MIS 307

SYSTEM ANALYSIS AND DESIGN COURSE

PROJECT

ARİF ESEN 17030411054

HAMZA YAVUZ 17030411056

İHSAN TARIK AKKOYUNLU 17030411057

ÖMER FARUK KIZILGEDİK 17030411015

CONTENTS

SYSTEM REQUEST FORM	
Request Date: 01.12.2020	
Project Title: MyCourse (Online Learning Platform)	
Project sponsor	MyCourse Company
Business need	This system have purpose to service as an online learning platform to the students in Turkey. Student will be access the our platform and they can watch the online lessons.
Business requirements	Using the web browsers students and personals can access the our system. The specific functionality that the system should have includes the following; ▪ Students can take private lessons ▪ Teacher can stream live lessons. ▪ Students can register the system. ▪ Admin can control the system ▪ Admin can create new students and personals.
Business value	▪ After the create this system, we hope the save about 540.000 Tl each year.(physical expenses) ▪ Also, we can register more students than normal (double) for only first year. In the total only students extra benefit about 500.000 Tl ▪ In the total we hope about 940.000 Tl save money thanks to the this system.
Specific issues or Constraints	▪ Security issues should be addressed as customers will enter the credit card information into the system. ▪ The current paper based informations should be transferred to new system. ▪ Need trainers to teach the new system to the teachers. ▪ Security issues should be addressed as customers will enter the credit card information into the system.

ANALYSIS STRATEGY

Organization gets decided to change their concept and they need completely new information system. Organization wants to be more efficiently They want to make use of pandemic conditions. In the face-to-face education, MyCourse organization did not need the information system like this but in the pandemic conditions, they have to use modern and improved information and education system.

We choose Business Process Reengineering (BPR) as Business Process Model (BPM) strategy because this decision is really risky, and their information system will radically redesigned. When we are planning, we thought that business processes will restructuring and existing system will change, so we have to choose BPR. This redesigned will take time, but we believe that new information system will really be useful for organization.

Organizations should be innovative in order to keep up with the competition and they should implement new technologies in their information system. Their new information system includes these new technologies.

There are so many benefits of BPR.BPR plays a major role in organizational performance improvement in terms of cost, delivery, quality, employee productivity, etc. It also helps to these. Ease business processes and systems, Organizations easily adapt to changing times, improve company profitability and sustain competitive advantage, Boost employee productivity and the last one Increase customer satisfaction.

PROJECT PLANNING PHASE

The Courses, like many institutions, have been closed due to the global pandemic and they continue their existence with distance education. One of our new customers is “MyCourse”. This course is new in this industry and wants to bring an innovative dimension to distance education in order to turn this global crisis into an opportunity for itself. We arranged a meeting with our technology consultant to help this customer. Thus, we will both provide solutions to our customers' needs and indicate the limits of our own company.

In the project, we aim to bring students and teachers together more comfortably and quickly in the virtual environment during the pandemic process. These are students who need additional support while preparing for the university entrance exam but cannot attend private educational institutions. One of our goals is to establish an online classroom system.

A certain fee will be charged when the student is registered. We can also get ads that can be useful to the site and increase profits.

AGILE DEVELOPMENT

Since we need a short time to set up the system, we preferred agile development. So, the cycles were short, and we quickly found a solution to the problem.

Estimating Project Duration Using Industry Standards

	PLANNING	ANALYSIS	DESIGN	IMPLEMENTATION
TYPICAL INDUSTRY STANDARDS FOR BUSINESS APPLICATIONS	%10	%20	%40	%30
ESTIMATES VASED IN ACTUAL FIGURES FOR FIRST STAGE OF SDLC	%20	%30	%30	%20

TASK ID	TASK NAME	ASSIGNED TO	ESTIMATED			ACTUAL		
			Duration	Start Date	Finish Date	Start Date	Finish Date	Duration Variance
1	Planning	Arif Hamza İhsan Ömer	15	01.12.2020	15.12.2020	14.12.2020	20.12.2020	6
1.1	Planning Table	Hamza	30	15.12.2020	18.01.2021	15.12.2020	20.01.2021	36
1.1	Work sharing	Arif Hamza İhsan Ömer	3	21.12.2020	24.12.2020	21.12.2020	23.12.2020	3
2	Analysis	Arif Hamza İhsan Ömer	15	25.12.2020	10.01.2020	28.12.2020	15.01.2021	18
2.1	Use case analysis	İhsan	5	28.12.2020	02.01.2021	28.12.2020	30.12.2020	2
2.2.1	Context diagram	Arif	6	30.12.2020	05.01.2021	01.01.2021	05.01.2021	5
2.2.2	DFD level 0	Arif	4	05.01.2021	09.01.2020	05.01.2021	10.01.2021	5
2.2.3	DFD level 1	Ömer	5	10.01.2021	16.01.2021	10.01.2020	15.01.2021	5
2.2.4	Physical DFD	Ömer	5	15.01.2021	20.01.2021	15.01.2021	17.01.2021	2
2.3	ER data model	İhsan	2	10.01.2021	12.01.2021	10.01.2021	11.01.2021	1
2.3.1	ERD and database model	Hamza	3	13.01.2021	16.01.2021	12.01.2021	14.01.2021	2
3	Design	Arif Hamza İhsan Ömer	6	14.01.2021	20.01.2021	14.01.2021	19.01.2021	5
3.1	Architecture design	Arif İhsan	3	14.01.2021	17.01.2021	14.01.2021	16.01.2021	2

		Ömer						
3.2	Software and hardware specifications	Hamza	3	15.01.2021	18.01.2021	16.01.2021	17.01.2021	1
3.3	User İnterface	Hamza	2	18.01.2021	20.01.2021	17.01.2021	18.01.2021	1
3.4	Structure chart	Arif	2	18.01.2021	20.01.2021	17.01.2021	19.01.2021	2
3.5	Implement the system	İhsan	1	19.01.2021	20.01.2021	18.01.2021	19.01.2021	1
3.6	Support the system	İhsan	1	19.01.2021	20.01.2021	18.01.2021	19.01.2021	1

GATHERING SYSTEM REQUIREMENTS

The Courses, like many institutions, have been closed due to the global pandemic and they continue their existence with distance education. One of our new customers is “MyCourse”. This course is new in this industry and wants to bring an innovative dimension to distance education in order to turn this global crisis into an opportunity for itself. We arranged a meeting with our technology consultant to help this customer. Thus, we will both provide solutions to our customers' needs and indicate the limits of our own company.

Important notes from the Interview

Interviewed: Kenan Memedow the manager of the Course and Vladimir Dokumov the Math Lecturer

Interviewer: İhsan Tarık Akkoyunlu the Technology consultant.

Purpose of the Interview:

- Understanding what system needs.
- Which options are the best for both of us?

Summary of the Interview:

Since the course is new in the industry, using the latest technologies makes the system to be created more effective. There are main requirements of system to be created:

- Information for management.
- A storage where teacher and student information and records are kept as a basic requirement.
- Another basic need is a handy user interface.
- A mobile application for future periods.
- A healthy web security.

Other important points:

- There must be minimum 6 months warranty.
- IT Support minimum first 6 months.
- Data security (personal data privacy).

The Requirements Definitions

We discussed the system requirements in order to provide the needs of our customer in the most accurate way by consulting our software developers. Deciding is very important as there are so many alternatives to use. Since storing data and the user interface are the most important demands of our customer, it is necessary to decide to build the system on these two requirements.

Functional Requirements

Course Management System Requirements

- The only authorization of management page.
- Database of students and personal records.
- Deleting record of personal or student.
- Creating new record of personal or student.

Student Management System Requirements

- Only authorization of manager or advisor.
- Creating new record of student.
- Deleting record of student.
- Request to manager of management issues.

Personal System Requirements

- Tracking Lessons.
- Creating new Live Lesson Sessions.
- Closing Sessions.
- Messaging on system.
- Students information.

Student System Requirements

- Tracking Lessons.
- Accessing live Lesson Sessions.
- Messaging on system.

Non-Functional Requirements

Security

- All operations need to be authenticated and audited via logger.
- Signed accounts need to request via cookies.

Operational

- User interface for both mobile and desktop.
- Effective live streamer for live lesson sessions.

Performance

- Caching should be server-side.
- Records do not need to be updated unless there is no creating or deleting request.
- Fetching type of student and lecturer should be lazy.

There is no constraint.

Use Cases

Use Case Name: Monitoring admin panel	ID: ADMIN-M-1	Priority: Highest
Actor: Admin		
Description: This use case is used for managing the system. This phase has kind of administration sceneries for manipulating whole system. Also, teachers use this panel to stream live lessons.		
Trigger: Logging the system from specific admin logon screen and adminpanel endpoint.		
Preconditions: • User must request to adminpanel endpoint. • User must be login from admin logon screen with admin name and password.		
Normal Courses: 1. This process begins with successful admin login. 2. The system monitors all system transactions from database logs and CRUD operations buttons. 2.1. Admin selects create to create new student or personal record on the database. 2.2. Admin selects read to read all information from specified record from database. 2.3. Admin selects update to update specified record from database. 2.4. Admin selects delete to delete specified record from database.		
Alternative Courses: 2.1.a. Admin selects creates already registered record. • System shows error dialog box. 2.2.a. Admin selects delete already deleted record. • System shows error dialog box.		
Postconditions: • System redirects to the adminpanel again after successful transaction. • System update whole database after each admin transactions.		
Exceptions: E1. System shows 403 Forbidden for unauthorized requests.		

Use Case Name: Admin Create Record	ID: ADMIN-CRUD-1	Priority: High
Actor: Admin		
Description: This use case is used for creating new student or personal record for the firm.		
Trigger: Admin select create-button on the admin panel.		
Preconditions: • User must request to adminpanel endpoint. • User must be login from admin logon screen with admin name and password. • Student must be registered on the system to send request.		
Normal Courses: 1. Admin selects student record to create. 1.1. System checks payment options. 1.2. System fills information box from temporal student database. 2. Admin selects personal record to create. 2.1. System asks for information to be created personal record. 2.2. Admin fills information steps. 3. Admin selects preview before the commit operation. 4. Admin selects save button. 5. System redirect to admin panel.		
Alternative Courses: 1.1.a. System shows dialog box if there is no payment option. 3.a Admin select save button. 1. System redirects to preview screen.		
Postconditions: • System redirects to the adminpanel again after successful transaction. • System update whole database after each admin transactions.		
Exceptions: E1. System shows 403 Forbidden for unauthorized requests.		

Use Case Name: Admin Delete Record	ID: ADMIN-CRUD-4	Priority: High
Actor: Admin		
Description: This use case is used for deleting student or personal record on the system.		
Trigger: Admin select delete-button on the admin panel.		
Preconditions: • User must request to adminpanel endpoint. • User must be login from admin logon screen with admin name and password.		
Normal Courses: 6. Admin selects student record to delete. 1.1. System checks all conditions. 1.2. System shows all logs about the record. 7. Admin selects personal record to delete 2.1. System asks for information to be delete personal record. 2.2. Admin fills information steps. 8. Admin selects preview before the commit operation. 9. Admin selects save button. 10. System redirect to admin panel.		
Alternative Courses: 2.2.a. Admin does not enter information. 1. System shows dialog box. 3.a Admin select save button. 1. System redirects to preview screen.		
Postconditions: • System redirects to the adminpanel again after successful transaction. • System update whole database after each admin transactions.		
Exceptions: E1. System shows 403 Forbidden for unauthorized requests.		

Use Case Name: Admin Read Record	ID: ADMIN-CRUD-3	Priority: High
Actor: Admin		
Description: This use case is used for updating registered profiles.		
Trigger: Admin selects update button.		
Preconditions: • User must request to adminpanel endpoint. • User must be login from admin logon screen with admin name and password. • User must list the records.		
Normal Courses: 11. Admin selects student record to update. 1.1. System lists all the students who are registered. 1.2. Admin clicks on any student record to get information from database. 1.3. Admin clicks update button on the record row. 1.4. System shows information form for updating the record. 12. Admin selects personal record to list. 2.1. System lists all personal records. 2.2. Admin clicks on any personal record to get information from database. 2.3. Admin clicks update button on the record row. 2.3. System shows information form for updating the record. 13. Admin selects preview. 14. Admin clicks save to commit transaction. 15. System redirect to admin panel.		
Alternative Courses: 1.1.a. Admin does not select 1. System shows dialog box. 4.a Admin select save button. 1. System redirects to preview screen.		
Postconditions: • System redirects to the adminpanel again after successful transaction. • System update whole database after each admin transactions.		
Exceptions: E1. System shows 403 Forbidden for unauthorized requests.		

Use Case Name: Admin Read Record	ID: ADMIN-CRUD-2	Priority: High
Actor: Admin		
Description: This use case is used for reading record or records which is registered system from database.		
Trigger: Admin selects list-button on the admin panel.		
Preconditions: • User must request to adminpanel endpoint. • User must be login from admin logon screen with admin name and password.		
Normal Courses: 16. Admin selects student record to list. 1.1. System lists all the students who are registered. 1.2. Admin clicks on any student record to get information from database. 17. Admin selects personal record to list. 2.1. System lists all personal records. 2.2. Admin clicks on any personal record to get information from database. 18. System shows other management operation buttons on each record in the list. 19. System redirect to admin panel.		
Alternative Courses: 1.1.a. Admin does not select. 1. System shows dialog box. 4.a Admin select save button. 1. System redirects to preview screen.		
Postconditions: • System redirects to the adminpanel again after successful transaction. • System update whole database after each admin transactions.		
Exceptions: E1. System shows 403 Forbidden for unauthorized requests.		

Use Case Name: Student registration	ID: STD-REG-1	Priority: High
Actor: Student		
Description: This use case is used for when a student wants to register to the system.		
Trigger: Student clicks the register button.		
Preconditions: • The user that wants to join the system must be a student. • The student should have an email or cell phone and number.		
Normal Courses: 1. System shows the form to get information of student. 2. The student enters the information. 3. System shows payment options and channel of payment section. 4. The student plans payment options and the money suspend on the bank system. 5. System store the options and notify the admin about payment and all information store on the temporal student information database.		
Alternative Courses: 4.a. Student does not select any options. 1. Payment options never disappear, and student must do selection. 5.an If admin does not create a new student the money sends back to bank account.		
Postconditions: • Student must activate the account via email or phone number.		
Exceptions: E1. System is not up to date 1. User gets server error. E2. Bank service is not active. 2. Payment options is not allowed and all safe.		

