# CHAPTER 1 : INTRODUCTION

## Problem Statement

Certificate Printing is a paper based work which is itself a great difficulty to complete it on time. Paper based work consumes a lot of time of people. In the present day, when people want to save their time as much as possible. This certificate printing has consumed more time than expected. Not only this, people are printing the certificate manually by typing all those names, symbol number, grade, GPA, school/college name. This is a tiresome work to do for just an ordinary worker with little skill of technology.

## Motivation

In this today’s world, the importance of certificate is increasing. So, this system has to be made to fulfill that requirements.The quality of students highly depends upon the certificate of the users and students. A good institution should be able to make the certificate in a better way fulfilling all the criteria for making a certificate of any individual. The chief objective of any good institution is to provide facilities and services to the students fast and easy. So, the motivation of our project is to make such a system which can make the certificate printing system easier taking some feedback from all the facilitators of the institutions.

## Objectives

* To manage all the information of students and facilitators.
* To update information easily.
* To provide certificates to the students easily.
* To build a software for certificate printing system.
* To computerize and automate the work for certificate making system.
* To facilitate the facilitators to provide their feedback for the students under certain criteria.

## Project Scope

In this era of technology, still institutions are completing their works manually. They are making their certificates through MS Word and manually typing each student name, symbol number, GPA and manually entering their character or behavior.

This ***“Certificate Printing System”*** will automate all the facilities mentioned above both for the admins and facilitators.

## Limitations

* Certificate Printing is based on facilitators feedback only but not the other staffs.
* There are limited number of criteria to evaluate the students character.
* It can only print one student certificate at a time.
* Require skill and knowledge of computers.
* Based on Windows Operating System only.

## Report Organization

**Chapter 1:** This chapter explains about the overview, introduction,problem statement, motivations, project scopes and limitations of the system.

**Chapter 2:** This chapter covers all the history, methods, requirement specification and feasibility analysis and structured system requirements.

**Chapter 3:** Design of the result management system is explained in detail with all the necessary diagrams and brief functionality.

**Chapter 4:** Process of implementation and testing is described along with all the tools used for the development.

**Chapter 5:** Conclusion and future scope of the application are explained.

# CHAPTER 2 : REQUIREMENT ANALYSIS

## 2.1 Literature Review

The Certificate Printing System Documentation is an all-encompassing guide that elucidates the functionality, structure, and utilization of the software application, specifically devised for the efficient management and production of certificates. Addressing both administrators and users, this documentation furnishes a comprehensive comprehension of the system's features and the navigation of its functionalities. It encompasses the evolution of certificate-related software solutions that preceded the current system, drawing insights from their successes and limitations.

From these experiences, the Certificate Printing System has been meticulously designed to address previous shortcomings and provide an advanced, user-friendly interface. It also covers the software's architecture, installation procedures, user roles and permissions, certificate data handling, generation of certificates, printing infrastructure, system maintenance, troubleshooting, and support mechanisms. This documentation ensures that users are adept in harnessing the software's potential, thereby facilitating seamless certificate generation, precise data management, and streamlined administrative operations.

The Certificate Printing System Documentation serves as a comprehensive guide, shedding light on the software's functionality, structure, and practical application in efficiently managing and producing certificates.

Tailored for both administrators and users, this documentation provides a thorough understanding of the system's features and effective navigation through its functions. Building upon lessons learned from previous certificate-related software iterations, this documentation highlights the evolution of solutions in the certificate management domain, integrating insights from past successes and limitations to refine and optimize the current Certificate Printing System.

This documentation further covers the software's architecture, installation processes, user role assignments, certificate data management, the certificate generation process, printing infrastructure setup, ongoing system maintenance, troubleshooting techniques, and avenues for user support. Through this comprehensive documentation, users gain proficiency in leveraging the software's capabilities, enabling seamless certificate generation, accurate data management, and efficient administrative workflows.

## 2.2 Problem Definition

The problem before having computerized system includes:

* **File lost:** When computerized system wasn't made, the file used to get deleted or lost due to some certain problems of the system or users.
* **File damaged:** The file where they used to record the details of all the users manually as a certificated used to be damaged due to the update of the system or software.
* **Difficult to make certificates:** When there were no automated system, the certificate making system always used to be a challenge.
* **Space consuming:** The manual file system used to consume more space as every students require a separate file to store their details.

