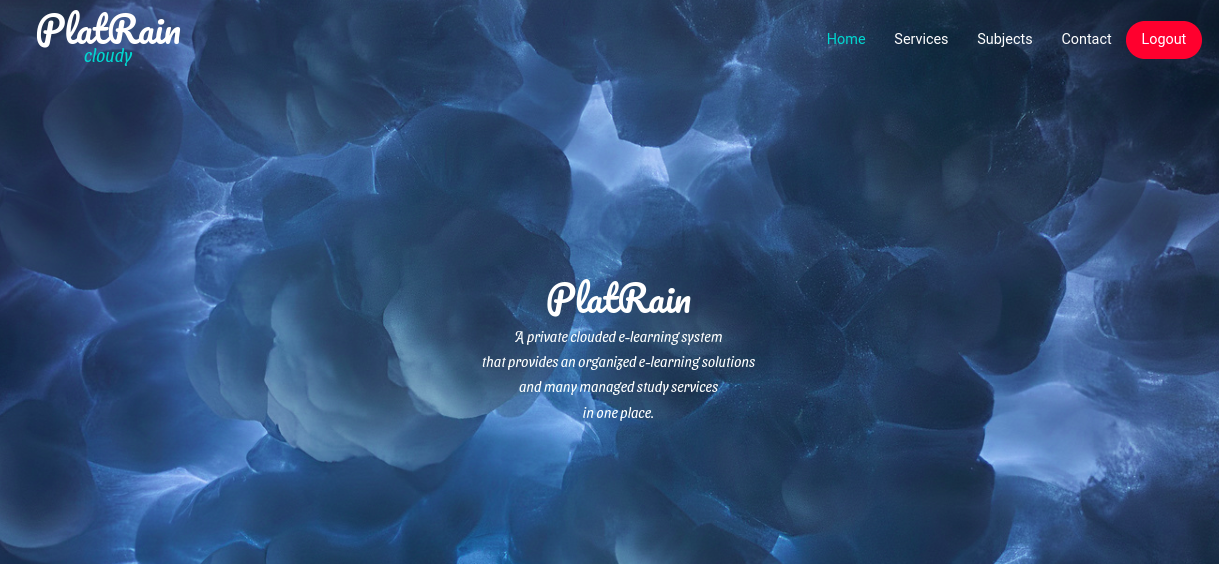
## PlatRain :

A private clouded e-learning system that provides an organized e-learning solutions and many managed study services in one place.

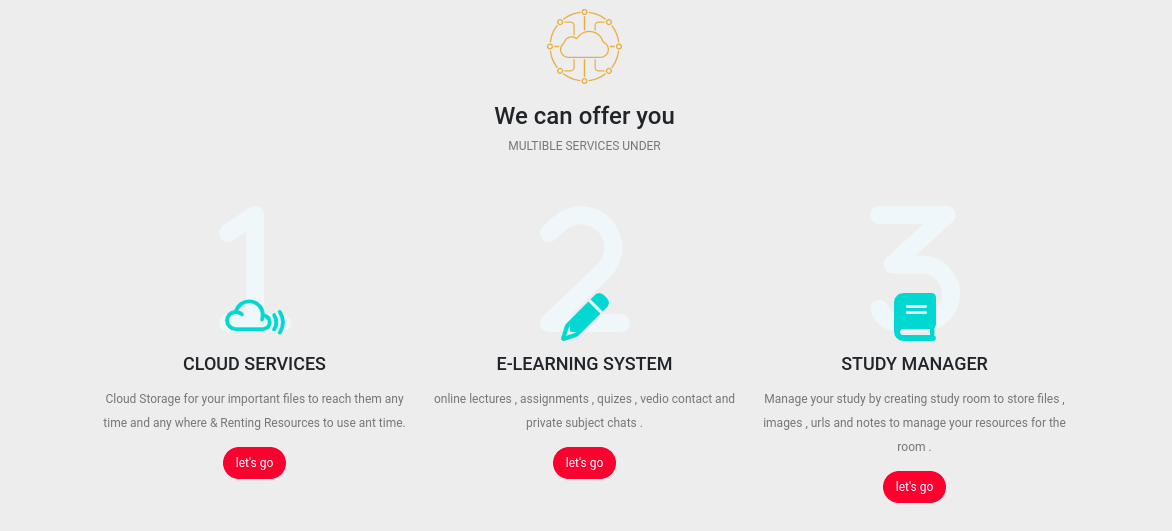


home11

## The three main services we provide :

In home page ,user can see three main taps for the three main services that our system provides :

* cloud services.
* E-learning platform.
* User files ,notes,links, … manager.



home-services

# E-learning Platform :

## What is E-learning System ?

​ E-learning (also called electronic learning) is any type of learning that takes place through or with a computer and is primarily facilitated through the Internet but can also be accomplished with CD-ROMs and DVDs, streaming audio or video and other media.

## Why E-learning System ?

​ The purpose of e-learning is to allow people to learn for personal accomplishment or to earn a professional degree,without physically attending a traditional university or academic setting. Applied for all levels of schooling from grade school to graduate degrees, e-learning is versatile enough to accommodate all learning styles.

​ Actually, a tragedy like covid-19 virus changed many of things in our world ,and changed our look to many things , it showed that we always need alternatives to things we used to do or to use . E-learning strategy was a very good way out from the pickle of non ability to communicate face to face .

## Types of E-Learning :

​ There are a number of types of e-learning that depend on the amount of physical interaction. Entirely online e-learning occurs without any face-to face interaction. Course work and materials are distributed electronically through email, websites, online forums and/or CDs or DVD-ROMs. Combined learning uses a combination of Internet-directed instruction, as well as face-to-face interaction. Most traditional colleges and universities use combined learning as students learn in physical classrooms, with instruction augmented by online lessons. For those learning for personal accomplishment, e-learning can also use a combination of e-learning types, as they can be entirely self-directed, or they can use the assistance of an expert in their selected field.

## There are many E-learning Platforms already , then why a private one ?

​ A big faculty like faculty of engineering that has a department like department of computer and systems control has the ability of building its own E-learning System that includes the university and the faculty specific taste ,look, and style, then why not ?

## Why such a great faculty should use a system made by students not professionals ?

For many reasons :

* to have students who have the ability to build such an important and big thing , that is great for the university and the faculty to the eyes of other universities because that means :
  + the faculty has qualified students.
  + the faculty trusts its well learned students and can depend on them.
  + the faculty supports and guides its students and let them help in its development .
  + the faculty wakes on the steps of our country and our president on supporting digital transformation and moving step by step to digital Egypt .
* to have your private system with you own taste and style and achieves your requirements specifically and can be managed and controlled by you , it is a great thing .

## construction and implementation steps of the E-learning System :

## Analysis and overview :

### Authentication System :

* 
* lock

Website authentication is the security process that allows users to verify their identities in order to gain access to their personal accounts on a website.

This process occurs behind the scenes any time an individual logs into an online account, including social media profiles, eCommerce sites, rewards programs, online banking accounts, and more.

When a user creates a new account on a website, they create a unique ID and key that will be used in the future to verify their identity and allow them *back* into the account. That ID and key are then stored in a highly secure web server to compare future credentials against.

