Project Documentation

On

UNIVERSITY GRADING SYSTEM (NJALA UNIVERSITY)

Submitted By:
KADIATU KAMARA 56390
LOIS B. SOGBEH   55019
SUZAN KOROMA   56386

In partial fulfillment for the award of the degree of

B. SCIENCE DEGREE (HONS)

In

BUSINESS AND INFORMATION TECHNOLOGY
SCHOOL OF TECHNOLOGY
NJALA UNIVERSITY, NJALA CAMPUS

DEPARTMENT OF COMPUTER SCIENCE & INFORMATION TECHNOLOGY

SCHOOL OF TECHNOLOGY

NJALA UNIVERSITY, NJALA CAMPUS

CERTIFICATE

Certified that this is a compiled record of the project work titled

UNIVERSITY GRADING SYSTEM
(NJALA UNIVERSITY)

Done by

KADIATU KAMARA 56390
LOIS B. SOGBEH 55019
SUZAN KOROMA 56386

of VIII semester Business & Information Technology in the year 2022 in partial fulfillment of the requirements for the award of Degree of Bachelor of Science (Hons) in Business & Information Technology of Njala University, Njala Campus.

Mr. Michael Komba Tommy　　　　　　　　　　　*Alhaji Dr. Foday F.M Kallon*
Project Supervisor　　　　　　　　　　　　　　*Head of Department*

ACKNOWLEDGEMENT

TABLE OF CONTENTS

ABSTRACT

University grading system is a project which aims in developing a computerized system to maintain all the daily work of school, department with regards to student grades .This project has many features which are generally not available in normal grading systems like facility of student login and a facility of lecturers, department and school admin login .It also has a facility of admin login through which the admin can monitor the whole system .It also has facility of an online notes resource center where lecturers can upload information about course materials or seminars being held in our university or nearby universities and after proper verification from the concerned institution organizing the seminar. It has also a facility where student after logging in their accounts can see list of available learning materials and notes, view their grades sent by lecturers and students can request to the department with issues regarding missing grades in the online request form.

Students are also able to print their progress report and view grades by semester. The lecturer after logging into his/her account i.e. Lecturer admin account can add student grades and add student learning materials.

The Super Admin has total privilege over the system as he/she is able to create any type of user, manage and remove them from the system

The School and Department Admin also have their own level of user privilege as they are granted and restricted by the Super Admin.

This system is built entirely using the Object-Oriented Programming paradigm and its support the multi-user login functionality

Overall this project of ours is being developed to help the students as well as staff
of the university to maintain student grades in the best way possible and also reduce the human efforts.

CHAPTER 1

INTRODUCTION

This chapter gives an overview about the aims, objectives, background and operation environment of the system.

1.1 PROJECT AIMS AND OBJECTIVES

The project aims and objectives that will be achieved after completion of this project are discussed in this subchapter. The aims and objectives are as follows:

- Online student access to grades
- Request column for students to the department and lecturer with regards to the issue of missing grades.
- A separate column for digital grading system.
- Student login page where student can view grades submitted by lecturer.
- A lecturer login page where lecturer can add student grades and notes with regards to lectures.

1.2 BACKGROUND OF PROJECT

University Grading System is an application which refers to grading systems which are generally small or medium in size. It is used by universities to manage student grades using a computerized system where he/she can record student grades, facilitate the addition of new students, faculty members and other users etc.

Grades and student maintenance modules are also included in this system which would keep track of the students using the system and also a detailed description about the student grades. With this computerized system there will be no loss of student record or student grade records which generally happens when a non-computerized system is used.

In addition, a progress report module is also included in Grading System. If user's role is Super admin, the user is able to generate different kinds of functions like lists of students registered, list of sub-admins and lecturers, issue and return reports on system usage.

All these modules are able to help the different levels of admins to manage the system with more convenience and in a more efficient way as compared to previous grading systems which are not computerized.

1.5 OPERATION ENVIRONMENT

PROCESSOR	INTEL CORE PROCESSOR OR BETTER PERFORMANCE
OPERATING SYSTEM	WINDOWS WINDOWS7 – Current Windows 11, UBUNTU
MEMORY	2GB RAM OR MORE
HARD DISK SPACE	MINIMUM 3 GB FOR DATABASE USAGE FOR FUTURE
DATABASE	MY SQL

CHAPTER 2

SYSTEM ANALYSIS

In this chapter, we will discuss and analyze about the developing process of Grading System including software requirement specification (SRS) and comparison between existing and proposed system. The functional and non-functional requirements are included in SRS part to provide complete description and overview of system requirement before the developing process is carried out. Besides that, existing vs proposed provides a view of how the proposed system will be more efficient than the existing one.

2.1 SOFTWARE REQUIREMENT SPECIFICATION

2.1.1 GENERAL DESCRIPTION

PRODUCT DESCRIPTION:

University grading System is a computerized system which help users (Super Admin, School, department and lecturer admins) to manage the student grading daily activity in an electronic format. It reduces the risk of paper work such as file lost which leads to missing grades, file damaged and time consuming. It can help user to manage the processing of student data and grades more effectively and timesaving.

PROBLEM STATEMENT:

The problem occurred before having computerized system includes:

- File lost (Missing Grades)

  When computerized system is not implemented file is always lost because of human environment. Sometimes due to some human error there may be a loss of student grades.

- File damaged

When a computerized system is absent most grade sheets file is damaged and this also leads to loss of data – integrity which happens as a result of some human error like spilling of water by some member on file accidentally. Besides some natural disaster like flooding or fires may also damage the files.

- Difficult to search for student grades (records)
  When there is no computerized system there is always a difficulty in searching for a particular student records if the records are large in number.

- Space consuming
  After the number of records become large the space for physical storage of file and records also increases if no computerized system is implemented.

- Cost consuming
  As there is no computerized system the to add each record paper will be needed which will increase the cost for the management of grading system.

2.1.2 SYSTEM OBJECTIVES

- Improvement in control and performance
  The system is developed to cope up with the current issues and problems of student grades. The system can add users (students, admins), validate users and is also bug free.

- Save cost
  After computerized system is implemented less human force will be required to maintain the grading system thus reducing the overall cost.

- Save time
  Admins are able to search for grade records by using few clicks of mouse and few search keywords thus saving his/her valuable time.