Use Case Name: Student Panel	ID: ST-1	Priority: High
Actor: Student		
Description: This use case is used for when the student login the system and monitoring panel.		
Trigger: User login the system and want to monitor the panel.		
Preconditions: • System must be active and up to date. • Student must be login the system.		
Normal Courses: 1. The student login the system. 2. Student access the panel. 3. System shows students stats. 4. System shows student information and setting. 5. Student check lessons and their times. 6. System contains other online teachers or student information. 7. System tracks activity on the panel.		
Alternative Courses: 4.a System notify user if there is an online lesson. 6.a System close session if there is no activity on the system for 1 hour.		
Postconditions: • Student should close the session after the activities.		
Exceptions: E1. System can be offline or closed. 1. Students get 500 error.		

Use Case Name: Access Live Lesson Stream	ID: ST-2	Priority: Medium
Actor: Student		
Description: This use case is used for when a student wants to join a teacher lesson streaming.		
Trigger: Student sends request to teacher lesson streamer.		
Preconditions: • Teacher must start a new session or there must be any live session. • Student must be logon on the system. • Student must not be on the another live stream.		
Normal Courses: 1. Student login the system and clicks join live stream. 1.1 Students selects lesson student wants to join. 2. System checks connection conditions. 3. System redirect student to stream. 4. System shows stream options on the session. 5. System logs students' activities and store it student logs temporal database. 6. Private Lesson (If student want, s/he can create private lesson session if teacher accepted) 7. System redirect to main student panel when stream is done.		
Alternative Courses: 1.1.a Student does not select lesson. 1. System pops up message box. 2.a. System does not establish stream and student. 1. System redirects to lessons screen.		
Postconditions: 1. System disconnect the stream server. 2. System checks attendants.		
Exceptions: E1. Teacher disconnects the server. 1. System throws 404 error. E2. System gets request overload and suspend all requests. 1. Student gets 500 internal server errors.		

Use Case Name: Streaming Live Lesson	ID: TCHR-1	Priority: High

Actor: Teacher

Description: This use case is used for when teacher should stream live lesson and publish the event on the system. Teachers has a different ID and they login the system on admin panel.

Trigger: Teacher clicks to stream the live lesson on the admin panel.

Preconditions:

- Teacher must be logon on the system in the admin panel.
- System must establish condition of stream on the server.

Normal Courses:

1. Teacher clicks to stream button on the admin panel.
2. System redirect to stream panel and shows stream options.
3. System shows attendants and teacher can see all students that joined the stream.
4. System shows student hardware conditions that allowed by the client.
5. Private lesson (Teacher can coach to students that want to private lesson)
6. Teacher closes the stream.

Alternative Courses:

2.a Stream server is not active.

1. System shows message box.

3.a. System allow different student to the stream.

1. Teacher warns user and kick user from the streaming.

Postconditions:

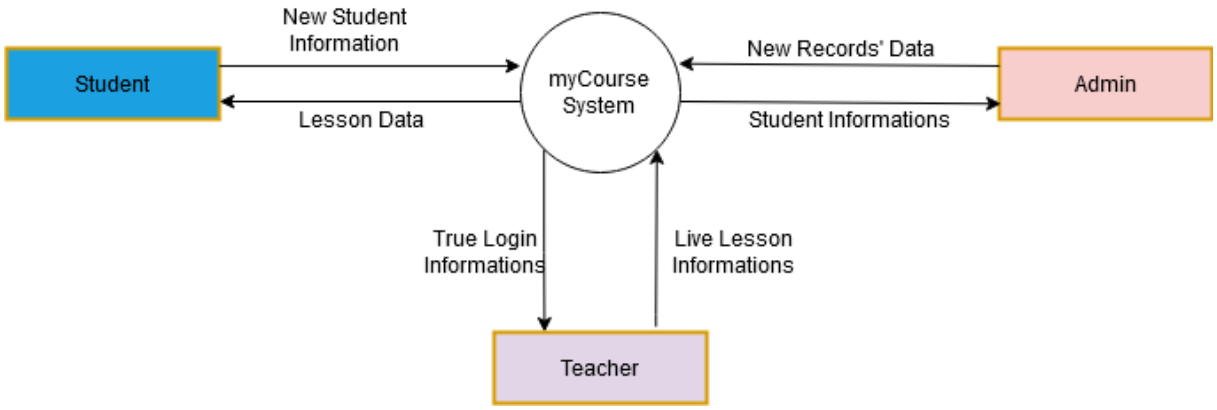
- Teacher should be close streaming after the lesson.

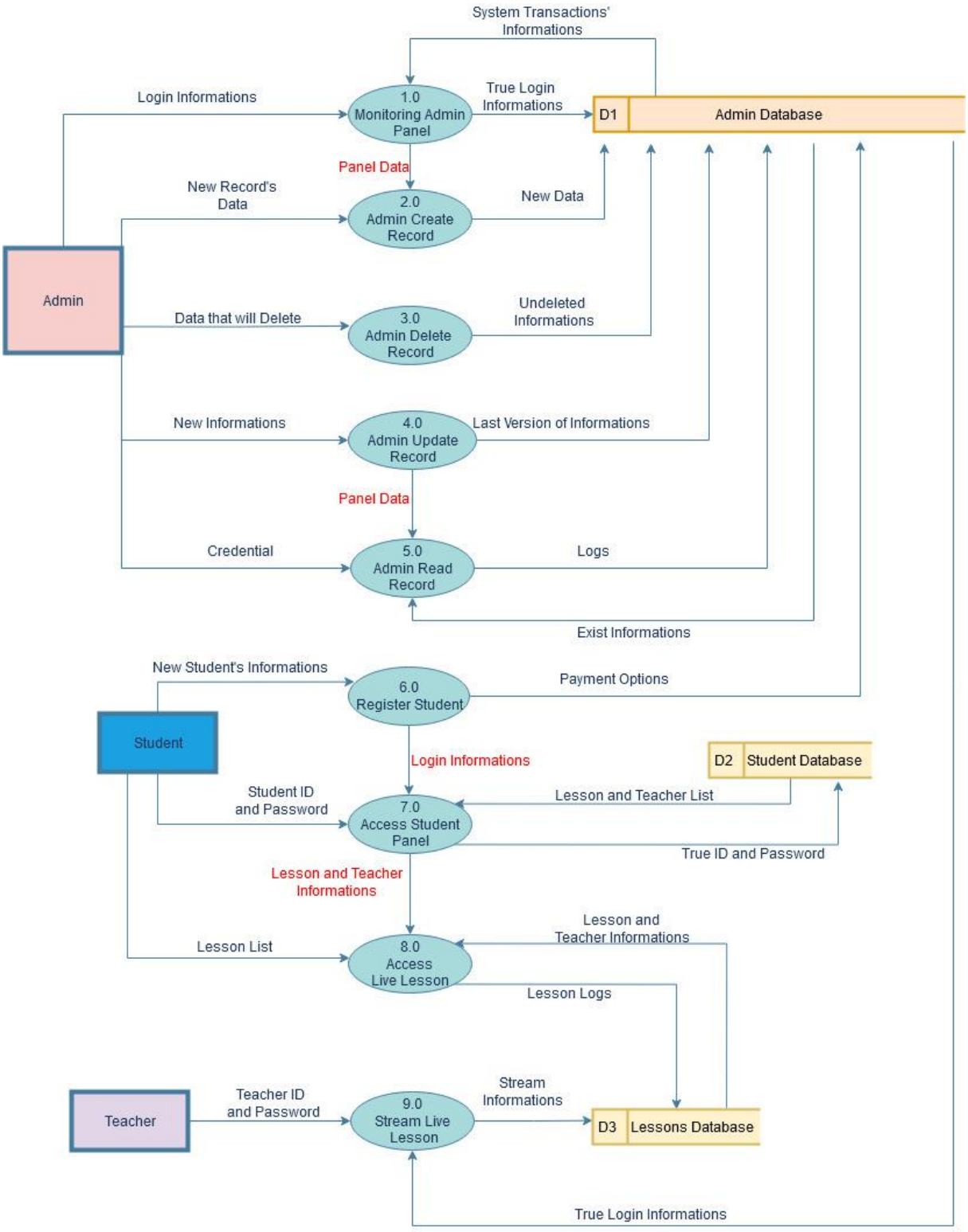
Exceptions:

E1. System disconnects the server.

1. System 500 internal server error.

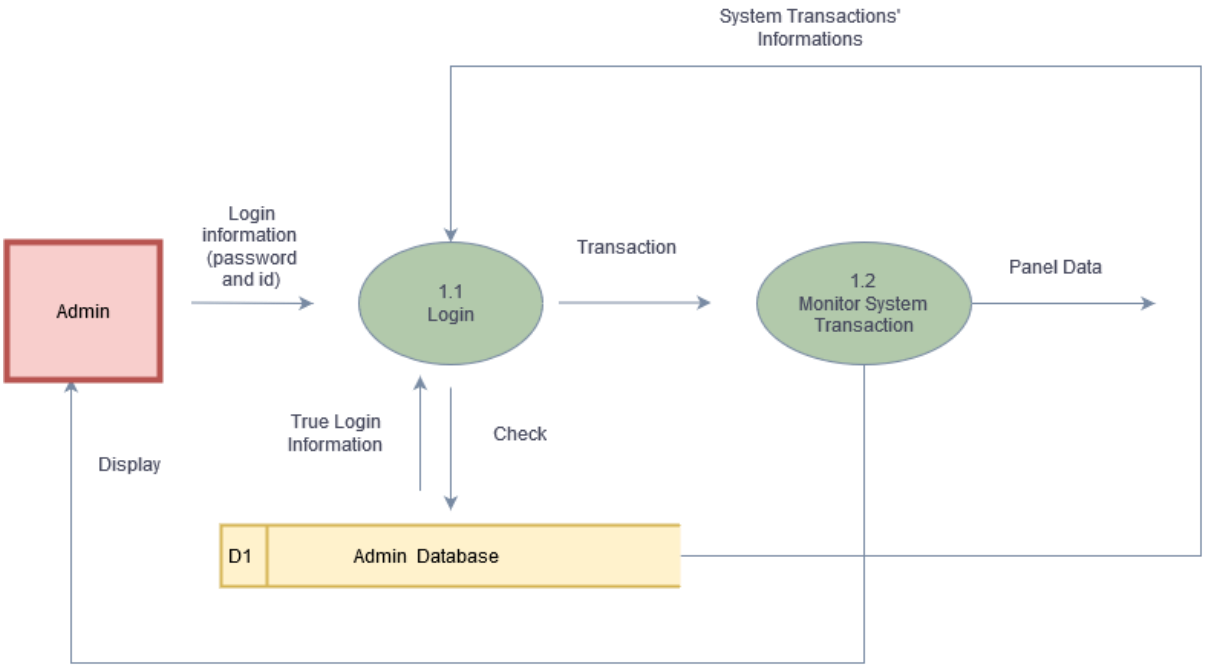
CONTEXT DIAGRAM & DFDs



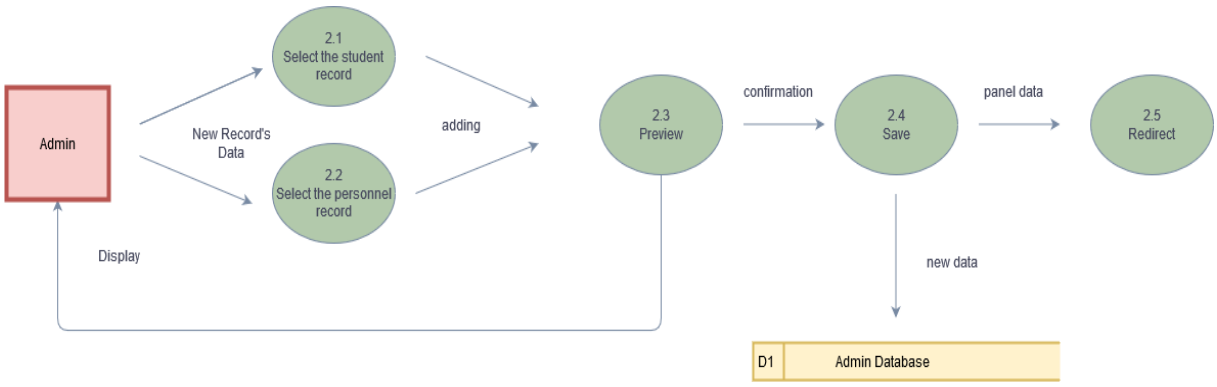

System Transactions' Informations
Login Informations
1.0 Monitoring Admin Panel
True Login Informations
D1 Admin Database
Panel Data
New Record's Data
2.0 Admin Create Record
New Data
Admin
Data that will Delete
3.0 Admin Delete Record
Undeleted Informations
New Informations
4.0 Admin Update Record
Last Version of Informations
Panel Data
Credential
5.0 Admin Read Record
Logs
Exist Informations
New Student's Informations
6.0 Register Student
Payment Options
Student
Login Informations
D2 Student Database
Student ID and Password
7.0 Access Student Panel
Lesson and Teacher List
True ID and Password
Lesson and Teacher Informations
Lesson and Teacher Informations
Lesson List
8.0 Access Live Lesson
Lesson Logs
Teacher
Teacher ID and Password
9.0 Stream Live Lesson
Stream Informations
D3 Lessons Database
True Login Informations

