Exam Result Management System

Exam Result Management System

Exam Result Management System

ABSTRACT

The main objective of this project is to handle, manage and compute the exam results operations and provide examination result to the student in a simple way.
The project is useful for students and faculties for getting the results in simple manner.
By a result analyzer with subject status and marks is an application tool for displaying the results in secure way.

The system is intended for the student. And the privileges that are provided to student are to read and execute his/her result by providing user name and password for secure login and in case of new student the registration is available. And the guest user has the privilege only to read.

The whole result analyzer will be under the control of the administrator and the admin as the full privileges to read, write and execute the result. And admin gives the privileges to the Teacher and student and the guest user to access the result. The student can share or download his/her result.

CHAPTER ONE:

INTRODUCTION

Exam Result Management System

1.1Introduction

This project is focused on creating an automated students result management system using forms and reports. This is a computerized examinations results management system for faculty student's examination records. The manual method of students' academic result processing was found to be tedious, especially when carried out for a large number of students, this makes the entire process time-consuming and error prone. The system designed is meant to register students as soon as they have paid their departmental registration and only then they will be able to view their results. The system presents a single platform that will be used to manage the processing of all examination records within the faculty. The data used for testing was obtained from the Faculty of Computers and Information (FCI) and an empirical evaluation of the system shows that the system expedites the processing of students' examination results and the reporting of it.

1.2Objectives

The main aim is to develop the managing methods of control system and there calculation, make control operations easier for staff, students also benefits as every student can sign in to see his results.

All can enjoy the following benefits:

- Flexibility in use.
- Adaptable to changes.
- Easier maintenance.
- Simpler interface.
- Quicker development.

1.3Purpose

The EMS purpose is to make grades observation easier and cover many faculties of our university.

1.4Methodology

There are many types of management models: - waterfall model.
- Agile model.
- Hybrid model.

Waterfall model:
Used when there is a clear view of what the final results of the system.

Agile model:
Used when there need a speed to develop the system, in this model there must have skilled developers who are adaptable and able to think independently.

Hybrid model:
This model combine of both agile and waterfall and increase speed, improved quality.

So our project work hybrid model to get the most advantages:
- High-level design.
- Clear requirement analysis.
- Waterfall principles in designing.
- Powerful coding.
- Agile methodology of testing.
- Agile implementation.

1.5Tools

Our project (EMS) developed using the experience of MySQL, PHP with XAMPP control panel server. Web languages like:

- HTML5
- CSS
- JS
- BOOTSTRAP
- AJAX
- JQUERY
- ANGULAR

Exe:

- BI TOOL
- WORKBENCH

CHAPTER TWO:

ANALYSIS AND REQUIREMENT

Exam Result Management System

2.1 Introduction

Exam Result Management System (EMS) is the methodology of repairing and computing exams results. In this chapter we will see what the problems EMS solves and what are the requirements.

2.2 Who can use EMS?

There are two main members use or interact with our project:

- Staff :
  Staff (admin)
  Staff (user)

- Students.

Each one of them has its own domain in EMS project. We will show each one in detail.

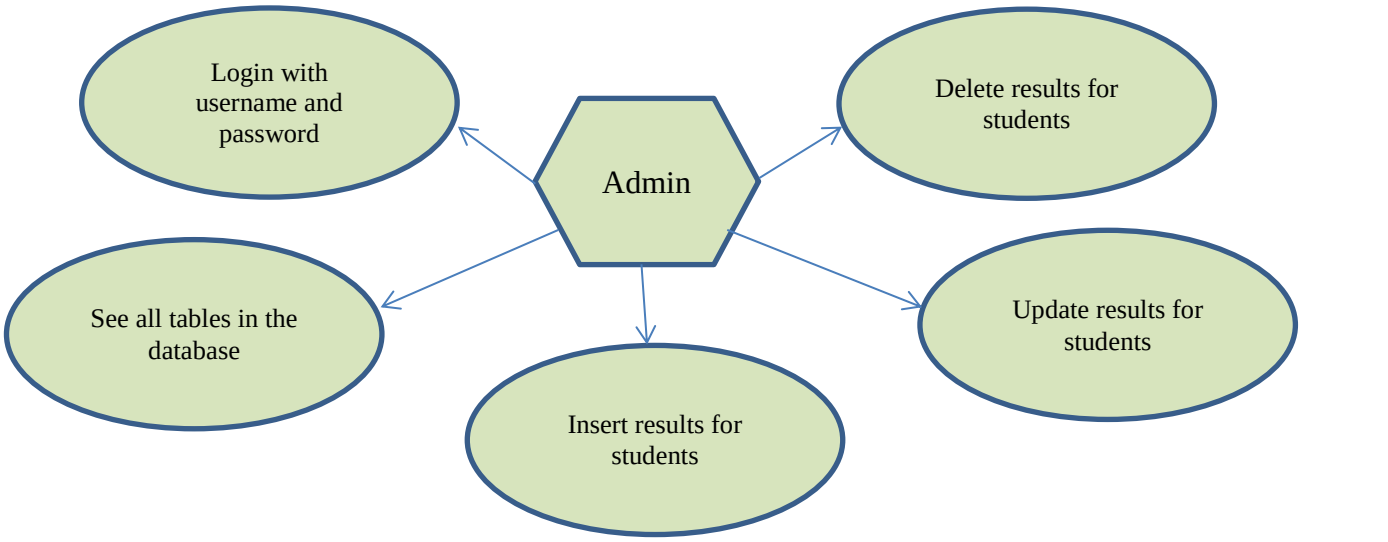
2.2.1 Staff (admin)

They some members of faculty staff have its own login password and a username which different form staff (users) because each one of them has different scope.

Admins authorizes are:

- Login with a unique username and password.
- See all tables in the database.
- Insert results for students.
- Update results for students.
- Delete results for students if needed.

(See figure 2.2.1)



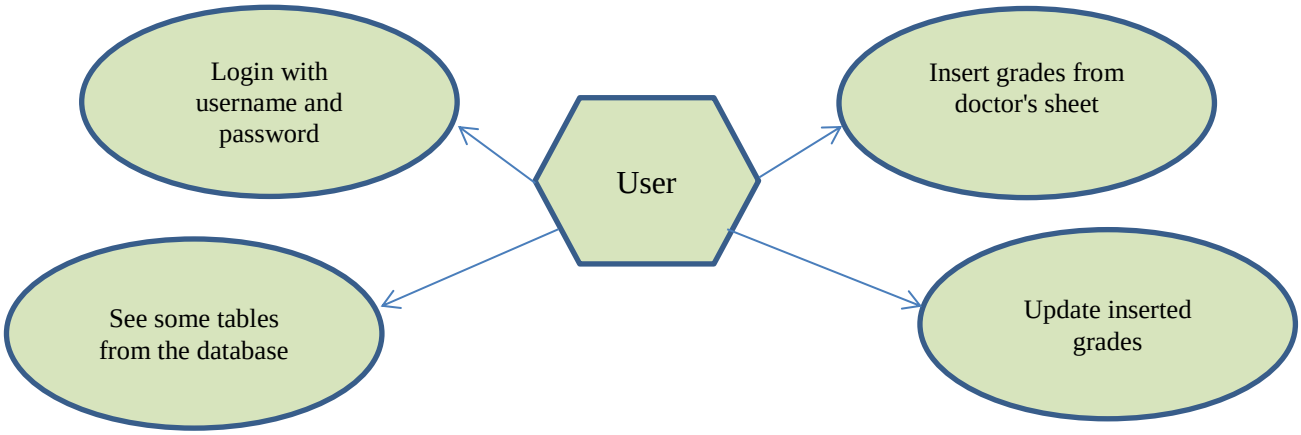
(Figure 2.2.1)

2.2.2 Staff (users)

Other group of faculty staff its job insert grades or copy results from doctors sheet which contains students grades (mid-grade, final-grade,..). They also authorize for:

- Login with its username and password given from a table in the database.
- See some tables from the database.
- Insert grades in its location as our project work.
- Update inserted grades.

(See figure 2.2.2)



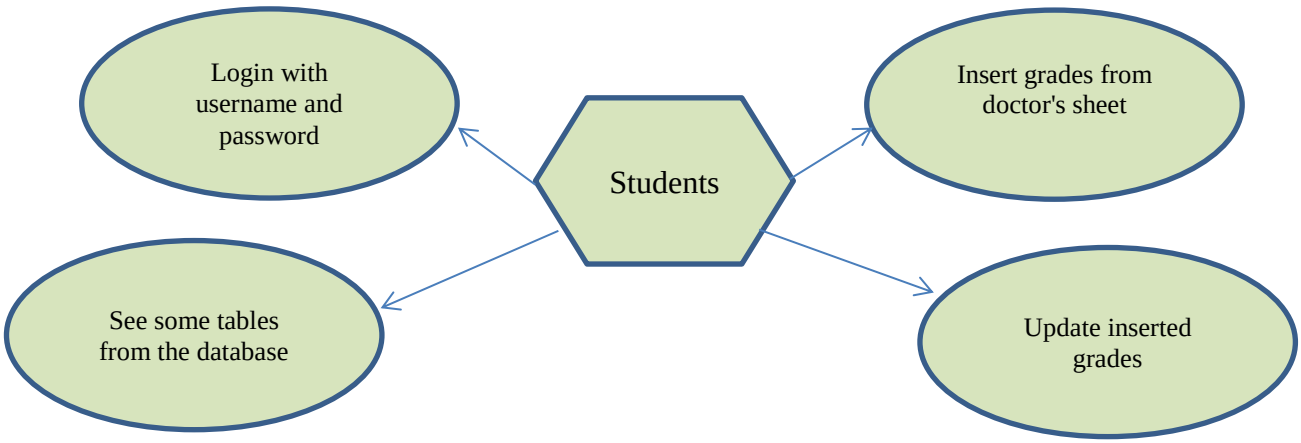
(Figure 2.2.2)

2.2.3 Students

Students also will be able to use our project, as students are the main members which staff works to help and make everything easy as they can. Each student has many options to do:

- Login with his ID and Personal Number which inserted in the database.
- See his results in a specific semester.
- Can see his GPA for credit hours system.
- Can see a report for his progress in last years.

(See figure 2.2.3)



(Figure 2.2.3)

2.3 Operations analysis and Flow Charts

Now let's see how the project works to do our operations for all members who use this project, operations which we provide are as follow:

Exam Result Management System

- How members login.
  - For admin (see Flow Chart 2.3.1.1).
  - For users (see Flow Chart 2.3.1.2).
  - For students (see Flow Chart 2.3.1.3).

- Admin specialization:

  1. See the results of a specific semester (see Flow Chart 2.3.2.1).
  2. See the results of a specific course (see Flow Chart 2.3.2.2).
  3. See a report for a specific course (see Flow Chart 2.3.2.3).
  4. See a report for a specific semester (see Flow Chart 2.3.2.4).
  5. See a report for a specific student (see Flow Chart 2.3.2.5).

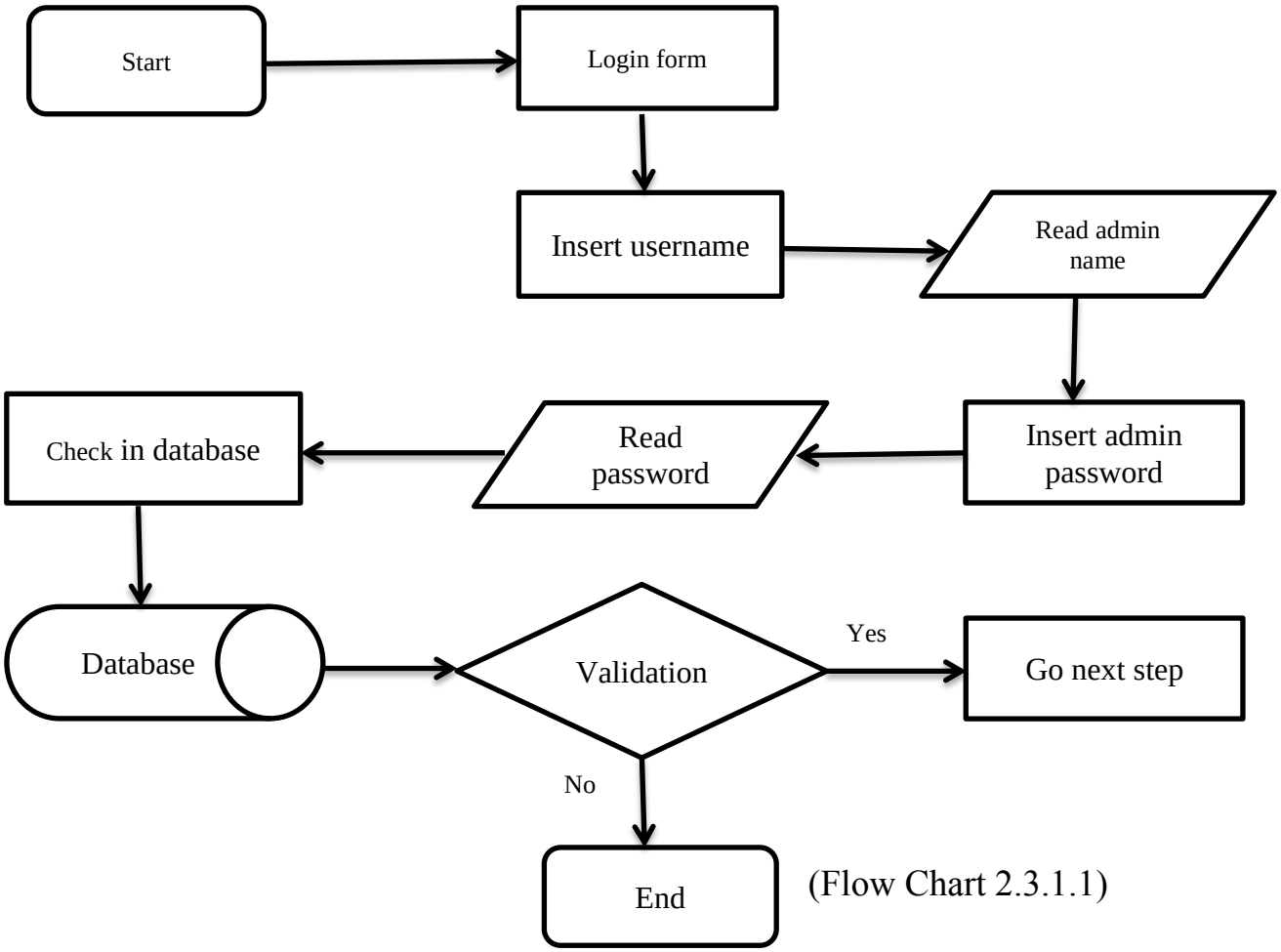
- Users specialization:

  1. See the results of a specific course (see Flow Chart 2.3.3.1).
  2. Insert grades for a specific course (see Flow Chart 2.3.3.2).
  3. Update grades for a specific course (see Flow Chart 2.3.3.3).
  4. Delete grades for a specific course if needed (see Flow Chart 2.3.3.4).

