



Bilkent Üniversitesi

Database Systems

Department of Computer Engineering

Spring 2021

Final Report

Online Course Platform – Sapientia

Project Group 4:

- Ömer Ünlüsoy - Section 01 - 21702136
- Elif Gamze Güliter - Section 01 - 21802870
- İrem Tekin - Section 01 - 21803267
- Ece Ünal - Section 01 - 21703149

Supervisor:

- Uğur Gündükay

Table of Contents

1. Introduction.....	4
1.1 Contribution of Each Group Member	5
2. Final E/R Diagram.....	6
3. Final Relation Schemas	7
3.1. Student.....	7
3.2. Instructor	7
3.3. Admin.....	7
3.4. Course	7
3.5. Section	7
3.6. Lecture	8
3.7. Quiz	8
3.8. Quiz_Question.....	8
3.9. Note	8
3.10. Course_Announcement	9
3.11. QnA_Entry_Student.....	9
3.12. QnA_Entry_Instructor.....	9
3.13. Complaint_Entry_Student	9
3.14. Complaint_Entry_Instructor	10
3.15. Wishlist	10
3.16. Enrolls	10
3.17. Certificate.....	10
3.18. Teaches	10
3.19. Evaluates_Complaint_Entry_Student.....	11
3.20. Evaluates_Complaint_Entry_Instructor.....	11
3.21. Discount	11
3.22. Zoom_Session	11
3.23. Take_Quiz.....	11
3.24. Take_Quiz_Question	12
3.25. Take_Lecture	12
4. Implementation Details	13
4.1. Languages	13

4.2. Database and Connection.....	13
4.3. Implementation	13
5. Advance Database Features.....	14
5.1. Student Main Page Course Query.....	14
5.2. Login Page Course Query.....	14
5.3. Admin Refund Request Query	15
5.4. Certificate Query	15
5.5. Lecture Content Order Query	15
6. User Manual.....	16
6.1. Instructor User Manual	16
6.2. Student User Manual	31
6.3. Admin User Manual	50
7. Website	62
8. Appendix	63
8.1. SQL Create Table Code	63

1. Introduction

In this project, we implemented a web-based course platform similar to Udemy. The purpose of the platform is to enable instructors to build their online courses as well as students to reach out to these contents and educate themselves. The name of the platform is **Sapientia** which means **wisdom** in Latin. In Sapientia, instructors can create courses about the topics that they are proficient about and students can purchase these courses in order to enroll. A course consists of several sections that have lectures as well as quizzes. Students can watch the lectures and solve quizzes. Students receive a certificate indicating that they have finished the online course if only if they watch all of the lectures and get at least the threshold value that is decided by the instructor as their quiz averages. In addition, Sapientia consists of students' wishlist, students' notes for a specific lecture, Q&A section for students and instructors to interact with each other, ratings of courses, discount or refund for courses, and at last, announcements made by instructors about a course.

1.1 Contribution of Each Group Member

1.1.1. Ömer Ünlüsoy

Contributed to E/R Diagram design. Designed the relational schemas. Designed the SQL create table queries. Implemented several data manipulation queries including search and filter. Implemented several pages like contents, view lecture, view quiz.

1.1.2. İrem Tekin

Contributed to E/R diagram, SQL queries for UI's and mockup designs in design report, UI design, data manipulation with SQL queries, general structure (navigation etc.) of the platform. Implemented most of the pages for student, instructor and admin with PHP and HTML and edited some CSS files for the pages.

1.1.3. Gamze Güliter

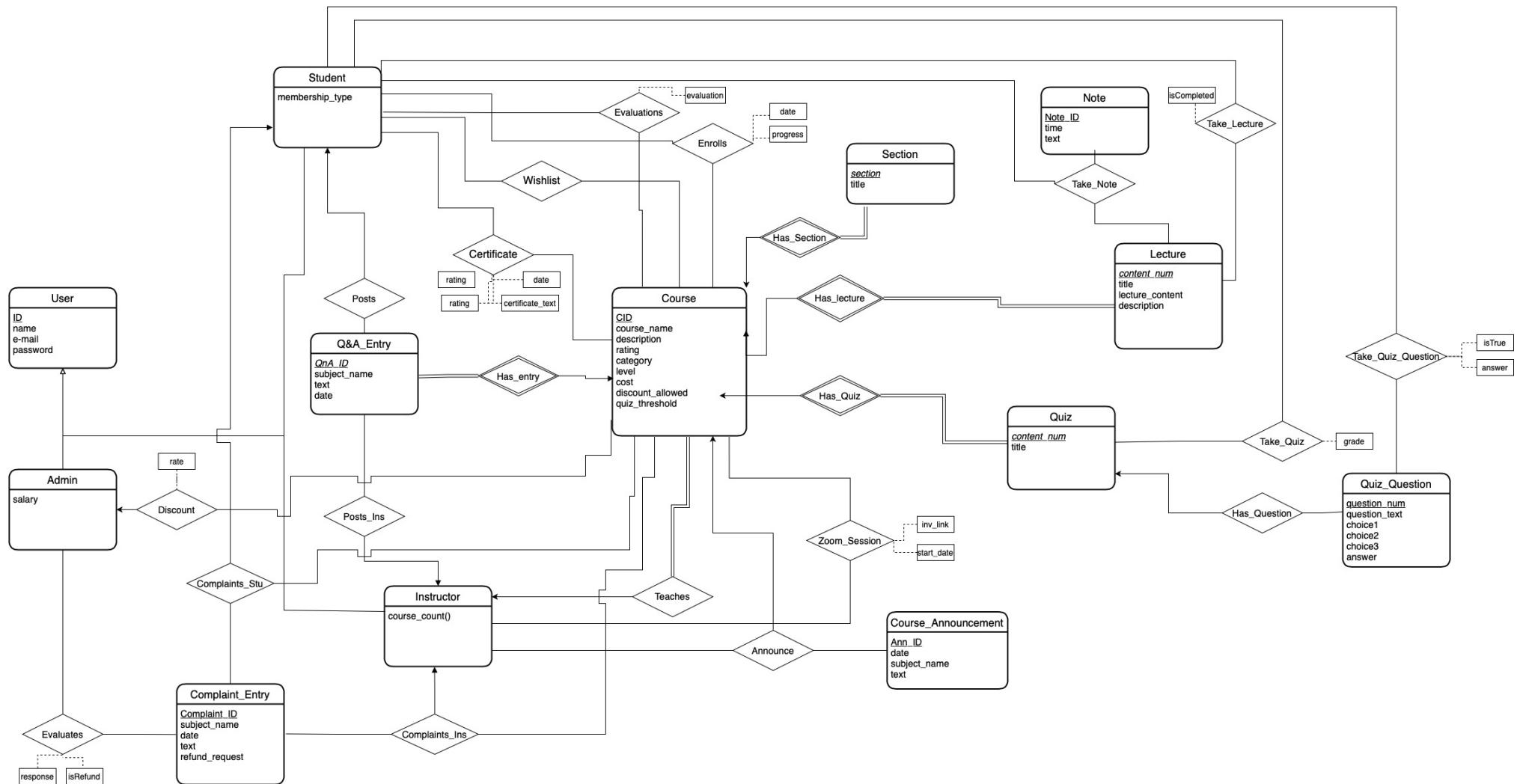
I have worked on Graphical User Interface most of the time, I implemented most of the pages with the help of İrem and Ece. I also helped my friends on back-end side. I assisted them on implementation of php and debugging process.

1.1.4. Ece Ünal

Contributed to E/R Diagram design. Helped to implement SQL queries, and UI design. Contributed to implementation of most of the front-end part. Contributed to debugging process for the PHP and HTML codes. Edited some CSS files for front-end to achieve a more user friendly design.

2. Final E/R Diagram

E-R Diagram of Sapientia



Note: Used software does not have a dashed underline so weak entities are marked with italic underline for this E/R.

3. Final Relation Schemas

3.1. Student

Student(SID, name, e_mail, password, membership_type)

3.2. Instructor

Instructor(IID, name, e_mail, password)

3.3. Admin

Admin(AID, name, e_mail, password, salary)

3.4. Course

Course(CID, IID, course_name, description, rating, category, level, cost, discount_allowed, quiz_threshold)

Foreign Key: IID REFERENCES Instructor(IID)

Unique: (IID, course_name)

3.5. Section

Section(CID, section, title)

Foreign Key: CID REFERENCES Course(CID)

Index: section

3.6. Lecture

Lecture(CID, section, content_num, title, lecture_content, description)

Foreign Key: (CID, section) REFERENCES Section(CID, section)

Unique: (CID, title)

Index: section

Index: content_num

3.7. Quiz

Quiz(CID, section, content_num, IID, section, title)

Foreign Key: (CID, section) REFERENCES Section(CID, section)

Foreign Key: IID REFERENCES Instructor(IID)

Unique: (CID, title)

Index: section

Index: content_num

3.8. Quiz_Question

Quiz_Question(CID, section, content_num, question_num, question_text, choice1, choice2, choice3, answer)

Foreign Key: (CID, section, content_num) REFERENCES Quiz(CID, section, content_num)

3.9. Note

Note(Note_ID, SID, section, content_num, time, text)

Foreign Key: SID REFERENCES Student(SID)

3.10. Course_Announcement

Course_Announcement(Ann_ID, IID, CID, date, subject_name, text)

Foreign Key: IID REFERENCES Instructor(IID)

Foreign Key: CID REFERENCES Course(CID)

3.11. QnA_Entry_Student

QnA_Entry_Student(QnA_ID_S, SID, CID, subject_name, text, date)

Foreign Key: SID REFERENCES Student(SID)

Foreign Key: CID REFERENCES Course(CID)

3.12. QnA_Entry_Instructor

QnA_Entry_Instructor(QnA_ID, QnA_ID_S, IID, CID, subject_name, text, date)

Foreign Key: QnA_ID_S REFERENCES QnA_Entry_Student(QnA_ID_S)

Foreign Key: IID REFERENCES Instructor(IID)

Foreign Key: CID REFERENCES Course(CID)

3.13. Complaint_Entry_Student

Complaint_Entry_Student(Complaint_ID, SID, CID, subject_title, text, date, refund_request)

Foreign Key: SID REFERENCES Student(SID)

Foreign Key: CID REFERENCES Course(CID)

3.14. Complaint_Entry_Instructor

Complaint_Entry_Instructor(Complaint_ID, SID, CID, subject_name, text, date)

Foreign Key: IID REFERENCES Instructor(IID)

Foreign Key: CID REFERENCES Course(CID)

3.15. Wishlist

Wishlist(SID, CID)

Foreign Key: SID REFERENCES Student(SID)

Foreign Key: CID REFERENCES Course(CID)

3.16. Enrolls

Enrolls(SID, CID, progress, date)

Foreign Key: SID REFERENCES Student(SID)

Foreign Key: CID REFERENCES Course(CID)

3.17. Certificate

Certificate(SID, CID, certificate_text, rating, comment, date)

Foreign Key: SID REFERENCES Student(SID)

Foreign Key: CID REFERENCES Course(CID)

3.18. Teaches

Teaches(IID, CID)

Foreign Key: IID REFERENCES Instructor(IID)

Foreign Key: CID REFERENCES Course(CID)

3.19. Evaluates_Complaint_Entry_Student

Evaluates_Complaint_Entry_Student(AID, Complaint_ID, response, isRefund)

Foreign Key: AID REFERENCES Admin(AID)

Foreign Key: IID Complaint_ID REFERENCES Complaint_Entry_Student(Complaint_ID)

3.20. Evaluates_Complaint_Entry_Instructor

Evaluates_Complaint_Entry_Instructor(AID, Complaint_ID, response)

Foreign Key: AID REFERENCES Admin(AID)

Foreign Key: Complaint_ID REFERENCES Complaint_Entry_Instructor(Complaint_ID)

3.21. Discount

Discount(AID, CID, rate)

Foreign Key: AID REFERENCES Admin(AID)

Foreign Key: CID REFERENCES Course(CID)

3.22. Zoom_Session

Zoom_Session(Session_ID, IID, CID, invite_link, start_date)

Foreign Key: IID REFERENCES Instructor(IID)

Foreign Key: CID REFERENCES Course(CID)

3.23. Take_Quiz

Take_Quiz(SID, CID, section, content_num, grade)

Foreign Key: SID REFERENCES Student(SID)

Foreign Key: (CID, section, content_num) REFERENCES Quiz(CID, section, content_num)

3.24. Take_Quiz_Question

Take_Quiz_Question(SID, CID, section, content_num, question_num, answer, isTrue)

Foreign Key: SID REFERENCES Student(SID)

Foreign Key: (CID, section, content_num, question_num) REFERENCES Quiz(CID, section, content_num, question_num)

3.25. Take_Lecture

Take_Quiz_Lecture(SID, CID, section, content_num, isCompleted)

Foreign Key: SID REFERENCES Student(SID)

Foreign Key: (CID, section, content_num) REFERENCES Lecture(CID, section, content_num)

4. Implementation Details

4.1. Languages

Four different languages have been used in this project. The first language is SQL which is used for database definition and modification. The second language is PHP. It is used to send queries to the database at the server side. The connection to the database has been established by PHP. The third language is HTML which is used to create Web user interface. HTML is selected due to its compatibility with PHP. Both PHP and HTML code is located at .php files and they can send information to each other which eases the coding. Last language is CSS, a style sheet language helped project to look pretty. Most HTML and CSS code have been produced by an application called Nicepage. IntelliJ and PHPstorm IDEs are used.

4.2. Database and Connection

MySQL has been used in this project for both database definition and modification. The database is located at one member's Dijkstra account so that every member can access. The connection is established by a PHP script.

4.3. Implementation

First, the E/R diagram and schemas are designed as the first stage of the implementation. SQL create table queries are designed and tables are created at a Dijkstra account. Main page structure of the website is determined and SQL database manipulation queries are designed for each page. Then, page user interfaces are implemented with HTML and CSS. At this point, the app created static HTML and CSS codes which was a major challenge for the UI part.

After implementing the UI, HTML files are turned into PHP files and UI components like buttons and text fields, are connected. After then, SQL manipulation queries are implemented in PHP files and each query is tested with several cases. Finally, page and session connections are implemented.

Gamze Güliter and Ece Ünal were mainly responsible from the web UI (HTML and CSS) while Ömer Ünlüsoy and İrem Tekin were mainly responsible for SQL queries and PHP. However, every group member has worked at every stage of the implementation.

5. Advance Database Features

5.1. Student Main Page Course Query

```
select *
from Discount right join Course on Course.CID = Discount.CID and
rate between $discount_min and $discount_max
natural join Instructor
where
cost between $price_min and $price_max
and category = $category and level = $level
and (course_name like '%$search_word%' or Instructor.name like '%$search_word%')
limit $page_num, $page_entry_num;
```

IID	AID	CID	rate	CID	course_name	description		
1	1	1	20.00	1	Music Theory for Beginners	This course aims to teach the basics of music theory to newcomers.		
rating	category	level	cost	discount_allowed	quiz_threshold	name	e_mail	password
95.00	Music	Beginner	30.99	0	8.00	Irem Tekin	irem@tekin	1234

Figure 1: Example run with rate between 0, 20; cost between 10, 50; category = Music; level = Beginner; search word = music

This query shows all the courses that satisfy student's criteria at the student main page. Everything starting with \$ sign is a variable taken from the student. This query joins three tables; uses price, level, and discount filters. It also implements search feature with “like” phrase. Finally, it uses “limit” to limit the number of Courses listed at each page.

