King Saud University
Collage of Computer and Information Sciences
Computer Science Department

“Educational-resource Sharing”

CSC 496 – Final Report

Prepared by:

Muath Oudah Alhwaiti	436105620
Faisal Abdullah Almusalam	436106118
Abdullah Saad Alshomrani	435105864
Waleed Mohammed Alotaibi	436104727

Supervised by:

Prof. Khalil El Hindi

Software project for the degree of Bachelor in Computer Science

First/Second Semester 1441/1442

Autumn/Spring 20202

Table of Contents

I. Acknowledgements

II. Abstract

Prior knowledge and exploiting the available materials make our life easier. So there are many common problems students face during their university life one of the main problems is the lack of educational resources of their ongoing or upcoming courses. To make the student lives easier and more enjoyable, we intend to develop a website and app for education resource sharing, where students can share their own educational materials about any course in order to help other students obtain them, such as helpful links, video lectures, notes, pictures, syllabus, and past exams, etc. The idea is that the students of a course can create a web page that only registered users can access, and each course will have a schedule that contain all the important dates for the midterm exams and final exam. A register user can access the webpage to upload or download any educational material. Also, the registered users will be notified once a new material is uploaded. Users can evaluate the material and can report any material that they see inappropriate to the webpage master. More features will be added to the system if we see them useful.

The system will be implemented using Python and the Django Framework. Python is a powerful programming language that has a lot of capabilities that we need to develop our project. We intend to implement the system using "Django Framework" and we will use SQLite for database management.

III. Arabic Abstract *

من خلال المعرفة المسبقة واستغلال الموارد المتاحة تصبح حياتنا أسهل، فيوجد الكثير من المشاكل التي تواجه الطلاب أثناء مرحلتهم الجامعيه و واحده من أهم المشاكل التي تواجههم هي المصادر المتعلقة بالمقررات الدراسية للملتحقين بها أو المقررات التي سيلتحقون بها في المستقبل، ولتصبح حياتهم العلميه أسهل وأكثر متعه قررنا بإنشاء موقع لمشاركة المصادر المتعلقه بالمقرر يسمى " مشاركة المصادر التعليميه" وهو عباره عن برنامج يمكن الوصول له من خلال المتصفح يسمح للطلاب بمشاركة المصادر الخاصه بهم فيما بينهم على سبيل المثال الروابط المفيده، المحاضرات المسجله، الملاحظات، الصور، التعريف بالمقرر و الإختبارات السابقه ...إلخ في مكان واحد.
فالفكره هي أن الطلاب الملتحقين بالمقرر يمكنهم إنشاء صفحه عن المقرر يمكن الوصول لها من خلال الطلاب المسجلين فقط. وكل مقرر لديه جدول يحتوي على أهم المواعيد على سبيل المثال مواعيد إختبارات منتصف الفصل والإختبار النهائي. أعضاء الموقع يمكنهم الوصول إلى المقررات للإضافه أو الإطلاع على المحتوى الخاص بالمقرر، وأيضاً سيصل كل عضو ملتحق بالمقرر إشعار عندما يتم إضافة مرجع عن المقرر الملتحق به، الموقع يسمح للأعضاء بتقييم متحوى المقرر والتبليغ عن المحتوى الغير مرغوب فيه. سيتم إضافة الكثير من الخصائص للموقع عندما نرأ بأنها مناسبه.
سيتم إنشاء الموقع بإستخدام بايثون و جانقو، بايثون من اللغات القويه التي توفر الكثير من الميزات في تطوير تطبيقات الإنترنت كما سنستخدم اسكيوال لايت لإدارة قواعد البيانات

Chapter 1: Introduction

As students, we have a common problem of finding the relevant course resources that may help us better understand a course. Searching for a lecture material may take time, effort, and may fail to find all relevant material.

We tend to solve this by building a web-based app which allows students to interact with each other and share courses material.

A Web app could be built in many languages, such as Java, JavaScript, and C#.

We tend to build the app using Python which is a powerful language with many powerful tools such as Django which give us a variety of features that we can build on.

Python was developed in 1989 and according to “edureka.co”. Today it is considered one of the most widely used programming languages due to the following properties:

- Open-source programming language.
- Extensive support modules and community development.
- Easy integration with web services.
- User-friendly data structures.
- GUI-based desktop applications.

This project will focus on making use of these properties to build an environment for students to share their course resources using Django.

The unique thing about our project that it's made for KSU IT students and it can be later expanded to other departments and even universities.

1.1 Problem Statement

Student have always face a problem in finding educational resources that are relevant to their courses. Such resources may help them better understand the courses and achieve better results. Some courses material is either sold in bookstore, sometimes with high prices that student cannot afford, or exchanged privately between some students which prevents the rest of students from using them.

The availability of educational resources such as lecture notes, past exam papers, video lectures, books, tutorials is critical for understanding a new subject and obtaining good results specially in critical times such as these times where face to face lectures have been suspended by the University due to the Corona virus outbreak. Also, a student may miss a lecture or more due to personnel matters and need extra educational resources to catch up with their colleagues.

These are some of the examples that regular student could face which results in poor performance. Video lectures have always been helpful and are easy to understand. Videos lectures usually contain too much details and take a lot of time until you take what you need from it. You could be watching a 15min video and at the end the outcome is zero while there is a student watching a 3min video which address the issue you need to understand.

Another problem students could be facing is knowing details about their upcoming courses that they will take later, student in general like having a brief outlook about what their upcoming courses looks like.

So they can be well prepared and know what there are facing in the feature, they ask question like:

- Does the course require knowing a specific language?
- How is the grading like?
- Is there a project?
- How do the past exams look like?

These questions are always ringing in the student ears, sometimes not knowing what the upcoming course looks like my affect the student grading or result.
We will address all these problems by developing a web-based system and apps that allow student to upload and download educational resources. The system will allow the user to search for the relevant resources as each resource will be associated with a set of keywords provided by the person who upload the resource. Also, the system will allow users to provide reviews of the available resources that may help other students decide on the material they want to download and use. Moreover, the system will collect statistics about each resource for example the number of times a resource was used, and the average review score etc.

1.2 Goals and Objectives

. Our ojective is to create a web-based system and app for sharing study resources between students to make their academic lives easier and more fulfilling. The system will allow the students to search for the relevant material via the keywords associated with each resource.

Goals:

We will achieve our objective by achieving the following goals

1. Design a Web-based system that allows users to create a webpage for each course. The course's page will contain a list of educational resources for that course ordered according to the average user reviews.
2. Design a database for storing the relevant educational resources such as video lectures, lectures notes, exam papers, assignments, quizzes etc.
3. Provide an interface through which the user can upload a new resource or download an available resource.
4. Provide a search engine to allow the user to search for the relevant sources based on a set of keywords.
5. source, or download an available resource.
6. Provide a search engine to allow the user to search for the relevant sources based on a set of keywords.

We will develop the system in steps

- we will write requirement document, including (User requirement and System requirements) then writing the software system requirement which is classified into functional and non-functional requirements, then construct the use case model and description, and UML diagrams.

- Then we will start the implementation using python "Django" and other language like HTML ..etc. which is needed to design User interface and make it easy to Develop the web app, also setting the database using "SQLite" that'll save and manage the data. After that we will create a prototype that we'll test it and debug it, the last thing we'll do is maintenance and make sure that everything is correct.

- Then we will test and evaluate our system. Our success lies in make an Integrated web that have features That helps students for a better understanding of their courses and Developing their academic skills, and make it easy to share resources by upload it and easy to read it and download it.
- Finally, we will document the system and produce the final report.

1	2	3	4	5	6	7	8	9	10	11	12	13	14
Building Background & Literature Review													
						Get Acquainted to Weka							
									Draw the flowchart of the proposed algorithm				
												Documentation	
Phase Two:													
1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**12**	**13**	**14**
Implementation of the Proposed Algorithm													
								Evaluation and extracting results					
												Documentation	

1.3 Solution

We mentioned before that we focus on making the website very easy to use and this could be by making a simple interface to the users, below are some of the solutions that we desire to do in order to make the website look easier , simpler and active.

Category:

As we said before that our goal to make the website easy and fast. This could be done by making a category for each course which will act as an archive, were student can add the course recourses. some categories for each course: Image, Notes, Slides, Videos.

Sharing:

Students tend share slides, videos, notes, images, and we will allow student to share any content that they think will benefit others at any extension.

Comminution:

Small chat box is very helpful, student could alert each other if there is very important part in a note they shared.

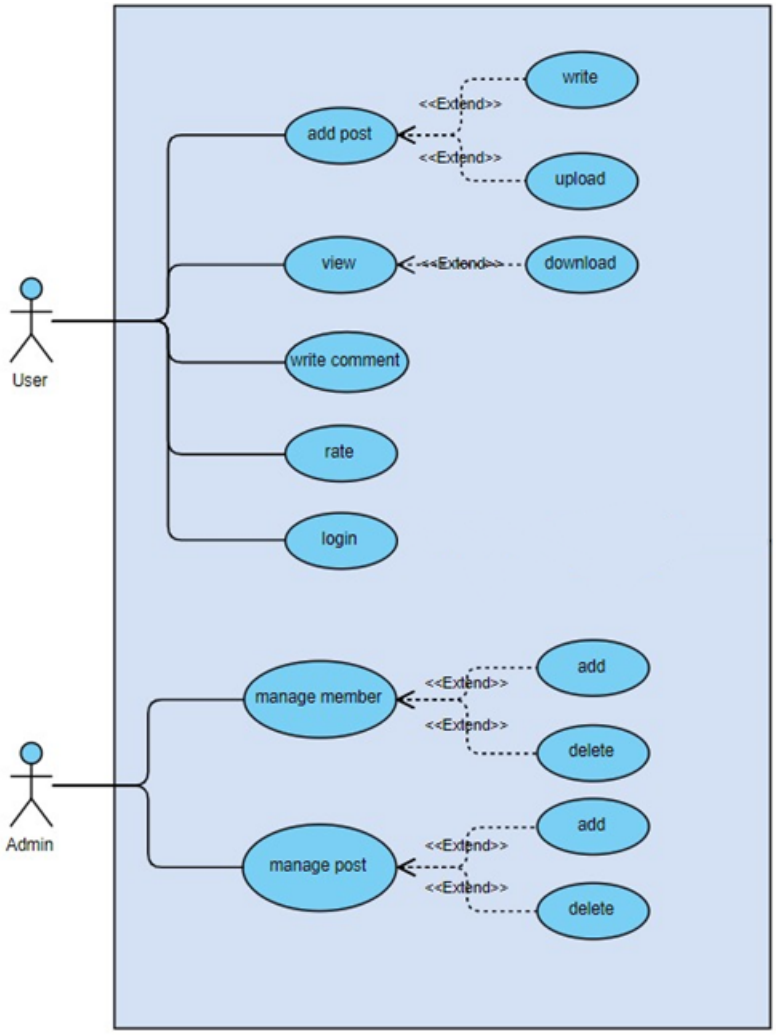
Accessibility:

Our goal is to make the website reachable and accessible only to specific users, this move will help the admin to have full control over users' content that they share.

