

**TRIBHUVAN UNIVERSITY
BHAKTAPUR MULTIPLE CAMPUS**

CSC 367: NET Centric Computing



Project Report on Staff Management System

Submitted By:

Ankit Nepal

Sec: A

Roll no: 09/23225

Year/Sem: 3rd / 6th

BSc.CSIT

Department of Computer Science and Information Technology

Submitted To:

Ramesh Kharbuja

Department of Computer Science and Information Technology

Acknowledgement

First and foremost, I would like to express my unequivocal gratitude and appreciation to the venerable individual: Sir Sushant Poudel, CSIT Coordinator, Bhaktapur Multiple Campus, for the continual guidance throughout the academic tenure, and the reasonable and interactive environment within the institute which sculpted us into a bunch of bona fide IT students.

We pay out great sense of gratitude to Mr. Ramesh Kharbuja, Net Centric Computing subject teacher of Bhaktapur Multiple Campus, for all those eternal efforts to encourage us to the highest peak and to provide us with this opportunity to prepare the project. I am immensely obliged to my friends for their elevating inspiration, kind supervision and encouraging guidance for the completion of this project.

On the very outset of this report, I would like to extend my heartfelt and sincere obligation toward all the personages who have helped me in this endeavor. Without this active guidance, help, cooperation and encouragement, I wouldn't have made headway in this project.

Abstract

The essence of technological evolution in Information and Technology suggests the inevitable fact that humans urge for sophistication and ease of access in every aspect of life. Thereby, this project adheres to the ideology and aims to provide accessibility and facility in the management of employee within the business organization. This project attempts to utilize the software engineering principles in the essence. And distinctly, this project is based on the implementation methodology of the full stack web application development using ASP.Net Core with code first approach, where both frontend and backend approaches were addressed, formulated and executed rationally and thoroughly. Further, the project development also emphasizes the necessity of system analysis of functional and non-functional requirements along with the proper sense of system design. With the OOP approach implemented, distinct structural diagrams: class diagram and data modeling diagram were formulated and studied. The testing endeavors involved the system testing and unit testing cases. Irrefutably, the final product so deduced is found to be within the expectations and justifies the objective of easing the process of managing the records of the employees.

Table of Contents

Acknowledgement	i
Abstract.....	ii
List of Figures	ii
Chapter 1	1
Introduction.....	1
Problem Statement.....	1
Objectives.....	1
Scope	2
Chapter-2.....	3
Requirement Analysis and Feasibility Analysis	3
Requirement Analysis.....	3
Feasibility Study.....	3
Chapter-3.....	5
System Design.....	5
Design	5
Structural Diagram.....	5
Behavioral Diagram.....	5
Chapter-4.....	6
Implementation	6
Design Tools.....	6
Programming Language.....	6
Database Platforms	6
Testing.....	6
Test Cases for Unit Testing	6
Test Cases for System Testing.....	6
Result.....	7
Result Analysis	7
User Interface	7
Chapter-5.....	11
Conclusion	11
Future Recommendations	11
Reference	11

List of Figures

Figure 1: Class Diagram of Staff Management System.....	5
Figure 2: Use Case Diagram for Staff Management System	5
Figure 3:Login page.....	7
Figure 4: Landing Page.....	7
Figure 5: Adding Staff Details.....	8
Figure 6: Display Grid View.....	8
Figure 7: Delete Staff Data	9
Figure 8: Database Code.....	9
Figure 9: Database	9
Figure 10:Print Preview	10

Chapter 1

Introduction

A staff management system is a tool for managing employee data and related tasks. It can help organizations keep track of employee information, such as contact details and job descriptions. Staff management systems can also include features for attendance tracking, scheduling, payroll processing, performance tracking, communication, access control, reporting, integration with other systems, mobile accessibility, training and support resources, customization options, data security measures, scalability, and user-friendly interface.

These systems can streamline administrative tasks, improve communication, enhance productivity, and reduce errors and inconsistencies. Staff management systems can be customized to fit the unique needs of an organization, whether it's a small business or a large enterprise. They can also be integrated with other software, such as accounting or HR systems. Overall, a staff management system can help organizations manage their workforce more efficiently and effectively.

Staff Management System is the system where one (manager or staff) requires to add the necessary details where one can view their personal as well as official records. Staff Management System aims to provide user simple medium so they can manage the staffs.

