**A Project report on**

**STUDENT MANAGEMENT SYSTEM**

A Dissertation submitted to JNTU Hyderabad in partial fulfillment of the academic requirements for the award of the degree.

## Bachelor of Technology

**In**

**Computer Science & Engineering**

Submitted by

**SANDUGULA SAI KARUN**

(14H51A05N7)

Under the esteemed guidance of

Mr.S.GNANENDER REDDY

(Associate Professor)



**Department of Computer Science & Engineering**

**CMR COLLEGE OF ENGINEERING AND TECHNOLOGY**

(An Autonomous Institution under UGC & JNTUH , Approved by AICTE, Permanently Affiliated to JNTUH, Accredited by NAAC with ’A’ Grade.)

KANDLAKOYA, MEDCHAL ROAD, HYDERABAD - 501401.

**2014- 2018**

**CMR COLLEGE OF ENGINEERING & TECHNOLOGY**

KANDLAKOYA, MEDCHAL ROAD, HYDERABAD – 501401

** DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING**

**CERTIFICATE**

This is to certify that the Project report entitled **"Student Management System"** being submitted by **S.Sai Karun, MVL Anulitha, B.Nanditha and K.Shashi Kiran *(14H51A05N7, 14H51A05M4, 14H51A05P8 and 14H51A05L5),*** in partial fulfillment for the award of **Bachelor of Technology in Computer Science and Engineering** is a record of bonafide work carried out his/her under my guidance and supervision.

The results embodies in this project report have not been submitted to any other University or Institute for the award of any Degree.

**Mr. Gnanender Reddy Dr. K. Vijaya Kumar**

**Associate Professor Professor and HOD**

**Dept. of CSE Dept. of CSE**

Submitted for viva voice Examination held on

**External Examiner**

**Acknowledgment**

With great pleasure I want to take this opportunity to express my heartfelt gratitude to all the people who helped in making this project work a grand success.

I am grateful to **Mr.Gnanender Reddy,** Associate Professor, Dept. of Computer Science and Engineering for his valuable suggestions and guidance during the execution of this project work.

I would like to thank **Dr. K. Vijaya Kumar**, Head of the Department of Computer Science and Engineering, for his moral support throughout the period of my study in CMRCET.

I am highly indebted to **Major Dr. V.A. Narayana**, Principal CMRCET for giving permission to carry out this project in a successful and fruitful way.

I would like to thank the Teaching & Non- teaching staff of Department of Computer Science and Engineering for their co-operation

Finally I express my sincere thanks to **Mr. Ch. Gopal Reddy**, Secretary, CMR Group of Institutions, for his continuous care. I sincerely acknowledge and thank all those who gave support directly and indirectly in completion of this project work.

**S. Sai Karun**

**14H51A05N7**

# ABSTRACT

A systematic educational institution requires the maintenance and organization

of student database. To further understand this, each institution has various

departments which are further classified into different sections. Student database

has to be maintained on the basis of a student's section as well as their department.

This information could be general details like sudent name, address, performance,

attendance, achievements and so on. Student database is a single module which has

to be maintained alongside several other modules, therefore requiring to be automated

and centralized as information from one module will be required by other modules.

Therefore, all the modules are interdependent.

For example, to view the marks of a student in their profile, will require the

assistance of the Examination branch module. Therefore, the project aims at efficiency

and accuracy. The administrators of the system will be able to record and retrieve

student information, manage their attendance with ease. The project therefore is

also useful for saving time by reducing the paper and manual work such as maintaining

the documents.

**TABLE OF CONTENTS**

|  |  |  |  |
| --- | --- | --- | --- |
| **CHAPTER NO.** | **TITLE** | | **PAGE NO.** |
|  | **LIST OF FIGURES** | | i |
|  | **ABSTRACT** | | ii |
| **1** | **INTRODUCTION** | | 1 |
|  | 1.1 | Purpose | 1 |
|  | 1.2  1.3  1.4  1.5 | Scope  Abbreviations  Client, Customer And Other Stakeholders  User Of Product | 1  2  2  4 |
| **2** | **BACKGROUND WORK** | | 6 |
|  | 2.1 | Existing System | 6 |
|  |  | 2.1.1 Disadvantages Of Existing System | 6 |
|  | 2.2 | Project Constraints  2.2.1 Mandated Constraints  2.2.2 Implementation environment of the current system | 6  6  6 |
|  | 2.3  2.4 | Functional Requirements  2.3.1The Scope of the Work  2.3.2 The Scope of the Product  2.3.3 Functional Requirements  Non - Functional Requirements  2.4.1 Look and Feel Requirements  2.4.2 Usability Requirements  2.4.3 Performance Requirements  2.4.4 Operational Requirements  2.4.5 Maintainability and Portability Requirements  2.4.6 Security Requirements  2.4.7 Legal Requirements | 8  8  8  9  10  10  10  10  10  11  11  11 |
| **3** | **PROPOSED SYSTEM** | | 13 |
|  | 3.1 | Proposed System  3.1.1 Advantages of Proposed System | 13  13 |
|  | 3.2 | Input & Output Design | 13 |
|  | 3.3 | Modules | 15 |
| **4** | **DESIGNING** | | 17 |
|  | 4.1 | UML Diagrams | 17 |
|  | 4.2 | ER Diagrams | 19 |
| **5** | **RESULTS AND DISCUSSION** | | 25 |
|  | 5.1 | Performance Analysis  5.1.1 System Study  5.1.2 System Testing | 25  25  26 |
| **6** | **CONCLUSION AND FUTUREWORK** | | 30 |

