



This is to certify that

Mr./Ms:

Student789491 ANH TRAN TUAN
Student789675 HIEN LAM VAN
Student789657 DUY VO THANH
Student788508 AN PHAM VO TRUONG

Has successfully Designed & Developed

--- Result Declarration Board ---

Submitted by:

Group 5

Date of Issue:

15/03/2015

Authorized Signature:

I. Problem Definition (Form No. 1):

Introduction:

Kingston Academy is an educational institution, which has three branches situated at three different locations of the city. In each branch there are more than 5000 students studying in the institution. Actually they maintain the records of the staff and the students through a stand alone application that is build by us. The records include the Personal records, Academic Records, Administration records, Professional records, Marks and other extracurricular activities.

Generally to increase the competition level between the students of the three branches and between the branches, they started the procedure where after conducting the exams the answer sheets will be interchanged between the branches for evaluation.

Later the first ten students who secured the maximum marks (toppers) and the institution from which the maximum toppers are there they will be provided with some prizes. Because of this there is a competition between the students and the branches. Once the examinations are completed the students are provided a week days or more number of days of vacation based on the vacation time like summer holidays, Christmas holidays, etc. In between the vacation period the result will be released and so because of this the students need to come back to the institution and then check their results and can apply for reevaluation if they feel if there is something that went wrong.

To increase the comfort level of the students, the institution as decided to incorporate a portal completely for the examination results. So they had approached us to help them by building the website for declaring the examination results.

A web-site incorporating all the information about the Result Declation Board is required with following functionality

1. The database for holding the following details:
 - Users information (the Academic department people and the admin)
 - Student's information like curriculum, semester, marks, etc.
2. Based on the user the application should judge who has logged in (Guest/Admin/Academic Department People) and display the site with the appropriate functionalities accordingly
3. The total marks and the percentage will be calculated by the application itself, based on the marks secured by the student and the maximum and minimum marks as assigned to that particular subject in that particular curriculum and semester
4. Based on the minimum marks of the subject the application should judge whether the student is failed or passed in that subject.
5. The results are to be entered and displayed in the form of a table.

II. Customer Requirement Specification(CRS) (Form No. 2):

Client: *Kingston Academy*

Business / Project Objective:

Website is capable of managing large records of communication service: details of users, students, subjects, marks... Records are stored in database which can automate the system. This website can easily solve the complexity and vastness of the system.

We provide convenient way for student to view mark, they can enquire regarding the connection by directly visiting to the website in anywhere which have internet and browser.

Academic department can easily search for the information of students, besides they can easily search for the mark of the student through the student roll number and edit, delete or update some table they already have permission

Beside, to increase the comfort level of students, students can register for re-evaluation or re-counting through internet too. Specially, system are update in real time with queries and newest notifications.

- **Input provide by the client**

- Inputs to the System:

- Student: Student Identity, Student Roll No, Student Name, Student Gender, Student Gender, Student Address, Student Phone, Student Email
- Staff: Staff Identity, Staff Password, Staff Full Name, Staff Address, Staff Phone, Staff Email, Staff Gender, Staff Question, Staff Answer, Staff Status, Branch Identity
- Branch: Branch Identity, Branch Name, Branch Address, Branch Email, Branch Phone
- Subject: Subject Identity, Subject Name
- Notification: Notification Identity, Notification Information, Notification Date
- Semester: Semester Identity, Semester Name, Curriculum Identity
- Curriculum: Curriculum Identity, Curriculum Name, Curriculum Period, Curriculum Fee
- Examination: Examination Identity, Examination Date, Examination Times
- ExamMark: Mark, Student Roll No, Subject Identity, Staff Identity
- Schedules: Schedules Identity, Schedules Name, Schedules Start Date, Schedules End Date, Schedules Fee
- Contact: Contact Identity, Contact Name, Contact Information, Contact Phone, Contact Email, Contact Time
- Regarding: Regarding Identity, Regarding Status, Regarding Date, Student Roll No, Schedules Identity
- Role: Role Identity, Role Name

- Output from the System:
 - List of Student
 - List of Staff
 - List of Branch
 - List of Subject
 - List of Notification
 - List of Semester
 - List of Curriculum
 - List of Examination
 - List of ExamMarks
 - List of Contact
 - List of Regarding
- Process Involved in the System:
 - Login
 - Logout
 - Change Password
 - Reset Password
 - Manage Student
 - Manage Staff
 - Manage Branch
 - Manage Subject
 - Manage Notification
 - Manage Curriculum
 - Manage Examination
 - Manage Exam_Mark
 - Manage Schedules
 - Manage Contact
 - Manage About
 - Manage Regarding
 - Caculate total marks
 - Caculate percentage
 - Judge the student
- **Expected Delivery Dates: 15/03/2015**
- **List of Deliverables:**
 - Document:
 - User manual
 - Installation guide
 - Document project
 - Source code
 - Database
- **Hardware Requirements:**
 - Client:
 - 1024 Megabytes of RAM or higher
 - A minimum computer system that will help you access browsers
 - Server:

- A minimum computer system that will help you access all the tools in the courses is a Pentium 200 or a higher capability CPU
- 2048 Megabytes of RAM or higher
- HDD: 20GB or higher
- Java Virtual Machine

- **Software Requirements:**

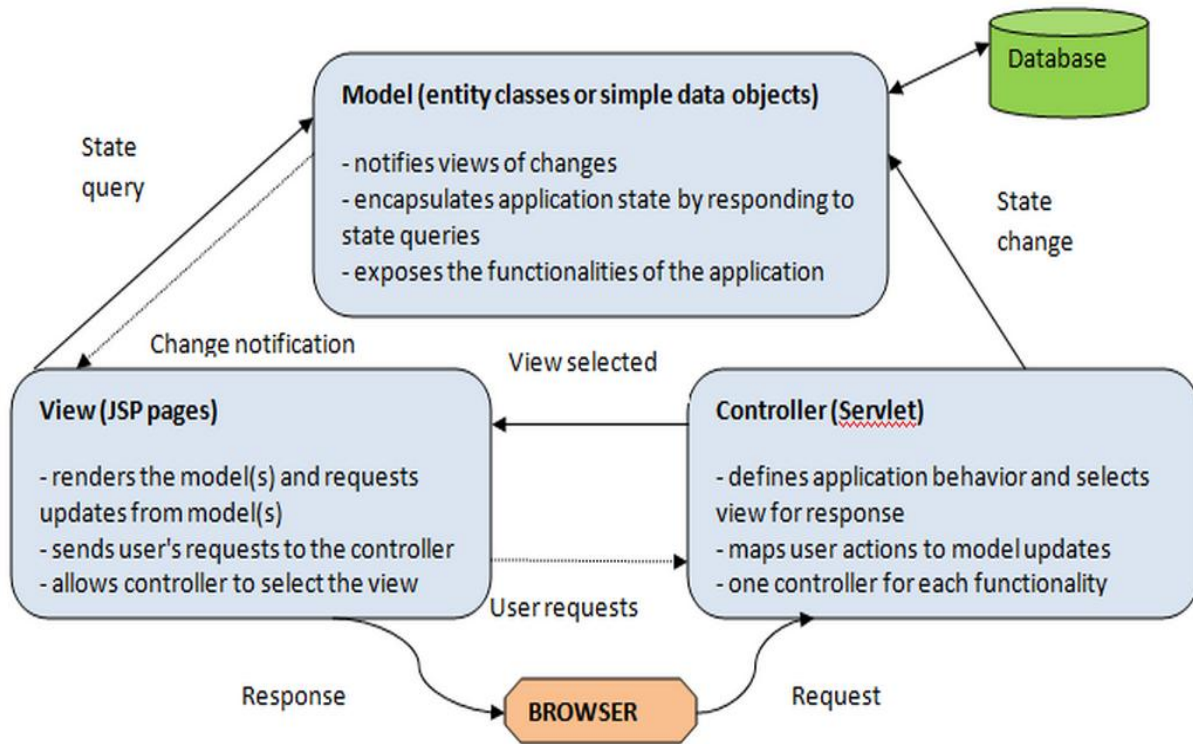
- Client:
 - Windows XP or later, MacOS, Linux
 - Browsers (IE, Chrome, Firefox, Opera, ...)
- Server:
 - Windows server 2003 or later
 - SQL server 2008 R2 or later
 - JSP/ Servlets with JSTL
 - JDBC Driver
 - Glassfish Server 4.0 or later
 - j2sdk1.7 or later / J2EE
 - Netbean 7.4 or later

- **Scope of work:**

- The database for holding the following details:
 - Users information (the Academic department people and the admin)
 - Student's information like curriculum, semester, marks, etc...
- Based on the user the application should judge who has logged in (Guest/Admin/Academic Department People) and display the site with the appropriate functionalities accordingly
- The total marks and the percentage will be calculated by the application itself, based on the marks secured by the student and the maximum and minimum marks as assigned to that particular subject in that particular curriculum and semester
- Based on the minimum marks of the subject the application should judge whether the student is failed or passed in that subject.
- The results are to be entered and displayed in the form of a table.

III. Requirement Analysis (Form No. 3):

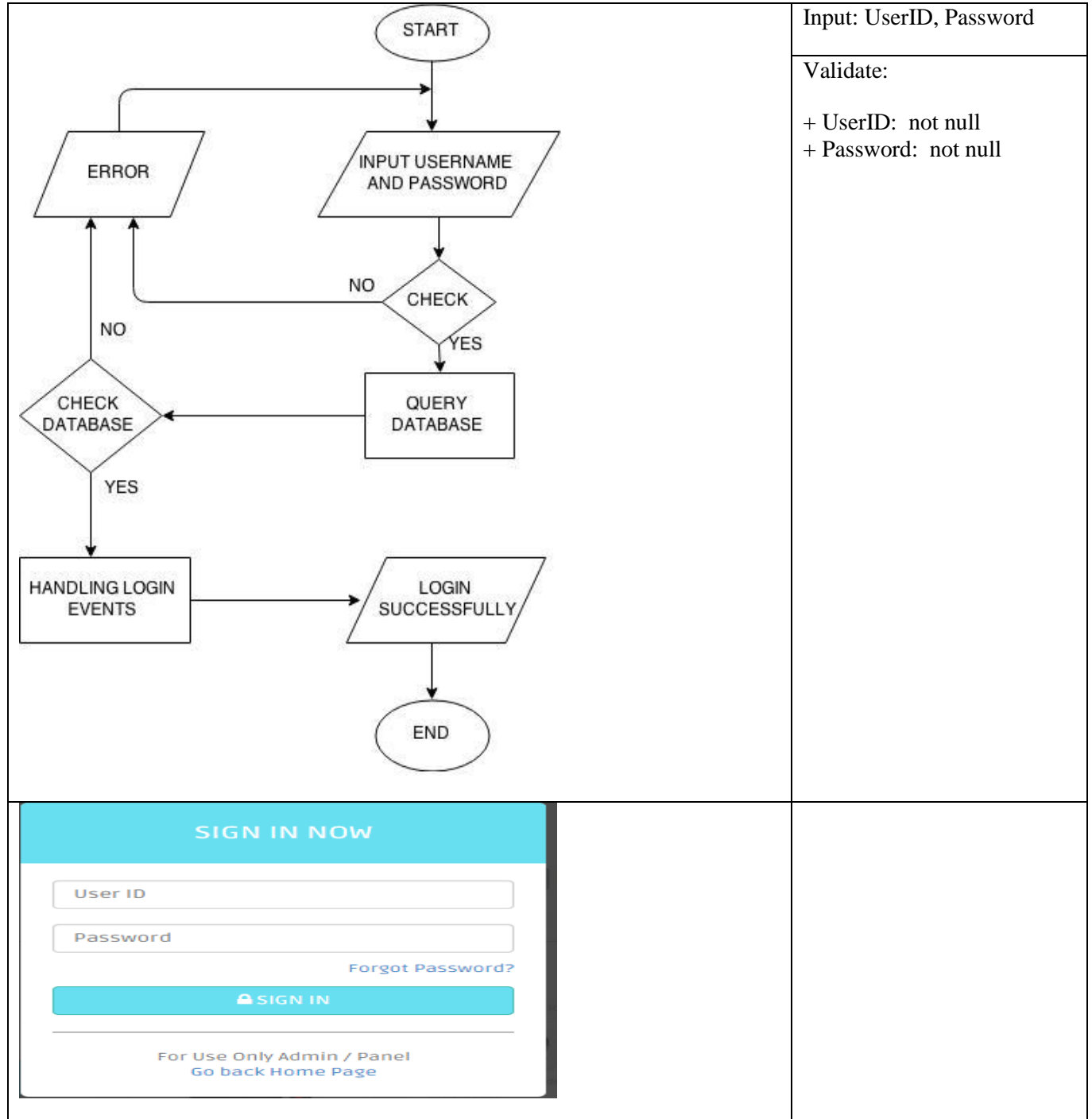
Since the application will be made of a Web-based distributed three-tier architecture to support multiple user transactions at the same time



