A

Mini Project On

**E-LIBRARY WITH URLS TUTORIAL , PDF BOOKS & VIDEOS**

(Submitted in partial fulfillment of the requirements for the award of Degree)

BACHELOR OF TECHNOLOGY

In

COMPUTER SCIENCE AND ENGINEERING

By

ANUGNA LINGAMPALLI (207R1A0590)

Under the Guidance of

**RAKSHITHA OKALI**

(Assistant Professor)

##### DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

**CMR TECHNICAL CAMPUS UGC AUTONOMOUS**

(Accredited by NAAC, NBA, Permanently Affiliated to JNTUH, Approved by AICTE, New Delhi) Recognized Under Section 2(f) & 12(B) of the UGCAct.1956, Kandlakoya (V), Medchal Road, Hyderabad-501401.

**2020-2024**

**DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING**



#### CERTIFICATE

This is to certify that the project entitled **“LIBRARY WITH URLS TUTORIAL , PDF BOOKS & VIDEOS”** being submitted by **ANUGNA LINGAMPALLI (207R1A0590),** in partial fulfillment of the requirements for the award of the degree of B.Tech in Computer Science and Engineering to the Jawaharlal Nehru Technological University Hyderabad, is a record of bonafide work carried out by them under our guidance and supervision during the year 2023-24.

The results embodied in this thesis have not been submitted to any other University or Institute for the award of any degree or diploma.

**Rakshitha Okali Dr. A. Raji Reddy**

(Assistant Professor) DIRECTOR

INTERNAL GUIDE

**Dr. K. Srujan Raju EXTERNAL EXAMINER**

HOD

**Submitted for viva voice Examination held on**

##### ACKNOWLEDGEMENT

Apart from my efforts , the success of any project depends largely on the encouragement and guidelines of many others. I take this opportunity to express my gratitude to the people who have been instrumental in the successful completion of this project.

I take this opportunity to express my profound gratitude and deep regard to my guide **Rakshitha Okali,** Assistant Professor for her exemplary guidance, monitoring and constant encouragement throughout the project work. The blessing, help and guidance given by her shall carry me a long way in the journey of life on which I am about to embark.

I also take this opportunity to express a deep sense of gratitude to the Project Review Committee (PRC) **G.Vinesh Shanker, Dr. J. Narasimharao, Ms. Shilpa, & Dr. K. Maheswari** for their cordial support, valuable information and guidance, which helped me in completing this task through various stages.

I am also thankful to **Dr. K. Srujan Raju,** Head, Department of Computer Science and Engineering for providing encouragement and support for completing this project successfully.

I am obliged to **Dr. A. Raji Reddy,** Director for being cooperative throughout the course of this project. I also express our sincere gratitude to Sri. **Ch. Gopal Reddy,** Chairman for providing excellent infrastructure and a nice atmosphere throughout the course of this project.

The guidance and support received from all the members of **CMR Technical Campus** who contributed to the completion of the project. I am grateful for their constant support and help.

Finally, I would like to take this opportunity to thank my family for their constant encouragement, without which this assignment would not be completed. I sincerely acknowledge and thank all those who gave support directly and indirectly in the completion of this project.

###### ANUGNA LINGAMPALLI (207R1A0590)

##### ABSTRACT

During the past recent years, there has been tremendous development reaming the concept of digital libraries, the biggest online platform of knowledge that can be stored and retrieved through online networks. Digital libraries are considered as the most complex form of data systems that associate with the digital document preservation, distributed database management, hypertext, filtering, information retrieval, and selective dissemination of information. This has really overcome geographical barrier offering a wide range of academic, research, and cultural resources with multimedia effects which can be accessed around the world over the distributed networks.

The study also highlighted the information on the digital library projects undertaken in countries. This project provides information to the audience on the subject matter in terms of what has been already discovered and explored on the importance of Digital Library and what all can be further explored. The literature pertaining to the studies relating to how digital libraries emerged discussed in this article. The idea is to brief the readers about the concept of library resources shifted into digital libraries with the help of technology and its growth sourced from already existing literature. The contemporary trends reflecting the current state of the library and how it has progressed over time also discussed here.

|  |  |  |
| --- | --- | --- |
| **FIGURE NO** | **FIGURE NAME** | **PAGE NO** |
| Figure 3.1 | Project Architecture for  E-Library with URL Tutorial , Pdf Books & Videos | 7 |
| Figure 3.2 | Use Case Diagram for  E-Library with URL Tutorial , Pdf Books & Videos | 8 |
| Figure 3.3 | Class Diagram for E-Library with URL Tutorial , Pdf Books & Videos | 9 |
| Figure 3.4 | Sequence diagram for  E-Library with URL Tutorial , Pdf Books & Videos | 10 |
| Figure 3.5  Figure 3.6  Figure 3.7 | Activity diagram for E-Library  with URL Tutorial , Pdf  Books & Videos  Collaboration Diagram for  E-Library with URL Tutorial ,  Pdf Books & Videos  Deployment Diagram for  E-Library with URL Tutorial ,  Pdf Books & Videos | 11  12  13 |

|  |  |  |
| --- | --- | --- |
| **SCREENSHOT NO.** | **SCREENSHOT NAME** | **PAGE NO**. |
| Screenshot 5.1 | Running Program | 24 |
| Screenshot 5.2 | Website View | 24 |
| Screenshot 5.3 | Admin Login | 25 |
| Screenshot 5.4 | Successful Admin Login | 25 |
| Screenshot 5.5  Screenshot 5.6  Screenshot 5.7  Screenshot 5.8  Screenshot 5.9  Screenshot 5.10  Screenshot 5.11  Screenshot 5.12  Screenshot 5.13 | Adding Books  Successful Adding of Books  Adding Videos  Successful Adding of Videos  Adding URL Tutorial  Successful Adding of URL Tutorial  Details of Available Books Videos URL Tutorial & Pdfs    One of the Pdfs in available list  New User Sign Up | 26  26  27  27  28  28  29  29  30 |
| SCREENSHOT NO. | SCREENSHOT NAME | PAGE NO. |
| Screenshot 5.14 | Successfully Signed Up | 30 |
| Screenshot 5.15 | User Login | 31 |
| Screenshot 5.16 | Successful Login and Start Search for material | 31 |
| Screenshot 5.17 | Search for URL Tutorial in available material | 32 |
| Screenshot 5.18  Screenshot 5.19  Screenshot 5.20  Screenshot 5.21  Screenshot 5.22 | Available URL Tutorials in the material  Open and read tutorial from URL  Search for Videos in available material  Available Videos in the material  Playing the Video | 32  33    33  34  34 |

###### ABSTRACT i

LIST OF FIGURES ii

LIST OF SCREENSHOTS iii

1. [INTRODUCTION](#_TOC_250030) 1

1.1 [PROJECT SCOPE 1](#_TOC_250029)

1.2 [PROJECT PURPOSE 1](#_TOC_250028)

1.3 [PROJECT FEATURES 1](#_TOC_250027)

1. [SYSTEM ANALYSIS](#_TOC_250026) 2

2.1 [PROBLEM DEFINITION 2](#_TOC_250025)

2.2 [EXISTING SYSTEM 2](#_TOC_250024)

2.2.1 LIMITATIONS OF THE EXISTING SYSTEM 3

2.3 [PROPOSED SYSTEM 3](#_TOC_250023)

2.3.1 ADVANTAGES OF PROPOSED SYSTEM 3

2.4 [FEASIBILITY STUDY 4](#_TOC_250022)

2.4.1 [ECONOMIC FEASIBILITY 4](#_TOC_250021)

2.4.2 [TECHNICAL FEASIBILITY 5](#_TOC_250020)

2.4.3 SOCIAL FEASIBILITY 5

2.5 [HARDWARE & SOFTWARE REQUIREMENTS 5](#_TOC_250019)

2.5.1 [HARDWARE REQUIREMENTS 5](#_TOC_250018)

2.5.2 [SOFTWARE REQUIREMENTS 6](#_TOC_250017)

1. [ARCHITECTURE](#_TOC_250016) 7

3.1 [PROJECT ARCHITECTURE 7](#_TOC_250015)

3.2 [DESCRIPTION 7](#_TOC_250014)

3.3 [USE CASE DIAGRAM 8](#_TOC_250013)

3.4 [CLASS DIAGRAM 9](#_TOC_250012)

3.5 [SEQUENCE DIAGRAM 10](#_TOC_250011)

3.6 [ACTIVITY DIAGRAM 11](#_TOC_250010)

3.7 COLLABORATION DIAGRAM 12

3.8 DEPLOYMENT DIAGRAM 13

1. IMPLEMENTATION 14

4.1 SAMPLE CODE 14

1. SCREENSHOTS 24
2. [TESTING 35](#_TOC_250009) 6.1 [INTRODUCTION TO TESTING 3](#_TOC_250008)5

6.2 [TYPES OF TESTING 3](#_TOC_250007)5

6.2.1 UNIT TESTING 35

6.2.2 [INTEGRATION TESTING 3](#_TOC_250006)6

6.2.3 [FUNCTIONAL TESTING 36](#_TOC_250005)

6.2.4 SYSTEM TESTING 37

6.2.5 WHITE BOX TESTING 37

6.2.6 BLACK BOX TESTING 37

6.3 [TEST CASES 38](#_TOC_250004)

6.3.1 [CLASSIFICATION 38](#_TOC_250003)

1. CONCLUSION & FUTURE SCOPE 39

7.1 PROJECT CONCLUSION 39

7.2 [FUTURE SCOPE 39](#_TOC_250002)

##### BIBLIOGRAPHY 40

8.1 [REFERENCES](#_TOC_250001) 40

8.2 [GITHUB LINK](#_TOC_250000) 40

# INTRODUCTION

#### INTRODUCTION

##### PROJECT SCOPE

This project is titled “E-Library with URL Tutorial , Pdf Books & Videos”. Develop an E-Library platform that provides easy access to a wide range of educational resources, including URL links, tutorial PDF books, and videos. The platform aims to facilitate learning and research by organizing and presenting these resources in an accessible and user-friendly manner.

##### PROJECT PURPOSE

The purpose of creating an e-library with URL tutorials, PDF books, and videos is to provide a comprehensive and accessible digital resource for learning and education. It is to democratize access to education and knowledge, support learning in diverse formats, and empower individuals to acquire new skills, enhance their knowledge, and pursue their educational goals.

##### PROJECT FEATURES

Creating an e-library with URL access to tutorial PDF books and videos requires careful planning and consideration of various features to ensure a user-friendly and efficient platform.

some of the essential features are :

* User Registration
* Search
* Access Control
* Content Updates
* Security and Privacy
* Admin Panel

## SYSTEM ANALYSIS

##### SYSTEM ANALYSIS

**SYSTEM ANALYSIS**

System Analysis is the important phase in the system development process. The System is studied to the minute details and analyzed. The system analyst plays an important role of an interrogator and dwells deep into the working of the present system. In analysis, a detailed study of these operations performed by the system and their relationships within and outside the system is done. A key question considered here is, “what must be done to solve the problem?” The system is viewed as a whole and the inputs to the system are identified. Once analysis is completed the analyst has a firm understanding of what is to be done.

##### PROBLEM DEFINITION

##### The problem at hand is to design and develop an e-library platform that offers a comprehensive collection of tutorial materials in various formats, including PDF books and videos, accessible via URLs. The primary goal of this e-library is to provide a user-friendly and efficient way for learners to access educational resources and enhance their knowledge and skills.

##### EXISTING SYSTEM

Concept of digital libraries has existed from the 20th Century. It is yet to follow many years of the 21st Century in building these digital libraries as strong and complete collections of records that have been visualized regarding them from a long time in the past (Okerson ,2009). The affluence within the knowledge has transformed the access systems for every stake holder in retrieval of main learning and significant information. The evolutions of digital libraries have happened significantly in the past few years. They are not only the digital corresponding entity of traditional (physical) libraries but they have become involved in networking systems. They have now the potential of managing communication as well as collaboration between diverse, universally spread user groups and communities (Tramboo et al. 2012).

###### DISADVANTAGES OF EXISTING SYSTEM

Following are the disadvantages of existing system:

* Physical Boundary
* Available for Specific Time period
* Worries about Misplaced or Stolen
* Lack of Books
* High Cost to buy books
* Penalty for late return

##### PROPOSED SYSTEM

The designing of innovative models and architectures are transforming this combination of information along with user requirements for giving improved well-versed decisions and enriching the customer experience. The vast volume of information within the communicating networks fetches vital challenges for assuring security basics within a disseminated setting. Blockchain, originally launched as a Bitcoin Cryptocurrency by Satoshi Nakamoto, has advanced beyond that. It facilitates a reliable platform for exchanging services and transactions through a disseminated network.

###### ADVANTAGES OF THE PROPOSED SYSTEM

* No Physical Boundary
* Round the clock availability
* No Worries about Misplaced or Stolen
* Updation of New Content
* No Cost or Less cost
* Multiple Access

##### FEASIBILITY STUDY

The feasibility of the project is analyzed in this phase and business proposal is put forth with a very general plan for the project and some cost estimates. During system analysis the feasibility study of the proposed system is to be carried out. This is to ensure that the proposed system is not a burden to the company. For feasibility analysis, some understanding of the major requirements for the system is essential.

