1. Introduction

1.1 Company Profile

**Company Name: -** Prakshal IT Academy

**Business Type: -** IT Development, Training and Academy

**Developed For: -** E-Education

**Address: -** Gulam Baba Mill Compound, B/H, Param Doctor House,

Near Railway Station, Lal Darwaja, Surat, Gujarat 395003

**Phone No: -** +91 97277 20806

**Email Id: -** [info@prakshal.com](mailto:info@prakshal.com)

**Website: - https://**[**www.prakshal.com/**](http://www.prakshal.com/)

 **About**

Established in the year of 2000, Prakshal is an ISO 9001:2015 certified firm that offers

certified courses in the field of Information Technology that open exciting opportunities for

people aspiring for a career in the IT industry.

1. Proposed System

2.1 Scope

In general, students are taking lecture and also going to classes for notes. And still they might get good notes or may be not. They also have to pay twice just for the materials.

In the case of college students also they are facing so many problems for notes and materials. If there is any institute which is running any courses, they are also providing notes and materials. They are also taking daily, weekly or monthly exams.

So, that is very time consuming to take exam physically at one place and that is taking too much paper waste.

So, using this application, Students can get materials provided by faculties directly. They can get notification of every exams and any other events by mail. So that they would not face any problem. As well as, they can also study anywhere and anytime through this application.

During the exam, there will be no paper waste and Students can give exam from home or anywhere else in some circumstances. And also get immediate result. This result can help Admin to generate certificate of that student. Faculty’s work load will be also reduce.

2.2 Objective

* To provide material easily to students anytime, anywhere.
* Students can give exam online without physically presence at institute.
* Reduce work load of faculties.
* Easy to know the progress of all the students.
* Paper waste will be reduced.

2.3 Constraints

The constraints associated in the development of any software system includes those that are common to all system. The common constraints may include lack of managerial partition, tighter development schedule etc.

Some of the constraints are listed below with which we have developed this system. It requires

* H/W Constraints
  + 4GB RAM
  + 1TB HDD
  + Intel Core i3
* S/W Constraints
* Windows 10
* Sublime Editor
* PhpMyAdmin
* Android Studio

2.4 Advantages

* By using this, the amount of paperwork will be reduced.
* All the materials will be saved on the server. So, there will be no issue of server.
* All the Data’s can be easily accessed by the organization.

2.5 Limitations

* The chances of cheating during exam are increased.

1. Environment Specification

3.1 Hardware and Software Requirements

* **Hardware Requirement: -**

Processor Speed : Intel Core I3 with 2.00 GHz

RAM : 4.00 GB or Higher

Hard Disk : 80GB or Higher

* **Software Requirement**: -

**Server-Side Software:**

Operating System **:** Windows 10

Language **:** PHP with Codeignitor

Technologies **:** HTML5, jQuery, AJAX, CSS3, Bootstrap

Database **:** PhpMyAdmin

Tools **:** Sublime Text, XAMPP

**Client-Side Software:**

Operating System **:** Android

1. System Planning

4.1 Feasibility Study

Before starting the development of the system, one should give considerable amount of time for feasibility study because the successful completion of the project depends upon feasibility.

The feasibility of our project has been judged on the basis of time, technology, resources available and project length.

* **Time: -**

This project takes at least 4 months to be completed if we take help of reuses components otherwise it will take 6 months to be completed.

We will not make use of components and therefore will be able to complete the project in 6 months. Thus, according to time feasibility is not that right.

* **Technology: -**

The necessary technology, front-end development tool, back-end database technology and various other tools namely installation tools, etc. For developing the system, are already available within the organization. So, this problem is feasible.

* **Resources: -**

We need good knowledge software engineers and practitioners. We need Net connections. We have all the resources in desired amount.

* **Project Size: -**

The project size might be about 10000 LOC. This is just the rough assumption because we don’t have any basis of the past projects.

Thus, the project overall feasibility is normal and therefore we have undertaken this project.

4.2 Software Engineering Model

* We followed Agile Development Model for our system so that we can add the functionality as the requirement comes.
* We were receiving the requirements and as per that developing the system till end.



4.3 Risk Analysis

**Project risks can be identified by answering following questions,**

* What can go wrong?
* What is the likelihood?
* What will the damage be?
* What can we do about it?

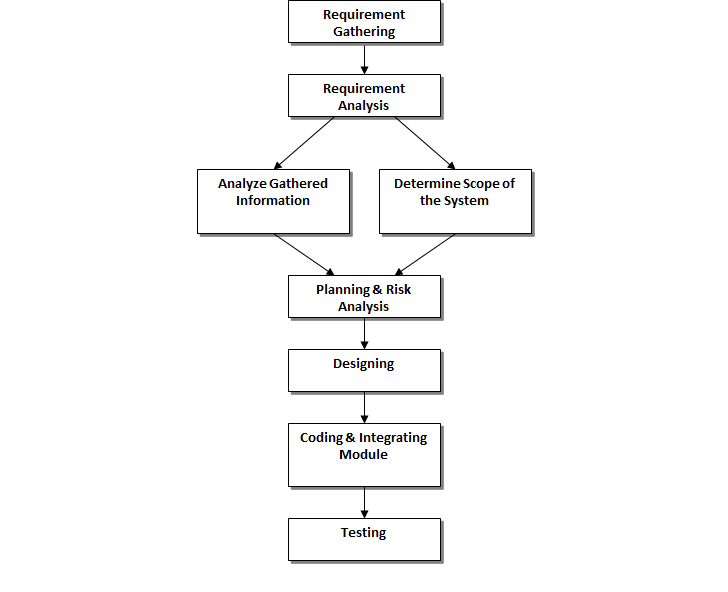
**Following are the risk types:**

* Product size - risks associated with the overall size of the software to be built or modified.
* Business impact - risks associated with constraints imposed by management or the marketplace.
* Customer characteristics - risks associated with the sophistication of the customer and the developer's ability to communicate with the customer in a timely manner
* Process definition - risks associated with the degree to which the software process has been defined and is followed by the development organization.
* Development environment - risks associated with the availability and quality of the tools to be used to build the product.
* Technology to be built - risks associated with the complexity of the system to be built and the "newness" of the technology that is packaged by the system.

Staff size and experience - risks associated with the overall technical and project experience of the software engineers who will do the work

4.4 Project Schedule

**4.4.1. Task Dependency**

****

**4.4.2. Timeline Chart**

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Week** | **1** | **2** | **3** | **4** | **5** | **6** | **7** | **8** | **9** | **10** | **11** | **12** | **13** | **14** | **15** |
| **Task Name** |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| **1)Requirement Analysis** |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| Collected requirement from our guide |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| Analyse gathered information |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| Determine different modules |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| **2)Planning & Risk**  **Analysis** |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| Analyse data for possible risk |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| Identify technical risk |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| Determine different modules |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| **3) Designing** |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| Design basic interface of the site |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| Design database tables |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| Design web form for modules |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| **4)Coding and integrating**  **modules** |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| Implement logic for different modules |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| Implement database connectivity |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| Implement report |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| integrate different modules |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| **5)Testing** |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| Validate input control and check accuracy of Reports |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |

**4.4.3. Project Table**

|  |  |
| --- | --- |
| **Sr. No.** | **TABLE NAME** |
| 1. | Admin |
| 2. | Category |
| 3. | City |
| 4. | Course |
| 5. | Contact\_Us |
| 6. | Events |
| 7. | Exam\_Schedule |
| 8. | Exam\_Type |
| 9. | Material |
| 10 | Notification |
| 11 | Questions |
| 12 | Results |
| 13 | State |
| 14 | Student\_Notification |
| 15 | Syllabus |
| 16 | Users |
| 17 | Users\_Course |

1. System Analysis

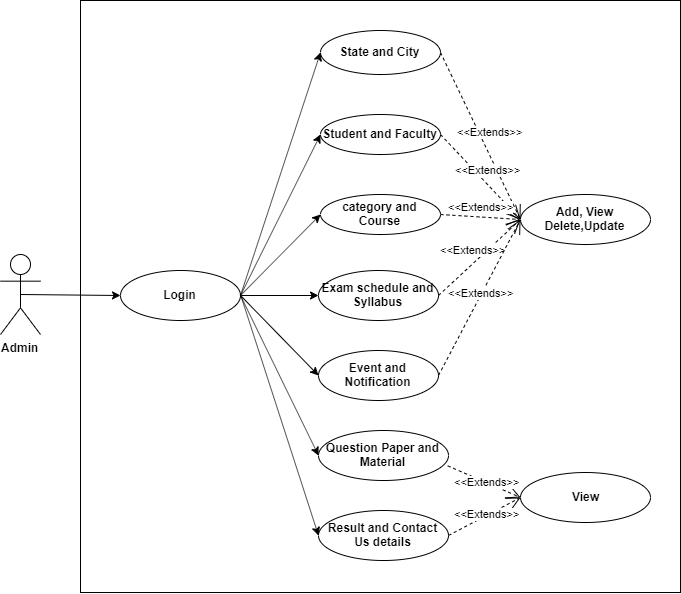
5.1 Detailed SRS

* **Admin** can Login
* Add new students and teacher’s details and send them their username and password by mail.
* Add, Update or Delete New City and State Details
* Add, Update or Delete New Category and Course Details
* Add, Update or Delete New Syllabus Of course Details and arrange exam schedule for courses.
* Add, Update and Delete New Events details.
* Add new Notification details and send them to students by mail.
* Can View all the questions of exam, all results and End user’s contact comments.
* **Teacher** can login and update own profile as well as password
* Add, update and delete material and question papers for courses
* View courses details and their exam schedule.
* **Students** can login and update ow profile as well as password
* View course details as well as view and download materials of selected course.
* Give exam and get marks to see own progress
* View all events details.

