



CS 353 COURSE PROJECT

Project Proposal

Ucollege

Muhammed Emre Yıldız, Murat Angın, Osman Batur İnce, Ümit Yiğit Başaran

Instructor: Özgür Ulusoy

Teaching Assistant: Mustafa Can Çavdar

February 24, 2020

1. Introduction	2
2. Project Description	2
2.1. Why is a Database System Needed?	3
2.2. How a Database System Will Be Used?	3
2.3. Database Items	3
3. Requirements	5
3.1. Functional Requirements	5
3.1.1 User	5
3.1.2 Admin	5
3.1.3 Creator	6
3.2. Non-functional Requirements	6
3.2.1 Resilience	6
3.2.2 Performance	6
3.2.3 Extensibility	6
3.2.4 Usability	6
3.2.5 Data Integrity	6
3.2.6 Security	7
3.2.7 Scalability	7
4. Limitations	7
5. ER Diagram	8
6. Conclusion	9
7. Website	9

1. Introduction

Online course platforms became an essential part of students' lives because of the COVID-19 pandemic. Udemy, Coursera, and Udacity are some examples of online course platform websites. In this project, an online course platform will be proposed, designed, and also its system will be implemented.

This report starts with the project description, which has general information about the platform. Moreover, this section investigates why and how a database system will be used in this project. After that, the requirements of the projects will be explained. The requirements contain the functional requirements that describe the functionalities and the system's scope, the non-functional requirements that take the user experience into account with the criteria that the system should satisfy. In the limitations section, the constraints of the system and the user boundaries will be investigated. In the end, the ER diagram of the system will be presented.

2. Project Description

Ucollege is a platform where users can search for courses in different categories and buy the courses. Each course consists of lectures separated into different subjects. Courses have Q&A sections where users can ask course-related questions to the course creator, and creators answer their questions. Ucollege enables users to track how far they have come in their courses, and rate owned courses after completing them with a rating score and a comment. Moreover, users can add notes to their lectures. They can refund courses that they do not want for some valid and rational reason. Users can add courses to their wish lists and get certificates if they complete the course by finishing all of the lectures. Furthermore, users can complain about the courses to a site admin. Course creators can make announcements about their courses and publish new courses. Site admins offer discounts on courses that the creators allowed such actions, deal with the user complaints about the courses, and evaluate and conclude the users' return requests. In addition to the previous specifications, we have also added some new features such as carts, bookmarks, tags, and comments. Users can add courses to carts to buy courses in multiple, can add bookmarks to specific times at owned course lectures. Each course can have

multiple tags that allow users to look for specific types of courses. The project's aim is to create an interactive, effective learning platform for the users.

2.1. Why is a Database System Needed?

In general, data is persisted in file systems or database systems. As this project aims to serve concurrent users managing their data within the constraints, and there is a vast amount of data, the file system with the relations can cause serious problems. The relational model cannot be represented with file systems, as the constraints and the management of the data would not be accessible and maintainable. The concurrent access of multiple users to file systems would be chaotic as file systems cannot read or write as fast as a database system. Another reason to choose a database system is managing permissions and accessibility constraints of data. Database systems are more flexible in terms of restricting permissions. Thus, a database system allows us to store and use vast amounts of complex data more effectively, and this system is more feasible.

2.2. How a Database System Will Be Used?

The database system will be used for multiple purposes. The first one addresses to store, retrieve, and represent complex data easily and effectively. This data includes entities and relations inside the platform. Moreover, database queries are a huge advantage of a database system. They allow us to maximize normalization and allow users to search different types of entities such as courses or tags more efficiently. The updates, removals, and insertions of any kind of entry are simpler, and we can manage multiple tables easier.

2.3. Database Items

- **Person:** By the Person entity, we will store general information needed for all the users, admins, and creators. These will be username, photo, name, surname, email, and password.
- **User:** Users will have all the properties Person entity has got, besides, they will have `isCoursesVisible` attributes. If a user wants to hide the courses bought from other users he/she can change the `isCoursesVisible` attribute.