* Economic Feasibility
* Technical Feasibility
* Social Feasibility

###### ECONOMIC FEASIBILITY

This study is carried out to check the economic impact that the system will have on the organization. The amount of fund that the company can pour into the research and development of the system is limited. The expenditures must be justified. Thus the developed system as well within the budget and this was achieved because most of the technologies used are freely available. Only the customized products had to be purchased.

###### TECHNICAL FEASIBILITY

This study is carried out to check the technical feasibility, that is, the technical requirements of the system. Any system developed must not have a high demand on the available technical resources. This will lead to high demands on the available technical resources. This will lead to high demands being placed on the client. The developed system must have a modest requirement, as only minimal or null changes are required for implementing this system.

**SOCIAL FEASIBILITY**

The aspect of study is to check the level of acceptance of the system by the user. This includes the process of training the user to use the system efficiently. The user must not feel threatened by the system, instead must accept it as a necessity. The level of acceptance by the users solely depends on the methods that are employed to educate the user about the system and to make him familiar with it. His level of confidence must be raised so that he is also able to make some constructive criticism, which is welcomed, as he is the final user of the system.

##### HARDWARE & SOFTWARE REQUIREMENTS

###### HARDWARE REQUIREMENTS:

Hardware interfaces specify the logical characteristics of each interface between the software product and the hardware components of the system. The following are some hardware requirements.

* Processor: Pentium IV or higher
* RAM: 256 MB
* Space on Hard Disk: minimum 512

##### SOFTWARE REQUIREMENTS:

Software Requirements specifies the logical characteristics of each interface and software components of the system. The following are some software requirements,

* Operating system : Windows 7 Ultimate.
* Coding Language **:** Python.
* Front-End **:** Python.
* Designing **:** Html,css,javascript.
* Data Base **:** MySQL.

## ARCHITECTURE

##### ARCHITECTURE

##### PROJECT ARCHITECTURE

This project architecture shows the procedure followed for classification, starting from input to final prediction.



Figure 3.1: Project Architecture of E-Library with URL Tutorial , Pdf Books & Videos

###### DESCRIPTION

All existing library system where managing using manual entries or computer entries where admin or librarian has to record all details such as Book Name, student name, issue-date and applying penalty and all completing all this task is time consuming process and to overcome from this issue we are designing E-Library where ADMIN will upload various types of tutorials like Videos, tutorial URL and all types of files like PDF, WORD, PPT etc.

###### 

###### USE CASE DIAGRAM

A use case diagram in the Unified Modeling Language (UML) is a type of behavioral diagram defined by and created from a Use-case analysis. Its purpose is to present a graphical overview of the functionality provided by a system in terms of actors, their goals (represented as use cases), and any dependencies between those use cases. The main purpose of a use case diagram is to show what system functions are performed for which actor. Roles of the actors in the system can be depicted.



Figure 3.2: Use Case Diagram for E-Library with URL Tutorial , Pdf Books & Videos

##### 

##### CLASS DIAGRAM

In software engineering, a class diagram in the Unified Modeling Language (UML) is a type of static structure diagram that describes the structure of a system by showing the system's classes, their attributes, operations (or methods), and the relationships among the classes. It explains which class contains information.



Figure 3.3: Class Diagram for E-Library with URL Tutorial , Pdf Books & Videos

**SEQUENCE DIAGRAM**

A sequence diagram in Unified Modeling Language (UML) is a kind of interaction diagram that shows how processes operate with one another and in what order. It is a construct of a Message Sequence Chart. Sequence diagrams are sometimes called event diagrams, event scenarios, and timing diagrams.



Figure 3.4: Sequence Diagram for E-Library with URL Tutorial , Pdf Books & Videos

###### ACTIVITY DIAGRAM

Activity diagrams are graphical representations of workflows of stepwise activities and actions with support for choice, iteration and concurrency. In the Unified Modeling Language, activity diagrams can be used to describe the business and operational step-by-step workflows of components in a system. An activity diagram shows the overall flow of control.



Figure 3.5: Activity Diagram for E-Library with URL Tutorial , Pdf Book

**COLLABORATION DIAGRAM**

In collaboration diagram the method call sequence is indicated by some numbering technique as shown below. The number indicates how the methods are called one after another. We have taken the same order management system to describe the collaboration diagram. The method calls are similar to that of a sequence diagram. But the difference is that the sequence diagram does not describe the object organization where as the collaboration diagram shows the object organization.

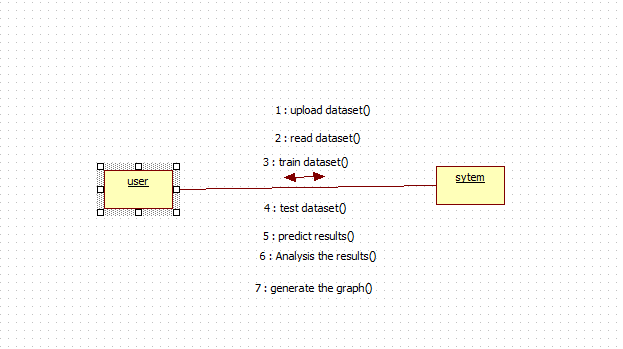


Figure 3.6: Collaboration Diagram for E-Library with URL Tutorial , Pdf Book

**DEPLOYMENT DIAGRAM**

Deployment diagram represents the deployment view of a system. It is related to the component diagram. Because the components are deployed using the deployment diagrams. A deployment diagram consists of nodes. Nodes are nothing but physical hardware’s used to deploy the application.



Figure 3.7: Deployment Diagram for E-Library with URL Tutorial , Pdf Book

## IMPLEMENTATION

##### 4.1 SAMPLE CODE

from django.shortcuts import render

from django.template import RequestContext

from django.contrib import messages

from django.http import HttpResponse

from django.conf import settings

import os

import pymysql

from django.core.files.storage import FileSystemStorage

from datetime import date

global username, password, contact, email, address

def SearchBookAction(request):

if request.method == 'POST':

query = request.POST.get('t1', False)

file\_type = request.POST.get('t2', False)

query = query.lower()

array = query.split(" ")

output = '<table border=1 align=center width=100%>'

font = '<font size="" color="black">'

arr = ['ID','Name','Description','Book Date','Book Type','Filename','Access Data']

output += "<tr>"

dup = []

for i in range(len(arr)):

output += "<th>"+font+arr[i]+"</th>"

con = pymysql.connect(host='127.0.0.1',port = 3306,user = 'root', password = '',database = 'elibrary',charset='utf8')

with con:

cur = con.cursor()

cur.execute("select \* FROM addbook")

rows = cur.fetchall()

for row in rows:

book\_id = row[0]

book\_name = row[1]

description = row[2]

book\_date = row[3]

book\_type = row[4]

filename = row[5]

books = book\_name.lower()

descs = description

for k in range(len(array)):

if array[k] in books or array[k] in descs:

if filename not in dup and book\_type == file\_type:

dup.append(filename)

output += "<tr><td>"+font+str(book\_id)+"</td>"

output += "<td>"+font+book\_name+"</td>"

output += "<td>"+font+description+"</td>"

output += "<td>"+font+book\_date+"</td>"

output += "<td>"+font+book\_type+"</td>"

output += "<td>"+font+filename+"</td>"

if book\_type == "Video":

output+='<td><a href="PlayVideo?t1='+filename+'"><img src=/static/images/video.png height=100 width=100/></a></td>'

elif book\_type == "URL":

output+='<td><a href="'+book\_name+'" target="\_blank"><img src=/static/images/url.png height=100 width=100/></a></td>'

else:

output+='<td><a href="http://127.0.0.1:8000/static/books/'+filename+'"><img src=/static/images/book.jpg height=100 width=100/></a></td>'

context= {'data':output}

return render(request, 'SearchResult.html', context)

def SearchBook(request):

if request.method == 'GET':

return render(request, 'SearchBook.html', {})

def DeleteFile(request):

if request.method == 'GET':

filename = request.GET['t1']

db\_connection = pymysql.connect(host='127.0.0.1',port = 3306,user = 'root', password = '', database = 'elibrary',charset='utf8')

db\_cursor = db\_connection.cursor()

student\_sql\_query = "delete from addbook where file\_name = '"+filename+"'"

db\_cursor.execute(student\_sql\_query)

db\_connection.commit()

os.remove("LibraryApp/static/books/"+filename)

output = filename+' deleted from database'

context= {'data':output}

return render(request, 'AdminScreen.html', context)

def PlayVideo(request):

if request.method == 'GET':

video = request.GET['t1']

output = '<source src="static/books/'+video+'" type="video/mp4">Your browser does not support the video tag.'

context= {'data':output}

return render(request, 'PlayVideo.html', context)

def ViewBooks(request):

if request.method == 'GET':

output = '<table border=1 align=center width=100%>'

font = '<font size="" color="black">'

arr = ['ID','Name','Description','Book Date','Book Type','Filename','Access Data','Delete File']

output += "<tr>"

for i in range(len(arr)):

output += "<th>"+font+arr[i]+"</th>"

con = pymysql.connect(host='127.0.0.1',port = 3306,user = 'root', password = '', database = 'elibrary',charset='utf8')

with con:

cur = con.cursor()

cur.execute("select \* FROM addbook")

rows = cur.fetchall()

for row in rows:

book\_id = row[0]

book\_name = row[1]

description = row[2]

book\_date = row[3]

book\_type = row[4]

filename = row[5]

output += "<tr><td>"+font+str(book\_id)+"</td>"

output += "<td>"+font+book\_name+"</td>"

output += "<td>"+font+description+"</td>"

output += "<td>"+font+book\_date+"</td>"

output += "<td>"+font+book\_type+"</td>"

output += "<td>"+font+filename+"</td>"

if book\_type == "Video":

output+='<td><a href="PlayVideo?t1='+filename+'"><img src=/static/images/video.png height=100 width=100/></a></td>'

elif book\_type == "URL":

output+='<td><a href="'+book\_name+'" target="\_blank"><img src=/static/images/url.png height=100 width=100/></a></td>'

else:

output+='<td><a href="http://127.0.0.1:8000/static/books/'+filename+'"><img src=/static/images/book.jpg height=100 width=100/></a></td>'

output+='<td><a href="DeleteFile?t1='+filename+'">Click Here</a></td>'

context= {'data':output}

return render(request, 'ViewBooks.html', context)

def AddUrlAction(request):

if request.method == 'POST':

global username

url = request.POST.get('t1', False)

desc = request.POST.get('t2', False)

today = date.today()

output = "none"

count = 0

con = pymysql.connect(host='127.0.0.1',port = 3306,user = 'root', password = '', database = 'elibrary',charset='utf8')

with con:

cur = con.cursor()

cur.execute("select count(\*) from addbook")

rows = cur.fetchall()

for row in rows:

count = row[0]

count = count + 1

db\_connection = pymysql.connect(host='127.0.0.1',port = 3306,user = 'root', password = '', database = 'elibrary',charset='utf8')

db\_cursor = db\_connection.cursor()

student\_sql\_query = "INSERT INTO addbook(book\_id,book\_name,description,book\_date,book\_type,file\_name) VALUES('"+str(count)+"','"+url+"','"+desc+"','"+str(today)+"','URL','"+url+"')"

db\_cursor.execute(student\_sql\_query)

db\_connection.commit()

print(db\_cursor.rowcount, "Record Inserted")

if db\_cursor.rowcount == 1:

context= {'data':url+' Details saved in Database'}

return render(request, 'AddBook.html', context)

else:

context= {'data':'Error in adding book details'}

return render(request, 'AddBook.html', context)

def AddUrl(request):

if request.method == 'GET':

return render(request, 'AddUrl.html', {})

def AddBook(request):

if request.method == 'GET':

return render(request, 'AddBook.html', {})

def AddBookAction(request):

if request.method == 'POST':

global username, password, contact, email, address

name = request.POST.get('t1', False)

desc = request.POST.get('t2', False)

book\_type = request.POST.get('t4', False)

book\_name = request.FILES['t3'].name

book\_data = request.FILES['t3']

fs = FileSystemStorage()

today = date.today()

output = "none"

count = 0

con = pymysql.connect(host='127.0.0.1',port = 3306,user = 'root', password = '', database = 'elibrary',charset='utf8')

with con:

cur = con.cursor()

cur.execute("select count(\*) from addbook")

rows = cur.fetchall()

for row in rows:

count = row[0]

count = count + 1

db\_connection = pymysql.connect(host='127.0.0.1',port = 3306,user = 'root', password = '', database = 'elibrary',charset='utf8')

db\_cursor = db\_connection.cursor()

student\_sql\_query = "INSERT INTO addbook(book\_id,book\_name,description,book\_date,book\_type,file\_name) VALUES('"+str(count)+"','"+name+"','"+desc+"','"+str(today)+"','"+book\_type+"','"+book\_name+"')"

db\_cursor.execute(student\_sql\_query)

db\_connection.commit()

print(db\_cursor.rowcount, "Record Inserted")

