



Date: 12/11/2017

CS5200 - SEC02
Database Management System
Project 16

Trainly.io

By

Sanket Mathur
Eric Ropiak
Prasad Tajane
Wei Du

Abstract

As a group, we set out to build a product which would allow students to complete training courses online. We built Trainly.io to connect students with the online resources that they need to do so.

More specifically, students had to be able to search for and enroll in classes. Once enrolled, they needed to be able to complete course materials in a specified order before a course was marked as complete. Users had to be able to create accounts and securely login. Additionally, there had to be a subset of users, faculty, who could create courses and respond to any questions regarding that course. Another subset of users, admins, needed to have the ability to approve faculty members or fellow admins.

There were primarily two large components that needed to be built out for the product to function. Firstly, we needed a database which could store large amounts of information relevant to the product. Secondly, we needed an interface that people could use to consume this data.

We chose to use a MariaDB database to hold all of the relevant data. Before writing any code, careful thought was put into the entities that would be needed to represent the system. ER and relational diagrams were constructed to display how these entities relate to each other in an organized manner. Once we agreed on the underlying representation for the system, DDL was used to create a working instance of the database, which was then filled with sample data. To ensure that our design was sound and that the database could perform the required tasks, we wrote queries to test actions like user registration, course enrollment and admin verification. Only after all of these pieces were in place did we move on to building a working user interface.

The interface that we chose to use was a Flask web application. There was a small amount of configuration that was needed to connect the application to the database, but once it was set up, we just had to pass our queries to the database and create an elegant display for the user to view the data. PyMySQL, a python library, was used to parameterize queries with user input to prevent SQL injection attacks. Bootstrap styling was used to create the display. Hashlib was used to encrypt user passwords (plus a randomly generated 4 character salt) using SHA256 before they were stored in the database.

Ultimately, we were able to produce a secured, polished system that allows users to sign up, enroll in courses, complete course materials, ask questions, and much more. We built Trainly.io, the future of the training industry!

Textual Description

Each user has a first name, a last name, a unique email address, a password, a profile picture, one or more contact phone numbers, and a contact address consisting of a street, a city, a postal code and a country. Thus, all the common attributes related to every type of user is being added into User relation whereas specific attributes are added in the specialised category Admin, Faculty.

In order to become a faculty member, a user must be approved by an administrator after registering. The date, time and the identity of the approving administrator, must be kept track of when a faculty member is approved. Similarly, to become an administrator a user must be given the role by another administrator, whose identity must be recorded, along with the time and date of the granting.

Every course has a unique course id, a name, a description, a cost, a creation date and at least one creator, who is a faculty member. Additionally, each course belongs to at least one topic (its primary topic), but may also belong to any number of secondary topics. A course consists of at least one course material, but can have any number of materials, which must be completed in a specified order.

Each course material can only be used in a single course and is identified by its order in its courses' sequence of materials. Each material has a name and must be either a downloadable file, a link, a post or a quiz. Downloadable files have a path, a size, and a type. Links have a URL and a flag indicating whether or not the URL is a video. A post contains a block of textual content, which has possibly been marked up. A quiz is comprised of a minimum passing score, and a list of quiz questions. Each quiz question has a number (unique to its quiz), text for the question, and a set of four multiple choice question answers. Each answer is represented by a letter (unique to the quiz question) and has text, feedback upon submission and an indication of whether or not it is the correct answer.

