

Feedback System

A PROJECT REPORT

Submitted to

Submitted by

**Ridakshi Sharma(23BCS10915)
Sahil Attri(23BCS10133)
Mohit Bhardwaj(23BCS10531)**

in partial fulfilment for the award of the degree

of

**Bachelor of Engineering
in**

Computer Science



Chandigarh University

NOV 2025



BONAFIDE CERTIFICATE

Certified that this project report “**Feedback System**” is the Bonafide work of **Ridakshi, Sahil, Mohit** who carried out the project work under my supervision.

SIGNATURE

SIGNATURE

HEAD OF THE DEPARTMENT

SUPERVISOR

TABLE CONTENT

Abstract	5
Chapter 1. INTRODUCTION.....	6-8
 1.1 . CLIENT IDENTIFICATION.....	6
 1.2 . IDENTIFIACTION OF PROBLEM	6
 1.3 . IDENTIFICATION OF TASKS	7
 1.4. TIMELINE.....	7
 1.5. ORGANIZATION OF THE REPORT.....	8
Chapter 2. LITERATURE REVIEW	9-11
 2.1 . TIMELINE OF THE REPORTED PROBLEM	9
 2.2. PROPOSED SOLUTIONS.....	9
 2.3. BIBLIOMETRIC ANALYSIS.....	10
 2.4. REVIEW SUMMARY.....	10
 2.5. PROBLEM DEFINITION.....	10
 2.6. GOALS/OBJECTIVES.....	10-11
Chapter 3. DESIGN FLOW.....	12-14
 3.1. FEATURES.....	12
 3.2. DESIGN CONSTRAINTS.....	12
 3.3. ANALYSIS FINALIZATION SUBJECT TO CONSTRAINT.....	13

3.4. DESIGN FLOW.....	13
3.5. DESIGN SELECTION.....	14
3.6. IMPLEMENTATION PLAN.....	14
Chapter 4.	15-17
4.1. IMPLEMENATATION OF SOLUTION.....	15-17
Chapter 5. CONCLUSION AND FUTURE WORK.....	18
5.1. CONCLUSION.....	18
5.2. FUTURE WORK.....	18
References.....	19

ABSTRACT

The Feedback System is a web-based platform designed to collect, manage, and analyze feedback from students about faculty and courses. The system ensures secure login for students, faculty, and administrators by providing a structured process through which feedback can be submitted, stored, and evaluated in an efficient manner.

This system replaces traditional manual methods, eliminating delays, reducing errors, and enabling the generation of insightful reports. Built using JSP, Servlets, and MySQL, the platform allows:

- Faculty to view feedback received.
- Admins can generate comprehensive reports.

The project improves the quality of teaching and enhances decision-making for academic administration.

CHAPTER 1.

INTRODUCTION

1.1. Client Identification/Need Identification/Identification of relevant Contemporary issue

The client for this project is [CU], which aims to enhance the quality of education by monitoring faculty performance and obtaining structured feedback from students. The client requires a system that is:

- Easy to use for students and faculty.
- Capable of generating analytical insights for administrators.
- Secure, reliable, and scalable.
- Web-based to allow remote access from any standard browser.

The Feedback System aligns with the client's requirements by providing a modular architecture that separates the presentation layer, business logic layer, and data storage layer. This ensures maintainability and scalability while supporting multiple users simultaneously.

1.2. Identification of Problem

Manual feedback systems present several challenges:

- Feedback forms must be physically distributed, which delays submission and processing.
- Data entry errors are common when transferring information from paper to spreadsheets.
- Analysis of feedback, including rating averages and trends, requires significant manual effort.
- Student responses may be biased due to lack of anonymity in physical forms.
- Faculty have no direct access to feedback in real-time, delaying performance evaluation.

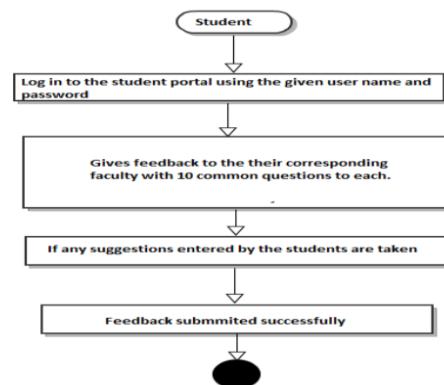
These limitations make it difficult to implement timely improvements in teaching quality. A web-based solution addresses these challenges by providing an automated, secure, and centralized feedback system.