**List of Figures**

|  |  |  |
| --- | --- | --- |
| **FIGURE NO.** | **TITLE** | **PAGE NO.** |
| 4.1.1 | Class Diagram | 17 |
| 4.1.2 | Use Case Diagram | 18 |
| 4.1.3  4.1.4  4.1.5  4.1.6  4.1.7 | Sequence Diagram  Collaboration Diagram  Component Diagram  Activity Diagram  Deployment Diagram | 19  20  21  22  23 |

|  |  |  |
| --- | --- | --- |
|  |  |  |

CHAPTER 1

INTRODUCTION

**1. INTRODUCTION**

Within the broad spectrum of technology-enabled training delivery, individual products have different strengths, weaknesses, architectures, supported computing platforms, licensing approaches, and levels of standards conformance. Depending on an organization’s interest, the field of potential products can vary widely.

More and more, these products are expected to support a “blended learning” approach – or the ability to support both traditional (i.e., instructor led and paper-based) as well as technology-enabled delivery (i.e., Web-based or CD-ROM). Depending on whether your organization is interested in content delivery, competency management, content authoring, traditional classroom, or fee for service requirements; the alternatives can vary significantly.

### The purpose of this document is to outline the characteristics of LMS and LCMS products, present some best practices and considerations for selecting an appropriate system, offer self-evaluation questions to help interested organizations focus on products that best fit their particular needs, and highlight suggested next steps for acquiring and implementing the new system.

* 1. **Purpose**

An LMS delivers and manages instructional content, and typically handles student registration, online course administration, and tracking, and assessment of student work. Some LMSs help identify progress towards learning or training goals.[]](https://en.wikipedia.org/wiki/Learning_management_system#cite_note-3) Most LMSs are web-based, to facilitate access. LMSs are often used by regulated industries (e.g. financial services and biopharma) for compliance training. Some LMS providers include "performance management systems", which encompass employee appraisals, competency management, skills-gap analysis, succession planning, and multi-rater assessments . Some systems support competency-based learning.

Though there are a wide variety of terms for digital aids or platforms for education, such as course management systems, virtual or managed learning platforms or systems, or computer-based learning environment, the term learning management system has become the ubiquitous term for products that help administer or deliver part or all of a course.

.

* 1. **Scope**

In 2017 and 2018, learning management systems are expected to become even more personalized, and to offer advanced functionality such as 3D learning. Experts also agree that modern learning management systems will offer more collaborative and social features, and will be delivered exclusively as cloud-hosted services. It is expected that gratification will continue to increase in prominence as will virtual learning.

**1.3 Abbreviations**

* **LMS** – Learning Management System
* **SRS** – Software requirements specification
* **PC** – Personal Computer
* **HDD** - Hard Disc Drive
* **RAM** – Random Access Memory
* **IE** – Microsoft Internet Explorer

**1.4Client, Customer and other Stakeholders**

The ***client*** is the person/s who pays for the development, and owner of the delivered system.

All client remarks will be improved immediately. Product deliverables have appropriated project schedule, approved by the client.

The customer is the person/s who will buy the product from the client.

In our case, the roles of the client and the customer are filled by the same company.