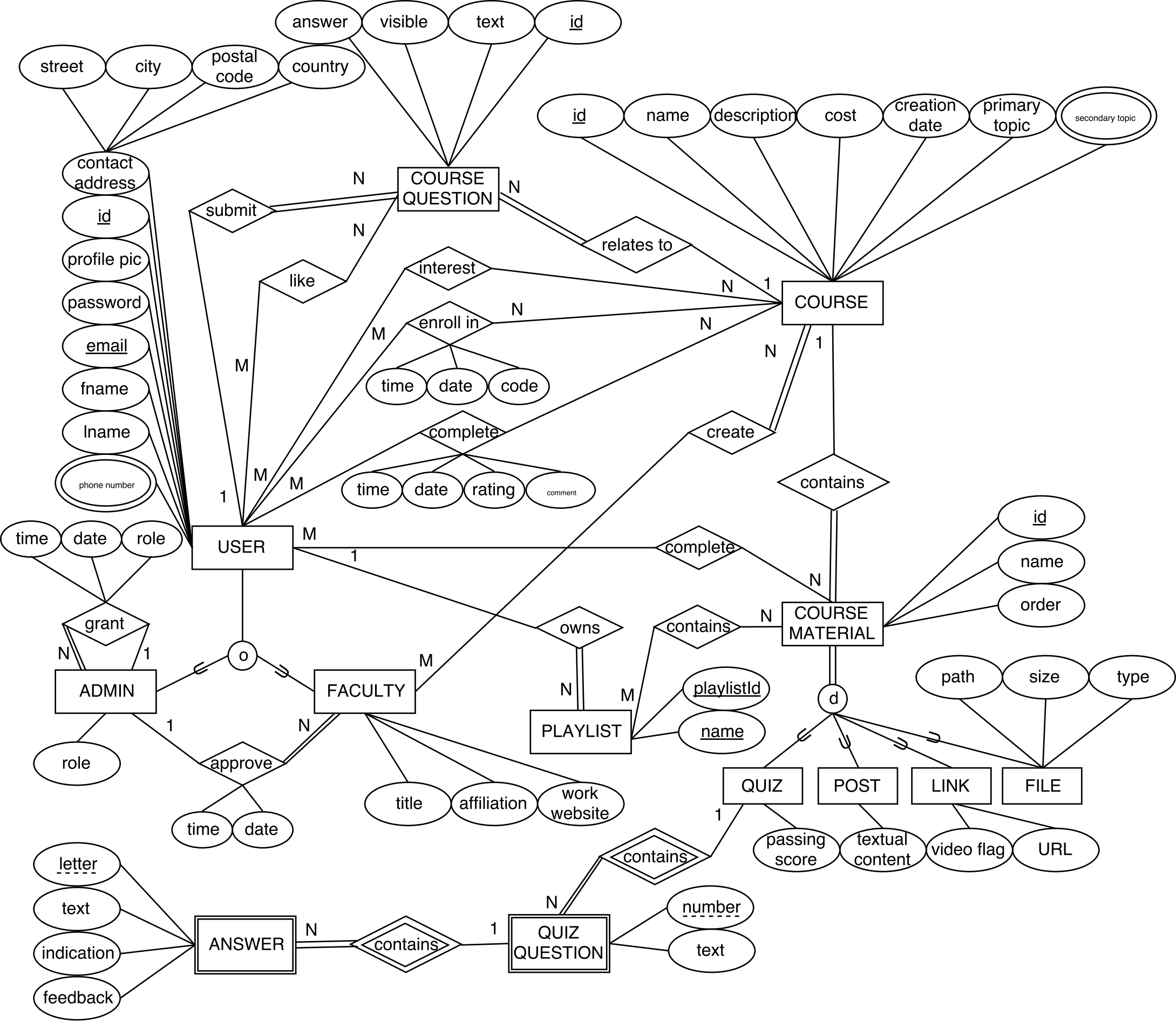
Each student has a list of courses that they are interested in. Students can also enroll in any number of courses. When they enroll in a course, the date and time of payment and the payment confirmation code must be recorded.

To complete a course, a student must complete all associated course materials in the specified order. When a course material is completed (i.e. a video is viewed, a file is downloaded, a quiz is completed, or a post is viewed), the time and date of completion must be recorded.

When a student completes a course, the date and time of completion are recorded. Additionally, if the student provides a rating (1-5 stars) or comments about the course, those also must be saved.

Students may also submit questions about courses they have enrolled in. Each question must relate to a course, and may additionally relate to one or more specific course materials. Each question is made up of a question id (which is unique within its course), the question text, along with a flag indicating whether or not the question has been made visible to all students by a faculty member. Additionally, the question may contain a textual answer, if one has been given by a faculty member. Students may like questions that they find to be useful.

