RSET
RAJAGIRI SCHOOL OF
ENGINEERING & TECHNOLOGY
(AUTONOMOUS)

Mini Project Report On

Automated Revision Note Generator

Submitted in partial fulfillment of the requirements for the award of the degree of

Bachelor of Technology

in

Computer Science & Engineering

By

ROHN RAPHAEL (U2103183)

SHAWN ANTONY SOBI (U2103195)

SHOBIN SHINO JOB (U2103196)

VINEET ABRAHAM KOSHY (U2103214)

Under the guidance of

Dr.Jisha G (Associate Professor,CS Dept.)

Department of Computer Science & Engineering
Rajagiri School of Engineering & Technology (Autonomous)
(Affiliated to APJ Abdul Kalam Technological University)
Rajagiri Valley, Kakkanad, Kochi, 682039
May 2024

CERTIFICATE

This is to certify that the mini project report entitled ***"Automated Revision Note Generator"*** *is a bonafide record of the work done by* ***Rohn Raphael (U2103183),Shawn Antony Sobi (U2103195),Shobin Shino Job (U2103196),Vineet Abraham Koshy (U2103214)****, submitted to the APJ Abdul Kalam Technological University in partial fulfillment of the requirements for the award of the degree of Bachelor of Technology (B. Tech.) in Computer Science and Engineering during the academic year 2023-2024.*

Dr.Jisha G
Project Guide
Associate Professor
Dept. of CSE
RSET

Dr.Uma Narayanan
Project Coordinator
Asst. Professor
Dept. of CSE
RSET

Dr. Preetha K G
Head of the Department
Professor
Dept. of CSE
RSET

ACKNOWLEDGEMENTS

We wish to express our sincere gratitude towards Dr P. S. Sreejith, Principal of RSET, and Dr. Preetha K.G., Head of the Department of Computer Science and Engineering for providing us with the opportunity to undertake our mini project, "Automated Revision Notes Generator".

We are highly indebted to our project coordinators, **Dr.Uma Narayanan**, Asst. Professor, Department of Computer Science and Engineering and **Ms. Liya Joseph**, Asst. Professor, Department of Computer Science and Engineering for their valuable support.

It is indeed our pleasure and a moment of satisfaction for us to express our sincere gratitude to our project guide **Dr Jisha G** for her patience and all the priceless advice and wisdom she has shared with us.

Last but not the least, we would like to express our sincere gratitude towards all other teachers and friends for their continuous support and constructive ideas.

Rohn Raphael

Shawn Antony Sobi

Shobin Shino Job

Vineet Abraham Koshy

Abstract

This project focuses on developing an Automated Revision Notes Generator aimed at enhancing exam preparation by analyzing previous year question papers. The system uses Natural Language Processing (NLP) algorithms to identify common and crucial topics, generating comprehensive and concise revision notes. Users can upload PDF notes, which are converted to text and compared with a dataset of keywords to identify and summarize the most important topics. The generated summaries, facilitated by GPT-3 (Gemini), provide targeted and efficient revision material, ultimately improving study efficiency and academic performance.

Contents

List of Figures

Chapter 1

Introduction

1.1 Background

In the realm of education, the volume of study material and the need for efficient revision strategies are critical challenges faced by students. Traditional methods of manual note-taking and summarization can be time-consuming and often fail to highlight the most important topics effectively. The rise of digital technologies presents an opportunity to enhance the learning process through automation and intelligent data analysis.

Automated Revision Note Generators are designed to address these challenges by leveraging advanced technologies such as Natural Language Processing (NLP) . These systems analyze large datasets, such as previous year question papers, to extract and summarize key topics. By doing so, they provide students with concise, relevant, and high-quality revision notes, enabling more focused and efficient study sessions.

This background sets the stage for the creation of an Automated Revision Note Generator, aimed at revolutionizing exam preparation and enhancing academic performance through innovative use of technology.

1.2 Problem Definition

To create an Automated Revision Notes Generator utilizing NLP and summarization api, providing students with personalized and condensed study materials extracted from previous year question papers for enhanced exam preparation.

1.3 Scope and Motivation

Scope

Uses NLP techniques for pattern identification. Takes a text document (which contains

notes of a particular subject) as input. Compares it with the available dataset to identify important topics. Summarize important topics in the given documents. Export functionalities for the generated revision notes.

Motivation

The challenge of sifting through extensive study material under time constraints often leads to inefficient exam preparation for students. Traditional note-taking methods are time-consuming and may not consistently highlight the most crucial topics. Leveraging advanced technologies like text pre-processing can automate and optimize this process, making revision more focused and effective. An Automated Revision Notes Generator can significantly reduce study time while enhancing the quality of revision materials. This innovation aims to empower students with tools that improve study efficiency and boost academic performance.

1.4 Objectives

- **Automated Note Generation**: Develop a system that automatically generates comprehensive and concise revision notes from previous year question papers.

- **Leverage Advanced Technologies**: Utilize Natural Language Processing (NLP) to identify and summarize key topics effectively.

- **Enhance Study Efficiency**: Provide students with high-quality, focused revision materials to optimize their study time and improve preparation.

- **User-Friendly Interface**: Design an intuitive interface that allows easy upload of question papers, customization of preferences, and access to generated notes.

- **Support Multiple Formats**: Enable users to save and share their revision notes in various formats, such as PDF and text files, for flexibility and convenience.

1.5 Challenges

1. **Accurate Topic Extraction**: Ensuring the product accurately identifies and extracts the most relevant topics from diverse and unstructured question papers.
2. **Text Summarization Quality**: Developing summarization algorithms that can condense information effectively without losing essential details, ensuring the notes are both concise and comprehensive.
3. **Handling Diverse Input Formats**: Creating a robust system capable of processing and converting various input formats of question papers (e.g., PDFs, images) into text for analysis.
4. **User Interface Design**: Designing an intuitive and user-friendly interface that caters to a wide range of user preferences and technical abilities, ensuring ease of use.
5. **Scalability and Performance**: Building a system that can handle large volumes of data and generate notes quickly, maintaining performance and accuracy as the dataset grows.

1.6 Assumptions

1. **Availability of Data**: Sufficient and diverse previous year question papers are available for analysis.
2. **User Technical Proficiency**: Users have basic technical skills to upload PDFs and interact with the software interface.
3. **Consistency in Question Papers**: The format and language used in the question papers are relatively consistent, facilitating more accurate topic extraction and summarization.
4. **Reliable Internet Access**: Users have reliable internet access for uploading question papers, processing data, and downloading generated notes.
5. **Third-Party Tools**: Necessary third-party APIs, libraries or tools are accessible and compatible with the development environment.

1.7 Societal / Industrial Relevance

The Automated Revision Notes Generator holds significant societal and industrial relevance by addressing the widespread challenge of efficient exam preparation. In the educational sector, this tool can revolutionize how students study, making learning more targeted and effective, thereby enhancing academic performance and reducing stress. For educators and institutions, it offers a means to support students with high-quality revision materials without additional workload. In the industry, such technology can be adapted for corporate training and continuous professional development, enabling employees to efficiently distill large volumes of information into actionable insights, thus fostering a more knowledgeable and competent workforce.

1.8 Organization of the Report

The report is structured as follows:

1. **Chapter 1**- Introduction: Provides background, objectives, and challenges of the project.
2. **Chapter 2**- Software Requirements Specification: Describes the project's overall description, including system features and nonfunctional requirements.
3. **Chapter 3**- System Architecture and Design: Presents the project's archi tecture, user interfaces, and implementation strategies

Chapter 2

Software Requirements Specification

2.1 Introduction

2.1.1 **Purpose:** The purpose of the revision note generator is to streamline and enhance the efficiency of the revision process for students by identifying and summarizing key topics from previous year question papers. This utility is specifically designed to aid in last-minute revisions, where time is of the essence and students need to focus on the most important and frequently tested concepts.

2.1.2 **Product Scope:** The Automated Revision Notes Generator is designed to analyze a repository of previous year question papers, utilizing Natural Language Processing (NLP) algorithms to identify recurring patterns, frequently asked questions, and common themes. The system generates concise and informative revision notes, providing students with an efficient and focused exam preparation tool. Its main goal is to improve the educational experience of students by providing them with a way to help them in their last minute revision.

2.2 Overall Description

2.2.1 **Product Perspective:** This product is a new, self-contained application designed specifically to address the need for efficient creation of revision notes by students. It is not part of a product family or intended as a replacement for existing systems, but rather as a standalone solution to enhance the revision process.While this application can function independently, it can also serve as a component within larger educational systems or platforms, augmenting their capabilities by providing a specialized tool for revision note generation.

2.2.2 **Product Functions:** The major functions are as follows:

NLP-based Topic Recognition and Identification:- Utilize Natural Language Processing (NLP) to identify subjects, topics, and recurring patterns within question papers.- Determine common and important topics based on relevance and significance.
Text Summarization and Revision Note Generation:- Apply text summarization algorithms to distill information from identified important topics.- Condense content while retaining core meanings, generating concise and informative revision notes.
Export and Accessibility Features*- Support export functionalities for saving and sharing generated revision notes in various formats (e.g., PDF, text).- Ensure limited offline access for continued work and compatibility with modern web browsers.

2.2.3 **Operating Environment:** The hardware platform includes modern server infrastructure like cloud-based virtual servers. The application is designed to be accessible across a wide range of devices, including desktop computers, laptops, tablets, and smartphones, without requiring specific hardware capabilities beyond internet connectivity and a web browser. It is compatible with major operating systems such as Windows and Linux. Relational databases such as MySQL, Git for version control and web servers such as Nginx or Apache to serve the Flask application might be required.

2.2.4 **Design and Implementation Constraints:** The main constraints for the revision note generator include compatibility limitations across different operating systems, web browsers, and devices, ensuring that the application functions seamlessly regardless of the user's platform. Resource constraints such as memory and processing power must be considered to optimize performance and ensure efficient operation, particularly during the note generation process. Security considerations are crucial to safeguard user data and privacy, requiring robust measures to protect against unauthorized access and data breaches. Technical constraints related to the capabilities of underlying technologies and frameworks may influence development decisions, impacting features and functionality. Usability challenges must be addressed to design an intuitive interface and ensure an effective user experience, enabling users to easily create and customize their revision notes.

2.2.5 **Assumptions and Dependencies:** In the development of the Automated Revision Notes Generator, several assumptions and dependencies have been identified. The software assumes the availability of a diverse and comprehensive repository of previous

year question papers, covering a wide range of subjects and topics. This assumption is crucial as the effectiveness of the system relies heavily on the richness and diversity of the ingested question papers. It is also important to assume that the text document produced by the user will cover all the topics of the particular subject. Additionally, the system assumes that users have access to a reliable internet connection for utilizing the Automated Revision Notes Generator. Since the system operates as a web-based application, uninterrupted internet connectivity is essential for various functionalities, including user interactions, data uploads, and accessing software features.