**Stakeholders include**:

* End - Users
* System Designer
* System Tester
* System Administrator

SRS identifies each type of stakeholder:

**Table 1. End - Users**

|  |  |
| --- | --- |
| Stakeholder Identification | Students and faculty |
| Knowledge needed by the project | Basic knowledge of Microsoft Windows-based interfaces, Internet browser’s interfaces. |
| Necessary degree of stakeholder’s involvement | Low. End – Users are not involved in software development, just in some kinds of sociological surveys |
| Degree of influence for that stakeholder | High. Software to be developed to satisfy first of all the End-Users. |

**Table 2. Customer**

|  |  |
| --- | --- |
| Stakeholder Identification | Customer |
| Knowledge needed by the project | Customer’s requirements, Project plan, business modeling and forecasting. |
| Necessary degree of stakeholder’s involvement | Middle. Customer supervises project, establishes budget |
| Degree of influence for that stakeholder | High. Software development process may be canceled by Customer. |

**Table 3. System Designer**

|  |  |
| --- | --- |
| Stakeholder Identification | System Designer |
| Knowledge needed by the project | Computer-based Design tools, Internet technologies |
| Necessary degree of stakeholder’s involvement | High. System Designer defines all software interfaces, how the product will look like, providing success of product from the End – Users point of view. |
| Degree of influence for that stakeholder | Low. System Designer will not use product to be developed in appropriate way. |

**Table 4. System Tester**

|  |  |
| --- | --- |
| Stakeholder Identification | System Tester |
| Knowledge needed by the project | Computer-based Testing tools, Internet technologies, Testing requirements and specifications. |
| Necessary degree of stakeholder’s involvement | Middle. System Tester is looking for mistakes in software and verify product’s possibilities. |
| Degree of influence for that stakeholder | Low. System Tester will not use product to be developed in appropriate way. |

**Table 5. System Administrator**

|  |  |
| --- | --- |
| Stakeholder Identification | System Administrator, College staff |
| Knowledge needed by the project | Strong computer skills, Internet and network technologies. |
| Necessary degree of stakeholder’s involvement | Low. System Administrator is not involved in software development process. |
| Degree of influence for that stakeholder | Middle. System Administrator will maintain and install product. |

**1.5 The Users of the Product**

**Table 8. Examiner**

|  |  |
| --- | --- |
| User name | College Examiners |
| User role | To maintain Exam results (add, delete, renew items and update databases) |

**Table 9. Students**

|  |  |
| --- | --- |
| User name | Students |
| User role | To use the College SCBMS in practice (from college or any place, where Internet connection is available) |

CHAPTER 2

BACKGROUND WORK

**2. BACKGROUND WORK**

* 1. **Existing System**

Traditional service discovery approaches of the web services technology are based on Universal Description, Discovery, and Integration (UDDI). However, centralized service registries used by UDDI may easily suffer from problems in an open SOC environment. To overcome the problems The Peer-to-Peer (P2P) technology provides a universal approach to improving reliability, scalability, and robustness of distributed systems by removing centralized infrastructures. Based on Distributed Hashing Table (DHT), structured P2P systems can achieve even data distribution and efficient query routing by controlling the topology and imposing constraints on the data distribution. In this technology also problems occur.

**2.1.1 DISADVANTAGES OF EXISTING SYSTEM:**

Although structured P2P can potentially improve the scalability of service discovery, directly applying DHT based P2P approaches to decentralized service discovery may be weak in guaranteeing the availability of published service descriptions. This is because DHT-based systems often distribute descriptions of functionally equivalent services to the same successor node, as they have the same or similar hashing values. If such a node fails, a service consumer will not be able to discover any of these services. This disadvantage may result in serious problems in open and dynamic SOC environments where unexpected failure of nodes cannot be avoided.

**2.2 Project Constraints**

**2.2.1 Mandated Constraints**

Next items must be used to verify software:

* + 1. For user home PC

Hardware

* Intel core i3 processor
* 50Mbytes free space on HDD
* 32Mbytes RAM
* Internet connection

Software

* MS Windows 7/8/XP
* MS IE, Netscape or chrome browsers with Java2 support

**2.2.2 Implementation environment of the current system**

This part of the specification will provide enough information about the environment for the designers to make the product successfully interact with its surrounding technology.

Here will be shown a diagram, with some kind of icon to represent each separate device or person (processor) and interfaces between the processors.

College workstation Remote PC

BS00580_ BS00580_

BD06675_Web - Interface

BS00554_

Integrated Databases

**2.3. FUNCTIONAL REQUIREMENTS**

**2.3.1 The Scope of the Work**

Software to be developed should be providing all necessary rules for exam staff and students.

There are several motivations to order new computer-based College SCBMS:

1. To modernize College Exam section department database
2. To optimize Exam section staff work and time
3. To join small Access-based database, where staff has stored information about student results
4. To check ability of commercial using of student credit based management systems.