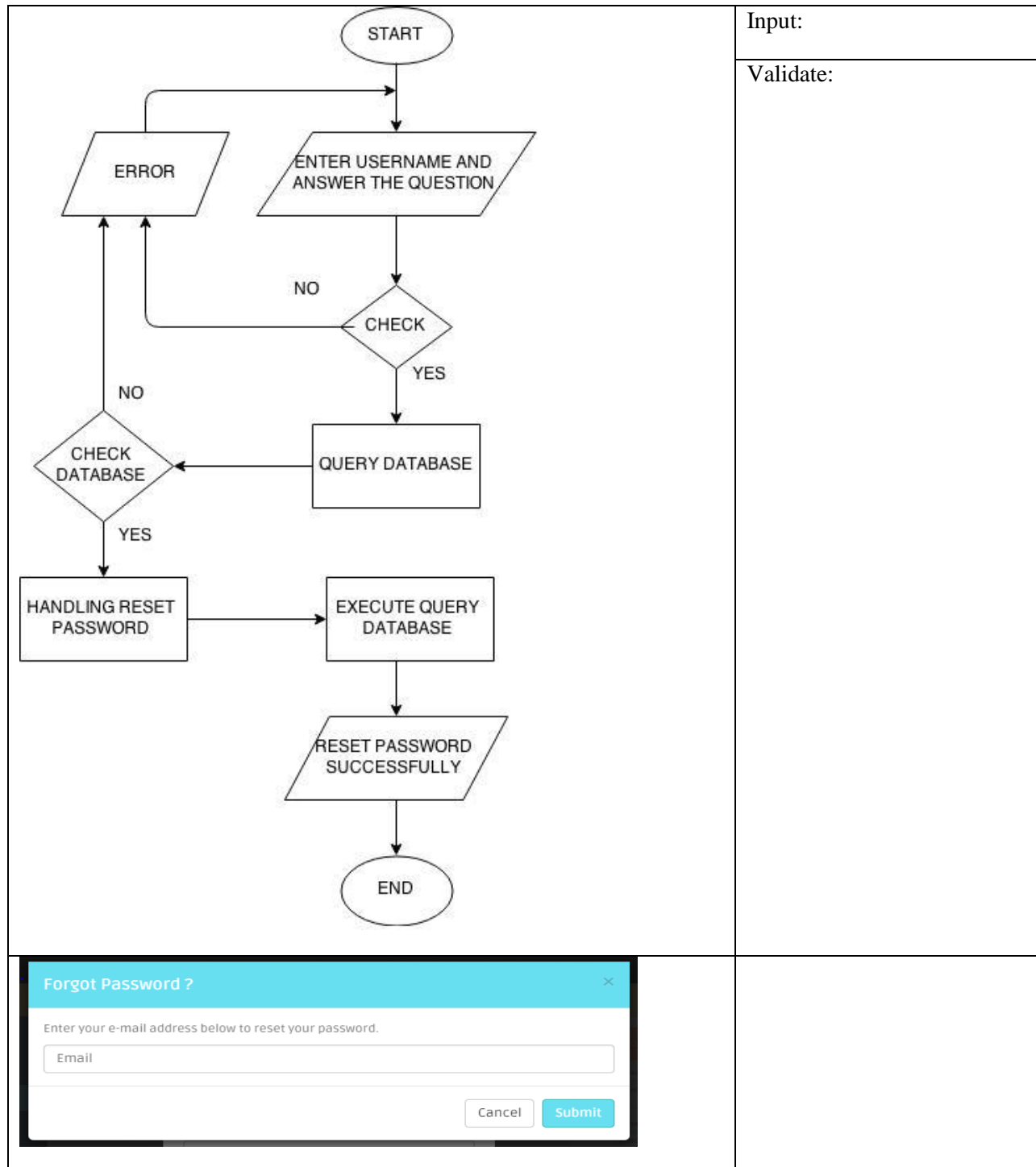
Web-Based Distributed 3-Tier Architecture of the Project

IV. Flow Chart (Form No.4):

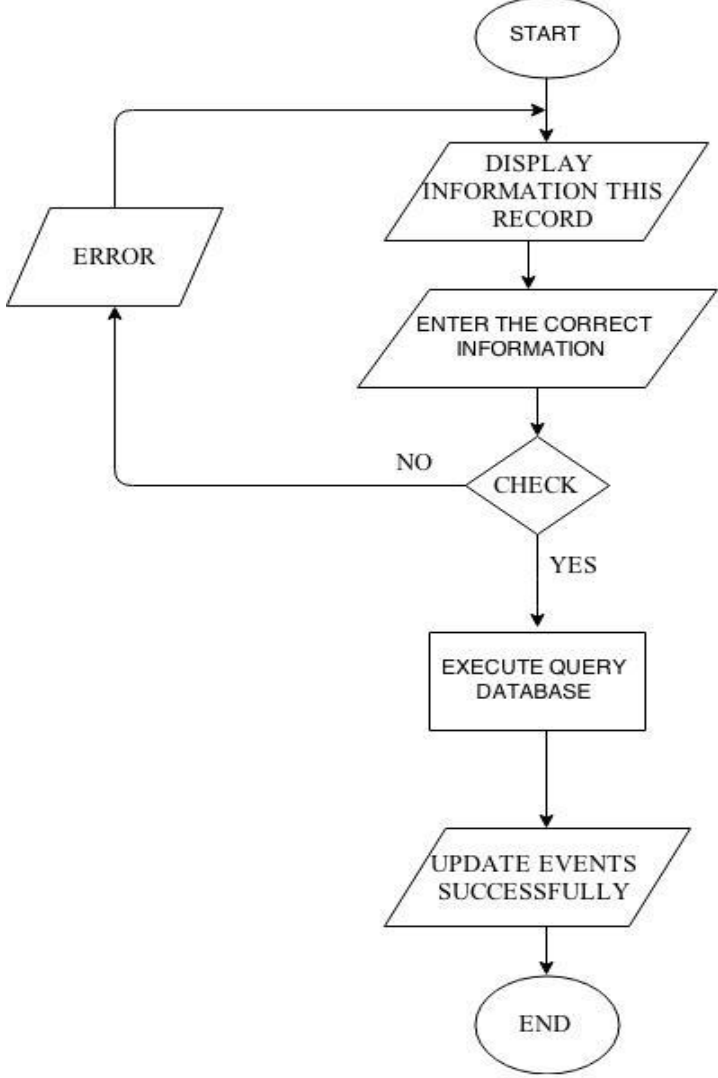
■ Login



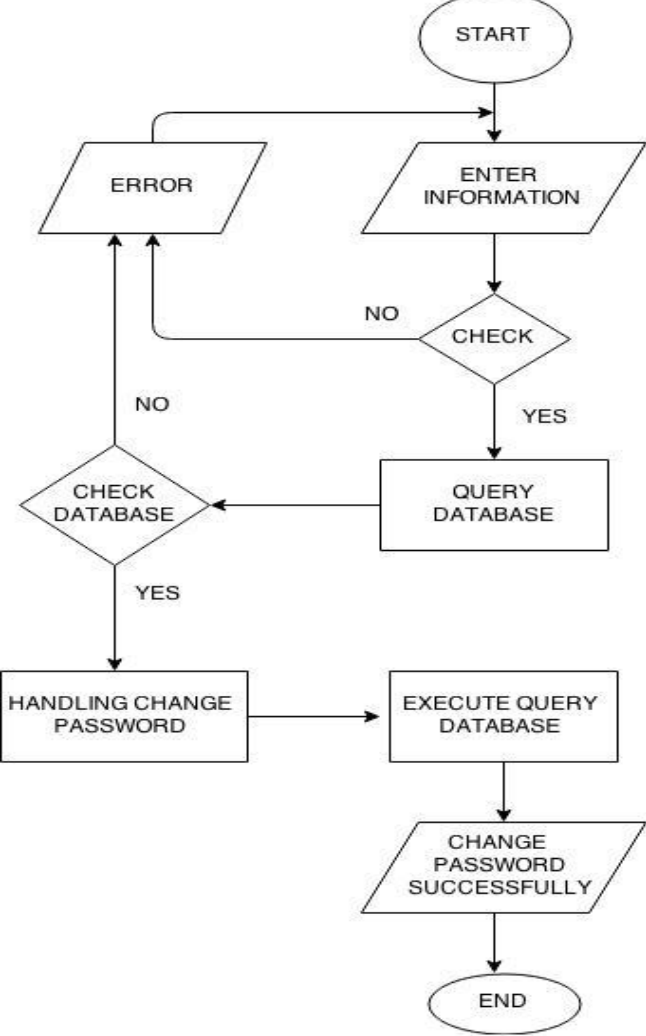
■ Reset password



■ Profile manage

 <pre> graph TD START([START]) --> DISPLAY[/DISPLAY INFORMATION THIS RECORD/] DISPLAY --> ENTER[/ENTER THE CORRECT INFORMATION/] ENTER --> CHECK{CHECK} CHECK -- NO --> ERROR[/ERROR/] ERROR --> ENTER CHECK -- YES --> EXECUTE[EXECUTE QUERY DATABASE] EXECUTE --> UPDATE[/UPDATE EVENTS SUCCESSFULLY/] UPDATE --> END([END]) </pre>	<p>Input:</p> <p>Validate:</p> <ul style="list-style-type: none"> +Full Name: not null +Address: not null +Email: not null +Phone number: not null +Birthday: not null
<p>> Profile Manage</p> <div data-bbox="212 1444 808 1843"> <div> <div>>Change password</div> <div>>Change secret question</div> </div> <div> <div>Staff Id</div> <div>Full Name</div> <div>Address</div> <div>Email</div> <div>Phone Number</div> <div>Birth Date</div> <div>Gender</div> <div>Start Date</div> </div> <div> <div>Staff Full Name</div> <div>Staff Address</div> <div>Staff Email</div> <div>Staff Phone Number</div> <div>Staff Birth Date</div> <div> <input checked="" type="radio"/> Male <input type="radio"/> Female </div> <div></div> </div> <div> <div>Update</div> <div>Clear All</div> </div> </div>	

■ Change password

 <pre> graph TD START([START]) --> ENTER[ENTER INFORMATION] ENTER --> CHECK{CHECK} CHECK -- NO --> ERROR[/ERROR/] CHECK -- YES --> QUERY[QUERY DATABASE] QUERY --> CHECK_DB{CHECK DATABASE} CHECK_DB -- NO --> ERROR CHECK_DB -- YES --> HANDLING[HANDLING CHANGE PASSWORD] HANDLING --> EXECUTE[EXECUTE QUERY DATABASE] EXECUTE --> SUCCESS[/CHANGE PASSWORD SUCCESSFULLY/] SUCCESS --> END([END]) </pre>	<p>Input:</p> <p>Validate:</p> <ul style="list-style-type: none"> + Staff Password: not null +New password: not null +Confirm password: not null
<p>> Profile Manage</p> <div> <div>>Change information</div> <div>>Change secret question</div> </div> <div> <div>Staff Password</div> <div>Staff Password</div> </div> <div> <div>New Password</div> <div>Enter new password</div> </div> <div> <div>Confirm Password</div> <div>Enter confirm password</div> </div> <div> <div>Change Password</div> <div>Clear All</div> </div>	

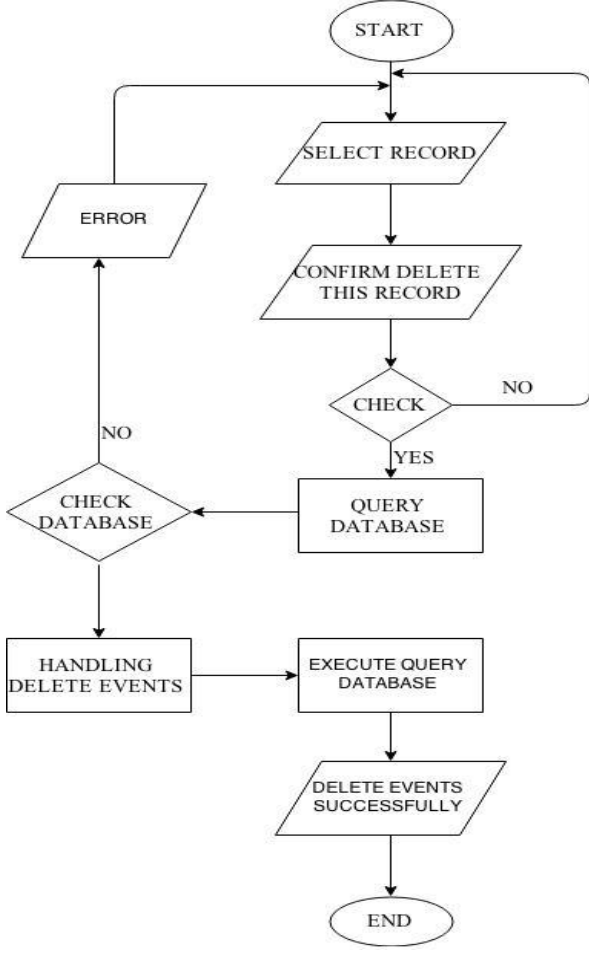
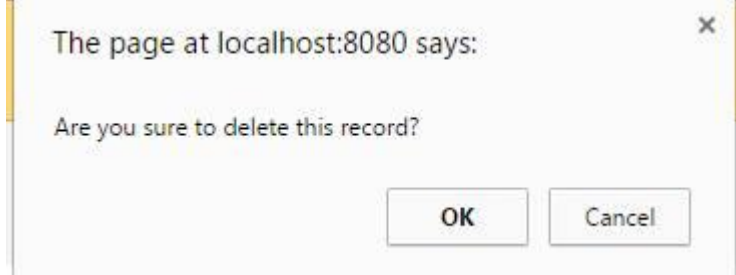
■ Add

<pre> graph TD START([START]) --> INPUT[/INPUT INFORMATION/] INPUT --> CHECK{CHECK} CHECK -- NO --> ERORR[/ERORR/] ERORR --> INPUT CHECK -- YES --> QUERY[QUERY DATABASE] QUERY --> CHECK_DB{CHECK DATABASE} CHECK_DB -- NO --> ERORR CHECK_DB -- YES --> SAVE[SAVE TO DATABASE] SAVE --> HANDLE[HANDLE ADD EVENTS] HANDLE --> EXEC[EXECUTE QUERY DATABASE] EXEC --> SUCCESS[ADD EVENTS SUCCESSFULLY] SUCCESS --> END([END]) </pre>	<p>Input:</p> <p>Validate:</p>
<p>> Staff Manage</p>	<p>* This diagram used for add: contact, branch, curriculum, regarding, semester, schedules, subjects, notification by admin; student, examination, mark by academic's people</p> <ul style="list-style-type: none"> + Actor: Staff + Maininterface: display information of data, with button add new on form + Interface: Add interface, with some textbox staff enter information of what they are manage + Handling: check information, when staff enter information, system will check that information to know they can add new or not + Database: table