1.3. Identification of Tasks

The key tasks of this project include:

- Developing a secure login system for students, faculty, and administrators.
- Creating dynamic JSP pages for feedback submission with input validation.
- Implementing servlets to handle database interactions efficiently.
- Designing a relational database in MySQL to store students, faculty, and feedback records.
- Implementing faculty dashboards to view consolidated feedback.
- Providing admin reports with analytics for decision-making.
- Ensuring security features such as hashed passwords and session management.
- Testing each module individually and performing integration testing.
- Designing a user-friendly interface to encourage student participation.

1.4. Timeline



1.5. Organization of the Report

The report is organized as follows:

Chapter 1: Introduction – Overview of the project, client, problem identification, tasks, and timeline.

Chapter 2: Literature Review – Analysis of existing solutions, proposed methods, and bibliometric insights.

Chapter 3: Design Flow – System architecture, features, constraints, finalization, and flow diagrams.

Chapter 4: Implementation of Solution – Technical implementation including database design, JSP pages, Servlets, and testing.

Chapter 5: Conclusion and Future Work – Summary of results, conclusions drawn, and future enhancements.

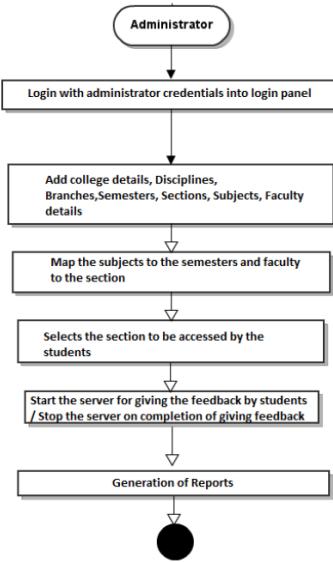
References – List of books, websites, and research papers consulted.

This organization ensures a clear understanding of the project from conceptualization to implementation.

CHAPTER 2.

LITERATURE REVIEW/BACKGROUND STUDY

2.1. Timeline of the reported problem



2.2. Proposed solutions

Automated feedback systems address the problems of manual systems:

- Web-Based Systems: Allow real-time feedback submission and viewing.
- Centralized Databases: Ensure secure storage, easy retrieval, and reduced redundancy.
- Analytical Dashboards: Facilitate trend analysis and performance monitoring.
- Role-Based Access: Ensures students, faculty, and administrators access only permitted data.
- Validation and Security: Reduces errors, protects sensitive data, and ensures feedback authenticity.

Our system implements all these features using JSP for front-end, Servlets for business logic, and MySQL for database management.

2.3. Bibliometric analysis

Studies indicate that:

- Automated feedback systems increase student participation rates.
- Centralized databases reduce data loss and duplication.
- Real-time analytics improves administrative decision-making by providing trends and statistical reports.
- Security and anonymity encourage students to provide honest feedback.

2.4. Review Summary

- Traditional manual feedback methods are inefficient and error-prone.
- Spreadsheet-based systems reduce manual effort but are not scalable.
- Web-based systems are efficient, scalable, and secure.
- Our proposed system integrates security, analytics and multi-role access, offering a comprehensive solution.

2.5. Problem Definition

Problem: Manual feedback collection is slow, insecure, and inaccessible remotely.

Solution: Develop a web-based automated system:

- Supports student feedback submission.
- Provides faculty dashboards.
- Enables administrators to generate reports.
- Ensures secure authentication and data integrity.

2.6. Goals/Objectives

Goals:

- Automate feedback collection and analysis.
- Reduce manual effort and human error.

- Provide analytical tools for academic decision-making.
- Ensure secure, role-based access.

Objectives:

- Design a normalized database schema for students, faculty, and feedback.
- Implement JSP pages and Servlets for all modules.
- Ensure input validation and data integrity.
- Provide a user-friendly interface for easy interaction.
- Test and validate system functionality thoroughly.
- Enable report generation for administrators.

