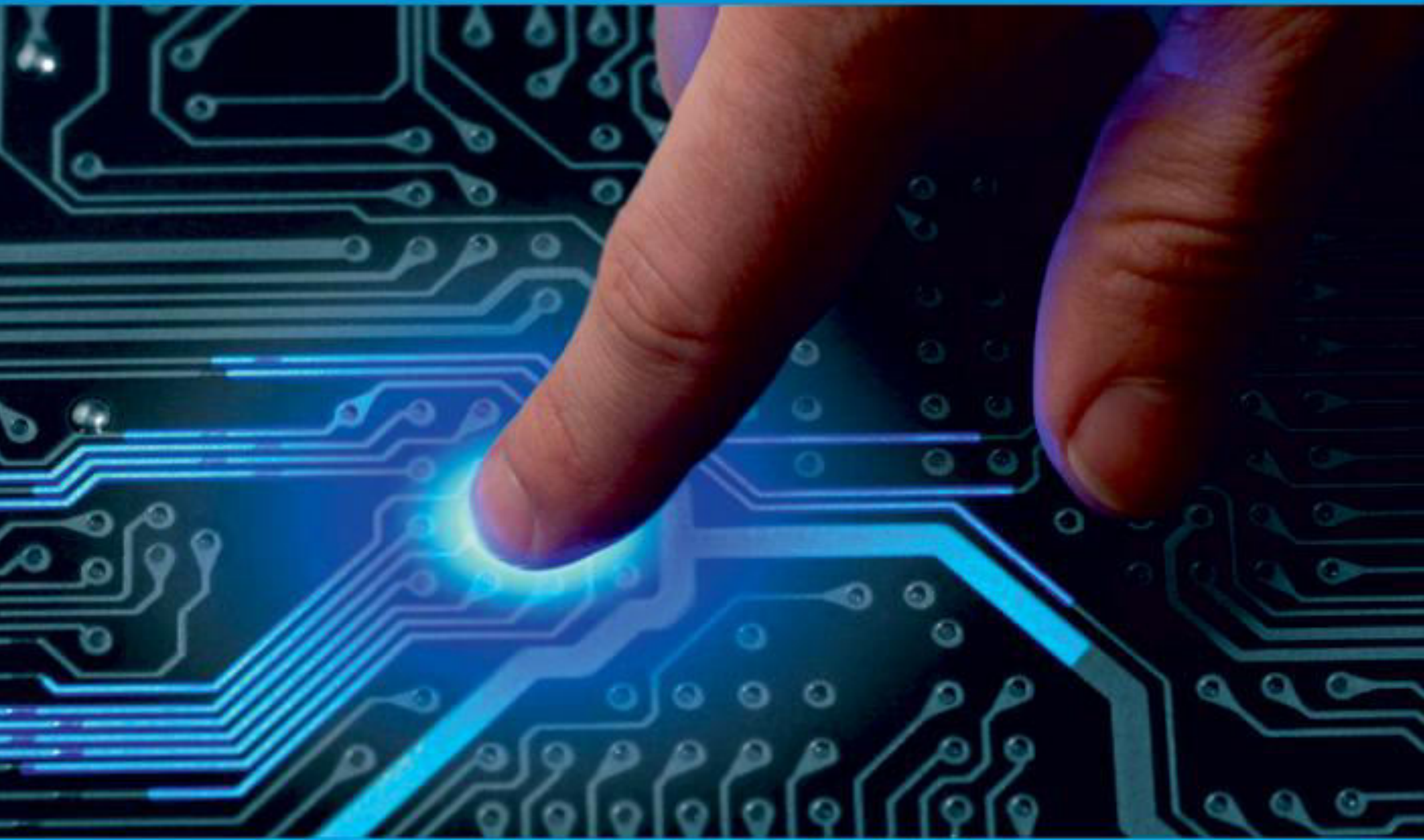
IJIRCCE

e-ISSN: 2320-9801 | p-ISSN: 2320-9798



INTERNATIONAL JOURNAL OF INNOVATIVE RESEARCH

IN COMPUTER & COMMUNICATION ENGINEERING

Volume 9, Issue 6, June 2021

INTERNATIONAL STANDARD SERIAL NUMBER INDIA

Impact Factor: 7.542

International Journal of Innovative Research in Computer and Communication Engineering

| e-ISSN: 2320-9801, p-ISSN: 2320-9798| www.ijircce.com | |Impact Factor: 7.542 |

|| Volume 9, Issue 6, June 2021 ||

| DOI: 10.15680/IJIRCCE.2021.0906077 |

Library Management Application: Index-Based Evaluation Using Keyword Extraction Algorithm

