RSET
RAJAGIRI SCHOOL OF
ENGINEERING & TECHNOLOGY
(AUTONOMOUS)

Project Report On

QGEN (Automated Question Paper Generation App)

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

Bachelor of Technology

in

Computer Science and Engineering

By

Nevin Aju (U2003151)
Rohan Jose Paul (U2003171)
Ronit John Daniel (U2003174)
Sebin Bejoy (U2003190)

Under the guidance of

Ms. Sangeetha Jamal

Department of Computer Science and Engineering
Rajagiri School of Engineering & Technology (Autonomous)
(Parent University: APJ Abdul Kalam Technological University)
Rajagiri Valley, Kakkanad, Kochi, 682039
April 2024

CERTIFICATE

This is to certify that the project report entitled ***"QGEN (Automated Question Paper Generation App)"*** *is a bonafide record of the work done by* ***Nevin Aju (U2003151)***, ***Rohan Jose Paul (U2003171)***, ***Ronit John Daniel (U2003174)***, ***Sebin Bejoy (U2003190)***, *submitted to the Rajagiri School of Engineering & Technology (RSET) (Autonomous) 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.*

Ms. Sangeetha Jamal
Project Guide
Asst. Professor
Dept. of CSE
RSET

Ms. Anita John
Project Coordinator
Asst. Professor
Dept. of CSE
RSET

Dr. Preetha K.G.
Professor & HoD
Dept. of CSE
RSET

ACKNOWLEDGMENT

We wish to express our sincere gratitude towards **Prof. Dr. P.S. Sreejith**, Principal of RSET, and **Dr.Preetha K.G.**, Professor, Head of the Department of Computer Science for providing us with the opportunity to undertake our project, "QGEN".

We are highly indebted to our project coordinators, **Ms.Anita John**, Assistant Professor, Department of Computer Science and Engineering and **Mr.Sajanraj TD**, Senior Research Fellow, 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 **Ms.Sangeetha Jamal**, Assistant Professor, Department of Computer Science and Engineering 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.

Nevin Aju

Rohan Jose Paul

Ronit John Daniel

Sebin Bejoy

Abstract

In response to the challenges faced in the educational and industrial sectors regarding the time-consuming nature of creating question papers, a solution named QGEN has been introduced. QGEN is a cutting-edge system specifically developed to automate the generation of multiple-choice questions or subjective questions from factual text. By utilizing QGEN, the arduous process of generating question papers is streamlined, allowing educators to focus more on teaching and improving the educational experience.

QGEN mainly utilizes a cutting-edge multi-task model based on the small version of the T5 Transformer. Leveraging the SQUAD, RACE dataset and a compact pre-trained T5 model, QGEN is trained to generate either MCQ's or subjective questions and to generate a final question paper.

By providing an efficient solution to the labor-intensive task of Question generation, QGEN emerges as a versatile tool, revolutionizing educational practices in both academic and industrial spheres.

Contents

List of Abbreviations

MCQ- Multiple Choice Question

NLP - Natural Language Processing

API - Application programming interface

ROUGE - Recall Oriented Understudy for Gisting Evaluation

BLEU - Bilingual Evaluation Understudy

RACE - Reading Comprehension Dataset From Examinations

SQuAD - Stanford Question Answering Dataset

RNN - Recurrent Neural Networks

JSON - JavaScript Object Notation

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Background

In today's age, education is the most important way of achieving success. When we discuss education, it is imperative to mention tests and examinations. Examinations prepare students in their quest for knowledge. So, having a proper examination paper and format is quite necessary. Now the traditional method of generating question papers has been manual. But this method can be ineffective at times owing to time consumption, repetition, etc. Thus having an Automated process of Question Paper Generation will be very effective.

Paper Based Systems:

- Human process.
- Slow as human labor is involved.

Paperless Based Systems:

- Automated process.
- Faster due to computer based automation.

1.2 Problem Definition

The manual creation of assessment questions places a considerable burden on faculty members, demanding extensive time and effort. This process often lacks the diversity and adaptability needed to meet the evolving educational landscape. The urgent call for innovation prompts the development of an automated system to streamline question paper creation. Such a system would not only alleviate the time constraints faced by faculty but also introduce efficiency, diversity, and adaptability in generating assessments. This

technological solution stands to revolutionize the educational assessment process, allowing educators to focus more on refining content and enhancing teaching methodologies while ensuring assessments align with modern standard

1.3 Scope and Motivation

1.3.1 Scope

The scope of this project involves developing and implementing an Automatic Question Paper Generator system using NLP techniques to streamline the creation of both subjective and objective questions for educational assessments. It encompasses various domains and subjects, meeting the diverse needs of educators and students across different fields of study. It also involves potential future enhancements and refinements to ensure the system's adaptability to evolving educational requirements and technological advancements. Additionally, the exploration of the system's impact on reducing the time and effort required for question generation is within its scope, ultimately benefiting educators and students in the assessment process.

1.3.2 Motivation

The motivation for this project originates from the growing need for subjective and objective based assessments in the realm of online learning, driving the requirement for an automated and effective question paper generation system. By developing such an application, we aim to harness the capabilities of artificial intelligence and machine learning to revolutionize the assessment process. It is rooted in the aspiration to ease the burden on educators by offering a cost-effective and time-efficient solution for assessment creation. It is driven by the potential of NLP techniques to improve the speed and precision of question generation, aligning with advancements in educational technology. It also arises from the goal of addressing the challenges associated with manual question generation and contributing to the evolution of assessment practices in educational institutions.

1.4 Objectives

The objective of QGEN, an Automated Question Paper Generation Application is to automate and streamline the process of creating question papers, in the field of education. It will optimize the assessment process in education, making it more efficient, and tailored to support student learning and achievement. By providing an easy user interface it makes question paper generation easier thus the application will meet the user's needs. The primary goals are:

- **Utility** - Question paper generators serve as a valuable tool in schools, colleges, and universities.
- **Time Saving** - Generating question papers manually can be a time-consuming task. Thus it eliminates the need for manual creation and selection of questions.
- **Scalability** - A question paper generator can handle the scalability effortlessly, generating papers for any number of students or exams.

1.5 Challenges

The different challenges associated with the project include the following:

- T5 model may generate irrelevant questions.
- The questions that are generated may not always be diverse.
- The questions generated may also need to be. examined by the faculty to ensure quality.

1.6 Assumptions

The assumptions made are as follows:

- User requires only 'Remember' type subjective questions and MCQ type questions.
- Input document should be in English.
- Input document does not contain equations.
- Input is in the form of PDF.

1.7 Societal / Industrial Relevance

The system's scalability allows for potential integration with online learning platforms and assessment management systems, enhancing its relevance in educational settings. Other societal and industrial impacts include:

- Reduced Stress and Anxiety
- Individualized Learning
- Technological Literacy
- Data-Driven Decision-Making
- Cost Savings
- Resource Optimization

1.8 Organization of the Report

The primary goal of this report explains the processes and efforts that went into developing a Question Paper Generation Application that helps in easing the burden on educators for setting question papers.

The first chapter deals with the general background of the project.

The second chapter includes the literature survey by showing a comparison between various methods used in question generation.

The third chapter contains the hardware and software requirements.

The fourth chapter describes the system overview, architectural design, module division and work schedule.

The fifth chapter contains the conclusion of the project and its future scope.

And lastly the report includes the various references used as well as appendixes of presentation, Vision, Mission, Programme Outcomes, Course Outcomes and CO-PO-PSO Mapping.

1.9 Chapter Conclusion

QGEN, an Automatic Question Paper Generator is a state-of-the-art tool powered by artificial intelligence, designed to revolutionize the creation of assessment questions. Its central goals encompass time-saving efficiency, ensuring consistency, scalability, and adaptability to dynamic curriculam. Its impact extends to the improvement of educational experiences for educators, institutions, and learners across diverse educational landscapes.

Chapter 2

Literature Survey

2.1 Leaf: Multiple-Choice Question Generation, Kristiyan Vachev et al., 2022 [1]

The paper presents Leaf, a system for generating multiple-choice questions from factual text. The system is designed to be used in educational settings, such as classrooms and MOOCs, to improve the efficiency of student examination setups. Leaf uses a combination of pre-processing techniques and machine learning models to generate high-quality questions with distractors. The system is open-source and can be adapted to different types of educational content.

Leaf uses a client-server architecture, where the client is used by the instructor to input the educational text and the required number of questions. The text is then passed through a REST API to the Multiple-Choice Question (MCQ) Generator Module, which performs pre-processing and generates the required number of question-answer pairs with distractors. The system uses a combination of pre-trained language models, such as T5 and sense2vec, to generate questions and distractors. The generated questions are evaluated using metrics such as BLEU, ROUGE, and METEOR.

The paper presents Leaf, a system for generating multiple-choice questions from factual text. The system is designed to be used in educational settings to improve the efficiency of student examination setups. Leaf uses a combination of pre-processing techniques and machine learning models to generate high-quality questions with distractors. The system is open-source and can be adapted to different types of educational content. In future work, the authors plan to experiment with larger pre-trained Transformers and to train

on additional data.

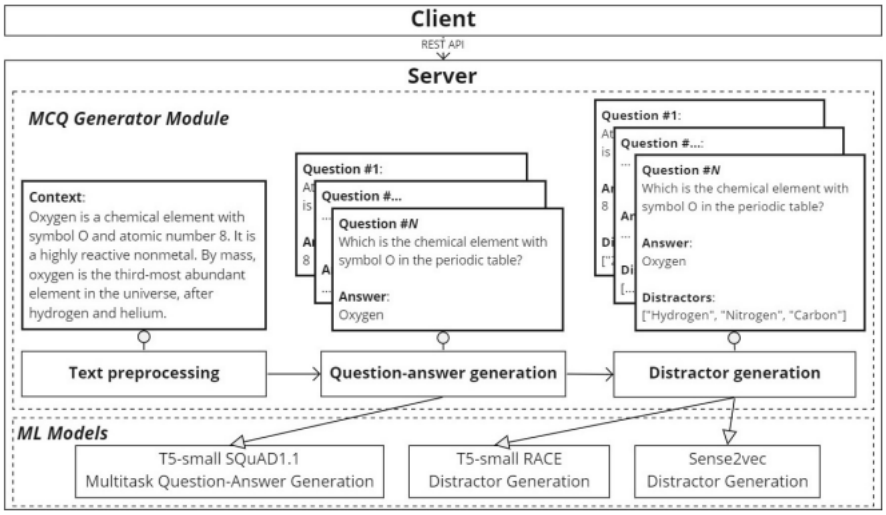


Figure 2.1: The general architecture of Leaf

2.2 Neural Question Generation from Text: A Preliminary Study, Qingyu Zhou et al., 2017 [2]

The paper focuses on the task of generating questions from natural language text using neural encoder-decoder models. The authors propose a method that uses a feature-rich encoder to encode answer position, POS and NER tag information to generate answer-focused questions. The encoder reads the input text and the answer position to produce an answer-aware input representation, which is fed to the decoder to generate an answer-focused question. The method is evaluated on the SQuAD dataset, which is composed of more than 100K questions posed by crowd workers on 536 Wikipedia articles.

The paper does a preliminary study on neural question generation from text with the SQuAD dataset, and the experiment results show that their method can produce fluent and diverse questions. The authors compare their method with a rule-based system and show that their method outperforms the rule-based system in terms of question diversity and quality. The generated questions can be answered by certain sub-spans of the given passage, which indicates that the method is able to capture the relevant information from the input text.

The authors suggest that the generated questions can be used for educational purposes and can also provide a large scale corpus of question-answer pairs. The method can be used to generate questions for educational materials, such as textbooks and online courses, and can also be used to augment existing question answering systems.

Overall, this research paper presents a novel approach to generating questions from natural language text using neural encoder-decoder models. The method is evaluated on the SQuAD dataset and shows promising results in generating fluent and diverse questions. The generated questions can be used for educational purposes and can also provide a large scale corpus of question-answer pairs. The authors suggest future work to investigate the potential of the method in improving question answering systems.

2.3 Automatic Multiple Choice Question Generation From Text : A Survey, Dhawaleswar Rao CH et al., 2020 [3]

This paper provides a systematic review of the literature on automatic generation of MCQ since the late 90's. It outlines a generic workflow for an automatic MCQ generation system, consisting of six phases, and discusses the techniques adopted in the literature for each phase. The paper also delves into the evaluation techniques used to assess the quality of system-generated MCQs and identifies areas for future research to enrich the literature in this field.
Here's a detailed explanation of each phase:

1. **Pre-Processing:** The pre-processing phase involves preparing the input text for further analysis. This may include tasks such as text normalization, structural analysis, and removal of non-text content.

2. **Sentence Selection:** In this phase, the system selects relevant sentences from the pre-processed text to form the basis for generating MCQs. The selection of sentences is crucial for ensuring that the generated questions are based on the most important and relevant information in the text.

3. **Key Selection:** Once the relevant sentences are identified, the key selection phase involves determining the key information or concepts within these sentences that will form the basis for the MCQs. This phase may involve identifying keywords, important facts, or concepts that are essential for constructing meaningful MCQs.