5.2. Login Page Course Query

```
select *
from Discount right join Course on Course.CID = Discount.CID
natural join Instructor
order by rate DESC
limit 0, 5;
```

IID	AID	CID	rate	CID	course_name	description		
1	1	2	20.00	2	Introduction to Badminton	This course aims to introduce badminton basics.		
2	1	1	20.00	1	Music Theory for Beginners	This course aims to teach the basics of music theory to newcomers.		
3	1	<null>	<null>	4	SQL 2	SQL 2		
4	3	<null>	<null>	12	CS 353	database course:)		
5	1	<null>	<null>	7	Linear Algebra for Beginners	Linear Algebra for Beginners course		
rating	category	level	cost	discount_allowed	quiz_threshold	name	e_mail	password
0.00	Sports	Beginner	12.99	1	6.00	Irem Tekin	irem@tekin	1234
95.00	Music	Beginner	30.99	0	8.00	Irem Tekin	irem@tekin	1234
0.00	Computer Science	Intermediate	25.00	0	7.00	Irem Tekin	irem@tekin	1234
0.00	Gardening	Intermediate	80.00	0	7.00	Mustafa Cavdar	cavdar@cavdar2	1234
0.00	Math	Beginner	45.89	1	10.00	Irem Tekin	irem@tekin	1234

Figure 2: Example run

This query lists top five courses according to their discount rate at the login page.

5.3. Admin Refund Request Query

```
select count(*) as refund_count
from Complaint_Entry_Student
where CID = '$CID' and refund_request = 1;
```

	refund_count	
	1	0

Figure 3: Example run with CID = 3

This query counts the number of refund request made by a specific student for admin.

5.4. Certificate Query

```
select *
from (
    select course_name, CID
    from Course
) as C_subquery
natural join Certificate
where SID = '$SID';
```

CID	course_name	SID	certificate_text	rating	comment	date
1	Music Theory for Beginners	1		95.00	well designed :)	2021-05-17
2	Introduction to Living with a Cat	1	<null>	10.00	<null>	<null>

Figure 4: Example run with SID = 1

This nested query shows every course, particular student has finished and got certificate for.

5.5. Lecture Content Order Query

```
select *
from Lecture
where CID = '$CID' and section = '$section_name'
order by content_num;
```

CID	section	content_num	title	lecture_content	description
1	1	1	Welcome to the Music Theory	upload/trial.mp4	WELCOME NEWCOMERS
1	1	2	Rhythm Theory	upload/trial.mp4	This lecture explains the Rhythm Theory

Figure 5: Example run with CID = 1 and section = 1

This query lists every lecture in a particular section of a course and orders according to their content number in the section.

6. User Manual

6.1. Instructor User Manual

6.1.1. Instructor Log In

Instructors will login to the Sapientia website by the Login Page.

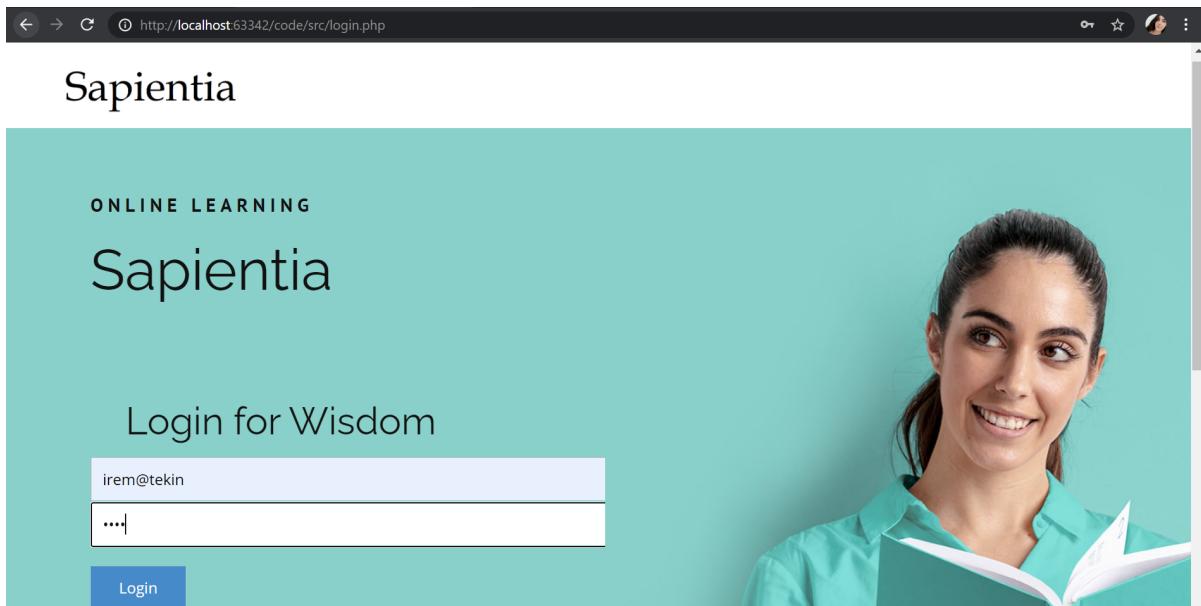


Fig. 6. Instructor Login

6.1.2. Instructor's Main Page

After log in, instructors will be directed to the main page which lists all the courses that the instructor teaches. In order to see this page, the instructor can also use the “**My Courses**” button in the navigation bar.

The screenshot shows a web browser window with the URL http://localhost:63342/code/src/instructor_main_courses.php#. The page title is "Sapientia". The top navigation bar includes links for "Account", "Forum", "My Courses", "My Announcements", and "Logout". A large red button labeled "Add Course" is centered above a table. The table has three columns: "Course Name", "Category", and "Level". It lists two courses: "Music Theory for Beginners" (Category: Music, Level: Beginner) and "Introduction to Badminton" (Category: Sports, Level: Beginner). Each course row contains four buttons: "EDIT", "STATISTICS", "COMPLAINT", and "ANNOUNCE". A cursor arrow is visible on the left side of the table.

Course Name	Category	Level
Music Theory for Beginners	Music	Beginner
Introduction to Badminton	Sports	Beginner

Fig. 7. Instructor's Main Page

6.1.3. Adding a Course to the Sapientia Website

In order to add a course, instructor should press the “**Add a Course**” button on the main page. This button will direct the instructor to the “**Add a Course**” page. In Add a Course page, the instructor should specify the name of the course that he/she will teach, category, the level of course, brief description about the course, its price and quiz threshold. Quiz threshold indicates the min grade that a student needs to collect from the quizzes in order to get a certificate from the course.

Add a Course

Course Name	Introduction to Living with a Cat
Category	Psychology
Level	Beginner
This course is for cat lovers.	
Price (\$)	40
Min. Quiz Average (.../10)	7

Fig.8. Adding a course to the Sapientia website

	Science					
SQL 3	Computer Science	Advanced	EDIT	STATISTICS	COMPLAINT	ANNOUNCE
Immanuel Kant	Philosophy	Intermediate	EDIT	STATISTICS	COMPLAINT	ANNOUNCE
Linear Algebra for Beginners	Math	Beginner	EDIT	STATISTICS	COMPLAINT	ANNOUNCE
HTML Basics	Graphic Design	Beginner	EDIT	STATISTICS	COMPLAINT	ANNOUNCE
Introduction to Living with a Cat	Psychology	Beginner	EDIT	STATISTICS	COMPLAINT	ANNOUNCE

Fig.9. Instructor's main page after adding a course with the name "Introduction to Living with Cats"

6.1.4. Editing an Added Course

In order to edit a course, the instructor should press the “Edit” button next to the course name in the main page. Editing a course includes uploading lecture videos, uploading quizzes, changing its name, price, quiz threshold, level, description etc.

As an **additional feature**, the instructor can schedule a zoom meeting with the enrolled students by giving a date and a zoom link. In order to schedule a zoom meeting, the instructor needs to use “Edit a Course” page.

The screenshot shows a web browser window titled "Edit a Course". The URL in the address bar is http://localhost:8084/Code/scr/Edit_Course.php. The page displays the following course information:

- Course Name: Introduction to Living with a Cat
- Category: Psychology
- Level: Beginner
- Description: This course is for cat lovers. We all love cats!!
- Price (\$): 30
- Min. Quiz Average (.../10): 7.00
- Zoom Link: (empty)

On the right side of the form, there are six pink rounded rectangular buttons, each with a plus icon and a label:

- ADD SECTIONS
- ADD LECTURE
- ADD QUIZ
- INVITE INSTRUCTORS
- EDIT LECTURES
- EDIT QUIZIES

The screenshot shows a web-based course editing interface. On the left, there are several input fields:

- Description:** A text area containing the text "This course is for cat lovers. We all love cats!!".
- Price (\$):** An input field containing "30".
- Min. Quiz Average (.../10):** An input field containing "7.00".
- Zoom Link:** A text area containing a Zoom link: "https://us02web.zoom.us/postattendee?id=43".
- Date:** A date input field showing "18.05.2021".
- Time:** A time input field showing "12.00".

On the right side, there are four pink rounded rectangular buttons:

- ADD QUIZ** (with a plus sign icon)
- INVITE INSTRUCTORS** (with a user icon)
- EDIT LECTURES** (with a pencil icon)
- EDIT QUIZIES** (with a pencil icon)

At the bottom center is a large red button labeled "Edit Course" with a hand cursor icon.

Fig 10 and 11. Edit a Course Page

6.1.5 Adding Sections to a Course

In Sapientia, instructors should add sections to their courses, then, add lectures and quizzes. In order to add a section to the course, the instructor should use the “**Add a Section**” button in the course’s edit page. While adding a section, instructor should specify a name and a number for the section.

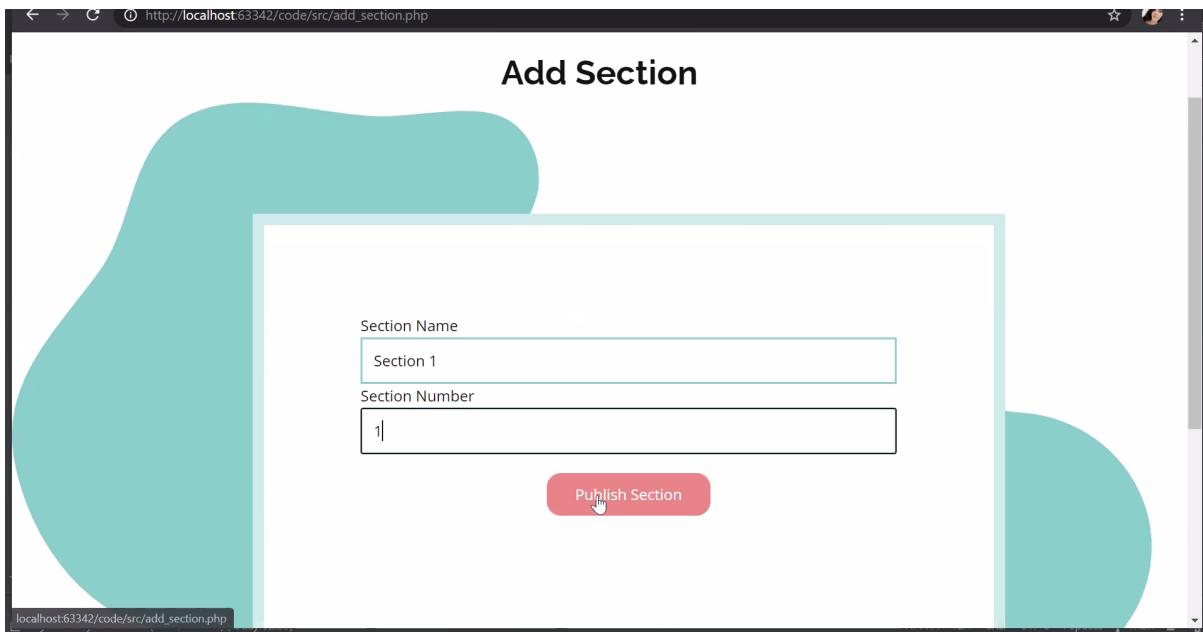


Fig.12 Adding a Section to the Course

6.1.6 Adding Lectures to the Course

In order to add a lecture to the course, an instructor first needs to add at least one section to the course. Following that, in order to add a lecture, the instructor should press the "**Add a Lecture**" button in the edit page. The button will direct the instructor to the "Add a Lecture" page. From that page, the teacher must specify a name, a number, brief description, and a section for the lecture.

In order to upload a lecture video, the instructor must press to the "**Submit Video**" button on the right corner and upload a lecture video from the local computer (See Figures 15,16)

