

Ahsanullah University of Science & Technology

Department of Computer Science & Engineering



[Teachers' Information and Assessment System]

CSE 3224

Information System Design
&
Software Engineering Lab

Submitted By:

Syed Sanzam	16.01.04.042
Nusrat Hossain	16.01.04.045
Saiful Islam	16.01.04.040

Date of Submission: **24 March, 2019**

Introduction:

An entity-relationship diagram (ERD) is crucial to creating a good database design. It is one of the parameters that indicates the robustness of a system. **Teachers' Information and Assessment System** certainly requires a strong database design as it will deal with a lot of different kind of data from different kinds of users. Understanding the data flow is also of great concerns for the developers themselves.

Names of Entities with Keys:

USER	
Primary Key	UserID

COMMENT	
Primary Key	CommnetID
Foreign Key	UserID, TeacherID

TEACHER_ATTRIBUTE	
Primary Key	AttributeID

TEACHER_ATTRIBUTE_STATUS	
Primary Key	AttributeStatusID

JUNCTION_TEACHER_ATTRIBUTE	
Primary Key	JunctionTeacherAttributeID
Foreign Key	TeacherID, AttributeID, AttributeStatusID

TEACHER	
Primary Key	TeacherID
Foreign Key	DepartmentID, StatusID

STATUS	
Primary Key	StatusID

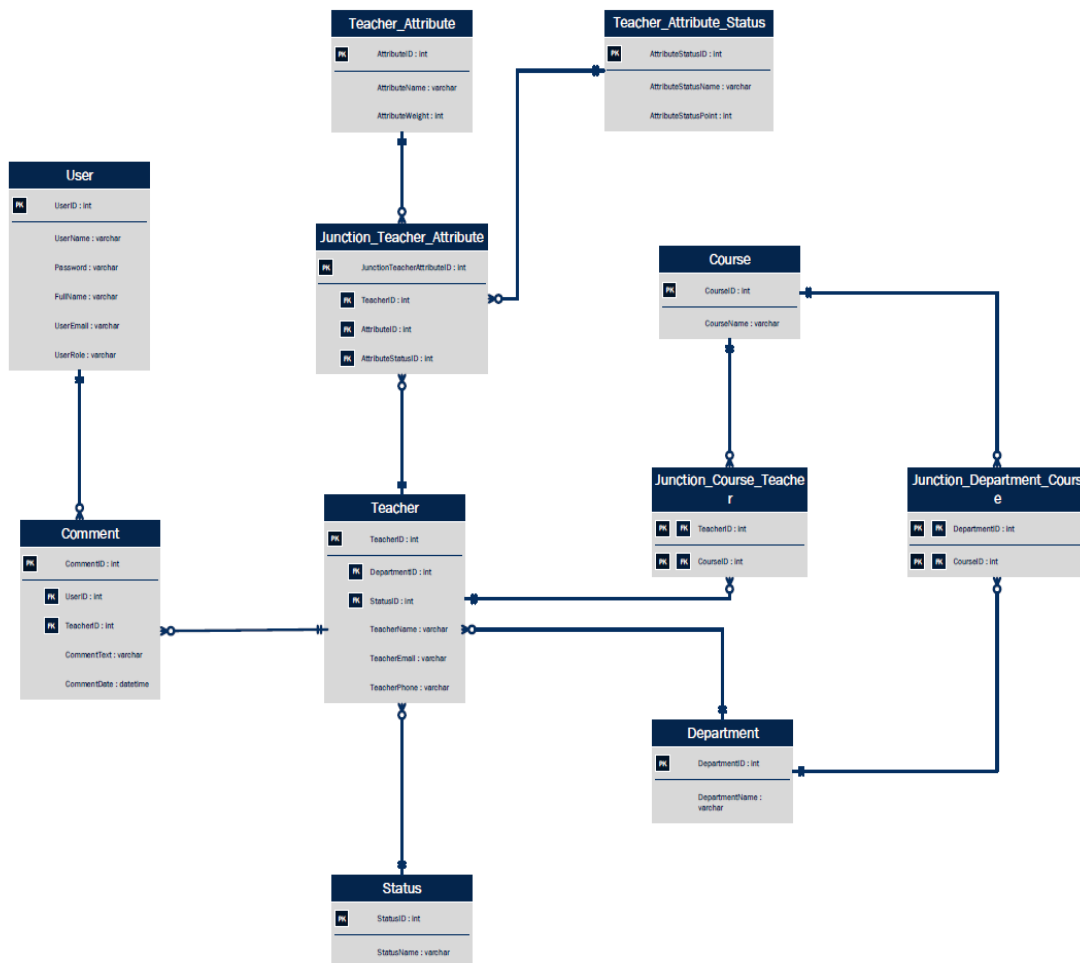
COURSE	
Primary Key	CourseID

JUNCTION_COURSE_TEACHER	
Primary Key	TeacherID, CourseID
Foreign Key	TeacherID, CourseID

