



Information Technology Institute

Full Stack .Net Track

Examination System DB

Presented by:

Mohamed Nabil Mohamed Abd El-Hafiz

Mohamed Hosni Thabet Ahmed

Hatem Mamdouh Abdelagani Abd elgayed

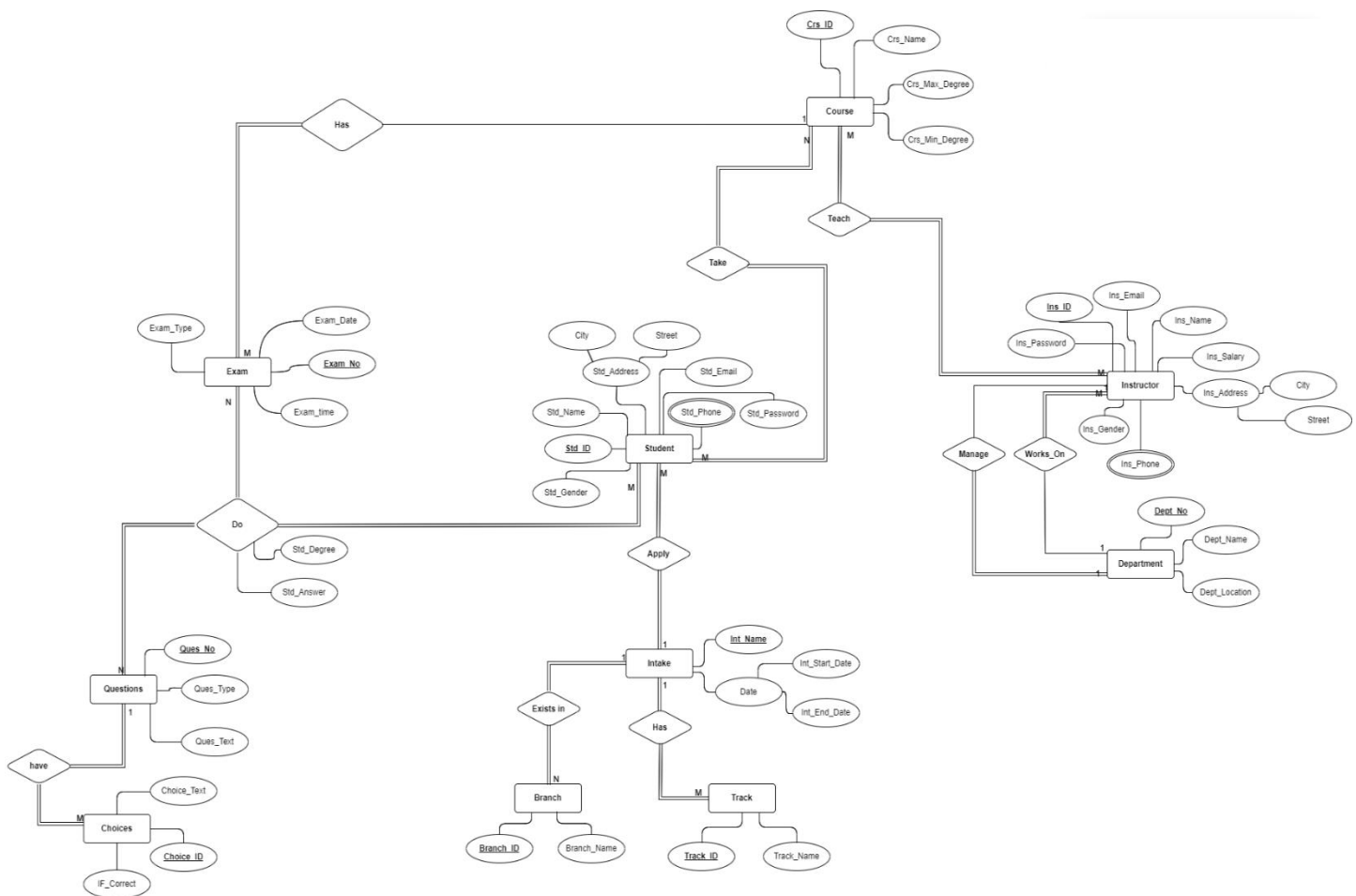
Mohamed Ashraf Fouad Mohamed