- Option of online access

  Student and Lecturer will be able to access the system online from anywhere.

- Lecture Notes

  Lecturer have a facility to upload lectures notes in a pdf file having size not more than 10mb

2.1.3 SYSTEM REQUIREMENTS

2.1.3.1 NON-FUNCTIONAL REQUIREMENTS

□ Product Requirements

EFFICIENCY REQUIREMENT

When the university system will be implemented staff and user will easily access the system and grades management will be very faster.

RELIABILITY REQUIREMENT

The system should accurately perform member registration, member validation, student learning resources, grades management etc.

USABILITY REQUIREMENT

The system is designed for a user-friendly environment so that student and staff of the university can perform the various tasks easily and in an effective way.

ORGANIZATIONAL REQUIREMENT

IMPLEMENTATION REQUIREMNTS

In implementing the whole system, it uses html in front end with php as server-side scripting language which will be used for database connectivity and the backend ie the database part is developed using MySQL.

DELIVERY REQUIREMENTS

The whole system is expected to be delivered in three months of time with a weekly evaluation by the project supervisor.

2.1.3.2 FUNCTIONAL REQUIREMENTS

1. NORMAL USER

1.1 USER LOGIN

Description of feature

This feature used by the user to login into system. They are required to enter username and password before they are allowed to enter the system. The username and password will be verified and if invalid username and password, the user will be redirected back to the landing page and not allowed to gain access to the system.

Functional requirements

- username is provided when they register
- The system must only allow user with valid username and password to enter the system
- The system performs authorization process which decides what user level users can gain access to.
- The user must be able to logout after they finished using system.

1.2 REGISTER NEW USER

Description of feature

This feature can be performed by the super admin , school admin and as well as the department admin to register new user into the system (such as admins and lecturers and students).

Functional requirements

- System must be able to verify information
- System must be able to delete information if information is wrong

1.5 SEARCH STUDENTS

DESCRIPTION OF FEATURE

This feature is found in top level admins part. we can search for the details of a particular student.

Functional requirements

- System must be able to search the database based on select search type
- System must be able to filter student details based on keyword entered
- System must be able to show the filtered student data in table view

1.6 GRADE ADDITION

DESCRIPTION OF FEATURE

This feature allows top and middle tier level admins such as super – admin, school – admin and departmental admin to add information such as student grade per semester and cumulatively.

Functional requirements

-System should be able to add detailed information about student grades.

-System should be able to display information to students individually in their respective dashboards.

2.1.4 SOFTWARE AND HARDWARE REQUIREMENTS

This section describes the software and hardware requirements of the system

2.1.4.1 SOFTWARE REQUIREMENTS

- Operating system- Windows 7 – current Windows 11 is used as the operating system as it is stable and supports more features and is more user friendly.
- XAMMP (Cross platform Apache) server that provides a local directory facility that the system runs on.
- Database MYSQL-MYSQL is used as database as it easy to maintain and retrieve records by simple queries which are in English language which are easy to understand and easy to write.
- Development tools and Programming language- HTML is used to write the whole code and develop webpages with CSS, JavaScript for styling and user interactivity work and php for sever side scripting.

2.1.4.2 HARDWARE REQUIREMENTS

- ➢ Intel core i7 2nd generation is used as a processor because it is fast than other processors an provide reliable and stable and we can run our pc for longtime. By using this processor, we can keep on developing our project without any worries. But also supports Intel core i3 and i5 quad core processors.
- ➢ Ram 1 Gb (at least needed) is used as it will provide fast reading and writing capabilities and will in turn support in processing

2.2   EXISTING VS PROPOSED SYSTEM

i.   Existing system does not have any facility of lecturer's login or student login whereas proposed system will have a facility of student login as well as lecturer login

ii.   Existing system does not have a facility of online viewing of books submitted by lecturers whereas proposed system has a facility of online viewing of books

iii.   Existing system does not have any option of lectures notes uploaded by lecturer whereas proposed system will have this facility

iv.   Existing system does not have any facility to generate student progress reports as well as manage student data whereas proposed system provides admins with a tool to generate student progress reports.

v.   Existing system does not have any facility for student request and suggestions where as in proposed system after logging in to their accounts student can request on issue with regards their grades as well as provide suggestions to improve the grading system.

2.3   SOFTWARE TOOLS USED

The whole Project is divided in two parts the front end and the back end.

2.3.1 Front end
The front end is designed using of html, Php, CSS, Java script and Bootstrap

- HTML- HTML or **Hyper Text Markup Language** is the main markup language for creating web pages and other information that can be displayed in a web browser.HTML is written in the form of HTML elements consisting of *tags* enclosed in angle brackets (like `<html>`), within the web page content. HTML tags most commonly come in pairs like `<h1>` and `</h1>`, although

some tags represent *empty elements* and so are unpaired, for example `<img>`. The first tag in a pair is the *start tag*, and the second tag is the *end*

tag (they are also called *opening tags* and *closing tags*). In between these tags web designers can add text, further tags, comments and other types of text-based content. The purpose of a web browser is to read HTML documents and compose them into visible or audible web pages. The browser does not display the HTML tags, but uses the tags to interpret the content of the page.HTML elements form the building blocks of all websites. HTML allows images and objects to be embedded and can be used to create interactive forms. It provides a means to create structured documents by denoting structural semantics for text such as headings, paragraphs, lists, links, quotes and other items.

It can embed scripts written in languages such as JavaScript which affect the behavior of HTML web pages.

- CSS- **Cascading Style Sheets** (**CSS**) is a style sheet language used for describing the look and formatting of a document written in a markup

language. While most often used to style web pages and interfaces written in HTML and XHTML, the language can be applied to any kind of XML document, including plain XML, SVG and XUL. CSS is a cornerstone specification of the web and almost all web pages use CSS style sheets to describe their presentation.CSS is designed primarily to enable the separation of document content from document presentation, including elements such as the layout, colors, and fonts. This separation can improve content accessibility, provide more flexibility and control in the specification

of presentation characteristics, enable multiple pages to share formatting, and reduce complexity and repetition in the structural content (such as by allowing for table less web design).CSS can also allow the same markup page to be presented in different styles for different rendering methods, such as on-screen, in print, by voice (when read out by a speech-based browser or screen reader) and on Braille-based, tactile devices. It can also be used to

allow the web page to display differently depending on the screen size or device on which it is being viewed. While the author of a document typically links that document to a CSS file, readers can use a different style sheet, perhaps one on their own computer, to override the one the author has

specified. However, if the author or the reader did not link the document to a specific style sheet the default style of the browser will be applied.CSS specifies a priority scheme to determine which style rules apply if more than one rule matches against a particular element. In this so-called *cascade*, priorities or *weights* are calculated and assigned to rules, so that the results are predictable.