if db\_cursor.rowcount == 1:

fs.save('LibraryApp/static/books/'+book\_name, book\_data)

context= {'data':book\_type+' Details saved in Database'}

return render(request, 'AddBook.html', context)

else:

context= {'data':'Error in adding book details'}

return render(request, 'AddBook.html', context)

def UserLogin(request):

global username

if request.method == 'POST':

username = request.POST.get('t1', False)

password = request.POST.get('t2', False)

con = pymysql.connect(host='127.0.0.1',port = 3306,user = 'root', password = '', database = 'elibrary',charset='utf8')

output = "none"

with con:

cur = con.cursor()

cur.execute("select username,password FROM register")

rows = cur.fetchall()

for row in rows:

if row[0] == username and row[1] == password:

username = row[0]

output = "success"

break

if output != 'none':

context= {'data':output}

return render(request, 'UserScreen.html', context)

if output == 'none':

context= {'data':'Invalid username'}

return render(request, 'Login.html', context)

def Register(request):

if request.method == 'GET':

return render(request, 'Register.html', {})

def index(request):

if request.method == 'GET':

return render(request, 'index.html', {})

def Login(request):

if request.method == 'GET':

return render(request, 'Login.html', {})

def AdminLogin(request):

if request.method == 'GET':

return render(request, 'AdminLogin.html', {})

def RegisterAction(request):

if request.method == 'POST':

global username, password, contact, email, address

username = request.POST.get('t1', False)

contact = request.POST.get('t3', False)

email = request.POST.get('t4', False)

address = request.POST.get('t5', False)

password = request.POST.get('t2', False)

fs = FileSystemStorage()

output = "none"

con = pymysql.connect(host='127.0.0.1',port = 3306,user = 'root', password = '', database = 'elibrary',charset='utf8')

with con:

cur = con.cursor()

cur.execute("select username,email FROM register")

rows = cur.fetchall()

for row in rows:

if row[0] == username:

output = username+" Username already exists"

break

if row[1] == email:

output = email+" Email id already exists"

break

if output == "none":

db\_connection = pymysql.connect(host='127.0.0.1',port = 3306,user = 'root', password = '', database = 'elibrary',charset='utf8')

db\_cursor = db\_connection.cursor()

student\_sql\_query = "INSERT INTO register(username,password,contact,email,address) VALUES('"+username+"','"+password+"','"+contact+"','"+email+"','"+address+"')"

db\_cursor.execute(student\_sql\_query)

db\_connection.commit()

print(db\_cursor.rowcount, "Record Inserted")

if db\_cursor.rowcount == 1:

context= {'data':'Signup Process Completed'}

return render(request, 'Register.html', context)

else:

context= {'data':'Error in signup process'}

return render(request, 'Register.html', context)

else:

context= {'data':output}

return render(request, 'Register.html', context)

def AdminLoginAction(request):

if request.method == 'POST':

user = request.POST.get('t1', False)

password = request.POST.get('t2', False)

if user == 'admin' and password == 'admin':

context= {'data':'Welcome '+user}

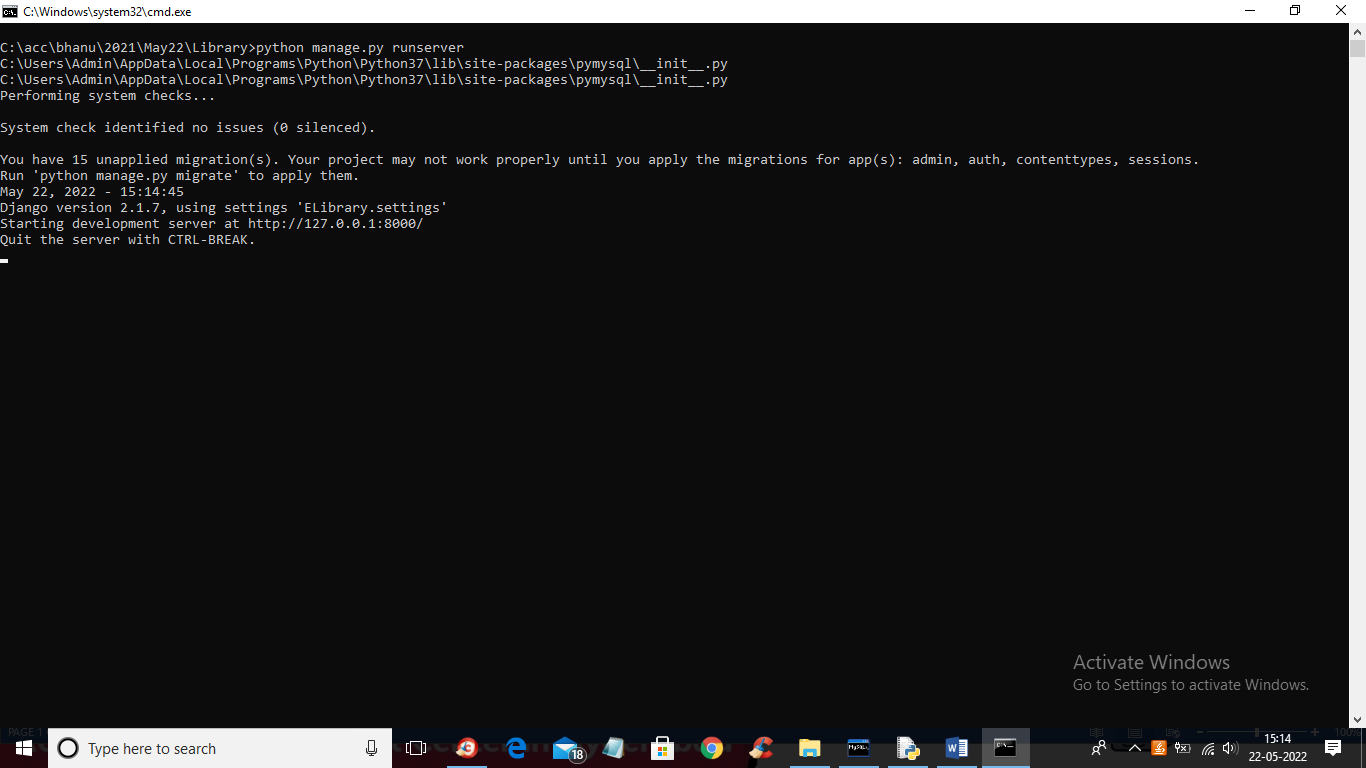
return render(request, 'AdminScreen.html', context)

else:

context= {'data':'Invalid login'}

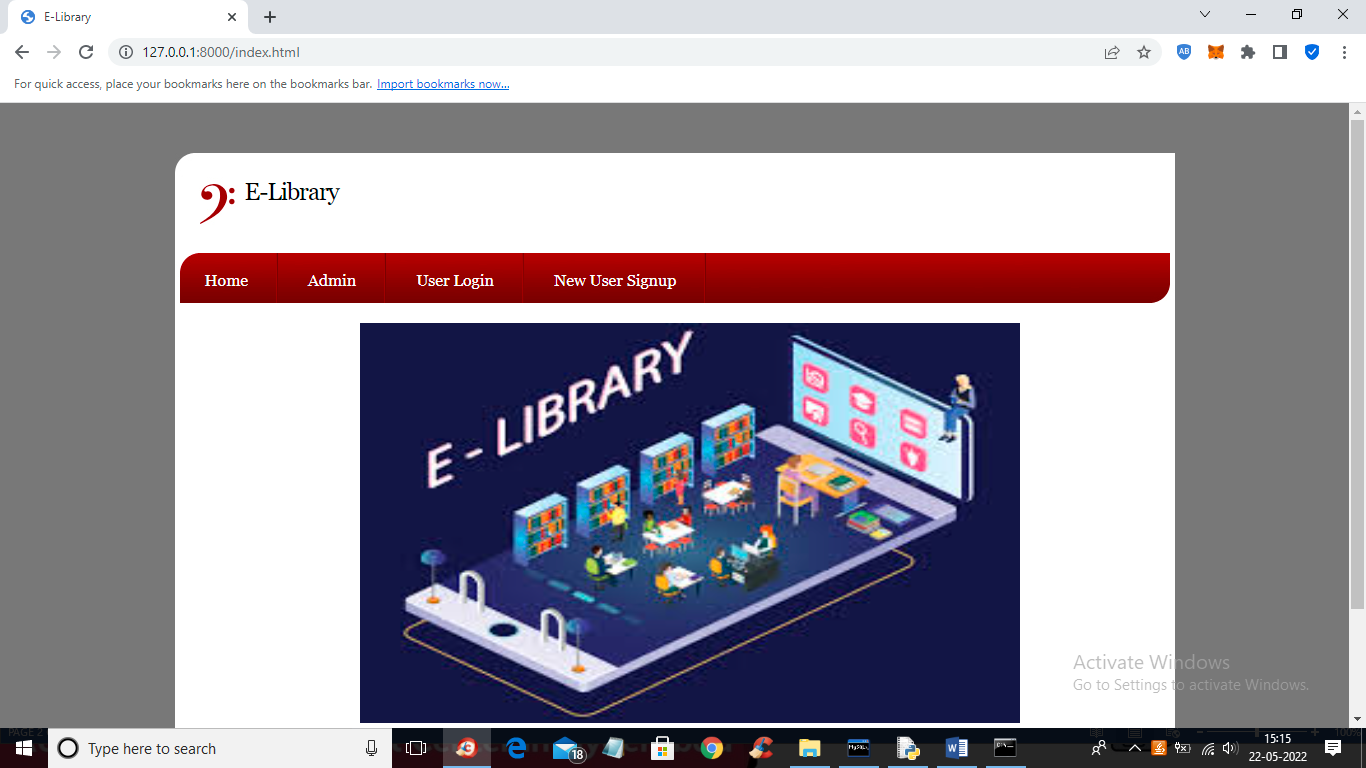
return render(request, 'AdminLogin.html', context)

## SCREENSHOTS

****

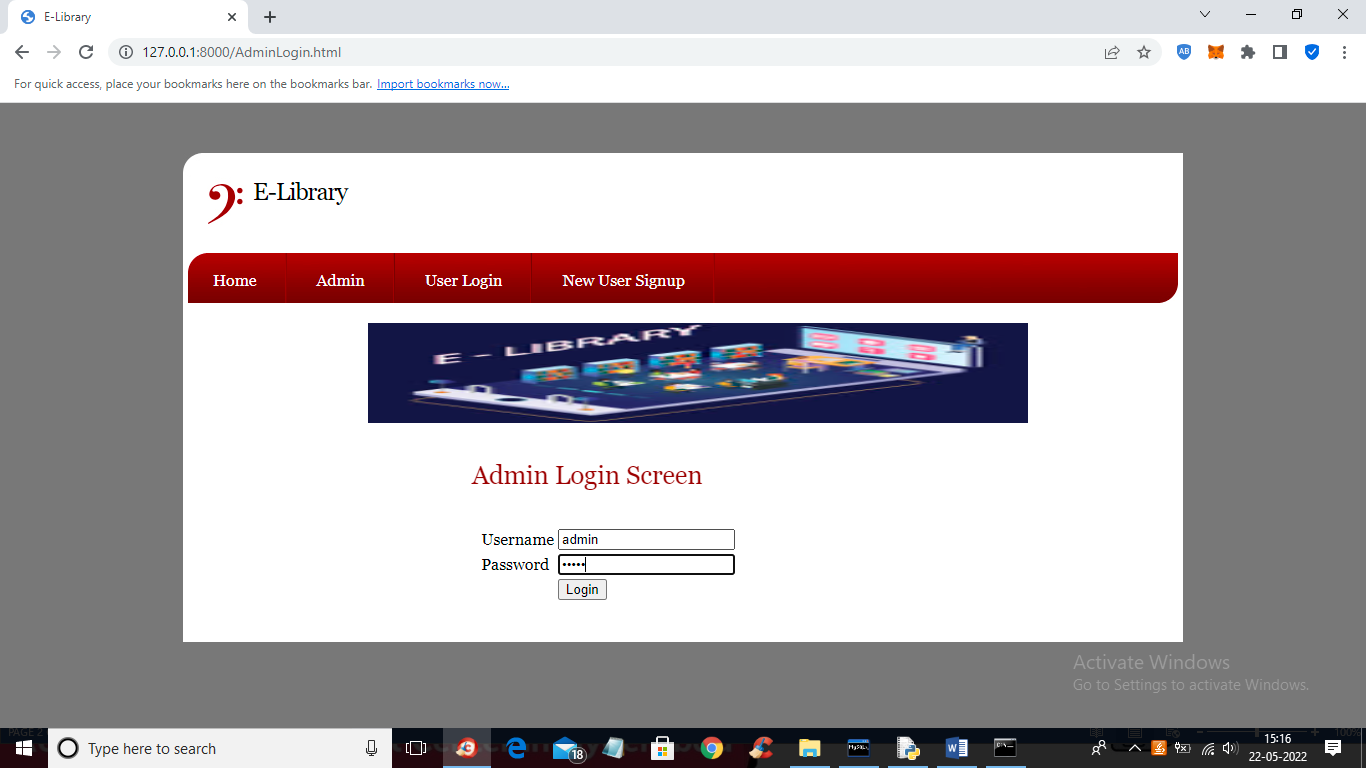
Screenshot 5.1 : Run program in Command Prompt