4. **Question Generation:** In this phase, the system formulates the actual multiple choice questions based on the selected sentences and key information. The formulation of clear, unambiguous, and effective questions is a critical aspect of this phase.

5. **Distractor Generation:** Distractors are the incorrect options provided in a multiple choice question. In this phase, the system generates distractors that are similar to the correct answer but are ultimately incorrect. The generation of high-quality distractors is essential for creating challenging and effective MCQs.

6. **Post-Processing:** The post-processing phase involves any additional steps or refinement processes applied to the generated MCQs. This may include tasks such as formatting the questions and ensuring grammatical correctness.

Overall, the paper provides a valuable overview of the techniques, evaluation methods, and future research directions in the field of automatic MCQ generation, making it a significant contribution to the literature in this area.

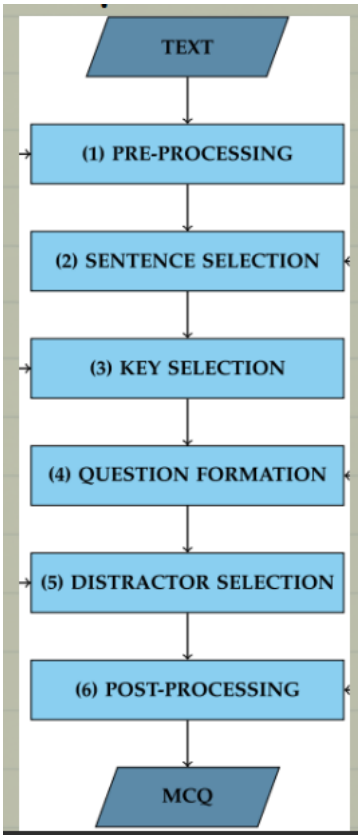


Figure 2.2: Flowchart of generating MCQ (survey paper)

2.4 Transformer-based End-to-End Question Generation, Luis Enrico Lopez et al.,2020 [4]

This paper focuses on the task of generating questions from a given context paragraph. The authors present a simple yet effective approach to this task using transformer-based finetuning techniques. The model is trained on a single pretrained language model and does not require the use of additional mechanisms, answer metadata, or extensive features.

The authors compare their approach to previous techniques for question generation, such as RNN-based Seq2Seq models and models that employ answer-awareness and other special mechanisms. They show that their model outperforms previous more complex RNN-based Seq2Seq models and performs on par with Seq2Seq models that employ answer-awareness and other special mechanisms.

The paper also discusses factors that affect the performance of the model, such as input data formatting and the length of context paragraphs. The authors provide examples of

the model's failure modes and possible reasons why it fails. They also discuss possible future work, such as incorporating additional features and mechanisms to improve the model's performance.

Overall, this research paper presents a promising approach to the task of generating questions from a given context paragraph using transformer-based finetuning techniques. The model is simple yet effective and outperforms previous more complex models. The paper provides valuable insights into the factors that affect the performance of the model and possible future directions for research in this area.

2.5 Scalable Educational Question Generation with Pre-trained Language Models, Sahan Bulathwela et al., 2023 [5]

This paper focuses on the development of a novel model for generating educational questions called EduQG. The authors argue that generating scalable educational questions is crucial for democratizing education and enabling self-assessment at scale. While existing language models are used for question generation, their utility in education has only been explored recently. This work demonstrates how a large language model can be adapted for educational question generation.

The paper begins by discussing related work in the field of AI systems capable of generating educational questions for technology-enhanced learning. The authors explain that this involves two main sub-tasks: Question Generation (QG), where a model generates a question based on given information, and Question Answering (QA), where a model generates a response to a question. QG is essential for QA, and both tasks are part of reading comprehension tasks. This paper focuses on QG specifically.

The authors then introduce their proposed model, EduQG, which is based on the T5 language model. EduQG is pre-trained on a large corpus of educational text and fine-tuned on a science question dataset. The paper describes the experiments conducted to validate the effectiveness of EduQG, including pre-training with educational text, investigating the impact of pre-training data size on question generation, and enhancing educational questions through fine-tuning with a science question dataset. The experi-

mental results show that pre-training and fine-tuning with domain-specific scientific text can outperform a state-of-the-art baseline, providing significant evidence for building an effective educational question-generation model.

The paper also discusses potential applications of EduQG in online education, such as enabling self-assessment and personalized learning. The authors suggest that EduQG can be adapted to generate questions for subjects other than science, and that future research could explore the use of EduQG in other domains.
Overall, this research paper presents a novel approach to generating educational questions that can help democratize education and enable self-assessment at scale. The authors demonstrate the effectiveness of their proposed model, EduQG, through a series of experiments and suggest potential applications for online education.

2.6 Summary and Gaps Identified

2.6.1 Summary of Literature Survey

Here is the summary of the existing methods used currently.

Paper	Method	Contribution
Transformer-based End-to-End Question Generation, Luis Enrico Lopez, Diane Kathryn Cruz, Jan Christian Blaise Cruz, Charibeth Cheng. [2020]	The method involves training a question generation model on the Stanford Question Answering Dataset (SQuAD) using transformer-based finetuning techniques	It was inferred from this paper that tranformer-based models are better than conventional RNN-based approaches
Scalable Educational Question Generation with Pre-trained Language Models, Sahan Bulathwela, Hamze Muse and Emine Yilmaz. [2023]	The method involves fine-tuning a Pre-trained T5 Language Model with domain-specific data to generate educational questions	This research paper gives us the idea that it is better to use a pre-trained model than creating one from scratch. The paper uses Google-T5

Paper	Method	Contribution
Neural Question Generation from Text: A Preliminary Study, Qingyu Zhou, Nan Yang, Furu Wei, Chuanqi Tan, Hangbo Bao, Ming Zhou. [2017]	The method uses neural encoder-decoder models with a feature-rich encoder and attention mechanism to generate questions.	This research paper presents the idea of using SQuAD 1.1 dataset for Fine-tuning the T5-base model
Automatic Multiple Choice Question Generation From Text : A Survey, Dhawaleswar Rao CH and Sujan Kumar Saha. [2018]	This paper explains methods used to generate MCQs. It performs text normalization, sentence selection, key selection based on pattern matching and distractor generation using WordNet.	Identifying and analyzing techniques for automatic MCQ generation, providing a structured workflow and critical analysis of existing methods. This review serves as a valuable resource for researchers .

Table 2.1: Comparison Table of Literature Survey

2.6.2 Gaps Identified

- **Limited Creativity:** An automated system may not be able to generate questions that are as creative or innovative as those generated by a human.
- **Limited Contextual:** While the proposed system utilizes advanced natural language processing techniques, it may still have limitations in understanding the context of the input text. This could result in the generation of questions that are not fully aligned with the intended meaning of the text.
- **Limited Flexibility:** An automated system may not be as flexible as a human in adapting to changes in the input text or adjusting the types of questions generated based on feedback from users.
- **Potential Bias:** An automated system may inadvertently introduce bias into the generated questions, particularly if the input text contains biased language or perspectives.

2.7 Chapter Conclusion

Chapter 2 provides a detailed overview of existing research in the field of automatic question generation from text. The paper [1] presents methods involve pre-processing techniques and the use of pre-trained language models to generate high-quality questions with distractors, contributing to its effectiveness in improving the educational process

. The paper [2] is related to neural question generation techniques demonstrating the effectiveness of their method on the SQuAD dataset, emphasizing its potential for educational use. The the paper [3] systematically reviews automatic multiple-choice question generation, presenting a generic workflow and discussing evaluation methods. The paper [4] focuses on transformer-based end-to-end question generation, showcasing a simple yet effective model that outperforms previous approaches. The final paper [5] introduces EduQG, a model based on the T5 language model, demonstrating its effectiveness in generating educational questions. The chapter concludes with a summary and identified gaps, highlighting potential limitations in creativity, contextual understanding, flexibility, and the risk of bias in automated question generation systems.

Chapter 3

Requirements

3.1 Hardware Requirements

The following are the hardware requirements to develop the QGEN Application.

- Hardware: Keyboard, Mouse, Monitor, CPU for a standard i3 processor.
- Processor: Intel Core i3.
- RAM: Minimum 8GB.

3.2 Software Requirements

The following are the softwares used in the development of the application.

3.2.1 Operating System: Windows 8 or above

Windows 8 is a personal computer operating system that is part of the Windows NT family. Windows 8 introduced significant changes to the Windows operating system and its user interface (UI), targeting both desktop computers and tablets.

It is a touch-optimized platform based on the modern Metro design architecture, which specifies how applications are delivered and rendered in the UI. Along with having a much different look and feel from its predecessor Windows 7, Windows 8 also boasted faster startup times and better performance.

3.2.2 Python 3.10

Python is a versatile and widely used programming language known for its simplicity and readability. It offers an extensive standard library and a vibrant ecosystem of third-party packages, making it suitable for various domains, including web development, data

analysis, artificial intelligence, and automation.It supports RegEX which is needed for this project.

It also supports multiple programming paradigms, including procedural, object-oriented, and functional programming, allowing developers to choose the approach that best suits their needs. Python 3.10 introduces improved error messages, making debugging and troubleshooting easier.

3.2.3 MongoDB 6.0

MongoDB is a popular NoSQL database management system that provides a flexible and scalable solution for storing and retrieving data. As a document-oriented database, MongoDB stores data in a JSON-like format called BSON, making it easy to work with and highly compatible with modern web applications.

It offers features such as high availability, horizontal scaling, and automatic sharding, allowing for seamless handling of large datasets and high traffic loads. With its flexible schema and dynamic document model, MongoDB is an excellent choice for agile development, rapid prototyping, and applications requiring frequent schema changes.

3.2.4 Flask(Python Web Framework)

Flask is a lightweight and versatile micro web framework for Python, widely acclaimed for its simplicity and efficiency in building web applications. With its minimalistic design and intuitive syntax, Flask offers developers a straightforward and elegant approach to web development.

It provides a wide range of features, including routing, templating, and request handling, making it suitable for projects of any size. Flask's modular structure allows developers to choose and integrate various extensions based on their specific needs, enhancing its flexibility and scalability.

Chapter 4

System Architecture

4.1 System Overview

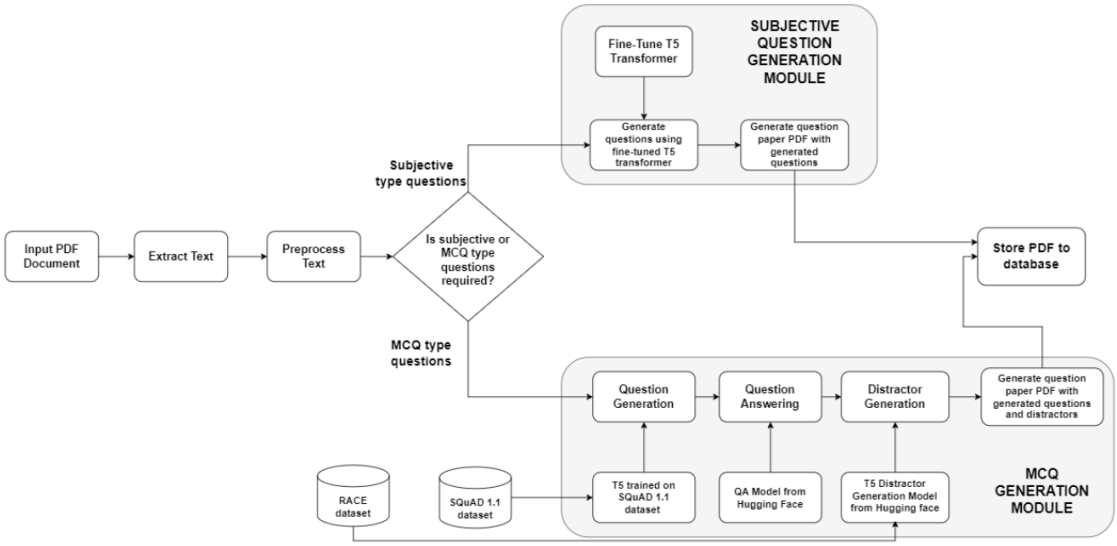


Figure 4.1: Architecture Diagram

- **Input Document:** The process begins with providing an input document. This document likely contains textual information that will be used to generate a question paper. It is a PDF document.
- **Extract Text from Document:** The system extracts the relevant text from the input document. This step involves isolating and capturing the necessary content that will be used to create questions.
- **Preprocess the Text:** Before generating questions, the extracted text undergoes preprocessing. This step may involve tasks such as tokenization, removing bullet points and other necessary stuff, thus cleaning the text to make it suitable for further processing.