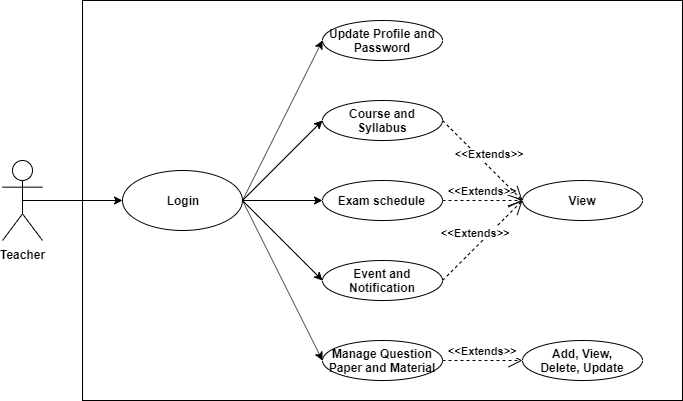
5.2 UML Diagrams

**5.2.1 Use Case Diagrams**

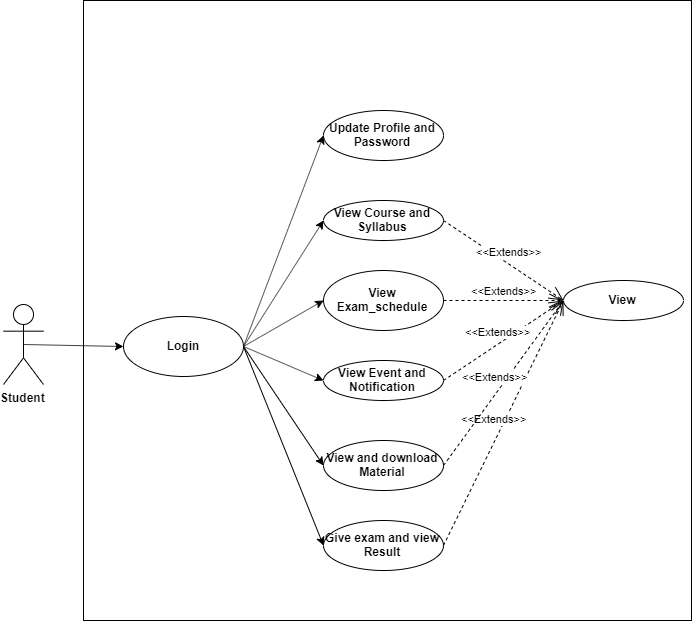
**1. Use Case for Admin**



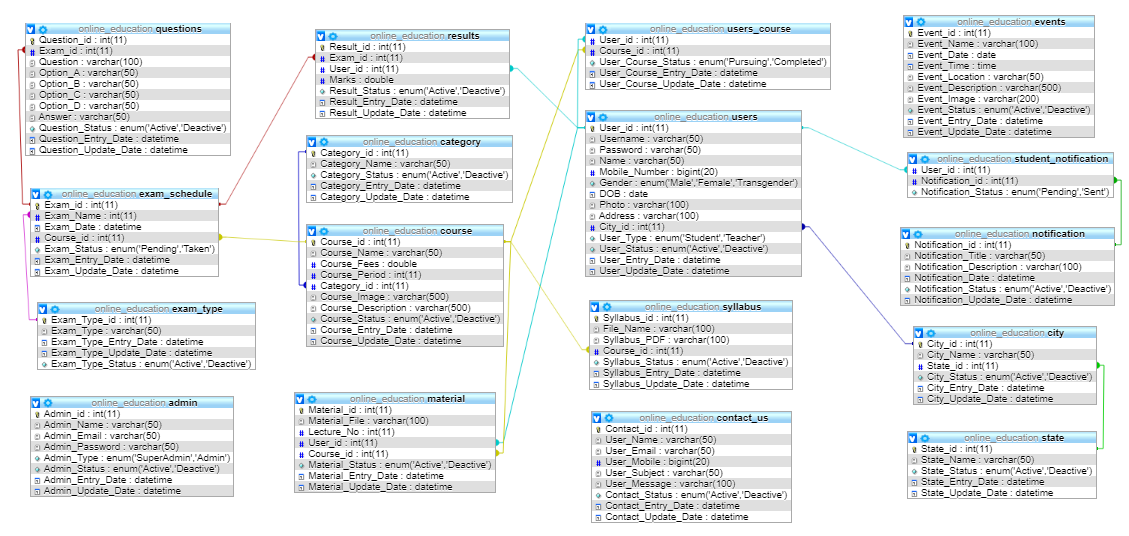
**2. Use case for Teacher**

****

**3. Use case for Student**

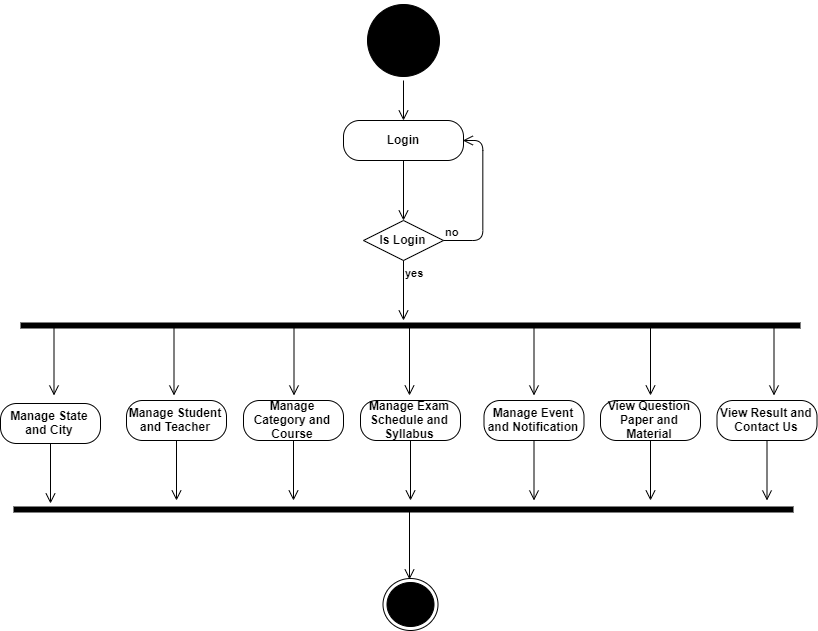
****

**5.2.2 Class Diagrams**

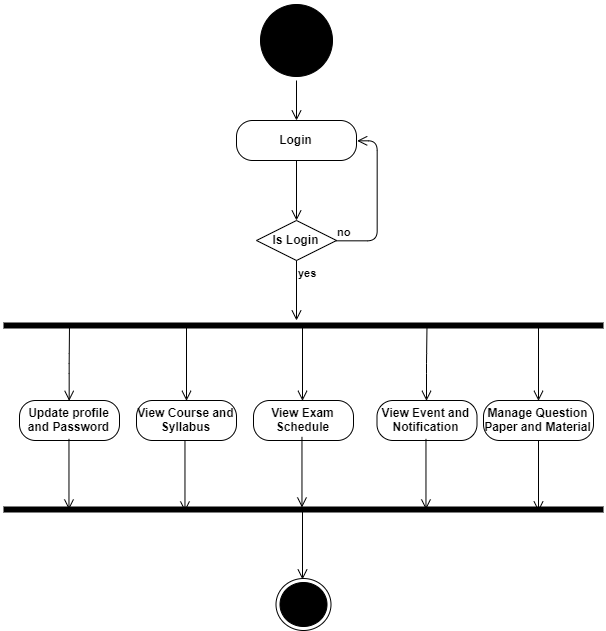
****

**5.2.3 Activity Diagrams**

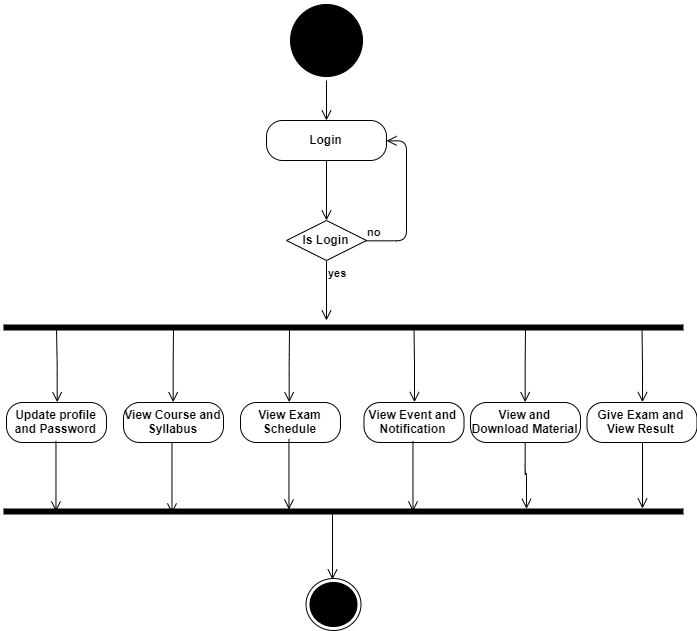
**5.2.3.1 Activity Diagram for Admin**



**5.2.3.2 Activity Diagram for Teacher**

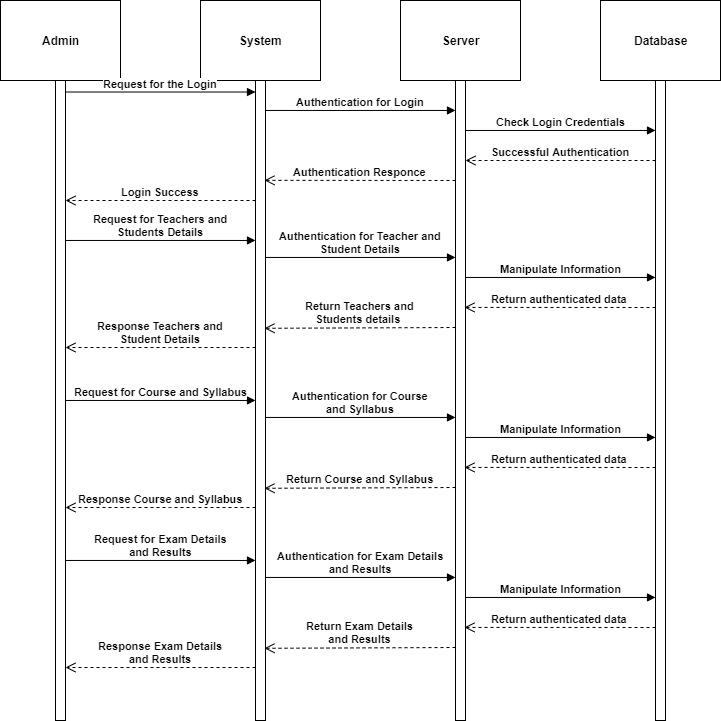


**5.2.3.3 Activity Diagram for Student**

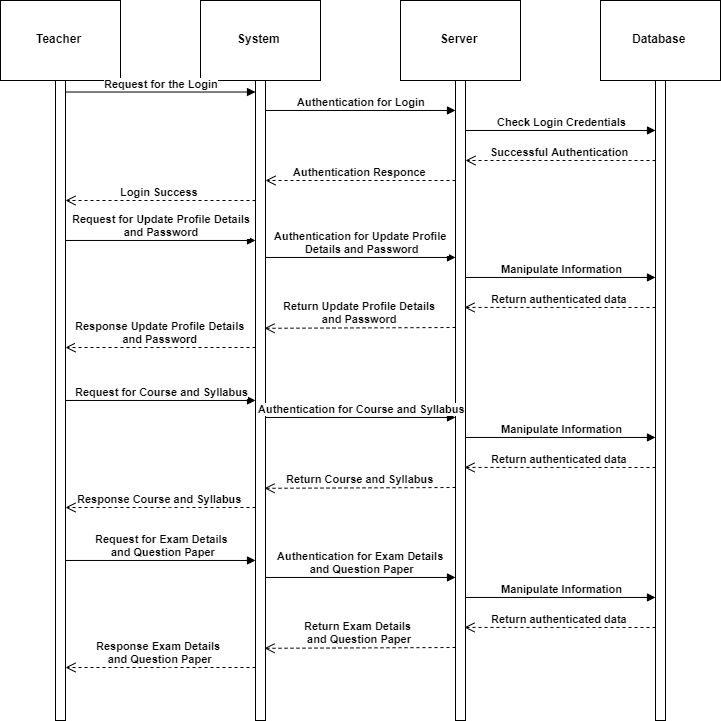
****

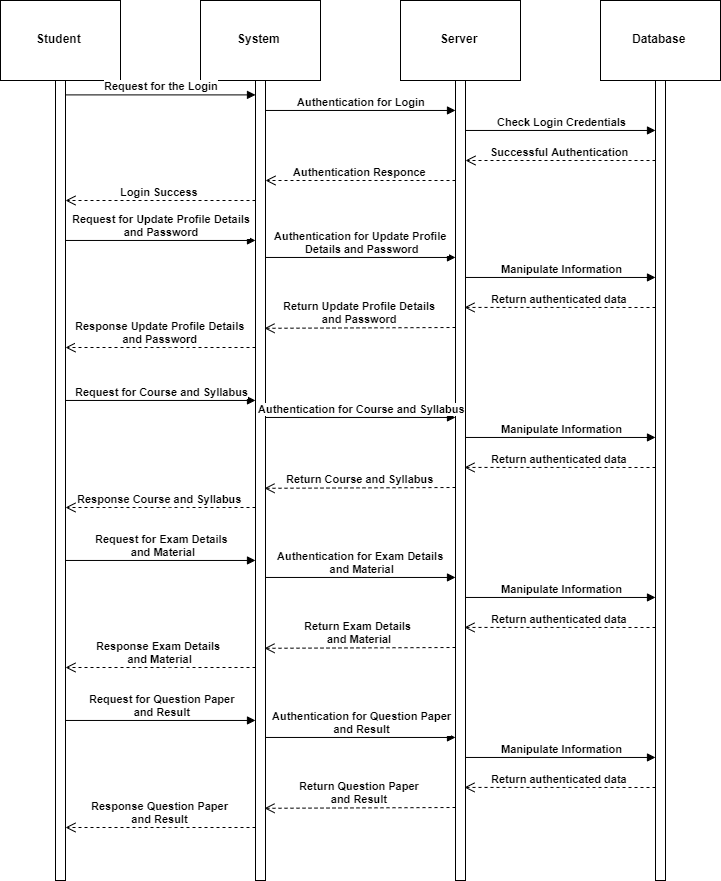
**5.2.4 Sequence Diagrams**

**5.2.4.1 Sequence Diagram for Admin**

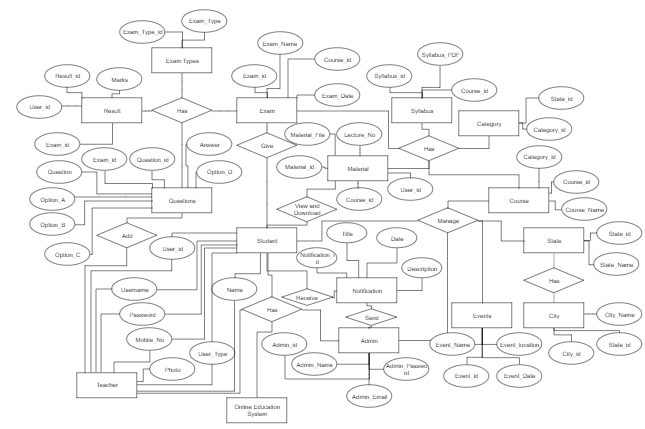
****

**5.2.4.2 Sequence Diagram for Teacher**

**5.2.4.3 Sequence Diagram for Student**



5.3 E-R Diagrams



1. Software Design

6.1 Database Design

1. Admin

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Sr. No** | **Column Name** | **Datatype** | **Constraint** | **Description** |
|  |  |  |  |  |
| 1. | Admin\_id | Integer | Primary Key | This is unique identification for the admin. |
| 2. | Admin\_Name | Varchar | - | This is the name for the Admin. |
| 3. | Admin\_Email | Varchar | - | This is the username of admin for login credentials. |
| 4. | Admin\_Password | Varchar | - | This is the password which is the secret field of particular Admin to login |
| 5. | Admin\_Type | Enum | - | This is the type of Admin. This may be superadmin or admin |
| 6. | Admin\_Status | Enum | - | This is status whether admin is active or Deactive |
| 7. | Admin\_Entry\_Date | Datetime | - | This is the date and time when the details are first time added. |
| 8. | Admin\_Update\_Date | Datetime | - | This is the date and time when the details are last time updated. |
|  |  |  |  |  |

1. Category

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Sr. No** | **Column Name** | **Datatype** | **Constraint** | **Description** |
|  |  |  |  |  |
| 1. | Category\_id | Integer | Primary Key | This is unique identification for the particular category of course. |
| 2. | Category\_Name | Varchar | - | This is the name for the category. |
| 3. | Category\_Status | Enum | - | This is status whether Category is active or Deactive |
| 4. | Category\_Entry\_Date | Datetime | - | This is the date and time when the Category is first time added. |
| 5. | Category\_Update\_Date | Datetime | - | This is the date and time when the category is last time updated. |

1. City

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Sr. No** | **Column Name** | **Datatype** | **Constraint** | **Description** |
|  |  |  |  |  |
| 1. | City\_id | Integer | Primary Key | This is unique identification for the City. |
| 2. | City\_Name | Integer |  | This is the name of city. |
| 3. | State\_id | Integer | Foreign Key | This is the reference key of state. |
| 4. | City\_Status | Enum | - | This is status whether City is active or Deactive |
| 5. | City\_Entry\_Date | Datetime | - | This is the date and time when the City is first time added. |
| 6. | City\_Update\_Date | Datetime | - | This is the date and time when the City is last time updated. |

1. Contact\_US

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Sr. No** | **Column Name** | **Datatype** | **Constraint** | **Description** |
|  |  |  |  |  |
| 1. | Contact\_id | Integer | Primary Key | This is unique identification for the Contact comments details. |
| 2. | User\_Name | Varchar | - | This is end user’s name |
| 3. | User\_Email | Varchar | - | This is end user’s email |
| 4. | User\_Mobile | Bigint | - | This is end user’s mobile no. |
| 5. | User\_Subject | Varchar | - | This is the subject. |
| 6. | User\_Message | Varchar | - | This is the message thar end user will comment to us. |
| 7. | Contact\_Status | Enum | - | This is status whether Contact is active or Deactive to show. |
| 8. | Contact\_Entry\_Date | Datetime | - | This is the date and time when the Contact is first time added. |
| 9. | Contact\_Update\_Date | Datetime | - | This is the date and time when the Contact is last time updated. |

1. Course

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Sr. No** | **Column Name** | **Datatype** | **Constraint** | **Description** |
|  |  |  |  |  |
| 1. | Course\_id | Integer | Primary Key | This is unique identification for the particular Course. |
| 2. | Course\_Name | Varchar | - | This is the name of course |
| 3. | Course\_Fees | Double |  | This Is the fees of particular course |
| 4. | Course\_Period | Int | - | This is the time period of course. This will be in months |