DFD Level-1

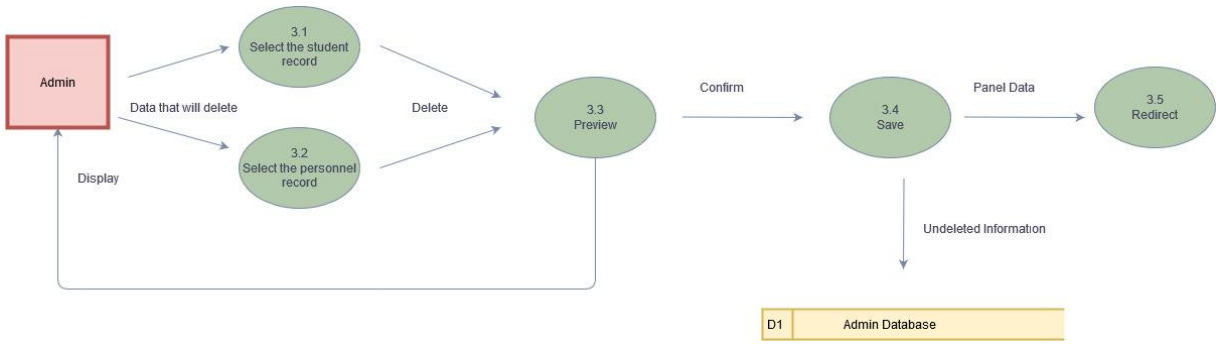
1.0 Monitoring Admin Panel



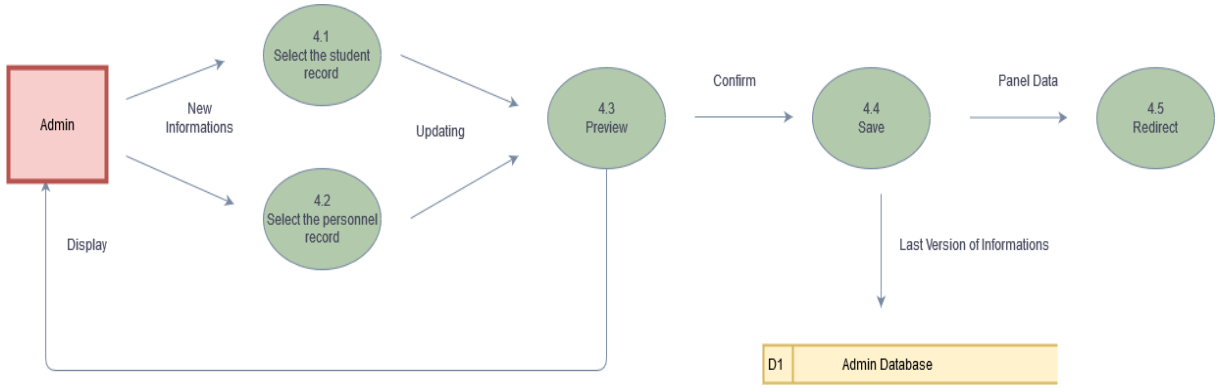
2.0 Admin Create Record



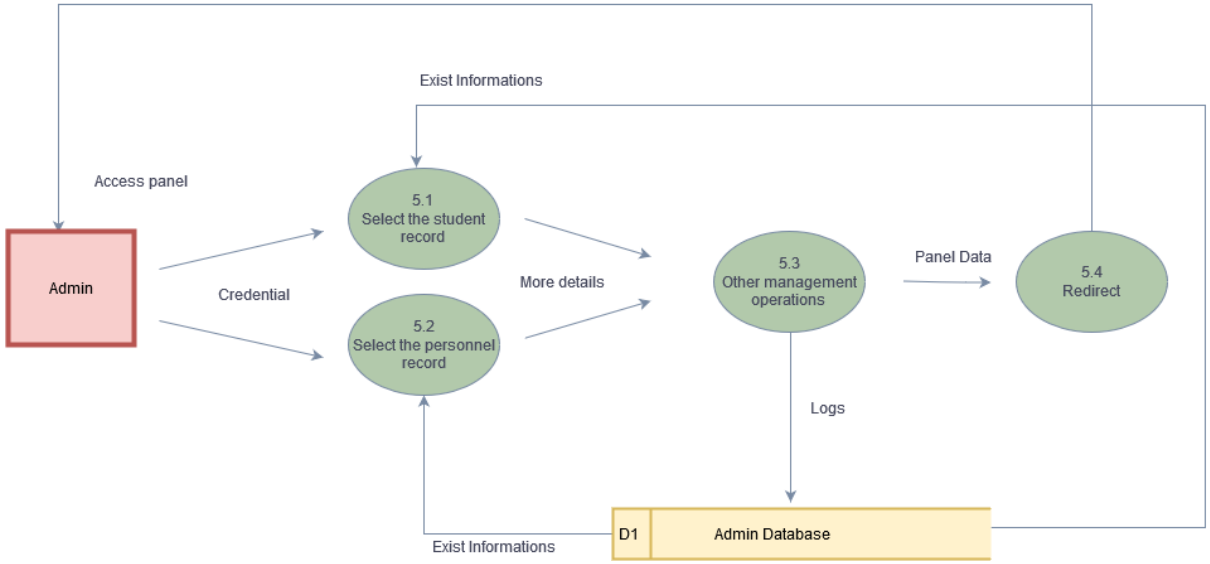
3.0 Admin Delete Record



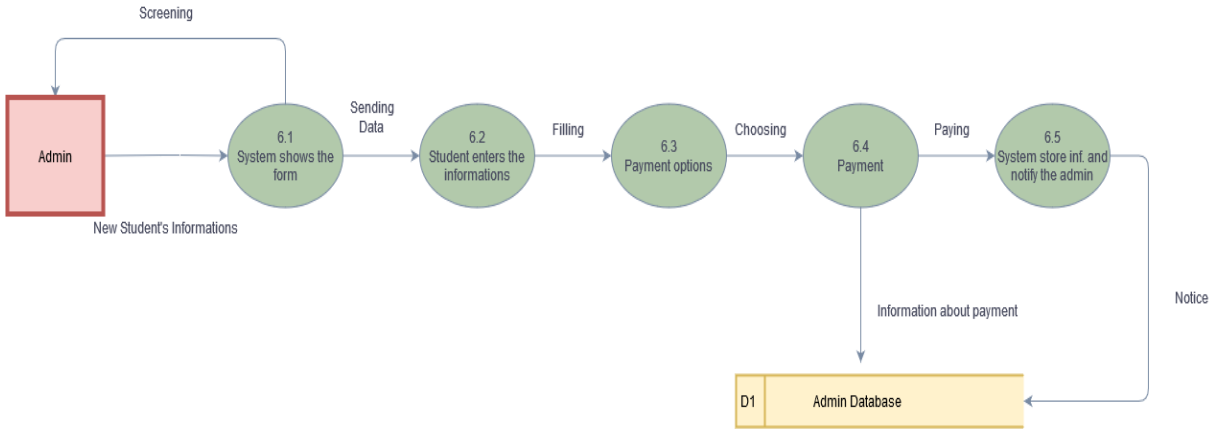
4.0 Admin Update Record



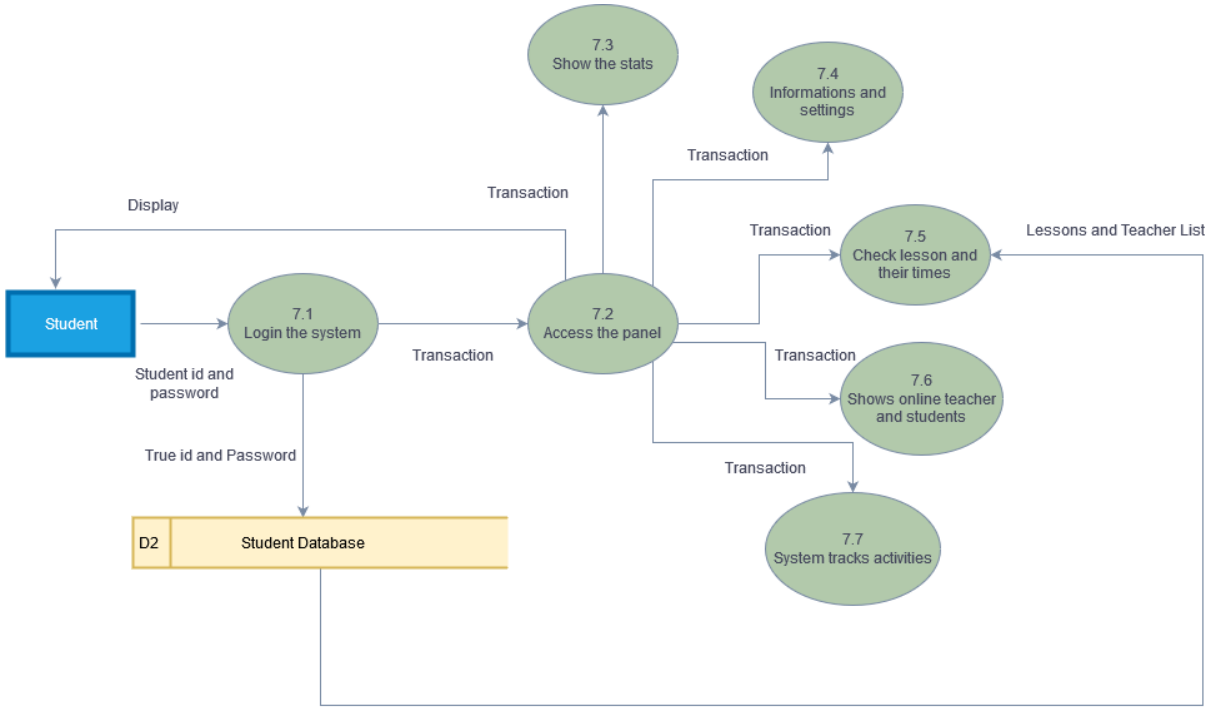
5.0 Admin Read Record



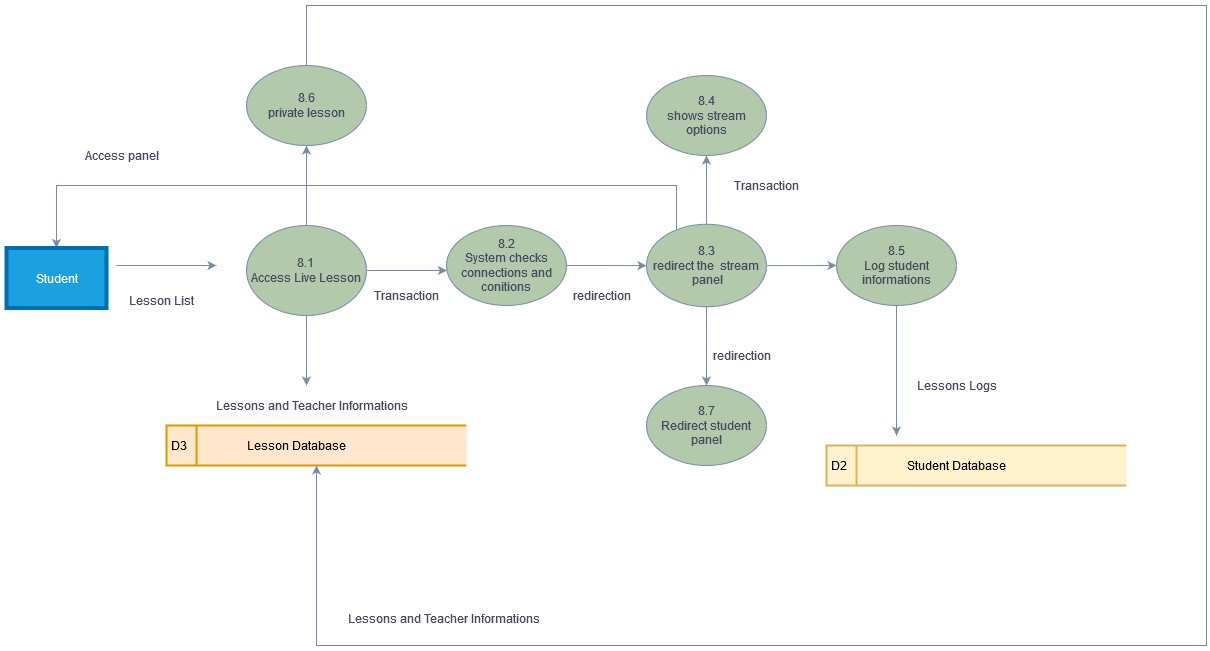
6.0 Register Student



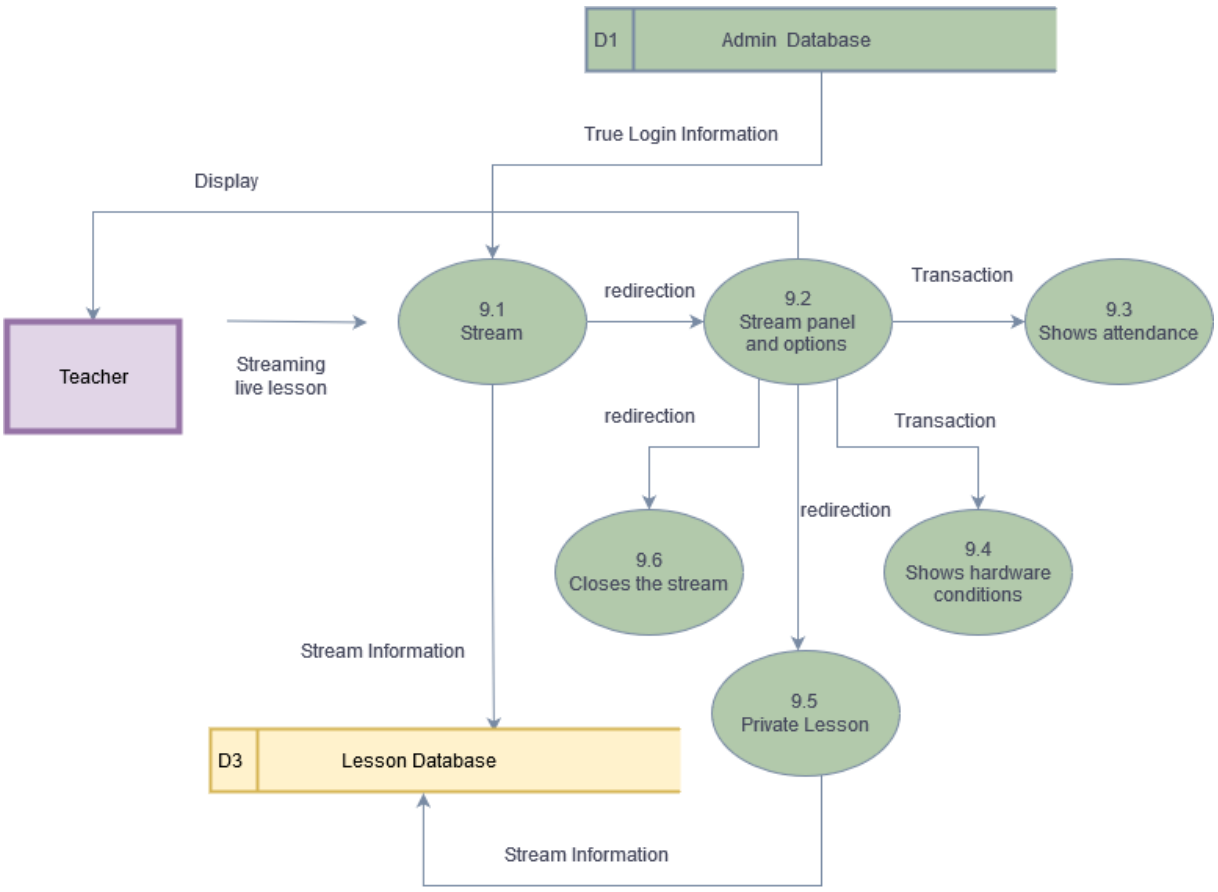
7.0 Access Student Panel



8.0 Access Live Lesson

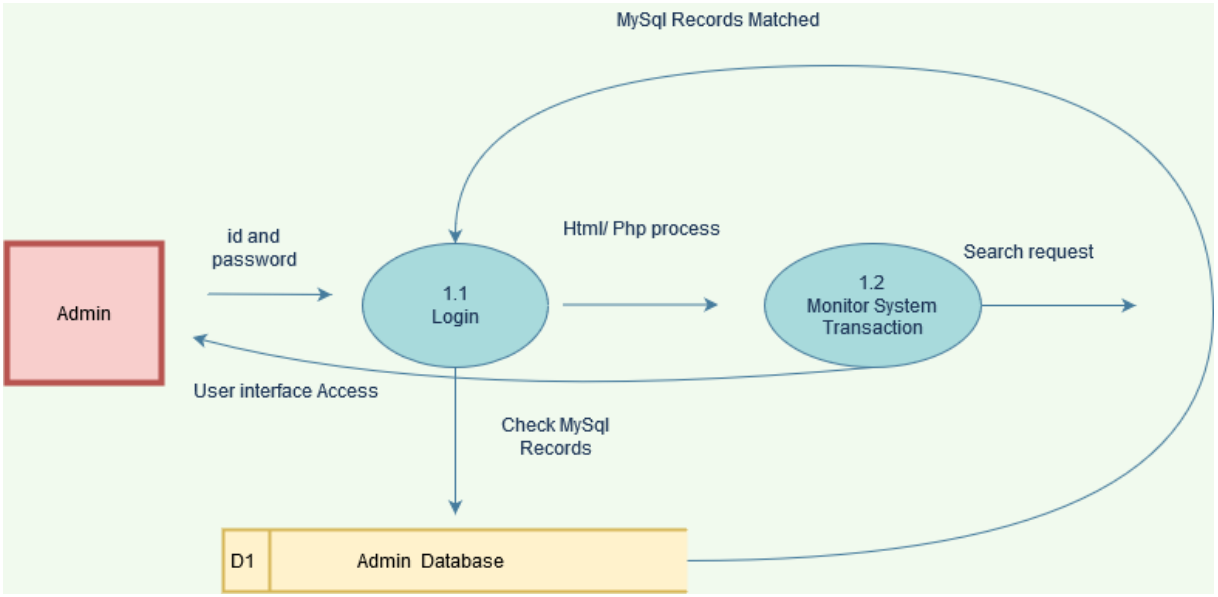


9.0 Stream Live Lesson

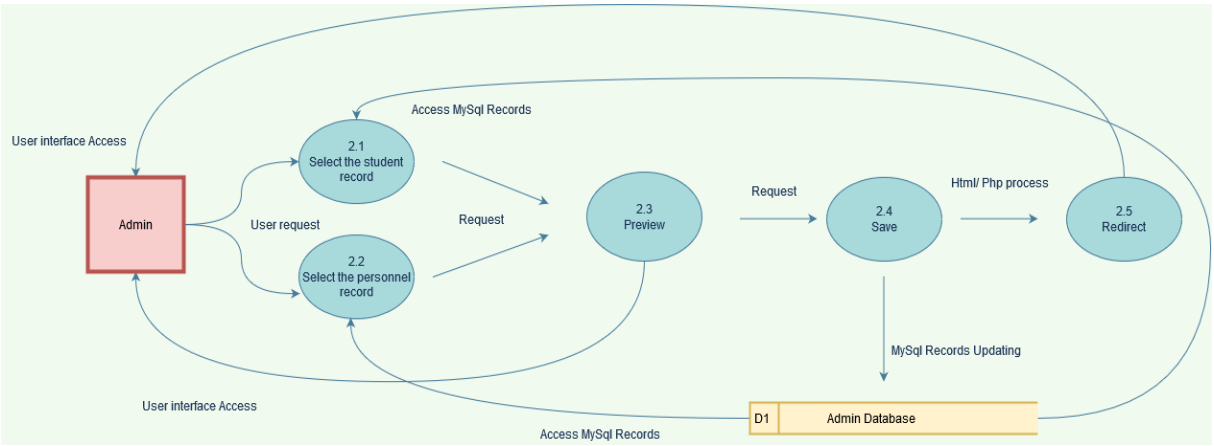


Physical DFD

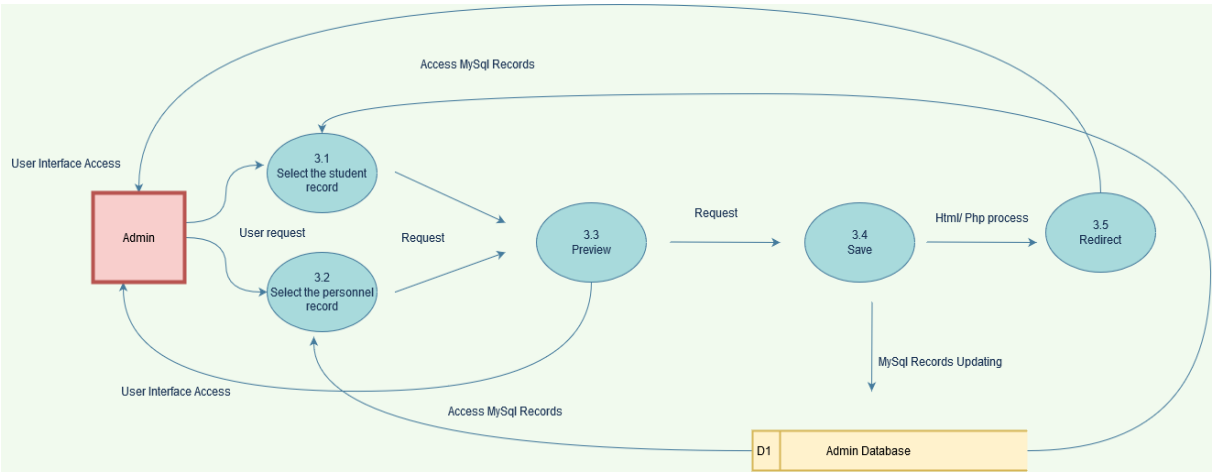
1.0 Monitoring Admin Panel



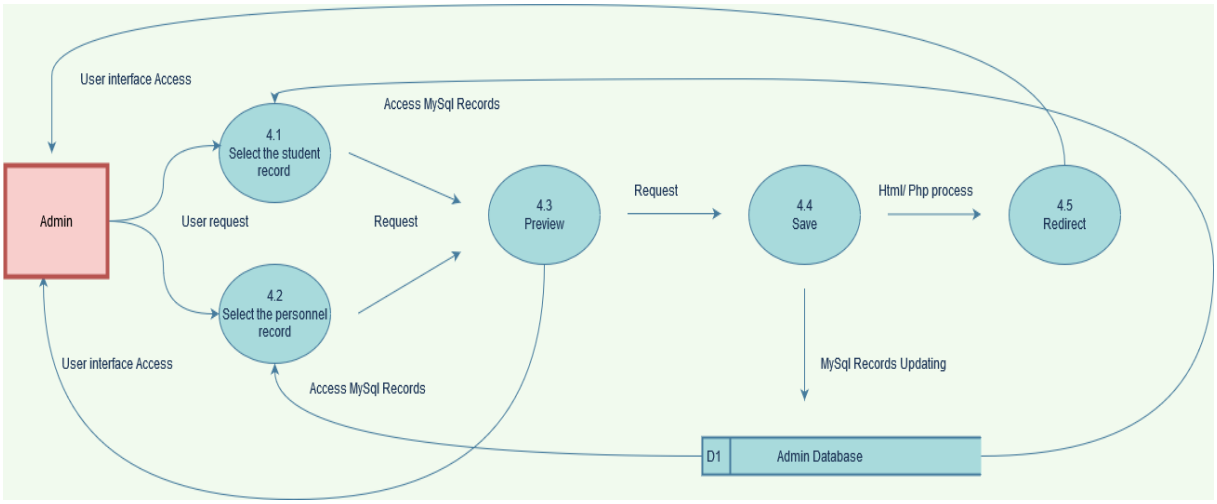
2.0 Admin Create Record



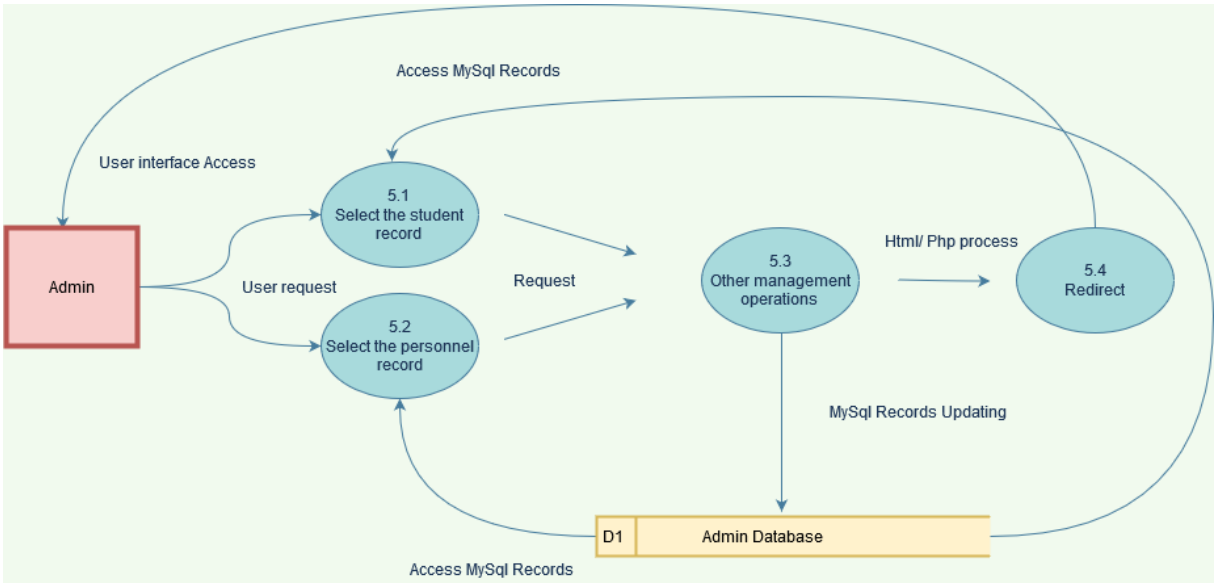
3.0 Admin Delete Record



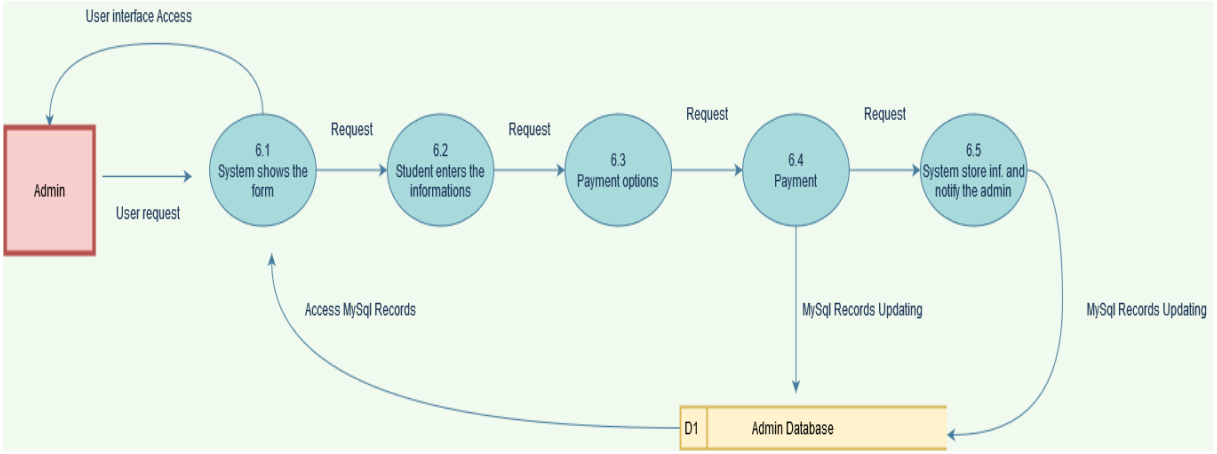
4.0 Admin Update Record



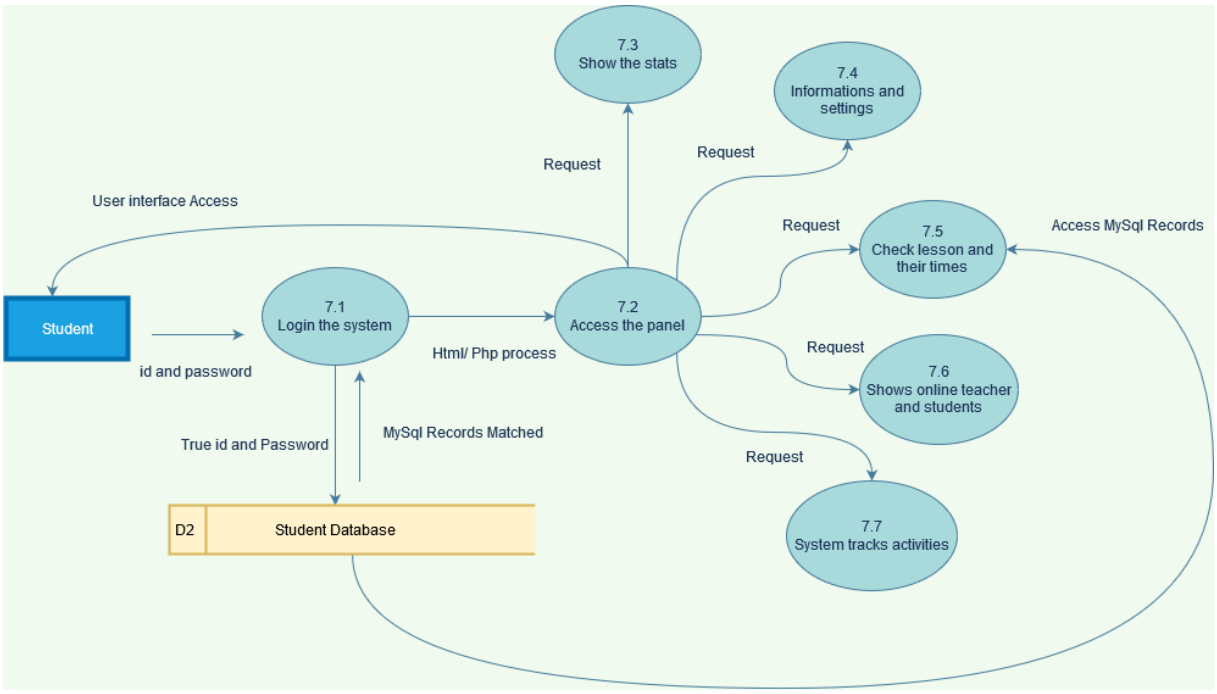
5. 0 Admin Read Record



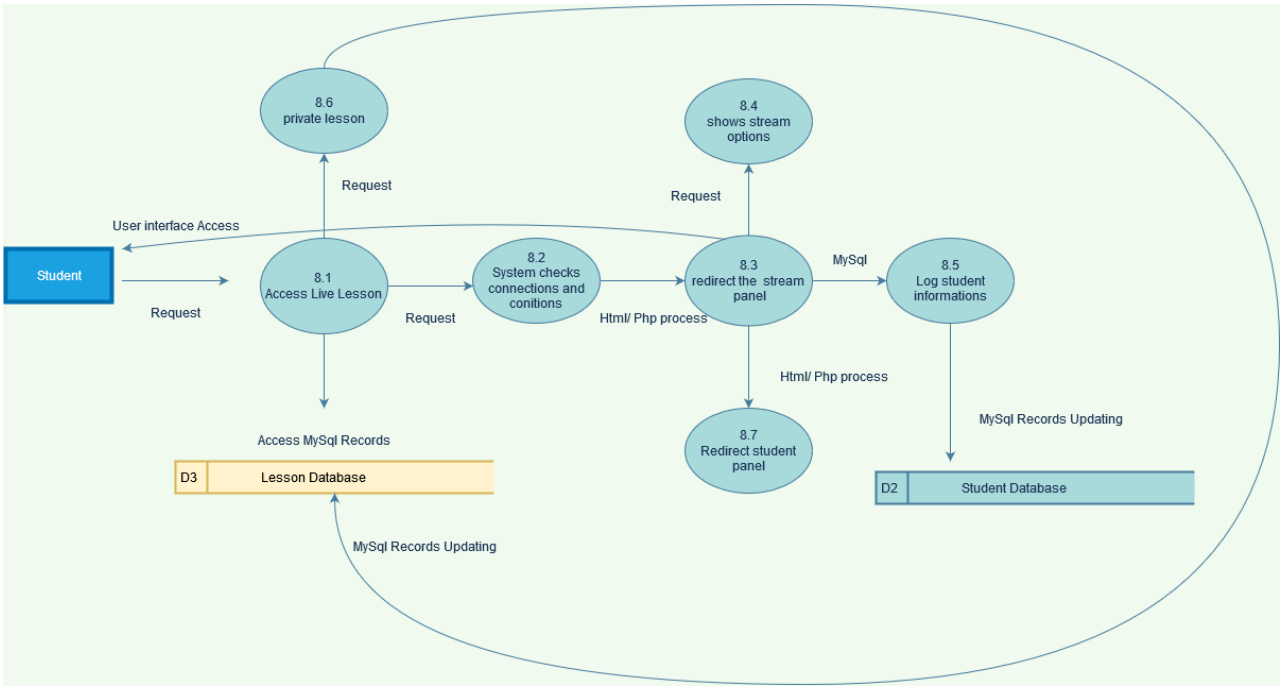
6.0 Register Student



