**“College Management System”**

**ACKNOWLEDGEMENT**

There are many people who have helped us directly or indirectly in the successful completion of our project. We would like to take this opportunity to thank one and all.

It is my sincere obligation to thank **xxxxxxx**, Head of Information Technology Department, xxxxxxxx, who has extended positive support and precious suggestions to make the project work a success.

I am very thankful to our project guide **xxxxxxxx** who has been inspiring guide and committed caretaker for his unflinching devotion. The encouragement and support by him, especially in carrying out this project motivated me to complete this project.

I would like to express deep sense of gratitude to our staff members of Dept. of Information Technology for their co-operation, which has given in the congruency to build up this project.

I would like to thank all our friends for their help and constructive criticism during my project period. Finally, I am very much indebted to my parents for their moral support and encouragement to achieve goals. I have no words to express my gratitude and still I am very thankful to my parents who have shown me this world and for every support they gave me.

**CONTENTS:**

1. **Abstract**
2. **Introduction** 
   1. **Purpose of document**
   2. **Scope for development for this project**
   3. **Definitions, Acronyms, and Abbreviations**
   4. **Overview of document**
   5. **General Description**
   6. **User view of product use**
   7. **User specific requirements**
3. **System design** 
   * 1. **Purpose**
     2. **Target audience**
     3. **Pre-requisites**
     4. **Top level interaction diagram**
     5. **Architectural strategies**
     6. **Logical view**
   1. **Data Flow Diagrams** 
      1. **UML Diagrams**
      2. **Tables**
4. **Screen shots**
5. **Testing**
6. **Conclusion**
7. **Future Enhancements**

**8.. Bibiliography/References**

1

**Abstract:**

This project is aimed at developing an Online Intranet **College Management System (CMS)** that is of importance to either an educational institution or a college. The system **(CMS)** is an Intranet based application that can be accessed throughout the institution or a specified department. This system can be used as an attendance monitoring system for the college. Students/staffs logging in may also access/search any information. Attendance and marks will be updated by staff.

This system **(C.M.S )** is being developed for an engineering college to maintain and facilitate easy access to information. For this the users need to be registered with the system after which they can access or modify data as per the permissions given to them.

**CMS** is an intranet based application that aims at providing information to all thelevels of management with in an organization. This system can be used as a knowledge/information management system for the college. For a given student/staff (technical/Non-technical) can access the system to either upload or download some information from the database.

1

**Introduction:**

The title of the project is “COLLEGE MANAGEMENT SYS TEM FOR COLLEGE” (CMS). CMS is an Intranet based application that aims at providing information to all the levels of management within an organization. This system can be used as a information management system for the college.

For a given student/staff (Technical / Non-technical) the Administrator creates login id & password, using these student/ staff (Technical / Non-technical) can access the system to either upload or download some information from the database.

The front-end will be HTML pages with Java Script for client side validation where as all business logics will be in Java reside at middle layer. And these layers will interact with third layer of database, which will be Oracle database. The web server will be Tomcat5.5. To start working on this project environment required is a server having Tomcat5.5 as web server, Oracle10g as database and Java Runtime Environment (JRE) as development environment.

The project is divided into 6 scenarios; each scenario can be developed independently. And knowledge of Java with oracle10g is desirable to execute this project.

**2.1 PURPOSE OF DOCUMENT:**

This document is the Software Requirement Specification (SRS) for the College Management System for College (CMS) project .The purpose of this document is to describe the functionality, requirements and general interface of the CMS.

**2.2 SCOPE FOR DEVELOPMENT OF THIS PROJECT:**

The requirement of the user is to:

* + Access/ Search information.
  + Login to the system through the first page of the application
  + Change the password after logging into the system
  + View/change his/her details.
  + Can get help through the help option to view different features of the system.
  + Students can give feedback on college/staff/any other student.
  + An admin login should be present who can read as well as remove any uploads

1. **DEFINITIONS ,ACRONYMS & ABBREVIATIONS:**

Browser

Click or Command Click

HTML

Java

Software used to view hypertext documents. Internet Explorer and Netscape Navigator are examples of browsers.

Refers to the user pressing a mouse button to select an object or cause some action to occur. On right-handed mice with more than one button, it usually means to press the leftmost button.

Hypertext Markup Language is a specification for graphical layout of a document. The specification calls for the document to be stored as text containing a series of tags that contain formatting information.

A programming language used to implement source code of our project.

1

User

Friendly

Oracle

JSP

Designed to be easy to use. Simple, practicable, accommodating, understandable and accessible.

Oracle database is a collection of data treated as a unit. The purpose of a database is to store and retrieve related information.

Java Server Page (JSP) is a technology for controlling the content or appearance of Web pages through the use of servlet, small programs that are specified in the Web page and run on the Web server to modify the Web page before it is sent to the user who requested it.

**2.4 OVERVIEW OF DOCUMENT:**

This document describes the product and its requirements and constraints. It provides a primarily non-technical description of the project targeted towards external audiences. This section includes information such as data requirements, functional requirements, and a general description of the product and its interaction with users from the perspective of the client. Section 3 provides specific technical requirements as the team understands them at this time and is intended for an internal audience. This section includes information such as external interface requirements, performance requirements, and any other technical requirements needed to design the software.

**2.5 GENERAL DESCRIPTION:**

**2.5.1 User Characteristics:**

The target audience for **CMS** product is the college students/staff (Technical/Non-

technical) .The users for this system are

* Administrator – The Super user of the system.
* Student – A user with limited access rights.
* Staff – A user of the system who has more access ri ghts than a normal user.

1

1. **Product Perspective:**

The product will be a standalone application and may be run on multiple systems

within an Intranet network. The product will require a keyboard, mouse and monitor to interface with the users. The minimum hardware requirements for the product are specified in this document.

**2.5.3 Overview of Functional Requirements:**

The client requires the following features:

* The administrator governs the working of the system.
* The staff can view the student’s details.
* A mechanism to uniquely identify each student
* The students can view their marks/attendance/exam schedules .
* The system should have a login.
* The system should have help feature.

**2.5.4:Non functional requirements**

*Client:*

*Hardware platform: PIII or above with*

*RAM of 128 or above MB*

*And 20GB or above of HD.*

*Software Platform: Java Enabled Browser*

*Server:*

*Hardware Platform: PIII or above with*

*RAM of 128 or above MB*

*And 20GB or above of HD.*

*SoftwarePlatform:HTML,Java,JavaScript,Oracle10g,tomcat5.0 windowsserver2003*.

1

***2.6 USER VIEW OF PRODUCT USE*:**

The front view of the system consists of different services provided by the system and a login page with which a user is prompted to login in to the system through his user name and password. Upon the student's login, his/her details are updated in to the system. When all data is entered, the student can not only view their details but also view their exams and report details. When the student's session ends, all data is saved. Staff can also login to the system with their id and password and has access to the system.

There will be a special login name and password (to prevent students in updating their details other than profile) to allow the teacher to access all student data in a table form. Here, the administrator can add students and staff to the system and change student and staff details.