- Students specialization:

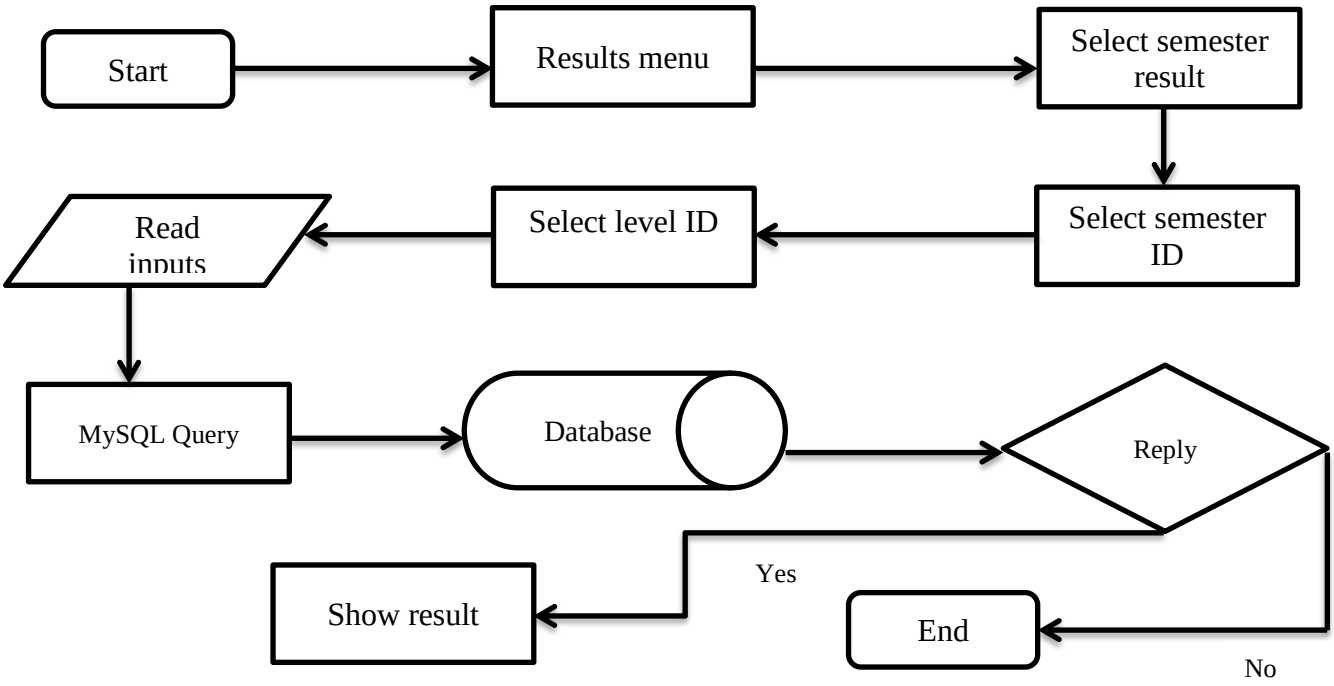
  1. See his results in a specific semester (see Flow Chart 2.3.4.1).
  2. See his report in a specific semester (see Flow Chart 2.3.4.2).

❖ Flow Charts Analysis:

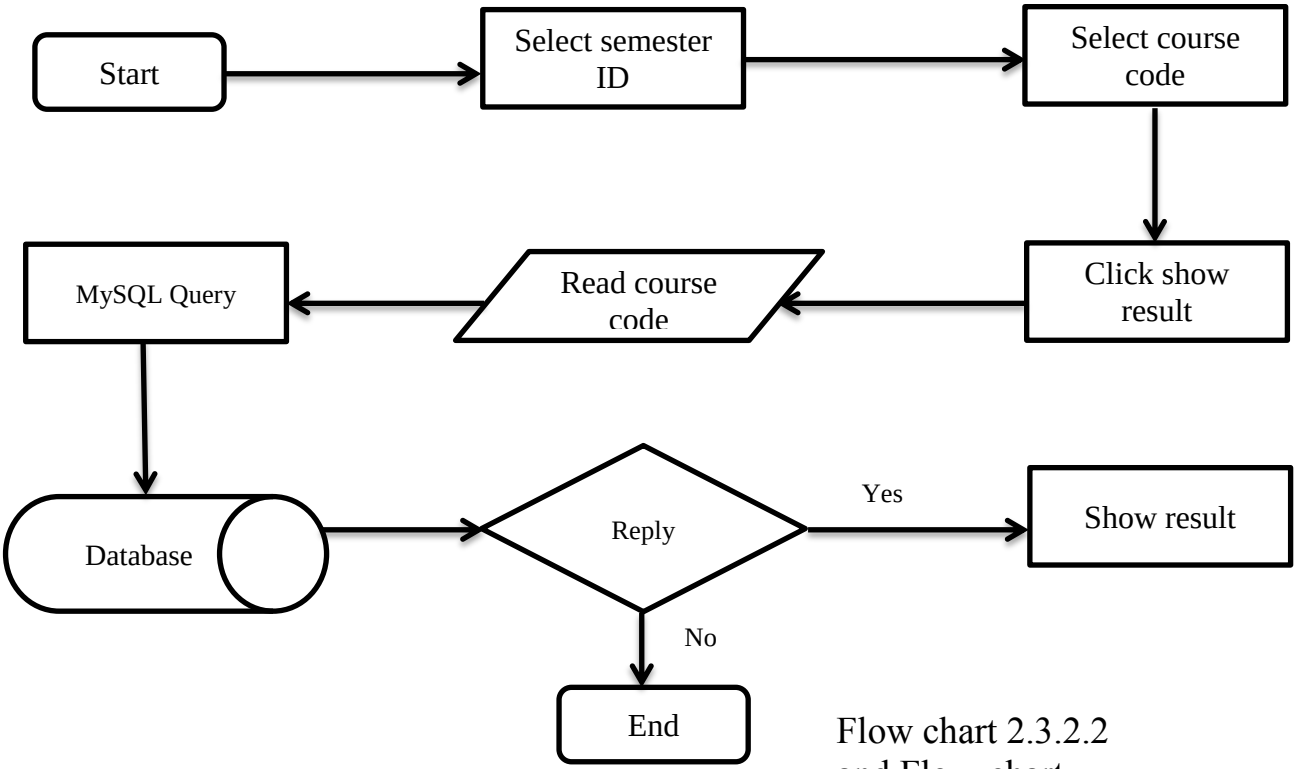


(Flow Chart 2.3.1.1)

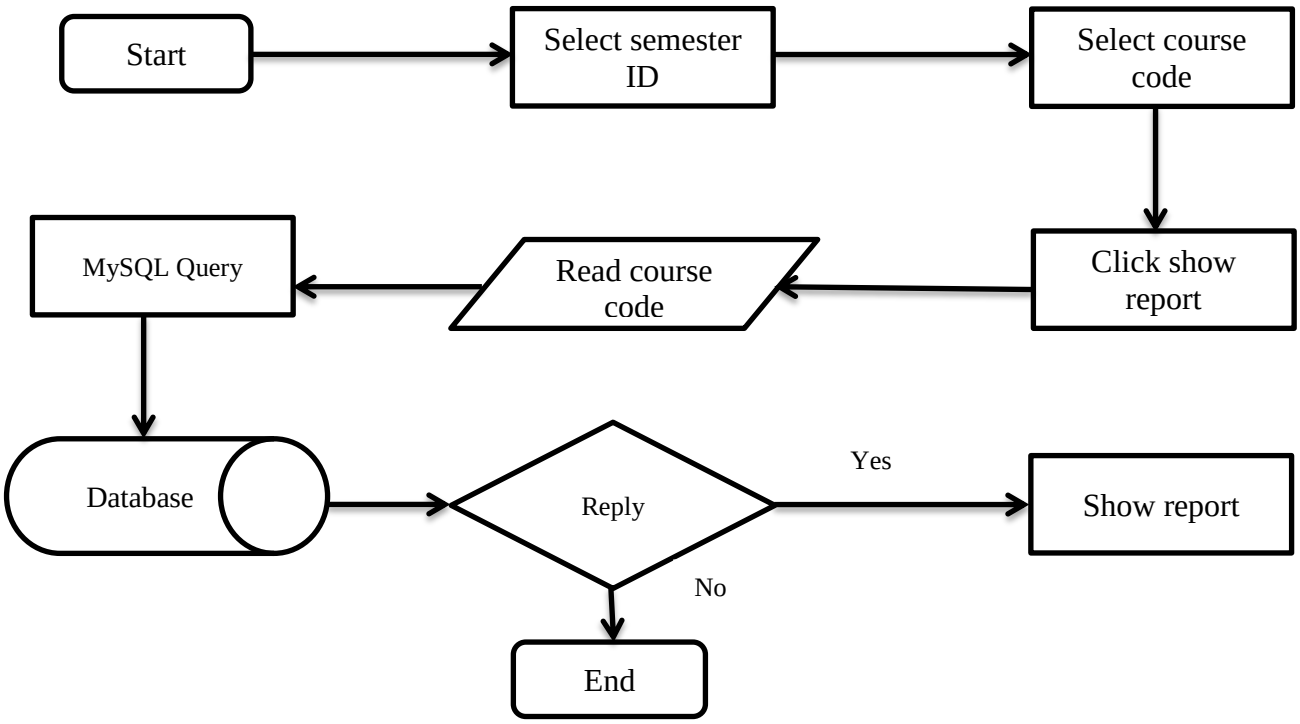
The same Flow chart for 2.3.1.2 and 2.3.1.2 the different users insert his own username and password and for students each one of them in username space he/she must inter his ID and in password space he/she must inter his personal number.



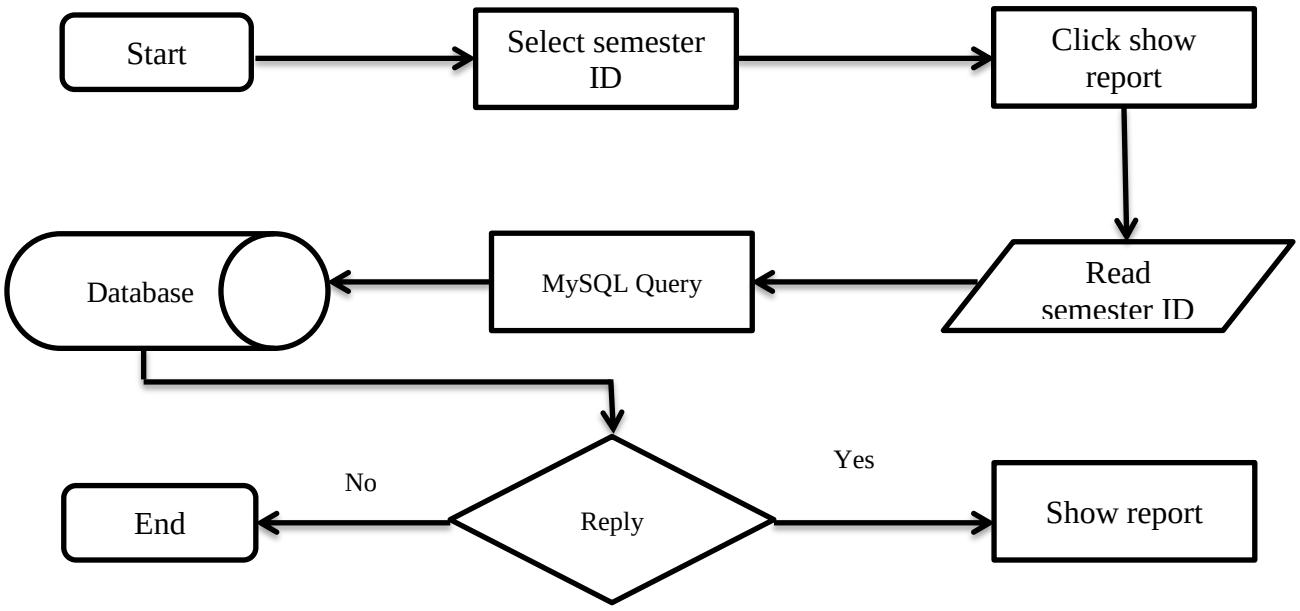
Flow chart 2.3.2.1



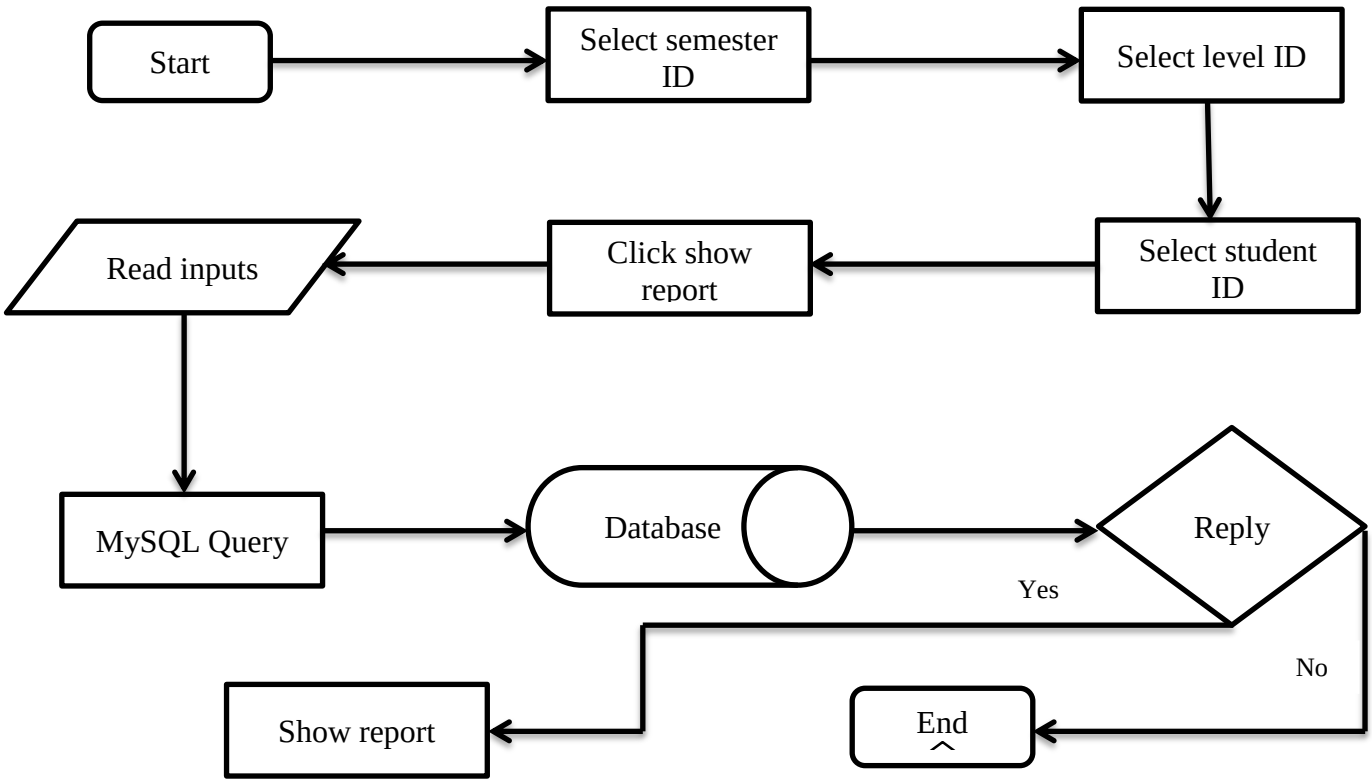
Flow chart 2.3.2.2 and Flow-chart 2.3.3.1 the same.



