**Student Feedback Management System**

## 1. Introduction

The Student Feedback Management System is a web-based application designed to collect, manage, and analyze feedback from students regarding courses, instructors, and academic experiences. This project aims to digitalize the traditional feedback process, enabling efficient data collection and real-time analysis.

## 2. Objectives

The main objectives of this project are:  
- To provide a user-friendly interface for students to submit feedback.  
- To allow administrators to view and analyze feedback efficiently.  
- To maintain a secure and structured database for storing feedback records.

## 3. Tools and Technologies Used

The following tools and technologies were used in the development of the project:  
- \*\*Frontend:\*\* HTML, CSS, JavaScript  
- \*\*Backend:\*\* PHP  
- \*\*Database:\*\* MySQL  
- \*\*Local Server:\*\* XAMPP (Apache & MySQL modules)  
- \*\*Port:\*\* 8080  
- \*\*Database File:\*\* database.sql

## 4. System Requirements

Hardware Requirements:  
- Processor: Intel Core i3 or higher  
- RAM: 4 GB minimum  
- Storage: 200 MB minimum for project files  
  
Software Requirements:  
- XAMPP Server  
- Web Browser (Chrome, Firefox, etc.)  
- Code Editor (VS Code, Sublime Text, etc.)

## 5. Implementation

The system is implemented using PHP as the backend scripting language and MySQL for database management. The application runs on XAMPP’s Apache server at http://localhost:8080/student\_feedback/. The database structure is imported from the database.sql file using phpMyAdmin.

## 6. Steps to Run the Project

1. Install and run XAMPP.  
2. Place the project folder inside the 'htdocs' directory of XAMPP.  
3. Open phpMyAdmin and create a new database (e.g., student\_feedback\_db).  
4. Import the database.sql file into the newly created database.  
5. Open a browser and go to http://localhost:8080/student\_feedback/.

## 7. Conclusion

The Student Feedback Management System successfully streamlines the process of collecting and managing feedback from students. By leveraging PHP and MySQL, the system ensures secure data handling and offers an intuitive interface for both students and administrators.