- **Selection of Question Paper type:** User is given option to choose between subjective or MCQ type questions for the question paper to be created.
- **Fine-Tune the T5 Transformer using SQuAD 1.1 Dataset:** The SQuAD 1.1 Dataset consists of 107,785 question-answer pairs on 536 articles. The T5 Transformer is fine-tuned using this dataset and this is used for question generation.
- **Generation of Answer:** The preprocessed text along with the question is fed into question answer model. The model generates the respective answer.
- **Generate Distractors:** The preprocessed text along with the question and answer is fed into the T5 distractor generation model. This enables the generation of relevant distractors for a specific MCQ question generated.
- **Generate Question Paper PDF by Adding the Questions and then an Answer Key:** The system combines the generated questions with other necessary components, such as formatting instructions and section headings, to create a complete question paper with answer key.
- **Store the PDF to Database:** The final question paper, now in PDF format, is saved and stored in a database. This step ensures that the question paper is readily accessible when needed.
- **Download the Question Paper PDF:** After the question paper is generated and stored in the database, the system provides a download option for users to access the PDF version. This can be achieved through a user interface where users can click a button to initiate the download process. With this, the system allows users to obtain the generated question paper as a downloadable PDF file for offline use.

4.2 Architectural Design

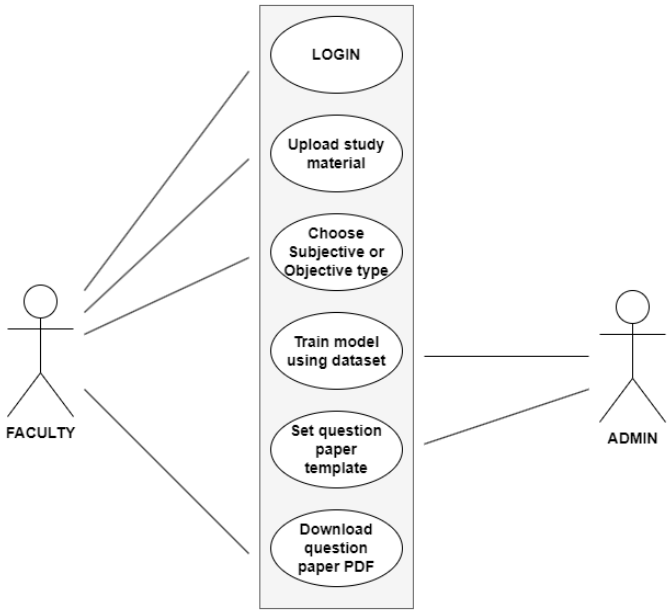


Figure 4.2: Use Case Diagram

4.3 Module Division

4.3.1 Modules

The project is divided into following five modules:

- **Module 1: Flask UI** - An interactive UI where faculties can generate question papers for various subjects and all previously generated question papers will be displayed.

    1. Login and Authentication
    2. User Dashboard with multiple subjects
    3. Uploading document
    4. View previously generated question papers

- **Module 2: Text Extraction and Preprocessing** - Text has to be extracted from input document. Preprocessing should be done on extracted text and make it suitable for transformer models.

1. Text Extraction using PyPDF2.
2. Text Preprocessing using RegEx.

- **Module 3: : MCQ Question Answer Generation** - Fine-tune T5 Transformer model using SQuAD 1.1 Dataset for question generation.
  Generate answer by feeding preprocessed text and question into question-answer model.

- **Module 4: Distractor Generation** - To create contextual distractors, utilize T5 distractor generation model to create relevant distractors for MCQ questions.

- **Module 5: Question Paper Generation** - Place the generated questions into the question paper with a predefined template.

  1. Generate PDF of question paper.
  2. Store the PDF in database.
  3. View and download previously generated question papers.

4.3.2 Work Breakdown and Responsibilities

- **Flask UI:** Nevin, Rohan, Ronit, Sebin
- **Text Extraction and Preprocessing:** Rohan, Sebin
- **MCQ Generation:** Rohan and Nevin
- **Distractor Generation:** Ronit and Sebin
- **Question Paper Generation:** Ronit, Nevin

4.3.3 Work Schedule - Gantt Chart

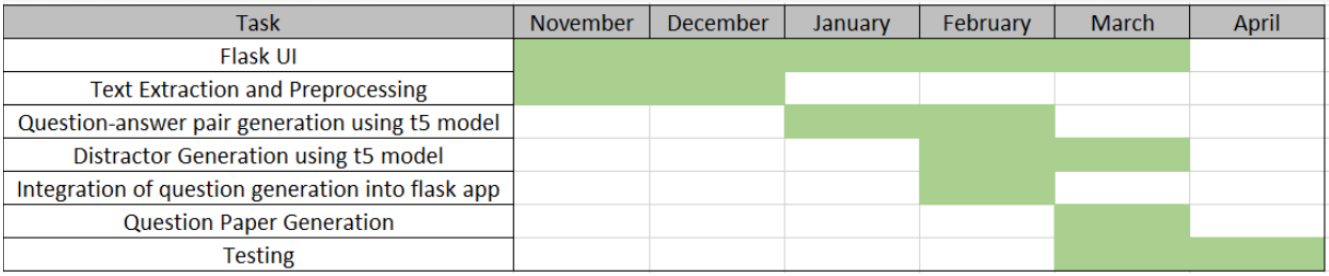


Figure 4.3: Gantt Chart

4.4 Chapter Conclusion

Chapter 4 offers a comprehensive overview of the system architecture and design for the automatic question paper generation project. The architecture diagram (Figure 4.1) provides a high-level view of key components, while the use case diagram (Figure 4.2) illustrates interactions between modules. The project is organized into five modules, addressing specific functionalities crucial for system execution, ranging from Flask UI for user interaction to modules focused on text extraction, question generation, and paper creation. The work breakdown and responsibilities section (4.3.2) assigns tasks based on expertise, promoting collaborative efforts for a well-rounded development process. The Gantt chart (4.3.3) visually outlines the project timeline, emphasizing the importance of adhering to deadlines. The outlined architecture, design, and work breakdown structure will serve as a foundation for successful system development. The modular approach enhances scalability, flexibility, and collaboration.

Chapter 5

System Implementation

5.1 Datasets Identified

The dataset utilized in the project is the Stanford Question Answering Dataset (SQuAD) version 1.1. SQuAD 1.1 is a widely-used dataset in the field of natural language processing (NLP) and machine reading comprehension tasks. It consists of a collection of questions on a set of Wikipedia articles, where the answers to these questions are from the corresponding articles.

Key properties of the SQuAD 1.1 dataset include:

- Questions and Contexts: It contains a diverse set of questions covering various topics sourced from Wikipedia articles. Each question is paired with a corresponding passage from the article that contains the answer.
- Answer Annotations: Human annotators have provided precise spans within the passages that contain the correct answers to the questions.
- Training and Evaluation Sets: The dataset is divided into training and evaluation sets, enabling the development and assessment of machine learning models for reading comprehension tasks.

Sample subsets of the dataset:

Context: "As of 2019, the total population of the world was estimated to be approximately 7.7 billion people according to the United Nations. This represents a significant increase from the estimated population of 7.3 billion in 2015. Population growth rates vary significantly by region, with some areas experiencing rapid population growth while

others are seeing declining populations due to factors such as fertility rates, migration, and mortality rates."

Question: What was the estimated world population in 2015?

Answer: 7.3 billion

Context: "The Great Wall of China is a series of fortifications made of stone, brick, tamped earth, wood, and other materials, generally built along an east-to-west line across the historical northern borders of China to protect the Chinese states and empires against the raids and invasions of the various nomadic groups of the Eurasian Steppe."

Question: What was the purpose of the Great Wall of China?

Answer: To protect the Chinese states and empires against raids and invasions

5.2 Proposed Methodology/Algorithms

1. Extract text from PDF input document using PyPDF2.
2. Preprocess extracted text by tokenization and removing unnecessary elements for further processing using RegEx.
3. Provide user option to choose between subjective or Multiple Choice Question (MCQ) type questions.
4. Fine-tune T5 Transformer model using SQuAD 1.1 Dataset for question generation.
5. Generate answer by feeding preprocessed text and question into question-answer model.
6. Utilize T5 distractor generation model to create relevant distractors for MCQ questions.
7. Combine generated questions with formatting instructions to create a complete PDF question paper.
8. Save final question paper in PDF format to a database for easy access.
9. Offer download option for users to retrieve the generated question paper as a PDF file.

5.3 User Interface Design

1. **Register Page:** This page provides the option for a new user to create an account. This username and password is stored in the database.
2. **Login Page:** This page allows user to login into the application.
3. **User Dashboard Page:** This is the welcome page. It has the option to create a new classroom and there are existing classrooms as well.
4. **Upload Page:** In this page user uploads the PDF document and it displays the preprocessed text and questions generated. The user can choose between subjective and MCQ for the type of question to be generated.
5. **Previously Generated Question Paper:** This page contains the list of the previously generated question papers under the respective subject.

5.4 Database Design

The database consist of two collections:

1. **users:** Each document of this collections comprises of:
    - Username
    - Password
    - A list of subjects for which the faculty has created a classroom
2. **pdf:** In this collection the elements are:
    - PDF id
    - User id (References the user id from the user collection)
    - Timestamp of when the question paper was generated
    - Name of the question paper generated
    - Data in binary format

5.5 Description of Implementation Strategies

The implementation strategies for this project are as follows:

1. Text Extraction: Utilize PyPDF2 to extract text from PDF input documents. This involves parsing the PDF files and extracting the textual content for further processing.

2. Text Preprocessing: Preprocess the extracted text using regular expressions (RegEx) for tokenization and removing unnecessary elements such as special characters, bullet points, and formatting artifacts. This step ensures that the text is clean and ready for further processing.

3. Question Type Selection: Implement a user interface or input mechanism to allow users to choose between generating subjective or Multiple Choice Question (MCQ) type questions. This ensures flexibility and customization according to user preferences.

4. Model Fine-Tuning: Fine-tune the T5 Transformer model using the SQuAD 1.1 Dataset. This involves training the model on the question-answer pairs from the dataset to adapt it to the specific task of question generation.

5. Answer Generation: Generate answers by feeding the preprocessed text and questions into the fine-tuned T5 Transformer model. The model predicts the answers based on the given context and question.

6. Distractor Generation: Utilize the T5 distractor generation model to create relevant distractors for MCQ questions. This involves providing the preprocessed text, question, and correct answer to the model, which generates plausible incorrect options.

7. Question Paper Creation: Combine the generated questions with formatting instructions to create a complete PDF question paper. This step involves organizing the questions, adding section headings, and formatting the document for readability and coherence.

8. PDF Storage: Save the final question paper in PDF format to a database for easy access and retrieval. This ensures that the generated question papers can be stored

securely and accessed when needed.

9. Download Option: Provide users with a download option to retrieve the generated question paper as a PDF file. This allows users to save the question paper locally for offline use or distribution.

5.6 Chapter Conclusion

In conclusion, the project outlines a systematic approach for generating question papers from PDF documents using the SQuAD 1.1 dataset and T5 Transformer models. By incorporating text extraction, preprocessing, and model fine-tuning, it aims to provide question papers and answer key for educational purposes. The user interface design allows users to upload documents, select question types, and retrieve generated question papers. With a well-structured database design, the project ensures efficient storage, retrieval, and distribution of generated question papers. Overall, this project presents a framework for automating the process of question paper generation, enhancing accessibility and efficiency in educational settings.

Chapter 6

Results and Discussions

6.1 Overview

Our automatic question paper generator has shown impressive effectiveness in simplifying the tedious process of creating examination materials. The end results showcase a significant reduction in time and effort required for question preparation with answering. Further analysis reveals the adaptability of the system to various educational contexts, offering educators a versatile tool to enhance the assessment process and promote effective learning outcomes.

