

Database Examination System Documentation

Prepared by:

- Abdallah Mostafa Draz Omara
- Omnia Abdelhamed Khalil Abdelhalem
- Mohamed Reyad Abdelraouf Amer
- Ahmed Abdelhakim Elsayed Mohamed
- Ezzat Mohamed Ezzat Shawky

1. Introduction

1.1 Overview

The Database Examination System is a structured database designed to manage and store examination-related data for ITI. The system efficiently handles student records, exam schedules, questions, and results within SQL Server. Data retrieval and reporting will be facilitated using SQL Server Reporting Services (SSRS) to generate insightful reports for administrators and instructors. The database ensures secure and efficient data management while providing structured queries and reports for examination tracking and analysis.

1.2 Purpose and Objectives

- Automate exam creation and management.
- Provide secure student authentication.
- Store and retrieve exam results efficiently.
- Facilitate real-time evaluation and reporting.

1.3 Scope

- User roles: Admin, Instructor, Student.
- Functions: Exam scheduling, question management, result processing.
- Security: Role-based access control, data encryption.

2. System Requirements

2.1 Hardware Requirements

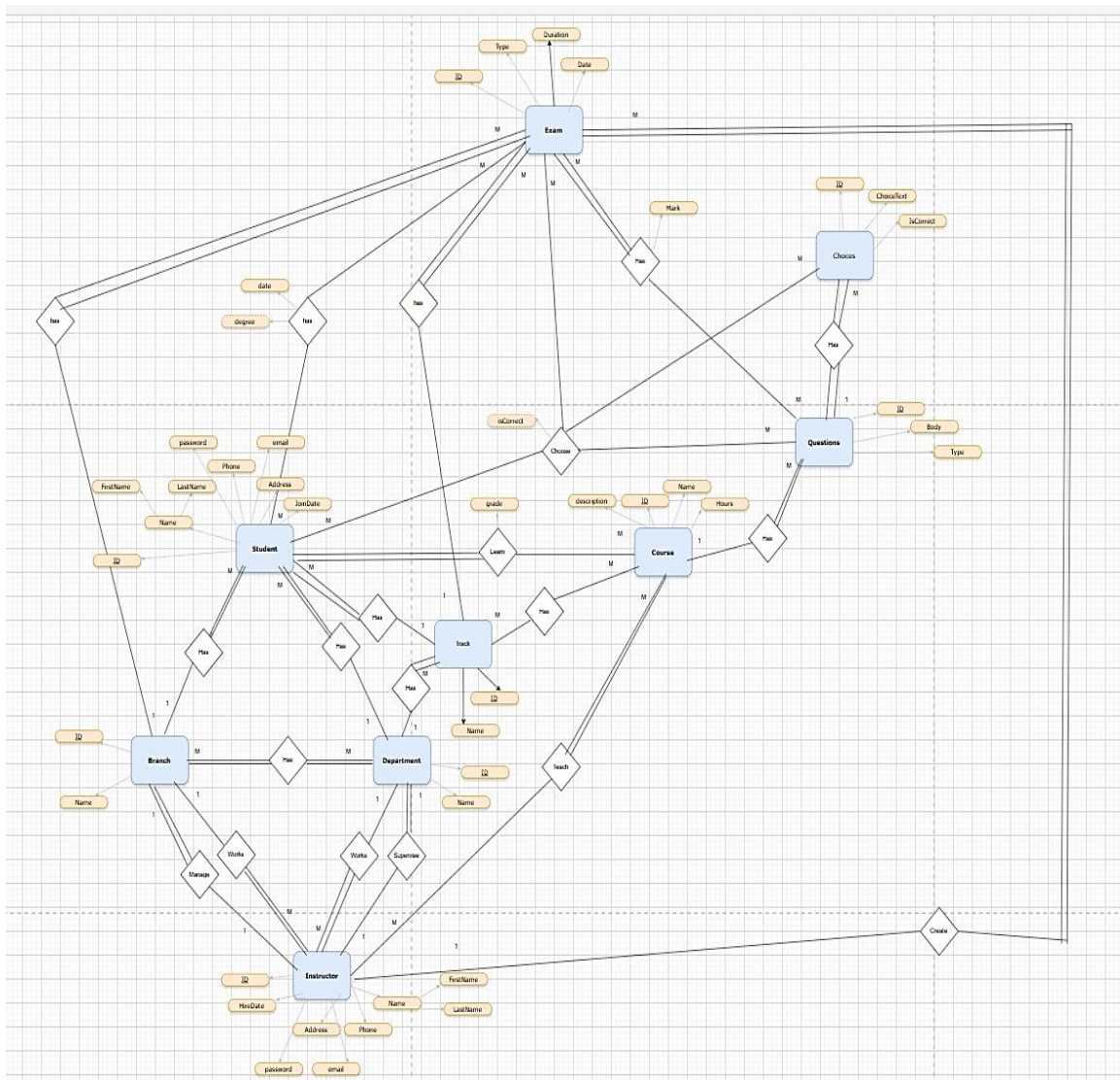
- Server with SQL Server installed.
- Minimum 8GB RAM, 100GB storage.