Problem Statement

These legacy systems typically have poor usability, with clunky interfaces and numerous steps required for routine tasks like updating employee information or tracking time and attendance. They lack flexibility and self-service capabilities, forcing employees to go through HR bottlenecks for basic needs. Traditionally designed for static in-office work, these systems struggle to support remote or flexible work arrangements that have become increasingly common. They also may not be optimized for regulatory compliance needs around data privacy, time tracking, etc. Overall, the manual nature, fragmented data, inflexible design, and poor user experience of legacy employee management systems make it arduous for both HR teams and employees to use them effectively. Modern solutions aim to solve these issues through automation, data integration, and improved analytics.

Objectives

The primary goal of implementing an staff management system is to create a centralized, integrated platform to streamline and automate routine HR processes. Key objectives include establishing a single source of truth for employee data across the organization, enhancing analytics and reporting capabilities, improving compliance management, and providing intuitive self-service access for employees and managers.

Scope

The scope of a staff management system is usually comprehensive, spanning core HR activities across the employee lifecycle. Key functions include applicant tracking, onboarding, personnel administration, benefits enrollment, payroll record. The system consolidates employee data into a single repository, standardizes HR processes through workflow automation, and provides self-service access via an employee portal.

Chapter-2

Requirement Analysis and Feasibility Analysis

Requirement Analysis

The general approach of the staff and the manager was observed and their expectations and requirements for better staff management experience was consistent throughout the observation. Any staff or employer would prefer an online medium for the records management which may as well make the work easier for HR.

1. Functional Requirements

The main requirement suggested for the system is the provision of a concrete and reliable online medium for staff management. The use of the so system can minimize the manual work and it provides the accurate record and can keep an eye on the employee.

2. Non-Functional Requirements

“While functional requirements define the system’s fundamental behavior, non-functional requirements set out how the system will carry out this function.

Several critical non-functional requirements must be considered when developing or selecting an employee management system. Foremost is security, as the system stores highly sensitive personal data, so encrypting data in transit and at rest, role-based access controls, and stringent authorization protocols are essential. The system should have high availability with minimal downtime via failover infrastructure and data backups. The system should offer API integration with existing enterprise platforms to enable data sharing. Scalability to support organizational growth is vital as well. Compliance capabilities to meet labor laws and regulations are also key.

Feasibility Study

Feasibility refers to the process that describes, identifies and evaluates the proposed system and selects the best system for proper functioning. For evaluating the feasibility, a feasibility study is conducted that helps in determining whether the system is possible to develop or not. There are three types of feasibility study, they are – technical feasibility, economic feasibility and behavioral feasibility.

1. Behavioral Feasibility

The exponential rise in the misuse of employment management system in the context of some organization supports the fact that the management system must be addressed more diligently. And hereby, the employee and workers want an efficient medium of records viewing, and our system can fulfill their demand. And hence, the feasibility scale is justified behaviorally and operationally.

2. Economic Feasibility

As it has high chances to be used by the organizational employees, so it can create a promising gain path for the organization. Thereby, the economic feasibility can be justified with proper provision of advertisement sections in the web application.

3. Technical Feasibility

The tools and software that are required for this project are easily available in the web. It doesn't require special environment to execute. Thus, it is affordable and it can be said to be technically feasible.

Chapter-3

System Design

Design

The approach ensued by the project involves the effective use of the prominent object-oriented programming and the data modelling is formulated accordingly. The structural and behavioral approach rendered by the OOP has been pictorially illustrated below.

Structural Diagram

This sort of diagrams in system design emphasizes the static structure of the system using objects, attributes, operations and relationships. It includes class diagrams and composite structure diagrams.

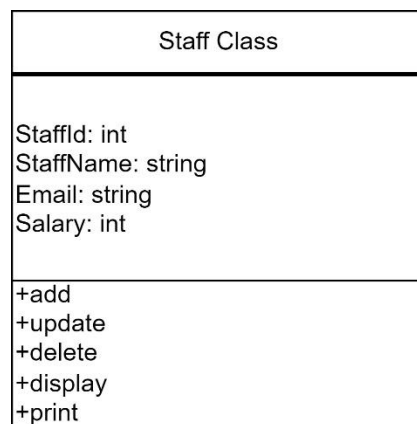


Figure 1: Class Diagram of Staff Management System

Behavioral Diagram

Behavioral diagram shows how the system works 'in motion', that is how the system interacts with external entities and users, how it responds to input or event and what constraints it operates under.