| 5. | Category\_id | Int | Foreign Key | This is the reference to the category to which course belongs |
| 6. | Course\_Image | Varchar | - | This is the image which will display on page. |
| 7. | Course\_Description | Varchar | - | This is the brief description of the course. |
| 8. | Course\_Status | Enum | - | This is status whether Course is active or Deactive |
| 5. | Course\_Entry\_Date | Datetime | - | This is the date and time when the Course is first time added. |
| 6. | Course\_Update\_Date | Datetime | - | This is the date and time when the Course is last time updated. |

1. Events

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Sr. No** | **Column Name** | **Datatype** | **Constraint** | **Description** |
|  |  |  |  |  |
| 1. | Event\_id | Integer | Primary Key | This is unique identification for the Event. |
| 2. | Event\_Name | Varchar |  | This is the Event Name |
| 3. | Event\_Date | Date |  | This is the date of the events |
| 4. | Event\_Time | Time |  | This is the time when the event will be started. |
| 5. | Event\_Location | Varchar | - | This is the place where the event will be organize |
| 6. | Event\_Description | Varchar | - | This is the brief description of the event |
| 7. | Event\_Image | Varchar |  | This is the path of the Image which will display |
| 8. | Event\_Status | Enum | - | This is status whether Event is done or will be in future. |
| 9. | Event\_Entry\_Date | Datetime | - | This is the date and time when the Event details is first time added. |
| 10. | Event\_Update\_Date | Datetime | - | This is the date and time when the Event details is last time updated. |

1. Exam\_Schedule

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Sr. No** | **Column Name** | **Datatype** | **Constraint** | **Description** |
|  |  |  |  |  |
| 1. | Exam\_id | Integer | Primary Key | This is the unique identification of the exam schedule |
| 2. | Exam\_Name | Varchar | - | This is the reference of the exam type. |
| 3. | Exam\_Date | Datetime | - | This is the date and time of the exam |
| 4. | Course\_id | Int | - | This is the reference of the course. |
| 5. | Exam\_Status | Enum | - | This is status whether Exam is pending or taken. |
| 6. | Exam\_Entry\_Date | Datetime | - | This is the date and time when the exam is first time arranged. |
| 7. | Exam\_Update\_Date | Datetime | - | This is the date and time when the Exam details are last time updated. |

1. Exam\_Type

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Sr. No** | **Column Name** | **Datatype** | **Constraint** | **Description** |
|  |  |  |  |  |
| 1. | Exam\_Type\_id | Integer | Primary Key | This is the unique identification for Exam Type. |
| 2. | Exam\_Type | Integer | Foreign Key | This is the type of the exam like daily or weekly. |
| 4. | Exam\_Type\_Status | Enum | - | This is status whether Exam\_Type is active or Deactive |
| 5. | Exam\_Type\_Entry\_Date | Datetime | - | This is the date and time when the Exam\_Type is first time added. |
| 6. | Exam\_Type\_Update\_Date | Datetime | - | This is the date and time when the Exam\_Type is last time updated. |

1. Material

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Sr. No** | **Column Name** | **Datatype** | **Constraint** | **Description** |
|  |  |  |  |  |
| 1. | Material\_id | Integer | Primary Key | This is unique identification for the particular Material. |
| 2. | Material\_file | Varchar | - | This is the path of the material file. |
| 3. | Lecture\_No | Int | - | This is the lecture number of that material |
| 4. | User\_id | Int | Foreign key | This is the reference to the teacher who will upload this material |
| 5. | Course\_id | Int | Foreign key | This is the reference of the course |
| 6. | Material\_Status | Enum | - | This is status whether Material is active or Deactive |
| 7. | Material\_Entry\_Date | Datetime | - | This is the date and time when the Material is first time added. |

1. Notification

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Sr. No** | **Column Name** | **Datatype** | **Constraint** | **Description** |
|  |  |  |  |  |
| 1. | Notification\_id | Integer | Primary Key | This is unique identification for the particular Notification. |
| 2. | Notification\_Title | Varchar | - | This is the title of the notification |
| 3. | Notification\_Description | Varchar | - | This is the brief message of notification content |
| 4. | Notification\_Date | Datetime | Foreign key | This is the date and time when the Notification Detail is first time added. |
| 5. | Notification\_Status | Enum | - | This is status whether Notification is active or Deactive |
| 6. | Notification\_Entry\_Date | Datetime | - | This is the date and time when the Notification detail is last time updated. |