6.2 Testing

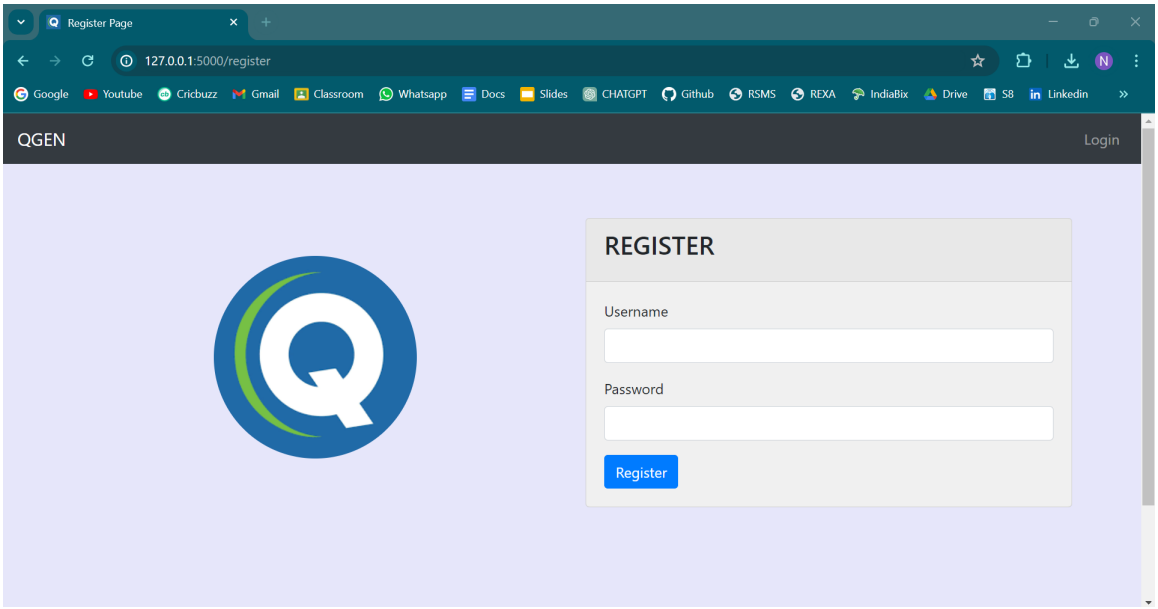


Figure 6.1: Register page

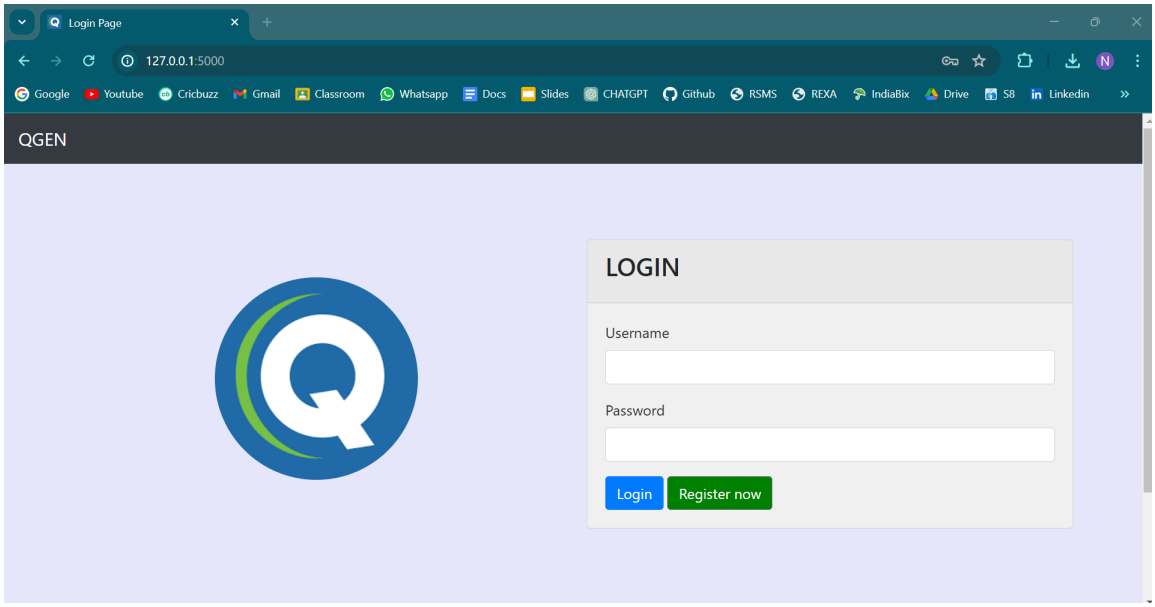


Figure 6.2: Login Page

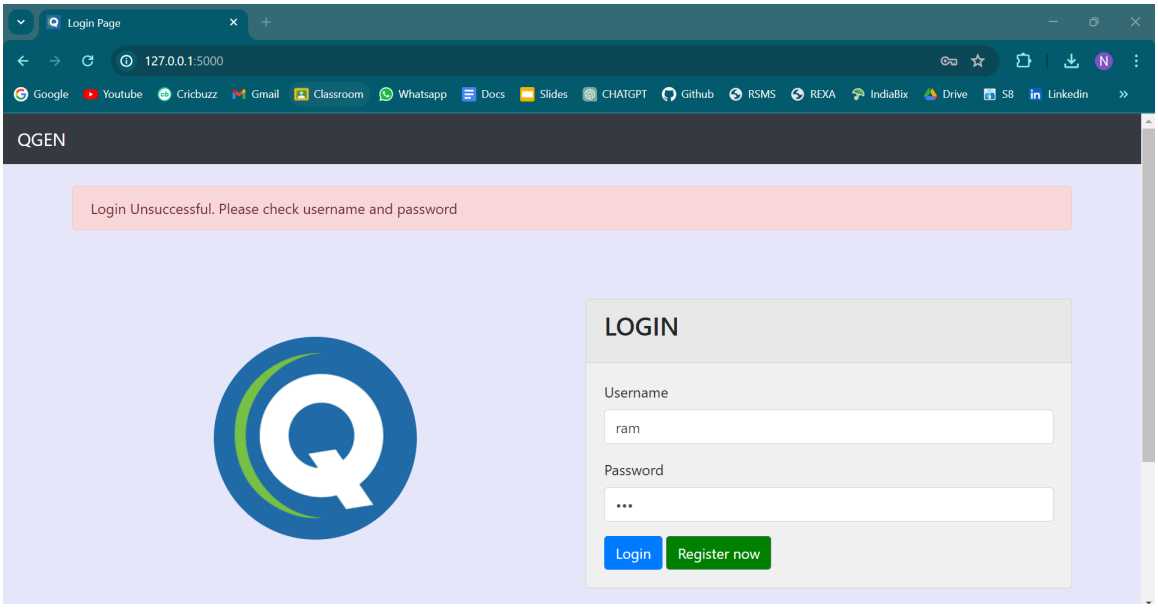


Figure 6.3: Login Unsuccessful

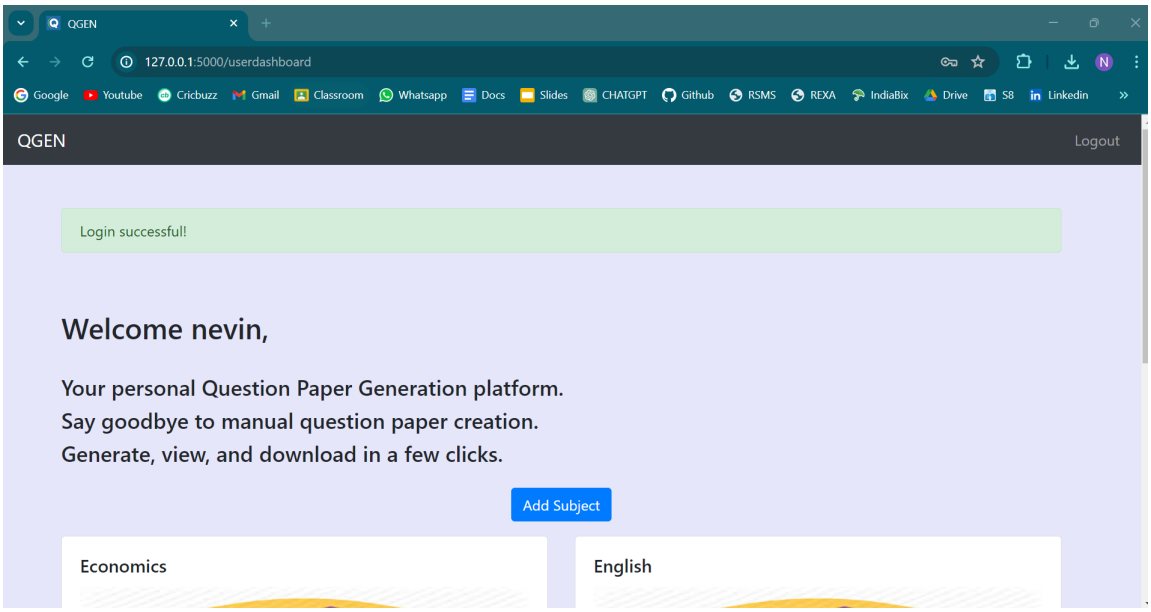


Figure 6.4: Dashboard

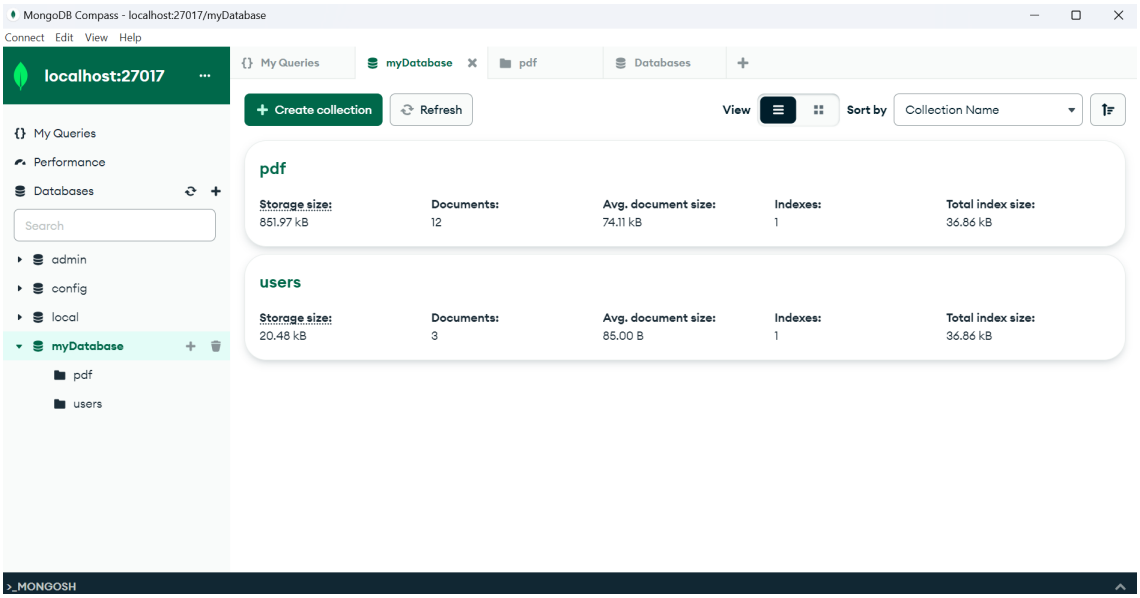


Figure 6.5: Database

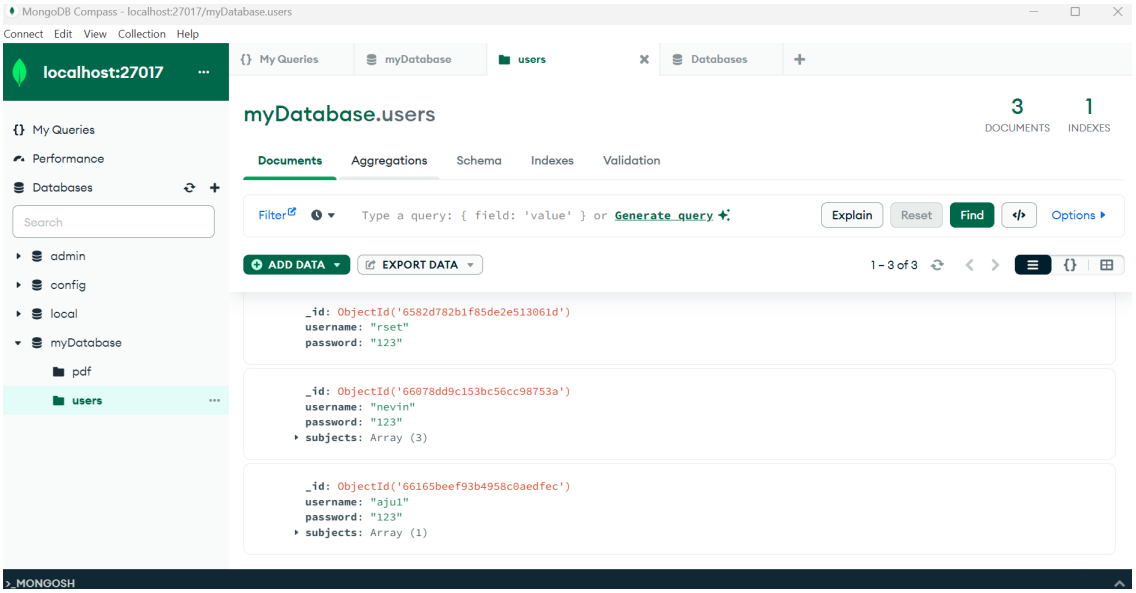


Figure 6.6: Users Collection

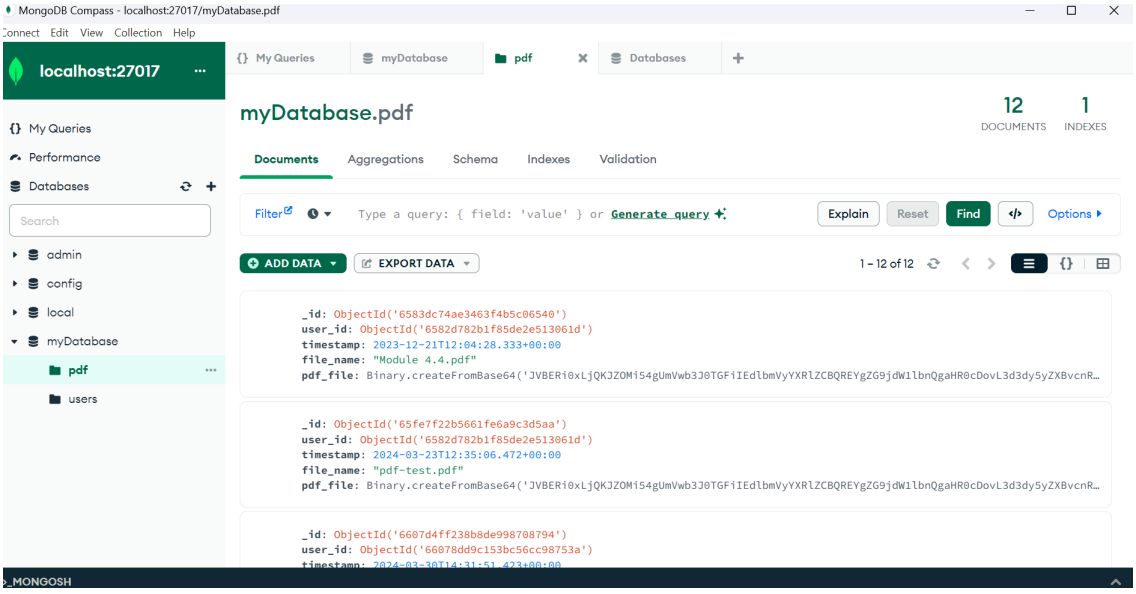


Figure 6.7: PDF Collection

