FUSICS

Team: _e03_team_GGWP

Team mates:

Md. Ibrahim Khalil
Tasmia Zerin
Ahmed Adnan
Mustahid Hasan

Table of Contents

FUSICS

Fusics is a Physics-based interactive learning desktop application. It uses a gamification model for teaching physics. It is a great platform for teacher-student interaction as well as virtual experimentation.

Problem Identification

During this global pandemic, it has become hard to conduct classes, as well as evaluating the performance of the students. Though it is possible to convey the study materials, the hands-on practical sessions are not possible to deliver. As a result, education during this time has become very tough and monotonous. This is resulting in the failure of proper communication between the teachers and the students. Our app satisfies the problems of both hands-on practical sessions as well as interactive learning. As the app interacts with the students through gamification, it encourages the students for studying, and thus, it has a structure for tracking the students through distance learning. Moreover, the app also connects the students with their teachers by using a virtual classroom.
This app is user-friendly and there is an option for offline use as well.
As Physics is difficult to visualize, this app focuses on simulating the concepts of Physics for a better understanding. Moreover, our target audience are higher secondary and secondary students, as the core concepts of physics must be clearly understood within these stages. As students don’t have access to devices/apparatus, they can conduct their hands-on practical with proper simulation. Our app more effectively teaches Physics than other e-learning softwares. As the available online

solutions of the hands-on practicals cannot be simulated, the students can easily simulate and visualize the topics through FUSICS.

Risk Analysis

1. The graphical simulations are in 2D which would have been better in 3D.
2. There are no sound effects in the simulations which would have helped the students to experiment with the simulations in a better way.
3. We could have made this app as a VR app but since VR technology is very costly and technical, using this technology would be a luxury in our country's socio-economic situation. This app is made for mass use and one of the primary goals of this app is to reach as many people as possible. So, if we implement the VR technology, there wouldn't be enough skilled and privileged people to use it.

Idea Formulation

The user first enters the game and sees the list of core topics of physics. Every topic has 3 sections. About, Experiment and Solve.

In the "About" section, the user can see the definition of the topic, and all the formulas related to it. Also, there is an animation of the topic.

In the "Experiment" section, the user can try his/her desired values to the result as simulation. For example, in a projectile, a user can add a ball, and add velocity angle to it and see the motion of the whole time. Also users can stop it any time to see the values of any instant. For checking collision, a rectangle can be added as an object. By this the user can understand the concept fully.

After that, in the "Solve" section, the user can evaluate her learnings'.There are two parts in the "Solve" section, "Practice" and "Exam"."Practice" is of two types, Category based and difficulty based. Category means any particular variable dependent (height, range, velocity etc). Difficulty consists of easy, medium and hard. In practice, a question will appear and the user has to answer it. after answering, she will see the correct answer. And if the user gives the wrong answer, solution and animation of both the wrong and correct answer will be given.

Then lastly, Exam is two types, single player and multiplayer. Single player is user adaptive, by giving correct answers users will be given

harder questions than the previous. In multiplayer users can play with friends, at most 10 people at a time. Everyone will be given the same question to answer within a particular time.

This app can authenticate teachers and students separately.A teacher can create a virtual classroom by generating a unique class code.The teacher can assign theoretical as well as practical tasks to the students. There is also a way to evaluate the performance of the students. Then a leaderboard with marks and solutions will be shown after the exam.
By this, the user can understand physics by learning, practicing, interacting and competing. So, let's "Fusics".

Implementation Plan and Work Progress

- Our app consists of a virtual experiment section which is a perfect substitute for live experimentation and hands-on practical sessions.
    - ❏ In this section, students can add any object (ex: ball) and any obstacles to that. So students can visualize the topic by simulating through these.
    - ❏ The ongoing experiment can be stopped at any time for further analysis and observation, as the instant values of all the parameters are shown every time.
    - ❏ Detecting collision can also be possible by adding any obstacle, so students can understand any real life example by visualizing.
- A virtual classroom can help both distant learners and instructors. The instructor can assign any task, question or experiment and evaluate the performance of the students.

- Structured modeling of learning procedures makes our app unique from any other learning software.
    - ❏ Any student can first get ideas about the topic, and then experiment as well as visualize.
    - ❏ If he/she wants to do better in the topic, then there is also a practice option for that.
    - ❏ Lastly, checking the progress and tracking his/her own performance, as well as competing with others can be done with the app.
- Our application is designed especially for secondary and higher secondary students. As our application is based on the curriculum of these levels, the students can easily be attracted and motivated to use it. As the app is made user friendly, anyone can use it without facing difficulties. In addition, the app is cost-efficient, as no apparatus is used for the virtual practicals.
- Peer learning using engaging content can be solved by multiplayer gaming. The proposed app can take exams by setting up questions for the students with their solution, visualization and evaluation.
- Gamification is a relatively novel technology in the educational and working area.
    - ❏ Student motivation is always something to keep in mind when planning lessons, as a teacher. By applying gamification to the classroom, students could be motivated to learn in new ways or enjoy otherwise tedious tasks.
    - ❏ When learners are engaged with learning they become active information-consumers who take the initiative and push forward their development.
    - ❏ When it comes to engagement in games there are two types of motivation: intrinsic and extrinsic. Gamification desires to

combine intrinsic motivation with extrinsic in order to raise motivation and engagement.

- ❑ Gamification uses structured model which contains these steps,

➔ **Play -** Setting the boundaries for play and allowing exploration and failure within;

➔ **Exposition** - Creating metaphors which integrate with real world settings;

➔ **Choice** - Putting the power of choosing in the user's hands;

➔ **Information** - Using engagement through game design and display to create concepts which enhance learning process;

➔ **Engagement** - Encouraging collaborative learning with other users who have similar real-world interests;

➔ **Reflection** - Assisting participants to find interests and past experiences which can enrich the learning process with deeper engagement

This proves that the app ineracts by using gamification and helps the students for a better understanding.

Use Case Scenario

DEFINITION OF USE CASE

A Use Case captures a contract that describes the system behavior under various conditions as the system responds to a request from one of its stakeholders. In essence, a Use Case tells a stylized story about how an end user interacts with the system under a specific set of circumstances. A Use Case diagram simply describes a story using corresponding actors who perform important roles in the story and makes the story understandable for the users. The first step in writing a Use Case is to define that set of \actors" that will be involved in the story. Actors are the different people that use the system or product within the context of the function and behavior that is to be described. Actors represent the roles that people play as the system operators.

Primary Actor

Primary actors interact directly to achieve required system function and derive the intended
benefit from the system. They work directly and frequently with the software.

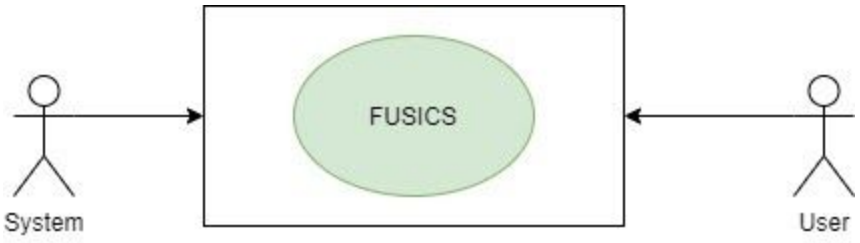
Secondary Actor

Secondary actors support the system so that primary actors can do their work.
They either produce or consume information.

Use Case diagrams give the non-technical view of the overall system.

Level 0

Use case Id: 0
Use case Name: FUSICS

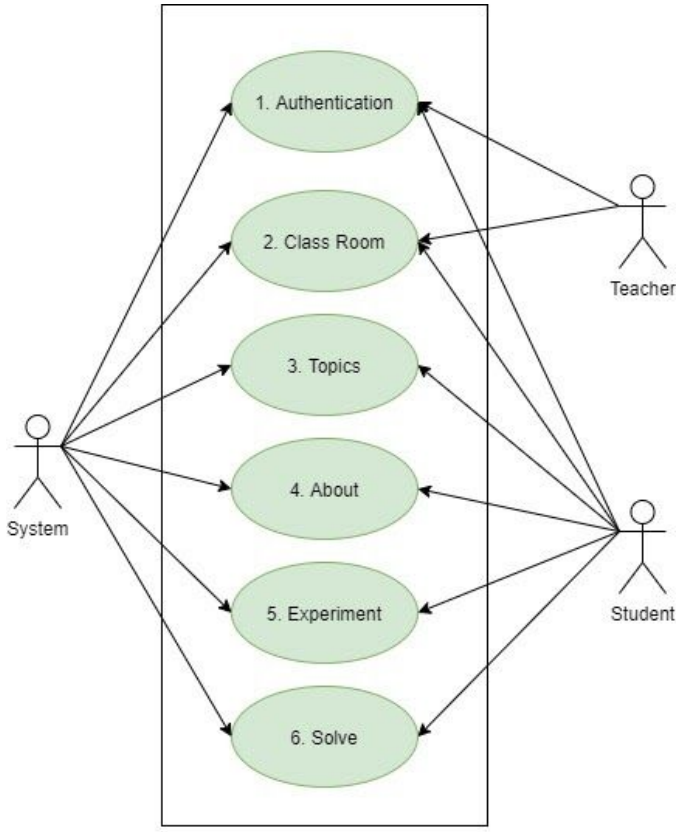


Level - 0

Primary Actor: User
Secondary Actor: System
Description: Low level structure of the application

Level 1

Use case Id: 1
Use case Name: FUSICS’s subsystems

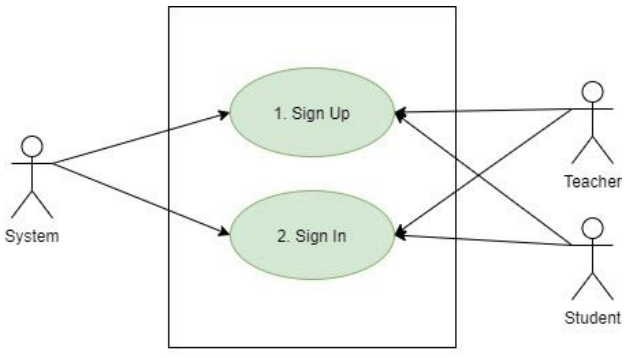


Level - 1

Primary Actor: Teacher & Student
Secondary Actor: System
Description: The diagram shows all the sub components of the application.

