CLO Based Evaluation System

Session: 2021 – 2025

Submitted by:

Muhammad Tahir 2021-CS-34

Supervised by:

Sir Samyan Wahla

Department of Computer Science and Engineering

University of Engineering and Technology

Lahore Pakistan

Acknowledgments

I express my gratitude to Allah Almighty for granting me the energy and ability to finish this project within the given time frame. I extend my appreciation to Mr. Nazeef Ul Haq for his supervision, guidance, and support throughout the project. His teachings and scheduling techniques helped me to produce the best possible project. I would also like to acknowledge Mr. Samyan Qayyum Wahla for his unwavering commitment to teaching Database Theory, which facilitated my project completion with ease. Furthermore, I feel fortunate to have friends who stood by me during the challenging moments of the project

Contents

List of Figures

List of Tables

Abstract

In this project a Desktop app namely 'CLO Based Evaluation System' has been developed. There are many evaluation systems which is managed by teachers and this project is also a reflection of it.

Teacher will manage student by adding their information and then will evaluate their assessments based on rubrics and rubrics level. He will also mark their attendance in this system and he will also be able to view different kinds of report to keep track on their performance.All the data will automatically be managed behind the scenes in database.

Project Details

1.1 Project Description:

This is a desktop software application designed to facilitate the assessment of students' work based on the rubrics defined for each Course Learning Outcome (CLO).This software is intended for single-user access by the course instructor, who has multiple functionalities available. The instructor can add multiple students to the course, as well as perform CRUD (Create, Read, Update, and Delete) operations on CLOs, their associated rubrics, and rubric levels.The instructor can also mark attendance for each student on the present day and can see attendence of previous days. Additionally, the instructor can create assignments and assess them based on rubric levels.The software generates several reports that the instructor can view. For example, in the CLO-wise report, the instructor can see the number of students who passed or failed each CLO, as well as the rubrics associated with each CLO and the percentage of marks obtained.In the Assessment-wise report, the instructor can view the obtained marks of each student in each assessment, along with the corresponding assesment component. The instructor can also view the results of each assessment, such as the number of students who passed and the average score obtained.Lastly, the Attendance report displays the attendance record for each student, as well as a day-wise report showing the number of students who were absent or present on a given day.

1.2 Business Need:

This Program is made to facilitate the professors or teacher of universities providing all the facilities in one platform. The Main purpose of the program is to provide a user-friendly interface and to lower the burdens of the staff to keep the manual records.

1.3 Motivation:

The main motivation for this project was to help to learn querying data inserted in a database. The focus was to teach us how to extract specific information from various relations within the database, ranging from simple to complex queries.To accomplish this goal, the project was implemented using C .NET Framework, with SQL Server integrated as the database management system. The software provides a user-friendly interface that facilitates the creation and execution of SQL queries to extract data from the database.Additionally, the project generates PDF reports using the iTextSharp library, which contain valuable information for the department to enhance the efficiency of the management system.

1.4 Project Actors:

There is only 1 users of this Program:

1.4.1 Teacher:

This program primarily caters to the needs of a single user Teacher who plays a vital role in managing all the essential data related to students, including their assessments and attendance records, among other modules.

Project Features

2.1 Project Features:

- He can View, Add, Delete, and Update students information .
- He can Manage CLOs, Rubrics and Rubric Levels.
- He can also mark the attendance of students.
- He can add multiple assessments based on rubrics and then can mark assessments based upon rubric levels of rubrics.
- He can also view reports of multiple kinds like CLO wise report, Assessment wise report, Attendance report etc.

Database Design

3.1 Lookup:

This table contains a list of lookup values used in other tables in the database. It has columns for LookupId (an auto-incremented primary key), Name (the name of the lookup value), and Category (the category of the lookup value).

3.2 Clo:

This table represents Course Learning Outcomes (CLOs) and has columns for Id (an auto-incremented primary key), Name (the name of the CLO), DateCreated (the date when the CLO was created), and DateUpdated (the date when the CLO was last updated).