## 2.3 Requirement Analysis

In our project, we have collected list of documents with sufficient and necessary requirements for the project development.

To derive the requirements, we have done better understanding of the products under development which we achieved through detailed and continuous communications with the project team throughout the software development process.

**2.3.1 Functional Requirement(Use Case)**

**1. Add Facilitator Info:** Admins can add the facilitator information in the system.

**2. Add Student Info:** Admins can add the student information in the system.

**3. View Facilitator Info:** Admins and facilitators can view the facilitator information.

**4. View Student Info:** Admins and facilitators can view the student information.

**5. Add Student Remarks:** Admins and facilitators can add the students character detail based on certain criteria.

**6. View Student Remarks:** Admins and facilitators can view students remarks.

**6. Auto Management:** The details of facilitators and students is managed automatically.

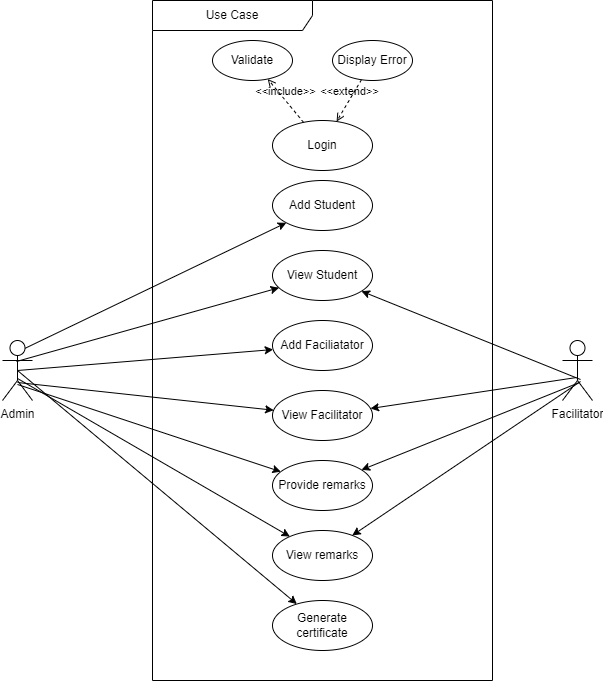


Fig 1 : Use Case

**2.3.1.1 Use Case Template**

| Use Case Id: | 1.0 |
| --- | --- |
| Use Case Name: | Log In |
| Created By: | Manoj Shrestha Last updated by: |
| Date Created: | August 24, 2023 Date last updated: August 24,2023 |
| Actor(s): | Admin, Facilitator |
| Description: | This use case allows users to login into the system and perform different functions according to the user roles. All the users have to enter their unique username and password. |
| Precondition: | Users have to validate their account. |
| Post-condition: | System displays the relevant homepage. |
| Normal Courses: | Users enter the username,password and choose user role.  System validates the username, password and user role.  System verifies the username, password and user role.  System displays the relevant homepage.  The use case ends. |
| Alternative Courses: | 1a. Upon missing username, password or user role.  System prompts for empty username, password or user role.  The use case resumes at step 1. |
| Exceptions: | Nil |
| Includes: |  |
| Priority: | High |
| Frequency of use: | 24 hours a day |
| Business Rule: | User must login through a registered and verified account. |
| Special Requirement: |  |
| Assumptions: | Nil |

| Use Case Id: | 2.0 |
| --- | --- |
| Use Case Name: | Add Student |
| Created By: | Manoj Shrestha Last updated by: |
| Date Created: | August 24, 2023 Date last updated: August 24,2023 |
| Actor(s): | Admin |
| Description: | This use case allows admins to fill up the student details into the system. |
| Precondition: | Admins have to gain access to their account. |
| Post-condition: | System displays a form to fill up the students details. |
| Normal Courses: | First admin must have to login into their account.  Then click on the Student in the navigation bar.  Then click on Add.  The student detail fill up form will appear.  The use case ends. |
| Alternative Courses: | 1a. Upon missing the details like name, address.  The system will displays a prompt with a message. The name and address cannot be empty. |
| Exceptions: | Nil |
| Includes: |  |
| Priority: | High |
| Frequency of use: | 24 hours a day |
| Business Rule: | User must login through a registered and verified account. |
| Special Requirement: |  |
| Assumptions: | Nil |