DEPARTMENT	
Primary Key	DepartmentID

JUNCTION_DEPARTMENT_COURSE	
Primary Key	DepartmentID, CourseID
Foreign Key	DepartmentID, CourseID

Entity Relationship (ER) diagram with the multiplicity:



Relational Model:

```
CREATE TABLE `user`  
(  
  `UserID` int(11) NOT NULL AUTO_INCREMENT,  
  `UserName` varchar(250) NOT NULL,  
  `Password` varchar(250) NOT NULL,  
  `FullName` varchar(250) NOT NULL,  
  `Email` varchar(250) NOT NULL,  
  `Role` varchar(250) NOT NULL,  
  PRIMARY KEY (`UserID`)  
)
```

```
CREATE TABLE `course`  
(  
  `CourseID` int(11) NOT NULL AUTO_INCREMENT,  
  `CourseName` varchar(250) NOT NULL,  
  PRIMARY KEY (`CourseID`),  
  UNIQUE KEY `CourseName` (`CourseName`)  
)
```

```
CREATE TABLE `department`  
(  
  `DepartmentID` int(11) NOT NULL AUTO_INCREMENT,  
  `DepartmentName` varchar(250) NOT NULL,  
  PRIMARY KEY (`DepartmentID`),  
  UNIQUE KEY `DepartmentName` (`DepartmentName`)  
)
```

```

CREATE TABLE `status`
(
  `StatusID` int(11) NOT NULL AUTO_INCREMENT,
  `StatusName` varchar(250) NOT NULL,
  PRIMARY KEY (`StatusID`),
  UNIQUE KEY `StatusName` (`StatusName`)
)

```

```

CREATE TABLE `teacher`
(
  `TeacherID` int(11) NOT NULL AUTO_INCREMENT,
  `TeacherName` varchar(250) NOT NULL,
  `TeacherEmail` varchar(250) NOT NULL,
  `TeacherPhone` varchar(250) NOT NULL,
  `DepartmentID` int(11) NOT NULL,
  `StatusID` int(11) NOT NULL,
  PRIMARY KEY (`TeacherID`),
  KEY `DepartmentID` (`DepartmentID`),
  KEY `StatusID` (`StatusID`),
  CONSTRAINT `teacher_ibfk_1` FOREIGN KEY (`DepartmentID`) REFERENCES
`department` (`DepartmentID`),
  CONSTRAINT `teacher_ibfk_2` FOREIGN KEY (`StatusID`) REFERENCES `status`
(`StatusID`)
)

```

```

CREATE TABLE `junction_course_teacher`
(
  `TeacherID` int(11) NOT NULL,
  `CourseID` int(11) NOT NULL,

```

```

PRIMARY KEY (`TeacherID`, `CourseID`),
KEY `CourseID` (`CourseID`),
CONSTRAINT `junction_course_teacher_ibfk_1` FOREIGN KEY (`CourseID`) REFERENCES
`course` (`CourseID`),
CONSTRAINT `junction_course_teacher_ibfk_2` FOREIGN KEY (`TeacherID`)
REFERENCES `teacher` (`TeacherID`)
)

```

```

CREATE TABLE `junction_department_course`
(
`DepartmentID` int(11) NOT NULL,
`CourseID` int(11) NOT NULL,
PRIMARY KEY (`DepartmentID`, `CourseID`),
KEY `CourseID` (`CourseID`),
CONSTRAINT `junction_department_course_ibfk_1` FOREIGN KEY (`DepartmentID`)
REFERENCES `department` (`DepartmentID`),
CONSTRAINT `junction_department_course_ibfk_2` FOREIGN KEY (`CourseID`)
REFERENCES `course` (`CourseID`)
)

```

```

CREATE TABLE `teacher_attribute`
(
`AttributeID` int(11) NOT NULL AUTO_INCREMENT,
`AttributeName` varchar(250) NOT NULL,
`AttributeWeight` int(11) NOT NULL,
PRIMARY KEY (`AttributeID`)
)

CREATE TABLE `junction_teacher_attribute` (
`JunctionTeacherAttributeID` int(11) NOT NULL AUTO_INCREMENT,
`TeacherID` int(11) NOT NULL,