1. Questions

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Sr. No** | **Column Name** | **Datatype** | **Constraint** | **Description** |
|  |  |  |  |  |
| 1. | Question\_id | Integer | Primary Key | This is unique identification for the particular question. |
| 2. | Exam\_id | Int | Foreign Key | This is the reference to the exam |
| 3. | Question | Varchar | - | This is the question |
| 4. | Option\_A | Varchar | - | This is the first option |
| 5. | Option\_B | Varchar | - | This is the second option |
| 6. | Option\_C | Varchar | - | This is the third option |
| 7. | Option\_D | Varchar | - | This is the fourth option |
| 8. | Answer | Varchar | - | This is the right answer |
| 9. | Question\_Status | Enum | - | This is status whether Question is active or Deactive |
| 10. | Question\_Entry\_Date | Datetime | - | This is the date and time when the Question is first time added. |
| 11. | Question\_Update\_Date | Datetime | - | This is the date and time when the Question is last time updated. |

1. Results

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Sr. No** | **Column Name** | **Datatype** | **Constraint** | **Description** |
|  |  |  |  |  |
| 1. | Result\_id | Integer | Primary Key | This is unique identification for the particular result. |
| 2. | Exam\_id | Int | Foreign Key | This is the reference to the exam |
| 3. | User\_id | Int | Foreign Key | This is the reference to the student |
| 4. | Marks | Double | - | This is the marks obtain in the exam |
| 5. | Result\_Status | Enum | - | This is status whether Result is active or Deactive |
| 6. | Question\_Entry\_Date | Datetime | - | This is the date and time when the Result is first time added. |

1. State

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Sr. No** | **Column Name** | **Datatype** | **Constraint** | **Description** |
|  |  |  |  |  |
| 1. | State\_id | Integer | Primary Key | This is unique identification for the particular State. |
| 2. | State\_Name | Varchar | - | This is the name for the State. |
| 3. | State\_Status | Enum | - | This is status whether State is active or Deactive |
| 4. | State\_Entry\_Date | Datetime | - | This is the date and time when the State is first time added. |
| 5. | State\_Update\_Date | Datetime | - | This is the date and time when the State is last time updated. |

1. Student\_Notification

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Sr. No** | **Column Name** | **Datatype** | **Constraint** | **Description** |
|  |  |  |  |  |
| 1. | User\_id | Integer | Foreign Key | This is the reference to the user |
| 2. | Notification\_id | Int | Foreign Key | This is the reference to the notification details |
| 3. | Notification\_Status | Enum | Foreign Key | This is status whether Notification is pending or sent |

1. Syllabus

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Sr. No** | **Column Name** | **Datatype** | **Constraint** | **Description** |
|  |  |  |  |  |
| 1. | Syllabus\_id | Integer | Primary Key | This is unique identification for the particular State. |
| 2. | File\_Name | Varchar | - | This is the name for the State. |
| 3. | Syllabus\_PDF | Varchar |  | This is the path of uploaded syllabus file |
| 4. | Course\_id | Int | Foreign key | This is the reference to the course. |
| 5. | State\_Status | Enum | - | This is status whether Syllabus is active or Deactive |
| 6. | State\_Entry\_Date | Datetime | - | This is the date and time when the Syllabus is first time added. |

1. Users

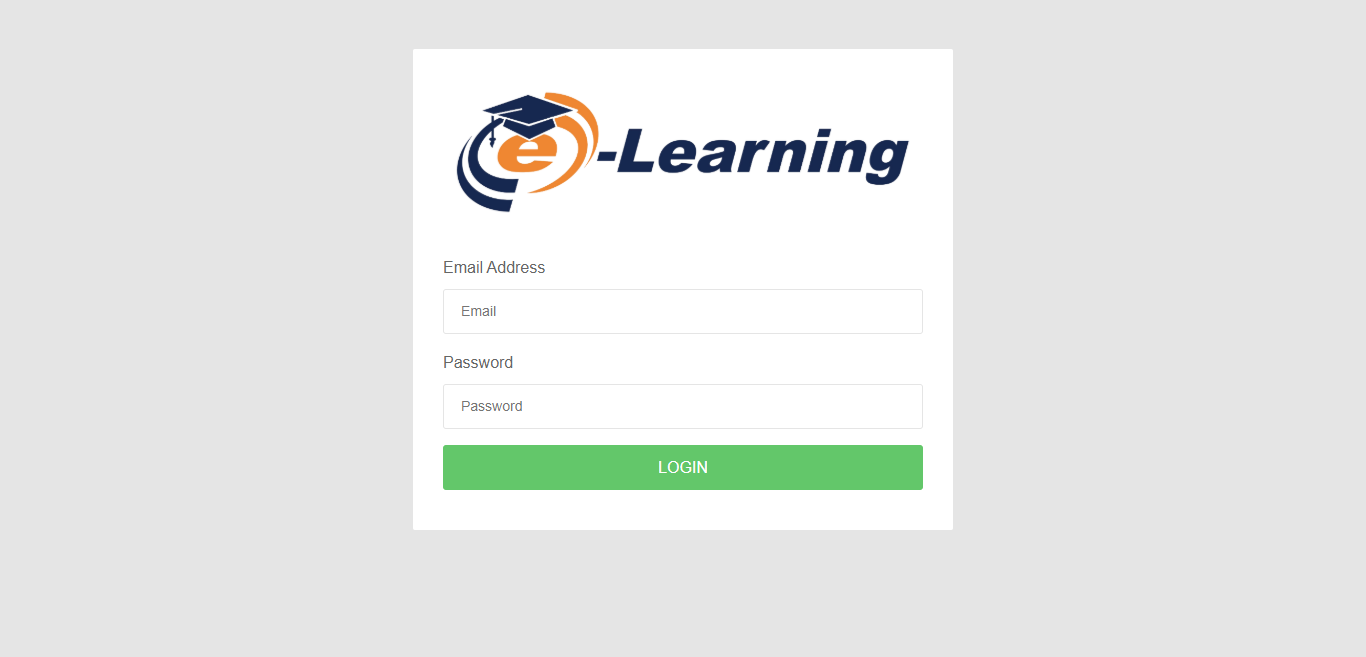
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Sr. No** | **Column Name** | **Datatype** | **Constraint** | **Description** |
|  |  |  |  |  |
| 1. | User\_id | Integer | Primary Key | This is unique identification for the particular User. |
| 2. | Username | Varchar | - | This is the username or email by which user will login |
| 3. | Password | Varchar | - | This is the password credential |
| 4. | Name | Varchar | - | This is the Name of the user |
| 5. | Mobile\_Number | Bigint | - | This is the contact number of the user |
| 6. | Gender | Enum | - | This is the gender of user. |
| 7. | DOB | Date | - | This is Birthdate of the user |
| 8. | Photo | Varchar | - | This is the path of user’s profile photo |
| 9. | Address | Varchar | - | This is the full address of user. |
| 10. | City\_id | Int | Foreign Key | This is the reference to the city |
| 11. | User\_Type | Enum | - | User may be Student or Teacher. |
| 12. | User\_Status | Enum | - | This is status whether User is active or Deactive |
| 13. | User\_Entry\_Date | Datetime | - | This is the date and time when the User detail is first time added. |
| 14. | User\_Update\_Date | Datetime | - | This is the date and time when the User detail is last time updated. |

1. Users\_Course

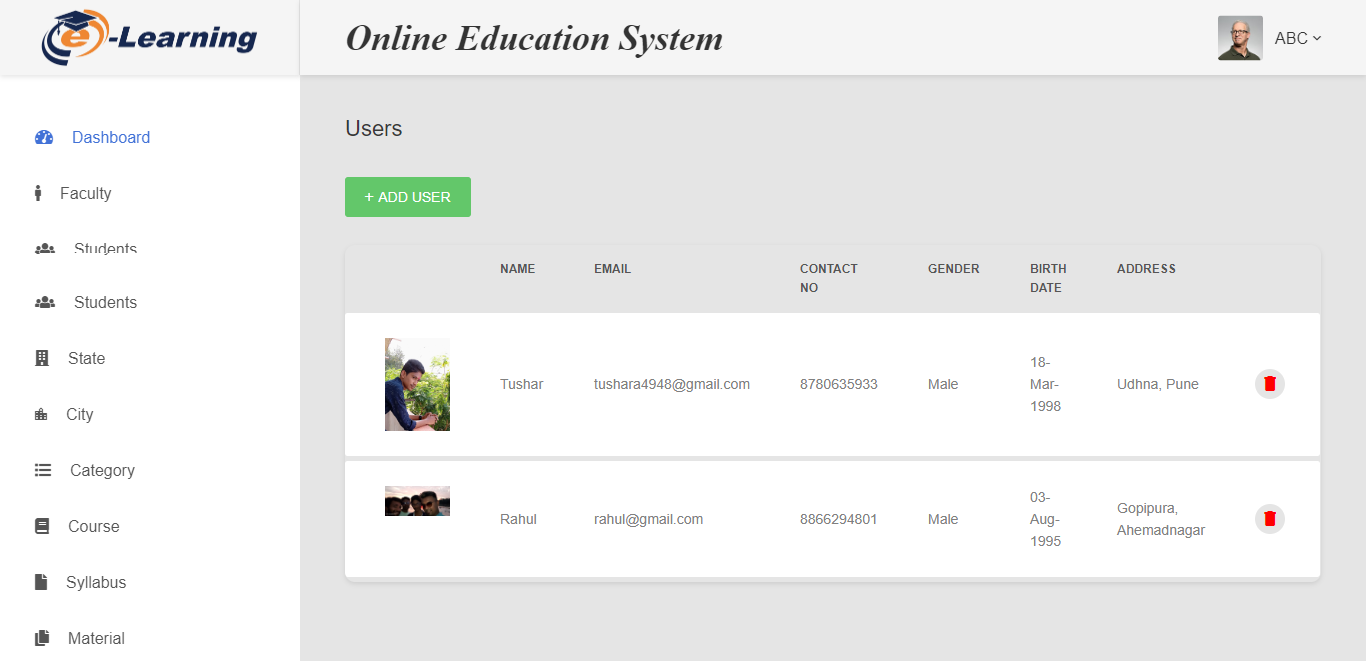
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Sr. No** | **Column Name** | **Datatype** | **Constraint** | **Description** |
|  |  |  |  |  |
| 1. | User\_id | Integer | Foreign Key | This is the reference to the user |
| 2. | Course\_id | Int | Foreign Key | This is the reference to the Course |
| 3. | User\_Course\_Status | Enum | Foreign Key | This is status whether User is pursuing the course or completed |
| 4. | User\_Course\_Entry\_Date | Datetime | - | This is the date and time when the User and Course detail is first time added. |
| 5. | User\_Course\_Update\_Date | Datetime | - | This is the date and time when the User and Course detail is last time updated. |