This could be done using Django Admin. The users will need to have accounts to access the systems.

1.4 Research Scope

2 The following use case diagram summarizes the main services that our system will provide and the actors that will have access to them.



Chapter 2: Background

A network can be defined as a group of computers and other devices connected in some ways to be able to exchange data. Computer networks are often classified in function of the geographical area that they cover:

- LAN: a local area network typically interconnects hosts that are up to a few or maybe a few tens of kilometers apart.
- MAN: a metropolitan area network typically interconnects devices that are up to a few hundred kilometers apart.
- WAN: a wide area network interconnect hosts that can be located anywhere on Earth.

The network architecture: A computer network consists of a collection of computers, printers and other equipment that is connected together so that they can communicate with each other, they can be broadly classified as using either a peer-to-peer or client/server architecture, the deference between them in the traditional client-server model, hosts act either as servers or as clients and a server serves a large number of clients. In the peer-to-peer model, all hosts act as both servers and clients and they play both roles. Peer-to-peer networks are more commonly implemented where less than ten computers are involved and where strict security is not necessary. All computers have the same status, hence the term 'peer', and they communicate with each other on an equal footing.

Client/server networks are more suitable for larger networks. A central computer, or 'server', acts as the storage location for files and applications shared on the network. Usually the server is a higher than average performance computer. The server also controls the network access of the other computers which are referred to as the 'client' computers.

Peer-to-Peer Networks VS Client/Server Networks	
Peer-to-Peer Networks	Client/Server Networks
Easy to set up	More difficult to set up
Less expensive to install	More expensive to install
Can be implemented on a wide range of operating systems	A variety of operating systems can be supported on the client computers, but the server needs to run an operating system that supports networking
More time consuming to maintain the software being used (as computers must be managed individually)	Less time consuming to maintain the software being used (as most of the maintenance is managed from the server)
Very low levels of security supported or none at all. These can be very cumbersome to set up, depending on the operating system being used	High levels of security are supported, all of which are controlled from the server. Such measures prevent the deletion of essential system files or the changing of settings
Ideal for networks with less than 10 computers	No limit to the number of computers that can be supported by the network
Does not require a server	Requires a server running a server operating system
Demands a moderate level of skill to administer the network	Demands that the network administrator has a high level of IT skills with a good working knowledge of a server operating system.

Internetwork A network of networks is called an internetwork, or simply the internet. It is the largest network in existence on this planet. The internet hugely connects all WANs and it can have connection to LANs and Home networks.
The transmission Control Protocol (TCP) is one of the most important protocols of Internet Protocols suite. It is most widely used protocol for data transmission in communication network. It is a connection-oriented transport service, it provides end-to-end reliability, resequencing, and flow control.
Feature:

- TCP is reliable protocol. That is, the receiver always sends either positive or negative acknowledgement about the data packet to the sender, so that the sender always has bright clue about whether the data packet is reached the destination, or it needs to resend it.
- TCP ensures that the data reaches intended destination in the same order it was sent.
- TCP is connection oriented. TCP requires that connection between two remote points be established before sending actual data.
- TCP provides error-checking and recovery mechanism.
- TCP provides end-to-end communication.
- TCP provides flow control and quality of service.
- TCP operates in Client/Server point-to-point mode.
- TCP provides full duplex server, i.e. it can perform roles of both receiver and sender.

The User Datagram Protocol (UDP) is simplest Transport Layer communication protocol available of the TCP/IP protocol suite. It involves minimum amount of communication mechanism. UDP is said to be an unreliable transport protocol but it uses IP services which provides best effort delivery mechanism. In UDP, the receiver does not generate an acknowledgement of packet received and in turn, the sender does not wait for any acknowledgement of packet sent. This shortcoming makes this protocol unreliable as well as easier on processing.

UDP deploy where the acknowledgement packets share significant amount of bandwidth along with the actual data. For example, in case of video streaming, thousands of packets are forwarded towards its users. Acknowledging all the packets is troublesome and may contain huge amount of bandwidth wastage. The best delivery mechanism of underlying IP protocol ensures best efforts to deliver its packets, but even if some packets in video streaming get lost, the impact is not calamitous and can be ignored easily. Loss of few packets in video and voice traffic sometimes goes unnoticed.

Features:

- UDP is used when acknowledgement of data does not hold any significance.
- UDP is good protocol for data flowing in one direction.
- UDP is simple and suitable for query-based communications.
- UDP is not connection oriented.
- UDP does not provide congestion control mechanism.
- UDP does not guarantee ordered delivery of data.
- UDP is stateless.
- UDP is suitable protocol for streaming applications such as VoIP, multimedia streaming.

Few of Application layer protocols are described below:

The Domain Name System (DNS) works on Client Server model. It uses UDP protocol for transport layer communication. DNS uses hierarchical domain-based naming scheme. The DNS server is configured with Fully Qualified Domain Names (FQDN) and email addresses mapped with their respective Internet Protocol addresses. A DNS server is requested with FQDN and it responds back with the IP address mapped with it. DNS uses UDP port 53.

The File Transfer Protocol (FTP) is the most widely used protocol for file transfer over the network. FTP uses TCP/IP for communication, and it works on TCP port 21. FTP works on Client/Server Model where a client requests file from Server and server sends requested resource back to the client. FTP uses out-of-band controlling i.e. FTP uses TCP port 20 for exchanging controlling information and the actual data is sent over TCP port 21.

Hyper Text Transfer Protocol (HTTP) is the foundation of World Wide Web. Hypertext is well organized documentation system which uses hyperlinks to link the pages in the text documents. HTTP works on client server model. When a user wants to access any HTTP page on the internet, the client machine at user end initiates a TCP connection to server on port 80. When the server accepts the client

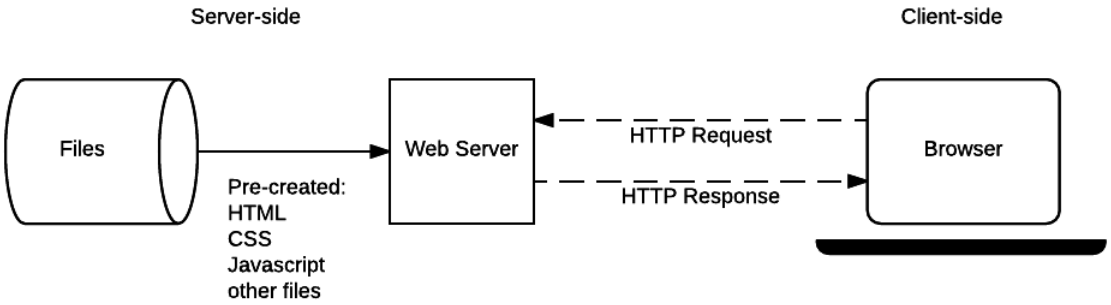
request, the client is authorized to access web pages. To access the web pages, a client normally uses web browsers, who are responsible for initiating, maintaining, and closing TCP connections. HTTP is a stateless protocol, which means the Server maintains no information about earlier requests by clients.

The Simple Mail Transfer Protocol (SMTP) is used to transfer electronic mail from one user to another. This task is done by means of email client software (User Agents) the user is using. User Agents help the user to type and format the email and store it until internet is available. When an email is submitted to send, the sending process is handled by Message Transfer Agent which is normally comes inbuilt in email client software.

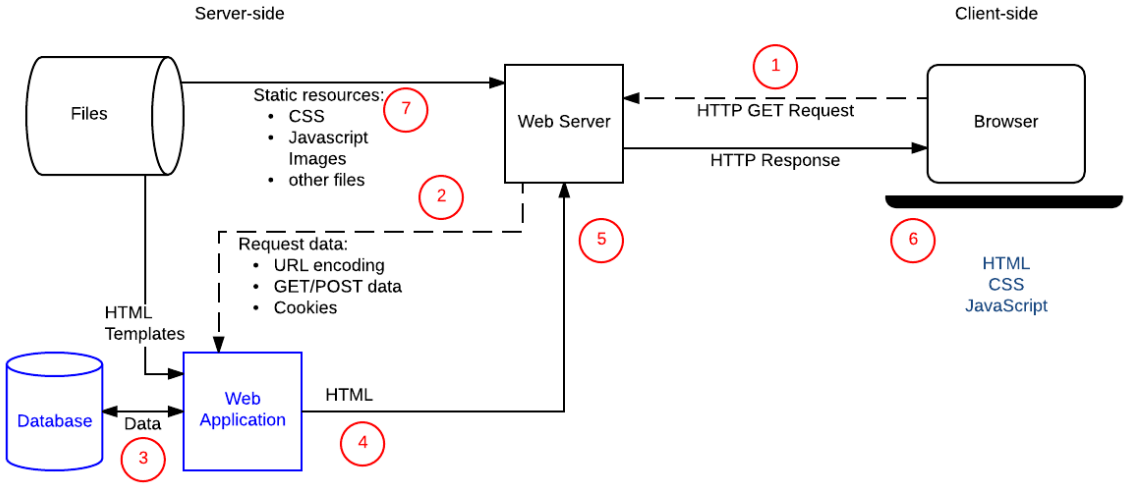
Message Transfer Agent uses SMTP to forward the email to another Message Transfer Agent (Server side). While SMTP is used by end user to only send the emails, the Servers normally use SMTP to send as well as receive emails. SMTP uses TCP port number 25 and 587.