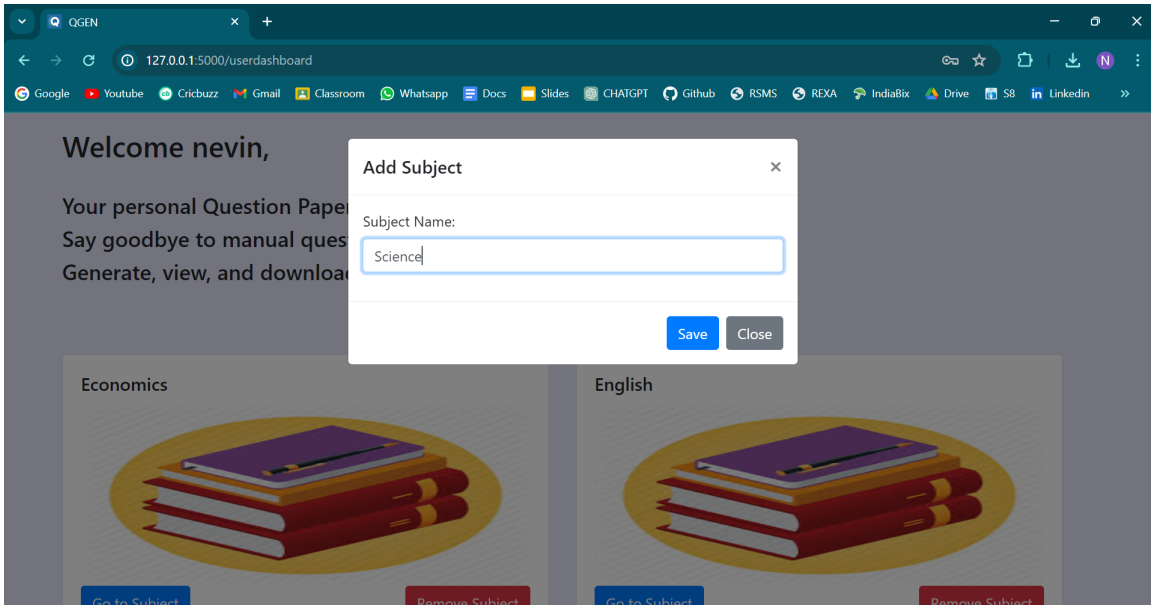


Figure 6.8: Add Subject

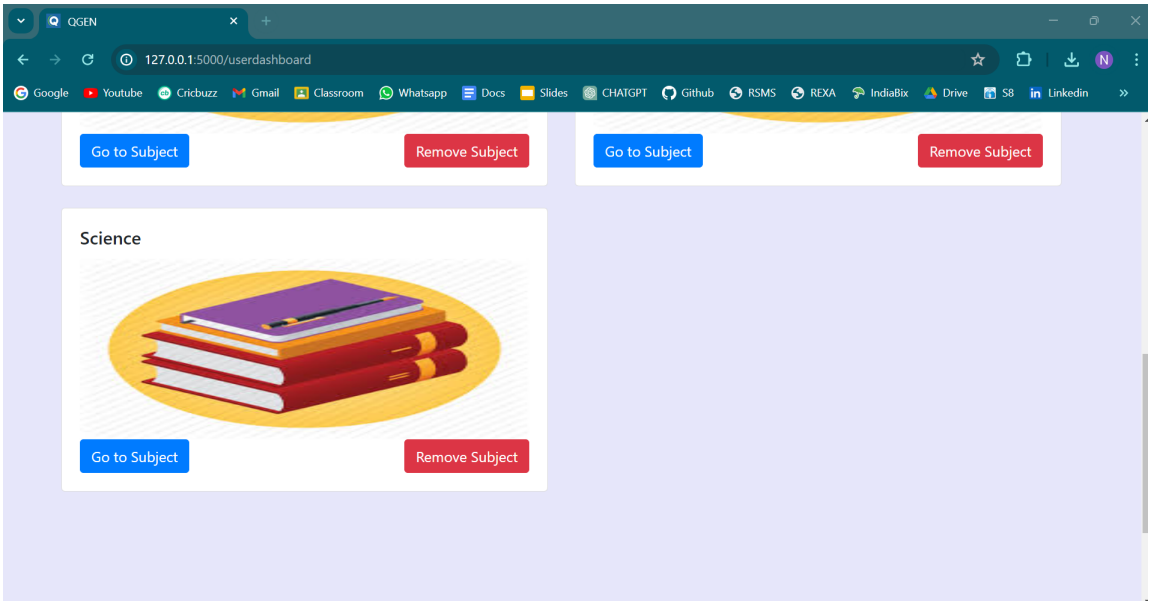


Figure 6.9: Successful addition of subject

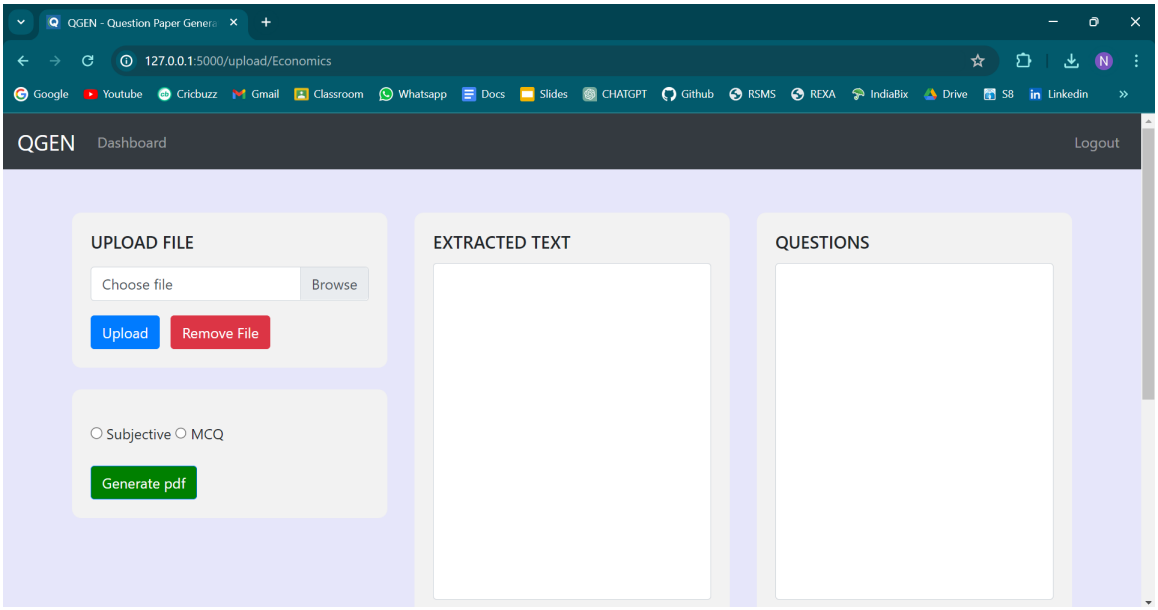


Figure 6.10: Upload Page

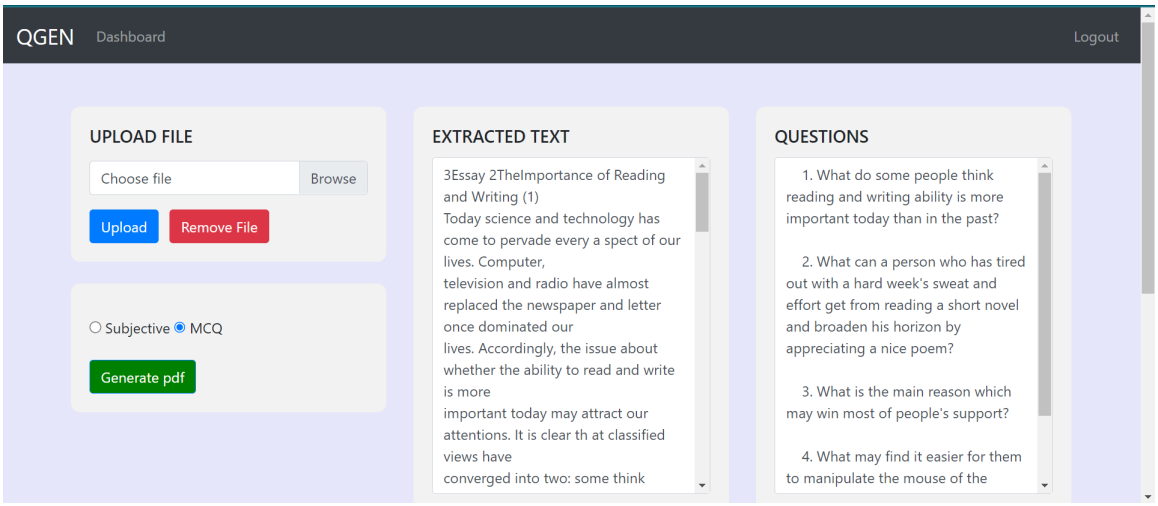


Figure 6.11: Upload Page after question generation

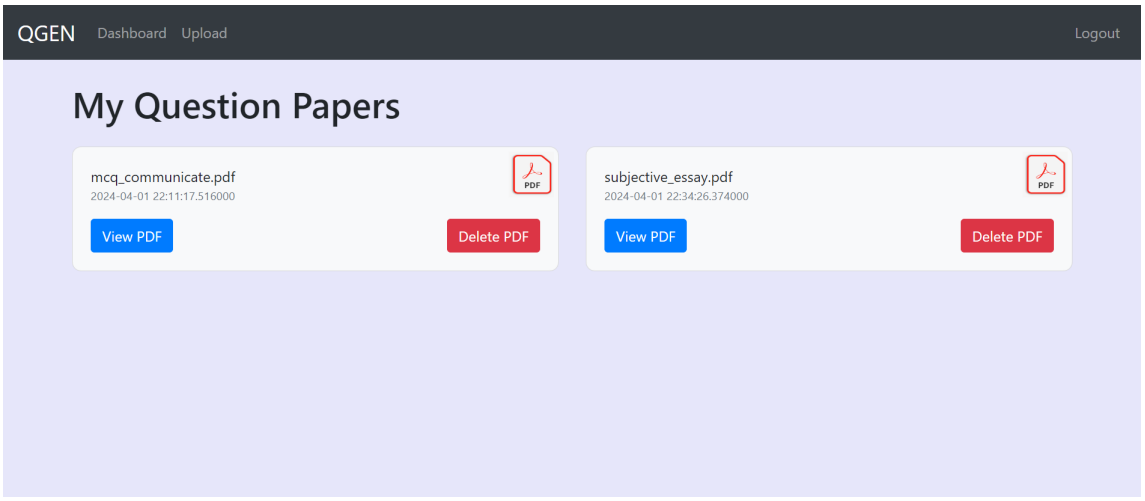


Figure 6.12: Question Paper Page

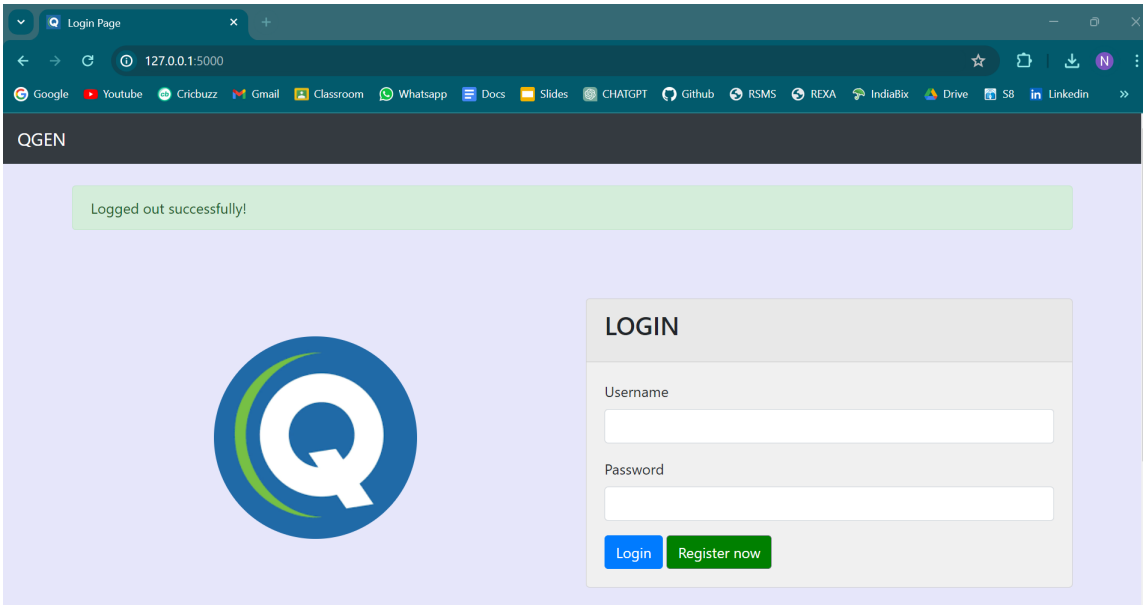


Figure 6.13: Screen after logging out

Model Question Paper

Reg No:
Name:

RAJAGIRI SCHOOL OF ENGINEERING & TECHNOLOGY
(AUTONOMOUS)

English

1) What does hesitate mean?

a) weakness

b) self- insecurity

c) hesitance

2) What will you surely learn from listening to others?

a) how to defend yourself

b) how to be heard

c) ne w things

d) how to speak

Figure 6.14: MCQ Question Paper

ANSWER KEY

1) self- insecurity

2) ne w things

3) listening part

4) your own thoughts

5) Establishing and developing effective communication

6) the listener

Figure 6.15: Answer Key

Model Question Paper

Reg No:
Name:

RAJAGIRI SCHOOL OF ENGINEERING & TECHNOLOGY
(AUTONOMOUS)

English

1) What is the main reason which may win most of people's support?

2) What do some people think reading and writing ability is more important today than in the past?

3) What can a person who has tired out with a hard week's sweat and effort get from reading a short novel and broaden his horizon by appreciating a nice poem?

4) What may find it easier for them to manipulate the mouse of the computer than to write things down by hand?

Figure 6.16: Subjective Question Paper

ANSWER KEY

1) if lack the reading and writing ability

2) relax themselves

3) lots of benefits and relief

4) children

Figure 6.17: Answer Key

6.3 Discussion

By implementing the project of automated question paper generator we found that the project gives us good questions but not grammatically correct. We generated questions based on different topics and we finally realized that school based topics are best for generating questions. There are no repeated questions however the questions generated are basic and not very difficult.

Our question paper however needs human evaluation as then only we can evaluate the quality of question generated. Based on such feedback the accuracy and precision can be provided helping to validate the quantitative metrics.

Chapter 7

Conclusions & Future Scope

7.1 Conclusion

The process of manually creating questions is a cumbersome one and so the system that is created helps to partially solve the difficulty of the task thus reducing human intervention and it is a cost and time effective system.

The project describes QGEN, an automated system that progresses from the traditional method of paper generation to an automated process by uploading the study material. The task involves fine-tuning a T5 Transformer model using the SQuAD 1.1 dataset to generate questions. Then, utilizing the trained model to generate answers by providing preprocessed text and questions. Subsequently, a T5 distractor generation model is employed to create relevant distractors for multiple-choice questions (MCQs). A question paper is generated using these questions and is downloaded in pdf format.

Thus QGEN represents a transformative tool in education and assessment. Its advanced natural language processing capabilities and adaptive learning features streamline the question creation process, offering educators efficient tools for evaluation. The system's potential applications span diverse educational domains, from e-learning platforms to professional certifications.

However, challenges such as bias, contextual limitations, and the need for continuous validation persist. Striking a balance between innovation and addressing these concerns will be crucial in maximizing the benefits of QGEN, ensuring they contribute positively to the evolving landscape of learning and evaluation.

7.2 Future Scope

We hope to extend the application by:

1. Input file can be of any file format.
2. Equations in the input file may be used to generate questions.
3. Setting the difficulty level of questions.
4. Allowing user to input number of questions he/she requires and select what type of questions he/she requires.
5. Extend other aspects of Bloom's Taxonomy to generate more diverse questions.

References

[1] K. Vachev, M. Hardalov, G. Karadzhov, G. Georgiev, I. Koychev, and P. Nakov, *Leaf: Multiple-Choice Question Generation*, 04 2022, pp. 321–328.

[2] Q. Zhou, N. Yang, F. Wei, C. Tan, H. Bao, and M. Zhou, "Neural question generation from text: A preliminary study," *ArXiv*, vol. abs/1704.01792, 2017. [Online]. Available: https://api.semanticscholar.org/CorpusID:9745861

[3] D. R. Ch and S. K. Saha, "Automatic multiple choice question generation from text: A survey," *IEEE Transactions on Learning Technologies*, vol. 13, pp. 14–25, 2020. [Online]. Available: https://api.semanticscholar.org/CorpusID:70335228

[4] L. E. Lopez, D. K. Cruz, J. C. B. Cruz, and C. K. Cheng, "Transformer-based end-to-end question generation," *ArXiv*, vol. abs/2005.01107, 2020. [Online]. Available: https://api.semanticscholar.org/CorpusID:218487550

[5] S. Bulathwela, H. Muse, and E. Yilmaz, "Scalable educational question generation with pre-trained language models," 05 2023.

[6] M. Agarwal and P. Mannem, "Automatic gap-fill question generation from text books," in *Proceedings of the 6th Workshop on Innovative use of NLP for Building Educational Applications*, Hyderabad, AP, India, 2011, pp. 56–64.

[7] Y. Liu, "Fine-tune BERT for extractive summarization," *arXiv preprint arXiv:1903.10318v2*, Sep 2019.

[8] A. Y. Satria and T. Tokunaga, "Automatic generation of english reference question by utilising nonrestrictive relative clause," in *Proceedings of the 9th International Conference on Computer Supported Education*, 2017, pp. 379–386.

[9] R. Nadig, J. Ramanand, and P. Bhattacharyya, "Automatic evaluation of wordnet synonyms and hypernyms," in *Proceedings of ICON-2008*, 2008.

[10] D. Coniam, "A preliminary inquiry into using corpus word frequency data in the automatic generation of english language cloze tests," *CALICO Journal*, vol. 14, no. 2–4, pp. 15–33, 1997.

Appendix A: Presentation

QGEN - Automatic Question Paper Generator

100% Presentation

Guide: Ms. Sangeetha Jamal, Asst. Professor, DCS

Team 4
Nevin Aju - U2003151
Rohan Jose Paul - U2003171
Ronit John Daniel - U2003174
Sebin Bejoy - U2003190

April 28, 2024

Overview

1. Problem Definition
2. Project objectives
3. Novelty of Idea and Scope of Implementation
4. Literature Review
5. Methodology
6. Architecture Diagram
7. Results
8. Work Division Among Team Members
9. Conclusion
10. Future Scope
11. References
12. Paper Publication(Accepted)

Problem Definition

Faculty members face significant challenges in manually generating questions for assessments, and there is a compelling need for the development of an automated system to streamline the question paper creation process.

Project objective

To develop a software that can generate a question paper which contain MCQs or subjective questions, depending on user ' s choice with the help of T5 Transformer models.

Novelty of Idea and Scope of Implementation I

1. Integration with Educational Platforms: Another aspect of the implementation scope is the integration of the Automatic Question Paper Generator with existing educational platforms such as learning management systems (LMS) or online assessment tools.

2.Subject Versatility: Ensure compatibility across diverse subjects for broad applicability.

3.Customization Options: Set questions according to student proficiency and educator preferences.

Literature Review

Paper	Method	Contribution
Transformer-based End-to-End Question Generation, Luis Enrico Lopez, Diane Kathryn Cruz, Jan Christian Blaise Cruz, Charibeth Cheng. [2020]	The method involves training a question generation model on the Stanford Question Answering Dataset (SQuAD) using transformer-based finetuning techniques	It was inferred from this paper that tranformer-based models are better than conventional RNN-based approaches
Scalable Educational Question Generation with Pre-trained Language Models, Sahan Bulathwela, Hamze Muse and Emine Yilmaz. [2023]	The method involves fine-tuning a Pre-trained T5 Language Model with domain-specific data to generate educational questions	This research paper gives us the idea that it is better to use a pre-trained model than creating one from scratch. The paper uses Google-T5

Paper	Method	Contribution
Neural Question Generation from Text: A Preliminary Study, Qingyu Zhou, Nan Yang, Furu Wei, Chuanqi Tan, Hangbo Bao, Ming Zhou. [2017]	The method uses neural encoder-decoder models with a feature-rich encoder and attention mechanism to generate questions.	This research paper presents the idea of using SQuAD 1.1 dataset for Fine-tuning the T5-base model
Automatic Multiple Choice Question Generation From Text : A Survey, Dhawaleswar Rao CH and Sujan Kumar Saha. [2018]	This paper explains methods used to generate MCQs. It performs text normalization, sentence selection, key selection based on pattern matching and distractor generation using WordNet.	Identifying and analyzing techniques for automatic MCQ generation, providing a structured workflow and critical analysis of existing methods. This review serves as a valuable resource for researchers .

Methodology

The project is divided into the following 5 modules:

Module 1: Flask UI - An interactive UI where faculties can generate question papers for various subjects and all previously generated question papers will be displayed.

1. Login and Authentication
2. User Dashboard with multiple subjects
3. Uploading document
4. View previously generated question papers

Module 2: Text Extraction and Preprocessing - Text has to be extracted from input document. Preprocessing should be done on extracted text and make it suitable for transformer models.

1. Text Extraction using PyPDF2
2. Text Preprocessing using RegEx

Module 3: : MCQ Question Answer Generation - Fine-tune T5 Transformer model using SQuAD 1.1 Dataset for question generation. Generate answer by feeding preprocessed text and question into question-answer model.

Module 4: Distractor Generation - To create contextual distractors, utilize T5 distractor generation model to create relevant distractors for MCQ questions.

Module 5: Question Paper Generation - Place the generated questions into the question paper with a predefined template.

1. Generate PDF of question paper.
2. Store the PDF in database.
3. View and download previously generated question papers.