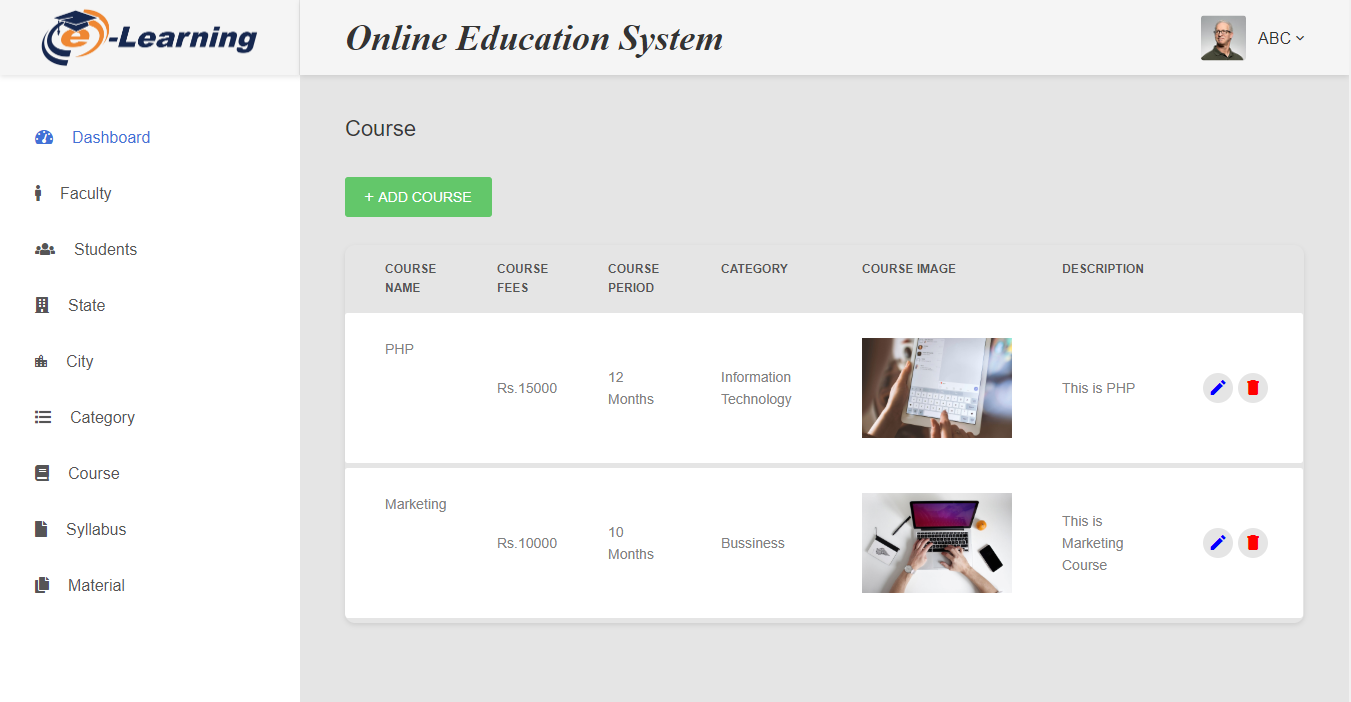
6.2 Interface Design sitemap followed with page