Level 2

Use case Id: 2
Use case Name: Authentication

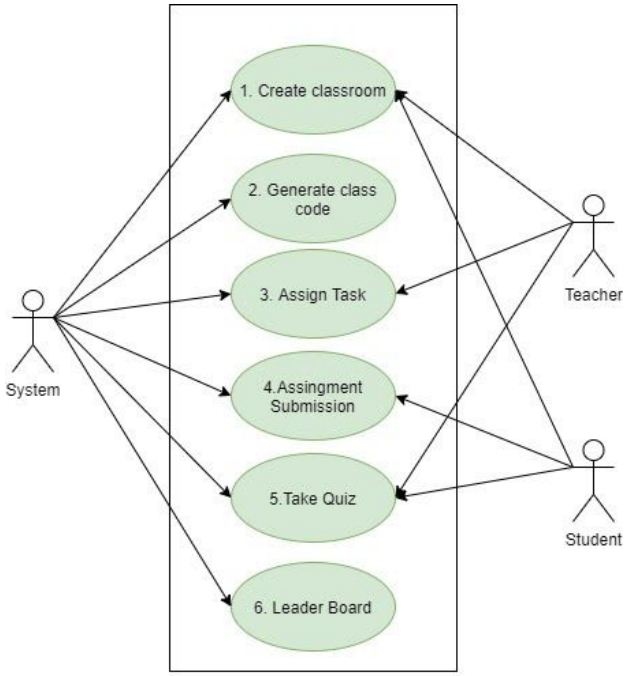


Level 1.1

Primary Actor: Student & Teacher
Secondary Actor: System
Description: User Authentication by sign in and login

Level 3

Use case Id:3
Use case Name: Virtual Classroom

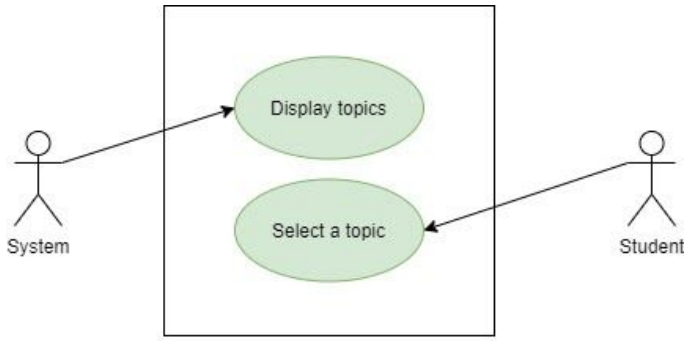


Level - 1.2

Primary Actor: Teacher & Student
Secondary Actor: System
Description: All attributes of the virtual classroom are shown here

Level 4

Use case Id: 4

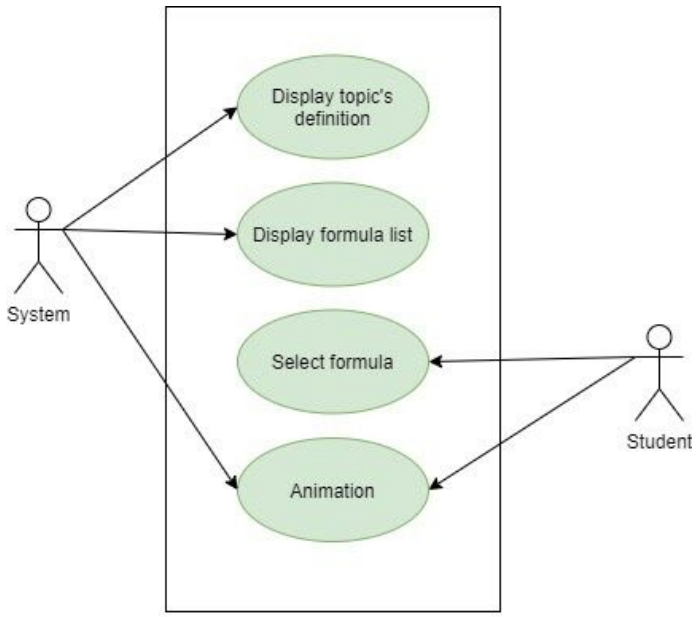


Level 1.3

Use case Name: Display topics
Primary Actor: Student
Secondary Actor: System
Description: Student can select the desired topic

Level 5

Use case Id: 5
Use case Name: About the topic

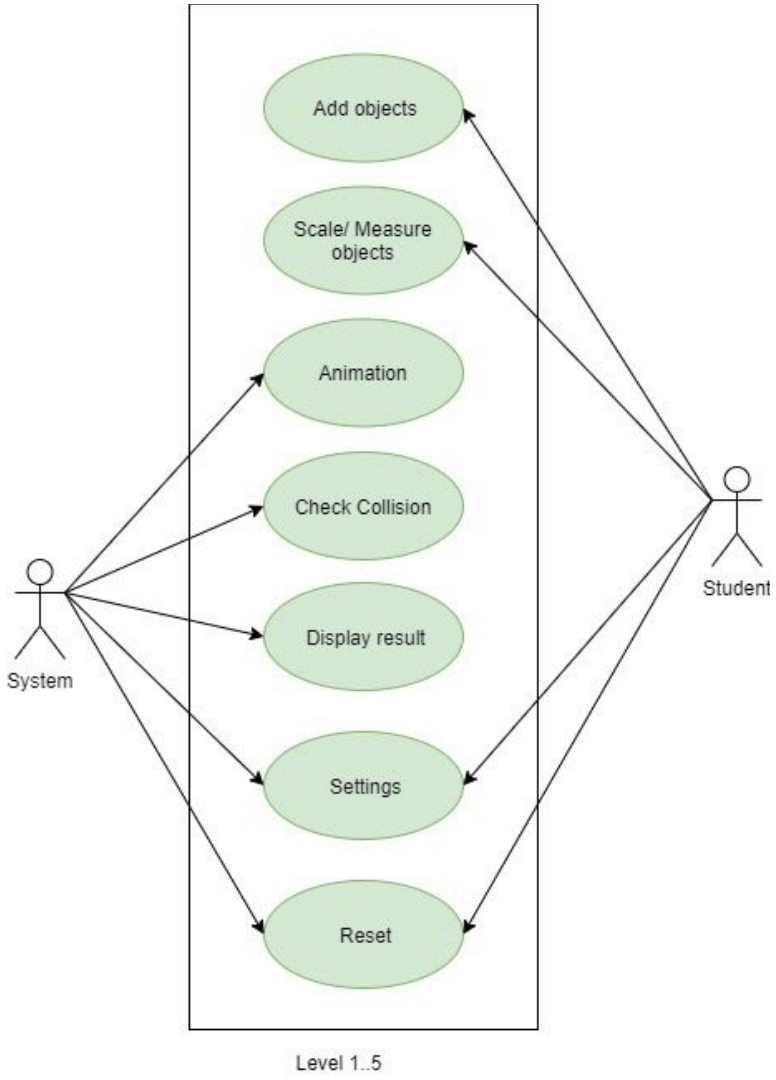


Level - 1.4

Primary Actor: Student
Secondary Actor: System
Description: Students can learn about the topic by animation, definitions and formulas.

Level 6

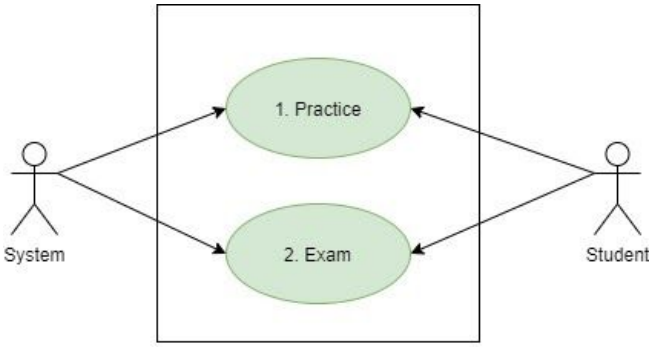
Use case Id: 6
Use case Name: Experiment



Primary Actor: Student
Secondary Actor: System
Description: Experiment by adding objects and checking collisions with the parameter values.

Level 7

Use case Id: 7
Use case Name: Solve

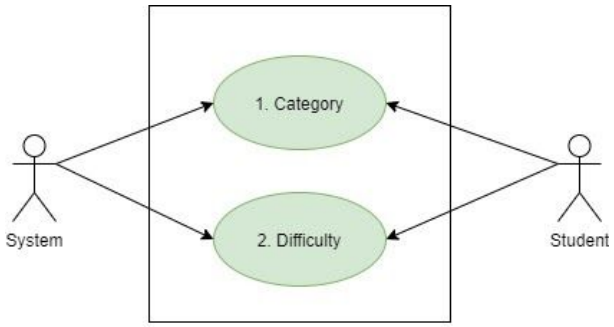


Level - 1.6

Primary Actor: Student
Secondary Actor: System
Description: Students can practice, as well as give exams.

Level 8

Use case Id: 8
Use case Name: Practice

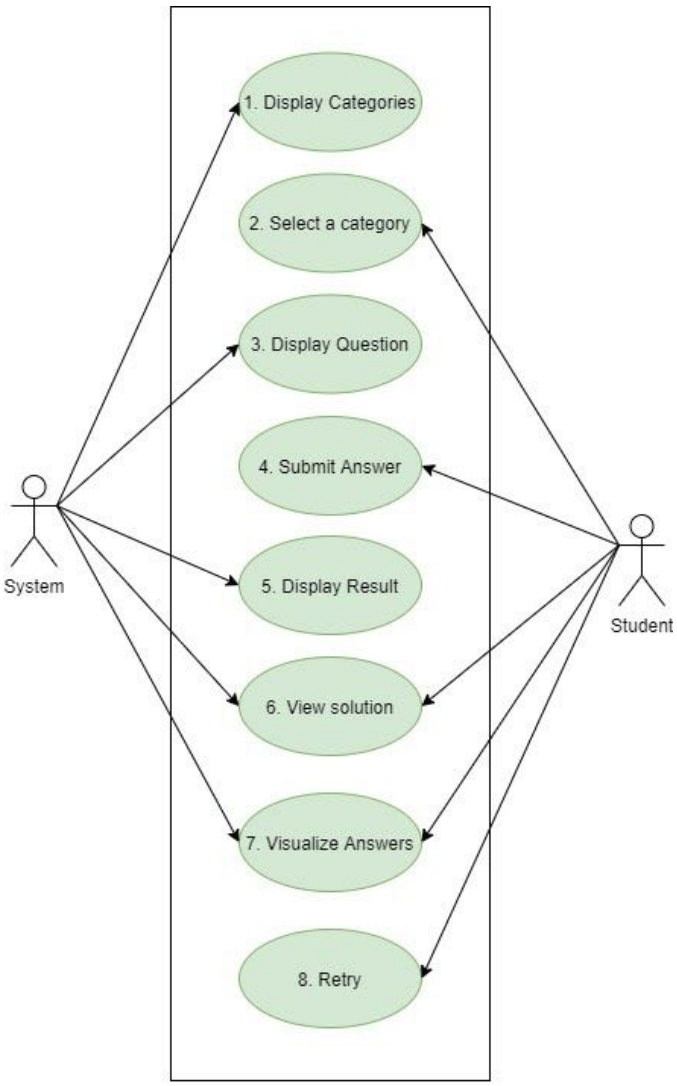


Level - 1.6.1

Primary Actor: Student
Secondary Actor: System
Description: Students can choose the category and the difficulty level of the problems.