2.3 External Interface Requirements

2.3.1 **User Interfaces:** Homepage and Welcoming Message: Upon entering the application, users are greeted with a clean, minimalistic homepage featuring a welcoming message that introduces the application and its benefits. The message is inviting and clearly communicates the app's purpose to facilitate last-minute revision by summarizing important topics from previous question papers.

Upload Section: A prominent feature on the homepage is the section where users can upload their notes in PDF format. The subject name is taken as input along with the uploaded file. This area is designed for ease of use, with clear instructions and a simple, intuitive interface that allows users to upload documents with just a few clicks.

2.3.2 **Hardware Interfaces:** The hardware interfaces include having a processor of intel i3 or above. The minimum requirement for RAM is 128 MB. No specific storage space required on the user's device since the application is web-based. A standard monitor and keyboard must be required for proper user interaction along with a compatible mouse. It can also be implemented in smart phones. A reliable internet connection is necessary for accessing and utilizing the web-based application.

2.3.3 **Software Interfaces:** Python is the primary programming language used along with flask, which is a web application framework for Python, used to create the web server. Data is stored as a csv file. Incoming data include the notes that are to be uploaded as pdf, in addition to the subject name. The previous year question papers also

are incoming data which are used to create the dataset to filter out important topics by finding the repeated questions. The outgoing data includes the auto generated revision notes which facilitate last minute revision. The notes can be exported as pdf.

2.3.4 **Communications Interfaces:** HTTP/HTTPS is the primary protocol for communication between the client and the server. FTP is used for file transfers. Synchronization techniques are required for data consistency across devices or sessions. This is done by the use of optimistic concurrency control or versioning techniques to manage simultaneous edits or updates.

2.4 System Features

2.4.1 Document input and Upload

2.4.1.1 Description and Priority This feature allows the users of the website to upload pdf or word documents. Giving an input document and uploading it is the initial step of this process.

2.4.1.2 Stimulus/Response Sequences Stimulus: when the user accesses the website they can drop the required document in the input field.The system responds by highlighting the upload button prompting the user to upload for further conversion. Response : Upon selecting the upload button, the document is now added as input to the system .

2.4.1.3 Functional Requirements

REQ- 1: The website provides a user-friendly interface, for uploading and converting text documents.

REQ- 2: The website should support standard text format ensuring compatibility.

REQ- 3: In the situation where the user inputs anything other than text(images or audio files,etc) an error message will be displayed

2.4.2 Note Generation

2.4.2.1 Description and Priority The extracted text from the input document is summarized by selecting most repeated topics according to priority and AI is used to summarize the text to short revision notes.

2.4.2.2 Stimulus/Response Sequences Stimulus: The system highlights the summarize button after the text from the document is extracted. The user can choose when to create the revision note. Response : Upon selecting the summarize button, the user is

navigated to a textbox that contains the revision note for the given document.

2.4.2.3 Functional Requirements

REQ-1: Provides a user-friendly interface

REQ-2: TBD (User can add specific topics that will have special priority of its own)

2.4.3 **Export produced Revision Notes**

2.4.3.1 Description and Priority Users can easily share their revision notes with peers, instructors, or mentors for collaboration or feedback purposes.Exporting revision notes provides users with a backup copy, ensuring they have access to their study materials even if they lose access to the online platform or encounter technical issues.

2.4.3.2 Stimulus/Response Sequences Stimulus: After the notes have been displayed, the export button gets highlighted Response: Selecting the export button downloads the summarized notes into the user's system.

2.4.2.3 Functional Requirements

REQ-1:Allows the user to select the desired document format for exporting the output.

REQ-2: Lets the user make personal changes to the text before exporting.

2.5 Other Nonfunctional Requirements

2.5.1 Performance Requirements

The major performance requirements include faster response time, which is a must for students who are about to appear for an examination. The faster the notes generated, the better. Summarization should not take more than 1 minute to complete. The loading of the web page should not take more than 3 seconds. It should also ensure data integrity, that is, ensure that user data and processed content are accurately saved and retrieved without loss or corruption. It should be optimized for efficient use of server resources (CPU, memory, storage) to handle processing of documents and generation of revision notes. Implementing caching strategies for frequently accessed data can reduce database load and improve response times.

2.5.2 **Safety Requirements:** The safety requirements for the Automated Revision Notes Generator focus on ensuring the secure and responsible use of the software, emphasizing the protection of user data, ethical usage, and adherence to relevant policies and regula-

tions. Users must agree to a code of conduct outlining ethical usage of the Automated Revision Notes Generator, prohibiting any misuse for academic dishonesty. To promote responsible usage and maintain the integrity of the educational process, users must adhere to ethical standards when utilizing the generated revision notes. This requirement ensures a respectful and fair environment for all users. The system must comply with relevant data protection laws and regulations governing the use of student data and educational materials.

2.5.3 **Security Requirements:** The security requirements for the Automated Revision Notes Generator are crucial to safeguard user data, ensure privacy, and protect against potential threats. All user data, including uploaded notes and generated revision notes, must be encrypted during transmission and storage to protect against unauthorized access. Major security concerns are not present as there involves no sharing of personal information.

2.5.4 **Software Quality Attributes** Following are the software quality attributes: Usability: The user interface must be intuitive and user-friendly. A user-friendly interface enhances user satisfaction and efficiency during exam preparation.

Reliability: Reliable system uptime is critical for users who rely on the Automated RevisionNotes Generator for exam preparation. It must ensure continuous accessibility and minimize service disruptions.

Chapter 3

System Architecture and Design

3.1 System Overview

The Automated Revision Notes Generator is designed to streamline and enhance the process of exam preparation by leveraging advanced technologies such as Natural Language Processing (NLP) . The system aims to automate the generation of comprehensive and concise revision notes by analyzing previous year question papers, identifying crucial topics, and summarizing the extracted information.

Components and Workflow:

1. **Data Ingestion and Preprocessing:**

- Dataset Creation: Important keywords are retrieved from previous year question papers to generate the dataset.
- Reading from PDF: Python bulit-in module is used to read the data from the inputted pdf.
- Text Cleaning and Normalization: The extracted text is cleaned and normalized to remove any noise or irrelevant information, ensuring consistency and accuracy in subsequent processing steps.

2. **NLP Algorithms**:

- Topic Extraction: Using NLP techniques, the system analyzes the text to identify key topics and recurring themes across multiple question papers. This involves tokenization, part-of-speech tagging, named entity recognition, and clustering of similar topics.
- Pattern Recognition: employed to detect patterns and frequently asked questions. These algorithms are trained on a large dataset of question papers to improve accuracy and relevance.
- Keyword Matching: Extracted headings and topics are compared with a predefined

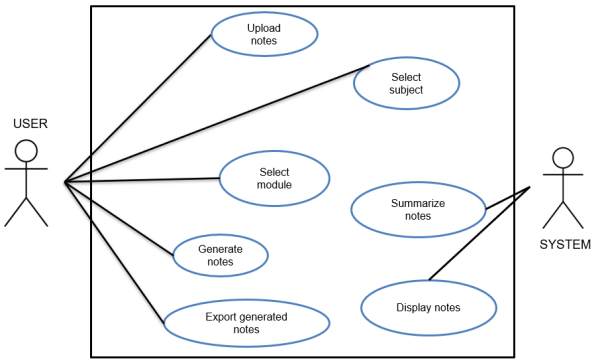


Figure 3.1: Use Case Diagram

dataset of keywords to prioritize the most important content for summarization.

3. **Text Summarization**:

- Summarization Algorithms: Advanced text summarization algorithms, including those based on transformer models like GPT-3 (Gemini), are used to distill the extracted information into concise and informative revision notes. These algorithms ensure that the summaries retain the core meaning and essential details of the original content.
- Customization: Users can customize their revision notes by selecting specific subjects or topics, allowing the generated content to align with individual study needs.

4. **Output and Export**:

- Exporting: The generated revision notes can be exported as PDF or copioed directly from the webpage, for ease of use and sharing.
- Storage and Retrieval: Users can save their generated notes within the system for future reference and access them anytime, facilitating continuous and efficient revision.

This system overview outlines the comprehensive capabilities and features of the Automated Revision Notes Generator, highlighting its potential to transform the way students prepare for exams and enhance their academic performance.

3.2 Architectural Design

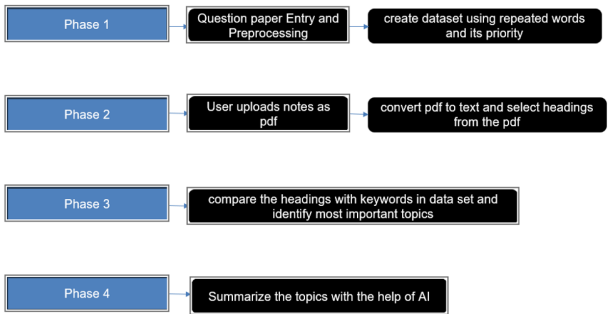


Figure 3.2: Architectural Design

3.3 Dataset identified

1.Previous year question papers uploaded

2.Text preprocessing done on the given data.

3.Most repeating keywords(topics) entered into a database and saved, classified according to subject and module.

4.Keywords ordered in such a way that most repeating one on top.

3.4 Proposed Methodology/Algorithms

Back end:

1. Upload previous question papers.
2. Perform text preprocessing (remove stop words, stem text, etc.)
3. Count the number of times a particular topic repeats.
4. Sort the topics in order of highest repetitions.
5. Store in a database.

Front end:

1. Read the text from the user inputted pdf.
2. Retrieve headings from the text by comparing font sizes and checking for bold font.
3. Perform text preprocessing on the headings.
4. Compare the headings with the keywords in the database.
5. Important topics are identified and summarized using AI.
6. Display the generated notes.