Lamiaa Mohamed Abd El-Mohsen

Supervisor: Sarah Salah

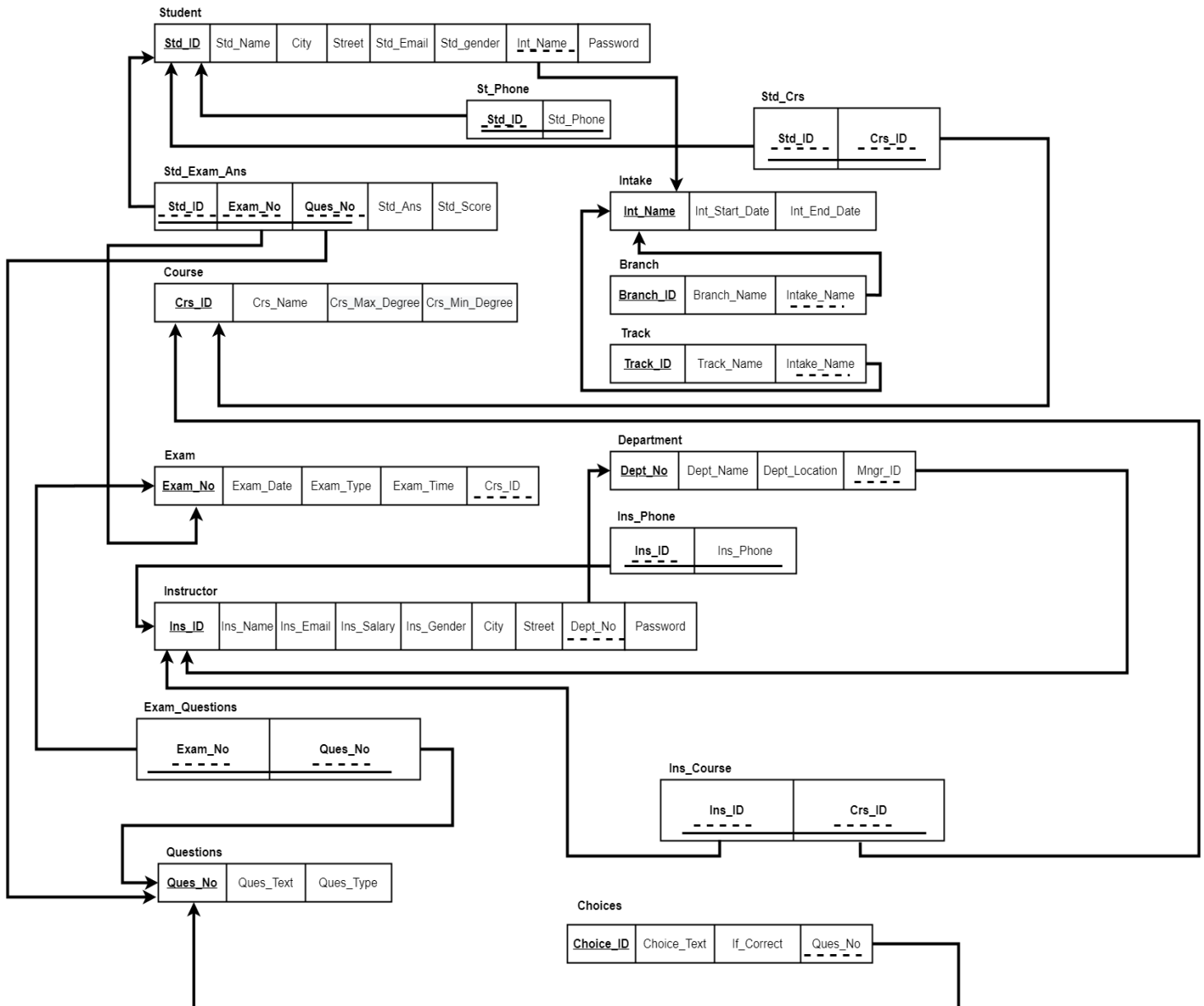
DB DESIGN (ERD)

Our Examination system ERD includes entities such as Student, department, Exam, questions, Instructors, courses, Branch, and their relationships to each other.