- **Admin:** Admin is a special entry for system administrators, they will have different privileges such as accepting/rejecting complaints, and creating discounts.
- **Creator:** Creator entity will be used by people who record and upload courses to the system. In order to advertise themselves well, they have different attributes such as about, website, LinkedIn, and Youtube. “website”, “linkedin”, and “youtube” will store respective URLs in order to show them in their profile. Also, a Creator can answer questions about their courses and allow different discounts on their courses coming from site admins if they want.
- **Course:** The Course entity will store id, title, description, thumbnail, and category. It will have different relations with a lot of different entities. They will have different lectures, different tags for search and filter purposes, discounts added by their Creators, questions & answers about them coming from both Users and Creators.
- **Lecture:** The Lecture entity will be used for each different session of a course. It will have a chapterName and title in order to distinguish different lectures. It also has an attribute called isVisible which can be adjusted by the Creator to show and hide a lesson from their Users and additionalMaterial which contains additional resources that are optional to add a lecture by the Creator. It has a date attribute and it will be used for ordering purposes and duration which stores the length of the lecture.
- **Note:** The Note entity will be used for personal notes taken by Users for a specific Lecture of a Course. It will contain title and content attributes which will be filled by the User.
- **Bookmark:** The Bookmark entity is a special timeline point that will be used to indicate important points of a lecture. It has got a timestamp attribute in order to detect the exact point of a bookmark in a Lecture.
- **Rating:** Users can leave ratings with comments under courses in order to show their feelings about a specific Course after completing it. Therefore the Rate entity will have ID, ratingScore, title, content, and date attributes.
- **Tag:** Each Course can contain different Tags that will be used to increase the filter and searchability among different Courses. Tag has name and color attributes.
- **Cart:** Cart entity will contain different courses which will be purchased by a User. Cart has the ID attribute.

- **Wishlist:** Wishlist will be used as a special list that contains courses a User will purchase.
- **Discount:** The Discount entity will store different attributes such as percentage, startDate, and endDate. A discount will be created by a site admin and will be allowed by a Creator.
- **Announcement:** Announcements are special entries that will contain different information about a course and they will be added by Creators.
- **Question:** Questions are special messages which will be shown on the QA page of a Course and Creators can answer them.
- **Answer:** Answers are special messages which will be shown on the QA page of a Course and they will be created by Creators in response to a specific Question.
- **Complain:** Complains are special messages which will be used for reports about a course. It will contain different sections like title, content, and refundNeeded. refundNeeded will be used to indicate whether or not a user requests a refund.

3. Requirements

3.1. Functional Requirements

3.1.1 User

Each user has a unique id, profile photo and isCourseAvailable attributes. Unique id is used to distinguish users. They have profile photos like commonly used social platforms and have isCourseAvailable attributes to give users an option to share their courses that they bought.

Users can buy a course and rate a course. Also, they can add a course to their wish lists that every user has. During lessons, users can take notes at specific lecture times. In addition, they can create bookmarks for courses. They can comment on courses and ask questions to creators about the courses. Also, a user can get a certificate when they finish a course in the Ucollege.

3.1.2 Admin

Admins have the same attributes as Person which are unique id, username, name, surname, photo, password, and email. Admins can offer a discount to creators for

their courses. However, the last decision is determined by the creators. In addition, admins can approve the complaints about the courses which are created by users.

3.1.3 Creator

Creator has the same attributes as the person. In addition to these attributes, creators have about, website, linkedin and youtube attributes. These attributes are used for creators to link their website, LinkedIn profile, and YouTube profile. They have courses that consist of lessons and publish them.

Also, creators can answer the questions which are asked from the user and also allow or reject the discount offer which is created by the admin in Ucollege.

3.2. Non-functional Requirements

3.2.1 Resilience

The system will be implemented carefully so that users will not confront errors and failures. If any error occurs, then this error will be detected and handled as quickly as possible.

3.2.2 Performance

The system should be as smooth as possible to make the user experience perfect. Therefore, the system implementation will be optimized as possible. Because in the project we must compete with large datasets, data operations will be also optimized.

3.2.3 Extensibility

The system will be designed in such a way that it is easy to add new features to the system and when these additions are made the system will not require major changes in the previous system.

3.2.4 Usability

Because user experience is important for the project, the system that we will implement should be easy to use as possible and should be understandable.

3.2.5 Data Integrity

In the entire life-cycle of the system, data has to be accurate and consistent. The operations and actions of the user and the system should be monitored to provide data integrity.

3.2.6 Security

The system to be applied will be designed to give importance to data security. The data of the user should not be shared with other third parties. Authentication should be implemented into our system and operations should be designed considering the roles of the users.