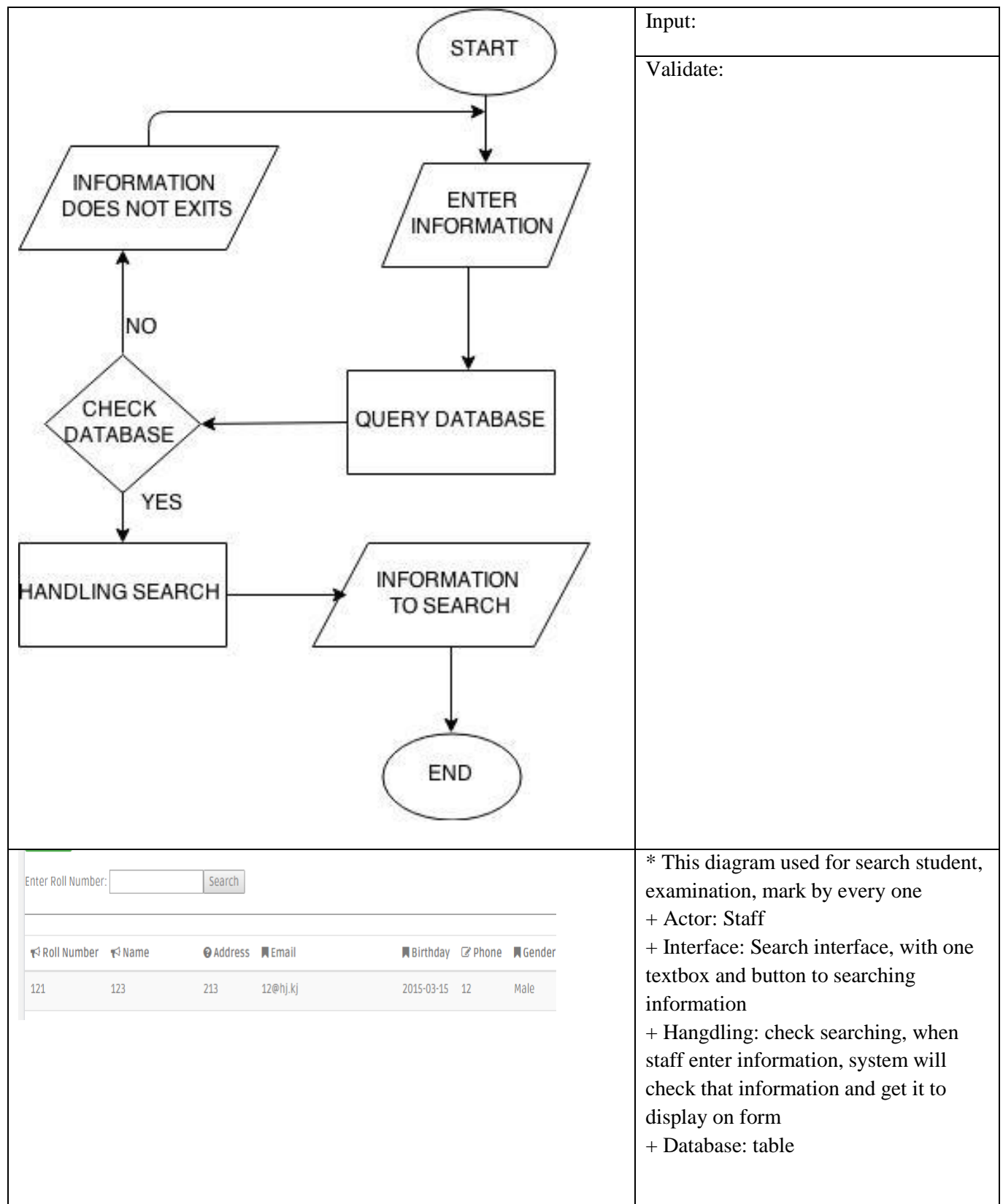
■ Edit

<pre> graph TD START([START]) --> SELECT[SELECT RECORD] SELECT --> DISPLAY[/DISPLAY INFORMATION THIS RECORD/] DISPLAY --> ENTER[/ENTER THE CORRECT INFORMATION/] ENTER --> CHECK{CHECK} CHECK -- NO --> ERROR[/ERROR/] ERROR --> ENTER CHECK -- YES --> HANDLING[HANDLING UPDATE EVENTS] HANDLING --> EXECUTE[EXECUTE QUERY DATABASE] EXECUTE --> SUCCESS[/UPDATE EVENTS SUCCESSFULLY/] SUCCESS --> END([END]) </pre>	<p>Input:</p> <p>Validate:</p>
<p>> Staff Manage</p>	<p>* This diagram used for update: contact, branch, curriculum, regarding, semester, schedules, subjects, notification, about by admin; student, examination, mark by academic's people</p> <ul style="list-style-type: none"> + Actor: Staff + Main interface: display information of data, with button update in single record + Interface: Update interface, with some textbox staff enter information of what they are manage + Handling: check information, when staff enter information, system will check that information to know they can update or not + Database: table

■ Delete

 <pre> graph TD START([START]) --> SELECT[/SELECT RECORD/] SELECT --> CONFIRM[/CONFIRM DELETE THIS RECORD/] CONFIRM --> CHECK{CHECK} CHECK -- NO --> SELECT CHECK -- YES --> QUERY[QUERY DATABASE] QUERY --> CHECK_DB{CHECK DATABASE} CHECK_DB -- NO --> ERROR[/ERROR/] ERROR --> SELECT CHECK_DB -- YES --> HANDLING[HANDLING DELETE EVENTS] HANDLING --> EXECUTE[EXECUTE QUERY DATABASE] EXECUTE --> SUCCESS[/DELETE EVENTS SUCCESSFULLY/] SUCCESS --> END([END]) </pre>	<p>Input:</p> <p>Validate:</p>
	<p>* This diagram used for delete: contact, branch, curriculum, regarding, semester, schedules, subjects, notification by admin; student, examination, mark by academic's people</p> <p>+ Actor: Staff</p> <p>+ Interface: display information of data, with button delete in single record</p> <p>+ Handling: check information, when staff accept delete, system will check that information to know they can delete or not</p> <p>+ Database: table</p>

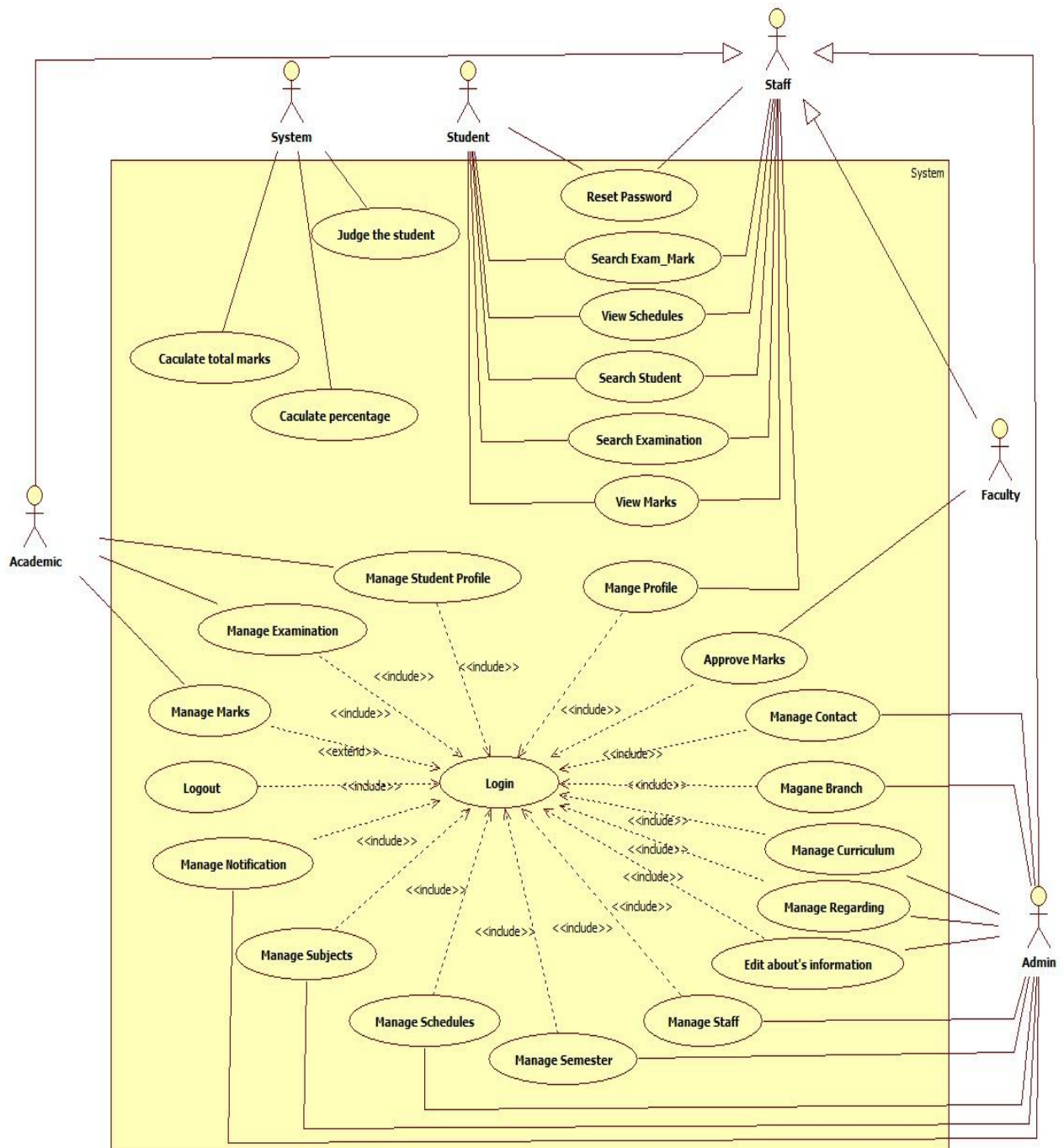
■ Search



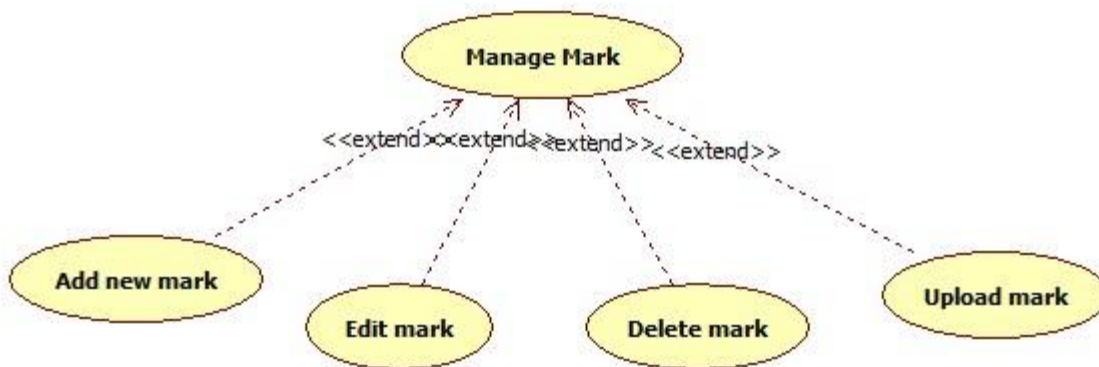
VI. Diagram (Form No.6)

VI.1 Use case:

+ Level 0



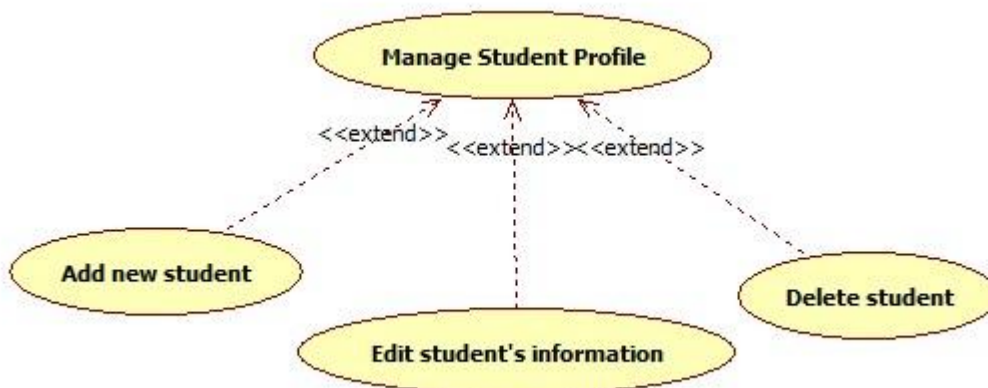
+ Level 1
. Manage Mark



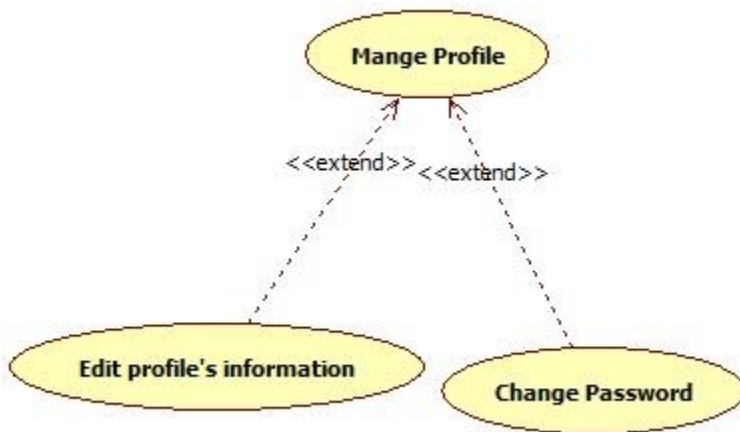
. Manage Examination



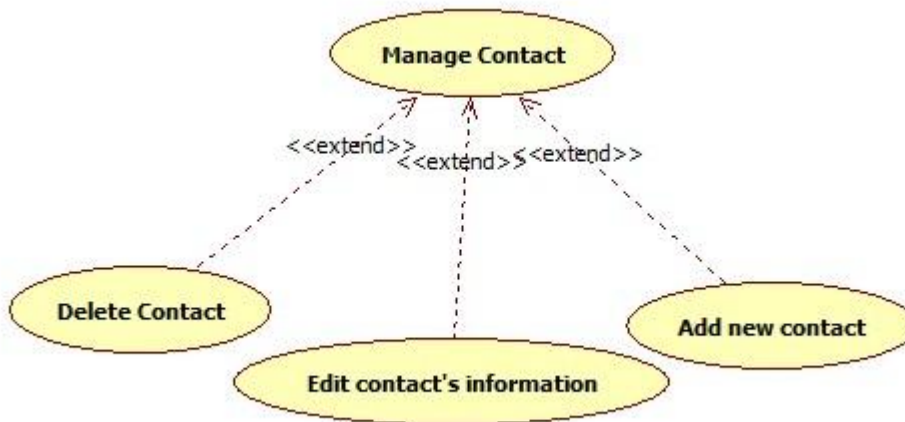
. Manage Student



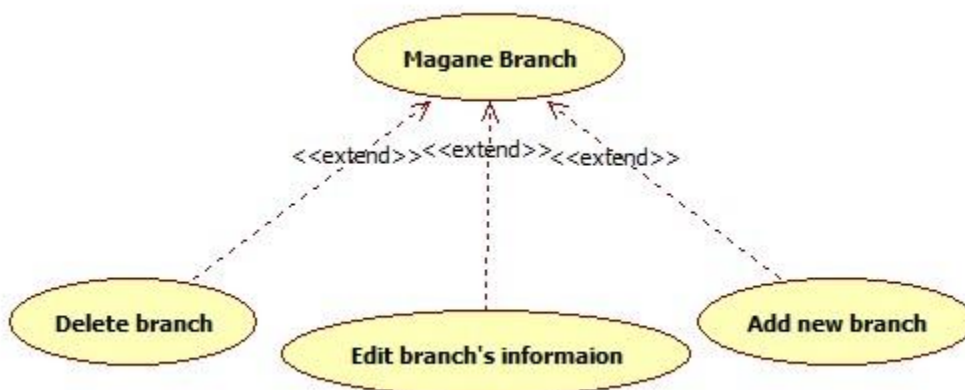
. Manage Profile



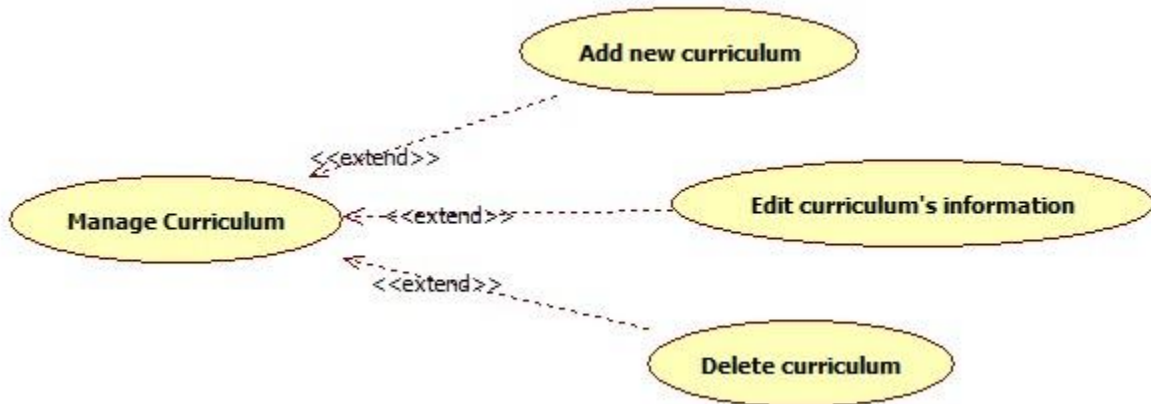
. Manage Contact



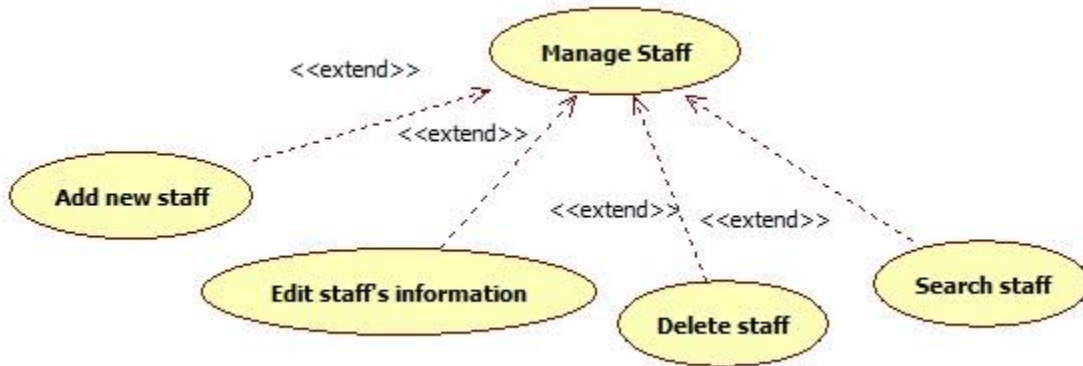
. Manage Branch



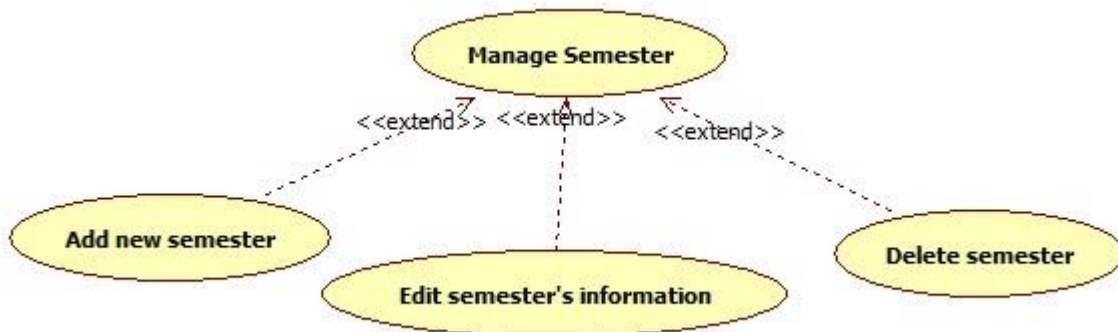
. Manage Curriculum



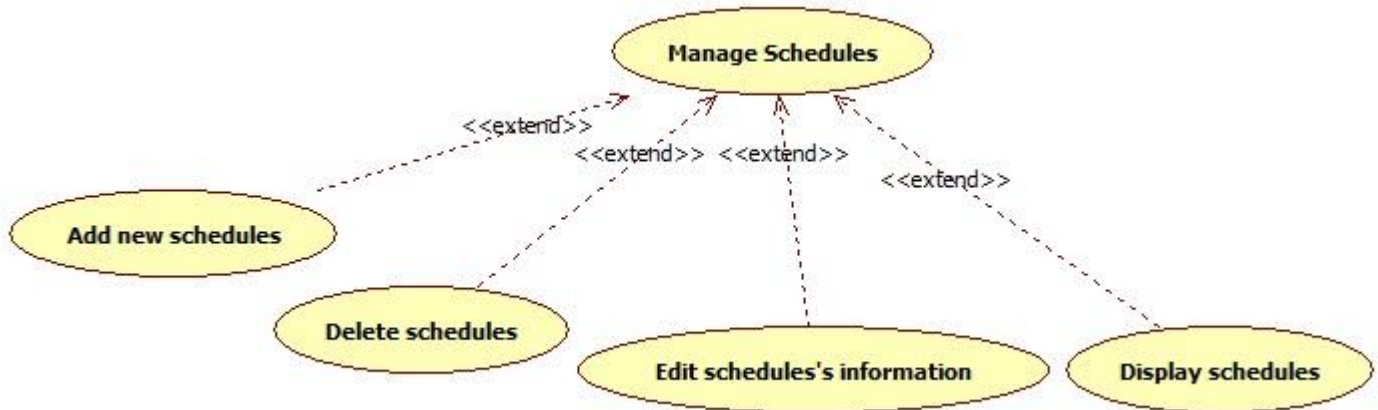
. Manage Staff



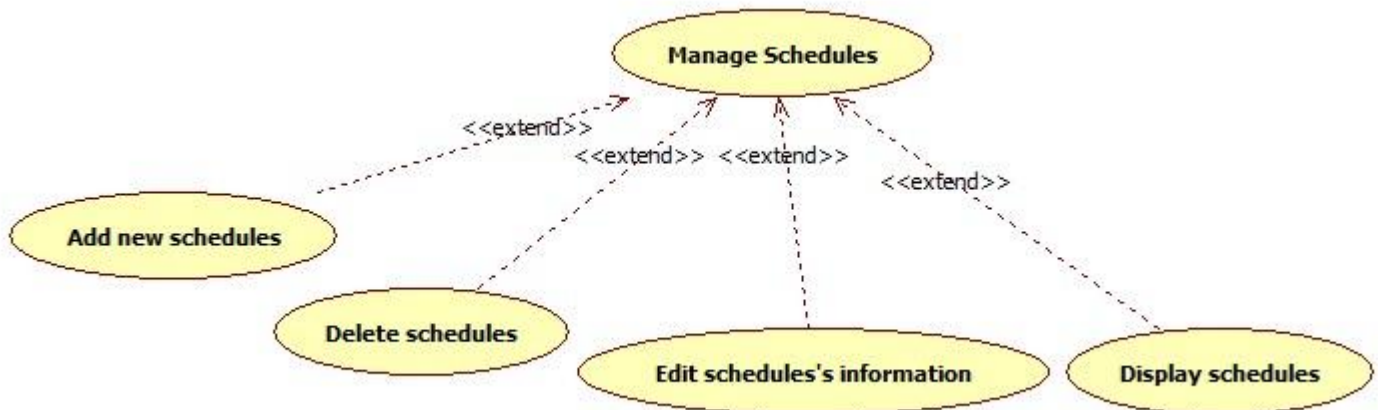
. Manage Semester



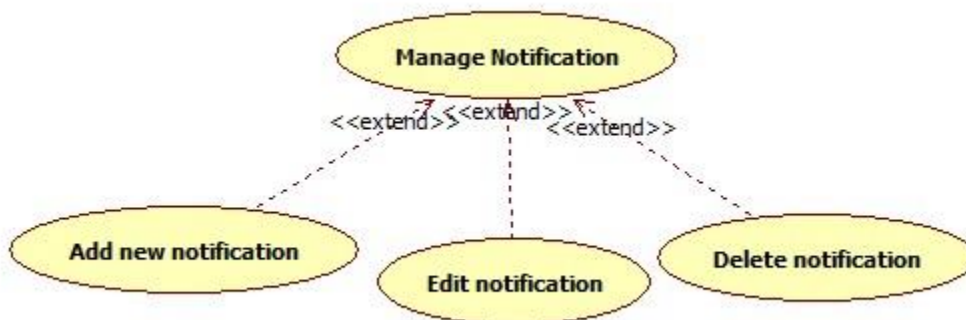
. Manage Schedules



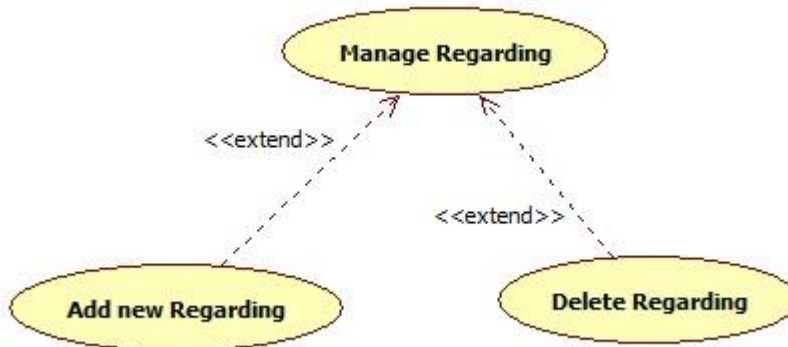
. Manage Subjects



. Manage Notification



. Manage regarding

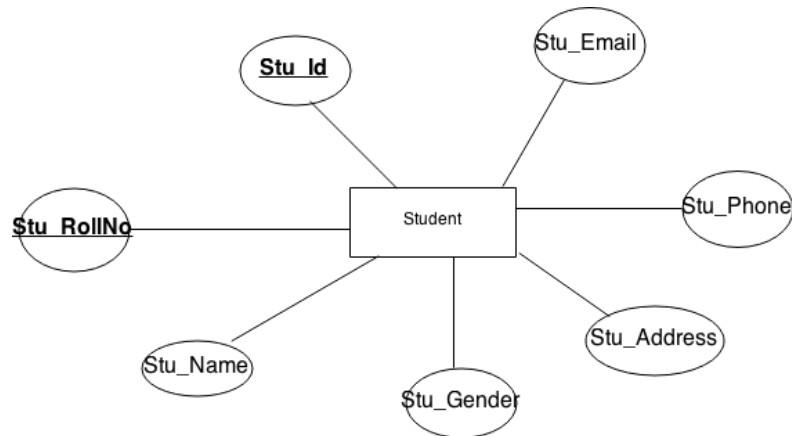


Result Declaration Board

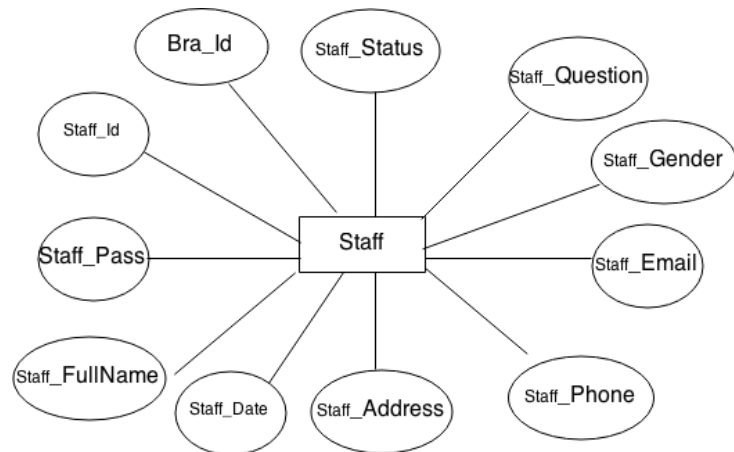


+Level 1

