Technical Report

Instructor Management System

Desktop Application Development Overview:

The Instructor Management System is a desktop application designed to streamline the process of managing student information and marks for instructors. This project is developed using C# and Windows Forms, the application provides a user-friendly interface for instructors to input, edit, and visualize student data. The system utilizes a DataTable to store and manage the information, ensuring efficient data handling and after the instructor finalizes the marks assigned to the students he will submit the marks assigned to the students.

Programming Languages and Frameworks:

The primary programming language for this application is C#, chosen for its compatibility with Windows Forms and ease of integration with databases especially in the Visual Studio 2022. The application heavily relies on .NET for database connectivity, utilizing SQL Server for storing login credentials and student information. The choice of these technologies helps to integrate the overall application into the proper functioning desktop application.

Application Logic and Functionality:

The core logic of the application revolves around **CRUD (Create, Read, Update, Delete)** operations for student records. Instructors can enter student details, including name, university number, field of study, and scores in three subjects. The application automatically calculates the average and determines whether the student has passed or failed based on a predefined threshold.

The system also includes features such as data visualization, allowing instructors to view the average marks of all students. The logic ensures data integrity by validating numeric input and handling exceptions as well. Additionally, the application enables instructors to edit existing records through double-click functionality on the DataGridView. This functionality becomes an ease of use of the application which makes things easier for the instructor.

Testing and Quality Assurance:

To ensure the reliability and correctness of the application, rigorous testing was performed. Unit testing was employed for individual components and functions, checking the accuracy of calculations, data input validation, and database interactions. Integration testing was conducted to verify the seamless integration of different modules.

The DataGridView's double-click functionality and data retrieval from the DataTable were specifically tested to guarantee smooth user interactions. Extensive error handling was implemented to provide informative messages in case of unexpected situations. This testing approach aimed to deliver a user-friendly application.

Deployment and Distribution:

The application is designed for desktop deployment, serving the specific needs of instructors. The deployment process involves packaging the application and its dependencies into an installer for easy installation on Windows machines. The choice of Windows Forms simplifies deployment, as it aligns with the native Windows environment.

For distribution, the installer can be shared with instructors, ensuring a straightforward setup process. The system's compatibility with Windows operating systems enhances its accessibility, making it suitable for a wide user base.

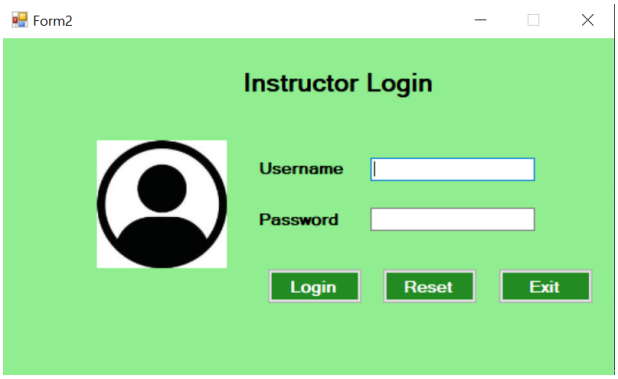
Security and Data Protection:

Security measures are implemented in the application to safeguard the sensitive information. The login system ensures that only authorized users can access the application. Passwords are stored securely, and SQL parameters are used to prevent SQL injection attacks. Regular updates and patches can be deployed to address emerging security threats, ensuring a secure environment for student data.