1. **SPECIFIC REQUIREMENTS:** 
   * 1. **External Interface Requirements:** 
        + Simple, Attractive, User Friendly
        + Self-Contained, Consistent, Self-Explanatory
        + Robust.
   1. **Detailed Description of Functional Requirements: 2.7.2.1 Administrator Configuration Screen:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  |  |  |  |  |
|  | **Purpose** |  | Allows administrator to add/remove student/staff records |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  | **Inputs** |  | Student/staff details |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  | **Processing** |  | Add/remove student/staff entries in database |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  | **Outputs** |  | Updated records. |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  | **Admin Configuration Screen** |  |

1

**2.7.2.2 Staff Configuration Screens:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  |  |  |  |  |
|  | **Purpose** |  | Allow teacher to view student records, view his details. |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  | **Inputs** |  | Student details/his personal details |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  | **processing** |  | Updates his details. |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  | **Outputs** |  | Updated records. |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  | **Staff Configuration Screen** |  |

**2.7.2.3 A mechanism for each student to be uniquely identified:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  |  |  |  |  |
|  | **Purpose** |  | To allow each student to keep track of their own progress |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  | **Inputs** |  | Students details |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  | **processing** |  | Verify student's record exists, prohibit duplicates |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  | **Outputs** |  | Launch 'Exercise Input' screen |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  | **Student Configuration Screen** |  |

**2.7.2.4 Exercise input screen:**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  |  |  |  |  |  |
|  | **Purpose** |  | Receives student’s daily input |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  | **Inputs** |  | Which inputs; in what form/format will inputs arrive; from what sources |  |  |
|  |  | input will be derived, legal domains of each input element |  |  |
|  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  | **processing** |  | Store new data to student history |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  | **Outputs** |  | launch summary screen |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  | **Exercise Input Screen** |  |  |

1

**2.7.2.5 A mechanism to change the password:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  |  |  |  |  |
|  | **Purpose** |  | Allows to change students/staff passwords. |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  | **Inputs** |  | Previous and new password. |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  | **processing** |  | Updates the new password. |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  | **Outputs** |  | Acknowledgement. |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  | **Change Password** |  |
|  | **2.7.2.6 Help Screen Mechanism:** | | |  |
|  |  |  |  |  |
|  | **Purpose** |  | To provide help about the system. |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  | **Inputs** |  | Users query. |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  | **processing** |  | Searches the relevant information. |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  | **Outputs** |  | Result for the requested query. |  |
|  |  |  |  |  |
|  |  |  |  |  |

**Help**

**2.7.3 Main modules of the system:**

**Campus Information:**

This module gives the information about

* **Buildings/Blocks:** It contains the information about the total number of blockspresent in the campus and also the number of rooms present in each block.
* **Laboratories:** This gives the information about the number of laboratoriespresent in each department.
* **Buses:** This gives the count of the buses deployed by the management.
* **Library:** students can borrow/return and can view status of books present in thelibrary

1

**Administration**:

This module deals mainly with,

* **Admission:** This mainly deals registering the students/staff andassigning them with a login id and password.
* **Accounts:** This keeps track of the financial details of the college.
* **Hostel:** It gives information about the college hostel for both boys andgirls.
* **Bus Routes:** This maintains the route details of the buses.

**Department Information:**

This module gives the information about,

* **Course:** This contains the information about the number of the coursesoffered by the college and number of seats present in each.
* **Staff:** This contains the number of staff available in each department.
* **Infrastructure:** This has the details of the assets allotted for eachdepartment.
* **Syllabus:** This provides the academic syllabus of the students fromdifferent branches.

**Staff Information:**

This module deals mainly with,

* **Profile:** This provides personal details of the staff.
* **Attendance:** This provides the staff with his/her attendance details.
* **Salary:** This provides the staff with his/her salary details.
* **Feedback:** This feature enables the staff to provide feedbacks to themanagement.
* **View Student Details:** This provides the staff to view the studentdetails.

1

**Student Information:**

This module gives information about,

* **Profile:** This provides personal details of the student.
* **Attendance:** This provides the student with his/her attendance details.
* **Marks:** This provides the internal/external marks of a student.
* **Feedback:** This feature enables the student to provide feedbacks tothe management.
* **Remarks:** It contains the remarks written by the faculties about astudent.
* **Change Password:** Provides the student to change his/her password.

**Examination Branch:** This module contains the information about,

* **Examination Form:** This gives the information about the issue of theexamination form.
* **Results:** This contains the internal/externals marks of a student.
* **Attendance:** It maintains the attendance of all the students (in order toissue the examination form).
* **Schedule:** This has the various examinations schedules/timetables.
* **Seating Arrangement:** This has the seating arrangements for thevarious examinations.

1

**2.7.4 Performance Requirements:**

1. Number of simultaneous users per machine: one
2. Number of files: one per user.
3. File Size: grows linearly with student’s history.

**2.7.5 Quality Attributes:**

1. **Security:** This feature is provided by login id and password.
2. **Reliability:** Must maintain data integrity. Computer crashes and misuseshould not affect a user's history.
3. **Simplicity:** Must be driven by a simple user interface.

1

**SYSTEM DESIGN**

1

**3.1 PURPOSE:**

Purpose of **College Management System for a College (CMS)** Design Document is to describe the design and the architecture of **CMS**. The design is expressed in sufficient detail so as to enable all the developers to understand the underlying architecture of **CMS.**

Logical architecture of **JDBC** driver, Server, **DML**, **DDL**, Session and Data Store are explained.

**3.2 TARGET AUDIENCE:**

This Design document is intended to act as a technical reference tool for developers involved in the development of **College Management System (CMS)**.

This document assumes that you have sufficient understanding of the following

**Concepts:**

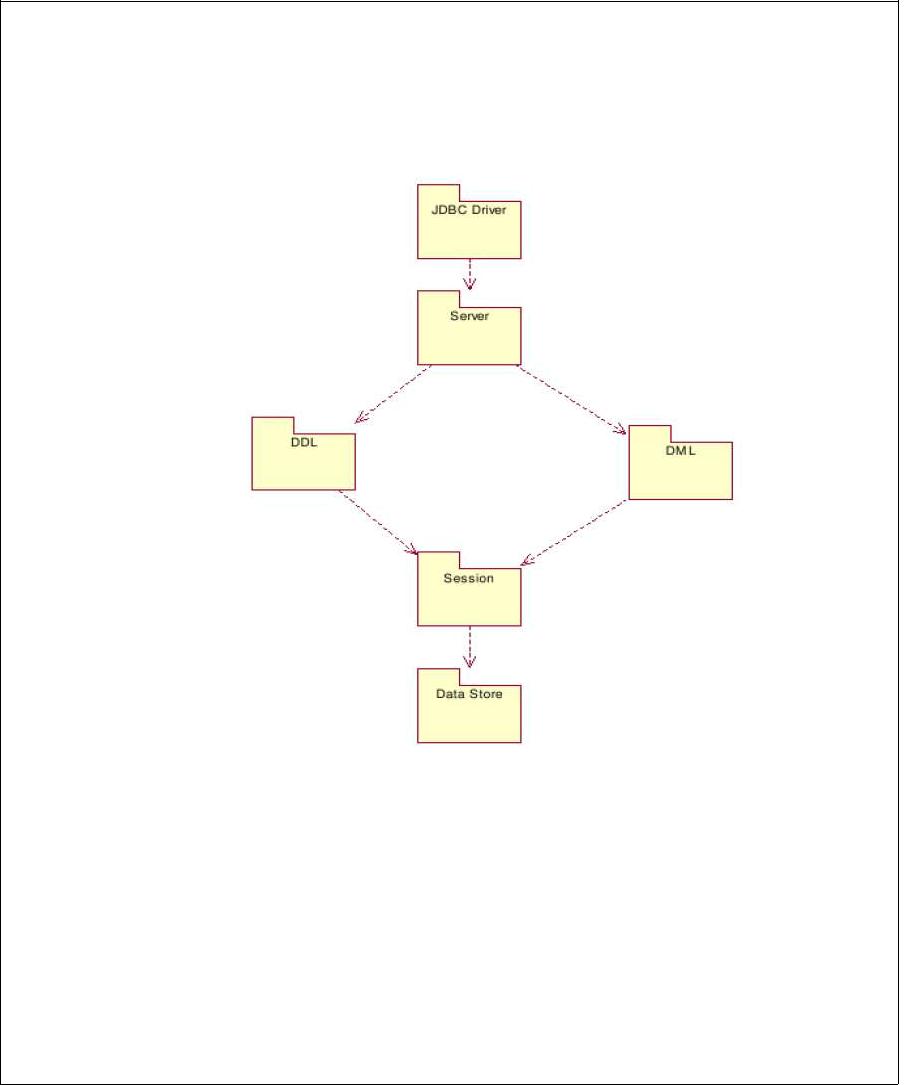
* RDBMS and its various component modules.
* SQL
* Java and JDBC
* Interaction Diagrams
* Classes and Interfaces