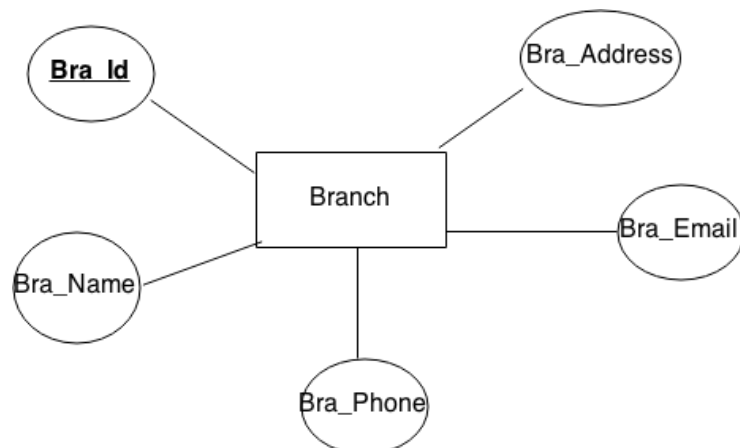
. Student



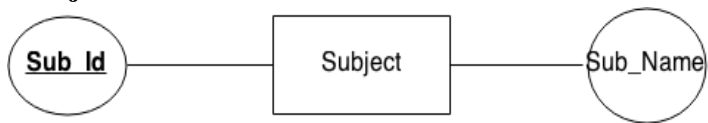
. Staff



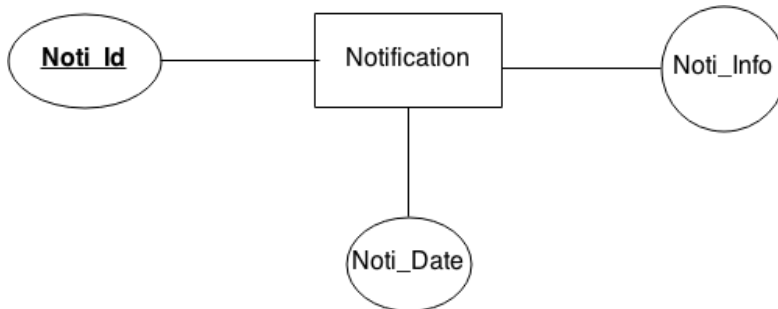
. Branch



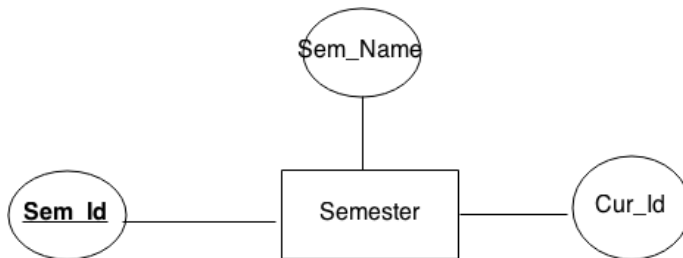
. Subjects



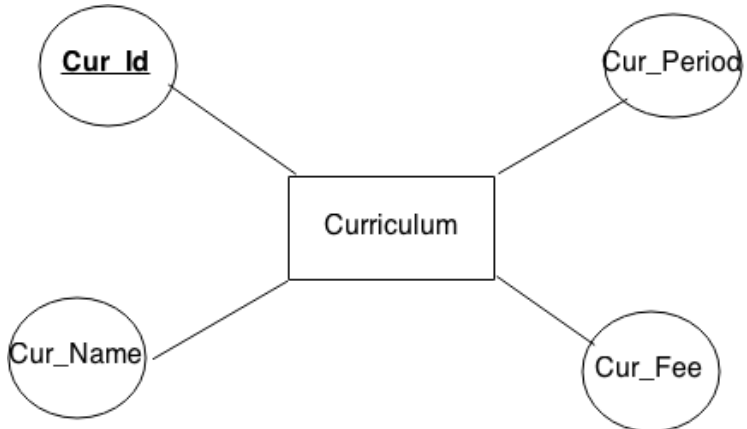
. Notification



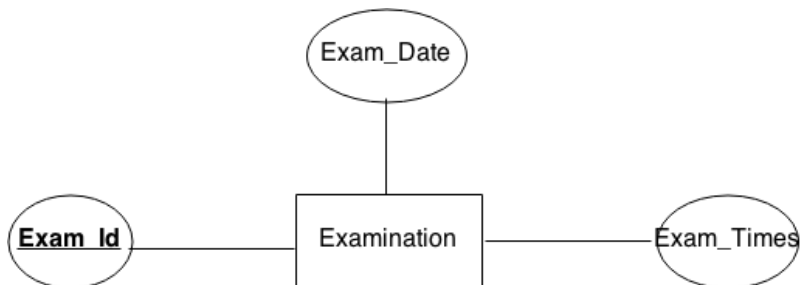
.Semester



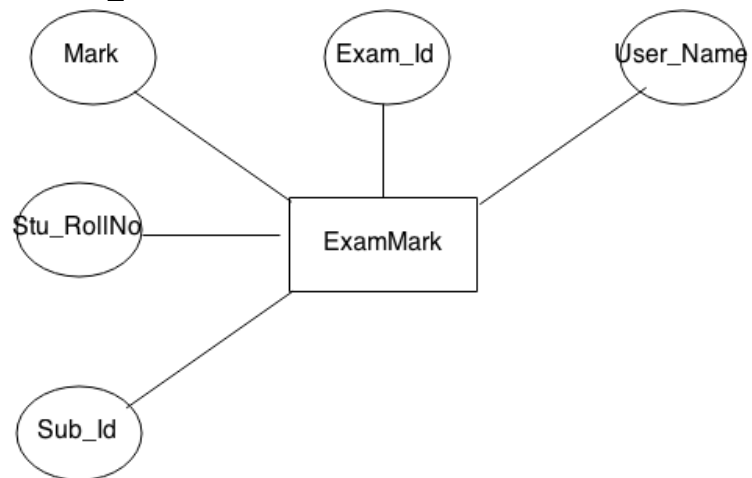
. Curriculum



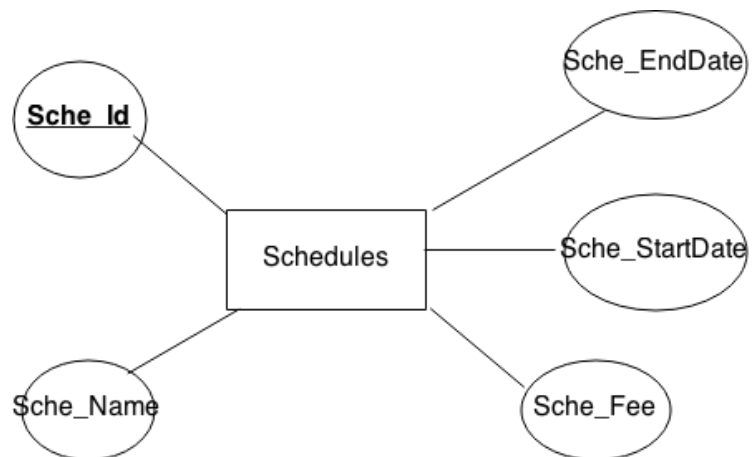
. Examination



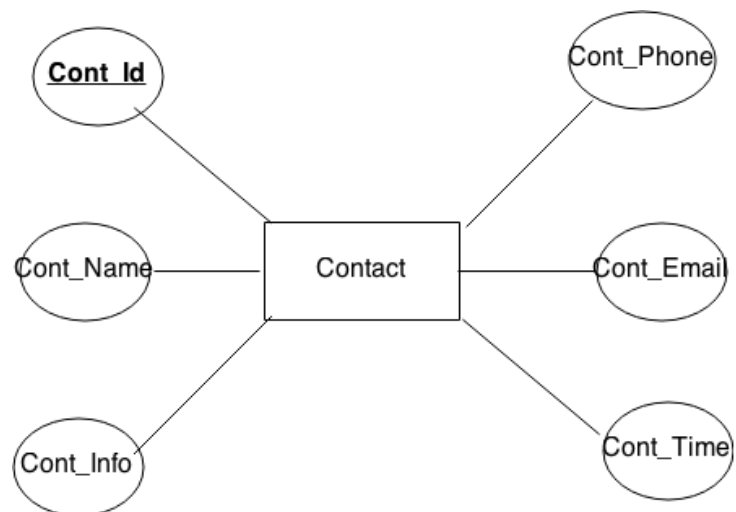
. Exam_Mark



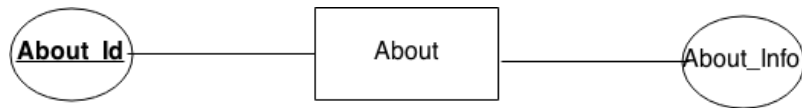
. Schedules



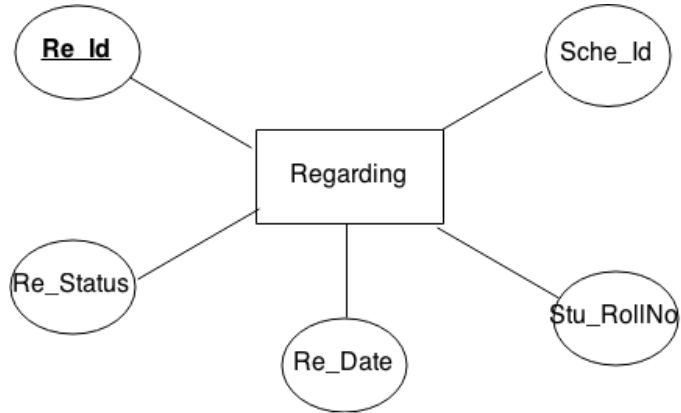
. Contact



. About



. Regarding

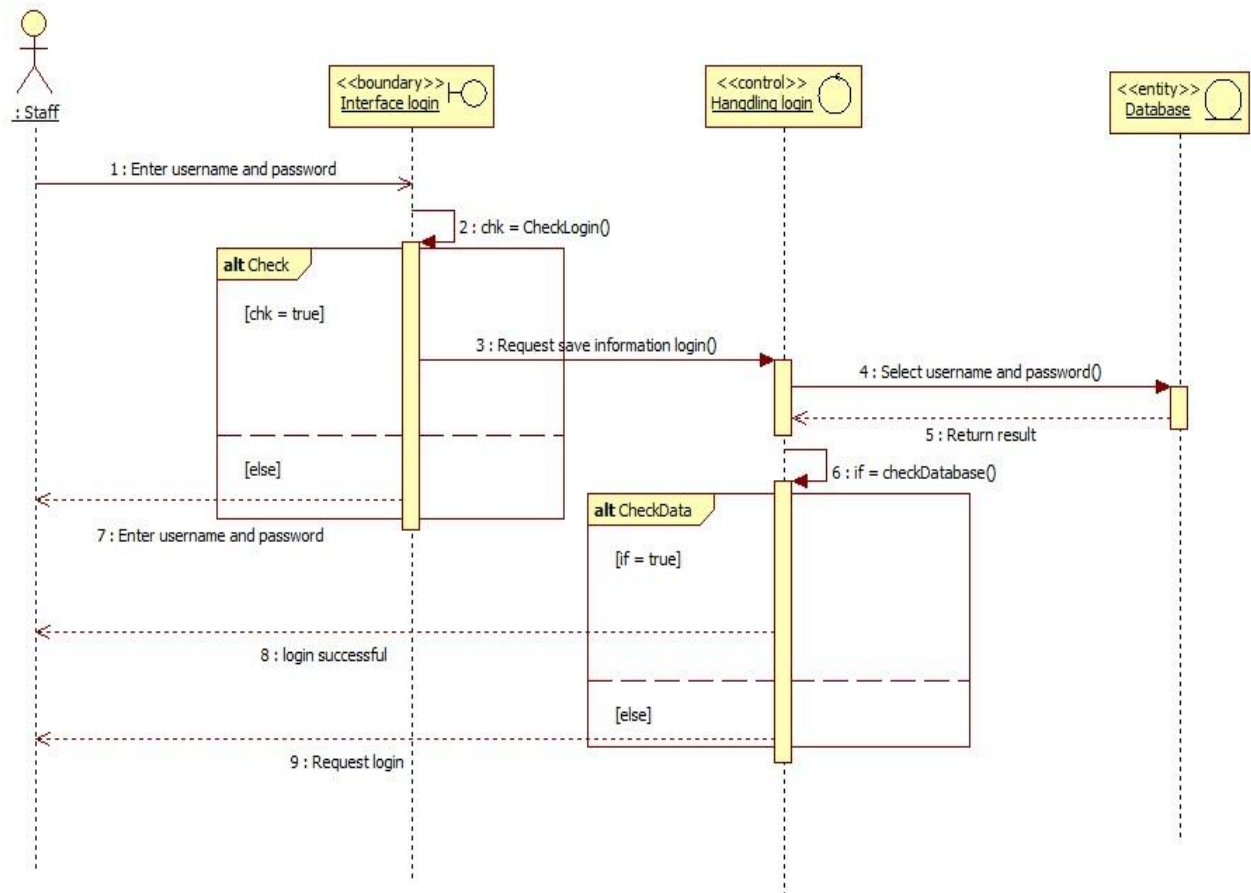


. Role



VI.2 Sequence Diagram

■ Login



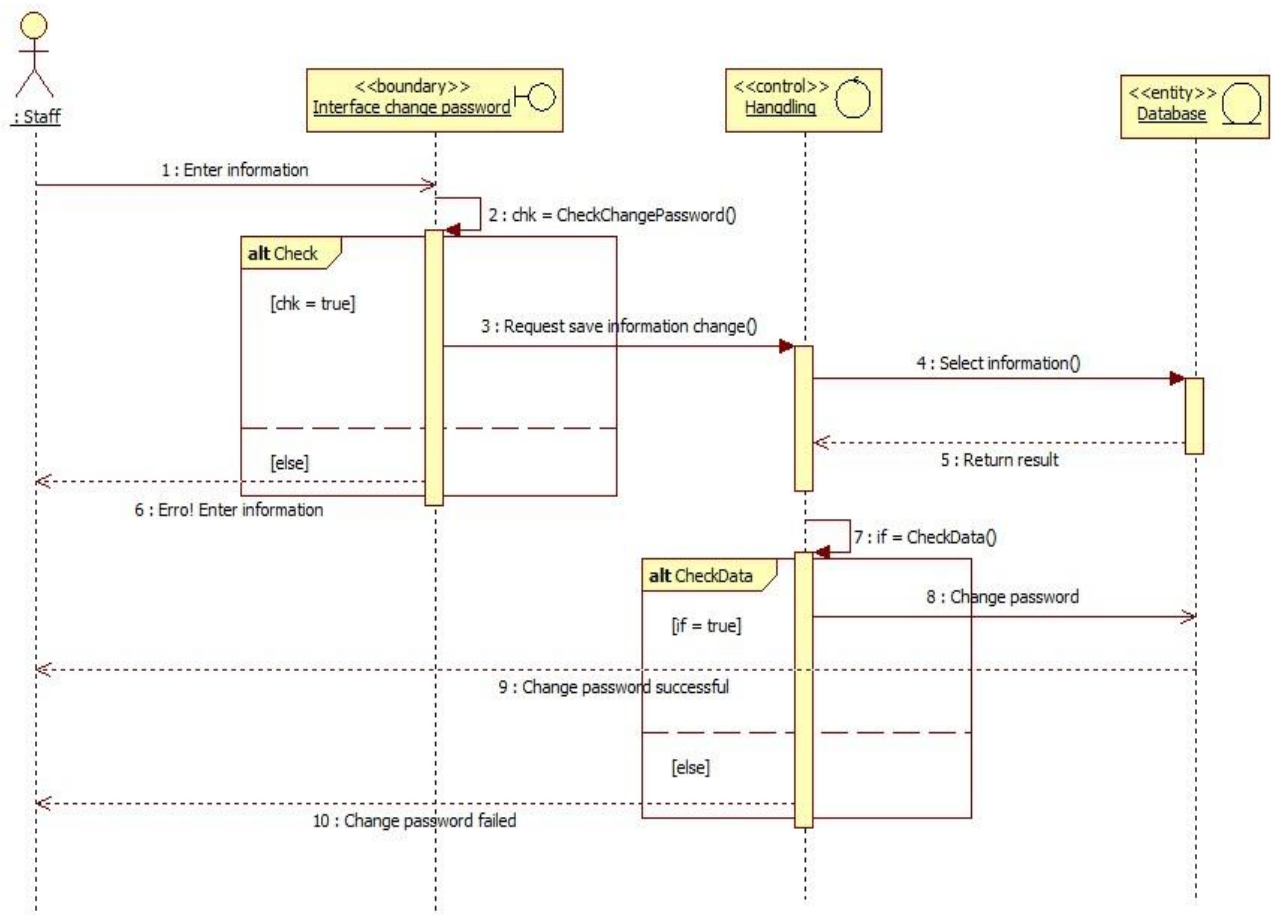
+ Actor: Staff

+ Interface: Login interface, with two textbox for staff enter their username and password to login