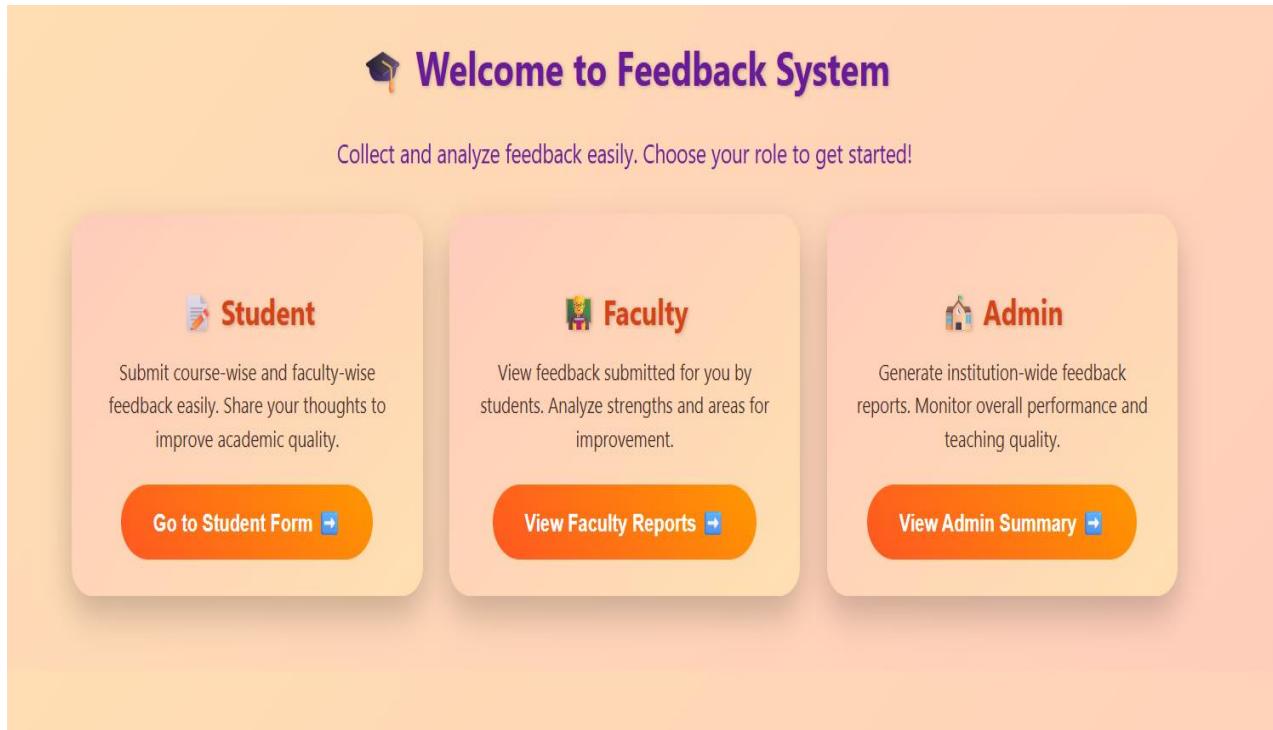
CHAPTER 3.

DESIGN FLOW/PROCESS

3.1. Evaluation & Selection of Specifications/Features

The Features section of the design describes the key functionalities provided to each type of user. For students, the system provides an intuitive web interface to submit feedback for multiple courses and faculty members. The feedback form includes rating options, comment boxes, and mandatory fields to ensure complete data collection. For faculty, the system provides a personalized dashboard displaying summarized feedback, average ratings, and detailed student comments. Administrators have access to analytical reports, trend charts, and export options, which help in monitoring faculty performance and making informed decisions. Additional features include role-based authentication, session management, data validation, and secure storage in a MySQL database to ensure data integrity and confidentiality.