- JAVA SCRIPT- **JavaScript** (**JS**) is a dynamic computer programming language. It is most commonly used as part of web browsers, whose implementations allow client-side scripts to interact with the user, control the browser, communicate asynchronously, and alter the document content that is displayed. It is also being used in server-side programming, game development and the creation of desktop and mobile applications. JavaScript is a prototype-based scripting language with dynamic typing and has first class functions. Its syntax was influenced by C. JavaScript copies many names and naming conventions from Java, but the two languages are

otherwise unrelated and have very different semantics. The key design principles within JavaScript are taken from the Self and Scheme programming languages. It is a multiparadigm language, supporting object-oriented, imperative, and functional programming styles. The application of JavaScript to use outside of web pages—for example, in PDF documents, site-specific browsers, and desktop widgets—is also significant. Newer and faster JavaScript VMs and platforms built upon them (notably Node.js) have also increased the popularity of JavaScript for server-side web applications. On the client side, JavaScript was traditionally implemented as

an interpreted language but just-in-time compilation is now performed by recent (post-2012) browsers.

- PHP- **PHP** is a server-side scripting language designed for web development but also used as a general-purpose programming language. PHP is now installed on more than 244 million websites and 2.1 million web servers. Originally created by Rasmus Lerdorf in 1995, the reference implementation of PHP is now produced by The PHP Group. While PHP originally stood for

  Personal Home Page, it now stands for *PHP: **Hypertext Preprocessor**,* a recursive acronym. PHP code is interpreted by a web server with a PHP processor module, which generates the resulting web page: PHP commands can be embedded directly into an HTML source document rather than calling an external file to process data. It has also evolved to include a command-line interface capability and can be used in standalone graphical applications. PHP is free software released under the PHP License. PHP can be deployed on most web servers and also as a standalone shell on almost every operating system and platform, free of charge.

2.3.2 BACK END- The back end is designed using MySQL which is used to design the databases

➢ MYSQL- **MySQL** ("My S-Q-L", officially, but also called "My Sequel") is (as of July 2013) the world's second most widely used open-source relational database management system (RDBMS). It is named after co-founder Michael Widenius daughter, My. The SQL phrase stands for Structured Query Language. The

  MySQL development project has made its source code available under the terms of the GNU General Public License, as well as under a variety of proprietary agreements. MySQL was owned and sponsored by a single for-profit firm, the Swedish company MySQL AB, now owned by Oracle Corporation. MySQL is a popular choice of database for use in web applications, and is a central component of the widely used LAMP open source web application software stack (and other 'AMP' stacks). LAMP is an acronym for "Linux, Apache,

MySQL, Perl/PHP/Python." Free-software-open source projects that require a full-featured database management system often use MySQL. For commercial use, several paid editions are available, and offer additional functionality.

Applications which use MySQL databases

include: TYPO3, MODx, Joomla, WordPress, phpBB, MyBB, Drupal and other software. MySQL is also used in many high-profile, large-scale websites, including Wikipedia, Google (though not for searches), Facebook, Twitter, Flickr, and YouTube

CHAPTER 3
SYSTEM DESIGN

3.1   TABLE DESIGN

VARIOUS TABELS TO MAINTAIN INFORMATION

1.   ADMIN TABLE FOR SYSTEM ADMINS

Field	Datatype	NULL	Default	Key	Extra
id	INT (11)	NOT NULL	NONE	PRIMARY KEY	AUTO_INCREMENT
username	VARCHAR (50)	NO	NONE		
password	VARCHAR (50)	NO	NONE		
role	INT (11)	NO	NONE	FOREIGN KEY	REFERENCES ROLE TABLE
fname	VARCHAR (50)	NO	NONE		
mname	VARCHAR (50)	NO	NONE		
lname	VARCHAR (50)	NO	NONE		
studentID	VARCHAR (50)	NO	NONE		
image	VARCHAR (50)	NO	NONE		
gender	INT (11)	NO	NONE		

program	INT (11)	NO	NONE	FOREIGN KEY	REFERENCES PROGRAM TABLE
level	INT (11)	NO	NONE		
school	INT (11)	NO	NONE	FOREIGN KEY	REFERENCES SCHOOL TABLE
department	INT (11)	NO	NONE	FOREIGN KEY	REFERENCES DEPARTMENT TABLE

2. TABLE FOR STORING INFORMATION ON COURSES

Field	Datatype	NULL	Default	Key	Extra
courseid	INT (11)	NOT NULL	NONE	PRIMARY KEY	AUTO_INCREMENT
coursecode	VARCHAR (50)	NO	NONE		
credithour	INT (5)	NO	NONE		
coursename	VARCHAR (50)	NO	NONE		

3. TABLE FOR STORING INFORMATION ON DEPARTMENTS

Field	Datatype	NULL	Default	Key	Extra
id	INT (11)	NOT NULL	NONE	PRIMARY KEY	AUTO_INCREMENT
department	VARCHAR (50)	NO	NONE		

4. TABLE FOR STORING GALLERY INFORMATION

Field	Datatype	NULL	Default	Key	Extra
id	INT (11)	NOT NULL	NONE	PRIMARY KEY	AUTO_INCREMENT
image	VARCHAR (50)	NO	NONE		

5. TABLE FOR STORING STUDENTS GRADES

Field	Datatype	NULL	Default	Key	Extra
gradeid	INT (11)	NOT NULL	NONE	PRIMARY KEY	AUTO_INCREMENT
grade	VARCHAR (1)	NO	NONE		
course	INT (11)	NO	NONE	FOREIGN KEY	REFERENCES COURSE TABLE
student	INT (11)	NO	NONE	FOREIGN KEY	REFERENCES COURSE TABLE
semester	VARCHAR (50)	NO	NONE		