Adarsh Pawar[1], Tejaswini Nikam[1], Sakshi Gaikwad[1], Nikita Kulkarni[2]

B.E. Students, Department of Computer Engineering, KJEI's Trinity Academy of Engineering, Pune, India[1]

Head of Department, Department of Computer Engineering, KJEI's Trinity Academy of Engineering, Pune, India[2]

ABSTRACT: A library management system (LMS) is ERP i.e., Enterprise Resource Planning software that is designed and developed to manage all the in-house functions of a library. A LMS increases the efficiency of users as well as librarian. We implemented a LMS which is a simple and clear search interface for all the users. This project aims in developing a system that will help the users to easily find the book related to the topic he/she needs through this application. This project has some additional features to the traditional library application, as it finds the books related to the topic by using of keyword extraction algorithm, so as if he/she does not know the name of the book still through the context he/she will find all the books containing that topic. In this paper, we implemented a library management system web application. This web application is implemented by using python Django web framework and SQLite. The library management system web application performs various necessary tasks and procedures that are necessary for a library to function well. This web application has two main roles, the one is admin/librarian, and the other is the user. The admin registers the book along with its index. The image of the index is processed through OCR, preprocessing and TextRank algorithm. This module leads to extraction of keywords from the index of the book and saves in the database with respect to the same. We provided a search bar where user or admin can search for the book using name of the book, author name, category, or keywords/tags.

KEYWORDS:Energy efficient algorithm; Manets; total transmission energy; maximum number of hops; network lifetime

I. INTRODUCTION

The traditional libraries are a place where books, periodicals and sometimes materials like videos and recordings are available for people to use or borrow. But today libraries are adopted to digitization. A digital library application is basically collection of documents in organized electronic form, available on the Internet, we have the options to search the available resources (referred to as books), by their title, author, and are sorted by categories.

The library application should perform all necessary tasks for a library system to run smoothly. The library management web application also contains a module along with all the necessary functionalities, which allows users to search the book with the help of the keywords extracted from the respective book. This unlocks a whole new experience for the users in the traditional library management system. The library management systems keyword extraction module works in a sequence as:

1. OCR
2. Pre-processing
3. TextRank algorithm

The OCR that is Optical Character Recognition, extracts all the characters from an image file or a scanned file effectively, providing the output in text, digital and editable form. The text received from OCR contains all words symbols numbers, which are not required by the TextRank algorithm for extracting keywords. So, in the text is pre-processed by removing symbols numbers and stop words, which results in rich output from the TextRank algorithm.

II. Literature Review

Optical character recognition is a science that can help to translate various types of documents, images into easily analysable and searchable data. OCR engine provides the accuracy for searchable data by tesseract behind the leading commercial engines [1].

Information to a computer system from printed documents or image files is to be stored to re-utilize information. It helps in automatically retrieve and store the information provided by ocr engine [2].

Text pre-processing is a vital step of text classification and text mining generally. It is used to convert the original text data into raw data structure, and they are served to distinguish between various categories are identified [3].

Keyword, Key-phrases gives the summary of the text or any information for searchable data to users. Automatic keywords or key-phrase extraction techniques helps us to overcoming this challenging task [4].

Semantic textual similarity describes various regression techniques of supervised model to analyse degree of semantic equivalence between two text fragments are word phrases, sentences, paragraphs, or documents [5] .

OCR is software that converts printed text and images into digitized form such that it can be manipulated by machine [6]

In this paper for Automatic Keyword extraction embedding vectors are used to compute a similarity measure as an edge weight. Incorporating sentence importance scores derived from the TextRank model at a sentence level enhances an overall performance [7].

In this Firstly, we use the Readability algorithm to extract the text of the web page, and study the PageRank algorithm and TextRank algorithm, and then use the TextRank algorithm to extract keywords, key sentences, and abstracts. We also develop the web application that processes web page and text file. The application can input URL, text file, or text paragraph, then application can complete the extraction of main content, abstract, keywords and key sentences[8].

The proposed e-library system's characteristics were identified as important predictors of perceived usefulness and of the perceived ease of use of an e-library system. Utilizing the well-established technology acceptance model as a theoretical framework, critical external variables, which consist of two individual differences, three interface characteristics, and two system characteristics, were expected to significantly influence the intention to use an e-library[9]

III. Application Module

A. *Admin and User Login:*

The admin registers itself for the first time and then further access the library management web application using the login credentials. And has access to the various module provided. The user has its registration module through which it can generate the login credentials to access the system.