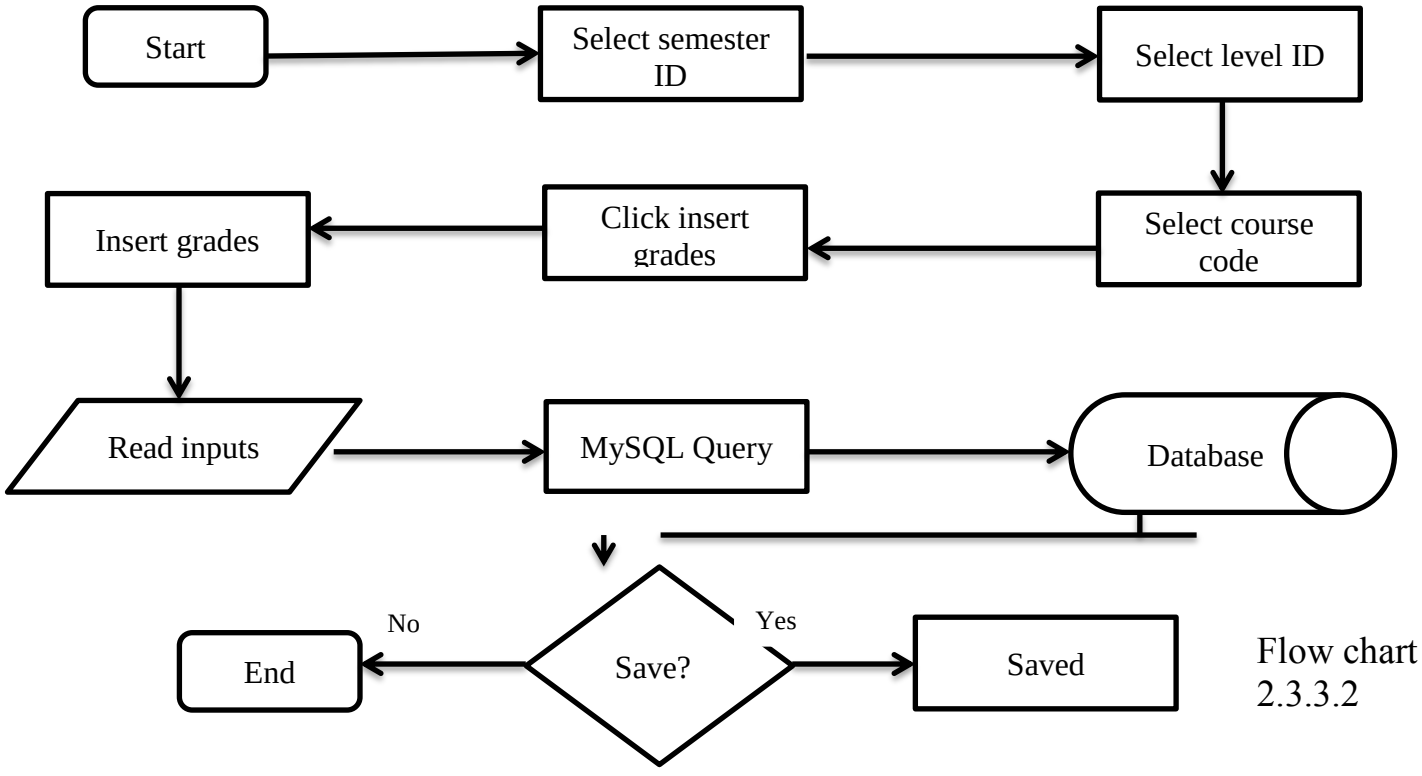
Flow chart 2.3.2.3



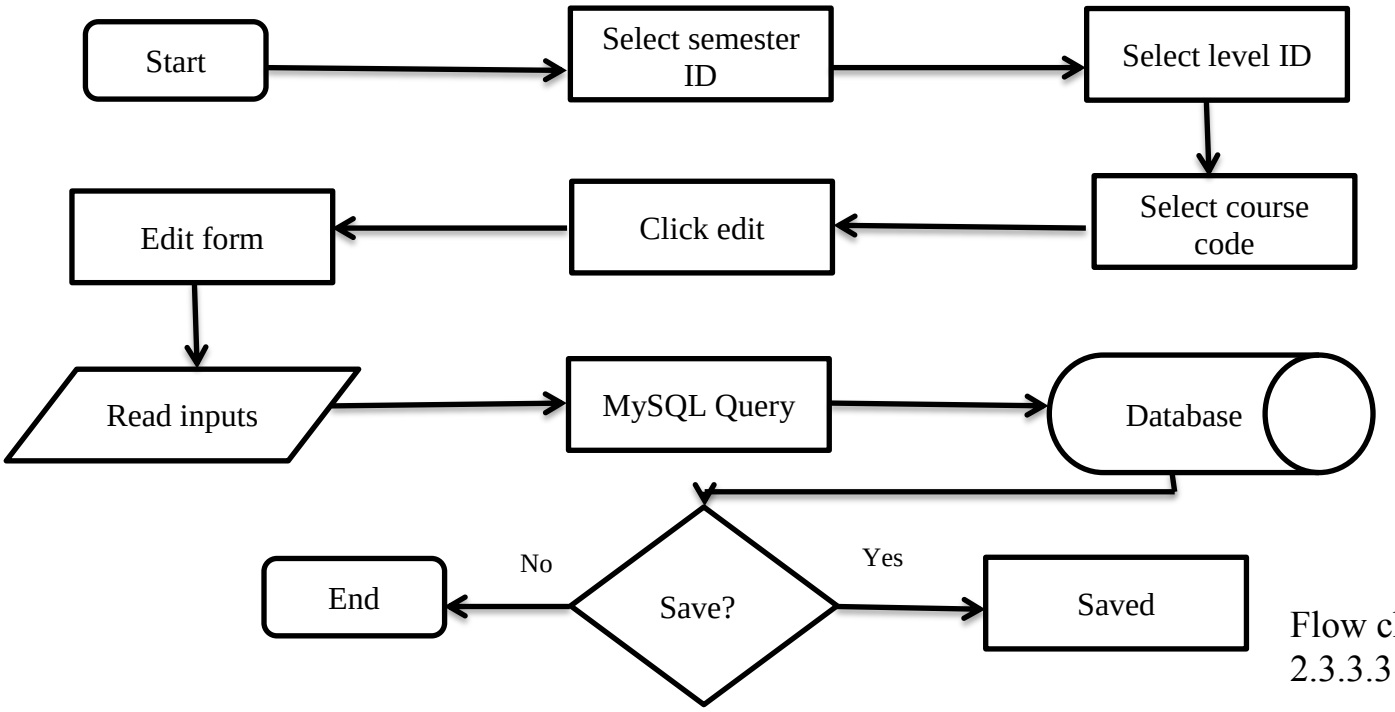
Flow chart 2.3.2.4



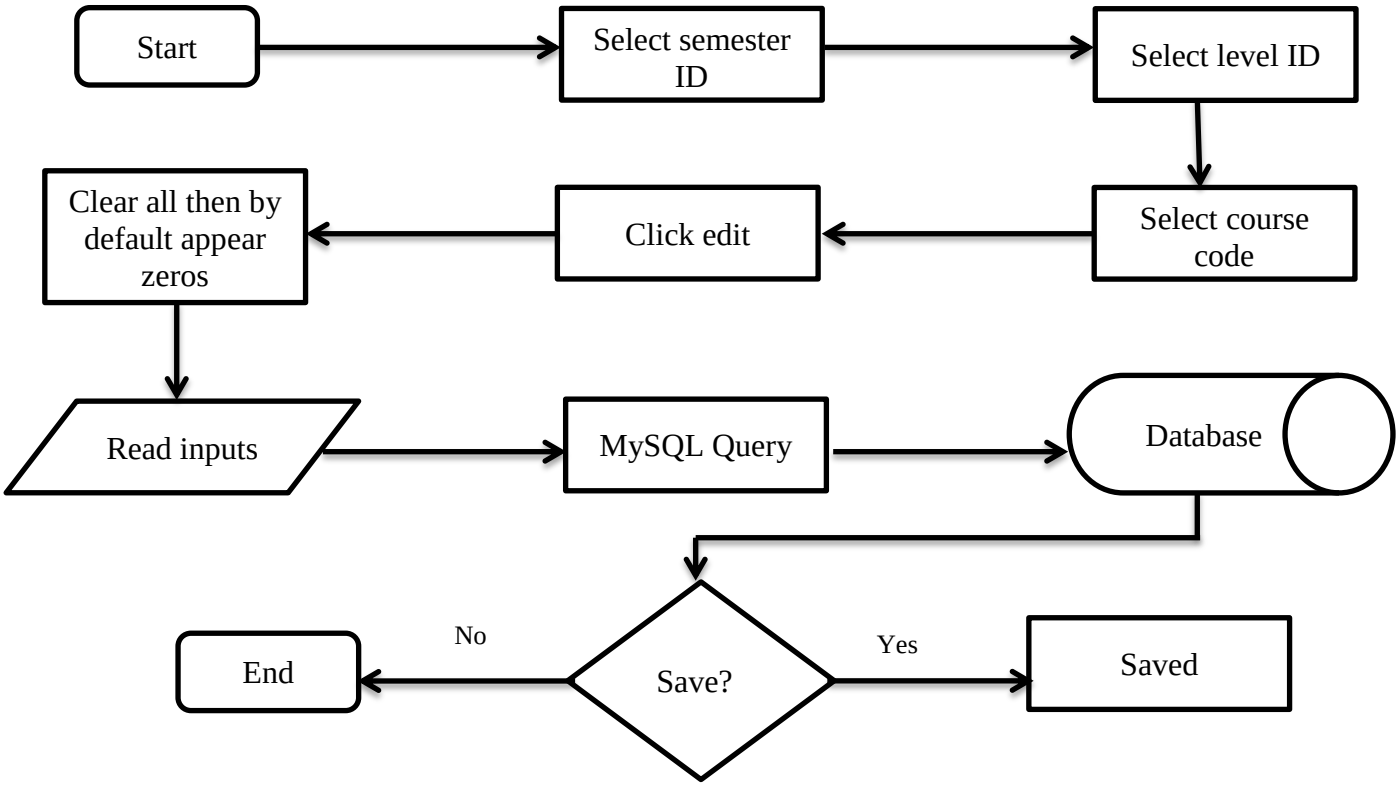
Flow chart 2.3.2.5



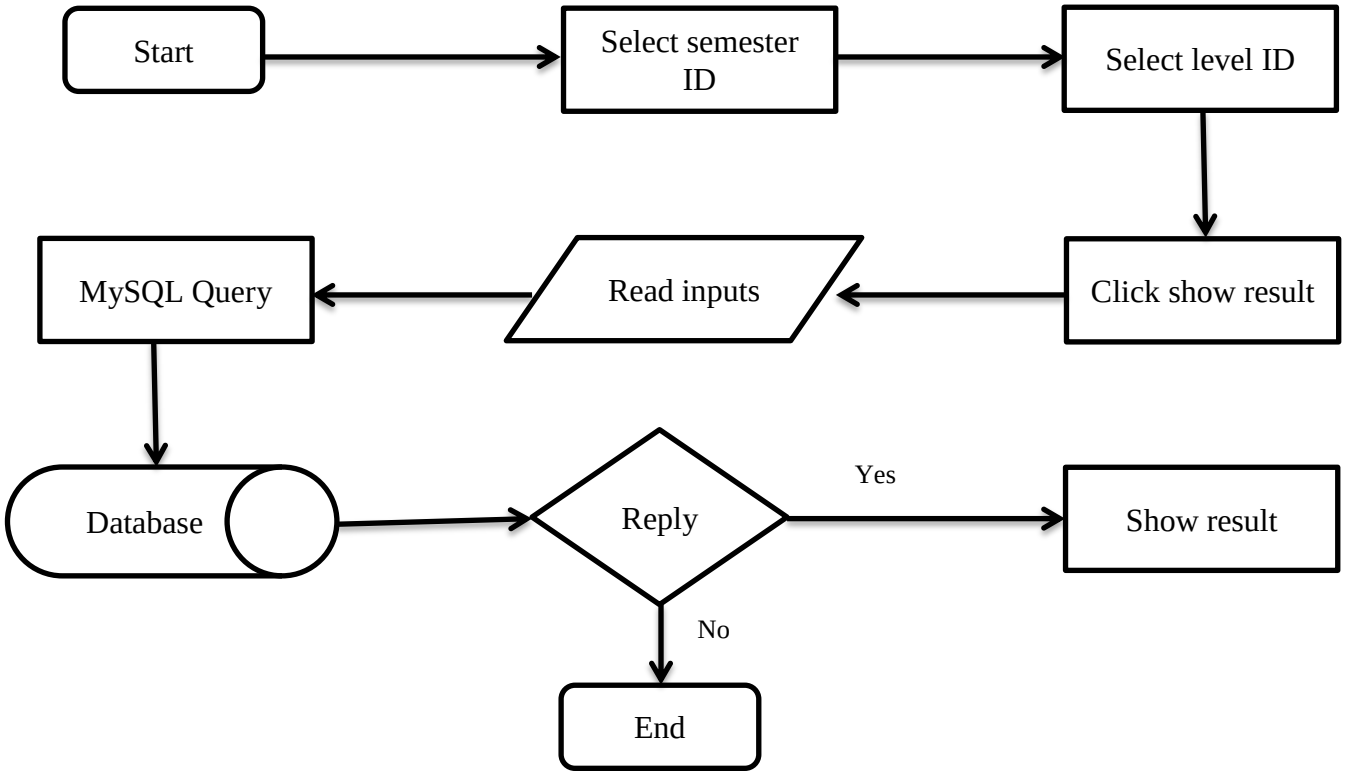
Flow chart 2.3.3.2



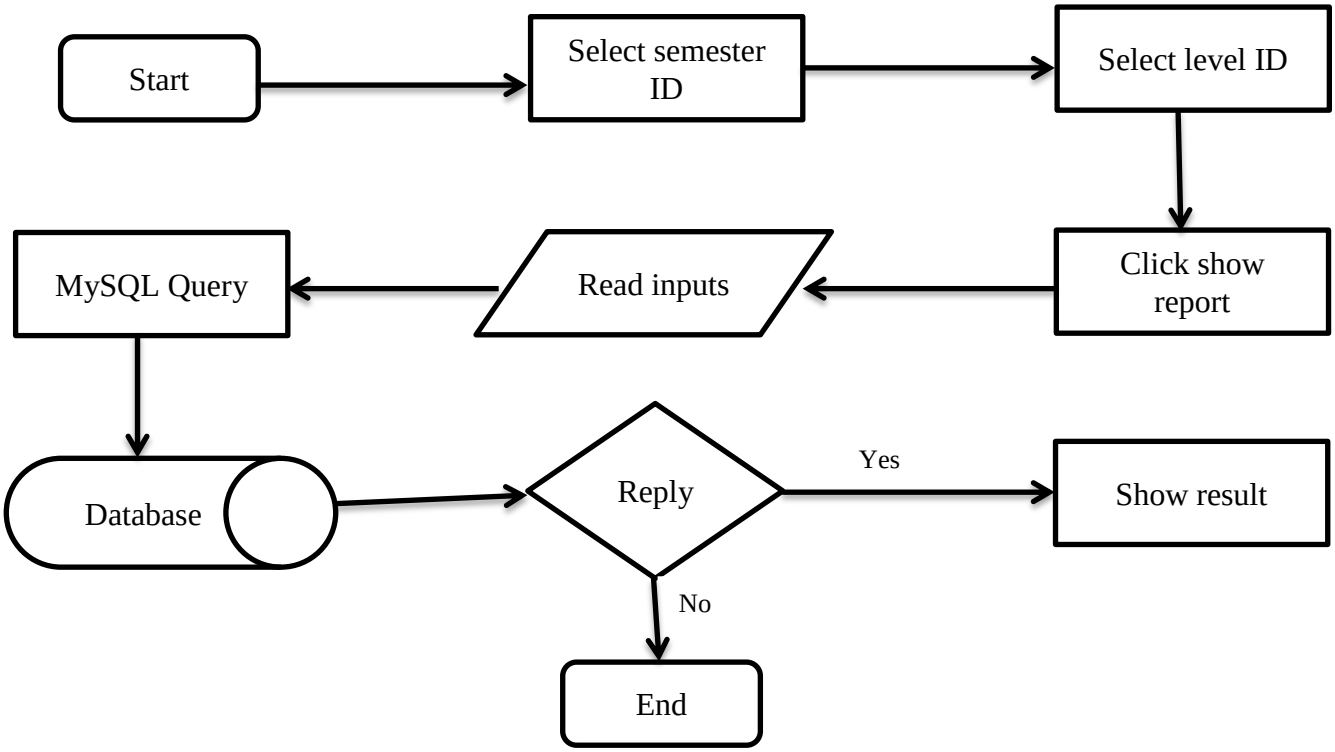
Flow chart 2.3.3.3



Flow chart 2.3.3.4



Flow Chart 2.3.4.1



Flow Chart 2.3.4.2

2.4 Project Requirements

There are four main requirements for our project EMS, characterized as follow:

- Input Requirements.
- Interface Requirements.
- Functional Requirements.
- Non-functional Requirements.