6. TABLE FOR STORING PROGRAM INFORMATION

Field	Datatype	NULL	Default	Key	Extra
id	INT (11)	NOT NULL	NONE	PRIMARY KEY	AUTO_INCREMENT
program	VARCHAR (50)	NO	NONE		
department	VARCHAR (50)	NO	NONE	FOREIGN KEY	REFERENCES DEPARTMENT TABLE
school	VARCHAR (50)	NO	NONE	FOREIGN KEY	REFERENCES SCHOOL TABLE

7. TABLE FOR STORING STUDENT REVIEW INFORMATION

Field	Datatype	NULL	Default	Key	Extra
courseid	INT (11)	NOT NULL	NONE	PRIMARY KEY	AUTO_INCREMENT
title	VARCHAR (50)	NO	NONE		
review	VARCHAR (50)	NO	NONE		
name	VARCHAR (50)	NO	NONE		
image	VARCHAR (50)	NO	NONE		
role	INT (11)	NO	NONE	FOREIGN KEY	REFERENCES ROLE TABLE

8. TABLE FOR STORING USER ROLE

Field	Datatype	NULL	Default	Key	Extra
id	INT (11)	NOT NULL	NONE	PRIMARY KEY	AUTO_INCREMENT
role	VARCHAR (50)	NO	NONE		

9. TABLE FOR STORING SCHOOL INFORMATION

Field	Datatype	NULL	Default	Key	Extra
id	INT (11)	NOT NULL	NONE	PRIMARY KEY	AUTO_INCREMENT
school	VARCHAR (50)	NO	NONE		

10. TABLE FOR STORING SLIDER INFORMATION

| Field	Datatype	NULL	Default	Key	Extra
id	INT (11)	NOT NULL	NONE	PRIMARY KEY	AUTO_INCREMENT
image	VARCHAR (50)	NO	NONE		