3.3 ClassAttendance

This table contains information about the attendance of students in a class. It has columns for Id (an auto-incremented primary key) and AttendanceDate (the date when the attendance was taken).

3.4 Assessment:

This table contains information about assessments given to students. It has columns for Id (an auto-incremented primary key), Title (the title of the assessment), DateCreated (the date when the assessment was created), TotalMarks (the total marks that can be obtained in the assessment), and TotalWeightage (the weightage of the assessment)

3.5 Rubric:

This table represents the grading rubrics used for assessments. It has columns for Id (the primary key), Details (the details of the rubric), and CloId (the foreign key referencing the Clo table).

3.6 Student:

This table contains information about the students in the database. It has columns for Id (an auto-incremented primary key), FirstName (the first name of the student), LastName (the last name of the student), Contact (the contact information of the student), Email (the email address of the student), and RegistrationNumber (the registration number of the student)

3.7 Assessment Component:

This table stores information about the different components of an assessment. Each assessment may consist of multiple components, such as multiple-choice questions, essay questions, or practical tasks. This table includes fields such as component ID, component name, description, and weightage (percentage of the overall assessment score).

3.8 Rubric Level:

This table stores information about the different levels in a rubric used for grading an assessment component. Each rubric level represents a different level of achievement for a particular component. For example, in a rubric for Code Design question, the rubric levels might be "Excellent", "Good", "Satisfactory", "Poor", and "Very Poor". This table includes fields such as rubric level ID, level name, description, and the corresponding range of marks or scores.

3.9 StudentResult:

This table contains information about the results of each assessment for each student. Each record in the table corresponds to a single assessment component for a single student. The table includes the student's ID, the assessment component ID, the date of the assessment, the student's score on the component, and the rubric level achieved.

3.10 Database Design:

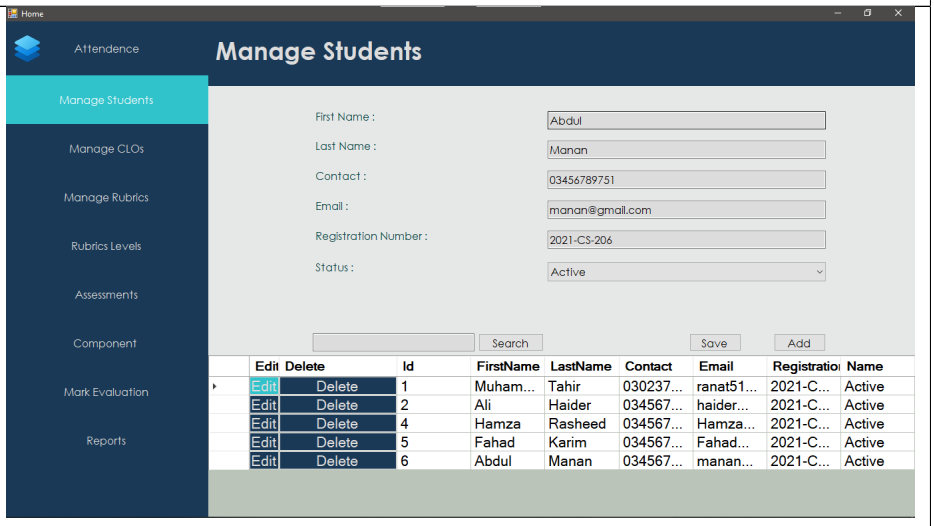
Below Image shows the database design of rubric based evaluation system.

