SWEN 6837

SOFTWARE ENGINEERING CAPSTONE PROJECT

ONLINE EXAMINATION

GROUP 5 – SPRING 2017

INSTRUCTOR – Dr. MICHAEL J. FINDLER

TEAM MEMBERS
KARTHIK PALAKURTHI
SAI KIRAN IPPILI
CHANDRAMOULI ADDANKI
RAMPRASAD SHANKARRAMAN

University
of Houston
Clear Lake

SWEN 6837-02

Software Engineering Capstone

Instructor – Dr. Michael J. Findler

Online Examination

Submitted by:
Karthik Palakurthi
Sai Kiran Ippili
Chandramouli Addanki
Ramprasad Shankarraman

TABLE OF CONTENTS

1.4.5 MODIFY EXAM SCHEDULE

1.5 VIEW REPORTS

1.5.1 PRINT QUESTIONS

1.5.2 PRINT STUDENT RESULTS

2 STUDENT

2.1 REGISTRATION

2.2TAKE EXAM

2.2.1 SELECT COURSE

2.2.2 START EXAM

2.3 VIEW GRADES

3 ADMIN

3.1 ADD DEPARTMENTS

3.2 ADD COURSES

3.3 ADD FACULTY

3.4 ADD FACULTY TO COURSE

3.5 TAKE BACKUP

3.6 MAINTENANCE

4 SEND REMINDERS

4.1 REMINDER TO FACULTY

4.2 REMINDER TO STUDENT

4.3 REMINDER TO ADMIN

1. PROJECT DESCRIPTION

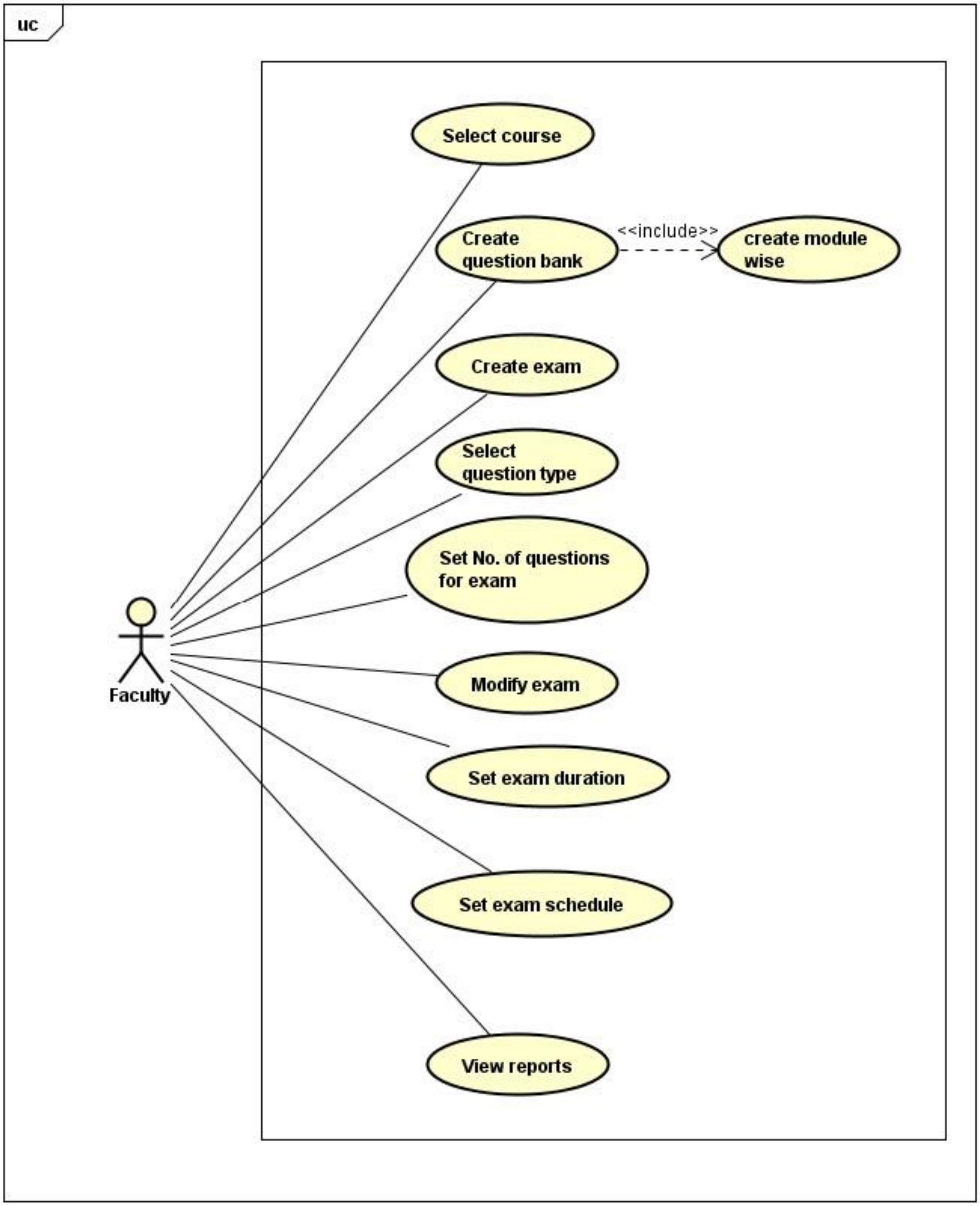
Project Description

This project is about online examination, which is evolved in order to suppress the paper examination which has many disadvantages like, hard retrieval of data, no long storage, no backups, reviewing, filtering.

This web application provides facility to conduct online examination. It saves time as it allows number of students to take exam at a time and displays the results as the exam complete, so no need to wait for the result as per paper format. Results are automatically generated by the server I in the new system. Faculty has a privilege to create, modify and delete the test papers and its particular questions. Student can register, login and give the test with his specific id and can see results. The faculty has more number of functions to be performed as a user of this application, Faculty is the one who create exam and set timer for the exam. The faculty has the DML operations like view, edit, and create the test. The Reminder function in the systems helps student, admin and faculty in reminding their task scheduled at some particular date and time.

2. USECASES

1. Use cases – Faculty



1.1 Select course

1.1.1 Brief Description

The faculty selects course from the list of courses he/she she is teaching.

1.1.2 Preconditions

The faculty has with him/her the list of courses that he/she teaches.

1.1.3 Flow of Events

- The faculty has the list of courses that are being taught for that semester.
- Faculty selects the course for which he/she is going to upload the list of questions.

1.1.4 Alternate Flow

N/A

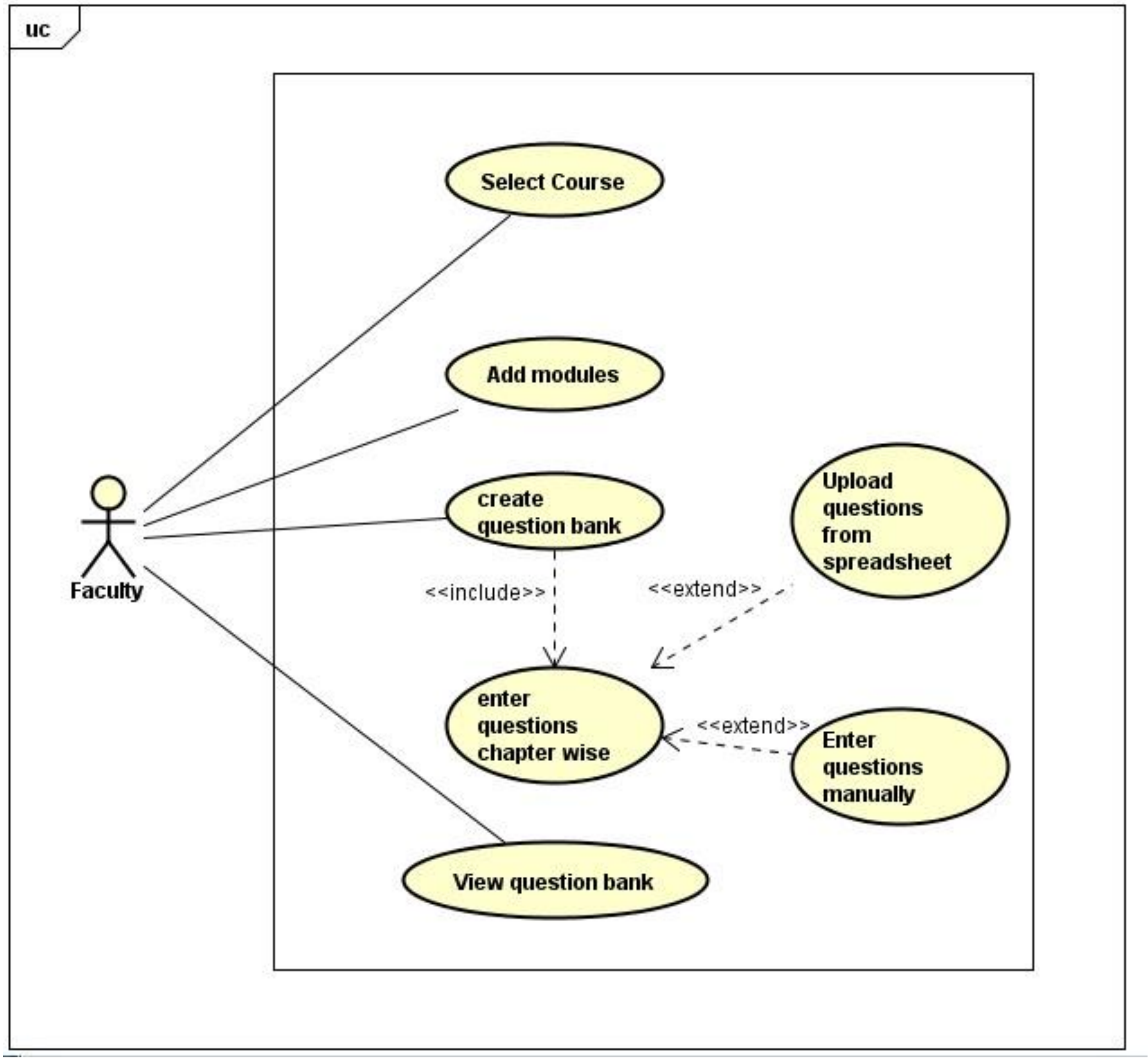
1.1.5 Post conditions

Faculty is able to select the courses from the list of courses.

1.1.6 Actors

Faculty is the only actor here.

1.2 Create question bank and Create module wise questions



1.2.1 Brief Description

The faculty creates Module wise question bank for the quizzes and exams.

1.2.2 Preconditions

- Faculty has selected the course for which he/she is going to create a Module wise question bank.

1.2.3 Flow of events

- Faculty has selected the course from the list of course(s) he/she will be teaching that particular semester.
- Faculty creates a Module wise question bank for that particular course.

1.2.4 Alternate Flow

N/A

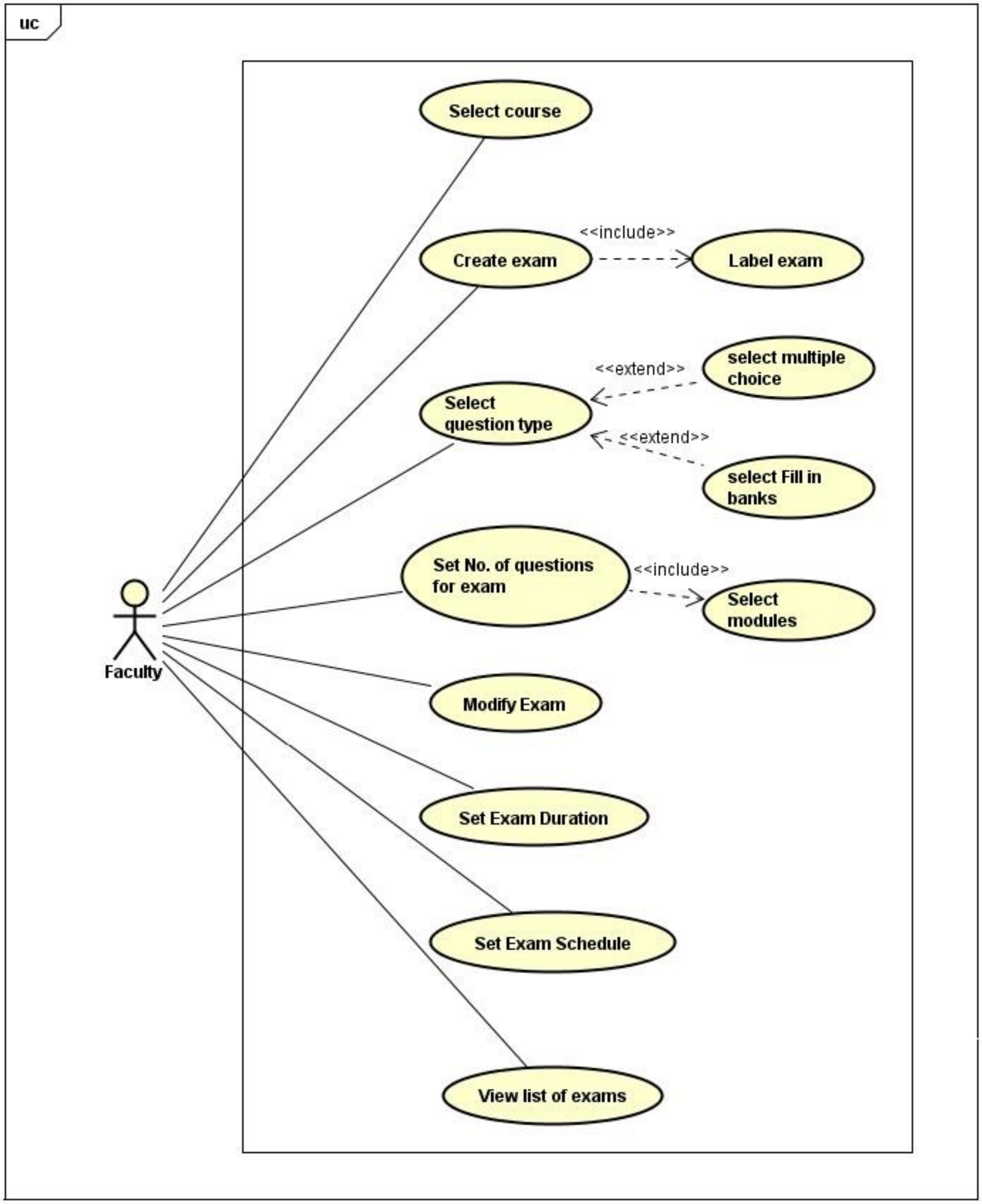
1.2.5 Post conditions

The Admin is able to create a question bank for the course Module wise

1.2.6 Actors

Faculty is the only actor here.

1.3 Create exam



1.3.1 Brief Description

The faculty creates exam for the courses from time to time.

1.3.2 Preconditions

Faculty has created the Module wise questions for the course.

1.3.3 Flow of events

- The faculty has created the Module wise question bank for the course.
- Faculty then creates exam for that particular course and uses the already uploaded questions from the question bank for the exam.

1.3.4 Alternate flow

N/A

1.3.5 Post conditions

Faculty is able to create exam for that particular course.

1.3.6 Actors

Faculty is the only actor here.

1.4 Select question type

1.4.1 Brief Description

The faculty selects the type of questions that are going to be asked in the examination.

1.4.2 Preconditions

- The faculty has already uploaded the Module wise questions.

1.4.3 Flow of events

- Faculty selects the type of question, say for example whether it is multiple choice questions and fill in the blanks.

1.4.4 Alternate Flow

N/A

1.4.5 Post conditions

The faculty is able to select the question type when he/she is creating the

1.4.6 Actors

Faculty is the only actor here.

1.5 Set Number of questions for exam

1.5.1 Brief Description

The faculty selects the number of questions for the examination.

1.5.2 Preconditions

- The faculty has already uploaded the Module wise questions and selects the type of questions that is going to be asked in the examination.

1.5.3 Flow of events

- Faculty uploads the Module wise questions and s
- Later he/she sets the number of questions for the exam.

1.5.4 Alternate Flow

N/A

1.5.5 Post conditions

The faculty is able to sets the number of questions for the examination.

1.5.6 Actors

Faculty is the only actor here.

1.6 Modify exam

1.6.1 Brief Description

The faculty modifies exam (questions, exam duration and question type).

1.6.2 Preconditions

- The faculty has already created exam and has uploaded the Module wise question bank.

1.6.3 Flow of events

- Faculty wishes to modify exam say for example, the examination question, exam duration and question type.
- Faculty modifies it if he/she feels the need to do so.

1.6.4 Alternate Flow

N/A

1.6.5 Post conditions

The Admin is able to select the question type when he/she is creating the

1.6.6 Actors

Faculty is the only actor here.

1.7 Set exam duration

1.7.1 Brief Description

The faculty creates the exam duration.

1.7.2 Preconditions

- The faculty has already created exam and has selected the number of questions.

1.7.3 Flow of events

- Faculty has created the number of questions, question type, etc.
- The faculty then sets the exam duration.

1.7.4 Alternate Flow

N/A

1.7.5 Post conditions

The faculty is able to select the exam duration.

1.7.6 Actors

Faculty is the only actor here.

1.8 Set exam schedule

1.8.1 Brief Description

The faculty sets the exam schedule.

1.8.2 Preconditions

- The faculty has already uploaded exam questions, set the question type.

1.8.3 Flow of events

- Faculty decides on the exam schedule, the date, time and the venue where the examination is scheduled to be conducted.

1.8.4 Alternate Flow

N/A

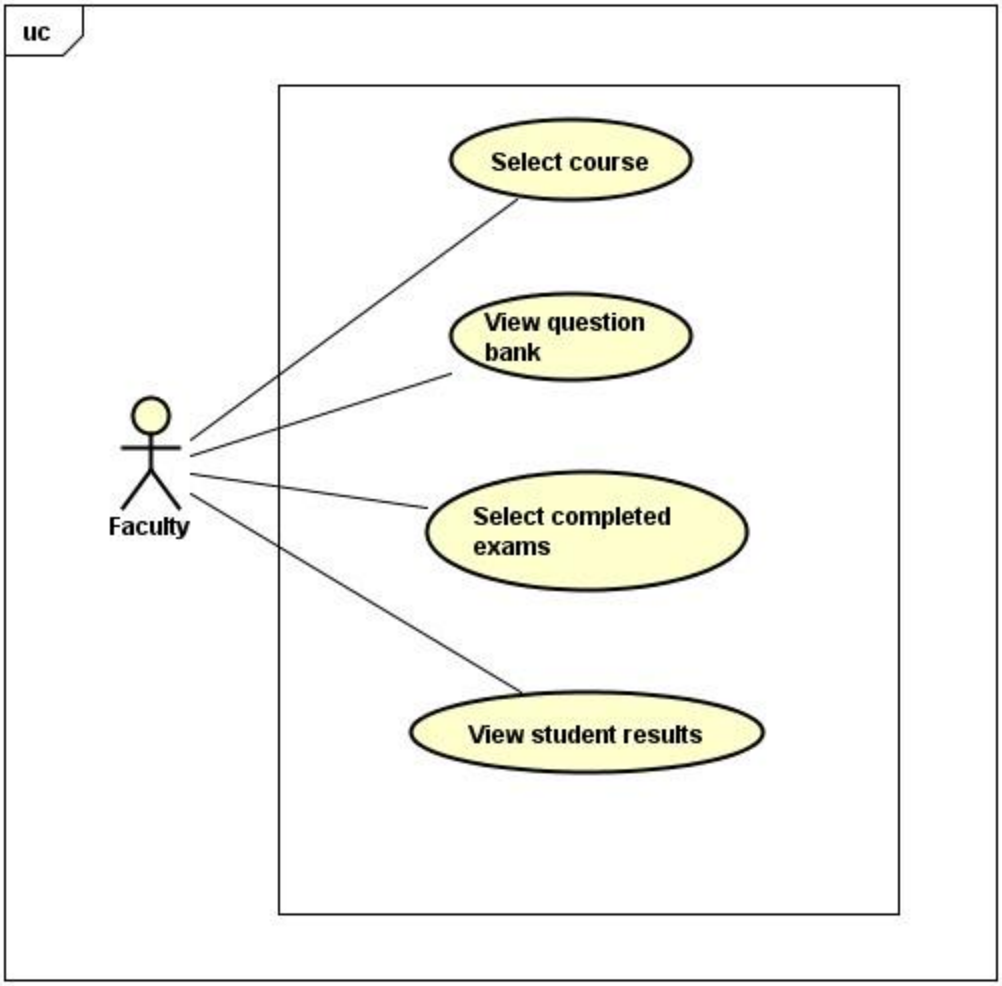
1.8.5 Post conditions

The faculty is able to select the examination schedule.

1.8.6 Actors

Faculty is the only actor here.

1.9 View reports



1.9.1 Brief Description

The faculty views the student’s exam result reports.

1.9.2 Preconditions

- The student has completed writing the examination.

1.9.3 Flow of events

- The student writes the examination.
- The faculty views the exam result in the form of a report.