In above screen python DJANGO server has started and now open browser and enter URL as http;//127.0.0.1:8000/index.html and press enter key to get below screen



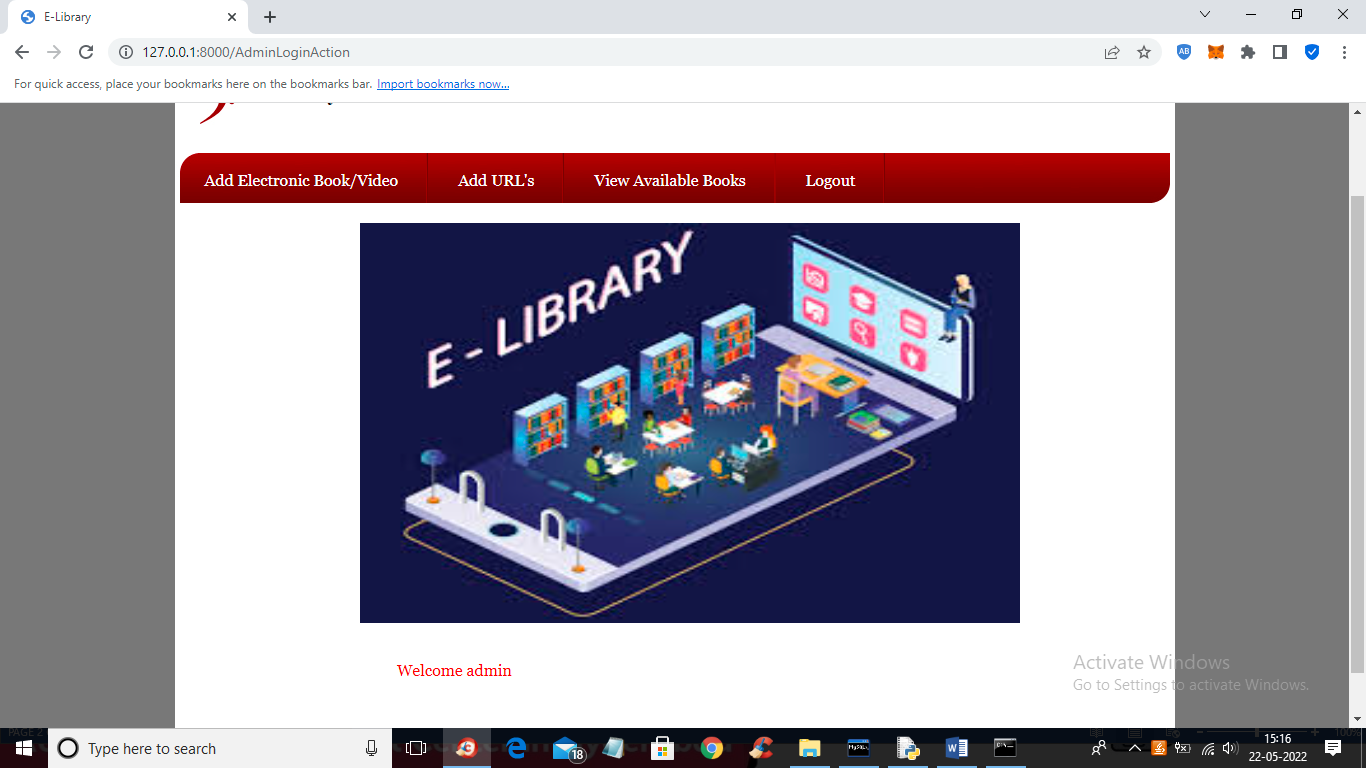
Screenshot 5.2 : Website View

Click on ‘Admin’ link to get the login screen



Screenshot 5.3 : Admin Login

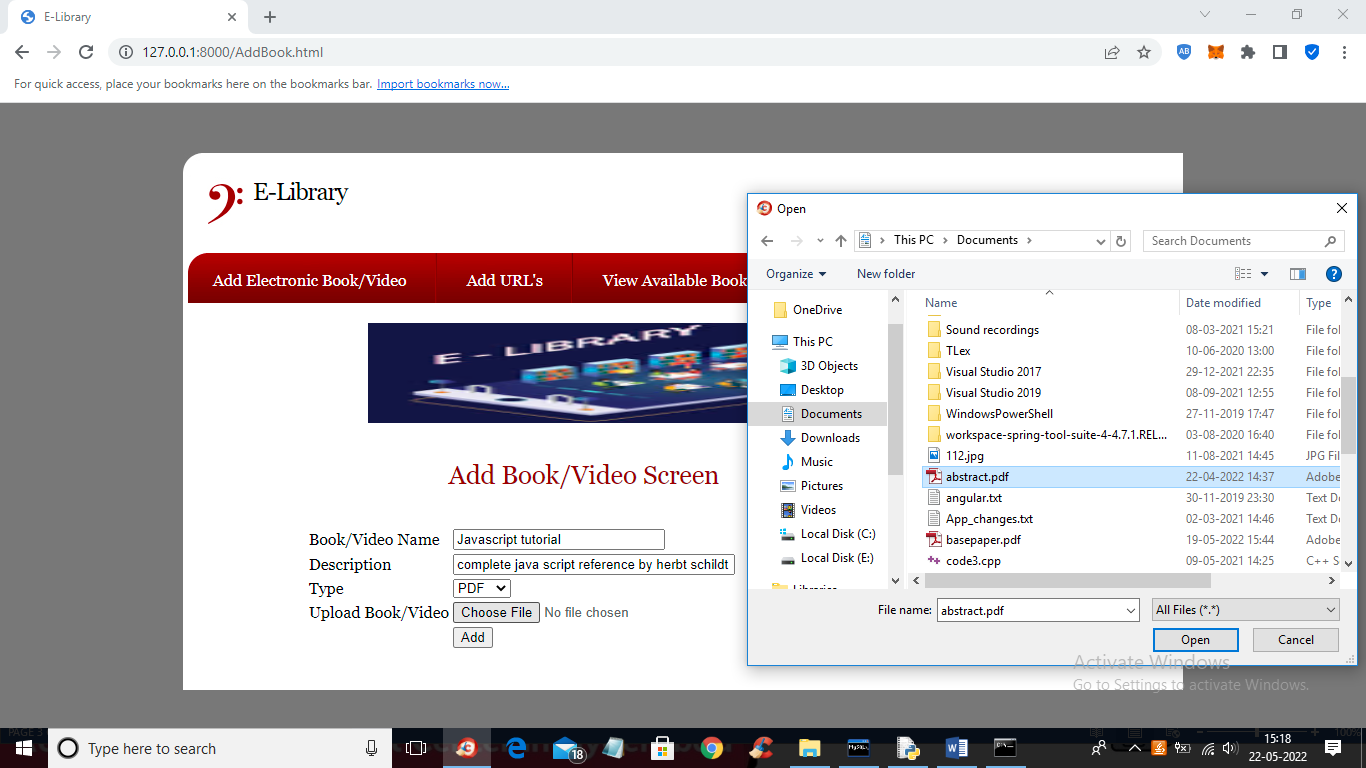
Enter the details of the Username and Password and click on “Login”



Screenshot 5.4 : Successful Admin Login

The login of Admin is Successful and now click on ‘Add Electronics Book/Video’

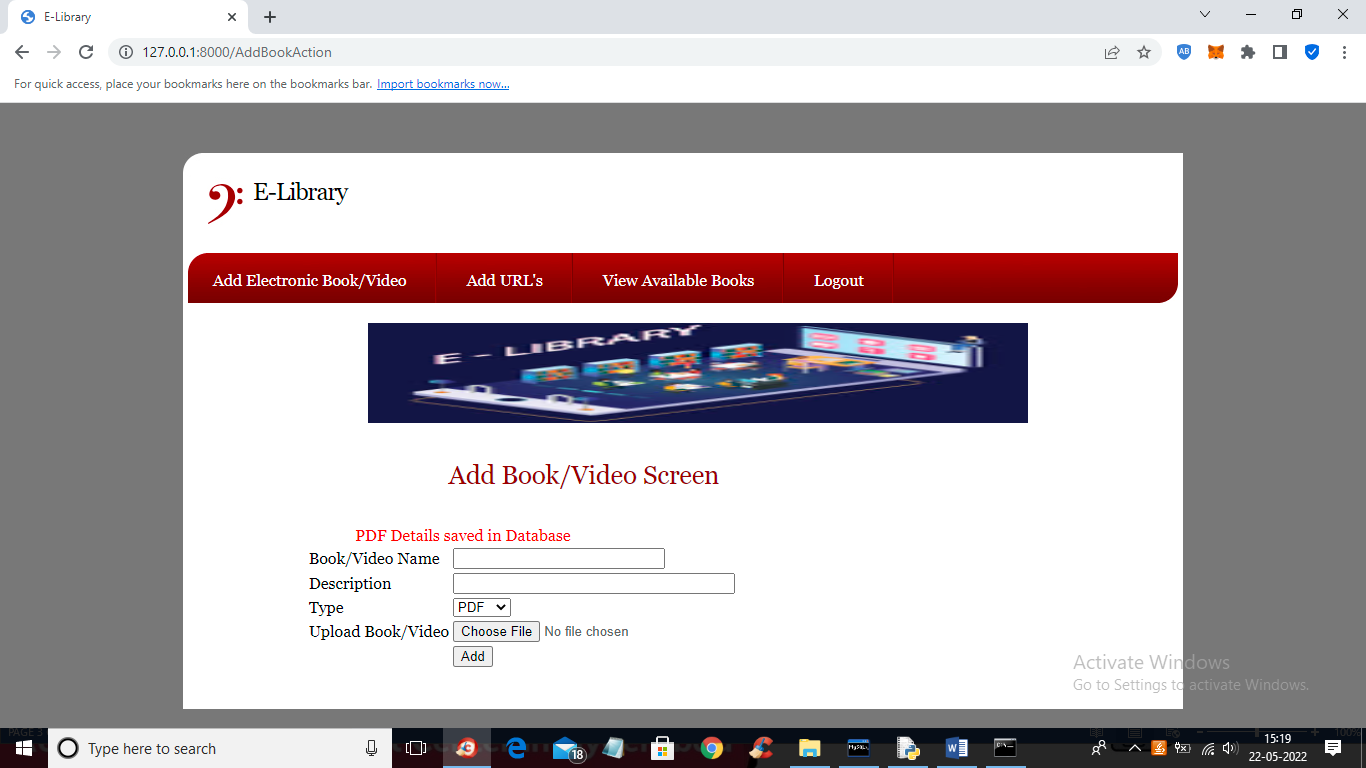
link to add book or video



Screenshot 5.5 : Adding Books

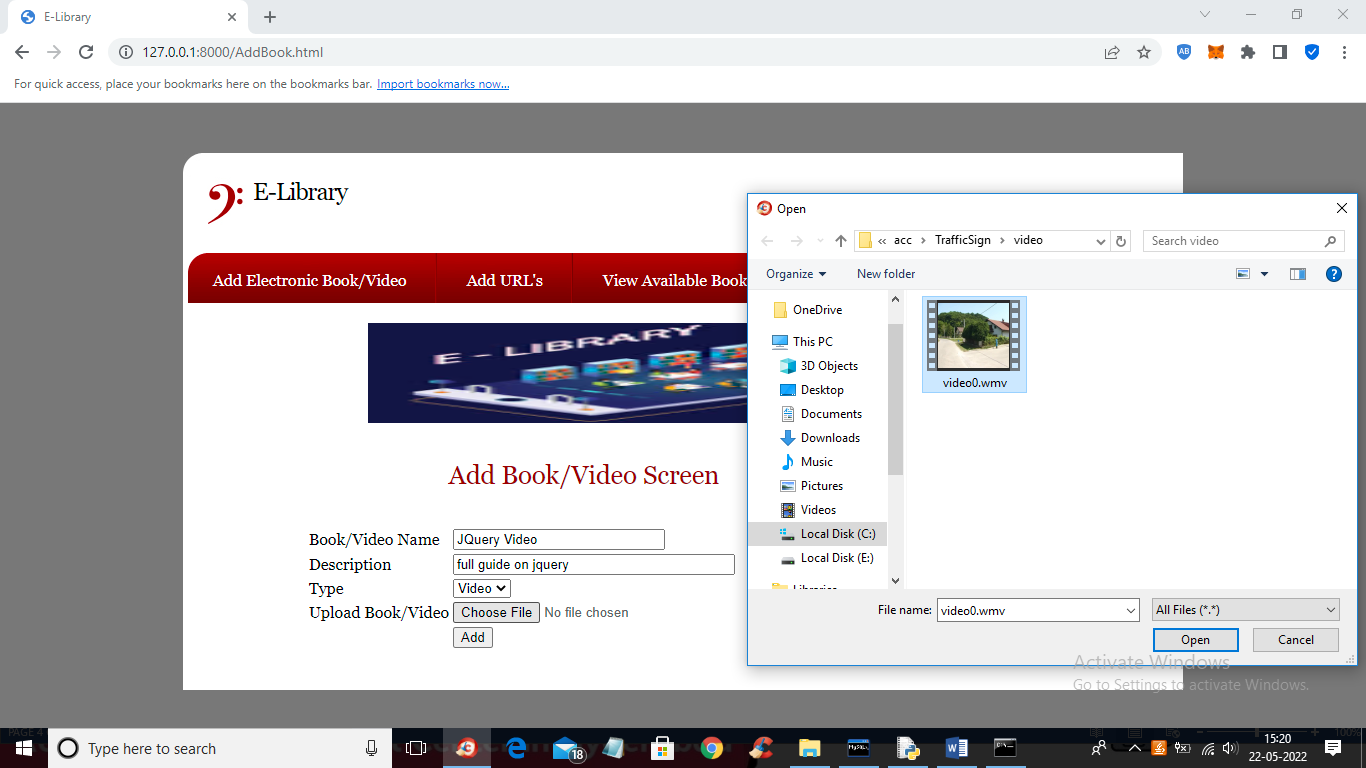
Admin is uploading PDF file and by clicking on ‘Open’ and ‘Add’ button to