### Authentication applied in our case using some steps :

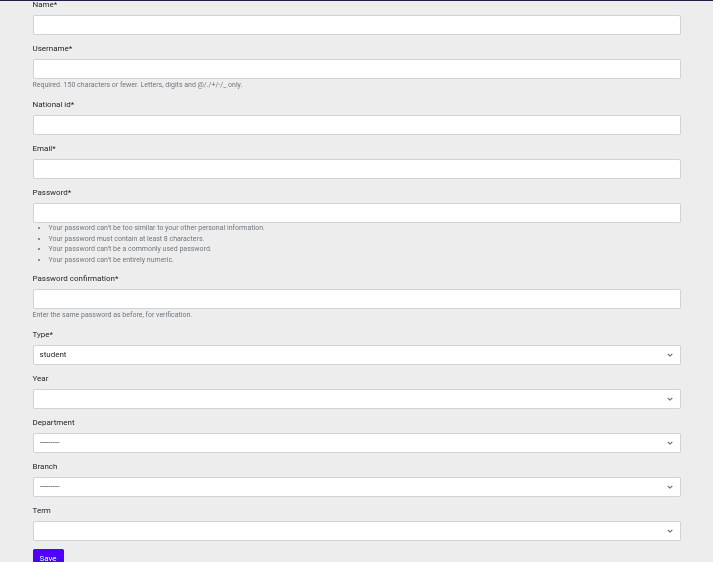
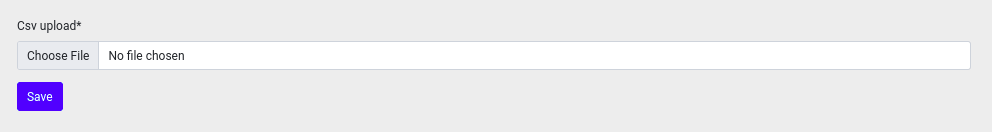
### Creating an account for all students and stuff :

* Only the administrator of the system can add users to the system .

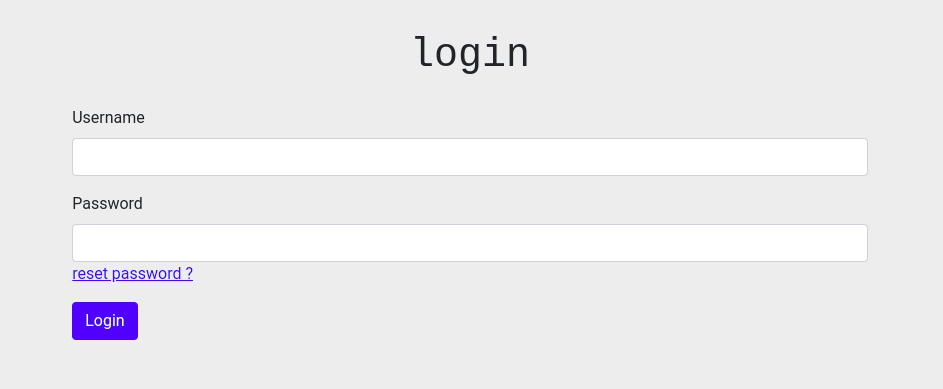
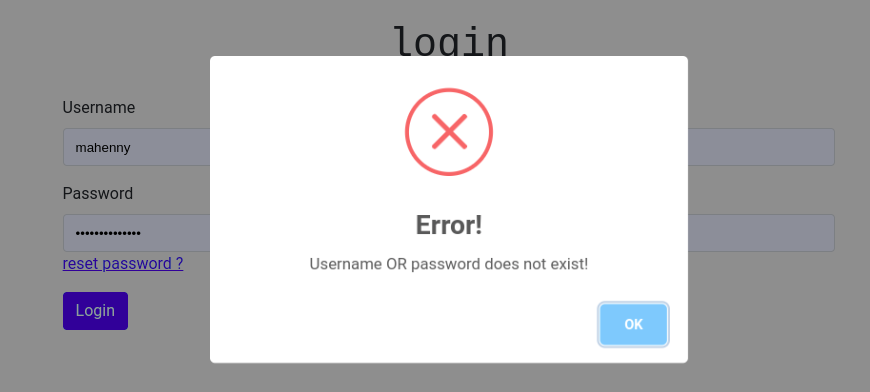
### why not every student can create an account to himself ?

* + to prevent any one except the students and the stuff of the faculty to access the system data .
  + this gives us a kind of organization .

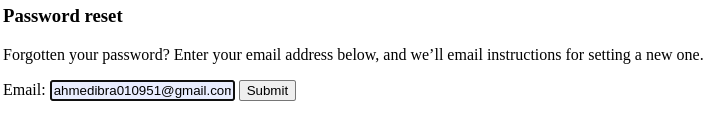
### three types of users in our system :

* there are three types of users in the system . they differs from each other in permissions and who can access what.
* they are:
  + Administrator .
  + Stuff .
  + Students .
* ### Two ways the admin can add users to the system:
  + adding single user by completing a form of user needed data .
  + 
  + adduser
  + adding a csv file of users in one time .
  + 
  + add-scv

### Login system :

* all student , staff members have accounts that can be accessed by user name and password .
* the system checks if this user has an account or not .
* 
* login
* if user has an account it will take him to the home page and can access the data he is permitted to access, but if the user name or password is wrong ,this error message will appear :
* 
* login-error

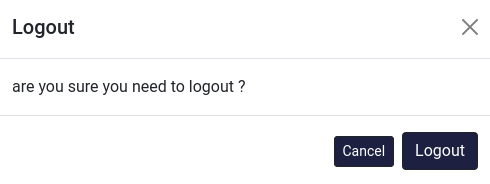
### Reset Password functionality :

* in case that the user forgot his password ,he can reset it by using his e-mail.
* 
* resetpassword

### Logout functionality :

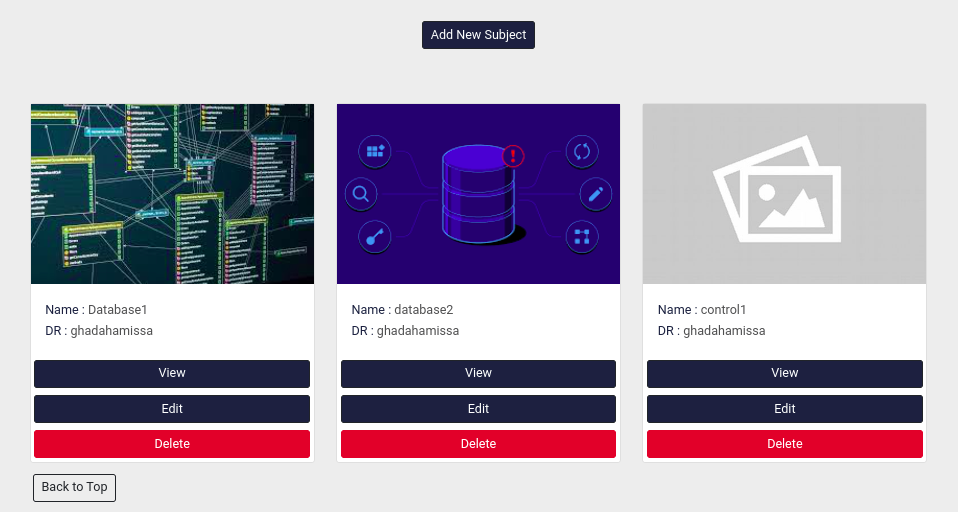
* there is a logout button that appears instead of the login button in the navigation bar so that we can logout the system when we finish using it , then user can login whenever he wants .