Level 9

Use case Id: 9
Use case Name: Practice by Category

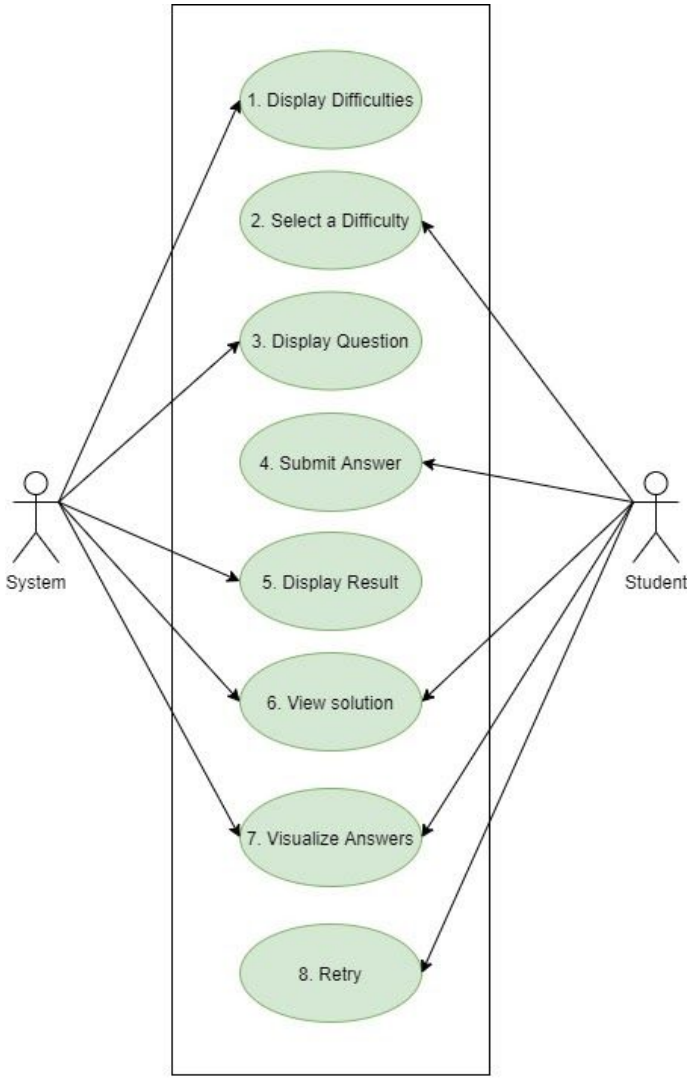


Level - 1.6.1.1

Primary Actor: Student
Secondary Actor: System
Description: Student can practice in any category as his/her wish.

Level 10

Use case Id: 10

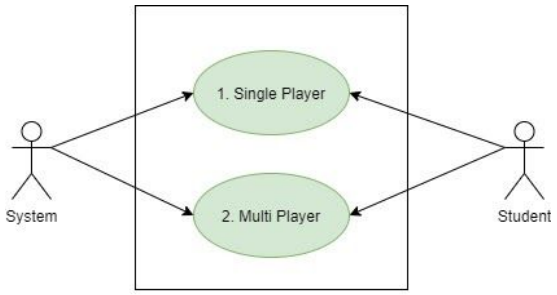


Level - 1.6.1.2

Use case Name: Practice by Difficulty
Primary Actor: Student
Secondary Actor: System
Description: Students can practice by choosing any difficulty, submit answers and see solutions by visualizing answers.

Level 11

Use case Id: 11

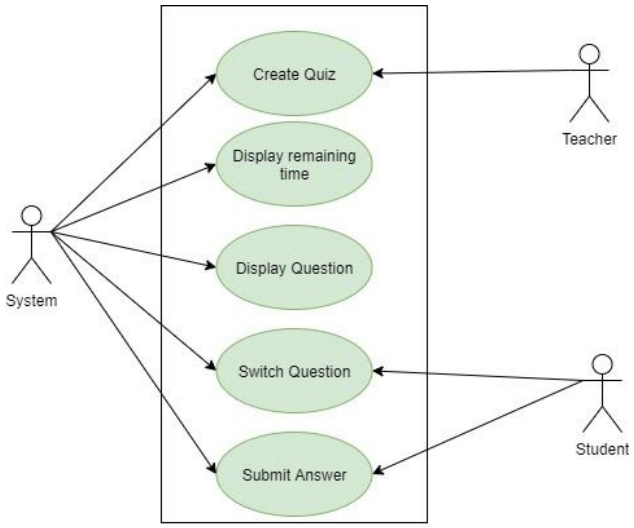


Level - 1.6.2

Use case Name: Exam Categories
Primary Actor: Student
Secondary Actor: System
Description: Student can give exam himself/herself only, as well as competing with other students.

Level 12

Use case Id: 12

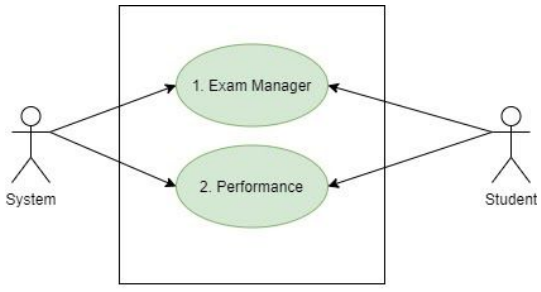


Level - 1.2.4

Use case Name: Exam Categories
Primary Actor: Teacher & Student
Secondary Actor: System
Description: This is a single player exam module which is accessible to both teachers and students.

Level 13

Use case Id: 13
Use case Name: Multiplayer Exam Overview

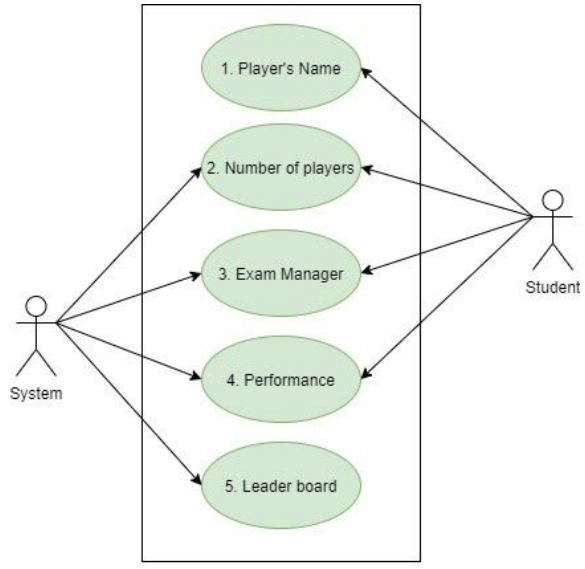


Level - 1.6.2.1

Primary Actor: System
Secondary Actor: Student
Description: Application can take exams and the performance of the user is shown after the exam.

Level 14

Use case Id: 14

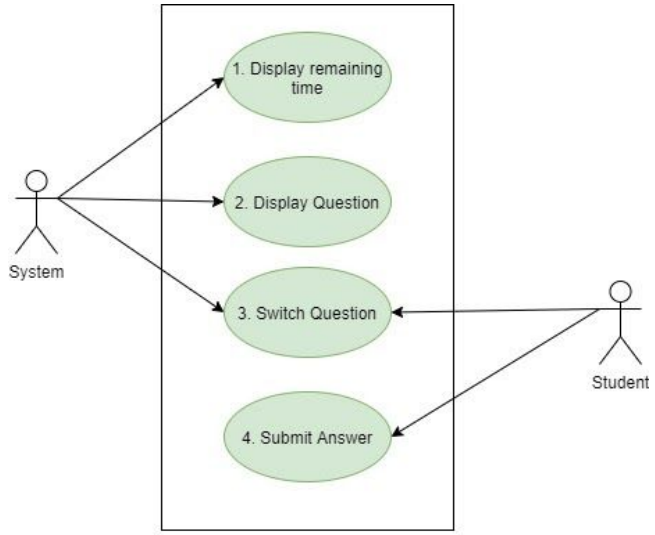


Level - 1.6.2.2

Use case Name: Multiplayer Exam Structure
Primary Actor: Student
Secondary Actor: System
Description: The preliminary information regarding the exam can be selected by the user.

Level 15

Use case Id: 15

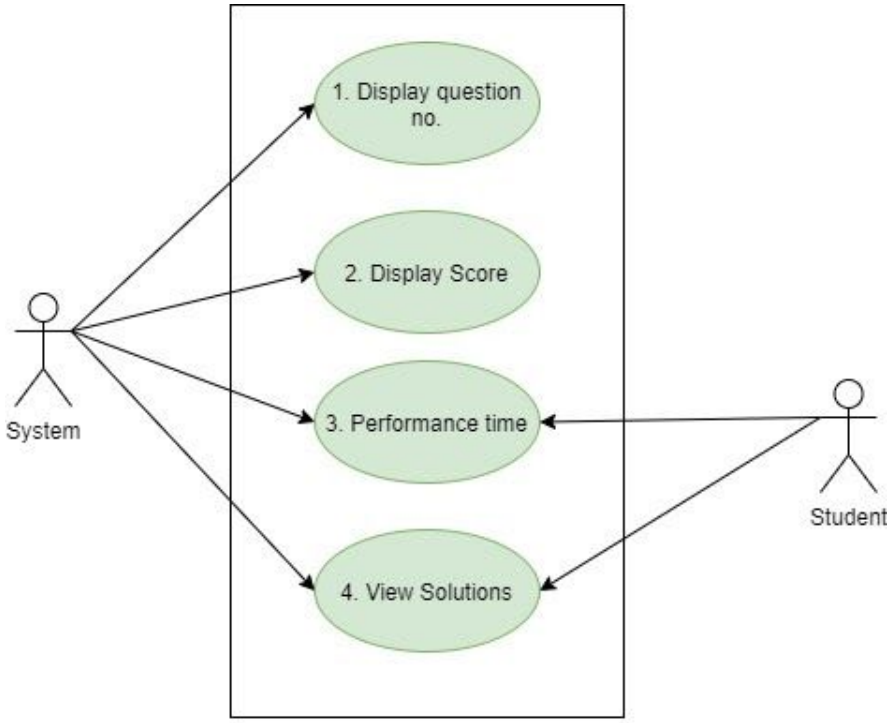


Level - 1.6.2.1.1

Use case Name: Multiplayer Exam answering questions
Primary Actor: Student
Secondary Actor: System
Description: The student can do various activities regarding the exam.