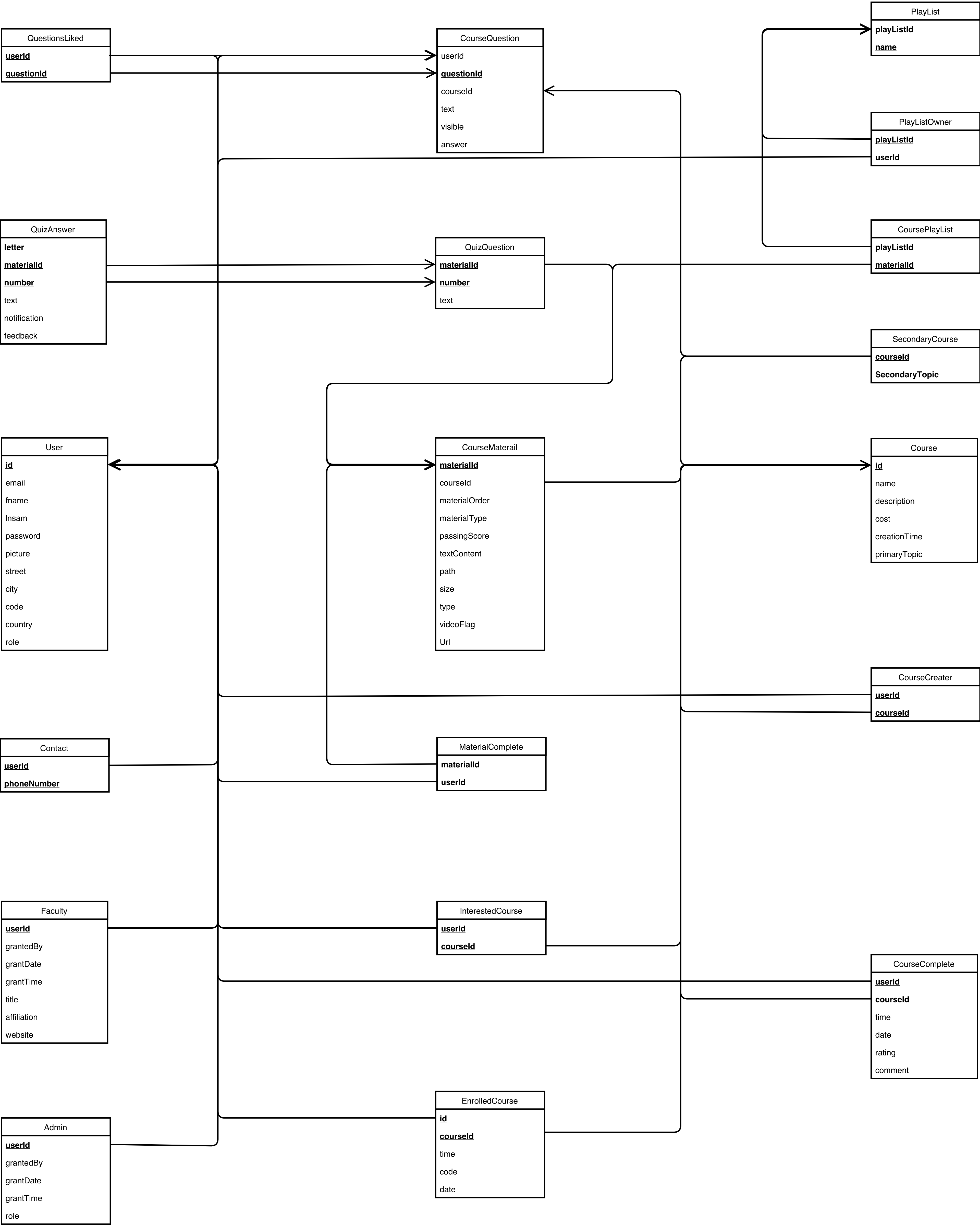
A student can also create a playlist of course materials. Each playlist has a unique id, a unique name and contains an ordered list of course materials. Students can create, delete and edit playlists.



As each user has some common attributes such as name, email and contact, those attributes are merged into one generalized entity 'User' whereas specific attributes are added in the specialized entity Admin, Faculty.

All attributes of course are added into the course entity and the materials, questions related to that course are saved into another entity course question, course material. Course material is further categorized into four specialized formats – quiz, post, link and file. Quiz has two different sections questions and answers, they are represented with two different weak entities.

Student can create playlist out of the course material and thus it is delineated in the playlist entity.



Physical Design :

Tables have been created for each of the entities and many to many relationships. Apart from that as system is not doing any heavy calculations and the data is small there is no need to work on query optimization and thus, we do not have any index in our system so far. However, depending on the profiling and DBMS query explain we can add index or work on physical and design tuning in future if problem arises.

The system has 2 before insert triggers on table faculty and administrator. This is done in order satisfy the constraint that the fields access granted date and time is not left blank and updated with proper time stamp. So whenever a user is given faculty or admin access, an entry is inserted in the appropriate table and the trigger automatically puts the timestamp in those records.