**2.3.2 The Scope of the Product**

Features provided by the learning management systems. :

1. Store necessary information about students details :

* Student name
* Subjects names

1. Allow a search item by student.
2. System will provide admin to add, modify, and remove items to/from the database, and check availability of the item.
3. System will allow users to get information about his/her status after authorization procedures:

* User name
* Pass word
* Student number

**2.3.3 Functional requirements**

Functional requirements are the following:

* 1. The LMS should store all information about student’s subjects.
  2. The LMS should store all information about subjects.
  3. The LMS allow searching items by author, title or keywords
  4. The LMS should support 10students’ requests simultaneously.
  5. The LMS should allow staff to add items in database, and check availability of the items.
  6. The LMS should generate request’s reports for student results every day, on base of which exams staff could make decisions about promoting to next academics years
  7. The LMS should create notification and send to students by e-mail
  8. The LMS should allow students to get their time table and status.
  9. The LMS should provide to students search from the college computers (LMS) from outside the college through College site (web-based) though the Internet.
  10. The LMS should provide access to previous Access-based database, online databases
  11. The LMS will be integrated with other colleges and universities and allow to check content status.

**2.4. Non - Functional Requirements**

**2.4.1 Look and Feel Requirements**

According to the Customer requirements, the College LMS should include following interfaces:

* + The LMS interfaces will the same for college staff and students based on Java application. Differences will depend on users’ functions.
  + The LMS interface for system administrator will include Java application, Command Line, System files
  + Web interface. This interface will provide search, request and renew procedures, connection with other online databases. Web interface should work correctly in different browsers.

**2.4.2 Usability Requirements**

* Economical and clear interface
* The interface should contain prompts and help to avoid making mistakes
* The product should be used by people with no training

**2.4.3 Performance Requirements**

* + Any interface between a user and LMS should have a maximum response time of 5 seconds
* The response should be fast enough to avoid users’ response collisions
* The LMS should be available for use 24 hours per day, 365 days per year.
* The LMS should support 50 students database

**2.4.4 Operational Requirements**

* + The LMS should be used on all compatible workstations with 50Mbytes free space on HDD for exam department workstations.
  + The LMS should be correctly implemented in different Internet browsers
  + The LMS should correctly interface if MS Access applications and MS SQL Server

**2.4.5 Maintainability and Portability Requirements**

* + Changes (new students addition, password changes, database changes) must be verified once per semester at least
  + The LMS should provide automatically notification to students by e-mail about total number of credits, exam results, availability of next year and etc.
  + The LMS is expected to run under MS Windows 7/8//XP

**2.4.6 Security Requirements**

* + The LMS should provide databases’ modification only for system administrator after authorization procedures
  + Access to the LMS is permitted only for College student and staff after authorization procedures

**2.4.7 Legal Requirements**

* Personal information should be protected
* The LMS should comply with quality assurance standards

CHAPTER 3

PROPOSED SYSTEM

**3. PROPOSED SYSTEM**

**3.1 PROPOSED SYSTEM:**

This paper proposes Chord4S, a Chord-based decentralized service discovery approach that supports service description distribution and discovery in a P2P manner. Chord is selected because it is well recognized for its flexibility and scalability and is considered suitable in large scale SOC environments.

**3.1.1ADVANTAGES OF PROPOSED SYSTEM:**

Chord4S takes advantages of the basic principles of Chord for nodes organization, data distribution and query routing.

* The main aim of designing Chord4S is to largely improve the availability of service descriptions in volatile environments by distributing descriptions of functionally equivalent services to different successor nodes.
* In case one node fails, a service consumer is still able to find functionally equivalent services that are stored at other successor nodes. Another two features of Chord4S are to support service discovery with wildcard(s) and QoS awareness.
* Chord4S extends Chord’s original routing protocol to support discovery of multiple functionally equivalent services at nodes with one query, which is necessary for different successor negotiation of a Service Level Agreement and selection of optimal service providers

**3.1 Input & Output Design**

**INPUT DESIGN**

The input design is the link between the information system and the user. It comprises the developing specification and procedures for data preparation and those steps are necessary to put transaction data in to a usable form for processing can be achieved by inspecting the computer to read data from a written or printed document or it can occur by having people keying the data directly into the system. The design of input focuses on controlling the amount of input required, controlling the errors, avoiding delay, avoiding extra steps and keeping the process simple. The input is designed in such a way so that it provides security and ease of use with retaining the privacy. Input Design considered the following things:

* What data should be given as input?
* How the data should be arranged or coded?
* The dialog to guide the operating personnel in providing input.
* Methods for preparing input validations and steps to follow when error occur.

**OBJECTIVES**

1. Input Design is the process of converting a user-oriented description of the input into a computer-based system. This design is important to avoid errors in the data input process and show the correct direction to the management for getting correct information from the computerized system.

2. It is achieved by creating user-friendly screens for the data entry to handle large volume of data. The goal of designing input is to make data entry easier and to be free from errors. The data entry screen is designed in such a way that all the data manipulates can be performed. It also provides record viewing facilities.