B. *Admin and User Modules:*

The admin once logged in has access to the following modules:

1. Add new books: Add new book to the database.
2. View available books: View available books in their library.
3. View students: View registers users of the library.
4. Issue book: Issue a book to the registered user.
5. View issued books: View issued books to the student.

The user has access to the following module once it logins:

1. View issued book: The user can view the details of the issued book.

C. *Universal Search Bar:*

The search bar is provided in the homepage menu of the library management system application. This universal search bar can be accessed by booth admin and the user. The reason to name it with prefix universal as it can take any input related to the book you need as in the form of book name, author name, category, or keywords.

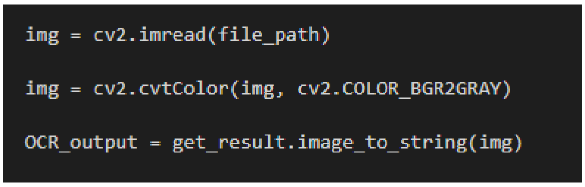
IV. SYSTEM DESIGN

The system is implemented using python Django framework and for database SQLite. The keyword extraction module consists of OCR, pre-processing and TextRank keyword extraction algorithm.

1. In the front-end the admin will give input as an image file of the index, while registering the book.
2. This index image of the book is processed by our keyword extraction module and are stored in the database with their respective book.
3. The user and admin now can access the information of the book using the universal search bar.

KEYWORD EXTRACTION MODULE

1. *OCR:*
   The input image of the index is processed with the help of the opencv library provided in python for when dealing with image files.



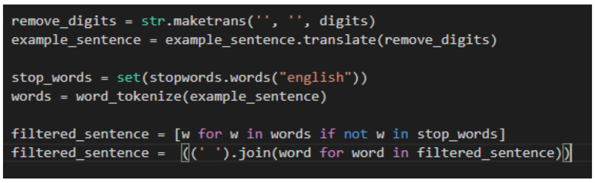
```
img = cv2.imread(file_path)

img = cv2.cvtColor(img, cv2.COLOR_BGR2GRAY)

OCR_output = get_result.image_to_string(img)
```

Fig 1. OCR module

2. *Pre-processing:*
   The output text words will be processed to exclude any unwanted characters as numbers and symbols. These unwanted characters are removed. With the help of NLTK stop words all the stop words will also be removed from the text, which will provide us with more proper and quality input for our further processing on the text for extracting keywords.



```
remove_digits = str.maketrans('', '', digits)
example_sentence = example_sentence.translate(remove_digits)

stop_words = set(stopwords.words("english"))
words = word_tokenize(example_sentence)

filtered_sentence = [w for w in words if not w in stop_words]
filtered_sentence =  (('  ').join(word for word in filtered_sentence))
```

Fig 2. Pre-processing module

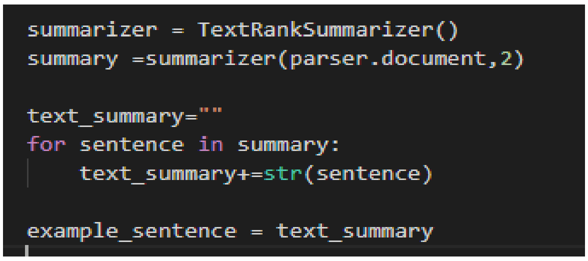
3. *TextRank Algorithm:*
   TextRank is a general purpose, graph based ranking algorithm for NLP. TextRank is an automatic summarization technique [4].
   Basic steps of textRank algorithm are:
   - First step is to extract the sentences from the document, either by splitting at whitespaces or full stops.
   - From the sentences extracted in step 1 Graph is created. The nodes represent the sentences, while the weight on the edges between two nodes is found by using a Similarity function, like Cosine Similarity or Jaccard Similarity.

- This step involves finding importance (scores) of each node by iterating the algorithm until convergence, i.e., until consistent scores are obtained.
- The sentences are sorted in a descending order based upon their scores. The first *k* sentences are chosen to be a part of the text summary.