1.9.4 Alternate Flow

N/A

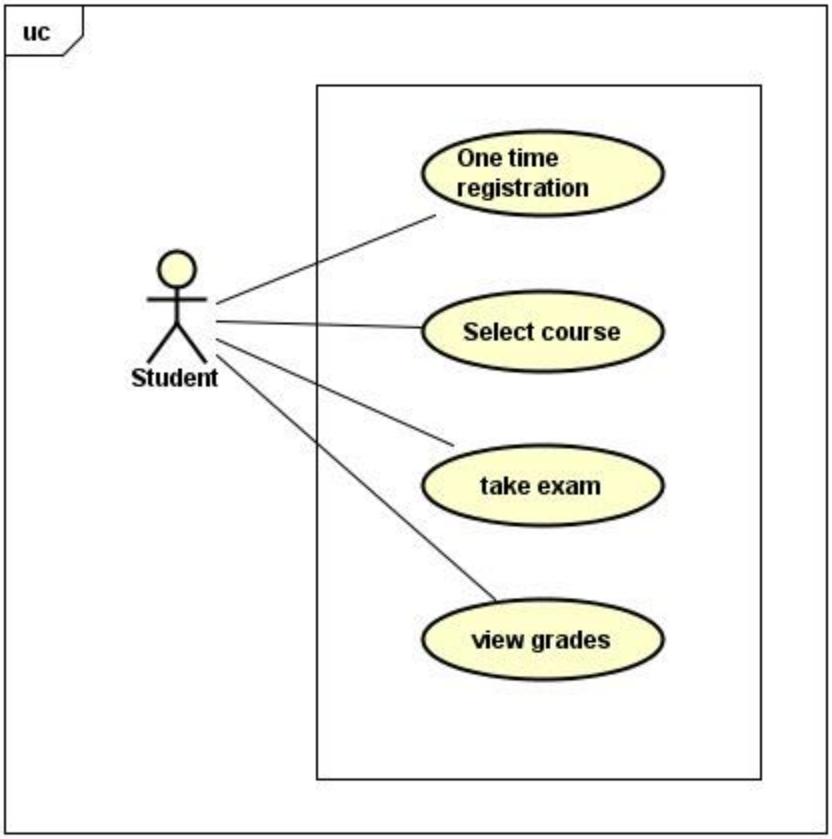
1.9.5 Post conditions

The faculty is able to view the exam results in the form of reports.

1.9.6 Actors

Faculty is the only actor here.

2. Student Over-View



2.1 One Time Registration

2.1.1 Brief description

This use-case gives us the one-time registration of student.

2.1.2 Preconditions

The student should be a valid person to register.

2.1.3 Flow of events

The student here will have the one-time registration after a successful registration he can select the course.

2.1.4 Alternate Flow

N/A

2.1.5 Post conditions

Successfully completes the registration and proceeds.

2.1.6 Actors

Student.

2.2 Select Course

2.2.1 Brief description

This use-case gives us the student selecting the course.

2.2.2 Preconditions

The student should be a valid student and should have registered for the exam.

2.2.3 Flow of events

The student will select the course where he wants to take an exam.

2.2.4 Alternate Flow

N/A

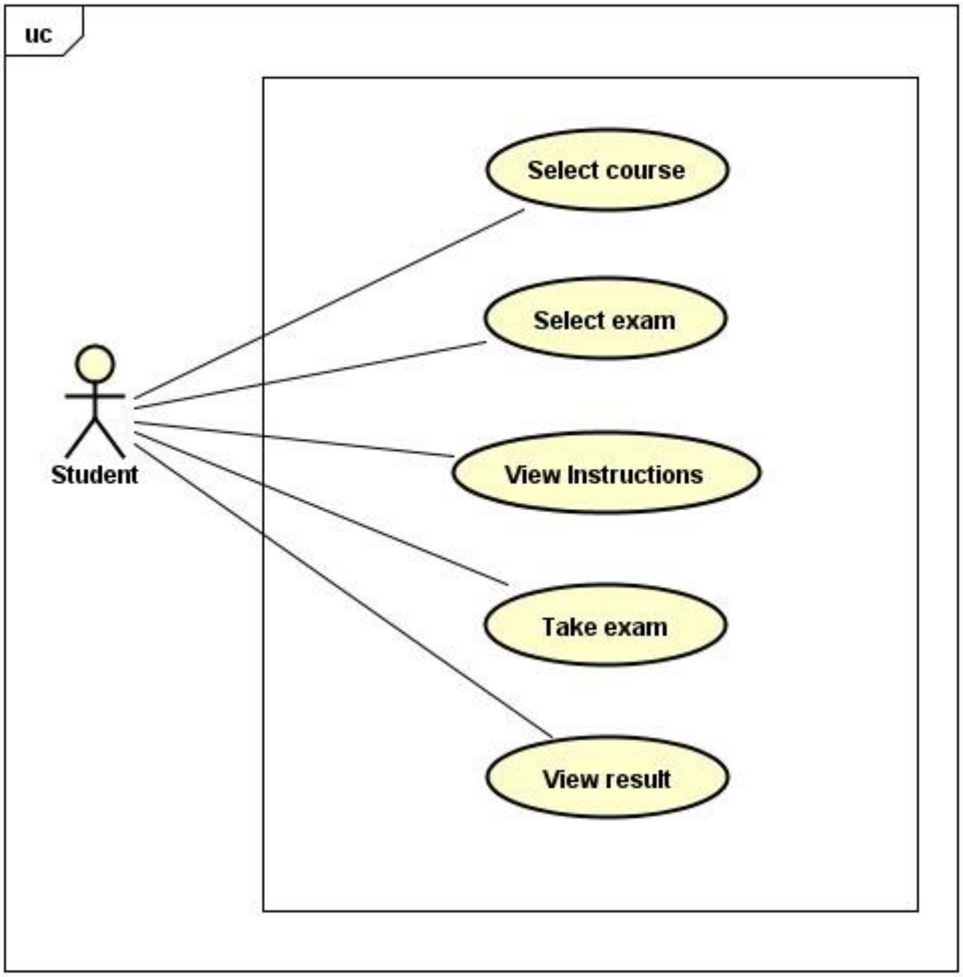
2.2.5 Post conditions

Successfully selects the course and move to take an exam.

2.2.6 Actors

Student

2.3 Take Exam


uc
Student
Select course
Select exam
View Instructions
Take exam
View result

2.3.1 Brief description

This use-case gives us the information about student taking exam.

2.3.2 Preconditions

The student should have to select the course for which he wants to take an exam.

2.3.3 Flow of events

The student will take an exam.

2.3.4 Alternate Flow

N/A

2.3.5 Post conditions

Successfully completes taking examination.

2.3.6 Actors

Student

2.4 View Grades

2.4.1 Brief description

This use-case gives us the information about student viewing grades.

2.4.2 Preconditions

The student should have successfully completed in taking an exam.

2.4.3 Flow of events

The student will view the grades.

2.4.4 Alternate Flow

N/A

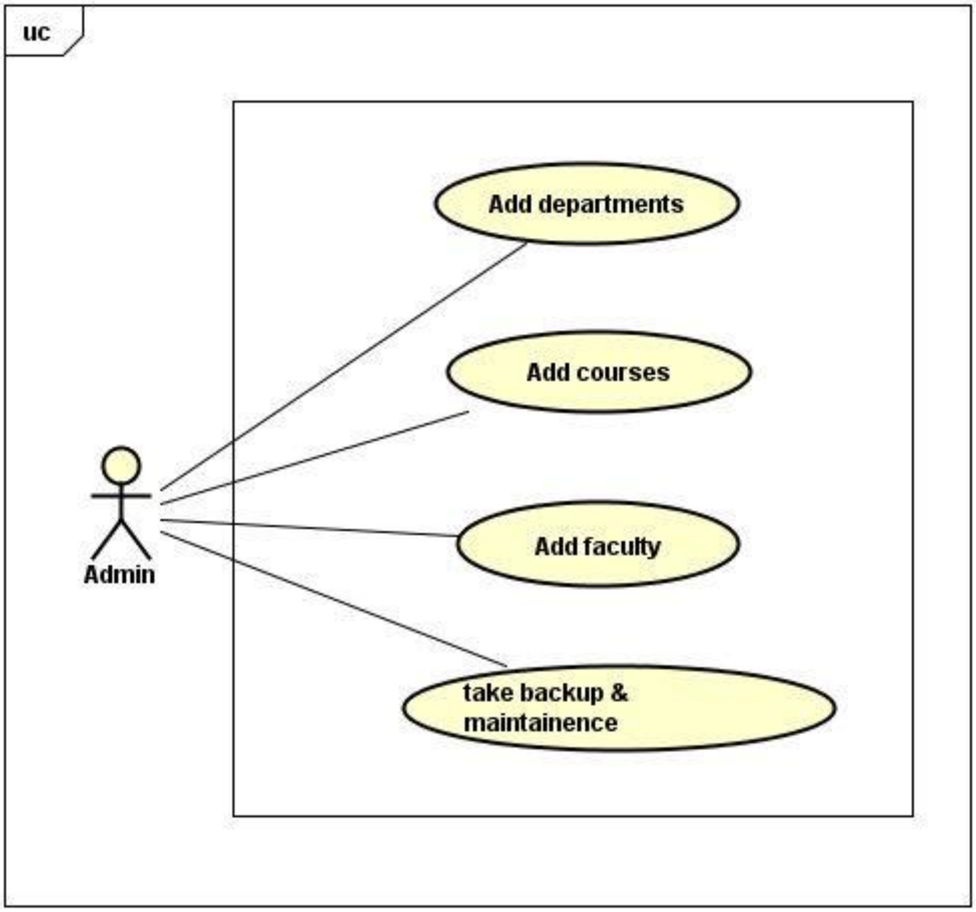
2.4.5 Post conditions

Successfully views the grade.

2.4.6 Actors

Student

3 Admin Overview (Use case spelling mistake)



3.1 Add Departments

3.1.1 Brief description

This use-case gives us the information about adding departments by admin.

3.1.2 Preconditions

There should be at least one department available in the school.

3.1.3 Flow of events

The admin will add the departments.

3.1.4 Alternate Flow

N/A

3.1.5 Post conditions

Successfully completes adding departments.

3.1.6 Actor

Admin.

3.2 Add Courses

3.2.1 Brief description

This use-case gives us the information about adding courses to department.

3.2.2 Preconditions

Admin should add department prior to adding of courses.

3.2.3 Flow of events

The admin will add the courses to department section.

3.2.4 Alternate Flow

N/A

3.2.5 Post conditions

Successfully completes adding courses.

3.2.6 Actor

Admin.

3.3 Add Faculty

3.3.1 Brief description

This use-case gives us the information about admin adding faculty to respected courses.

3.3.2 Preconditions

Admin should add course prior to adding of faculty.

3.3.3 Flow of events

The admin will add the faculty to their respected courses.

3.3.4 Alternate Flow

N/A

3.3.5 Post conditions

Successfully completes adding faculty.

3.3.6 Actor

Admin.

3.4 Take Back-up and Maintenance

3.4.1 Brief description

This use-case gives us the information about admin taking back-up and maintenance of the system.

3.4.2 Preconditions

There should be some new data to take back up and no pre condition for maintenance.

3.4.3 Flow of events

The admin will take back up of data and maintain the system up to date.

3.4.4 Alternate Flow

N/A

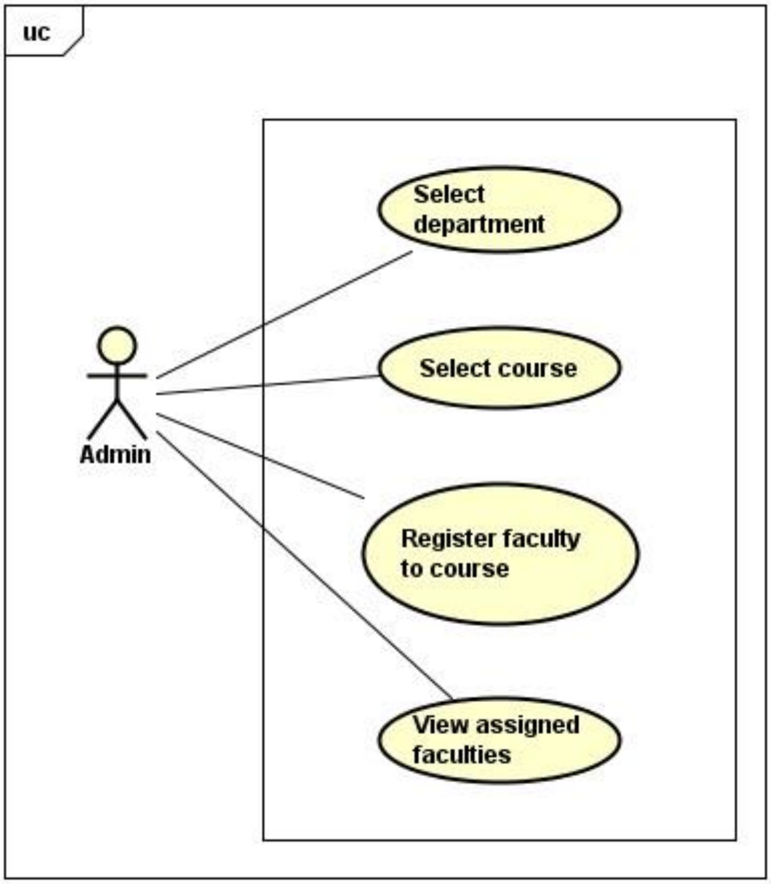
3.4.5 Post conditions

Successfully completes taking backup and maintaining system to up to date.

3.4.6 Actors

Admin.

4 Adding Faculty



4.1 Select Department

4.1.1 Brief description

This use-case gives us the information about admin selecting department to add faculty to the system.

4.1.2 Preconditions

There should be at least one faculty before adding faculty to a department.

4.1.3 Flow of events

The admin will select the respective department to add faculty.

4.1.4 Alternate Flow

N/A

4.1.5 Post conditions

Successfully completes selecting the department.

4.1.6 Actor

Admin.

4.2 Select Course

4.2.1 Brief description

This use-case gives us the information about admin selecting course in to the department.

4.2.2 Preconditions

There should be at least one department selected.

4.2.3 Flow of events

The admin will select the respective course in department.

4.2.4 Alternate Flow

N/A

4.2.5 Post conditions

Successfully completes selecting the course.

4.2.6 Actors

Admin.

4.3 Register faculty to course

4.3.1 Brief description

This use-case gives us the information about admin registering faculty to the selected course.

4.3.2 Preconditions

There should be at least one course selected before adding faculty.

4.3.3 Flow of events

The admin will add faculty to respective course.

4.3.4 Alternate Flow

N/A

4.3.5 Post conditions

Successfully completes registering the faculty to course.

4.3.6 Actor

Admin.

4.4 View Assigned Faculties

4.4.1 Brief description

This use-case gives us information on assigned faculties.

4.4.2 Preconditions

At least on faculty should have been registered.

4.4.3 Flow of events

Admin can view the assigned faculties.

4.4.4 Alternate Flow

N/A

4.4.5 Post conditions

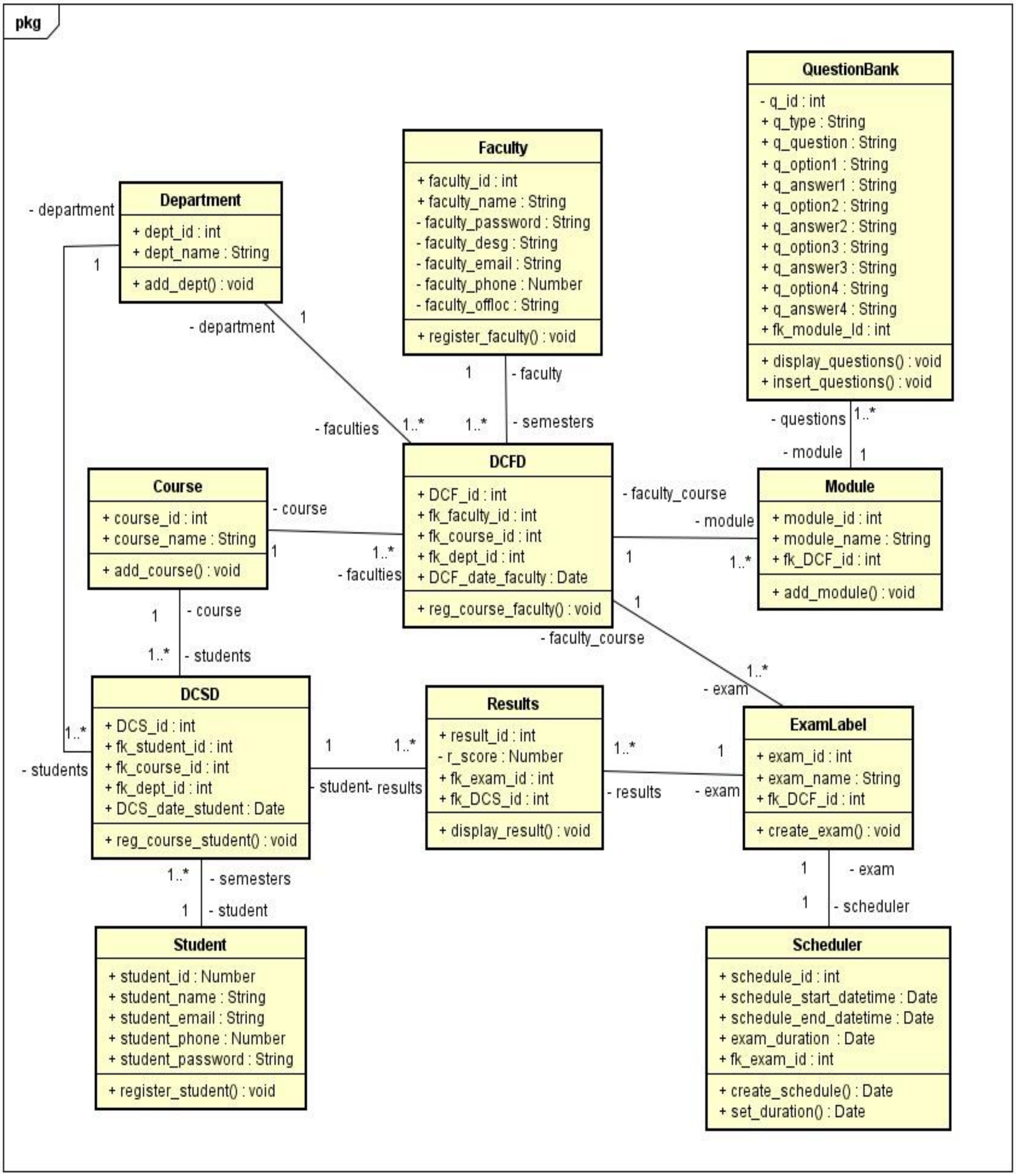
Successfully views the assigned faculty.