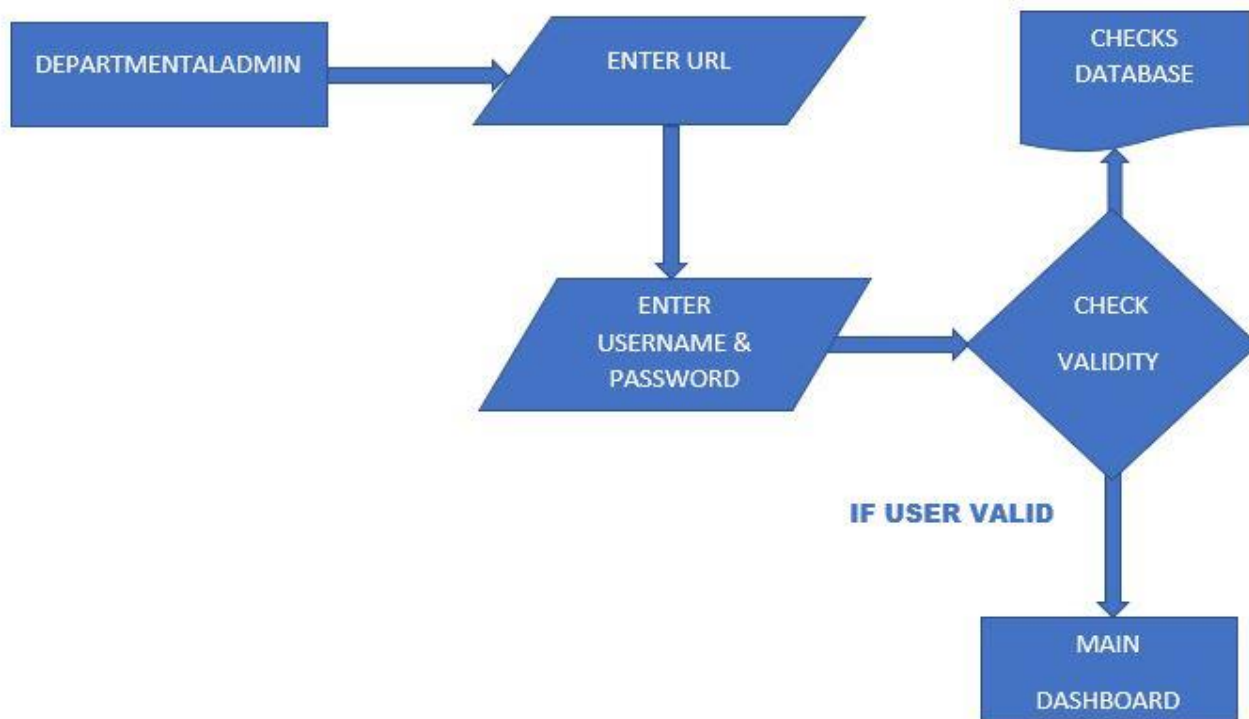## 3.2   DATA FLOW DIAGRAMS

DATA FLOW DIAGRAM FOR SUPERADMIN LOGIN
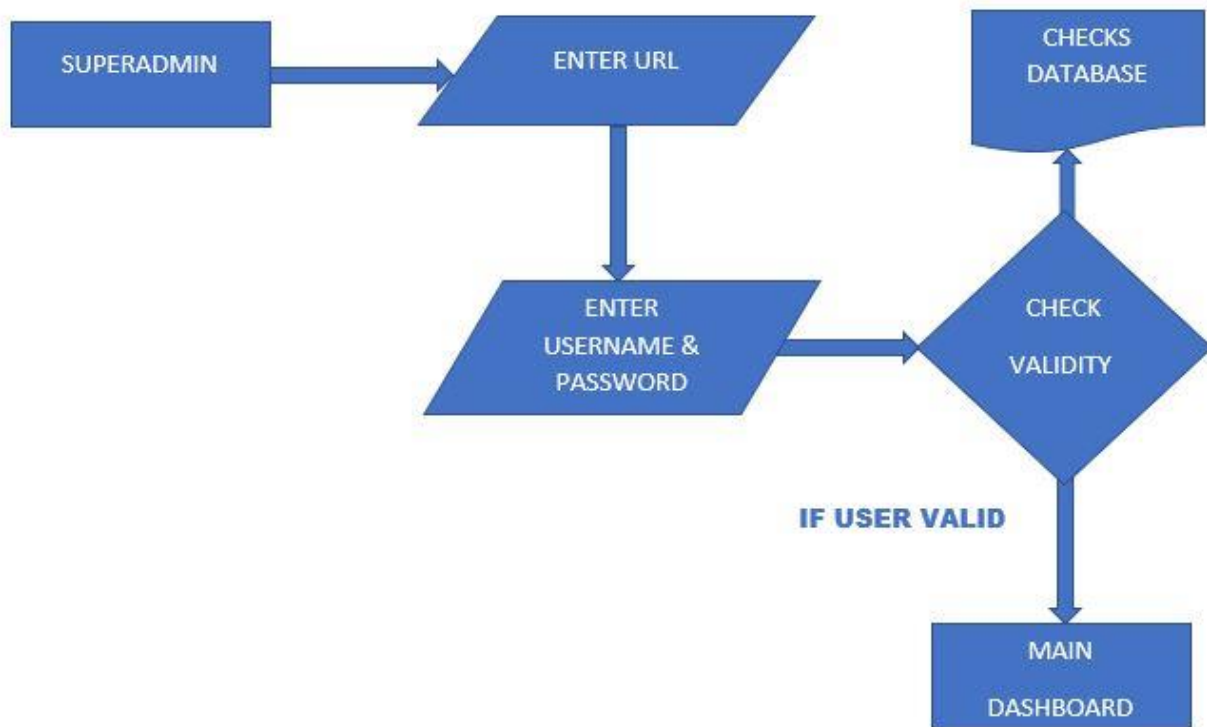
After entering the home page of the website, Super - admin can choose the LOGIN option where they are asked to enter username & password, and if he/she is a valid user then they will be redirected to the main dashboard.

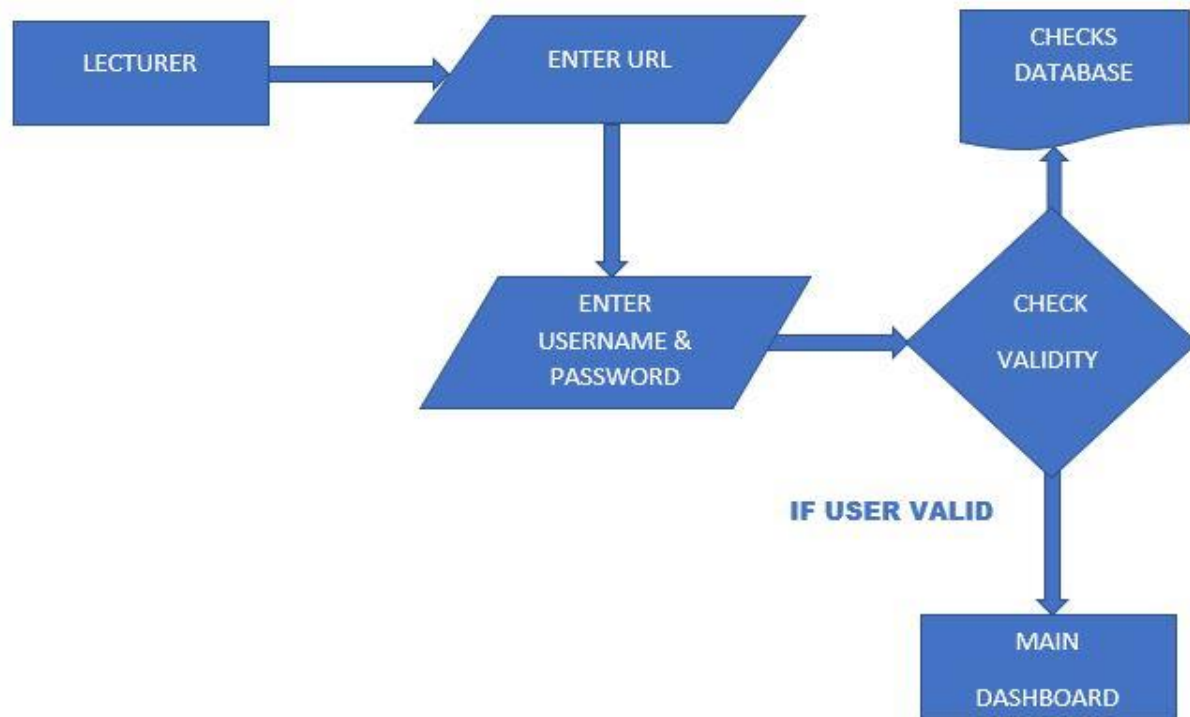## DATA FLOW DIAGRAM FOR SCHOOL ADMIN LOGIN
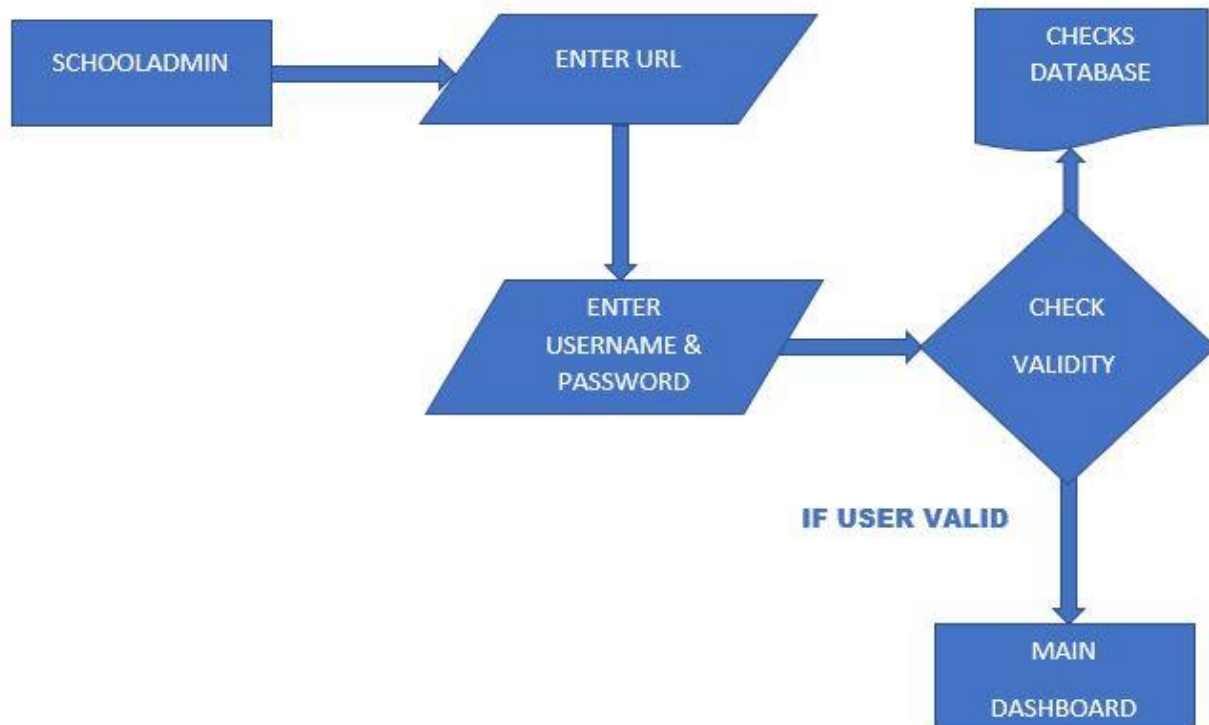
After entering the home page of the website, School - admin can choose the LOGIN option where they are asked to enter username & password, and if he/she is a valid user then they will be redirected to the main dashboard.