### Logout assurance :

* 
* logout-ensure

### Subjects :

Every subject has information about it,lectures,assignments,and quizes.



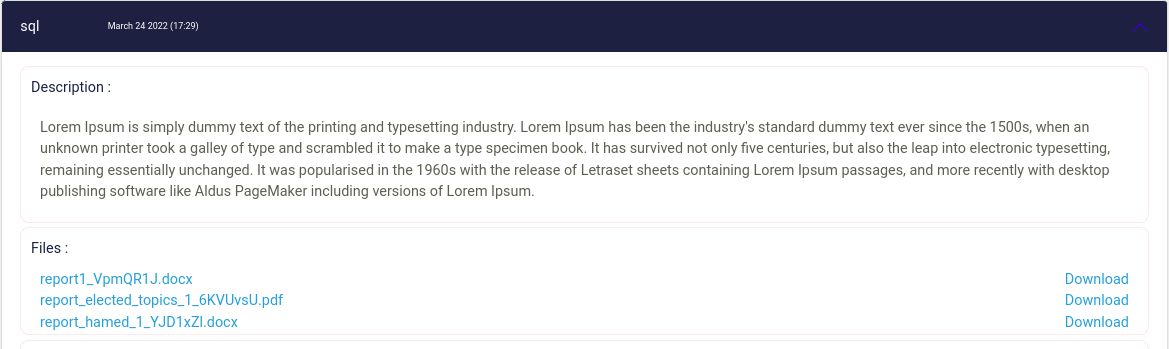
subjectss

* if the authenticated user is administrator , he can see all subjects and all lectures,assignments,quizes of this subject.
* if the authenticated user is stuff , he can see and access subjects that belong to him and all lectures,assignments,quizes of these subjects.
* if the authenticated user is student , he can see subjects that belong to him and all lectures,assignments,quizes of these subjects.
* stuff can add ,edit , or remove subject :
* 
* add-subject

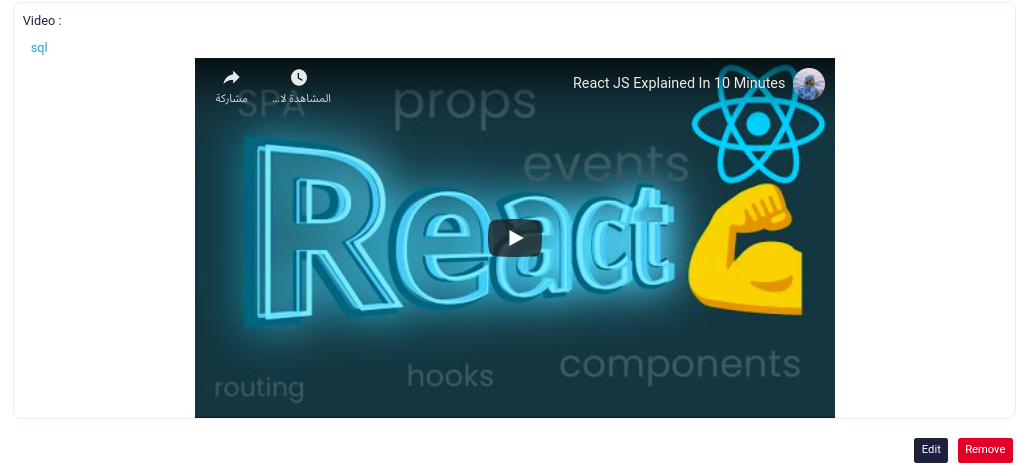
### Lectures :



addlectrueee

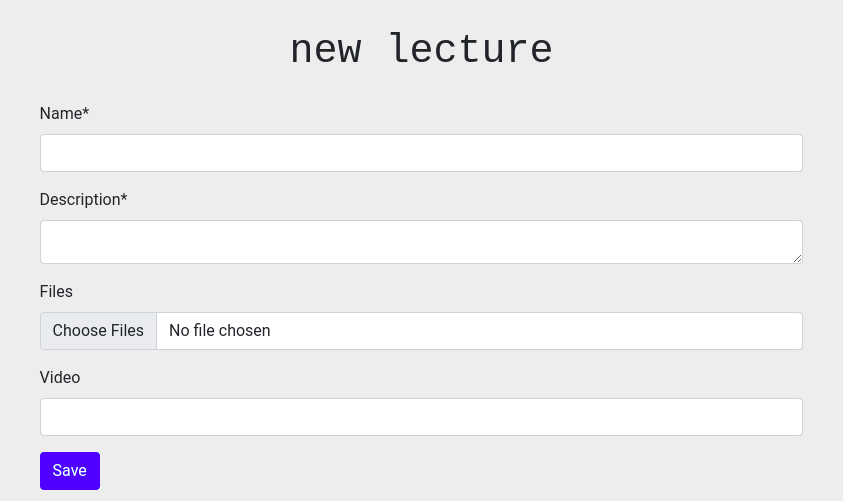


lecture1



lecture2

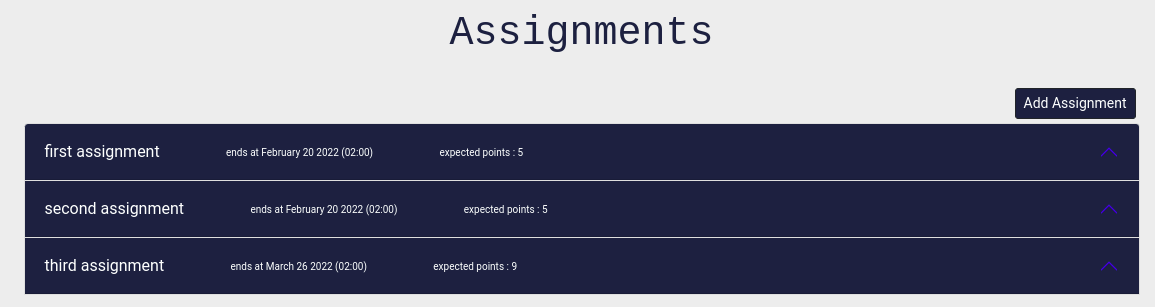
* show all lectures of a subject.
* every lecture has title , date added , video link can be watched directly in frame or on youtube ,files that can opened or downloaded .
* stuff can add ,edit , or remove lecture :