Static and Dynamic Websites:

A static website includes a series of HTML and CSS files where each HTML file represents a web page of the website. In static sites, the format and number of pages are fixed which offers fixed information to the client. Even though two webpages of the site consist of same content such as header or footer, both webpages would have two different versions of the same. So, in order to make changes in the header or footer section, you need to make the changes in both the files manually. This is how sites were created in the early days of the internet.



Dynamic websites use server technologies to create web page dynamically when someone visits the page. When a user visits a certain web address, the server looks into the bunch of variety of pieces of information to write a single united webpage which is then displayed to the user. In a nutshell, dynamic websites can change the webpage dynamically while the user is accessing the page on the browser. Server-side programming (JSP, PHP, Python) is used to change page's content at run time.



Web Application:

The term web application refers to a software system that provides a user interface through a web browser.

Examples of web applications include blogs, online shopping, search engines, etc.

Web applications can be simple consisting of only static web pages or they can be dynamic and interactive.

Static web pages are stored in the file system of web server usually displays the same information to all visitors.

Web applications use a combination of server-side scripts (PHP and ASP) to handle the storage and retrieval of the information,

and client-side scripts (JavaScript and HTML) to present information to users.

How a web application works?

Web applications are usually coded in browser-supported language such as JavaScript and HTML as these languages rely on the browser to render the program executable.
Some of the applications are dynamic, requiring server-side processing. Others are completely static with no processing required at the server.
The web application requires a web server to manage requests from the client, an application server to perform the tasks requested, and, sometimes, a database to store the information.
Application server technology ranges from ASP.NET, ASP and ColdFusion, to PHP and JSP.

Database: A database system is an integrated collection of related files, along with details of the interpretation of the data contained therein. Basically, database system is nothing more than a computer-based record keeping system i.e. a system whose overall purpose is to record and maintain data.

Database management system (DBMS) is a software system that allows access to data contained in a database. The objective of the DBMS is to provide a convenient and effective method of defining, storing and retrieving the information contained in the database. The DBMS interfaces with the application programs, so that the data contained in the database can be used by multiple applications and users. In addition, the DBMS exerts centralized control of the database, prevents fraudulent or unauthorized users from accessing the data, and ensures the privacy of the data.

SQL is Structured Query Language, which is a computer language for storing, manipulating and retrieving data stored in a relational database.

SQL is the standard language for Relational Database System. All the Relational Database

Management Systems (RDMS) like MySQL, MS Access, Oracle, Sybase, Informix, Postgres and SQL Server use SQL as their standard database language.

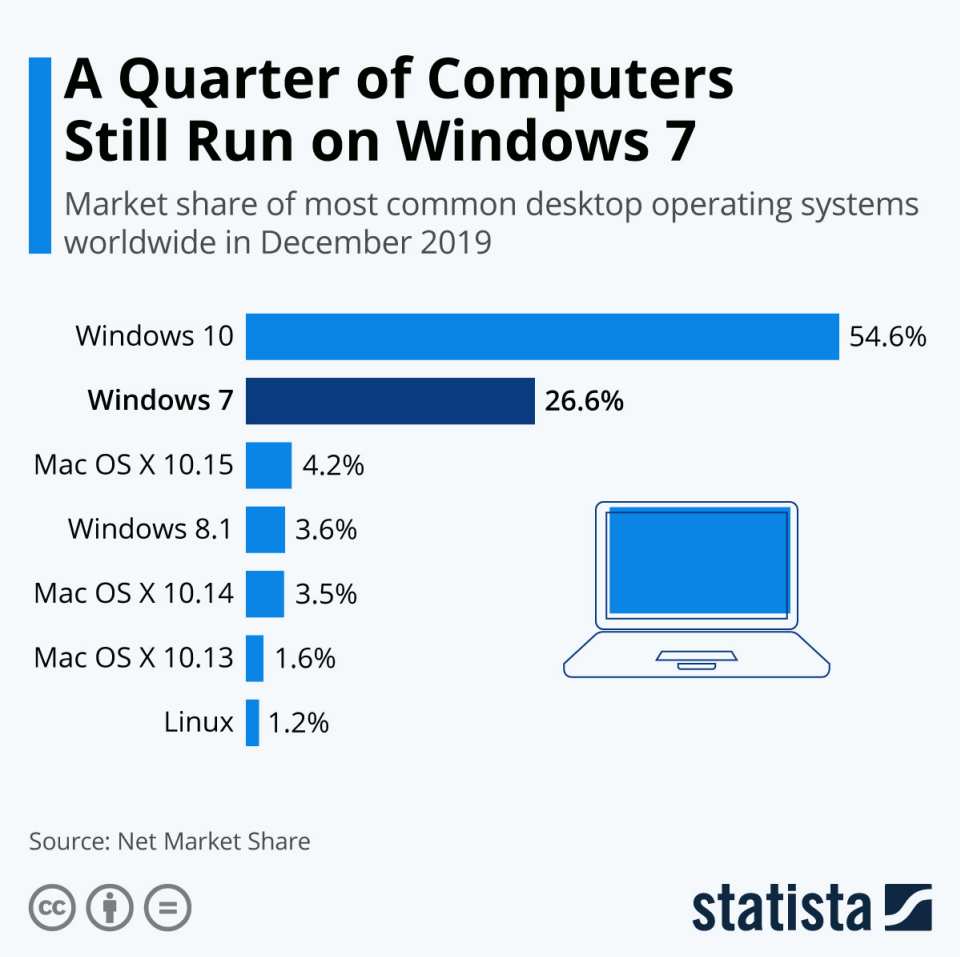
The operating system:

An operating system is the most important software that runs on a computer. which acts as an interface between the end user and computer hardware.
Every computer must have at least one OS to run other programs.
An application like Chrome, MS Word, Games, etc needs some environment in which it will run and perform its task.
The OS helps you to communicate with the computer without knowing how to speak the computer's language.
It is not possible for the user to use any computer or mobile device without having an operating system.

Types of operating systems:

Operating systems usually come pre-loaded on any computer you buy. Most people use the operating system that comes with their computer,
but it's possible to upgrade or even change operating systems.
The three most common operating systems for personal computers are Microsoft Windows, macOS, and Linux.

Here is a statistic about most common desktop operating system Worldwide.



Search Engines:

Today, Google and other search engines are smarter than ever, they use machine learning to help process and rank information, and can understand natural human speech. But the internet wasn't always so easy to navigate!

There was a time when you had to know the exact wording of a website's title to find it. Search results were riddled with spam. Getting new content indexed by the search engines could take weeks to complete.

Search engines certainly have changed! In this timeline, learn about the history and evolution of search engines from 1990 to the present day.

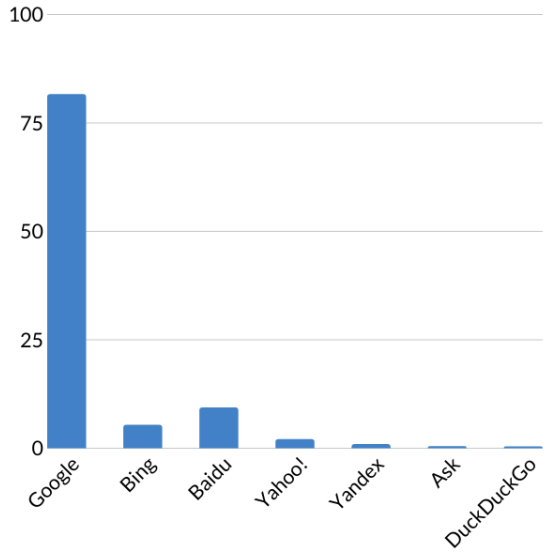
Let's take a look at the most popular search engines.

TOP SEARCH ENGINES

GOOGLE

81.5%

Google is the most popular search engine with a stunning 81.5% market share compared to 5.29% of second in place Bing.



RELIABLESOFT.NET @RELIABLESOFTNET

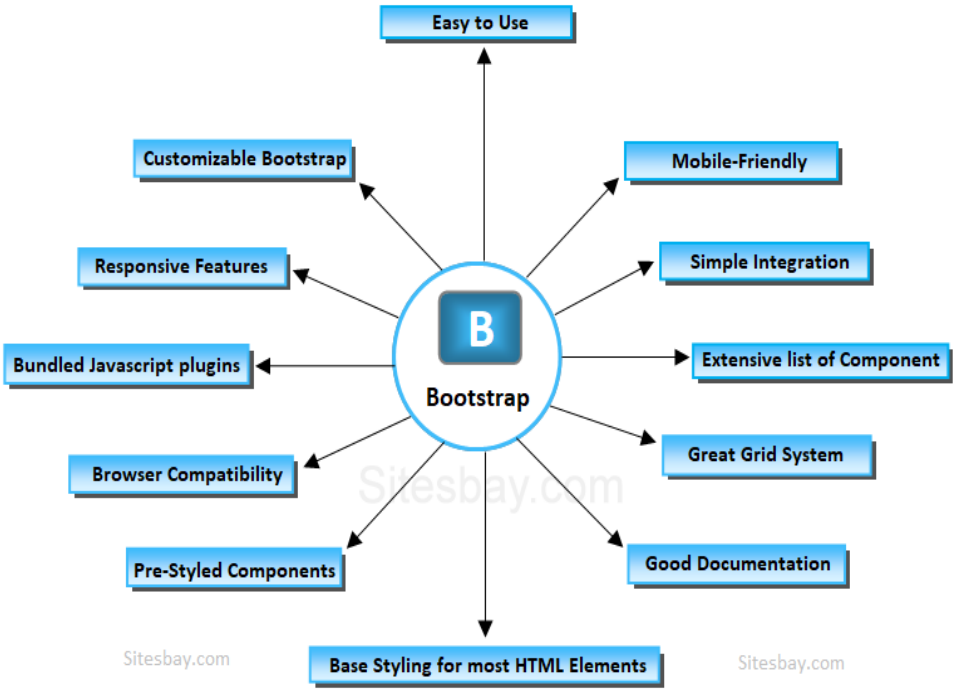
Django Tools and Packages:

Bootstrap Framework: is a free and open-source CSS framework that makes it easier to create websites and web application user interfaces.