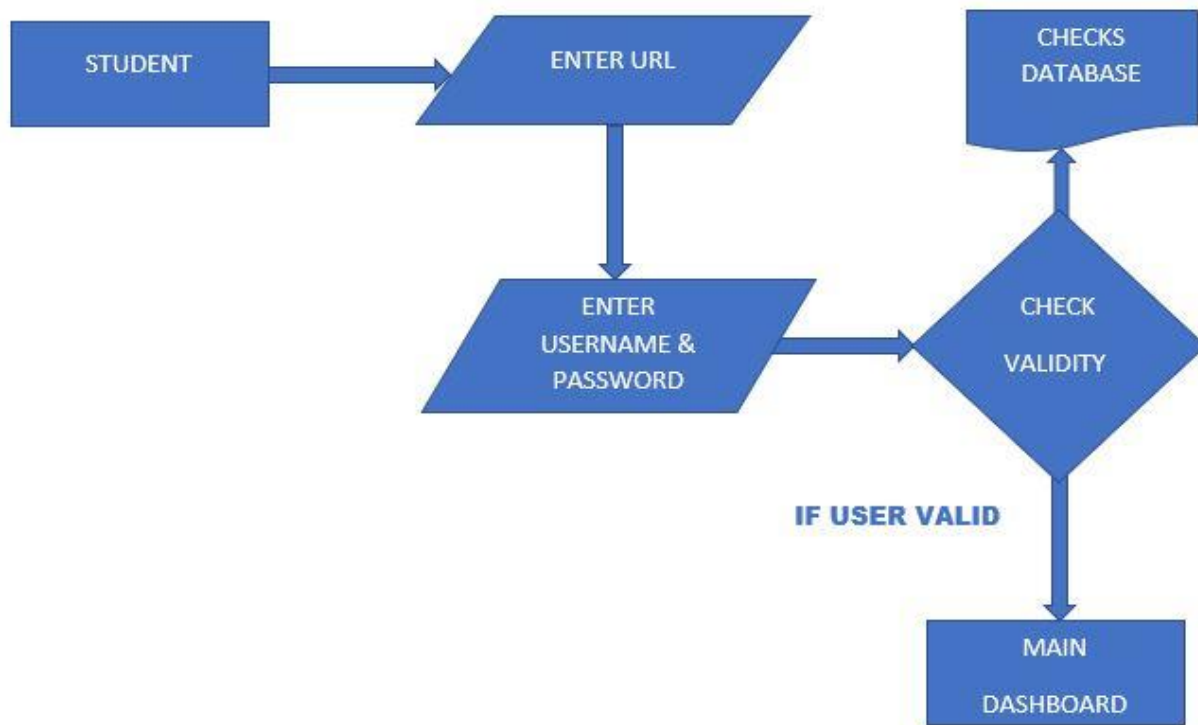## DATA FLOW DIAGRAM FOR DEPARTMENTAL ADMIN LOGIN

After entering the home page of the website, Departmental - admin can choose the LOGIN option where they are asked to enter username & password, and if he/she is a valid user then they will be redirected to the main dashboard.

.