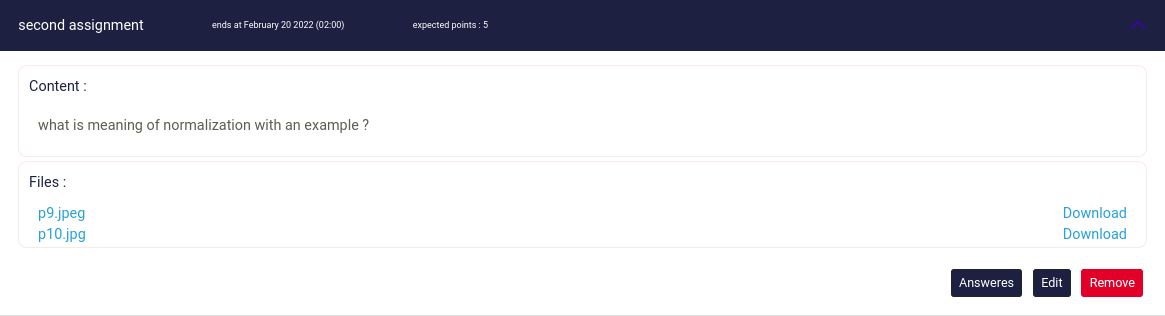
add-lectue

### Assignments :

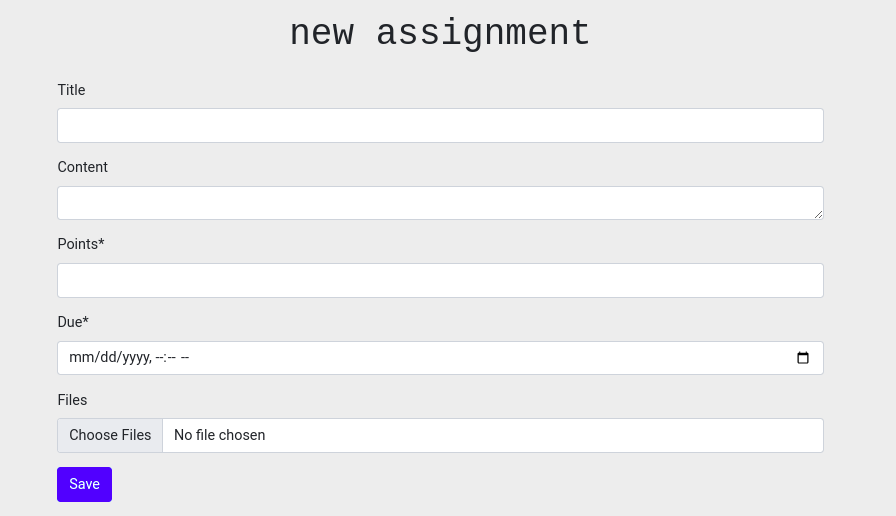
* doctor can add new assignment .
* doctor can grade assignment after the assignment final date finishes.
* doctor can see all students degrees in an assignment.
* student can send an answer.
* student can say his answer and edit or delete it before the assignment final date finishes.