**1. Admin and Teacher Login**

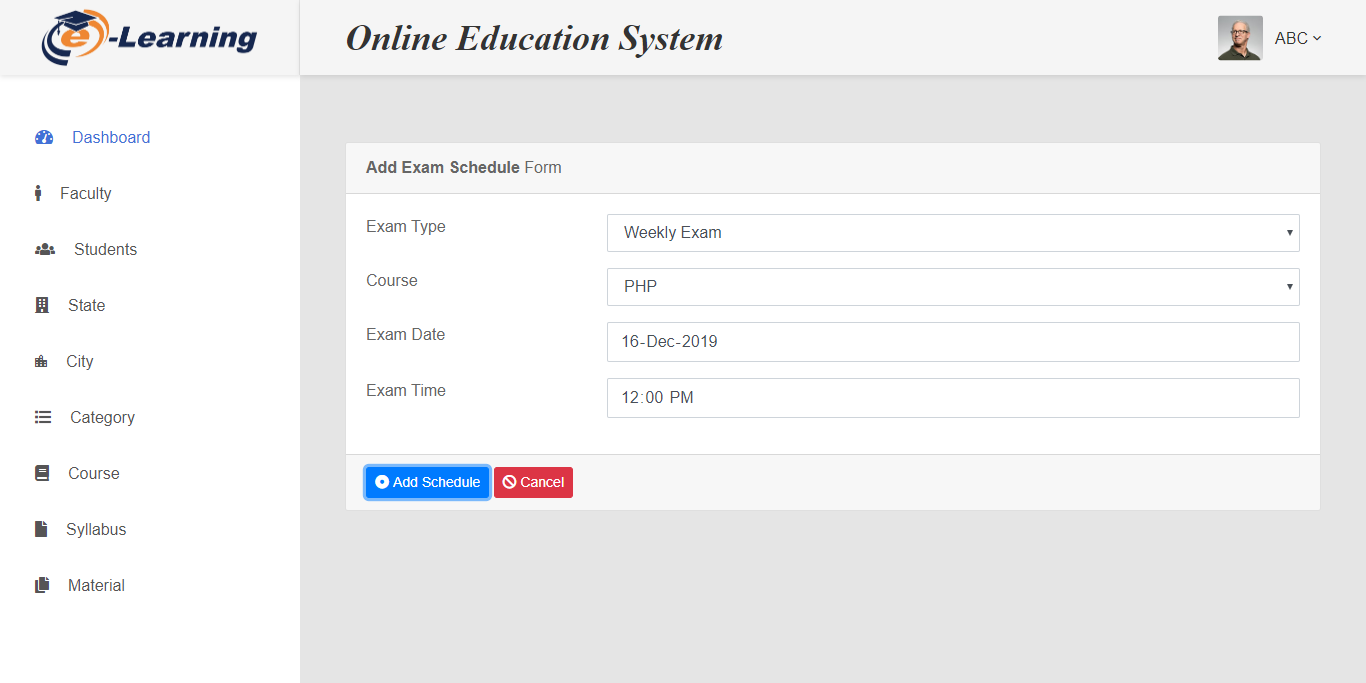
****

**Admin**

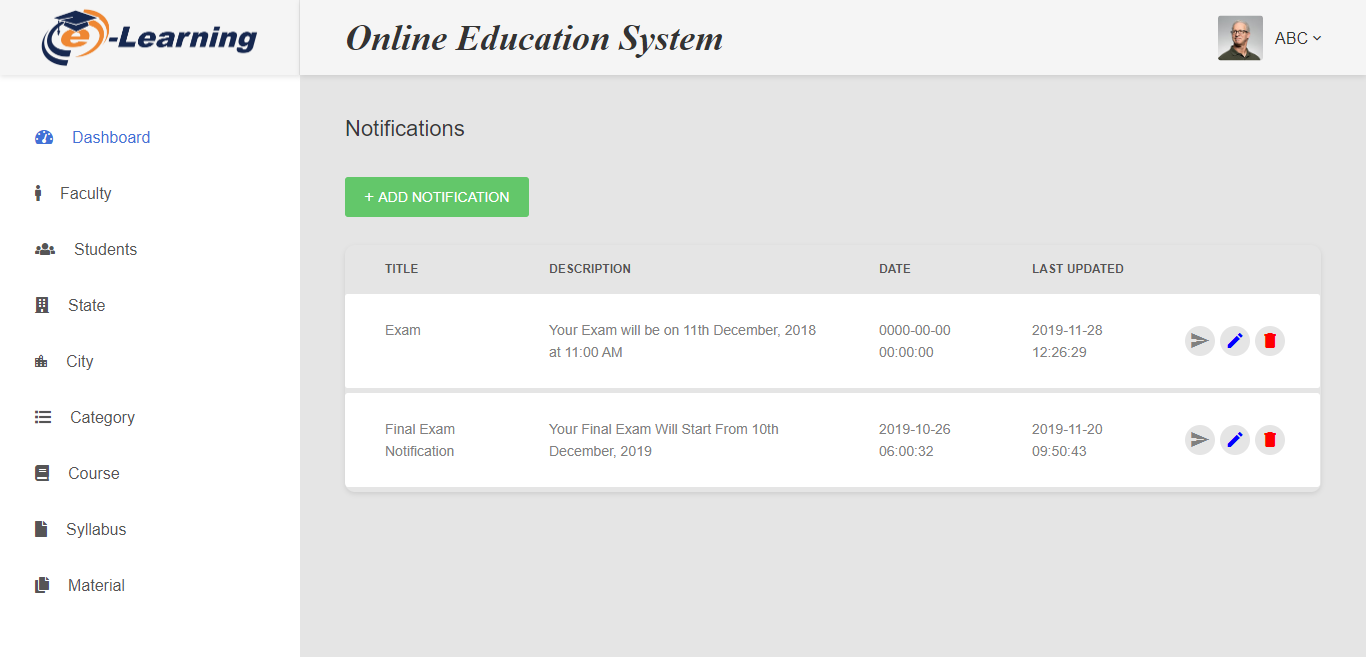
**2. View User**

**3. View Course**

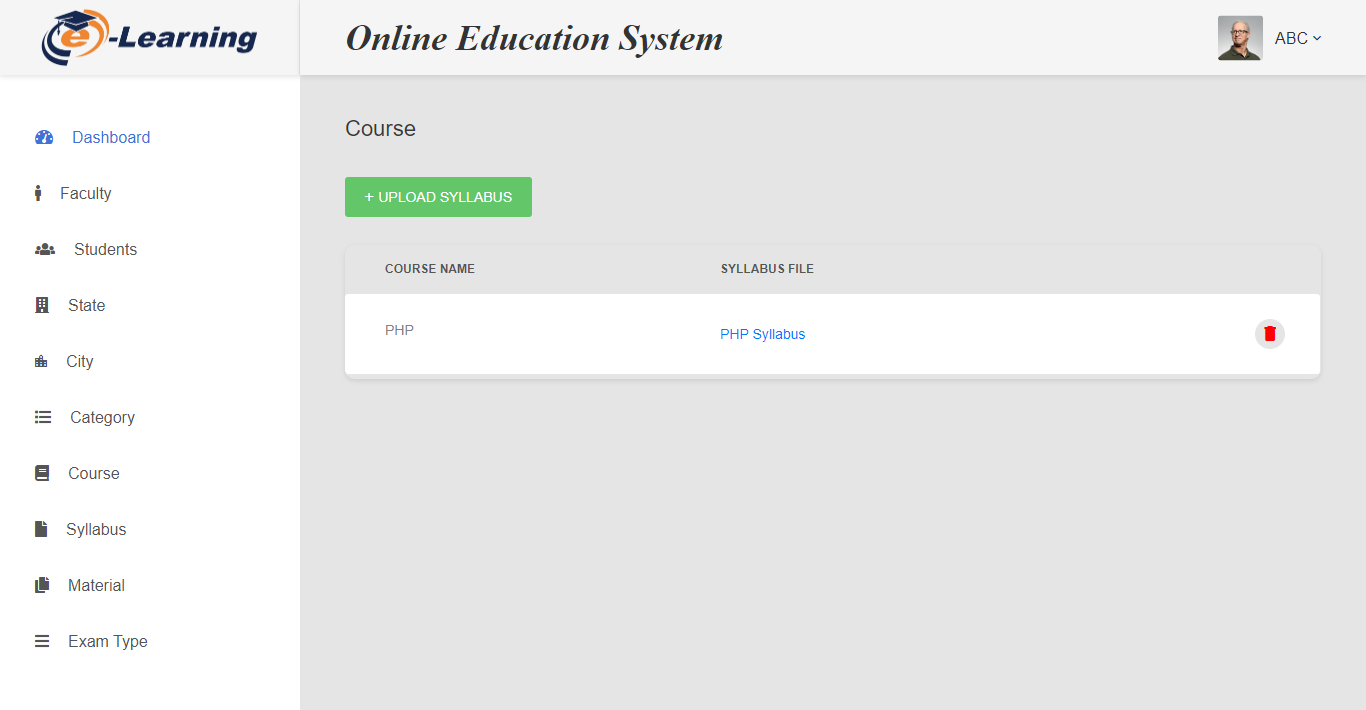
**4. Add Exam Schedule**

****

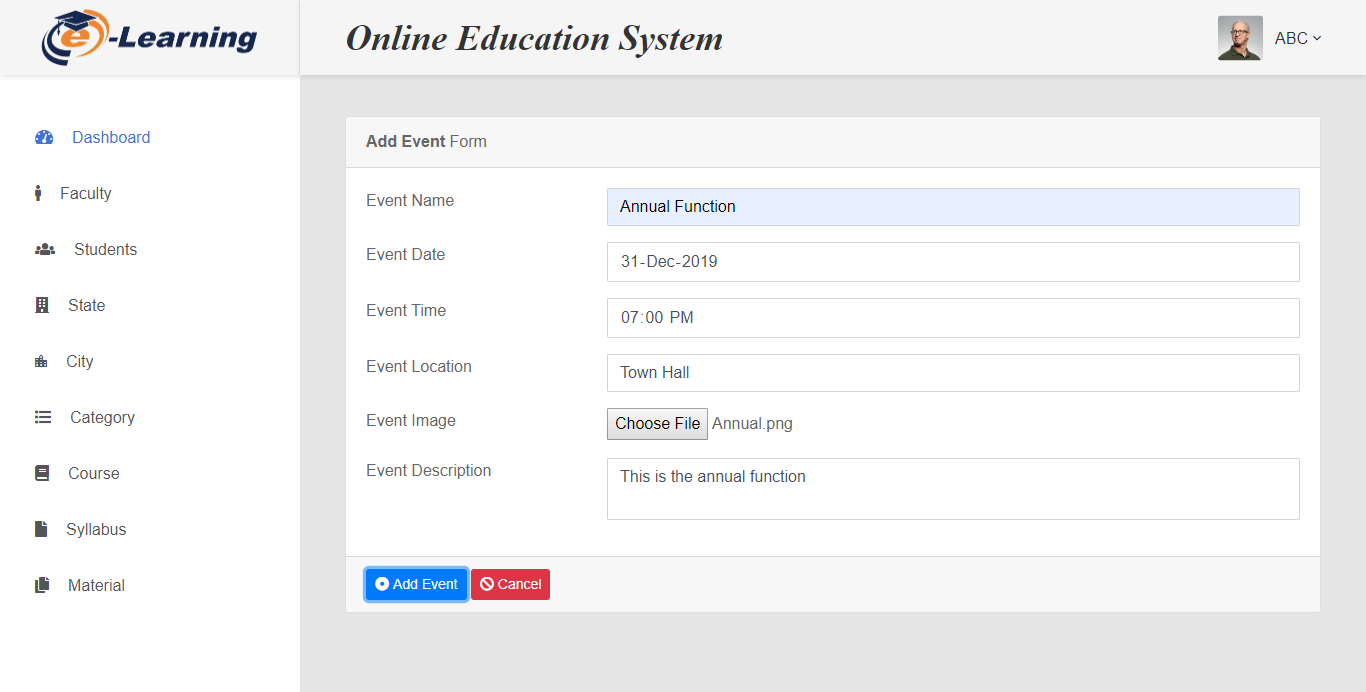