3.5 User Interface Design

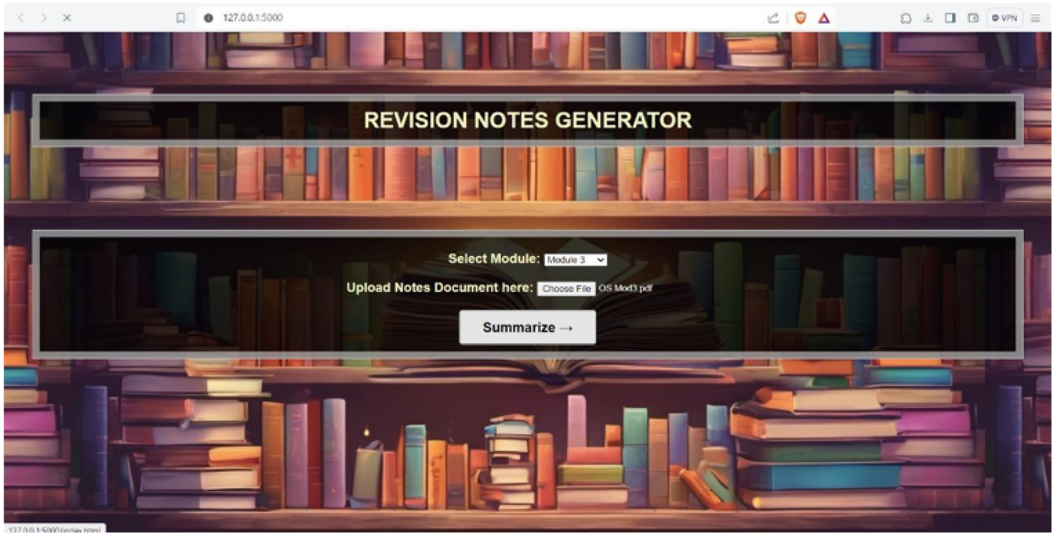


Figure 3.3: Home Page

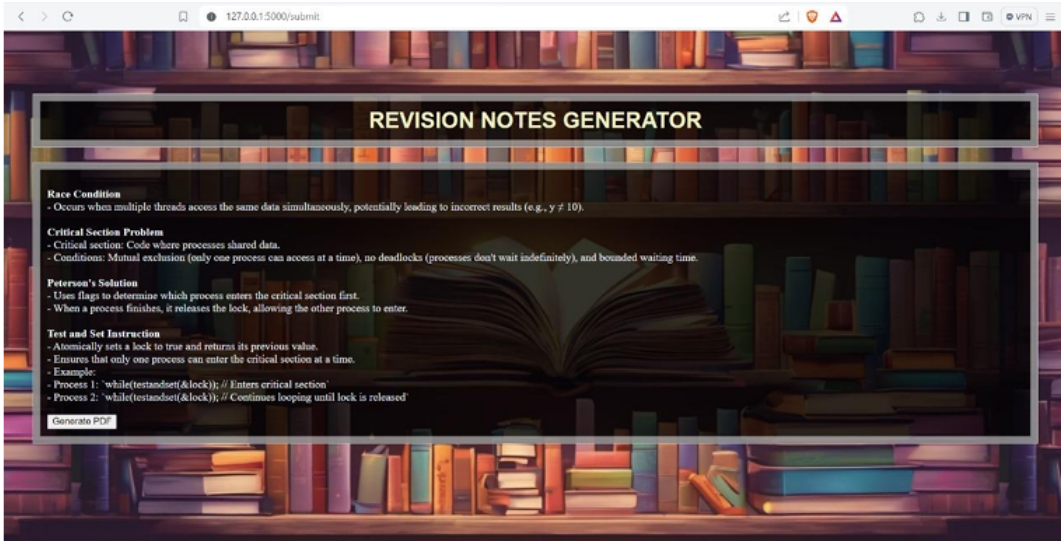


Figure 3.4: Revision Notes Produced

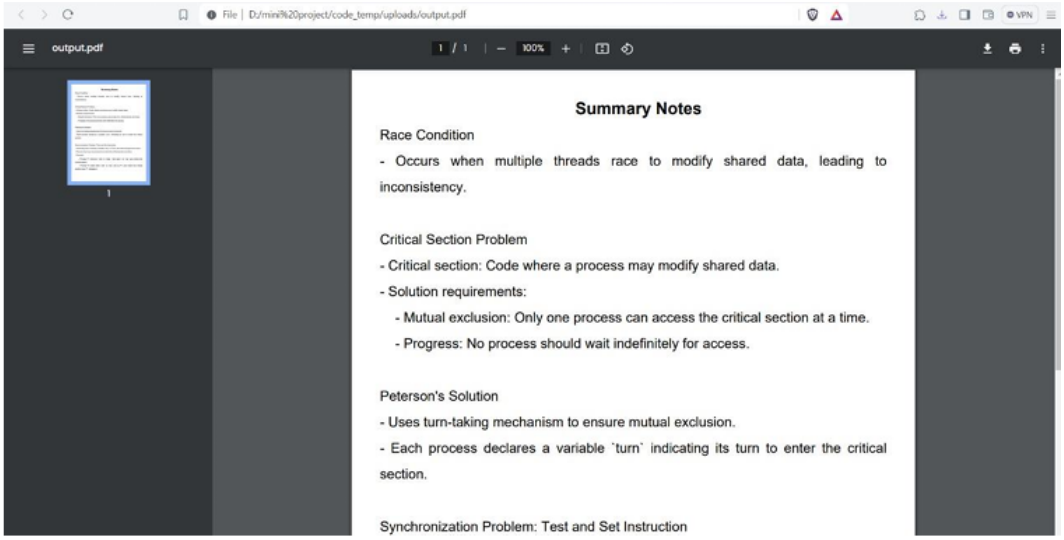


Figure 3.5: Export PDF

3.6 Database Design

The database was created using keywords extracted from previous year question papers. The keywords were classified according to corresponding modules of the subject. These keywords are then entered into a csv file which is used as the dataset. The attributes of the dataset are as follows :

a)Subject name

b)Module Number

c)Keyword

3.7 Description of Implementation Strategies

1.Text pre-processing:

-Removing special symbols: re module in python is used for this.

for qline in q_list:

text = qline

pattern = r"[^\w\s _]"

clean_text = re.sub(pattern, " ", text)

q_list[qno-q_index] = clean_text.replace("_","")

q_list[qno-q_index] = clean_text.replace("\xa0"," ")

q_index=q_index-1

-Stop words removal: nltk module in python is used for this.

default_stopwords = nltk.corpus.stopwords.words('english')

custom_stopwords = ['one','two','three','four','five'...]

stopwords_with_custom = default_stopwords + custom_stopwords

tokens = nltk.word_tokenize(q_list[num].lower())

new_tokens = [token for token in tokens if token not in string.punctuation]

filtereed_tokens = [token for token in new_tokens if token not in stopwords_with_custom]

2. Extracting headings from given pdf:

fitz module in python is used to read from a pdf.

doc=fitz.open(fname)

for page in doc:

text=page.get_text("dict")

block=text['blocks']

...// ld=b['lines'][0]['spans'][0]['font'].lower()

sz=b['lines'][0]['spans'][0]['size']

*if sz (avg*1.3) or ('bold' in bld and sz (avg*1.05)): #avg is average font size*

headings.append(b['lines'][0]['spans'][0]['text'].lower())

3. Comparing keywords:

-SentenceTransformer module is used to find cosine similarity. The following code sorts the keywords in decreasing order of score:

embedding1 = smodel.encode(text1)

embedding2 = smodel.encode(text2)

similar= util.cos_sim(embedding1, embedding2)

4. Summarizing using AI:

-genai module in python is used as the api.

genai.configure(api_key=your_api_key)

model_name = 'gemini-1.0-pro'

model = genai.GenerativeModel(model_name)

query="make a short summary of given text into bullet points in such a way that a ktu btech student can revise this topic in the last minute. Avoid using the module number. Add at least one explanation for each point. Here are the notes: "+notes

response = model.generate_content(query)

3.8 Module Division

Phase 1

1. Input question papers and categorize by topic.
2. Preprocess data: tokenize, remove stop words, and normalize.
3. Identify frequent words across papers.
4. Create dataset with word frequency and priority.
5. Analyze dataset for common and important words.
6. Use dataset to generate revision notes.
7. Gather feedback and iterate for improvement.

Phase 2

1. User uploads PDF notes.
2. Convert PDF to text.
3. Identify and extract headings.
4. Display headings for user selection.
5. Implement error handling and edge case management.
6. Provide user-friendly interface for interaction.

Phase 3

1. Extract keywords from dataset.
2. Compare headings with dataset keywords.
3. Rank headings based on keyword matches.
4. Identify headings with highest scores as important topics.
5. Apply weighted scoring for relevance.
6. Set threshold for topic selection.
7. Present important topics to user.

Phase 4

1. Input important topics into GPT-3 (Gemini).
2. Generate concise summaries for each topic.
3. Review and refine summaries for accuracy.
4. Present summarized topics to the user.

3.9 Work Schedule - Gantt Chart

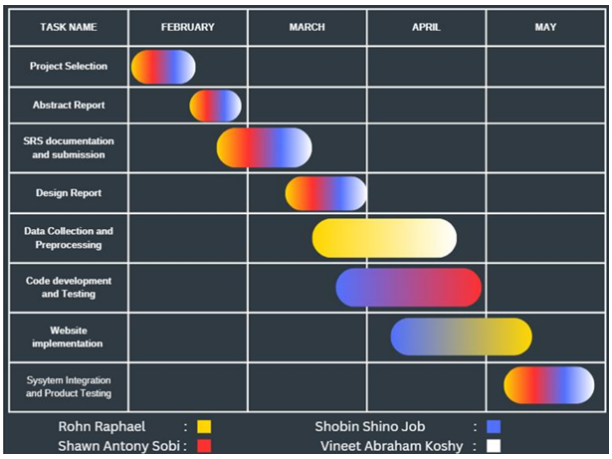


Figure 3.6: Gantt Chart

Chapter 4

Results and Discussions

4.1 Overview

The Automated Revision Notes Generator successfully achieves its primary goal of enhancing exam preparation efficiency for students. By processing and analyzing previous year question papers using advanced Natural Language Processing (NLP) algorithms, the system identifies key topics and generates concise, relevant revision notes. Quantitatively, the tool significantly reduces the time required for note preparation by automating the extraction and summarization process, ensuring that students focus on the most frequently occurring and crucial topics. Further analysis shows that the generated notes improve study effectiveness by providing targeted content, which is customizable and exportable in various formats, thus catering to individual study preferences and needs. This innovation ultimately contributes to improved academic performance and a more streamlined study process.