FIGURE 3.1: Database design of project

Use Cases

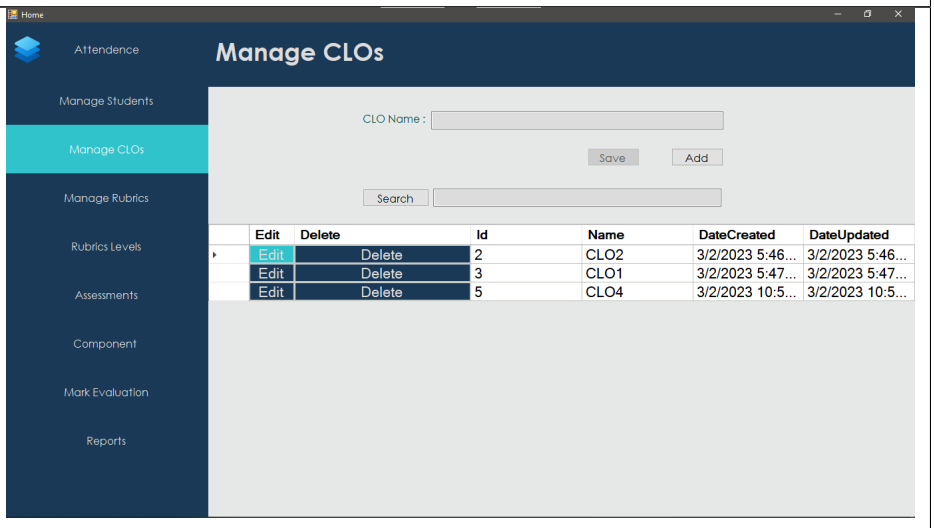
4.1 Use Case 1(Student CRUD):

Table 4.1: Student CRUD

Use Case ID	U01
Name	Student CRUD
Actor	Teacher
Description	Teacher will be able to add students and will be able to perform different operations such as search, view and delete.
GUI	

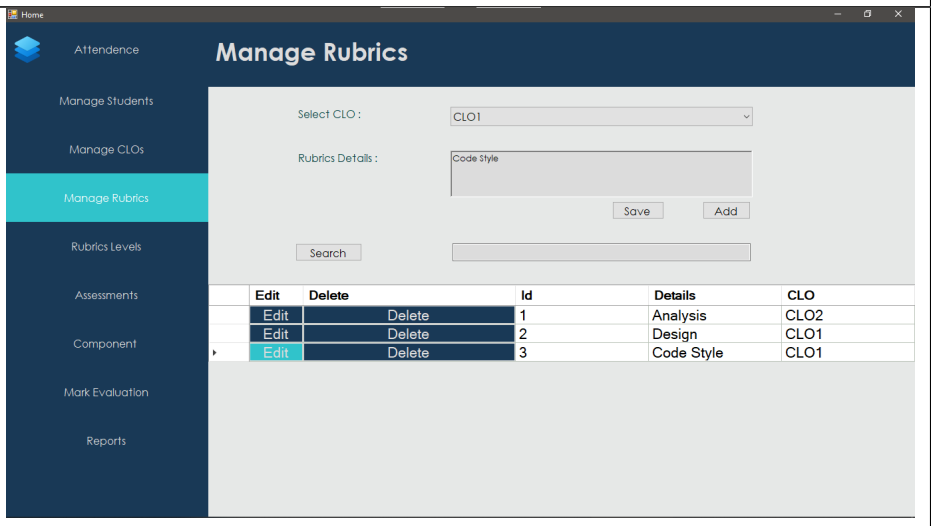
4.2 Use Case 2(CLO CRUD):

TABLE 4.2: Used to describe use-case 2 of CLO

Use Case ID	U02
Name	CLO CRUD
Actor	Teacher
Description	Teacher can add CLOs into his course and can edit, view and delete according to its need.
GUI	