**5. Notification**

****

**6. View Syllabus**

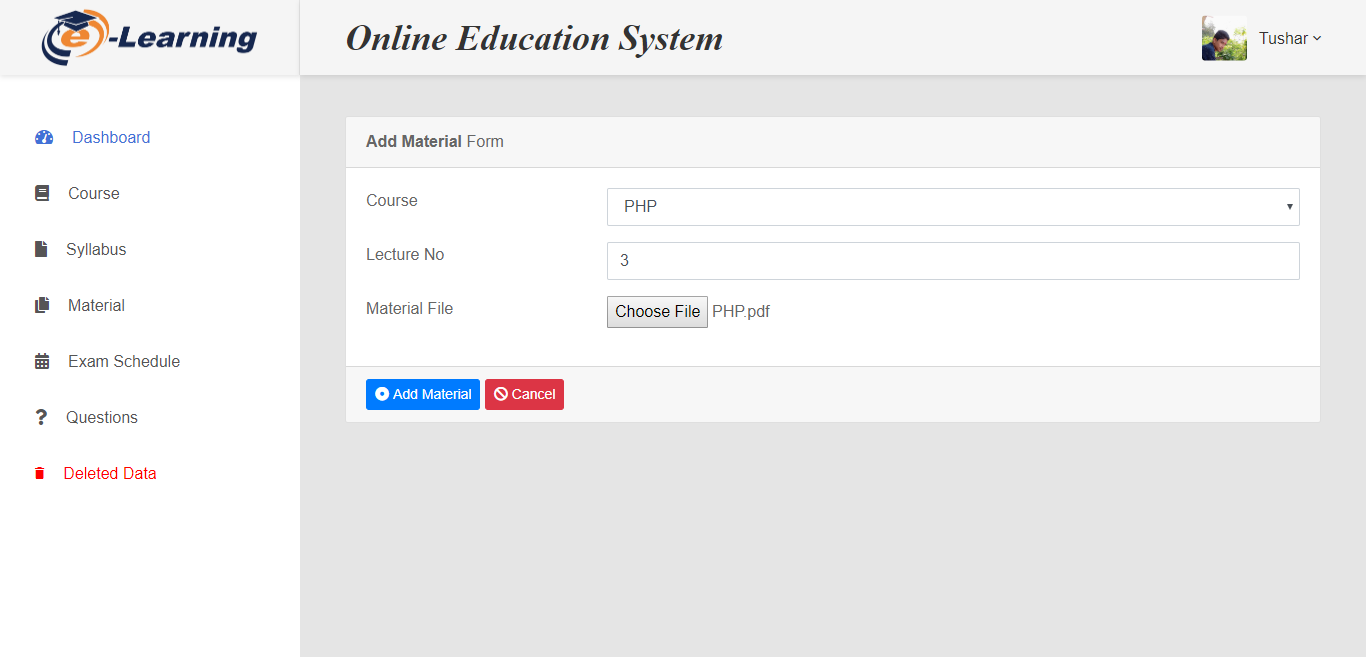
****

**7. Add Event**

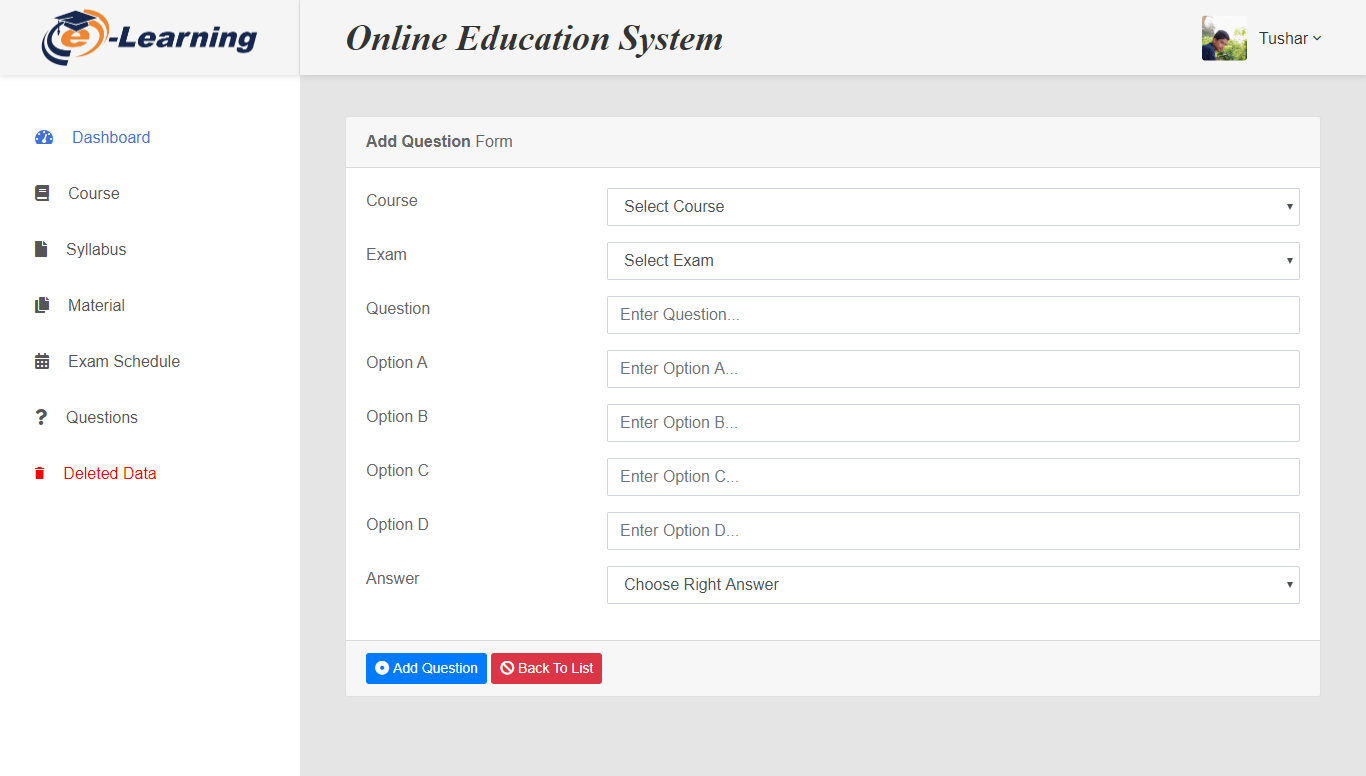
****

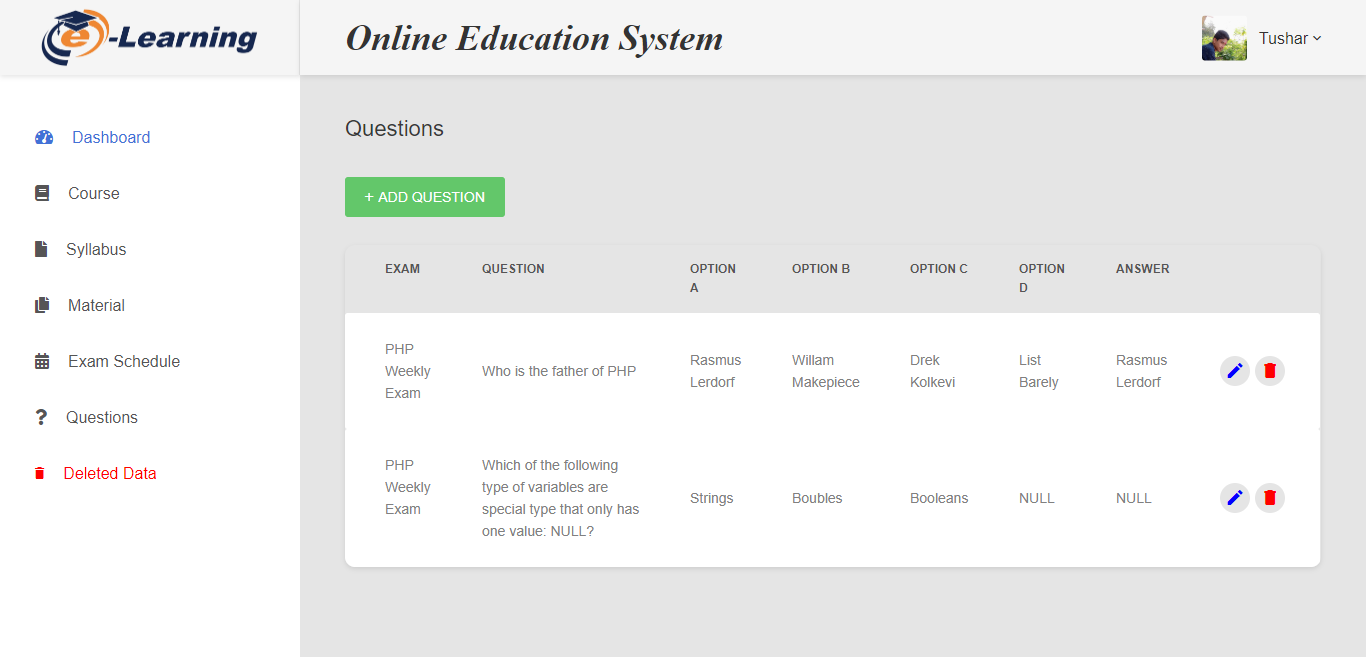
**Teacher**

**8. Add Material**

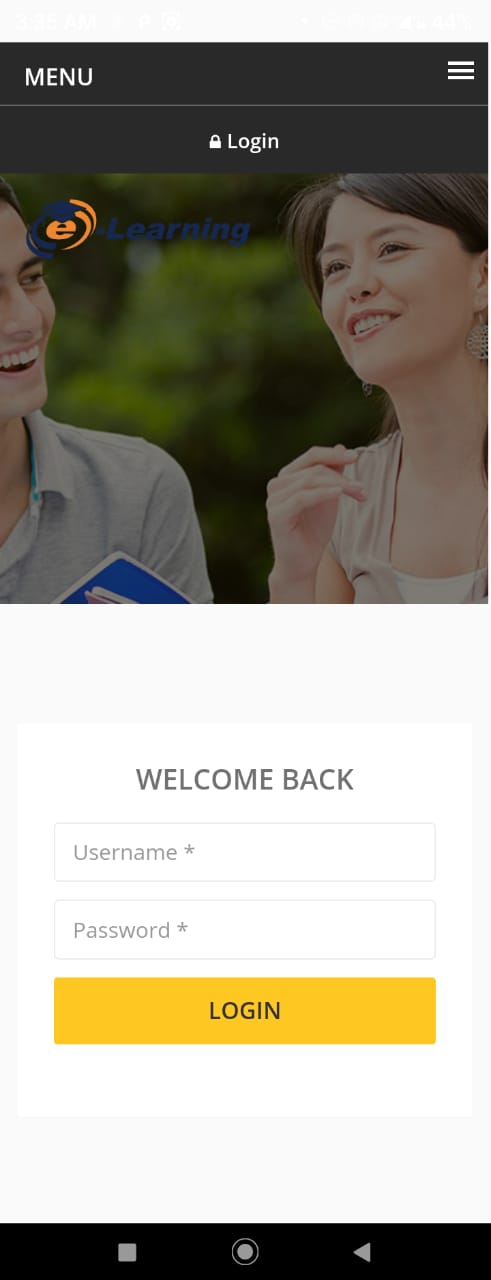
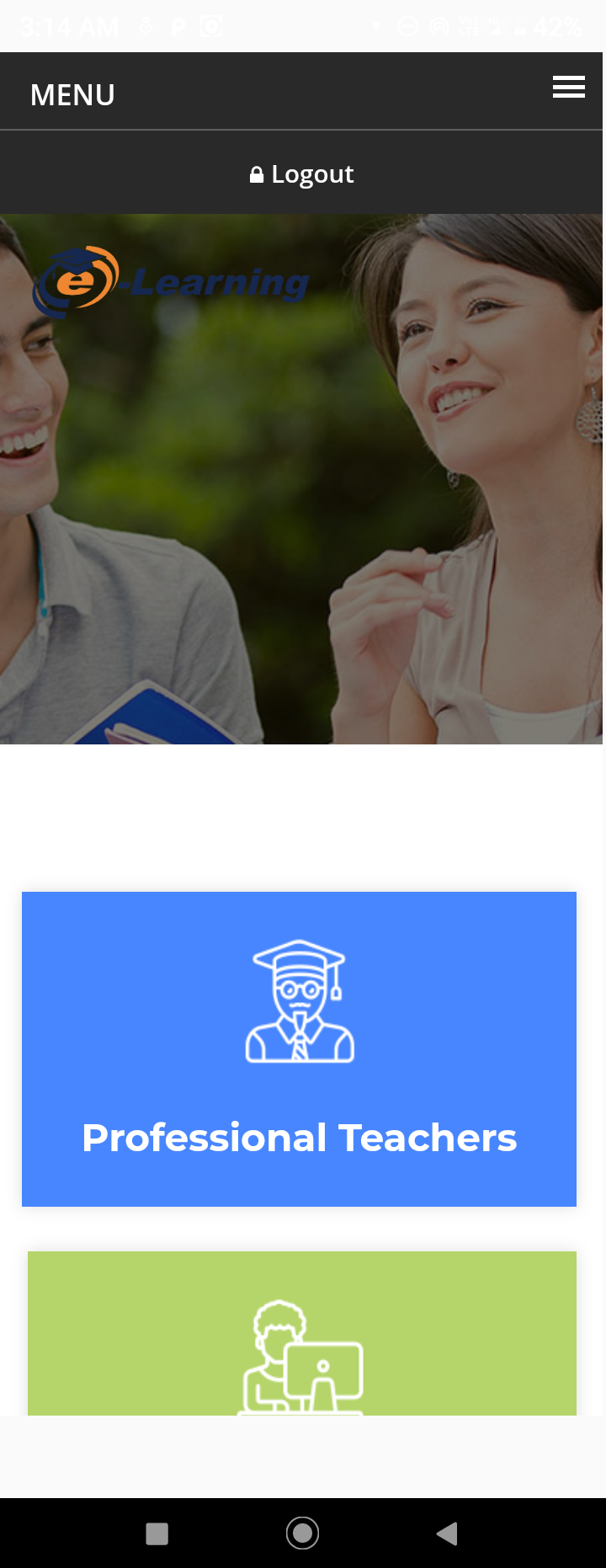
****