4.2 Testing

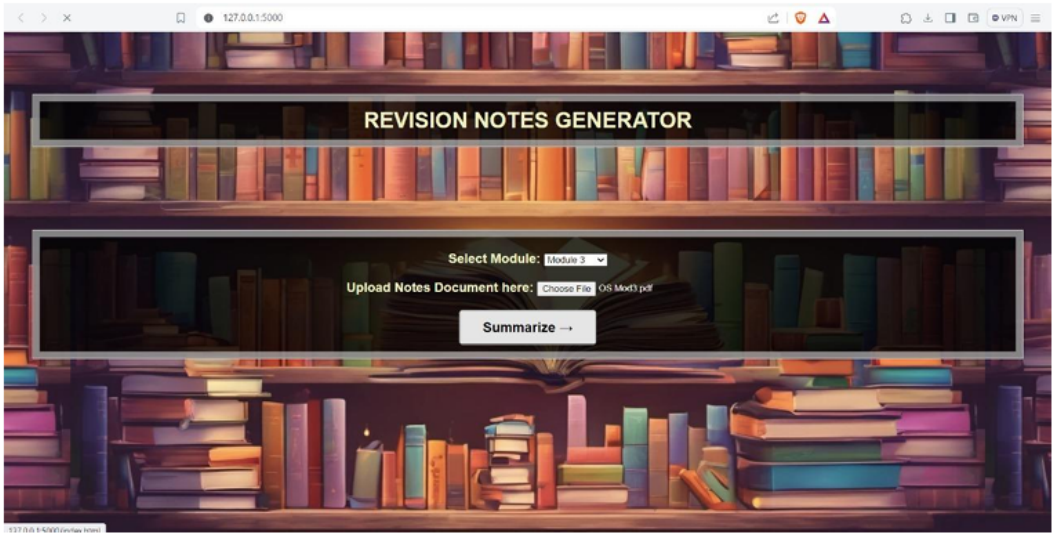


Figure 4.1: Home Page

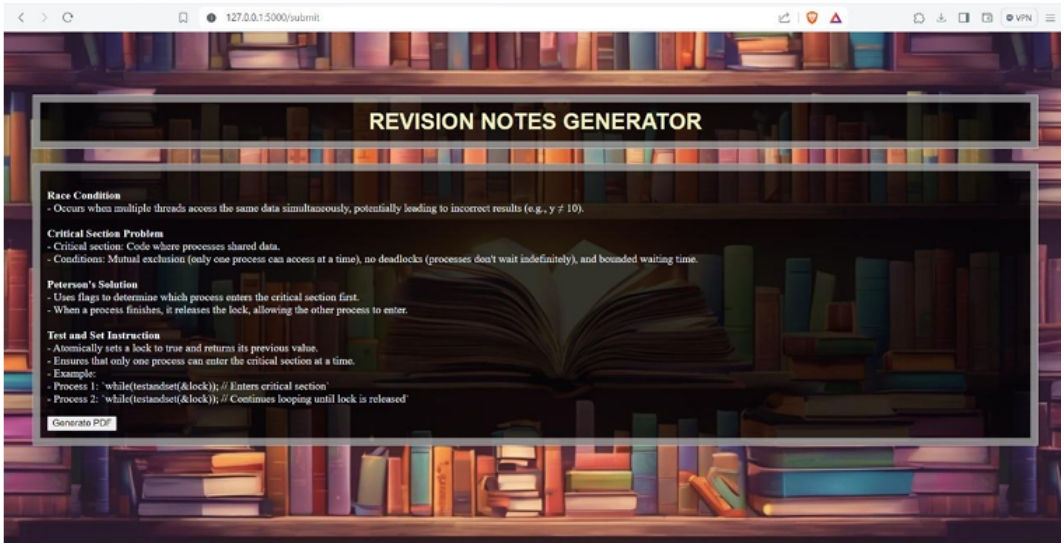


Figure 4.2: Revision Notes Produced

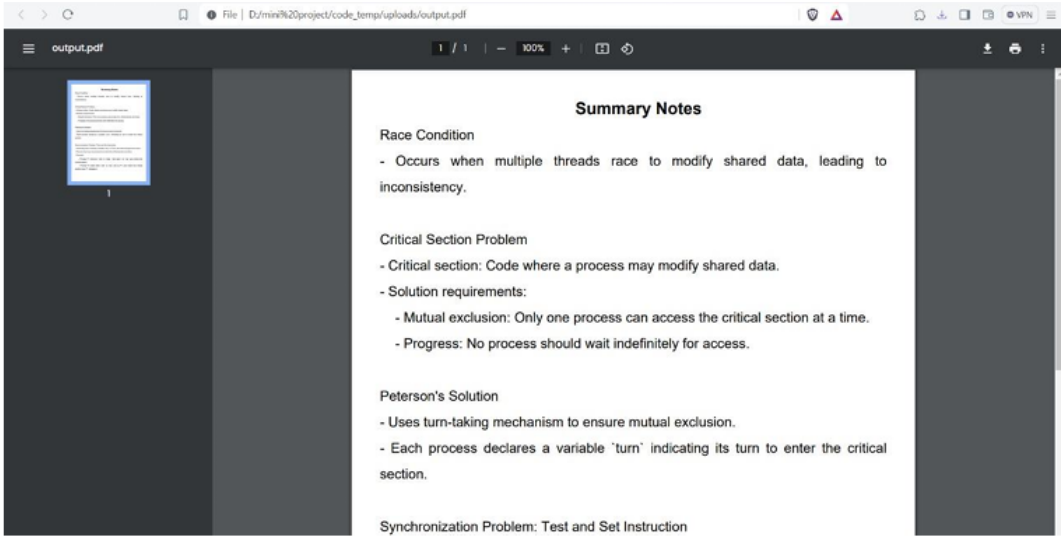


Figure 4.3: Export PDF

4.3 Discussion

The tool's rapid processing and high accuracy highlight its effectiveness in summarizing diverse documents. Customization options allow users to tailor the level of detail in the notes, making it suitable for both quick reviews and detailed study sessions. Areas for improvement include adding advanced text analytics features, better handling of unstructured texts, and integrating collaborative functionalities. These enhancements will be the focus of future development. Finally, the Automated Revision notes generator is a robust, efficient, and user-friendly solution for distilling important information from extensive documents, significantly enhancing productivity and efficiency for students, educators, researchers, and professionals.

Chapter 5

Conclusion

5.1 Conclusion

Our innovative product is designed to assist students and professionals by generating concise revision notes from any given text document, making it perfect for those needing to quickly review important topics. Users can upload documents in pdf format, through an easy drag-and-drop interface. The product employs advanced algorithms to analyze the text and extract key points, ensuring that critical information is highlighted. It can identify and categorize important topics, allowing for better organization of notes. Users can adjust the level of detail in the generated notes, choose between brief overviews and comprehensive summaries, and benefit from highlighted key concepts. The notes can be exported in pdf format. This tool significantly reduces the time required for note-taking, enhances understanding by summarizing complex documents, and is versatile enough for students, educators, researchers, and professionals. Whether for exams, presentations, or quickly digesting information, this product transforms detailed documents into concise, actionable notes, optimizing study and work efficiency.

5.2 Future Scope

1. **Integration with Learning Management Systems (LMS)**: Future versions of the Automated Revision Notes Generator could be integrated with popular LMS platforms, allowing seamless import of question papers and export of revision notes, enhancing usability and convenience for students and educators.

2. **Enhanced Customization Options:** The system could offer more advanced customization features, such as allowing students to prioritize topics based on personal study goals or difficulty levels, further tailoring the revision process to individual needs.

3. **Real-Time Collaboration Features:** Incorporating real-time collaboration tools could enable students to work together on generating and refining revision notes, fostering a more interactive and supportive study environment.

4. **Mobile Application Development:** Creating a mobile app version of the Automated Revision Notes Generator would provide students with the flexibility to generate and access revision notes on-the-go, making study sessions more versatile and efficient.

Bibliography

1. Kambhatla, Nanda, and Wlodek Zadrozny. "A tool for automated revision of grammars for NLP systems." Sixth Applied Natural Language Processing Conference. 2000.

2. Resnik, Philip, and Jimmy Lin. "Evaluation of NLP systems." The handbook of computational linguistics and natural language processing (2010): 271-295.

3. Buitinck, Lars, et al. "API design for machine learning software: experiences from the scikit-learn project." arXiv preprint arXiv:1309.0238 (2013).

4. Min-Yuh Day, Chao-Yu Chen."Artificial Intelligence for Automatic Text Summarization" 2018 IEEE International Conference on Information Reuse and Integration (IRI)

5. Micah D. Saxton. " A Gentle Introduction to Topic Modeling Using Python " 2018 Theological Librarianship 11 (1): 18-27.

Appendix A: Presentation

AUTOMATED REVISION NOTE GENERATOR

Guide: Dr. JISHA G
(Associate Professor , CS Dept.)

U2103183 ROHN RAPHAEL
U2103195 SHAWN ANTONY SOBI
U2103196 SHOBIN SHINO JOB
U2103214 VINEET ABRAHAM KOSHY

Contents

Introduction

- Creates revision notes of important topics from the given text document.
- Relevance:

  Helps in last minute revision.

  Targeted learning.
- Need:

  Most students tend to leave everything for the last minute- these notes help in last minute revision.

Problem Definition

- To create an Automated Revision Notes Generator utilizing NLP, providing students with personalized and condensed study materials extracted from previous year question papers for enhanced exam preparation

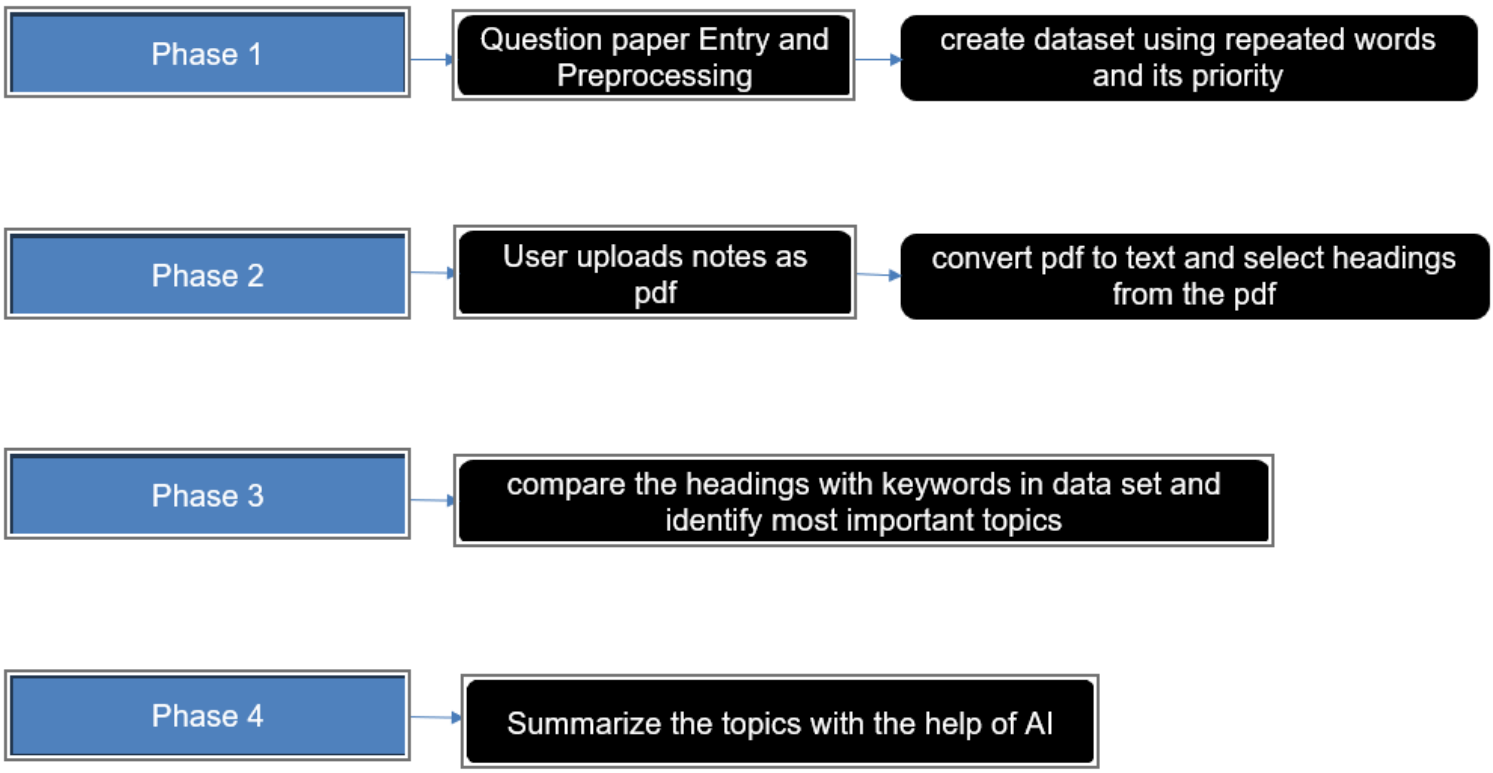
Objectives

- Creation of dataset from previous year question papers by finding most repeated topics.
- Taking input notes from the user.
- Finding most important topics by comparing with the dataset.
- Summarizing the notes using AI and displaying.