2.4.1 Input Requirements

- Login key which represented in username and password

For admins/users/students each one of them has its own user name and password which are inserted by Admin Registration Team. They must know their key. This key will lead to the second step according to the accessing key (for user, admin, student) .

- Select to go the next step

  - For admin: there are multiple options like chose semester ID, course ID, and level ID to see the results of the selected course for the student in the selected level. Chose faculty name, semester ID, and level ID to see the selected semester result for the selected level.

  - For users: they can chose semester ID, course ID, and level ID to see the results of the selected course for the students in the selected level and can update this results. Also they can chose semester ID to see a report of this semester. They able to print the results for a specific semester. Able to select a specific course to see a report of it.

- For students: each student can select semester ID and level ID to see his results in this semester, also he/she able to select semester ID to see his report in this semester.

- Save changes

After any action that admin/user/student take there would be alarm to ask he/she if the project save last changes or not ,so be careful in your actions and changes.

2.4.2 Interface Requirements

- ✓ User interface should be in the simplest design.
- ✓ User interface should provide reports about students and curses.
- ✓ User interface should provide help articles.
- ✓ User interface should provide a secured login for admins/users/students.
- ✓ User interface should be resalable and understood.
- ✓ User interface should be responsive.

2.4.3 Functional Requirements

Project should cover all the requirements of control management system which concerned with exams and its computations.

Provide reports to help supervisors in student's registration in next semesters adding to this that reports student himself could see them.

Project should run on all faculties' systems in Suez University.

2.4.4 Non-Functional Requirements

There are many hidden requirements which should be care about to make the project in the best way, as follow:

- Performance:

  Our project developed to help the university organization in managing exams operations so the performance should be good and understood, and should be fast, despite many users accessing the database at once, and responsibility.

- Reliability:

  Our project available any time in the week without any problems.

- Usability:

  Project must prevent admins/users/students from changing anything they not authorized whit. So each group has its own schedules.

CHAPTER THREE:

TOOLS

Exam Result Management System

3.1Introduction:

EMS implemented to cover the managing the part of control system which has its data bases and there many interfaces for staff and students. So the technologies which the EMS implemented with are:

- PHP/MySQL
- XAMPP
- WORK BENCH

Now let's see what this are .., how we used this .., what is the features of this technology.

MySQL is available under a dual licensing scheme. You can use it under an open source license like stack overflow free as long as you are willing to meet the terms of that license.

3.2.1MySQL Strengths

MySQL main competitors are postgreSQL , Microsoft SQL server , and oracle .
MySQL has much strength, as following:

- Portability.
- Easy of configuration.
- Low cost.
- High performance.
- Availability of support.
- Source code availability.

3.2.1.1Portability

MySQL can be used on many different UNIX systems as under Microsoft windows.

3.2.2What is SQL?

SQL stands for structure query language. It is the most standard language for accessing relational database management systems.

SQL is used to store data and retrieve it from a database. It is used in database systems such as MySQL, Oracle, postgreSQL, Sybase and Microsoft SQL server.

There is an ANSL standard for SQL, and database systems such as MySQL generally strive to implement this standard. There are some subtle differences between standard SQL and MySQL.

Some of these differences are planned to become standard in future versions of MySQL, and some are deliberate differences.

3.2.3The new in MySQL

Major changes in MySQL include:

- Views.
- Stored producers.
- Cursor support.
- Basic trigger support.

Other changes include more ANSL standard compliance and speed improvements. As follow:

Our version is 4.x which includes:

- Sub query support.
- Improved support for internationalization.
- Stored producers.
- Transaction-safe storage engine innoDB included as standard.
- MySQL query cache.

Supporting for-partitioning:

- Row based replication.
- Logging to tables.
- Event scheduling.

3.3Xampp

Xampp is a small and light apache distribution containing the most common web development technologies in a single package. Its contents, small size, and portability make it the ideal tool for students developing and testing applications in php and MySQL.

Xampp is available as a free download in two specific packages: full and lite.

- Full package download provides a wide array of development tools.
- Lite contains the necessary technologies that meet the Ontario skills. competition standards.

3.4MySQL Statements

- inserting statement.
- retrieving statement.
- update statement.
- deleting statement.
- dropping statement.

3.4.1Inserting statement

Recall that RDBMSs contain tables, which in turn contain rows of data organized into columns. Each row in a table normally describes some real world object or relationship. And the column values for that row store information about the real-world object. You can use the INSERT statement to put rows of data into the database.

```
INSERT [INFO] table [(column1, column2 ...)]
Values (value2, value3 ...);
```

3.4.2Retrieving statement

The workhorse of SQL is the select statement. It's used to retrieve data from a database by selecting rows that match specified criteria from a table.

There are a lot of an options and different ways to use the SELECT statement. The basic form of a SELECT is:

```
SELECT [options] items
 [INTO file details]
 FROM tables
 [WHERE conditions]
 [GROUP BY group type]
 [HAVING where definition]
 [ORDER BY order type]
 [LIMIT limit_criteria]
 [Procedure name (argument)];
```

3.4.3Update statement

In addition to retrieving data from the database, you often want to change it. For example, you might want to increase the grade of a student in the database. You can do this using update statement.

The usual form of an update statement is:

```
UPDATE [LOW PRIMARY] [IGNORE] table name
SET columnl-expression1, column2-expression2,...
[WHERE condition]
[ORDER BY order criteria]
[ORDER BY order criteria];
```

3.4.4Deleting statement

Deleting rows from the database is simple. You can do this using the DELETE statement, which generally looks like this:

```
DELETE [LOW PRIMARY] [QUICK] IGNORE] FROM table
[WHERE condition]
[ORDER BY order_cols]
[LIMIT number];
```

On its own, all the rows in a table will be deleted, so be careful! Usually, you want to delete specific rows, and you can specify the ones you want to delete with a Where clause. You might do this, if, for example, a particular student were no longer failing or if a particular doctor hadn't placed any grades for a student in a course and you wanted to do some housekeeping:

```
Delete from students
Where course id = a number;
```

The LIMIT clause can be used to limit the maximum numbers of rows that are actually deleted or selected. ORDER BY is usually used in conjunction with LIMIT. LOW PRIMARY and IGNORE works as they elsewhere. QUICK may faster on MyISAM tables.

3.4.5Dropping statement

At times, you may want to get rid of an entire table. You can do this with DROP TABLE statement. This process is very simple, and it looks like this:

```
DROP TABLE tablename;
```

This query deletes all the rows in the table and the table itself, so be careful using it.

3.5PHP

PHP is a server-side scripting language designed specifically for the Web. Within a HTML page you can embed PHP code that will be executed each time the page is visited. Your PHP code is interpreted at the web server and generates HTML or other output that the visitor will see.

PHP is an Open Source project, which means you have access to the source code and can use, alter, and redistribute it all without charge.

PHP Originally stood for Personal Home Page but was changed in line the GNU recursive naming convention (GNU -Gnu's Not Unix) and now stands for PHP Hypertext Preprocessor.

This version saw a complete rewrite of the underlying Zend engine and some major improvements to the language.

The home page for PHP is available at http://www.php.net
The home page for Zend Technologies is http://www.zend.com

3.5.1Some of PHP's features

Comparison to these products, PHP has much strength, including the following:

- Performance.
- Scalability.
- Built-in libraries for many common web tasks.
- Availability of support and documentation.
- Ease of learning and use.
- Strong object-oriented support.
- Interfaces to many different database systems.
- Flexibility of development approach.
- Low-cost.
- Portability.

3.5.1.1Performance

PHP is very fast .Using a single inexpensive server, you can serve millions of hits per day. Benchmarks published by Zend Technologies (http://www.zend.com) show PHP outperforming its competition.

3.5.1.2Scalability

PHP has what Ramus Leadoff frequently refers to as a "shared-nothing" architecture this means that you can effectively and cheaply implement horizontal scaling with large numbers of commodity servers.

3.5.1.3Built-in Libraries

Because PHP was designed for use on the Web, it has many built-in functions for performing many useful web-related tasks, you can generate images on the fly, connect to web services and other network services, parse XML. Send email, work with cookies, and generate PDF documents, all with just a few lines of code.

3.5.1.4Portability

PHP is available for many different operating systems. You can write PHP code on free Unix-like operating systems such as Linux and FreeBSD, commercial UNIX versions such as Solaris and IRIX, OS X, or on different versions of Microsoft Windows. Well-written code will usually work without modification on a different system running PHP.

3.6The new in PHP

You may have recently moved to PHP 4.x versions. As you would expect in a new major Vers1on, it has some significant changes. The Zend engine beneath PHP has been rewritten for this version. Major new features are as follows:

- Better object-oriented support built around a completely new object model.
- Exception for scalable, maintainable error handling.
-SimpleXML for easy handling of XML data.

Other changes include moving some extensions out of the default PHP install and into the PECL library, improving streams support, and adding SQLite.

PHP 5.2 added a number of useful features including:
- JSON extension for better JavaScript interoperability.
- Better date and time handling.
- The new input filtering extension for security purpose.
- File uploads progress tracking.

3.7Other PHP-Database interfaces

PHP supports libraries for connecting to a large number of databases, including Oracle, Microsoft SQL Server, and PostgreSQL.

In general, the principles of connecting to and querying any of these databases are much the same. The individual function names vary, and different databases have slightly different functionality, but if you can connect to MySQL, you should be able to easily adapt your knowledge to any of the others.

If you want to use a database that doesn't have a specific library available in PHP, you can use the generic ODBC functions. ODBC, which stands for Open Database Connectivity, is a standard for

Connections to databases. It has the most limited functionality of any of the function sets, for fairly obvious reasons. If you have to be compatible with everything, you can't exploit the special futures of anything.

3.7.1Uploadingfiles

In PHP functionality is Support for uploading files, Instead of files coming from the server to the browser using HTTP, they go in the opposite direction that is, from the browser to the server. Usually, you implement this configuration with an HTML from interface.

3.8Why Use PHP and MYSQL?

One of best features of both PHP and MYSQL is that they work with any major operating system and many of the minor ones.

The majority of PHP code can be written to be portable between operating systems that are operating system dependent.

Whatever hardware, operating system, and web s server you choose, we believe you should seriously consider using PHP and MYSQL.

3.9HTML5

HITMLS's goals are grand. The specification is sprawling an often misunderstood. Given the confusion, the goals of this chapter are not only to summarize what is new about HTML5 and provide a roadmap to the element reference that follows, but to also expose some of the myths and misconceptions about this exciting new approach to markup.

3.9.1HTML Politics as Usual

The Web is an interesting place technology-swine because the mob tends to rule. Very often, well-defined specifications will be built only to be avoided or replaced by ad hoc Specifications that appear to spring out of nowhere.

HTML5 tries to tame the mob and bring a bit of order to the chaos, but that doesn't come easily, particularly when politics and competition are involved. On the Web, there are those who promote openness, and those who promote new proprietary features for their own browsers. Some will label such organizations good or bad, and declare their technology the one true way over others, such promotion of us versus them can create loyal followers, but the author finds some of the discussion more than a bit disingenuous.

Web technologies that were once maligned as proprietary Microsoft features, such as inner HTML, content editable, Ajax XML http request object, and more, have been quietly absorbed into the open Web community. Other capabilities such as CSS transformations, behaviors, Web fonts, and animations found in Internet Explorer-in many cases for the better part of a decade- are also maligned as proprietary only to be reintroduced with slight syntax differences by other browser vendors to hails of the progress of the open Web, "Today proprietary, tomorrow standard'" seems to be the rule of Web standards, and it would seem that now HITMLS is doing its part to Continue politics as usual. Google has already begun a tremendous push to promote HTML5.

The problem is the term is basically being used as a comparison as to what a major competitor is not supporting, more than a lucid discussion of the emerging technology. Unfortunately, from my observations, when most people speak of HTML5, it is more as a code for open Web or even anti-Microsoft, which reminds me of other misused terms of the last browser battles. Let's hope that cool heads prevail in the standards fights that will likely ensue.

3.9.2Imperfect Improvement

HTML5 is an imperfect improvement for an imperfect Web world. We simply can't force the masses to code their markup right; TIMLS doesn't try to accomplish this fool's errand but instead finds a reasonable path of defining what to do with such malformed markup at the browser level. The HIML5 specification is too big. It's a sprawling specification and covers many things. However, it tries to document that which is ad hoc and make decisions about issues left unsolved. Something is better than nothing. The HITMLS specification is a work in progress. Writing a book about such a moving target is more than a bit of a challenge.

3.10CSS

CSS is a language that describes the style of an HTML document.

CSS describes how HTML elements should be displayed.

CSS is used to define styles for your web pages, including the design, layout and variations in display for different devices and screen sizes.

3.10.1Why CSS?

CSS care about the design and how you can implement your project ,so we use the CSS sheets to make our project more enhanced and to be easy for users .

3.10.2CSS Solved a Big Problem

HTML was NEVER intended to contain tags for formatting a web page!

HTML was created to describe the content of a web page, like:

<h1>This is a heading</h1>

 This is a paragraph.

When tags like <font>, and color attributes were added to the HTML 3.2 specification, it started a nightmare for web developers. Development of large websites, where fonts and color information were added to every single page, became a long and expensive process.

CHAPTER FOUR:

PROJECT DESIGN AND IMPLEMENTATION

4.1 Introduction

After we analysis the project , know the used tools , and project requirements, now in this chapter we will see the design and coding static and dynamic pages with HTML , CSS and PHP including the require information from our database.

4.2 Overall system design objectives

The overall system design objectives done to provide an efficient, modular design that will reduce the system's complexity, facilitate change and result in an easy implementation.

This will be accomplished by designing strongly cohesion system with minimal coupling, in addition, this document will provide good interface design models that are consistent user friendly and will provide straight forward transition through the various system functions.