It is an unsupervised graph-based technique used to extract the summary of a text. It uses the theory behind the PageRank algorithm [4]. TextRank model can be expressed as a weighted directed graph G= (V, E). The graph consists of a set of points V and a set of edges E and the set of edges E is the subset of V *V. The weight of the edge of arbitrary two points i and j are Wij. For a given point Vi, In V(i) represents the set of points that point to the point Vi, Out V(i) represents the set of points that point from the point Vi. TR V(i) represents the score of the point Vi obtained by the TextRank model. The formula of TextRank model can be defined as shown below:

$$WS(V_i) = (1-d) + d * \sum_{V_j \in In(V_i)} \frac{w_{ji}}{\sum_{V_k \in Out(V_j)} w_{jk}} WS(V_j)$$

In this formula, d is the same as d in the PageRank algorithm [3]. When using the TextRank algorithm to calculate the score of the points of the graph, it is required to specify the initial value of any given value to the points in the graph and then recursively compute the score of the point until convergence. After each point is convergent, the final score of the point represents the importance of the point in the graph.



```
summarizer = TextRankSummarizer()
summary =summarizer(parser.document,2)

text_summary=""
for sentence in summary:
    text_summary+=str(sentence)

example_sentence = text_summary
```

Fig 3. Keywords extraction module

V. Results

We have obtained decent results from TextRank algorithm. The following is an example of it:

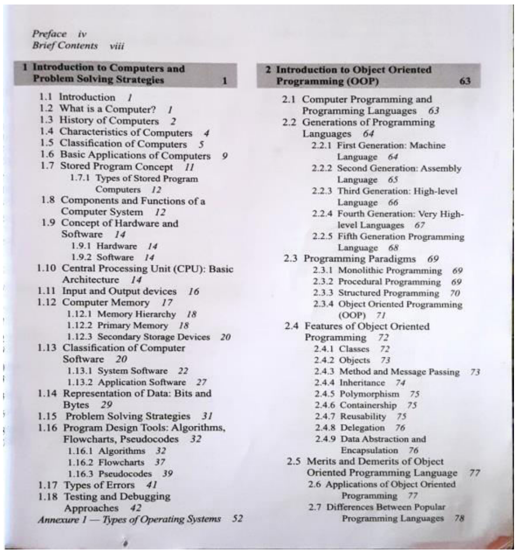
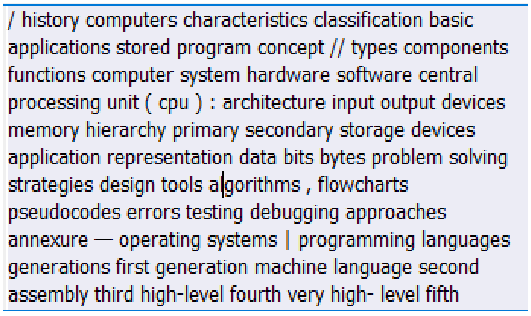


Fig 4. Input image

| e-ISSN: 2320-9801, p-ISSN: 2320-9798| www.ijircce.com | |Impact Factor: 7.542 |

|| Volume 9, Issue 6, June 2021 ||

| DOI: 10.15680/IJIRCCE.2021.0906077 |



/ history computers characteristics classification basic applications stored program concept // types components functions computer system hardware software central processing unit (cpu) : architecture input output devices memory hierarchy primary secondary storage devices application representation data bits bytes problem solving strategies design tools algorithms , flowcharts pseudocodes errors testing debugging approaches annexure — operating systems | programming languages generations first generation machine language second assembly third high-level fourth very high- level fifth

Fig 5. Output Image

Screenshots of the system:

A. User Interface:

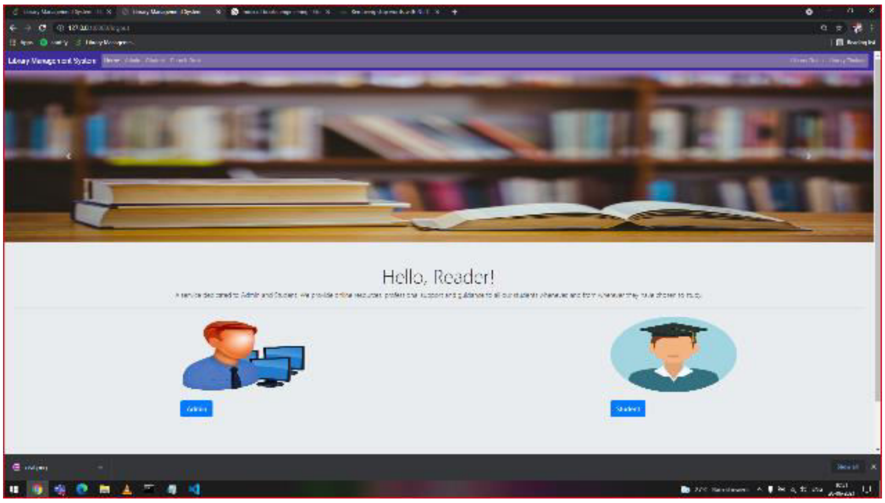


Fig 6. Home Page

B. Admin Interface:

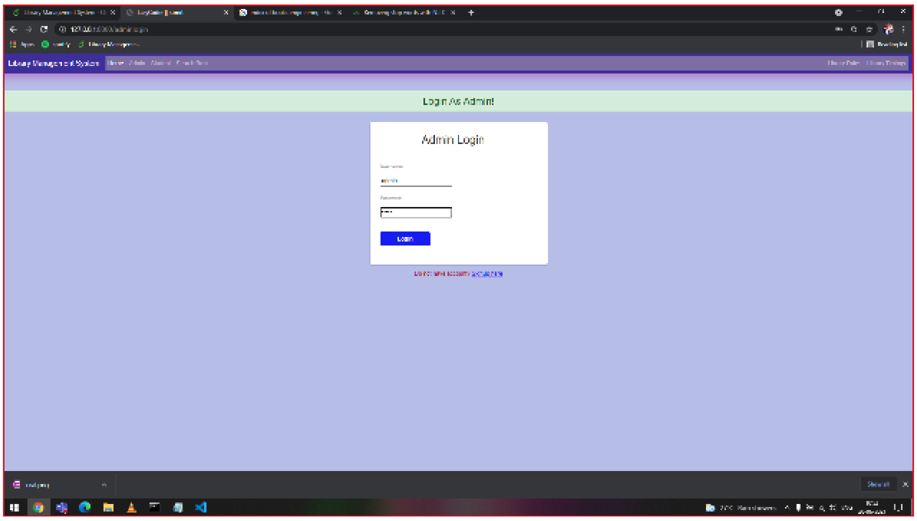
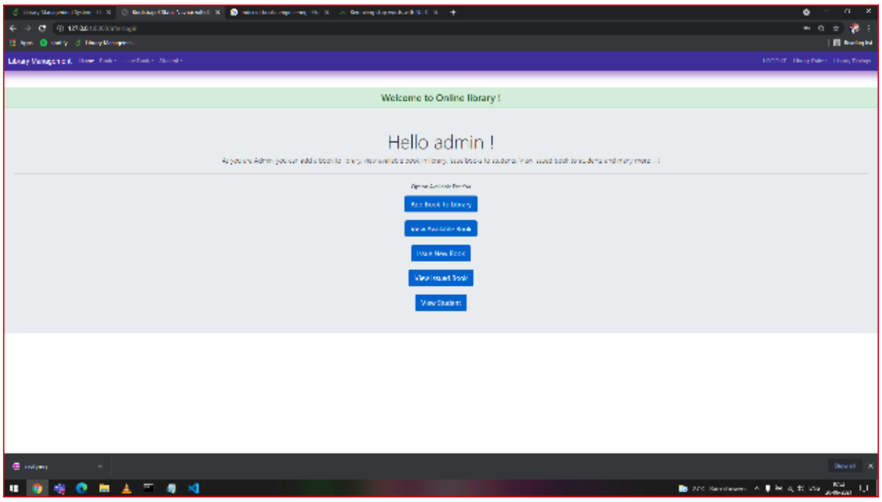


Fig 7. Admin Login

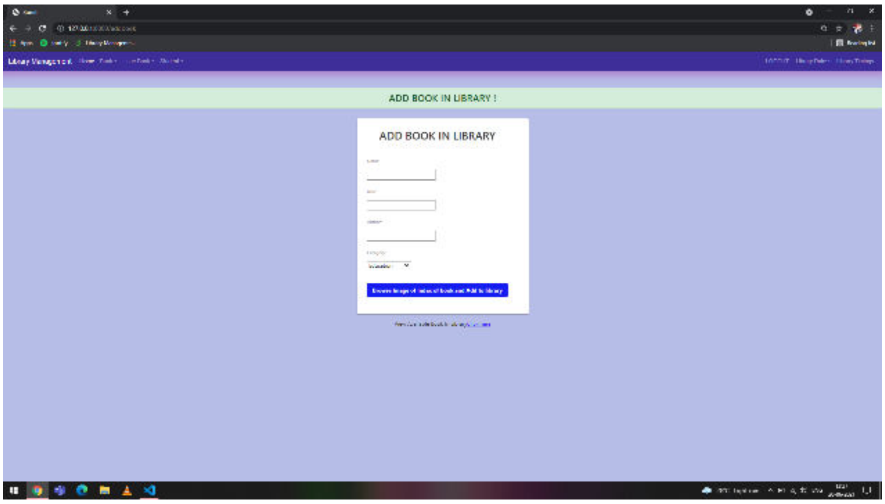


Welcome to Online library !