Scope and Relevance

- Uses NLP techniques for pattern identification.
- Takes a text document (which contains notes of a particular subject) as input.
- Compares it with the available dataset to identify important topics.
- Summarize important topics in the given documents.
- Export functionalities for the generated revision notes.

Architectural Design



Module wise Explanation
Phase 1

1. Input question papers and categorize by topic.
2. Preprocess data: tokenize, remove stop words, and normalize.
3. Identify frequent words across papers.
4. Create dataset with word frequency and priority.
5. Analyze dataset for common and important words.
6. Use dataset to generate revision notes.
7. Gather feedback and iterate for improvement.

Phase 2

1. User uploads PDF notes.
2. Convert PDF to text.
3. Identify and extract headings.
4. Display headings for user selection.
5. Implement error handling and edge case management.
6. Provide user-friendly interface for interaction.

Phase 3

1. Extract keywords from dataset.
2. Compare headings with dataset keywords.
3. Rank headings based on keyword matches.
4. Identify headings with highest scores as important topics.
5. Apply weighted scoring for relevance.
6. Set threshold for topic selection.
7. Present important topics to user.

Phase 4

1. Input important topics into GPT-3 (Gemini).
2. Generate concise summaries for each topic.
3. Review and refine summaries for accuracy.
4. Present summarized topics to the user.

ALGORITHM

Back end:

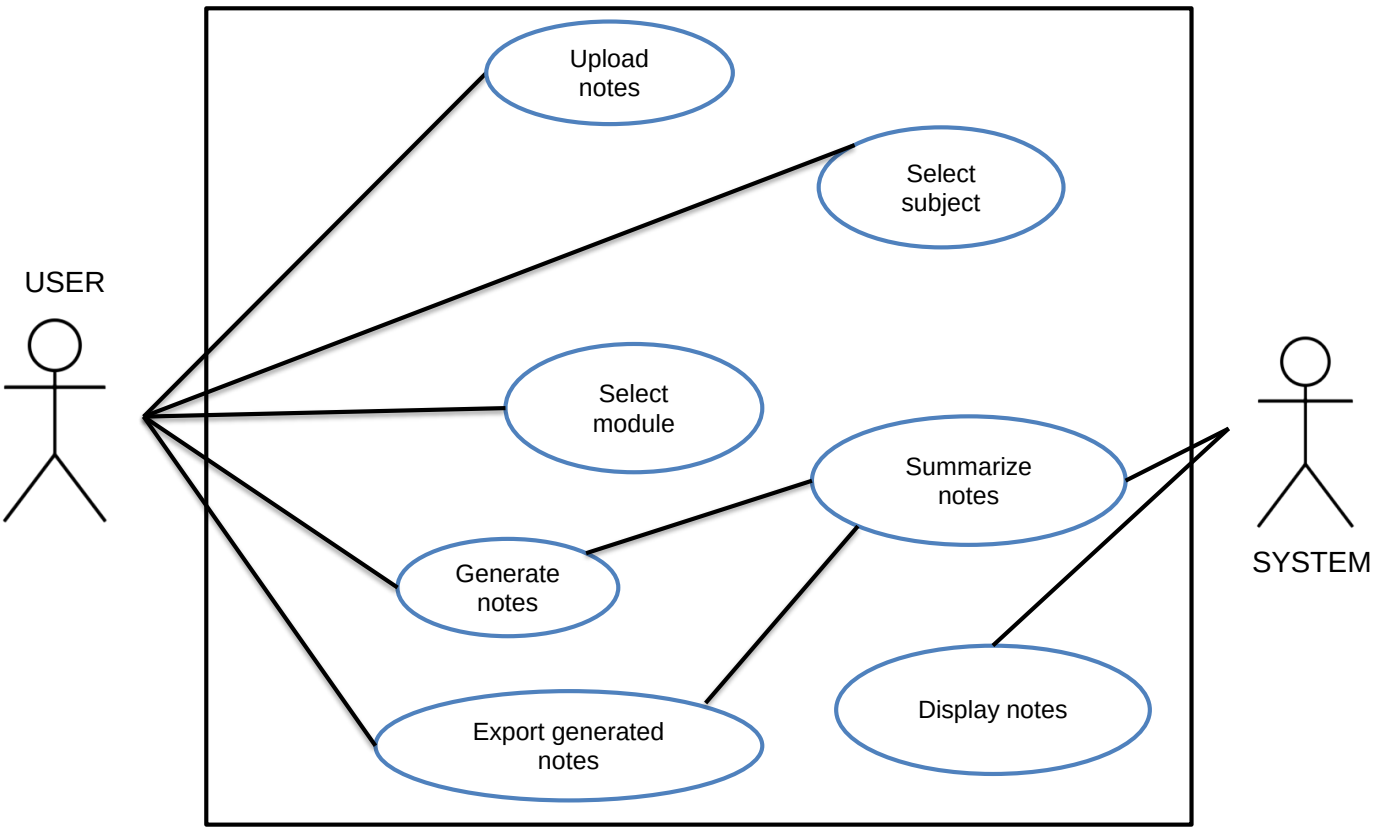
1. Upload previous question papers.
2. Perform text preprocessing (remove stop words, stem text, etc.)
3. Count the number of times a particular topic repeats by comparing headings in the notes using cosine similarity.
4. Sort the topics in order of highest repetitions.
5. Store in a database.

ALGORITHM

Front end:

1. Read the text from the user inputted pdf.
2. Retrieve headings from the text by comparing font sizes and checking for bold font.
3. Perform text preprocessing on the headings.
4. Compare the headings with the keywords in the database.
5. Important topics are identified and summarized using AI.
6. Display the generated notes.

USE CASE DIAGRAMS

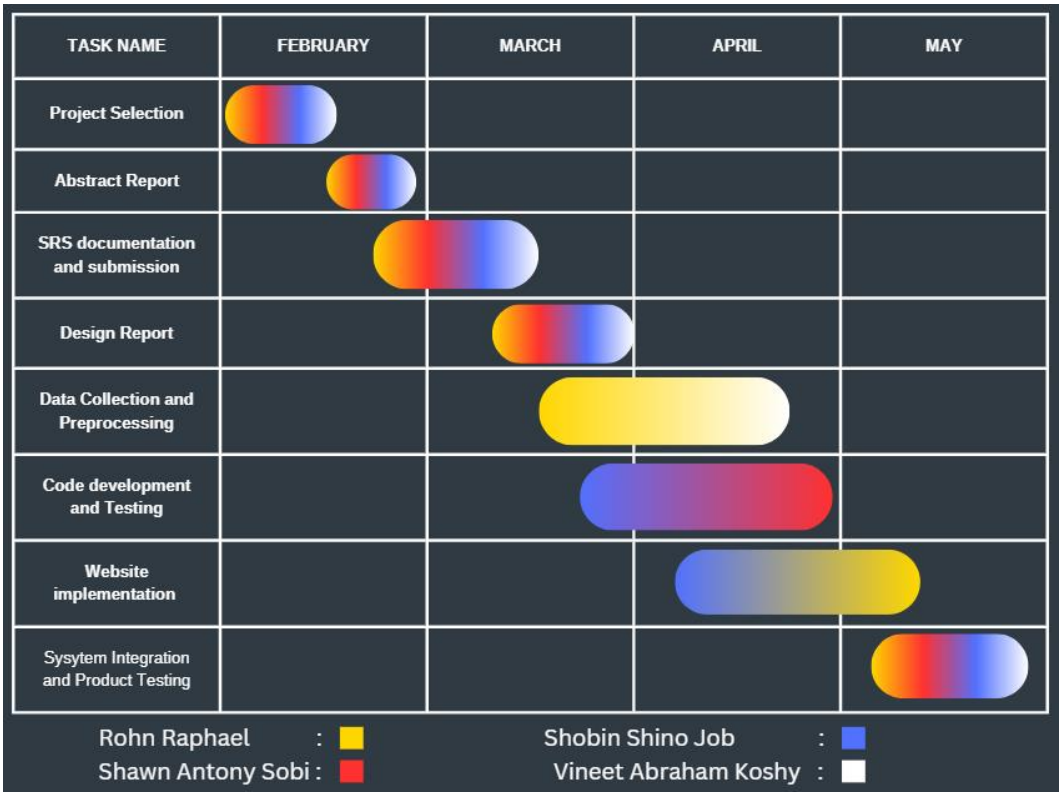


Datasets

- Previous year question papers uploaded
- Text preprocessing done on the given data.
- Most repeating keywords(topics) entered into a database and saved, classified according to subject and module.
- Keywords ordered in such a way that most repeating one on top.