3. When the data is entered it will check for its validity. Data can be entered with the help of screens. Appropriate messages are provided as when needed so that the user will not be in maize of instant. Thus the objective of input design is to create an input layout that is easy to follow

**OUTPUT DESIGN**

A quality output is one, which meets the requirements of the end user and presents the information clearly. In any system results of processing are communicated to the users and to other system through outputs. In output design it is determined how the information is to be displaced for immediate need and also the hard copy output. It is the most important and direct source information to the user. Efficient and intelligent output design improves the system’s relationship to help user decision-making.

1. Designing computer output should proceed in an organized, well thought out manner; the right output must be developed while ensuring that each output element is designed so that people will find the system can use easily and effectively. When analysis design computer output, they should Identify the specific output that is needed to meet the requirements.

2. Select methods for presenting information.

3. Create document, report, or other formats that contain information produced by the system.

The output form of an information system should accomplish one or more of the following objectives.

* Convey information about past activities, current status or projections of the
* Future.
* Signal important events, opportunities, problems, or warnings.
* Trigger an action.
* Confirm an action.

**3.3 MODULES**

1. **Authentication and authorization**
2. **Data enter**
3. **Change Password**

**Authentication and authorization**: - This Module is having three sub modules called admin,staff & Students; where the students will get registered in to application by giving the information like this name, mailed, year seem Password Etc. and Can login into application by giving this Role Number Year, Seem if he is Authentication user the application will authorization to access to the student to Access our Data The other user of our project will be admin who will have Direct authorization with a default user name & password

**Data enter**: - In this module the admin will be uploading the students Internal & external marks in to the Data base

**Change Password**- By using this Module the Admin Can change password any time in our application for any type security.

CHAPTER 4

DESIGNING

**4. DESIGING**

**4.1 UML DIAGRAM**

**CLASS DIAGRAM:**

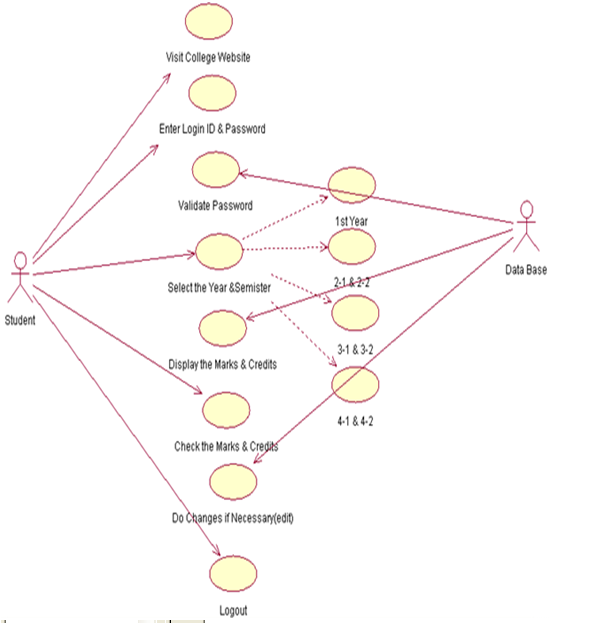
The class diagram is a static diagram. It represents the static view of an application. Class diagram is not only used for visualizing, describing and documenting different aspects of a system. The below class diagram consists of classes such as student, college& admin or database or server. These classes consist of attributes which are the details & operations which can be performed.All the class diagrams are dependent on eachother & maintain relationships.