Hello admin !

As you are Admin you can add a book to library, view available book in library, issue books to students, view issued book to students and many more ...!

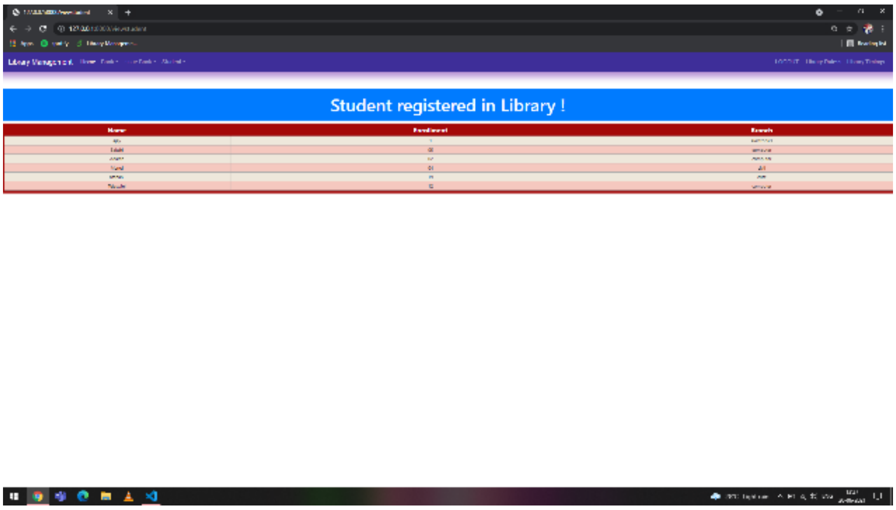
Fig 8. Admin Home Page



ADD BOOK IN LIBRARY !

ADD BOOK IN LIBRARY

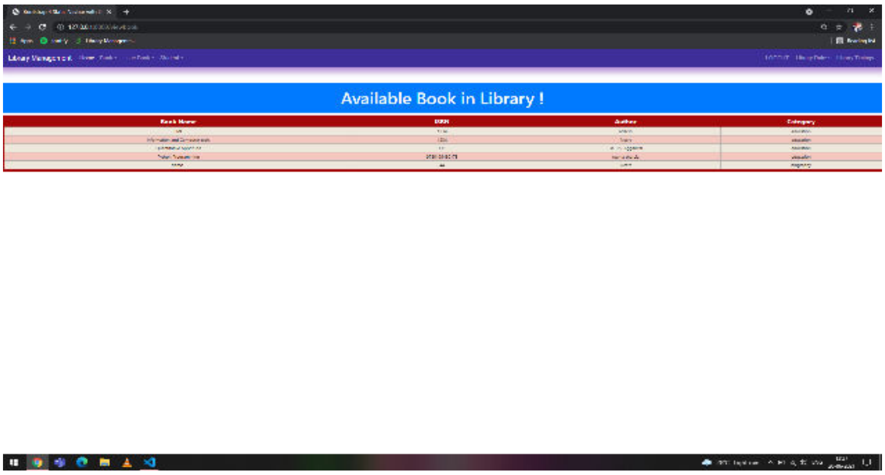
Fig 9. Add Book Page



Student registered in Library !

Name	[illegible]	[illegible]
[illegible]	[illegible]	[illegible]
[illegible]	[illegible]	[illegible]
[illegible]	[illegible]	[illegible]
[illegible]	[illegible]	[illegible]
[illegible]	[illegible]	[illegible]
[illegible]	[illegible]	[illegible]