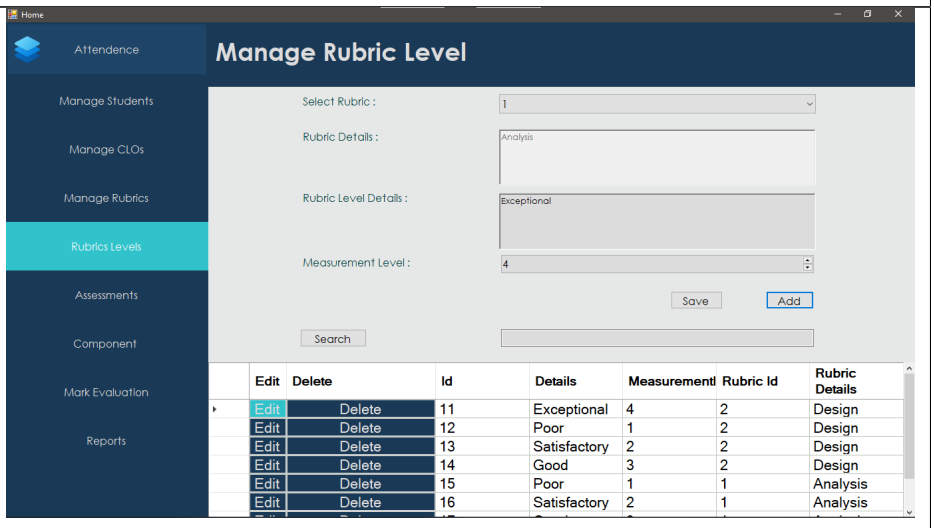
4.3 Use Case 3(Rubrics CRUD):

TABLE 4.3: Rubrics CRUD

Use Case ID	U03
Name	Rubrics CRUD
Actor	Teacher
Description	Teacher will bE adding the Rubrics of each Clo in this case. He will also perform CRUD Operations on rubrics as per need.
GUI	

4.4 Use Case 4(Rubrics Level CRUD):

TABLE 4.4: Rubrics Level CRUD

Use Case ID	U04
Name	Rubrics Level CRUD
Actor	Teacher
Description	Teacher will be adding Rubrics Level of each Rubrics and will be performing CRUD operations on rubric Level.
GUI	

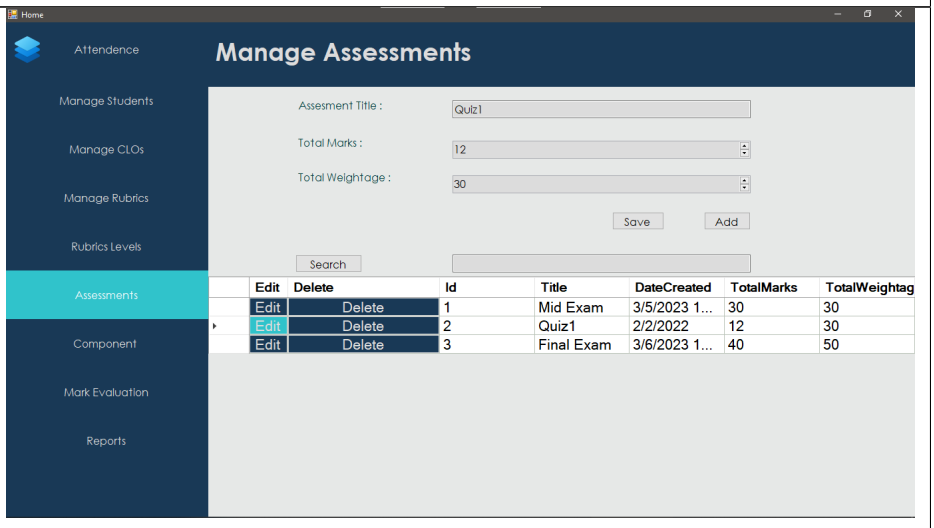
4.5 Use Case 5(Attendance):

Table 4.5: Mark Attendance

Use Case ID	U05
Name	Mark Attendence
Actor	Teacher
Description	Teacher will be able or edit to mark attendance of the students
GUI	**Attendance Report** RegistrationNumber | Name | 02/02/2021 | 11/03/2023 2021-CS-206 | Abdul | | Present 2021-CS-38 | Ali | | Absent 2021-CS-14 | Fahad | | Present 2021-CS-26 | Hamza | | Absent 2021-CS-34 | Muhammad | | Present