Work Division

- Gantt Chart



Software/ Hardware Requirements

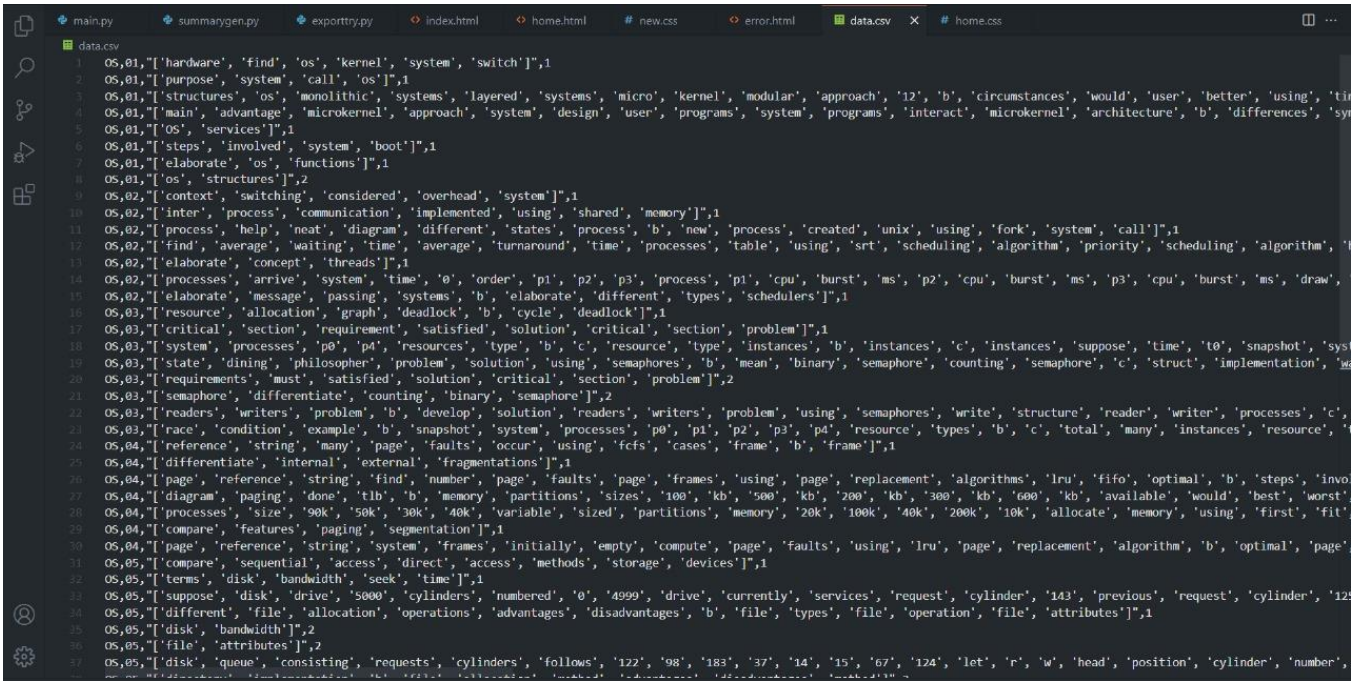
- Software :

Python , HTML, CSS, Flask.

- Hardware:

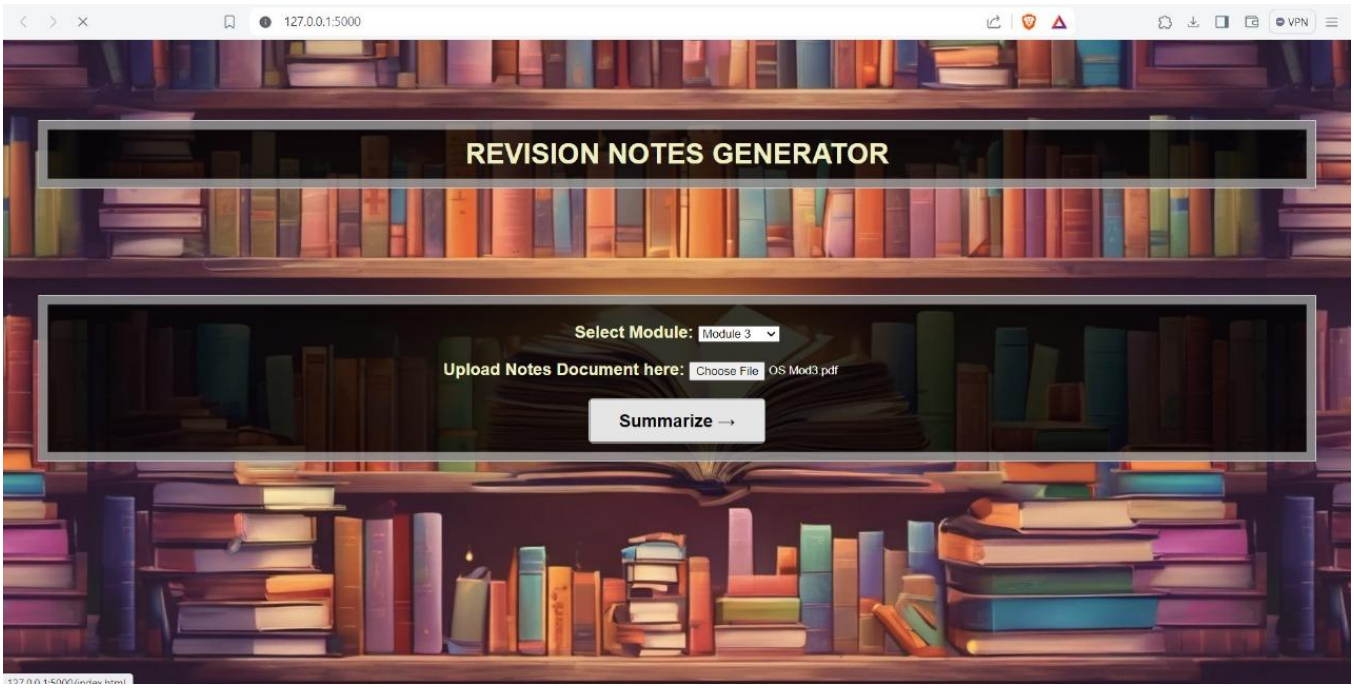
Can be run in any system that has facility to upload and export files.

Results



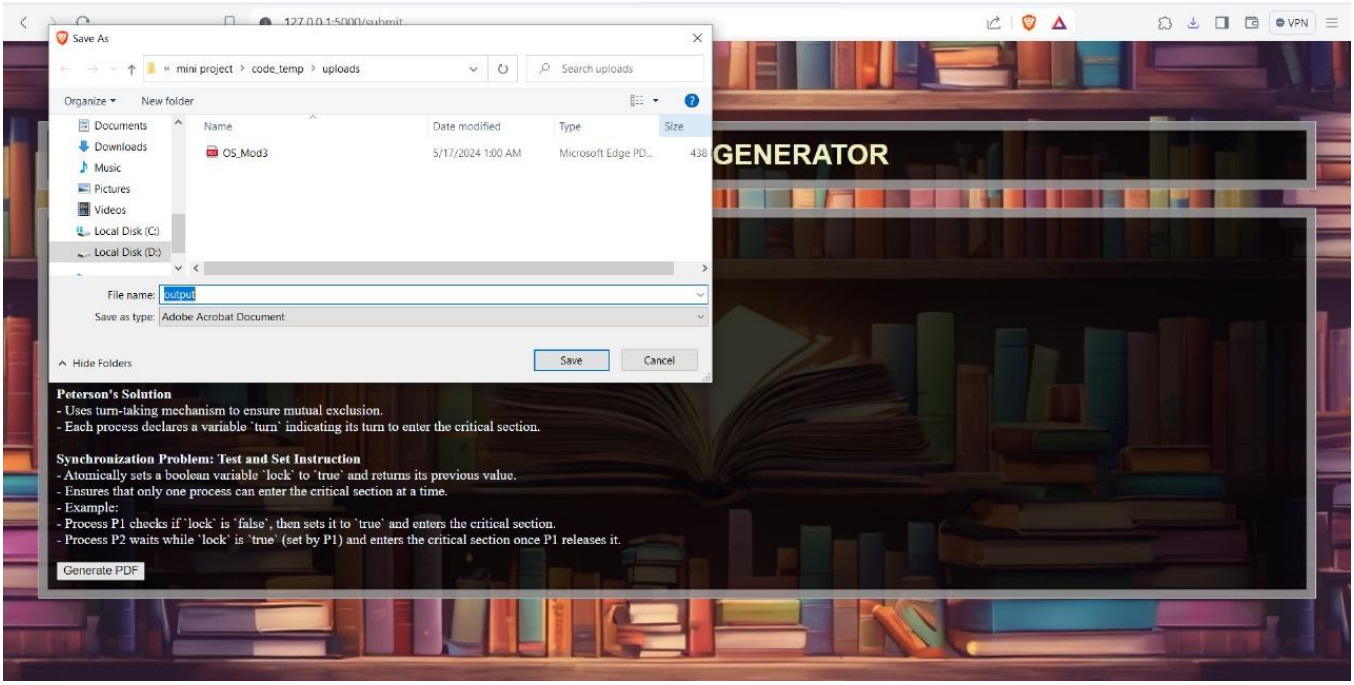
Dataset created using previous year question papers