2.2 Software Requirements

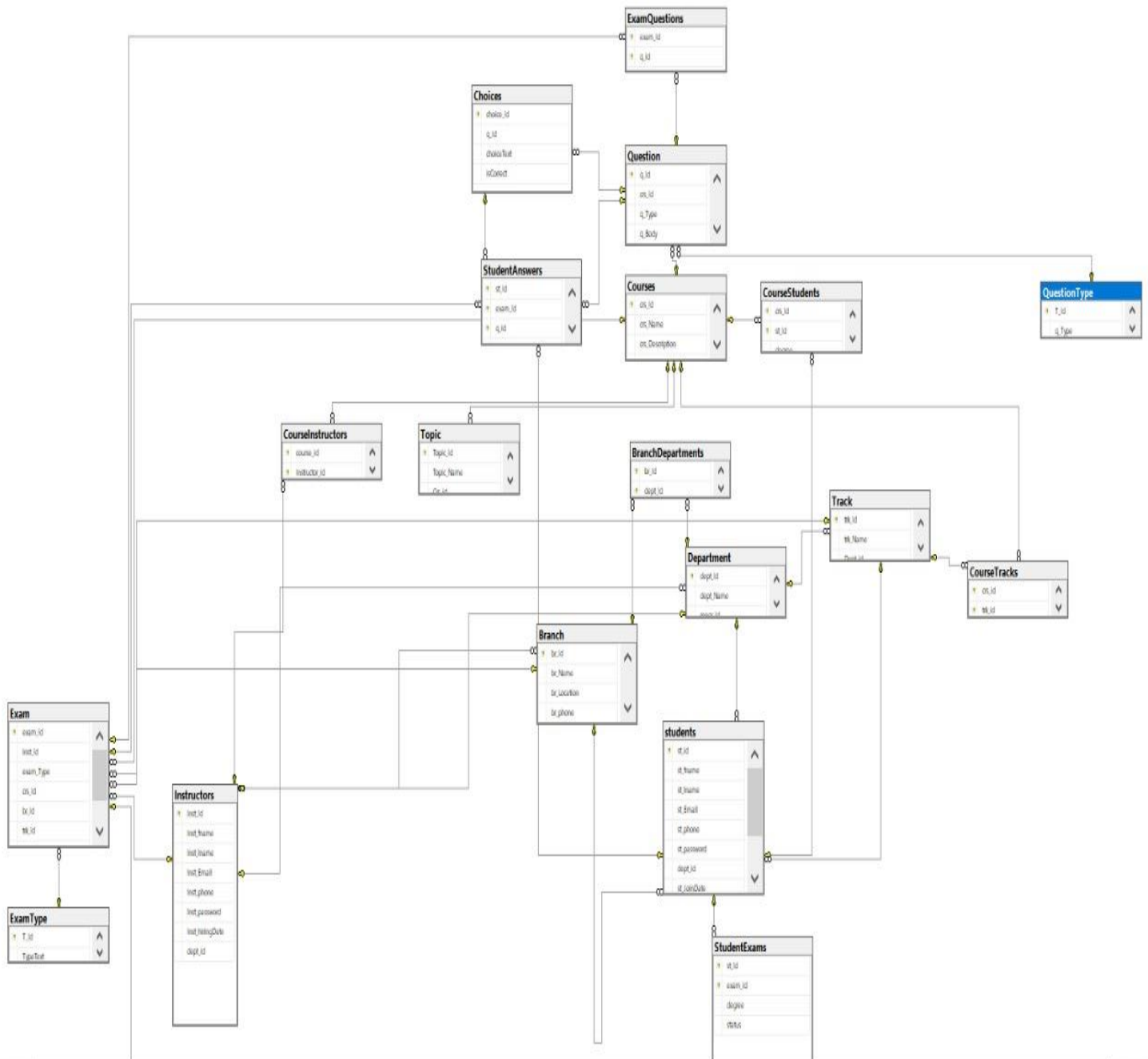
- SQL Server 2019 or later.
- Windows Server/Linux.