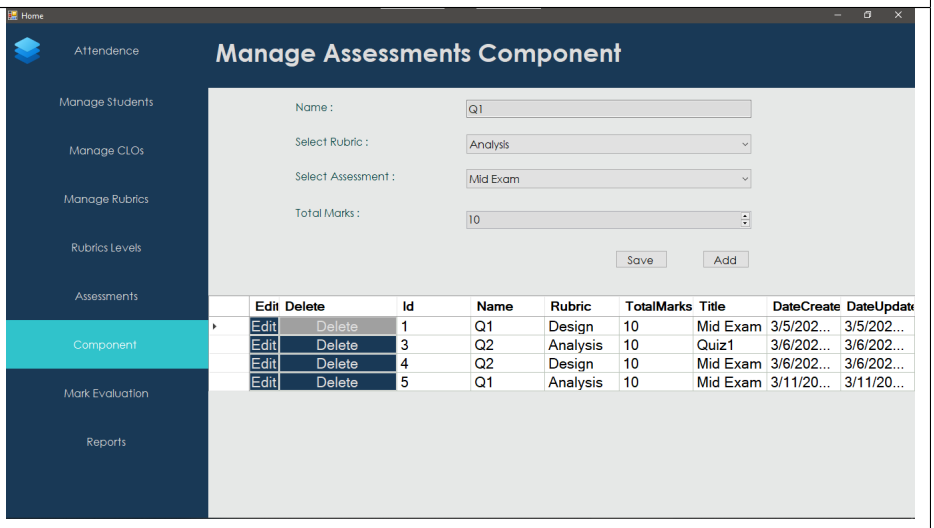
4.6 Use Case 6(Assessment CRUD):

TABLE 4.6: Assessment CRUD

Use Case ID	U06
Name	Make Assessment CRUD
Actor	Teacher
Description	He will be able to make and perform CRUD operations on assessments.
GUI	

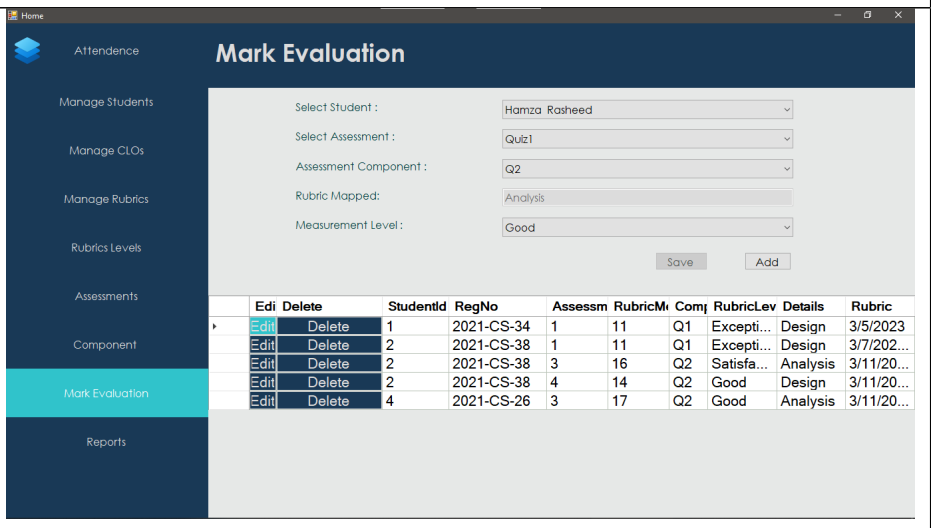
4.7 Use Case 7(Assessment Components CRUD):

TABLE 4.7: CRUD Assessment Components

Use Case ID	U07
Name	Assessment Components CRUD
Actor	Teacher
Description	Teacher will be able to add assessment components of each assessment and can perform CRUD on it.
GUI	