add book to database and similarly admin can also upload video



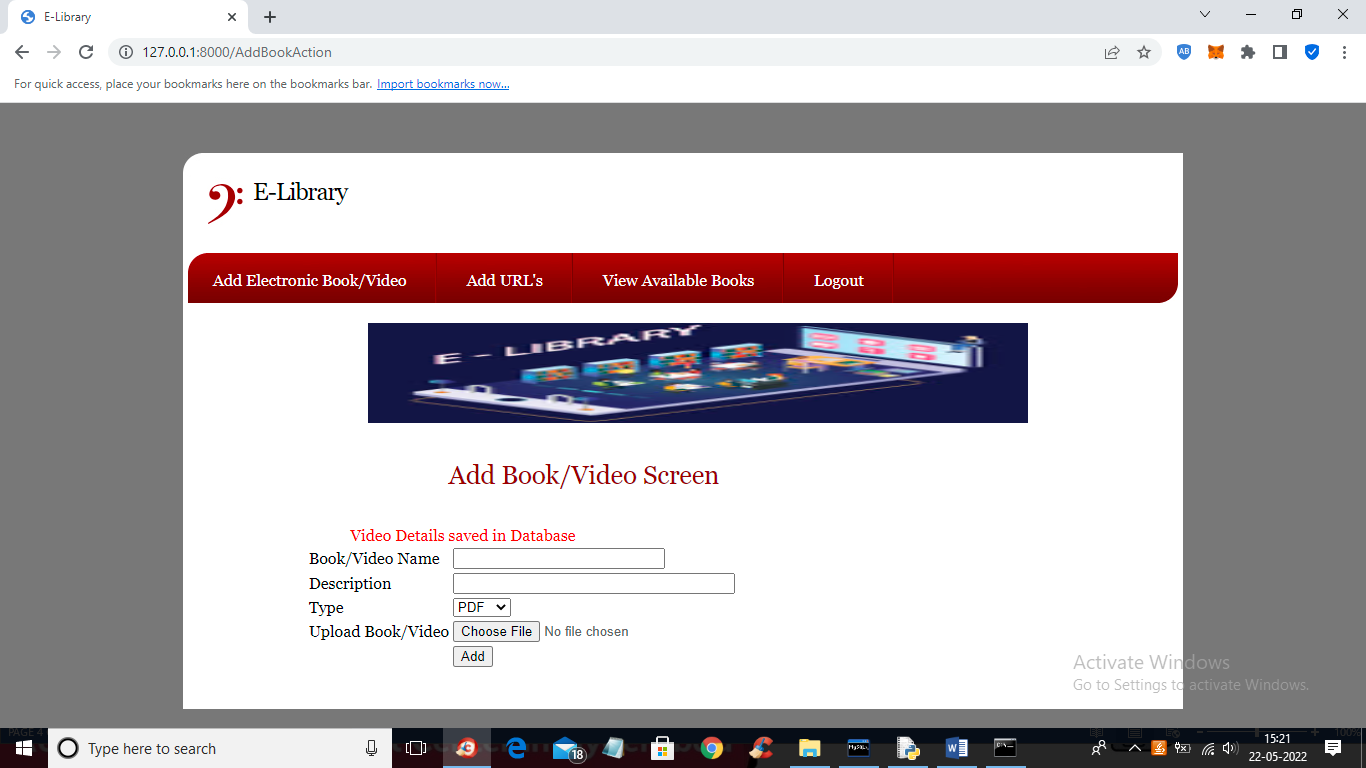
Screenshot 5.6 : Successful Adding of Books

PDF details added to database and similarly you can add other books also



Screenshot 5.7 : Adding Videos

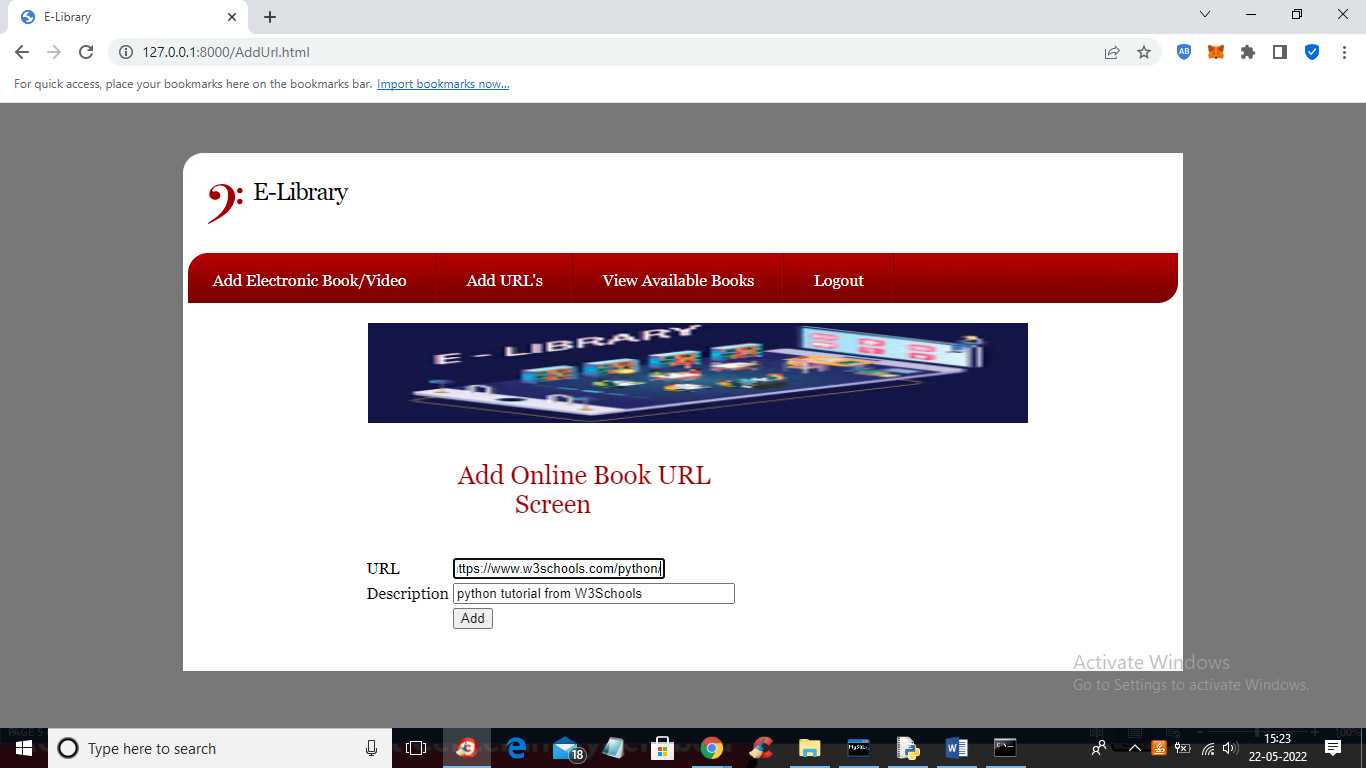
Admin is uploading video tutorial and click button “Add” to add details



Screenshot 5.8 : Successful Adding of Videos

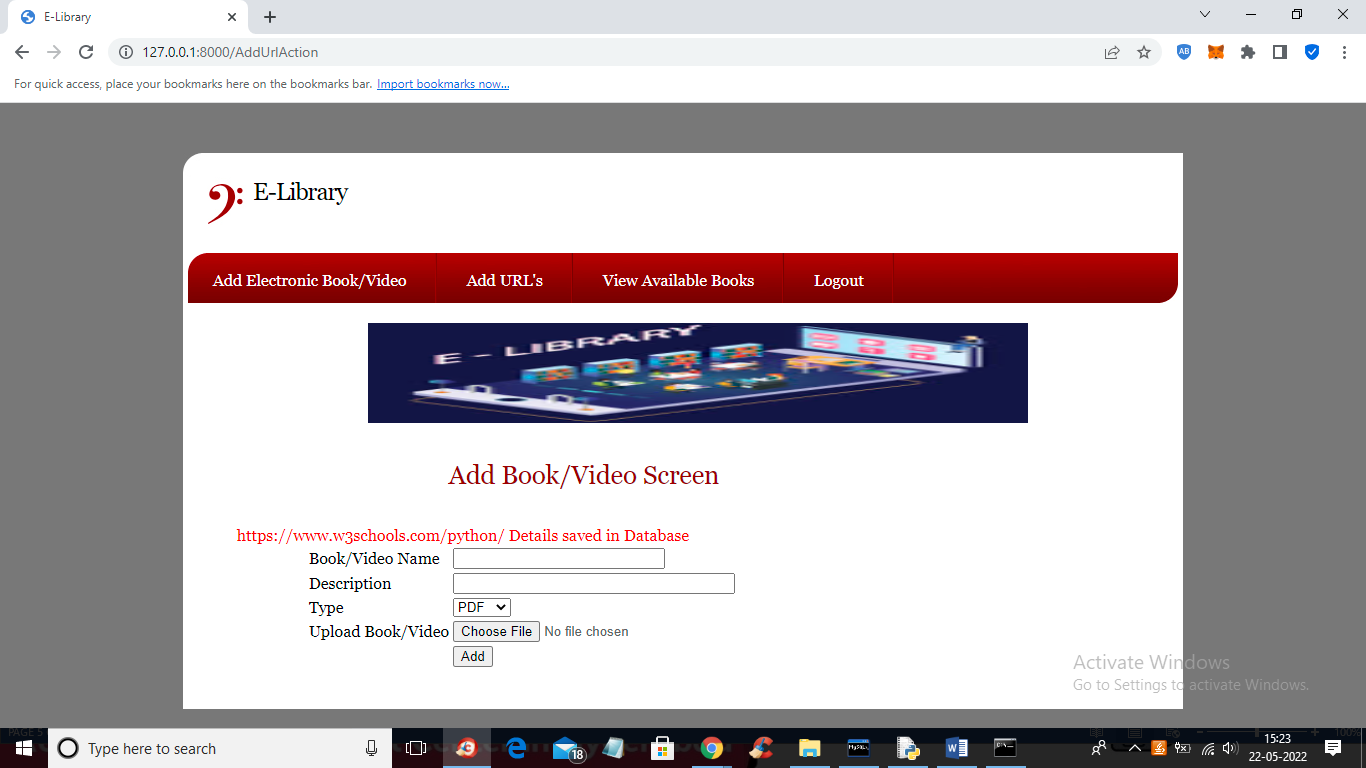
Video details saved in database and now click on ‘Add URL’s’ link to