Level 16

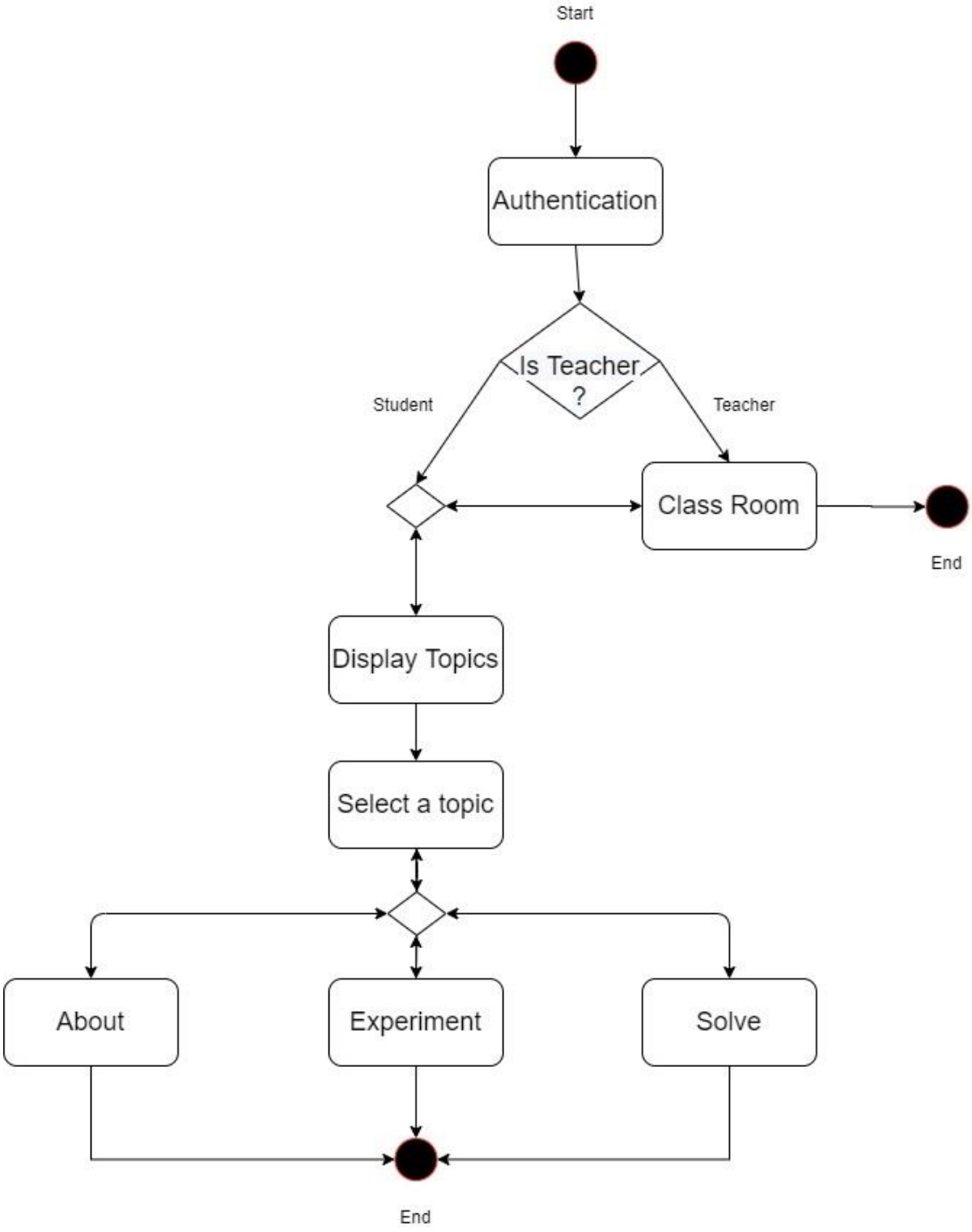
Use case Id: 16
Use case Name: Performance evaluation



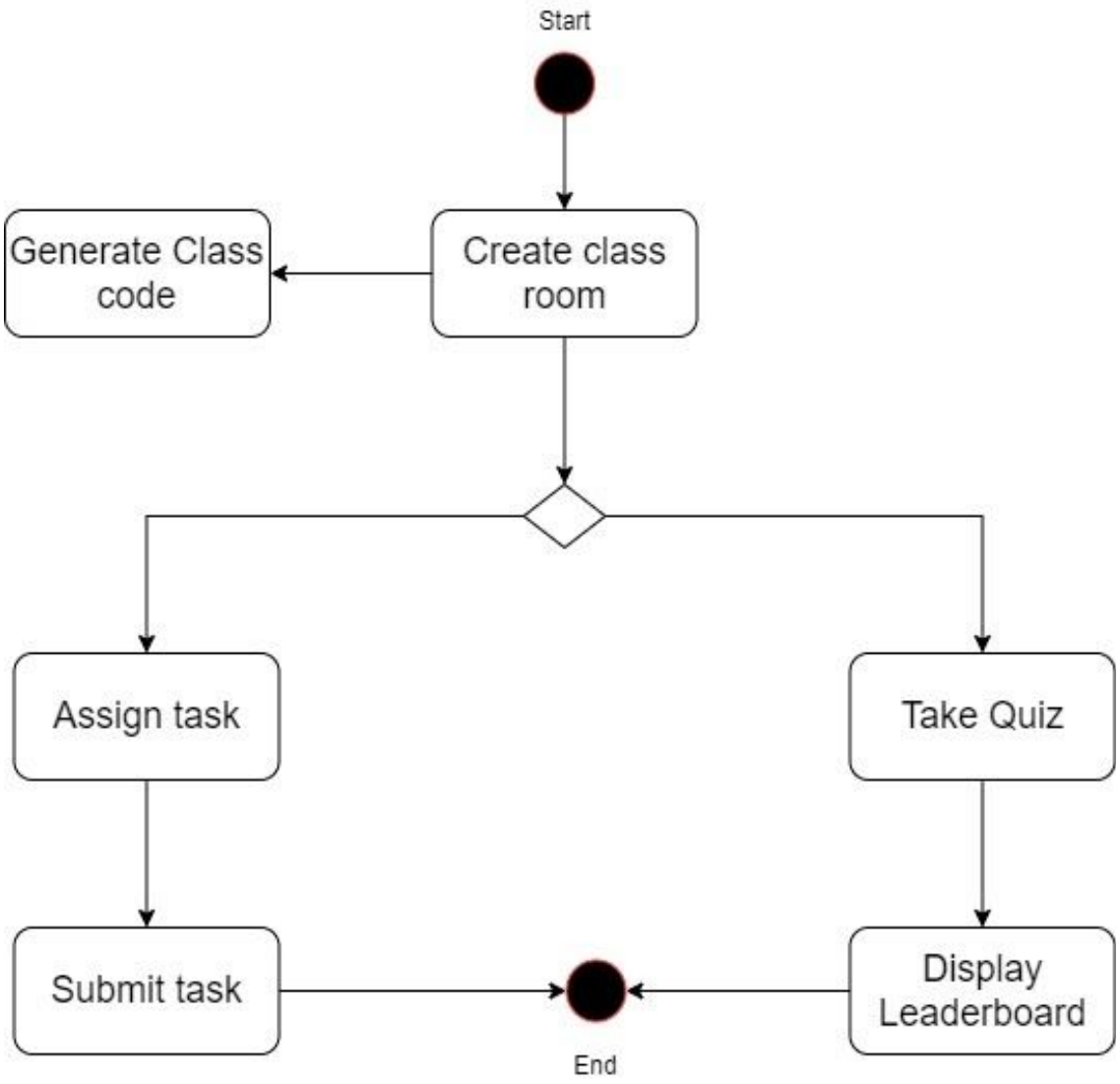
Level - 1.6.2.1.2

Primary Actor: Student
Secondary Actor: System
Description: Student's performance is evaluated here.

Activity Diagram:

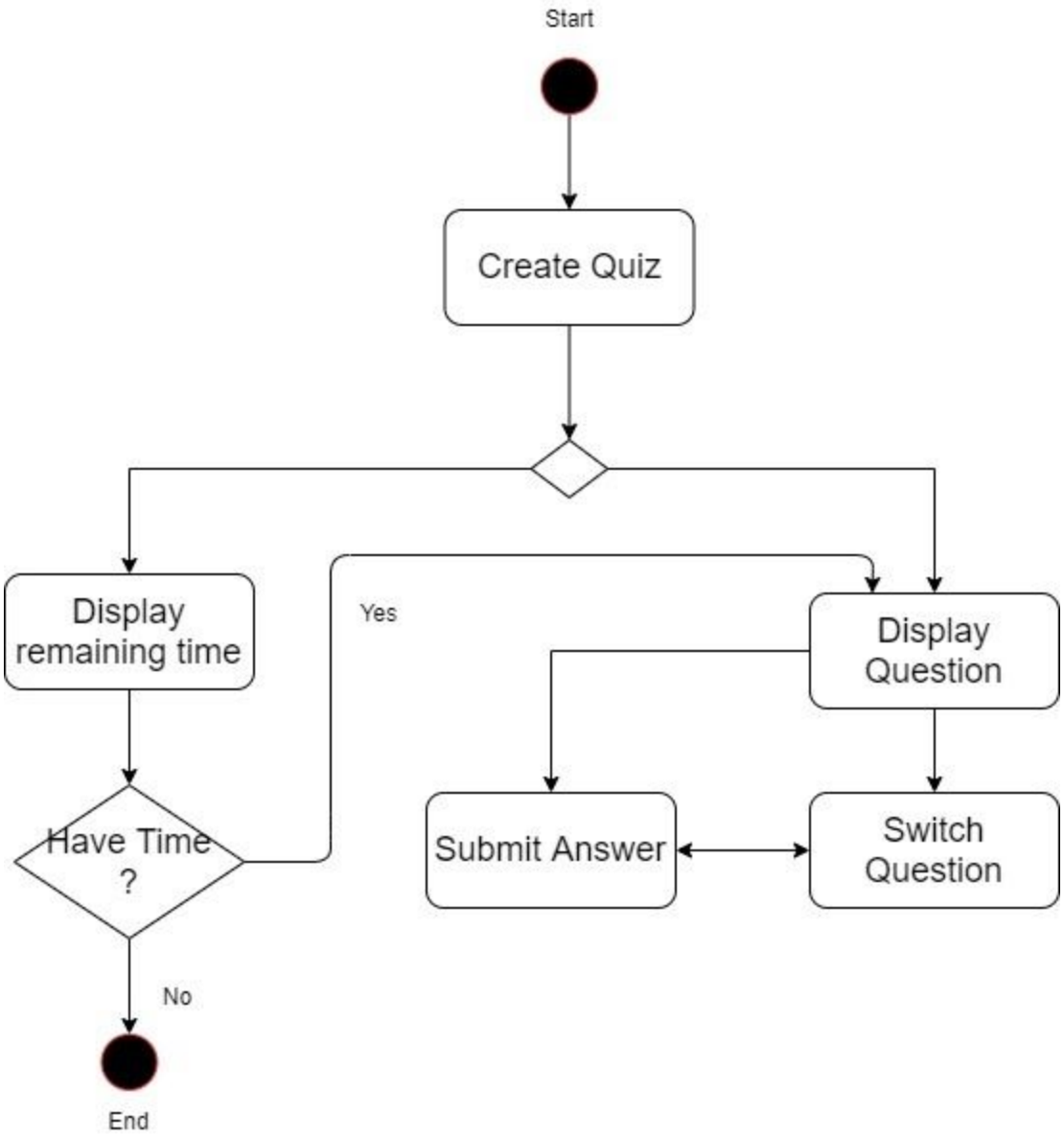


Description: User can authenticate his/her profession.If he/she is a teacher ,he/she can create and manage classroom related sections.
If a user is a student he/she can select his/her topics. In a particular topic , there are three sections namely about,experiment and solve.