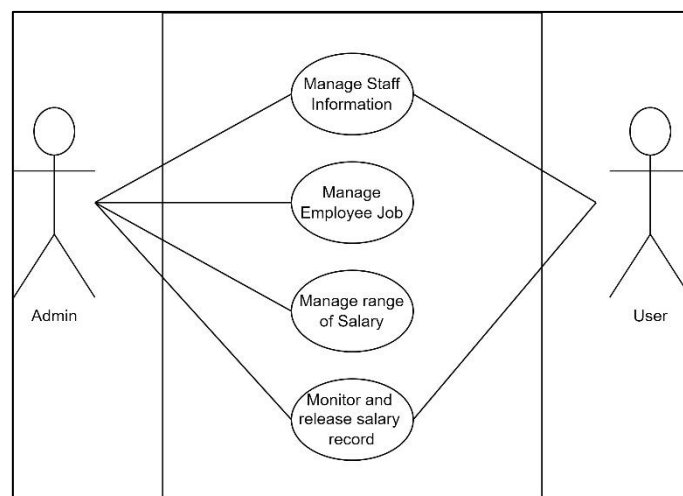


Figure 2: Use Case Diagram for Staff Management System

Chapter-4

Implementation

The implementation methodology involves the full stack web development package namely ASP.Net Core 3.1 that delivers all the basic tools for implementing the system design with ease and efficiency. The implementation is based on one of the popular programming languages i.e., C# that provides any developer with effective set of features and functionality to sculpt the fully functional web application in accordance and resonance with the prior system design drafts.

However, to be precise, the implementation involved two major development phase first, frontend development and second, the backend development which conclusively summed up to a full stack web application.

Design Tools

Different types of tools were used for the implementation of different diagrams and charts used in this report such as use case diagram, class diagrams, etc. Draw.io is the most used design tool for these diagrams. Some other tools are Lucid Chart and Smart Draw.

Programming Language

The main programming language can be distinctly stated as C#, however, there are instances of use of various language throughout the implementation, which is listed out below:

- Programming Language: C#, MySQL, MySQL Server
- IDE: Visual Studio Community 2022

Database Platforms

For the database platform, the need of local database was fulfilled by the availability and implementation of the service provided by MySQL Server. It provides fluent connectivity to the APIs and responsive behavior to data alters.

Testing

Test Cases for Unit Testing

Unit testing involved the essential testing performed on the individual components and modules like web components, backend components: especially, to emphasize their independence and overall functionality. Unit testing provides a sort of living documentation of the system. Developers looking to learn what functionality is provided by a unit, and how to use it, can look at the unit tests to gain a basic understanding of the unit's interface.

Test Cases for System Testing

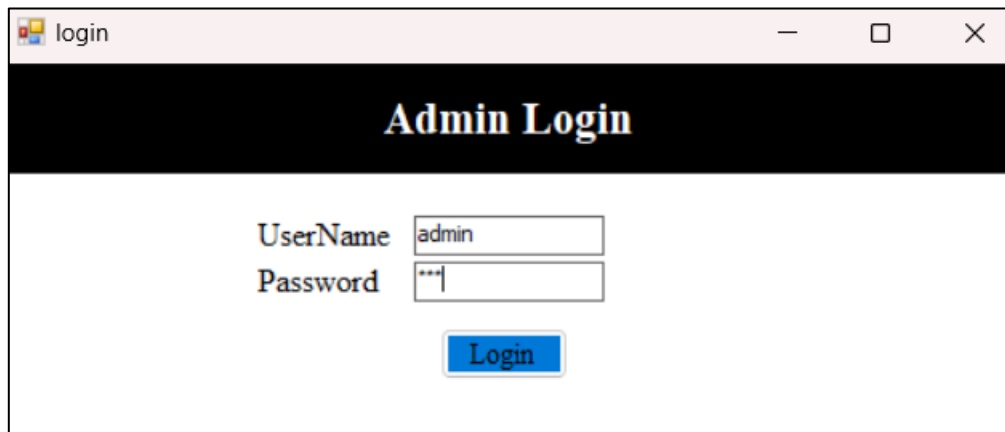
System testing was monitored and supervised to ascertain the integrity of those units i.e. individual components and modules amongst each other. The co-ordinates of the components such as authorization along with the database management was also tested.