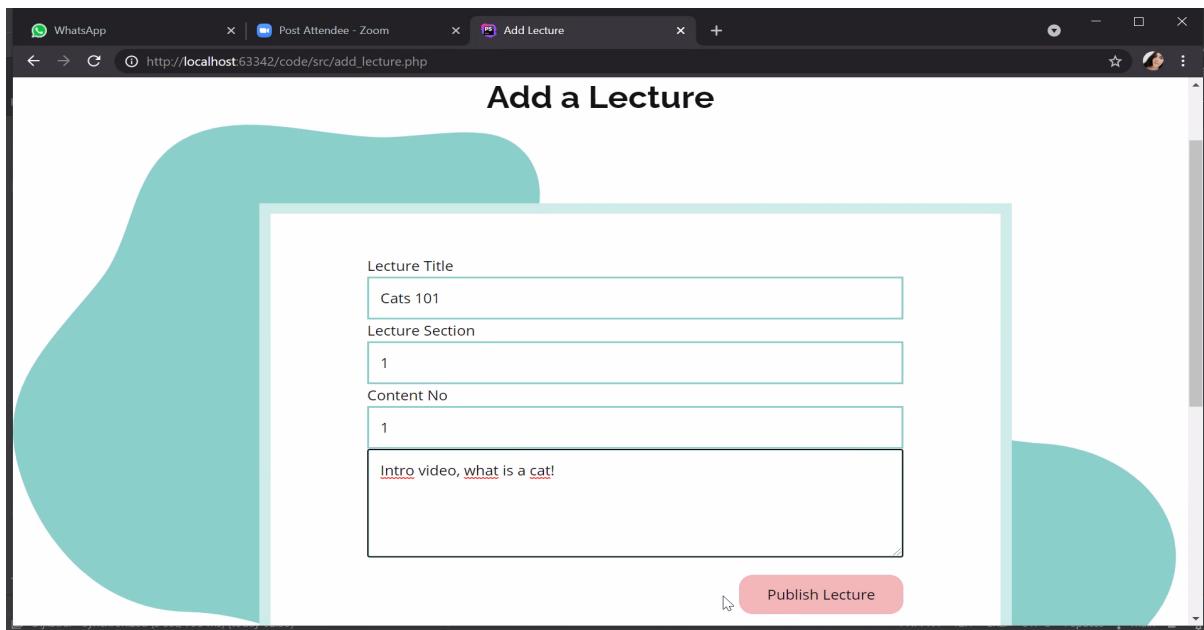


Fig. 13. Add a Lecture Page

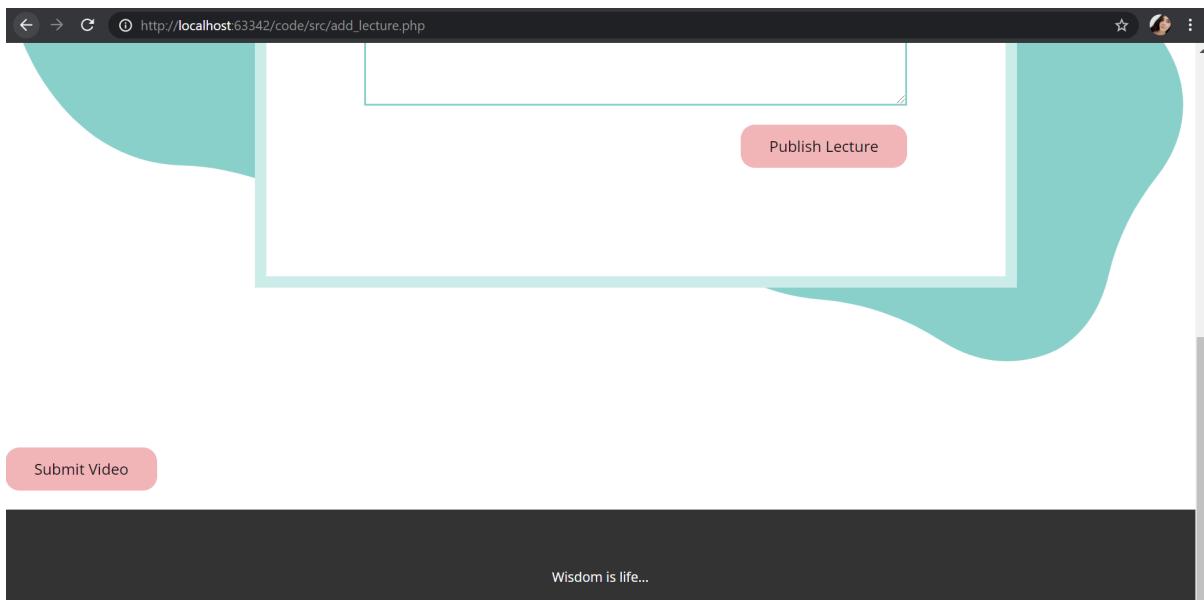


Fig.14 Submit Video Button

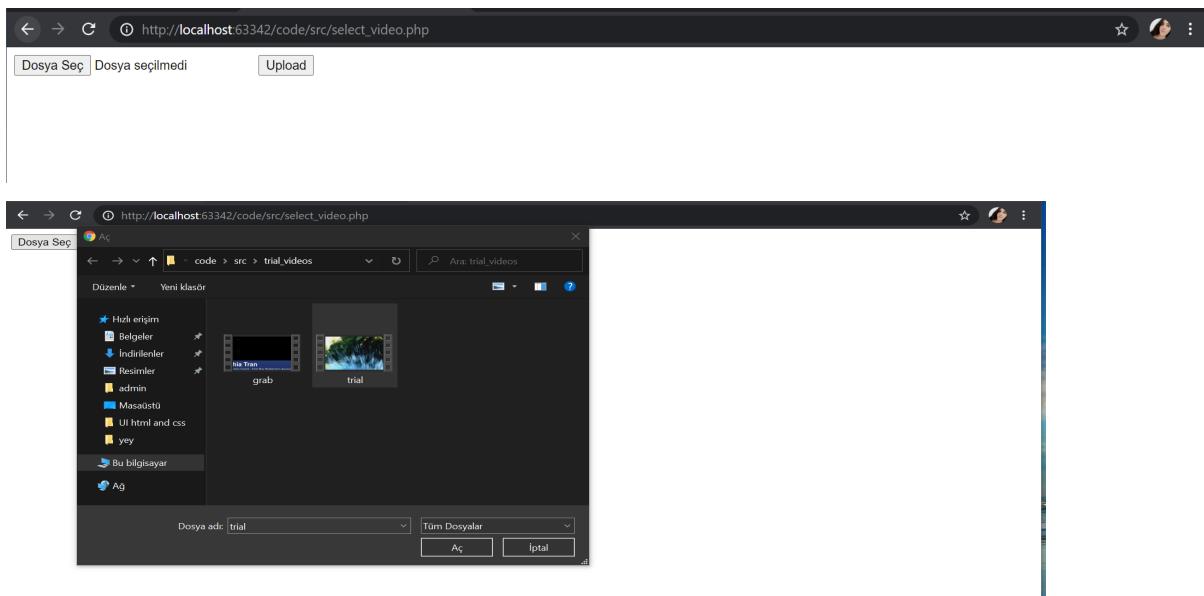


Fig 15 and 16. Uploading a Lecture Video from Local Computer

6.1.7 Adding Quizzes to the Course

In order to add a course to the course, an instructor first needs to add at least one section to the course. Following that, in order to add a quiz, the instructor should press the “**Add a Quiz**” button in the edit page. The button will direct the instructor to the “Add a Quiz” page. From that page, the teacher must specify a name, a number and a section for the quiz.

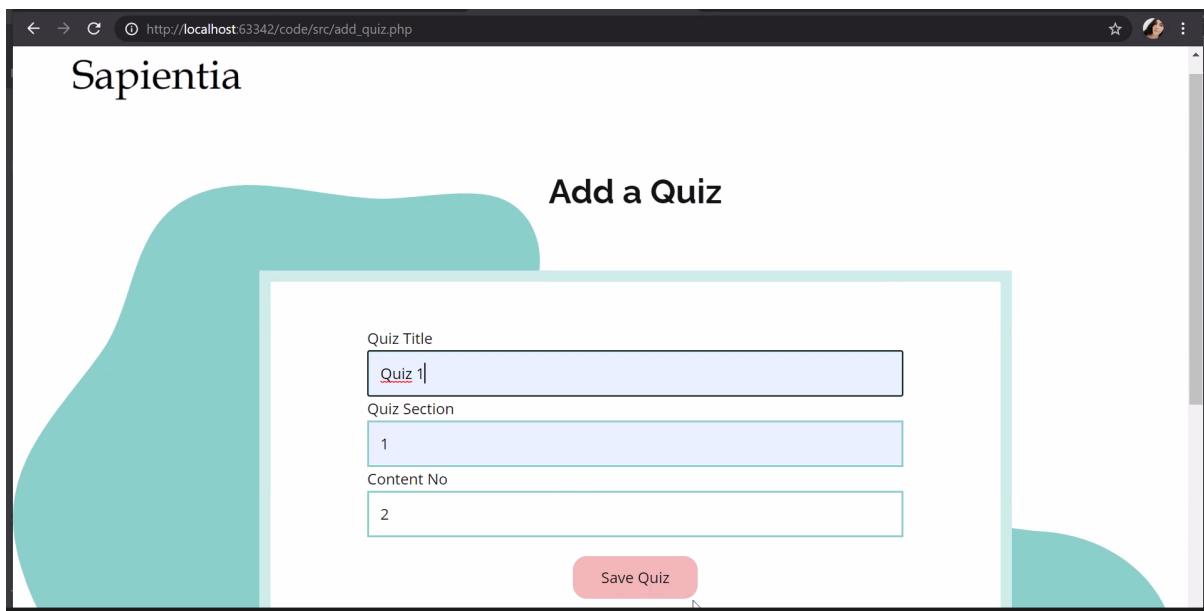


Fig .17 Adding a Quiz

After adding a quiz, instructor should add questions to that quiz. In “**Add a Quiz Question**” page, instructor should write a question, three options and indicate the correct answer of the question. Following that, the instructor can either continue adding questions or finish adding by pressing to the “**Finish**” button.

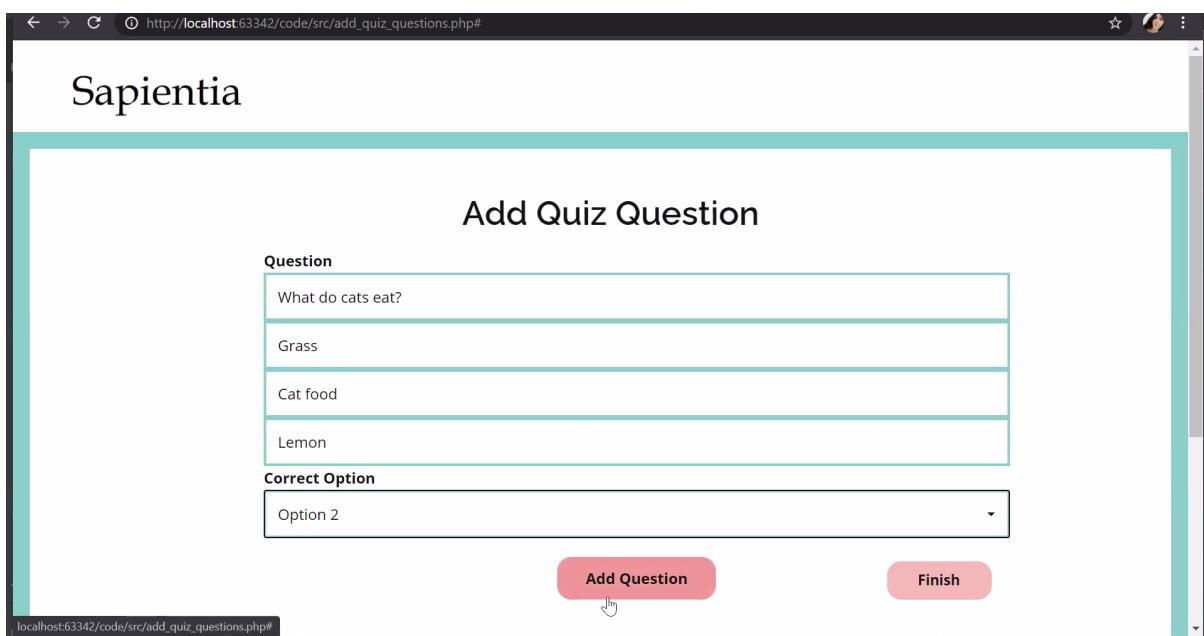


Fig 18. Adding Questions to a Quiz

6.1.8 Inviting an Instructor to a Course

As an **additional feature**, instructors can invite other instructors to the course as content creators. In order to do this, the instructor should press the “**Invite an Instructor**” button on the “**Edit Course**” page. The button will direct the instructor to the page shown below. From there, the instructor should give the name and email address of the other instructor in order to add him/her as the content creator.

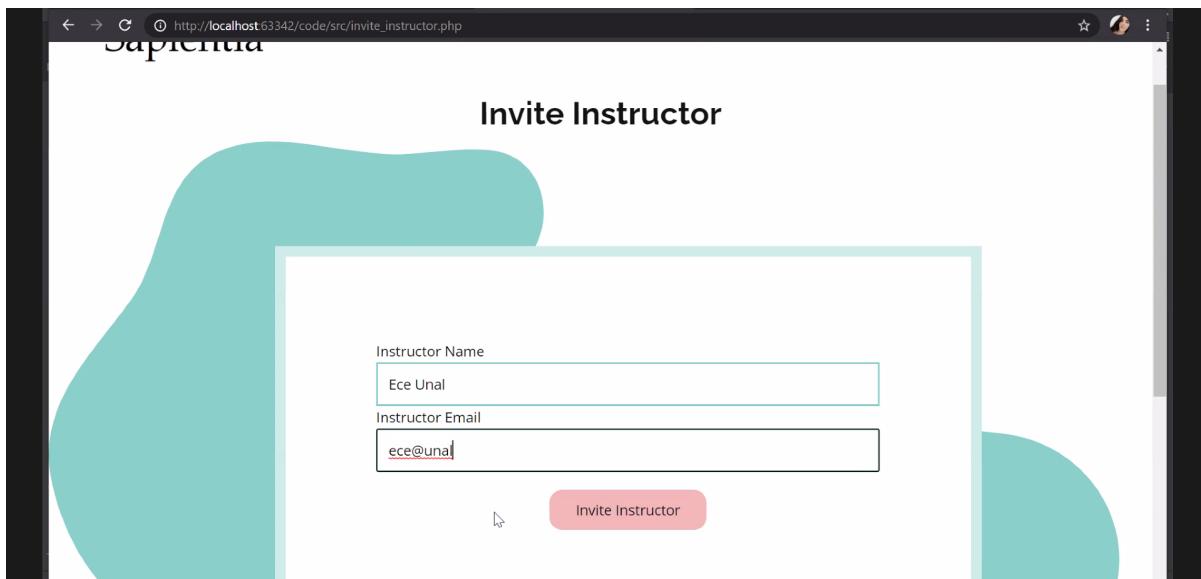


Fig 19. Inviting Another Instructor to the Course.

6.1.9 Filling a Complaint About a Course

An instructor can add a complaint such as a technical issue about his/her course to the admin. In order to fill a complaint, he/she needs to press to the “**Fill a Complaint**” button which is next to the course’s name in the main page. From this page, the instructor must specify a subject, date and a brief description about the complaint and press to the “**Send Complaint**” button.

The screenshot shows a web page titled "Fill a Complaint" at the top center. Below the title, there are three input fields: "Subject" containing "I cannot upload video", "Date" containing "17.05.2021", and "Description" containing "I cannot upload video with size 2 mb". A "Send Complaint" button is located at the bottom right of the form area. The background features a large teal shape on the left and a red shape on the right.

Subject
I cannot upload video

Date
17.05.2021

Description
I cannot upload video with size 2 mb

Send Complaint

Fig 20. Filing a Complaint About a Course

6.1.10 Making an Announcement About a Course

An instructor can make an announcement about his/her course to the enrolled students. In order to make it, he/she needs to press the "**Announce**" button which is next to the course's name on the main page. The button will direct the instructor to the "Make Announcement" page. From there, the instructor should specify a subject, date and a description about the announcements.

Subject
New lecture coming

Date
17.05.2021

Description
I will share a video today, please|

Announce

Fig 21. Making an Announcement About a Course

An instructor can see all of his/her announcements in the Sapientiae system by pressing the “**My Announcements**” button in the navigation bar. From there, the instructor can also delete any announcement by pressing to the “**DELETE**” button.

Course Name	Announcement
Music Theory for Beginners	We have a discount for this week.
SQL 2	New video is not coming this week!
Introduction to Living with a Cat	I will share a video today, please go and watch. Good day!

Fig 22. My Announcements Page

The screenshot shows a web browser window for the Sapientia platform. The URL is http://localhost:63342/code/src/instructor_announcements.php#. The page has a header with the Sapientia logo and navigation links for Account, Forum, My Courses, My Announcements, and Logout. A red "Add Course" button is visible above a table. The table has two rows. The first row contains "Course Name" and "Announcement" columns. The second row contains "SQL 2" and "New video is not coming this week!". Each row has a "DELETE" button to its right.

Course Name	Announcement
SQL 2	New video is not coming this week!

Fig 23. My Announcements Page After Deleting the Announcement About the Course
“Introduction to Living with a Cat”

6.1.11 Forum

An instructor can see all the questions from enrolled students about his/her courses from the “**Forum**” button in the navigation bar. From there, the instructor can answer any question about his courses by simply pressing to the “**Answer Question**” button next to the question itself.