3. Database Design

3.1 Entity-Relationship Diagram (ERD)



3.2 Mapping



3.3 Database Schema

- Branch (BranchID, BranchName)
- BranchDepartments (BranchDepartmentID, BranchID, DepartmentID)
- Choices (ChoiceID, QuestionID, ChoiceText, IsCorrect)
- Courses (CourseID, CourseName, TrackID)
- CourseInstructors (CourseInstructorID, CourseID, InstructorID)
- CourseStudents (CourseStudentID, CourseID, StudentID)
- CourseTracks (TrackID, TrackName)
- Department (DepartmentID, DepartmentName)
- Exams (ExamID, Title, Date, Duration, ExamTypeID, InstructorID)
- ExamQuestions (ExamQuestionID, ExamID, QuestionID)
- ExamType (ExamTypeID, TypeName)
- Instructors (InstructorID, Name, Email, PhoneNumber)
- Questions (QuestionID, QuestionTypeID, QuestionText)
- QuestionType (QuestionTypeID, TypeName)
- StudentAnswers (StudentAnswerID, StudentExamID, QuestionID, SelectedChoiceID)
- StudentExams (StudentExamID, StudentID, ExamID, Score)
- Students (StudentID, Name, Email, PhoneNumber)
- Tracks (TrackID, TrackName)

4. Implementation Details

4.1 Stored Procedures and Functions

- BranchInsert: Adds a new branch.
- BranchUpdate: Updates an existing branch
- BranchDelete: Deletes a branch.
- ChoicesInsert: Adds choices for a question.
- ChoicesUpdate: Updates choices for a question.
- ChoicesDelete: Deletes choices for a question.
- CourseStudentsSelect: Retrieves students in a course.
- StudentExamsInsert: Registers a student for an exam.
- StudentExamsUpdate: Updates the score of a student.
- QuestionsSelectAll: Retrieves all questions.

With the same Stored Procedures, they are added to each table in the same way.

5. Queries and Reports

5.1 Sample Queries

Fetch all students registered for an exam:

```
```sql
SELECT S.StudentID, S.Name, E.ExamID, E.Title
FROM Students S
JOIN StudentExams SE ON S.StudentID = SE.StudentID
JOIN Exams E ON SE.ExamID = E.ExamID;
```
```

Retrieve top-performing students:

```
```sql
SELECT TOP 10 S.StudentID, S.Name, AVG(SE.Score) AS AverageScore
FROM Students S
JOIN StudentExams SE ON S.StudentID = SE.StudentID
GROUP BY S.StudentID, S.Name
ORDER BY AverageScore DESC;
```
```

Generate detailed exam reports:

```
```sql
SELECT E.Title, Q.QuestionText, SA.SelectedChoiceID, C.ChoiceText
FROM Exams E
JOIN ExamQuestions EQ ON E.ExamID = EQ.ExamID
JOIN Questions Q ON EQ.QuestionID = Q.QuestionID
JOIN StudentAnswers SA ON Q.QuestionID = SA.QuestionID
JOIN Choices C ON SA.SelectedChoiceID = C.ChoiceID;
```
```

6. Security and Backup

6.1 User Roles and Permissions

- Admin: Full access.
- Instructor: Manage exams and results.
- Student: Take exams and view results.

