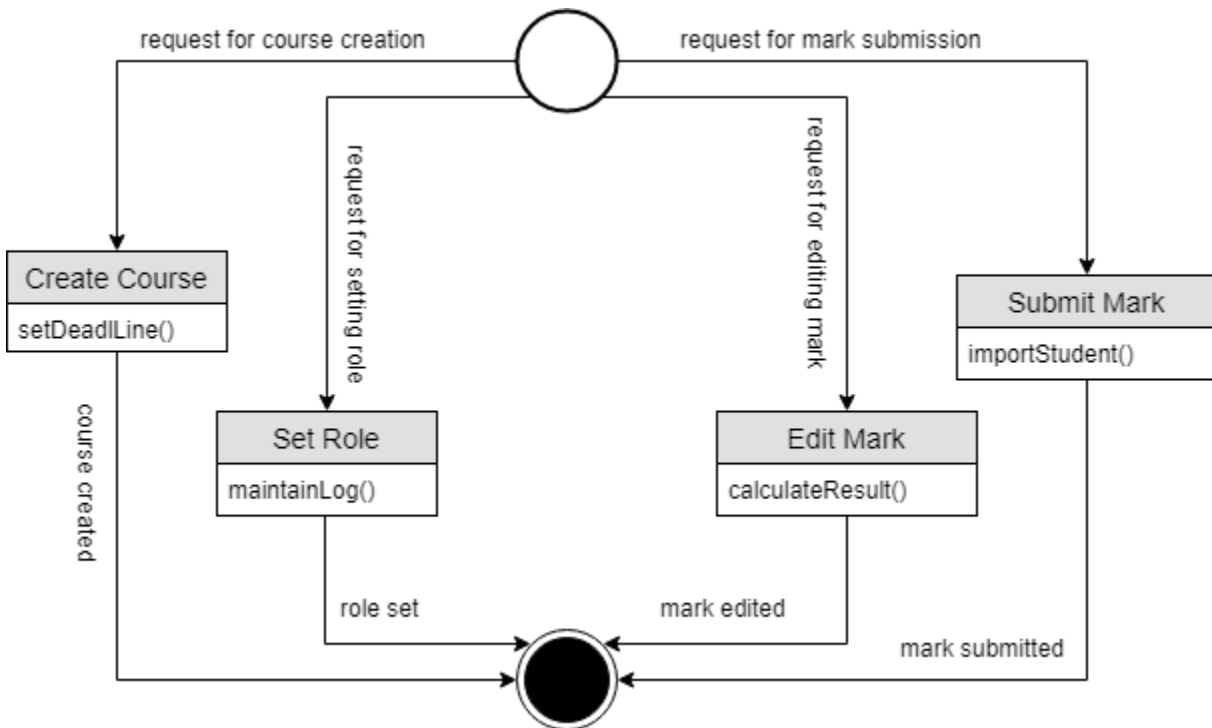


Figure 31: State Transition diagram: Administrator: Faculty Member

### 8.1.2.3 STATE TRANSITION DIAGRAM: STUDENT

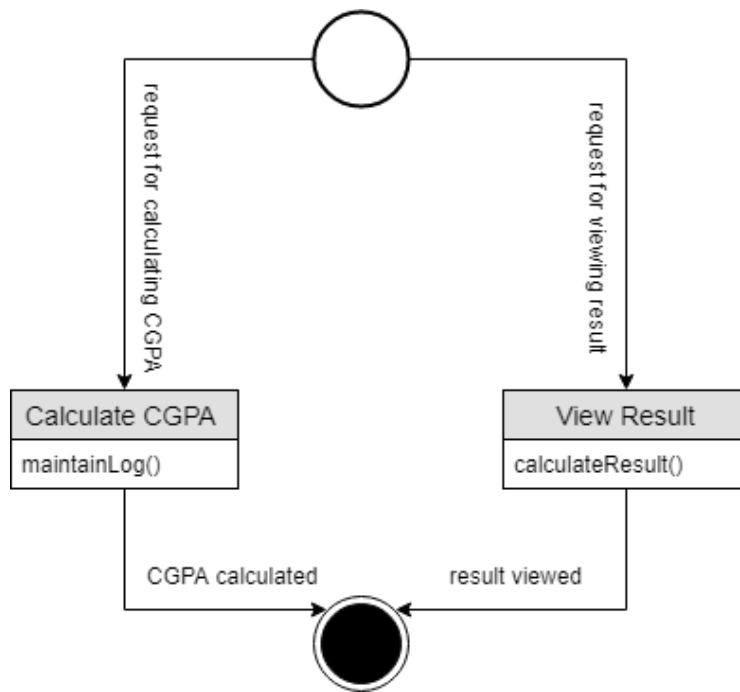


Figure 32: State Transition diagram: Student