| Use Case Id: | 2.1 |
| --- | --- |
| Use Case Name: | View Student |
| Created By: | Manoj Shrestha Last updated by: |
| Date Created: | August 24, 2023 Date last updated: August 24,2023 |
| Actor(s): | Admin |
| Description: | This use case allows users to view the student details from the system. |
| Precondition: | Users have to gain access to their account. |
| Post-condition: | System displays a form to fill up the students details. |
| Normal Courses: | First admin must have to login into their account.  Then click on the Student in the navigation bar.  The student detail will appear.  The use case ends. |
| Alternative Courses: | 1a. Upon not getting student info:  System displays a empty table with an error. |
| Exceptions: | Nil |
| Includes: |  |
| Priority: | High |
| Frequency of use: | 24 hours a day |
| Business Rule: | User must login through a registered and verified account. |
| Special Requirement: |  |
| Assumptions: | Nil |

| Use Case Id: | 3.0 |
| --- | --- |
| Use Case Name: | Add Facilitator |
| Created By: | Manoj Shrestha Last updated by: |
| Date Created: | August 24, 2023 Date last updated: August 24,2023 |
| Actor(s): | Admin |
| Description: | This use case allows admins to fill up the facilitator details into the system. |
| Precondition: | Admins have to gain access to their account. |
| Post-condition: | System displays a form to fill up the facilitator details. |
| Normal Courses: | First admin must have to login into their account.  Then click on the Facilitator in the navigation bar.  Then click on Add.  The facilitator detail fill up form will appear.  The use case ends. |
| Alternative Courses: | 1a. Upon missing the details like name, address.  The system will displays a prompt with a message. The name and address cannot be empty. |
| Exceptions: | Nil |
| Includes: |  |
| Priority: | High |
| Frequency of use: | 24 hours a day |
| Business Rule: | User must login through a registered and verified account. |
| Special Requirement: |  |
| Assumptions: | Nil |

| Use Case Id: | 3.1 |
| --- | --- |
| Use Case Name: | View Facilitator |
| Created By: | Manoj Shrestha Last updated by: |
| Date Created: | August 24, 2023 Date last updated: August 24,2023 |
| Actor(s): | Admin |
| Description: | This use case allows users to view the facilitator details from the system. |
| Precondition: | Users have to gain access to their account. |
| Post-condition: | System displays the facilitators details. |
| Normal Courses: | First admin must have to login into their account.  Then click on the Facilitator in the navigation bar.  The facilitator detail will appear.  The use case ends. |
| Alternative Courses: | 1a. Upon not getting facilitator info:  System displays a empty table with an error. |
| Exceptions: | Nil |
| Includes: |  |
| Priority: | High |
| Frequency of use: | 24 hours a day |
| Business Rule: | User must login through a registered and verified account. |
| Special Requirement: |  |
| Assumptions: | Nil |

**2.3.2 Non-functional Requirement**

1. **Security**: - Only authorize users can access the system with username and password
2. **Performance**: - Easy tracking of records and update can be done.
3. **User Friendly**: - The system is very interactive.
4. **Availability**: The system is available all the time, no time constraint