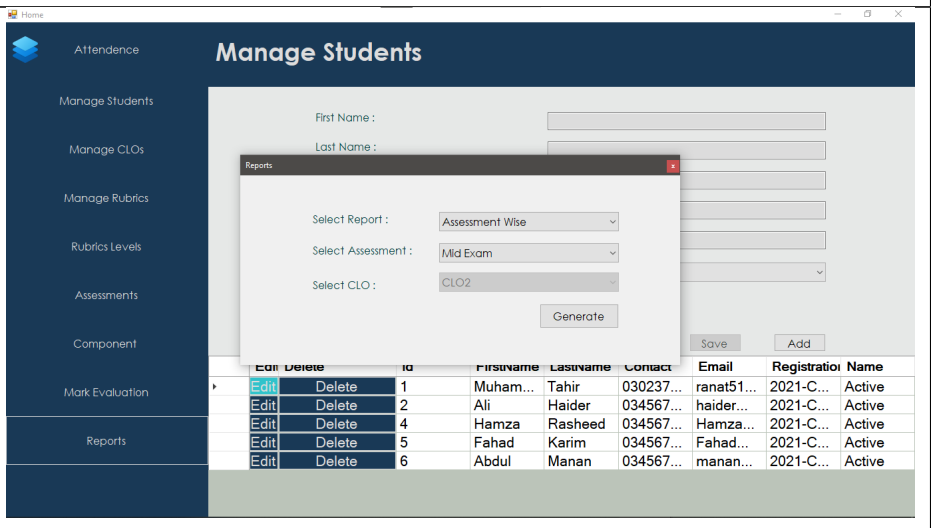
4.8 Use Case 8(Mark Evaluation):

TABLE 4.8: Mark Evaluation

Use Case ID	U08
Name	Mark Evaluation
Actor	Teacher
Description	Teacher will be evaluating all the students against each assessment component based on rubrics and rubric level and can delete it as well.
GUI	

4.9 Use Case 9(Generate Reports):

TABLE 4.9: Generate Reports

Use Case ID	U09
Name	Generate Reports
Actor	Teacher
Description	Teacher can generate different reports and can export as pdf to desired location.
GUI	

Operational Details

Here is a road map for using the program from scratch. Follow the following steps to use the program efficiently and smoothly:

- Start the program by clicking on the .exe file.
- Now click on Student Section button and then add students into the system.You can also edit,view and delete students from here.
- After adding the student go to the CLO portion by clicking on the CLO button and add CLO and perfrom CRUD on it..
- Now Add Rubrics and rubric levels and perform CRUD operations.
- After that add assessment component..
- Now Evaluate the students against each assessment component based on rubric and rubric levels.
- Teacher can also mark attendance of the students on daily basis.
- Teacher can also view different reports such as CLO report, Attendance report, Assessment report and Student report.

er of students who were absent or present on a given day.

Reports

In this project 4 reports are being generated based on the requirements include attendance report and the class result report and assessment wise report of each student in CLO or assessments.

6.1 Attendence Reports:

This report shows the attendence status of every student on each day.

Attendance Report

RegistrationNumber	Name	02/02/2021	11/03/2023
2021-CS-206	Abdul		Present
2021-CS-38	Ali		Absent
2021-CS-14	Fahad		Present
2021-CS-26	Hamza		Absent
2021-CS-34	Muhammad		Present

FIGURE 6.1: Attendence report

6.1.1 Query:

```
DECLARE @cols AS NVARCHAR(MAX), @query AS NVARCHAR(MAX)
WITH DateList AS (
```

```
SELECT DISTINCT FORMAT(CC.AttendanceDate, 'dd/MM/yyyy') AS
AttendanceDateFormatted
FROM ClassAttendance CC
)
SELECT @cols = STUFF((SELECT distinct ',' +
QUOTENAME(AttendanceDateFormatted)
FROM DateList
FOR XML PATH(''), TYPE
).value('.', 'NVARCHAR(MAX)')
,1,1,'')
SET @query = 'SELECT RegistrationNumber, Name, ' + @cols + '
FROM
(
SELECT s.RegistrationNumber, s.FirstName as Name,
FORMAT(ca.AttendanceDate, ''dd/MM/yyyy'') AS AttendanceDateFormatted,
ISNULL(L.Name, ''N/A'') AS AttendanceStatus
FROM student s
LEFT JOIN StudentAttendance sa ON s.Id = sa.StudentId
JOIN ClassAttendance CA ON CA.Id = SA.AttendanceId
LEFT JOIN Lookup L ON L.LookupId = SA.AttendanceStatus
) AS source_table
PIVOT
(
MAX(AttendanceStatus)
FOR AttendanceDateFormatted IN (' + @cols + ')
) AS pivot_table'
EXECUTE(@query)
```

6.2 Class Result:

This report shows the combined class result of every assessment.