4.4.6 Actors

Admin

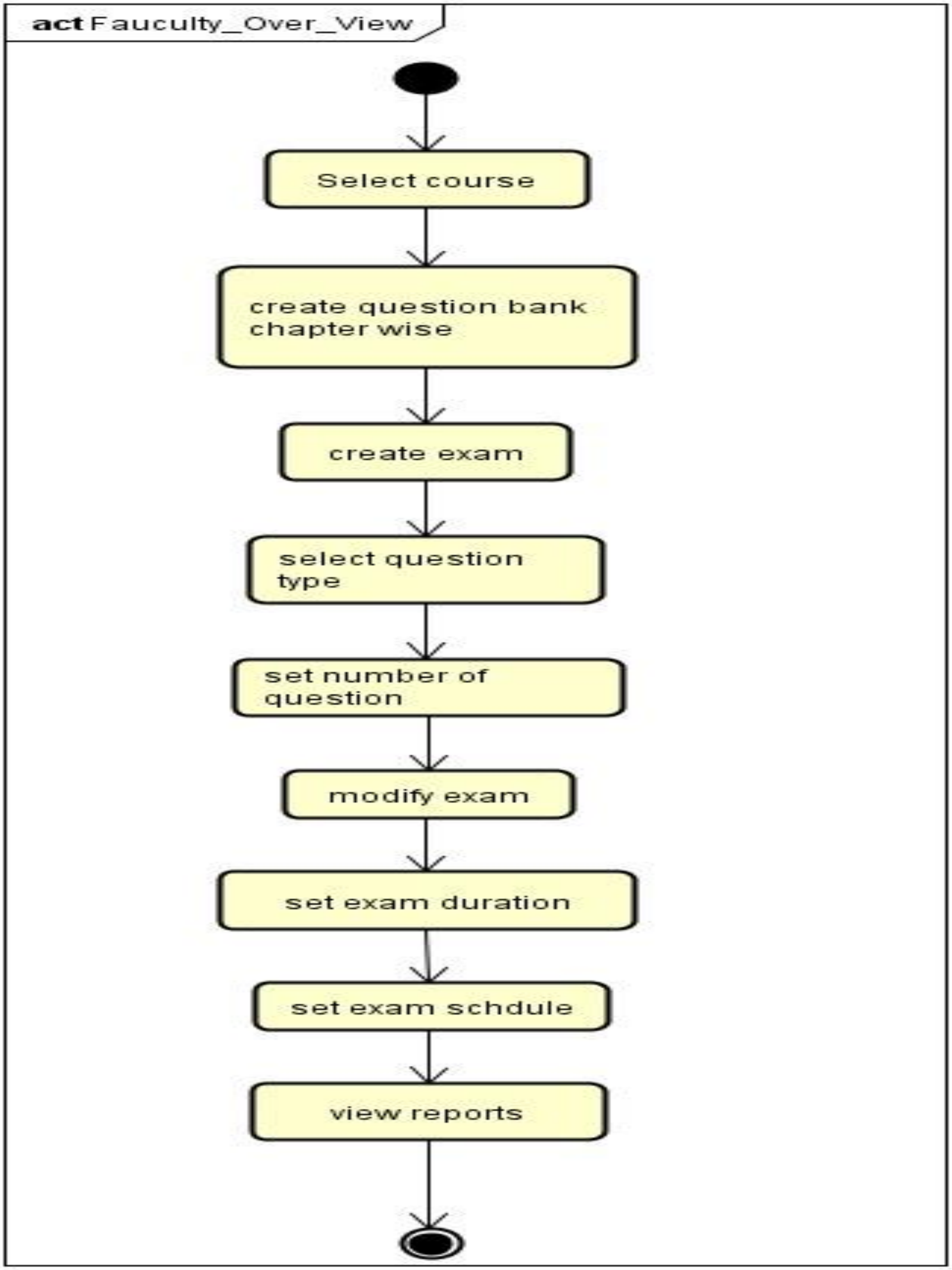
3. CLASS DIAGRAM

Class Diagram – Online Examination

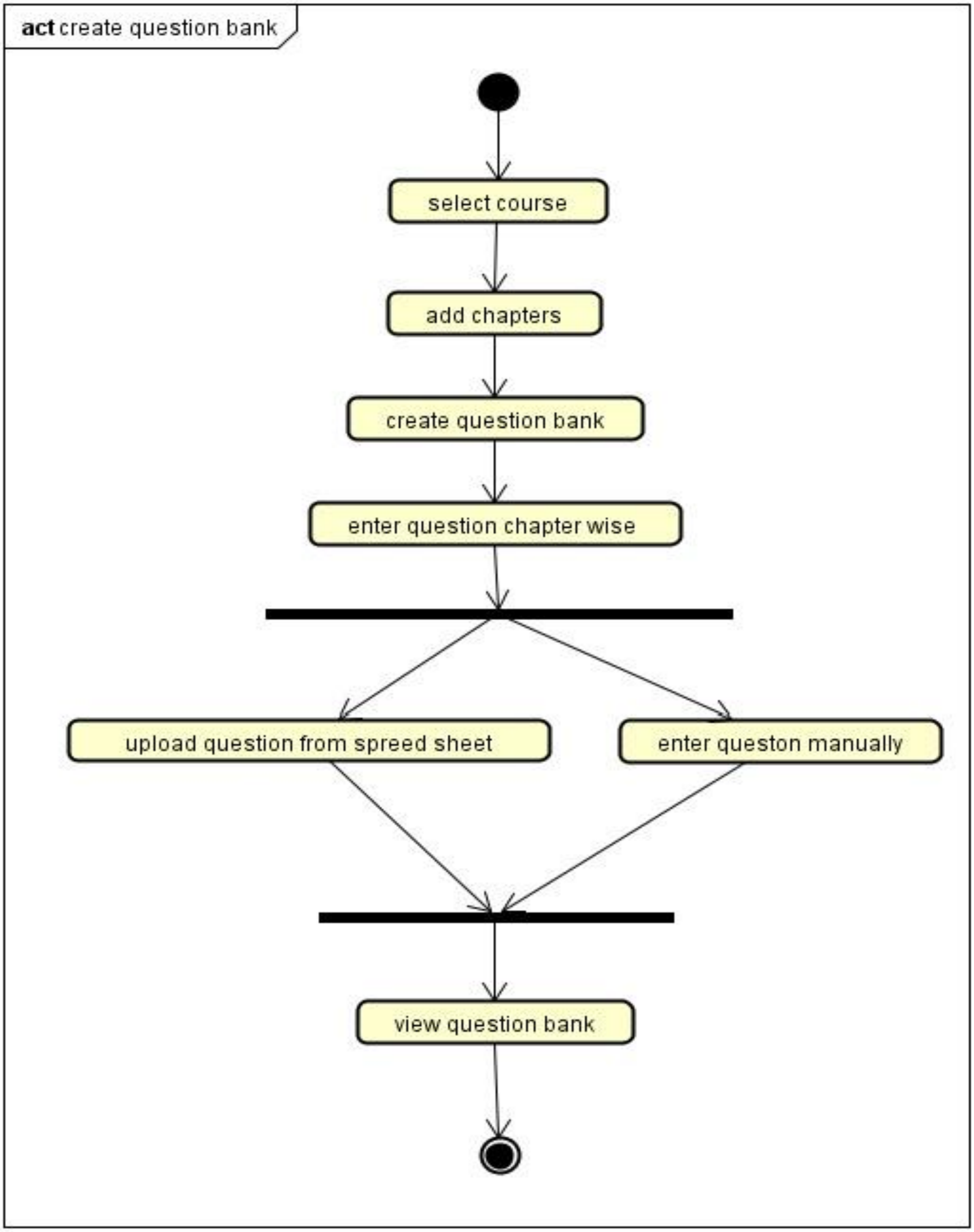


4. ACTIVITY DIAGRAMS

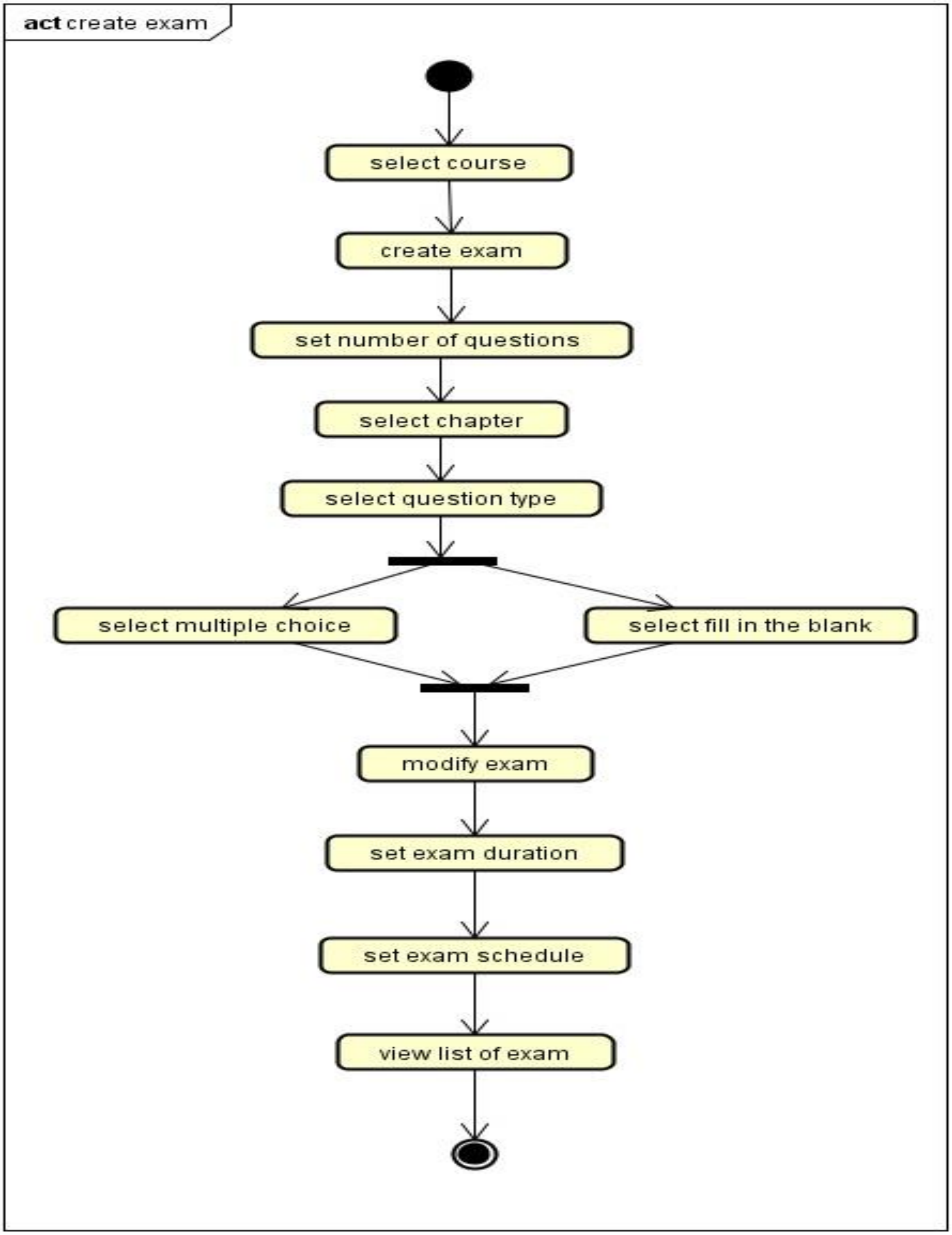
Faculty


act Faucculty_Over_View
Select course
create question bank chapter wise
create exam
select question type
set number of question
modify exam
set exam duration
set exam schdule
view reports