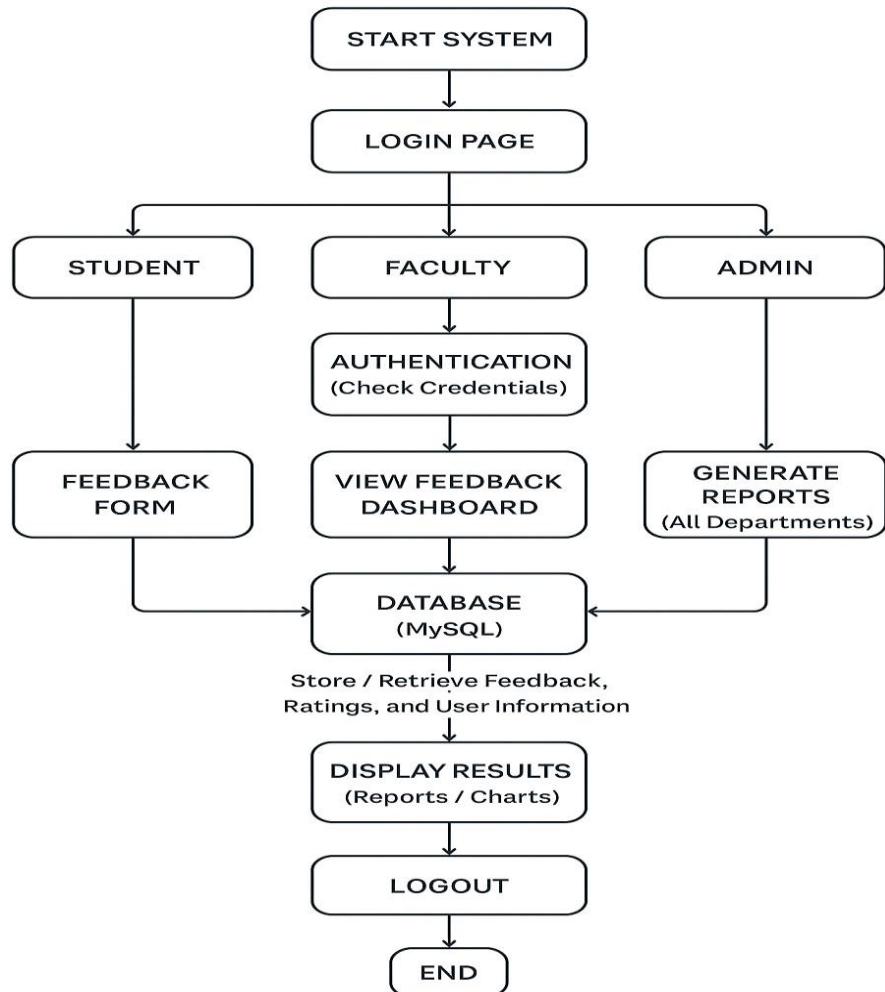
3.2. Design Constraints



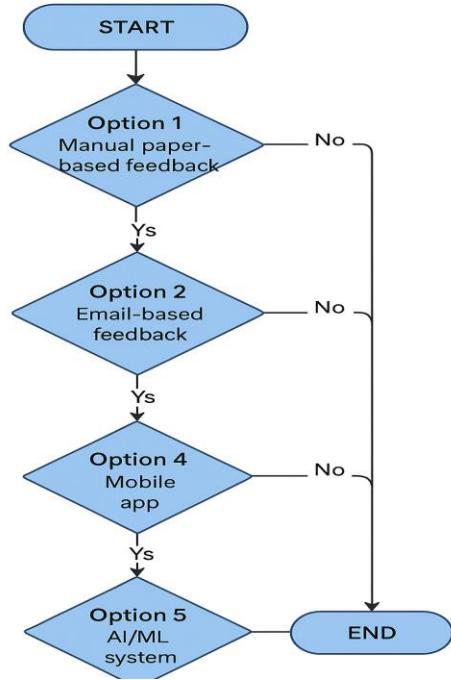
3.3. Analysis and Feature finalization subject to constraints

The Analysis Finalization Subject to Constraints section discusses how the design choices satisfy these constraints. For instance, using JSP and Servlets separates the presentation layer from business logic, making the system easier to maintain and scale. The relational database ensures efficient data storage and retrieval while enforcing referential integrity through foreign key relationships. Security measures such as hashed passwords, HTTPS connections, and session timeouts are incorporated to maintain confidentiality and prevent unauthorized access. The design also considers future enhancements, ensuring that new features can be added without major architectural changes.

3.4. Design Flow



3.5. Design Selection



3.6. Implementation plan

The implementation of the Feedback System was carried out in several structured phases to ensure smooth development and deployment.

The first phase involved **requirement analysis and planning**, where system objectives, user needs, and technical specifications were defined. The second phase focused on **database design** using MySQL, creating tables for students, faculty, courses, and feedback with proper relationships and constraints.

Next, **front-end development** was done using HTML, CSS, and JavaScript to create a simple, user-friendly interface for login, feedback submission, and dashboard views. The **back-end integration** used JSP and Servlets to handle authentication, feedback processing, and report generation securely.

Once development was complete, **testing** was conducted, including unit, integration, and user acceptance testing, to ensure accuracy and reliability. Finally, the system was **deployed** on the Apache Tomcat server, and documentation was prepared for users and administrators to facilitate easy operation and maintenance.