The screenshot shows a web-based forum interface. At the top, there is a navigation bar with links for 'Account', 'Forum', 'My Courses' (which is underlined, indicating it is the active page), and 'My Announcements'. On the right side of the top bar is a 'Logout' link. Below the navigation bar, the title 'Q&A Section' is centered. A table follows, with a header row containing 'Student Name' and 'Question'. Three rows of data are present in the table:

Student Name	Question
Gamze Guliter 2	I cannot fasten the video, I want 3x.
Gamze Guliter 2	Explain please!
Gamze Guliter 2	asd2

For each question in the table, there is a red-bordered button labeled 'ANSWER QUESTION' on its right side.

Fig 24. Instructor's Forum Page

After pressing the “Answer Question” button, the instructor will be directed to the “Answer the Question” page. From there the instructor can submit an answer for the enrolled student’s question about a course.

The screenshot shows a web-based form for answering a question. At the top, there is a navigation bar with links for 'Account', 'Forum', 'My Courses' (underlined), and 'My Announcements'. On the right side of the top bar is a 'Logout' link. Below the navigation bar, the title 'Answer the Question' is centered. The question text 'I cannot fasten the video, I want 3x.' is displayed. Below the question is a text input field with the placeholder 'Write Your Answer Below' and the instruction 'You cannot do that in Sapientia go use Udemy. :)'. At the bottom of the input field is a red-bordered button labeled 'Post Answer'. The bottom of the screen features a black footer bar with various icons: Mute, Start Video, Security, Participants (2), Chat, Share Screen, Record, Reactions, and an End button.

Fig 25. Answer the Question Page

6.1.12 Statistics

An instructor can see the statistics about his/her courses such as the average student rating, number of enrolled students and the number of refund requests about the course. In order to see the “Statistics” page, the instructor should press to the “Statistics” button which is next to the course’s name in the main page.

The screenshot shows a web browser window with the URL http://localhost:63342/code/src/instructor_course_statistics.php#. The page is titled "Sapientia". At the top, there are navigation links: Account, Forum, My Courses (which is underlined), My Announcements, and Logout. Below the navigation, the course title "Introduction to Living with a Cat" is displayed with a bar chart icon. Underneath the title, the word "Statistics" is centered. The page lists three statistics: "Number of Enrolled Students : 1", "Rating : 10.00", and "Number Of Refund Requests: 0".

Fig 26. Statistics about the Course “Introduction to Living with a Cat”

6.2. Student User Manual

6.2.1 Student Sign Up

Students will sign up to the Sapientia website by the Signup Page as entering their name, email, password and clicking the “Sign Up as Student” button.

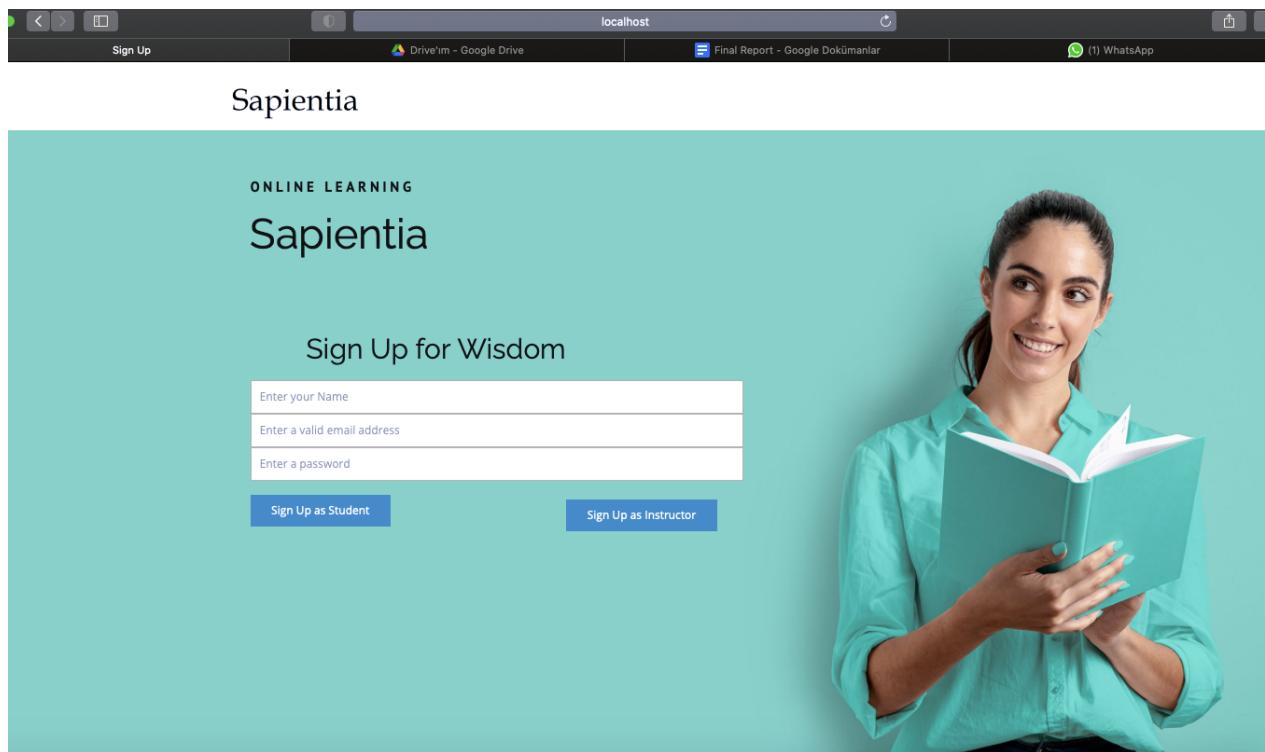


Fig 27. Student Sign Up

6.2.2 Student Login

Students will login to the Sapientia website by the Login Page.

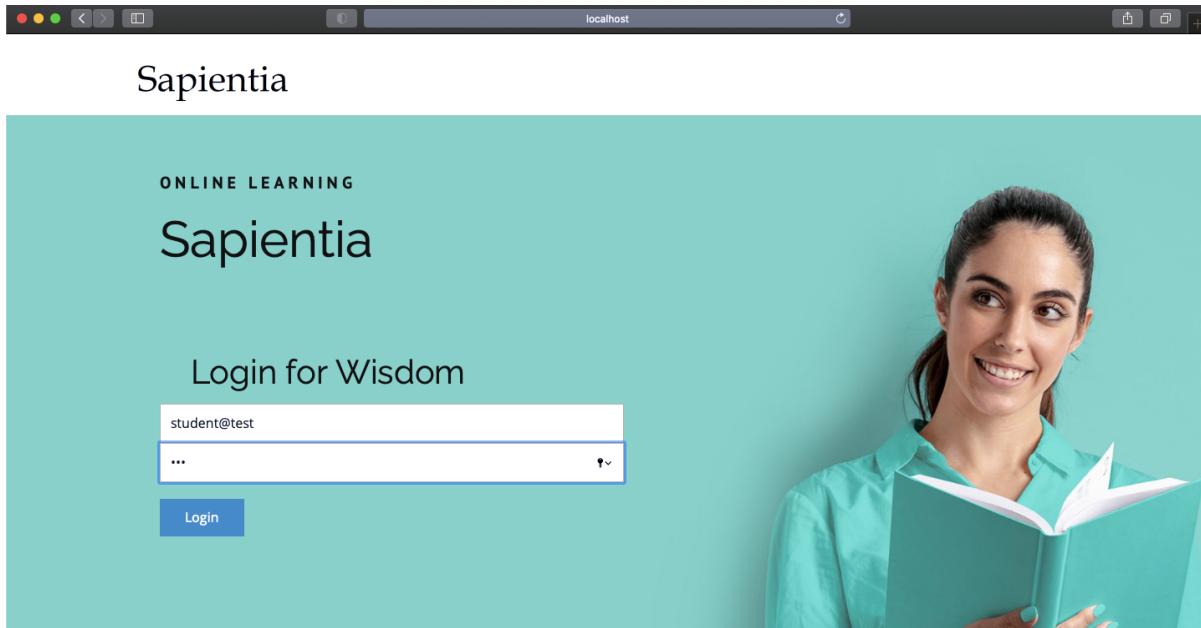


Fig 28. Student Login

6.2.3 Student's Main Page

In this page, student will see the courses included in the Sapientia. This page is opened after student logs in. In order to see this page, the student can also use the "**Main Page**" button in the navigation bar. Student can search a course or an instructor by a keyword with the search bar as entering the keyword here and pressing the Enter button on their keyboard. Furthermore, student can filter the course list by course level, category, price and discount by changing the values in the select boxes according to their need and clicking the "**Filter**" button.

Moreover, in the courses table that labeled as "Popular Courses" there are two buttons for each course: "**VIEW**" and "**ADD TO WISHLIST**". By using the "**VIEW**" button, student can see the course page that includes course details such as course description and comments about the course. By using the "**ADD TO WISHLIST**" button, student can add the course to their wishlist.

The figure below shows the page before the filtering.

The figure displays two screenshots of the Sapientia application interface. The top screenshot shows the search interface with filtering options, and the bottom screenshot shows the resulting list of popular courses.

Top Screenshot (Search Interface):

- Header: Account, Notifications, My Courses, Fill a Complaint, Certificates, Wishlist, Main Page, Logout.
- Search bar: Search in Courses or Instructors...
- Filtering dropdowns:
 - Level: All
 - Category: All
 - Price: All
 - Discount: All
- Filter button: Filter

Bottom Screenshot (Popular Courses List):

Popular Courses

Course Name	Price	Category	Level	Instructor		
Music Theory for Beginners	30.99₺ 24.79₺	Music	Beginner	Irem Tekin	VIEW	ADD TO WISHLIST
Introduction to Badminton	12.99₺ 10.39₺	Sports	Beginner	Irem Tekin	VIEW	ADD TO WISHLIST
SQL 1	22.00₺	Computer Science	Beginner	Irem Tekin	VIEW	ADD TO WISHLIST
SQL 2	25.00₺	Computer Science	Intermediate	Irem Tekin	VIEW	ADD TO WISHLIST
SQL 3	45.99₺	Computer Science	Advanced	Irem Tekin	VIEW	ADD TO WISHLIST
Immanuel Kant	67.99₺	Philosophy	Intermediate	Irem Tekin	VIEW	ADD TO WISHLIST
Linear Algebra for Beginners	45.89₺	Math	Beginner	Irem Tekin	VIEW	ADD TO WISHLIST

HTML Basics	77.89₺	Graphic Design	Beginner	Irem Tekin	VIEW	ADD TO WISHLIST
Introduction to Living with a Cat	30.00₺	Psychology	Beginner	Irem Tekin	VIEW	ADD TO WISHLIST
Tulip Science	55.99₺	Gardening	Beginner	Ece Unal	VIEW	ADD TO WISHLIST

[LEFT](#) [RIGHT](#)

Fig. 29. Student Main Page

After filtering with the keyword ‘sql’ by entering the value to search bar for example:

Popular Courses

Course Name	Price	Category	Level	Instructor		
SQL 1	22.00₺	Computer Science	Beginner	Irem Tekin	VIEW	ADD TO WISHLIST
SQL 2	25.00₺	Computer Science	Intermediate	Irem Tekin	VIEW	ADD TO WISHLIST
SQL 3	45.99₺	Computer Science	Advanced	Irem Tekin	VIEW	ADD TO WISHLIST

[LEFT](#) [RIGHT](#)

Fig. 30. Filtered table by keyword

Listing only advanced level courses:

The screenshot shows a web browser window with the URL `localhost`. The page title is "student main". On the left, there's a sidebar with navigation links: "COURSES", "INSTRUCTORS", "ABOUT", "CONTACT", "LOGOUT". The main content area has a search bar at the top with placeholder text "Search in Courses or Instructors...". Below it is a filter section with dropdown menus for "Level" (set to "Advanced"), "Category" (set to "All"), "Price" (set to "All"), and "Discount" (set to "All"). A "Filter" button is located below the dropdowns. The heading "Popular Courses" is followed by a table with the following data:

Course Name	Price	Category	Level	Instructor		
SQL 3	45.99₺	Computer Science	Advanced	Irem Tekin	VIEW	ADD TO WISHLIST
Life of Nazim Hikmet	78.99₺	Literature	Advanced	Ece Unal	VIEW	ADD TO WISHLIST
How to Sell Drugs Online (fast)	199.99₺	Medicine	Advanced	Ece Unal	VIEW	ADD TO WISHLIST

Fig. 31. Filtered table by level

Listing advanced courses in Computer Science category:

The screenshot shows a web browser window with the URL `localhost`. The page title is "student main". The navigation links on the left are "COURSES", "INSTRUCTORS", "ABOUT", "CONTACT", "LOGOUT". The main content area has a search bar at the top with placeholder text "Search in Courses or Instructors...". Below it is a filter section with dropdown menus for "Level" (set to "Advanced"), "Category" (set to "Computer Science"), "Price" (set to "All"), and "Discount" (set to "All"). A "Filter" button is located below the dropdowns. The heading "Popular Courses" is followed by a table with the following data:

Course Name	Price	Category	Level	Instructor		
SQL 3	45.99₺	Computer Science	Advanced	Irem Tekin	VIEW	ADD TO WISHLIST

Below the table are two buttons: "LEFT" and "RIGHT". At the bottom center of the page is the number "1".

Fig. 32. Filtered table by label and category

Listing courses between prices 35-50 liras:

The screenshot shows a search bar at the top with placeholder text "Search in Courses or Instructors...". Below it is a filter section with dropdown menus for Level (All), Category (All), Price (\$15-\$35), and Discount (All). A teal "Filter" button is located to the right of the dropdowns. Below the filter section is a heading "Popular Courses" followed by a table.

Course Name	Price	Category	Level	Instructor		
Music Theory for Beginners	30.99₺ 24.79₺	Music	Beginner	Irem Tekin	VIEW	ADD TO WISHLIST
SQL 1	22.00₺	Computer Science	Beginner	Irem Tekin	VIEW	ADD TO WISHLIST
SQL 2	25.00₺	Computer Science	Intermediate	Irem Tekin	VIEW	ADD TO WISHLIST
Introduction to Living with a Cat	30.00₺	Psychology	Beginner	Irem Tekin	VIEW	ADD TO WISHLIST

Fig. 33. Filtered table by price

Listing courses between prices 35-50 liras and includes “sql” keyword:

The screenshot shows a search bar at the top with the keyword "sql" entered. Below it is a filter section with dropdown menus for Level (All), Category (All), Price (\$15-\$35), and Discount (All). A teal "Filter" button is located to the right of the dropdowns. Below the filter section is a heading "Popular Courses" followed by a table.

Course Name	Price	Category	Level	Instructor		
SQL 1	22.00₺	Computer Science	Beginner	Irem Tekin	VIEW	ADD TO WISHLIST
SQL 2	25.00₺	Computer Science	Intermediate	Irem Tekin	VIEW	ADD TO WISHLIST

At the bottom of the table area, there are navigation buttons labeled "LEFT" and "RIGHT", and a page number "1".

Fig. 34. Filtered table by price and keyword

There are 10 courses displayed per table page. To see the remaining courses, user can click the left and right buttons to navigate to table page. In the screen shot of the website below, the second page of the table is shown since user clicked the “RIGHT” button to increment the page number.