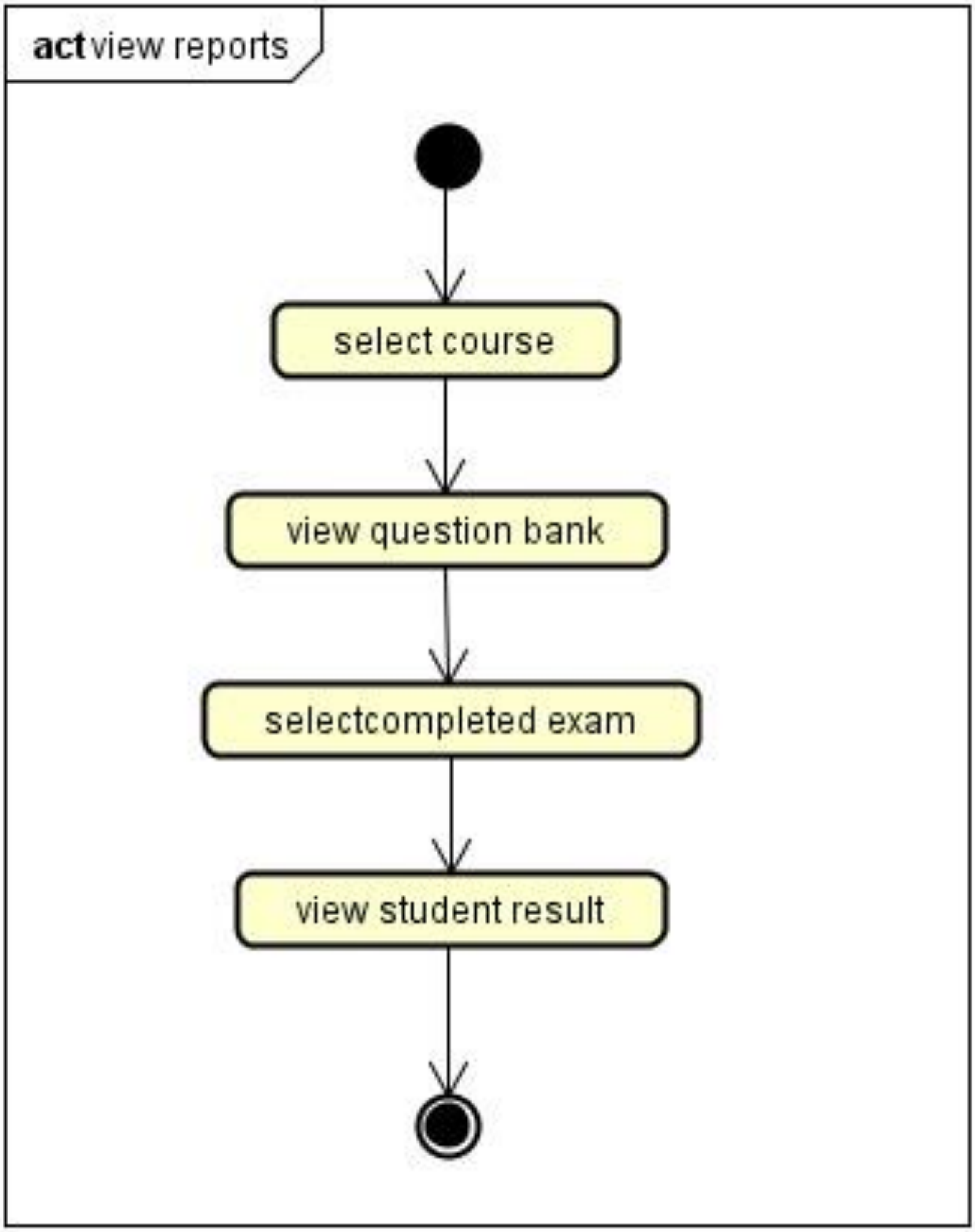
Faculty Create Question Bank



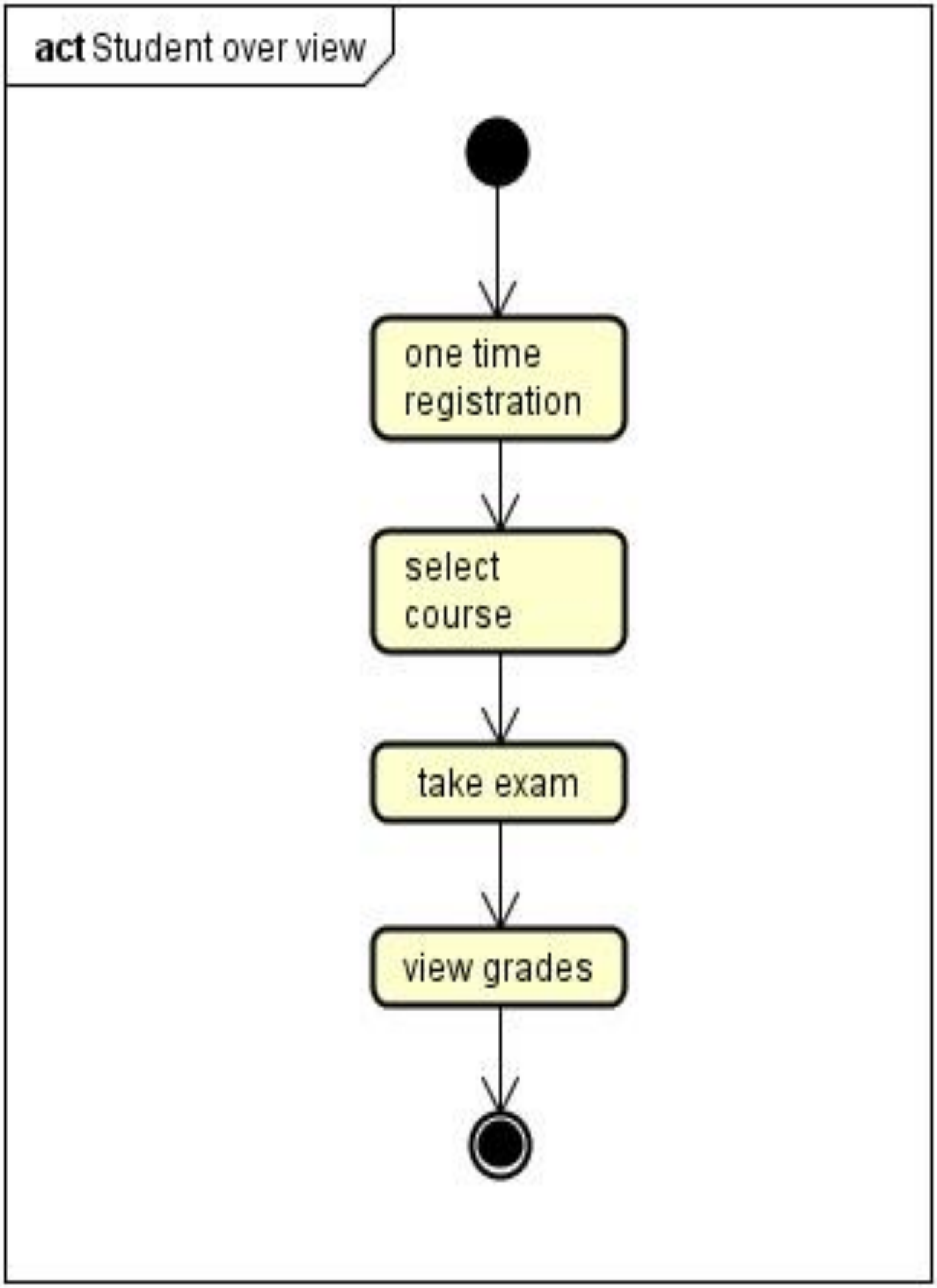
Faculty Create Exam



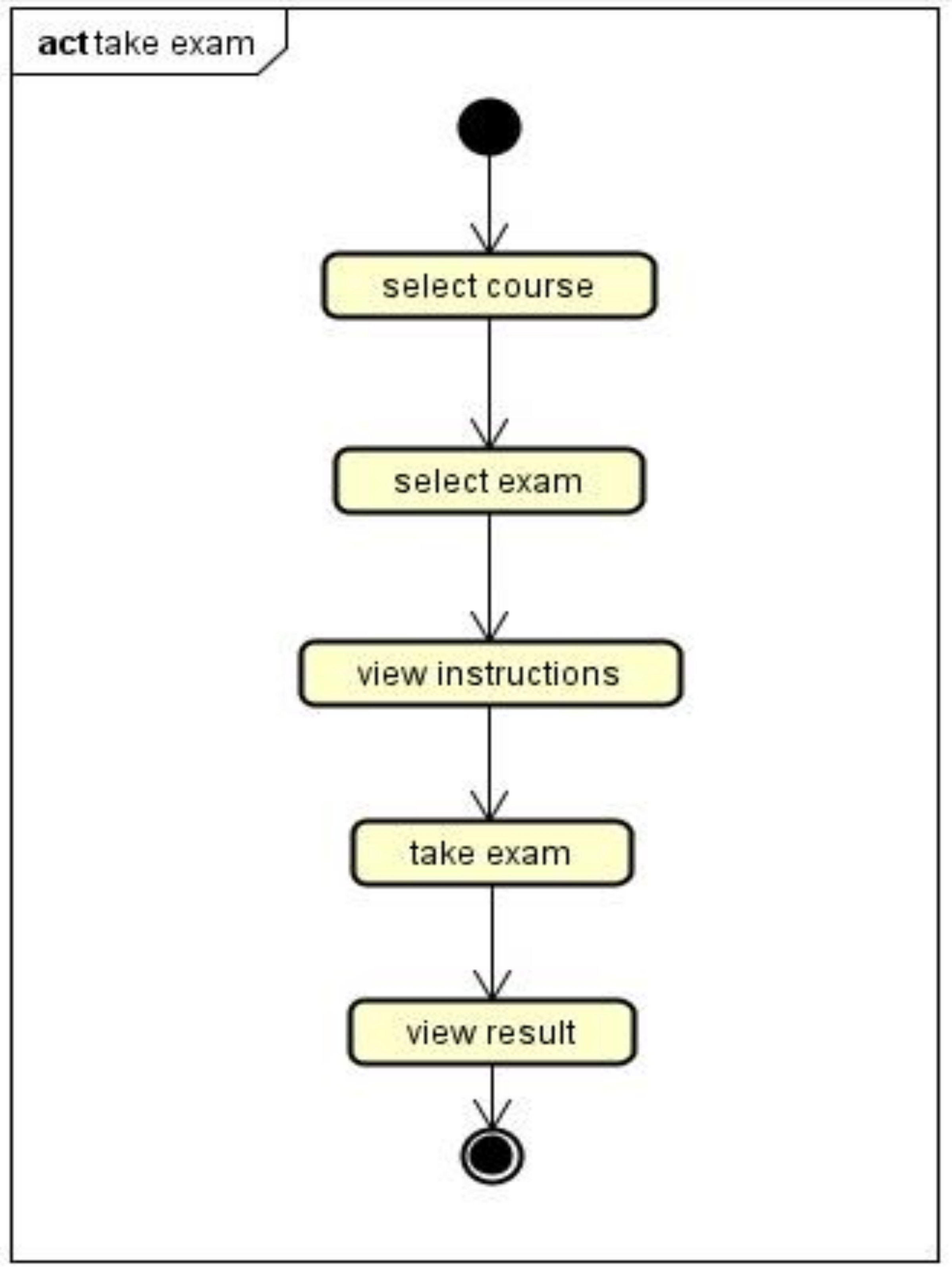
Faculty View Reports



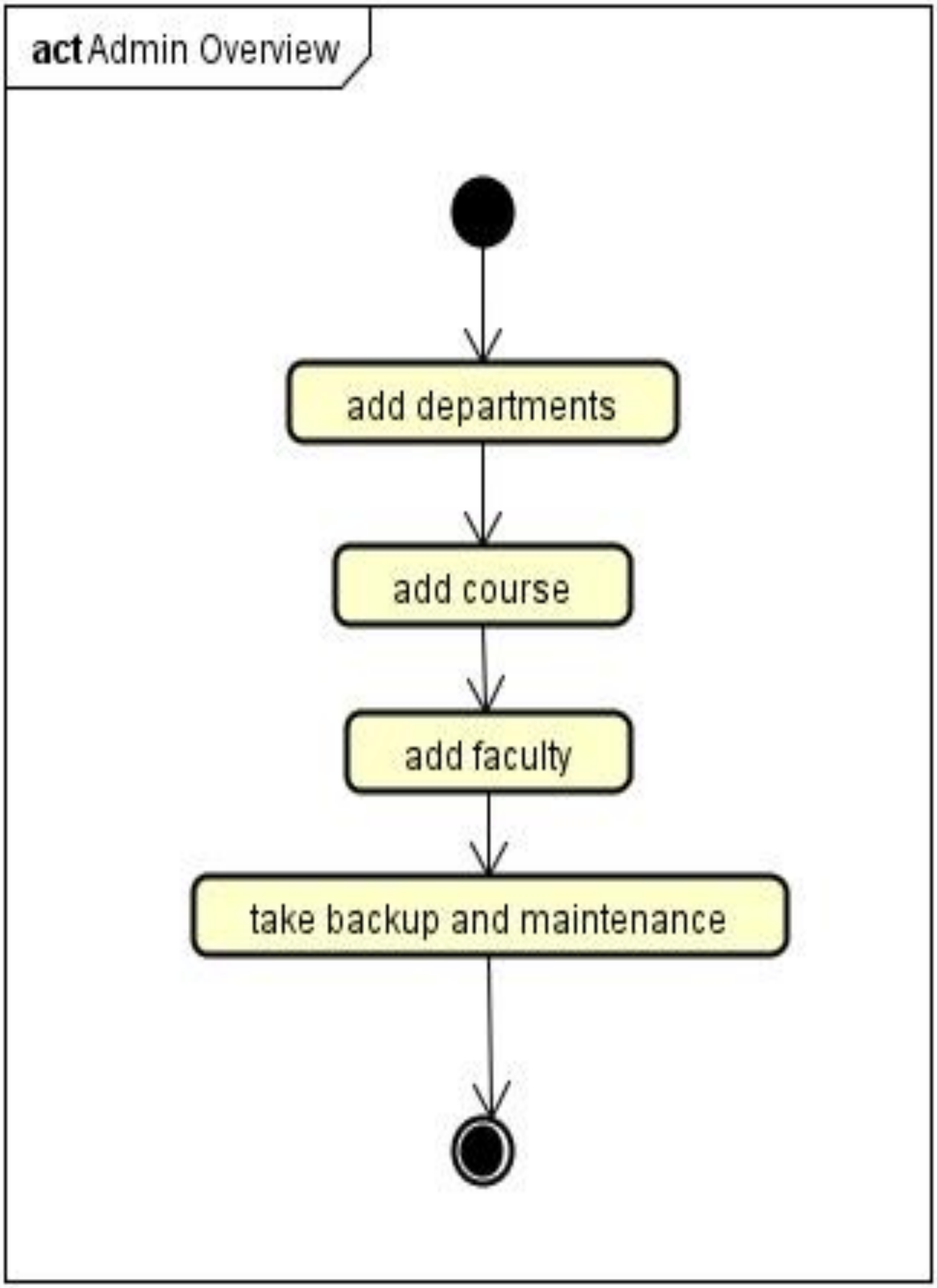
Student


act Student over view
one time registration
select course
take exam
view grades

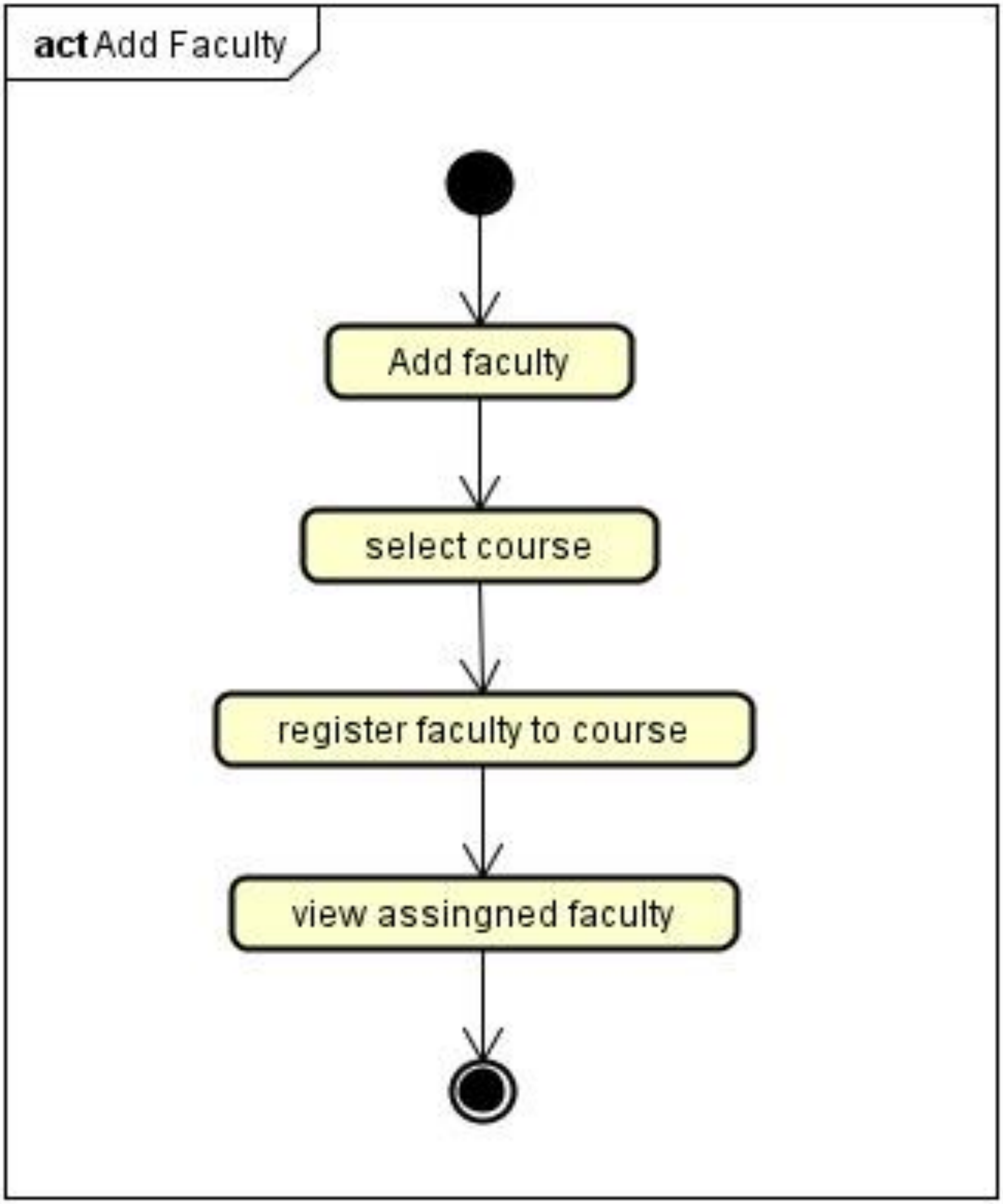
Student Take Exam



Admin

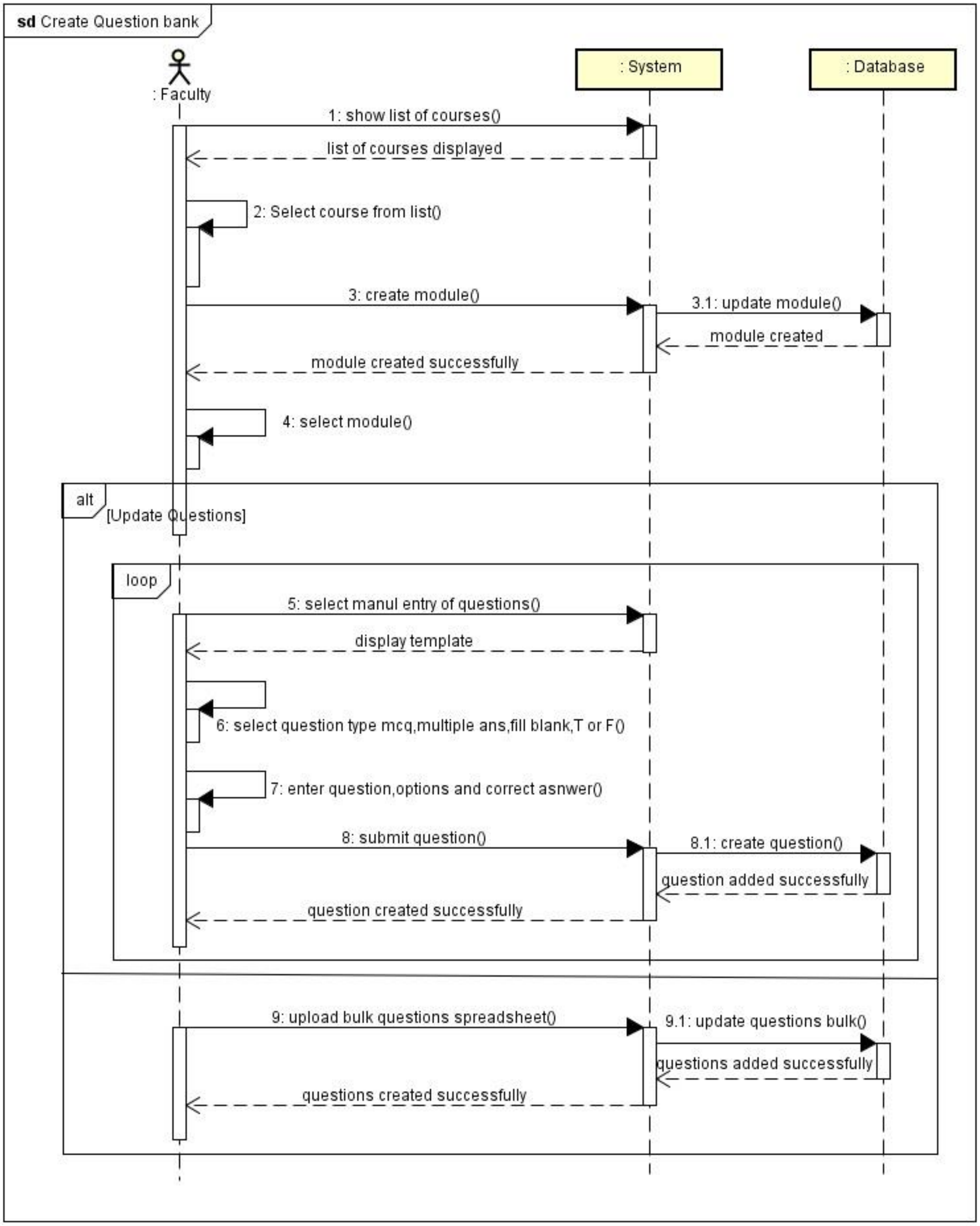

act Admin Overview
add departments
add course
add faculty
take backup and maintenance

Admin Add Faculty

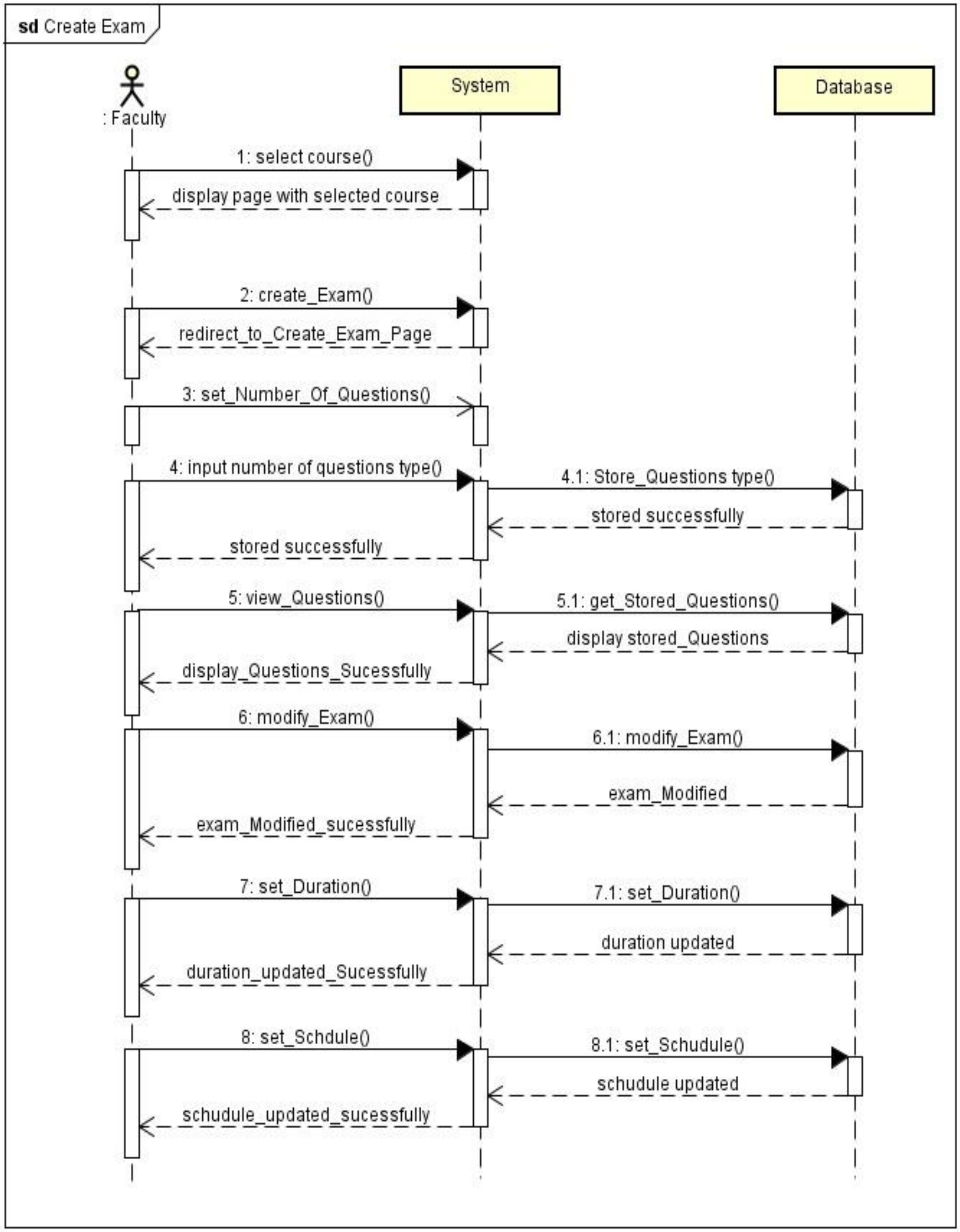


5. SEQUENCE DIAGRAMS

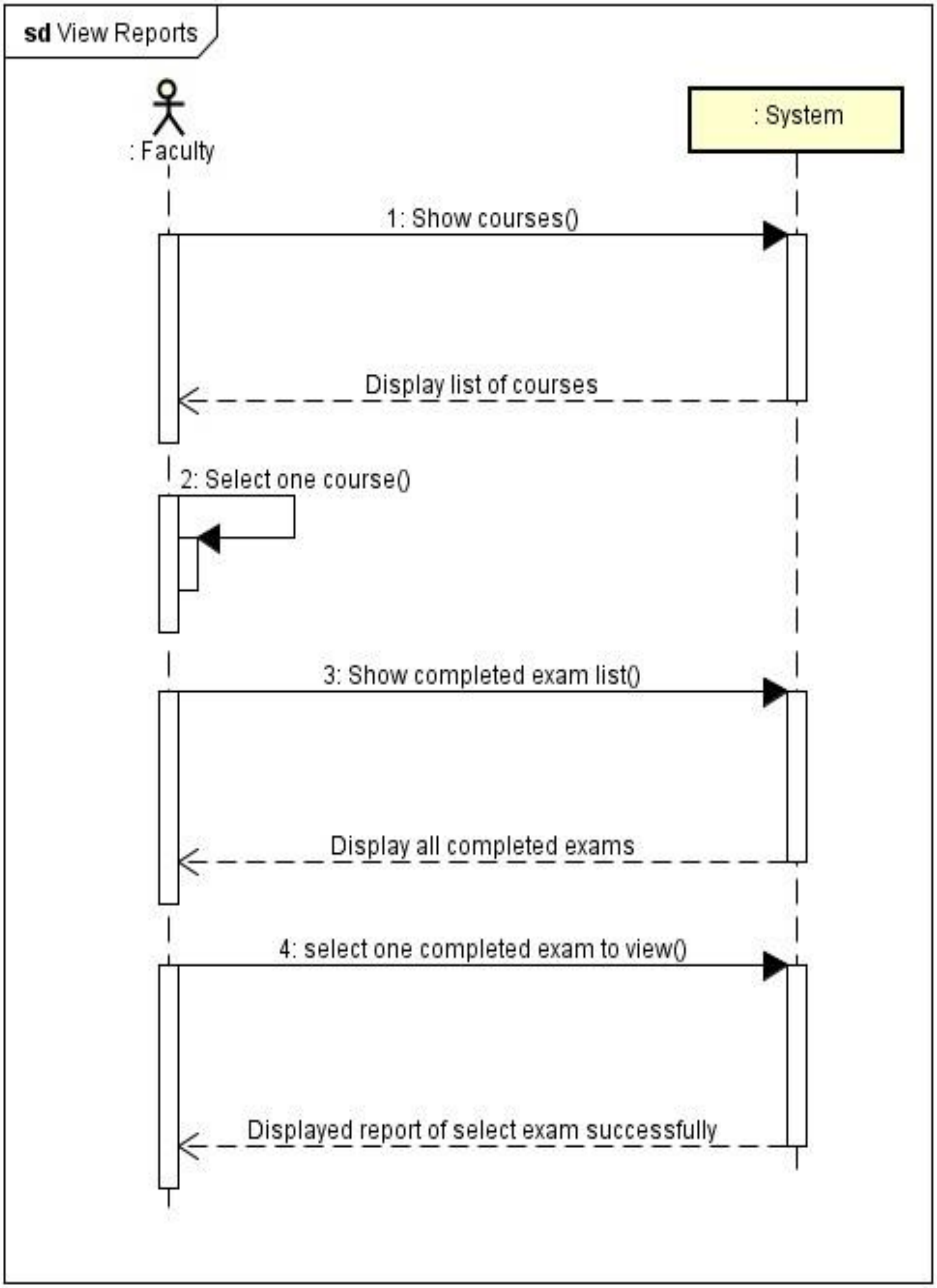
Faculty - Create Question Bank


sd Create Question bank
: Faculty
: System
: Database
1: show list of courses()
list of courses displayed
2: Select course from list()
3: create module()
3.1: update module()
module created
module created successfully
4: select module()
alt
[Update Questions]
loop
5: select manul entry of questions()
display template
6: select question type mcq,multiple ans,fill blank,T or F()
7: enter question,options and correct asnwer()
8: submit question()
8.1: create question()
question added successfully
question created successfully
9: upload bulk questions spreadsheet()
9.1: update questions bulk()
questions added successfully
questions created successfully