**3.3 PRE-REQUISITES:**

**CMS** requires Java **JRE** 1.5 or higher. Since **CMS** is written in Java, it can run onany platform that supports the Java runtime environment 1.5 or higher. The compiled files are contained in Java Archives (**JAR’**s) and have to be defined in the CLASSPATH environment variable.

**3.4 TOP LEVEL INTERACTION DIAGRAM:**

1



Logical architecture of **JDBC** driver, Server, **DML**, **DDL**, Session and Data Store

**Interaction Diagram**

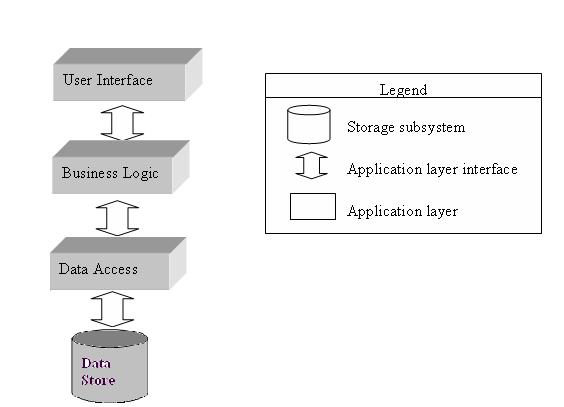
**3.5 ARCHITECTURAL STRATEGIES:**

1

The architectural design of a software project is simply the design of the entire software system. This includes the hierarchy of the modules and also which modules are present in the system. A good architectural design will create a clear and fair balance between cohesion (each module has only one distinct purpose), coupling (no two modules depend completely on each other), abstraction (seeing modules in full and not in detail), hierarchy (logical modules stem from others) and partitioning (logically grouping modules together) of the software modules.

**3.6. Logical View:**

it provides the user with an abstract view of the overall system functionality. .



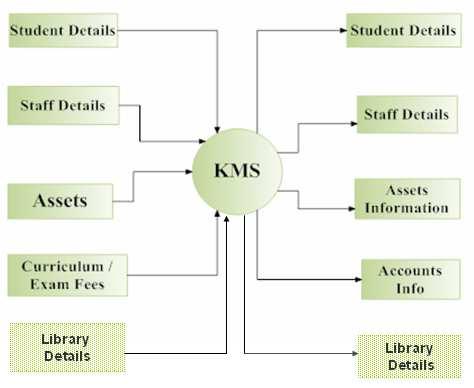
**Abstract view of CMS**

1

**3.7..Data flow diagrams:**

The whole architectural structure stems from the original flow design. Below are the Level 0, Level 1, Level 2 and Level3 Data Flow Diagrams (**DFD**).. The diagrams below more accurately portray the data flow through our system. These diagrams take precedence over the diagrams in the requirements document.

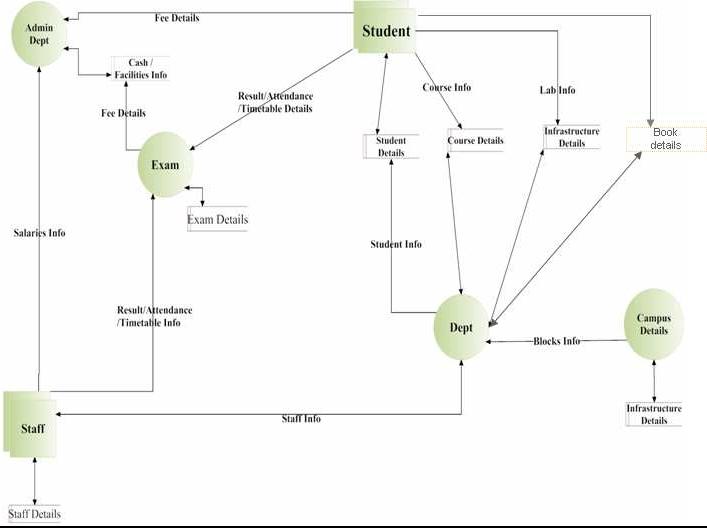
**Level 0:**



**Level 0 DFD**

1

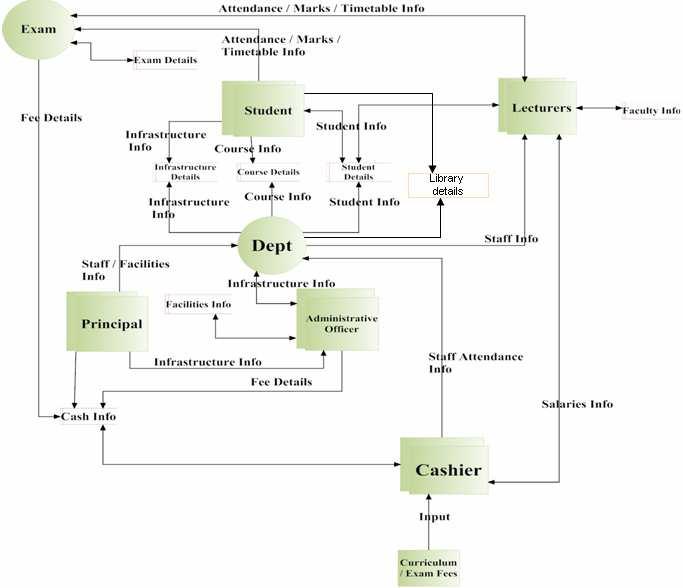
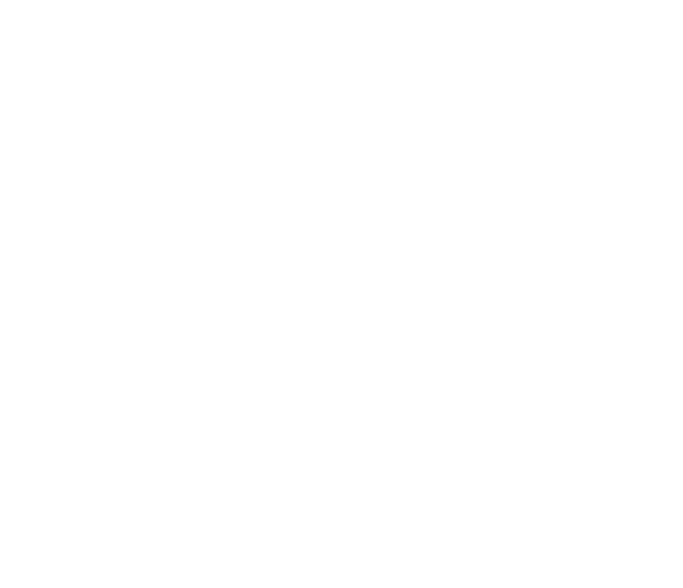
**Level 1:**



