

# College Management Software

CS814 Course Project Report

Submitted by

SAMYAK JAIN (Roll No. 202IS022)

KARMANYA GAJANAN RAUT (Roll No. 202IS013)



Department of Computer Science and Engineering  
N.I.T. Karnataka  
P.O. Srinivasnagar, Surathkal,  
Mangalore-575025  
Karnataka India

**January, 2021**

## **Contents**

CHAPTER 1: INTRODUCTION	3
CHAPTER 2: AUTHORIZATION MECHANISM	5
1. CLASS DESIGN	6
2. USE CASE DIAGRAM	7
CHAPTER 3: CONCLUSION	8
CHAPTER 4: REFERENCES	9

# 1. INTRODUCTION

The management system “**College Management software**” highlights key information about the college to the viewers . The System provides the users with an interface that helps the user use the software efficiently.It contains informative stuffs such as academics faculties info, year wise placement stats, current ongoing r&d projects, mou with companies,etc. Admin has access to manage all information flow in the management software. Admin is able to dynamically add the information. Students are only permitted to view the college information. Special interface for users ,professors and head of department is provided . So overall the management software loads all the dynamic contents quickly to enhance user experience.

## **Types of Users**

- 1) Students
- 2) Professors
- 3) Head of department
- 4) Administrator

## **Functionalities of Users**

### **1) Students :**

- Students can view the information such as placements stats, any new announcement,course structure and faculty info .
- They can also update their profile on the portal.

### **2) Professors :**

- Professors can view the information such as placements stats, any new announcement, course structure and other faculty info .
- They can also update their profile as well as course information thorough the portal.
- They can also view the profiles of students.

### **3) Head of department :**

- HOD has the same privilege as professors with additional permission to add any new courses to the existing course structure .
- They monitor the ongoing courses on the portal

4) **Administrator :**

- Admin has authority to delete any course from the existing course structure .
- .He/she can also edit the information such as College info,contact details,faculty info ,etc to the website.
- Overall admin manages all the information on the portal

## **2. AUTHORIZATION MECHANISM**

### **Need for RBAC based authorization:**

The College management software should allow access to only authorized users who have active sessions and correct credentials. It also has varying levels of access based on their roles and responsibilities. When the user data increases, it becomes very difficult to manage the database of users with their roles and responsibilities. So there arises the possibility of leak of sensitive information to unauthorized users. This is where Role based Access control Mechanism comes into play. It is a mechanism that restricts system access. It involves setting permissions and privileges to enable access to authorized users. Most large organizations use role-based access control to provide their employees with varying levels of access based on their roles and responsibilities. This protects sensitive data and ensures users can only access information and perform actions they need to do their jobs.

### **Components of RBAC:**

RBAC is implemented using the database approach like there are three major components.

- User - Role Relationship
- Permission - Role Relationship
- Role -Role Relationship (Role Hierarchy)