The screenshot shows a web application running on localhost. At the top, there are three dropdown menus: 'All' under 'Category', 'Discount' under 'Category', and 'All' under 'Instructor'. Below these is a 'Filter' button. The main content area is titled 'Popular Courses' and contains a table with the following data:

Course Name	Price	Category	Level	Instructor		
Life of Nazim Hikmet	78.99₺	Literature	Advanced	Ece Unal	VIEW	ADD TO WISHLIST
How to Sell Drugs Online (fast)	199.99₺	Medicine	Advanced	Ece Unal	VIEW	ADD TO WISHLIST
CS 353	80.00₺	Gardening	Intermediate	Mustafa Cavdar2	VIEW	ADD TO WISHLIST

At the bottom of the table, there are navigation buttons labeled 'LEFT' and 'RIGHT'. The number '2' is centered between them, indicating this is the second page of the results.

Fig. 35. Second page of table

6.2.4 Course Page

After clicking the “VIEW” button at main page, student can see the course instructor(s), price, rating, category, level for the selected course. Furthermore, at the bottom part of the page, student can see the comments for the course and at the right side, the description of the course is shown. Before purchasing the course, there are two buttons displayed in this page: “ADD TO WISHLIST” and “BUY”. As the names suggest, student can add the course to their wishlist or buy the course by clicking those

buttons.

The screenshot shows a web browser window with the title bar 'localhost'. The main content area displays a course page for 'SQL 3'. At the top, there's a navigation bar with links: 'View Course', 'Sapientia', 'Account', 'Notifications', 'My Courses' (which is underlined), 'Fill a Complaint', 'Certificates', 'Wishlist', 'Main Page', and 'Logout'. Below the navigation bar, the course title 'SQL 3' is centered. To the left of the title, there's a table with course details:

Instructor(s):	Irem Tekin
Price :	45.99₺
Rating :	Not rated yet!
Category:	Computer Science
Level:	Advanced

To the right of the table, there's a section titled 'About the Course:' with the text 'SQL 3 for advance users.' Below the table, it says 'Quiz: 0, Lecture: 0'. At the bottom right of the page are two buttons: 'Add to Wishlist' and 'Buy'.

Fig. 36. Course page before purchase

6.2.5 After Buying the Course - Course Page

After purchasing the course, course page changes to the version shown below. Student can reach this page from the list that will be displayed by clicking the “**My Courses**” at the navigation bar.

By clicking the “**View Lectures**” button, student can be navigated to the page that shows course contents, which are lectures and quizzes.

The screenshot shows a web browser window with three tabs: 'Course Page' (active), 'localhost' (address bar), and 'Final Report - Google Dokümanlar'. The main content area displays a course titled 'SQL 3'. It includes basic course details: Instructor(s) - Irem Tekin, Rating - Not rated yet!, Category - Computer Science, and Level - Advanced. Below these are sections for 'About the Course' (SQL 3 for advance users) and 'Completeness' (Quiz: 0, Lecture: 0). A red 'View Lectures' button is visible. At the bottom, there's a form titled 'Ask a Question' with fields for 'Subject' (containing 'Question Subject') and 'Question' (containing 'Question'). A red 'Post Question' button is at the bottom left. To the right of the question form is a link 'To see all Q&A's: Go to Course Forum'.

Fig. 37. Course page after purchase by filling the form that labeled as “Ask a Question” student can ask a question about the course to the instructor. In this form, it is asked to write the subject of the question and question itself. After that, student can publish the question by clicking the “Post Question” button.

A zoomed-in view of the 'Ask a Question' form. The title 'Ask a Question' is at the top. Below it, there are two input fields: 'Subject' (containing 'Q1?') and 'Question' (containing 'Question question'). Both fields have a blue border. At the bottom is a red 'Post Question' button.

With the help of the “**Go to Course Forum**” student can go to the forum page where the Q&As are displayed.

To see all Q&A's : [Go to Course Forum](#)

Fig. 38. and 39. Ask Question and Go to Course Forum

6.2.6 Course Forum

In this page, student can view all of the questions that asked to the instructor. Also, by clicking the “**VIEW ANSWER**” button, student will be navigated to the answer page for the related question.

The screenshot shows a web browser window with the title 'localhost'. The tabs at the top include 'Forum', 'Drive' (im - Google Drive), 'Final Report - Google Dokümanlar', and '(2) WhatsApp'. The main content area is titled 'Sapientia' and has a navigation bar with links for 'Account', 'Notifications', 'My Courses', 'Fill a Complaint', 'Certificates', 'Wishlist', 'Main Page', and 'Logout'. Below this is a teal header bar with the text 'Q&A Section'. A table lists four student questions, each with a 'VIEW ANSWER' button.

Student Name	Question	
Gamze Gulter 2	I cannot fasten the video, I want 3x.	VIEW ANSWER
Gamze Gulter 2	Explain please!	VIEW ANSWER
Gamze Gulter 2	asd2	VIEW ANSWER
Student Test	Question question	VIEW ANSWER

Fig. 40. Course forum

6.2.7 View answer to a forum question

Student can see the answer of the question that they wanted to learn in this page.

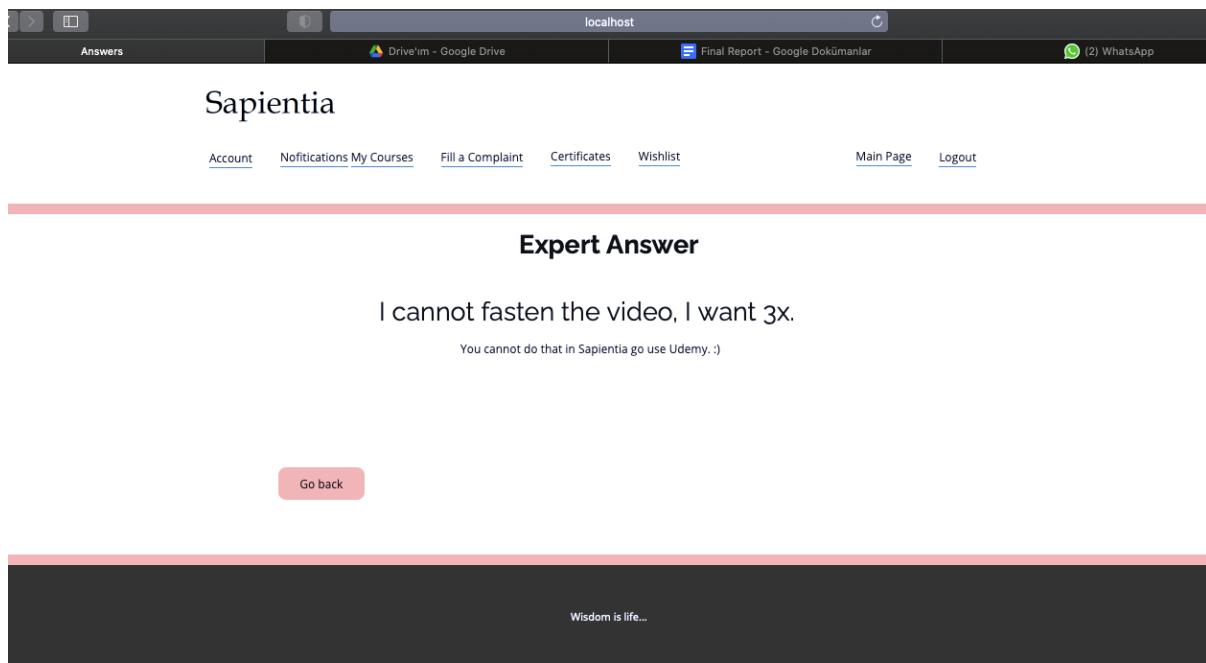


Fig. 41. Answer of the question

6.2.8 See Contents of the Course

Student can see the contents of a course, which are lectures and quizzes in this page. **We divided contents as sections to achieve a more neat look.**

The screenshot shows a web browser window with the title 'localhost' at the top. The address bar contains 'Drive'ın - Google Drive, Final Report - Google Dokümanları, and WhatsApp. Below the address bar, the 'Sapientia' logo is displayed, followed by a navigation menu with links: Account, Notifications, My Courses, Fill a Complaint, Certificates, Wishlist, Main Page, and Logout. A pink horizontal bar spans across the page. Below this, the word 'Contents' is centered. The main content area features a grid layout with three columns. The first column contains section titles: 'Section 1', 'Section 2', and 'Section 3'. The second column contains subject names: 'Fundamentals', 'What is Music?', and 'Grand Staff'. The third column contains action buttons: 'View Lecture' (for sections 1 and 2) and 'Solve Quiz' (for section 1). Below the grid, a dark gray box contains the text 'Wisdom is life...'.

Fig. 42. Contents of a course

6.2.9 Watch Lecture

In this page, student can watch the lecture video and take notes. Student can stop the video by clicking the video image. Also, by clicking the “**Back to content**” button, user can navigate to the Contents page. With clicking “**Previous**” and “**Next**” buttons, student can go to previous and next lecture pages, respectively.

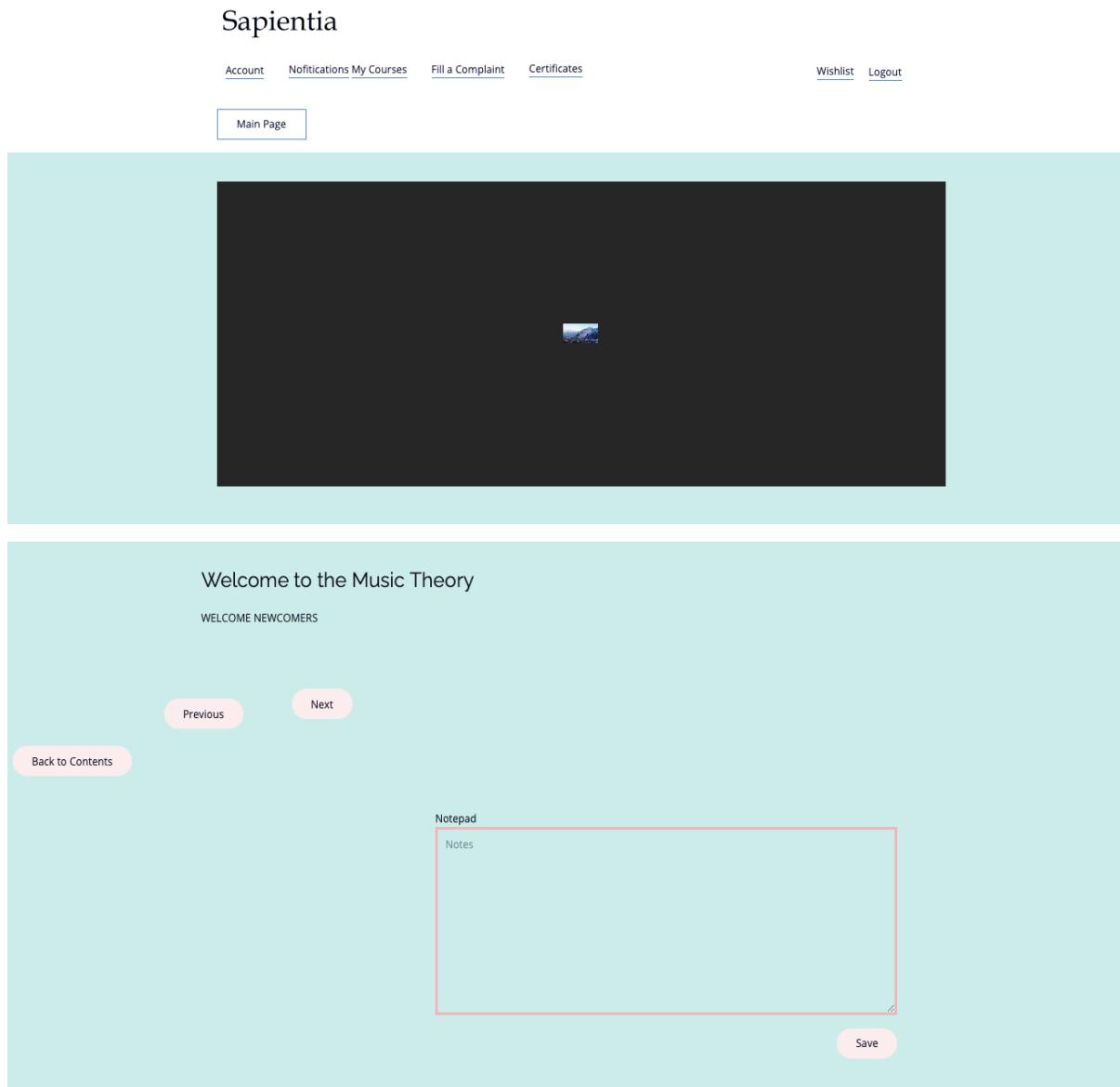


Fig. 43. Watch Lecture page

By filling the text area shown below and hitting the “Save” button, students can take notes for the related lecture.

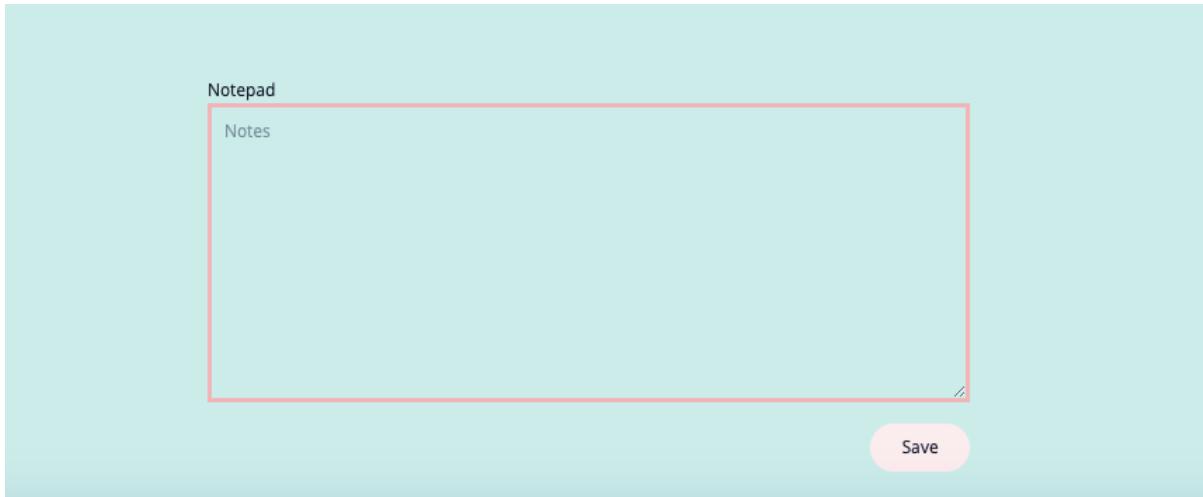


Fig. 44. Notepad section

6.2.10 Solve Quiz

In this page, student can see the quiz question and select one of the choices. With clicking “Previous” and “Next” buttons, student can go to previous and next lecture quiz questions, respectively. Finally, student can submit all of their answers for related quiz by clicking the “Submit Quiz” button. Furthermore, student can see their grade after submitting the quiz, and by clicking the “Back to Contents” button, they can navigate to the Contents page.

Music Theory for Beginners

Rhythm Quiz 1

Question

What is rhythm?

smtg about computers (maybe)
 a strong, regular repeated pattern of movement or
 no idea!