Screenshots of the running system:

1. Landing page of the system. We have tried to make it informative. It has 3 users shown under the Premade users block which can be used for testing. All the 3 users have different roles.

Overall in the database there are 30 users, segregated as:

10001-10010 : Admin

10011-10020 : Faculty

10021-10030 : Student

Using the Sign In button in the top right corner, a user can login into the system.

Trainly.io

Sign In

Welcome to Trainly.io!

Welcome to Trainly.io, the CS5200 semester project by Eric, Prasad, Sanket and Wei. The website provides an interface for interacting with the database that we have created and populated.

Premade users

Here are some premade users that you can use to test functionality. If you click on a user's email, you will be signed in as that user. Alternatively, you can use their credentials [here](#).

Email	Role	Password
Georgi.10001@somemail.com	Admin	somepass
Eberhardt.10013@somemail.com	Faculty	somepass
Ramzi.10021@somemail.com	Student	somepass

Quick Links

Tasks	Reports
Register New User	Materials Remaining Per Course (Student)
Authenticate Faculty/Admin (Admin)	\$\$ Made by Each Course Creator (Admin)
Categorized List of Student's Courses (Student)	Sort Courses by Topic (Student)
Enroll in a Course (Student)	Sort Courses by Completion Percent (Student)
See Ordered Course Materials (Student)	See Courses With Unanswered Questions (Faculty)
Mark Material Complete (Student)	
Certificate of Completion (Student)	
Account History (Student)	

2. Login page - a registered user can login into the system using their email as the username and password. Access to different functionality will be given to user based upon their roles.

Use the Sign up here, link (at the bottom of the sign in block) to register for new user.

Welcome Back!

Email:

Password:

Sign In

Don't have an account? Sign up [here](#).

3. Sign-up page: A new user can register himself into the system. Initially every new user is a student. Admin can change the roles of any user based on the request.

Welcome To Trainly.io!

First Name:

Last Name:

Phone Number:

Street:

City:

Postal Code:

Country:

Profile Picture:

Choose File

No file chosen

Email:

Password:

Confirm Password:

Sign up

Already have an account? Sign in [here](#).

ADMIN:

- Once login as admin into the system, an admin can list all the courses into the system under the Find Courses link at the top.

[Trainly.io](#) [My Courses](#) [Find Courses](#) [Admin](#) [Account Info](#) [Sign Out](#)

Courses

Search by: [Topic](#) | [Completion Percentage](#) | [Keyword](#)

Name	Description	Topic(s)	Course Rating	
Databases Programming	This overview course focuses on the uses of relational and object-oriented databases for storing and managing information. Topics covered include computer database terminology and the evolution of the modern database. Database management systems (DBMS) such as Oracle, MySQL, Microsoft SQL Server, and Microsoft Access are introduced along with query languages. Hands-on instruction includes the creation of simple databases, inputting data, and developing basic queries.	Database, Database Design, Programming in SQL, Relational Database	4.5000	☆
Developing Web Databases	Internet retailers collect an incredible amount of data in order to complete a sale, including customer names, addresses, and credit card numbers. The data is inputted by the buyer and stored in a relational database for retrieval by the seller.	Database	2.5000	☆
Natural Language Processing	Introduces the computational modeling of human language; the ongoing effort to create computer programs that can communicate with people in natural language; and current applications of the natural language field, such as automated document classification, intelligent query processing, and information extraction. Topics include computational models of grammar and automatic parsing, statistical language models and the analysis of large text corpora, natural language semantics and programs that understand language, models of discourse structure, and language use by intelligent agents. Course work includes formal and mathematical analysis of language models and implementation of working programs that analyze and interpret natural language text. Knowledge of statistics is helpful.	Computational Modeling of Human Language	--	☆
Computer Graphics	Charts a path through every major aspect of computer graphics with varying degrees of emphasis. Discusses hardware issues: size and speed; lines, polygons, and regions; modeling, or objects and their relations; viewing, or what can be seen (visibility and perspective); rendering, or how it looks (properties of surfaces, light,	Programming in C	--	☆

- Admin can look at its account information under under the Account Info, found on the top right corner of the screen, beside the Sign out button.

[Trainly.io](#) [My Courses](#) [Find Courses](#) [Admin](#) [Account Info](#) [Sign Out](#)

Account Info

Email: Georgi.10001@somemail.com
Name: Georgi Facello
Contact Number(s): 5250491525
Address: Street123, Boston, USA, 02115

Account History

No history found associated with your account.