Bootstrap is especially useful as a base layer of CSS to build sites with responsive web design.

What's a bootstrap Features?

It is a web framework that focuses on simplifying the development of informative web pages. Let's take a look at some of the main Features.



Easy to use: Anybody with just basic knowledge of HTML and CSS can start using Bootstrap.

Responsive features: Bootstrap's responsive CSS adjusts to phones, tablets, and desktops.

Mobile-Friendly: Mobile-first approach: In Bootstrap 3, mobile-first styles are part of the core framework.

Simple Integration: Bootstrap can be simply integrated along with distinct other platforms and frameworks, on existing sites and new ones too and one more things you can also utilize particular elements of Bootstrap along with your current CSS.

Pre-styled Components: Bootstrap approaches with pre-styled components for alerts, dropdowns, nav bars, etc.

Customizable Bootstrap: The Bootstrap can be customized as per the designs of your project.

Browser compatibility: Bootstrap is compatible with all modern browsers (Chrome, Firefox, Internet Explorer, Safari, and Opera).

Great grid system: Bootstrap is built on responsive 12-column grids, layouts and components. Whether you need a fixed grid or a responsive, it's only a matter of a few changes.

Bundled JavaScript plugins: The components such as drop down menu are made interactive with the numerous JavaScript plugins bundled in the bootstrap package.

Extensive list of components: Whether you need drop down menus, pagination or alert boxes, Bootstrap has got your covered. Some of the components pre styled are; Dropdowns, Button Groups, Navigation Bar, Breadcrumbs, Labels & Badges, Alerts, Progress Bar, And many others.

Base styling for most HTML elements: A website has many different elements such as headings, lists, tables, buttons, forms, etc. The HTML elements for which styles are provided are; Typography Code, Tables, Forms, Buttons, Images, Icons.

Good documentation: Not only does Bootstrap offer styling for almost every element a typical website or web application requires, it also provides a great documentation with examples and demo that only make it more easier for even someone new.

Let's take a look to some of tool and Packages that we a tend to use:

Login Tool: (Django-Allauth): Django-Allauth is a reusable Django application that solves our registration and authentication needs. Whether we need a local or social registration system, Django-Allauth has you covered.

The project supports multiple authentication schemes, such as username or email address. Once a user has signed up, multiple strategies are supported for account verification ranging from none to email verification.

Multiple social and email accounts are also supported. Pluggable signup forms are also supports which allows asking additional questions during registration.

Open Source Django Packages:

Django is built around the concept of reusable apps: self-contained packages that provide re-usable features.

we can build our site by composing these reusable apps, together with our own site-specific code.

There's a rich and varied ecosystem of reusable apps available for your use.

Django-Colorfield Packages: simple color field for your models with a nice color-picker in the admin-interface. Run pip install Django-colorfield Add colorfield to settings. INSTALLED_APPS, Run python manage.py collectstatic

Django-typogrify: Typogrify is a collection of Django template filters that help prettify your web typography by preventing ugly quotes and widows and providing CSS hooks to style some special cases.

Django-simple-history: This package keeps a history of record changes. If a client has access to the database through the admin panel,

they might change something by mistake or may want to return to the previous version of the text.

Chapter 3: Literature Review

Python is a fast-growing language, and it became popular because its highly productive compared to others. Many famous apps that we see today is built by python like: Quora, Spotify, YouTube, Yahoo, Nasa, Instagram, Redditt and Drobox.

Since the project team are students and still learning, we choose Python over other languages because it has an active community support, and easy to learn in short time.

Other than that if we compare C++, to python, python has a massive advantage in term of portability, Which mean you simply can use the python code on any operating system while C++ can't.

And in comparing between java and python, java is fast, and you need to define the data types, while python is dynamic.

Python code is always shorter than java, java tend to use curly brackets and semicolon while python not.
But in the end all of the languages could be helpful and give the output that we need.

Django or flask?

Python has strong web development tools like, flask, Django, Cherrypie.
Talking about Django, it allows out of the box functionality, "manage.py" can perform the most framework like, starting the development of the web server creating admin, collecting static files etc…
Django build large and complex web apps while flask build lightweight web applications.
Flask is minimal framework, but that doesn't mean it I can't build big apps.
Netfilx, Reddit, Uber are built with Flask frame work.
A key advantage for Flask over Django is the flexibility, flexibility here means that the developer can implement everything exactly as they want it, using a huge range of external libraries and add-ons, making it flexible and extensible, while Django has Built-in feature and modules, which offer less freedom.

Comparisons with similar web websites:

Koizat vs Educational-Resource Sharing:

"Koizat.com" is a web app that allows students to add only images of their past exams.
There is no restriction in entering the website and seeing the content as well as adding image.
One of the main problem we faced when we used this website is that sometimes there is unwanted content, like banter images or images that are not related, this becomes a weakness to "koizat.com" .

Our project will give the students the ability to add anything is related to the course whether its an image, note, pdf etc…

We tend to restrict entrance to the website and make it reachable only to registered users with their KSU id and password, we choose to restrict it in this way to make the website safe for the students, if a student add unwanted content for banter he knows that he is reachable to the university by his student ID.

StuDocu:

StuDocu is a crowdsourced online learning and sharing platform for students where students can share study material.

StuDocu is mainly used for book summaries, lecture notes and exam questions.

Their business model is based on subscription access for premium documents

StuDocu allows users to upload past exams from their university courses and, in most cases, the universities own the copyright on these documents and, as a result, requests are often made to remove these types of documents.

The web has several advantage let's take some of them :

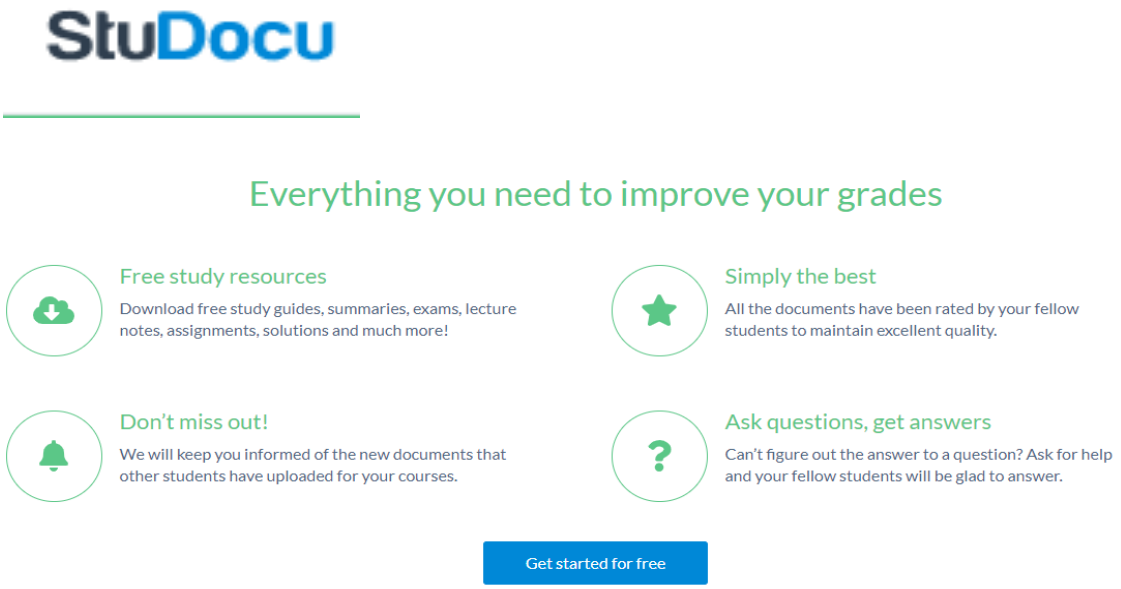
- It has large amount of contents .
- It has a content rating feature.

- It has a small chat for each post.

And now let's take some of the disadvantage and the differences between us:

- one of the main disadvantage of the web is, it has paid Premium account that open all the contents for you, and Not all students have the ability to pay our web is totally free for all content .

- The web contains several universities Which affects the quality of the content If you search for references on math for example You will find all the references for all universities, This causes messiness in the content in general, Our web is directed specifically to King Saud University



Koofers:

Koofers is a private company, founded in 2008 and headquartered in Reston, VA (a part of the Washington, DC metropolitan area). Koofers is an interactive community that serves the academic needs of college students through information sharing. College students have access to the service by creating an account with their e-mail address.

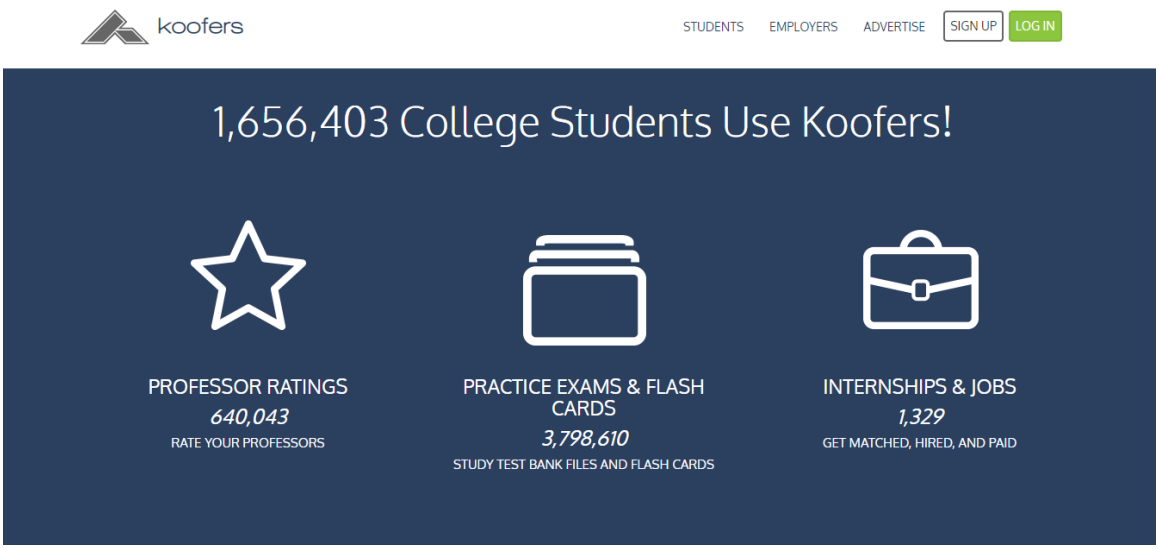
Koofers' services span the academic calendar from course selection through final exams, and include interactive flashcards, course & instructor ratings, professors' grading histories, and an online library for sharing past exams and study materials.virtual exam.