**Level 1 DFD**

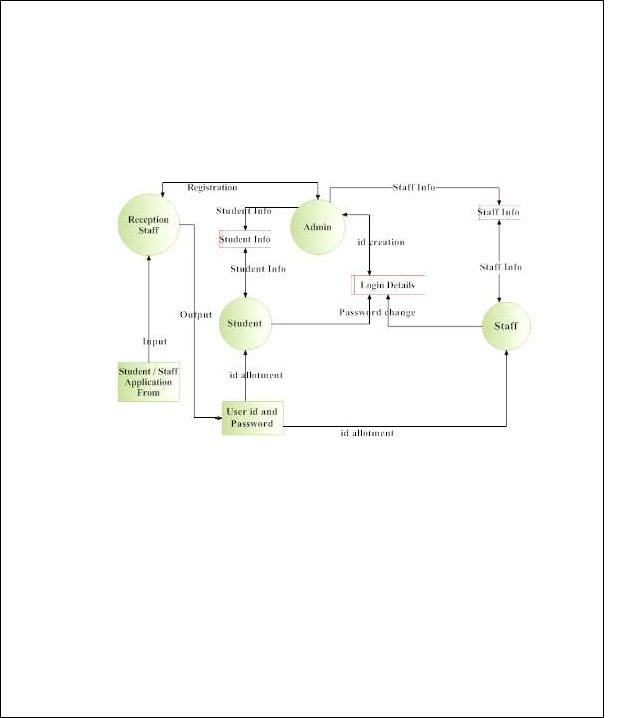
1

**Level 2 for Department:**



**Level 2 DFD for Department**

1



**Level 2 for Admin:**

**Level 2 DFD for Admin**

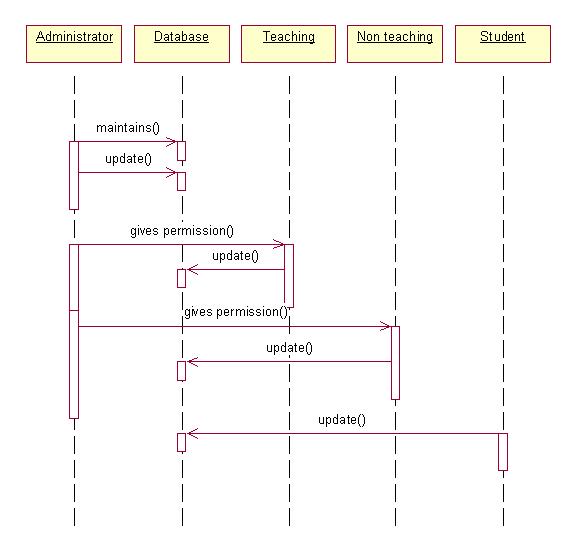
1

**3.8. UML diagrams:**

**Logical View:**

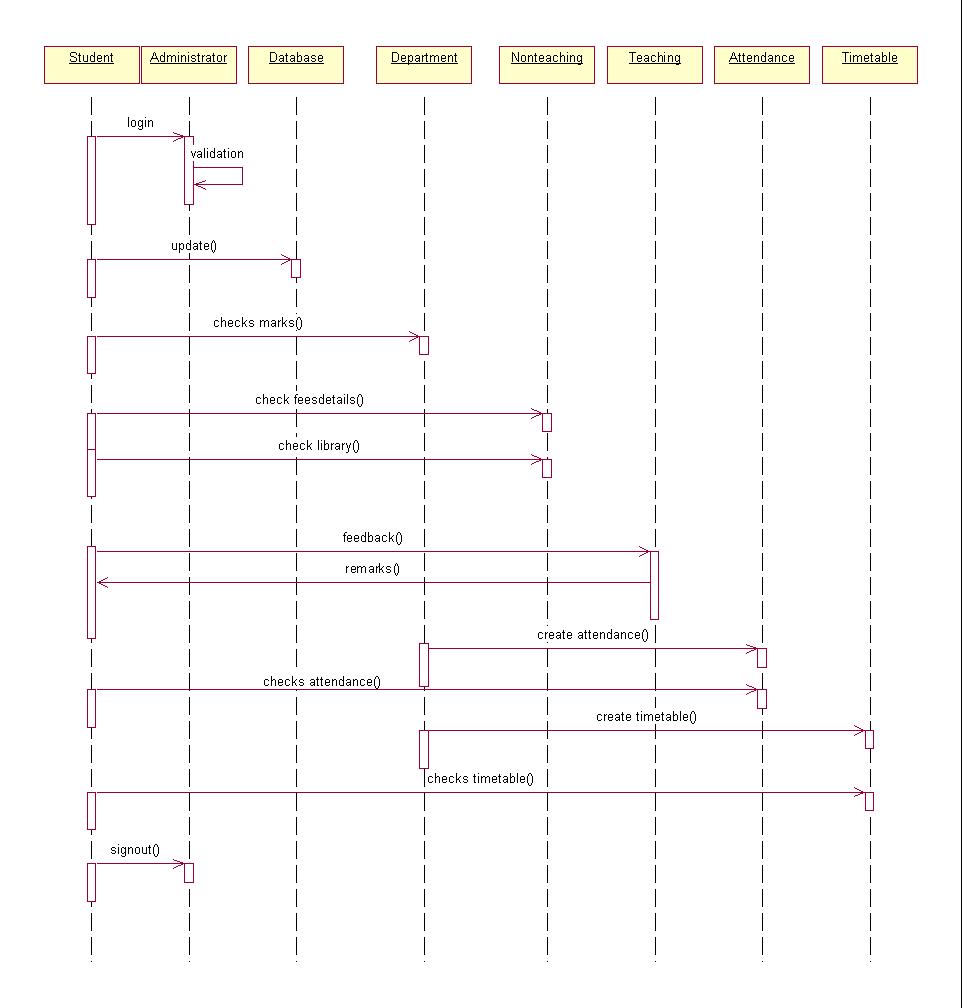
**Sequence Diagram:**

**Administrator:**



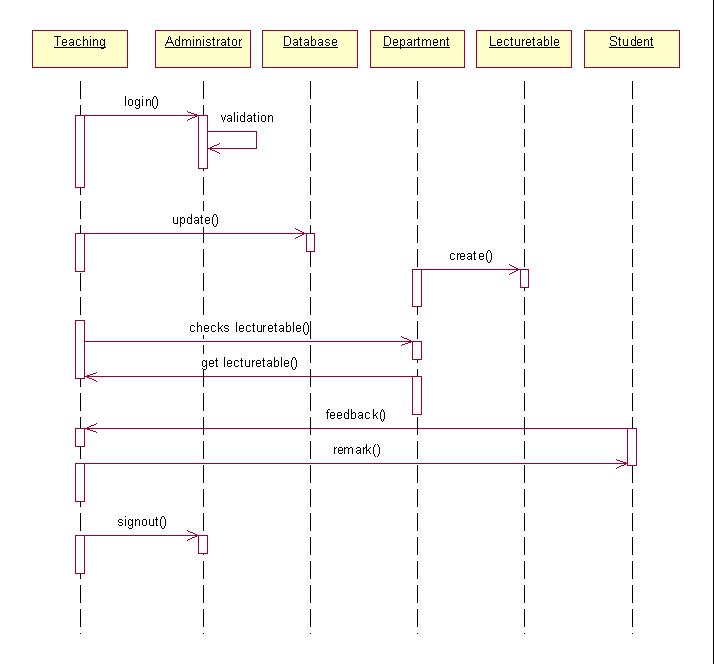
1

**Student:**



1

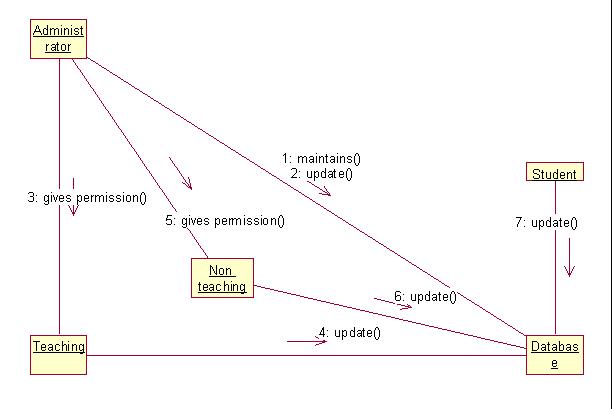
**Staff:**



1

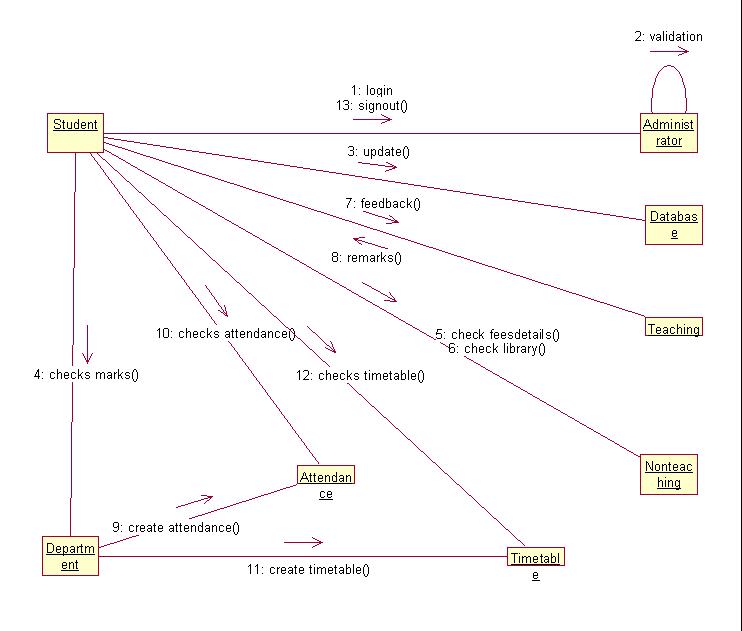
**Collaborations Diagram:**

**Administrator:**



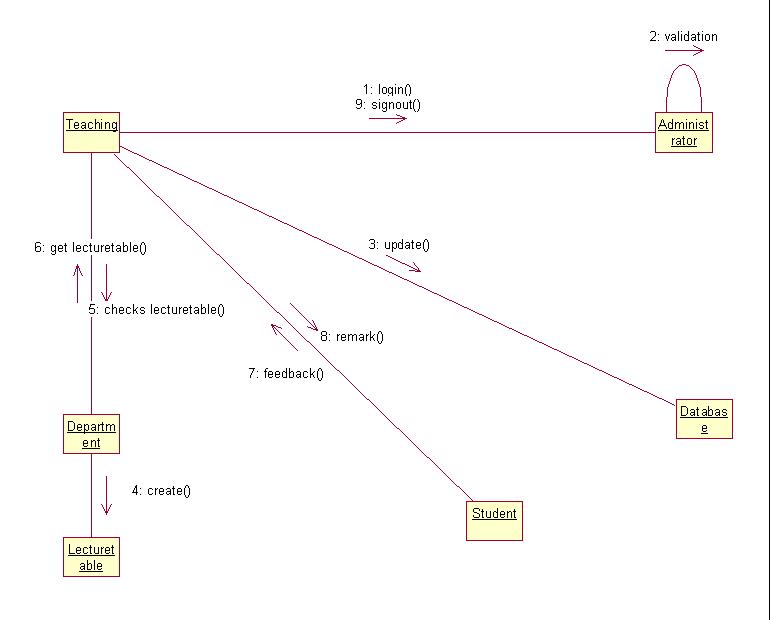
1

**Student:**



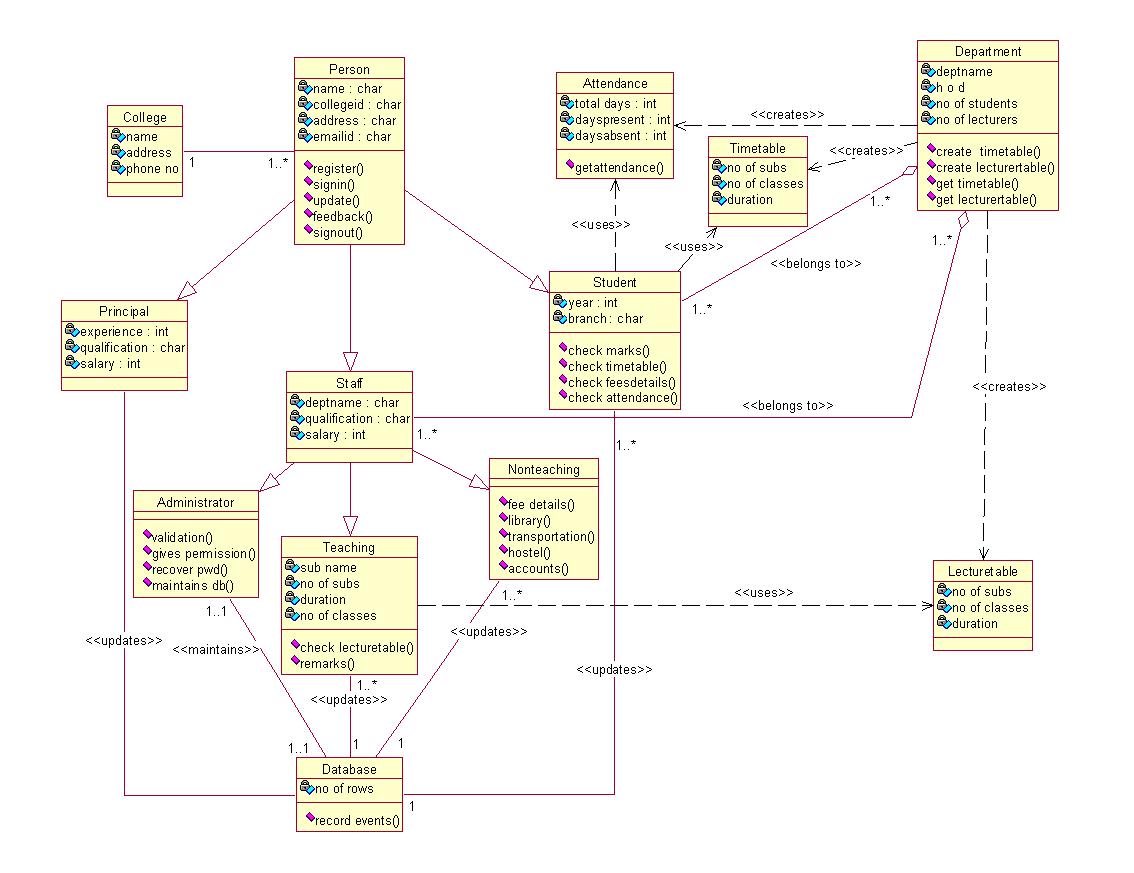
1

**Staff:**



1

**Class Diagram:**

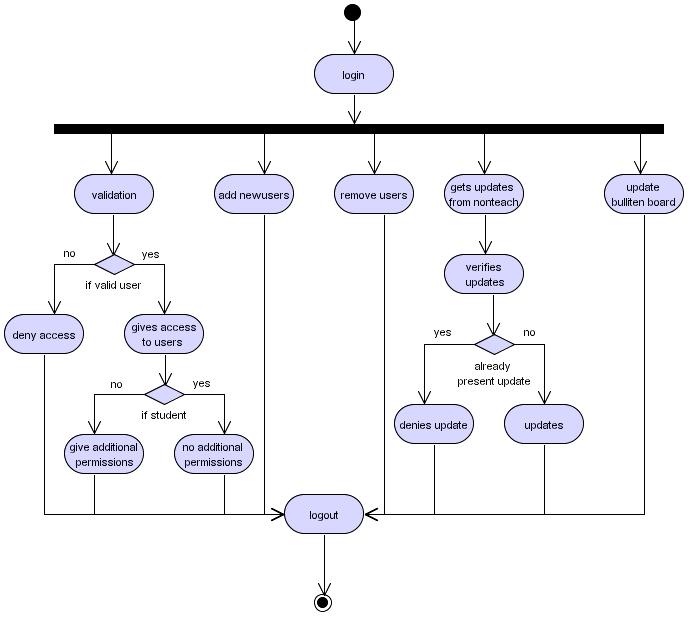


**Class Diagram**

1

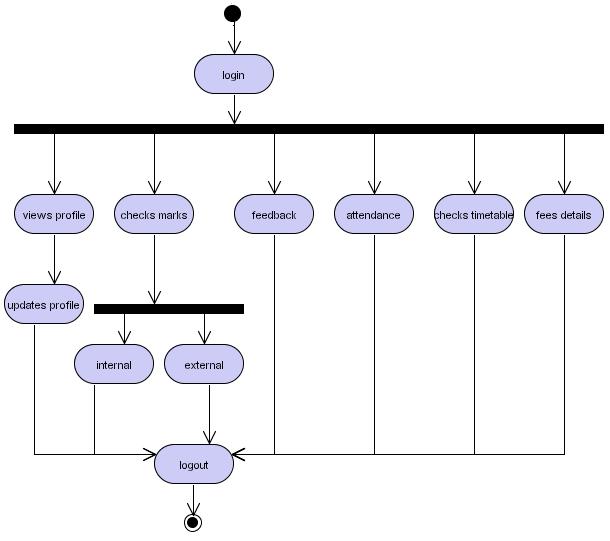
**Activity Diagrams:**

**Administrator:**



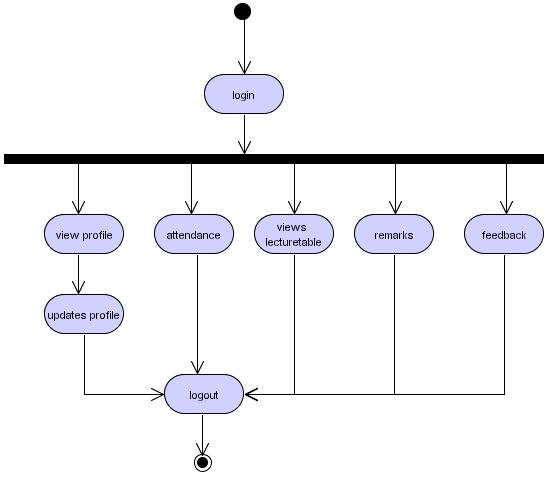
1

**Student:**



1

**Staff:**



1

**3.9.Tables:**

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Table no: | |  |  | 1 |  |  |  |  |
|  |  |  |  |  |  |  |  |  |
| Table Name: | |  |  | Users | |  |  |  |
|  |  |  |  |  |  | |  |  |