## 2.4 Feasibility Study

## 2.5 Structuring System Requirements

**2.5.1 ER Diagram**

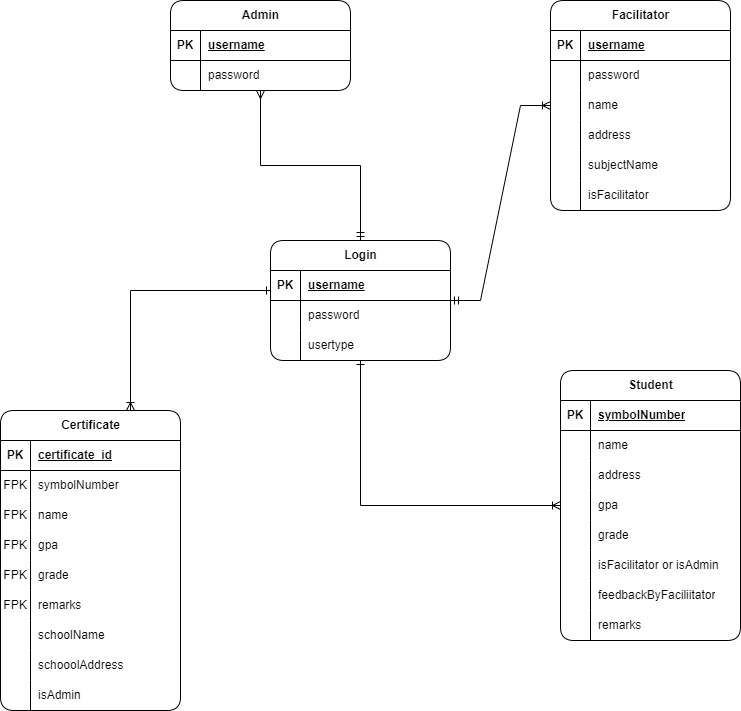
****

Fig 2: ER Diagram

The above diagram explains the relationship between the database where rectangle represents entity, oval represents attributes and Diamond represents relation. There are five entities with their respective attributes. Admin acts as one of the entities and has attributes like admin\_ id, admin\_ username, admin\_ password. And another entity Facilitator acts as another entity and it has attributes like facilitator\_ id, facilitator\_ username, facilitator\_password, facilitator\_name,facilitator\_address, facilitator\_subjectName. The student is another entity and it has attributes like student\_ id, students\_ name,student\_address, student\_gpa, student\_grade, and student\_remarks. The login includes login\_id,login\_pass and login\_usertype. The certificate is another entity and it has attributes like certificate\_id, certificate\_schoolName, certificate\_schoolAddress and certificate\_isAdmin.

## **2.5.2 Process Modeling(DFD Level-0)**

## DFDLevel0

## Fig 3: Level-0 DFD

The above diagram shows the DFD zero level where system prints certificate for the student. It’s a basic overview of the whole Certificate Printing System where admins and facilitator login to the system. System takes the students information and sends data to admin and facilitator profile. Admin performs updating data and adding new students and facilitators records in addition admin manages student’s data, facilitators data and certificate information and update report.

**2.5.3 Process Modeling(DFD Level-1)**

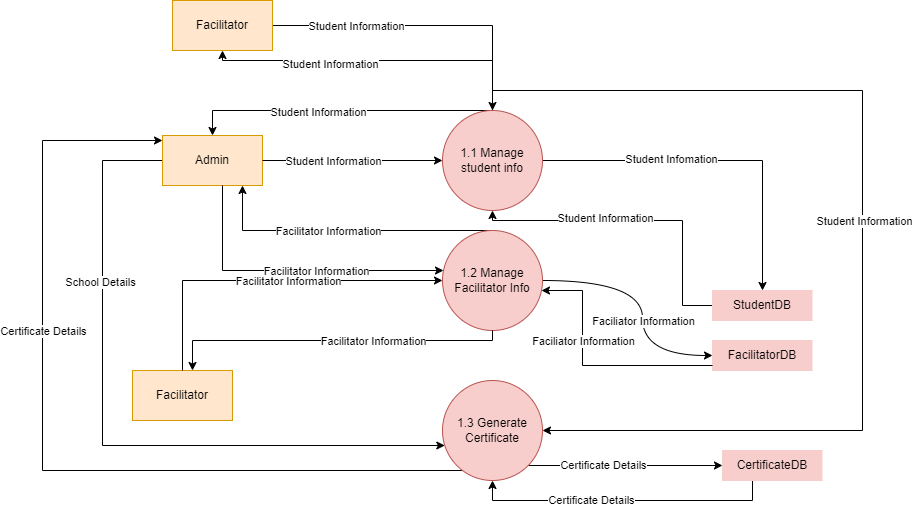


Fig 4: Level-1 DFD

The above diagram is the DFD level-1, where it explains the working process of system. These procedures require information such as record of students, and facilitators from which served as the bases for the Certificate Printing System. There is login process in the system which is authenticated and authorized by system. Admins and facilitators login to the system via username and password then system verify it. If username and password is validated then the system opens the home page, if the username and password is invalidated then the system notifies unauthorized.

# Chapter 3 : System Design

## 3.1 System Architecture and Overview

System Design is the process of designing the architecture, components, and interfaces for a system so that it meets the end-user requirements. The designs can be defined in graphical or textual modelling languages. Also, it is the process of creating or altering systems, along with the processes, practices, models. And methodologies used to develop them.