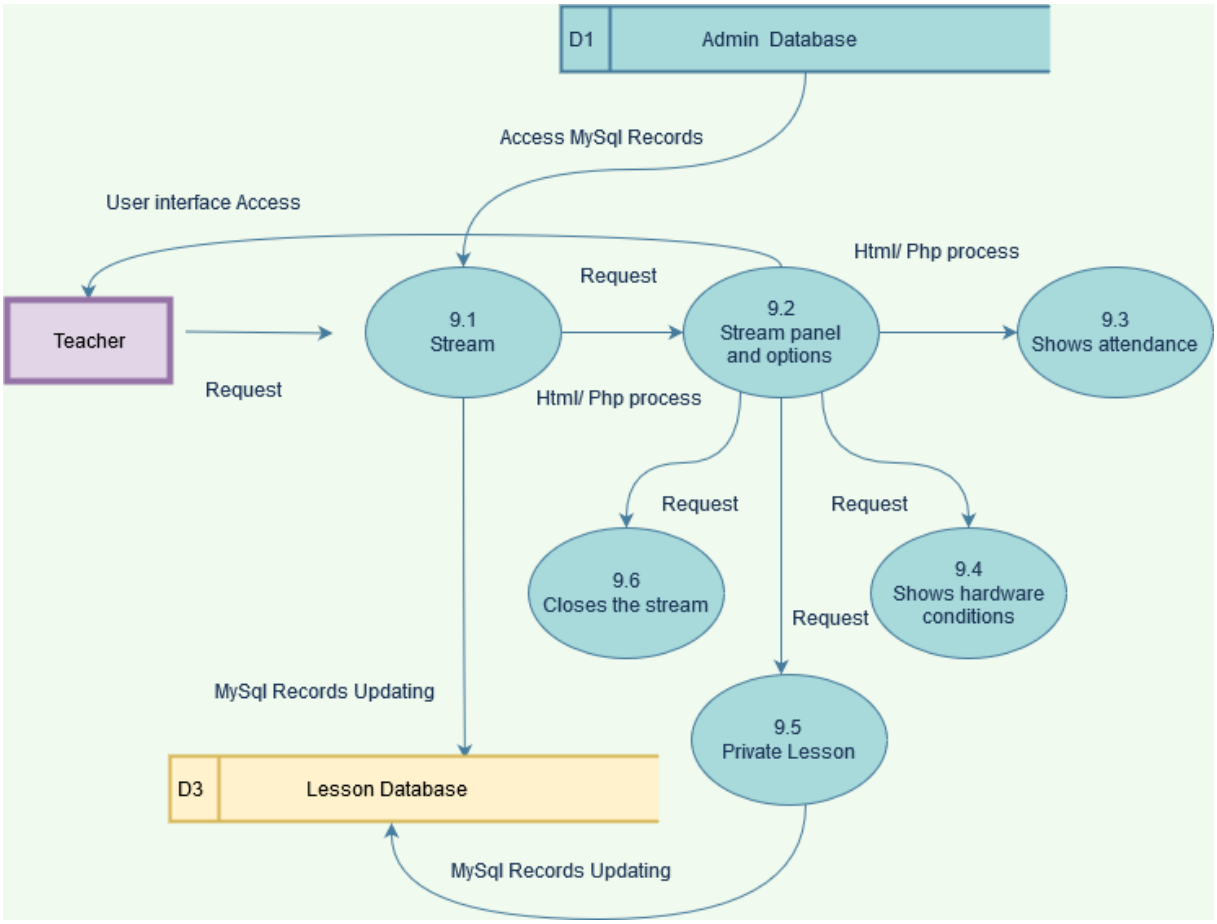
7.0 Access Student Panel



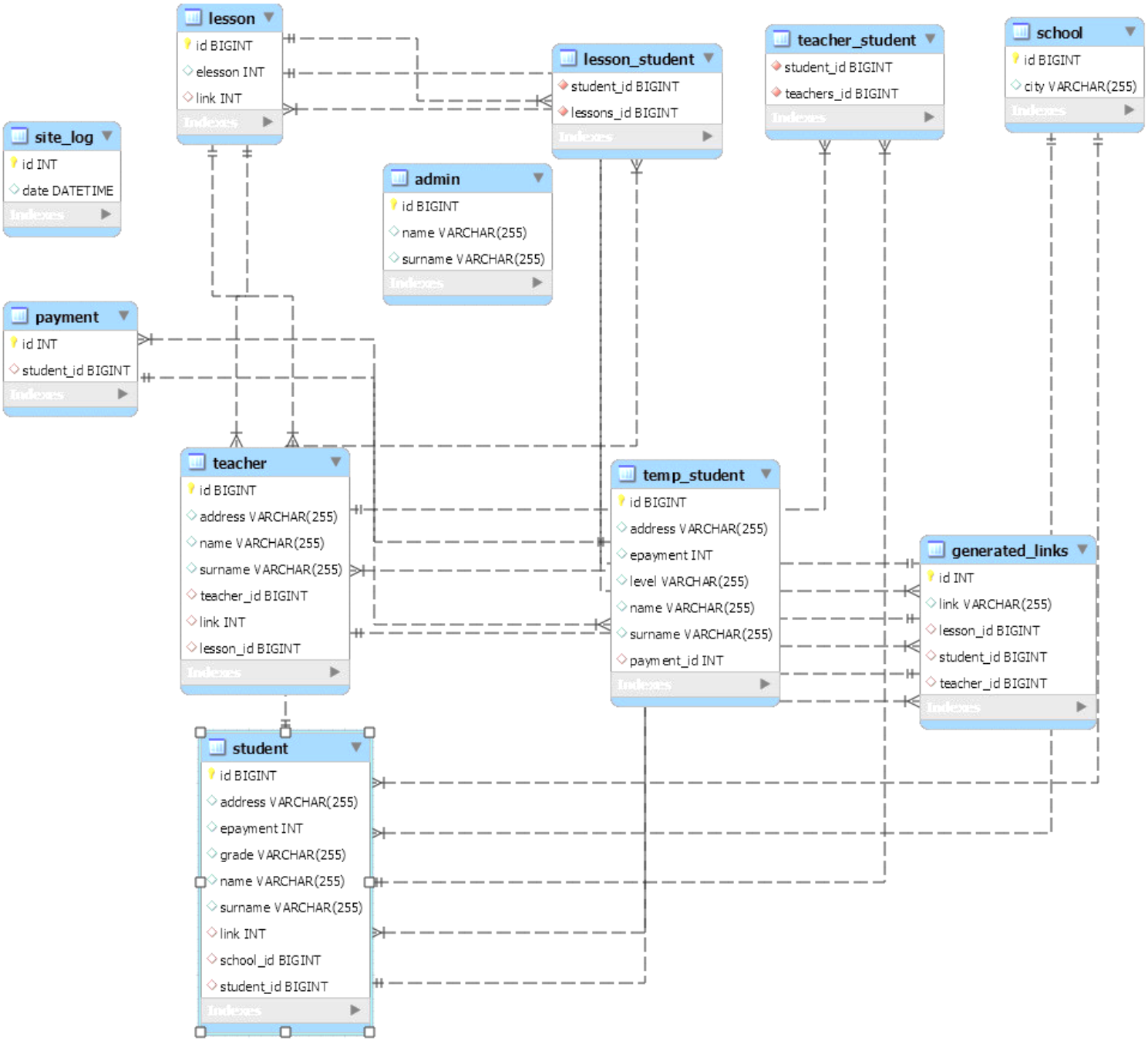
8.0 Access Live Lesson



9.0 Stream Live Lesson



ER


lesson
id BIGINT
elesson INT
link INT
Indexes
lesson_student
student_id BIGINT
lessons_id BIGINT
Indexes
teacher_student
student_id BIGINT
teachers_id BIGINT
Indexes
school
id BIGINT
city VARCHAR(255)
Indexes
site_log
id INT
date DATETIME
Indexes
admin
id BIGINT
name VARCHAR(255)
surname VARCHAR(255)
Indexes
payment
id INT
student_id BIGINT
Indexes
teacher
id BIGINT
address VARCHAR(255)
name VARCHAR(255)
surname VARCHAR(255)
teacher_id BIGINT
link INT
lesson_id BIGINT
Indexes
temp_student
id BIGINT
address VARCHAR(255)
epayment INT
level VARCHAR(255)
name VARCHAR(255)
surname VARCHAR(255)
payment_id INT
Indexes
generated_links
id INT
link VARCHAR(255)
lesson_id BIGINT
student_id BIGINT
teacher_id BIGINT
Indexes
student
id BIGINT
address VARCHAR(255)
epayment INT
grade VARCHAR(255)
name VARCHAR(255)
surname VARCHAR(255)
link INT
school_id BIGINT
student_id BIGINT
Indexes

Entities

1- Lesson

It is the main place of the classroom. Everyone's aim here is lessons. Lesson entity includes lesson_id and lesson_elesson.

2- Student

Students try to learn lessons from the site. Each student can have more than one teacher. Student entity includes student_id, student_name, student_surname, student_epayment, student_grade, student_adress.

3- Teacher

Teachers are responsible for teaching students on the site. Each teacher can teach in only one area. Teacher entity includes teacher_id, teacher_name, teacher_surname, teacher_adress.

4- Generated link

It is the connection point required for students and teachers to enter the lessons. Generated link entity includes generated_link_link, generated_link_id.

5- **School**

Students must go to school. School entity includes school_id and school_city.

6- Temp_student

It is the entity that the students' information is registered in the system.Temp_student entity includes Temp_student_id, temp_student_level, temp_student_adress, temp_student_epayment, temp_student_name, temp_student_surname.