| Table Description: | |  |  | Contains usernames and passwords of all | | | | |
|  |  |  |  | the users in the system. | | | | |
|  |  |  |  |  |  |  |  |  |
| Seq no | Column name | Column description | Column | |  | PK/FK? | Null/Not | Remarks |
|  |  |  | type | |  |  | Null |  |
|  |  |  |  | |  |  |  |  |
| 1 | User name | User name | VarChar | |  | PK | Not Null |  |
|  |  |  |  | |  |  |  |  |
| 2 | Password | Password | VarChar | |  |  |  |  |
|  |  |  |  | |  |  |  |  |
| 3 | Designation | Admin/student/staff | varChar | |  |  |  |  |
|  |  |  |  |  |  |  |  |  |

**Users**

**Relationships: None**

**Constraints:**

* **Key Constraints:** username is the primary key and must be unique.
* **Referential Integrity:** N/A
* **Check Constraints:** None
* **Domain Constraints:** None

**Expected volume:** None

1

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Table No: |  |  |  | 2 |  |  |  |
|  |  |  |  |  |  |  |  |
| Table Name: |  |  |  | Student | |  |  |
|  |  |  |  |  |  |  |  |
| Table Description: | |  |  | Contains all the details of students i.e student | | | |
|  |  |  |  | name, address, phone number, etc. | | | |
|  |  |  |  |  |  |  |  |