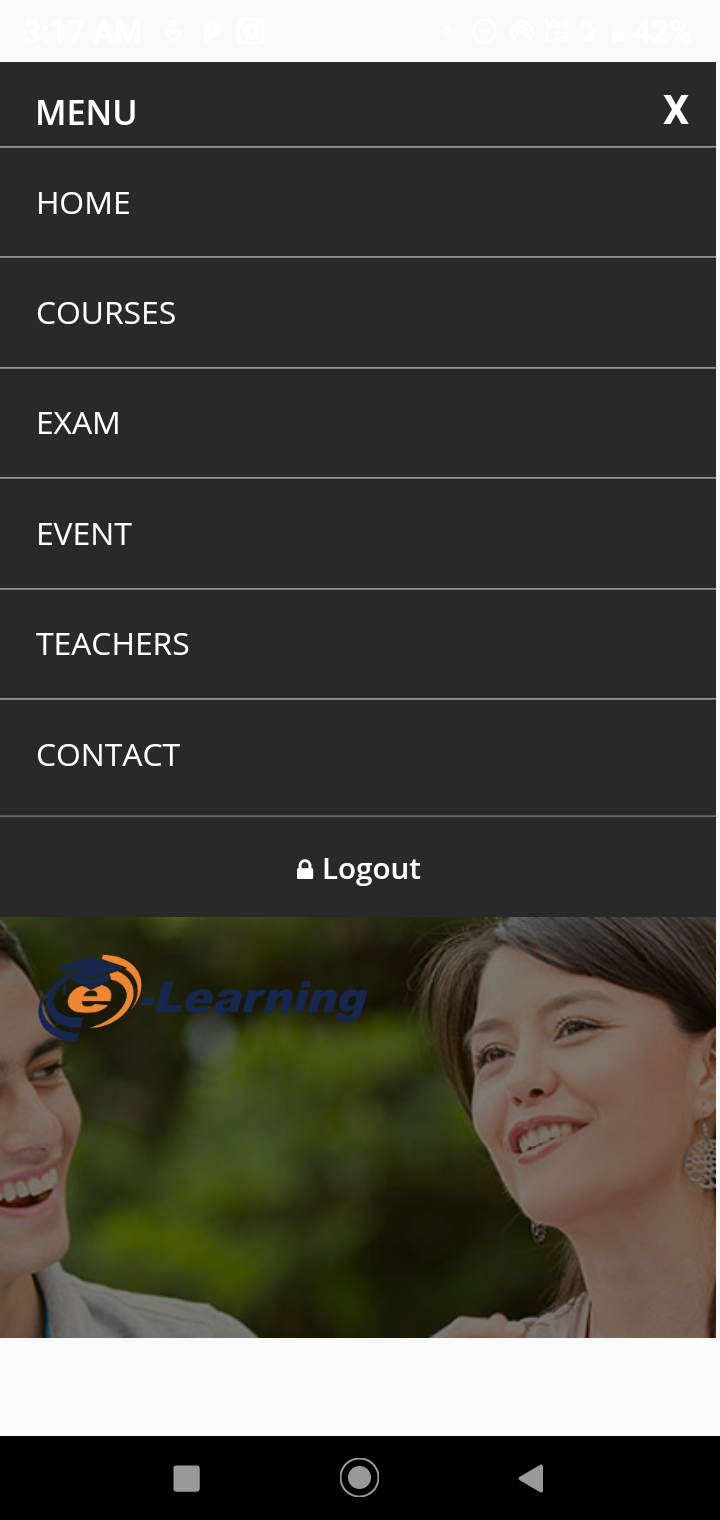
**9. Add Question**

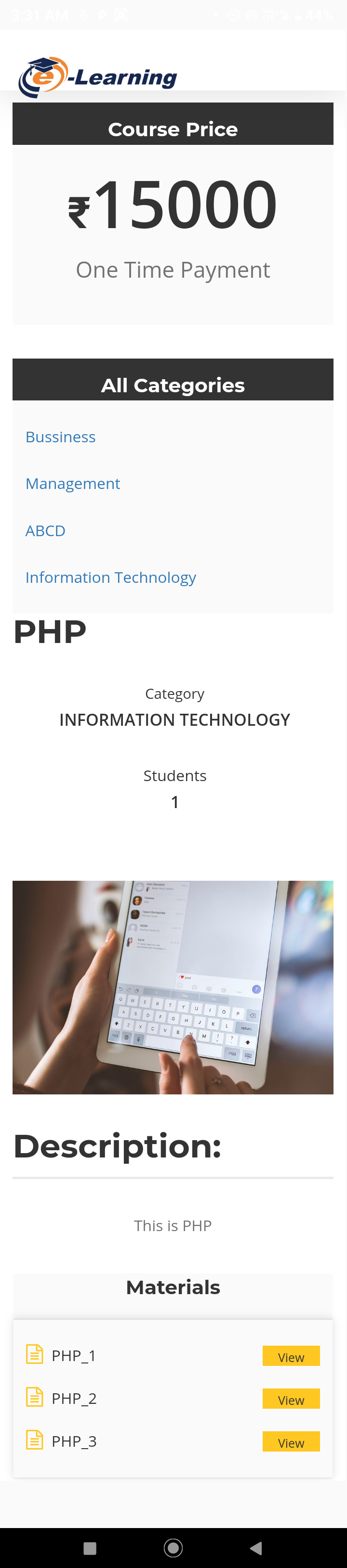
****

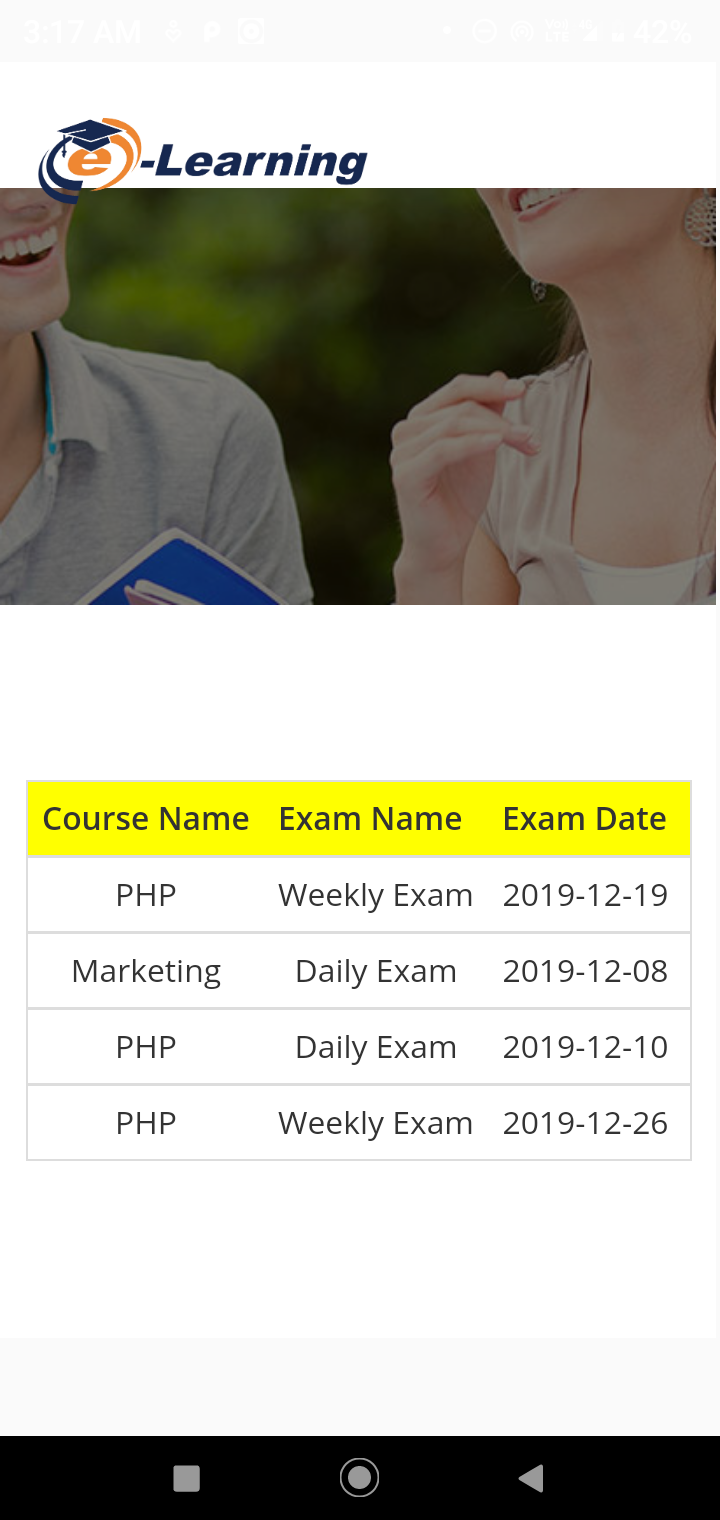
**10. View Questions**

**Student**

**11. Login and Home Page**

**12. Menu**

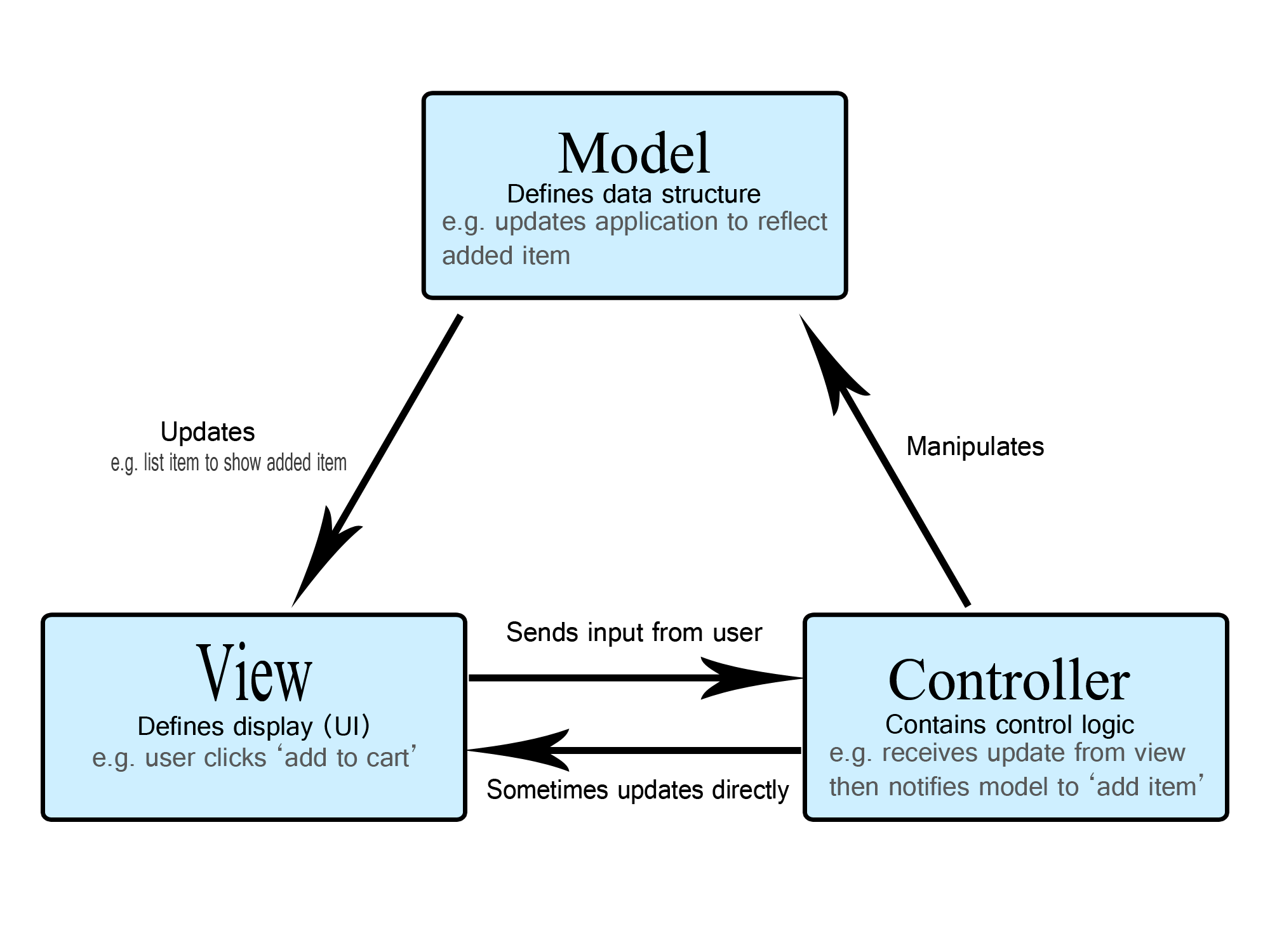
**13. Courses**

**15. Exam**

6.3 Architecture Design

Model View Controller (MVC) is a software architecture pattern, commonly used to implement user interfaces: it is therefore a popular choice for architecting web apps. In general, it separates out the application logic into three separate parts, promoting modularity and ease of collaboration and reuse. It also makes applications more flexible and welcoming to iterations.

To make this a little more clearly, let's imagine a simple shopping list app. All we want is a list of the name, quantity and price of each item we need to buy this week. Below we'll describe how we could implement some of this functionality using MVC.



* **The Model: -**

The model defines what data the app should contain. If the state of this data changes, then the model will usually notify the view (so the display can change as needed) and sometimes the controller (if different logic is needed to control the updated view).

* **The View: -**

The view defines how the app's data should be displayed.

* **The Controller: -**

The controller contains logic that updates the model and/or view in response to input from the users of the app.

1. Testing

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Page Name** | **View** | **Insert** | **Update** | **Delete** | **Validate** | **Recover**  **Deleted** |
| Category | **✓** | **✓** | **✓** | **✓** | **✓** | **✓** |
| City | **✓** | **✓** | **✓** | **✓** | **✓** | **✓** |
| Contact\_Us | **✓** | **✓** | **🗶** | **🗶** | **✓** | **🗶** |
| Course | **✓** | **✓** | **✓** | **✓** | **✓** | **✓** |
| Events | **✓** | **✓** | **✓** | **✓** | **✓** | **✓** |
| Exam\_Schedule | **✓** | **✓** | **✓** | **✓** | **✓** | **✓** |
| Exam\_Type | **✓** | **✓** | **✓** | **✓** | **✓** | **✓** |
| Material | **✓** | **✓** | **✓** | **✓** | **✓** | **✓** |
| Notification | **✓** | **✓** | **✓** | **✓** | **✓** | **✓** |
| Question | **✓** | **✓** | **✓** | **✓** | **✓** | **✓** |
| Results | **✓** | **✓** | **🗶** | **🗶** | **✓** | **🗶** |
| State | **✓** | **✓** | **✓** | **✓** | **✓** | **✓** |
| Student\_Notification | **✓** | **🗶** | **🗶** | **🗶** | **-** | **🗶** |
| Syllabus | **✓** | **✓** | **✓** | **✓** | **✓** | **✓** |
| Users | **✓** | **✓** | **✓** | **✓** | **✓** | **✓** |
| Users\_Course | **✓** | **-** | **-** | **-** | **-** | **-** |

7.1 Unit Testing

1. Future Enhancement

* There can be online attendance of Students and Teachers as well.
* We can provide softcopy of the certificate at our application as well.
* We can give admission online also by registration.
* We can provide lecture online to students via live streaming.

1. Reference

* <https://www.w3schools.com/jquery/>
* <http://www.stackoverflow.com>
* <http://www.drawio.com>
* <https://go.gliffy.com/go/html5/13180101>
* <http://www.c-sharpcorner.com>