7- Payment

Each student has to pay to be able to take the course. Payment entity includes payment_id.

Lesson-Student Relationship:

The student must always study. Here, too, students have to take which courses in the system, which courses they can take and can view them.

Many students can learn more than one lesson.

Lesson-Student Relationship has many-to-many (M:N) relationship.

Lesson-Teacher Relationship:

Teachers become experts in only one area. That's why they teach only one area.

One teacher teach just one lesson.

Lesson-Teacher Relationship has one-to-one (1:1) relationship.

Teacher-Student Relationship:

Teachers are obliged to teach students lessons.

Many teachers have many students.

Teacher-Student relationship has many-to-many (M:N) relationship.

Student-Generated Link Relationship:

These are the links that students use to connect to live lessons.

A link is only for 1 student.

Student-General Link relationship has one-to-one (1:1) relationship.

Teacher Generated Link Relationship:

These are the links that teacher use to connect to live lessons.

A link is only for 1 teacher.

Teacher-General Link relationship has one-to-one (1:1) relationship.

Student-School Relationship:

Every student has to go to school to get a diploma.

Each student can only attend one school.

Student-School relationship has one-to-many (1:M) relationship.

Student-Temp Student Relationship:

The information of each registered student must be recorded in the system.

Each student can only have 1 temp file.

Student-Temp Student relationship has one-to-one (1:1) relationship.

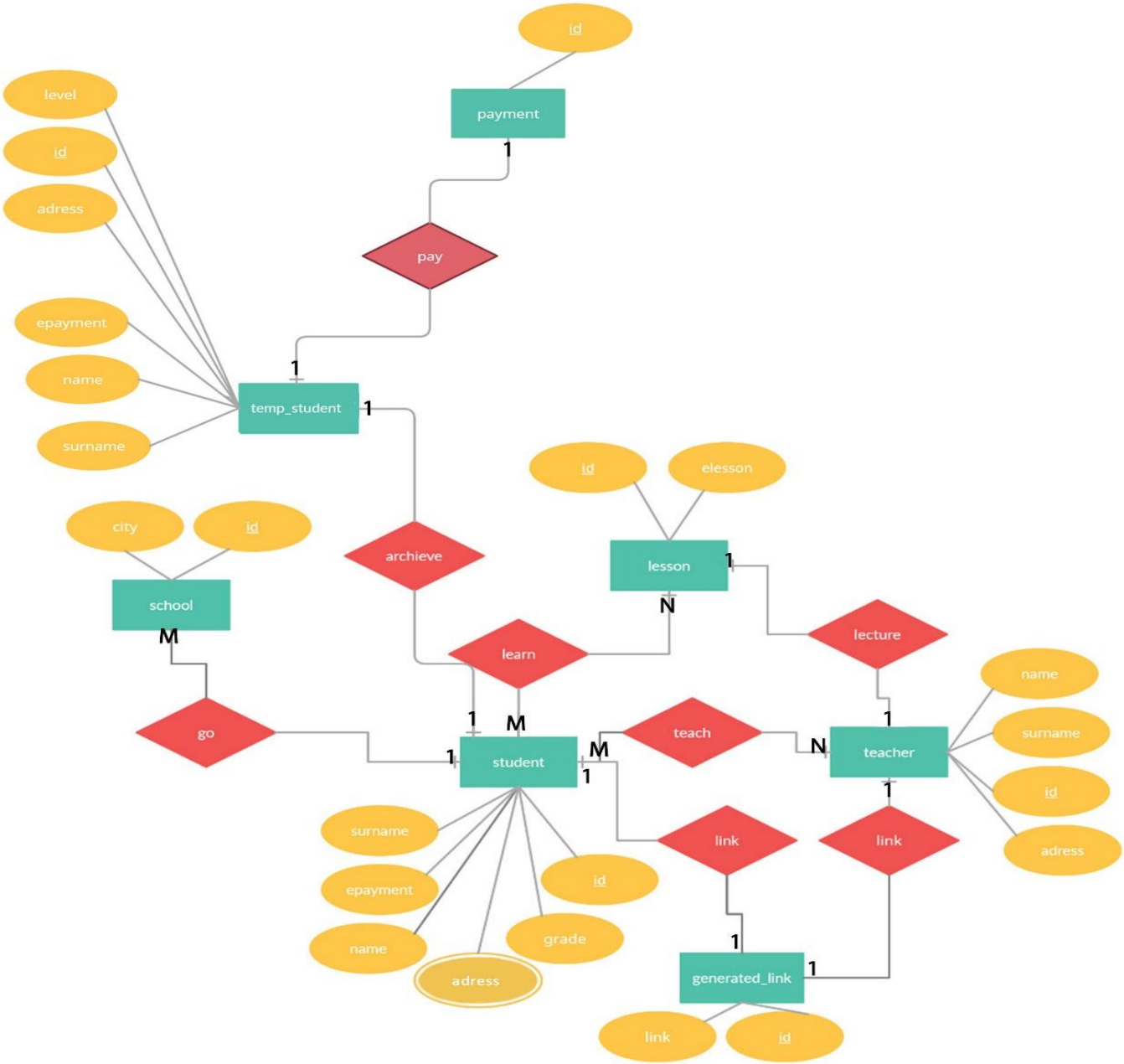
Temp Student-Payment Relationship:

Every student enrolled in the system has to pay in order to take lessons.

One student only pays for himself.

Temp Student-Payment relationship has one-to-one (1:1) relationship.

ERD


id
payment
level
id
adress
1
pay
epayment
name
surname
1
temp_student
1
id
elesson
city
id
archieve
lesson
1
school
N
lecture
M
learn
name
1
M
1
M
teach
N
1
go
1
student
1
teacher
surname
id
adress
surname
link
link
epayment
id
name
grade
1
adress
generated_link
1
link
id

ARCHITECTURAL DESIGN OF SYSTEM

1. Operational Requirements

1.1 Technical Environment Requirements

- The system will work over the Web environment with Google chrome and Firefox.

  (System is a website and you can access the lessons in this website)

- The system support only pc platform and tablets, it not support the mobil platforms and others as an application.

  (laptop, desktop and tablets with web browsers.)

- The system is accessible only from Turkey.
  (The system created only for Turkish students. Also, it contain security constraint.)

- All users must be online to access the system because this systeam an online platform.
  (Wifi or local area network)

- The system based on cloud computing.

- In MyCourse system will use mysql

  MySQL is a free database management system and also, it is so common thanks to this taking support and finding expert is easier than other database management systems.

- MyCourse system will based on Linux Operating system.

Because Linux systems are common, free and easy to solve system problems.

Also, finding expert is easier than other server management systems.

1.2 System Integration Requirements

- The system must be able to export and import Excell spreadsheets.
- The system will write to every log admin database.
- The system must be entegrated banking and accounting systems.

1.3 Portability Requirements

- In this version system will be support only computers.
- MyCourse web site do not support mobile devices in this version.
- System will work over Internet browser in this version.

1.4 Maintainability Requirements

- Next versions, system will be support mobil diveces for browser and application.

(iOS and android)

- Next versions , system will have an deskop application but system will countinue to sopport web broswers.(Chrome and firefox)
- First update will come after one year and others will come every six month regularly.

(In first year will be observation for demands and densities.)

2. Performance Requirements

2.1 Speed Requirements

- All databases must me updated in real time.
- In live lesson and streaming lesson response time must be less than 30 seconds.
- In other any transections response time must be less than 5 seconds.
- New students must reported the admin once everyday.

2.2 Capacity Requirements

- there will be a maximum of 3050 simultaneous users at peak use time (60 percent of students plus all personals)
- The system will store data on approximately 5000 students for a total of about 10 mb of data (logs and lessons times)

2.3 Availability and Reliability Requirements

- The system will be available 7/24, with the exception of scheduled maintenance.
- (every month wednesday 02:00-05:00)
- Time of scheduled maintenance must not more than 3 hours.
- The system will have %99.98 % uptime performance

3. Security Requirements

3.1 System value estimates

- A complete loss of all system data is estimated to cost abouth 20.000 tl.

(Student data, personal data and lessons time.)

- In system outage, there is no lost money but users system can lose confidence.

3.2 Access Control Requirements

- In MyCourse system, we have limitation. Only main admin can access everywhere , others have some limitation. We use for this personal id and password.
- Teachers, can access students data and their attandance and they can stream lesson.
- Students can access the live lessons and they can see online students and teacher.

3.3 Encryption and Authentication Requirements

- Only personal information will be encrypted from the user's computer to the web site to provide personal security.

- All users must enter user id and password for do anything on MyCourse website.

3.4 Virus control requirements

- The system will be checked at regular intervals for viruses. (per month).
- Unauthorized persons can not upload anything on the system.
- All uploaded files will be checked by system.

4. Cultural and Political Requirements

4.1 Multilingual requirements

This system will use only Turkey, so all information based on Turkish.

- System language will be turkish. (All options and all buttons will be turkish.)

4.2 Customization Requirements

- Entering the telephone number must be start with +90
- Using on the system must be local turkey hour format. (utc+3)
- All date fields must use only day-month-year in the system

4.3 Making Unstated Norms Explicit

MyCourse system created for Turkish student so it is not international system. Therefore, there is no unstated norms in the system.

4.4 Legal requirements

- System store personal data about 2 years because of the Turkish kvkk
- System is subjected to Turkish law and rules regarding personal data.

Network Architecture

In this system (MyCourse) will based on client-server architecture. We choose client-server architecture because of some reasons.

- In the client-server architecture you can easly increase or decrease the storage and processing capabilities of the server .

If any server overloaded you can add one more server or there is not enough request, you can reduce one server. Thanks to this architecture, server cost will increase gradual not too much.

- Client-server architecture support different type of client and servers.

Thanks to this property MyCourse system will not live mismatch.

- When use the Internet standarts, you can easly spread and somewhat independent. (presentation logic application logic, and data access logic)
- When a server fall in client-server architecture , other servers not apply from them. They continue the run in the system.

Client-server architecture some complex in technical issues but it is not problem for our customer because our firm will set-up and update the system. Therefore, they will not need any technical expert or etc.

Clients

MyCourse is a decentralized system so all students and personals must have their own client. (computer) Therefore we cannot determine any client for the system. It is totally dependent to the students and personals.

Client-Server Tiers

In the MyCourse system we will use three-tiers (client-server)

- Flexibility
- Distributing the application layer across many nodes.
- More secure

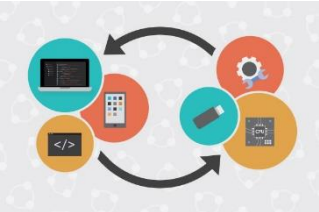
This tier can some disadvantages with internet connection and integrating of the three tiers. However, we have very-well experts and our servers will be based on cloud. That's why these problems will not affect this system.

Cloud

MyCourse system will use infrastructure as a Service cloud delivery model. (IaaS)

- Applications
- Runtimes
- Security and integration
- Database
- Servers

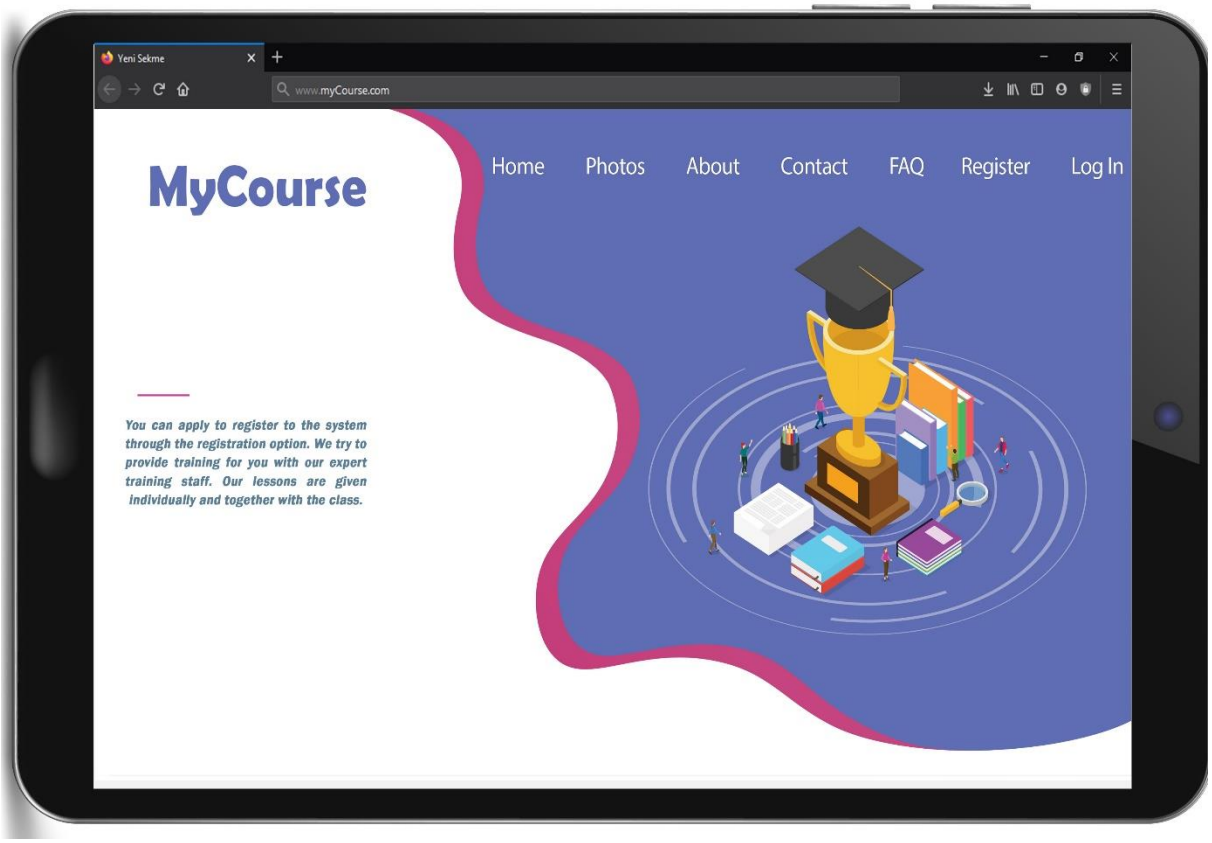
We chose IaaS as the cloud system because we can control the above features. Storage and network problems will be managed by vendor of cloud system. Also, cloud deployment model of the MyCourse will be public cloud because of the low price and increasable storage and bandwidth. In public cloud model we will not pay any cost for set-up and we will not lose any space floor. Also, we do not pay maintenance cost. We save too much money.