+ Handling: check login, when staff enter their username and password, system will compare their information to know they can enter system or not

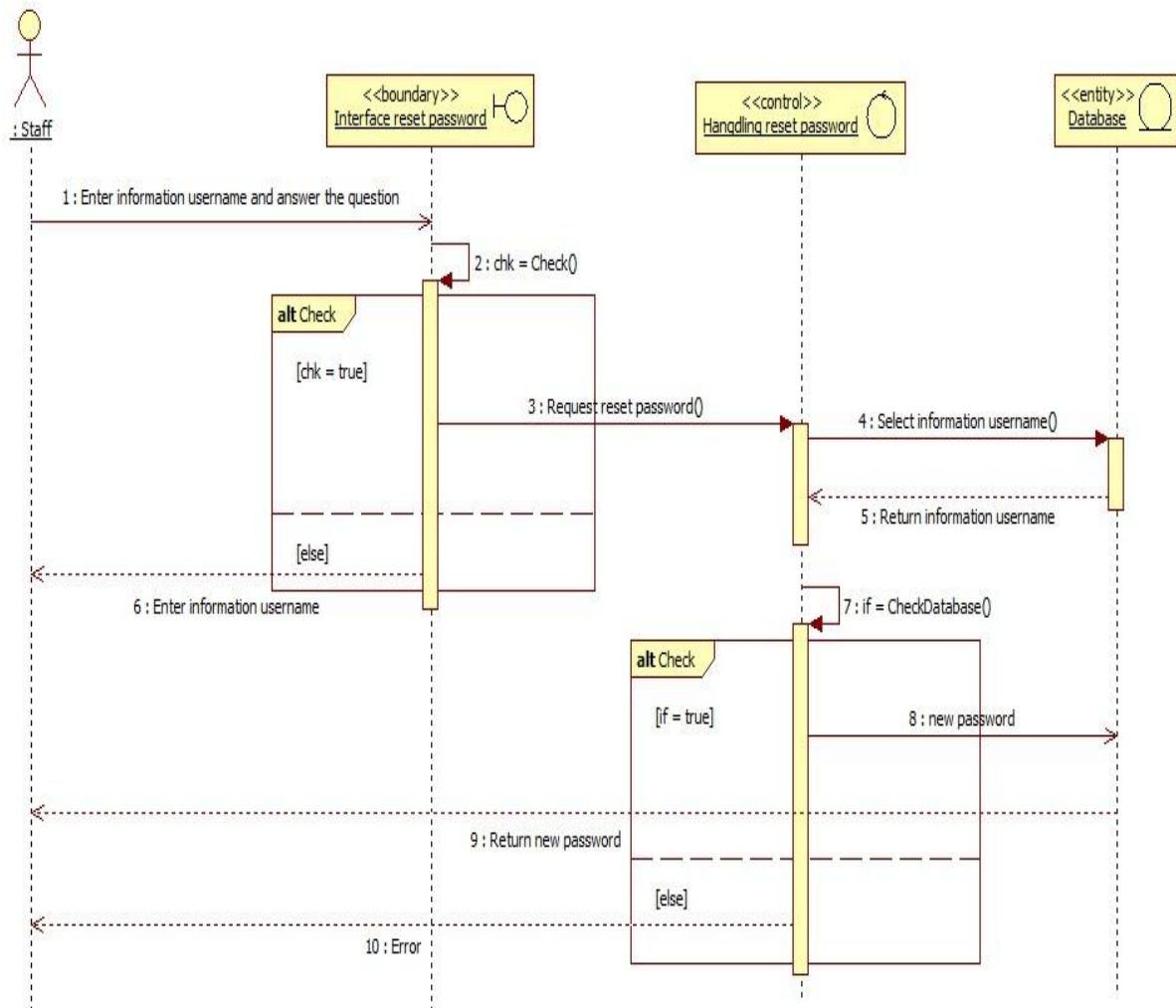
+ Database: staff table to get username and password

■ Change password



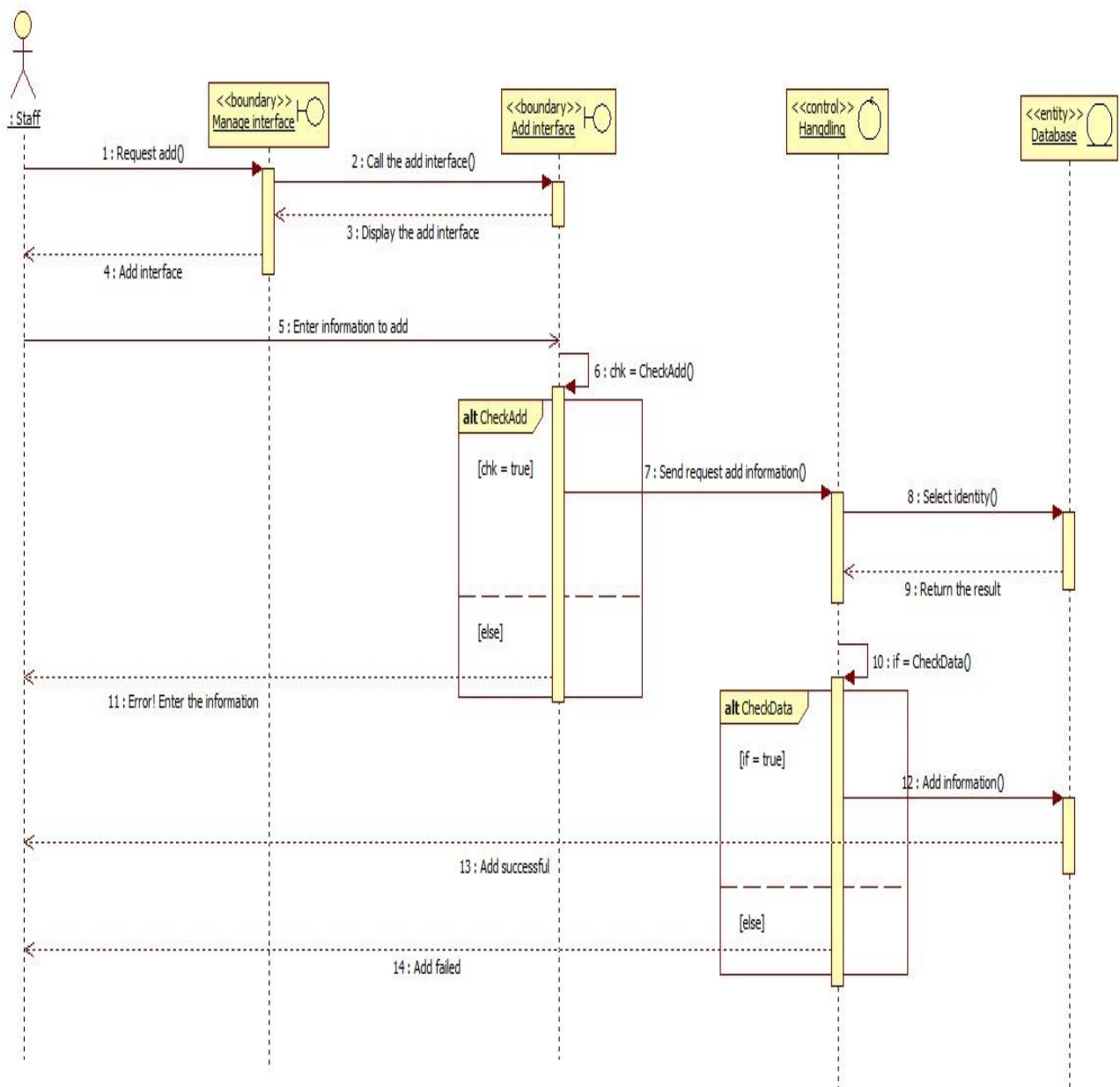
- + Actor: Staff
- + Interface: Change password interface, with three textbox for staff enter their old password, new password and confirm new password to change password
- + Handling: check change password, when staff enter their password, system will compare their information to know they can change password or not
- + Database: staff table to get old password

■ Reset password



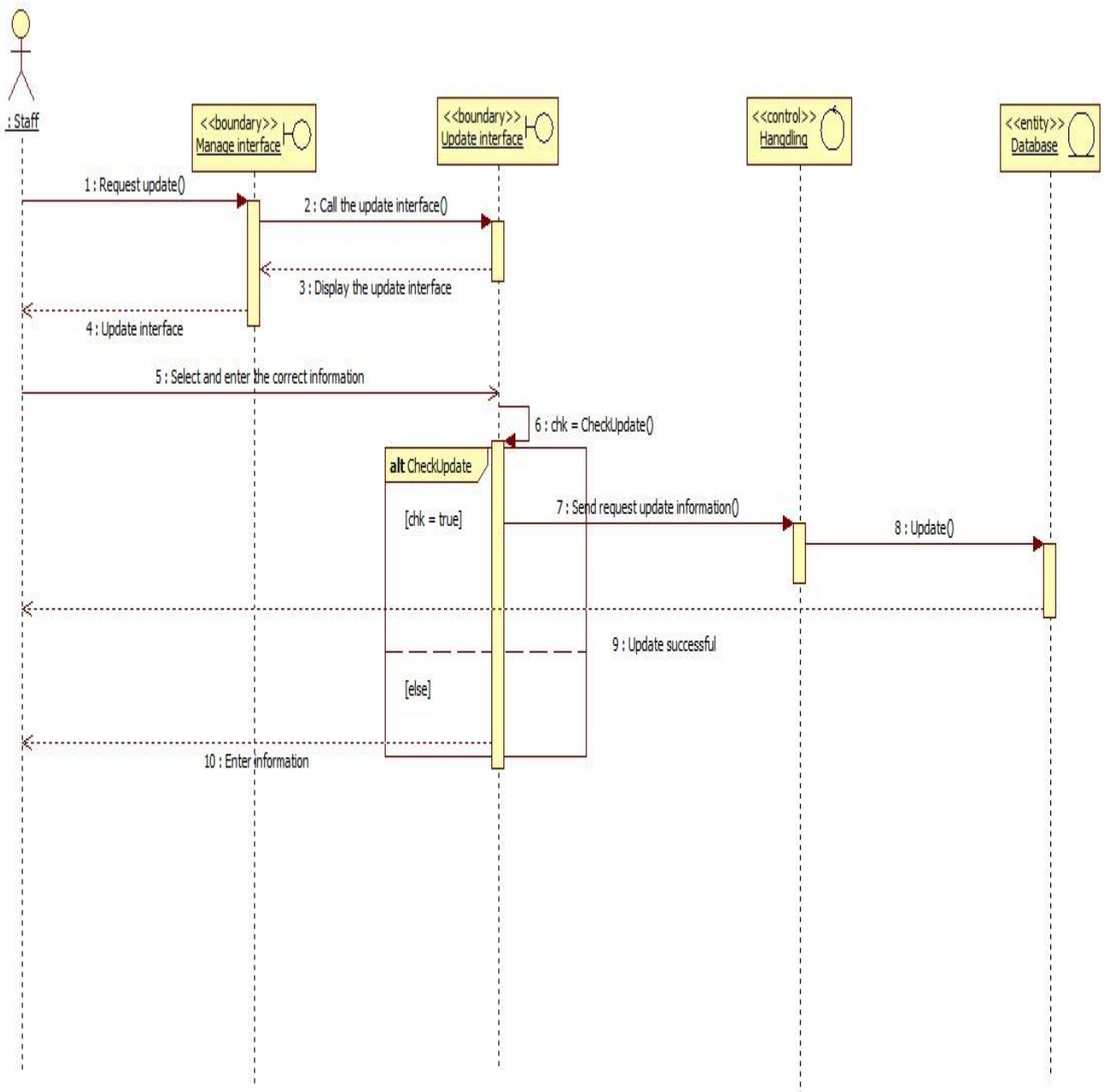
- + Actor: Staff
- + Interface: Reset password interface, with two textbox and one select list box for staff enter their username, selected secret question and enter answer to reset password
- + Handling: check reset password, when staff enter their username, answer and question, system will compare their information to know they can reset password or not
- + Database: staff table to get username, question and answer

■ Manage + Add



- * This diagram used for add: contact, branch, curriculum, regarding, semester, schedules, subjects, notification by admin; student, examination, mark by academic's people
- + Actor: Staff
- + Maininterface: display information of data, with button add new on form
- + Interface: Add interface, with some textbox staff enter information of what they are manage
- + Handling: check information, when staff enter information, system will check that information to know they can add new or not
- + Database: table

+ Update



* This diagram used for update: contact, branch, curriculum, regarding, semester, schedules, subjects, notification, about by admin; student, examination, mark by academic's people

