COLLEGE MANAGEMENT

(Case Study)

Project Description

A college management system is an online software designed to help colleges and higher education institutes to manage their day to day activities efficiently. students and parents can collaborate on a single platform and improve their efficiency and performance.

the advantages of using a college management system:

1. Better and improved admission rate
2. Better teacher performance
3. Drastic reduction in manual tasks
4. Elimination of paper based tasks
5. Improved fee management
6. Makes the institute smart and ready for the future.
7. Improved institute reputation
8. better resource utilization

Users: Admin, Student, Teacher, Department staff

Functional requirement:

Admin

- The admin register and login into the system
- The Admin Add/delete/update account
- The admin can see the Account Information
- The admin can see the user details list
- The admin can active and inactive the account of user.

Student

- The student can login to account
- The student can join to classes
- The student can change the password and pin of accounts
- The student can view exams
- The student can download materials
- The student can record all transaction

Teacher

- The Teacher can access system by username and password.
- Only teacher can mark student attendance
- The teacher can login and register
- The teacher can upload materials
- The teacher can do exams
- The teacher can check and update attendance
- The teacher can check and update grades

Department Staff

- The staff can enter students and (educational) university information
- The staff can create documents for educational process
- The staff can view student information and exam information
- The staff can manage students and courses
- The staff can delete accounts
- The staff can manage teacher/staff finance

Nonfunctional Requirement:

Usability

A user requirement might specify that the system should be easy to navigate and have an intuitive user interface. This is a non-functional requirement related to usability.

Performance

Users may have requirements related to system performance, such as response times for certain operations or the ability to handle a specific number of concurrent users. These are non-functional performance requirements.

Security

Users might specify security requirements, such as the need for secure authentication and data encryption, which are non-functional requirements focused on security.

Reliability

User requirements might include expectations about system uptime and reliability. These are non-functional requirements related to the system's dependability.