Fig 10. Student List

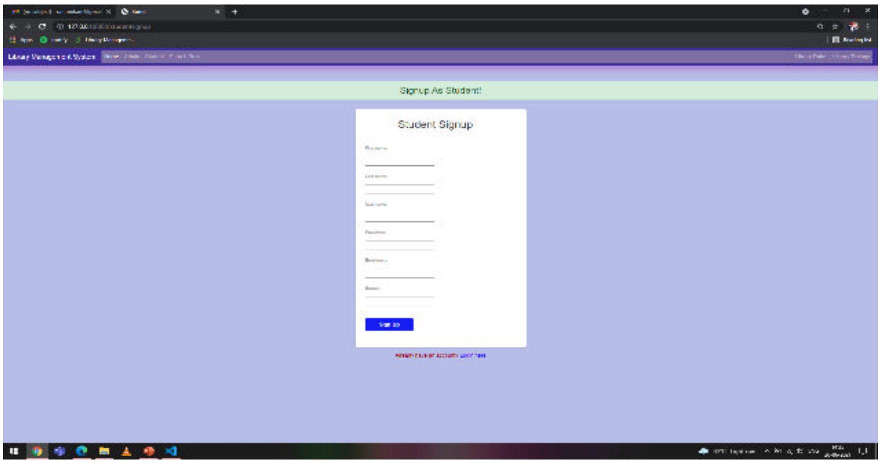


Available Book in Library !

Book Name	ISBN	Author	Category
[illegible]	[illegible]	[illegible]	[illegible]
[illegible]	[illegible]	[illegible]	[illegible]
[illegible]	[illegible]	[illegible]	[illegible]
[illegible]	[illegible]	[illegible]	[illegible]
[illegible]	[illegible]	[illegible]	[illegible]

Fig 11. Book List

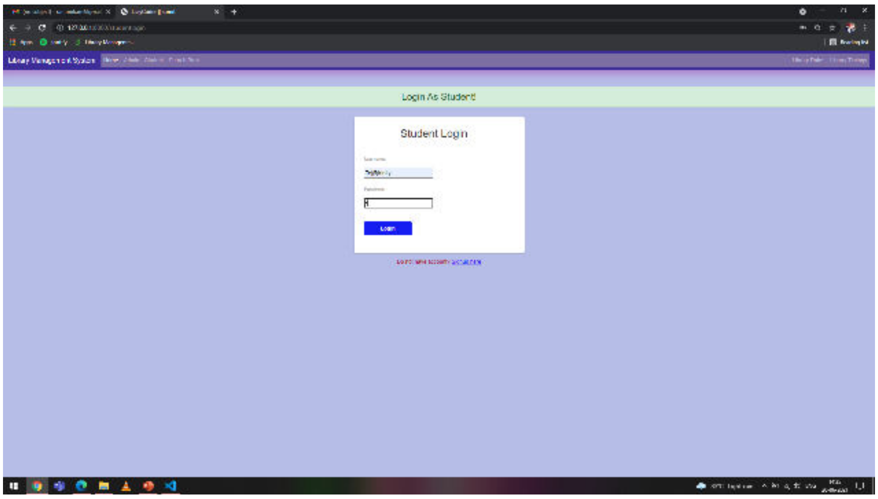
C. Student Interface:



Signup As Student!

Student Signup

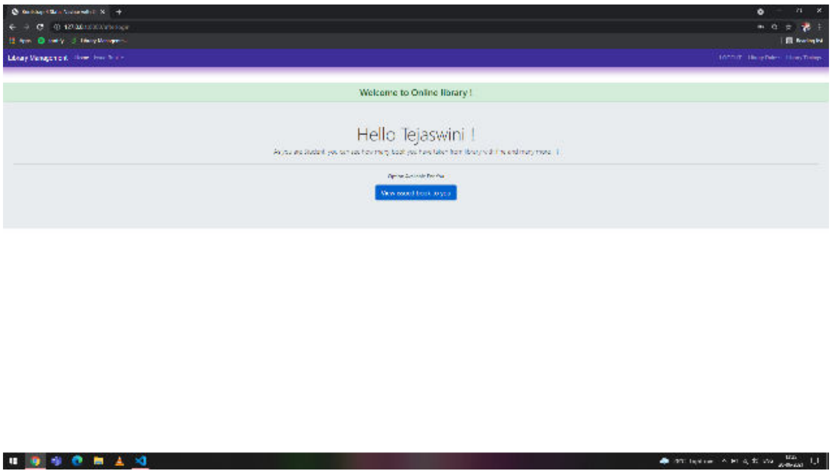
Fig 12. Student Registration Form



Login As Student!