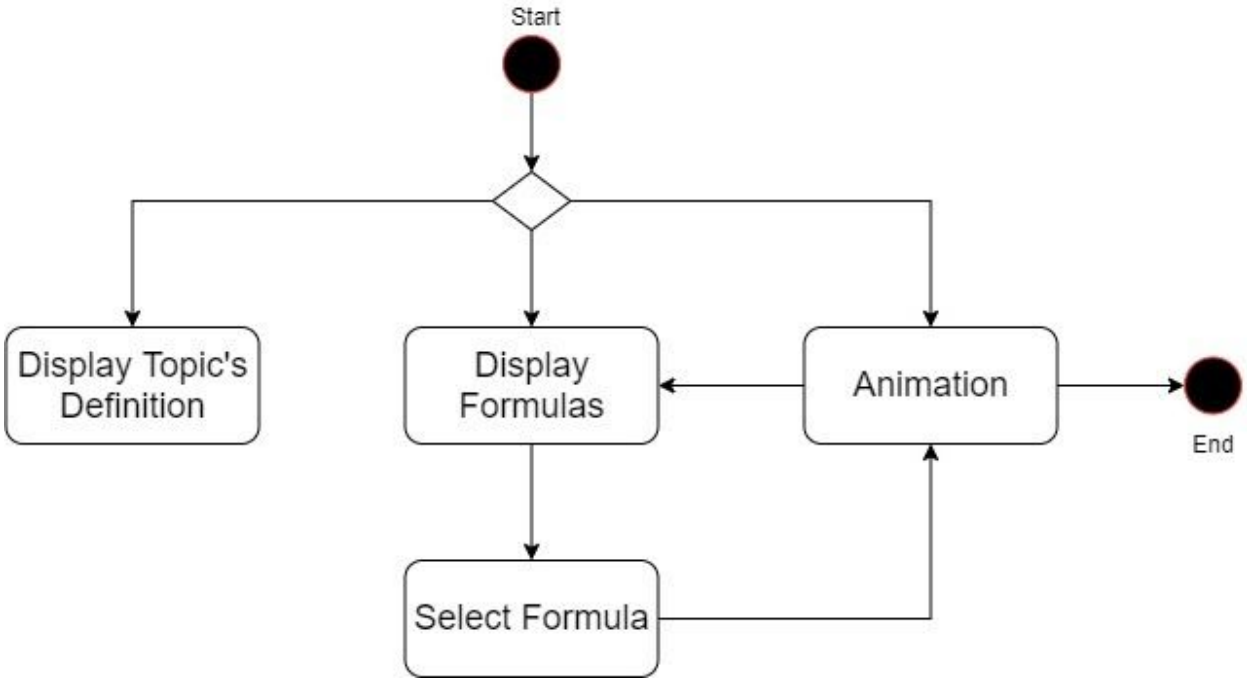


Description: In the virtual classroom section, teachers can create a class and students can enter the class using class code.There teacher can

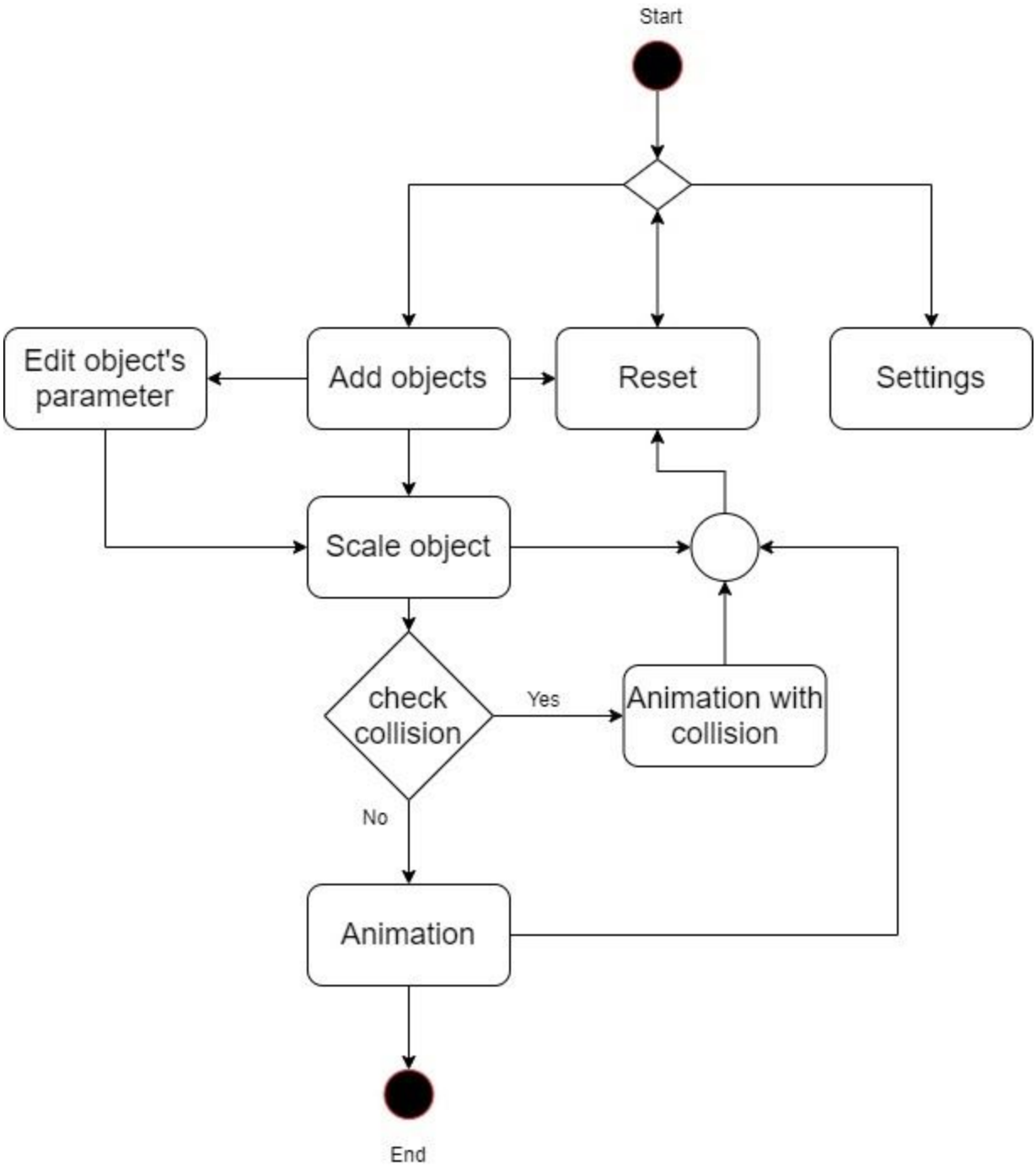
assign tasks & take quizzes where students can submit their class task. After taking the quiz, the teacher can evaluate students' performance.



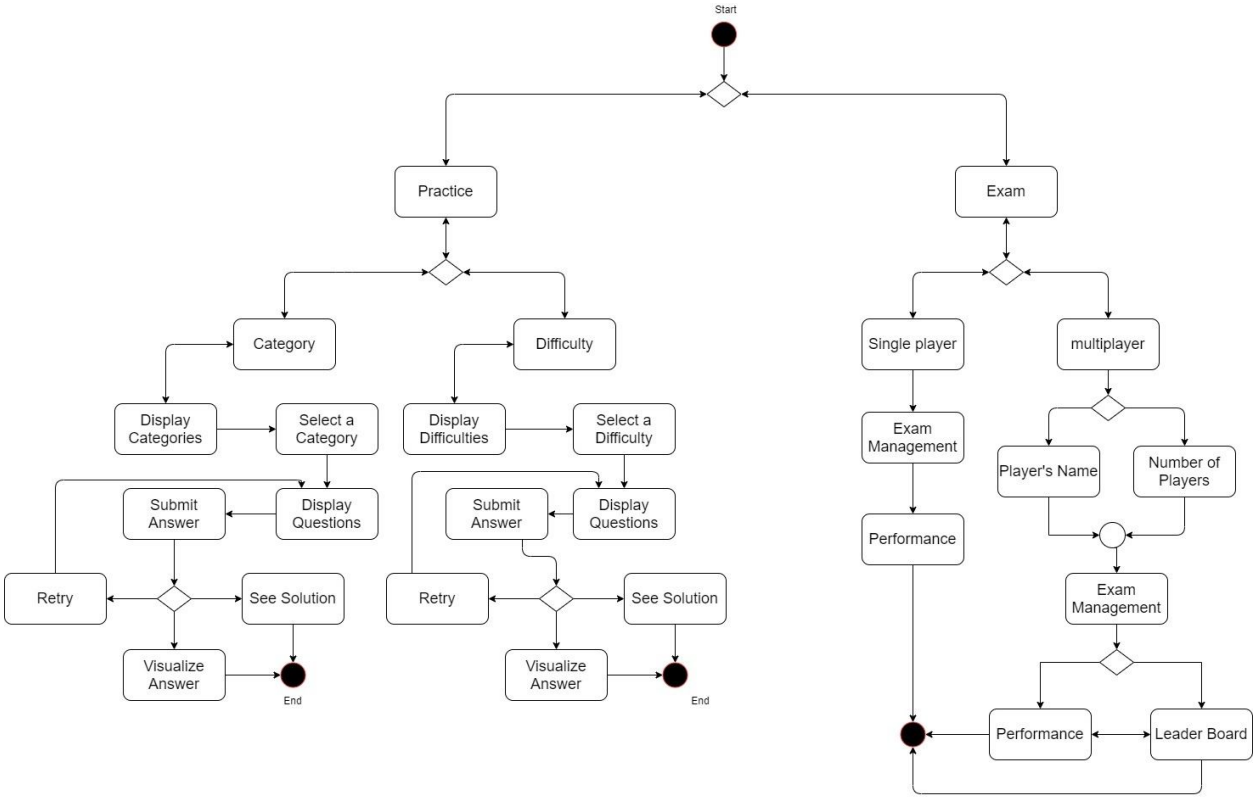
Description: If we dive into our create quiz section, a set of questions is to be solved within a certain time. Students can skip questions and evaluate his/her learnings.



Description: Students can select their topic and gain knowledge about the topic using definition, formulation and animation.



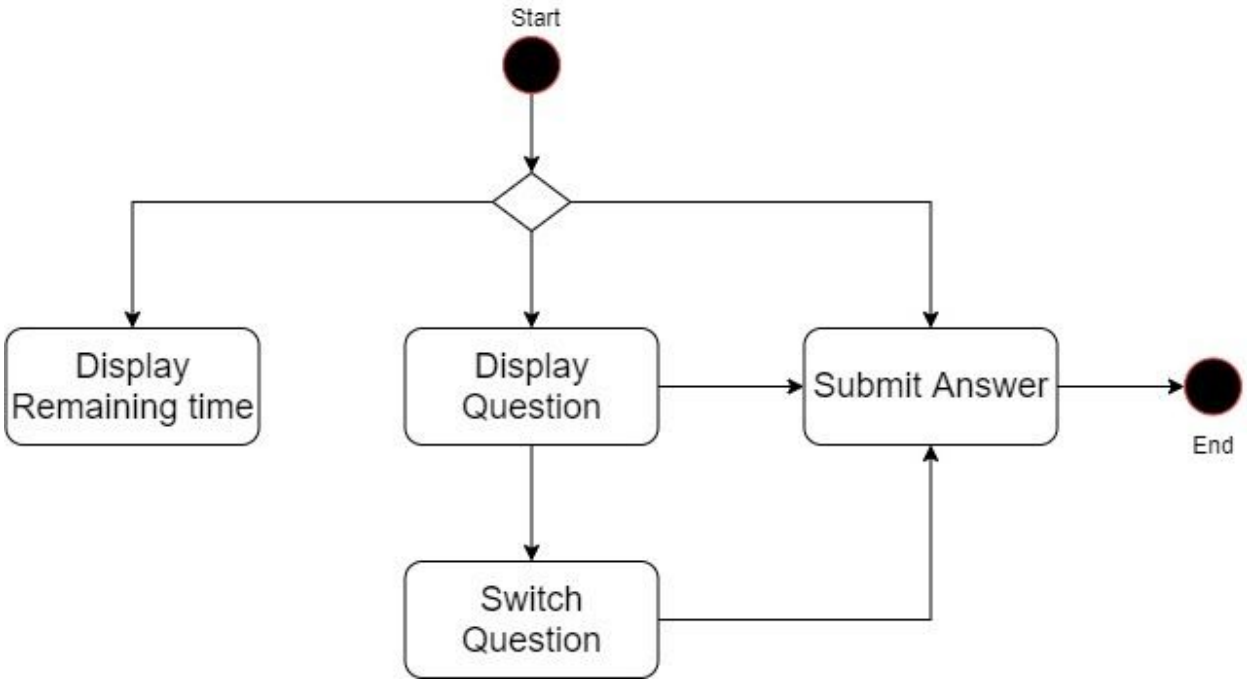
Description: In the experiment section, a student can edit an object's parameter where he/she can add objects, reset objects, scale objects.He/she can check collision between objects and obstacles. He/she can start animation and pause anytime to the current condition of the object (ex: velocity, time).