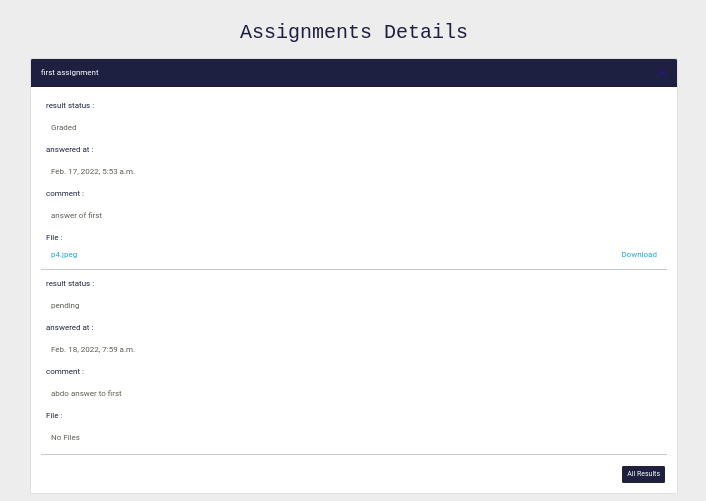
assignmentsss



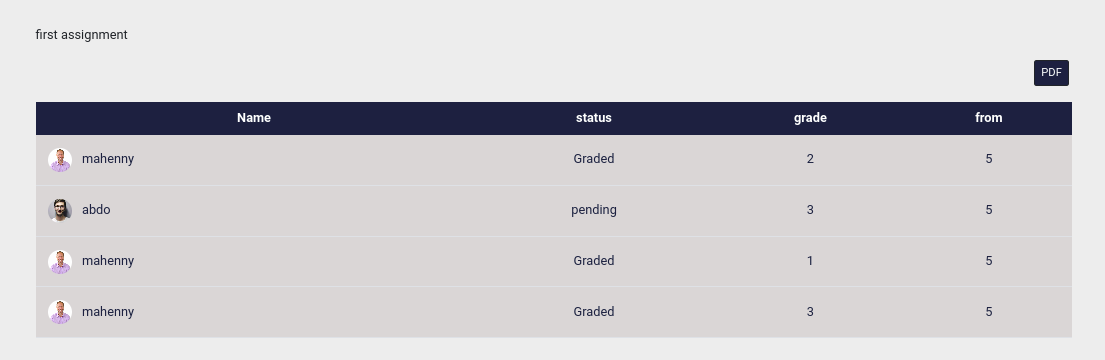
assignment



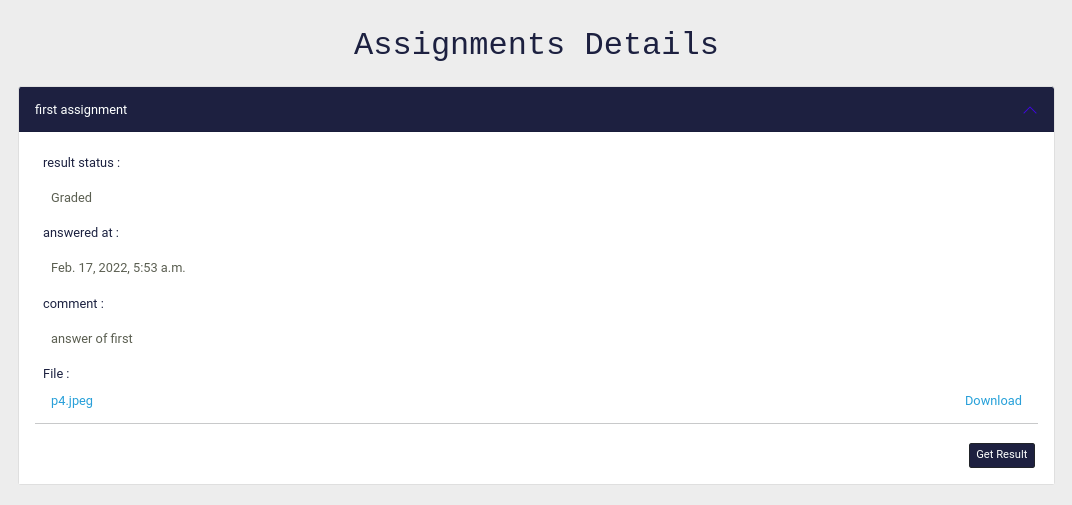
add-assignment



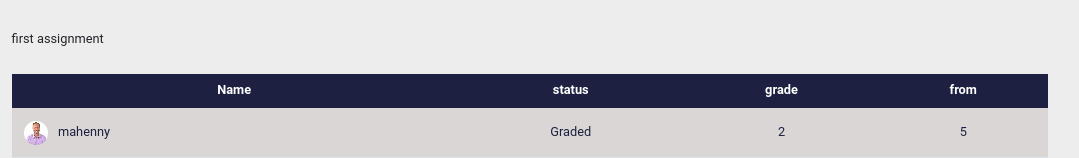
assignment-allanswers



assignment-allresults



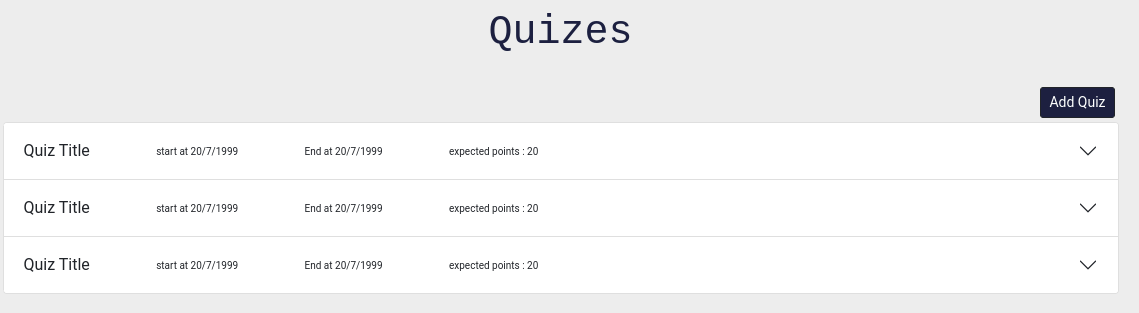
assignment-student-answer



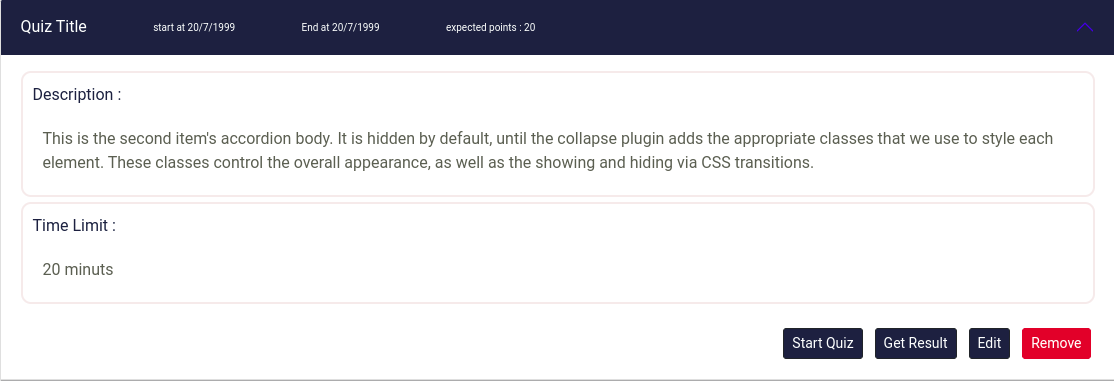
assignment-student-result

### Quizes :

* every quiz consists of a number of questions (mcq , true false) .
* doctor can add new quiz .
* doctor can grade assignment after the quiz final date finishes.
* doctor can see all students degrees in an quiz.
* doctor can pick random questions.
* doctor can add new questions or choose from ones added before .
* doctor can allow more than a trial for the quiz.
* student can send an answer.
* student can say his answer and right answers and his degree it as soon as he clicked finish button.



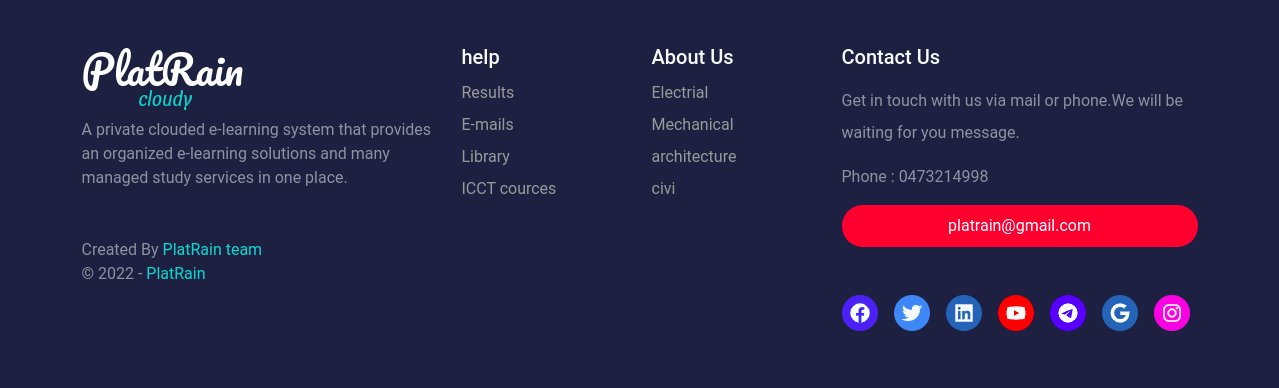
quizes



quiz

### contact :

the footer in the home page has all necessary links that user may need to contact the original website of the faculty:

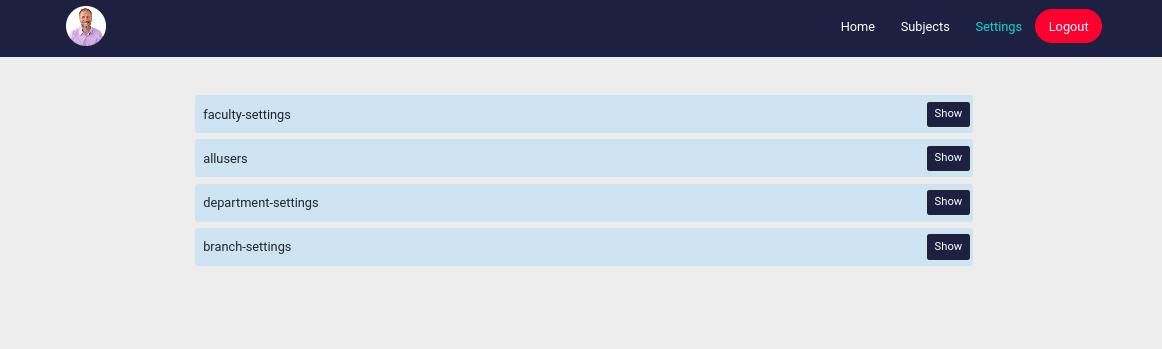


footerrrr

### settings :

if the user is Admin ,he will find a tap in the navigation bar called settings to lead him to the settings page where the admin can :

* add faculty information.
* edit or delete faculty.
* show,search,filter,add,edit,delete users.
* show,add,edit,remove department or branch.