| Seq no | Column | Column | Column | | PK/FK? | Null/Not | Remarks |
|  | Name | Description | Type | |  | Null |  |
|  |  |  |  | |  |  |  |
| 1 | Studentid | User name of | Varchar | | PK | Not Null |  |
|  |  | the student |  |  |  |  |  |
|  |  |  |  | |  |  |  |
| 2 | Firstname | First name of | Varchar | |  |  |  |
|  |  | the student |  |  |  |  |  |
|  |  |  |  | |  |  |  |
| 3 | Lastname | Last name of | Varchar | |  |  |  |
|  |  | the student |  |  |  |  |  |
|  |  |  |  | |  |  |  |
| 4 | Fathersname | Student’s | Varchar | |  |  |  |
|  |  | father name |  |  |  |  |  |
|  |  |  |  | |  |  |  |
| 5 | Address | Student’s | Varchar | |  |  |  |
|  |  | permanent |  |  |  |  |  |
|  |  | address |  |  |  |  |  |
|  |  |  |  | |  |  |  |
| 6 | Gender | Sex of the | Varchar | |  |  |  |
|  |  | student i.e M/F |  |  |  |  |  |
|  |  |  |  | |  |  |  |
| 7 | Dateofbirth | Date of Birth | Varchar | |  |  |  |
|  |  |  |  | |  |  |  |
| 8 | Dateofjoin | Joining date | Varchar | |  |  |  |
|  |  |  |  | |  |  |  |
| 9 | Course | CSE/ECE/EEE | Varchar | |  |  |  |
|  |  |  |  | |  |  |  |
| 10 | Phoneno | Phone number | Varchar | |  |  |  |
|  |  |  |  |  |  |  |  |