#### 8.1.2.4 STATE TRANSITION DIAGRAM: AUTHENTICATION

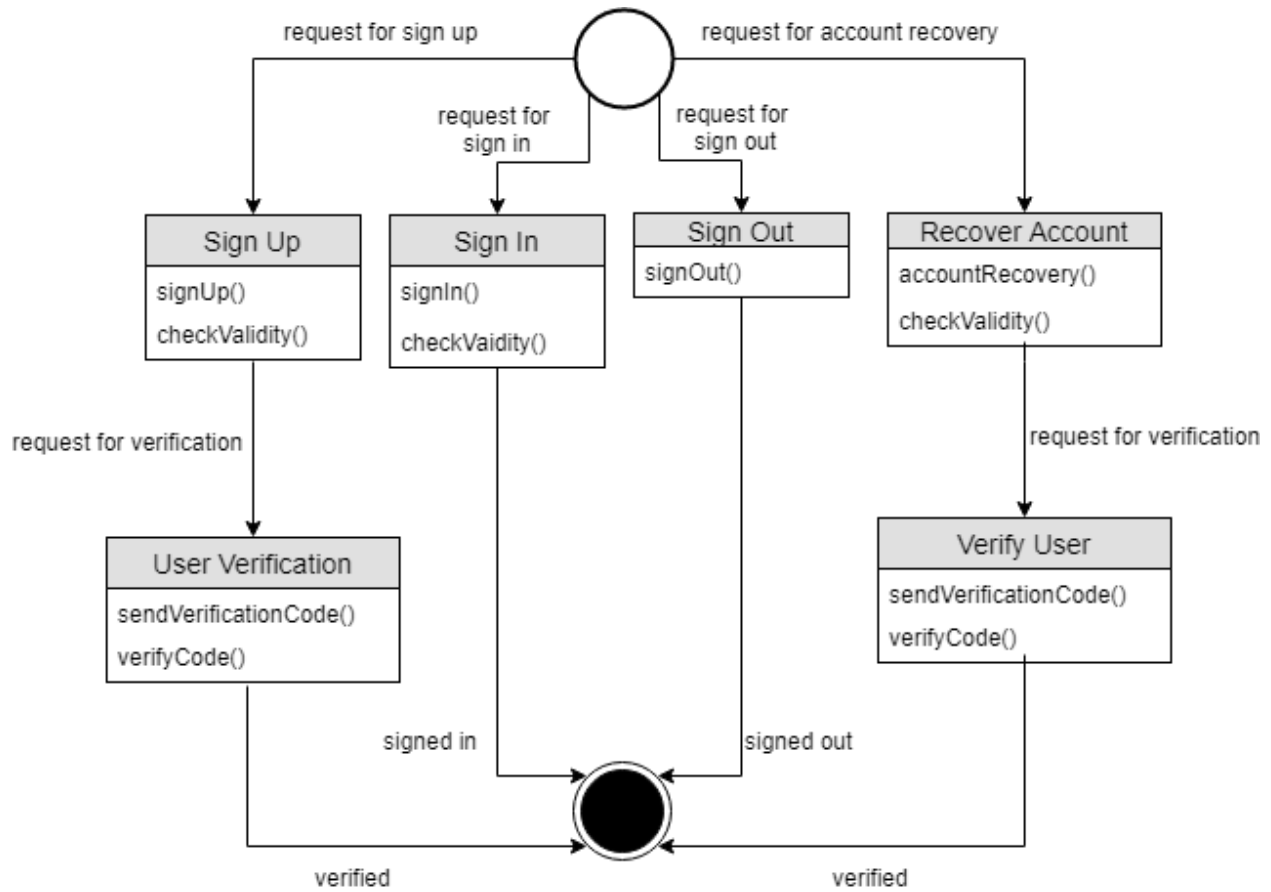


Figure 33: State Transition diagram: Authentication

### 8.1.2.5 STATE TRANSITION DIAGRAM: SPREADSHEET

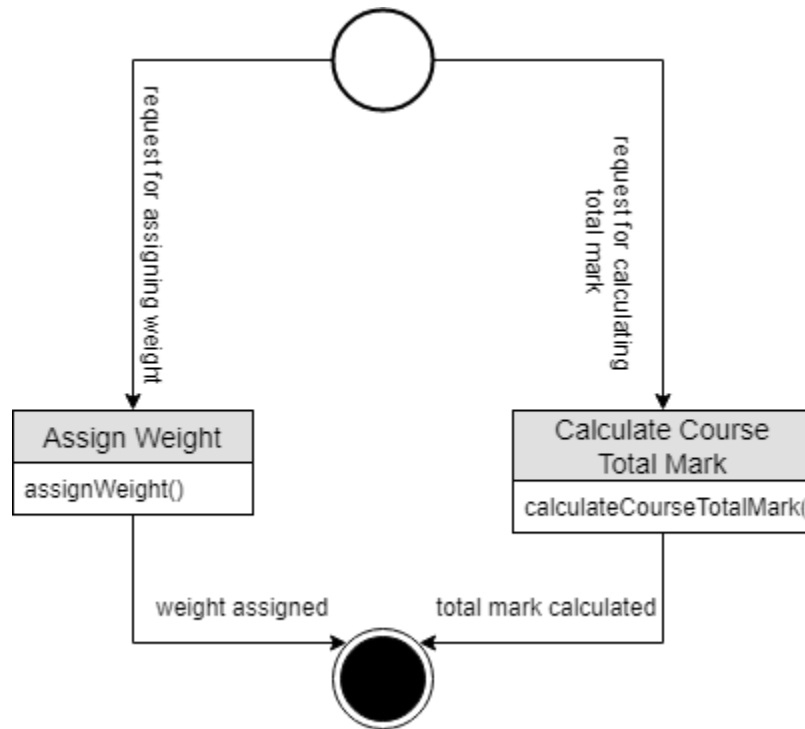