6. Admin can get the list of all the users in the system and the roles associated with them. He/She can also change the role of one or more users.

Trainly.io

My CoursesFind CoursesAdmin

Account InfoSign Out

[View faculty earnings report here.](#)

Current Users

Email	First Name	Last Name	Role	Update	Delete
Yongqiao.10026@somemail.com	Yongqiao	Bertziss	Student	✓	✕
Tzvetan.10007@somemail.com	Tzvetan	Zielinski	Admin	✓	✕
Suzette.10024@somemail.com	Suzette	Pettey	Student	✓	✕
Sumant.10009@somemail.com	Sumant	Peac	Admin	✓	✕
Shah.10022@somemail.com	Shah	Famili	Student	✓	✕
Saniya.10008@somemail.com	Saniya	Kalloufi	Admin	✓	✕
Ramzi.10021@somemail.com	Ramzi	Erde	Student	✓	✕
Prasadram.10025@somemail.com	Prasadram	Heyers	Student	✓	✕
Patricio.10012@somemail.com	Patricio	Bridgland	Faculty	✓	✕
Parto.10003@somemail.com	Parto	Bamford	Admin	✓	✕
Otmar.10029@somemail.com	Otmar	Herbst	Student	✓	✕
Mayuko.10020@somemail.com	Mayuko	Warwick	Faculty	✓	✕
Mary.10011@somemail.com	Mary	Sluis	Faculty	✓	✕
Lillian.10019@somemail.com	Lillian	Haddadi	Faculty	✓	✕

7. Admin can change the role of any user by selecting the available roles and clicking on the check mark under the Update for that corresponding user. Admin can also delete a user by clicking on the X sign under the Delete column.

Trainly.io

My CoursesFind CoursesAdmin

Account InfoSign Out

[View faculty earnings report here.](#)

Current Users

Email	First Name	Last Name	Role	Update	Delete
Yongqiao.10026@somemail.com	Yongqiao	Bertziss	Admin Faculty ✓ Student	✓	✕
Tzvetan.10007@somemail.com	Tzvetan	Zielinski	Admin	✓	✕
Suzette.10024@somemail.com	Suzette	Pettey	Student	✓	✕
Sumant.10009@somemail.com	Sumant	Peac	Admin	✓	✕
Shah.10022@somemail.com	Shah	Famili	Student	✓	✕
Saniya.10008@somemail.com	Saniya	Kalloufi	Admin	✓	✕
Ramzi.10021@somemail.com	Ramzi	Erde	Student	✓	✕
Prasadram.10025@somemail.com	Prasadram	Heyers	Student	✓	✕
Patricio.10012@somemail.com	Patricio	Bridgland	Faculty	✓	✕

8. Admin can also look at the earnings report of a faculty by clicking on the link “View faculty earning report here”

[Trainly.io](#) [My Courses](#) [Find Courses](#) [Admin](#)

Earnings Report

Shows an organized list of how much money has been made in courses created by faculty members.

Creator Email	Earnings
Eberhardt.10013@somemail.com	\$8000
Anneke.10006@somemail.com	\$5000
Saniya.10008@somemail.com	\$1000
Tzvetan.10007@somemail.com	\$1000

FACULTY:

9. Faculty can get the list of courses under the Find Courses link.

Trainly.io

My Courses

Find Courses

Faculty

Account Info

Sign Out

Courses

Search by: Topic | Completion Percentage | Keyword

Name	Description	Topic(s)	Course Rating	
Databases Programming	This overview course focuses on the uses of relational and object-oriented databases for storing and managing information. Topics covered include computer database terminology and the evolution of the modern database. Database management systems (DBMS) such as Oracle, MySQL, Microsoft SQL Server, and Microsoft Access are introduced along with query languages. Hands-on instruction includes the creation of simple databases, inputting data, and developing basic queries.	Database, Database Design, Programming in SQL, Relational Database	4.5000	☆
Developing Web Databases	Internet retailers collect an incredible amount of data in order to complete a sale, including customer names, addresses, and credit card numbers. The data is inputted by the buyer and stored in a relational database for retrieval by the seller.	Database	2.5000	☆
Natural Language Processing	Introduces the computational modeling of human language; the ongoing effort to create computer programs that can communicate with people in natural language; and current applications of the natural language field, such as automated document classification, intelligent query processing, and information extraction. Topics include computational models of grammar and automatic parsing, statistical language models and the analysis of large text corpora, natural language semantics and programs that understand language, models of discourse structure, and language use by intelligent agents. Course work includes formal and mathematical analysis of language models and implementation of working programs that analyze and interpret natural language text. Knowledge of statistics is helpful.	Computational Modeling of Human Language	--	☆
Computer Graphics	Charts a path through every major aspect of computer graphics with varying degrees of emphasis. Discusses hardware issues: size and speed; lines, polygons, and regions; modeling, or objects and their relations; viewing, or what can be seen (visibility and perspective); rendering, or how it looks (properties of surfaces, light, and color); transformations, or moving, placing, distorting, and animating and	Programming in C	--	☆