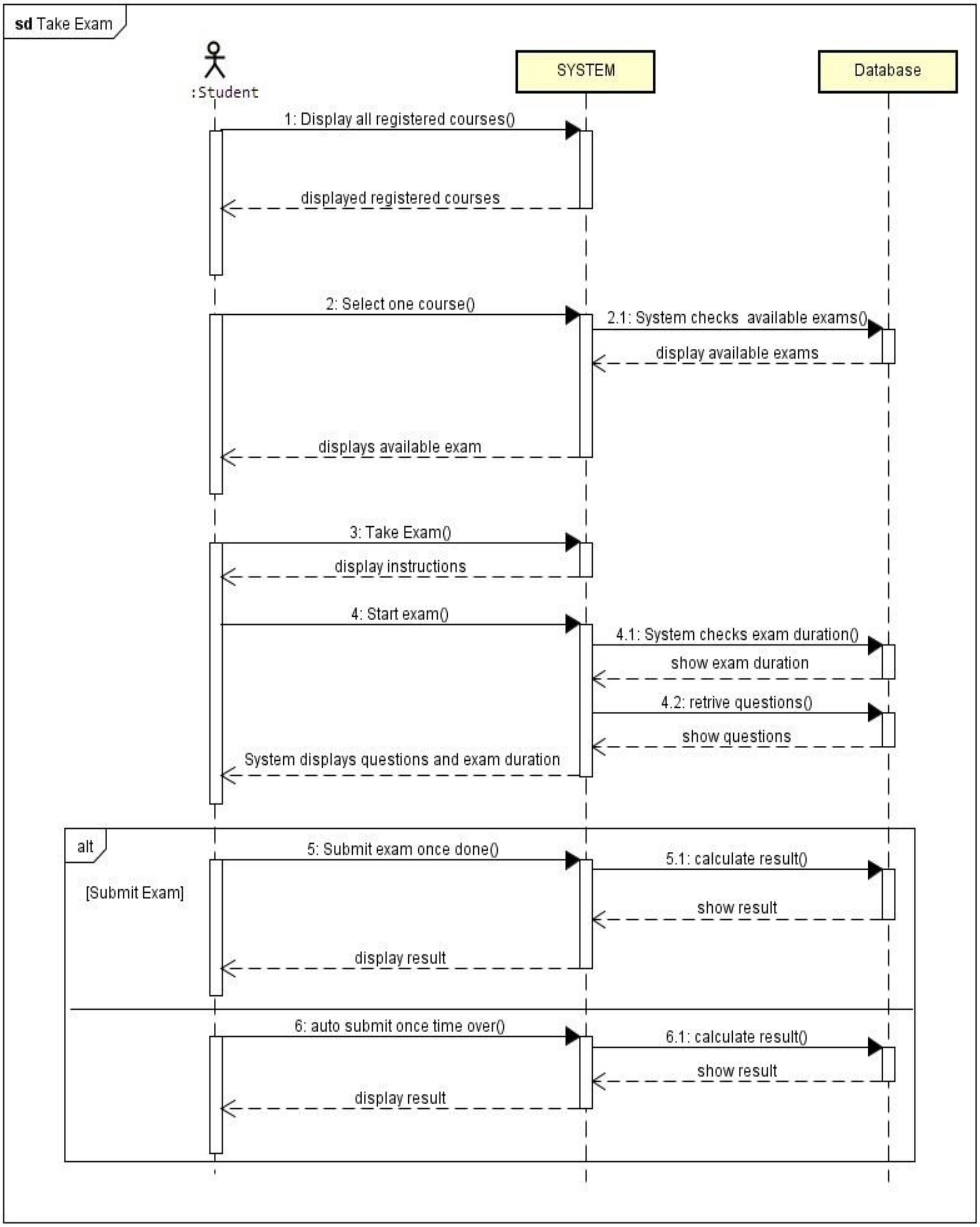
Faculty – Create Exam



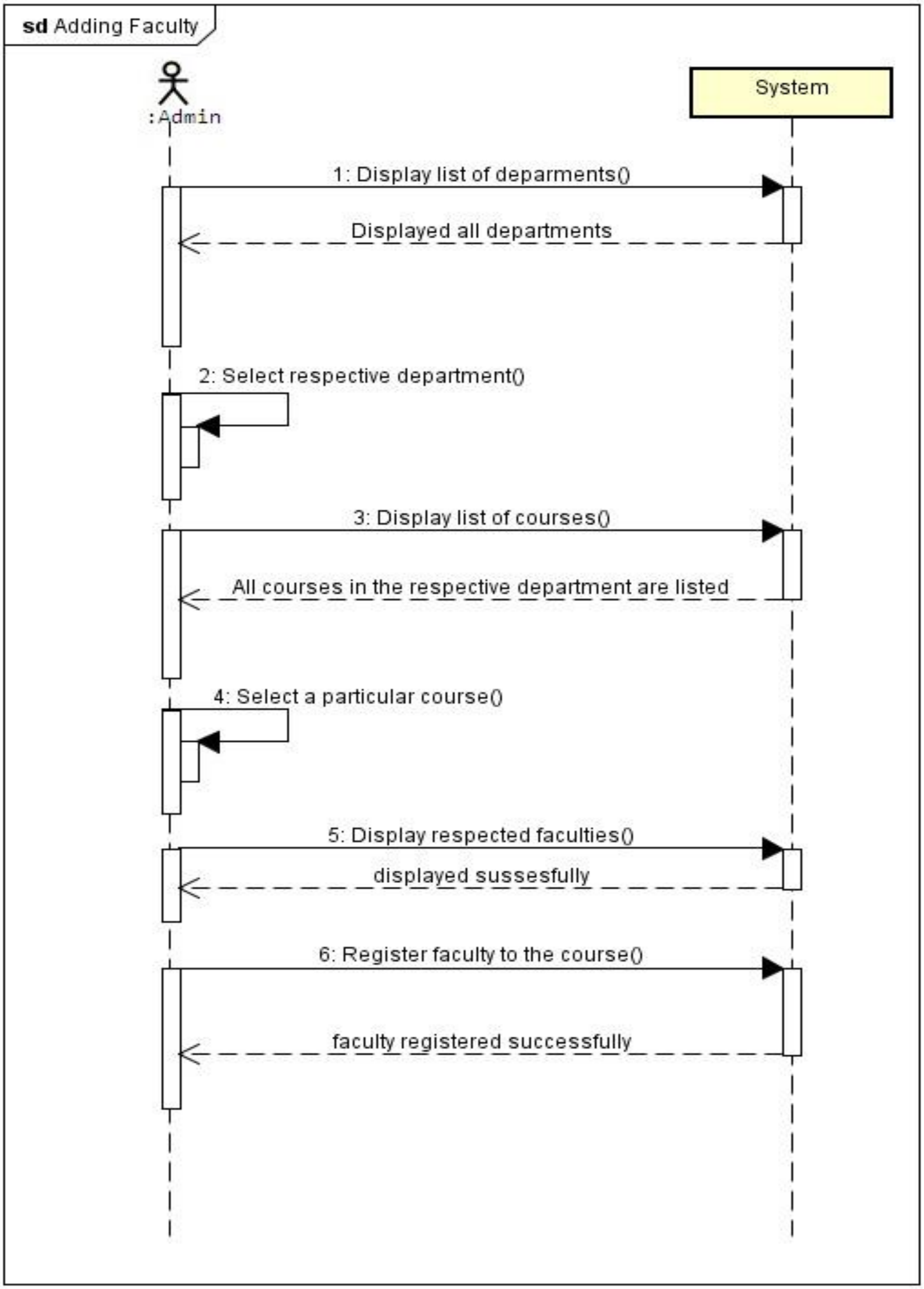
Faculty – View Reports


sd View Reports
: Faculty
: System
1: Show courses()
Display list of courses
2: Select one course()
3: Show completed exam list()
Display all completed exams
4: select one completed exam to view()
Displayed report of select exam successfully

Student – Take Exam



Admin – Add Faculty

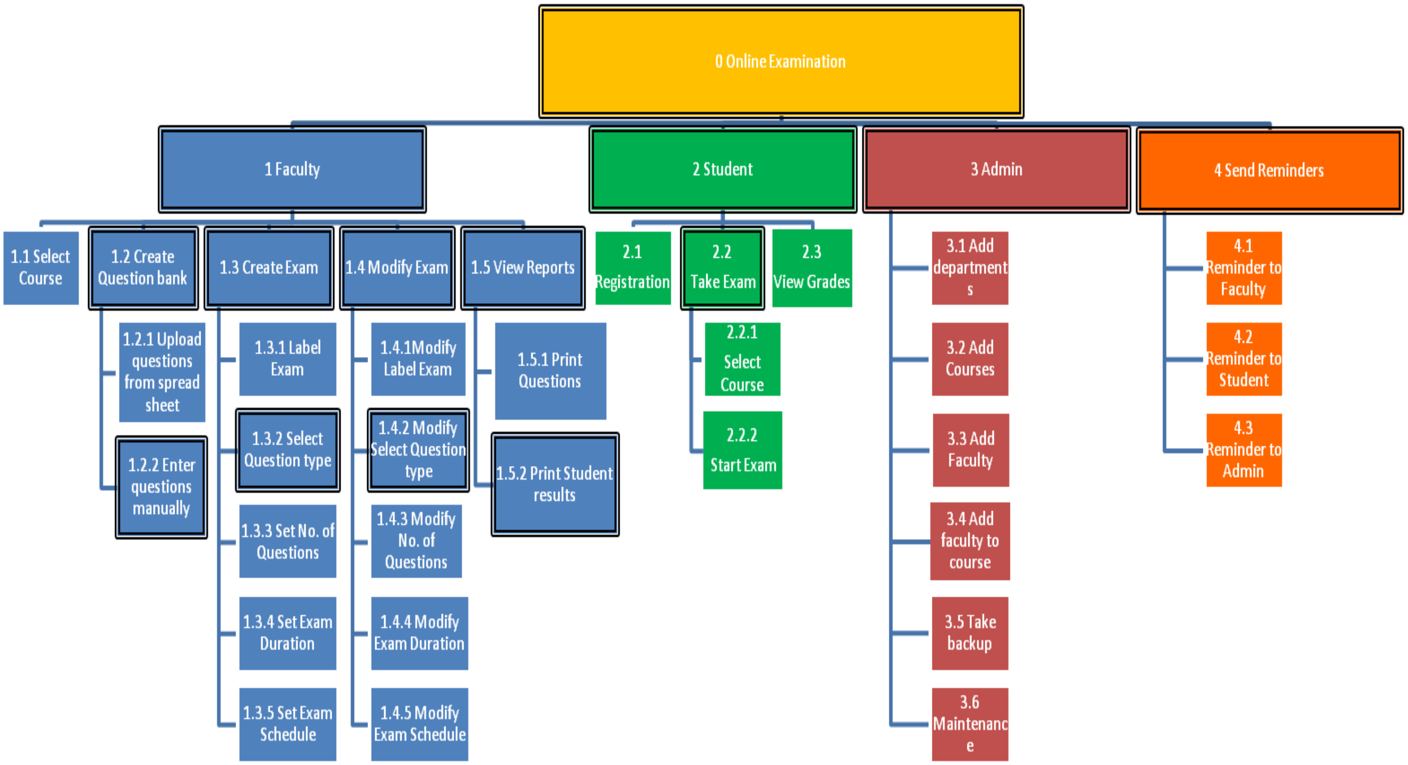

sd Adding Faculty
:Admin
System
1: Display list of deparments()
Displayed all departments
2: Select respective department()
3: Display list of courses()
All courses in the respective department are listed
4: Select a particular course()
5: Display respected faculties()
displayed sussesfully
6: Register faculty to the course()
faculty registered successfully

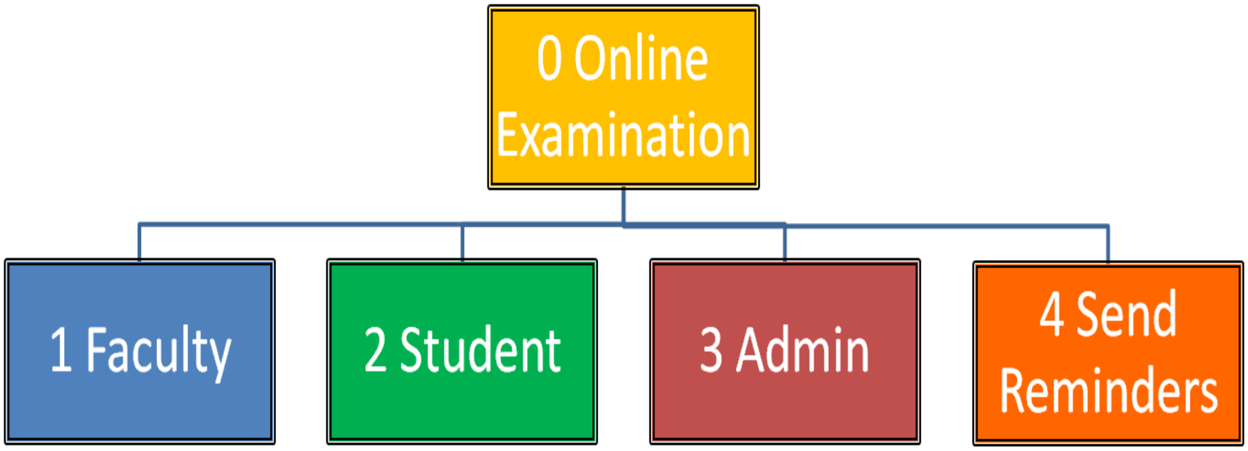
6. TASK ANALYSIS HIERARCHY LIST

- 0 Online Examination
    - 1 Faculty
    - 1.1 Select Course
    - 1.2 Create Question bank
      - 1.2.1 Upload questions from spread sheet
      - 1.2.2 Enter questions manually
    - 1.3 Create Exam
      - 1.3.1 Label Exam
      - 1.3.2 Select Question type
      - 1.3.3 Set No. of Questions
      - 1.3.4 Set Exam Duration
      - 1.3.5 Set Exam Schedule
    - 1.4 Modify Exam
      - 1.4.1Modify Label Exam
      - 1.4.2 Modify Select Question type
      - 1.4.3 Modify No. of Questions
      - 1.4.4 Modify Exam Duration
      - 1.4.5 Modify Exam Schedule
    - 1.5 View Reports
      - 1.5.1 Print Questions
      - 1.5.2 Print Student results
  - 2 Student
    - 2.1 Registration
    - 2.2Take Exam
      - 2.2.1 Select Course

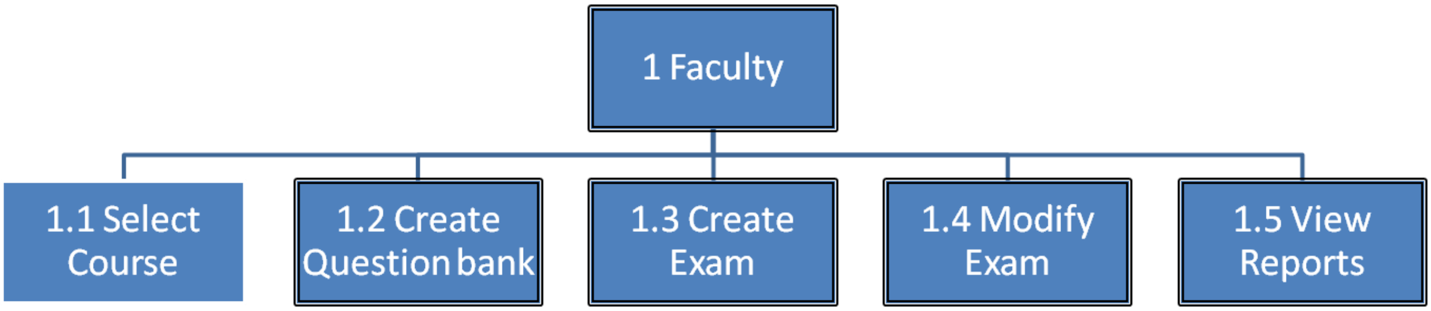
        - 2.2.2 Start Exam
    - 2.3 View Grades
- 3 Admin
    - 3.1 Add departments
    - 3.2 Add Courses
    - 3.3 Add Faculty
    - 3.4 Add faculty to course
    - 3.5 Take backup
    - 3.6 Maintenance
- 4 Send Reminders
    - 4.1 Reminder to Faculty
    - 4.2 Reminder to Student
    - 4.3 Reminder to Admin

7. TASK ANALYSIS DIAGRAM

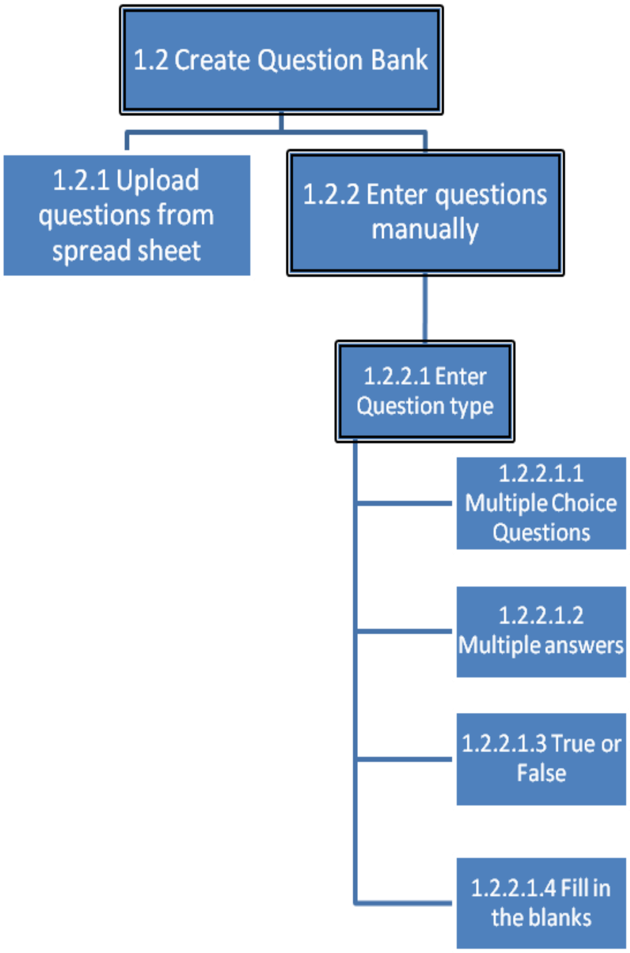

0 Online Examination
1 Faculty
2 Student
3 Admin
4 Send Reminders
1.1 Select Course
1.2 Create Question bank
1.3 Create Exam
1.4 Modify Exam
1.5 View Reports
1.2.1 Upload questions from spread sheet
1.2.2 Enter questions manually
1.3.1 Label Exam
1.3.2 Select Question type
1.3.3 Set No. of Questions
1.3.4 Set Exam Duration
1.3.5 Set Exam Schedule
1.4.1Modify Label Exam
1.4.2 Modify Select Question type
1.4.3 Modify No. of Questions
1.4.4 Modify Exam Duration
1.4.5 Modify Exam Schedule
1.5.1 Print Questions
1.5.2 Print Student results
2.1 Registration
2.2 Take Exam
2.3 View Grades
2.2.1 Select Course
2.2.2 Start Exam
3.1 Add departments
3.2 Add Courses
3.3 Add Faculty
3.4 Add faculty to course
3.5 Take backup
3.6 Maintenance
4.1 Reminder to Faculty
4.2 Reminder to Student
4.3 Reminder to Admin