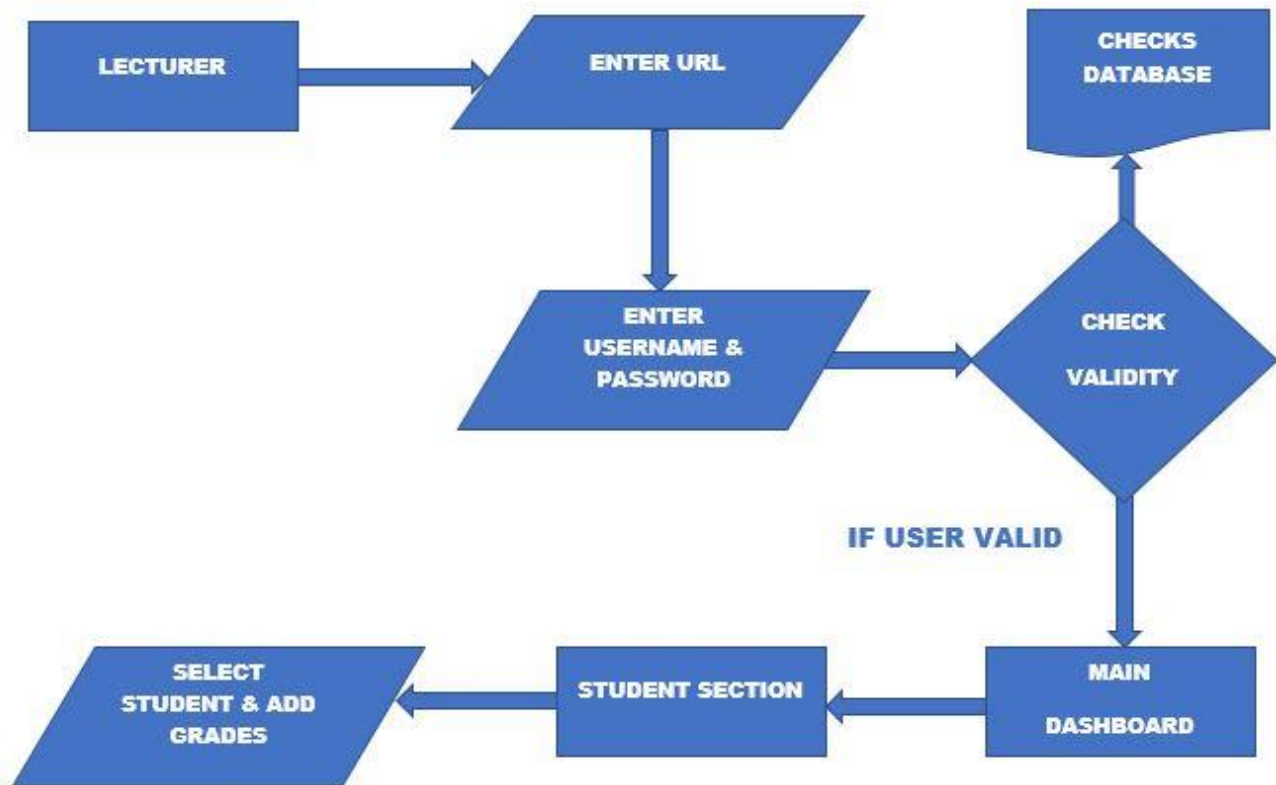## DATA FLOW DIAGRAM FOR LECTURER LOGIN

After entering the home page of the website, Lecturer - admin can choose the LOGIN option where they are asked to enter username & password, and if he/she is a valid user then they will be redirected to the main dashboard.

DATA FLOW DIAGRAM FOR STUDENT LOGIN

After entering the home page of the website, Student can choose the LOGIN option where they are asked to enter username & password, and if he/she is a valid user then they will be redirected to the main dashboard.

DATA FLOW DIAGRAM FOR LECTURER ADDING STUDENT GRADES

After entering the main dashboard of the website, Lecturer can now add student grades based on courses selected by student.
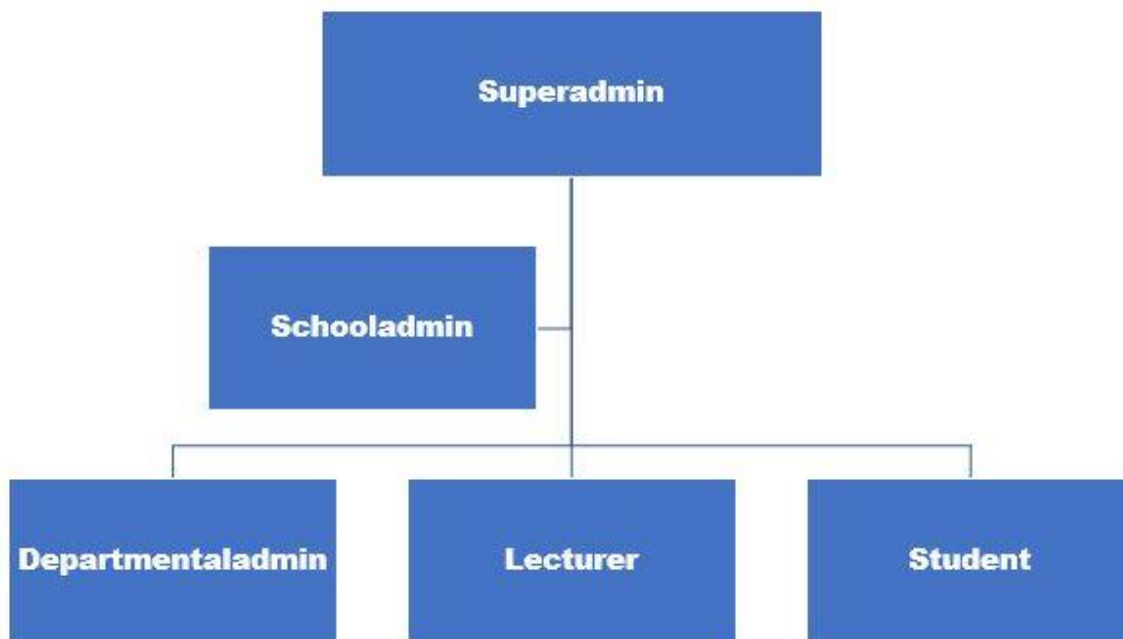
DATA FLOW DIAGRAM FOR LECTURER ADDING STUDENT NOTES
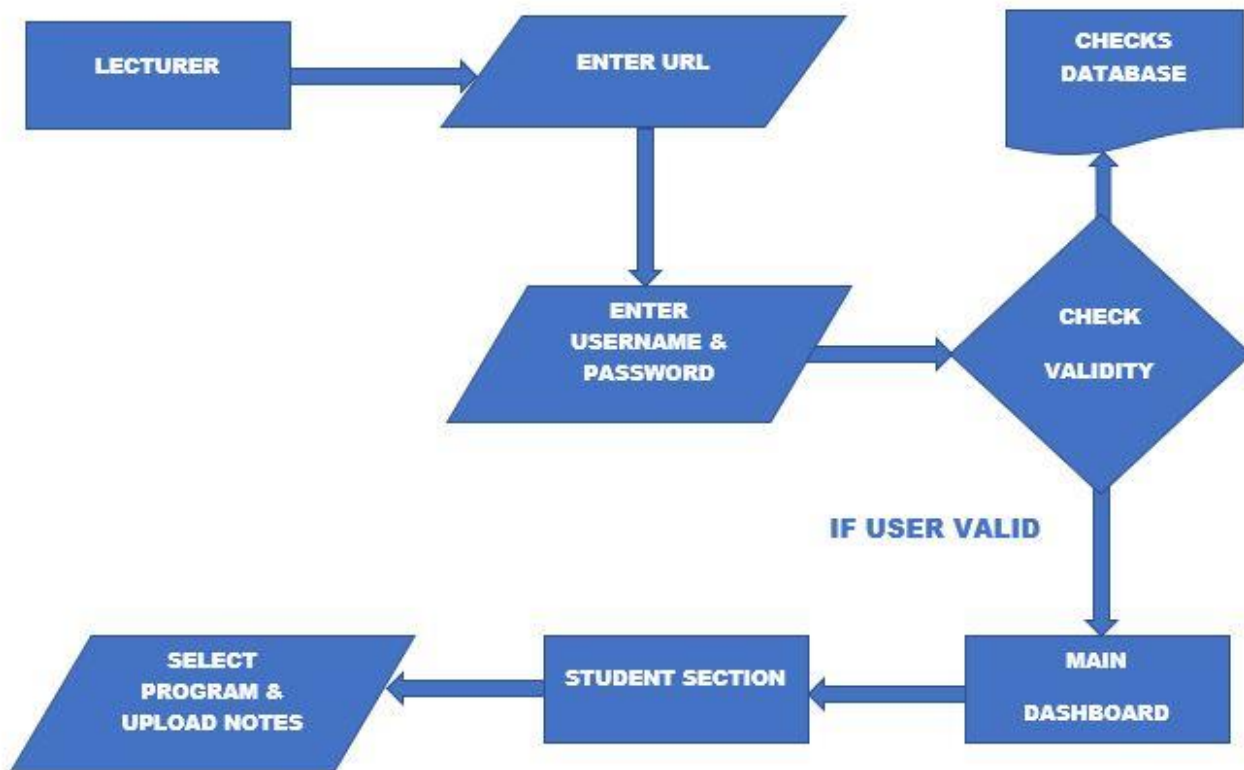
After entering the main dashboard of the website, Lecturer can now add the student notes based on courses selected by student and their various semester.