10. Under the Faculty link, a faculty can get the list of courses they need to perform some action under “Courses Needing Attention”. Also, they can see the list of other courses, together with the number of students enrolled and completed the course.

Trainly.io

My Courses

Find Courses

Faculty

Courses Needing Attention

No courses that require your attention.

All Courses

Name	Enrolled	Completed
Databases Programming	10	4
Natural Language Processing	1	0
Computer Graphics	3	0
Compilers	0	0
Developing Web Databases	2	2

11. Faculty can answer the questions posted by the students for a course.

[Trainly.io](#) [My Courses](#) [Find Courses](#) [Faculty](#) [Account Info](#) [Sign Out](#)

Developing Web Databases

Internet retailers collect an incredible amount of data in order to complete a sale, including customer names, addresses, and credit card numbers. The data is inputted by the buyer and stored in a relational database for retrieval by the seller.

Questions

What tools will be used in this course? [Like this question](#)

Related materials:

We will be using MySql for database

[Answer Question](#)

[Make this question visible](#)

12. Faculty can also get the details about their account under the link Account Info

[Trainly.io](#) [My Courses](#) [Find Courses](#) [Faculty](#) [Account Info](#) [Sign Out](#)

Account Info

Email: Eberhardt.10013@somemail.com
Name: Eberhardt Terkki
Contact Number(s): 9480591821
Address: Street123, Seattle, USA, 202012

Account History

No history found associated with your account.

STUDENT:

13. Students can get the list of courses they are associated with under the link My Courses. Current Courses will list all the courses they are enrolled in and currently pursuing. Under the Course Completed, they can see the list of courses they have completed so far. Student can even mark some of the courses they are interested to take, and can get the list under the block Interested Courses.

Trainly.io

My Courses

Find Courses

Account Info

Sign Out

Current Courses

Name	Description	Topic(s)	Materials Remaining	Course Rating
Developing Web Databases	Internet retailers collect an incredible amount of data in order to complete a sale, including customer names, addresses, and credit card numbers. The data is inputted by the buyer and stored in a relational database for retrieval by the seller.	Database	2	2.5000
Natural Language Processing	Introduces the computational modeling of human language; the ongoing effort to create computer programs that can communicate with people in natural language; and current applications of the natural language field, such as automated document classification, intelligent query processing, and information extraction. Topics include computational models of grammar and automatic parsing, statistical language models and the analysis of large text corpora, natural language semantics and programs that understand language, models of discourse structure, and language use by intelligent agents. Course work includes formal and mathematical analysis of language models and implementation of working programs that analyze and interpret natural language text. Knowledge of statistics is helpful.	Computational Modeling of Human Language	0	--
Web Development	Discusses Web development for sites that are dynamic, data driven, and interactive. Focuses on the software development issues of integrating multiple languages, assorted data technologies, and Web interaction. Considers ASP.NET, C#, HTTP, HTML, CSS, XML, XSLT, JavaScript, AJAX, RSS/Atom, SQL, and Web services. Requires each student to deploy individually designed Web experiments that illustrate the Web technologies and at least one major integrative Web site project. Students may work as a team with the permission of the instructor. Each student or team must also create extensive documentation of their goals, plans, design decisions, accomplishments, and user guidelines. All source files must be open and be automatically served by a sources server.	HTML and CSS Basics, Angular JS, Node JS	1	--
Game Programming	Introduces the different subsystems used to create a 3D game, including rendering, animation, collision, physics, audio, trigger systems, game logic, behavior trees, and simple artificial intelligence. Offers students an opportunity to learn the inner workings of game engines and how to use multiple libraries such as physics and graphics libraries to develop a game. Discusses graphics pipeline, scene graph	Gaming Design	1	--