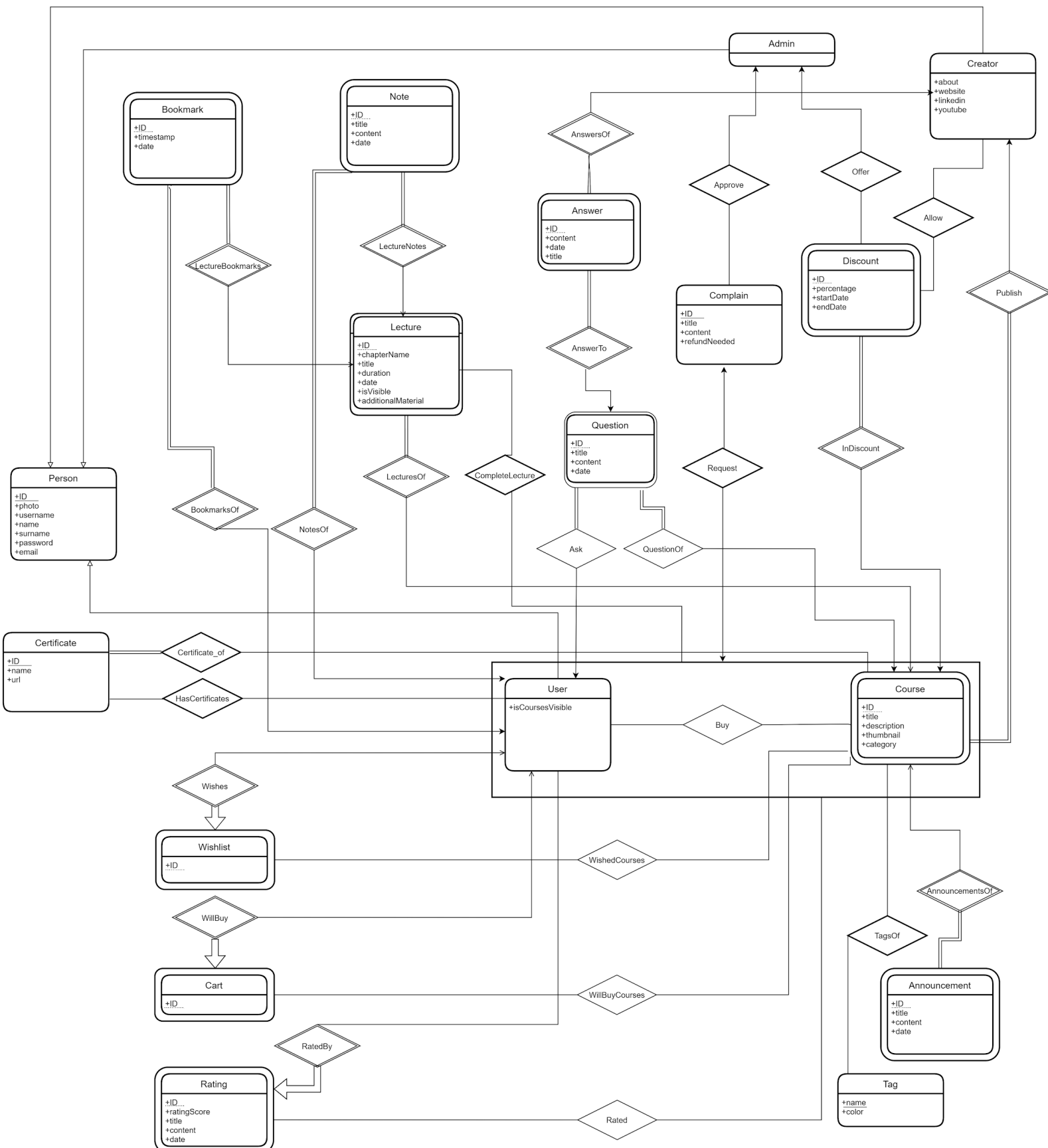
3.2.7 Scalability

The system should be developed in a way that it can provide service to a large audience. Also, it must be resistant to large data operations.

4. Limitations

- Users cannot comment if the course is not bought.
- Users cannot ask questions if the course is not bought.
- Users cannot rate the course if the course is not bought.
- Users cannot take notes if the course is not bought.
- Users cannot complain about the course if the course is not bought.
- Users cannot get certificates until finishing the course.
- Each user can determine whether to share their courses or not.
- Only admins can approve the comments.
- Offers are made by admins, however, approved by creators.
- Creators can only delete or edit their own courses.
- Admins can delete or edit courses as well.
- Admins can ban users or creators.

5. ER Diagram



6. Conclusion

In conclusion, we will implement a web application called Ucollage, an online course platform where users can publish their courses, buy courses, track their process, etc. The proposed system will be user-friendly, consistent, secure, and scalable.

Unnecessary entities, relations, and redundant attributes were cleaned from our system design. Also, constraints of the application are indicated. The proposed E-R diagram is designed for readers to understand. In this report, we explained the importance and aim of the application. The design of the database system, entities, attributes, and relationships were explained. Also, functional and non-functional requirements of the proposed system were indicated in this report.

7. Website

<https://muratangin187.github.io/CS353-Project/>