```

```

`AttributeID` int(11) NOT NULL,
`AttributePoint` int(11) NOT NULL,
PRIMARY KEY (`JunctionTeacherAttributeID`),
KEY `AttributeID` (`AttributeID`),
KEY `TeacherID` (`TeacherID`,`AttributeID`),
CONSTRAINT `junction_teacher_attribute_ibfk_1` FOREIGN KEY (`TeacherID`)
REFERENCES `teacher` (`TeacherID`),
CONSTRAINT `junction_teacher_attribute_ibfk_2` FOREIGN KEY (`AttributeID`)
REFERENCES `teacher_attribute` (`AttributeID`)
)

CREATE TABLE `teacher_attribute_status`
(
`AttributeID` int(11) NOT NULL,
`AttributeStatus` varchar(250) NOT NULL,
`AttributePoint` int(11) NOT NULL,
PRIMARY KEY (`AttributeID`,`AttributeStatus`),
CONSTRAINT `teacher_attribute_status_ibfk_1` FOREIGN KEY (`AttributeID`)
REFERENCES `teacher_attribute` (`AttributeID`)
)

CREATE TABLE `comment`
(
`CommentID` int(11) NOT NULL AUTO_INCREMENT,
`UserID` int(11) NOT NULL,
`TeacherID` int(11) NOT NULL,
`CommentText` text NOT NULL,
`CommentDate` datetime NOT NULL,
PRIMARY KEY (`CommentID`),
KEY `UserID` (`UserID`),

```


KEY **TeacherID** (TeacherID),

CONSTRAINT `comment_ibfk_1` FOREIGN KEY (UserID) REFERENCES `user` (UserID), CONSTRAINT `comment_ibfk_2` FOREIGN KEY (TeacherID) REFERENCES `teacher` (TeacherID))

Dummy data for justification:

DEPARTMENT

Department ID	Department Name
1	Computer Science and Engineering
2	Electrical Engineering
3	Civil Engineering

COURSE

Course ID	Course Name
1	Information System Design and Software Engineering
2	Microcontroller Based System Design
3	Operating System
4	Digital System Design
5	Database Lab
6	Assembly Language Programming Lab
7	Software Development - V
8	Mathematics I

JUNCTION_DEPARTMENT_COURSE

Department ID	Course ID
1	1
1	2
1	8
2	8
3	8

STATUS

Status ID	Status Name
1	Guest Teacher
2	Lecturer
3	Assistant Professor
4	Professor

TEACHER

Teacher ID	Teacher Name	Teacher Email	Teacher Phone	Department ID	Status ID
1	Mir Tafseer Nayeem	tafseer.nayeem@gmail.com	01912584949	1	3
2	Shoeb Shahriar	shahriar2529@gmail.com	01917256784	1	3
3	Moinul Hoque	moincse@yahoo.com	01817579779	1	3
4	Fayza Amreen	amreen.aust32@gmail.com		1	2

JUNCTION_TEACHER_COURSE

Teacher ID	Course ID
1	1
1	5
1	6
2	4
2	7
3	3
4	1

TEACHER_ATTRIBUTE

Attribute ID	Attribute Name	Attribute Weight
1	Professionalism	5
2	Patience	3
3	Integrity	1

Note: In this predefined table, “Attribute Weight” indicates the overall significance of an attribute which will be later used to calculate a teacher’s *Overall Score*.

Like, for example, Overall Score = Attribute Weight * Average Value [*of a particular attribute*]

“Average Value” will be queried from *JUNCTION_TEACHER_ATTRIBUTE* table.

TEACHER_ATTRIBUTE_STATUS

Attribute Status ID	Attribute Status Name	Attribute Status Point
1	Little	1
2	Moderate	2
3	Excellent	3

JUNCTION_TEACHER_ATTRIBUTE

J_Teacher_Attribute_ID	Teacher ID	Attribute ID	Attribute Status ID
1	1	1	3
2	1	2	2
3	1	3	1
4	1	1	1
5	1	2	2

Note : When reviewing, the data for each attribute for each teacher will be inserted into this table. In like manner, when visiting a particular teacher's profile, queries to find the total numeric values for each attribute will be performed in order to find an average value.

USER

User ID	Username	Password	Full Name	Email	Role
1	sanzamsyed	ENCODED	Syed Sanzam	sanzamsyed71@gmail.com	Student
2	nusratanika	ENCODED	Nusrat Anika	nusratanika04@gmail.com	Student
3	saifulislam	ENCODED	Saiful Islam	saifulislam@gmail.com	Student
4	randomvisitor	ENCODED	The Visitor	thevee@gmail.com	Visitor

COMMENT

Comment ID	User ID	Teacher ID	Comment Text	Comment Date
1	1	1	Lorem Ipsum	2019-03-17 18:29:57
2	1	2	Lorem Ipsum	2019-03-21 20:34:45
3	2	1	Lorem Ipsum	2019-03-21 11:54:45

Conclusion:

Building a strong connection between different kinds of data is pivotal before developing any kind of system application. As it is a fundamental part of the system, relationship accuracy should be at its best. The goal is always to make a software that is sustainable and is able to serve a purpose.