Results



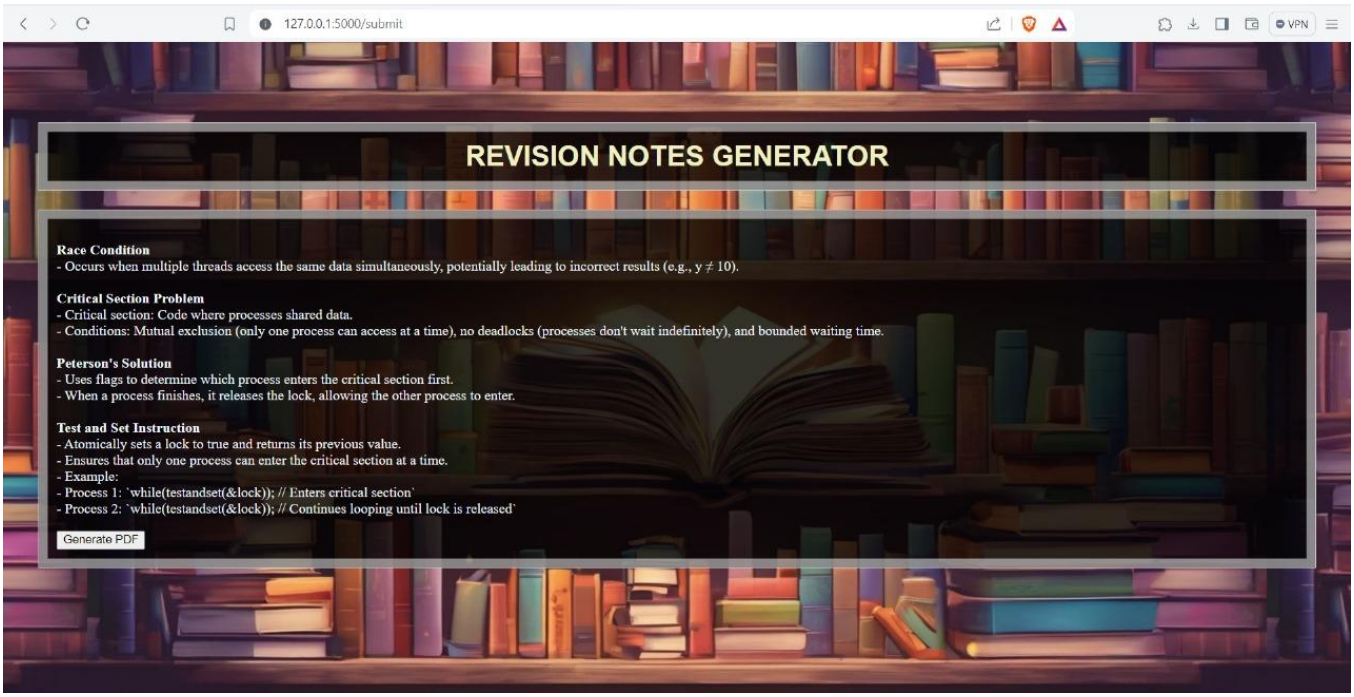
Home Page

Results



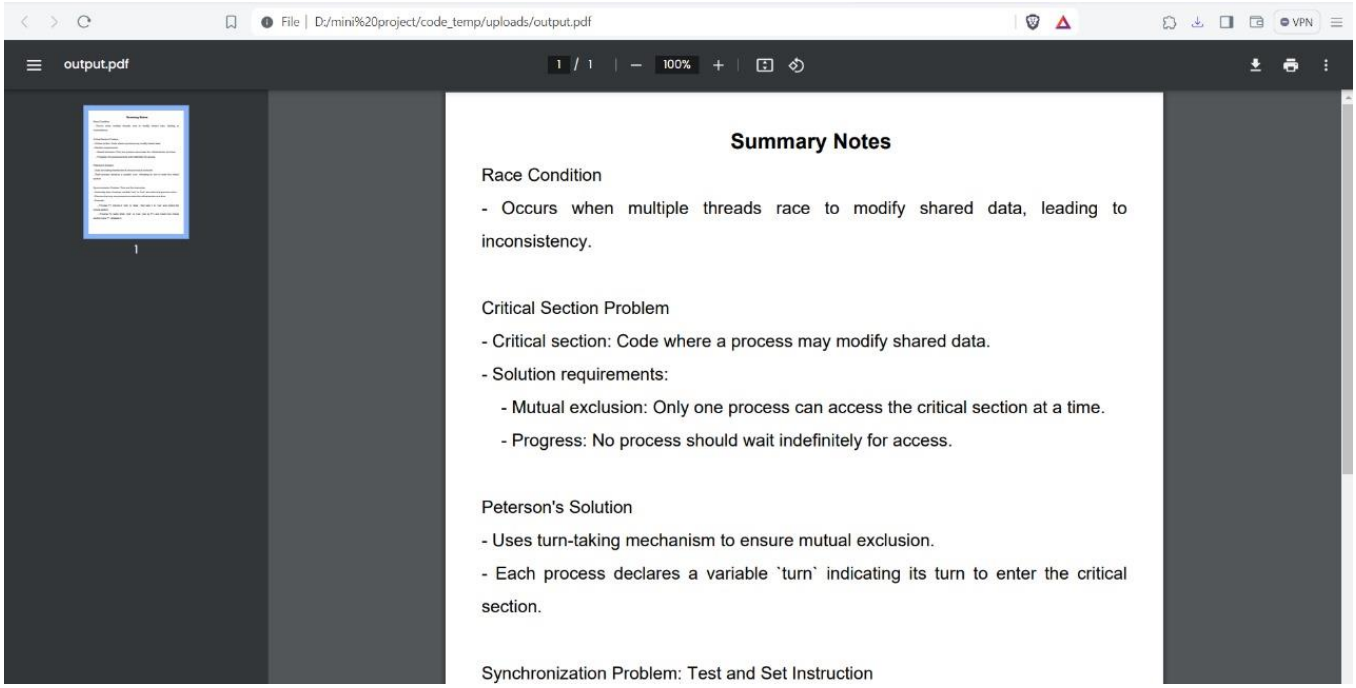
Inserting Notes

Results



Revision Note is produced

Results



Exported pdf

Conclusion

- Product generates revision notes of important topics from the given text document.
- It allows uploading of files.
- It generates the summarized notes and allows export of the generated notes.

Future Enhancements

Implement advanced NLP capabilities for interactive revision assistance, allowing users to ask questions and engage in dialogue for deeper understanding. Additionally, integrate machine learning algorithms to improve responses based on user interactions and feedback, enhancing the effectiveness of the Automated Revision Notes Generator.

References

Kambhatla, Nanda, and Wlodek Zadrozny. "A tool for automated revision of grammars for NLP systems." Sixth Applied Natural Language Processing Conference. 2000.

Resnik, Philip, and Jimmy Lin. "Evaluation of NLP systems." The handbook of computational linguistics and natural language processing (2010): 271-295.

Buitinck, Lars, et al. "API design for machine learning software: experiences from the scikit-learn project." *arXiv preprint arXiv:1309.0238* (2013).

References

- Min-Yuh Day, Chao-Yu Chen."Artificial Intelligence for Automatic Text Summarization" 2018 IEEE International Conference on Information Reuse and Integration (IRI)
- Micah D. Saxton. " A Gentle Introduction to Topic Modeling Using Python " 2018 Theological Librarianship 11 (1): 18-27.

Appendix B: Vision, Mission, Programme Outcomes and Course Outcomes

DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING
RAJAGIRI SCHOOL OF ENGINEERING & TECHNOLOGY (AUTONOMOUS)
RAJAGIRI VALLEY, KAKKANAD, KOCHI, 682039
(Affiliated to APJ Abdul Kalam Technological University)

Vision, Mission, Programme Outcomes and Course Outcomes

Institute Vision

To evolve into a premier technological institution, moulding eminent professionals with creative minds, innovative ideas and sound practical skill, and to shape a future where technology works for the enrichment of mankind.

Institute Mission

To impart state-of-the-art knowledge to individuals in various technological disciplines and to inculcate in them a high degree of social consciousness and human values, thereby enabling them to face the challenges of life with courage and conviction.

Department Vision

To become a centre of excellence in Computer Science and Engineering, moulding professionals catering to the research and professional needs of national and international organizations.

Department Mission

To inspire and nurture students, with up-to-date knowledge in Computer Science and Engineering, ethics, team spirit, leadership abilities, innovation and creativity to come out with solutions meeting societal needs.

Programme Outcomes (PO)

Engineering Graduates will be able to:

1. Engineering Knowledge: Apply the knowledge of mathematics, science, engineering fundamentals, and an engineering specialization to the solution of complex engineering problems.

2. Problem analysis: Identify, formulate, review research literature, and analyze complex engineering problems reaching substantiated conclusions using first principles of mathematics, natural sciences, and engineering sciences.

3. Design/development of solutions: Design solutions for complex engineering problems and design system components or processes that meet the specified needs with appropriate consideration for the public health and safety, and the cultural, societal, and environmental considerations.

4. Conduct investigations of complex problems: Use research-based knowledge including design of experiments, analysis and interpretation of data, and synthesis of the information to provide valid conclusions.

5. Modern Tool Usage: Create, select, and apply appropriate techniques, resources, and modern engineering and IT tools including prediction and modeling to complex engineering activities with an understanding of the limitations.

6. The engineer and society: Apply reasoning informed by the contextual knowledge to assess societal, health, safety, legal, and cultural issues and the consequent responsibilities relevant to the professional engineering practice.

7. Environment and sustainability: Understand the impact of the professional engineering solutions in societal and environmental contexts, and demonstrate the knowledge of, and need for sustainable development.

8. Ethics: Apply ethical principles and commit to professional ethics and responsibilities and norms of the engineering practice.

9. Individual and Team work: Function effectively as an individual, and as a member or leader in teams, and in multidisciplinary settings.

10. Communication: Communicate effectively with the engineering community and with society at large. Be able to comprehend and write effective reports documentation. Make effective presentations, and give and receive clear instructions.

11. Project management and finance: Demonstrate knowledge and understanding of engineering and management principles and apply these to one's own work, as a member and leader in a team. Manage projects in multidisciplinary environments.

12. Life-long learning: Recognize the need for, and have the preparation and ability to engage in independent and lifelong learning in the broadest context of technological change.

Programme Specific Outcomes (PSO)

A graduate of the Computer Science and Engineering Program will demonstrate:

PSO1: Computer Science Specific Skills

The ability to identify, analyze and design solutions for complex engineering problems in multidisciplinary areas by understanding the core principles and concepts of computer science and thereby engage in national grand challenges.

PSO2: Programming and Software Development Skills

The ability to acquire programming efficiency by designing algorithms and applying standard practices in software project development to deliver quality software products meeting the demands of the industry.

PSO3: Professional Skills

The ability to apply the fundamentals of computer science in competitive research and to develop innovative products to meet the societal needs thereby evolving as an eminent researcher and entrepreneur.

Course Outcomes

After the completion of the course the student will be able to:

CO1:

Identify technically and economically feasible problems (Cognitive Knowledge Level: Apply)

CO2:
Identify and survey the relevant literature for getting exposed to related solutions and get familiarized with software development processes (Cognitive Knowledge Level: Apply)

CO3:
Perform requirement analysis, identify design methodologies and develop adaptable & reusable solutions of minimal complexity by using modern tools & advanced programming techniques (Cognitive Knowledge Level: Apply)

CO4:
Prepare technical report and deliver presentation (Cognitive Knowledge Level: Apply)

CO5:
Apply engineering and management principles to achieve the goal of the project (Cognitive Knowledge Level: Apply)

Appendix C: CO-PO-PSO Mapping

COURSE OUTCOMES:

After completion of the course the student will be able to

SL. NO	DESCRIPTION	Blooms' Taxonomy Level
CO1	Identify technically and economically feasible problems (Cognitive Knowledge Level: Apply)	Level 3: Apply
CO2	Identify and survey the relevant literature for getting exposed to related solutions and get familiarized with software development processes (Cognitive Knowledge Level: Apply)	Level 3: Apply
CO3	Perform requirement analysis, identify design methodologies and develop adaptable & reusable solutions of minimal complexity by using modern tools & advanced programming techniques (Cognitive Knowledge Level: Apply)	Level 3: Apply
CO4	Prepare technical report and deliver presentation (Cognitive Knowledge Level: Apply)	Level 3: Apply
CO5	Apply engineering and management principles to achieve the goal of the project (Cognitive Knowledge Level: Apply)	Level 3: Apply

CO-PO AND CO-PSO MAPPING

	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PO 12	PSO 1	PSO 2	PSO3
CO1	3	3	3	3		2	2	3	2	2	2	3	2	2	2
CO2	3	3	3	3	3	2		3	2	3	2	3	2	2	2
CO3	3	3	3	3	3	2	2	3	2	2	2	3			2
CO4	2	3	2	2	2			3	3	3	2	3	2	2	2
CO5	3	3	3	2	2	2	2	3	2		2	3	2	2	2