4.3 Database Tables

Our project (EMS) implemented using multiple tables as following:

4.3.1 Course

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
1	ID	int(11)			No	None			Change Drop More
2	Code	varchar(255)	utf8_general_ci		Yes	None			Change Drop More
3	ArabicName	varchar(255)	utf8_general_ci		Yes	None			Change Drop More
4	Course	varchar(255)	utf8_general_ci		Yes	None			Change Drop More
5	Credits	int(11)			Yes	None			Change Drop More
6	Prerequisits	varchar(255)	utf8_general_ci		Yes	None			Change Drop More
7	CourseTypeID	int(11)			Yes	None			Change Drop More
8	Teaching Hours Lecture	int(11)			Yes	None			Change Drop More
9	Teching Hours Tutorial	varchar(255)	utf8_general_ci		Yes	None			Change Drop More
10	TeachingHours Practical	int(11)			Yes	None			Change Drop More
11	Midterm	decimal(8,2)			Yes	None			Change Drop More
12	Year Work Grades O	decimal(8,2)			Yes	None			Change Drop More
13	Year Work Grades PE	decimal(8,2)			Yes	None			Change Drop More
14	YearWorkGrades G	decimal(8,2)			Yes	None			Change Drop More
15	FinalExam	decimal(8,2)			Yes	None			Change Drop More
16	TimeofExam	int(11)			Yes	None			Change Drop More
17	LeveLID	int(11)			Yes	None			Change Drop More
18	DepartmentID	int(11)			Yes	None			Change Drop More
19	OrderCode	int(11)			Yes	None			Change Drop More
20	StatusID	int(11)			Yes	None			Change Drop More
21	CurriculumID	int(11)			Yes	None			Change Drop More

4.3.2 Course-grade

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
1	ID	int(11)			Yes	None			Change Drop More
2	Grade_English	varchar(255)	utf8_general_ci		Yes	None			Change Drop More
3	Grade_Arabic	varchar(255)	utf8_general_ci		Yes	None			Change Drop More
4	OrderCode	int(11)			Yes	None			Change Drop More
5	Points	float			Yes	None			Change Drop More
6	Percentage	float			Yes	None			Change Drop More
7	CurriculumID	int(11)			Yes	None			Change Drop More
8	Notes	varchar(255)	utf8_general_ci		Yes	None			Change Drop More

4.3.3 Course-semester

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
1	ID	int(11)			No	None			Change Drop More
2	SemesterID	int(11)			Yes	None			Change Drop More
3	CourseID	int(11)			Yes	None			Change Drop More
4	DoctorID	int(11)			Yes	None			Change Drop More

4.3.4 Curriculum

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
1	ID	int(11)			No	None			Change Drop More
2	CurriculumEngName	varchar(255)	utf8_general_ci		Yes	None			Change Drop More
3	CurriculumArabicName	varchar(255)	utf8_general_ci		Yes	None			Change Drop More
4	DepartmentID	int(11)			Yes	None			Change Drop More
5	statusID	int(11)			Yes	None			Change Drop More
6	OrderCode	int(11)			Yes	None			Change Drop More
7	FacultyID	int(11)			Yes	None			Change Drop More
8	Limit_Fail_Final	decimal(18,4)			Yes	None			Change Drop More
9	Limit_Fail_Total	decimal(18,4)			Yes	None			Change Drop More

4.3.5 Department

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
1	ID	int(11)			No	None			Change Drop More
2	Departmenttxt	varchar(255)	utf8_general_ci		Yes	None			Change Drop More
3	DeptartmentCode	varchar(255)	utf8_general_ci		Yes	None			Change Drop More
4	FacultyID	int(11)			Yes	None			Change Drop More
5	OrderCode	int(11)			Yes	None			Change Drop More
6	CurriculumID	mediumtext	utf8_general_ci		Yes	None			Change Drop More

4.3.6 Doctor

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
1	ID	int(11)			No	None			Change Drop More
2	NameTxt	varchar(255)	utf8_general_ci		Yes	None			Change Drop More
3	StatusID	int(11)			Yes	None			Change Drop More
4	OrderCode	int(11)			Yes	None			Change Drop More
5	Arabic_doctorName	varchar(255)	utf8_general_ci		Yes	None			Change Drop More

4.3.7 Faculty

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
1	ID	int(11)			No	None			Change Drop More
2	FacultyName_Arabic	varchar(255)	utf8_general_ci		Yes	None			Change Drop More
3	FacultyDescription	mediumtext	utf8_general_ci		Yes	None			Change Drop More
4	OrderCode	int(11)			Yes	None			Change Drop More
5	FacultyName_English	varchar(255)	utf8_general_ci		Yes	None			Change Drop More

4.3.8Level

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
1	ID	int(11)			No	None			Change Drop More
2	LevelTxt	varchar(255)	utf8_general_ci		Yes	None			Change Drop More
3	LevelNumber	int(11)			Yes	None			Change Drop More
4	LevelCreditHours	float			Yes	None			Change Drop More
5	LevelTxt_Arabic	varchar(255)	utf8_general_ci		Yes	None			Change Drop More
6	CurriculumID	int(11)			Yes	None			Change Drop More

4.3.9 Registration

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
1	ID	int(11)			No	None			Change Drop More
2	Course_semesterID	int(11)			Yes	None			Change Drop More
3	StudentID	int(11)			Yes	None			Change Drop More
4	YearWork	decimal(18,4)			Yes	None			Change Drop More
5	Practical	decimal(18,4)			Yes	None			Change Drop More
6	MidTermExam	decimal(18,4)			Yes	None			Change Drop More
7	FinalExam	decimal(18,4)			Yes	None			Change Drop More
8	statusID	int(11)			Yes	None			Change Drop More
9	AdvisorApprovalID	int(11)			Yes	None			Change Drop More

4.3.10 Semester

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
1	ID	int(11)			No	None			Change Drop More
2	Semester	varchar(255)	utf8_general_ci		Yes	None			Change Drop More
3	SemesterFullName	varchar(255)	utf8_general_ci		Yes	None			Change Drop More
4	SemesterArabicName	varchar(255)	utf8_general_ci		Yes	None			Change Drop More
5	SemesterEnumID	int(11)			Yes	None			Change Drop More
6	StatusID	int(11)			Yes	None			Change Drop More

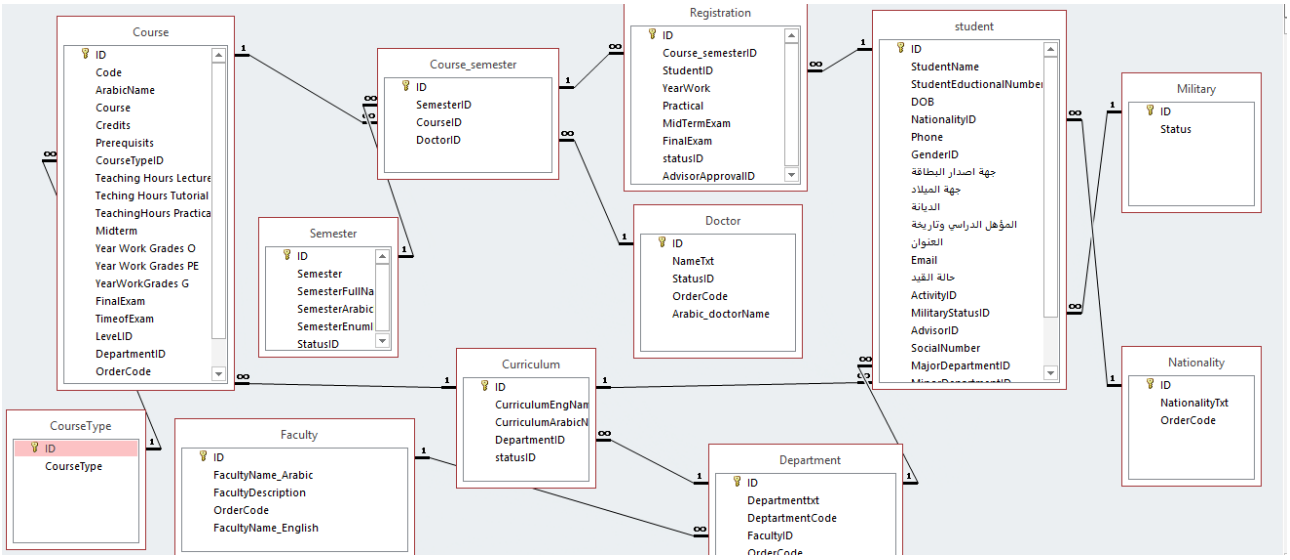
4.3.11 Student

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
1	ID	int(11)			No	None			Change Drop More
2	StudentName	varchar(255)	utf8_general_ci		Yes	None			Change Drop More
3	StudentEductionalNumber	varchar(255)	utf8_general_ci		Yes	None			Change Drop More
4	StudentEductionalNumber1	int(11)			Yes	None			Change Drop More
5	DOB	varchar(40)	utf8_general_ci		Yes	None			Change Drop More
6	NationalityID	int(11)			Yes	None			Change Drop More
7	Phone	varchar(255)	utf8_general_ci		Yes	None			Change Drop More
8	GenderID	int(11)			Yes	None			Change Drop More
9	جهة اصدار البطاقة	varchar(255)	utf8_general_ci		Yes	None			Change Drop More
10	جهة الميلاد	varchar(255)	utf8_general_ci		Yes	None			Change Drop More
11	الديانة	varchar(255)	utf8_general_ci		Yes	None			Change Drop More
12	المؤهل الدراسي وتاريخة	varchar(255)	utf8_general_ci		Yes	None			Change Drop More
13	العنوان	varchar(255)	utf8_general_ci		Yes	None			Change Drop More
14	Email	varchar(255)	utf8_general_ci		Yes	None			Change Drop More
15	حالة القيد	varchar(255)	utf8_general_ci		Yes	None			Change Drop More
16	ActivityID	int(11)			Yes	None			Change Drop More
17	MilitaryStatusID	int(11)			Yes	None			Change Drop More
18	AdvisorID	int(11)			Yes	None			Change Drop More
19	SocialNumber	varchar(255)	utf8_general_ci		Yes	None			Change Drop More
20	MajorDepartmentID	int(11)			Yes	None			Change Drop More
21	MinorDepartmentID	int(11)			Yes	None			Change Drop More
22	StatusID	int(11)			Yes	None			Change Drop More
23	MilitaryEducationID	int(11)			Yes	None			Change Drop More
24	CurriculumID	int(11)			Yes	None			Change Drop More

4.3.12 Users

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
1	Username	varchar(50)	utf8_general_ci		No	None			Change Drop More
2	Email	varchar(250)	utf8_general_ci		No	None			Change Drop More
3	password	varchar(50)	utf8_general_ci		No	None			Change Drop More

4.3.13 Tables Relationship



4.4 Design with coding

Here we will show the design of all pages for every user in EMS and we also will see the implementation of each page as we show in the previous chapter (flow charts).

4.4.1 Login form

Here we design a simple form for login which manage project users to see the EMS operations.

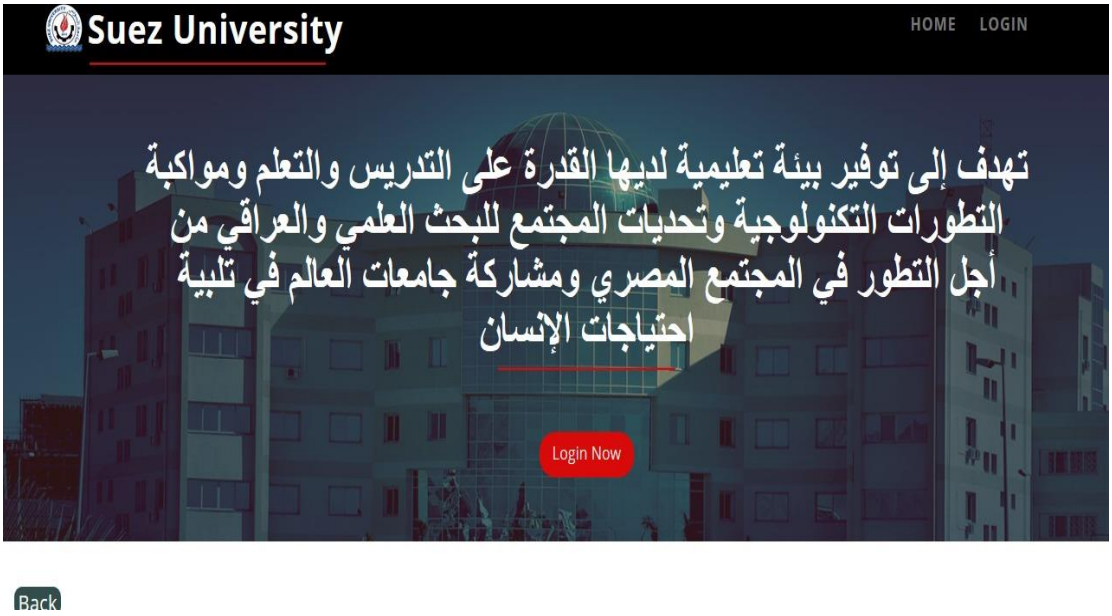


Figure 4.1

4.4.1 Implementation

```
<body ng-app="BloodBankMod" ng-controller="homeCtrl">

    <div class="mynav navbar navbar-inverse navbar-fixed-top " id="menu">
        <div class="container">
            <div class="navbar-header">
                <button type="button" class="navbar-toggle" data-toggle="collapse" data-target=".navbar-collapse">
                    <span class="icon-bar"></span>
                    <span class="icon-bar"></span>
                    <span class="icon-bar"></span>
                </button>

                <a class="navbar-brand" href="#">
                    <img class="logo-custom" src="assets/img/suez.png" alt=Suez University Drop Logo" />
                </a>
                <h1>Suez  University</h1>
            </div>
            <div class="navbar-collapse collapse move-me">
                <ul class="nav navbar-nav navbar-right">
                 <li>
                     <a href="#home">HOME</a>
                  </li>

                  <li>
                    <a href="log.html ">LOGIN</a>
                  </li>

                </ul>
            </div>

        </div>
    </div>
```

```
<div class="home-sec" id="home">

    <div class="overlay">

        <div class="container">

            <div class="row text-center ">

                <div class="col-lg-12  col-md-12 col-sm-12">

                    <div class="flexslider set-flexi" id="main-section">

                        <ul class="slides move-me">

                            <!-- Slider 01 -->
                            <li>
                                <h3>اهلا بكم في
جامعة السويس</h3>

                                <h1>السويس مؤسسة تعليمية علمية رائدة ضمن مجموعة الجامعات المصرية العامة والخاصة</h1>

                            </li>
                            <!-- End Slider 01 -->

                            <!-- Slider 02 -->
                            <li>


                                <h1>أجل التطور في المجتمع المصري ومشاركة جامعات العالم في تلبية احتياجات الإنسان</h1>

                                <a href="log.html" class="btn btn-lg" id="donateBtn">
                                    Login Now
                                </a>
```

```
<div id="about-sec" class="container set-pad">
    <div class="row text-center">
        <div class="col-lg-8 col-lg-offset-2 col-md-8 col-sm-8 col-md-offset-2 col-sm-offset-2">
            <h1 data-scroll-reveal="enter from the bottom after 0.2s" class="header-line">About Us</h1>
            <p data-scroll-reveal="enter from the bottom after 0.3s">
                كليات وتضم الجامعة أعتبـــاراً من العام الجامعي 2016/2017 اثنى عشر كلية وبإجمالي عدد الطلاب 12354
            </p>
        </div>

    </div>
    <!--/.HEADER LINE END-->


</div>
<!-- ABOUT SECTION END-->

<div id="articleWrapper">
    <div id="article-sec">
        <div class="container set-pad">
            <div class="row text-center">

                <div class="col-lg-8 col-lg-offset-2 col-md-8 col-sm-8 col-md-offset-2 col-sm-offset-2">
                    <h1 data-scroll-reveal="enter from the top after 0.1s" class="header-line">ARTICLES </h1>
                </div>

            </div>
            <!--/.HEADER LINE END-->

            <div class="row">