Let's take some of the advantages :

- it has a large amount of contents.
- it has feature called "Take as Test" which allow you to take a virtual exam .
- every account has profile that Contains all user activities.
-

Let's take some of the disadvantages:

- the post that has low rating should go down of the page.
- All universities are grouped together this creates clutter in the content.



What is our benefit from this project:

Python in general have proven itself in many fields not only in web developing, AI ,machine learning, analyzing , advanced computing and much more.
Learning python and taking our graduation project in it will give us a boost to learn other libraries to expand our knowledge.

Chapter 4: System Analysis

We will develop the system in such a way the fulfils many functional and non-functional requirements

4.1 Functional Requirements

- The user should be able to add content.
- The user should be able to delete only the content he have added.
- The user should be able to rate the content.
- The user should be able to see who sent the content.
- The admin have the ability to delete content.
- The admin have the ability to restrict entrance.

4.2 Non-Functional Requirements

- The website will be developed using python Django.
- The website will be secure by allowing only registered KSU student to enter.
- The content uploaded have the extension"docx,pdf,img,png,mp4,ppt,jpg,txt"
- The website interface of the website will be support English language.
- The website will not allow the user to upload a file over 500mb.
- The website will have a responsive and friendly user interface.

Chapter 5: System Design *

In this chapter we clearly describe our software design using diagrams.

5.1 System use-cases:

Use Case ID:	1		
Use Case Name:	Create an Account		
Actors:	Admin	**Trigger:**	Clicks the 'Create Account' button.
Precondition:	No account with same information is in the system.		
Post Condition:	User account created and added to database.		
Description:	Enable to create an account to access the system.		
Normal Flow:	1- Click on the "create account" button. 2- The admin enter the user all information needed. 3- Enter the information. 4- The system checks if all information valid and there is no account with the same information. The system creates the account.		

Use Case ID:	2		
Use Case Name:	Login		
Actors:	User	**Trigger:**	Enter user information and login
Precondition:	The user cant enter the website		
Post Condition:	User enter the website		
Description:	Enable entrance to website		
Normal Flow:	1. User enter his id and password that has been sent to his email. 2. The system pop up “you successfully entered”.		
Alternative Flow:	1. User enter id and password. 2. The system pop up “your id and password don’t match our record please contact system admin”.		

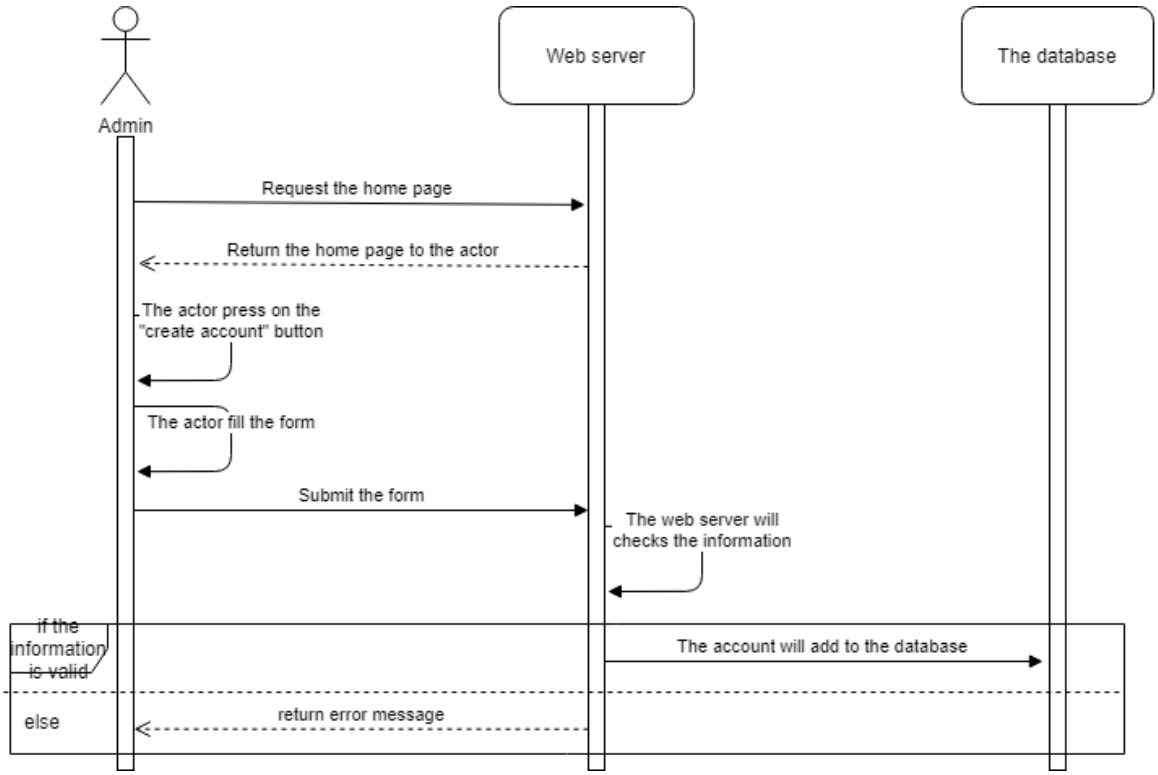
Use Case ID:	3		
Use Case Name:	Upload a file		
Actors:	Admin, User	**Trigger:**	Clicks the ‘upload’ button.
Precondition:	No files added.		
Post Condition:	User and admin upload successfully.		
Description:	Enable adding files to the website.		
Normal Flow:	1- Clicks on “upload” button. 2- Admin and user chose a specified file. 3- Then they click on upload. 4- A message pop up “file uploaded succecfully”		
Alternative Flow:	1- “file is not supported, please choose other files” 2- “file space is larger than 500mb, please choose other files”		

Use Case ID:	4		
Use Case Name:	Delete user		
Actors:	Admin	**Trigger:**	Clicks the ‘delete account’ button.
Precondition:	Account isn’t deleted.		
Post Condition:	User account successfully deleted.		
Description:	Deleting user.		
Normal Flow:	1- the admin choose specified user to delete. 2- Clicks on delete user. 3- Message pop up “user successfully deleted ”.		

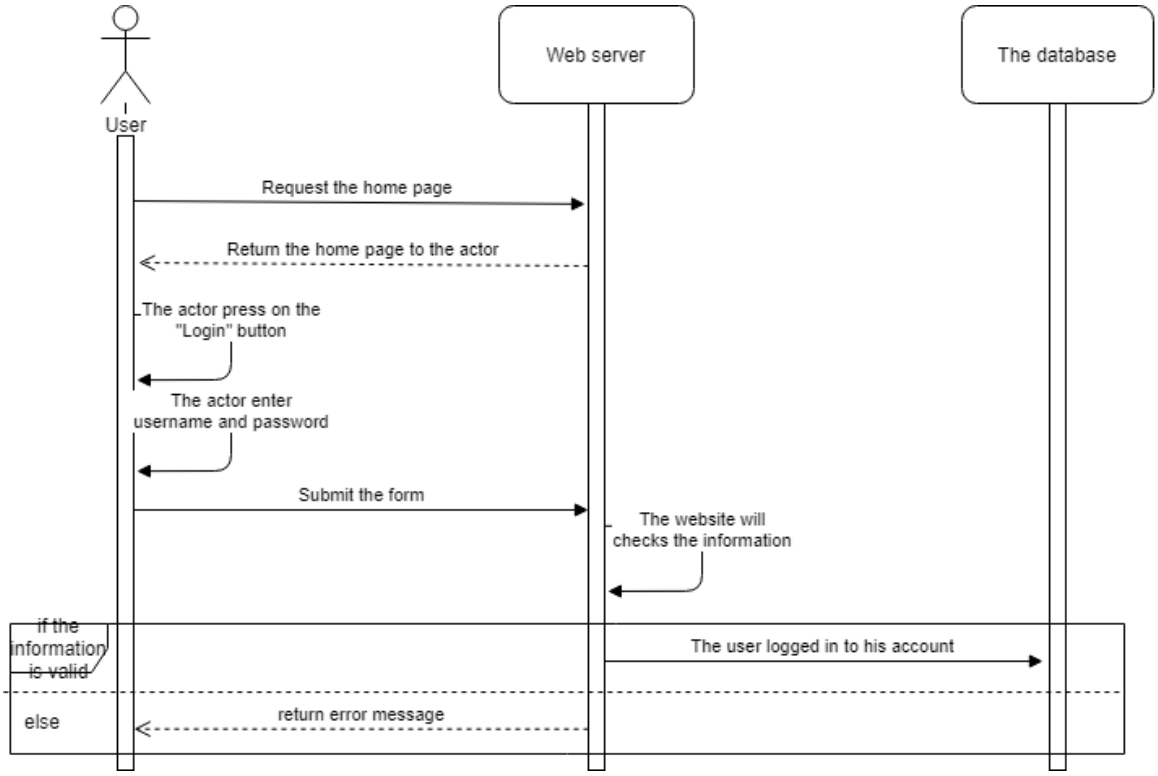
Use Case ID:	5		
Use Case Name:	Add comments		
Actors:	Admin, User	**Trigger:**	Clicks the ‘add post’ button.
Precondition:	No comment has been added		
Post Condition:	Admin and user comment has been added.		
Description:	Enable user and admin to add comments		
Normal Flow:	1- Click on add post. 2- Write a comment. 3- After writing , clicks on add post.		