add URL for tutorials



Screenshot 5.9 : Adding URL Tutorial

Admin is adding URL tutorial and click button “Ädd” and gets below output



Screenshot 5.10 : Successful Adding of URL Tutorial

Tutorial is added and now click on ‘View Available Books’ link to

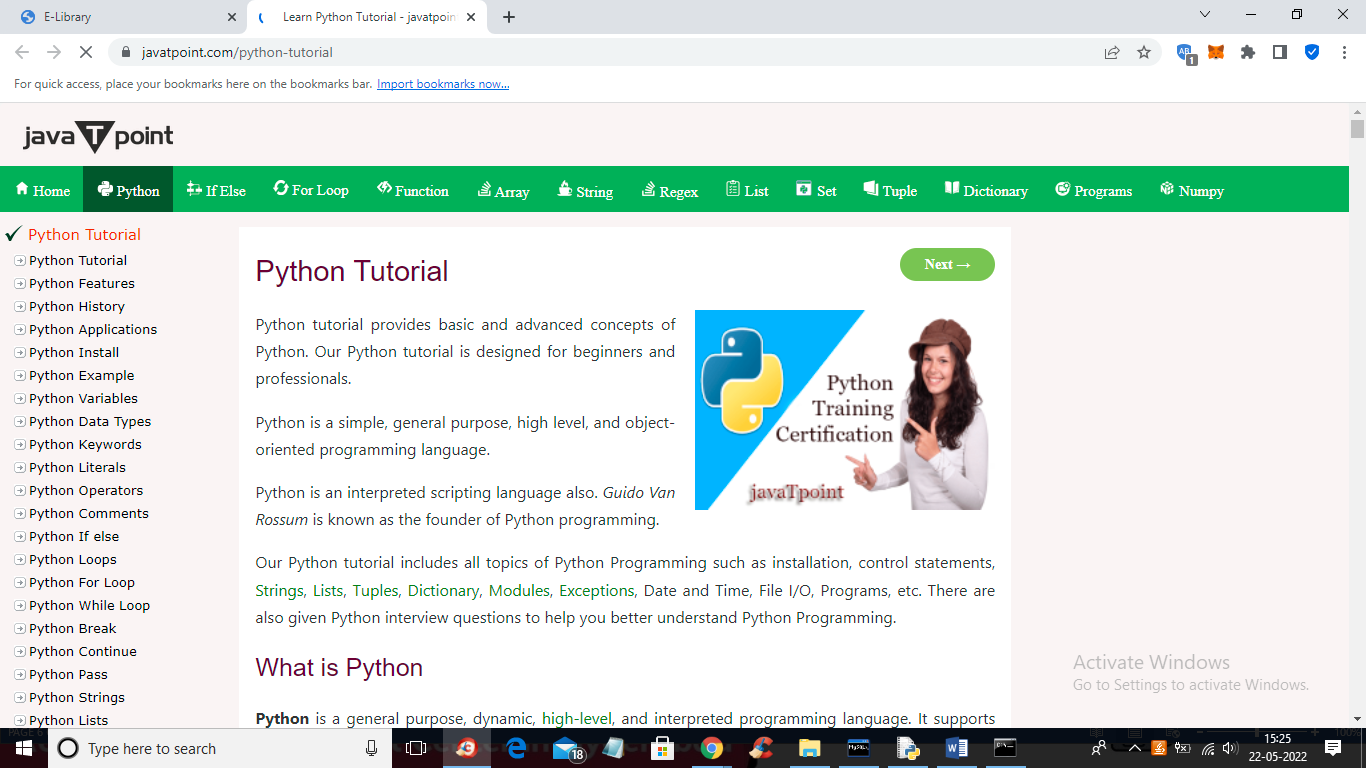
view available material



Screenshot 5.11 : Details of Available Books Videos URL Tutorial & Pdfs

Admin can view all book details and then click on desired image to view its tutorial or

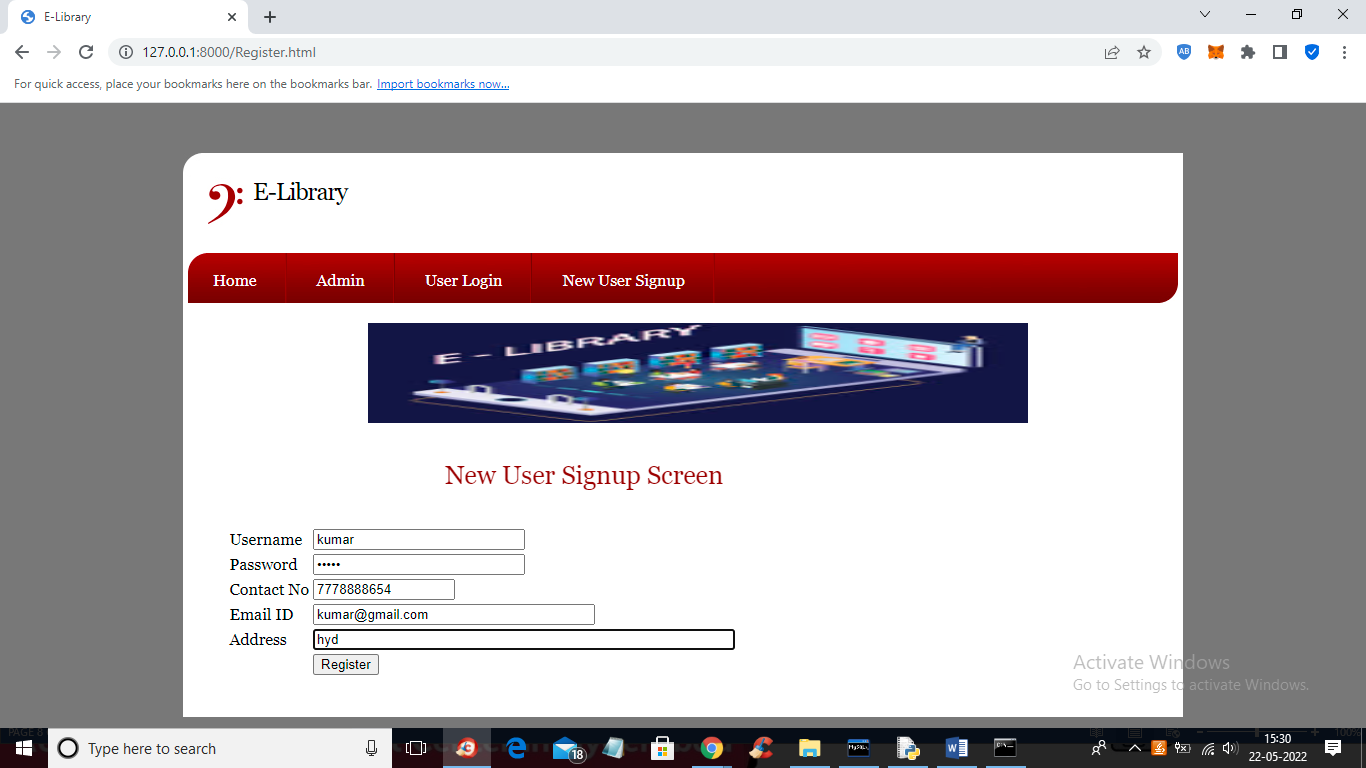
“Click Here” to delete that tutorial. So admin can add and delete tutorial



Screenshot 5.12 : One of the Pdf’s in available list

Tutorial URL is opened and in the same way we can see Videos, Pdf Tutorials

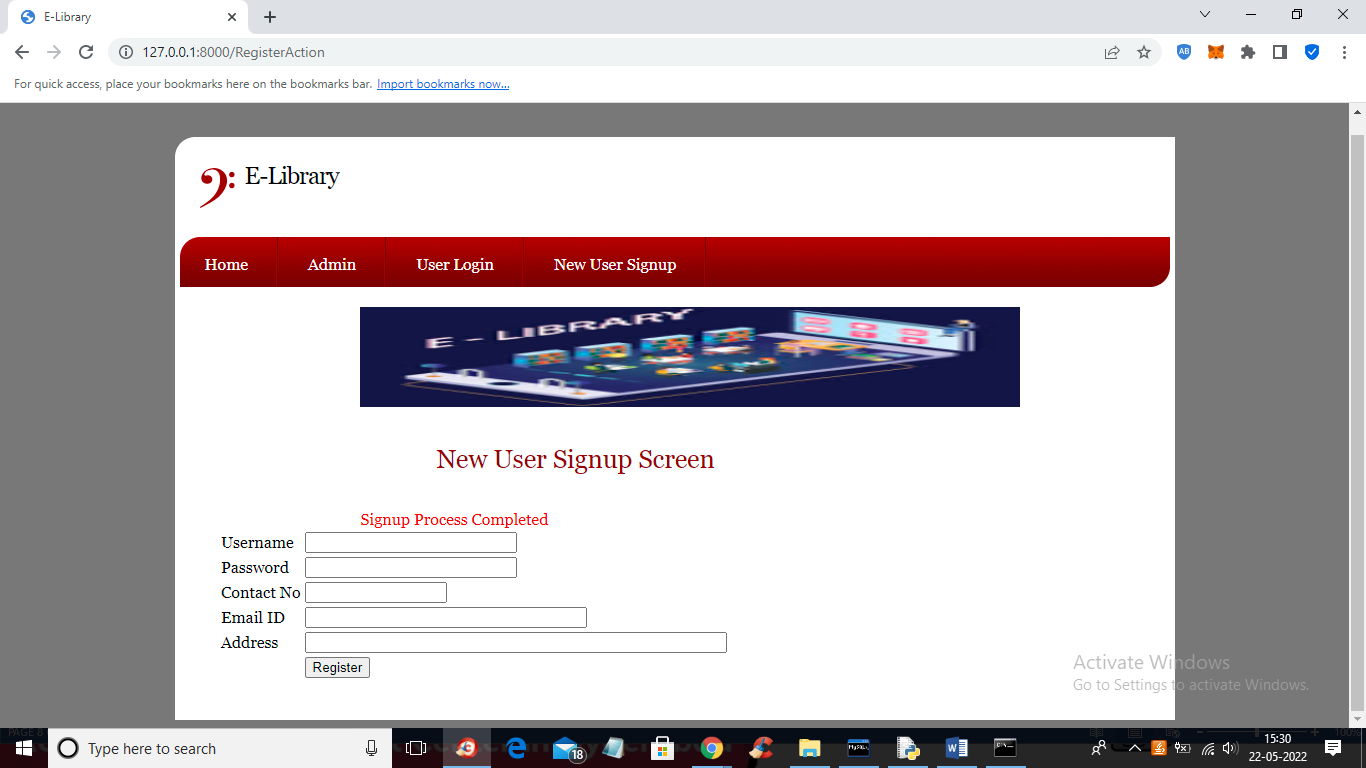
and other available material



Screenshot 5.13 : New User Sign Up

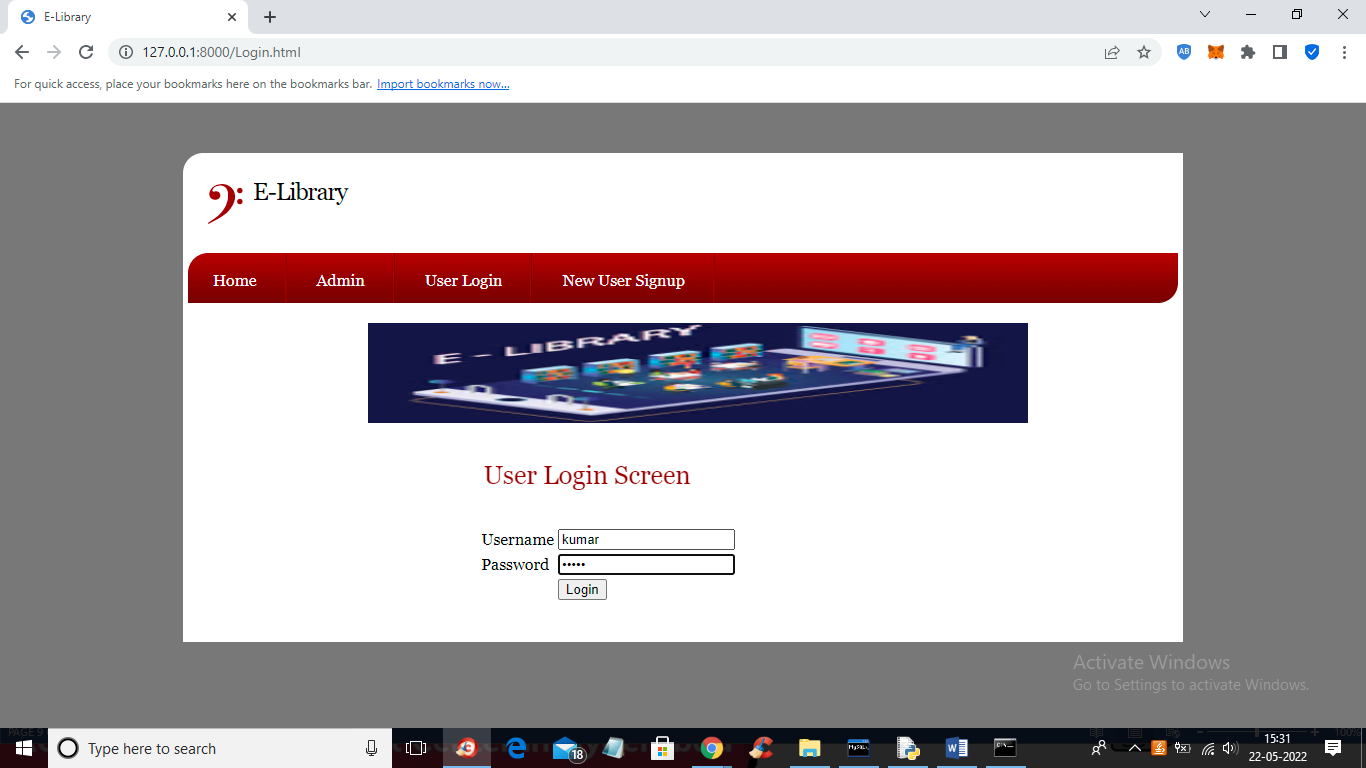
Student is signing up by entering the details and click button “Register”