```

```
<div class="row">

    <div class="col-lg-4  col-md-4 col-sm-4" data-scroll-reveal="enter from the bottom after 0.5s">
        <div class="article-div" ng-click="showArticle1=!showArticle1">
            <img src="assets/img/ashSuezu.jpeg" class="img-rounded" />
            <h3>رؤية الجامعة</h3>
            <hr />
            <p>
                رؤية الجامعة تتمثل في مواجهة التحديات و التغيرات الكونيه ومواكبة التكنولوجيا
            </p>
        </div>
    </div>

    <div class="col-lg-4  col-md-4 col-sm-4" data-scroll-reveal="enter from the top after 0.5s">
        <div class="article-div" ng-click="showArticle2=!showArticle2">
            <img src="assets/img/scifac-01.jpeg" class="img-rounded" />
            <h3>رسالة الجامعة</h3>
            <hr />
            <p>
                تعتبرجامعة السويس مؤسسه اكاديميه لها هدف محورى لتنمية الفكر و الارتقاء بالفرد

            </p>
        </div>
    </div>

    <div class="col-lg-4  col-md-4 col-sm-4" data-scroll-reveal="enter from the bottom after 0.5s">
        <div class="article-div" ng-click="showArticle3=!showArticle3">
            <img src="assets/img/DSC_47ss69.jpeg" class="img-rounded" />
            <h3>أهداف الجامعة</h3>
            <hr />
            <p>
                اعداد الكوادر البشرية المتخصصة في مجال العلوم و المعرفة الحديثة
```

```
<div class="containerDiv" ng-show="showArticle1" ng-click="showArticle1 =!showArticle1">
    <!--containerDiv1-->
    <div class="articleContainerDiv">
        <!--HeaderDiv-->
        <div class="headerDiv">
            <!--imgDiv-->
            <div class="imgDiv">
                <img class="img-thumbnail" src="assets/img/ashSuezu.jpeg">
            </div>
            <h1>رؤية الجامعة </h1>


            <hr/>
        </div>

        <!--articleDiv-->
        <div class="content">

            <ul>
                <li>والتطلع لأن تكون منارة للعلم والبحث العلمى و المشاركة المجتمعية فى مصر و العالم</li>
            </ul>


        </div>
    </div>
</div>
<!-- ARTICLE1 PANE END-->

<div class="containerDiv" ng-show="showArticle2" ng-click="showArticle2 =!showArticle2">
    <!--containerDiv1-->
    <div class="articleContainerDiv">
        <!--HeaderDiv-->
```

```
<div class="row set-row-pad">
    <div class="col-lg-6 col-md-6 col-sm-6 " data-scroll-reveal="enter from the left after 0.4s">
        <img src="assets/img/ashSuezu.jpeg" class="img-thumbnail" />
    </div>

    <div class="col-lg-4 col-md-4 col-sm-4 col-lg-offset-1 col-md-offset-1 col-sm-offset-1">
        <div class="panel-group" id="accordion">
            <div class="panel panel-default" data-scroll-reveal="enter from the right after 0.5s">
                <div class="panel-heading">
                    <h4 class="panel-title">
                        <a data-toggle="collapse" data-parent="#accordion" href="#collapse1" class="collapsed">

                        </a>
                    </h4>
                </div>

                <div id="collapse1" class="panel-collapse collapse" style="height: 0px;">
                    <div class="panel-body">

                    </div>
                </div>
            </div>

            <div class="panel panel-default" data-scroll-reveal="enter from the right after 0.7s">
                <div class="panel-heading">
                    <h4 class="panel-title">
                        <a data-toggle="collapse" data-parent="#accordion" href="#collapse2" class="collapsed">

                        </a>
                    </h4>
                </div>
```

4.4.2 Index pages implementation

```
<div >
<form action="index.php" method="post" style="margin-top:100px">
<center id="selector">
<label>
faculty <select>
<option> fci </option>
</select>
<span> Level </span><select name="level" id="level" >
</div>
<?php

$DB=mysqli_connect("localhost","root","","gpa") or die("error:".mysqli_error());



$getlevel=mysqli_query($DB,"SELECT level.ID,level.LevelNumber
FROM level ");



while($row = mysqli_fetch_row( $getlevel ))
{

    $level=$row['1'];
    echo " <option> ".$row['1']." </option>";


}

        echo" </select>";
        echo"<span>Semester</span> <select name='semester' id='semester'>";
```

```
$getsemester=mysqli_query($DB, "SELECT semester.ID,semester.Semester
FROM semester ");



while($row = mysqli_fetch_row( $getsemester ))
{

$semesterID=$row['1'];

    echo " <option value=".$row['0']."> ".$row['1']." </option>";


}


?>
 </select>
<input type="submit"value="Go" class="sh">
</center>

 </label>
</form>

    <div id="db">
    <?php



$DB=mysqli_connect("localhost","root","","gpa") or die("error:".mysqli_error());
mysqli_query($DB,"SET NAMES utf8");

$levelID='';
 $semesterID='';
```

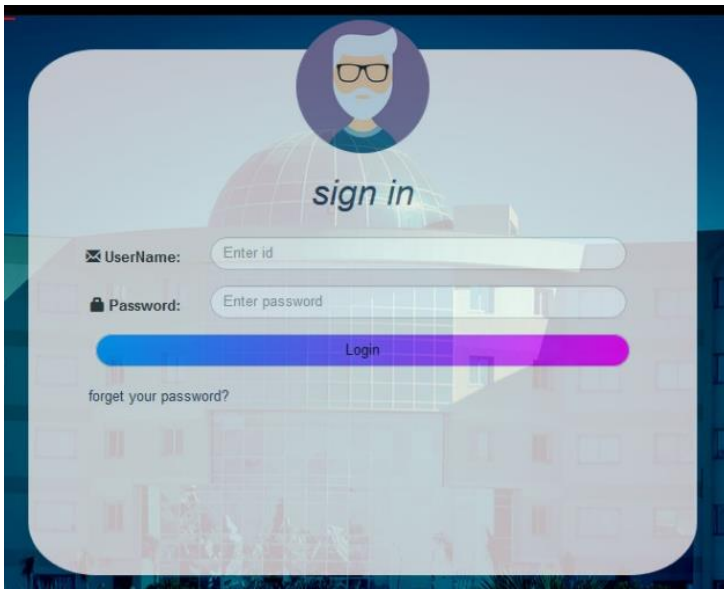
```
$levelID='';
 $semesterID='';
if(isset($_POST['level'])){
$levelID=$_POST['level'];
$semesterID=$_POST['semester'];

$getsemester=mysqli_query($DB,"SELECT semester.Semester
FROM semester where ID = '$semesterID'  ");

while($row1 = mysqli_fetch_row( $getsemester ))
{
 $seme=$row1['0'];
}

echo"<center style='font-size:25px;color:white;'> <bold style='color:red;weight:30px;background-color:white;'>Level

    }

$getcourse=mysqli_query($DB,"CALL sp_course_semester('$semesterID','$levelID')");

$num_r=mysqli_num_rows($getcourse);
if($num_r>0)
{
    echo"
<center>
<div class='container'>
<input type='text' id='myInput' onkeyup='myFunction()' placeholder='Search'>
<table border=5px id='myTable'>
```

```
while($row=mysqli_fetch_row($getcourse))
{
    $idd=$row['0'];

    echo "<tr id='search' class='table-hover'> <td> <a href='marks.php?idd=$idd & sem=$semesterID & lvl=$levelID'
    echo $row['1']."</a></td><td><a href='marks.php?idd=$idd & sem=$semesterID & lvl=$levelID' >";
    echo $row['2']."</a></td></tr>";









    }
    echo"<center><a href='db1.php?semester=$semesterID & level=$levelID' ><button type='button'
    style='background-color:red;width:70px;font-size:18 ;color:rgb(255,255,255) ;border-radius:10px;
    height:30px'> <b>Edit</b></button></center><br>";

 echo"</table>";

 echo " <center><a href='absence.php?idd=$idd & sem=$semesterID & lvl=$levelID' ><button type='button'
    style='background-color:#b2568a;width:70%;font-size:17 ;color:rgb(255,255,255) ;border-radius:10px;
    height:30px'> <b>Absence</b></button></center> " ;

 }else{echo"<center><h1></h1>";
        /*echo"<mark><span>There is no Courses in<br> ".$seme." and level ".$levelID."</span>";*/
        echo"</center>";
 }
 $_SESSION["level"]='';
$_SESSION["semester"]='';
if(isset($_POST['level'])){
 $_SESSION["level"] = $_POST['level'];
```

4.4.3 Excel JS

```
function exportTableToExcel(tableID, filename = ''){
    var downloadLink;
    var dataType = 'application/vnd.ms-excel';
    var tableSelect = document.getElementById(tableID);
    var tableHTML = tableSelect.outerHTML.replace(/ /g, '%20');

    // Specify file name
    filename = filename?filename+'.xls':'excel_data.xls';

    // Create download link element
    downloadLink = document.createElement("a");

    document.body.appendChild(downloadLink);

    if(navigator.msSaveOrOpenBlob){
        var blob = new Blob(['\ufeff', tableHTML], {
            type: dataType
        });
        navigator.msSaveOrOpenBlob( blob, filename);
    }else{
        // Create a link to the file
        downloadLink.href = 'data:' + dataType + ', ' + tableHTML;

        // Setting the file name
        downloadLink.download = filename;

        //triggering the function
        downloadLink.click();
    }
}
```

4.4.4 Login form



4.4.4 Implementation in HTML

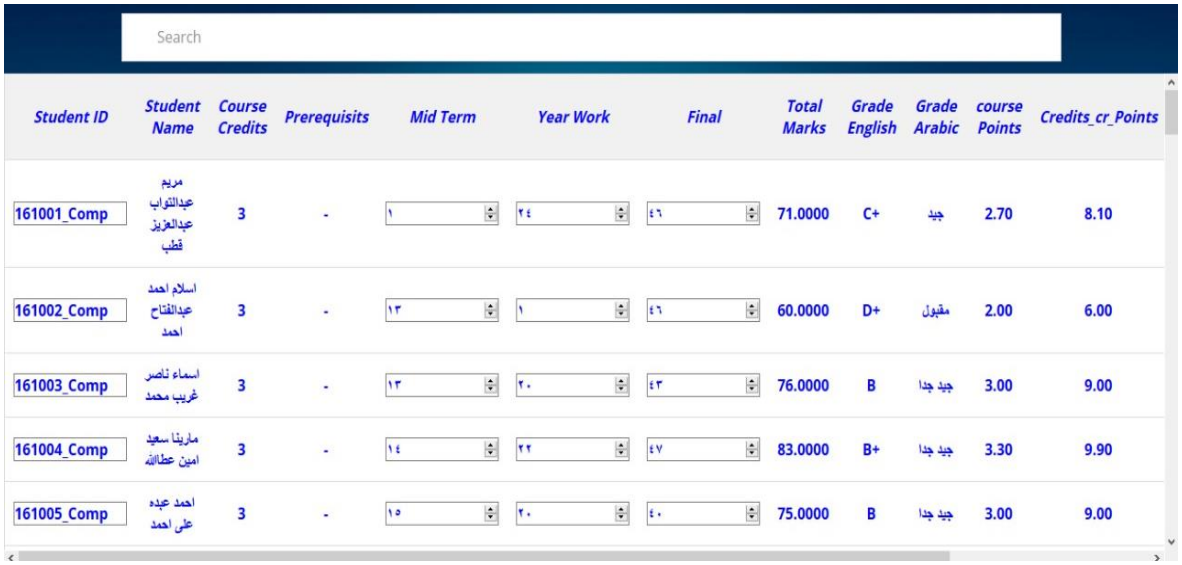
```
<div class="container" id="login">
    <form class="form-horizontal" method="post" action="log.php">
        <div class="form-group">
            <div><img src="assets/img/man.png"alt="loginpho" id="logo"></div>
            <div class="header"><h1>sign in</h1></div>
        </div>
        <div class="form-group">
        <label class="control-label col-sm-3" for="email"><span class="glyphicon glyphicon-envelope" style="margin: 5
        <div class="col-sm-8">
          <input type="text" class="form-control" id="email" placeholder="Enter id" name="id" required autocomplete="
        </div>
        </div>
```

```
            <div class="form-group">
              <label class="control-label col-sm-3" for="pwd"><i class="glyphicon glyphicon-lock" style="margin: 5px"></i>Pa
              <div class="col-sm-8">
                <input type="password" class="form-control" id="pwd" placeholder="Enter password" name="pwd" required autoc
              </div>
            </div>
            <div class="form-group">
              <div class=" col-sm-14">
                <button type="submit" class="btn btn-default" name="submit">Login</button><br><br>
              </div>
              <a href="#">forget your password?</a>
            </div>

          </form>
        </div>
</div>
<div><a id='mina' href="javascript:history.go(-1)"onMouseOver="self.status.referrer;return true">Back</a></div>
```

4.4.4 Implementation in PHP

```
<?php
session_start();
$mysql_hostname="localhost";
$mysql_username="pola";
$mysql_password="1234";
$mysql_dbname="gpa";
$connect=mysqli_connect($mysql_hostname, $mysql_username, $mysql_password, $mysql_dbname) or die(mysql_error());
$id=($_POST['id']);
$password=($_POST['pwd']);
$_SESSION['id']=$id;
$_session['pwd']=$password;
$get_users=mysqli_query($connect,"select * from student where (StudentEductionalNumber1='$id' )and socialnumber='$passwor
 or die(mysqli_error());
$pass=mysqli_query($connect,"select * from users where (Username='$id' or Email='$id')and password='$password'") or die(m
echo mysqli_num_rows($get_users);
echo mysqli_num_rows($pass);
while($row=mysqli_fetch_assoc($get_users)){
    $db_user=$row['id'];
    $db_pass=$row['pwd'];
}
while($row=mysqli_fetch_assoc($pass)){
    $db_user=$row['id'];
    $db_pass=$row['pwd'];
}
if(mysqli_num_rows($get_users)==1){
     header("Location:marks1.php");
     $_session['id']=$id;
    exit();
}
else if(mysqli_num_rows($pass)==1){
    header("Location:index.php");
    exit();
}
```