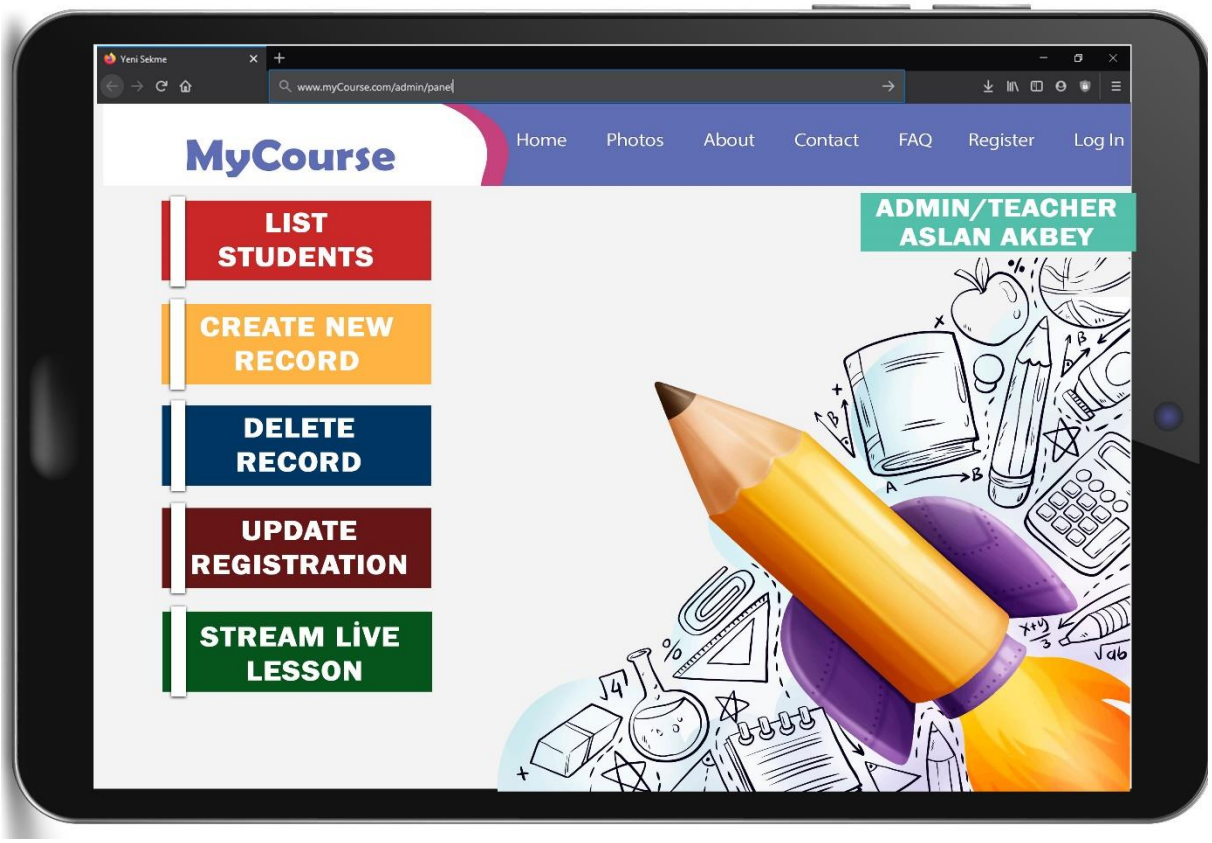
Software and Hardware Specifications

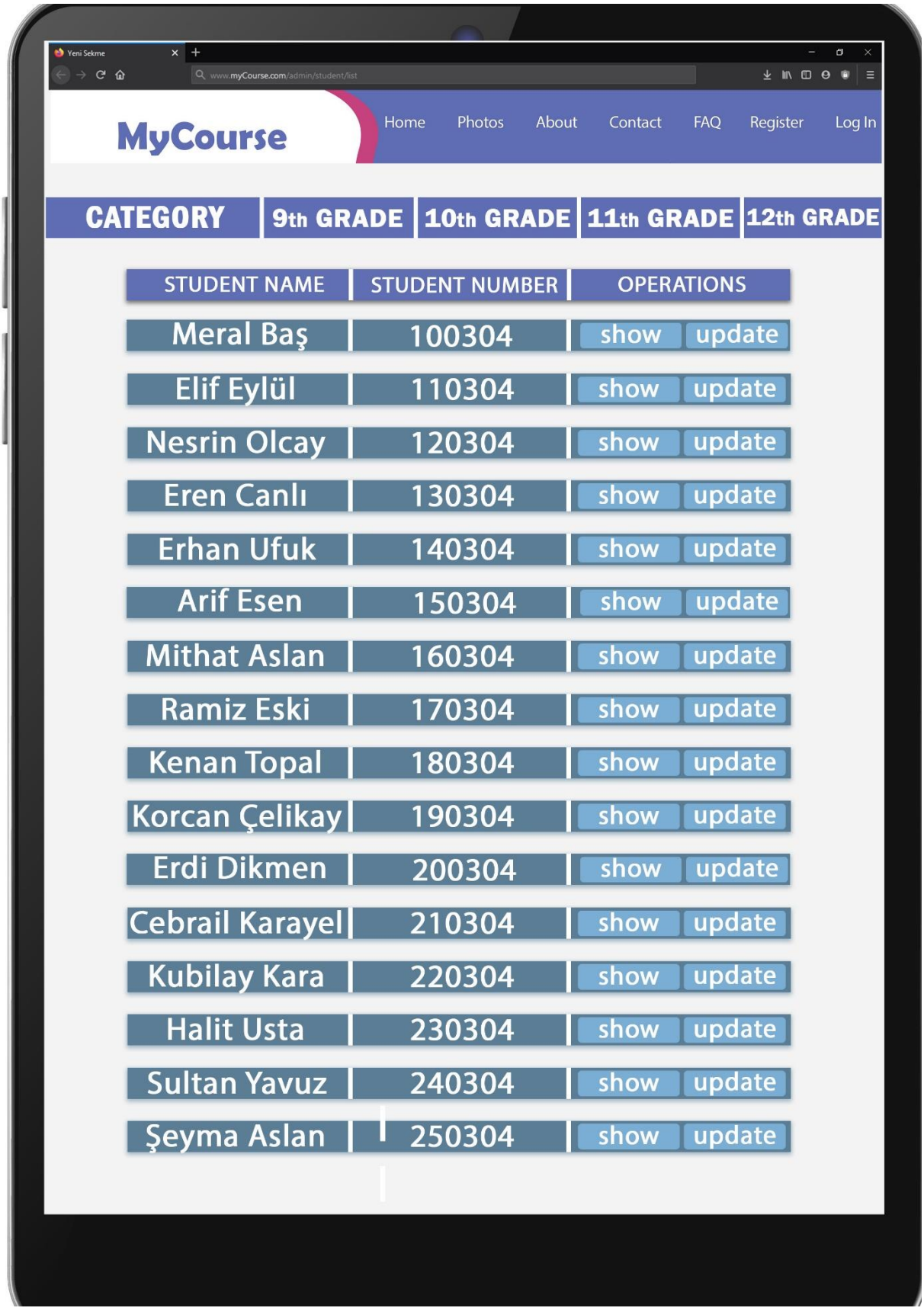
	Standard Client	Standard Web Server	Standard Application Server	Standard Database Server
Operating System	-Windows -MAC	-Linux	Linux	-Linux
Special Software	-Adobe Flash Player	-IIS	IBM	-mysql
Hardware	-500 GB disk drive -Camera -Microphone Intel® Core™ i5-7200U -15.6” LCD monitor	-1 TB disk drive -Intel® Xeon® W-2200	-500 GB -Intel Xeon® E5-2680 v4 2.4 GHz	-2 TB disk drive -Mysql
Network	-2.0 Mbits / sec upload -5.0 Mbits / sec download	Dual 100 Mbps Ethernet	Dual 100 Mbps Ethernet	Dual 100 Mbps Ethernet