A picture containing text, map

Description generated with high confidence

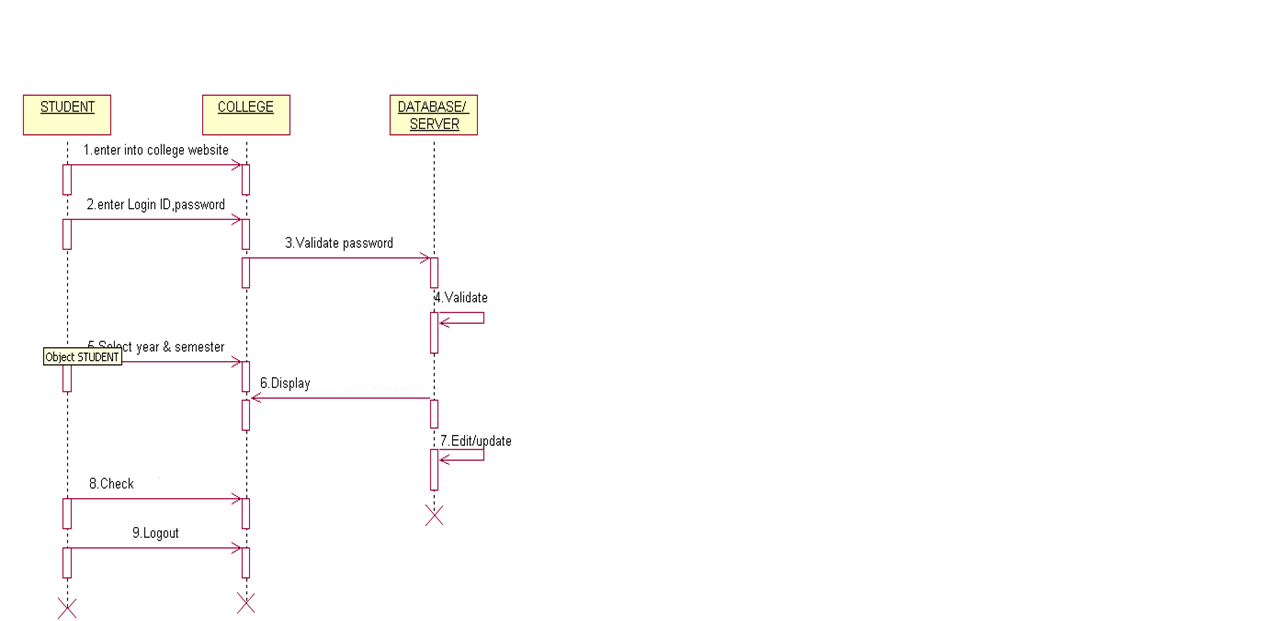
**USECASE DIAGRAM:**

**A Use case Diagram is a diagram that shows a set of Use cases and actors and their relationships. These diagrams are used to model the static use case view of a system. The below diagram explains how the operations are performed between the student &database. A student visits the site & gets logged in & checks the required marks &credits, which is displayed only if password is valid. Later the database updates if any changes are required also & logout after checking.**

****

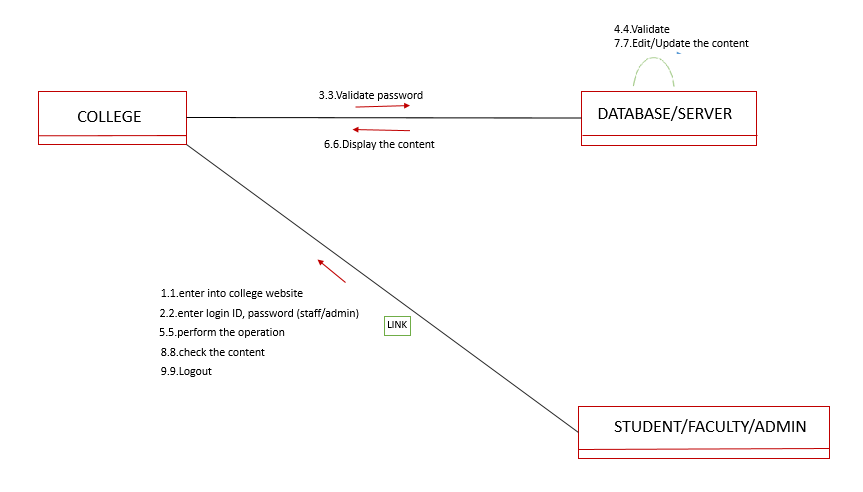
**SEQUENCE DIAGRAM:**

The first interaction diagram is sequence diagram. It shows the object organization as shown below. In the below diagram, each method is called one after the other & then perform their action or task. The methods are followed in a sequence by numbering each method one by one. The student visits the college site to check marks &credits. To open the database, the student logins with ID & password and the password is validated by the database. Later marks are checked & server keeps on updating if necessary changes have to be done & later logs out after checking the details. Once the diagram is finished a cross symbol is used to tell that task is completed or ended or finished.



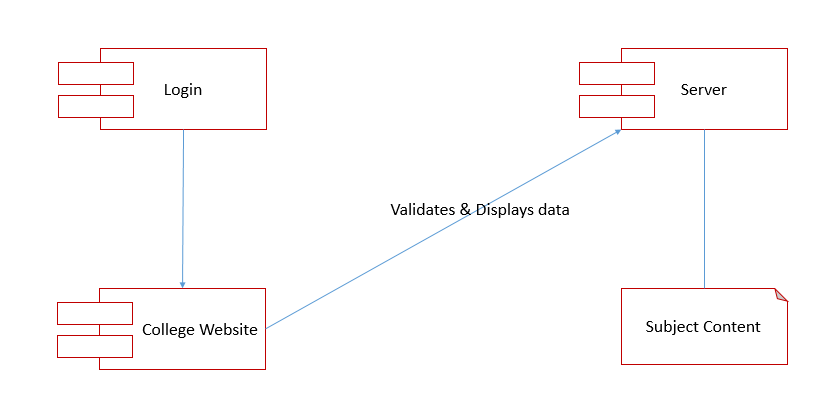
**COLLABORATION DIAGRAM:**

The second interaction diagram is collaboration diagram. It shows the object organization as shown below. The methods used in sequence are shown together by numbering. The methods are described by nodes &links. It follows same number sequence as explained in sequence diagram by calling methods one after the other.