Result

Result Analysis

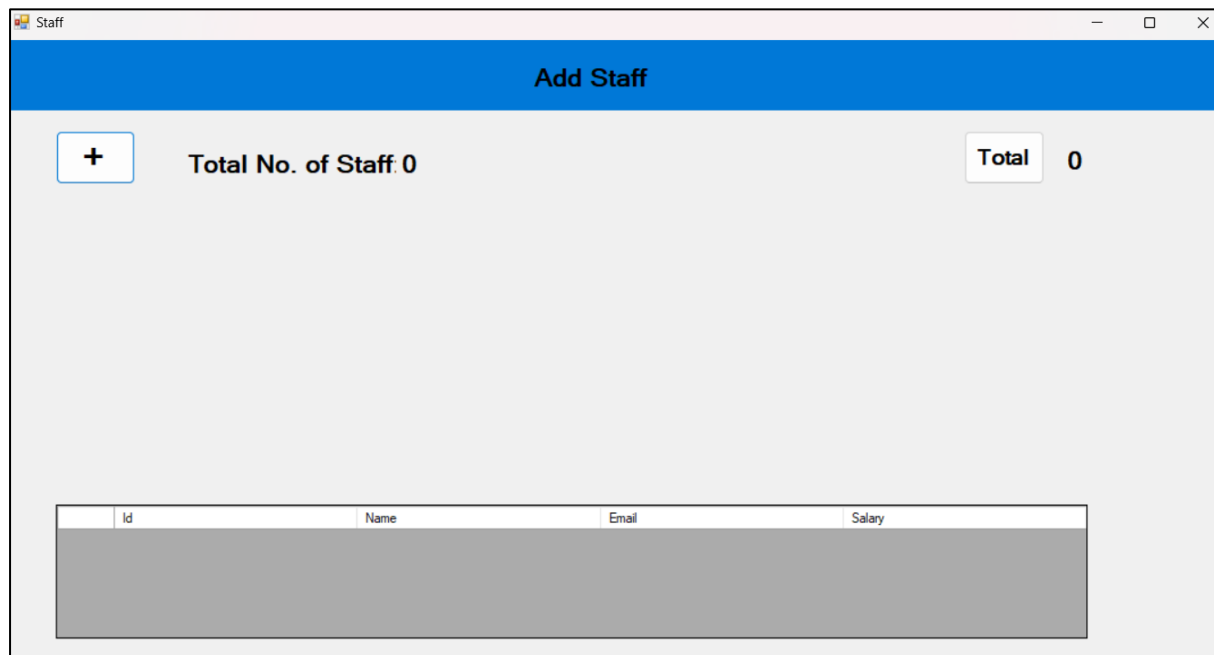
The overall consequential product was well within the expectation and requirements drafted earlier in the system development phase. The user interface worked articulately with responsive behavior and there were very few visible non uniformity in the overall interface design of the system. The functional aspects also were sound and full operational.

User Interface



The screenshot shows a web browser window titled "login". The page has a black header with the text "Admin Login" in a stylized, outlined font. Below the header, there are two input fields: "UserName" with the value "admin" and "Password" with masked characters "***". A blue "Login" button is positioned below the password field.

Figure 3:Login page



The screenshot shows a web browser window titled "Staff". The page has a blue header with the text "Add Staff". Below the header, there is a section with a blue button containing a "+" sign, followed by the text "Total No. of Staff: 0". To the right, there is a "Total" label and the value "0". Below this section, there is a table with four columns: "Id", "Name", "Email", and "Salary". The table body is currently empty, showing only the header row.

Figure 4: Landing Page

Staff

+

Total No. of Staff: 0

Total 0

Staff Info.

Id: 1

Name: abc

Email: abc@gmail.com

Salary: 10000

Display

Print

Save

Insert

Update

Remove

Id	Name	Email	Salary

Figure 5: Adding Staff Details

Staff

+

Total No. of Staff: 0

Total 0

Staff Info.

Id: 4

Name: name

Email: name@gmail.com

Salary: 15000

Display

Print

Save

Insert

Update

Remove

Id	Name	Email	Salary
1	abc	abc@gmail.com	10000
2	xyz	xyz@gmail.com	20000
3	pqr	pqr@gmail.com	5000
4	name	name@gmail.com	15000

Figure 6: Display Grid View

Staff Info.