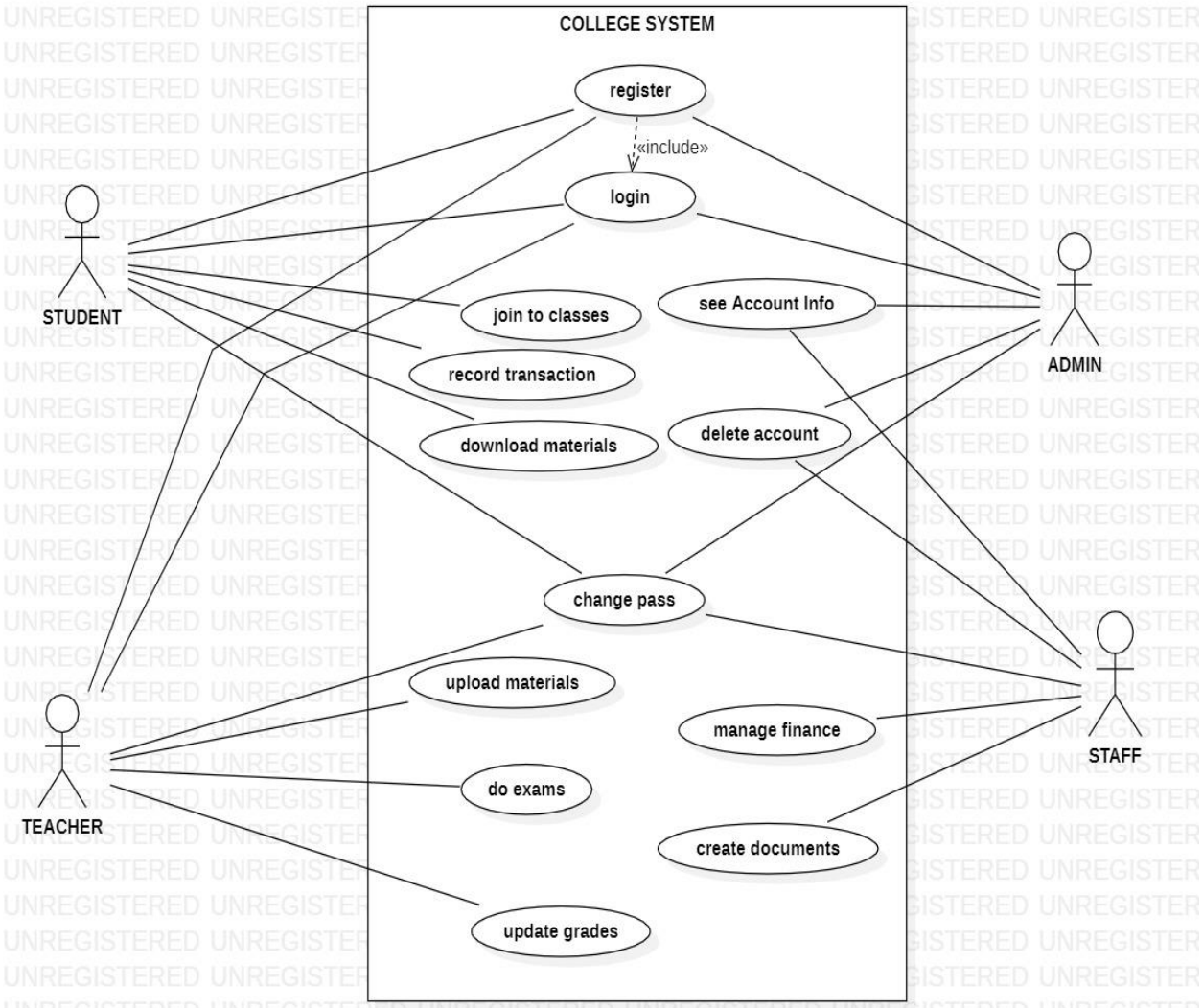
Scalability

If users anticipate that the system will need to handle a growing number of users or data, scalability requirements are non-functional requirements.