CHAPTER 4.

RESULTS ANALYSIS AND VALIDATION

4.1. Implementation of solution

1. System Overview

- The Feedback System was developed as a web-based application to automate and simplify the process of collecting and analyzing student feedback.
- It integrates multiple modules including user authentication, feedback submission, and report generation.

2. Technologies Used

- Frontend: HTML, CSS, JavaScript for creating user-friendly interfaces.
- Backend: JSP and Servlets for processing user requests.
- Database: MySQL for storing and managing data securely.
- Server: Apache Tomcat for deployment and hosting.

3. Database Implementation

- Tables such as Students, Faculty, Courses, and Feedback were created in MySQL.
- Primary and foreign keys were used to maintain relationships among tables.
- Proper indexing and constraints were applied to ensure efficient data retrieval and consistency.

4. Front-End Implementation

- The interface was designed to be simple, responsive, and user-friendly.
- Students can log in and submit feedback easily.
- Faculty can view feedback reports and performance summaries.

- Administrators can monitor institutional performance and generate department-wide reports.

5. Back-End Implementation

JSP and Servlets handle business logic and data communication between the front end and database.

The system includes modules for:

- Login authentication
- Feedback form submission
- Report generation
- Session management

Data validation ensures accurate and complete information is stored in the database.

6. Testing Phase

- Unit Testing: Each module (login, feedback, report) was tested individually.
- Integration Testing: Ensured smooth communication between modules.
- System Testing: Checked system performance, security, and reliability.
- User Acceptance Testing: Conducted with actual users to validate ease of use and functionality.

7. Deployment Phase

- The system was hosted on the Apache Tomcat Server.
- Configurations were set for multiple user logins and secure database connections.
- Backup systems were added for data safety and reliability.

8. Documentation and Training

- User manuals were prepared for students, faculty, and administrators.
- Training sessions were conducted to guide users on how to operate the system effectively.
- Technical documentation was prepared for future developers for maintenance and upgrades.

9. Outcome

- The implemented system is efficient, secure, and scalable.
- It automates feedback collection and reporting, reduces manual effort, and increases transparency.
- The system successfully bridges communication between students, faculty, and administrators.

CHAPTER 5 CONCLUSION AND FUTURE WORK

5.1. Conclusion

The development and implementation of the Feedback System have successfully achieved the main objective of creating an automated, efficient, and user-friendly platform for collecting and analyzing student feedback. This system eliminates manual paperwork and human errors, ensuring accurate and real-time feedback management.

The project provides separate interfaces for students, faculty, and administrators, making the process transparent and structured. Students can submit feedback easily, faculty can review performance reports, and administrators can generate institutional summaries for quality improvement.

By using technologies like HTML, CSS, JSP, Servlets, and MySQL, the system ensures a stable and secure environment for data handling. Proper authentication, validation, and reporting mechanisms enhance data reliability and security. The system demonstrates the effectiveness of automation in academic evaluation and contributes to improving the overall educational experience.

In conclusion, the Feedback System is a practical and scalable solution that streamlines the communication channel between students and faculty while promoting transparency, accountability, and continuous improvement within the institution.

5.2. Future work

Although the Feedback System fulfills its current objectives, several enhancements can be implemented in the future to increase its efficiency and usability:

- Integration of AI/ML-based sentiment analysis to analyze textual comments automatically and provide deeper insights.
- Addition of mobile app support to make the feedback system more accessible for students on smartphones and tablets.
- Implementation of data visualization dashboards for administrators using graphs and analytics tools.

REFERENCES

1. *Herbert Schildt*, “Java: The Complete Reference,” McGraw-Hill Education, 11th Edition.
2. *Paul Deitel & Harvey Deitel*, “Internet and World Wide Web: How to Program,” Pearson Education.
3. Oracle Java Documentation – <https://docs.oracle.com>
4. MySQL Developer Guide – <https://dev.mysql.com/doc>
5. W3Schools Online Tutorials – <https://www.w3schools.com>
6. TutorialsPoint – JSP and Servlets Basics – <https://www.tutorialspoint.com>
7. GeeksforGeeks – JSP and Web Development Concepts – <https://www.geeksforgeeks.org>
8. JavaTPoint – Servlet and JDBC Tutorials – <https://www.javatpoint.com>