4.4.5 Marks edit page



4.4.5 Implementation

```
$getinfo=mysqli_query($DB,"SELECT
        `student`.`StudentEductionalNumber` AS `StudentCode`,
        `student`.`StudentName` AS `StudentName`,
        `course`.`Credits` AS `CourseCredits`,
        `course`.`Prerequisits` AS `Prerequisits`,
        `registration`.`MidTermExam`,`registration`.`YearWork`,`registration`.`FinalExam`,
        IF((IFNULL(`registration`.`FinalExam`, 0) < 0),
            IFNULL(`registration`.`FinalExam`, 0),
            ((IFNULL(`registration`.`FinalExam`, 0) + IFNULL(`registration`.`YearWork`, 0)) + IFNULL(`registration`.
        IF((`registration`.`FinalExam` = -(300)),
            (SELECT
                    `course_grade`.`Grade_English`
                FROM
                    `course_grade`
                WHERE
                    ((IFNULL(`student`.`CurriculumID`, 0) = `course_grade`.`CurriculumID`)
                        AND (`course_grade`.`OrderCode` = 54))
                LIMIT 1),
            IF((`registration`.`FinalExam` = -(200)),
                (SELECT
                        `course_grade`.`Grade_English`
                    FROM
                        `course_grade`
                    WHERE
                        ((IFNULL(`student`.`CurriculumID`, 0) = `course_grade`.`CurriculumID`)
                            AND (`course_grade`.`OrderCode` = 53))
                    LIMIT 1),
                IF((`registration`.`FinalExam` = -(100)),
                    (SELECT
                            `course_grade`.`Grade_English`
                        FROM
                            `course_grade`
```

```
  WHERE
      ((IFNULL(`student`.`CurriculumID`, 0) = `course_grade`.`CurriculumID`)
          AND (`course_grade`.`OrderCode` = 52))
  LIMIT 1),
((((IFNULL(`registration`.`FinalExam`, 0) + IFNULL(`registration`.`YearWork`, 0)) + IFNULL(`registration`.`MidTermExam`
          `curriculum`.`Limit_Fail_Total`
      FROM
          `curriculum`
      WHERE
          (IFNULL(`student`.`CurriculumID`, 0) = `curriculum`.`ID`)
      LIMIT 1))),
  (SELECT
          `course_grade`.`Grade_English`
      FROM
          `course_grade`
      WHERE
          ((IFNULL(`student`.`CurriculumID`, 0) = `course_grade`.`CurriculumID`)
              AND (`course_grade`.`OrderCode` = 50))
      LIMIT 1),
  IF((IFNULL(`registration`.`FinalExam`, 0) < (`course`.`FinalExam` * (SELECT
              `curriculum`.`Limit_Fail_Final`
          FROM
              `curriculum`
          WHERE
              (IFNULL(`student`.`CurriculumID`, 0) = `curriculum`.`ID`)
          LIMIT 1))),
      (SELECT
              `course_grade`.`Grade_English`
          FROM
              `course_grade`
          WHERE
              ((IFNULL(`student`.`CurriculumID`, 0) = `course_grade`.`CurriculumID`)
```

```
    (SELECT
        `course_grade`.`Grade_English`
    FROM
        `course_grade`
    WHERE
        ((IFNULL(`student`.`CurriculumID`, 0) = `course_grade`.`CurriculumID`)
            AND (`course_grade`.`Percentage` <= ((IFNULL(`registration`.`FinalExam`, 0) + IFNULL(`registration`.
    LIMIT 1))))) AS `Grade_English`,
ion`.`FinalExam` = -(300)),

urse_grade`.`Grade_Arabic`

urse_grade`

FNULL(`student`.`CurriculumID`, 0) = `course_grade`.`CurriculumID`)
  AND (`course_grade`.`OrderCode` = 54))
),
ration`.`FinalExam` = -(200)),
  `course_grade`.`Grade_Arabic`
M
  `course_grade`
RE
  ((IFNULL(`student`.`CurriculumID`, 0) = `course_grade`.`CurriculumID`)
      AND (`course_grade`.`OrderCode` = 53))
IT 1),
egistration`.`FinalExam` = -(100)),
LECT
    `course_grade`.`Grade_Arabic`
  FROM
    `course_grade`
  WHERE
```

```
        AND (`course_grade`.`OrderCode` = 52))
    LIMIT 1),
((((IFNULL(`registration`.`FinalExam`, 0) + IFNULL(`registration`.`YearWork`, 0)) + IFNULL(`registration`.`MidTermExam`, 0)
        `curriculum`.`Limit_Fail_Total`
    FROM
        `curriculum`
    WHERE
        (IFNULL(`student`.`CurriculumID`, 0) = `curriculum`.`ID`)
    LIMIT 1))),
(SELECT
        `course_grade`.`Grade_Arabic`
    FROM
        `course_grade`
    WHERE
        ((IFNULL(`student`.`CurriculumID`, 0) = `course_grade`.`CurriculumID`)
            AND (`course_grade`.`OrderCode` = 50))
    LIMIT 1),
IF((IFNULL(`registration`.`FinalExam`, 0) < (`course`.`FinalExam` * (SELECT
            `curriculum`.`Limit_Fail_Final`
        FROM
            `curriculum`
        WHERE
            (IFNULL(`student`.`CurriculumID`, 0) = `curriculum`.`ID`)
        LIMIT 1))),
    (SELECT
            `course_grade`.`Grade_Arabic`
        FROM
            `course_grade`
        WHERE
            ((IFNULL(`student`.`CurriculumID`, 0) = `course_grade`.`CurriculumID`)
                AND (`course_grade`.`OrderCode` = 51))
        LIMIT 1),
    (SELECT
```

```
    (SELECT
        `course_grade`.`Grade_Arabic`
    FROM
        `course_grade`
    WHERE
        ((IFNULL(`student`.`CurriculumID`, 0) = `course_grade`.`CurriculumID`)
            AND (`course_grade`.`Percentage` <= ((IFNULL(`registration`.`FinalExam`, 0) + IFNULL(`registratic
    LIMIT 1)))))) AS `Grade_Arabic`,
gistration`.`FinalExam`, 0) < (`course`.`FinalExam` * (SELECT
urriculum`.`Limit_Fail_Final`

urriculum`

FNULL(`student`.`CurriculumID`, 0) = `curriculum`.`ID`)
1))),

LL(`registration`.`FinalExam`, 0) + IFNULL(`registration`.`YearWork`, 0)) + IFNULL(`registration`.`MidTermExam`, 0)

(SELECT
        `course_grade`.`Points`
    FROM
        `course_grade`
    WHERE
        ((IFNULL(`student`.`CurriculumID`, 0) = `course_grade`.`CurriculumID`)
            AND (`course_grade`.`Percentage` <= ((IFNULL(`registration`.`FinalExam`, 0) + IFNULL(`registration`.`
    LIMIT 1),
  2))) AS `Course_Points`,

  `course_grade`.`Points`
OM
  `course_grade`
ERE
  ((IFNULL(`student`.`CurriculumID`, 0) = `course_grade`.`CurriculumID`)
```

```
  ((IFNULL(`student`.`CurriculumID`, 0) = `course_grade`.`CurriculumID`)
     AND (`course_grade`.`Percentage` <= ((IFNULL(`registration`.`FinalExam`, 0) + IFNULL(`registration`.`YearWork`, 0))
MIT 1) * `course`.`Credits`),
`Course_cr_points`,
egistration`.`FinalExam`, 0) < (`course`.`FinalExam` * (SELECT
urriculum`.`Limit_Fail_Final`

urriculum`

FNULL(`student`.`CurriculumID`, 0) = `curriculum`.`ID`)
1)))
IFNULL(`registration`.`FinalExam`, 0) + IFNULL(`registration`.`YearWork`, 0)) + IFNULL(`registration`.`MidTermExam`, 0)) <
urriculum`.`Limit_Fail_Total`

urriculum`

FNULL(`student`.`CurriculumID`, 0) = `curriculum`.`ID`)
1)))),

urse`.`Credits`, 0)) AS `CourseCredits_Completed`,
orDepartmentID` AS `DepartmentID`


r`

semester`
tion` ON ((`course_semester`.`ID` = `registration`.`Course_semesterID`))) ON ((`course`.`ID` = `course_semester`.`CourseID

`.`statusID` <> 2) and  course_semester.semesterID='$semesterID'  and course.Code='$id' and course.LevelID='$levelID'
.`StudentEductionalNumber`");

ws($getinfo);
h row($getinfo))
```

```
ws($getinfo);
h_row($getinfo))


h'>";
row['0']."<input  type=hidden  name='idd[]' value='".$row['0']."'></td>";
ow['1']."</td>";
$row['2']."</td>";
$row['3']."</td>";
ut type=number min='0' max='15' name='mid[]' value='".$row['4']."'></td>";
ut type=number min='0' max='25' name='year[]' value='".$row['5']."'></td>";
ut type=number min='0' max='60' name='final[]' value='".$row['6']."'></td><td>";
."</td><td>";
."</td><td>";
."</td><td>";
]."</td>";
ow['11']."</td>";
ow['12']."</td>";
ow['13']."</td>";


ype=hidden name='mid1[]' value='".$row['4']."'>";
ype=hidden name='year1[]' value='".$row['5']."'>";
ype=hidden name='final1[]' value='".$row['6']."'>";









en name='lvl' value='".$levelID."'>";
en name='code' value='".$id."'>";
en name='sem' value='".$semesterID."'>";
```

4.4.6 Results (one course & final)

4.4.6.1 One course

Student ID	Student Name	Course Credits	Prerequisits	Mid Term	Year Work	Final	Total Marks	Grade English	Grade Arabic	course Points	Credits_cr_Points	Course Credits Completed	Departr ID
161001_Comp	مريم عبدالتواب عبدالعزيز قطب	3	-	1.0000	24.0000	46.0000	71.0000	C+	جيد	2.70	8.10	3	1
161002_Comp	اسلام احمد عبدالفتاح احمد	3	-	13.0000	1.0000	46.0000	60.0000	D+	مقبول	2.00	6.00	3	3
161003_Comp	اسماء ناصر غريب محمد	3	-	13.0000	20.0000	43.0000	76.0000	B	جيد جدا	3.00	9.00	3	3
161004_Comp	مارينا سعيد امين عطاالله	3	-	14.0000	22.0000	47.0000	83.0000	B+	جيد جدا	3.30	9.90	3	3
[illegible]_Comp	احمد عبده على احمد	3	-	15.0000	20.0000	40.0000	75.0000	B	جيد جدا	3.00	9.00	3	

4.4.6.2 Final

Subjects / Number of Credits per Subject		CS Fundamentals CS121 — 3 Credits				English Language HUM111 — 3 Credits				
StudentEductionalNumber	StudentName	CourseMarks	CourseEnglish	CoursePoints	Cr_points	CourseMarks	CourseEnglish	CoursePoints	Cr_points	Co
161017_Comp مستجد	ابراهيم مطاوع ابراهيم نصرالدين		-	-		77.0000	B	3.00	9.0	
161051_Comp باق	عبدالرحمن مصطفى السيد أحمد النجار		-	-		63.0000	D+	2.00	6.0	
161001_Comp مستجد	مريم عبدالتواب عبدالعزيز قطب	71.0000	C+	2.70	8.1	82.0000	B+	3.30	9.9	
161002_Comp مستجد	اسلام احمد عبدالفتاح احمد	60.0000	D+	2.00	6.0	78.0000	B	3.00	9.0	

4.4.6 Implementation

```
<?php

$DB=mysqli_connect("localhost","root","","gpa") or die("error:".mysqli_error());
mysqli_query($DB,"SET NAMES utf8");
    echo" </select>";
    echo"Curriculum <select name='curriculum' id='curriculum'>";

$getcurriculum=mysqli_query($DB,"SELECT curriculum.ID,curriculum.curriculumEngName
FROM curriculum ");



while($row = mysqli_fetch_row( $getcurriculum ))
{

$curriculumID=$row['1'];

  echo " <option value=".$row['0']."> ".$row['1']." </option>";


}
echo" </select>";

//////////////////////////////////////////
  echo"Semester <select name='semester' id='semester'>";
$getsemester=mysqli_query($DB,"SELECT semester.ID,semester.Semester
FROM semester ");



while($row = mysqli_fetch_row( $getsemester ))
{
```

```
$semesterID=$row['1'];

    echo " <option value=".$row['0']."> ".$row['1']." </option>";


}
echo" </select>";

//////////////////////////////////////////////
  echo"level <select name='level' id='level'>";


$getlevel=mysqli_query($DB,"SELECT level.ID,level.levelTxt
FROM level ");


echo"<option value='null'>null</option>";

while($row = mysqli_fetch_row( $getlevel ))
{

$levelID=$row['1'];

    echo " <option value=".$row['0']."> ".$row['1']." </option>";


}

echo" </select>";

echo"Department <select name='Department' id='Department'>";
```

```
$getDepartment=mysqli_query($DB,"SELECT Department.ID,DeptartmentCode
FROM Department ");

echo"<option value='null'>null</option>";
while($row = mysqli_fetch_row( $getDepartment ))
{

$departmentID=$row['1'];

    echo " <option value=".$row['0']."> ".$row['1']." </option>";

}


echo" </select>";

?>

<input type="submit"value="Go" name="go">
</center>

 </label>
</form>

<button onclick="myFunction()"><b>Print this page</button>

<button onclick="exportTableToExcel('myTable', 'Results-data')"><b>Export Result To Excel File</button>


<div class='table-wrapper-scroll-y my-custom-scrollbar'>
<table border=5px id='myTable' id='example'class='display' cellspacing='0' class='table table-bordered table-striped mb-0
```

```php
<?php
$DB=mysqli_connect("localhost","root","","gpa") or die("error:".mysqli_error());
mysqli_query($DB,"SET NAMES utf8");


$levelID='';
 $semesterID='';
$curriculumID='';
 $departmentID='';
if(isset($_POST['level'])){
$curriculumID=$_POST['curriculum'];
$semesterID=$_POST['semester'];
$levelID=$_POST['level'];
$departmentID=$_POST['Department'];
}



$getcourseth=mysqli_query($DB,"SELECT DISTINCT Course.course,Course.Credits,Course.Code
FROM student INNER JOIN (Course INNER JOIN (Course_semester INNER JOIN
(Registration INNER JOIN StudentLevel ON Registration.StudentID = StudentLevel.StudentID)  ON
Course_semester.ID = Registration.Course_semesterID) ON Course.ID = Course_semester.CourseID)
ON student.ID = Registration.StudentID
WHERE student.curriculumID=$curriculumID and Course_semester.SemesterID=$semesterID and if($departmentID is null,true
GROUP BY Course.Code, Course.ArabicName,
(Course.Midterm+Course.`Year Work Grades O`+Course.`Year Work Grades PE`+ Course.`YearWorkGrades G`+Course.FinalExam)
Course_semester.CourseID, Course_semester.SemesterID, Course.Course") ;

echo"
<center>
<input type='text' id='myInput' placeholder='Search'>
</center>";
 echo"
```

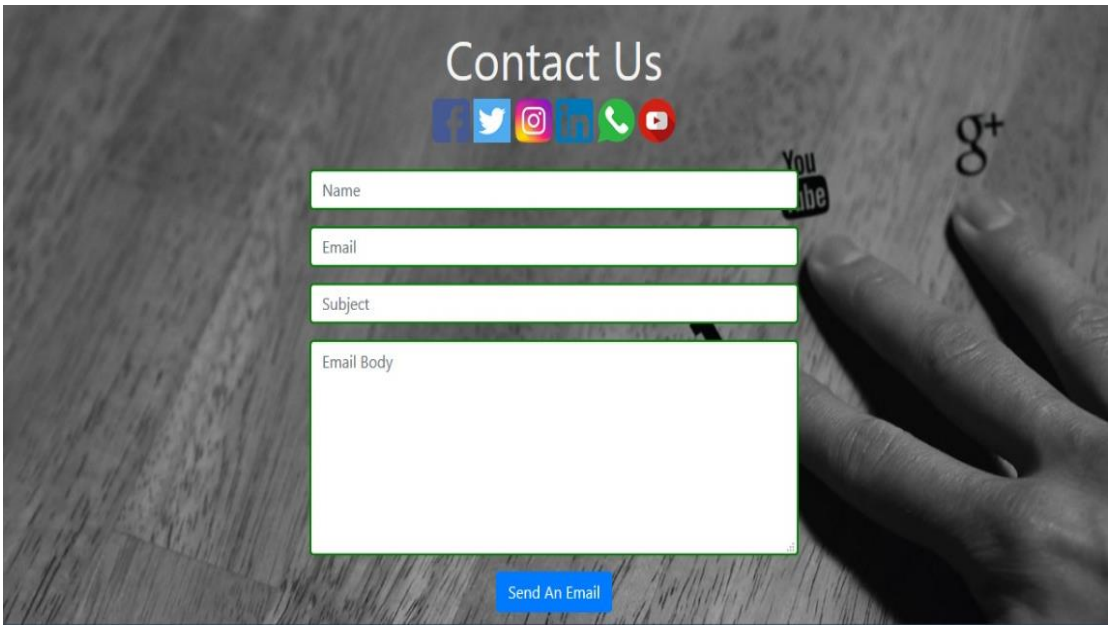
```
$getpoints=mysqli_query($DB,"CALL sp_cursor_semester($curriculumID,$semesterID,$levelID,$departmentID)");


$count8=$countAll+3;
while($row=mysqli_fetch_row($getpoints)){

  echo"
  <tr id='search'><td>".$row['0']."</td>
  <td>".$row['2']."</td>";
  $n=3;
  $ncolor=3;
  $countAll=$count*4;
    z:
  if($countAll!=0)
  {
    $fcolor=$row[$n];

    if($fcolor<50 & $fcolor>12 or $fcolor=='F' or $fcolor=='-' or $fcolor=='ل ر'){echo"<td style=background-color:gray
      else{echo"<td>".$row[$n]."</td>";}
  $n++;
  $countAll--;

  goto z;
}
echo"
  <td>".$row[$count8]."</td>
  <td>".$row[$count8+1]."</td>
  <td>".$row[$count8+2]."</td>
  <td>".$row[$count8+3]."</td>
  <td>".$row[$count8+4]."</td>
  <td>".$row[$count8+5]."</td>
  <td>".$row[$count8+6]."</td>
```

```
$_SESSION['curriculum']='';
$_SESSION['semester']='';
$_SESSION['level']='';
$_SESSION['Department']='';
if(isset($_POST['go'])){
$_SESSION['curriculum']=$_POST['curriculum'];
$_SESSION['semester']=$_POST['semester'];
$_SESSION['level']=$_POST['level'];
$_SESSION['Department']=$_POST['Department'];}

echo"</table></div>";
?>
<script>
var country = '<?php echo $_SESSION["curriculum"]; ?>';
var country1 = '<?php echo $_SESSION["semester"]; ?>';
var country2='<?php echo $_SESSION["level"];?>'
var country3='<?php echo $_SESSION["Department"];?>'
if(country !== '' || country1!=='' || country2!=='' || country3!==''){
    document.getElementById("curriculum").value = country;
    document.getElementById("semester").value = country1;
    document.getElementById("level").value = country2;
    document.getElementById("Department").value = country3;

}
</script>
 <script>
        function scrollFunction() {
      if (document.body.scrollTop > 20 || document.documentElement.scrollTop > 20) {
        document.getElementById("myBtn").style.display = "block";
      } else {
        document.getElementById("myBtn").style.display = "none";
      }
    }
```