DB DESIGN (MAPPING)

Mapping refers to the process of creating an association between different entities or attributes in the system



DB IMPLEMENTATION

Creating a database typically involves creating tables to store information about the. We used the Bulk Insert command. This command allows for the efficient insertion of large amounts of data into tables from a data file. The data file format is CSV in our system

```
11  -- Table Course
12  CREATE table Course
13  (
14      Crs_Id int Primary key,
15      Crs_Name nvarchar(30) not null,
16      Max_Degree int,
17      Min_Degree int
18  )
19
20  -- Table Question
21  CREATE table Question
22  (
23      Ques_No int primary key ,
24      Ques_Text nvarchar(max),
25      Ques_Type varchar(20)
26  )
```

Bulk Insert command:

```
1  -- insert Course data
2  Bulk insert Course
3  From 'C:\ITI\SQL\Course.csv'
4  with (
5      FIELDTERMINATOR = ',',
6      ROWTERMINATOR = '\n',
7      FIRSTROW = 2
8  )
9
10 -- Insert Question data
11 Bulk insert Question
12 From 'C:\ITI\SQL\Question.csv'
13 with (
14     FIELDTERMINATOR = ',',
15     ROWTERMINATOR = '\n',
16     FIRSTROW = 2
17 )
```

For each table in the DB at least have insert, update, delete, and select Stored Procedure

Insert Stored Procedure: insert new row into the instructor_Course table. It can take input parameters such as the @Ins_ID, @Crs_ID and check if user have permission to do this

```
354  /* Insert Instructors Course */
355  alter procedure Proc_Add_New_Instructor_Course  @manegerId int ,
356                                                    @Ins_Id int ,
357                                                    @Crs_Id int
358  as
359  begin try
360      if Not Exists(select 1 from Vw_Show_Manger_ID where Mngr_Id = @manege
361      begin
362          print 'You do not have access to this table '
363      end
364      else
365          begin
366              begin try
367                  insert into Ins_Course
368                      values( @Ins_Id , @Crs_Id)
369                  print 'Insert Done Successfully'
370              end try
```

Update Stored Procedure: update existing rows in the Branch table. It can take input parameters @Branch_ID, @Branch_Name, @Intake_Name, and check if user have permission to do this.

```
CREATE OR ALTER PROCEDURE UpdateBranchIfManager
    @ManagerID INT,
    @Branch_Id INT,
    @Branch_Name NVARCHAR(50),
    @Intake_Name VARCHAR(30)
AS
BEGIN
    IF EXISTS (SELECT 1 FROM Department WHERE Mngr_ID = @ManagerID)
    BEGIN
        IF EXISTS (SELECT 1 FROM Branch WHERE Branch_Id = @Branch_Id)
        BEGIN
```

Delete Stored Procedure: delete row from the instructor table parameter @Ins_ID,
and check if user have permission to do this.

```
/* Delete Instructors */
create procedure Proc_Delete_Instructor @Manger_Id int , @Ins_Id int
as
begin try
    if Not Exists(select 1 from Vw_Show_Manger_ID where Mngr_Id = @Manger
begin
    select 'You do not have access to this table ' [Error Msg]
end
else
begin
    if Not Exists(select 1 from Instructor where Ins_ID = @Ins_Id and
begin
    select 'This ID Not Found' As [Error Msg]
end
end
```

Create Exam is a stored procedure created to create exam, to course id with the exam number if the Course id is not existed print an error message, @Exam_No, @Crs_ID, @Exam_Time, @NumberOf_TF_Question, @NumberOfMCQ_Question parameters

```
--Create Exam
alter procedure SP_Add_Exam
    @Instructor_Id int,
    @Crs_ID int, @Exam_No int,
    @Exam_Time int, @NumberOfMCQ_Question int,
    @NumberOf_TF_Question int, @Exam_Type varchar(20)
as
begin try
    if not exists (select * from Course where Crs_Id = @Crs_ID)
        begin
            select 'Selected course not exist' AS 'ErrorMessage'
        end
    else if not exists (
        select 1
        from Ins_Course
```