[Previous](#) [Next](#) [Submit Quiz](#)

Grade : .../ 10

[Back to Contents](#)

Fig. 45. Solve Quiz Page

6.2.11 View Wishlist

Student can see their wishlist in this page. By clicking the “VIEW”, “BUY”, “REMOVE” buttons, student can view the course page, buy the course or remove the course from the wishlist respectively.

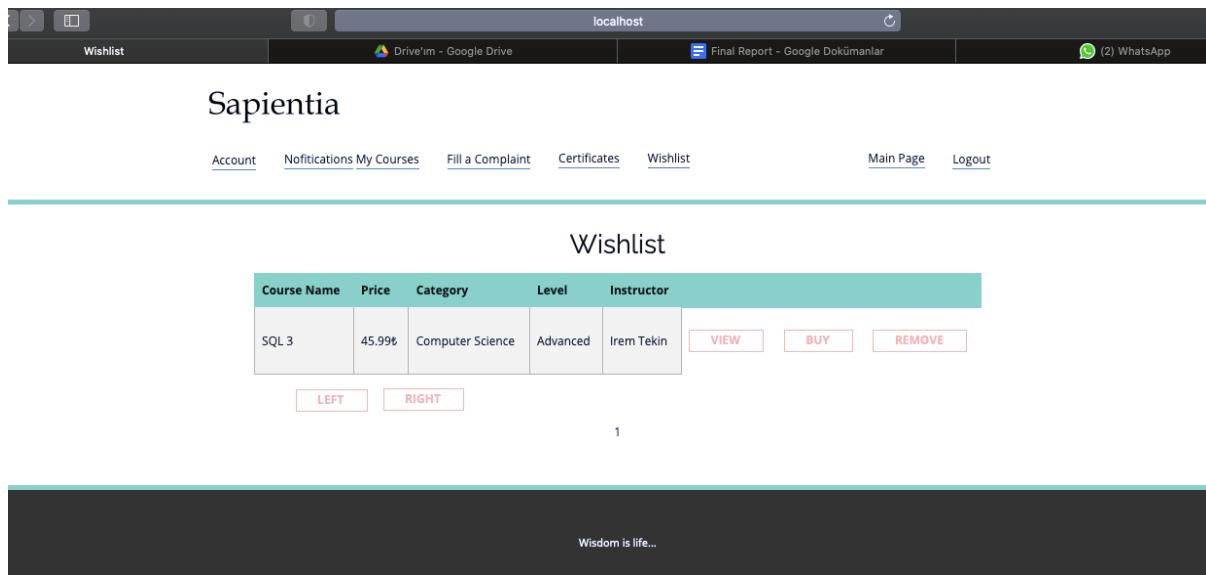


Fig. 46. Wishlist page

6.2.12 View Certificates

In this page, student can see the list of certificates that they got. By clicking the “VIEW CERTIFICATE” as the name indicates, student can see the certificate for the related course.

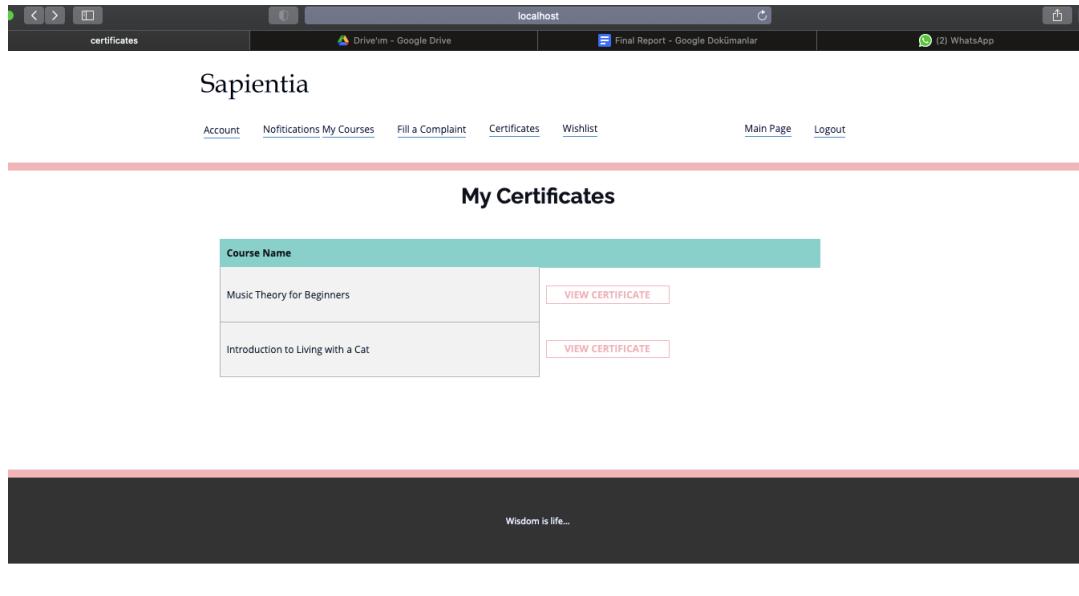


Fig. 47. View Certificates page

6.2.13 Certificate Page

Student can see the certificate for a course that they successfully completed in this page. Moreover, student can send comment about the course by filling the text area labeled as “**Comment Course**” and clicking the “**Send Comment**” button. Also student can rate the course by filling the text area labeled as “**Rate Course**” and clicking the “**Rate**” button.

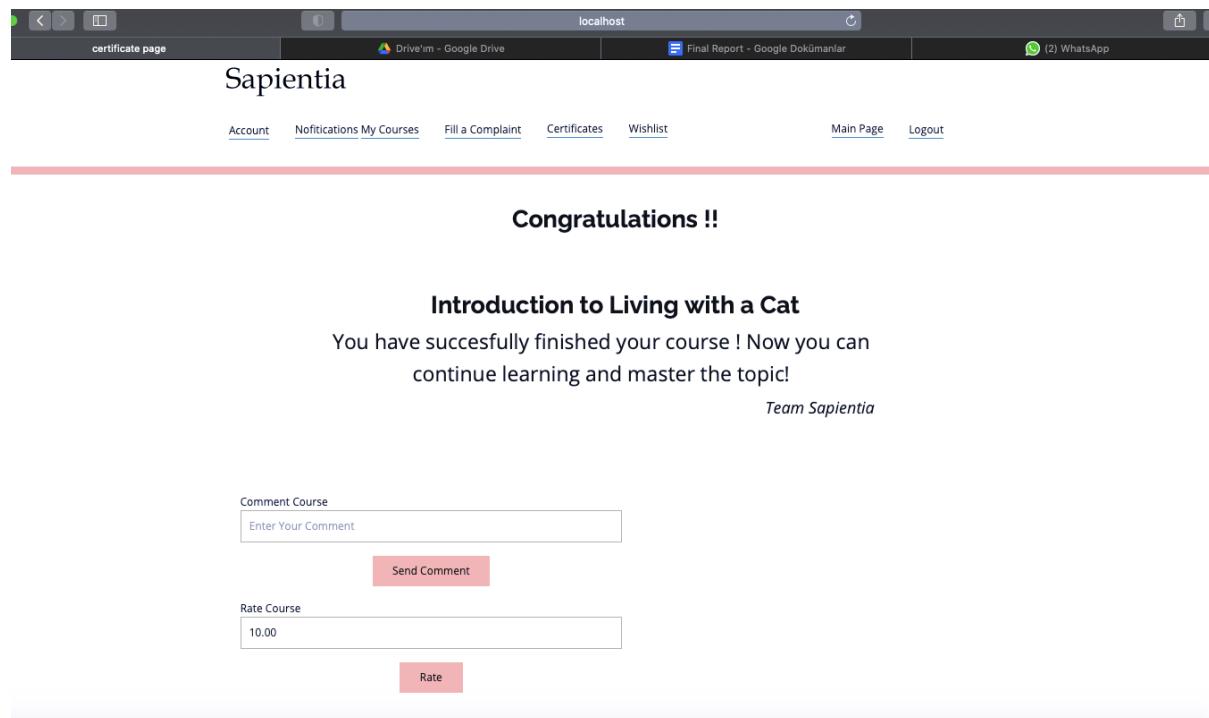


Fig. 48. Certificate page

6.2.14 Filling a Complaint About a Course

A student can add a complaint about a course by the help of this page. In order to fill a complaint he/she needs to press to the “**Fill a Complaint**” button which is next to the course’s name in the main page. From this page, the student must specify a subject, date and a brief

description about the complaint and press to the “**Send Complaint**” button. Also, selecting the “**Refund Request**” choice, student can request a refund from the administration.

The screenshot shows a web browser window titled "localhost" with the URL "Fill a complaint - student". The page is part of the Sapientia LMS system. At the top, there is a navigation bar with links: Account, Notifications, My Courses, Fill a Complaint (which is underlined), Certificates, Wishlist, Main Page, and Logout. Below the navigation bar, the title "Fill a Complaint" is centered. A large form is displayed, enclosed in a red border. The form fields include "Subject" (with placeholder "subject"), "Date" (with placeholder "2021-05-17"), and a "Description" text area. At the bottom of the form, there are two radio buttons: "Refund Request" and "Other". A pink "Send Complaint" button is located at the bottom right of the form area. The background of the page features a teal abstract shape on either side of the form.

Fig. 49. Fill a Complaint page

6.3. Admin User Manual

6.3.1 Admin Login

Sapientia admin can login to the system using the login page.

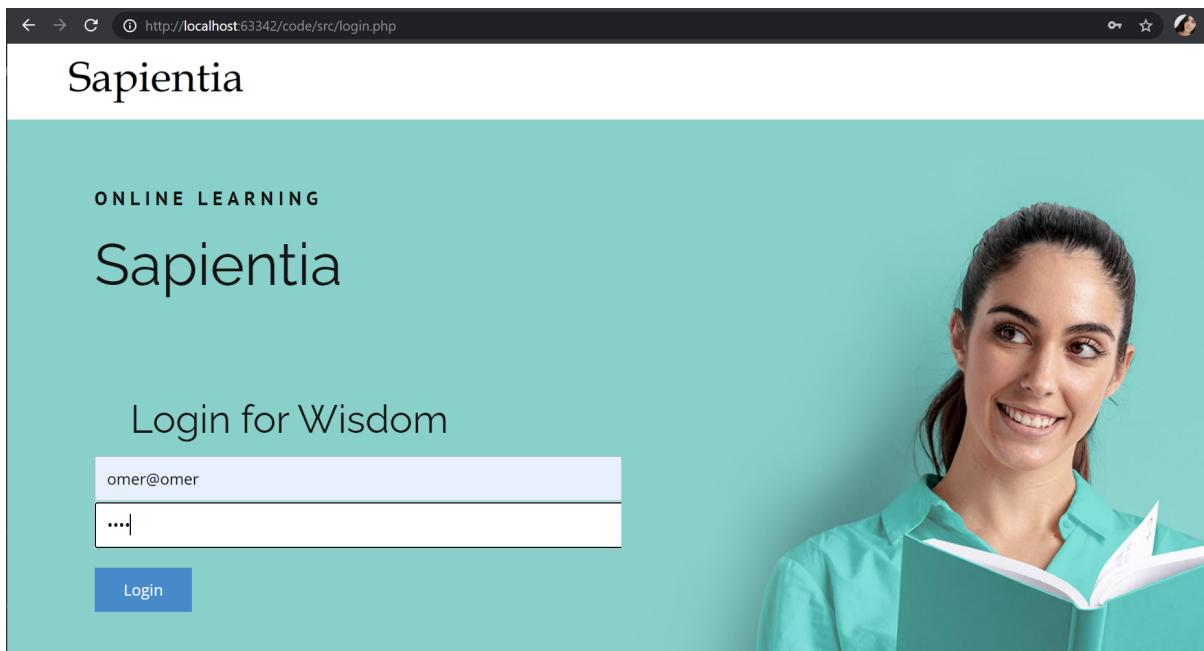


Fig. 49. Admin Login

A screenshot of a database management interface showing an "Admin" table. The table has columns: AID, name, e_mail, and password. There is one row with the values: 1, omer, omer@omer, and 1234. The interface includes a toolbar with various icons and a "Filter Criteria" section.

Fig. 50. Admin Table

6.3.2 Admin Main Page and Filtering

Sapientia admin sees a main page which consists of all courses that are added by instructors. Admin can filter the courses, view the courses statistics and offer discount using this page.

The screenshot shows a web browser window with the URL http://localhost:63342/code/src/admin_main.php#. At the top, there is a search bar labeled "Search in Courses or Instructors...". Below it is a "Filter" section with dropdown menus for "Level" (Advanced), "Category" (All), "Price" (All), and "Discount" (All). A "Filter" button is located at the bottom of this section. Below the filter section, the word "Courses" is centered above a table. The table has a header row with columns: Course Name, Price, Category, Level, and Instructor. The data row for "SQL 3" shows: Course Name (SQL 3), Price (45.99₺ 29.43₺), Category (Computer Science), Level (Advanced), and Instructor (Irem Tekin). To the right of this row are two buttons: "VIEW STATISTICS" and "OFFER DISCOUNT".

Course Name	Price	Category	Level	Instructor
SQL 3	45.99₺ 29.43₺	Computer Science	Advanced	Irem Tekin

Courses

VIEW STATISTICS OFFER DISCOUNT

Fig. 51. Admin Main Page and Filtering

6.3.3 View Course Statistics

Admin can view the statistics about a course by clicking the **view statistics** button. In statistics page, the number of students who enrolled the course, the rating of the course and the number of refund requests for the course are displayed.

The screenshot shows a web browser window with the URL http://localhost:63342/code/src/admin_course_statistics.php#. The page title is "Sapientia". The main content area displays statistics for a course named "SQL 3 Statistics". The statistics include:

- Number of Enrolled Students : 1
- Rating : 0.00
- Number Of Refund Requests: 2

A footer banner at the bottom of the page contains the text "Wisdom is life...".

