E-LEARNING MANAGEMENT SYSTEM DATABASE DESIGN DOCUMENT

Project Group 14

Table of Contents

Section 1: Project Description

Background:

As the Covid pandemic goes on, the word "remote" has become a lot more familiar to us all. Particularly, the need for remote learning has stimulated a steady market growth with more universities or third-party individuals investing in online learning management systems. For students or teachers, remote learning/teaching allows users to engage in materials we are interested in anywhere, anytime, and in any form, we'd prefer video, audio, documents etc. at a relatively lower cost, providing unrivaled flexibility and convenience. However, with the rapid increase in e-learning users as well as the educational content, database management skills and technologies are required to eradicate the disarray that may be caused by traditional flat-files methods.

Mission Statement:

With the rapidly increasing rate of the demand for e-learning, the number of online courses is increasing. This project's aim is to make the management of education system flexible with easy accessibility, effectiveness, and reasonable cost. In this project, we'll be storing information of user's, the course providers, course, etc., manage and provide better services.

Objectives:

- Database of courses that can be accessed by the user under one place, here users will be able to enroll for one or more course and complete certifications.
- Providing data to determine the popularity for the course depending upon how many users have taken that course.
- Providing data insights from the dataset that will help the platform owners to make an informed decision based on analyzed data.

Section 2: Business Rules

Business rules plays an integral role to the layout of the database and will show how tables are interact with each other. There are the main business rules for our project:

- **The USER ACCOUNT can be either an ADMIN, STUDENT, or INSTRUCTOR, but can not be more than one at the same time.**
  A supertype/subtype relationship is included with a disjoint rule.

- **A new user must fill in key information upon registering especially the account type.**
  A student account user will be taking courses and quizzes, a instructor account user will be posing and hosting courses, while admin deals with potential issue from the two type of user may encounter.

- **INSTRUCTOR Teaches COURSE, and COURSE is taught by INSTRUCTOR.**
  An instructor must have at least one course to teach to be qualified as an instructor, while a course must have one instructor.

- **COURSE Has CONTENT, and CONTENT is related to COURSE.**
  A course must have some content and empty courses are not allowed, and a content must and can only belongs to one course.

- **LEARNER Makes a REGISTRATION, and REGISTRATION Happens for a COURSE.**
  A learner must register for something to be qualified as a learner. A learner may register for one or more courses.

- **REGISTRATION Generates INVOICE, and INVOICE is generated for a REGISTRATION.**
  An invoice in generated when a registration happens, recording the necessary of the registration. An invoice belongs to only one single registration.

- **LEARNER Gives COURSE RATING, and COURSE Has COURSE RATING.**
  Learners may give course rating to evaluate the experience they had. While giving a rating is optional for a learner, a course rating must belong to only one learner and course pairing.

- **LEARNER Completes CERTIFICATE, and there is CERTIFICATE For a COURSE.**
  A learner receives certificate when a course is finished. A course may issue many certificates and a learner may receive multiple certificates from different courses.

- **CONTENT Has QUIZ, and there is a QUIZ for a CONTENT.**
  Quiz is generated to evaluate learner's mastery of a certain course content. Quizzes are deemed necessary for us and should be included in contents.

- **LEARNER Takes a QUIZ TAKEN, and QUIZ TAKEN Is for QUIZ.**
  As quiz is mandatory for course content and learner is required to be registered in a course or more to be qualified as a learner, therefore a learner must take one or more quiz during a course.

Section 3: Table Description

With the real-world learning management systems in mind, we design the following entities/tables for our project. Their name, datatype and other information are as follows:

3.1 User:

Each user will create a USER ACCOUNT, and the USER ACCOUNT has three subtypes as defined in our business rules. All key information will be stored upon registration in the corresponding tables. Their details are as follows.

3.1.1 USER_ACCOUNT:

Columns	Description	Data Type	Key
AccountId	Unique ID to identify the user	int	PK
AccountUsername	Username created upon registration	varchar(100)	
AccountPassword	Unencrypted password, will be varbinary datatype after encryption.	varchar(100)	
FirstName	First name of the user	varchar(100)	
LastName	Last name of the user	varchar(100)	
EmailId	Email address of the user	varchar(100)	
ContactNo	Contact number of the user, mostly phone number	bigint	
CreatedOn	Date and time the account is created	datetime	
AccountType	Type of the account, either ADMIN, LEARNER, or INSTRUCTOR	varchar(20)	