Architecture Diagram

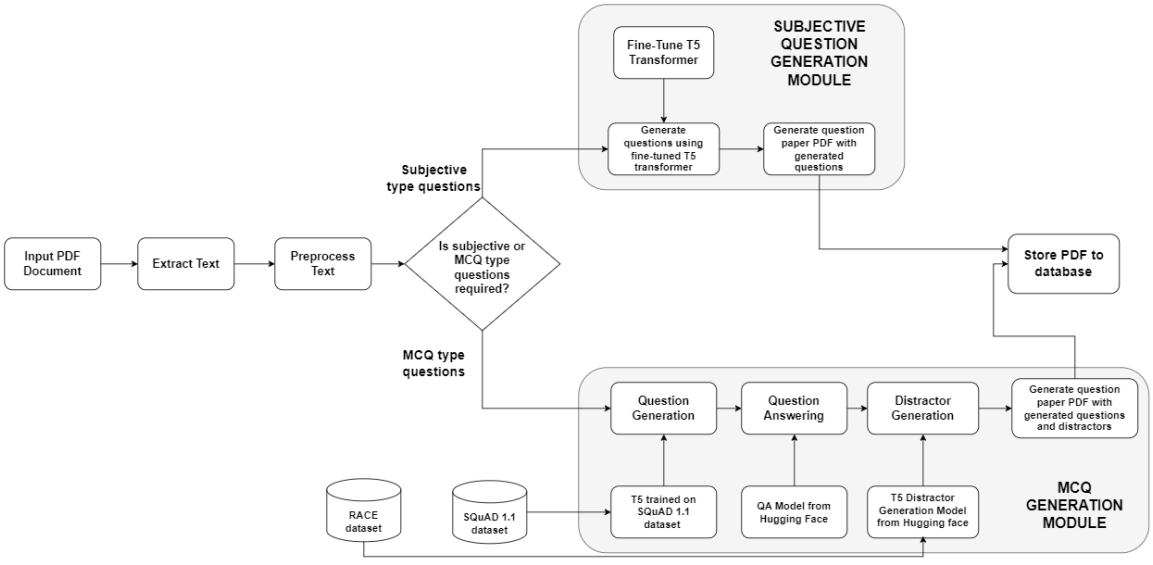


Figure: Architecture Diagram

Results

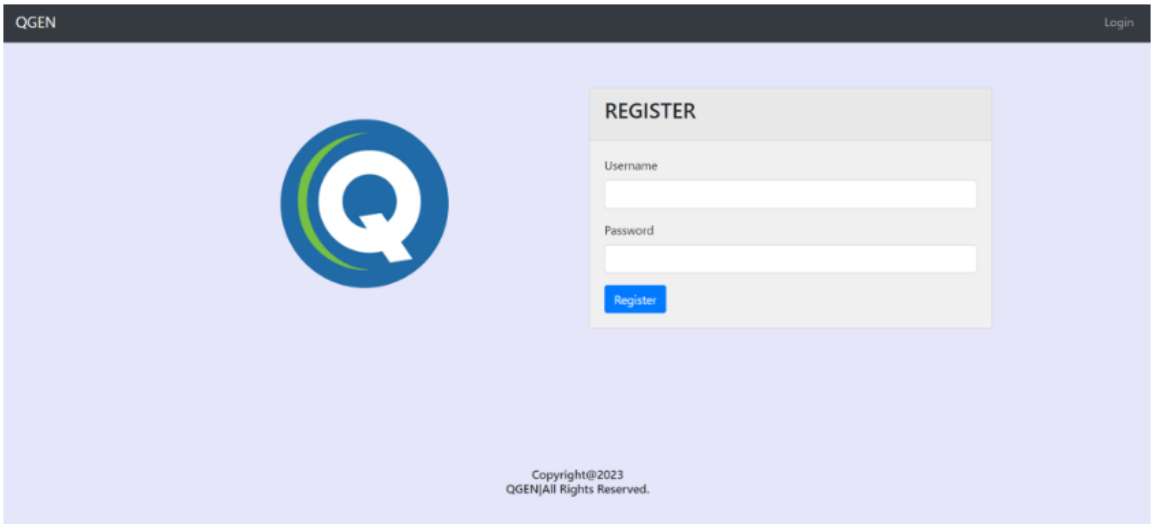


Figure: Register page

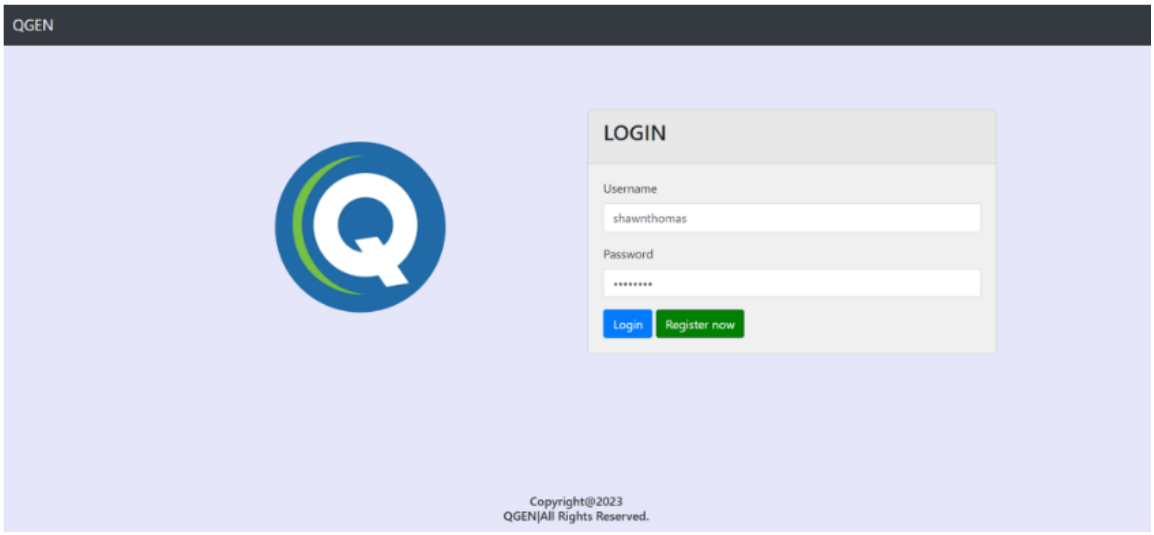


Figure: Login Page

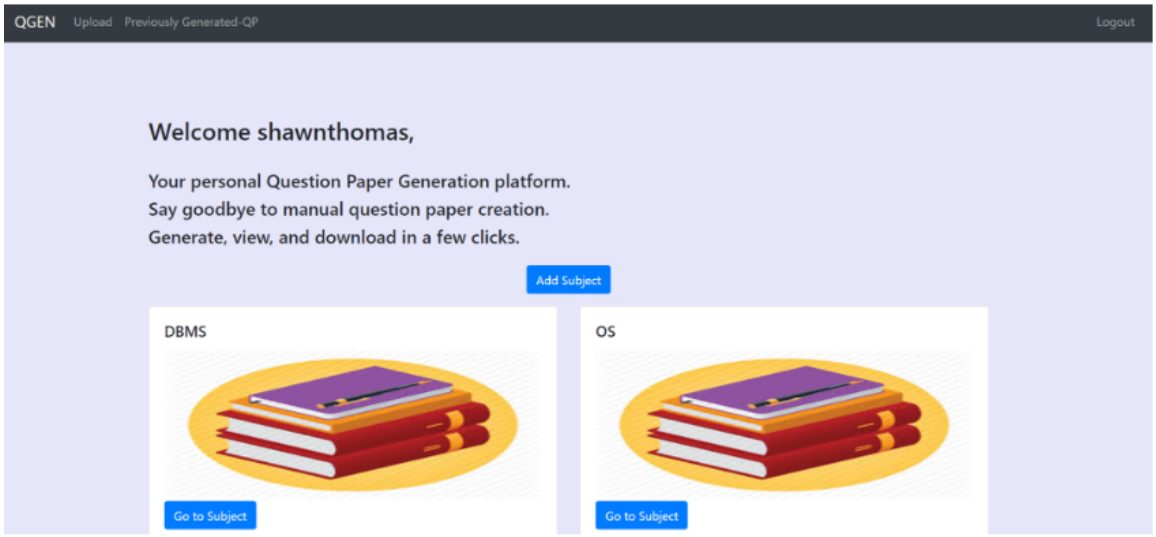


Figure: User Dashboard Page

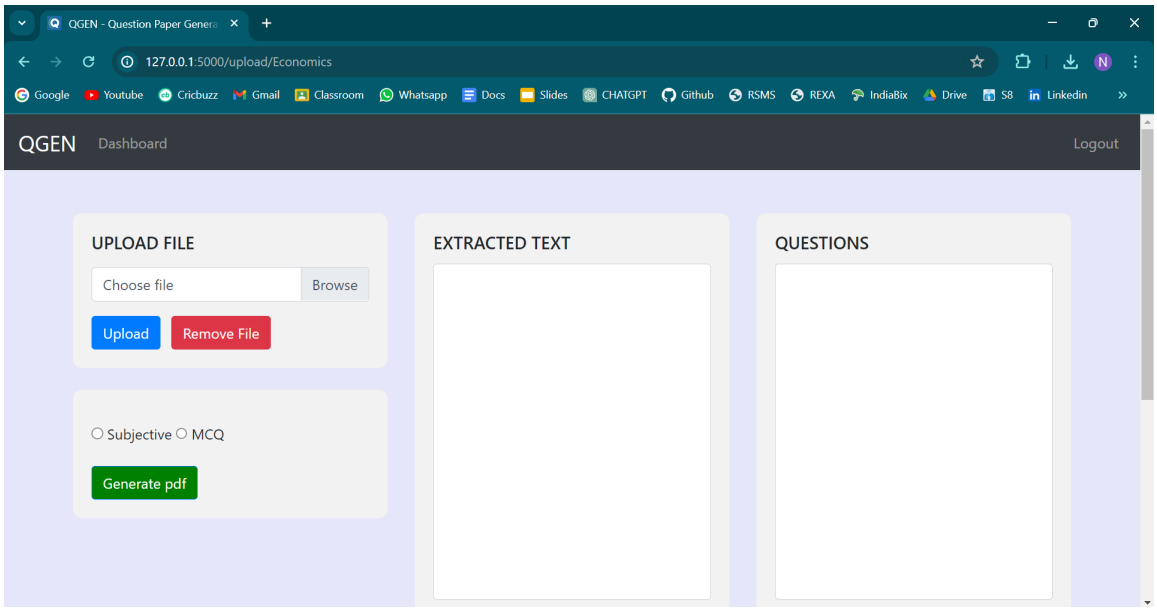


Figure: Upload Page

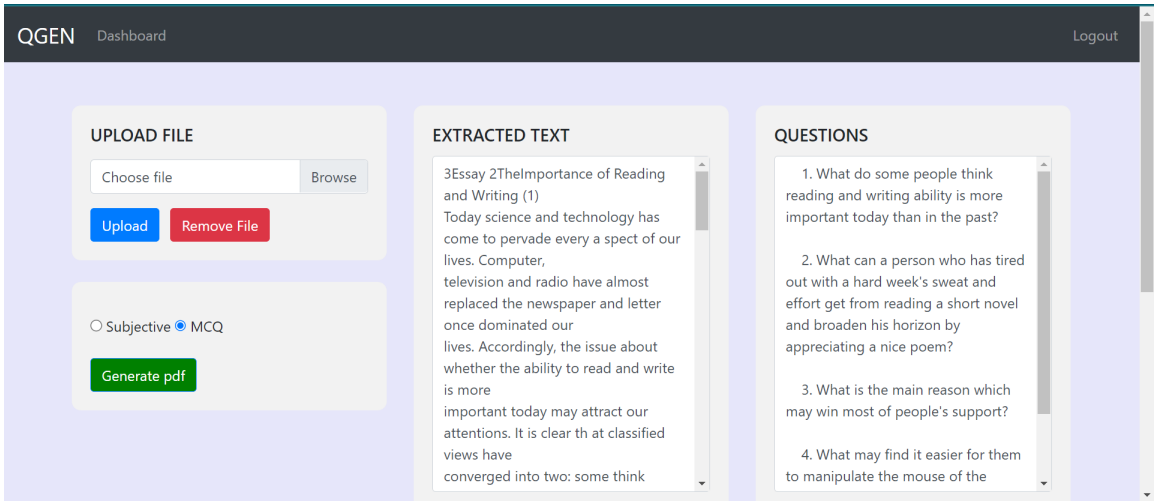
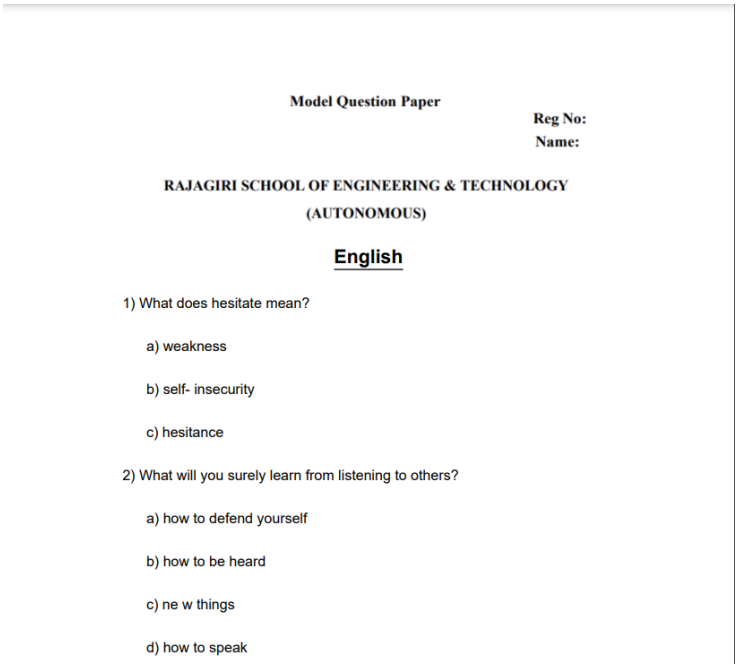


Figure: Upload Page after question generation



Model Question Paper

Reg No:
Name:

RAJAGIRI SCHOOL OF ENGINEERING & TECHNOLOGY
(AUTONOMOUS)

English

1) What does hesitate mean?

a) weakness

b) self- insecurity

c) hesitance

2) What will you surely learn from listening to others?

a) how to defend yourself

b) how to be heard

c) ne w things

d) how to speak

Figure: MCQ Question Paper

ANSWER KEY

1) self- insecurity

2) ne w things

3) listening part

4) your own thoughts

5) Establishing and developing effective communication

6) the listener

Figure: Answer Key

Model Question Paper

Reg No:
Name:

RAJAGIRI SCHOOL OF ENGINEERING & TECHNOLOGY

(AUTONOMOUS)

English

1) What is the main reason which may win most of people's support?

2) What do some people think reading and writing ability is more important today than in the past?

3) What can a person who has tired out with a hard week's sweat and effort get from reading a short novel and broaden his horizon by appreciating a nice poem?

4) What may find it easier for them to manipulate the mouse of the computer than to write things down by hand?