0 Online Examination
1 Faculty
2 Student
3 Admin
4 Send Reminders

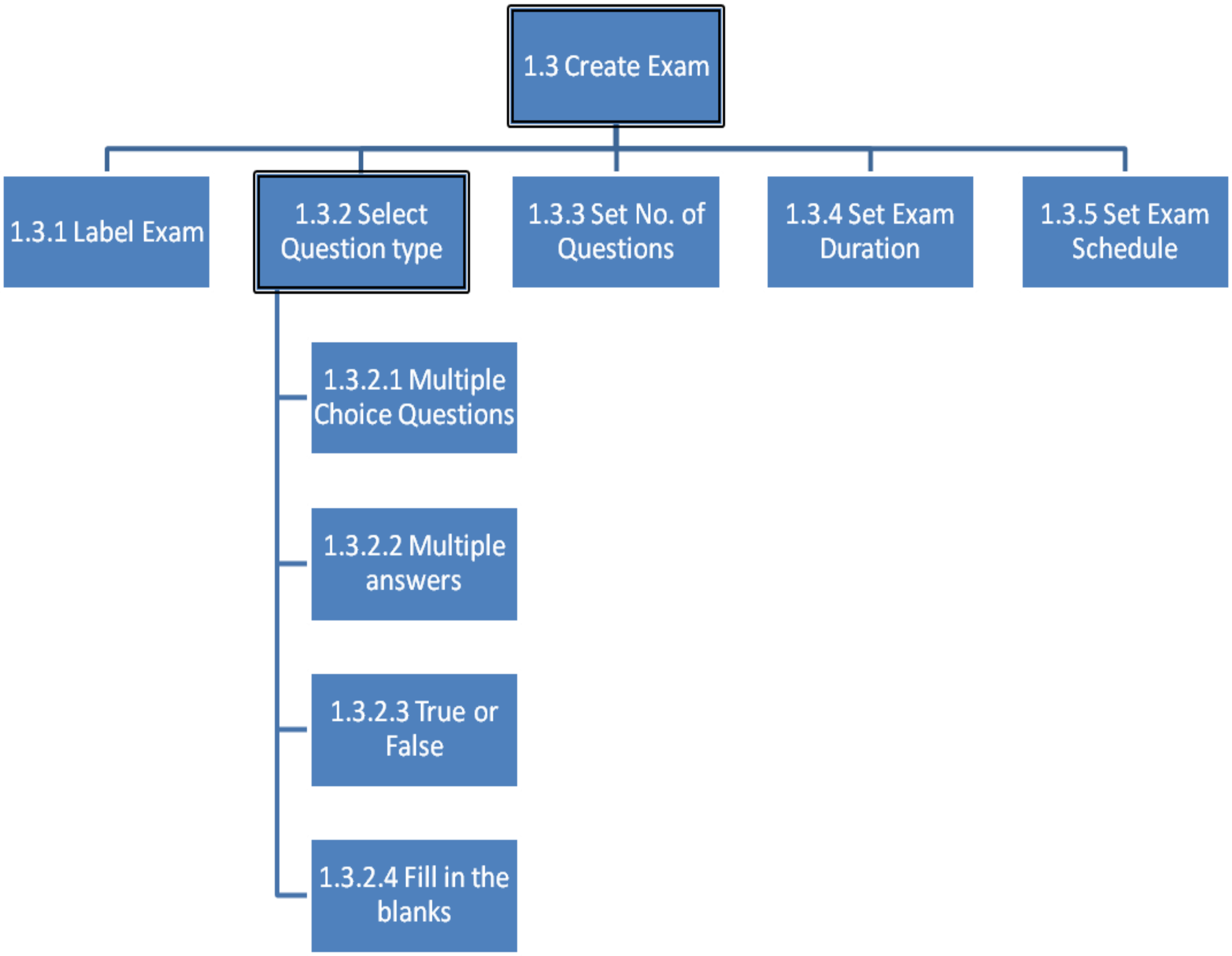
1 Faculty


1 Faculty
1.1 Select Course
1.2 Create Question bank
1.3 Create Exam
1.4 Modify Exam
1.5 View Reports

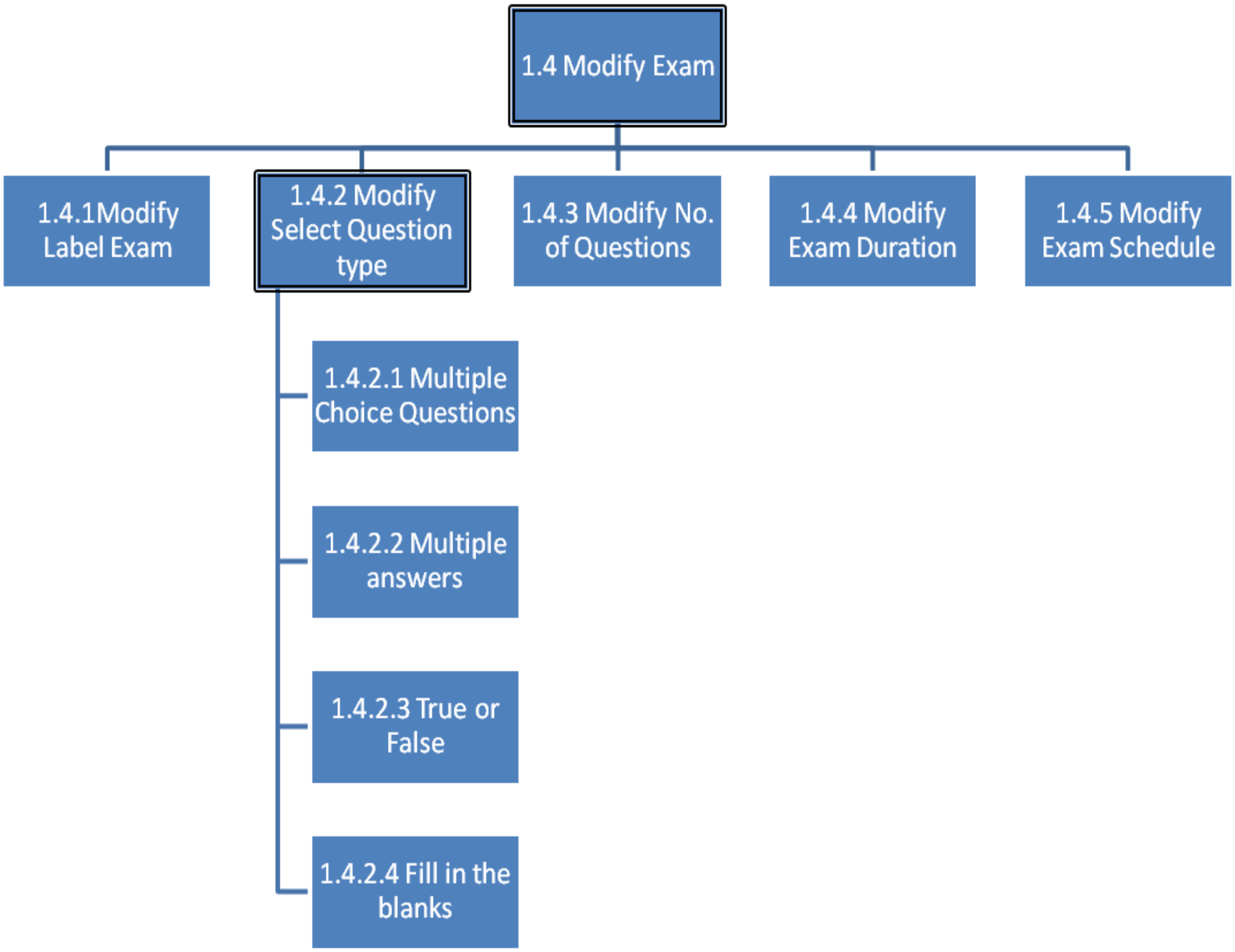
1.2 Create Question Bank


1.2 Create Question Bank
1.2.1 Upload questions from spread sheet
1.2.2 Enter questions manually
1.2.2.1 Enter Question type
1.2.2.1.1 Multiple Choice Questions
1.2.2.1.2 Multiple answers
1.2.2.1.3 True or False
1.2.2.1.4 Fill in the blanks

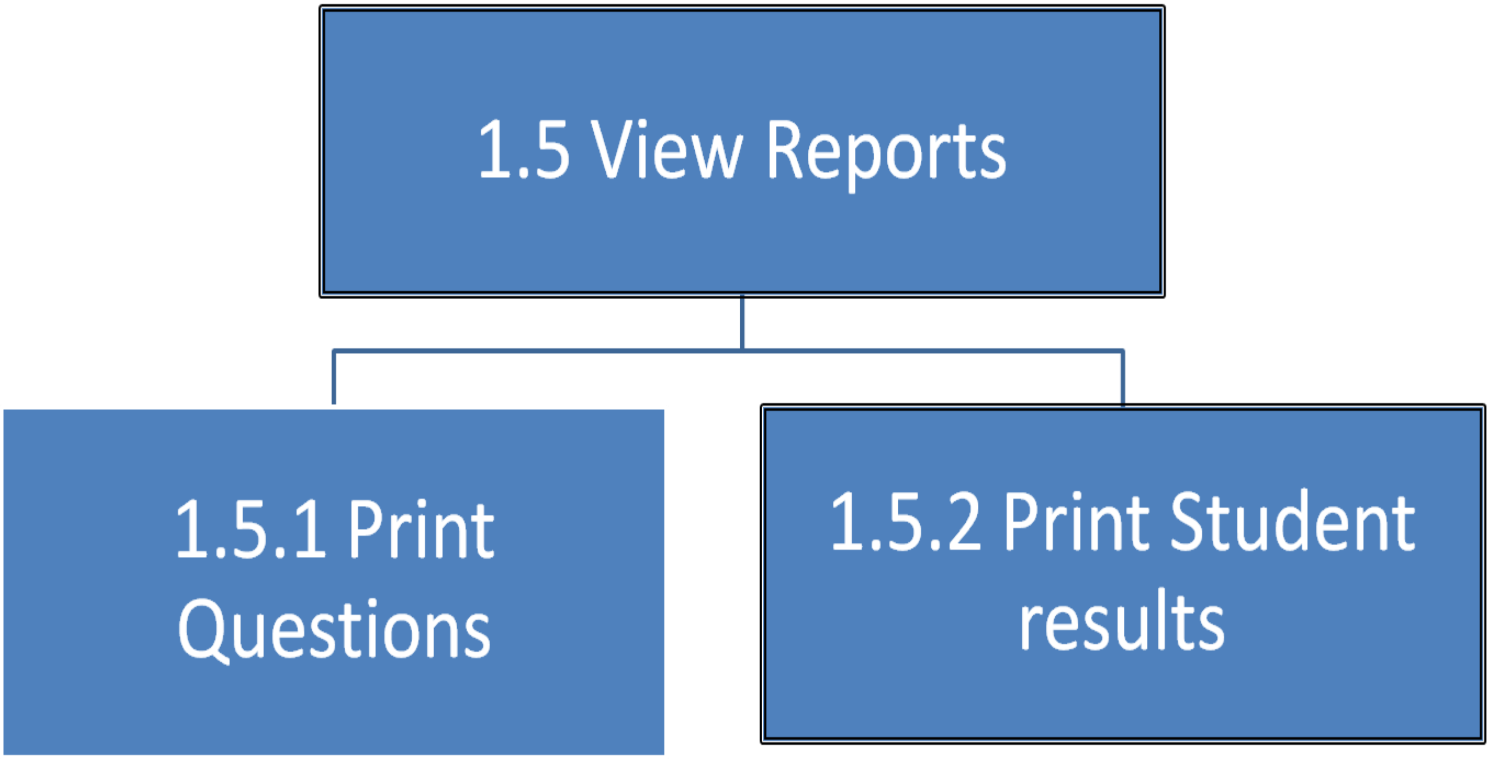
1.3 Create Exam



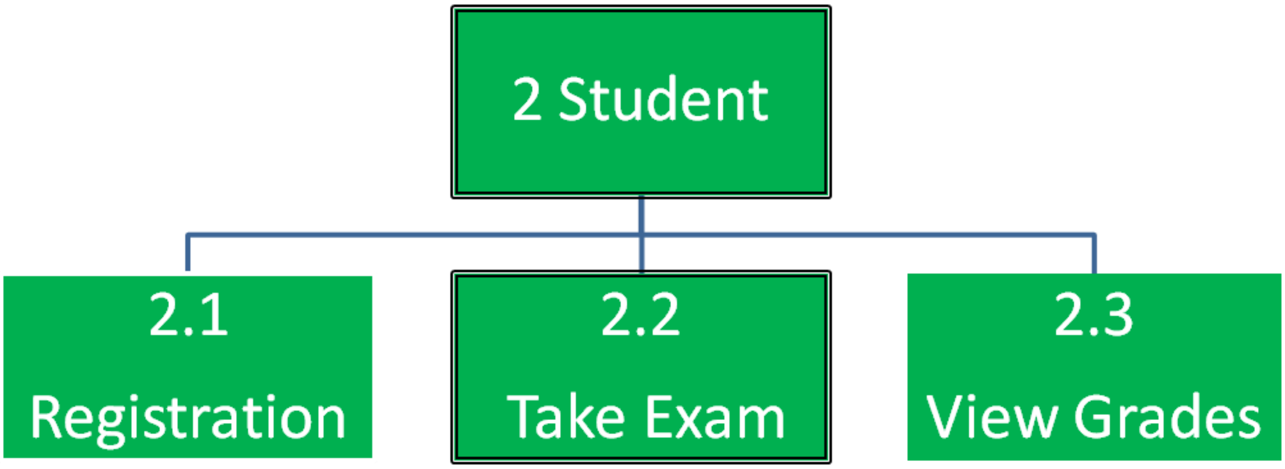
1.4 Modify Exam


1.4 Modify Exam
1.4.1Modify Label Exam
1.4.2 Modify Select Question type
1.4.3 Modify No. of Questions
1.4.4 Modify Exam Duration
1.4.5 Modify Exam Schedule
1.4.2.1 Multiple Choice Questions
1.4.2.2 Multiple answers
1.4.2.3 True or False
1.4.2.4 Fill in the blanks

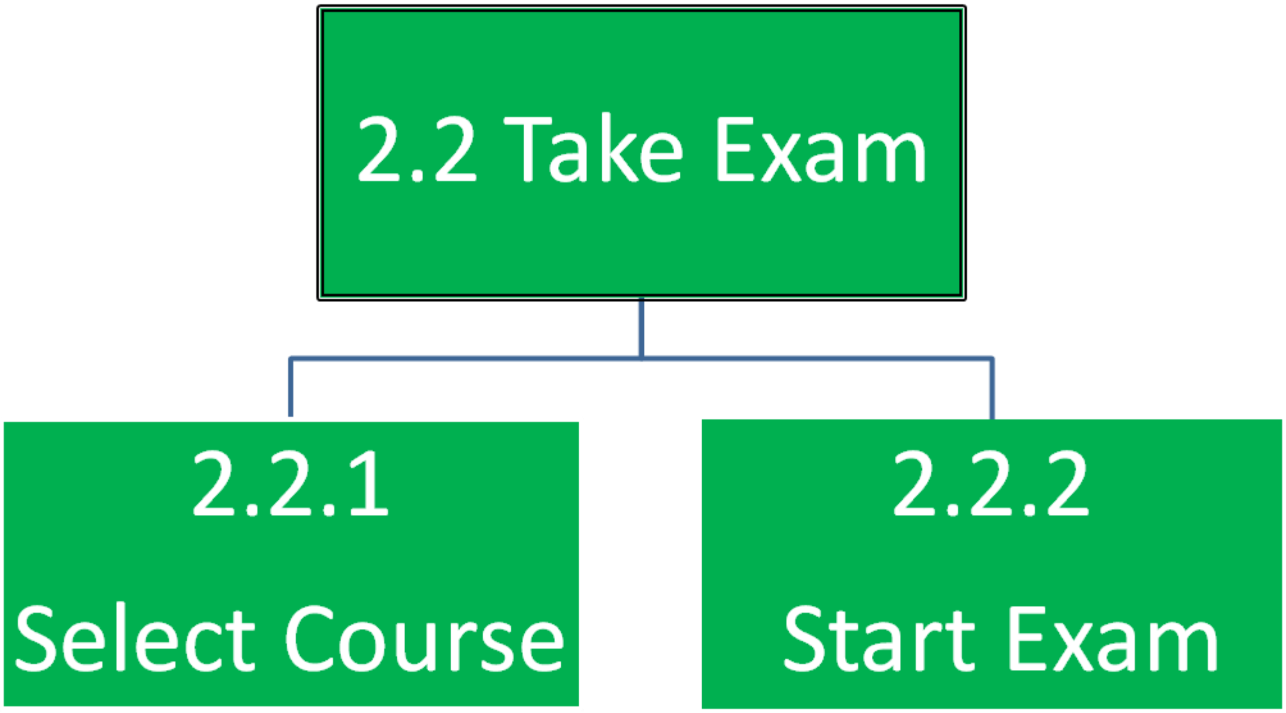
1.5 View Reports



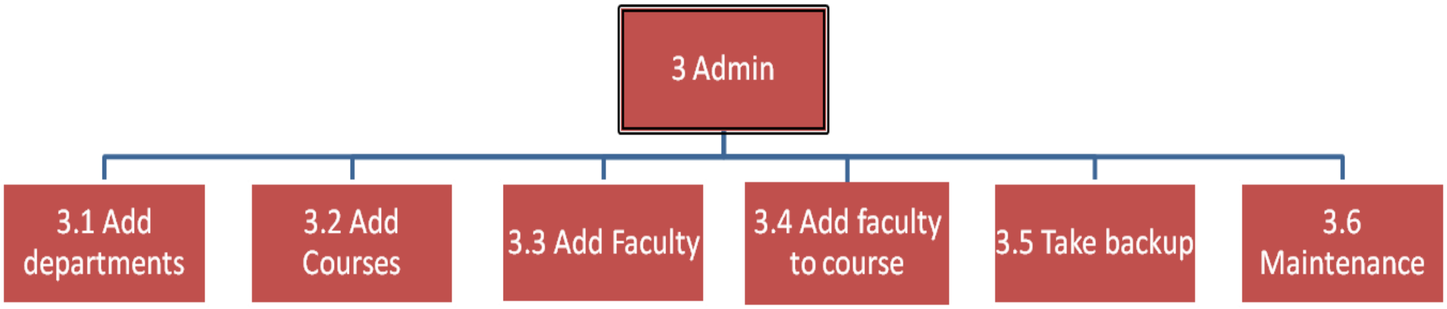
2 Student



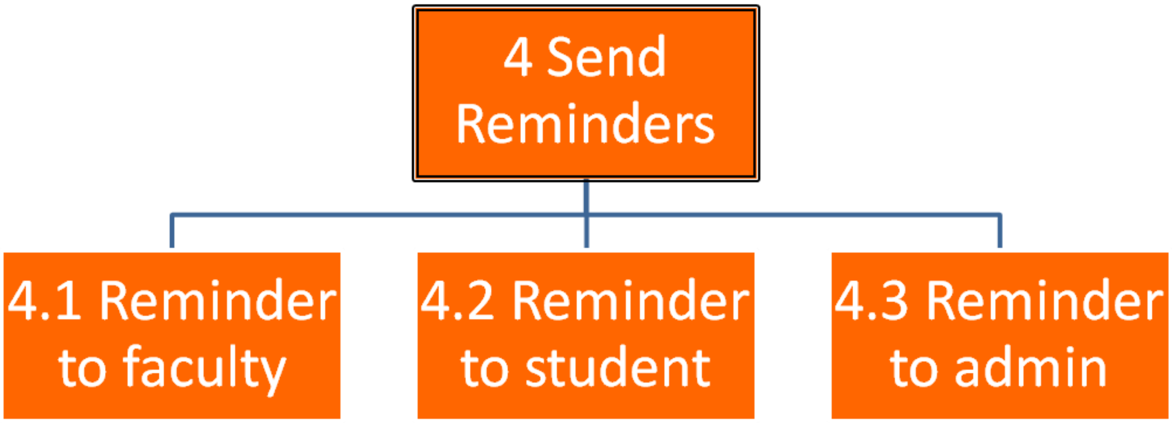
2.2 Take Exam



3 Admin


3 Admin
3.1 Add departments
3.2 Add Courses
3.3 Add Faculty
3.4 Add faculty to course
3.5 Take backup
3.6 Maintenance

4. Send Reminder



8. SPECIFICATIONS

1.1 Select course

1. **What is the goal of the task?**

   The goal is to select the course from the list of courses that the faculty is teaching for the particular semester.

2. **What subtasks define this task?**

   None

3. **Is this task a sub unit of larger task?**

   No

4. **What non interface functions does this task require?**

   The website will load the list of courses that the faculty will teaching that particular semester.

5. **What kind of inputs or actions does this task require from the user?**

   The faculty should open the website and check for the list of courses he/she shall be teaching.

6. **What kind of outputs/results are expected by performing this task?**

   The courses are loaded, so that the faculty uses it.

7. **What automatic actions does this expect from the system?**

   None

8. **What special Characteristics of the task should we record?**

   None

9. **In this sub tree, is there a task that must come before this one?**

   No

10. In this sub tree, is there a task for which this one is required to be immediately preceding? Is there any specific sequence in which the task must be performed?

No

11. Which if any primary, classes/ entities are involved in this subtask?

Database

12. How can this task fail (or end in non-completion)?

The task can fail if the courses are not loaded successfully from the database.

13. How frequently is this task performed?

Every time when the faculty selects the course.

14. How open is this task especially in terms of its sequence and inputs?

The task is performed as the faculty will be teaching more than one course and it is used to switch over between the different courses he/she is teaching.

15. What if any, are the specific usability expectations (e.g. Ease of Use) for this task and do we anticipate determining if we have satisfied the user expectations?

None

1.2 Create question bank

1. **What is the goal of the task?**

   The goal is to create a question bank for each course(s) the faculty is teaching.

2. **What subtasks define this task?**

   The following are the subtasks:
   - Upload questions from spread sheet
   - Enter questions manually

3. **Is this task a sub unit of larger task?**

   No

4. **What non interface functions does this task require?**

   None

5. **What kind of inputs or actions does this task require from the user?**

   The faculty creates the question bank for the course(s) he/she is teaching for the particular.

6. **What kind of outputs/results are expected by performing this task?**

   Question bank will be created for the particular course.

7. **What automatic actions does this expect from the system?**

   None

8. **What special Characteristics of the task should we record?**

   None

9. **In this sub tree, is there a task that must come before this one?**

   No

10. **In this sub tree, is there a task for which this one is required to be immediately preceding? Is there any specific sequence in which the task must be performed?**

    No

11. **Which if any primary, classes/ entities are involved in this subtask?**

    Database

12. **How can this task fail (or end in non-completion)?**

    The task can fail if the question bank is not created successfully.

13. **How frequently is this task performed?**

    The first time when a question bank is created for a particular course.

14. **How open is this task especially in terms of its sequence and inputs?**

    The task is performed when the faculty creates a question bank for a course.

15. **What if any, are the specific usability expectations (e.g. Ease of Use) for this task and do we anticipate determining if we have satisfied the user expectations?**

    None

1.2.1Upload questions from spread sheet

1. **What is the goal of the task?**

   The goal is to upload the questions for the question bank from a spread sheet.

2. **What subtasks define this task?**

   None

3. **Is this task a sub unit of larger task?**

   Yes, it is a sub unit of Create question bank.

4. **What non interface functions does this task require?**

   None

5. **What kind of inputs or actions does this task require from the user?**

   The faculty prepares the questions in a spread sheet and later uploads them for a particular course.

6. **What kind of outputs/results are expected by performing this task?**

   Questions in the form of spread sheet will be uploaded successfully.

7. **What automatic actions does this expect from the system?**

   None

8. **What special Characteristics of the task should we record?**

   None

9. **In this sub tree, is there a task that must come before this one?**

   No

10.In this sub tree, is there a task for which this one is required to be immediately preceding? Is there any specific sequence in which the task must be performed?

No

11.Which if any primary, classes/ entities are involved in this subtask?

Database

12.How can this task fail (or end in non-completion)?

The task can fail if the questions in spread sheet are not uploaded successfully due to network connectivity or some other technical issue.

13.How frequently is this task performed?

Whenever the faculty wants to add a new set of questions to the course's question bank or when he/she feels the need to update the questions for a particular course.

14.How open is this task especially in terms of its sequence and inputs?

The task is performed when the faculty has a set of questions in a spread sheet.

15. What if any, are the specific usability expectations (e.g. Ease of Use) for this task and do we anticipate determining if we have satisfied the user expectations?

None

1.2.2 Enter the questions manually

1. **What is the goal of the task?**

   The goal is to enter the questions manually instead of going with the uploading for the questions from a spread sheet.

2. **What subtasks define this task?**

   None

3. **Is this task a sub unit of larger task?**

   Yes, it is a sub unit of Create question bank.

4. **What non interface functions does this task require?**

   None

5. **What kind of inputs or actions does this task require from the user?**

   The faculty creates the questions manually on the website.

6. **What kind of outputs/results are expected by performing this task?**

   Questions will be successfully created when the faculty enters the questions manually on the website.