Fig. 52. View Course Statistics

6.3.4 Discount Management - Apply Discount

In the main page, the **offer discount button** is displayed for the courses that are allowed to get discounts. Admin can manage discounts of a course by clicking the **offer discount button**. The button leads admin to course's price and discount information. Admin can change the discount rate or delete the discount. To apply discount, admin writes a new discount rate, selects apply discount and saves changes. The change in the price of the course is displayed on the main page.

http://localhost:63342/code/src/admin_main.php#

Courses

Course Name	Price	Category	Level	Instructor	
Music Theory for Beginners	30.99₺ 24.79₺	Music	Beginner	Irem Tekin	VIEW STATISTICS
Introduction to Badminton	12.99₺ 10.39₺	Sports	Beginner	Irem Tekin	VIEW STATISTICS OFFER DISCOUNT
SQL 3	45.99₺ 29.43₺	Computer Science	Advanced	Irem Tekin	VIEW STATISTICS OFFER DISCOUNT
SQL 1	22.00₺	Computer Science	Beginner	Irem Tekin	VIEW STATISTICS
SQL 2	25.00₺	Computer Science	Intermediate	Irem Tekin	VIEW STATISTICS

http://localhost:63342/code/src/admin_offer_discount.php#

Sapientia

Add Admin [Student Complaints](#) [Instructor Complaints](#) Main Page Logout

SQL 3

Determined Price by Instructor (Discount will be applied to this price)

45.99

Applied Discount Rate

36.00

New Discount Rate

20

Apply Discount
 Delete Discount

[Save Changes](#)

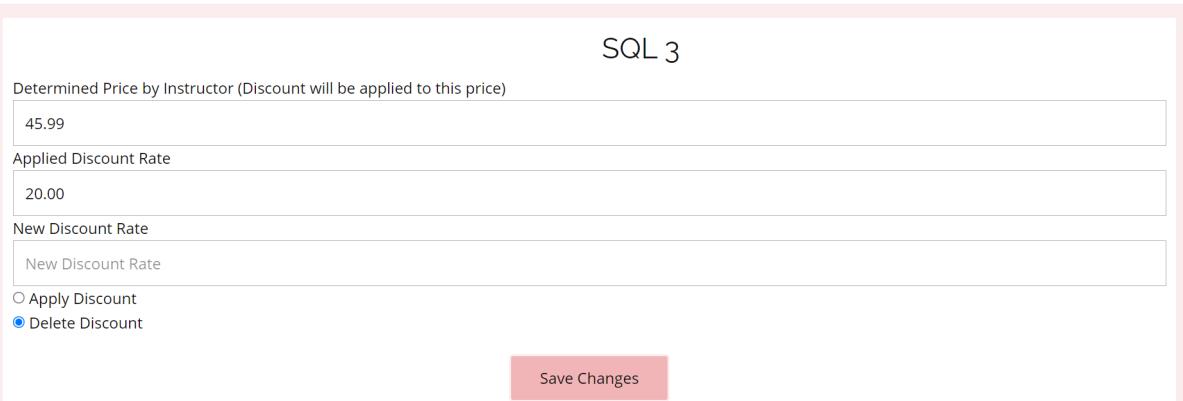


Course Name	Price	Category	Level	Instructor	
Music Theory for Beginners	30.99₺ 24.79₺	Music	Beginner	Irem Tekin	VIEW STATISTICS
Introduction to Badminton	12.99₺ 10.39₺	Sports	Beginner	Irem Tekin	VIEW STATISTICS OFFER DISCOUNT
SQL 3	45.99₺ 36.79₺	Computer Science	Advanced	Irem Tekin	VIEW STATISTICS OFFER DISCOUNT
SQL 1	22.00₺	Computer Science	Beginner	Irem Tekin	VIEW STATISTICS
SQL 2	25.00₺	Computer Science	Intermediate	Irem Tekin	VIEW STATISTICS

Fig. 53., 54. and 55. Apply Discount

6.3.4 Discount Management - Delete Discount

Admin can also delete an already applied discount by selecting delete discount and select changes. The change in the price of the course is displayed on the main page.



SQL 3

Determined Price by Instructor (Discount will be applied to this price)

Applied Discount Rate

New Discount Rate

Apply Discount
 Delete Discount

[Save Changes](#)



Introduction to Badminton	12.99₺ 10.39₺	Sports	Beginner	Irem Tekin	VIEW STATISTICS	OFFER DISCOUNT
SQL 1	22.00₺	Computer Science	Beginner	Irem Tekin	VIEW STATISTICS	
SQL 2	25.00₺	Computer Science	Intermediate	Irem Tekin	VIEW STATISTICS	
SQL 3	45.99₺	Computer Science	Advanced	Irem Tekin	VIEW STATISTICS	OFFER DISCOUNT
Immanuel Kant	67.99₺	Philosophy	Intermediate	Irem Tekin	VIEW STATISTICS	
Linear Algebra for Beginners	45.89₺	Math	Beginner	Irem Tekin	VIEW STATISTICS	OFFER DISCOUNT

Fig. 56. and 57. Delete Discount

6.3.5 Manage Student Complaints

Students can complain about courses, in their complaints can ask for a refund or not. Admin can view all the complaints by clicking **Student Complaints button** in the header. If a student asks for a refund, **refund button** is displayed next to the student complaint. If student doesn't ask for a refund, **noted button** is displayed next to **ignore button**.

6.3.5.1 Ignore Student Complaint

Admin can ignore a complaint by clicking **ignore button**. When admin ignores the complaint the complaint is deleted.

http://localhost:63342/code/src/admin_student_complaints.php

Course Name	Student Name	Complaint Subject	Complaint	Date	
Music Theory for Beginners	Gamze Guliter 2	Course could not satisfy my ex	Apparently I have some other type of expectations from a course (you know ;)	2021-05-17	IGNORE REFUND
SQL 3	Gamze Guliter 2	I hate this course	I hate it	2021-05-17	IGNORE REFUND
SQL 3	Gamze Guliter 2	Video Fast Mode Problem	123	2021-05-17	IGNORE NOTED
SQL 3	Gamze Guliter 2	complaint1	wd	2021-05-17	IGNORE NOTED

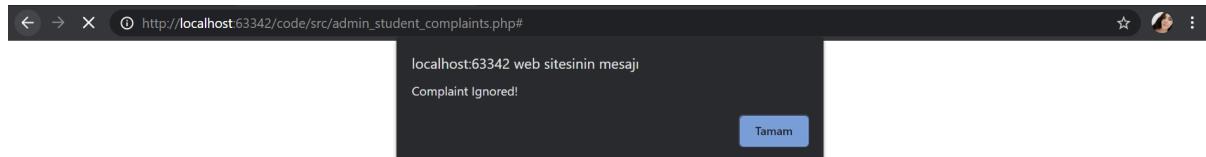


Fig. 58. and 59. Ignore Student Complaint

http://localhost:63342/code/src/admin_student_complaints.php#

Course Name	Student Name	Complaint Subject	Complaint	Date	
SQL 3	Gamze Guliter 2	i hate this course	i hate it	2021-05-17	IGNORE REFUND
SQL 3	Gamze Guliter 2	Video Fast Mode Problem	123	2021-05-17	IGNORE NOTED
SQL 3	Gamze Guliter 2	complaint1	wd	2021-05-17	IGNORE NOTED
SQL 3	Gamze Guliter 2	sdfg	qwefg	2021-05-17	IGNORE REFUND

Fig. 60. After ignore Student Complaint

6.3.5.2 Note Student Complaint

Admin can use **noted button** if admin doesn't want to ignore a complaint which doesn't include a refund request.

http://localhost:63342/code/src/admin_student_complaints.php#

Course Name	Student Name	Complaint Subject	Complaint	Date	
SQL 3	Gamze Guliter 2	i hate this course	i hate it	2021-05-17	IGNORE REFUND
SQL 3	Gamze Guliter 2	Video Fast Mode Problem	123	2021-05-17	IGNORE NOTED
SQL 3	Gamze Guliter 2	complaint1	wd	2021-05-17	IGNORE NOTED
SQL 3	Gamze Guliter 2	sdfg	qwefg	2021-05-17	IGNORE REFUND



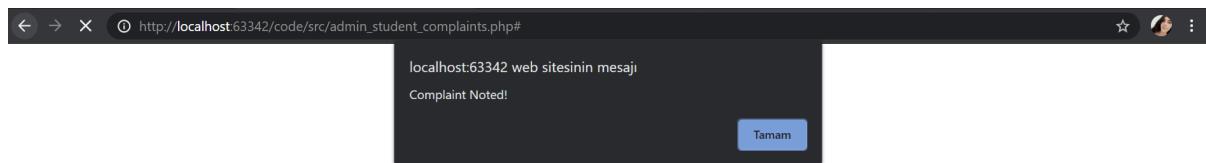


Fig. 61. and 62. Note Student Complaint

A screenshot of the Sapientia application interface. The title bar says "Sapientia". The navigation menu includes "Add Admin", "Student Complaints" (which is underlined), "Instructor Complaints", "Main Page", and "Logout". The main content area is titled "Student Complaints" and contains a table with three rows of student complaints. The table columns are: Course Name, Student Name, Complaint Subject, Complaint, and Date. The first row has values: SQL 3, Gamze Guliter 2, i hate this course, i hate it, 2021-05-17. The second row has values: SQL 3, Gamze Guliter 2, complaint1, wd, 2021-05-17. The third row has values: SQL 3, Gamze Guliter 2, sdfg, qwefg, 2021-05-17. To the right of each row are two buttons: "IGNORE" and "REFUND".

Course Name	Student Name	Complaint Subject	Complaint	Date		
SQL 3	Gamze Guliter 2	i hate this course	i hate it	2021-05-17	IGNORE	REFUND
SQL 3	Gamze Guliter 2	complaint1	wd	2021-05-17	IGNORE	NOTED
SQL 3	Gamze Guliter 2	sdfg	qwefg	2021-05-17	IGNORE	REFUND

Fig. 63. After Note Student Complaint

6.3.5.3 Refund Student

http://localhost:63342/code/src/admin_student_complaints.php

Add Admin Student Complaints Instructor Complaints Main Page Logout

Student Complaints

Course Name	Student Name	Complaint Subject	Complaint	Date	
SQL 3	Gamze Guliter 2	i hate this course	i hate it	2021-05-17	<button>IGNORE</button> <button>REFUND</button>
SQL 3	Gamze Guliter 2	complaint1	wd	2021-05-17	<button>IGNORE</button> <button>NOTED</button>
SQL 3	Gamze Guliter 2	sdfg	qwefg	2021-05-17	<button>IGNORE</button> <button>REFUND</button>

http://localhost:63342/code/src/admin_student_complaints.php#

localhost:63342 web sitesinin mesajı
Complaint accepted, Refund Approved!

Tamam

Fig. 64. and 65. Refund Student

http://localhost:63342/code/src/admin_student_complaints.php#

Add Admin Student Complaints Instructor Complaints Main Page Logout

Student Complaints

Course Name	Student Name	Complaint Subject	Complaint	Date	
SQL 3	Gamze Guliter 2	complaint1	wd	2021-05-17	<button>IGNORE</button> <button>NOTED</button>
SQL 3	Gamze Guliter 2	sdfg	qwefg	2021-05-17	<button>IGNORE</button> <button>REFUND</button>

Fig. 66. After Refund

6.3.6 Manage Instructor Complaints

Instructors can complain about their courses to admin. Admin can ignore or note the complaints.

The screenshot shows a web application titled "Sapientia". The main menu includes "Add Admin", "Student Complaints", "Instructor Complaints", "Main Page", and "Logout". The "Instructor Complaints" section displays two rows of data:

Course Name	Instructor Name	Complaint Subject	Complaint	Date	Action
Music Theory for Beginners	Irem Tekin	Discount	How can i open my course for discount allow?	2021-05-17	IGNORE NOTED
Introduction to Living with a Cat	Irem Tekin	I cannot upload video	I cannot upload video with size 2 mb	2021-05-17	IGNORE NOTED

Red arrows point to the "IGNORE" and "NOTED" buttons for each row.

Fig. 67. Ignore and Note the Complaints

The screenshot shows the same web application and menu as Fig. 67. The "Instructor Complaints" section now displays the same two rows of data, but the "Action" column is empty, indicating that the complaints have been processed.

Course Name	Instructor Name	Complaint Subject	Complaint	Date	Action
Music Theory for Beginners	Irem Tekin	Discount	How can i open my course for discount allow?	2021-05-17	
Introduction to Living with a Cat	Irem Tekin	I cannot upload video	I cannot upload video with size 2 mb	2021-05-17	

Fig. 68. After Ignore and Note the Complaints

6.3.7 Add Admin

An admin can add other admins by clicking the **add admin button** in the header and giving information about a new admin.

ONLINE LEARNING

Sapientia

Add Admin

Merve
merve@merve
...

Add admin

Fig. 69. Add Admin

	AID	name	e_mail	password
1	1	omer	omer@omer	1234
2	3	Merve	merve@merve	123

Fig. 70. Admin Table

7. Website

The link of GitHub page of the project is the following:

https://github.com/Sapientia-Course-Platform/CS353_Project

The link of Reports Website is the following:

https://sapientia-course-platform.github.io/CS353_Project/

8. Appendix

8.1. SQL Create Table Code

```
use omer_unlusoy;

DROP TABLE IF EXISTS Take_Quiz_Question;
DROP TABLE IF EXISTS Take_Lecture;
DROP TABLE IF EXISTS Take_Quiz;
DROP TABLE IF EXISTS Zoom_Session;
DROP TABLE IF EXISTS Discount;
DROP TABLE IF EXISTS Evaluates_Complaint_Entry_Student;
DROP TABLE IF EXISTS Evaluates_Complaint_Entry_Instructor;
DROP TABLE IF EXISTS Teaches;
DROP TABLE IF EXISTS Enrolls;
DROP TABLE IF EXISTS Wishlist;
DROP TABLE IF EXISTS Certificate;
DROP TABLE IF EXISTS QnA_Entry_Student;
DROP TABLE IF EXISTS QnA_Entry_Instructor;
DROP TABLE IF EXISTS Complaint_Entry_Student;
DROP TABLE IF EXISTS Complaint_Entry_Instructor;
DROP TABLE IF EXISTS Course_Announcement;
DROP TABLE IF EXISTS Note;
DROP TABLE IF EXISTS Quiz_Question;
DROP TABLE IF EXISTS Quiz;
DROP TABLE IF EXISTS Lecture;
DROP TABLE IF EXISTS Section;
DROP TABLE IF EXISTS Course;

DROP TABLE IF EXISTS Student;
DROP TABLE IF EXISTS Instructor;
DROP TABLE IF EXISTS Admin;

/*
General Assumptions:
ID          -> INT AUTO_INCREMENT
referencing ID -> INT NOT NULL
short string -> VARCHAR(30)
mid string   -> VARCHAR(50)
long string   -> VARCHAR(70)
description  -> VARCHAR(300)
money        -> NUMERIC(12, 2) OR NUMERIC(8, 2)
%            -> NUMERIC(5, 2)
date         -> DATE
URL          -> VARCHAR(1024)

Special Cases:
membership_type -> ENUM('BRZ', 'SLV', 'GLD')
refund_request  -> BOOLEAN
duration        -> TIME
lecture_content -> BLOB NOT NULL
membership_type -> ENUM('BRZ', 'SLV', 'GLD') NOT NULL DEFAULT 'BRZ'
```

```

answer          -> ENUM('choice1', 'choice2', 'choice3') NOT NULL DEFAULT
'choice1'

Constraints:
Student-Course cannot be in Wishlist, Enrolls, and Certificate all together.
adds Wishlist -> Enrolls (remove from Wishlist) -> get Certificate (add
Certificate not remove Enrolls)
Lecture and Quiz content_num

Future:
ENGINE = InnoDB; can be added

 */

/*
Assumptions:
*/
CREATE TABLE Student(
    SID           INT AUTO_INCREMENT,
    name          VARCHAR(50) NOT NULL,
    e_mail         VARCHAR(50) NOT NULL UNIQUE,
    password       VARCHAR(30) NOT NULL,
    membership_type ENUM('BRZ', 'SLV', 'GLD') NOT NULL
DEFAULT 'BRZ',
    PRIMARY KEY (SID)
);

/* Assumptions:
include functions [course_count(), current_income()]
*/
CREATE TABLE Instructor(
    IID           INT AUTO_INCREMENT,
    name          VARCHAR(50) NOT NULL,
    e_mail         VARCHAR(50) NOT NULL UNIQUE,
    password       VARCHAR(30) NOT NULL,
    PRIMARY KEY (IID)
);

CREATE TABLE Admin(
    AID           INT AUTO_INCREMENT,
    name          VARCHAR(50) NOT NULL,
    e_mail         VARCHAR(50) NOT NULL UNIQUE,
    password       VARCHAR(30) NOT NULL,
    salary         NUMERIC(12, 2) NOT NULL DEFAULT 0.00,
    PRIMARY KEY (AID)
);

/* Assumptions:
include functions [rating(), student_count(), lecture_count()]
constraints for category and level
0 <= cost <= 999,999.99
discount_allowed = True OR False
lecture_completed_threshold = %70
0 <= certificate_threshold <= 100.00
attention: FOREIGN KEY (creator_IID) REFERENCES Instructor(IID) ON DELETE
NO ACTION ON UPDATE CASCADE
*/