We have developed *“Certificate Printing System”* which doesn’t require Web-server for its functioning. To use this system we need a C#, .NET, laptop / desktop and the users.

## 3.2 System Design

**3.2.1 Database Schema**

**3.2.2 Data Dictionary**

**3.2.3 UML Class Diagram**

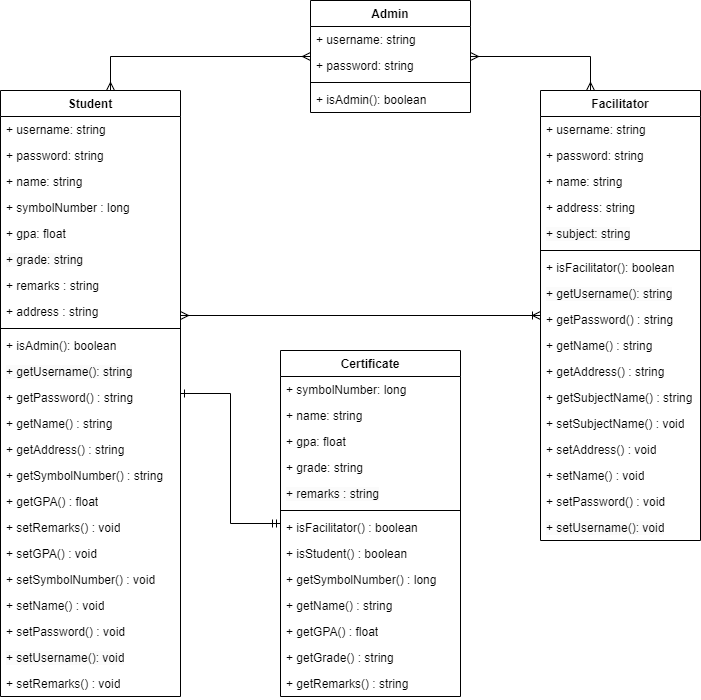
****

Fig 5: Class Diagram

**3.2.3 UML Activity Diagram**

**3.3 Interface Design**

# Chapter 4 : Implementation and testing

## Implementation and Overview

A Prototyping Model has been used to develop this application. The prototyping model is a technique for quickly building a function but incomplete model of the information system. There are several kinds of prototyping but they all intend to reduce risk by building a quick and dirty replica or mock up of the intended system. It can be used to demonstrate technical feasibility when the technical risk is high. It can also be used to better understand and elicit user requirements. In either case, the goal is to reduce risk and limit costs by increasing understanding of proposed solutions before committing more resources.

## Technology used in my system

**4.2.1 C#**

C#( C-Sharp) is a versatile and modern programming language developed by Microsoft. Known for its strong typing, object-oriented features, and integration with .NET Framework, C# is widely used for creating a wide range of applications, from desktop software and games to web services and mobile apps.

**4.2.2 .NET framework**

The .NET framework, developed by Microsoft, is a comprehensive software platform that provides developers with tools and libraries for building and running various types of applications, including web, desktop and mobile applications. It offeres a runtime environment, a vast class library, and support for multiple programming languages like C# and VB.NET. The .NET framework simplifies application development by handling common tasks and enabling seamless communication between different software components.

**4.2.3 mysql.data**

MySQL.Data is a crucial component of the MySQL Connector/NET package, serving as a data provider that facilitates the connection between MySQL databases and .NET applications. It offers a set of classes, methods, and interfaces that allow developers to interact with MySQL databases using .NET languages like C#. MySQL.Data enables tasks such as establishing connections, executing queries, and managing data, ensuring efficient and reliable communication between the application and the database.

**4.2.4 MySQL**

MySQL is an open-source relational database management system that provides a reliable and efficient platform for storing, managing, and retrieving structured data. Developed by Oracle, MySQL is widely used for various applications, from small-scale web projects to large enterprise-level solutions, due to its scalability, speed, and strong support for standard SQL queries. Its versatility and active community support have made MySQL a popular choice among developers seeking robust database solutions.

## 4.3 Testing

# Chapter 5 : CONCLUSION and recommendations

## 5.1 Conclusion

## 5.2 Recommendation