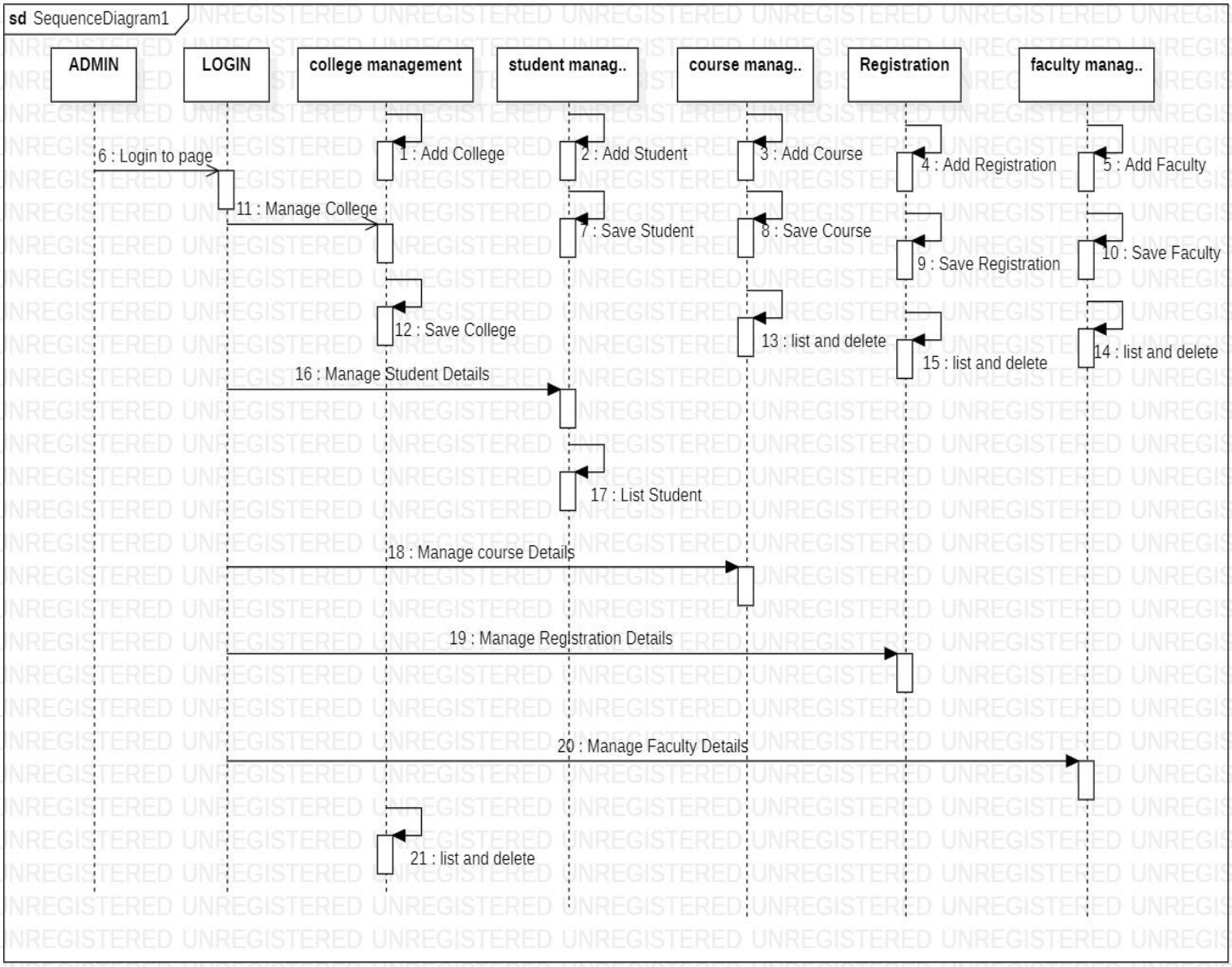
Quality Attributes

These are the characteristics of the system that determine its overall quality. Examples of quality attributes include security, performance, and usability.

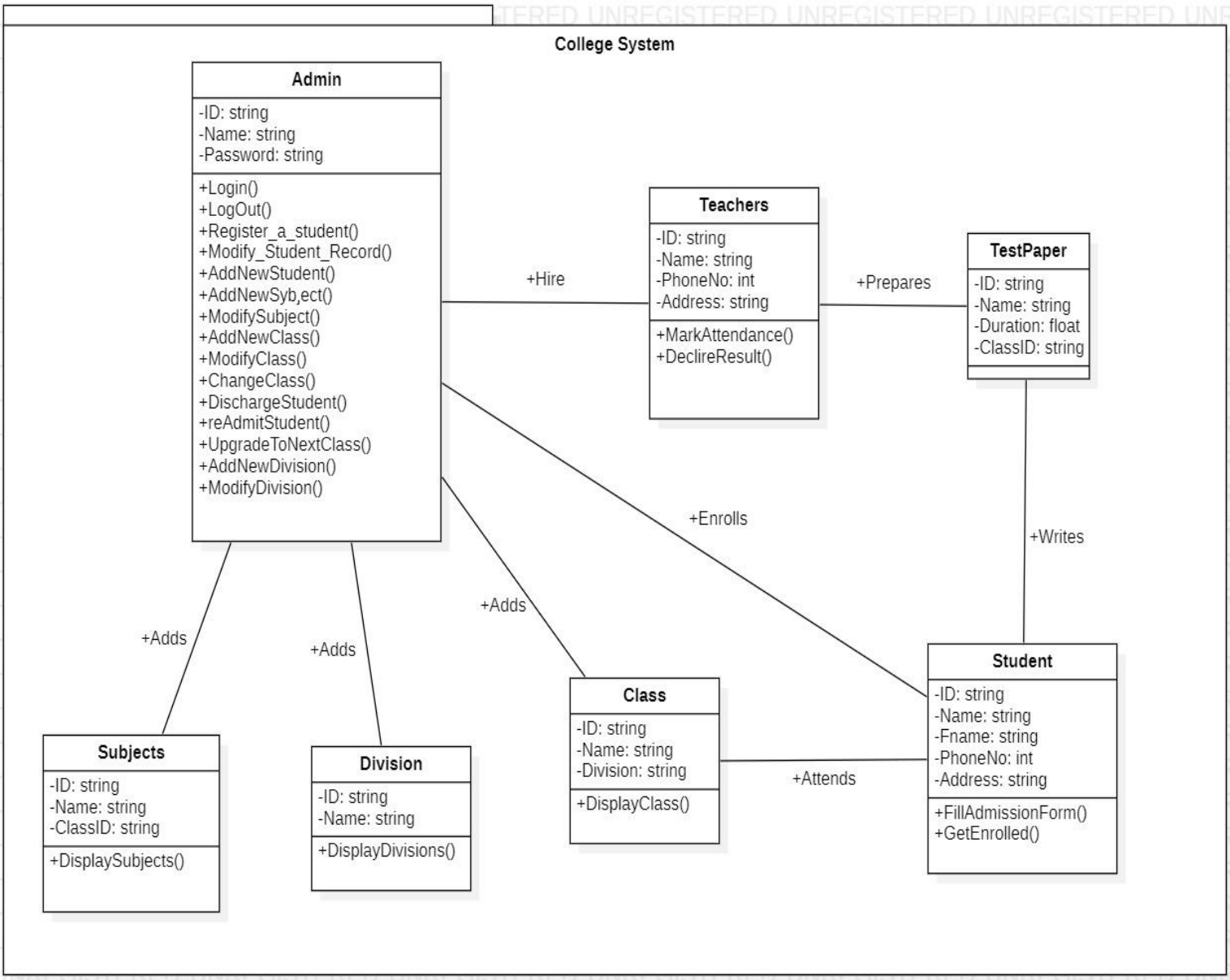
Use-Case Diagram:



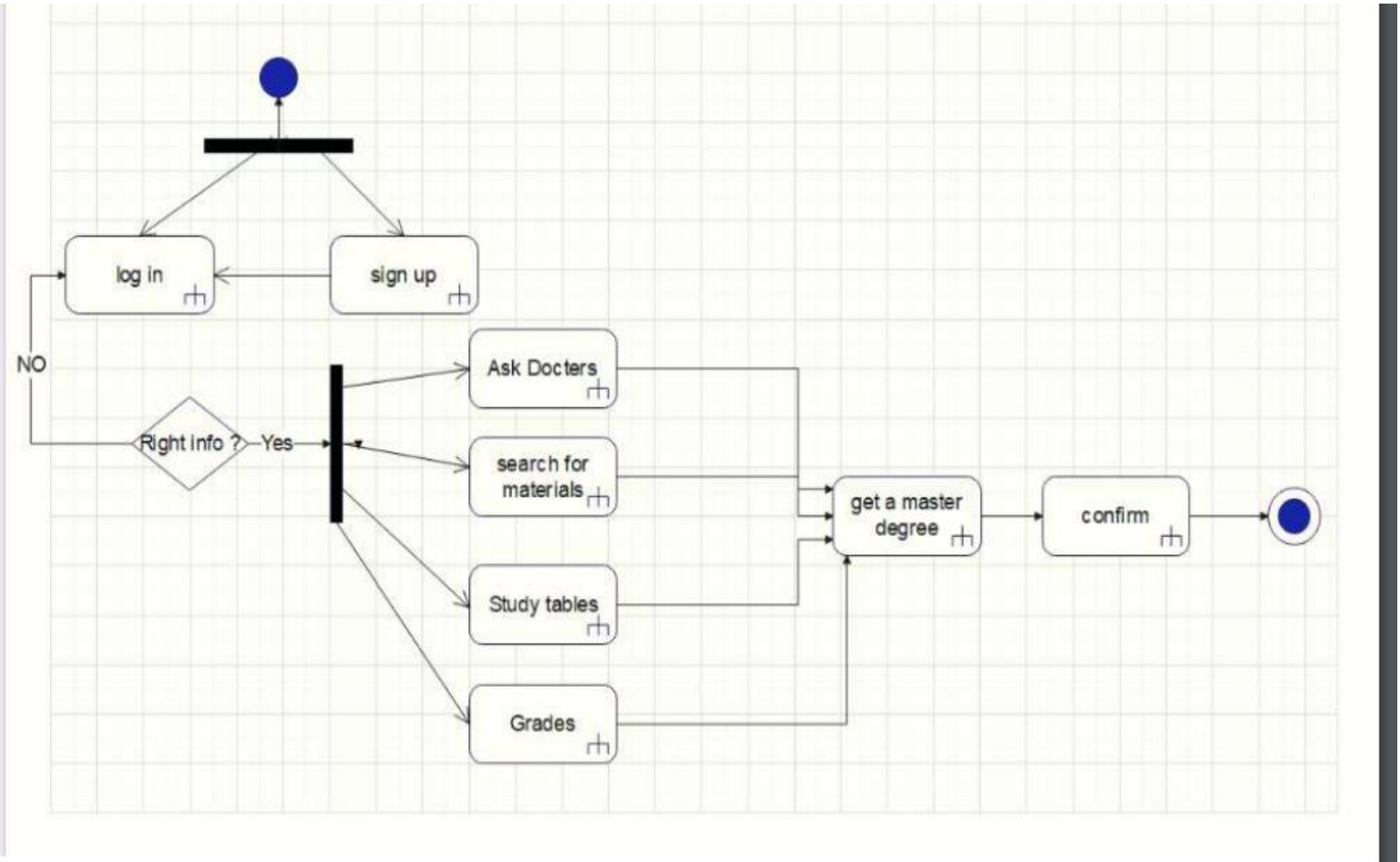
Sequence Diagram:



Class Diagram:



Activity Diagram:



Team Members:

- Khaled Kamal Badr
- Hussain Talha Hussain
- Maha Abd-Elmaksoud
- Nahla Muhammed Saeed