5.2 Interaction Diagrams:

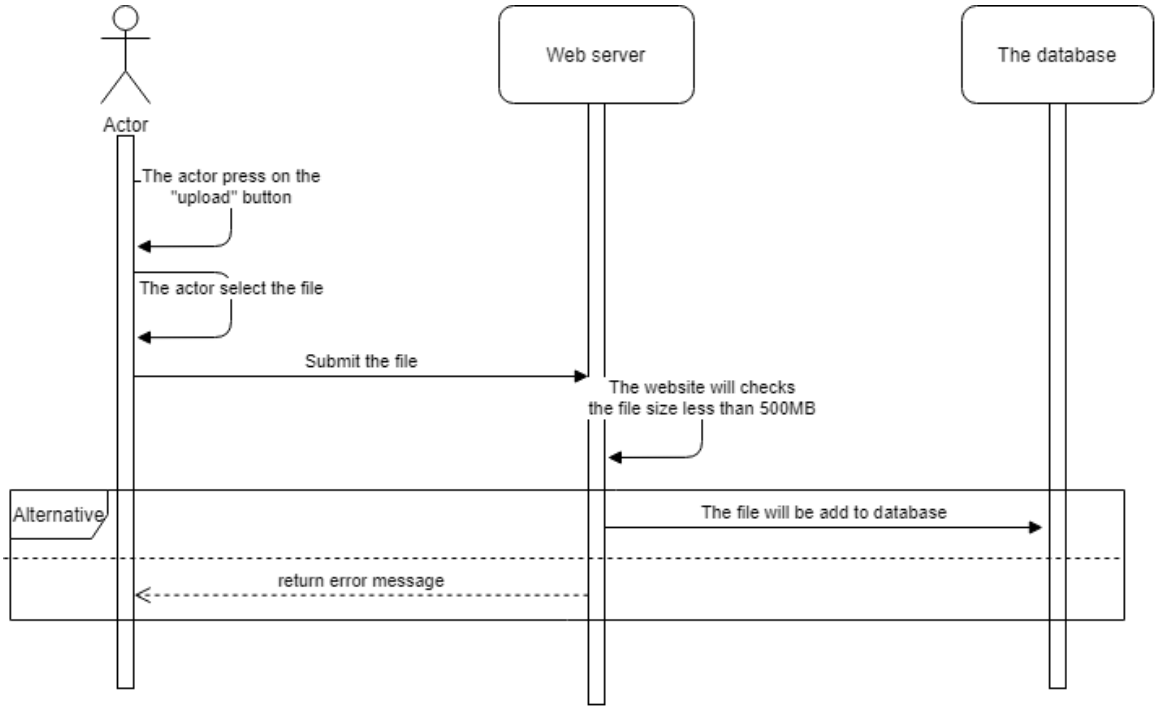
Create an account



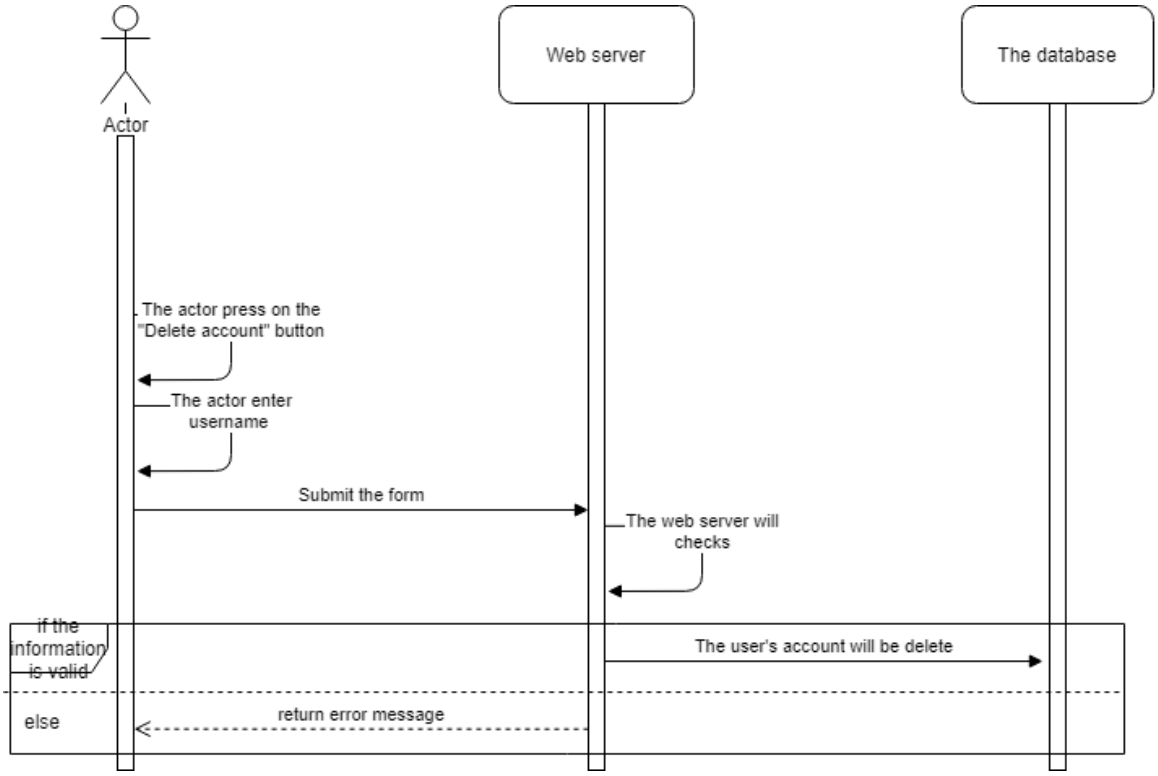
Login


User
Web server
The database
Request the home page
Return the home page to the actor
The actor press on the "Login" button
The actor enter username and password
Submit the form
The website will checks the information
if the information is valid
The user logged in to his account
else
return error message

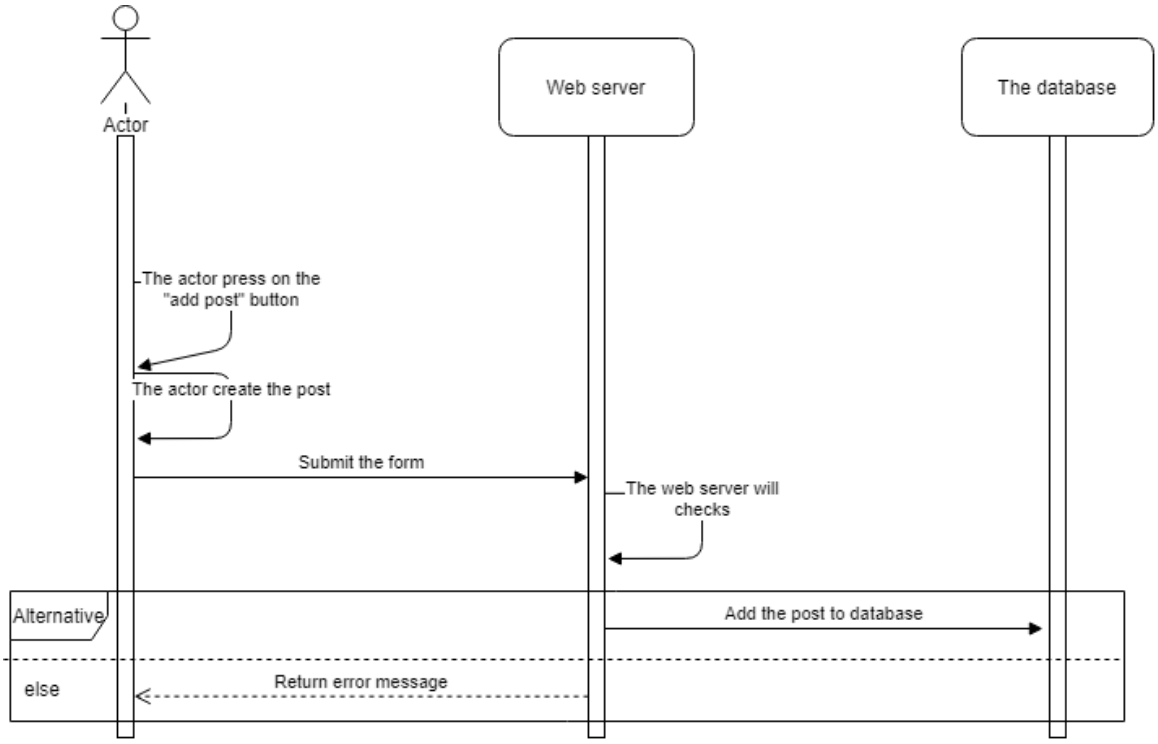
Upload a file



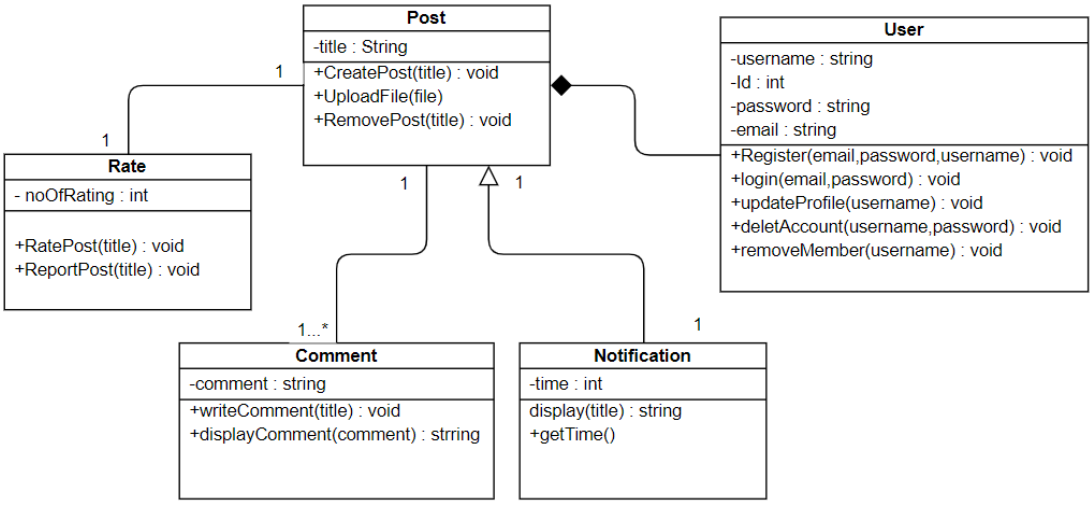
Delete user


Actor
Web server
The database
The actor press on the "Delete account" button
The actor enter username
Submit the form
The web server will checks
if the information is valid
The user's account will be delete
else
return error message