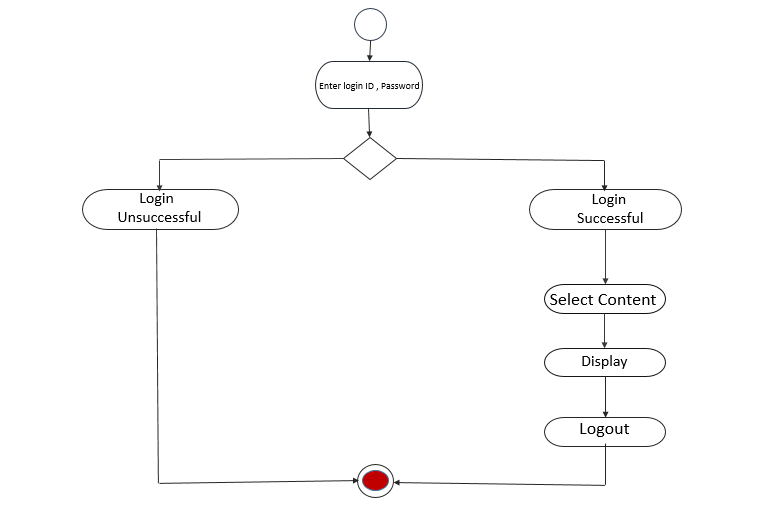
**COMPONENT DIAGRAM:**

A **component** is a structured class representing a modular part of a system with encapsulated content and whose manifestation is replaceable within its environment. The below component diagram explains the behaviour of database, & college defined in terms of provided with their required interfaces. It shows the organization & dependencies among the set of these components. It just addresses the static view of a system.



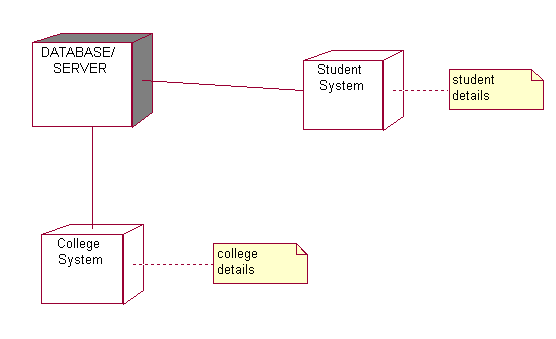
**ACTIVITY DIAGRAM:**

An activity diagram shows the flow from activity to activity .An activity is an ongoing non atomic execution within a state machine .The below activity diagram explains the activity of a student & database responding to it. The dynamic view of the system is addressed. The flow of control is maintained among the objects either in sequence or branched. If the decision of login is successful then only tasks are performed otherwise the activity is ended.

****

**DEPLOYMENT DIAGRAM:**

The Deployment Diagram also helps to model the physical aspect of an Object-Oriented software system. It models the run-time configuration in a static view and visualizes the distribution of components in an application. The below diagram explains how the student & college system depend on the database & how they perform their task. These systems contain software & hardware components & keep updating the required details.

****

CHAPTER 5

RESULTS AND DISCUSSION

**5. RESULTS AND DISCUSSION**

**5.1 Performance Analysis**

**5.1.1 System Study**

**FEASIBILITY STUDY**

The feasibility of the project is analyzed in this phase and business proposal is put forth with a very general plan for the project and some cost estimates. During system analysis the feasibility study of the proposed system is to be carried out. This is to ensure that the proposed system is not a burden to the company. For feasibility analysis, some understanding of the major requirements for the system is essential.

Three key considerations involved in the feasibility analysis are

* ECONOMICAL FEASIBILITY
* TECHNICAL FEASIBILITY
* SOCIAL FEASIBILITY

**ECONOMICAL FEASIBILITY**

This study is carried out to check the economic impact that the system will have on the organization. The amount of fund that the company can pour into the research and development of the system is limited. The expenditures must be justified. Thus the developed system as well within the budget and this was achieved because most of the technologies used are freely available. Only the customized products had to be purchased.

### TECHNICAL FEASIBILITY

This study is carried out to check the technical feasibility, that is, the technical requirements of the system. Any system developed must not have a high demand on the available technical resources. This will lead to high demands on the available technical resources. This will lead to high demands being placed on the client. The developed system must have a modest requirement, as only minimal or null changes are required for implementing this system.