Student Login

Fig 13. Student Login Form

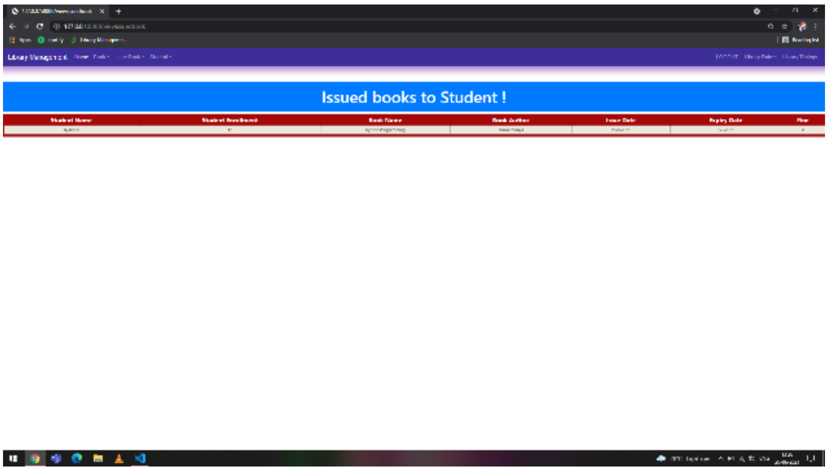


Welcome to Online library !

Hello Tejaswini !

As you are Student, you can see how many book you have taken from library with [illegible] and many more. !

Fig 14. Student Home Page



Issued books to Student !

Student Name	Student Enrollment	Book Name	Book Author	Issue Date	Expiry Date	Fine
[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]

Fig 15. Issued Book

VI. CONCLUSION

In this paper, we have implemented a system using OCR by converting image file into editable text file, by extracting the characters and various text pre-processing approaches with their implementations and applied TextRank algorithm to summarize and generate keyword which are useful to search therelated book that makes the system more efficient and user- friendly to the users.

V. FUTURE WORK

- Improve the quality of extracted keywords and key-phrases from the book.
- Optimize certain keyword extraction algorithm, which can extract keywords and key phrases from the books having smallsize index.
- Implementation of intelligent keyword extraction algorithm by providing the algorithm with some knowledge or learning method.
- Implementation of more fast and efficient search techniques for large datasets

REFERENCES

1. N. Islam, Z. Islam, and N. Noor, “A survey on optical character recognition system,” arXiv, no. December 2016, 2017.
2. K. Hamad and M. Kaya, “A Detailed Analysis of Optical Character Recognition Technology,” Int. J. Appl. Math. Electron. Comput., vol. 4, no. Special Issue-1, pp. 244–244, 2016, doi: 10.18100/ijamec.270374.
3. A. I. Kadhim, “An Evaluation of Preprocessing Techniques for Text Classification,” Int. J. Comput. Sci. Inf. Secur., vol. 16, no. 6, pp. 22–32, 2018.
4. M. G. Thushara, T. Mownika, and R. Mangamuru, “A comparative study on different keyword extraction algorithms,” Proc. 3rd Int. Conf. Comput. Methodol. Commun. ICCMC 2019, no. June, pp. 969–973, 2019, doi: 10.1109/ICCMC.2019.8819630.
5. T. Putta, “Semantic Textual Similarity Using Machine Learning Algorithms,” no. 8, pp. 10–15, 2017.
6. M. G. Marne, P. R. Futane, S. B. Kolekar, A. D. Lakhadive, and S. K. Marathe, “Identification of Optimal Optical Character Recognition (OCR) Engine for Proposed System,” Proc. - 2018 4th Int. Conf. Comput. Commun. Control Autom. ICCUBEA 2018, pp. 1–4, 2018, doi: 10.1109/ICCUBEA.2018.8697487.
7. P. Wongchaisuwat, “Automatic Keyword Extraction Using TextRank,” 2019 IEEE 6th Int. Conf. Ind. Eng. Appl. ICIEA 2019, pp. 377–381, 2019, doi: 10.1109/IEA.2019.8714976.
8. X. You, “Automatic Summarization and Keyword Extraction from Web Page or Text File,” 2019 IEEE 2nd Int. Conf. Comput. Commun. Eng. Technol. CCET 2019, pp. 154–158, 2019, doi: 10.1109/CCET48361.2019.8989315.
9. H. Jeong, “An investigation of user perceptions and behavioral intentions towards the e-library,” Libr. Collect. Acquis. Tech. Serv., vol. 35, no. 2–3, pp. 45–60, 2011, doi: 10.1016/j.lcats.2011.03.018.

INTERNATIONAL JOURNAL OF INNOVATIVE RESEARCH

IN COMPUTER & COMMUNICATION ENGINEERING