settings

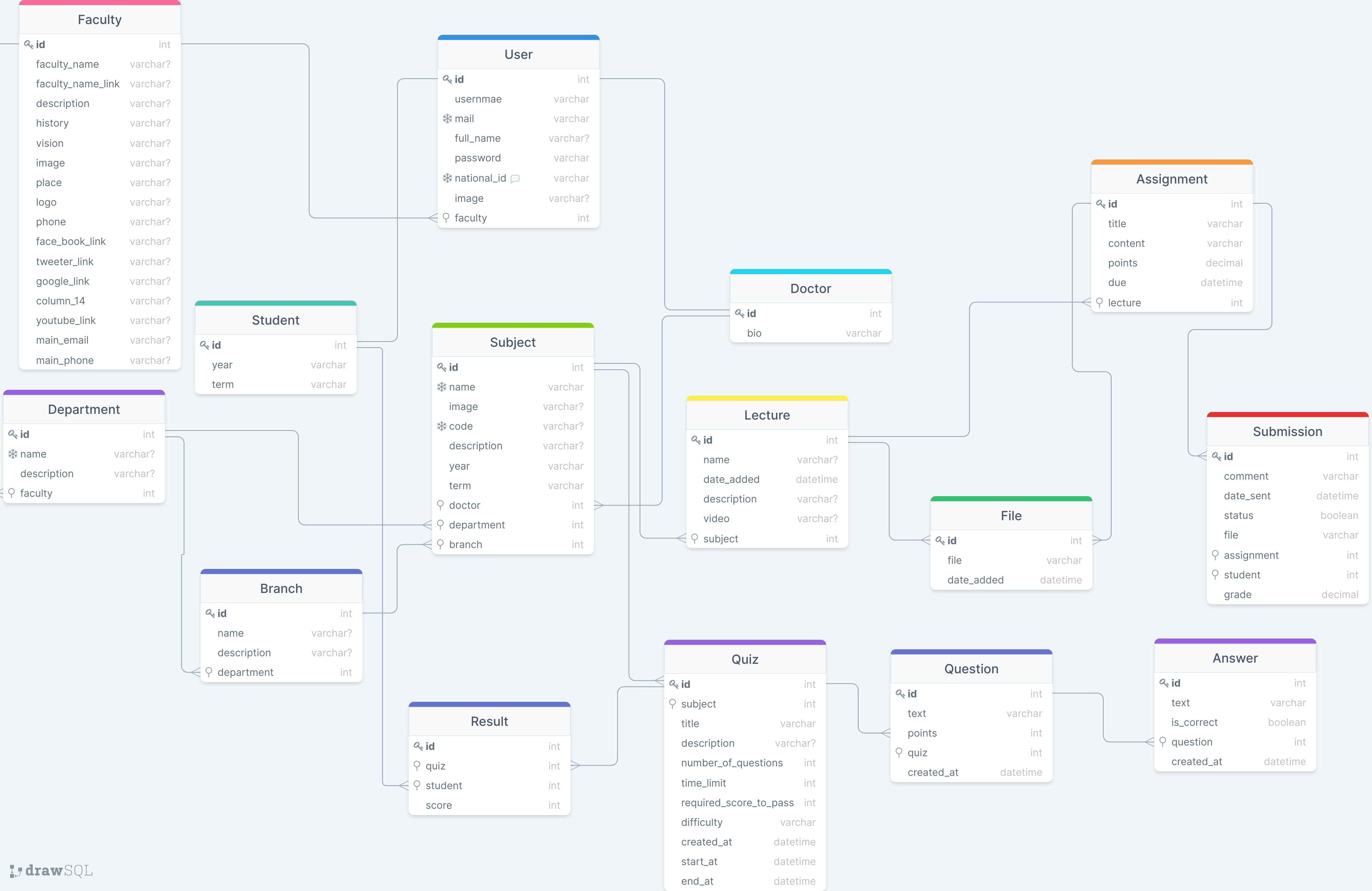
### chat room :

still working on it…….

### video conference :

still working on it…….

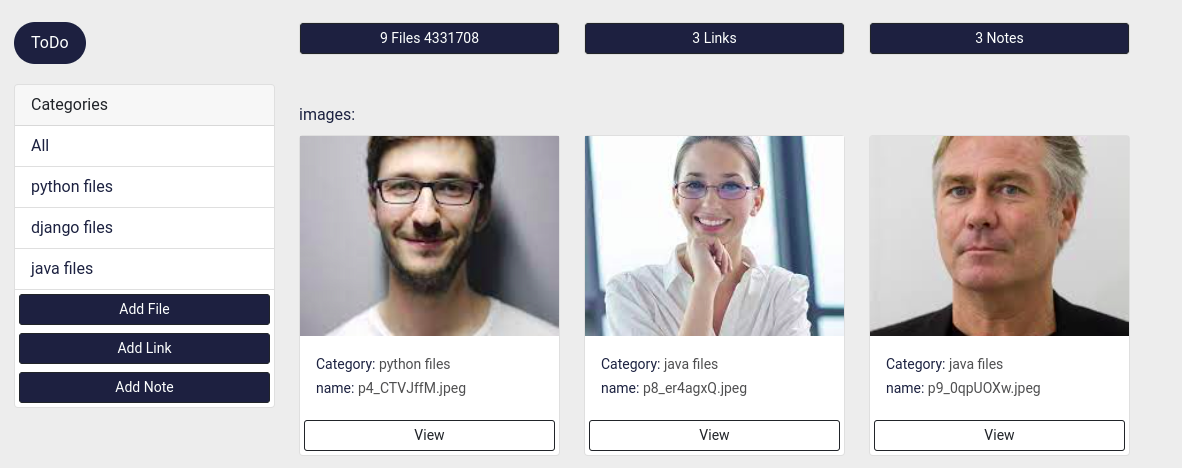
## ER model design :



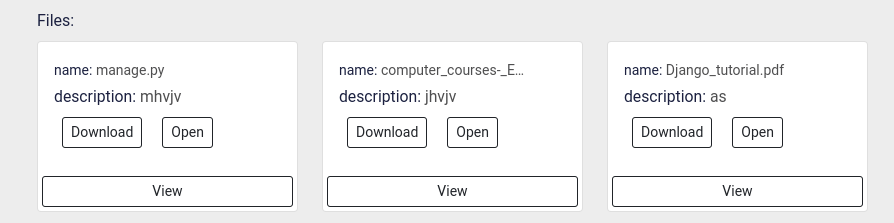
PlatRain-ER-p1

# User Data Manager :

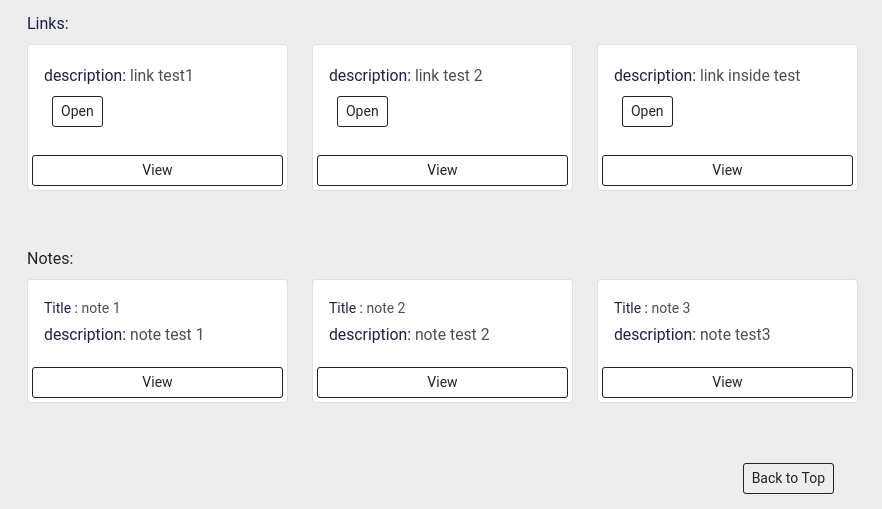
* user can create categories to keep his data organized in.
* user can upload files , images with names , sizes , description and can access them any time either by showing or downloading.
* user can store notes , links with description and access.
* to-do part to manage user’s day and tasks .
* user can easily get his own files,images,notes,links,tasks by date or , name , or category .