to get below output



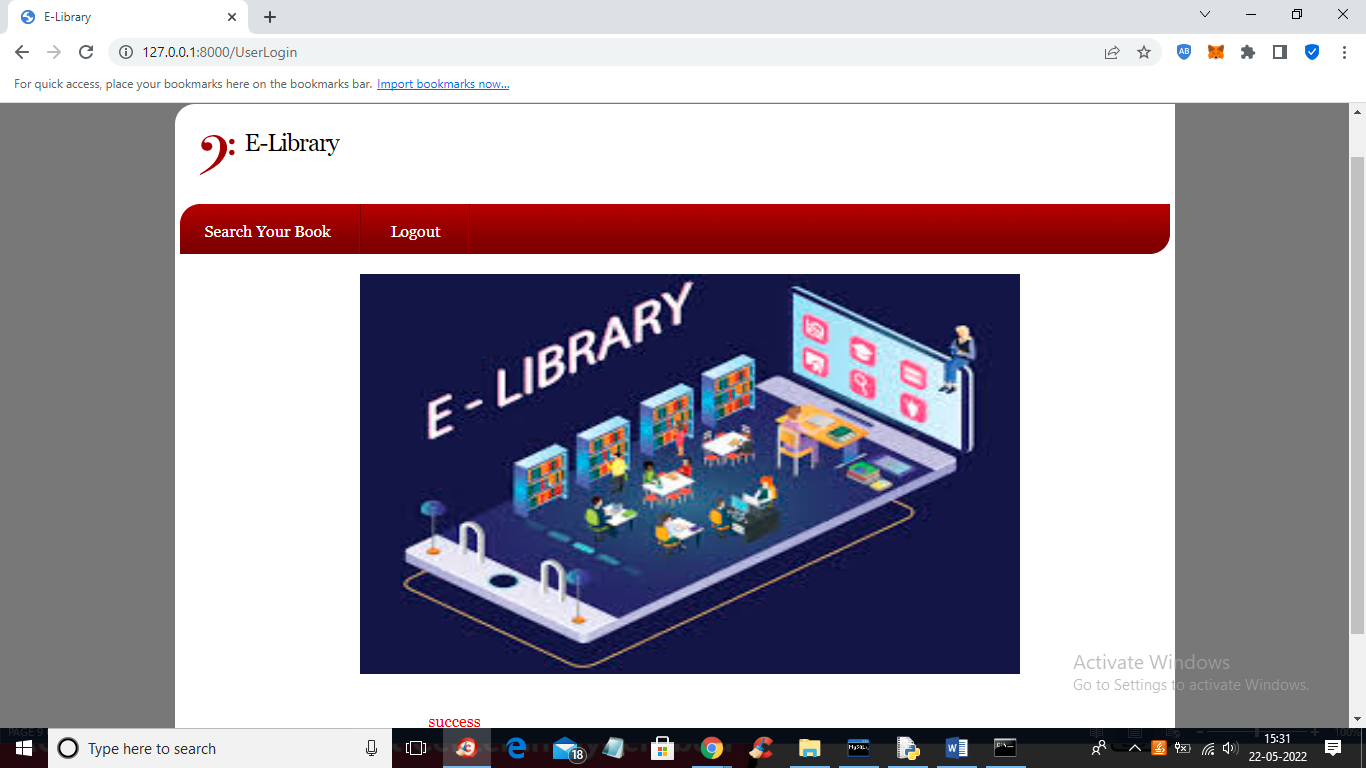
Screenshot 5.14 : Successfully Signed Up

Signup process completed and now click on ‘User Login’ link to login



Screenshot 5.15 : User Login

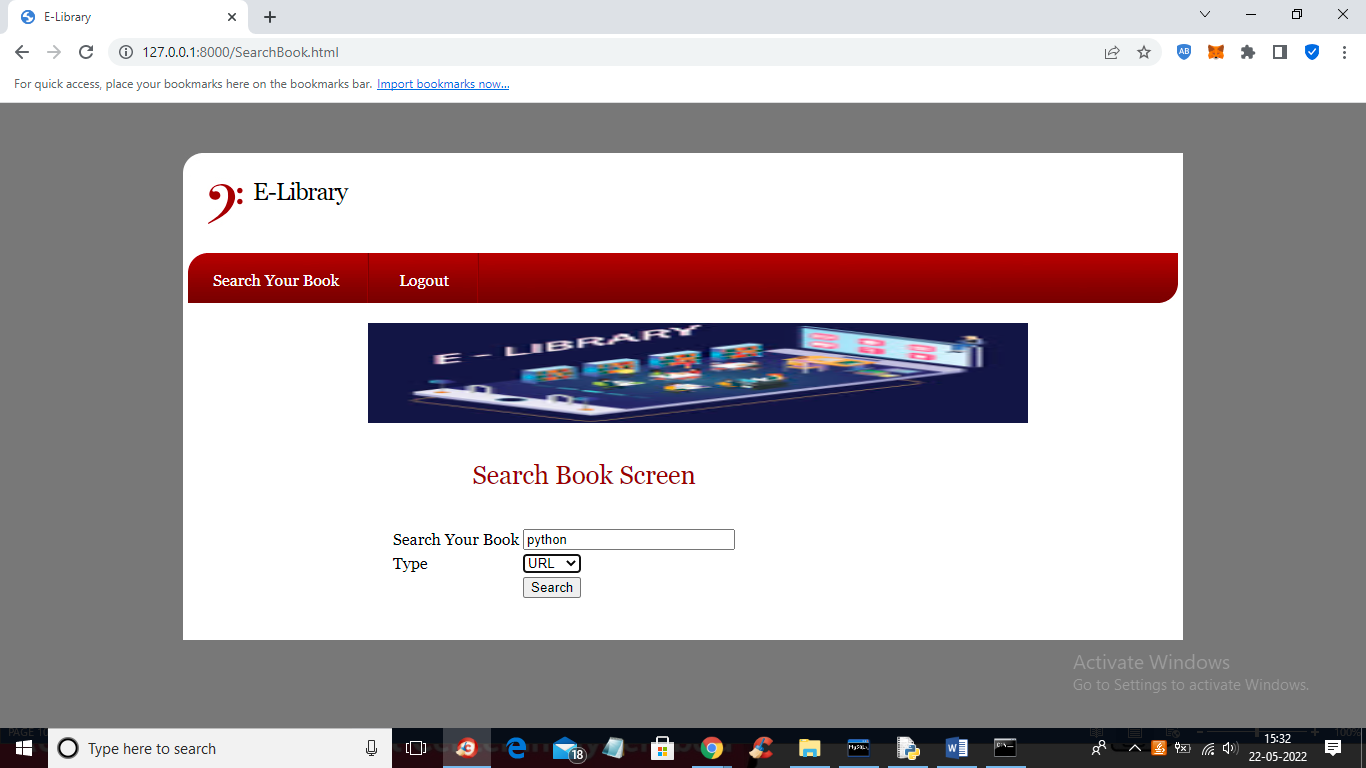
Enter the username and password for User login and click “Login”



Screenshot 5.16 : Successful Login and Start Search for material

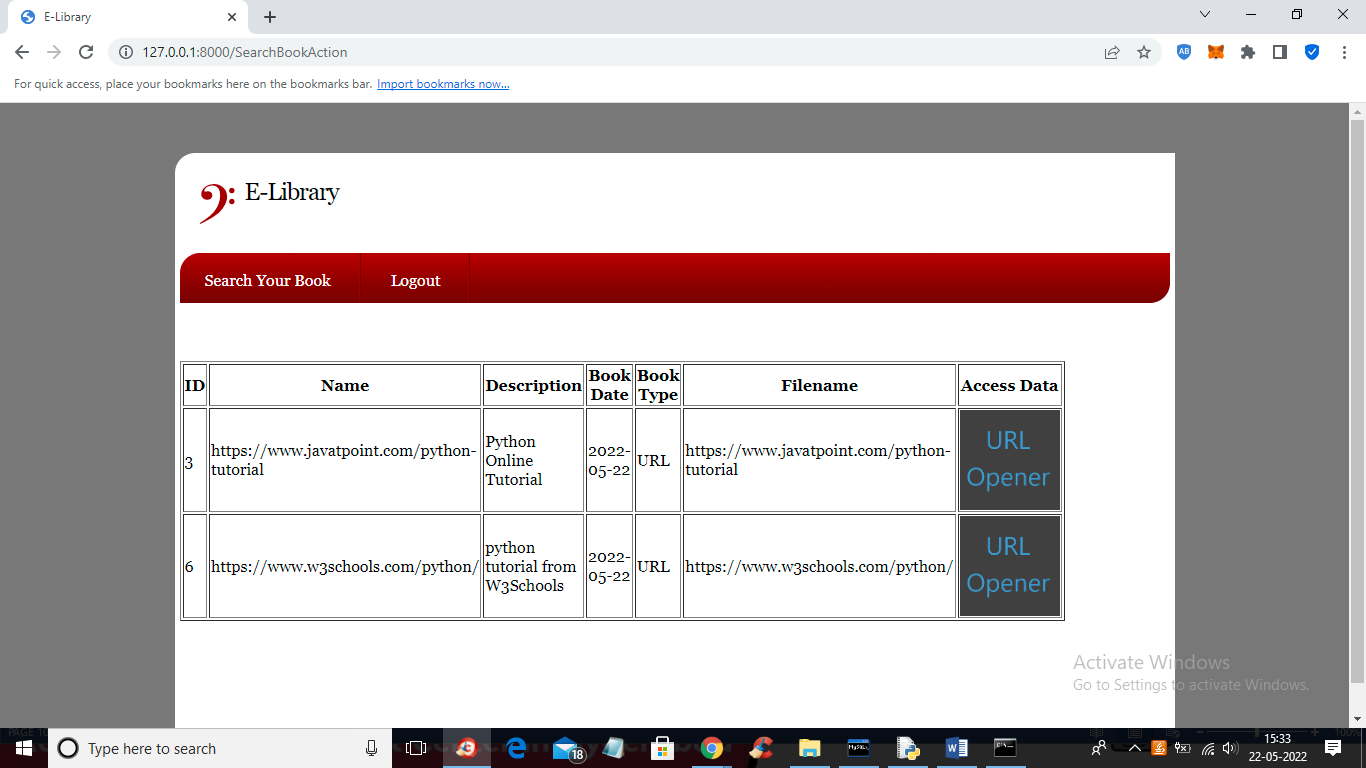
Login of user is successful and Click on ‘Search Your Book’ link we get the

next screen



Screenshot 5.17 : Search for URL Tutorial

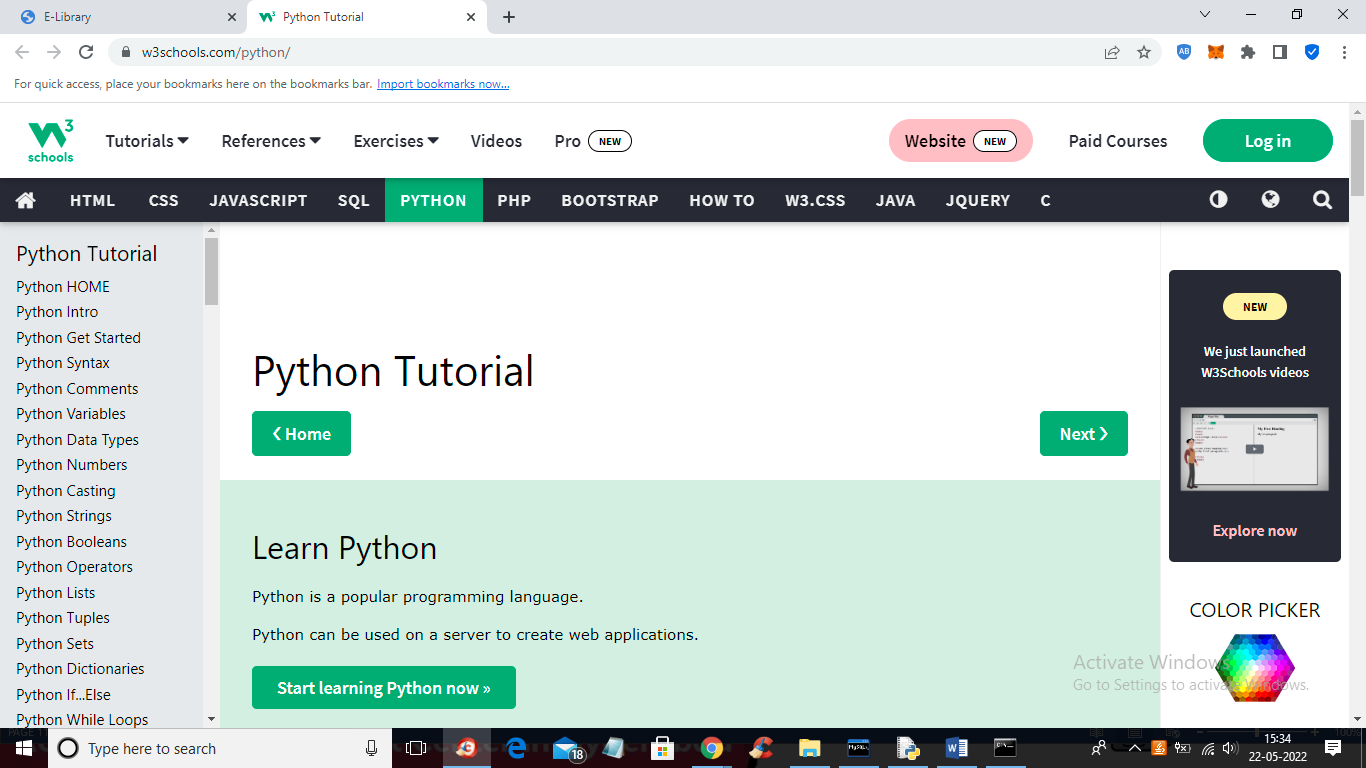
User is searching for python tutorial with URL and the user get below output