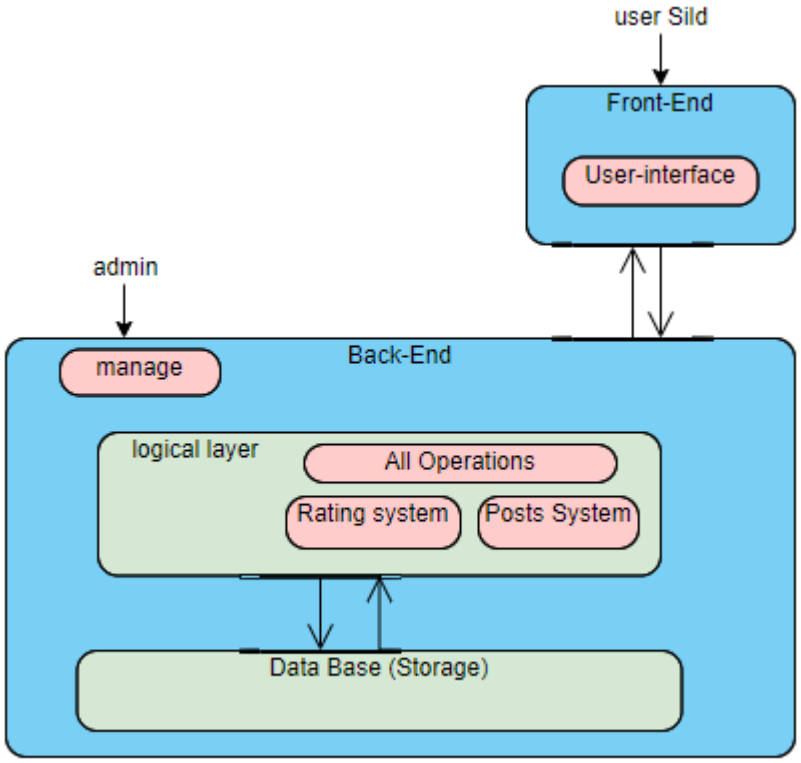
Add post


Actor
Web server
The database
The actor press on the "add post" button
The actor create the post
Submit the form
The web server will checks
Alternative
Add the post to database
else
Return error message

5.3 Class Diagram:



System Architecture:



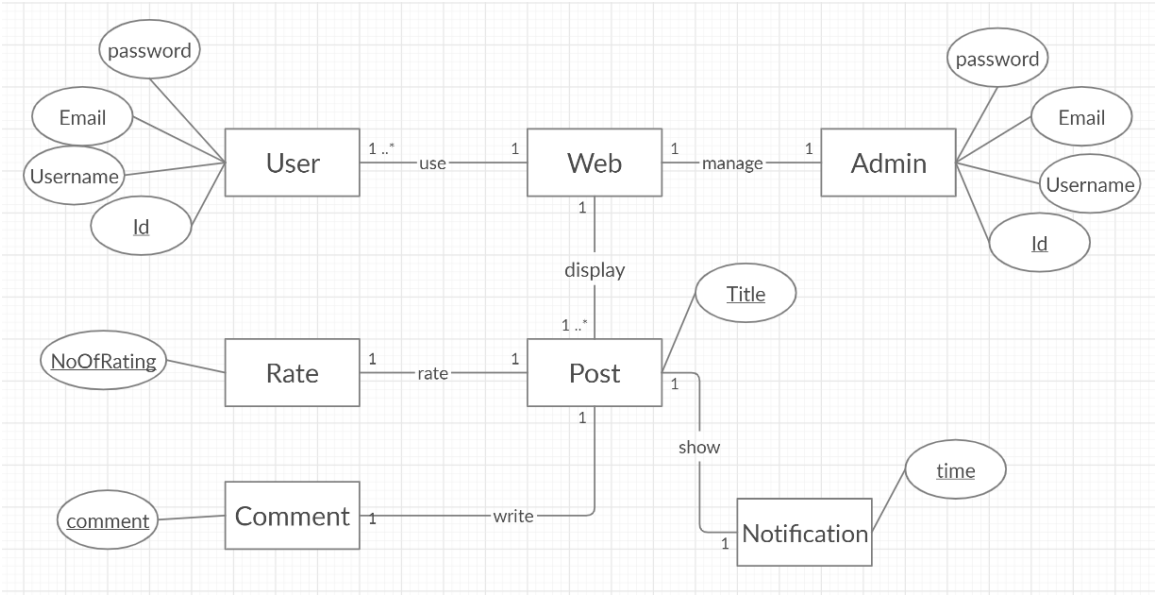
User side:
Users will use browser to use the system, The front-end should work as a user interface and collect data from the users.
and handle it to the back end.

Mange:
It's for admin to control panel of the system, it provide some feature like remove member, delete a reported posts.

Logical layer:
It's the mean part of the system, It has all operation like add post, Rate post, Write comment ...etc.
It receive request from the users and admin then the operation will be performed.
It handles the communications with the database.

Database:
It store all data of the user's info and the content of the post and number of rating and comments.
It has communications with the logical layer.

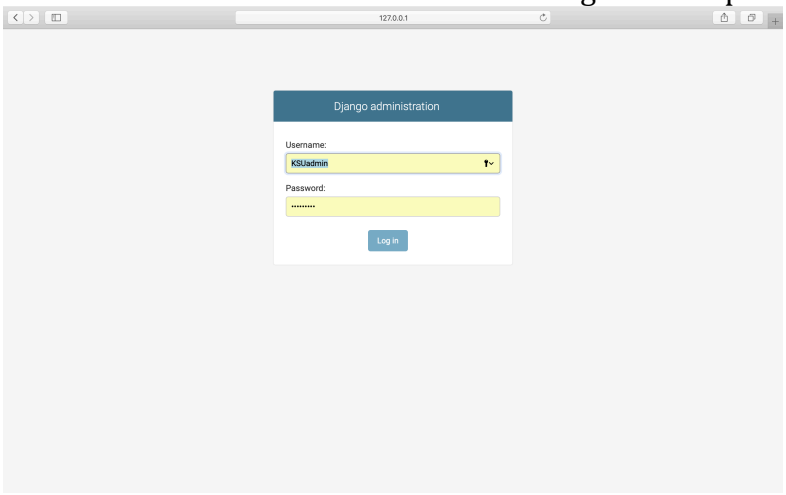
5.4 Database Design:



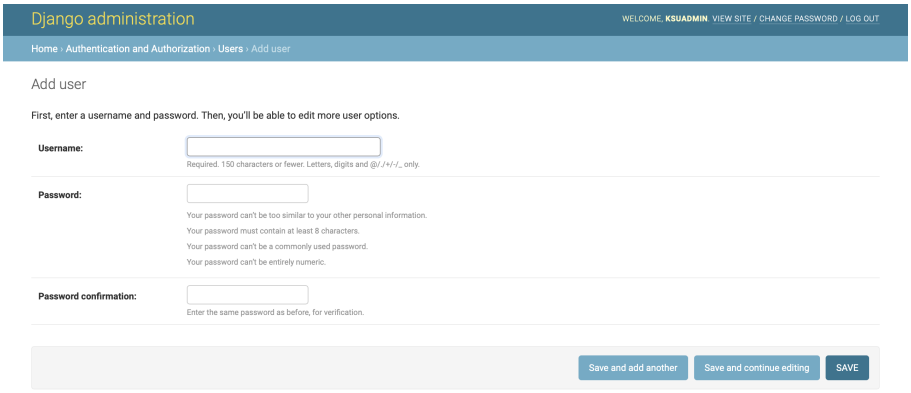
5.5 User Interface Prototype:

In the section we will provide graphical description of the user interface.

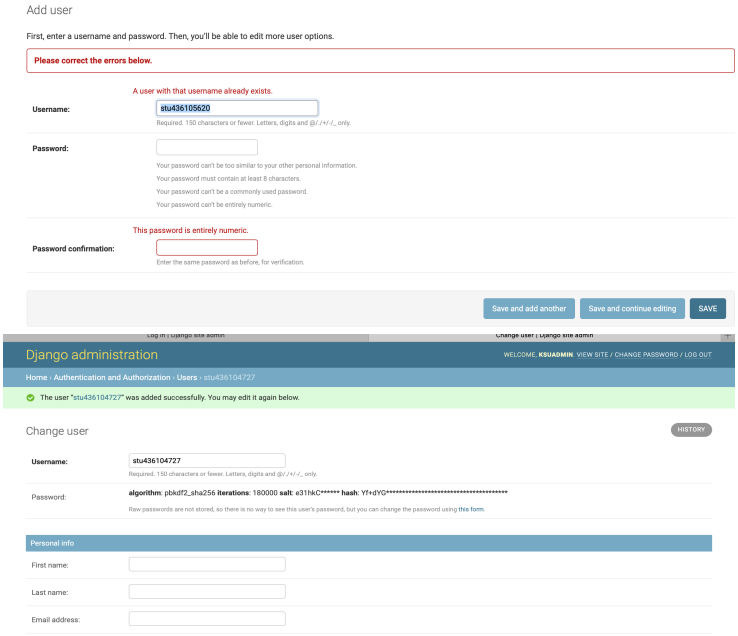
1. This is the main super user login page provided by Django, it allows the admin to add student and staff and give the capability to access the website.