7. **What automatic actions does this expect from the system?**

   None

8. **What special Characteristics of the task should we record?**

   None

9. **In this sub tree, is there a task that must come before this one?**

   No

10. In this sub tree, is there a task for which this one is required to be immediately preceding? Is there any specific sequence in which the task must be performed?

No

11. Which if any primary, classes/ entities are involved in this subtask?

Database

12. How can this task fail (or end in non-completion)?

NA

13. How frequently is this task performed?

Whenever the faculty wants to add a new set of questions to the course’s question bank or when he/she feels the need to update the questions for a particular course.

14. How open is this task especially in terms of its sequence and inputs?

The task is performed when the faculty wants to add a new set of questions and don’t want to upload the questions in the form of a question bank.

15. What if any, are the specific usability expectations (e.g. Ease of Use) for this task and do we anticipate determining if we have satisfied the user expectations?

None

1.2.2.1 Enter Question type

1 What is the goal of the task?

The goal is to select the question type manually when the faculty decides to enter the questions manually rather than uploading the questions in the form of a spread sheet.

2 What subtasks define this task?

The following are the subtasks:

- Multiple choice questions
- Multiple answers
- True or False
- Fill in the blanks

3 Is this task a sub unit of larger task?

Yes, it is a sub unit of Enter question manually.

4 What non interface functions does this task require?

None

5 What kind of inputs or actions does this task require from the user?

The faculty selects the type of question on the website and based on the question type appropriate questions will be created by the faculty.

6 What kind of outputs/results are expected by performing this task?

The question type will have selected successfully by the faculty.

7 What automatic actions does this expect from the system?

None

8 What special Characteristics of the task should we record?

None

9 In this sub tree, is there a task that must come before this one?

No

10 In this sub tree, is there a task for which this one is required to be immediately preceding? Is there any specific sequence in which the task must be performed?

No

11 Which if any primary, classes/ entities are involved in this subtask?

Database

12 How can this task fail (or end in non-completion)?

NA

13 How frequently is this task performed?

The task is performed whenever the faculty decides to create questions manually. While creating questions manually, the faculty has to choose the type of question he/she is going to create.

14 How open is this task especially in terms of its sequence and inputs?

This task is performed when the faculty decides to enter the questions manually, during that time the faculty has to choose the type of questions whether the question will be multiple choice questions, multiple answers, True or False or Fill in the blanks.

15 What if any, are the specific usability expectations (e.g. Ease of Use) for this task and do we anticipate determining if we have satisfied the user expectations?

None

1.3 Create Exam

1 What is the goal of the task?

The goal of the task is to create an exam on the website for a particular course.

2 What subtasks define this task?

The following are the subtasks:

- Label exam
- Select question type
- Set number of questions
- Set exam duration
- Set exam schedule

3 Is this task a sub unit of larger task?

No

4 What non interface functions does this task require?

None

5 What kind of inputs or actions does this task require from the user?

The faculty after logging in to the website, selecting the course and uploading the question bank will create an exam.

6 What kind of outputs/results are expected by performing this task?

Create an exam is a larger task and several other sub tasks needs to be completed before the exam is created on the website.

7 What automatic actions does this expect from the system?

None

8 What special Characteristics of the task should we record?

None

9 In this sub tree, is there a task that must come before this one?

No

10 In this sub tree, is there a task for which this one is required to be immediately preceding? Is there any specific sequence in which the task must be performed?

No

11 Which if any primary, classes/ entities are involved in this subtask?

Database

12 How can this task fail (or end in non-completion)?

NA

13 How frequently is this task performed?

The task is performed before an exam is scheduled for the students.

14 How open is this task especially in terms of its sequence and inputs?

This task is performed when an exam needs to be scheduled for the students who are registered for a course and based on the course's schedule the faculty creates the exam whenever necessary.

15 What if any, are the specific usability expectations (e.g. Ease of Use) for this task and do we anticipate determining if we have satisfied the user expectations?

None

1.3.1 Label Exam

1 **What is the goal of the task?**

The goal of the task is to name the exam.

2 **What subtasks define this task?**

None

3 **Is this task a sub unit of larger task?**

This is a sub unit of create exam.

4 **What non interface functions does this task require?**

None

5 **What kind of inputs or actions does this task require from the user?**

The faculty after logging in to the website and uploading the question banks, will create an exam whenever required.

6 **What kind of outputs/results are expected by performing this task?**

The exam will be named successfully by the faculty.

7 **What automatic actions does this expect from the system?**

None

8 **What special Characteristics of the task should we record?**

None

9 **In this sub tree, is there a task that must come before this one?**

No

10 **In this sub tree, is there a task for which this one is required to be immediately preceding? Is there any specific sequence in which the task must be performed?**

No

11 Which if any primary, classes/ entities are involved in this subtask?

Database

12 How can this task fail (or end in non-completion)?

NA

13 How frequently is this task performed?

The task is performed whenever the faculty decides to name an exam for a particular course whether it is a mid-term or final exam.

14 How open is this task especially in terms of its sequence and inputs?

This task is performed when an exam is created by the faculty for the course.

15 What if any, are the specific usability expectations (e.g. Ease of Use) for this task and do we anticipate determining if we have satisfied the user expectations?

None

1.3.2 Select Question type

Refer 1.2.2.1 Select question type

1.3.3 Set Number of Questions

1 **What is the goal of the task?**

The goal of the task is to set the number of questions.

2 **What subtasks define this task?**

None

3 **Is this task a sub unit of larger task?**

This is a sub unit of Create exam.

4 **What non interface functions does this task require?**

None

5 **What kind of inputs or actions does this task require from the user?**

The faculty after creating the exam, will be setting the number of questions for the examination.

6 **What kind of outputs/results are expected by performing this task?**

The faculty will successfully be able to set the number of questions.

7 **What automatic actions does this expect from the system?**

None

8 **What special Characteristics of the task should we record?**

None

9 **In this sub tree, is there a task that must come before this one?**

No

10 In this sub tree, is there a task for which this one is required to be immediately preceding? Is there any specific sequence in which the task must be performed?

No

11 Which if any primary, classes/ entities are involved in this subtask?

Database

12 How can this task fail (or end in non-completion)?

NA

13 How frequently is this task performed?

The task is performed whenever there is an examination that is going to be conducted.

14 How open is this task especially in terms of its sequence and inputs?

Only the faculty can set the number of questions.

15 What if any, are the specific usability expectations (e.g. Ease of Use) for this task and do we anticipate determining if we have satisfied the user expectations?

None

1.3.4 Set Exam Duration

1 What is the goal of the task?
The goal of the task is to set the exam duration.

2 What subtasks define this task?

None

3 Is this task a sub unit of larger task?

It is a sub unit of Create exam

4 What non interface functions does this task require?

None

5 What kind of inputs or actions does this task require from the user?

The faculty will be setting the exam duration.

6 What kind of outputs/results are expected by performing this task?

The faculty will successfully be able to set the exam duration.

7 What automatic actions does this expect from the system?

None

8 What special Characteristics of the task should we record?

None

9 In this sub tree, is there a task that must come before this one?

No

10 In this sub tree, is there a task for which this one is required to be immediately preceding? Is there any specific sequence in which the task must be performed?

No

11 Which if any primary, classes/ entities are involved in this subtask?

Database

12 How can this task fail (or end in non-completion)?

NA

13 How frequently is this task performed?

The task is performed whenever there is an examination that is going to be conducted.

14 How open is this task especially in terms of its sequence and inputs?

Only the faculty can set the examination duration.

15 What if any, are the specific usability expectations (e.g. Ease of Use) for this task and do we anticipate determining if we have satisfied the user expectations?

None

1.3.5 Set Exam Schedule

1 **What is the goal of the task?**
The goal of the task is to set the exam schedule.

2 **What subtasks define this task?**

None

3 **Is this task a sub unit of larger task?**

It is a sub unit of Create exam

4 **What non interface functions does this task require?**

None

5 **What kind of inputs or actions does this task require from the user?**

The faculty will be setting the exam schedule.

6 **What kind of outputs/results are expected by performing this task?**

The faculty will successfully be able to set the exam schedule.

7 **What automatic actions does this expect from the system?**

None

8 **What special Characteristics of the task should we record?**

None

9 **In this sub tree, is there a task that must come before this one?**

No

10 **In this sub tree, is there a task for which this one is required to be immediately preceding? Is there any specific sequence in which the task must be performed?**

No

11 Which if any primary, classes/ entities are involved in this subtask?

Database

12 How can this task fail (or end in non-completion)?

NA

13 How frequently is this task performed?

The task is performed whenever there is an examination that is going to be conducted.

14 How open is this task especially in terms of its sequence and inputs?

Only the faculty can set the examination schedule.

15 What if any, are the specific usability expectations (e.g. Ease of Use) for this task and do we anticipate determining if we have satisfied the user expectations?

None

1.4.1 Modify Label Exam

1 What is the goal of the task?

The goal of the task is to modify exam label on the website.

2 What subtasks define this task?

None

3 Is this task a sub unit of larger task?

None

4 What non interface functions does this task require?

None

5 What kind of inputs or actions does this task require from the user?

The faculty open the website, selects the course and modifies the exam label.

6 What kind of outputs/results are expected by performing this task?

The faculty will successfully be able to modify the exam label.

7 What automatic actions does this expect from the system?

None

8 What special Characteristics of the task should we record?

None

9 In this sub tree, is there a task that must come before this one?

No

10 In this sub tree, is there a task for which this one is required to be immediately preceding? Is there any specific sequence in which the task must be performed?

No

11 Which if any primary, classes/ entities are involved in this subtask?

Database

12 How can this task fail (or end in non-completion)?

NA

13 How frequently is this task performed?

The task is performed whenever the faculty feels a need to modify the exam label.

14 How open is this task especially in terms of its sequence and inputs?

Only the faculty can modify the exam label.

15 What if any, are the specific usability expectations (e.g. Ease of Use) for this task and do we anticipate determining if we have satisfied the user expectations?

None

1.4.2 Modify Select Question type

1 **What is the goal of the task?**

The goal of the task is to modify the exam question type on the website.

2 **What subtasks define this task?**

The following are the sub tasks:

- Multiple choice questions
- Multiple answers
- True or False
- Fill in the blanks

3 **Is this task a sub unit of larger task?**

None

4 **What non interface functions does this task require?**

None

5 **What kind of inputs or actions does this task require from the user?**

The faculty open the website, selects the course and modifies the question type.

6 **What kind of outputs/results are expected by performing this task?**

The faculty will successfully be able to modify the question type.

7 **What automatic actions does this expect from the system?**

None

8 **What special Characteristics of the task should we record?**

None

9 In this sub tree, is there a task that must come before this one?

No

10 In this sub tree, is there a task for which this one is required to be immediately preceding? Is there any specific sequence in which the task must be performed?

No

11 Which if any primary, classes/ entities are involved in this subtask?

Database

12 How can this task fail (or end in non-completion)?

NA

13 How frequently is this task performed?

The task is performed whenever the faculty feels a need to modify the exam question type.

14 How open is this task especially in terms of its sequence and inputs?

Only the faculty can modify the exam question type.

15 What if any, are the specific usability expectations (e.g. Ease of Use) for this task and do we anticipate determining if we have satisfied the user expectations?

None

1.4.3 Modify No. of Questions

1 What is the goal of the task?

The goal of the task is to modify the number of questions on the website.

2 What subtasks define this task?

None

3 Is this task a sub unit of larger task?

None

4 What non interface functions does this task require?

None

5 What kind of inputs or actions does this task require from the user?

The faculty open the website, selects the course and modifies the number of questions for the exam.

6 What kind of outputs/results are expected by performing this task?

The faculty will successfully be able to modify the number of questions for the examination.

7 What automatic actions does this expect from the system?

None

8 What special Characteristics of the task should we record?

None

9 In this sub tree, is there a task that must come before this one?

No

10 In this sub tree, is there a task for which this one is required to be immediately preceding? Is there any specific sequence in which the task must be performed?

No

11 Which if any primary, classes/ entities are involved in this subtask?

Database

12 How can this task fail (or end in non-completion)?

NA

13 How frequently is this task performed?

The task is performed whenever the faculty feels a need to modify the number of questions for the exam.

14 How open is this task especially in terms of its sequence and inputs?

Only the faculty can modify the number of questions for the exam.

15 What if any, are the specific usability expectations (e.g. Ease of Use) for this task and do we anticipate determining if we have satisfied the user expectations?

None

1.4.4 Modify Exam Duration

1 What is the goal of the task?

The goal of the task is to modify the exam duration.

2 What subtasks define this task?

None

3 Is this task a sub unit of larger task?

None

4 What non interface functions does this task require?

None

5 What kind of inputs or actions does this task require from the user?

The faculty open the website, selects the course and modifies the exam duration.

6 What kind of outputs/results are expected by performing this task?

The faculty will successfully be able to modify the examination duration.

7 What automatic actions does this expect from the system?

None

8 What special Characteristics of the task should we record?

None

9 In this sub tree, is there a task that must come before this one?

No

10 In this sub tree, is there a task for which this one is required to be immediately preceding? Is there any specific sequence in which the task must be performed?

No

11 Which if any primary, classes/ entities are involved in this subtask?

Database

12 How can this task fail (or end in non-completion)?

NA

13 How frequently is this task performed?

The task is performed whenever the faculty feels a need to modify the exam duration.

14 How open is this task especially in terms of its sequence and inputs?

Only the faculty can modify the exam duration.

15 What if any, are the specific usability expectations (e.g. Ease of Use) for this task and do we anticipate determining if we have satisfied the user expectations?

None

1.4.5 Modify Exam Schedule

1 What is the goal of the task?

The goal of the task is to modify the exam schedule.

2 What subtasks define this task?

None

3 Is this task a sub unit of larger task?

None

4 What non interface functions does this task require?

None

5 What kind of inputs or actions does this task require from the user?

The faculty open the website, selects the course and modifies the exam schedule.

6 What kind of outputs/results are expected by performing this task?

The faculty will successfully be able to examination schedule.

7 What automatic actions does this expect from the system?

None

8 What special Characteristics of the task should we record?

None

9 In this sub tree, is there a task that must come before this one?

No

10 In this sub tree, is there a task for which this one is required to be immediately preceding? Is there any specific sequence in which the task must be performed?

No

11 Which if any primary, classes/ entities are involved in this subtask?

Database

12 How can this task fail (or end in non-completion)?

NA

13 How frequently is this task performed?

The task is performed whenever the faculty feels a need to modify the exam schedule.

14 How open is this task especially in terms of its sequence and inputs?

Only the faculty can modify the exam schedule.

15 What if any, are the specific usability expectations (e.g. Ease of Use) for this task and do we anticipate determining if we have satisfied the user expectations?

None

1.5 View Reports

1. **What is the goal of the task?**

Faculty should be able to view the reports of the students.

2. **What subtasks define this task?**
The subtasks include:

- Printing the questions of a particular exam and
- Viewing the reports of the student.

3. **Is this task a sub unit of larger task?**
No

4. **What non interface functions does this task require?**
None

5. **What kind of inputs or actions does this task require from the user?**
Faculty should login to the application.

6. **What kind of outputs/results are expected by performing this task?**
Faculty will be able to view student reports.

7. **What automatic actions does this expect from the system?**
Faculty should have the option to print the questions and reports of that exam.

8. **What special Characteristics of the task should we record?**
None

9. **In this sub tree, is there a task that must come before this one?**
The exam should be completed by all the students.

10. **In this sub tree, is there a task for which this one is required to be immediately preceding? Is there any specific sequence in which the task must be performed?**
This task must be performed only after the exam is taken by all students.

11. Which if any primary, classes/ entities are involved in this subtask?
No

12. How can this task fail (or end in non-completion)?
If the faculty is not able to view reports.

13. How frequently is this task performed?
Whenever faculty want to view reports.

14. How open is this task especially in terms of its sequence and inputs?
This task is performed when faculty want to view the reports.

15. What if any, are the specific usability expectations (e.g. Ease of Use) for this task and do we anticipate determining if we have satisfied the user expectations?
Yes, when faculty is able to view reports.

1.4.6 Print Questions

1. **What is the goal of the task?**

   Faculty should be able to print the questions of the exam.

2. **What subtasks define this task?**

   None

3. **Is this task a sub unit of larger task?**

   Yes, View reports

4. **What non interface functions does this task require?**

   None

5. **What kind of inputs or actions does this task require from the user?**

   User should view the reports.

6. **What kind of outputs/results are expected by performing this task?**

   All the questions should get print.

7. **What automatic actions does this expect from the system?**

   None

8. **What special Characteristics of the task should we record?**

   None

9. In this sub tree, is there a task that must come before this one?

The exam should be completed by all the students.

10. In this sub tree, is there a task for which this one is required to be immediately preceding? Is there any specific sequence in which the task must be performed?

This task must be performed only after the exam is taken by all students.

11. Which if any primary, classes/ entities are involved in this subtask?

No

12. How can this task fail (or end in non-completion)?

If the faculty is not able to print the questions.

13. How frequently is this task performed?

Whenever faculty want to print the questions.

14. How open is this task especially in terms of its sequence and inputs?

This task is performed when faculty want to print the questions.

15. What if any, are the specific usability expectations (e.g. Ease of Use) for this task and do we anticipate determining if we have satisfied the user expectations?

Yes, when faculty is able to print the questions.

1.4.7 Print student results

1. **What is the goal of the task?**

   Faculty should be able to print student results.

2. **What subtasks define this task?**

   None

3. **Is this task a sub unit of larger task?**

   Yes, View reports

4. **What non interface functions does this task require?**

   None

5. **What kind of inputs or actions does this task require from the user?**

   Faculty should view the reports.

6. **What kind of outputs/results are expected by performing this task?**

   All the student results should get print.

7. **What automatic actions does this expect from the system?**

   None

8. **What special Characteristics of the task should we record?**

   None

9. In this sub tree, is there a task that must come before this one?

The exam should be completed by all the students.

10. In this sub tree, is there a task for which this one is required to be immediately preceding? Is there any specific sequence in which the task must be performed?

This task must be performed only after the exam is taken by all students.

11. Which if any primary, classes/ entities are involved in this subtask?

No

12. How can this task fail (or end in non-completion)?

If the faculty is not able to print the student results

13. How frequently is this task performed?

Whenever faculty want to print the student results.

14. How open is this task especially in terms of its sequence and inputs?

This task is performed when faculty want to print the student results

15. What if any, are the specific usability expectations (e.g. Ease of Use) for this task and do we anticipate determining if we have satisfied the user expectations?

Yes, when faculty is able to print the student results.

1.0 Student

1. What is the goal of the task?

To Perform Some Student Operations.

2. What subtasks define this task?

- Registration
- Take Exam
- View Grades

3. Is this task a sub unit of larger task?

No

4. What non interface functions does this task require?

Update of student database as soon as student change or enter details.

5. What kind of inputs or actions does this task require from the user?

Student should be a valid ongoing student in the school.

6. What kinds of outputs/results are expected by performing this task?

Student can either go for Registration, Write Exam or View his/her Grades.

7. What automatic actions does this expect from the system?

Returning to respected pages selected by student.

8. What special Characteristics of the task should we record?

It gives us the gate way for the student either to register or view grades, take exam.

9. In this sub tree, is there a task that must come before this one?

This is the main parent.

10. In this sub tree, is there a task for which this one is required to be immediately preceding? Is there any specific sequence in which the task must be performed?

This is the main parent.

11. Which if any primary, classes/ entities are involved in this subtask?

This is the main parent.

12. How can this task fail (or end in non-completion)?

When the server is not responding.

13. How frequently is this task performed?

It is like a home page for every student. The frequency is very high.

14. How open is this task especially in terms of its sequence and inputs?

This task is open to main section of any sequence of input from student side.

15. What if any, are the specific usability expectations (e.g. Ease of Use) for this task and do we anticipate determining if we have satisfied the user expectations?

N/A

2.1 Registration

1. **What is the goal of the task?**

   This is one-time registration.

2. **What subtasks define this task?**

   There is no sub task.

3. **Is this task a sub unit of larger task?**

   It is a sub unit of registration.

4. **What non interface functions does this task require?**

   The database has to store the details of the student when he registers for one-time registration.

5. **What kind of inputs or actions does this task require from the user?**

   The user has to enter his/her details in the registration form.

6. **What kinds of outputs/results are expected by performing this task?**

   The student will be registered for the exam.

7. **What automatic actions does this expect from the system?**

   Saving the profile of student in data base and after completing the registration form it is directed to homepage.

8. **What special Characteristics of the task should we record?**

   Accepting the student data and saving in database which is followed by automatic return to home page.

9. **In this sub tree, is there a task that must come before this one?**

   No, this is the first task to be performed.

10. In this sub tree, is there a task for which this one is required to be immediately preceding? Is there any specific sequence in which the task must be performed?

This has to be performed I order to take the exam and then he can view results.

11. Which if any primary, classes/ entities are involved in this subtask?

Database.

12. How can this task fail (or end in non-completion)?

When the server is not responding, if the Student enters wrong format of data in the respective fields.

13. How frequently is this task performed?

Every time if the student wants to wrote the exam he has to perform this task of one-time registration.

14. How open is this task especially in terms of its sequence and inputs?

This task is open to students who take online examination.