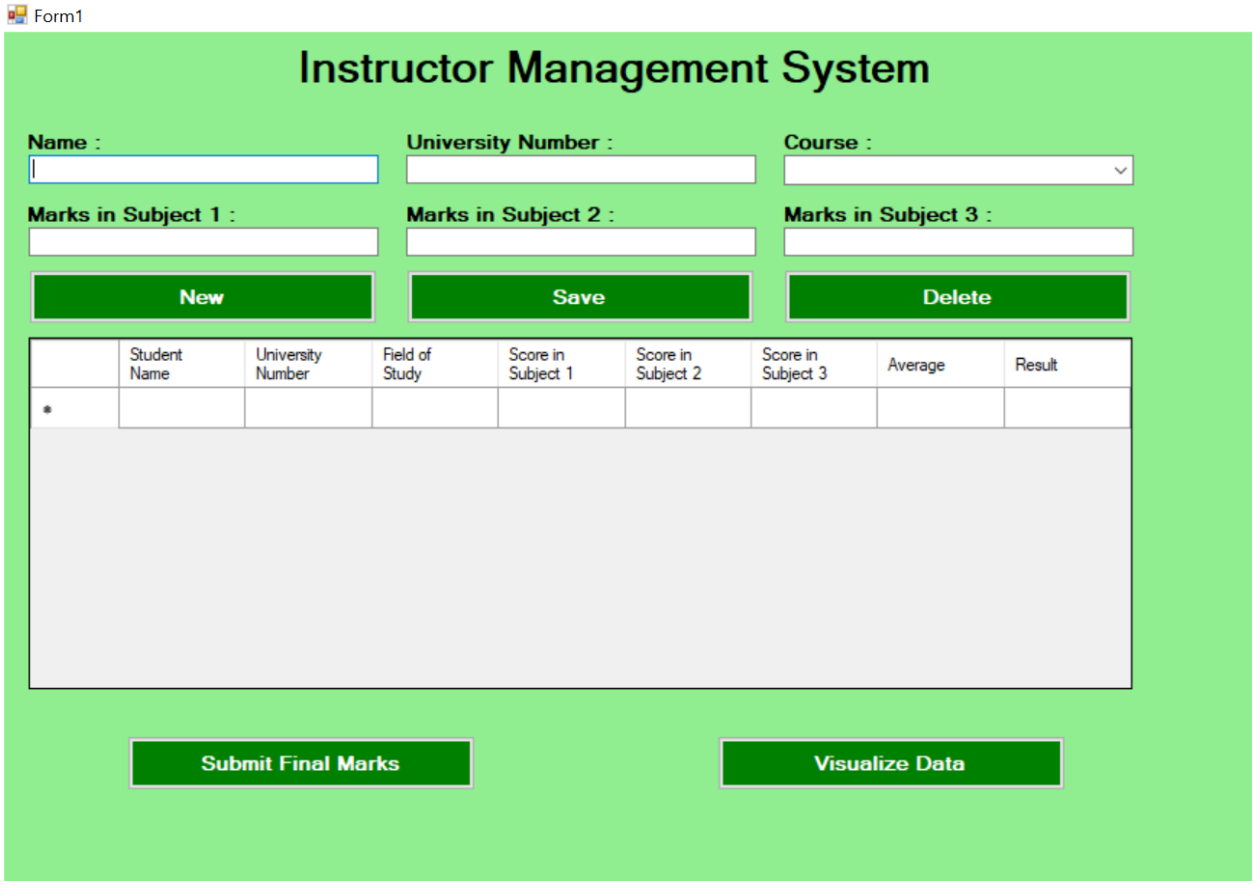
Case Studies and Examples:

The application's effectiveness is demonstrated through case studies of instructors successfully using it to manage student records. Instructors have reported increased efficiency in entering and retrieving student information. The visualization feature has proven valuable for instructors to quickly assess average marks and identify trends.