DATA FLOW DIAGRAM OF THE ENTIRE SYSTEM USER LEVEL

Summary of the entire user level and privileges of the system.

CHAPTER 5

SYSTEM TESTING

The aim of the system testing process was to determine all defects in our project. The program was subjected to a set of test inputs and various observations were made and based on these observations it will be decided whether the program behaves as expected or not.

Our Project went through two levels of testing

1. Unit testing
2. Integration testing

UNIT TESTING

Unit testing is undertaken when a module has been created and successfully reviewed. In order to test a single module, we need to provide a complete environment. i.e. besides the module we would require

- The procedures belonging to other modules that the module under test calls
- Non local data structures that module accesses
- A procedure to call the functions of the module under test with appropriate parameters

Unit testing was done on each and every module that is described under module description of chapter 4

1. Test for the Super admin module

- Testing user login form-This form is used to sign in the different level of administrators of the system. In this we enter the username and password if both are correct the user will get redirected to the specific dashboard based on

their role otherwise if any of data is wrong it will get redirected back to the login page and again ask for username and password again.

- School addition – In this section the super admin can verify the function responsible for creating the school module is working as intended.

- Department addition – In this section the super admin can also verify the function responsible for creating the department module is properly functioning.

- Program addition – In this section the super admin can verify the function (method) responsible for creating and adding a particular program module is working as intended.

- Sub admins addition - In this section the super admin can verify and add sub admins (school, department and lecturer admins) into system database, that will help in reducing the overall burden on the super admin. It contains add, edit and delete buttons if user click add button data will be added to admin database, if user click edit button data will be edited and modified into admin database and if he/she clicks delete button the sub admin data will be deleted from the system.

- Student addition - In this section the super admin can verify student details from student academic info and then only add student details to main grading system database it contains add, edit and delete buttons if user click add button data will be added to student database, if user click edit button data will be edited and modified student data inside the database and if he clicks delete button the student data will be deleted from the system.

- System utility settings - Admin can handle other system settings like gallery images, review and slider images.

2. Test for the Sub admins module

(A) School Admin

- Department addition – In this section the school admin can also verify the function responsible for creating the department module is properly functioning.

- Program addition – In this section the school admin can verify the function (method) responsible for creating and adding a particular program module is working as intended.

- Student addition - In this section the school admin can verify student details from student academic info and then only add student details to main grading system database it contains add, edit and delete buttons if user click add button data will be added to student database, if user click edit button data will be edited and modified student data inside the database and if he clicks delete button the student data will be deleted from the system.

(B) Department Admin

- Program addition – In this section the department admin can verify the function (method) responsible for creating and adding a particular program module is working as intended.

    - Student addition - In this section the department admin can verify student details from student academic info and then only add student details to main grading system database it contains add, edit and delete buttons if user click add button data will be added to student database, if user click edit button data will be edited and modify student data inside the database and if he clicks delete button the student data will be deleted from the system.

- Lecturer addition - In this section the department admin can verify lecturer info and then only add lecturer details to main grading system database it contains add, edit and delete buttons if user click add button data will be added to student database, if user click edit button data will be edited and modify lecturer data

inside the database and if he clicks delete button the student data will be deleted from the system.

(C) Lecturer Admin

- Grades addition – In this section the lecturer admin can verify the function (method) responsible for adding a particular student grade module is working as intended.
- Learning materials addition - In this section the lecturer admin can verify notes and upload notes into main grading system database. It contains add, edit and delete buttons if user click add button data will be added to notes database, if user click edit button data will be edited and modify notes data inside the database and if he clicks delete button the notes uploaded will be deleted from the system.

INTEGRATION TESTING

In this type of testing we test various integration of the project module by providing the input. The primary objective is to test the module interfaces in order to ensure that no errors are occurring when one module invokes the other module.

CHAPTER 6

CONCLUSION & FUTURE SCOPE

This website provides a computerized version of university grading system which will benefit the students as well as the staff of the university.

It makes entire process online where student can access their grades, staff can generate student data and process student grades. It also has a facility for student login where student can login and can see views of books issued by their respective lecturers. It has a facility of lecturer's login where lecturers can add lecture notes and also give necessary suggestion to the improve of the system and also add student grades.

There is a future scope of this facility that many more features such as online lectures video tutorials can be added by teachers as well as online assignments submission facility , a feature Of group chat where students can discuss various issues with regards to their studies which can be added to this project thus making it more interactive more user friendly and project which fulfills each users need in the best way possible

CHAPTER 7
REFERENCES

- http://www.w3schools.com/html/html_intro.asp

- http://www.w3schools.com/css/css_background.asp

- http://www.w3schools.com/js/js_datatypes.asp

- http://www.w3schools.com/sql/sql_insert.asp

- http://www.w3schools.com/sql/sql_update.asp

- http://www.w3schools.com/php/php_forms.asp

- Fundamentals of software engineering by Rajib mall, PHI learning

- Web development and application development by Ivan Byross BPB publications