Student answer sheet is a stored procedure created to get the score of students in student id with the exam number if the student id is not existed print an error message, @Exam_No, @std_ID

```
--Student Score
alter procedure SP_Student_AnswerSheet
    @Std_ID int, @Exam_No int
as
begin try
    if not exists (select * from Exam where Exam_No = @Exam_No)
    begin
        select 'The Exam does not exist' AS 'ErrorMessage'
    end
    else if not exists (select * from Student where Std_ID = @Std_ID)
    begin
        select 'The Student does not exist' AS 'ErrorMessage'
    end
    else if not exists (select * from Std_Exam_Ans where (Std_ID =
    begin
```

Full exam is a stored procedure created to show the full exam using cursor if the exam id is not existed print an error message, @Exam_No

```
--Exam Structure With Cursor
alter procedure SP_FullExam
    @Exam_ID int
as
begin try
    if not exists(select * from Exam where Exam_No = @Exam_ID)
    begin
        select 'The Exam does not exist' AS 'ErrorMessage'
    end
    else
    begin
        declare @Content int
        create table #ShowExam(
            Questions varchar(max)
        )
        declare ExamCursor cursor
        for
```

Trig Audit Update Ins Salary is a Trigger created to get the data of instructor if he tries to change his salary, and insert data in another table.

```
create table Instructor_Salary
(
    ID int ,
    Ins_Name nvarchar(30),
    OldSalary int,
    NewSalary int ,
    ModifiedBy nvarchar(max),
    ModifiedDate date
)
create trigger Trig_Audit_Update_Ins_Salary
on Instructor
after update
as
begin
    if(update(Ins_Salary))
        begin
            insert into Instructor_Salary
```

View Exam Full Details a view to Show Full Details About Exam .

```
1 create view Vw_Exam_Full_Details
2 as
3 select E.Exam_No , E.Exam_Date , E.Exam_Type , E.Exam_Time ,
4 B.Branch_Name , B.Intake_Name ,
5 T.Track_Name ,
6 C.Crs_Name
7 from Branch B, Intake Intk, Track T , Student S , Course C , Std_Crs SC , Exam E
8 where B.Intake_Name = Intk.Int_Name
9 and T.Intake_Name = Intk.Int_Name
10 and S.Int_Name = Intk.Int_Name
11 and SC.Crs_ID = C.Crs_Id
12 and C.Crs_Id = E.Crs_ID
```


<u>Team Member</u>	<u>Tasks</u>
1-Mohamed Nabil	<ul style="list-style-type: none"> • Create Table Department. • Create Table Instructor. • Create Table Ins_Course. • Create Table Ins_Phone. • procedure SP_Add_Exam • procedure SP_Student_AnswerSheet • procedure SP_StudentSolution • procedure SP_FullExam (<u>With Cursor</u>) • procedure SP_Student_Ex_Answers • procedure Show_Departments • Procedure Insert_Department • Procedure Update_Departments • procedure Delete_Department • procedure GetStdInfoByDeptNo • procedure DetailsCoursesStudentbyInstuctorID • View View_Student_Department • Index Index_Department • Index Index_Instructor (Name) • Index Index_Instructor2 (E-mail) • Bulk Insert • Account and Permissions

2-Mohamed Hosny

- Create Table Std_Phone.
- Create Table Std_Exam_Ans.
- Create Table Std_Crs.
- **procedure** Proc_Show_Instructor_Details
- **procedure** Proc_Add_New_Instructor
- **procedure** Proc_Delete_Instructor
- **procedure** Proc_Update_Instructor
- **procedure** Proc_Show_Instructor_Phones
- **procedure** Proc_Add_New_Instructor_Phone
- **Procedure** Proc_Delete_Instructor_phone
- **Procedure** Proc_Update_Instructor_Phone
- **procedure** Proc_Show_Instructor_Course
- **procedure** Proc_Add_New_Instructor_Course
- **procedure** Proc_Delete_Instructor_Course
- **View** Vw_Show_All_Instructor
- **View** Vw_Show_Manger_ID
- **View** Vw_Show_All_Instructor_Phone
- **View** Vw_Show_All_Instructor_Course
- **View** Vw_Training_Manager_Details
- **View** Vw_Instructor_Works_Dept
- **View**
Vw_Show_All_InstructorName_CourseName_DeptName
- **View** Vw_Exam_Full_Details
- **View** Vw_CountInstructor_ForAllDept
- **View** Vw_Show_Instructor_Salary_Tab
- **Index** Ins_Gender_Index (Instructor)
- **Index** Ins_Email_Index (Instructor)
- **Index** Ins_Salary_Index (Instructor)
- **Index** Ins_City_Index (Instructor)
- **Index** Ins_Street_Index (Instructor)
- **Trigger** Trig_Ins_Crs_Insert_Delete
- **Trigger** Trig_Audit_Update_Ins_Salary
- **Trigger** Trig_Audit_delete_Ins