Id:

Name:

Email:

Salary:

Buttons: Dis, Sa, Update, Remove

Modal: Record Deleted Successfully

	Id	Name	Email	Salary
▶	1	abc	abc@gmail.com	10000
	2	xyz	xyz@gmail.com	20000
	3	pqr	pqr@gmail.com	5000

Figure 7: Delete Staff Data

Update

```

1 CREATE TABLE [dbo].[Stafftable] (
2     [Id] INT NOT NULL,
3     [Name] VARCHAR (50) NULL,
4     [Email] VARCHAR (50) NULL,
5     [Salary] INT NULL,
6     PRIMARY KEY CLUSTERED ([Id] ASC)
7 );
8
9

```

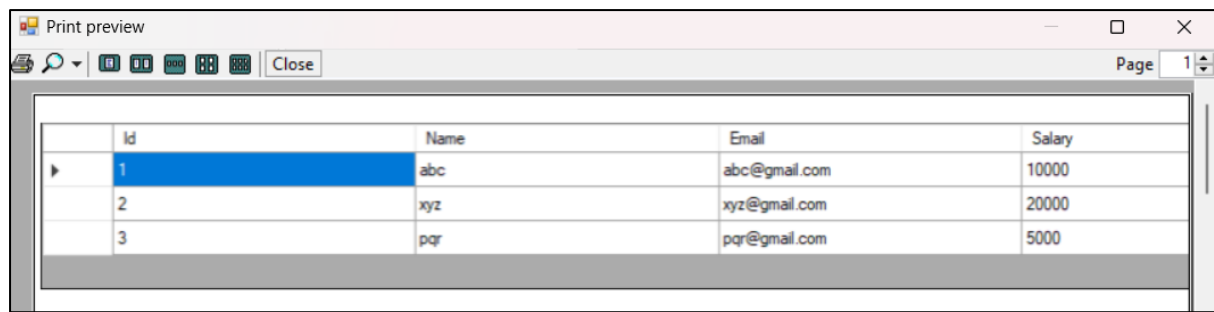
Figure 8: Database Code

dbo.Stafftable [Data]

Max Rows: 1000

	Id	Name	Email	Salary
▶	1	abc	abc@gmail.com	10000
	2	xyz	xyz@gmail.com	20000
	3	pqr	pqr@gmail.com	5000
⊞	NULL	NULL	NULL	NULL

Figure 9: Database



Print preview

Close

Page 1

	Id	Name	Email	Salary
▶	1	abc	abc@gmail.com	10000
	2	xyz	xyz@gmail.com	20000
	3	pqr	pqr@gmail.com	5000

Figure 10:Print Preview

Chapter-5

Conclusion

The implementation of a comprehensive staff management system has delivered immense value to our organization by centralizing and optimizing core HR processes. The streamlined workflows and self-service portal have reduced administrative overhead for HR while providing employees and managers with modern tools to access information and complete tasks. Powerful reporting and analytics are yielding data-driven insights to inform strategic workforce planning. Integrations with existing systems created a unified HR data repository, eliminating silos.

Compliance management and audit capabilities help ensure adherence to labor laws and regulations. While change management efforts were essential to drive adoption, employee feedback on system usability and utility has been overwhelmingly positive. In summary, this project achieved the core goals of automation, standardization, data consolidation, and improved user experience across the employee lifecycle. It has transformed employee services, elevated HR's strategic role, and enabled more efficient people management. The success provides a template for future enhancements and expansion of the platform.

Future Recommendations

While the current system has delivered substantial improvements, additional features could further optimize HR processes and analytics. Integrating predictive analytics to flag talent risks would enable more proactive retention strategies. Machine learning algorithms could analyze performance data to identify high-potential employees for advancement. Expanded mobile access would provide even greater flexibility. Offering more online training content and personalized learning pathways would boost development. Introducing compensation management modules would centralize pay planning and analysis. Enhancing collaboration tools could connect remote employees. Strengthening culture assessment surveys would provide insights to guide engagement initiatives. Providing APIs for external recruiting and assessment platforms could enhance hiring. These recommendations would build on the existing foundation to help attract, develop and retain top talent. With continuous innovation, the HR system can remain a strategic advantage in managing the evolving workforce.

Reference

[1]<https://www.javatpoint.com/er-diagram-for-employee-management-system>

[2]<https://learn.microsoft.com/en-nz/dotnet/>