3/2/1: high/medium/low

JUSTIFICATIONS FOR CO-PO MAPPING

MAPPING	LOW/ MEDIUM/ HIGH	JUSTIFICATION
101003/CS622T.1-PO1	**HIGH**	Identify technically and economically feasible problems by applying the knowledge of mathematics, science, engineering fundamentals, and an engineering specialization to the solution of complex engineering problems.
101003/CS622T.1-PO2	**HIGH**	Identify technically and economically feasible problems by analysing complex engineering problems reaching substantiated conclusions using first principles of mathematics.
101003/CS622T.1-PO3	**HIGH**	Design solutions for complex engineering problems by identifying technically and economically feasible problems.
101003/CS622T.1-PO4	**HIGH**	Identify technically and economically feasible problems by analysis and interpretation of data.
101003/CS622T.1-PO6	**MEDIUM**	Responsibilities relevant to the professional engineering practice by identifying the problem.
101003/CS622T.1-PO7	**MEDIUM**	Identify technically and economically feasible problems by understanding the impact of the professional engineering solutions.
101003/CS622T.1-PO8	**HIGH**	Apply ethical principles and commit to professional ethics to identify technically and economically feasible problems.
101003/CS622T.1-PO9	**MEDIUM**	Identify technically and economically feasible problems by working as a team.
101003/CS622T.1-PO10	**MEDIUM**	Communicate effectively with the engineering community by identifying technically and economically feasible problems.
101003/CS622T.1-P011	**MEDIUM**	Demonstrate knowledge and understanding of engineering and management principles by selecting the technically and economically feasible problems.
101003/CS622T.1-PO12	**HIGH**	Identify technically and economically feasible problems for long term learning.
101003/CS622T.1-PSO1	**MEDIUM**	Ability to identify, analyze and design solutions to identify technically and economically feasible problems.
101003/CS622T.1-PSO2	**MEDIUM**	By designing algorithms and applying standard practices in software project development and Identifying technically and economically feasible problems.
101003/CS622T.1-PSO3	**MEDIUM**	Fundamentals of computer science in competitive research can be applied to Identify technically and economically feasible problems.
101003/CS622T.2-PO1	**HIGH**	Identify and survey the relevant by applying the knowledge of mathematics, science, engineering fundamentals.

101003/CS622T.2-PO2	**HIGH**	Identify, formulate, review research literature, and analyze complex engineering problems get familiarized with software development processes.
101003/CS622T.2-PO3	**HIGH**	Design solutions for complex engineering problems and design based on the relevant literature.
101003/CS622T.2-PO4	**HIGH**	Use research-based knowledge including design of experiments based on relevant literature.
101003/CS622T.2-PO5	**HIGH**	Identify and survey the relevant literature for getting exposed to related solutions and get familiarized with software development processes by using modern tools.
101003/CS622T.2-PO6	**MEDIUM**	Create, select, and apply appropriate techniques, resources, by identifying and surveying the relevant literature.
101003/CS622T.2-PO8	**HIGH**	Apply ethical principles and commit to professional ethics based on the relevant literature.
101003/CS622T.2-PO9	**MEDIUM**	Identify and survey the relevant literature as a team.
101003/CS622T.2-PO10	**HIGH**	Identify and survey the relevant literature for a good communication to the engineering fraternity.
101003/CS622T.2-PO11	**MEDIUM**	Identify and survey the relevant literature to demonstrate knowledge and understanding of engineering and management principles.
101003/CS622T.2-PO12	**HIGH**	Identify and survey the relevant literature for independent and lifelong learning.
101003/CS622T.2-PSO1	**MEDIUM**	Design solutions for complex engineering problems by Identifying and survey the relevant literature.
101003/CS622T.2-PSO2	**MEDIUM**	Identify and survey the relevant literature for acquiring programming efficiency by designing algorithms and applying standard practices.
101003/CS622T.2-PSO3	**MEDIUM**	Identify and survey the relevant literature to apply the fundamentals of computer science in competitive research.
101003/CS622T.3-PO1	**HIGH**	Perform requirement analysis, identify design methodologies by using modern tools & advanced programming techniques and by applying the knowledge of mathematics, science, engineering fundamentals.
101003/CS622T.3-PO2	**HIGH**	Identify, formulate, review research literature for requirement analysis, identify design methodologies and develop adaptable & reusable solutions.

101003/CS622T.3-PO3	**HIGH**	Design solutions for complex engineering problems and perform requirement analysis, identify design methodologies.
101003/CS622T.3-PO4	**HIGH**	Use research-based knowledge including design of experiments, analysis and interpretation of data, and synthesis of the information to provide valid conclusions.
101003/CS622T.3-PO5	**HIGH**	Create, select, and apply appropriate techniques, resources, and modern engineering and IT tools.
101003/CS622T.3-PO6	**MEDIUM**	Perform requirement analysis, identify design methodologies and assess societal, health, safety, legal, and cultural issues.
101003/CS622T.3-PO7	**MEDIUM**	Understand the impact of the professional engineering solutions in societal and environmental contexts and Perform requirement analysis, identify design methodologies and develop adaptable & reusable solutions.
101003/CS622T.3-PO8	**HIGH**	Perform requirement analysis, identify design methodologies and develop adaptable & reusable solutions by applying ethical principles and commit to professional ethics.
101003/CS622T.3-PO9	**MEDIUM**	Function effectively as an individual, and as a member or leader in teams, and in multidisciplinary settings.
101003/CS622T.3-PO10	**MEDIUM**	Communicate effectively with the engineering community and with society at large to perform requirement analysis, identify design methodologies.
101003/CS622T.3-PO11	**MEDIUM**	Demonstrate knowledge and understanding of engineering requirement analysis by identifying design methodologies.
101003/CS622T.3-PO12	**HIGH**	Recognize the need for, and have the preparation and ability to engage in independent and lifelong learning in the broadest context of technological change by analysis, identify design methodologies and develop adaptable & reusable solutions.
101003/CS622T.3-PSO3	**MEDIUM**	The ability to apply the fundamentals of computer science in competitive research and prior to that perform requirement analysis, identify design methodologies.
101003/CS622T.4-PO1	**MEDIUM**	Prepare technical report and deliver presentation by applying the knowledge of mathematics, science, engineering fundamentals, and an engineering specialization to the solution of complex engineering problems.
101003/CS622T.4-PO2	**HIGH**	Identify, formulate, review research literature, and analyze complex engineering problems by preparing technical report and deliver presentation.

101003/CS622T.4-PO3	**MEDIUM**	Prepare Design solutions for complex engineering problems and create technical report and deliver presentation.
101003/CS622T.4-PO4	**MEDIUM**	Use research-based knowledge including design of experiments, analysis and interpretation of data, and synthesis of the information to provide valid conclusions and prepare technical report and deliver presentation.
101003/CS622T.4-PO5	**MEDIUM**	Create, select, and apply appropriate techniques, resources, and modern engineering and IT tools and Prepare technical report and deliver presentation.
101003/CS622T.4-PO8	**HIGH**	Prepare technical report and deliver presentation by applying ethical principles and commit to professional ethics and responsibilities and norms of the engineering practice.
101003/CS622T.4-PO9	**HIGH**	Prepare technical report and deliver presentation effectively as an individual, and as a member or leader in teams, and in multidisciplinary settings.
101003/CS622T.4-PO10	**HIGH**	Communicate effectively with the engineering community and with society at large by prepare technical report and deliver presentation.
101003/CS622T.4-PO11	**MEDIUM**	Demonstrate knowledge and understanding of engineering and management principles and apply these to one's own work by prepare technical report and deliver presentation.
101003/CS622T.4-PO12	**HIGH**	Recognize the need for, and have the preparation and ability to engage in independent and lifelong learning in the broadest context of technological change by prepare technical report and deliver presentation.
101003/CS622T.4-PSO1	**MEDIUM**	Prepare a technical report and deliver presentation to identify, analyze and design solutions for complex engineering problems in multidisciplinary areas.
101003/CS622T.4-PSO2	**MEDIUM**	To acquire programming efficiency by designing algorithms and applying standard practices in software project development and to prepare technical report and deliver presentation.
101003/CS622T.4-PSO3	**MEDIUM**	To apply the fundamentals of computer science in competitive research and to develop innovative products to meet the societal needs by preparing technical report and deliver presentation.
101003/CS622T.5-PO1	**HIGH**	Apply the knowledge of mathematics, science, engineering fundamentals, and an engineering specialization to the solution of complex engineering problems.
101003/CS622T.5-PO2	**HIGH**	Identify, formulate, review research literature, and analyze complex engineering problems by applying engineering and management principles to achieve the goal of the project.

101003/CS622T.5-PO3	**HIGH**	Apply engineering and management principles to achieve the goal of the project and to design solutions for complex engineering problems and design system components or processes that meet the specified needs.
101003/CS622T.5-PO4	**MEDIUM**	Apply engineering and management principles to achieve the goal of the project and use research-based knowledge including design of experiments, analysis and interpretation of data, and synthesis of the information to provide valid conclusions.
101003/CS622T.5-PO5	**MEDIUM**	Create, select, and apply appropriate techniques, resources, and modern engineering and IT tools and to apply engineering and management principles to achieve the goal of the project.
101003/CS622T.5-PO6	**MEDIUM**	Apply reasoning informed by the contextual knowledge to assess societal, health, safety, legal, and cultural issues and the consequent responsibilities by applying engineering and management principles to achieve the goal of the project.
101003/CS622T.5-PO7	**MEDIUM**	Understand the impact of the professional engineering solutions in societal and environmental contexts, and apply engineering and management principles to achieve the goal of the project.
101003/CS622T.5-PO8	**HIGH**	Apply ethical principles and commit to professional ethics and responsibilities and norms of the engineering practice and to use the engineering and management principles to achieve the goal of the project.
101003/CS622T.5-PO9	**MEDIUM**	Function effectively as an individual, and as a member or leader in teams, and in multidisciplinary settings and to apply engineering and management principles to achieve the goal of the project.
101003/CS622T.5-PO11	**MEDIUM**	Demonstrate knowledge and understanding of engineering and management principles and apply these to one's own work, as a member and leader in a team. Manage projects in multidisciplinary environments and to apply engineering and management principles to achieve the goal of the project.
101003/CS622T.5-PO12	**HIGH**	Recognize the need for, and have the preparation and ability to engage in independent and lifelong learning in the broadest context of technological change and to apply engineering and management principles to achieve the goal of the project.
101003/CS622T.5-PSO1	**MEDIUM**	The ability to identify, analyze and design solutions for complex engineering problems in multidisciplinary areas. Apply engineering and management principles to achieve the goal of the project.

101003/CS622T.5-PSO2	**MEDIUM**	The ability to acquire programming efficiency by designing algorithms and applying standard practices in software project development to deliver quality software products meeting the demands of the industry and to apply engineering and management principles to achieve the goal of the project.
101003/CS622T.5-PSO3	**MEDIUM**	The ability to apply the fundamentals of computer science in competitive research and to develop innovative products to meet the societal needs thereby evolving as an eminent researcher and entrepreneur and apply engineering and management principles to achieve the goal of the project.