3-Mohamed Ashraf

- Create Table Intake
- Create Table Exam_Questions
- Create Table Student
- Procedure PS_validate_id
- Procedure SP_dispaly_Intake_Table_forTM
- Procedure SP_dispaly_Intake_Table_forST
- Procedure SP_dispaly_Intake_Table_forINS
- Procedure SP_DisplayIntake_Table
- Procedure SP_update_Intake_Table_forTM
- Procedure SP_delete_Intake_Table_forTM
- Procedure SP_delete_Intake_Table
- Procedure SP_insert_Intake_Table_forTM
- Procedure SP_insert_Intake_Table
- Procedure SP_Display_Std_Crs_Tablefor_TM
- Procedure SP_DisplayStd_Crs_Table
- Procedure SP_DisplayStd_Crs_Table
- Procedure SP_Std_Crs_Table_forTM
- Procedure SP_update_std_crs_Table
- Procedure SP_delete_Std_Crs_Table_forTM
- Procedure SP_delete_Std_Crs_Table
- Procedure SP_Insert_Std_Crs_TableforTM
- Procedure SP_insert_STD_CRS_Table
- Procedure SP_Insert_Std_Phone_Table
- View StudentInfo
- View intakeInfo
- View Std_PhoneInfo
- View std_crsInfo
- Index index_Int_Start_Date
- Index index_Int_End_Date
- Index index_student_std_Name
- Index index_student_std_Gender
- Index index_student_std_Email
- Index index_student_City
- Index index_student_Street

4-Hatem Mamdouh

- Create Table Track.
- Create Table Choices.
- Create Table Exam.
- `procedure` AddCourseIfManager
- `procedure` UpdateCourseIfManager
- `procedure` DeleteCourseIfManager
- `procedure` AddQuestion
- `procedure` UpdateQuestion
- `procedure` DeleteQuestion
- `Procedure` AddBranchIfManager
- `Procedure` UpdateBranchIfManager
- `procedure` DeleteBranchIfManager
- `procedure` sp_GetInstructorDetails
- `Function` dbo.GetBranchAndManager
- `Function` dbo.GetInstructorsAndCourses
- `View` vw_AllQuestions
- `View` vw_InstructorsCourses
- `Index` idx_Crs_Name (Course)
- `Index` idx_Max_Degree
- `Index` idx_Min_Degree
- `Trigger` trg_Branch_Delete
- `Trigger` trg_Branch_Update
- `Trigger` trg_Branch_Insert
- `Trigger` trg_Course_Update
- `Trigger` trg_Course_Insert
- `Trigger` trg_Question_Delete
- `Trigger` trg_Question_Update
- `Trigger` trg_Question_Insert
- `Trigger` trg_Instructor_Delete
- `Trigger` trg_Instructor_Update
- `Trigger` trg_Instructor_Insert

5- Lamiaa
Mohamed

- Create Table Courses.
- Create Table Questions.
- Create Table Branch.
- `procedure` Proc_Choice_Details
- `procedure` Proc_Add_Choice
- `procedure` Proc_Delete_Choice
- `procedure` Proc_Update_Choices
- `procedure` Proc_Track_Details
- `procedure` Proc_Add_Track
- `Procedure` Proc_Delete_Track
- `Procedure` Proc_Update_Track
- `procedure` Track_Details
- `procedure` Proc_Add_Student
- `procedure` Proc_Delete_Student
- `procedure` Proc_Update_Student
- `view` Vw_Choises_Details
- `view` Vw_Instructor_ID
- `view` Vw_Track_Details
- `view` Vw_Manager_ID
- `view` Vw_Student_Details