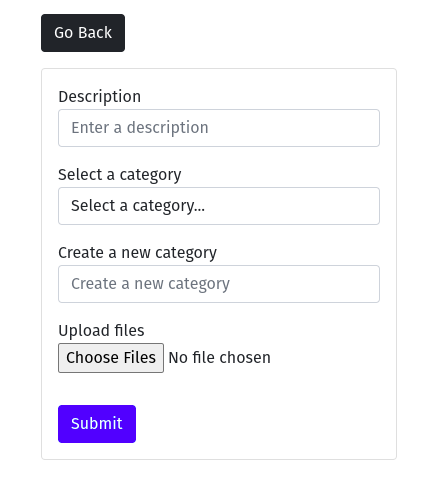
usermanager1



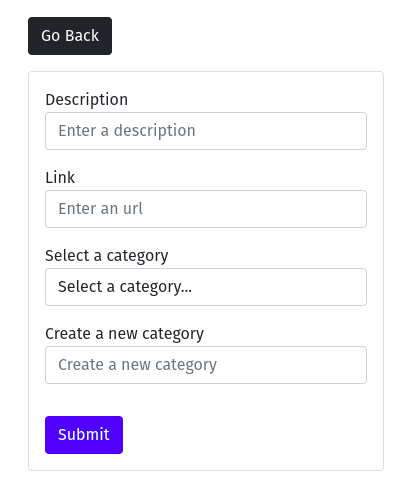
usermanager2



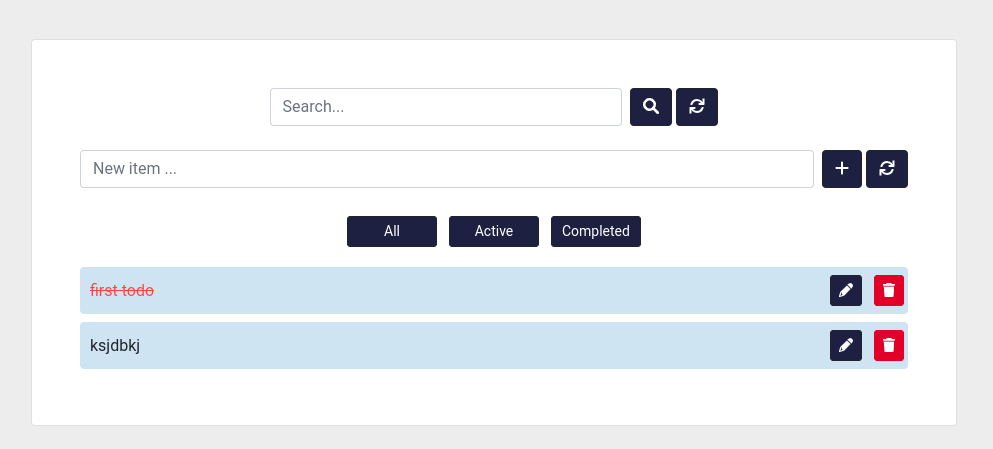
usermanager3



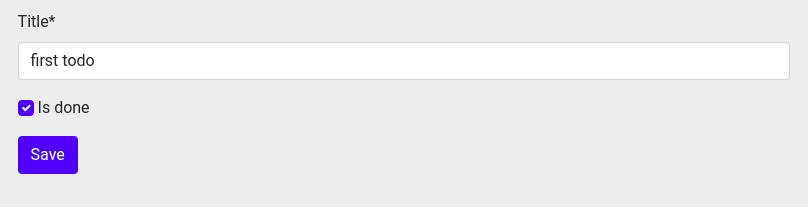
addfile



addlink

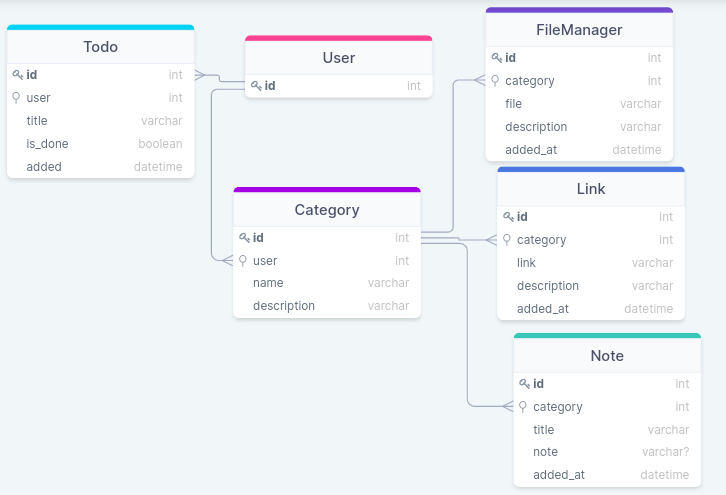


todo



todo-edit

## ER model design :



er2

# Technologies and tools used in the system :

## used tools and technologies versions :

* Django==4.0.2
* crispy-bootstrap5==0.6
* django-crispy-forms==1.14.0
* cryptography==36.0.2
* mysqlclient==2.1.0
* Pillow==9.0.1
* sqlparse==0.4.2
* urllib3==1.26.8
* html5
* css3
* bootstrap5

## Design pattern :

### MVT :

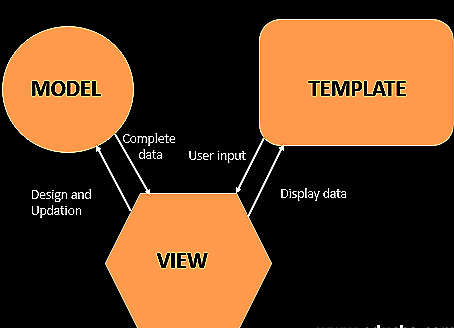
MVT is a software design pattern for developing a web application.

### MVT Structure has the following three parts :

​ **Model:** The model is going to act as the interface of your data. It is responsible for maintaining data. It is the logical data structure behind the entire application and is represented by a database (generally relational databases such as MySql, Postgres).

​ **View:** The View is the user interface — what you see in your browser when you render a website. It is represented by HTML/CSS/Javascript and Jinja files.

​ **Template:** A template consists of static parts of the desired HTML output as well as some special syntax describing how dynamic content will be inserted.



mvt

## Back-end implementation :

### Python :

* Python is an interpreted, object-oriented, high-level programming language with dynamic semantics. Its high-level built in data structures, combined with dynamic typing and dynamic binding, make it very attractive for Rapid Application Development, as well as for use as a scripting or glue language to connect existing components together. Python’s simple, easy to learn syntax emphasizes readability and therefore reduces the cost of program maintenance. Python supports modules and packages, which encourages program modularity and code reuse. The Python interpreter and the extensive standard library are available in source or binary form without charge for all major platforms, and can be freely distributed.