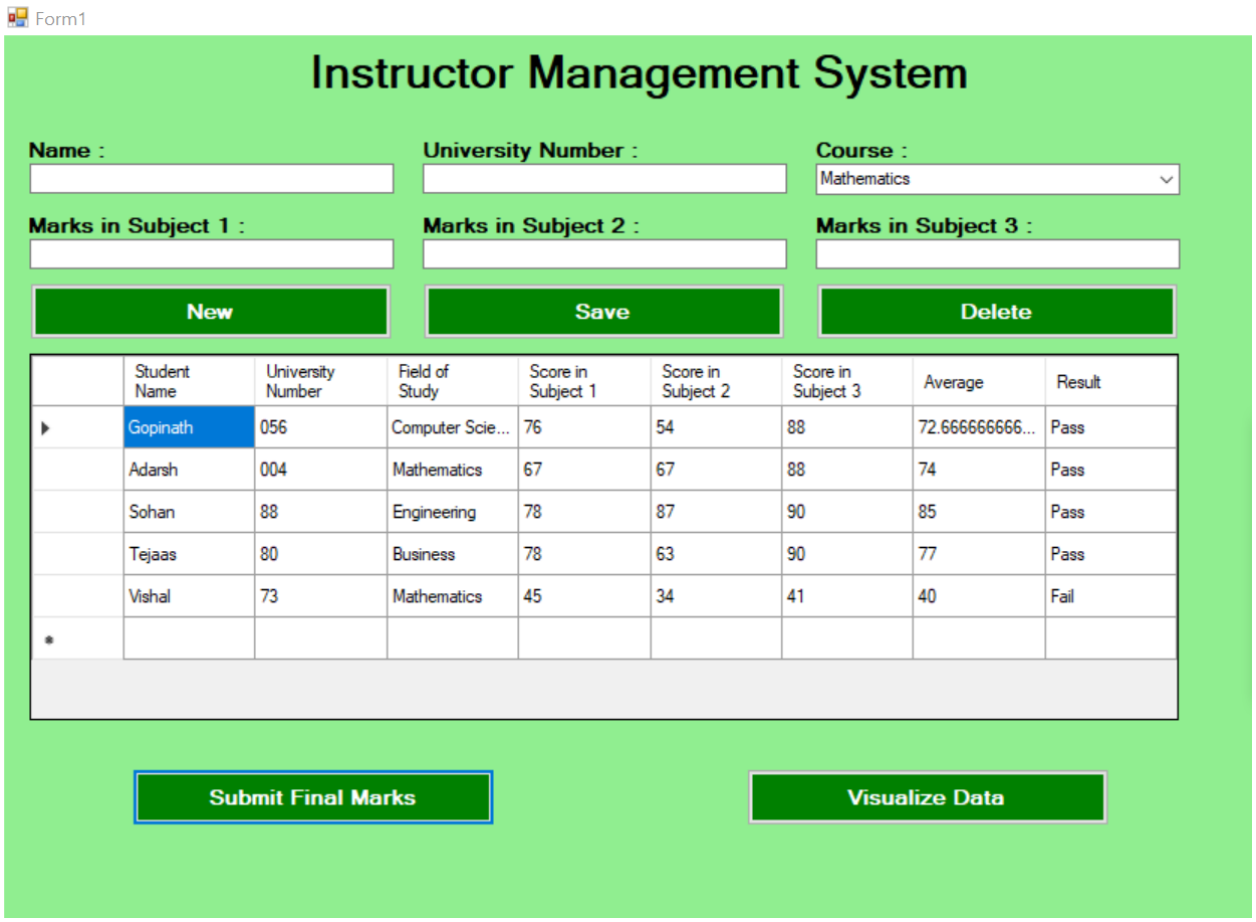
The Screenshots of the project along with the walk through like how the instructor would proceed with the application is shown below



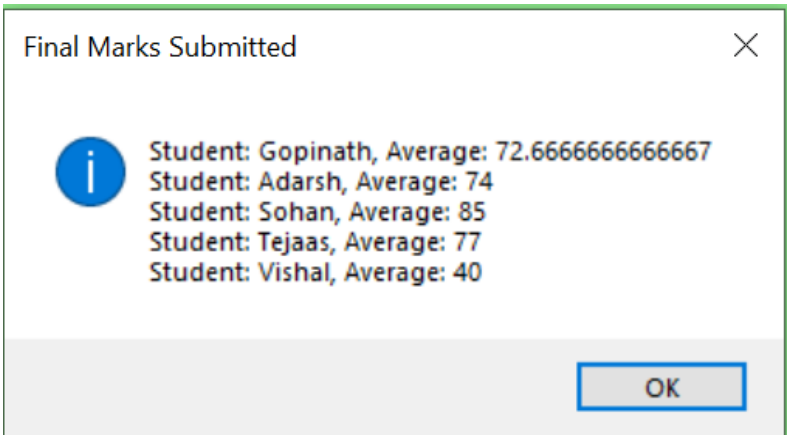
1) As soon as opening the project the application will ask the user for the Username and Password for the authentication purpose. It will validate the given username and the password with the SQL database if both the fields are entered correctly and by clicking the “login” button the dashboard is displayed otherwise an error message will pop up. If you click on the reset button all the text fields will be empty and by clicking on the exit button the application will quit.