Class Result

Student Id	Name	Total Marks	Total Obtained	Percentage obtained
2	Ali Haider	30	22.5	75.00
4	Hamza Rasheed	10	7.5	75.00
1	Muhammad Tahir	20	12.5	62.50

FIGURE 6.2: Class Result

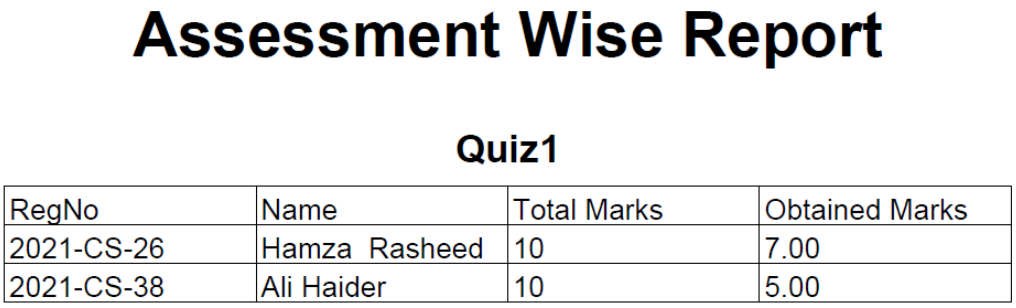
6.2.1 Query:

```
SELECT N.StudentId, N.Name, N.[Total Marks], N.[Total Obtained],
CAST((N.[Total Obtained] / N.[Total Marks]) * 100
AS NUMERIC(18,2)) AS [Percentage obtained]
FROM (
SELECT B.StudentId, B.Name, SUM(B.TotalMarks) AS [Total Marks],
SUM(B.[Obtained Marks]) AS [Total Obtained]
FROM (
SELECT T.StudentId, T.Name, T.TotalMarks,
(CAST(T.[Student obtained rub leve] AS FLOAT) / 4) * T.TotalMarks AS
[Obtained Marks] FROM (
SELECT S.StudentId, (St.FirstName + ' ' + St.LastName)AS [Name],
A.RubricId, A.TotalMarks, (SELECT MeasurementLevel FROM RubricLevel R
WHERE R.Id = S.RubricMeasurementId) AS [Student obtained rub leve]
FROM StudentResult S
JOIN AssessmentComponent A ON S.AssessmentComponentId = A.Id
JOIN Student ST ON St.Id = S.StudentId
```

```
) AS T
) AS B
GROUP BY B.StudentId, B.Name
) AS N
ORDER BY [Percentage obtained] DESC
```

6.3 Assessment Wise Report:

This report shows the assessment wise result of every student.



Assessment Wise Report

Quiz1

RegNo	Name	Total Marks	Obtained Marks
2021-CS-26	Hamza Rasheed	10	7.00
2021-CS-38	Ali Haider	10	5.00

FIGURE 6.3: Assessment Wise Result

6.3.1 Query:

```
SELECT Stu.RegistrationNumber, Stu.FirstName + ' ' + Stu.LastName
AS Name,
SUM(AC.TotalMarks) AS [Total Marks],
SUM(CAST(RL.MeasurementLevel * AC.TotalMarks / 4 AS decimal(10, 2)))
AS [Obtained Marks]
FROM StudentResult StuR
JOIN Student Stu ON Stu.Id = StuR.StudentId
JOIN RubricLevel RL ON StuR.RubricMeasurementId = RL.Id
JOIN AssessmentComponent AC ON AC.Id = StuR.AssessmentComponentId
JOIN Assessment A ON A.Id = AC.AssessmentId
```

```
JOIN Rubric R ON R.Id = RL.RubricId
WHERE A.Title = '" + assessment + "'
GROUP BY A.Title, Stu.RegistrationNumber,
Stu.FirstName, Stu.LastName, R.Details
```