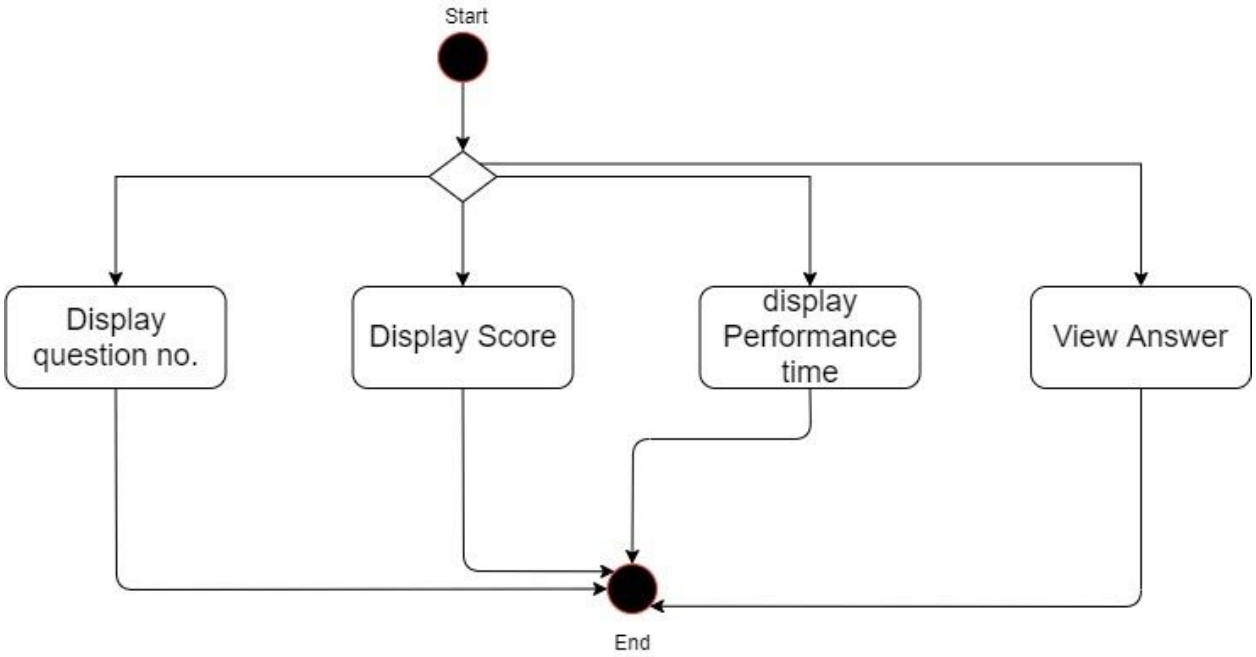
Description: In the "Solve" section, the user can evaluate her learnings'.There are two parts in the "Solve" section, "Practice" and "Exam"."Practice" is of two types, Category based and difficulty based. Category means any particular variable dependent (height, range, velocity etc). Difficulty consists of easy, medium and hard. In practice, a question will appear and the user has to answer it. after answering, she will see the correct answer. And if the user gives the wrong answer, solution and animation of both the wrong and correct answer will be given.

Then lastly, Exam is two types, single player and multiplayer. Single player is user adaptive, by giving correct answers users will be given harder questions than the previous. In multiplayer users can play with

friends, at most 10 people at a time. Everyone will be given the same question to answer within a particular time.



Description: During the single player examination, a timer counts the remaining time and the user can switch between questions and submit answers.



Description: During multiplayer mode, the examination interface displays questions and a scoreboard is presented showing various performance parameters.

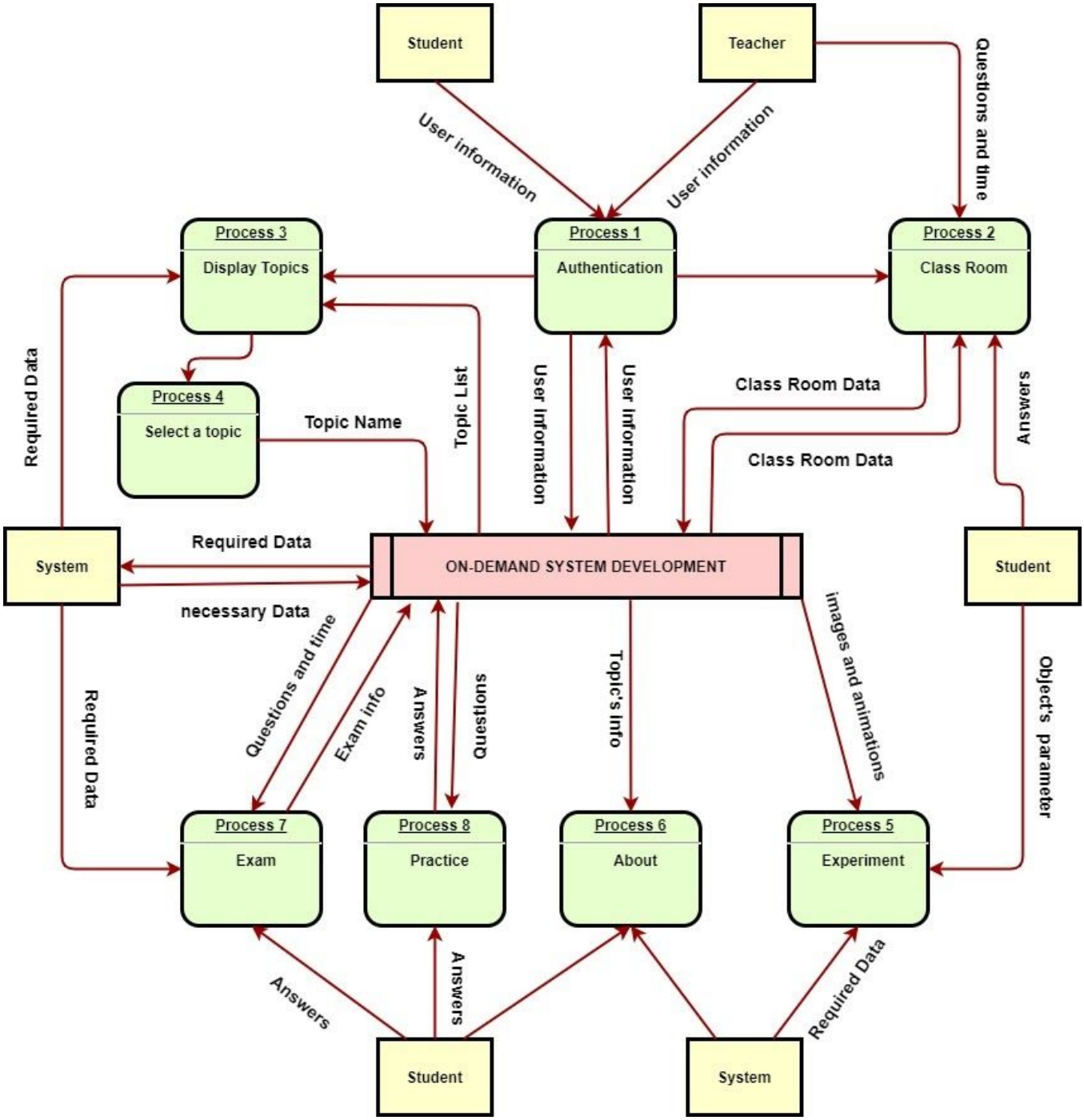
Data Flow Diagram (DFD)

Definition :

A Data Flow Diagram (DFD) is a traditional visual representation of the information flows within a system.It shows how data enters and leaves the system, what changes the information, and where data is stored.

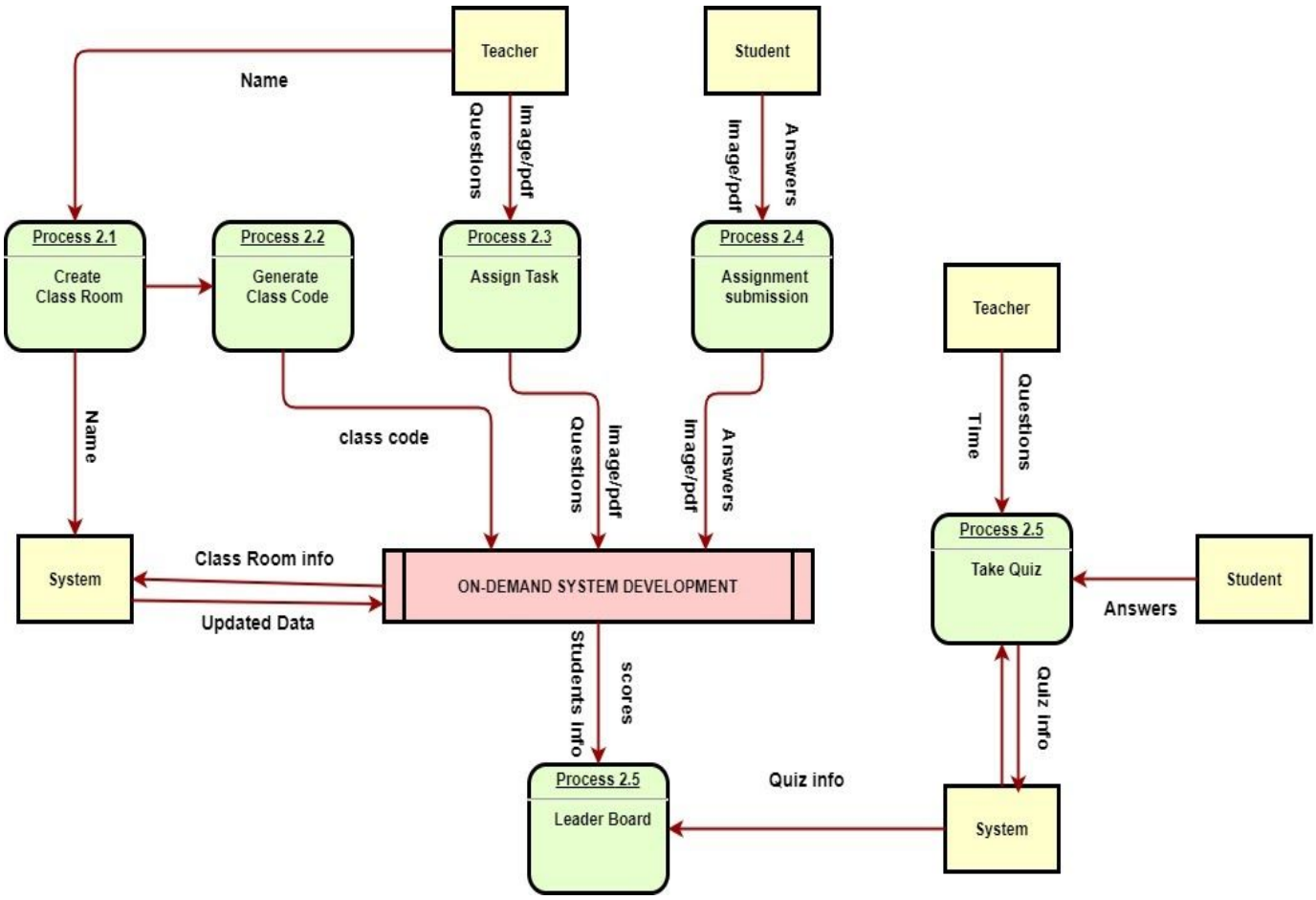
0-level DFD:

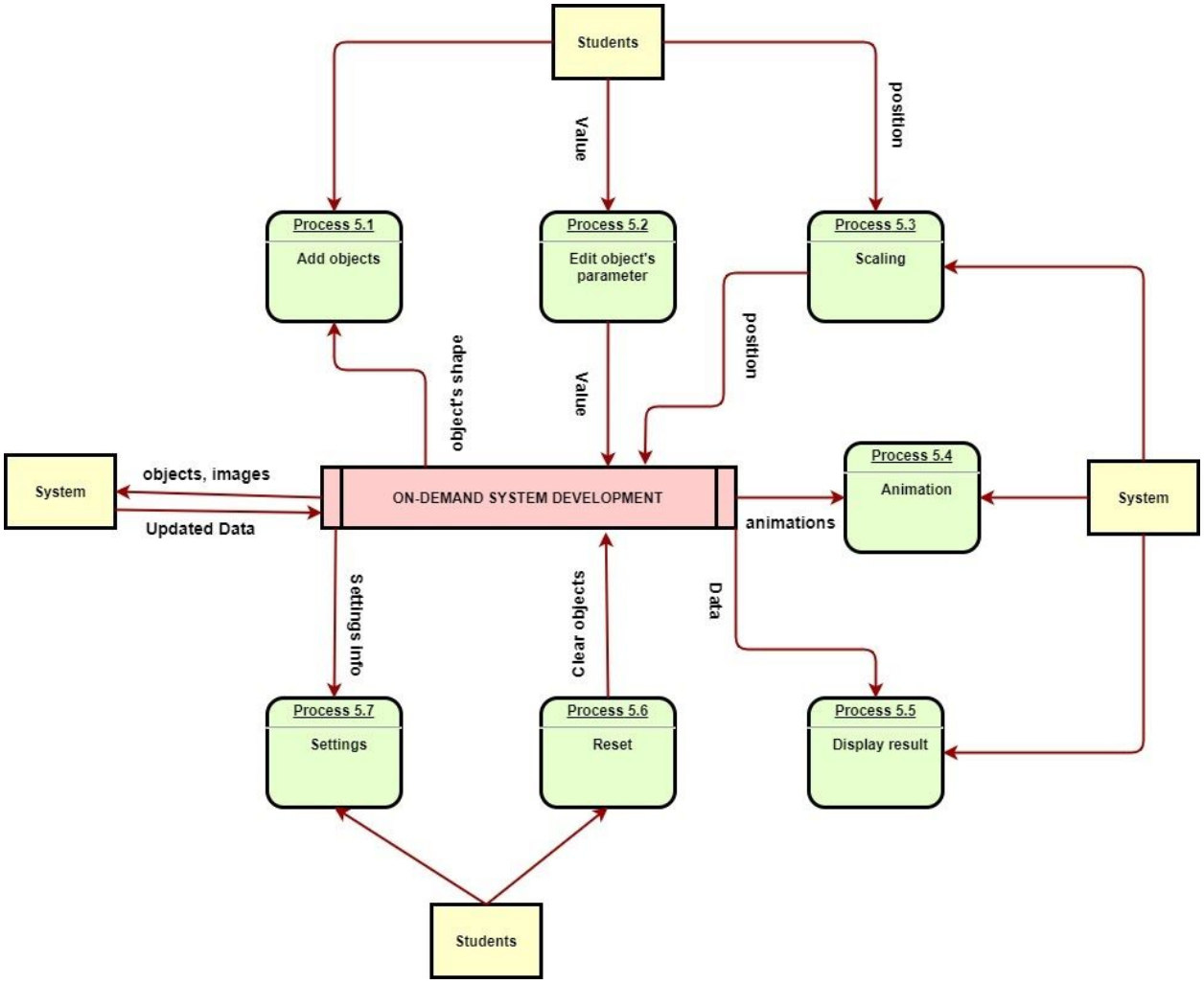
It’s designed to be an abstraction view, showing the system as a single process with its relationship to external entities.

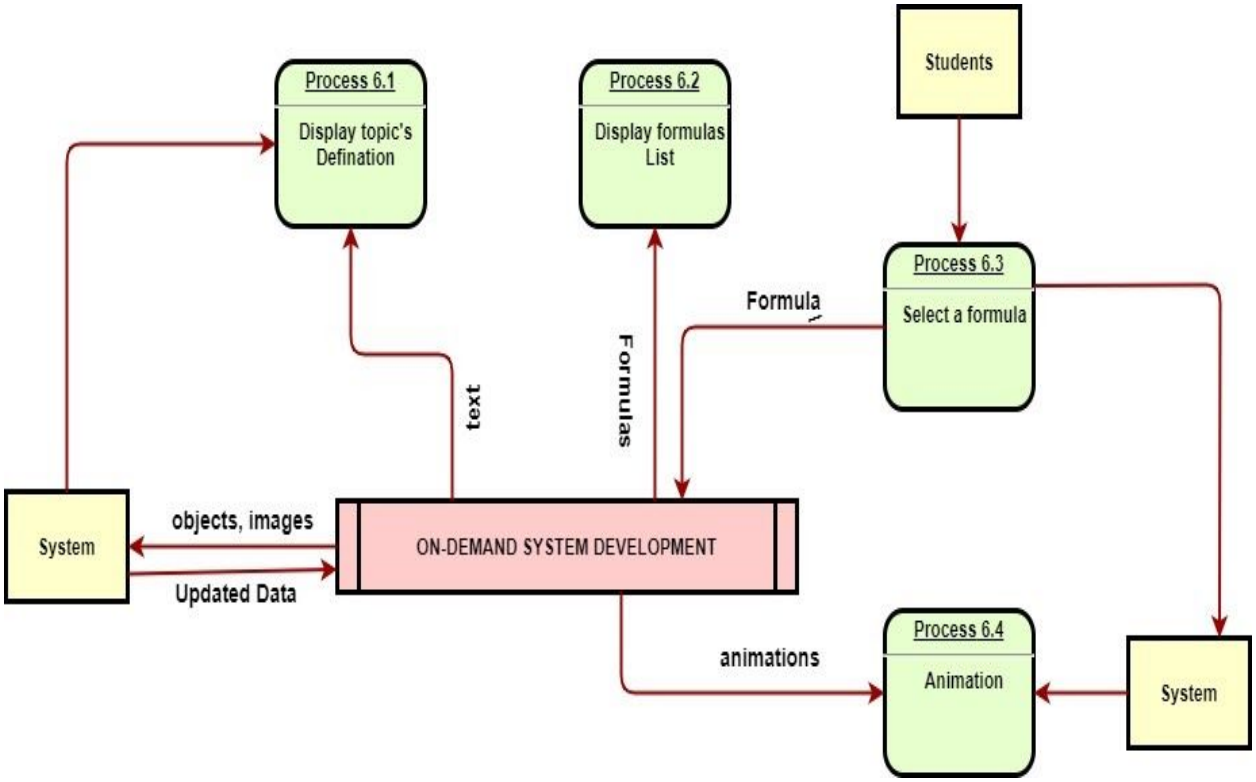

Student
Teacher
User information
User information
Questions and time
Process 3
Display Topics
Process 1
Authentication
Process 2
Class Room
Required Data
Process 4
Select a topic
Topic Name
Topic List
User Information
User Information
Class Room Data
Class Room Data
Answers
System
Required Data
ON-DEMAND SYSTEM DEVELOPMENT
Student
necessary Data
Questions and time
Exam info
Answers
Questions
Topic's Info
images and animations
Required Data
Object's parameter
Process 7
Exam
Process 8
Practice
Process 6
About
Process 5
Experiment
Answers
Answers
Required Data
Student
System

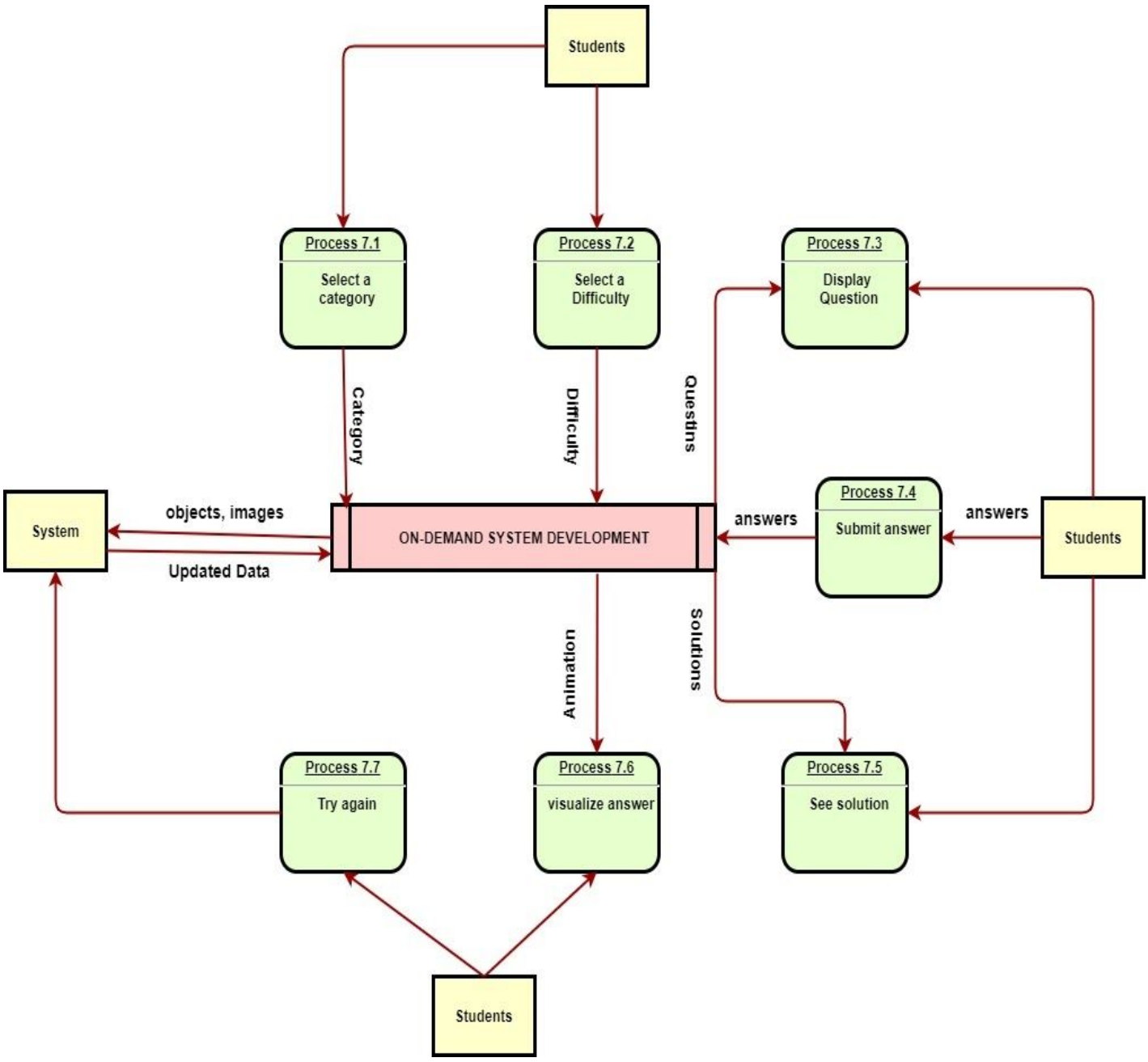
1-level DFD:

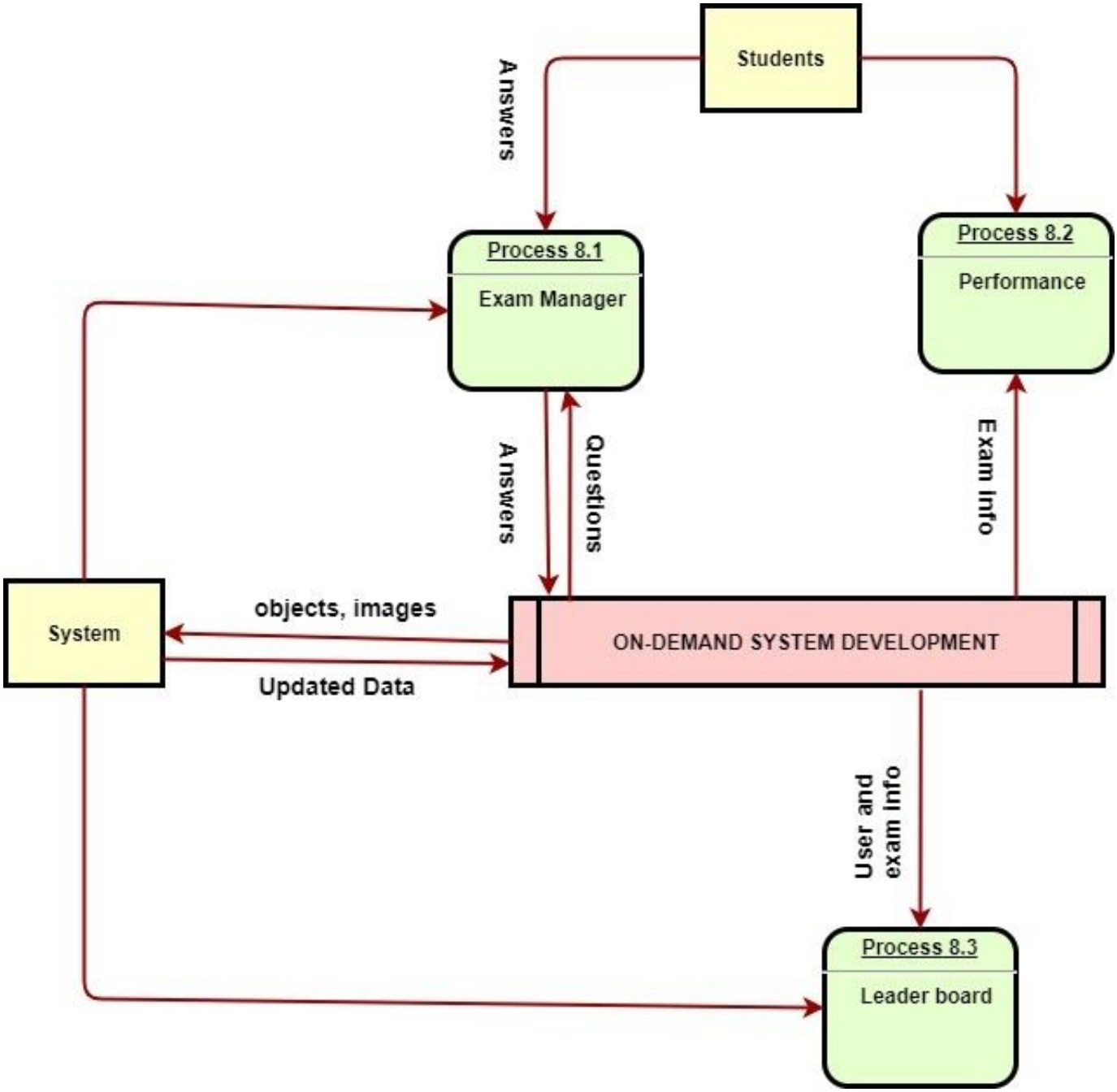
In this level we highlight the main functions of the system and break down the high level process of 0-level DFD into subprocesses.




Students
Value
position
Process 5.1
Add objects
Process 5.2
Edit object's parameter
Process 5.3
Scaling
object's shape
Value
position
System
objects, images
Updated Data
ON-DEMAND SYSTEM DEVELOPMENT
Process 5.4
Animation
animations
System
Settings info
Clear objects
Data
Process 5.7
Settings
Process 5.6
Reset
Process 5.5
Display result
Students

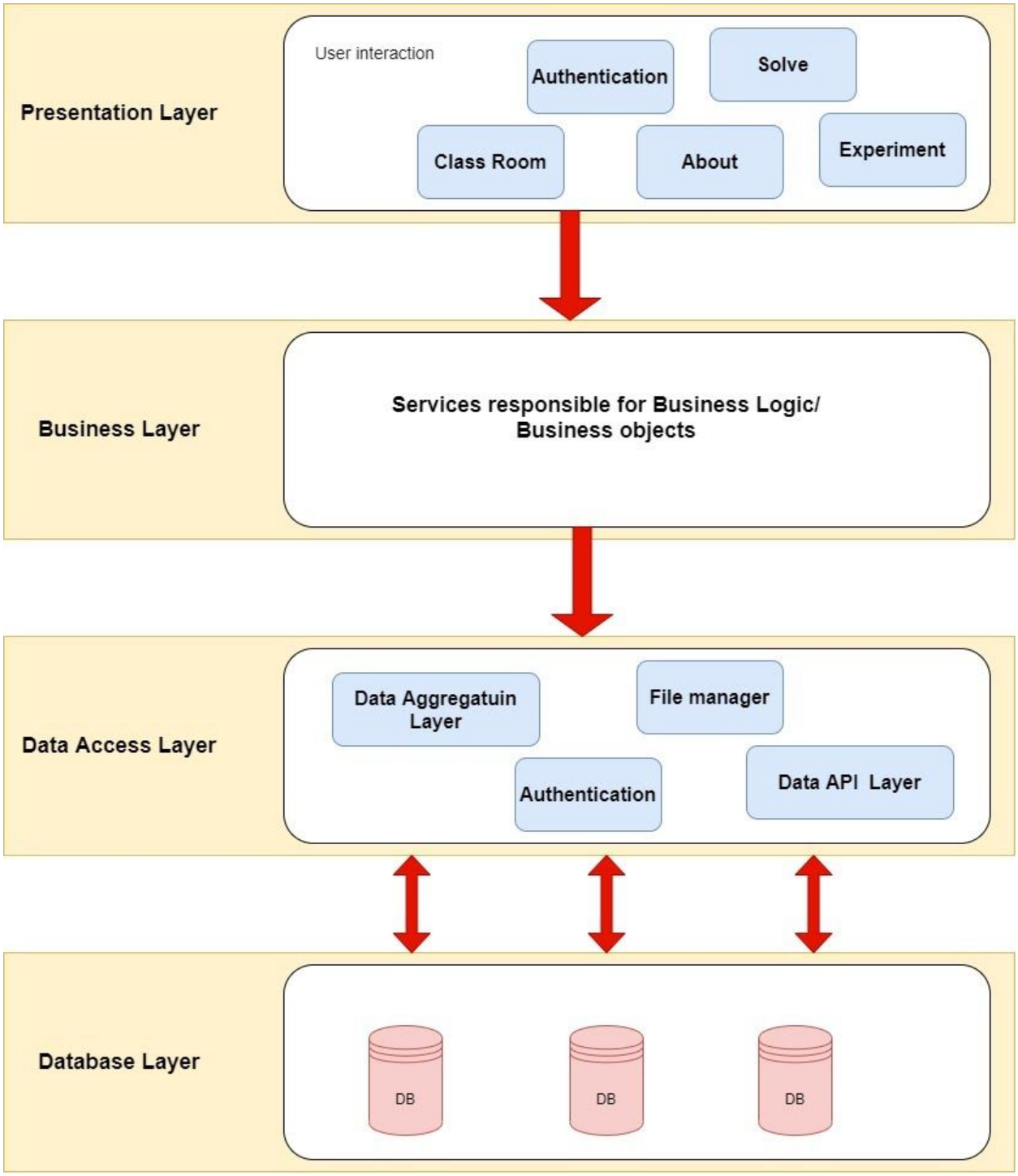

Process 6.1
Display topic's Defination
Process 6.2
Display formulas List
Students
Process 6.3
Select a formula
Formula
text
Formulas
System
objects, images
Updated Data
ON-DEMAND SYSTEM DEVELOPMENT
animations
Process 6.4
Animation
System


Students
Process 7.1
Select a category
Process 7.2
Select a Difficulty
Process 7.3
Display Question
Category
Difficulty
Questions
System
objects, images
Updated Data
ON-DEMAND SYSTEM DEVELOPMENT
answers
Process 7.4
Submit answer
answers
Students
Animation
Solutions
Process 7.7
Try again
Process 7.6
visualize answer
Process 7.5
See solution
Students


Students
Answers
Process 8.1
Exam Manager
Process 8.2
Performance
Answers
Questions
Exam Info
System
objects, images
Updated Data
ON-DEMAND SYSTEM DEVELOPMENT
User and exam info
Process 8.3
Leader board

Architecture Diagram

Definition:
Architecture serves as a blueprint for a system. It provides an abstraction to manage the system complexity and establish a communication and coordination mechanism among components.It defines a structured solution to meet all the technical and operational requirements, while optimizing the common quality attributes like performance and security.


Presentation Layer
User interaction
Authentication
Solve
Class Room
About
Experiment
Business Layer
Services responsible for Business Logic/
Business objects
Data Access Layer
Data Aggregatuin Layer
File manager
Authentication
Data API Layer
Database Layer
DB
DB
DB