### Django :

* Django is a high-level Python web framework that encourages rapid development and clean, pragmatic design. Built by experienced developers, it takes care of much of the hassle of web development, so you can focus on writing your app without needing to reinvent the wheel. It’s free and open source.
* **why django ?**
  + Ridiculously fast.
  + Django was designed to help developers take applications from concept to completion as quickly as possible.
  + Reassuringly secure.
  + Django takes security seriously and helps developers avoid many common security mistakes.
  + Exceedingly scalable.
  + Some of the busiest sites on the web leverage Django’s ability to quickly and flexibly scale.
* API :
* API is the acronym for Application Programming Interface, which is a software intermediary that allows two applications to talk to each other. Each time you use an app like Facebook, send an instant message, or check the weather on your phone, you’re using an API.
* **Django Rest Framework**
* Django REST framework is a powerful and flexible toolkit for building Web APIs.

## Database implementation :

We used Mysql database and ORM to build our system database store.

**what is database**

A database is an organized collection of structured information, or data, typically stored electronically in a computer system. A database is usually controlled by a [database management system (DBMS)](https://www.oracle.com/database/what-is-database/#WhatIsDBMS). Together, the data and the DBMS, along with the applications that are associated with them, are referred to as a database system, often shortened to just database.

Data within the most common types of databases in operation today is typically modeled in rows and columns in a series of tables to make processing and data querying efficient. The data can then be easily accessed, managed, modified, updated, controlled, and organized. Most databases use structured query language (SQL) for writing and querying data.

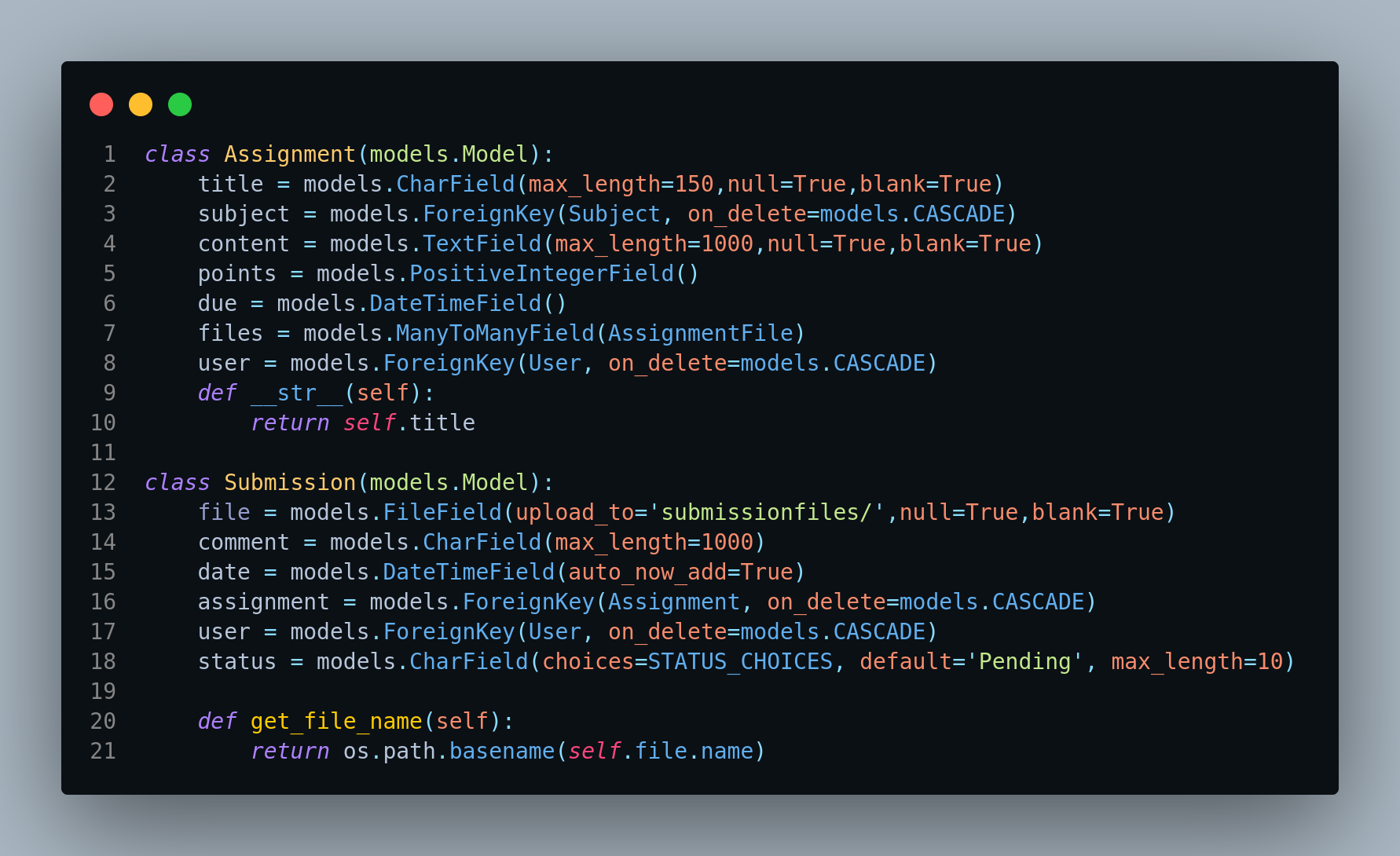
**SQL**

SQL is a programming language used by nearly all [relational databases](https://www.oracle.com/database/what-is-database/#relational) to query, manipulate, and define data, and to provide access control. SQL was first developed at IBM in the 1970s with Oracle as a major contributor, which led to implementation of the SQL ANSI standard, SQL has spurred many extensions from companies such as IBM, Oracle, and Microsoft. Although SQL is still widely used today, new programming languages are beginning to appear.

**ORM**

[Object-Relational Mapping](https://en.wikipedia.org/wiki/Object-relational_mapping) (ORM) is a technique that lets you query and manipulate data from a database using an object-oriented paradigm. When talking about ORM, most people are referring to a *library* that implements the Object-Relational Mapping technique, hence the phrase “an ORM” .

example shots from the system:



shot1

## Front End technologies :

### Html :

* HTML is the standard markup language for creating Web pages.
  + HTML stands for Hyper Text Markup Language.
  + HTML is the standard markup language for creating Web pages.
  + HTML describes the structure of a Web page.
  + HTML consists of a series of elements.
  + HTML elements tell the browser how to display the content.
  + HTML elements label pieces of content such as “this is a heading”, “this is a paragraph”, “this is a link”, etc.

### Css :

* CSS is the language we use to style a Web page.
  + CSS stands for Cascading Style Sheets.
  + CSS describes how HTML elements are to be displayed on screen, paper, or in other media.
  + CSS saves a lot of work. It can control the layout of multiple web pages all at once.
  + External stylesheets are stored in CSS files.

### Javascript :

* JavaScript is a scripting or programming language that allows you to implement complex features on web pages every time a web page does more than just sit there and display static information for you to look at displaying timely content updates, interactive maps, animated 2D/3D graphics, scrolling video jukeboxes, etc.
* It is the third layer of the layer cake of standard web technologies, two of which ([HTML](https://developer.mozilla.org/en-US/docs/Learn/HTML) and [CSS](https://developer.mozilla.org/en-US/docs/Learn/CSS)) we have covered in much more detail in other parts of the Learning Area.

### Bootstrap :

* **Bootstrap** is the most popular **CSS Framework** for developing responsive and mobile-first websites.
* We made use of it to simple and fast the implementation of front end.