4.4.7 Save

```php
<?php

$DB=mysqli_connect("localhost","root","","gpa") or die("error:".mysqli_error());

$sql="update course,doctor,semester,course_semester set
  doctor.NameTxt = '$_POST[nametxt]'
WHERE Course.ID = Course_semester.CourseID and doctor.ID = Course_semester.DoctorID and course.code='$_POST[code]'";



if(mysqli_query($DB,$sql))

      header("location:db1.php?semester=$_POST[semesterID] & level=$_POST[level]");
else
      echo "NOT UPDATE";



?>
```

4.4.8 Sending email



4.4.8 Implementation

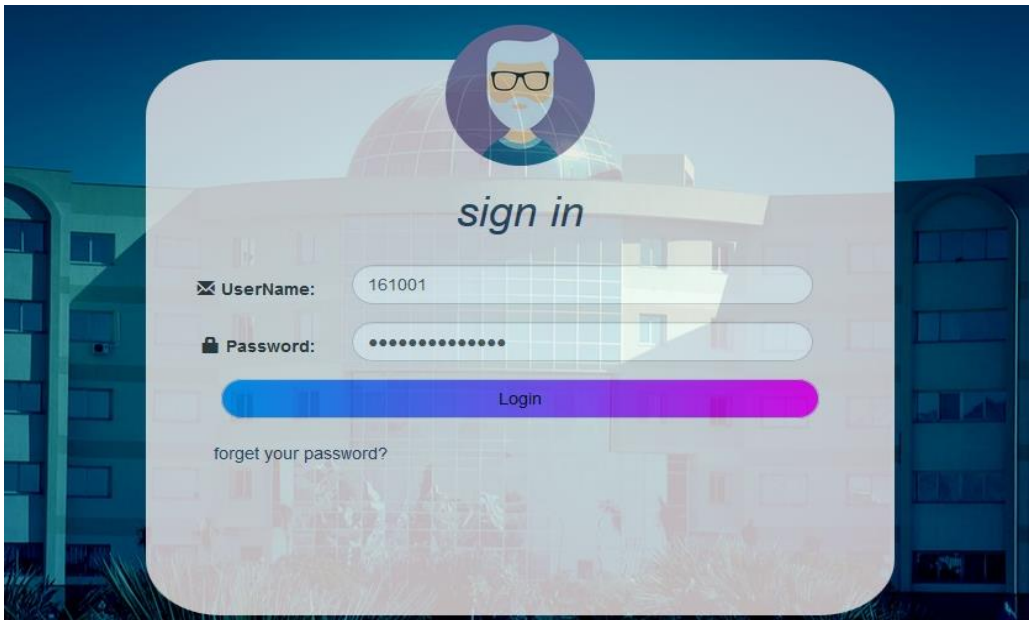
```php
<?php
    use PHPMailer\PHPMailer\PHPMailer;

    if (isset($_POST['name']) && isset($_POST['email'])) {
        $name = $_POST['name'];
        $email = $_POST['email'];
        $subject = $_POST['subject'];
        $body = $_POST['body'];

        require_once "PHPMailer/PHPMailer.php";
        require_once "PHPMailer/SMTP.php";
        require_once "PHPMailer/Exception.php";

        $mail = new PHPMailer();

        //SMTP Settings
        $mail->isSMTP();
        $mail->Host = "smtp.gmail.com";
        $mail->SMTPAuth = true;
```

```
    $mail->Username = "minayesser@gmail.com";
    $mail->Password = '123456789Mm';
    $mail->Port = 587; //587
    $mail->SMTPSecure = "tls"; //tls

    //Email Settings
    $mail->isHTML(true);
    $mail->setFrom($email, $name);
    $mail->addAddress("polanasser35@gmail.com");
    $mail->Subject = $subject;
    $mail->Body = $body;

    if ($mail->send()) {
        $status = "success";
        $response = "Email is sent!";
```

```
    //Email Settings
    $mail->isHTML(true);
    $mail->setFrom($email, $name);
    $mail->addAddress("polanasser35@gmail.com");
    $mail->Subject = $subject;
    $mail->Body = $body;

    if ($mail->send()) {
        $status = "success";
        $response = "Email is sent!";
    } else {
        $status = "failed";
        $response = "Something is wrong: <br><br>" . $mail->ErrorInfo;
    }

    exit(json_encode(array("status" => $status, "response" => $response)));
}
```

4.4.9 Student (login & result)

4.4.9.1 Login



4.4.9.2 Result

Subjects Number of Credits per Subject		CS Fundamentals CS121 3 Credits				English Language HUM111 3 Credits			
StudentEductionalNumber	StudentName	CourseMarks	CourseEnglish	CoursePoints	Cr_points	CourseMarks	CourseEnglish	CoursePoints	Cr_points
161001_Comp مسجد	مريم عبدالتواب عبدالعزيز قطب	71.0000	C+	2.70	8.1	82.0000	B+	3.30	9.9

4.4.9 Implementation

```
<?php


$DB=mysqli_connect("localhost","root","","gpa") or die("error:".mysqli_error());
mysqli_query($DB,"SET NAMES utf8");
    echo" </select>";

    echo"Curriculum <select name='curriculum' id='curriculum'>";


$getcurriculum=mysqli_query($DB,"SELECT curriculum.ID,curriculum.curriculumEngName
FROM curriculum ");



while($row = mysqli_fetch_row( $getcurriculum ))
{

$curriculumID=$row['1'];

  echo " <option value=".$row['0']."> ".$row['1']." </option>";


}
echo" </select>";

//////////////////////////////////////////
  echo"Semester <select name='semester' id='semester'>";
$getsemester=mysqli_query($DB,"SELECT semester.ID,semester.Semester
FROM semester ");
```

```
    echo"Semester <select name='semester' id='semester'>";
$getsemester=mysqli_query($DB,"SELECT semester.ID,semester.Semester
FROM semester ");



while($row = mysqli_fetch_row( $getsemester ))
{

$semesterID=$row['1'];

    echo " <option value=".$row['0']."> ".$row['1']." </option>";


}
echo" </select>";

////////////////////////////////////////////////
  echo"level <select name='level' id='level'>";


$getlevel=mysqli_query($DB,"SELECT level.ID,level.levelTxt
FROM level ");


echo"<option value='null'>null</option>";

while($row = mysqli_fetch_row( $getlevel ))
{

$levelID=$row['1'];
```

```
while($row = mysqli_fetch_row( $getlevel ))
{

$levelID=$row['1'];

    echo " <option value=".$row['0']."> ".$row['1']." </option>";


}

echo" </select>";

echo"Department <select name='Department' id='Department'>";


$getDepartment=mysqli_query($DB,"SELECT Department.ID,DeptartmentCode
FROM Department ");

echo"<option value='null'>null</option>";
while($row = mysqli_fetch_row( $getDepartment ))
{

$departmentID=$row['1'];

    echo " <option value=".$row['0']."> ".$row['1']." </option>";

}


echo" </select>";
```

```
?>

<input type="submit"value="Gooo" name="go">


 </label>
</form>

<button onclick="myFunction()"><b>Print this page</button>

<button onclick="exportTableToExcel('myTable', 'Results-data')"><b>Export Result To Excel File</button>

<table border="5px" id='myTable'>


<?php
$DB=mysqli_connect("localhost","root","","gpa") or die("error:".mysqli_error());
mysqli_query($DB,"SET NAMES utf8");


$levelID='';
 $semesterID='';
$curriculumID='';
 $departmentID='';
if(isset($_POST['level'])){
$curriculumID=$_POST['curriculum'];
$semesterID=$_POST['semester'];
$levelID=$_POST['level'];
$departmentID=$_POST['Department'];
}
```

```
$getcourse=mysqli_query($DB,"SELECT DISTINCT Course.Code
FROM student INNER JOIN (Course INNER JOIN (Course_semester INNER JOIN
(Registration INNER JOIN StudentLevel ON Registration.StudentID = StudentLevel.StudentID)  ON
Course_semester.ID = Registration.Course_semesterID) ON Course.ID = Course_semester.CourseID)
ON student.ID = Registration.StudentID
WHERE student.curriculumID=$curriculumID and Course_semester.SemesterID=$semesterID and if($departmentID is null,true,s
GROUP BY Course.Code, Course.ArabicName,
(Course.Midterm+Course.`Year Work Grades O`+Course.`Year Work Grades PE`+ Course.`YearWorkGrades G`+Course.FinalExam),
Course_semester.CourseID, Course_semester.SemesterID, Course.Course") ;

echo"
<th>Total_CR_points</th>
<th>GPA_Semester</th>
<th>Total_points</th>
<th>SumOfCredits</th>
<th>CGPA_Total_points</th>
<th>CGPA_grade</th>
<th>CGPA_Points</th>
<th>CGPA_SumOfCredits</th>
";

$count=0;
echo"<tr><th>StudentEductionalNumber</th>
<th>StudentName</th>
";

while($row=mysqli_fetch_row($getcourse)){


  echo"
  <th>CourseMarks</th>
  <th>CourseEnglish</th>
```

```
echo $count;
echo "<br> All ".$countAll;

$getpoints=mysqli_query($DB,"call sp_cursor_semester_std($curriculumID,$semesterID,$levelID,$departmentID)");


$count8=$countAll+3;
while($row=mysqli_fetch_row($getpoints)){

  echo"
  <tr><td>".$row['0']."</td>
  <td>".$row['2']."</td>";
  $n=3;
  $ncolor=3;
  $countAll=$count*4;
    z:
  if($countAll!=0)
  {
    $fcolor=$row[$n];

    if($fcolor<50 & $fcolor>12 or $fcolor=='F' or $fcolor=='-' or $fcolor=='لا رد'){echo"<td style=background-color:gr
      else{echo"<td>".$row[$n]."</td>";}
  $n++;
  $countAll--;

  goto z;
}
echo"
  <td>".$row[$count8]."</td>
  <td>".$row[$count8+1]."</td>
  <td>".$row[$count8+2]."</td>
```

```
$_SESSION['curriculum']='';
$_SESSION['semester']='';
$_SESSION['level']='';
$_SESSION['Department']='';
if(isset($_POST['go'])){
$_SESSION['curriculum']=$_POST['curriculum'];
$_SESSION['semester']=$_POST['semester'];
$_SESSION['level']=$_POST['level'];
$_SESSION['Department']=$_POST['Department'];}

echo"</table>";
?>
<script>
var country = '<?php echo $_SESSION["curriculum"]; ?>';
var country1 = '<?php echo $_SESSION["semester"]; ?>';
var country2='<?php echo $_SESSION["level"];?>'
var country3='<?php echo $_SESSION["Department"];?>'
if(country !== '' || country1!=='' || country2!=='' || country3!==''){
    document.getElementById("curriculum").value = country;
    document.getElementById("semester").value = country1;
    document.getElementById("level").value = country2;
    document.getElementById("Department").value = country3;

}
</script>
<div id="footer">
  &copy 2019 controlsystem.com | All Rights Reserved | <a href="index1.php">Contact Us</a>
</div>
</body>
</html>
```

4.4.10 Top & Back buttons

4.4.10 Implementation

```
    function scrollFunction() {
  if (document.body.scrollTop > 20 || document.documentElement.scrollTop > 20) {
    document.getElementById("myBtn").style.display = "block";
  } else {
    document.getElementById("myBtn").style.display = "none";
  }
}

// When the user clicks on the button, scroll to the top of the document
function topFunction() {
  document.body.scrollTop = 0; // For Safari
  document.documentElement.scrollTop = 0; // For Chrome, Firefox, IE and Opera
}
```

4.4.11 Update

```php
<?php

$DB=mysqli_connect("localhost","root","","gpa") or die("error:".mysqli_error());



$mid1 = array();
                foreach($_POST['mid1'] as $val)
                {
                $mid1[] = (int) $val;
                }


$year1 = array();
                foreach($_POST['year1'] as $val)
                {
                $year1[] = (int) $val;
                }
$final1 = array();
                foreach($_POST['final1'] as $val)
                {
                $final1[] = (int) $val;
                }
$ids = array();
                foreach($_POST['idd'] as $val)
                {
                $ids[] = (int) $val;
                }
$mid = array();
                foreach($_POST['mid'] as $val)
                {
                $mid[] = (int) $val;
                }
$year = array();
                foreach($_POST['year'] as $val)
                {
                $year[] = (int) $val;
                }
$final = array();
                foreach($_POST['final'] as $val)
                {
                $final[] = (int) $val;
                }
```

```
$co=$_POST['code'];
$semID=$_POST['sem'];
$count=$_POST['lenth'];

$x=0;
while($x<$count)
{

    if (($mid[$x]>15 || $mid[$x]<0 || $year[$x]>25 || $year[$x]<0 || $final[$x]>60 || $final[$x]<0 )||($mid1[$x]==$mid[$x]
```

```
$x=0;
while($x<$count)
{

    if (($mid[$x]>15 || $mid[$x]<0 || $year[$x]>25 || $year[$x]<0 || $final[$x]>60 || $final[$x]<0 )||($mid1[$x]==$mid[$x
        {

            $x=$x+1;
        }
    else
        {

mysqli_query($DB,"update registration set registration.MidTermExam='$mid[$x]',registration.YearWork ='$year[$x]',registra
WHERE registration.Course_semesterID = (select course_semester.ID from course_semester WHERE course_semester.SemesterID =
and registration.StudentID=(select Student.ID from Student WHERE  Student.StudentEductionalNumber1='$ids[$x]')");

        $x=$x+1;
        }




}
    header("location:marksedit.php?idd=$_POST[code] & sem=$_POST[sem] & lvl=$_POST[lvl]");

?>
```

CHAPTER FIVE:

CONCLUSION AND REFERENCES

5.1 Conclusion

This project has helped to emphasis the necessity of an automated student result system in faculties, as it simplifies the process of compiling and presenting students results. Computerization of this process reduces the time, human involvement and errors that could occur while making use of the manual method. Most student information systems in use today are server-based, with the application residing on a central computer server and being accessed by client applications at various places within and even outside the faculty. The project utilizes an applications web and database server. The application is meant to ease the processing of students' results in faculties. The application was successfully developed, tested, and found to be working as expected. It is capable of storing and retrieving academic records with high speed and accuracy, and presenting useful information to its users. Its qualities are the reduction in the cost of processing students results (an example would be the cost of purchase of papers) reduction in the time spent in the computation of student's grades and the elimination of duplication of duplication of resources in terms of manpower and infrastructure. The system is flexible and runs in a web browser. It is reasonably secure, enforces data integrity from the use of a relational database management system, it also minimizes data redundancy and is user-friendly. With this application, the processing of students' results is automated, thereby reducing processing time and increasing accuracy. "Automation Minimizes Redundancy: The use of Relational Database Management System for storage of students' data will help in minimizing the redundancy in database in addition to maintain consistency, integrity and security of data."

5.2 References

5.2.1 Books

- Joseph Valacich, Joey George and Jeffrey A.Hoffer"essentials of systems analysis and design" fifth edition.
- object oriented software engineering timothy c. lethbridge and robert laganiere.
- ian sommerville software engineering 9th edition solution manual
- luke welling php and mysql web development
- ramez elmasri and shamkant b.navathe"fundamentals of database system" 6th

5.2.2 Websites

- www.w3school.com
- www.php.com
- www.tutorialspoint.com
- www.stackoverflow.com