15. What if any, are the specific usability expectations (e.g. Ease of Use) for this task and do we anticipate determining if we have satisfied the user expectations?

The student has to follow the field data types while filling the data. Failure to follow will incur in error

2.2 Take Exam

1. What is the goal of the task?

The goal of this task is to take exam.

2. What subtasks define this task?

The sub tasks are:

- Select course
- Start exam.

3. Is this task a sub unit of larger task?

Yes, it is a sub unit of the student task.

4. What non interface functions does this task require?

It needs to save the data of what student has taken the exam and it has to calculate the score on back end.

5. What kind of inputs or actions does this task require from the user?

The user has to answer to the question or we can say like the student has to take the exam.

6. What kinds of outputs/results are expected by performing this task?

The student will complete the examination.

7. What automatic actions does this expect from the system?

Saving the result of the exam taken by the student and calculating the correct answers of the exam.

8. What special Characteristics of the task should we record?

Saving huge amount of data in a small interval of time with huge number of users accessing it.

9. In this sub tree, is there a task that must come before this one?

Yes, this is a sub tree of student.

10. In this sub tree, is there a task for which this one is required to be immediately preceding? Is there any specific sequence in which the task must be performed?

The student has to register before taking the examination, so Registration is the first task that has to be completed prior to examination.

11. Which if any primary, classes/ entities are involved in this subtask?

No

12. How can this task fail (or end in non-completion)?

This task can fail when there is a server issue or failure to save the inputs of the database or having wrong calculation of the logic which leads to incorrect results.

13. How frequently is this task performed?

Every time if the student wants to write the exam he has to perform this task.

14. How open is this task especially in terms of its sequence and inputs?

This task is open to every student who takes the exam.

15. What if any, are the specific usability expectations (e.g. Ease of Use) for this task and do we anticipate determining if we have satisfied the user expectations?

N/A

2.3 View grades

1. **What is the goal of the task?**

   The goal of the task is to view grades of the attempted exam.

2. **What subtasks define this task?**
   It has no sub task.

3. **Is this task a sub unit of larger task?**
   It is a sub unit of student.

4. **What non interface functions does this task require?**
   This task is purely based on graphical representation of the student.

5. **What kind of inputs or actions does this task require from the user?**
   The user has to just view his grades and the input is from the previous task where the score is calculated and given as input to this module.

6. **What kinds of outputs/results are expected by performing this task?**
   The student will be referring to the grade score of his/her given examination.

7. **What automatic actions does this expect from the system?**
   It has no automatic action.

8. **What special Characteristics of the task should we record?**
   Viewing the result in the crystal reports will be used in this section.

9. **In this sub tree, is there a task that must come before this one?**
   Yes, the student has to take exam before viewing the result.

10. **In this sub tree, is there a task for which this one is required to be immediately preceding? Is there any specific sequence in which the task must be performed?**
    Yes, taking exam is the preceding task before viewing the result.

11. Which if any primary, classes/ entities are involved in this subtask?
No.

12. How can this task fail (or end in non-completion)?
This task can fail when the incorrect total of the student is given as input. This task can also fail with malfunction of crystal reporting tool.

13. How frequently is this task performed?
Every time the student writes the exam then this task is performed.

14. How open is this task especially in terms of its sequence and inputs?
This task is open to numerical inputs.

15. What if any, are the specific usability expectations (e.g. Ease of Use) for this task and do we anticipate determining if we have satisfied the user expectations?
The crystal report gives the clean report of the marks, which gives a ease of understanding to the student.

2.2.1 Select Course

1. **What is the goal of the task?**
   The goal of the task is to select the course.

2. **What subtasks define this task?**
   It has no sub task.

3. **Is this task a sub unit of larger task?**
   It is a sub unit of take exam.

4. **What non interface functions does this task require?**
   This has no non interface functionality.

5. **What kind of inputs or actions does this task require from the user?**
   The user has to select the course he is enrolled too.

6. **What kinds of outputs/results are expected by performing this task?**
   The student will be redirected to the selected course.

7. **What automatic actions does this expect from the system?**
   It has no automatic actions.

8. **What special Characteristics of the task should we record?**
   Viewing the desired selected course.

9. **In this sub tree, is there a task that must come before this one?**
   Yes, the student has to select the 'take exam' option.

10. **In this sub tree, is there a task for which this one is required to be immediately preceding? Is there any specific sequence in which the task must be performed?**
    Yes, taking exam is the preceding task before selecting the course.

11. **Which if any primary, classes/ entities are involved in this subtask?**
    No.

12. How can this task fail (or end in non-completion)?

This task can fail when the incorrect course is viewed instead of users selected option or breakdown of server.

13. How frequently is this task performed?

Every time the student takes the exam this task has to be performed.

14. How open is this task especially in terms of its sequence and inputs?

It is very frequent as every student has to select a course to take the exam.

15. What if any, are the specific usability expectations (e.g. Ease of Use) for this task and do we anticipate determining if we have satisfied the user expectations?

N/A

2.2.2 Start Exam

1. **What is the goal of the task?**
   The goal of the task is to start the exam.

2. **What subtasks define this task?**
   It has no sub task.

3. **Is this task a sub unit of larger task?**
   It is a sub unit of take exam.

4. **What non interface functions does this task require?**
   This task no non interface functions.

5. **What kind of inputs or actions does this task require from the user?**
   The user has to just click on start exam when he is ready.

6. **What kinds of outputs/results are expected by performing this task?**
   The student will be redirected to exam page when they click on start exam.

7. **What automatic actions does this expect from the system?**
   It has no automatic action.

8. **What special Characteristics of the task should we record?**
   There are no special characteristics.

9. **In this sub tree, is there a task that must come before this one?**
   Yes, the student has to select the course before clicking on start exam.

10. **In this sub tree, is there a task for which this one is required to be immediately preceding? Is there any specific sequence in which the task must be performed?**
    Yes, selecting exam is the preceding task.

11. **Which if any primary, classes/ entities are involved in this subtask?**
    No.

12. How can this task fail (or end in non-completion)?
This task can fail when there is crash of the server.

13. How frequently is this task performed?
Every time the student writes the exam then this task is performed.

14. How open is this task especially in terms of its sequence and inputs?
N/A

15. What if any, are the specific usability expectations (e.g. Ease of Use) for this task and do we anticipate determining if we have satisfied the user expectations?
N/A

3 Admin

1. **What is the goal of the task?**

   Admin should be able to perform different operations on the application.

2. **What subtasks define this task?**

   The subtasks are:

   - Add departments
   - Add courses
   - Add faculty
   - Add faculty to course
   - Take backup
   - Maintenance

3. **Is this task a sub unit of larger task?**

   No

4. **What non interface functions does this task require?**

   None

5. **What kind of inputs or actions does this task require from the user?**

   User should login to the application as Admin.

6. **What kind of outputs/results are expected by performing this task?**
   User can perform any kind of operation on the application.

7. **What automatic actions does this expect from the system?**

   User should automatically get admin access to the application.

8. What special Characteristics of the task should we record?

None

9. In this sub tree, is there a task that must come before this one?

No

10. In this sub tree, is there a task for which this one is required to be immediately preceding? Is there any specific sequence in which the task must be performed?

No

11. Which if any primary, classes/ entities are involved in this subtask?

No

12. How can this task fail (or end in non-completion)?

If the user is not able to perform any modifications to the application.

13. How frequently is this task performed?

Whenever application needs maintenance.

14. How open is this task especially in terms of its sequence and inputs?

This task is performed when administrator want to access the application.

15. What if any, are the specific usability expectations (e.g. Ease of Use) for this task and do we anticipate determining if we have satisfied the user expectations?

The design of the application satisfies the user.

3.1 Add Departments

1. **What is the goal of the task?**
   Administrator should be able to add/modify any department information to the application.

2. **What subtasks define this task?**

   No

3. **Is this task a sub unit of larger task?**

   Yes, it is sub unit of Admin.

4. **What non interface functions does this task require?**
   The changes done by the admin in this subtask should get committed in the database.

5. **What kind of inputs or actions does this task require from the user?**

   User should have admin access to perform this task

6. **What kind of outputs/results are expected by performing this task?**

   Departments should successfully be added to the application.

7. **What automatic actions does this expect from the system?**

   None

8. **What special Characteristics of the task should we record?**

   None

9. In this sub tree, is there a task that must come before this one?

User should login as administrator.

10. In this sub tree, is there a task for which this one is required to be immediately preceding? Is there any specific sequence in which the task must be performed?

No

11. Which if any primary, classes/ entities are involved in this subtask?

No

12. How can this task fail (or end in non-completion)?

This sub task can be noted as a failure if user is not able to add departments.

13. How frequently is this task performed?

Whenever application requires maintenance.

14. How open is this task especially in terms of its sequence and inputs?

This task is performed before adding any course or faculty to the new department.

15. What if any, are the specific usability expectations (e.g. Ease of Use) for this task and do we anticipate determining if we have satisfied the user expectations?

Yes, User will be satisfied when he is able to add department successfully.

3.2 Add Courses

1. **What is the goal of the task?**

   Administrator will add/modify the courses in the application.

2. **What subtasks define this task?**

   No.

3. **Is this task a sub unit of larger task?**
   Admin.

4. **What non interface functions does this task require?**

   At least one department should have been added to the application.

5. **What kind of inputs or actions does this task require from the user?**

   User should give a proper name and select the respective department.

6. **What kind of outputs/results are expected by performing this task?**

   Course should get added to the respective department.

7. **What automatic actions does this expect from the system?**

   None

8. **What special Characteristics of the task should we record?**

   None

9. In this sub tree, is there a task that must come before this one?

Yes, department should be added before this task

10. In this sub tree, is there a task for which this one is required to be immediately preceding? Is there any specific sequence in which the task must be performed?

This task should be performed only after adding any new department and before adding any new faculty.

11. Which if any primary, classes/ entities are involved in this subtask?

Yes, Department.

12. How can this task fail (or end in non-completion)?

When the admin is not able to add course in the application.

13. How frequently is this task performed?

Whenever a new course is introduced.

14. How open is this task especially in terms of its sequence and inputs?

This tasks are performed by selecting a respective department.

15. What if any, are the specific usability expectations (e.g. Ease of Use) for this task and do we anticipate determining if we have satisfied the user expectations?

Yes, User will be satisfied if he is able to add course to the respective department.

3.3 Add Faculty

1. **What is the goal of the task?**

   Administrator will add/modify the faculty details in the application.

2. **What subtasks define this task?**

   No.

3. **Is this task a sub unit of larger task?**

   Admin.

4. **What non interface functions does this task require?**

   The faculty details should get committed in the database.

5. **What kind of inputs or actions does this task require from the user?**

   User should give a proper name and select the respective department.

6. **What kind of outputs/results are expected by performing this task?**

   Faculty should get added to the respective department.

7. **What automatic actions does this expect from the system?**

   None

8. **What special Characteristics of the task should we record?**

   None

9. In this sub tree, is there a task that must come before this one?

Yes, Department should get added to the application.

10. In this sub tree, is there a task for which this one is required to be immediately preceding? Is there any specific sequence in which the task must be performed?

This task should be performed only after adding any new department.

11. Which if any primary, classes/ entities are involved in this subtask?

Yes, Department.

12. How can this task fail (or end in non-completion)?

When the admin is not able to add faculty in the application.

13. How frequently is this task performed?

Whenever a new faculty joined.

14. How open is this task especially in terms of its sequence and inputs?

This tasks is performed by selecting a respective department.

15. What if any, are the specific usability expectations (e.g. Ease of Use) for this task and do we anticipate determining if we have satisfied the user expectations?

Yes, User will be satisfied if he is able to add faculty to the respective department.

3.4 Add Faculty to course

1. **What is the goal of the task?**
   Administrator has to add the faculty to the respective course in the application.

2. **What subtasks define this task?**

   No.

3. **Is this task a sub unit of larger task?**

   Admin.

4. **What non interface functions does this task require?**

   None

5. **What kind of inputs or actions does this task require from the user?**

   User should select the right faculty to the right course.

6. **What kind of outputs/results are expected by performing this task?**

   Faculty should get added to the respective course.

7. **What automatic actions does this expect from the system?**

   Faculty should be mapped to the respective course.

8. **What special Characteristics of the task should we record?**

   None

9. In this sub tree, is there a task that must come before this one?

Yes, adding the particular faculty to the application.

10. In this sub tree, is there a task for which this one is required to be immediately preceding? Is there any specific sequence in which the task must be performed?

This task should be performed only after adding that faculty to the application.

11. Which if any primary, classes/ entities are involved in this subtask?

Yes, Faculty.

12. How can this task fail (or end in non-completion)?

This task will fail if the respective faculty is not found in the application.

13. How frequently is this task performed?

Whenever a new faculty is added to the application.

14. How open is this task especially in terms of its sequence and inputs?

This tasks are performed by selecting a respective faculty and course.

15. What if any, are the specific usability expectations (e.g. Ease of Use) for this task and do we anticipate determining if we have satisfied the user expectations?

Yes, User will be satisfied if he is able to add faculty to the respective course.

3.5 Take backup

1. **What is the goal of the task?**

   Administrator has to back up the application.

2. **What subtasks define this task?**

   No.

3. **Is this task a sub unit of larger task?**

   Admin.

4. **What non interface functions does this task require?**

   None

5. **What kind of inputs or actions does this task require from the user?**

   Admin takes back up from time to time.

6. **What kind of outputs/results are expected by performing this task?**

   Backup is successful.

7. **What automatic actions does this expect from the system?**

   None

8. **What special Characteristics of the task should we record?**

   None

9. In this sub tree, is there a task that must come before this one?

None

10. In this sub tree, is there a task for which this one is required to be immediately preceding? Is there any specific sequence in which the task must be performed?

None

11. Which if any primary, classes/ entities are involved in this subtask?

None.

12. How can this task fail (or end in non-completion)?

If the backup of the application is failed.

13. How frequently is this task performed?

Daily.

14. How open is this task especially in terms of its sequence and inputs?

Taking backup of the application is an easy task.

15. What if any, are the specific usability expectations (e.g. Ease of Use) for this task and do we anticipate determining if we have satisfied the user expectations?

None.

3.6 Maintenance

1. **What is the goal of the task?**

Maintenance includes the application performance/database performance/User interface redesign.

2. **What subtasks define this task?**

No.

3. **Is this task a sub unit of larger task?**

Admin.

4. **What non interface functions does this task require?**

None

5. **What kind of inputs or actions does this task require from the user?**

Normal users should not connect the application during maintenance.

6. **What kind of outputs/results are expected by performing this task?**

The performance and user interface of the application would be better.

7. **What automatic actions does this expect from the system?**

None

8. What special Characteristics of the task should we record?

None

9. In this sub tree, is there a task that must come before this one?

No

10. In this sub tree, is there a task for which this one is required to be immediately preceding? Is there any specific sequence in which the task must be performed?

Application should get online once maintenance is over.

11. Which if any primary, classes/ entities are involved in this subtask?

None.

12. How can this task fail (or end in non-completion)?

When the maintenance doesn't bring any betterment in the application.

13. How frequently is this task performed?

Weekly.

14. How open is this task especially in terms of its sequence and inputs?

Maintenance is performed only after notifying all the users.

15. What if any, are the specific usability expectations (e.g. Ease of Use) for this task and do we anticipate determining if we have satisfied the user expectations?

No.

4.0 Reminder

1. **What is the goal of the task?**
The goal is to check reminder and send notifications to the admin, faculty and student.

2. **What subtasks define this task?**
It has the following sub-task .

3. **Is this task a sub unit of larger task?**
No.

4. **What non interface functions does this task require?**
The application has to check timer in the data base and then notify to actors.

5. **What kind of inputs or actions does this task require from the user?**
The system checks the time and date for the reminder.

6. **What kind of outputs/results are expected by performing this task?**
Notification of the reminders will be sent to the actor's.

7. **What automatic actions does this expect from the system?**
Sending notification to the actors.

8. **What special Characteristics of the task should we record?**
Sending automatic reminder's to all actors at their respective time interval.

9. **In this sub tree, is there a task that must come before this one?**
This is the parent task.

10. **In this sub tree, is there a task for which this one is required to be immediately preceding? Is there any specific sequence in which the task must be performed?**
No, this is parent task.

11. **Which if any primary, classes/ entities are involved in this subtask?**
Yes, database is the main entity in this task.

12. **How can this task fail (or end in non-completion)?**

When the reminder is not sent to particular actor.

13. How frequently is this task performed?

Whenever a reminder is set this task has its implication.

14. How open is this task especially in terms of its sequence and inputs?

This tasks is performed by the system and the input will be date and time.

15. What if any, are the specific usability expectations (e.g. Ease of Use) for this task and do we anticipate determining if we have satisfied the user expectations?

Yes, User will be satisfied by getting notifications of reminder.

4.1 Faculty Reminder

1. **What is the goal of the task?**
   The goal is to check reminder send notifications to the faulty.

2. **What subtasks define this task?**
   It has no sub task.

3. **Is this task a sub unit of larger task?**
   Yes, it is the sub task of send reminder.

4. **What non interface functions does this task require?**
   The application has to check timer in the data base and then notify to faculty.

5. **What kind of inputs or actions does this task require from the user?**
   The system checks the time and date for the reminder to work.

6. **What kind of outputs/results are expected by performing this task?**
   Notification of the reminders will be sent to the faculty.

7. **What automatic actions does this expect from the system?**
   Sending notification to the faculty.

8. **What special Characteristics of the task should we record?**
   Sending automatic reminder's to faculty at respective time interval.

9. **In this sub tree, is there a task that must come before this one?**
   No.

10. **In this sub tree, is there a task for which this one is required to be immediately preceding? Is there any specific sequence in which the task must be performed?**
    No.

11. **Which if any primary, classes/ entities are involved in this subtask?**
    Yes, database is the main entity in this task.

12. **How can this task fail (or end in non-completion)?**

When the reminder is not sent to faculty.

13. **How frequently is this task performed?**

Whenever a reminder is set to faculty this task has its implication.

14. **How open is this task especially in terms of its sequence and inputs?**

This tasks is performed by the system and it is constrained to only faculty, the input will be date and time.

15. **What if any, are the specific usability expectations (e.g. Ease of Use) for this task and do we anticipate determining if we have satisfied the user expectations?**

Yes, Faculty will be satisfied by getting notifications of reminder.

4.2 Student Reminder

1. **What is the goal of the task?**
   The goal is to check reminder and send notifications to the student.

2. **What subtasks define this task?**
   It has no sub-task .

3. **Is this task a sub unit of larger task?**
   Yes, Reminder.

4. **What non interface functions does this task require?**
   The application has to check timer in the data base and then notify to student.

5. **What kind of inputs or actions does this task require from the user?**
   The system checks the time and date for the reminder to work.

6. **What kind of outputs/results are expected by performing this task?**
   Notification of the reminders will be sent to the student.

7. **What automatic actions does this expect from the system?**
   Sending notification to the student.

8. **What special Characteristics of the task should we record?**
   Sending automatic reminder's to student at respective time interval.

9. **In this sub tree, is there a task that must come before this one?**
   No.

10. **In this sub tree, is there a task for which this one is required to be immediately preceding? Is there any specific sequence in which the task must be performed?**
    No.

11. **Which if any primary, classes/ entities are involved in this subtask?**
    Yes, database is the main entity in this task.

12. **How can this task fail (or end in non-completion)?**

When the reminder is not sent to student.

13. How frequently is this task performed?

Whenever a reminder is set to student this task has its implication.

14. How open is this task especially in terms of its sequence and inputs?

This tasks is performed by the system and it is constrained to only admin, the input will be date and time.

15. What if any, are the specific usability expectations (e.g. Ease of Use) for this task and do we anticipate determining if we have satisfied the user expectations?

Yes, Student will be satisfied by getting notifications of reminder.

4.3Admin Reminder

1. **What is the goal of the task?**
   The goal is to check reminder send notifications to the admin.

2. **What subtasks define this task?**
   It has no sub task.

3. **Is this task a sub unit of larger task?**
   Yes, it is the sub task of send reminder.

4. **What non interface functions does this task require?**
   The application has to check timer in the data base and then notify to admin.

5. **What kind of inputs or actions does this task require from the user?**
   The system checks the time and date for the reminder to work.

6. **What kind of outputs/results are expected by performing this task?**
   Notification of the reminders will be sent to the admin.

7. **What automatic actions does this expect from the system?**
   Sending notification to the admin.

8. **What special Characteristics of the task should we record?**
   Sending automatic reminder’s to admin at respective time interval.

9. **In this sub tree, is there a task that must come before this one?**
   No.

10. **In this sub tree, is there a task for which this one is required to be immediately preceding? Is there any specific sequence in which the task must be performed?**
    No.

11. **Which if any primary, classes/ entities are involved in this subtask?**
    Yes, database is the main entity in this task.

12. **How can this task fail (or end in non-completion)?**

When the reminder is not sent to admin.

13. How frequently is this task performed?

Whenever a reminder is set to admin this task has its implication.

14. How open is this task especially in terms of its sequence and inputs?

This tasks is performed by the system and it is constrained to only admin, the input will be date and time.

15. What if any, are the specific usability expectations (e.g. Ease of Use) for this task and do we anticipate determining if we have satisfied the user expectations?

Yes, Admin will be satisfied by getting notifications of reminder.