```

```

        UNIQUE (creator_IID, course_name)
*/
CREATE TABLE Course(
    CID                      INT AUTO_INCREMENT,
    IID                      INT NOT NULL,
    course_name              VARCHAR(70) NOT NULL,
    description              VARCHAR(300),
    rating                   NUMERIC(5, 2) DEFAULT
0.00,
    category                 VARCHAR(30),
    level                    VARCHAR(30),
    cost                     NUMERIC(8, 2) NOT NULL
DEFAULT 0.00,
    discount_allowed         BOOLEAN,
    quiz_threshold           NUMERIC(5, 2),
    PRIMARY KEY (CID),
    FOREIGN KEY (IID) REFERENCES Instructor(IID) ON DELETE
NO ACTION ON UPDATE CASCADE,
    UNIQUE (IID, course_name)
);

CREATE TABLE Section(
    CID                      INT NOT NULL,
    section                  INT NOT NULL,
    title                    VARCHAR(300),
    PRIMARY KEY (CID, section),
    FOREIGN KEY (CID) REFERENCES Course(CID) ON DELETE
CASCADE ON UPDATE CASCADE,
    INDEX (section)
);

/* Assumptions:
lecture_content is big data
lecture_content is blob (not recommended, try to store videos in file
directories and hold URLs)
Attention: UNIQUE (CID, title)
*/
CREATE TABLE Lecture(
    CID                      INT NOT NULL,
    section                  INT NOT NULL,
    content_num               INT NOT NULL,
    title                    VARCHAR(300) NOT NULL,
    lecture_content           VARCHAR(300),
    description              VARCHAR(300),
    PRIMARY KEY (CID, section, content_num),
    FOREIGN KEY (CID, section) REFERENCES Section(CID,
section) ON DELETE CASCADE ON UPDATE CASCADE,
    UNIQUE (CID, title),
    INDEX (section),
    INDEX (content_num)
);

CREATE TABLE Quiz(
    CID                      INT NOT NULL,
    section                  INT NOT NULL,
    content_num               INT NOT NULL,
    IID                      INT NOT NULL,

```

```

        title          VARCHAR(300) NOT NULL,
        PRIMARY KEY (CID, section, content_num),
        FOREIGN KEY (CID) REFERENCES Course(CID) ON DELETE
CASCADE ON UPDATE CASCADE,
        FOREIGN KEY (section) REFERENCES Section(section) ON
DELETE CASCADE ON UPDATE CASCADE,
        FOREIGN KEY (IID) REFERENCES Instructor(IID) ON DELETE
NO ACTION ON UPDATE CASCADE,
        UNIQUE (CID, title),
        INDEX (CID),
        INDEX (section),
        INDEX (content_num)
);

/*
Attention: answer ENUM('choice1', 'choice2', 'choice3') NOT NULL DEFAULT
'choice1'
answer = 1 => answer = choice1
*/
CREATE TABLE Quiz_Question(
    CID           INT NOT NULL,
    section       INT NOT NULL,
    content_num   INT NOT NULL,
    question_num  INT NOT NULL,
    question_text VARCHAR(300) NOT NULL,
    choice1       VARCHAR(50) NOT NULL,
    choice2       VARCHAR(50),
    choice3       VARCHAR(50),
    answer        ENUM('choice1', 'choice2',
'choice3') NOT NULL DEFAULT 'choice1',
    PRIMARY KEY (CID, section, content_num,
question_num),
    FOREIGN KEY (CID, section, content_num)
REFERENCES Quiz(CID, section, content_num) ON DELETE CASCADE ON UPDATE
CASCADE
);

CREATE TABLE Note(
    Note_ID        INT AUTO_INCREMENT,
    SID            INT NOT NULL,
    section        INT NOT NULL,
    content_num    INT NOT NULL,
    text           VARCHAR(300) NOT NULL,
    PRIMARY KEY (Note_ID),
    FOREIGN KEY (SID) REFERENCES Student(SID) ON DELETE
CASCADE ON UPDATE CASCADE
);

CREATE TABLE Course_Announcement(
    Ann_ID         INT AUTO_INCREMENT,
    IID            INT NOT NULL,
    CID            INT NOT NULL,
    date           DATE,
    subject_name   VARCHAR(30) NOT NULL,
    text           VARCHAR(300) NOT
NULL,
);

```

```

        PRIMARY KEY (Ann_ID),
        FOREIGN KEY (IID) REFERENCES
Instructor(IID) ON DELETE CASCADE ON UPDATE CASCADE,
        FOREIGN KEY (CID) REFERENCES Course(CID)
ON DELETE CASCADE ON UPDATE CASCADE
);

CREATE TABLE QnA_Entry_Student(
    QnA_ID_S          INT AUTO_INCREMENT,
    SID                INT NOT NULL,
    CID                INT NOT NULL,
    subject_name       VARCHAR(30) NOT NULL,
    text               VARCHAR(300) NOT NULL,
    date               DATE,
    PRIMARY KEY (QnA_ID_S),
    FOREIGN KEY (SID) REFERENCES Student(SID)
ON DELETE NO ACTION ON UPDATE CASCADE,
    FOREIGN KEY (CID) REFERENCES Course(CID) ON
DELETE CASCADE ON UPDATE CASCADE
);

CREATE TABLE QnA_Entry_Instructor(
    QnA_ID           INT AUTO_INCREMENT,
    QnA_ID_S         INT NOT NULL,
    IID              INT NOT NULL,
    CID              INT NOT NULL,
    subject_name     VARCHAR(30) NOT
NULL,
    text             VARCHAR(300) NOT
NULL,
    date             DATE,
    PRIMARY KEY(QnA_ID),
    FOREIGN KEY (QnA_ID_S) REFERENCES
QnA_Entry_Student(QnA_ID_S) ON DELETE NO ACTION ON UPDATE CASCADE,
    FOREIGN KEY (IID) REFERENCES
Instructor(IID) ON DELETE NO ACTION ON UPDATE CASCADE,
    FOREIGN KEY (CID) REFERENCES Course(CID)
ON DELETE CASCADE ON UPDATE CASCADE
);

CREATE TABLE Complaint_Entry_Student(
    Complaint_ID      INT
AUTO_INCREMENT,
    SID                INT NOT NULL,
    CID                INT NOT NULL,
    subject_title      VARCHAR(30) NOT
NULL,
    text               VARCHAR(300) NOT
NULL,
    date               DATE,
    refund_request     BOOLEAN,
    PRIMARY KEY (Complaint_ID),
    FOREIGN KEY (SID) REFERENCES
Student(SID) ON DELETE CASCADE ON UPDATE CASCADE,
    FOREIGN KEY (CID) REFERENCES
Course(CID) ON DELETE CASCADE ON UPDATE CASCADE
);

```

```

CREATE TABLE Complaint_Entry_Instructor(
    Complaint_ID          INT,
    AUTO_INCREMENT,
    IID                   INT NOT NULL,
    CID                   INT NOT NULL,
    subject_name          VARCHAR(30),
    NOT NULL,
    text                  VARCHAR(300),
    NOT NULL,
    date                 DATE,
    PRIMARY KEY (Complaint_ID),
    FOREIGN KEY (IID) REFERENCES Instructor(IID) ON DELETE CASCADE ON UPDATE CASCADE,
    FOREIGN KEY (CID) REFERENCES Course(CID) ON DELETE CASCADE ON UPDATE CASCADE
) ;

/*
Binary relationship between Student and Course
Attributes: date
*/
CREATE TABLE Wishlist(
    SID                  INT NOT NULL,
    CID                  INT NOT NULL,
    PRIMARY KEY (SID, CID),
    FOREIGN KEY (SID) REFERENCES Student(SID) ON DELETE CASCADE ON UPDATE CASCADE,
    FOREIGN KEY (CID) REFERENCES Course(CID) ON DELETE CASCADE ON UPDATE CASCADE
) ;

/*
Binary relationship between Student and Course
Attributes: progress, rating
*/
CREATE TABLE Enrolls(
    SID                  INT NOT NULL,
    CID                  INT NOT NULL,
    progress             NUMERIC(5, 2) DEFAULT 0.00,
    date                DATE,
    PRIMARY KEY (SID, CID),
    FOREIGN KEY (SID) REFERENCES Student(SID) ON DELETE CASCADE ON UPDATE CASCADE,
    FOREIGN KEY (CID) REFERENCES Course(CID) ON DELETE CASCADE ON UPDATE CASCADE
) ;

/*
Binary relationship between Student and Course
Attributes: date
*/
CREATE TABLE Certificate(
    SID                  INT NOT NULL,
    CID                  INT NOT NULL,
    certificate_text     VARCHAR(300),
    NOT NULL
)

```

```

rating          NUMERIC(5, 2),
comment         VARCHAR(300),
date            DATE,
PRIMARY KEY (SID, CID),
FOREIGN KEY (SID) REFERENCES Student(SID) ON
DELETE CASCADE ON UPDATE CASCADE,
FOREIGN KEY (CID) REFERENCES Course(CID) ON
DELETE CASCADE ON UPDATE CASCADE
);

/*
Binary relationship between Instructor and Course
*/
CREATE TABLE Teaches(
    IID             INT NOT NULL,
    CID             INT NOT NULL,
    PRIMARY KEY (IID, CID),
    FOREIGN KEY (IID) REFERENCES Instructor(IID) ON
DELETE CASCADE ON UPDATE CASCADE,
    FOREIGN KEY (CID) REFERENCES Course(CID) ON DELETE
CASCADE ON UPDATE CASCADE
);

/*
Binary relationship between Admin and Complaint_Entry
Attention: FOREIGN KEY (AID) REFERENCES Admin(AID) ON DELETE NO ACTION ON
UPDATE CASCADE
*/
CREATE TABLE Evaluates_Complaint_Entry_Student(
    AID            INT NOT
NULL,
    Complaint_ID   INT NOT
NULL,
    response        VARCHAR(300),
    isRefund        BOOLEAN,
    PRIMARY KEY (AID,
Complaint_ID),
    FOREIGN KEY (AID)
REFERENCES Admin(AID) ON DELETE NO ACTION ON UPDATE CASCADE,
    FOREIGN KEY (Complaint_ID)
REFERENCES Complaint_Entry_Student(Complaint_ID) ON DELETE CASCADE ON UPDATE
CASCADE
);

/*
Binary relationship between Admin and Complaint_Entry
Attention: FOREIGN KEY (AID) REFERENCES Admin(AID) ON DELETE NO ACTION ON
UPDATE CASCADE
*/
CREATE TABLE Evaluates_Complaint_Entry_Instructor(
    AID            INT
NOT NULL,
    Complaint_ID   INT
NOT NULL,
    response        VARCHAR(300),
    PRIMARY KEY (AID,
Complaint_ID),
    FOREIGN KEY (AID)
REFERENCES Admin(AID) ON DELETE NO ACTION ON UPDATE CASCADE
);

```

```

VARCHAR(300),
PRIMARY KEY (AID,
Complaint_ID),
FOREIGN KEY (AID)
REFERENCES Admin(AID) ON DELETE NO ACTION ON UPDATE CASCADE,
FOREIGN KEY
(Complaint_ID) REFERENCES Complaint_Entry_Instructor(Complaint_ID) ON DELETE
CASCADE ON UPDATE CASCADE
);

/*
Binary relationship between Admin and Course
Attributes: rate (e.g. %30)
Attention: FOREIGN KEY (AID) REFERENCES Admin(AID) ON DELETE NO ACTION ON
UPDATE CASCADE
*/
CREATE TABLE Discount(
    AID          INT NOT NULL,
    CID          INT NOT NULL,
    rate         NUMERIC(5, 2),
    PRIMARY KEY (AID, CID),
    FOREIGN KEY (AID) REFERENCES Admin(AID) ON DELETE NO
ACTION ON UPDATE CASCADE,
    FOREIGN KEY (CID) REFERENCES Course(CID) ON DELETE
CASCADE ON UPDATE CASCADE
);

/*
Binary relationship between Instructor and Course
Attributes: invite_link, start_date
Attention: it uses its own key to allow instructors to create several links
*/
CREATE TABLE Zoom_Session(
    Session_ID   INT AUTO_INCREMENT,
    IID          INT NOT NULL,
    CID          INT NOT NULL,
    invite_link  VARCHAR(1024),
    start_date   TIMESTAMP,
    PRIMARY KEY (Session_ID),
    FOREIGN KEY (IID) REFERENCES Instructor(IID) ON
DELETE CASCADE ON UPDATE CASCADE,
    FOREIGN KEY (CID) REFERENCES Course(CID) ON
DELETE CASCADE ON UPDATE CASCADE
);

/*
Binary relationship between Student and Quiz
Attributes: grade (updated according to Student_Take_Quiz_Question)
Attention: its key is Student(SID) U Quiz(CID, QID)
*/
CREATE TABLE Take_Quiz(
    SID          INT NOT NULL,
    CID          INT NOT NULL,
    section      INT NOT NULL,
    content_num  INT NOT NULL,
    grade        NUMERIC(5, 2),
    PRIMARY KEY (SID, CID, section, content_num),

```

```

        FOREIGN KEY (SID) REFERENCES Student(SID) ON DELETE
CASCADE ON UPDATE CASCADE
);

/*
Binary relationship between Student and Quiz_Question
keeps students' answers to each quiz question
Attributes: isTrue
*/
CREATE TABLE Take_Quiz_Question(
    SID           INT NOT NULL,
    CID           INT NOT NULL,
    section       INT NOT NULL,
    content_num   INT NOT NULL,
    question_num  INT NOT NULL,
    answer        ENUM('choice1',
'choice2', 'choice3') NOT NULL,
    isTrue         BOOLEAN,
    PRIMARY KEY (SID, CID, section,
content_num, question_num),
    FOREIGN KEY (SID) REFERENCES Student(SID)
ON DELETE CASCADE ON UPDATE CASCADE
);

/*
Binary relationship between Student and Lecture
keeps students' completion to each lecture (lecture_completed_threshold in
lecture (e.g. %70 for each lecture))
Attributes: isCompleted
*/
CREATE TABLE Take_Lecture(
    SID           INT NOT NULL,
    CID           INT NOT NULL,
    section       INT NOT NULL,
    content_num   INT NOT NULL,
    isCompleted   BOOLEAN,
    PRIMARY KEY (SID, CID, section, content_num),
    FOREIGN KEY (SID) REFERENCES Student(SID) ON
DELETE CASCADE ON UPDATE CASCADE,
    FOREIGN KEY (CID, section, content_num)
REFERENCES Lecture(CID, section, content_num) ON DELETE CASCADE ON UPDATE
CASCADE
);

/*
Example Insertions
*/
INSERT INTO Admin (name, e_mail, password, salary)
VALUES ('omer', 'omer@omer', '1234', 1000000000.00);

```