Completed Courses

Name	Description	Topic(s)	Completion Date	My Rating
Databases Programming	This overview course focuses on the uses of relational and object-oriented databases for storing and managing information. Topics covered include computer database terminology and the evolution of the modern database. Database management systems (DBMS) such as Oracle, MySQL, Microsoft SQL Server, and Microsoft Access are introduced along with query languages. Hands-on instruction includes the creation of simple databases, inputting data, and developing basic queries.	Database, Database Design, Programming in SQL, Relational Database	2015-01-02	5
Developing Web Databases	Internet retailers collect an incredible amount of data in order to complete a sale, including customer names, addresses, and credit card numbers. The data is inputted by the buyer and stored in a relational database for retrieval by the seller.	Database	2015-01-02	2

Interested Courses

Name	Description	Topic(s)	Course Rating
Discrete Structures	Introduces the mathematical structures and methods that form the foundation of computer science. Studies structures such as sets, tuples, sequences, lists, trees, and graphs. Discusses functions, relations, ordering, and equivalence relations. Examines inductive and recursive definitions of structures and functions. Discusses principles of proof such as truth tables, inductive proof, and basic logic. Also covers the counting techniques and arguments needed to estimate the size of sets, the growth of functions, and the space-time complexity of algorithms.	Algorithms, Data Structures	--

14. Student can get the list of all the courses under the link Find Courses. Course names are clickable which will take the student to the detail description page for the course.

Trainly.io

My Courses





Find Courses

Account Info

Sign Out

Courses

Search by: [Topic](#) | [Completion Percentage](#) | [Keyword](#)

Name	Description	Topic(s)	Course Rating
Databases Programming	This overview course focuses on the uses of relational and object-oriented databases for storing and managing information. Topics covered include computer database terminology and the evolution of the modern database. Database management systems (DBMS) such as Oracle, MySQL, Microsoft SQL Server, and Microsoft Access are introduced along with query languages. Hands-on instruction includes the creation of simple databases, inputting data, and developing basic queries.	Database, Database Design, Programming in SQL, Relational Database	4.5000 
Developing Web Databases	Internet retailers collect an incredible amount of data in order to complete a sale, including customer names, addresses, and credit card numbers. The data is inputted by the buyer and stored in a relational database for retrieval by the seller.	Database	2.5000 
Natural Language Processing	Introduces the computational modeling of human language; the ongoing effort to create computer programs that can communicate with people in natural language; and current applications of the natural language field, such as automated document classification, intelligent query processing, and information extraction. Topics include computational models of grammar and automatic parsing, statistical language models and the analysis of large text corpora, natural language semantics and programs that understand language, models of discourse structure, and language use by intelligent agents. Course work includes formal and mathematical analysis of language models and implementation of working programs that analyze and interpret natural language text. Knowledge of statistics is helpful.	Computational Modeling of Human Language	-- 
Computer Graphics	Charts a path through every major aspect of computer graphics with varying degrees of emphasis. Discusses hardware issues: size and speed; lines, polygons, and regions; modeling, or objects and their relations; viewing, or what can be seen (visibility and perspective); rendering, or how it looks (properties of surfaces, light, and color); transformations, or moving, placing, distorting, and animating and interaction, or drawing, selecting, and transforming.	Programming in C	-- 

15. A detailed description page of a course, including the information whether the student has finished the course or not. A link to the certificate if the student has finished the course. What are the course materials associated with this course and a link to ask question if in doubt.

Trainly.io

My Courses

Find Courses

Account Info

Sign Out

Databases Programming

This overview course focuses on the uses of relational and object-oriented databases for storing and managing information. Topics covered include computer database terminology and the evolution of the modern database. Database management systems (DBMS) such as Oracle, MySQL, Microsoft SQL Server, and Microsoft Access are introduced along with query languages. Hands-on instruction includes the creation of simple databases, inputting data, and developing basic queries.

Topic(s): Database, Database Design, Programming in SQL, Relational Database Enrolled on: 2015-01-02

[Course Questions](#)

Course Complete

Congratulations! You completed this course on 2015-01-02!