3.1.2 ADMIN:

Columns	Description	Data Type	Key
AdminId	Unique identifier for ADMIN referencing to AccountId in USER ACCOUNT	int	PFK
AdminRole	The field that an admin works in, maybe course admin, quiz admin, registration admin .etc.	varchar(100)	

3.1.3 LEARNER:

Columns	Description	Data Type	Key
LearnerId	Unique identifier for LEARNER referencing to AccountId in USER ACCOUNT	int	PFK
Occupation	The job of a learner, usually students but other occupations are accepted	varchar(100)	

3.1.4: INSTRUCTOR:

Columns	Description	Data Type	Key
InstructorId	Unique identifier for INSTRUCTOR referencing to AccountId in USER ACCOUNT	int	PFK
University	The university an instructor is from	varchar(100)	
FieldOfStudy	The field of expertise of an instructor	varchar(100)	

3.2 Course

The following section contains a series of entities/tables involving the information and content of the course created by instructors.

3.2.1 COURSE:

Columns	Description	Data Type	Key
CourseId	Unique ID to identify each course	varchar(50)	PK
InstructorId	Foreign key referencing to InstructorId in INSTRUCTOR	varchar(100)	FK
CourseName	Name of the course	int	
CourseFees	The cost of attending the course	int	

3.2.2 CONTENT:

Columns	Description	Data Type	Key
ContentId	Unique ID to identify each content of a course	varchar(50)	PK
CourseId	Foreign key referencing to CourseId in COURSE	varchar(50)	FK
Description	Field containing brief information of content	varchar(300)	

3.2.3 QUIZ:

Columns	Description	Data Type	Key
QuizId	Unique ID to identify each quiz of a content	int	PK
ContentId	Foreign key referencing to ContentId in CONTENT	varchar(50)	FK
QuizTime	The time required to take the quiz	int	
Attempts	Number of attempt a learner can have	int	
TotalMarks	Total marks of the quiz	int	
MinimumPassingMarks	Minimum marks a learner has to acquire to pass	int	

3.3 Actions

This section contains the associative entities that record the action and information of the interactions between entities.

3.3.1 REGISTRATION:

REGISTRATION is an associative entity connecting COURSE and LEARNER that record data generated for/when a student register for a course.

Columns	Description	Data Type	Key
RegistrationId	Surrogate key to identify each registration	int	PK
CourseId	Foreign key referencing to CourseId in COURSE	varchar(50)	FK
LearnerId	Foreign key referencing to LearnerId in LEARNER	int	FK
RegistrationDate	The date that registration happened	date	

3.3.2 QUIZ TAKEN:

QUIZ TAKEN is an associative entity connecting QUIZ and LEARNER that record data generated when a student takes a quiz.

Columns	Description	Data Type	Key
QuizTakenId	Surrogate key to identify each quiz take action	int	PK
QuizId	Foreign key referencing to QuizId in QUIZ	int	FK
LearnerId	Foreign key referencing to LearnerId in LEARNER	int	FK
QuizTakeDate	The date of a learner takes a quiz	date	
MarksObtained	The marks a learner get from a attempt	int	

3.4 Record:

This part contains entities generated to record and display key information for a user throughout his/her entire experience on our learning management system.

3.4.1 CERTIFICATE:

Columns	Description	Data Type	Key
CertificateId	Unique ID to identify each certificate for course completion	int	PK
CourseId	Foreign key referencing to CourseId in COURSE	varchar(50)	FK
LearnerId	Foreign key referencing to LearnerId in LEARNER	int	FK
CompletedOn	Date and time at which a learner completes a course	datetime	

3.4.2 COURSE RATING:

Columns	Description	Data Type	Key
LearnerId	Foreign key referencing to LearnerId in LEARNER	int	PFK
CourseId	Foreign key referencing to CourseId in COURSE	varchar(50)	PFK
RateDate	The date on which a rating is given	date	
Rating	The actual score of a rating	int	

3.4.3 INVOICE:

Columns	Description	Data Type	Key
InvoiceId	Unique ID to identify each invoice for registration	int	PK
RegistrationId	Foreign key referencing to RegistrationId in REGISTRATION	int	FK
PaymentMode	The payment method of a transaction	varchar(10)	
Date	The date on which the transaction happened	date	
Status	The status of the transaction	char(9)	

Section 4: Entity Relationship Diagram

An ER diagram is created based on the aforementioned business rules and tables. It serves as a guideline throughout our design of the database project. Our entity relationship diagram is attached below:

E Learning Management System