2. After login, the admin can choose Add user, then he writes the user name and password.

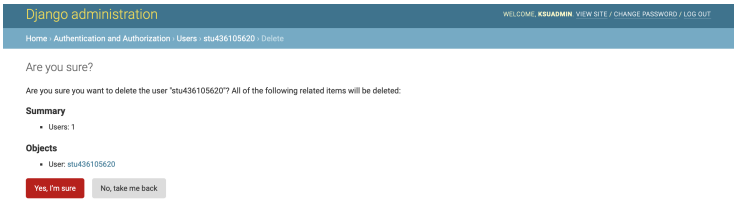


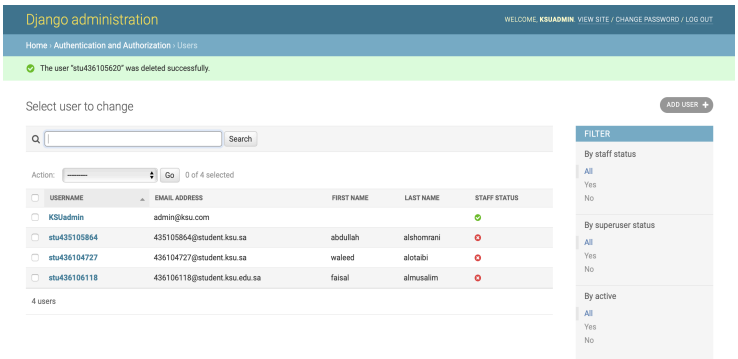
3. Then the system checks whether the user account already exist or no, if not the user is added, the admin can then add furthermore information and even give him an authorization to become an admin.



deleting:

the admin choose a specific user to delete after deleting a message will pop and inform the admin about successfully deleting the user.





5.6 Search Algorithm:

A Linear search algorithm will be implemented to help the student who are going to search for a specific course to find it fast. We will evaluate the performance of the algorithms and if we find it unsatisfactory, we will implement other algorithms such as a hashing algorithm or a binary search algorithm.

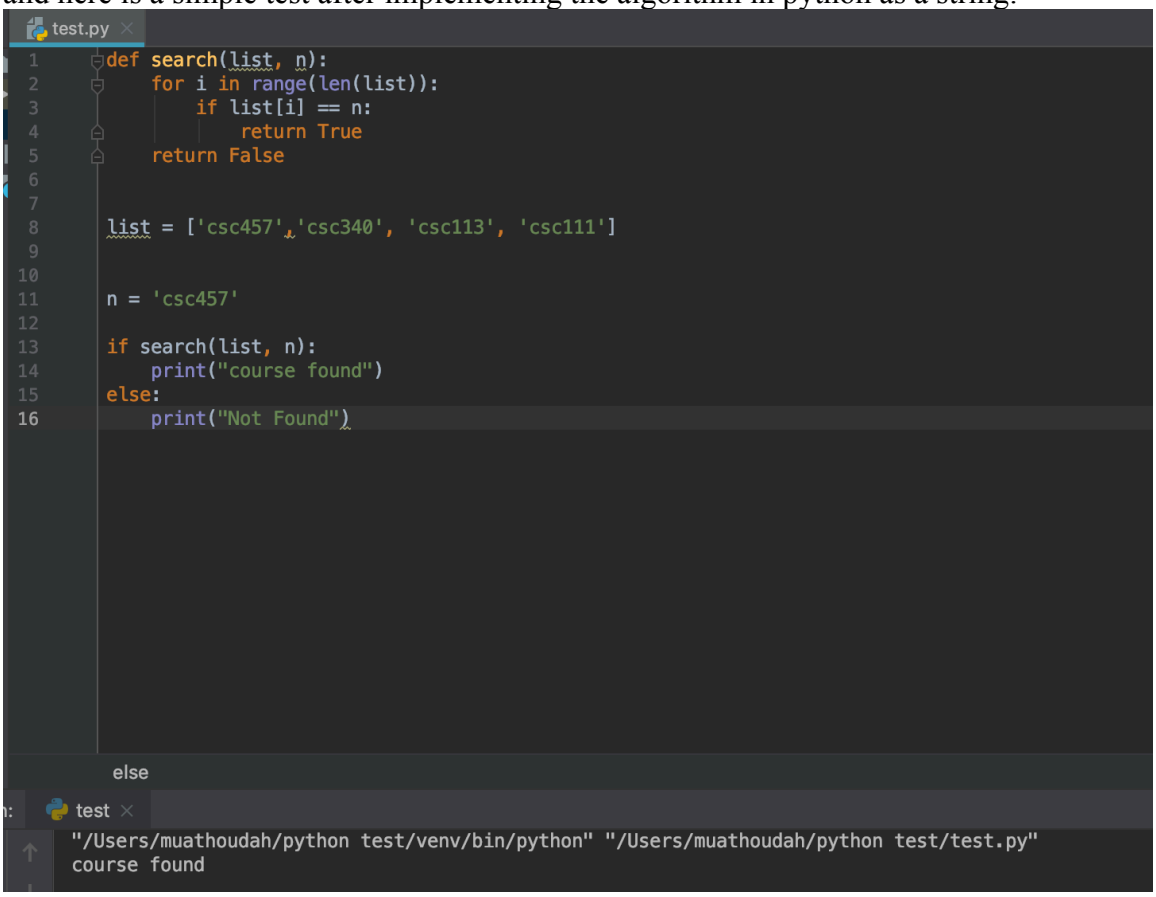
Pseudo code:

```
procedure linear_search (list, value)

   for each item in the list
      if match item == value
         return the item's location
      end if
   end for

end procedure
```

and here is a simple test after implementing the algorithm in python as a string:



Chapter 6: System Implementation**

This chapter will be written during the next term. This chapter should provide an exhaustive explanation of the implementation stage of your project, review and explain all used technologies, describe the adopted integration process, and mention any limitations in the system (if any).

You should also give a clear detailed description of the software's main interfaces and core logic. You can also include a walkthrough of the system by showing the sample interfaces to demonstrate other functionalities. Provide code snippet to show the main logic but do not give the whole code.

Chapter 7: System Testing **

This chapter will be written during the next term. This chapter should describe the test strategies and methodologies used to plan, organize, execute and manage the testing of your software project. Mention and explain any tools used for testing the software.

7.1 Unit testing

Show the results of testing each component separately.

7.2 Integration and regression testing

Describe how components where tested during the integration process, and report any issues or unexpected behavior and how it has been resolved.

7.3 Performance and stress testing

Measure and report the performance of your software, and explain the behavior of your system under extreme cases.

7.4 User acceptance testing

The purpose of acceptance testing is to confirm that the system is ready for operational use. During acceptance test, end-users (customers) of the system compare the system to its initial requirements. You should describe this process and may list people involved in testing, and the feedback you obtained from them.

7.5 Test cases

The goal of any given test case or set of test cases is to detect defects in the system being tested. This section should provide descriptions of various test cases to test each component in your application with all the possible actions and input. Each test case should include a brief description of the sequence of events being tested, the test data, testing environment, expected results, actual results, and whether the software passed or failed that test case.

Chapter 8: Conclusion

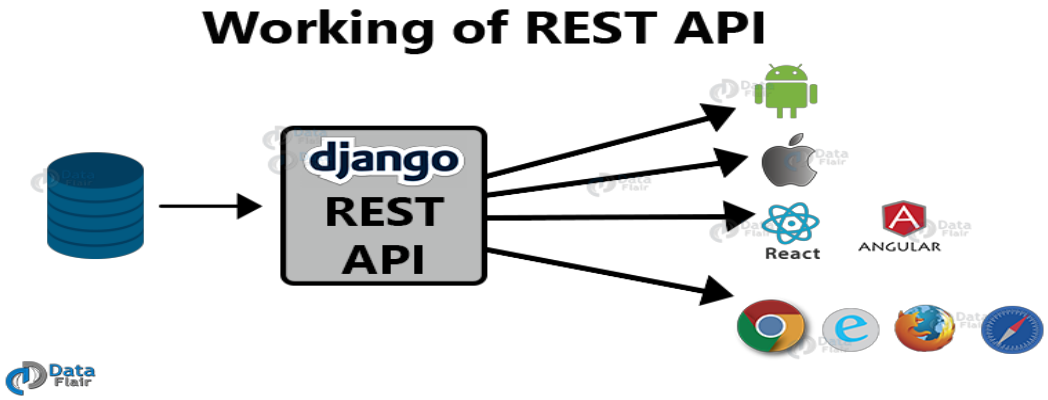
Our goal is to create a web-based system and app for sharing course material and resources between students to make their academic lives easier and improve their understanding of courses in the shortest possible time.

In our system, we will avoid the weakness of other websites and learn from it and make it advantages to our side, for example:

- Koizait is a public website open and reachable to all, our website would restrict access to KSU registered students.
- Studocu main disadvantage that you should become a premium member in order to reach the content, however our "ERS" is going to be free.
- Koofers is good website but the content is mixed which makes it hard for students to find their course notes.

Luckily python allows us to avoid these disadvantages by using Django, as well other libraries.

We intend to make the project accessible to students on phones and tablets. This can be achieved by using Django REST API with any language that support IOS and Andriod development.



References

1- Robert Gibb (2016) Stack path How a web application works .

https://blog.stackpath.com/web-application/

2- begin with java Introduction to Web Application

http://www.beginwithjava.com/servlet-jsp/web-application-overview/web-application.html

3- GURU99 operating system.

https://www.guru99.com/operating-system-tutorial.html

4- GCF Global type of os

https://edu.gcfglobal.org/en/computerbasics/understanding-operating-systems/1/

5- Martin Armstrong (2020) Statista

https://www.statista.com/chart/8552/windows-7-is-still-the-desktop-os-king/

6-worldStream - search engins

https://www.wordstream.com/articles/internet-search-engines-history

7- Oberlo - statistics of search engines

https://www.oberlo.com/statistics/search-engine-market-share

8- sitesbay - bootstrap

https://www.sitesbay.com/bootstrap/bootstrap-features-of-bootstrap

9- Andrzej Dabski-(2019) sunscrapers

https://sunscrapers.com/blog/10-django-packages-you-should-know/