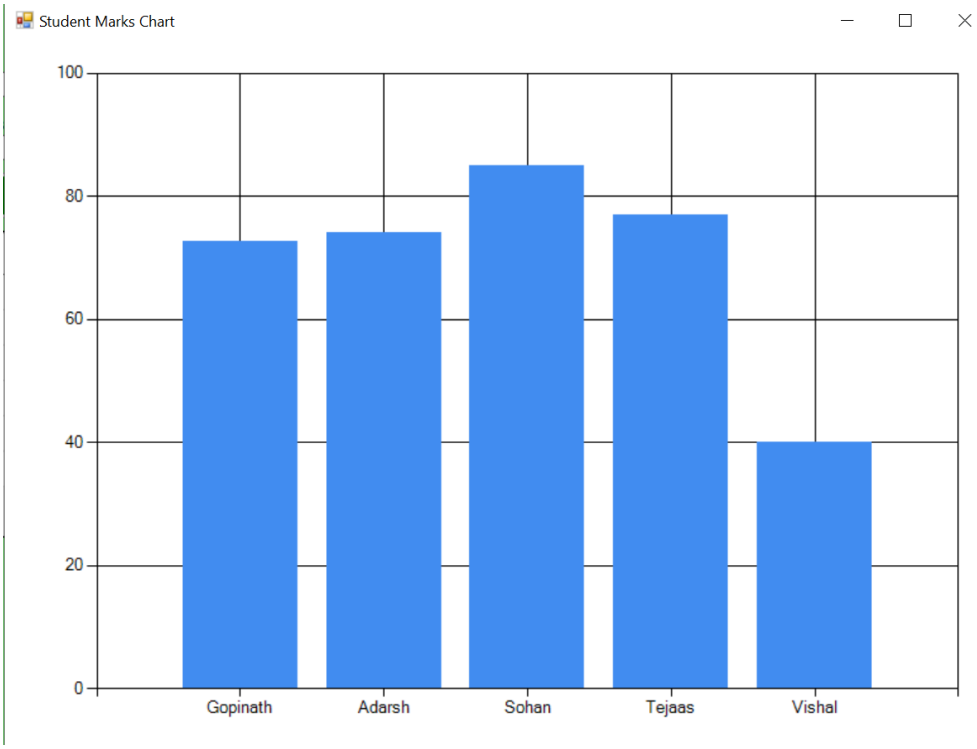
2) After clicking on the “login” button the Instructor will be able to see the above dashboard where he can enter all the values in the text field and when he clicks on the “Save” button the data will be inserted into the cell as shown below.



3) Once the data is inserted into the cell you can even delete or else you can even update the data by just double-clicking on the particular cell. And everytime you fill in all the fields you click on the save button the new record will be kept on inserting into the dashboard.



4) After submitting the Final marks the above pop-up will be shown just to make sure that the instructor has entered the correct details. It helps the instructor by showing only the name and average marks in the pop-up message because both of these details are easy to check rather than showing the entire data in the pop-up message.



5) If the instructor clicks on the “Visualize Data” button the above chart will pop-up. The above chart gives information about the average scores of the student as per the student's name.

Feedback from users has highlighted the intuitive design and ease of use as significant advantages. The application has streamlined the grading process, reducing manual effort and minimizing errors.

Conclusion:

The Instructor Management System represents a successful desktop application tailored for instructors managing student data. Leveraging C# and Windows Forms, the application provides a robust and user-friendly interface, promoting efficient data entry, retrieval, and visualization. Rigorous testing and security measures ensure reliability and protect sensitive information.

Future Trends and Challenges:

As technology evolves, future iterations of the application could explore integration with cloud services for enhanced accessibility and collaboration. Implementing additional features such as automated report generation or real-time collaboration could further enhance the system's capabilities.

Challenges may arise in adapting to evolving security threats and ensuring compatibility with newer operating systems. Regular updates and proactive monitoring of industry trends will be crucial to addressing these challenges and maintaining the application's effectiveness.

In conclusion, the Instructor Management System exemplifies the successful development of a desktop application tailored to specific user needs, with a foundation for future enhancements and adaptations.