+ Actor: Staff

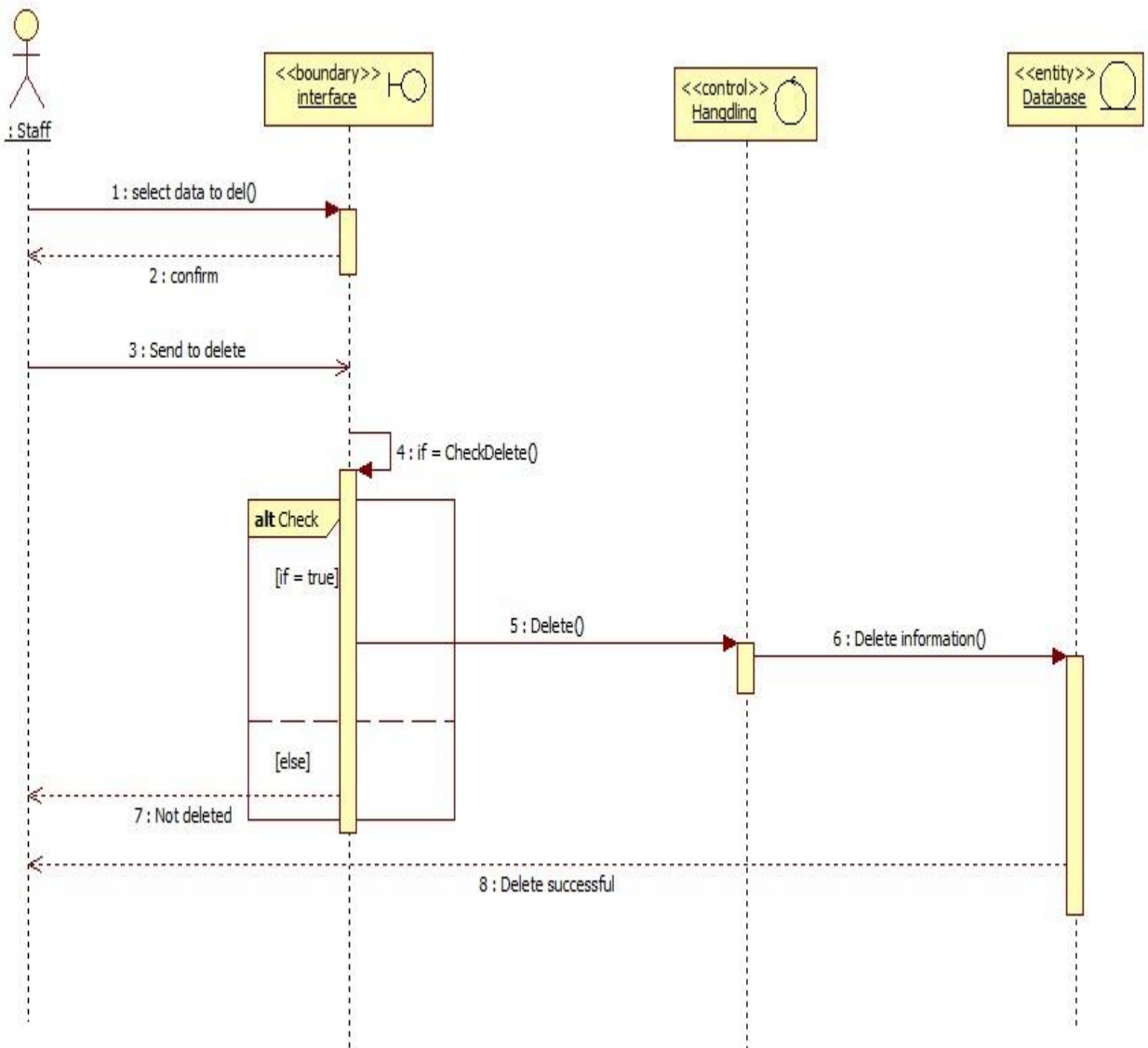
+ Main interface: display information of data, with button update in single record

+ Interface: Update interface, with some textbox staff enter information of what they are manage

+ Handling: check information, when staff enter information, system will check that information to know they can update or not

+ Database: table

+ Delete



* This diagram used for delete: contact, branch, curriculum, regarding, semester, schedules, subjects, notification by admin; student, examination, mark by academic's people

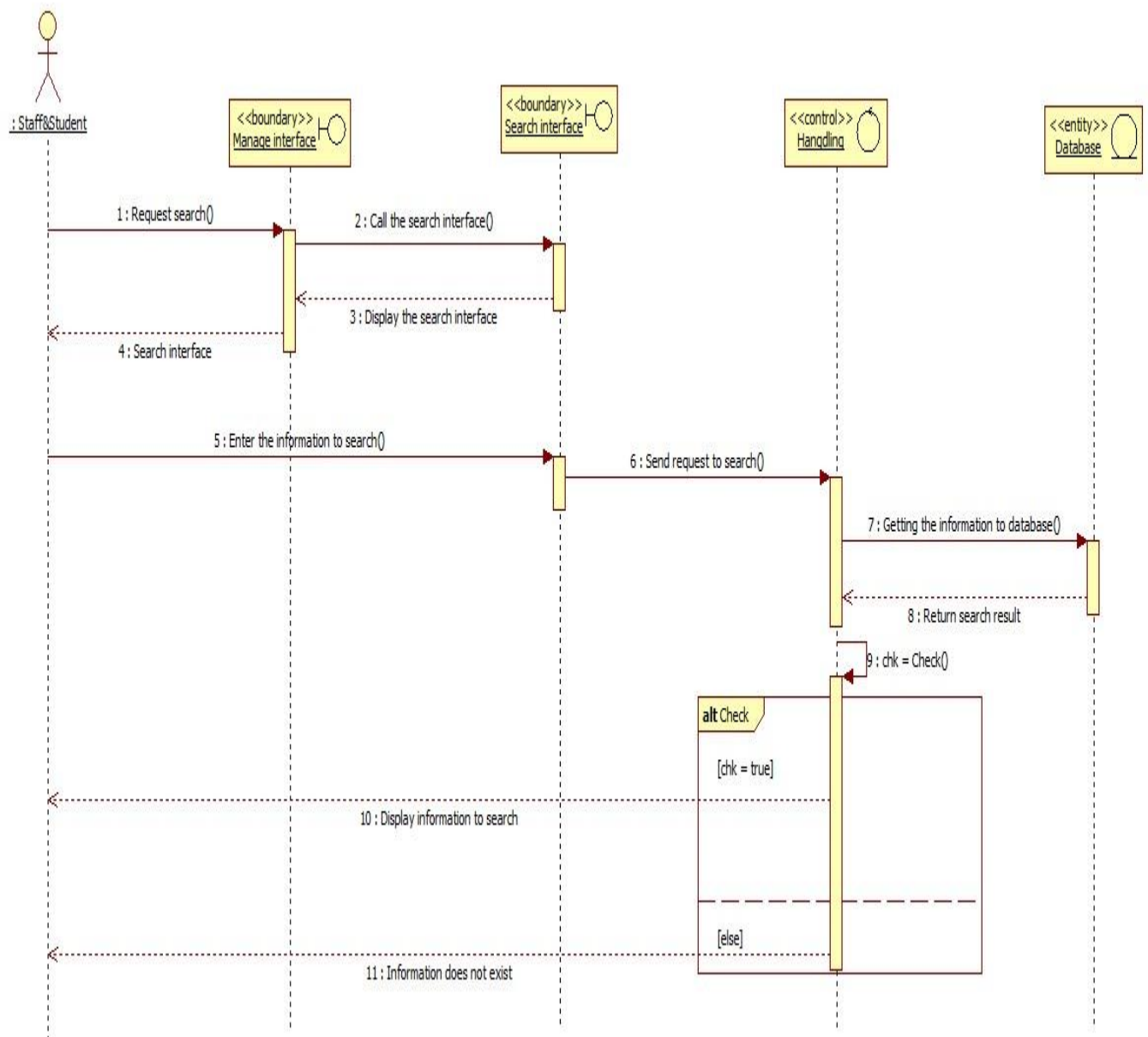
+ Actor: Staff

+ Interface: display information of data, with button delete in single record

+ Handling: check information, when staff accept delete, system will check that information to know they can delete or not

+ Database: table

■ Search



* This diagram used for search student, examination, mark by every one

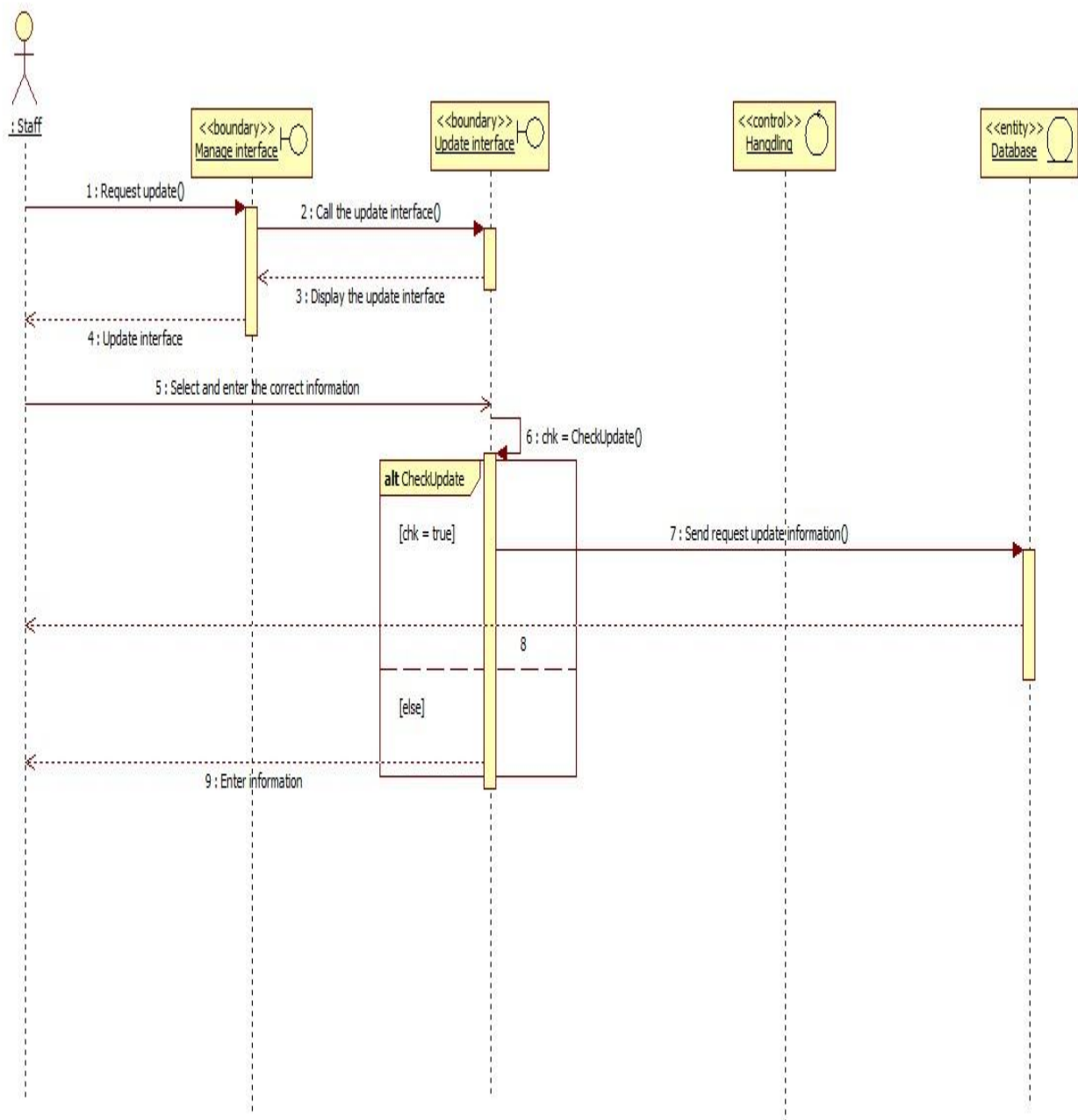
+ Actor: Staff and Student

+ Interface: Search interface, with one textbox and button to searching information

+ Handling: check searching, when staff enter information, system will check that information and get it to display on form