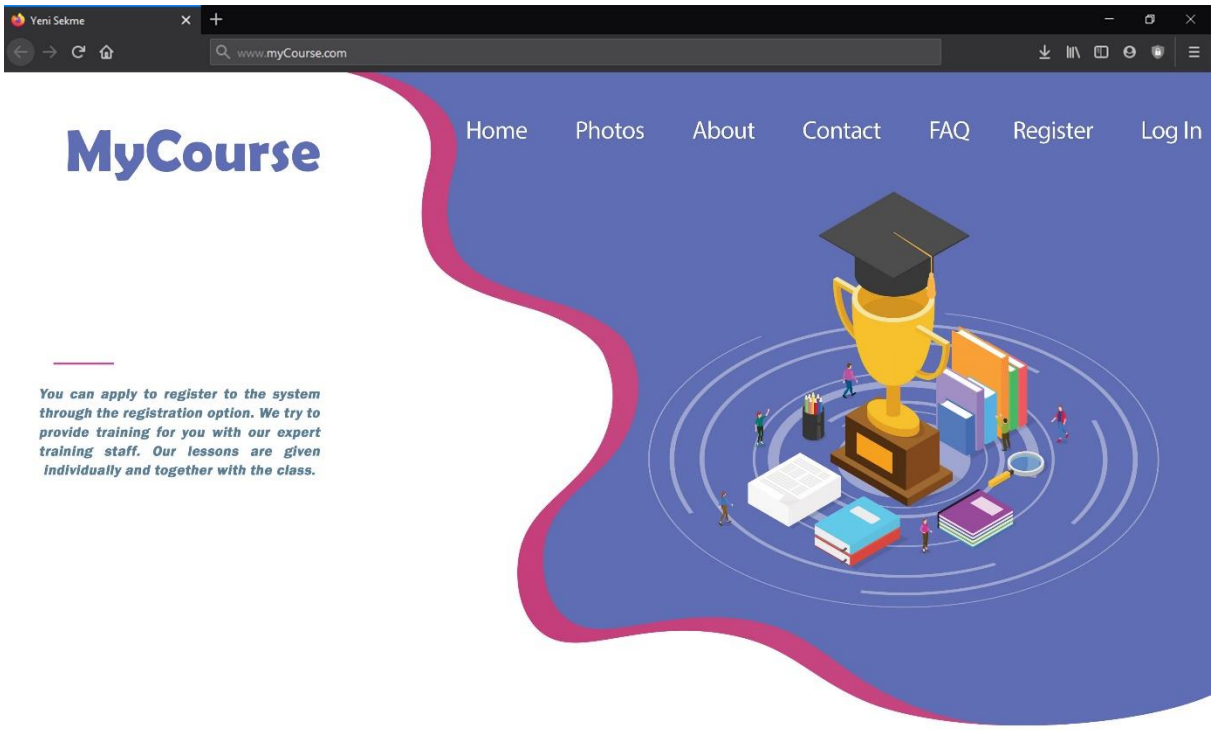
USER INTERFACES

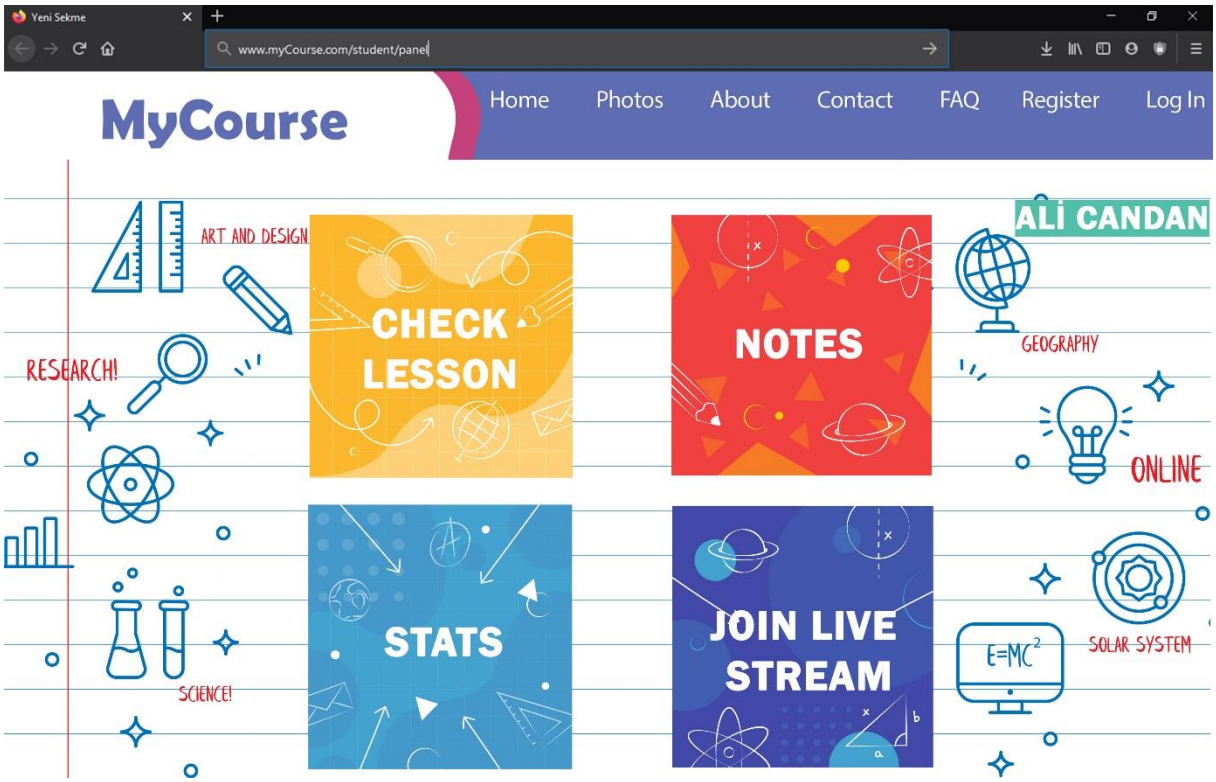


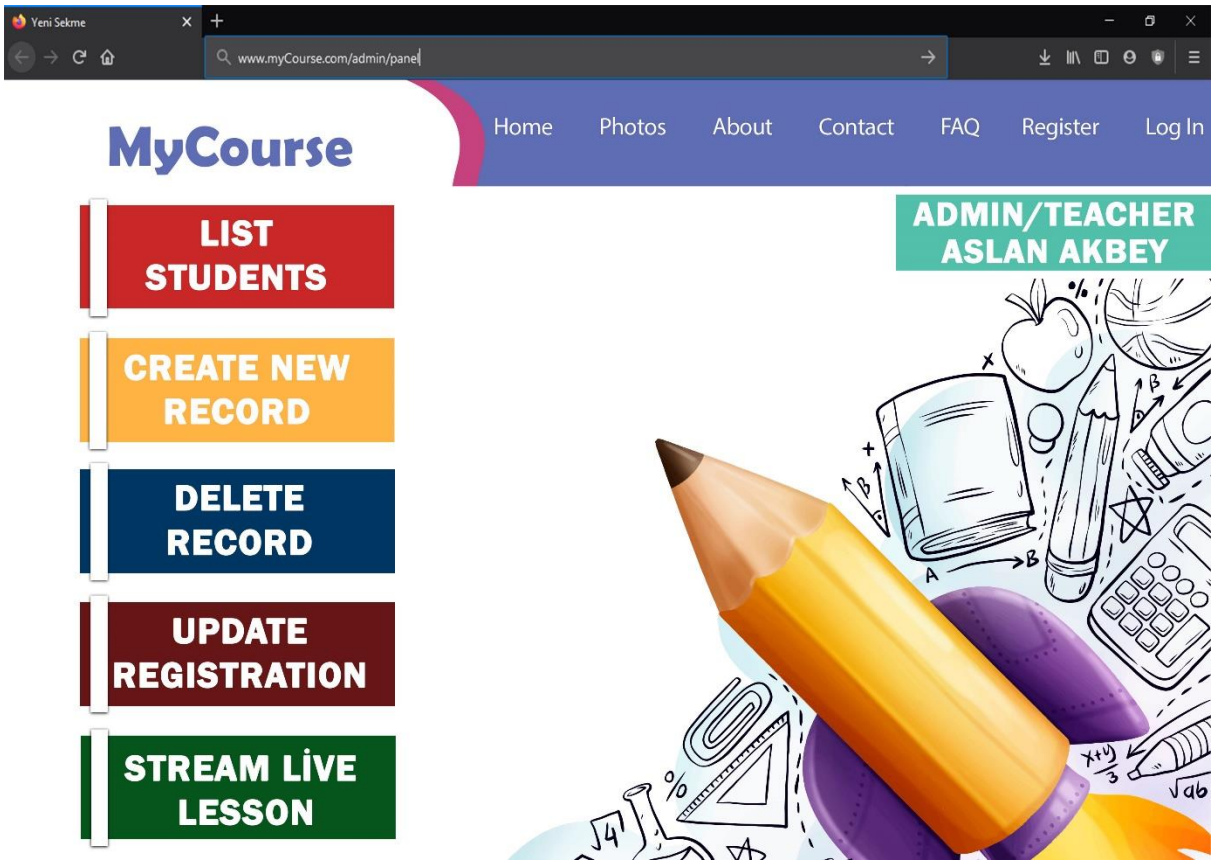

Yeni Sekme
www.myCourse.com/student/panel
MyCourse
Home
Photos
About
Contact
FAQ
Register
Log In
ART AND DESIGN
CHECK LESSON
NOTES
ALİ CANDAN
GEOGRAPHY
RESEARCH!
ONLINE
STATS
JOIN LIVE STREAM
E=MC²
SOLAR SYSTEM
SCIENCE!

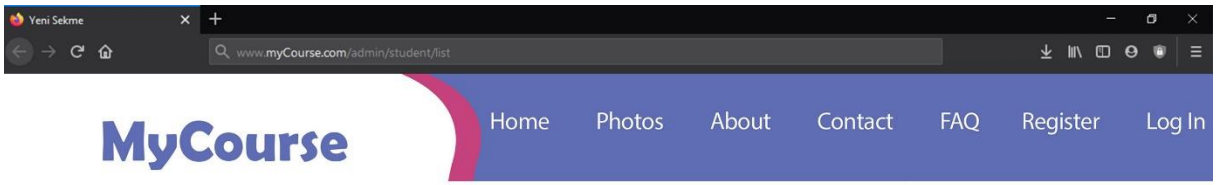

Yeni Sekme
www.myCourse.com/admin/panel
MyCourse
Home
Photos
About
Contact
FAQ
Register
Log In
LIST STUDENTS
CREATE NEW RECORD
DELETE RECORD
UPDATE REGISTRATION
STREAM LİVE LESSON
ADMIN/TEACHER
ASLAN AKBEY


Yeni Sekme
www.myCourse.com/admin/student/list
MyCourse
Home
Photos
About
Contact
FAQ
Register
Log In
CATEGORY
9th GRADE
10th GRADE
11th GRADE
12th GRADE
STUDENT NAME
STUDENT NUMBER
OPERATIONS
Meral Baş 100304 show update
Elif Eylül 110304 show update
Nesrin Olcay 120304 show update
Eren Canlı 130304 show update
Erhan Ufuk 140304 show update
Arif Esen 150304 show update
Mithat Aslan 160304 show update
Ramiz Eski 170304 show update
Kenan Topal 180304 show update
Korcan Çelikay 190304 show update
Erdi Dikmen 200304 show update
Cebrail Karayel 210304 show update
Kubilay Kara 220304 show update
Halit Usta 230304 show update
Sultan Yavuz 240304 show update
Şeyma Aslan 250304 show update


Yeni Sekme
www.myCourse.com
MyCourse
Home
Photos
About
Contact
FAQ
Register
Log In
You can apply to register to the system through the registration option. We try to provide training for you with our expert training staff. Our lessons are given individually and together with the class.


Yeni Sekme
www.myCourse.com/student/panel
MyCourse
Home
Photos
About
Contact
FAQ
Register
Log In
ART AND DESIGN
CHECK LESSON
NOTES
ALİ CANDAN
GEOGRAPHY
RESEARCH!
ONLINE
STATS
JOIN LIVE STREAM
SOLAR SYSTEM
E=MC²
SCIENCE!

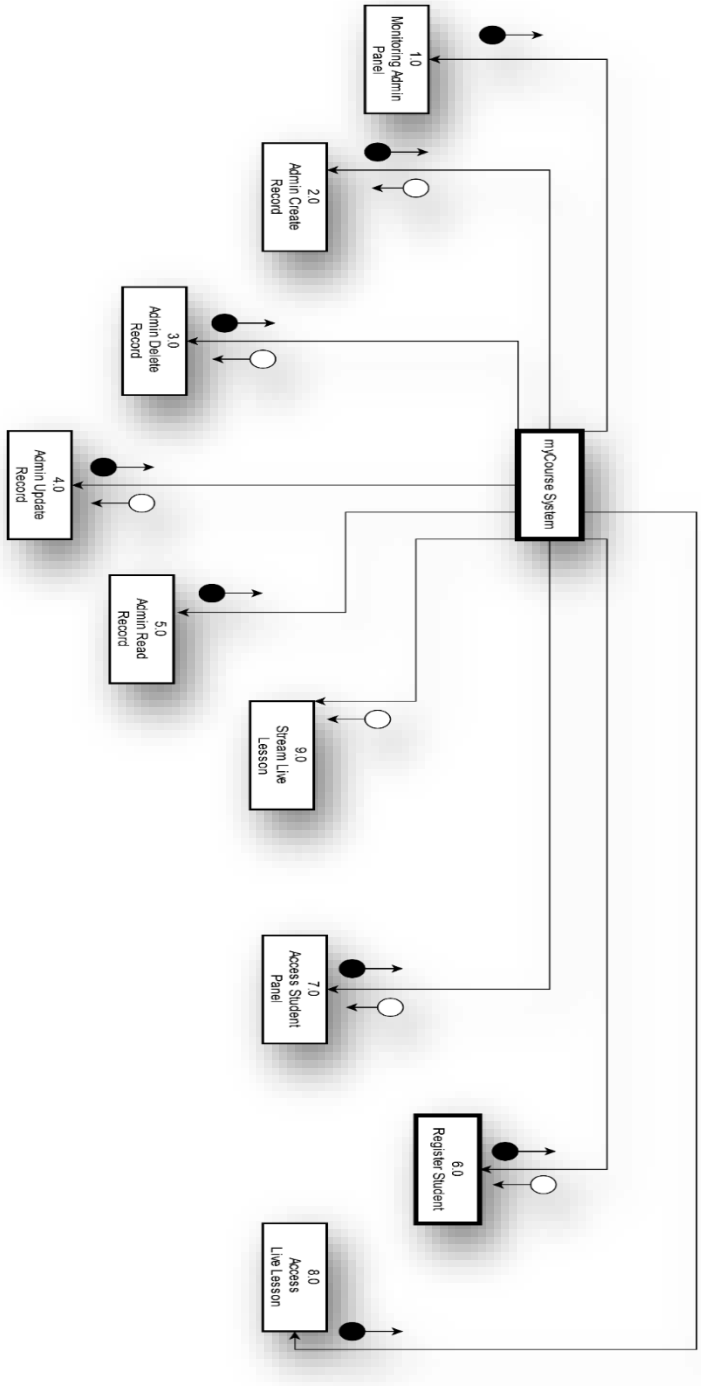

Yeni Sekme
www.myCourse.com/admin/panel
MyCourse
Home
Photos
About
Contact
FAQ
Register
Log In
LIST STUDENTS
CREATE NEW RECORD
DELETE RECORD
UPDATE REGISTRATION
STREAM LİVE LESSON
ADMIN/TEACHER
ASLAN AKBEY



CATEGORY	9th GRADE	10th GRADE	11th GRADE	12th GRADE

STUDENT NAME	STUDENT NUMBER	OPERATIONS
Meral Baş	100304	show update
Elif Eylül	110304	show update
Nesrin Olcay	120304	show update
Eren Canlı	130304	show update
Erhan Ufuk	140304	show update
Arif Esen	150304	show update
Mithat Aslan	160304	show update
Ramiz Eski	170304	show update
Kenan Topal	180304	show update
Korcan Çelikay	190304	show update
Erdi Dikmen	200304	show update
Cebrail Karayel	210304	show update
Kubilay Kara	220304	show update
Halit Usta	230304	show update
Sultan Yavuz	240304	show update
Şeyma Aslan	250304	show update

SYSTEM STUCTURE CHART



Admin Create Record:

On the admin panel, admin create the specified user. Admin gets request to create new user. System shows information about user. Admin creates it.

```
function createUser {
user =  tempDbService.getUser // get the temp user information
systemUserService.createUser(user) // save on the system
return the saved user //
}
```

Admin Delete Record:

On the admin panel, admin delete specified user. Admin checks the information of the user about constraint or requirements. If there is no constrain or requirements admin deletes the user.

```
function deleteUser{
user  = UserDb.getUser() // get user to be deleted
pass = checkConditions(user) // check conditions or requirements
if pass is true
system.delete(user) // if there is no conditions delete the user
else
send error // otherwise throw an error
}
```

Admin Update Record:

On the admin panel, admin gets the request for update a user. AdminS reach the update screen and system shows information of user to be updated. Admins accept the request and user is updated.

```
function updateUser{
user = getUserFromRequest() // get the user from requested data
pass = checkInformtions(user) // check it
if pass is true
system.updateUser(user) // update the user
else
send error // send an error
}
```

Admin List User:

When admin on the panel, system shows the all the users by category. Admin selects any category and system list the users by category.

```
function getUserList(){
userList = system.getUsersByCategory(category) // system generates the user by given category
return userList // returns the list of categorized user.
}
```

Implementing the System

The MyCourse project is composited well and prepared for end-user. All components and dependencies are packed and composited well for tests.

Before the deploying, the test engineers test the project on several parts such as unit test, unit tests, integration tests, system tests, and acceptance tests for bugs and errors.

Unit Test Demo Example

Purpose	Preconditions	Inputs	Outputs	
Students enter the system.	Student on the student sign in screen.	Correct usernames and passwords.	Students access the system	Success
Teachers enter the system.	Teacher on the teacher sign in screen.	Correct usernames and passwords.	Teachers access the system.	Success
Admins enter the system.	Admin on the manager sign.	Correct usernames and passwords.	Admins access the system.	Success

Acceptance test is critical phase. Our customers test the project for two parts. Alpha and beta.

Our customers test with temporal data for functionality and with real data for check whether the whole system works correctly.

After all tests, project goes to Merge to Trunk process. After that, the test engineers test the post commit test and gives a report for bugs or errors and bugs and errors are fixed as soon as possible. After the last processes DevOps deploy the project.

The developers that worked on the project are experienced and they develop the project as to be minimum system usage.

Supporting the System

Software, like all other products, is provided with warranty and technical support. However, software can face many problems, security, crash, etc. As an IT Consultancy we provide a huge network for IT support and our products have 6 months warranty and lifelong support service.

Server-side problems:

The MyCourse project has also have lifelong support service and the project is served on our server, so we always help our clients for server-side problems. The client-side problems may cause by many reasons and we never know without contact the clients.

Client-side problems:

The clients can open a ticket for bugs, error etc. or report for bugs. We provide remote-control support for client-side problems. If we can solve the problem ticket close with successfully.

If we could not solve it our field support team deal with the problems. After the work, the team reports the bugs. We discuss about it. If the problem is not caused by us, we will redirect the client to the cause of error.

So, we defend and support our products every time.

REFERENCES

- Systems Analysis and Design, by Harry Rosenblatt, Shelly Cashman, Cengage Learning, 10th edition, Mar 2013
- Practical Guide to Structured Systems Design by Page-Jones, Prentice-Hall 1988.

TOOLS

- JETBRAINS INTELLIJ IDE (back-end)
- ADOBE DREAMWEAVER (front-end)
- ADOBE PHOTOSHOP CS6 (graphical design)
- Draw.io (schemas)

Thanks to Assist. Prof. Dr. Vildan ATEŞ.

This has been an incredibly good experience for us MIS students.