Figure 34: State Transition diagram: Spreadsheet

### 8.1.2.6 STATE TRANSITION DIAGRAM: TABULATIONSHEET

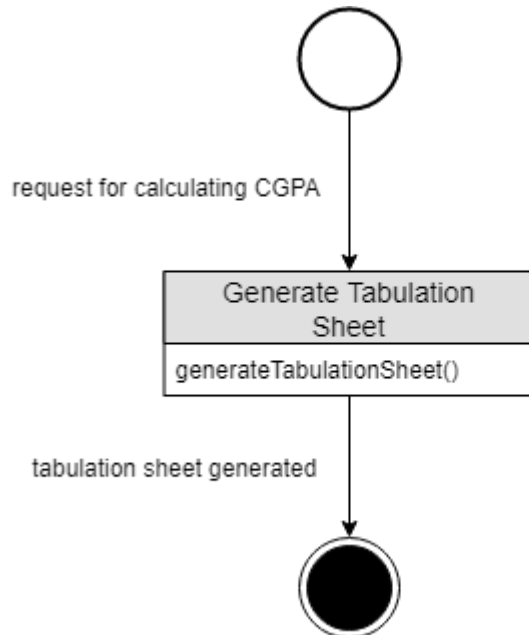


Figure 35: State Transition diagram: Tabulation Sheet

### **8.1.3 SEQUENCE DIAGRAM**