1

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 11 |  | Email | e-mail id | Varchar |  |  |  |
|  |  |  |  |  |  |  |  |
| 12 |  | Cas | Caste – | Varchar |  |  |  |
|  |  |  | OC/SC/BC/ST |  |  |  |  |
|  |  |  |  |  |  |  |  |
| 13 |  | Religion | Religion | Varchar |  |  |  |
|  |  |  |  |  |  |  |  |
| 14 |  | Nation | Nationality | Varchar |  |  |  |
|  |  |  |  |  |  |  |  |
| 15 |  | Idmarks | Identification | Varchar |  |  |  |
|  |  |  | marks |  |  |  |  |
|  |  |  |  |  |  |  |  |
| 16 |  | Designation | student | Varchar |  |  |  |
|  |  |  |  |  |  |  |  |
| 17 |  | Password | Password | Varchar |  |  |  |
|  |  |  |  |  |  |  |  |
| 18 |  | Image | URL of the | Varchar |  |  |  |
|  |  |  | students photo |  |  |  |  |
|  |  |  |  |  |  |  |  |
|  |  |  | **Student** | | | | |
|  | **Relationships: None** | |  |  |  |  |  |

**Constraints:**

* **Key Constraints:** student id is the primary key and must be unique.
* **Referential Integrity:** N/A
* **Check Constraints:** None
* **Domain Constraints:** None

**Expected volume:** None

1

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Table No: |  |  |  | 3 |  |  |  |
|  |  |  |  |  |  |  |  |
| Table Name: |  |  |  | Staff |  |  |  |
|  |  |  |  |  |  |  |  |
| Table Description: | |  |  | Contains all the details of staff i.e. staff name, | | | |
|  |  |  |  | address, phone number, etc. | | |  |
|  |  |  |  |  |  |  |  |
| Seq no | Column | Column | Column | | PK/FK? | Null/Not | Remarks |
|  | Name | Description | Type | |  | Null |  |
|  |  |  |  | |  |  |  |
| 1 | employeeid | User name of | Varchar | | PK | Not Null |  |
|  |  | the staff |  |  |  |  |  |
|  |  |  |  | |  |  |  |
| 2 | Firstname | First name of | Varchar | |  |  |  |
|  |  | the staff |  |  |  |  |  |
|  |  |  |  | |  |  |  |
| 3 | Lastname | Last name of | Varchar | |  |  |  |
|  |  | the staff |  |  |  |  |  |
|  |  |  |  | |  |  |  |
| 4 | Address | Staff’s | Varchar | |  |  |  |
|  |  | permanent |  |  |  |  |  |
|  |  | address |  |  |  |  |  |
|  |  |  |  | |  |  |  |
| 5 | Gender | Sex of the staff | Varchar | |  |  |  |
|  |  | i.e M/F |  |  |  |  |  |
|  |  |  |  | |  |  |  |
| 6 | Dateofbirth | Date of Birth | Varchar | |  |  |  |
|  |  |  |  | |  |  |  |
| 7 | Dateofjoin | Joining date | Varchar | |  |  |  |
|  |  |  |  | |  |  |  |
| 8 | Phoneno | Phone number | Varchar | |  |  |  |
|  |  |  |  | |  |  |  |
| 9 | Email | e-mail id | Varchar | |  |  |  |
|  |  |  |  | |  |  |  |
| 10 | Qualification | Qualification | Varchar | |  |  |  |
|  |  |  |  | |  |  |  |
| 11 | Experience | Experience | Varchar | |  |  |  |
|  |  |  |  |  |  |  |  |

1

|  |  |  |  |
| --- | --- | --- | --- |
| 12 | Cas | Caste – | Varchar |
|  |  | OC/SC/BC/ST |  |
|  |  |  |  |
| 13 | Religion | Religion | Varchar |

|  |  |  |  |
| --- | --- | --- | --- |
| 14 | Nation | Nationality | Varchar |

|  |  |  |  |
| --- | --- | --- | --- |
| 15 | Idmarks | Identification | Varchar |
|  |  | Marks |  |
|  |  |  |  |
| 16 | Designation | Staff | Varchar |

|  |  |  |  |
| --- | --- | --- | --- |
| 17 | Password | Password | Varchar |

|  |  |  |  |
| --- | --- | --- | --- |
| 18 | Image | URL of the | Varchar |
|  |  | staff’s photo |  |

**Staff**

**Relationships: None**

**Constraints:**

**Key Constraints:** employee id is the primary key and must be unique.

* **Referential Integrity:** N/A
* **Check Constraints:** None
* **Domain Constraints:** None

**Expected volume:** None

1

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Table No: |  |  |  | 4 |  |  |  |
|  |  |  |  |  |  |  |  |
| Table Name: | |  |  | feedback | |  |  |
|  |  |  |  |  |  |  |  |
| Table Description: | |  |  | Contains all the feedback given by the staff | | | |
|  |  |  |  |  |  |  |  |
| Seq No | Column | Column Description | Column | | PK/Fk? | Null/Not | Remarks |
|  | Name |  | Type | |  | Null |  |
|  |  |  |  | |  |  |  |
| 1 | fromuserid | Sending userid | Varchar | |  |  |  |
|  |  |  |  | |  |  |  |
| 2 | Feedbackon | College/Staff/students | Varchar | |  |  |  |
|  |  |  |  | |  |  |  |
| 3 | Comments | Comments | text | |  |  |  |
|  |  |  |  | |  |  |  |
| 4 | Touserid | Destinationuserid | Varchar | |  |  |  |
|  |  |  |  | |  |  |  |
| 5 | Pdate | Current date | Date | |  |  |  |
|  |  |  |  |  |  |  |  |

**Feedback**

**Relationships:** users

**Constraints:**

* **Key Constraints:** userid is the primary key and must be unique.
* **Referential Integrity:** user id match with userid in users table
* **Check Constraints:** None
* **Domain Constraints:** None

**Expected volume:** None

1

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Table No: |  |  |  | 5 |  |  |  |
|  |  |  |  |  |  |  |  |
| Table Name: |  |  |  | Seating |  |  |  |
|  |  |  |  |  |  |  |  |
| Table Description: | |  |  | Contains all the results the students. | | | |
|  |  |  |  |  |  |  |  |
| Seq No | Column | Column | Column | | PK/FK? | Null/Not | Remarks |
|  | Name | Description | Type | |  | Null |  |
|  |  |  |  | |  |  |  |
| 1 | Roomno | Room number | Int | |  |  |  |
|  |  |  |  | |  |  |  |
| 2 | Studentrange | Student | Varchar | |  |  |  |
|  |  | number range |  |  |  |  |  |
|  |  |  |  | |  |  |  |
| 3 | Branch | Student | Varchar | |  |  |  |
|  |  | Branch |  |  |  |  |  |
|  |  |  |  |  |  |  |  |

**Seating**

**Relationships:** none

**Constraints:**

* **Key Constraints:** none
* **Referential Integrity:** none
* **Check Constraints:** None
* **Domain Constraints:** None