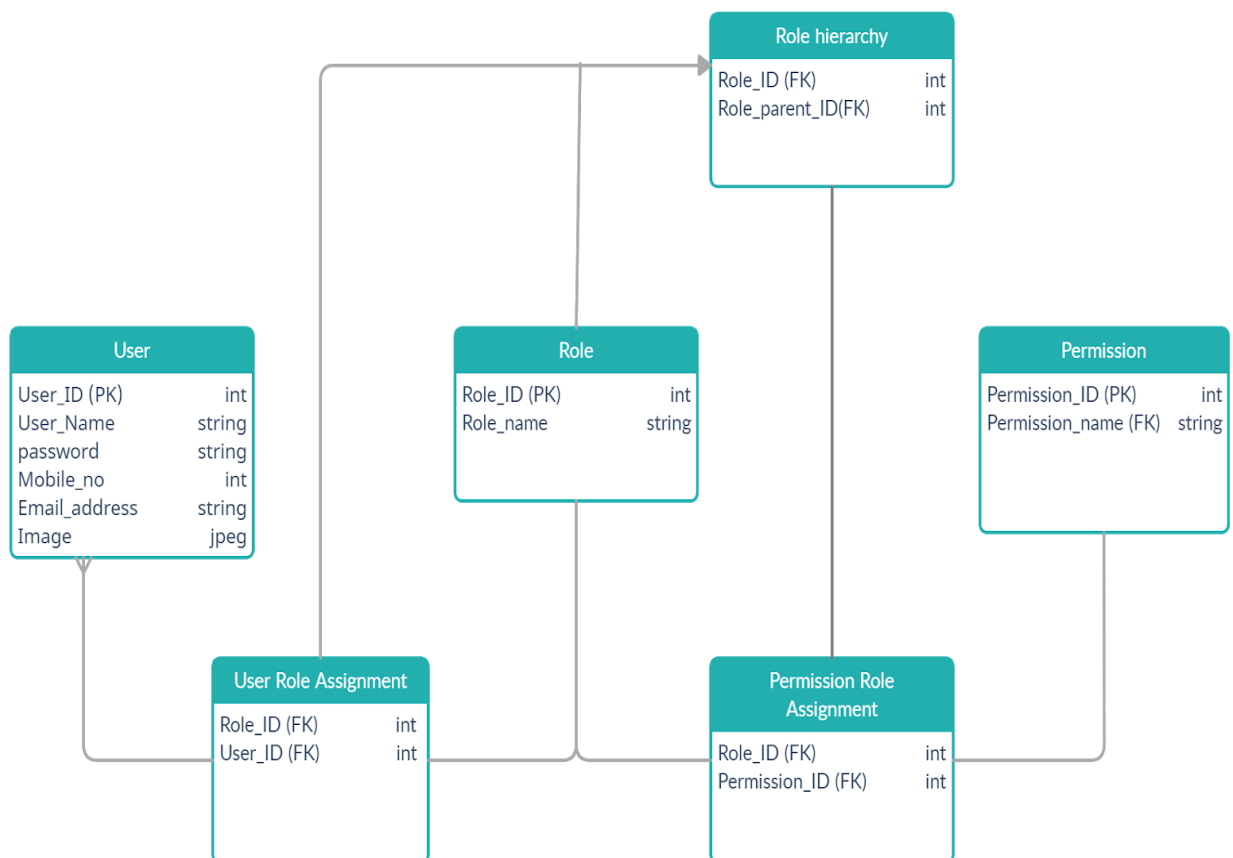
### **Components of Administrative model :**

Administration of RBAC allows the administrator to create, delete any users, roles and permission. In this application, permission for Administration of RBAC has been given to admin roles. Admin can define new roles, user or permission. He/she can also be able to role hierarchy, constraints, etc.

## Class Diagram

In the following class diagram which is consist of six tables shows the relationship between them:

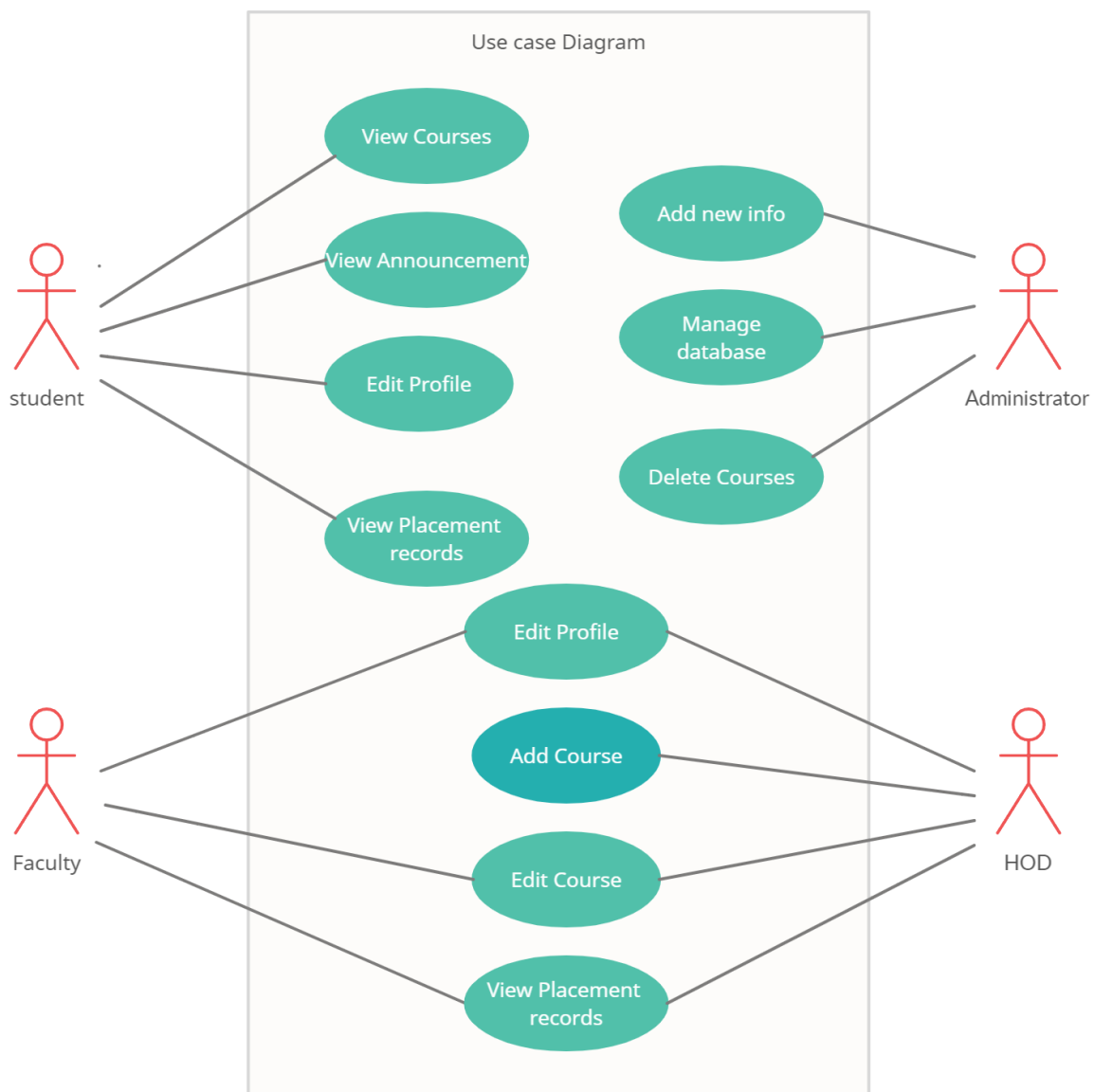
- User
- Role
- Permission
- User-Role assignment
- Role hierarchy
- Permission-Role assignment



## Use Case Diagram

In the following diagram there are four type of users Roles:

- Students role has following permissions:-  
{View Courses, View Announcement, Edit Profile, View placement records}
- Faculty role has following permissions:-  
{Edit Course View Announcement, Edit Profile, View placement records}
- HOD role has following permissions:-  
{Add Courses, View Announcement, Edit Profile, View placement records}
- Admin role has following permissions:-  
{Delete Courses, Manage database, Add new info}



### 3. Conclusion

Managing and auditing network access is essential to information security. Access can and should be granted on a need-to-know basis. With hundreds or thousands of users, security is more easily maintained by limiting unnecessary access to sensitive information based on each user's established role within the organization. Other advantages include:

1. Reducing administrative work and IT support. With RBAC, we can reduce the need for paperwork and password changes when a student is admitted or changes their role. Instead, we can use RBAC to add and switch roles quickly and implement them globally across operating systems, platforms and applications. It also reduces the potential for error when assigning user permissions. This reduction in time spent on administrative tasks is just one of several economic benefits of RBAC. It also helps to more easily integrate third-party users into your network by giving them predefined roles.
2. Maximizing operational efficiency. RBAC offers a streamlined approach that is logical in definition. Instead of trying to administer lower-level access control, all the roles can be aligned with the organizational structure of the business and users can do their jobs more efficiently and autonomously.

The webapp makes use of components of RBAC for ensuring scalability of the application . It groups the permission assignment into roles which makes it easier to manage. In this way RBAC helps in maintaining the user role permission very efficiently .

### 4. References



1. Sandhu, Ravi, David Ferraiolo, and Richard Kuhn. "The NIST model for role-based access control: towards a unified standard." ACM workshop on Role-based access control. Vol. 10. No. 344287.344301. 2000.  
<https://profsandhu.com/dissert/diss-qamar.pdf>
2. <https://www.php.net/docs.php>
3. [https://www.apachefriends.org/faq\\_windows.html](https://www.apachefriends.org/faq_windows.html)