This query calculates the percentage obtained by each student in their assessment results, and orders the result by the percentage in descending order. The query first selects data from the StudentResult, AssessmentComponent, and Student tables to get the student's name, rubric level, total marks, and obtained marks. Then, it uses subqueries to calculate the total marks and obtained marks for each student, and calculates the percentage obtained. Finally, the query orders the result by the percentage obtained in descending order

6.4 CLO Wise Report:

This report shows the CLO wise result of every student.

CLO Wise Report

CLO1

RegNo	Name	Total Marks	Obtained Marks
2021-CS-34	Muhammad Tahir	10	10.00
2021-CS-38	Ali Haider	20	17.00

FIGURE 6.4: CLO Wise Result

6.4.1 Query:

```
SELECT Stu.RegistrationNumber, Stu.FirstName + ' ' + Stu.LastName
AS [Student Name], SUM(AC.TotalMarks) AS [Total Marks],
SUM(CAST(RL.MeasurementLevel*AC.TotalMarks/4 AS decimal(10,2)))
AS [Obtained Marks]
FROM StudentResult StuR
JOIN Student Stu ON Stu.Id=StuR.StudentId
JOIN RubricLevel RL ON StuR.RubricMeasurementId=RL.Id
JOIN AssessmentComponent AC ON AC.Id=StuR.AssessmentComponentId
JOIN Assessment A ON A.Id=AC.AssessmentId
JOIN Rubric R ON R.Id=RL.RubricId
JOIN Clo ON Clo.Id=R.CloId
WHERE Clo.Id='2'
GROUP BY Stu.Id, Stu.RegistrationNumber, Stu.FirstName,
Stu.LastName, Clo.Id, Clo.Name
```

Testing

During the project, testing was performed at every step to ensure that data was linked correctly with every component. By resolving issues at lower levels, it ensured accurate testing at higher levels. Queries were tested through SQL Server and their outputs were checked before integrating them into the application's GUI. However, a challenge during testing was the lack of specific output declarations, making it difficult to determine query accuracy. Additionally, dealing with situations where data was deleted from one relation and not deleting associated information caused conflicts that required careful handling. To remove bugs from the code, the Visual Studio debugger was utilized. Finally, the application was made responsive by adjusting the anchors of objects.

Limitations

Following are the limitations of the project:

- Its a single user application.
- Login and Sign up form is missing.
- In this application we cannot delete specific information easily from database like before deleting CLO we must delete information from other all other table where information of that CLO in used.
- In this applicaiton once the attendence is marked we cannot edit it.
- Searching functionality is implemented only in specific modules.
- Generated reports are limited.
- Application is not linked with the LMS, thus data of students have to be added again in the system manually.

Future Work

The future work that can be done on the project are enlisted below:

- I want it to be a multi user application.
- Login and Sign up should be provided.
- User interface of the application will be improved.
- Attendence should be edited.
- Searching functinality shoduld be implemented properly.
- Number of reports should be increased.
- Applicaiton should be linked with lms and student can view their marks and attendence.

Conclusion

This midterm project was a comprehensive learning experience that taught us new concepts in database management systems. It was an eye-opening experience as it made us realize the tremendous effort and hard work that goes into creating a system that we use on a daily basis. The project helped us to think critically about the use of databases in real-world problems and how to query them effectively. We were able to apply the theoretical concepts we learned in class to solve practical problems efficiently. Prior to this, we used to rely on files to manage data, but now we understand the advantages and disadvantages of using databases. This knowledge has enabled us to analyze problems more effectively and improve the performance of our projects. The project also exposed us to various systems that utilize large amounts of data, making them more efficient. In summary, the project was an invaluable learning experience that taught us how to apply database management systems to solve real-world problems.