**Expected volume:** None

1

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Table No: |  |  |  | 6 |  |  |  |
|  |  |  |  |  |  |  |  |
| Table Name: |  |  |  | Results |  |  |  |
|  |  |  |  |  |  |  |  |
| Table Description: | |  |  | Contains all the results the students. | | | |
|  |  |  |  |  |  |  |  |
| Seq No | Column | Column | Column | | PK/Fk? | Null/Not | Remarks |
|  | Name | Description | Type | |  | Null |  |
|  |  |  |  | |  |  |  |
| 1 | studentid | User id | Varchar | | FK |  |  |
|  |  |  |  | |  |  |  |
| 2 | Internal | Internal Marks | Varchar | |  |  |  |
|  |  |  |  | |  |  |  |
| 3 | External | External Marks | Varchar | |  |  |  |
|  |  |  |  |  |  |  |  |

**Results**

**Relationships:** users

**Constraints:**

* **Key Constraints:** studentid is the Foreign key.
* **Referential Integrity:** student id match with userid in users table
* **Check Constraints:** None
* **Domain Constraints:** None

**Expected volume:** None

1

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Table No: |  |  |  |  | 7 |  |  |  |
|  |  |  |  |  |  |  |  |  |
| Table Name: |  |  |  |  | Attendance | | | |
|  |  |  |  |  |  |  |  |  |
| Table Description: | |  |  |  | Contains the attendance of the students/staff. | | | |
|  |  |  |  |  |  |  |  |  |
| Seq No | Column | Column |  | Column | | PK/FK? | Null/Not | Remarks |
|  | Name | Description |  | Type | |  | Null |  |
|  |  |  |  |  | |  |  |  |
| 1 | userid | User id |  | Varchar | | FK |  |  |
|  |  |  |  |  | |  |  |  |
| 2 | attendance | Attendace |  | Varchar | |  |  |  |
|  |  | percentage |  |  |  |  |  |  |
|  |  |  |  |  | |  |  |  |
|  |  |  | **Attendance** | | |  |  |  |

**Relationships:** users

**Constraints:**

* **Key Constraints:** userid is the Foreign key.
* **Referential Integrity:** userid match with userid in users table
* **Check Constraints:** None
* **Domain Constraints:** None

**Expected volume:** None

1

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Table No: | |  |  |  | 8 |  |  |  |
|  |  | |  |  |  |  |  |  |
| Table Name: | | |  |  | Accounts | | | |
|  |  | |  |  |  |  |  |  |
| Table Description: | | |  |  | Contains the fee details of the students. | | | |
|  |  |  |  |  |  |  |  |  |
| Seq no |  | Column | Column Description | Column | | PK/FK? | Null/Not | Remarks |
|  |  | Name |  | Type | |  | Null |  |
|  |  |  |  |  | |  |  |  |
| 1 |  | Studentid | User name of the staff | Varchar | | FK |  |  |
|  |  |  |  |  | |  |  |  |
| 2 |  | Firstname | First name of the staff | Varchar | |  |  |  |
|  |  |  |  |  | |  |  |  |
| 3 |  | Lastname | Last name of the staff | Varchar | |  |  |  |
|  |  |  |  |  | |  |  |  |
| 4 |  | Course | CSE/EEE/ECE/ME | Varchar | |  |  |  |
|  |  |  |  |  | |  |  |  |
| 5 |  | Payfor | Nature of payment | Varchar | |  |  |  |
|  |  |  | (bus, College,..) |  |  |  |  |  |
|  |  |  |  |  | |  |  |  |
| 6 |  | Yearofpay | Which year of the | Varchar | |  |  |  |
|  |  |  | course |  |  |  |  |  |
|  |  |  | (I/II/III/IV) |  |  |  |  |  |
|  |  |  |  |  | |  |  |  |
| 7 |  | Modeofpay | Mode of payment | Varchar | |  |  |  |
|  |  |  | (DD/Cheque) |  |  |  |  |  |
|  |  |  |  |  | |  |  |  |
| 8 |  | Amount | Amount Paid | Varchar | |  |  |  |
|  |  |  |  |  |  |  |  |  |

**Accounts**

**Relationships:** users

**Constraints:**

* **Key Constraints:** studentid is the Foreign key.
* **Referential Integrity:** studentid match with userid in users table
* **Check Constraints:** None
* **Domain Constraints:** None

**Expected volume:** None

1

**Test cases:**

**Test case 1: Authentication**

Test case#: validity of name and password/priority (H, L)-High Test objective: To check the name of the user

Test description: To check whether user name and password is valid or not Requirements verified: valid user

Test Environment: Tomcat server, database.

|  |  |  |  |
| --- | --- | --- | --- |
| **Test setup/reconditions:** | | |  |
|  | **Actions** |  | **Expected results** |
| 1. | Correct name and pwd entered | | The user will be logged in |
| 2. | Wrong name and pwd entered | | Generates an error report |
| Pass: yes | | Condition pass: ………. Fail | : …………….. |
| Problems/issues: None | | |  |
| Notes: successfully tested and executed. | | |  |

**Test case 2: e-mail verification**

Test case#: validation of e-mail/priority (H, L)-High Test objective: validity check for e-mail

Test description: To check whether the e-mail is valid or not Requirements verified: valid e-mail

Test Environment: Tomcat server, database.

|  |  |  |
| --- | --- | --- |
| **Test setup/reconditions:** | |  |
|  | **Actions** | **Expected results** |
| 1. | Correct e-mail entered | validation cleared |
| 2. | Wrong e-mail entered | displays an error prompt |

1

Pass: yes Condition pass: ………. Fail: ……………..

Problems/issues: None

Notes: successfully tested and executed.

**Test case 3: Date verification**

Test case#: validation of date/priority (H, L)-High Test objective: validity check for date

Test description: To check whether the given date is valid or not Requirements verified: valid date

Test Environment: Tomcat server, database.

**Test setup/reconditions:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | **Actions** |  | **Expected results** | |
| 1. | Correct date entered | | validation cleared | |
| 2. | Wrong date entered | | displays an error prompt | |
| Pass: yes | | Condition pass: ………. | Fail: … | ………….. |
| Problems/issues: None | | |  |  |
| Notes: successfully tested and executed. | | |  |  |

1

**Conclusion:**

The project entitled as **College Management System** is the system that deals with the

issues related to a particular institution.

This project is successfully implemented with all the features mentioned in system requirements specification.

The application provides appropriate information to users according to the chosen service.

The project is designed keeping in view the day to day problems faced by a college. Deployment of our application will certainly help the college to reduce unnecessary

wastage of time in personally going to each department for some information.

Awareness and right information about any college is essential for both the development of student as well as faculty. So this serves the right purpose in achieving the desired requirements of both the communities.

1

**Future Enhancement:**

Online examination module would be introduced to conduct online examination.

Scheduling of the staff. i.e. , time table setting of the staff

Further, the faculty can upload the videos of their lectures on to this site and students who had missed those classes can view those videos.

1

**References:**

1. Internet & World Wide Web: How to Program Deitel, PJ Deitel.
2. Web Development with Java Server Pages BY Duane K.Fields and Mark A.Kolb.
3. The Complete Refernce Java2 HerbertSchildt.
4. Core Servlets and Java Server Pages By Marty Hall.
5. Apache Jakarta-Tomcat by James Goodwill.
6. Practical PostgreSQL by John Worsley, Joshua Drake

Web Sites: www.sves-srpt.ac.in

www.kings.cam.ac.uk

www.wellington-college.school.nz