[See your certification!](#)

You rated this course: 5

Course Materials

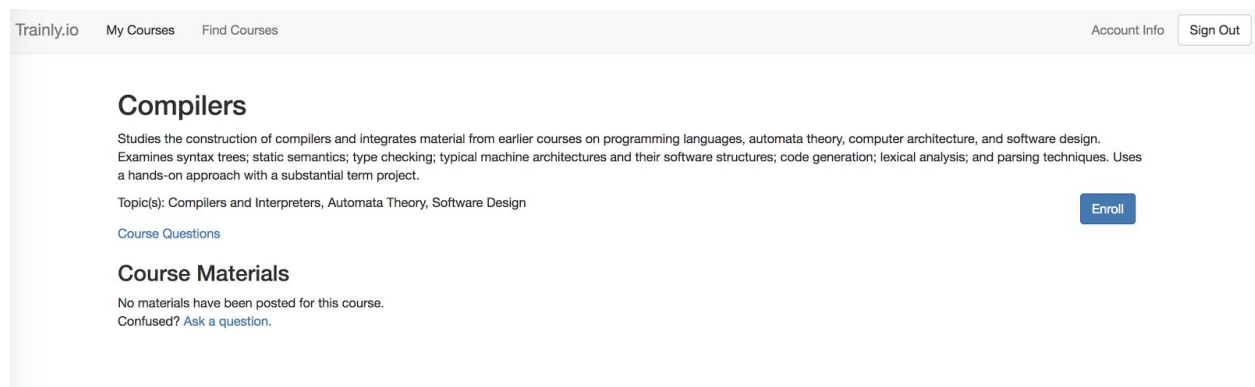
Order	Name	Material Type	View	Completion Date
1	Programming Link	Link	Visit	2015-01-02
2	Programming File	File (Type: java, Size: 1000)	/somepath/file	2015-01-02
4	Programming Quiz	Quiz	View	2015-01-02

Confused? [Ask a question.](#)

16. Certificate of course completed



17. A course detail page of a course which student has not completed and can enroll into it by clicking on the Enroll button.



18. Student can ask question related to a course which will be answered by the faculty members.

Trainly.io

My Courses

Find Courses

Account Info

Sign Out

Artificial Intelligence

Introduces the fundamental problems, theories, and algorithms of the artificial intelligence field. Includes heuristic search; knowledge representation using predicate calculus; automated deduction and its applications; planning; and machine learning. Additional topics include game playing; uncertain reasoning and expert systems; natural language processing; logic for common-sense reasoning; ontologies; and multiagent systems.

Ask your question:

Select any course materials this question applies to:

Artificial Intelligence File

Ask Question

19. Student can see their account information under the link Account Info. They can see their account history related to the course details about enrolled date, completion date, amount paid and transaction id.

Trainly.io

My Courses

Find Courses

Account Info

Sign Out

Account Info

Email: Ramzi.10021@somemail.com
Name: Ramzi Erde
Contact Number(s): 2084433135, 4439225298, 9540565281
Address: Street123, Seattle, USA, 065001

Account History

Course	Enrollment Date	Completion Date	Amount Paid	Confirmation Code
Databases Programming	2015-01-02	2015-01-02	500	TXN100001
Natural Language Processing	2015-01-02	None	500	TXN100030
Computer Graphics	2015-01-02	None	500	TXN100022
Developing Web Databases	2015-01-02	2015-01-02	500	TXN100020
Web Development	2015-01-02	None	500	TXN100029
Programming in C++	2015-01-02	None	500	TXN100011
Game Programming	2015-01-02	None	500	TXN100027
Artificial Intelligence	2015-01-02	None	500	TXN100025
GRAND TOTAL:			4000	

Project Retrospective:

The project start from the very beginning of any real time project by designing a database system from the description of a system. The database design must be good to make a perfect working system. This project has given us a proper understanding and knowledge of database design and structure together with development of ER model and physical design of the system.

There is no point which we can mark as easiest or hardest for the project but we have put majority of our effort in building a good and detailed ER diagram and to make sure that the physical design is in 3NF form.

Additionally, working in team helped a lot in completing this project on time and in the best possible way. Every large scale project has a team associated with it which and the contribution of every team member led to its success. In this project every team member has contributed their knowledge of database systems to make it a better product.

Conclusion:

We have built a training system to train a student in the courses available in Trainly.io. Student can register himself, enroll in a course, ask a question about a course or like other students questions and take quizzes to check its progress. A faculty manages courses, decide the course content and answer the students' questions regarding a course. To manage all of those things there will be some administrators who are responsible for managing the details of the users.

Overall, we have successfully developed an interactive and polished training system convenient for students to learn and interact with faculty members in the best possible way. Since almost all of this has been covered, in future we can add a payment portal for the students to pay the fee for the enrolled courses and an on demand live chat or video room for the students to communicate with the faculty members.