Screenshot 5.18 : Available URL Tutorials

User will get all URLS for python tutorial and can click on ‘URL Opener’

link to visit that URL



Screenshot 5.19 : Open and read tutorial from URL

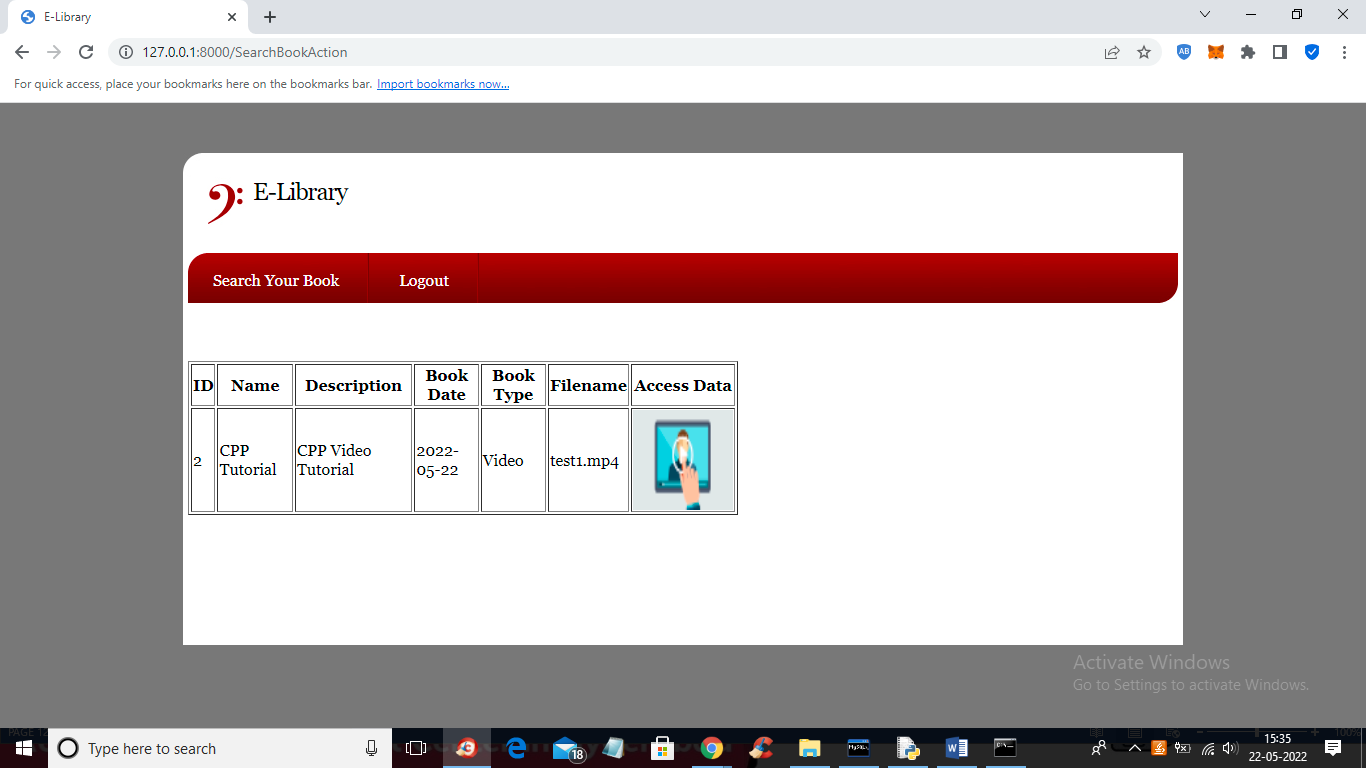
User can open and read tutorial from URL and can also search for other material



Screenshot 5.20 : Search for Videos

User searching for CPP video based tutorial and click button “search” to get

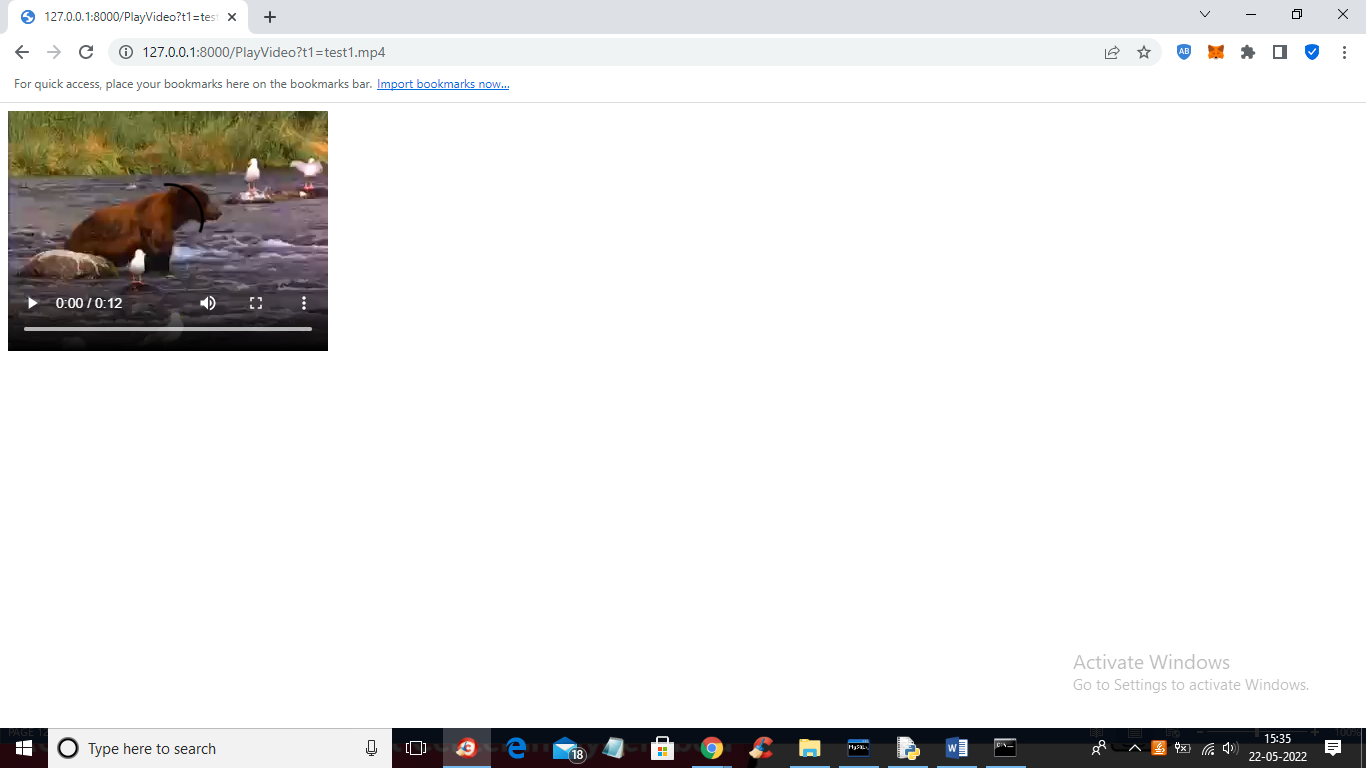
below output



Screenshot 5.21 : Available Videos

User finds the available tutorials and user can click on ‘PLAY ICON” to

play tutorial



Screenshot 5.22 : Playing Selected Video

Tutorial is playing and similarly you can search any other tutorial

## TESTING

#### TESTING

##### INTRODUCTION TO TESTING

The purpose of testing is to discover errors. Testing is the process of trying to discover every conceivable fault or weakness in a work product. It provides a way to check the functionality of components, sub assemblies, assemblies and/or a finished product It is the process of exercising software with the intent of ensuring that the Software system meets its requirements and user expectations and does not fail in an unacceptable manner. There are various types of test. Each test type addresses a specific testing requirement.

##### TYPES OF TESTING

###### UNIT TESTING

Unit testing involves the design of test cases that validate that the internal program logic is functioning properly, and that program inputs produce valid outputs. All decision branches and internal code flow should be validated. It is the testing of individual software units of the application .it is done after the completion of an individual unit before integration. This is a structural testing, that relies on knowledge of its construction and is invasive. Unit tests perform basic tests at component level and test a specific business process, application, and/or system configuration. Unit tests ensure that each unique path of a business process performs accurately to the documented specifications and contains clearly defined inputs and expected results.

###### INTEGRATION TESTING

Integration tests are designed to test integrated software components to determine if they actually run as one program. Testing is event driven and is more concerned with the basic outcome of screens or fields. Integration tests demonstrate that although the components were individually satisfaction, as shown by successfully unit testing, the combination of components is correct and consistent. Integration testing is specifically aimed at exposing the problems that arise from the combination of components.

###### FUNCTIONAL TESTING

Functional tests provide systematic demonstrations that functions tested are available as specified by the business and technical requirements, system documentation, and user manuals.

Functional testing is centered on the following items:

Valid Input : identified classes of valid input must be accepted.

Invalid Input : identified classes of invalid input must be rejected.

Functions : identified functions must be exercised.

Output : identified classes of application outputs must be exercised.

Systems/Procedures : interfacing systems or procedures must be invoked.

Organization and preparation of functional tests is focused on requirements, key functions, or special test cases. In addition, systematic coverage pertaining to identify Business process flows; data fields, predefined processes, and successive processes must be considered for testing. Before functional testing is complete, additional tests are identified and the effective value of current tests is determined.

**System Testing**

System testing ensures that the entire integrated software system meets requirements. It tests a configuration to ensure known and predictable results. An example of system testing is the configuration oriented system integration test. System testing is based on process descriptions and flows, emphasizing pre-driven process links and integration points.

**White Box Testing**

White Box Testing is a testing in which in which the software tester has knowledge of the inner workings, structure and language of the software, or at least its purpose. It is purpose. It is used to test areas that cannot be reached from a black box level.

**Black Box Testing**

Black Box Testing is testing the software without any knowledge of the inner workings, structure or language of the module being tested. Black box tests, as most other kinds of tests, must be written from a definitive source document, such as specification or requirements document, such as specification or requirements document. It is a testing in which the software under test is treated, as a black box .you cannot “see” into it. The test provides inputs and responds to outputs without considering how the software works.

##### 

**TEST CASES:**

**CLASSIFICATION**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **S.NO** | **Test Case** | **Excepted Result** | **Result** | **Remarks(IF Fails)** |
| 1. | User Register | If User registration successfully. | Pass | If already user email exist then it fails. |
| 2. | User Login | If Username and password is correct then it will getting valid page. | Pass | Un Register Users will not logged in. |
| 3. | User View User | Show our dataset | Pass | If Data set Not Available fail. |
| 4. | View Fast History Results | The Four Alarm Score Should be Displayed. | Pass | The Four Alarm Score Not Displaying fail |
| 5. | User Prediction | Display Review with true results | Pass | Results not True Fail |
| 6. | Show Detection  process | Display Detection  process | Pass | Results Not True Fail |
| 7. | Show Eye Blink Process | Display Eye Blink Process | Pass | If Results not Displayed Fail. |
| 8. | Admin login | Admin can login with his login credential. If success he get his home page | Pass | Invalid login details will not allowed here |
| 9. | Admin can activate the register users | Admin can activate the register user id | Pass | If user id not found then it won’t login |
| 10. | Results | For our Four models the accuracy and F1 Score | Pass | If Accuracy And F1 Score Not Displayed fail |

1. **. CONCLUSION**

##### 7 . CONCLUSION & FUTURE SCOPE

##### PROJECT CONCLUSION

It makes entire process online where student can search books, staff can generate

reports and do book transactions. It also has a facility for student login where student can login and can see status of books issued as well request for book or give some suggestions. It has a facility of teacher’s login where teachers can add lectures notes and also give necessary suggestion to library and also add info about workshops or events happening in our college or nearby college in the online notice board.

##### FUTURE SCOPE

There is a future scope of this facility that many more features such as online lectures video tutorials can be added by teachers as well as online assignments submission facility , a feature Of group chat where students can discuss various issues of engineering can be added to this project thus making it more interactive more user friendly and project which fulfills each users need in the best way possible.

### . BIBLIOGRAPHY

##### 8. BIBLIOGRAPHY

##### REFERENCES

* <http://www.w3schools.com/html/html_intro.asp>
* [http://www.Udemy.com/css/css\_background.asp](http://www.udemy.com/css/css_background.asp)
* <http://www.w3schools.com/js/js_datatypes.asp>

##### GITHUB LINK

* https://github.com/Anu-Lingampalli/E-LIBRARY