**SOCIAL FEASIBILITY**

The aspect of study is to check the level of acceptance of the system by the user. This includes the process of training the user to use the system efficiently. The user must not feel threatened by the system, instead must accept it as a necessity. The level of acceptance by the users solely depends on the methods that are employed to educate the user about the system and to make him familiar with it. His level of confidence must be raised so that he is also able to make some constructive criticism, which is welcomed, as he is the final user of the system.

* + 1. **System Testing**

The purpose of testing is to discover errors. Testing is the process of trying to discover every conceivable fault or weakness in a work product. It provides a way to check the functionality of components, sub-assemblies, assemblies and/or a finished product It is the process of exercising software with the intent of ensuring that the

Software system meets its requirements and user expectations and does not fail in an unacceptable manner. There are various types of test. Each test type addresses a specific testing requirement.

**TYPES OF TESTS**

**Unit testing**

Unit testing involves the design of test cases that validate that the internal program logic is functioning properly, and that program inputs produce valid outputs. All decision branches and internal code flow should be validated. It is the testing of individual software units of the application .it is done after the completion of an individual unit before integration. This is a structural testing, that relies on knowledge of its construction and is invasive. Unit tests perform basic tests at component level and test a specific business process, application, and/or system configuration. Unit tests ensure that each unique path of a business process performs accurately to the documented specifications and contains clearly defined inputs and expected results.

**Integration testing**

Integration tests are designed to test integrated software components to determine if they actually run as one program. Testing is event driven and is more concerned with the basic outcome of screens or fields. Integration tests demonstrate that although the components were individually satisfaction, as shown by successfully unit testing, the combination of components is correct and consistent. Integration testing is specifically aimed at exposing the problems that arise from the combination of components.

**Functional test**

Functional tests provide systematic demonstrations that functions tested are available as specified by the business and technical requirements, system documentation, and user manuals.

Functional testing is centered on the following items:

**Valid Input** : identified classes of valid input must be accepted.

**Invalid Input** : identified classes of invalid input must be rejected.

**Functions**  : identified functions must be exercised.

**Output**  : identified classes of application outputs must be exercised.

**Systems/Procedures**: interfacing systems or procedures must be invoked.

Organization and preparation of functional tests is focused on requirements, key functions, or special test cases. In addition, systematic coverage pertaining to identify Business process flows; data fields, predefined processes, and successive processes must be considered for testing. Before functional testing is complete, additional tests are identified and the effective value of current tests is determined.

**System Test**

System testing ensures that the entire integrated software system meets requirements. It tests a configuration to ensure known and predictable results. An example of system testing is the configuration oriented system integration test. System testing is based on process descriptions and flows, emphasizing pre-driven process links and integration points.

**White Box Testing**

White Box Testing is a testing in which in which the software tester has knowledge of the inner workings, structure and language of the software, or at least its purpose. It is purpose. It is used to test areas that cannot be reached from a black box level.

**Black Box Testing**

Black Box Testing is testing the software without any knowledge of the inner workings, structure or language of the module being tested. Black box tests, as most other kinds of tests, must be written from a definitive source document, such as specification or requirements document, such as specification or requirements document. It is a testing in which the software under test is treated, as a black box .you cannot “see” into it. The test provides inputs and responds to outputs without considering how the software works.

**Unit Testing:**

Unit testing is usually conducted as part of a combined code and unit test phase of the software lifecycle, although it is not uncommon for coding and unit testing to be conducted as two distinct phases.

**Test strategy and approach**

Field testing will be performed manually and functional tests will be written in detail.

**Test objectives**

* All field entries must work properly.
* Pages must be activated from the identified link.
* The entry screen, messages and responses must not be delayed.

**Features to be tested**

* Verify that the entries are of the correct format
* No duplicate entries should be allowed
* All links should take the user to the correct page.

# Integration Testing

Software integration testing is the incremental integration testing of two or more integrated software components on a single platform to produce failures caused by interface defects.

The task of the integration test is to check that components or software applications, e.g. components in a software system or – one step up – software applications at the company level – interact without error.

**Test Results:** All the test cases mentioned above passed successfully. No defects encountered.

**Acceptance Testing**

User Acceptance Testing is a critical phase of any project and requires significant participation by the end user. It also ensures that the system meets the functional requirements.

**Test Results:** All the test cases mentioned above passed successfully. No defects encountered.

CHAPTER 6

CONCLUSION AND FUTURE WORK

**6. CONCLUSION AND FUTURE WORKS**

**Conclusion:**

**Present days there are so many educational tools. Learning Management System enables the students to learn with a systematically and understand subjects with a wide perspective. Because it is online it delimits the barriers and provides international accessibility.**