6.2 Backup and Recovery Strategies

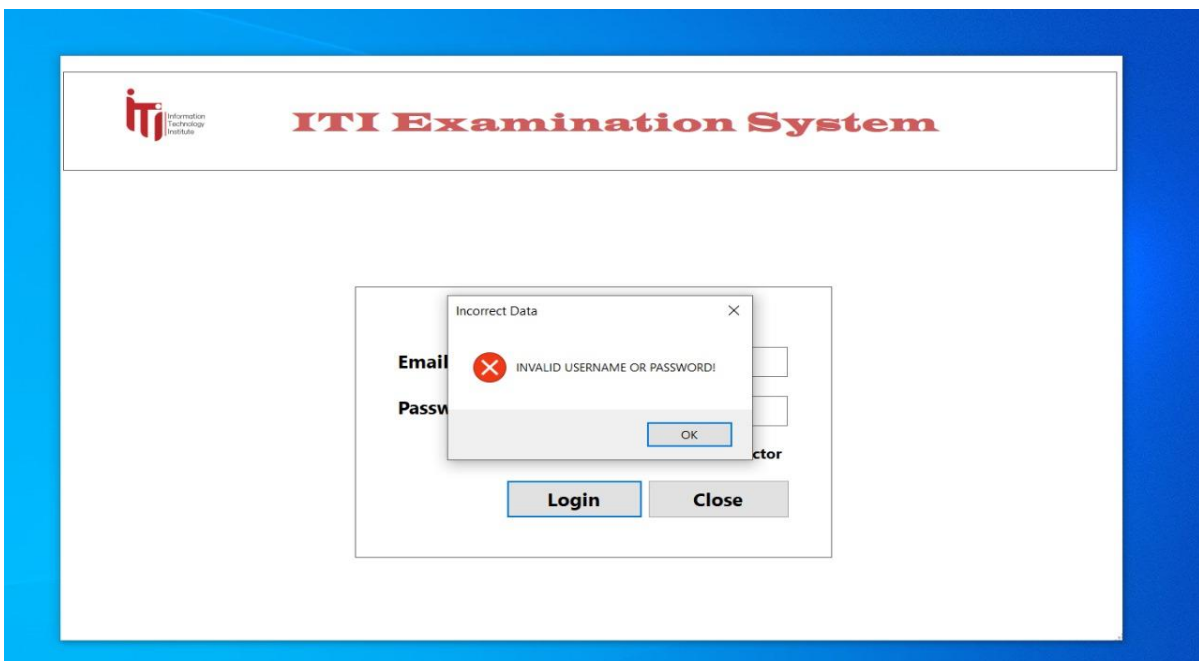
- Scheduled backups.
- Disaster recovery plan.

7. Future Enhancements

- AI-based question generation.
- Enhanced reporting and analytics.
- Desktop app integration.

Desktop Application Integration

- To enhance user interaction and facilitate database management, a desktop application has been developed. This application provides a user-friendly interface for administrators, instructors, and students to interact with the database seamlessly. The desktop application communicates with the SQL Server database using secure and optimized queries, ensuring efficient and real-time data access.





ITI Examination System

Branches Departments Tracks Instructors Students Courses Topics Questions Exams

Branches

Branch Name:

Location:

Phone:

Manager:

Save

Search

Update

Delete

Logout



ITI Examination System

Branches Departments Tracks Instructors Students Courses Topics Questions Exams

Departments

Department Name:

Manager:

abdallah hany
ahmed alaa
ali gamal
ayman fathy

Save

Search

Update

Delete

Logout



ITI Examination System

Branches Departments Tracks Instructors Students Courses Topics Questions Exams

Departments

Department Name

Manager:

Saved Data

Department Added Successfully

OK

Save

Search

Update

Delete

Logout



ITI Examination System

Branches Departments Tracks Instructors Students Courses Topics Questions Exams

Tracks

Track Name:

Fullstack Web Development .NET

Department:

Web Development

Web Development

Save

Search

Update

Delete

Logout



ITI Examination System

Branches Departments Tracks **Instructors** Students Courses Topics Questions Exams

Instructors

First Name: Last Name:
Email: Password:
Phone: Confirm:
Hire Date: Department:

Tracks:

| Track Name | |
|------------|-------------------------------------|
| Python | <input checked="" type="checkbox"/> |

Save

Search

Update

Delete

Logout

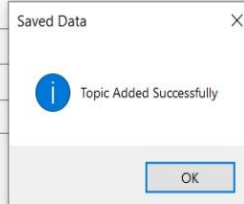


ITI Examination System

Branches Departments Tracks Instructors Students Courses **Topics** Questions Exams

Topics In Course

Course Name:
Topic Name:



Save

Search

Update

Delete

Logout



ITI Examination System

Branches Departments Tracks Instructors Students Courses Topics Questions Exams

Courses

Course Name:

Hours:

Description:

Tracks:

| Track Name | |
|---------------------|-------------------------------------|
| AI | <input type="checkbox"/> |
| PHP | <input type="checkbox"/> |
| Python | <input type="checkbox"/> |
| FrontEnd | <input type="checkbox"/> |
| Fullstack Web De... | <input checked="" type="checkbox"/> |

Instructors:

| Instructor Name | |
|-----------------|-------------------------------------|
| Ayman Lotfy | <input checked="" type="checkbox"/> |
| Ahmed Samir | <input type="checkbox"/> |
| Eman Monier | <input checked="" type="checkbox"/> |

Save

Search

Update

Delete

Logout



ITI Examination System

Branches Departments Tracks Instructors Students Courses Topics Questions Exams

Questions

Course Name:

Question Type:

Marks:

Question:

Correct Answer

True ☒

False ☐


Save

Search

Update

Delete

Logout



ITI Examination System

[Branches](#) [Departments](#) [Tracks](#) [Instructors](#) [Students](#) [Courses](#) [Topics](#) [Questions](#) [Exams](#)

Questions

Course Name: OOP Using C#

Question Type: Choose One

Marks: 5

Question: Which of the following is NOT oop Principle ?

Choices

1. Abstraction ☐

2. Encapsulation ☐

3. Casting ☒

4. Inheritance ☐

Save
Search
Update
Delete

Logout

8- Reports using (SSRS)

- Report that returns the students' information according to Department No parameter:

Department No 1
View Report

| st id | st fname | st lname | st Email | st phone | dept id | dept Name |
|-------|----------|----------|----------------|-----------|---------|-----------|
| 1 | mona | mostafa | mona@gmail.com | 010203040 | 1 | OS |
| 3 | anas | ahmed | anas@gmail.com | 010504080 | 1 | OS |

- Report that takes the student ID and returns the grades of the student in all courses:

| | | |
|----------------|-------|-------------|
| Student ID | 1 | View Report |
| 1 | of 1 | 100% |
| mona | | |
| crs Name | Grade | |
| data structure | 50 | |
| php | 40 | |

- number of students per course:

| | | |
|----------------|---------------|-------------|
| Instructor ID | 2 | View Report |
| 1 | of 1 | 100% |
| Course Name | Student Count | |
| data structure | 3 | |
| html | 0 | |

- Report that takes course ID and returns its topics:

Crs Id

[View Report](#)

1 of 1 100%

| Topic Name |
|------------|
| advanced |
| OOP |

- Report that takes exam number and returns Questions in it:

examid

[View Report](#)

1 of 1 100%

| q Body | q marks |
|--------|---------|
| q1body | 5 |
| q2body | 4 |
| q3body | 3 |

- Report that takes exam number and the student ID then returns the Questions in this exam with the student answers:

Exam ID

2

Student ID

1

View Report

1 of 1

100%

1

mona mostafa

| | |
|--------|------------|
| q body | choicetext |
| q1body | false |
| q2body | b- choice2 |
| q3body | false |

Thank you ,,,