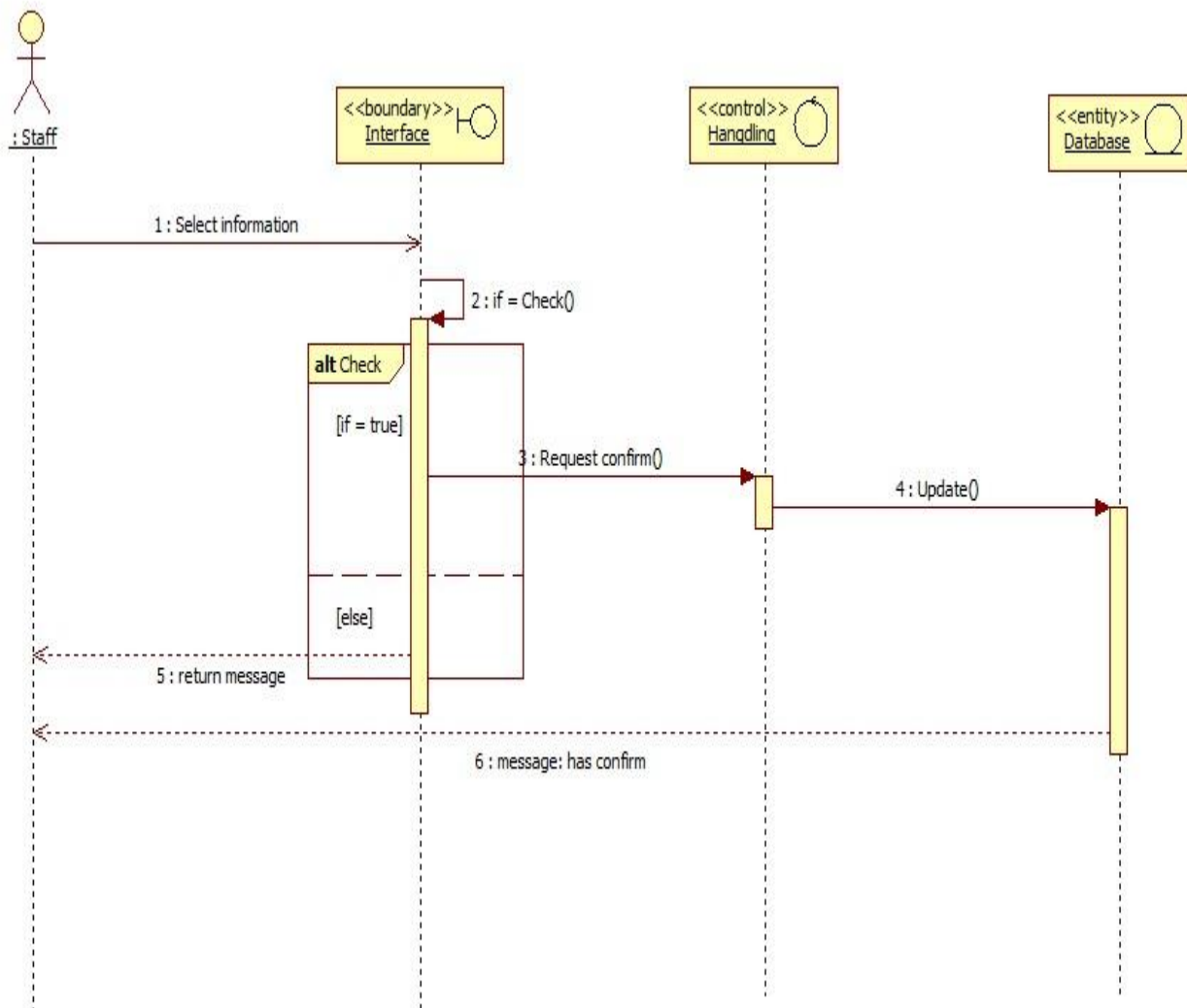
+ Database: table

■ Manage profile



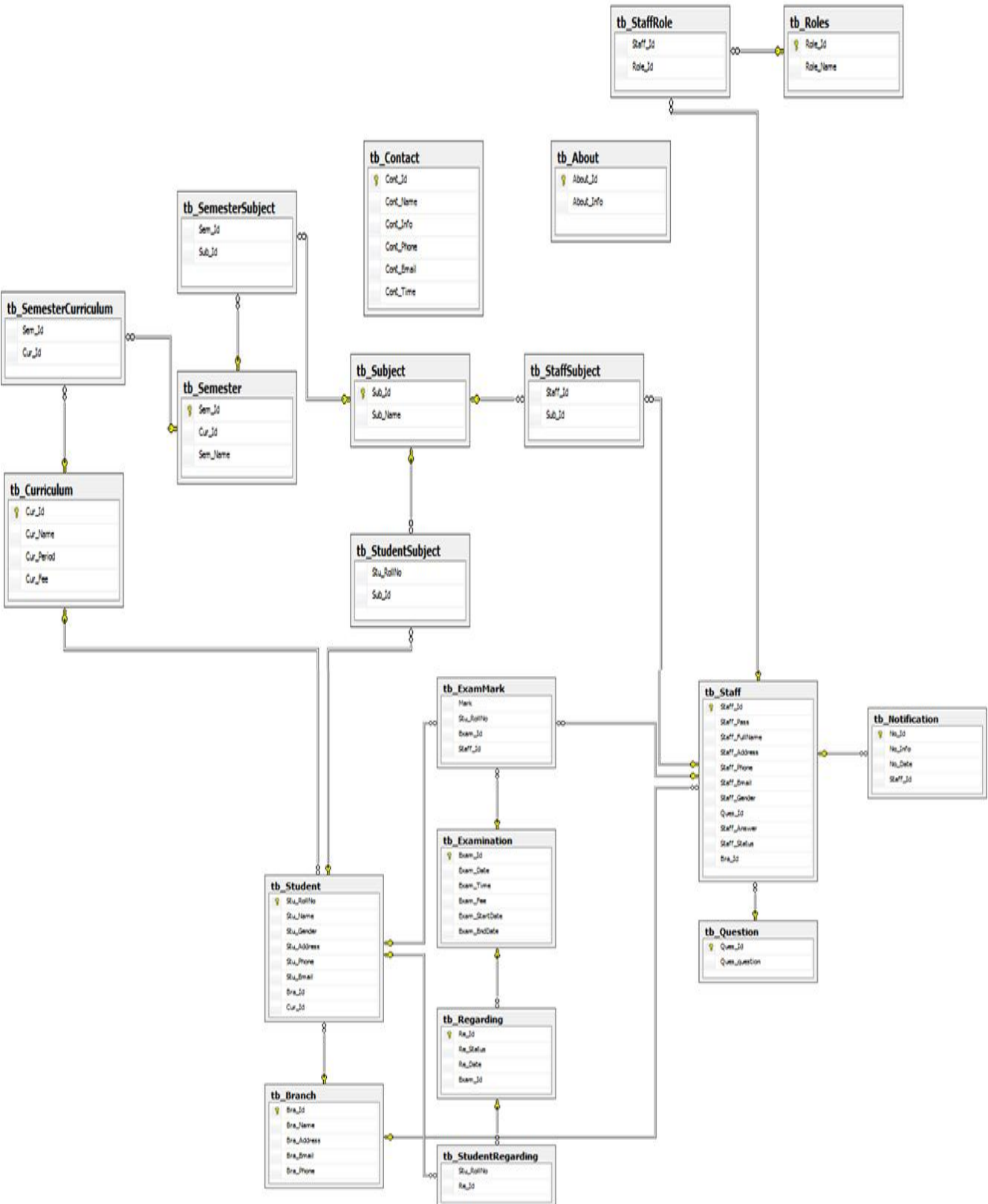
- + Actor: Staff
- + Interface: Update interface, with some textbox for staff enter information
- + Handling: check information, when staff update their information, system will check that information to know they can update or not
- + Database: staff table

■ Approve marks



- + Actor: Staff
- + Interface: Approve mark interface, for faculty approve mark upload by academic's people
- + Handling: check information, when faculty accept, system will display all mark of student on result page
- + Database: mark table

VII. Database Design/Structure (Form No.7)



■ About table

Field Name	Data Type	Constraints	Description
About_Id	nvarchar(10)	Primary key	About's identifier
About_Info	Nvarchar(MAX)	Allow null	About' information

■ Branch table

Field Name	Data Type	Constraints	Description
Bra_Id	nvarchar(10)	Primary key	Branch's identifier
Bra_Name	nvarchar(50)	Allow null	Branch's name
Bra_Address	nvarchar(50)	Allow null	Branch's address
Bra_Email	nvarchar(50)	Allow null	Branch's email
Bra_Phone	nvarchar(20)	Allow null	Branch's phone

■ Contact table

Field Name	Data Type	Constraints	Description
Cont_Id	nvarchar(10)	Primary key	Contact's identifier
Cont_Name	nvarchar(50)	Allow null	Contact's name
Cont_Info	nvarchar(50)	Allow null	Contact's information
Cont_Phone	nvarchar(20)	Allow null	Contact's phone
Cont_Email	nvarchar(50)	Allow null	Contact's email
Cont_Time	date	Allow null	Contact's time
Cont_Status	bit	Allow null	Contact's status

■ Curriculum table

Field Name	Data Type	Constraints	Description
Cur_Id	nvarchar(20)	Primary key	Curriculum's identifier
Cur_Name	nvarchar(50)	Allow null	Curriculum's name
Cur_Period	nvarchar(50)	Allow null	Curriculum's period
Cur_Fee	float	Allow null	Curriculum's fee

■ Examination table

Field Name	Data Type	Constraints	Description
Exam_Id	nvarchar(20)	Primary key	Examination's identifier
Sub_Id	nvarchar(20)	Allow null	Subject's identifier
Exam_Date	date	Allow null	Examination's date
Exam_Time	time(7)	Allow null	Examination's time
Exam_Fee	int	Allow null	Examination's fee
Exam_StartDate	date	Allow null	Examination's star date
Exam_EndDate	date	Allow null	Examination's end date

■ ExamMark table

Field Name	Data Type	Constraints	Description
Mark	float	Allow null	Exam mark
Stu_RollNo	nvarchar(20)	Foreign key references to Student Table	Student's roll no
Exam_Id	nvarchar(20)	Foreign key references to Examination Table	Examination identifier
Staff_Id	nvarchar(20)	Foreign key references to Staff Table	Staff's identifier

■ Notification table

Field Name	Data Type	Constraints	Description
No_Id	nvarchar(20)	Primary key	Notification's identifier
No_Title	nvarchar(MAX)	Allow null	Notification's title
No_Info	nvarchar(MAX)	Allow null	Notification's information
No_Date	date	Allow null	Notification's date
Staff_Id	nvarchar(20)	Foreign key references to Staff Table	Staff's identifier

■ Question table

Field Name	Data Type	Constraints	Description
Ques_Id	nvarchar(10)	Primary key	Question's identifier
Ques_question	nvarchar(50)	Allow null	Question

■ Regarding table

Field Name	Data Type	Constraints	Description
Re_Id	nvarchar(20)	Primary key	Regarding's identifier
Stu_RollNo	nvarchar(20)	Allow null	Student's roll no
Re_Status	nvarchar(MAX)	Allow null	Regarding's status
Re_Date	date	Allow null	Regarding's date
Exam_Id	nvarchar(20)	Foreign key references to Examination Table	Examination's identifier

■ Roles table

Field Name	Data Type	Constraints	Description
Role_Id	nvarchar(10)	Primary key	Roles's identifier
Role_Name	nvarchar(10)	Allow null	Roles's name

■ Semester table

Field Name	Data Type	Constraints	Description
Sem_Id	nvarchar(10)	Primary key	Semester's identifier
Sem_Name	nvarchar(20)	Not null	Semester's name

■ Semester Curriculum table

Field Name	Data Type	Constraints	Description
Sem_Id	nvarchar(10)	Foreign key references to Semester Table	Semester's identifier
Cur_Id	nvarchar(20)	Foreign key references to Curriculum Table	Curriculum's identifier

■ Semester Subject table

Field Name	Data Type	Constraints	Description
Sem_Id	nvarchar(10)	Foreign key references to Semester Table	Semester's identifier
Sub_Id	nvarchar(20)	Foreign key references to Subject Table	Subject's identifier

■ Staff table

Field Name	Data Type	Constraints	Description
Staff_Id	nvarchar(20)	Primary key	Staff's identifier
Staff_Pass	nvarchar(50)	Allow null	Staff's pass word
Staff_FullName	nvarchar(50)	Allow null	Staff's full name
Staff_Address	nvarchar(50)	Allow null	Staff's address
Staff_Birth	date	Allow null	Staff's birth
Staff_Phone	nvarchar(20)	Allow null	Staff's phone
Staff_Email	nvarchar(20)	Allow null	Staff's email
Staff_Gender	bit	Allow null	Staff's gender
Ques_Id	nvarchar(10)	Allow null	Question's identifier
Staff_Answer	nvarchar(50)	Allow null	Staff's answer
Staff_StartDate	date	Allow null	Staff's start date
Staff_Status	bit	Allow null	Staff's status
Bra_Id	nvarchar(20)	Foreign key references to Branch Table	Branch's identifier

■ Staff Role table

Field Name	Data Type	Constraints	Description
Staff_Id	nvarchar(20)	Foreign key references to Staff Table	Staff role identifier
Role_Id	nvarchar(10)	Foreign key references to Role Table	Roles's identifier

■ Staff Subject table

Field Name	Data Type	Constraints	Description
Staff_Id	nvarchar(20)	Foreign key references to Staff Table	Staff subject identifier
Sub_Id	nvarchar(20)	Foreign key references to Subject Table	Subject's identifier

■ Student table

Field Name	Data Type	Constraints	Description
Stu_RollNo	nvarchar(20)	Primary key	Student's roll no
Stu_Name	nvarchar(50)	Allow null	Student's name
Stu_Gender	bit	Allow null	Student's gender
Stu_Address	nvarchar(50)	Allow null	Student's address
Stu_Birth	date	Allow null	Student's birth
Stu_Phone	nvarchar(20)	Allow null	Student's phone
Stu_Email	nvarchar(50)	Allow null	Student's email
Bra_Id	nvarchar(10)	Foreign key references to Branch Table	Branch's identifier
Cla_Id	nvarchar(20)	Foreign key references to Curriculum Table	Curriculum identifier

■ Student Regarding table

Field Name	Data Type	Constraints	Description
Stu_RollNo	nvarchar(20)	Foreign key references to Student Table	Student regarding roll no
Re_Id	nvarchar(20)	Foreign key references to Regarding Table	Regarding's identifier

■ Student Subject table

Field Name	Data Type	Constraints	Description
Stu_RollNo	nvarchar(20)	Foreign key references to Student Table	Student subject roll no
Sub_Id	nvarchar(20)	Foreign key references to Subject Table	Subject's identifier

■ Subject table

Field Name	Data Type	Constraints	Description
Sub_Id	nvarchar(20)	Primary key	Subject's identifier
Sub_Name	nvarchar(30)	Allow null	Subject's name

VIII. Task Sheet (Form No.8)

Project Ref. No.: Project title: <i>Result Declaration Boards</i> Activity Plan Prepared by :		Date of Preparation of Activity Plan : 26/03/2014			
Sr. No.	Task	Actual Start Date	Actual Days	Team Member/ Names	Status
1	Problem Definition (Form 1)	21/01/2015	1	ANH TRAN TUAN	90%
2	Customer Requirement Specification (Form 2)	22/01/2015	8	ANH TRAN TUAN HIEN LAM VAN DUY VO THANH	80%
3	Diagram (Form 6) - UseCase Diagram	24/01/2015	5	ANH TRAN TUAN HIEN LAM VAN	70%
4	Diagram (Form 6) - Entity Relationship (ER)	24/01/2015	5	ANH TRAN TUAN DUY VO THANH	70%
5	Architecture & Design of the Project (Form No. 3):	23/01/2015	1	ANH TRAN TUAN	80%
6	Database Design/Structure (Form No.7)	29/01/2015	1	ANH TRAN TUAN	100%
7	Sequence Diagram, Flow Chart Diagram, Code: + Add Branch + Update Branch + Delete Branch + Add Notification + Updated Notification + Delete Notification + Add Class + Updated Class + Delete Class + Judge the student + View mark, notification + Add Exam_Mark + Update Exam_Mark + Delete Exam_Mark	30/01/2015	44	ANH TRAN TUAN	96%
8	Sequence Diagram, Flow Chart Diagram, Code: + Add Student + Update Student + Delete Student + Search Student + Course Registration + Add Examination + Update Examination + Delete Examination + Add Regarding	30/01/2015	44	DUY VO THANH	95%

	+ Delete Regarding				
9	Sequence Diagram, Flow Chart Diagram, Code: + Login + Change Password + Change Secret Question + Reset Password + Edit Profile + Add Staff + Update Staff + Delete Staff	30/01/2015	44	AN PHAM VO TRUONG	68%
10	Sequence Diagram, Flow Chart Diagram, Code: + Add Subject + Update Subject + Delete Subject + Add Curriculum + Update Curriculum + Delete Curriculum + Send Contact	30/01/2015	44	HIEN LAM VAN	70%
11	Design website	20/01/2015	54	ANH TRAN TUAN	90%