Figure: Subjective Question Paper

ANSWER KEY

1) if lack the reading and writing ability

2) relax themselves

3) lots of benefits and relief

4) children

Figure: Answer Key

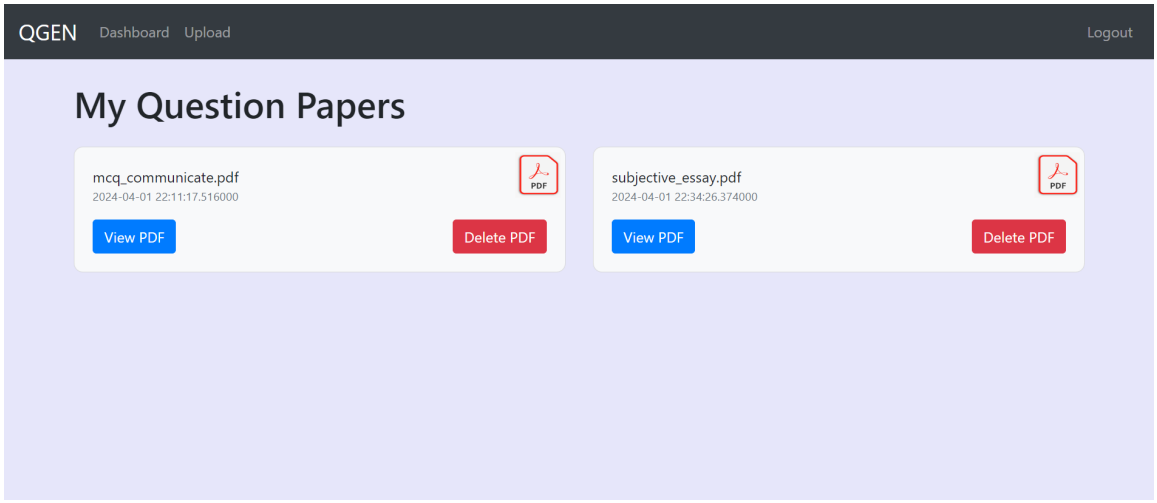


Figure: Question Paper Page

Work Division Among Team Members

1. Flask UI: Nevin, Rohan, Ronit, Sebin
2. Text Extraction and Preprocessing: Rohan, Sebin
3. MCQ Generation: Rohan and Nevin
4. Distractor Generation: Ronit and Sebin
5. Question Paper Generation: Ronit, Nevin

Conclusion

The process of manually creating questions is a tough one and so the system that is created helps to partially solve the difficulty of the task thus reducing human intervention and it is a cost and time effective system.

Future Scope

We hope to extend the application by:

1. Input file can be of any file format.
2. Equations and figures in the input file may be used to generate questions.
3. Setting the difficulty level of questions.
4. Allowing user to input number of questions he/she requires
5. Extend other aspects of Bloom's Taxonomy to generate more diverse questions.

References

[1] Vachev, K., Hardalov, M., Karadzhov, G. E., Georgiev, G,Koychev, I., Nakov, P. (2022). Leaf: Multiple-Choice Question Generation. arXiv (Cornell University). https://doi.org/10.48550/arxiv.2201.09012
[2] Lopez, Luis Enrico, Diane Kathryn Cruz, Jan Christian Blaise Cruz and Charibeth Ko Cheng. “Transformer-based End-to-End Question Generation.” ArXiv abs/2005.01107 (2020)
[3] Q. Zhou, N. Yang, F. Wei, C. Tan, H. Bao, and M. Zhou, “Neural question generation from text: A preliminary study,” ArXiv, vol. abs/1704.01792, 2017. [Online]. Available: https://api.semanticscholar.org/CorpusID:9745861
[4] D. R. Ch and S. K. Saha, “Automatic multiple choice question generation from text: A survey,” IEEE Transactions on Learning Technologies, vol. 13, pp. 14–25, 2020. [Online]. Available: https://api.semanticscholar.org/CorpusID:70335228

[5]M. Agarwal and P. Mannem, “Automatic gap-fill question generation from text books,” in Proceedings of the 6th Workshop on Innovative use of NLP for Build- ing Educational Applications, Hyderabad, AP, India, 2011, pp. 56–64.
[6] A. Y. Satria and T. Tokunaga, “Automatic generation of english reference question by utilising nonrestrictive relative clause,” in Proceedings of the 9th International Conference on Computer Supported Education, 2017, pp. 379–386
[7] Y. Liu, “Fine-tune BERT for extractive summarization,” arXiv preprint arXiv:1903.10318v2, Sep 2019
[8] R. Nadig, J. Ramanand, and P. Bhattacharyya, “Automatic evaluation of wordnet synonyms and hypernyms,” in Proceedings of ICON-2008, 2008. 24

Paper Publication (Accepted)

Rohan Jose Paul, Sangeetha Jamal, Sebin Bejoy, Ronit John Daniel and Nevin Aju, ”QGen: Automated Question Paper Generator”, was accepted and presented at the **5th INTERNATIONAL CONFERENCE ON INNOVATIVE TRENDS IN INFORMATION TECHNOLOGY IIIT Kottayam, Kerala, India** on 15th March, 2024.

The paper will be published soon on IEEE Xplore

Thank You!

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

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 (CO)

Course Outcome 1: Model and solve real world problems by applying knowledge across domains (Cognitive knowledge level: Apply).

Course Outcome 2: Develop products, processes or technologies for sustainable and socially relevant applications (Cognitive knowledge level: Apply).

Course Outcome 3: Function effectively as an individual and as a leader in diverse teams and to comprehend and execute designated tasks (Cognitive knowledge level: Apply).

Course Outcome 4: Plan and execute tasks utilizing available resources within timelines, following ethical and professional norms (Cognitive knowledge level: Apply).

Course Outcome 5: Identify technology/research gaps and propose innovative/creative solutions (Cognitive knowledge level: Analyze).

Course Outcome 6: Organize and communicate technical and scientific findings effectively in written and oral forms (Cognitive knowledge level: Apply).

Appendix C: CO-PO-PSO Mapping

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	PSO 3
CO 1	2	2	2	1	2	2	2	1	1	1	1	2	3		
CO 2	2	2	2		1	3	3	1	1		1	1		2	
CO 3									3	2	2	1			3
CO 4					2			3	2	2	3	2			3
CO 5	2	3	3	1	2							1	3		
CO 6					2			2	2	3	1	1			3

3/2/1: high/medium/low

JUSTIFICATIONS FOR CO-PO MAPPING

MAPPING	LOW/MEDIUM/ HIGH	JUSTIFICATION
100003/ CS722U.1-PO1	M	Knowledge in the area of technology for project development using various tools results in better modeling.
100003/ CS722U.1-PO2	M	Knowledge acquired in the selected area of project development can be used to identify, formulate, review research literature, and analyze complex engineering problems reaching substantiated conclusions.

100003/ CS722U.1-PO3	M	Can use the acquired knowledge in designing solutions to complex problems.
100003/ CS722U.1-PO4	M	Can use the acquired knowledge in designing solutions to complex problems.
100003/ CS722U.1-PO5	H	Students are able to interpret, improve and redefine technical aspects for design of experiments, analysis and interpretation of data, and synthesis of the information to provide valid conclusions.
100003/ CS722U.1-PO6	M	Students are able to interpret, improve and redefine technical aspects by applying contextual knowledge to assess societal, health and consequential responsibilities relevant to professional engineering practices.
100003/ CS722U.1-PO7	M	Project development based on societal and environmental context solution identification is the need for sustainable development.
100003/ CS722U.1-PO8	L	Project development should be based on professional ethics and responsibilities.
100003/ CS722U.1-PO9	L	Project development using a systematic approach based on well defined principles will result in teamwork.
100003/ CS722U.1-PO10	M	Project brings technological changes in society.

100003/ CS722U.1-PO11	H	Acquiring knowledge for project development gathers skills in design, analysis, development and implementation of algorithms.
100003/ CS722U.1-PO12	H	Knowledge for project development contributes engineering skills in computing & information gatherings.
100003/ CS722U.2-PO1	H	Knowledge acquired for project development will also include systematic planning, developing, testing and implementation in computer science solutions in various domains.
100003/ CS722U.2-PO2	H	Project design and development using a systematic approach brings knowledge in mathematics and engineering fundamentals.
100003/ CS722U.2-PO3	H	Identifying, formulating and analyzing the project results in a systematic approach.
100003/ CS722U.2-PO5	H	Systematic approach is the tip for solving complex problems in various domains.
100003/ CS722U.2-PO6	H	Systematic approach in the technical and design aspects provide valid conclusions.
100003/ CS722U.2-PO7	H	Systematic approach in the technical and design aspects demonstrate the knowledge of sustainable development.

100003/ CS722U.2-PO8	M	Identification and justification of technical aspects of project development demonstrates the need for sustainable development.
100003/ CS722U.2-PO9	H	Apply professional ethics and responsibilities in engineering practice of development.
100003/ CS722U.2-PO11	H	Systematic approach also includes effective reporting and documentation which gives clear instructions.
100003/ CS722U.2-PO12	M	Project development using a systematic approach based on well defined principles will result in better teamwork.
100003/ CS722U.3-PO9	H	Project development as a team brings the ability to engage in independent and lifelong learning.
100003/ CS722U.3-PO10	H	Identification, formulation and justification in technical aspects will be based on acquiring skills in design and development of algorithms.
100003/ CS722U.3-PO11	H	Identification, formulation and justification in technical aspects provides the betterment of life in various domains.
100003/ CS722U.3-PO12	H	Students are able to interpret, improve and redefine technical aspects with mathematics, science and engineering fundamentals for the solutions of complex problems.

100003/ CS722U.4-PO5	H	Students are able to interpret, improve and redefine technical aspects with identification formulation and analysis of complex problems.
100003/ CS722U.4-PO8	H	Students are able to interpret, improve and redefine technical aspects to meet the specified needs with appropriate consideration for the public health and safety, and the cultural, societal, and environmental considerations.
100003/ CS722U.4-PO9	H	Students are able to interpret, improve and redefine technical aspects for design of experiments, analysis and interpretation of data, and synthesis of the information to provide valid conclusions.
100003/ CS722U.4-PO10	H	Create, select, and apply appropriate techniques, resources, and modern engineering and IT tools for better products.
100003/ CS722U.4-PO11	M	Students are able to interpret, improve and redefine technical aspects by applying contextual knowledge to assess societal, health and consequential responsibilities relevant to professional engineering practices.
100003/ CS722U.4-PO12	H	Students are able to interpret, improve and redefine technical aspects for demonstrating the knowledge of, and need for sustainable development.
100003/ CS722U.5-PO1	H	Students are able to interpret, improve and redefine technical aspects, apply ethical principles and commit to

		professional ethics and responsibilities and norms of the engineering practice.
100003/ CS722U.5-PO2	M	Students are able to interpret, improve and redefine technical aspects, communicate effectively on complex engineering activities with the engineering community and with society at large, such as, being able to comprehend and write effective reports and design documentation, make effective presentations, and give and receive clear instructions.
100003/ CS722U.5-PO3	H	Students are able to interpret, improve and redefine technical aspects to demonstrate knowledge and understanding of the engineering and management principle in multidisciplinary environments.
100003/ CS722U.5-PO4	H	Students are able to interpret, improve and redefine technical aspects, recognize the need for, and have the preparation and ability to engage in independent and life-long learning in the broadest context of technological change.
100003/ CS722U.5-PO5	M	Students are able to interpret, improve and redefine technical aspects in acquiring skills to design, analyze and develop algorithms and implement those using high-level programming languages.
100003/ CS722U.5-PO12	M	Students are able to interpret, improve and redefine technical aspects and contribute their engineering skills in computing and information engineering domains like

		network design and administration, database design and knowledge engineering.
100003/ CS722U.6-PO5	M	Students are able to interpret, improve and redefine technical aspects and develop strong skills in systematic planning, developing, testing, implementing and providing IT solutions for different domains which helps in the betterment of life.
100003/ CS722U.6-PO8	H	Students will be able to associate with a team as an effective team player for the development of technical projects by applying the knowledge of mathematics, science, engineering fundamentals, and an engineering specialization to the solution of complex engineering problems.
100003/ CS722U.6-PO9	H	Students will be able to associate with a team as an effective team player to Identify, formulate, review research literature, and analyze complex engineering problems
100003/ CS722U.6-PO10	M	Students will be able to associate with a team as an effective team player for designing solutions to complex engineering problems and design system components.
100003/ CS722U.6-PO11	M	Students will be able to associate with a team as an effective team player, use research-based knowledge and research methods including design of experiments, analysis and interpretation of data.

100003/ CS722U.6-PO12	H	Students will be able to associate with a team as an effective team player, applying ethical principles and commit to professional ethics and responsibilities and norms of the engineering practice.
100003/ CS722U.1-PSO1	H	Students are able to develop Computer Science Specific Skills by modeling and solving problems.
100003/ CS722U.2-PSO2	M	Developing products, processes or technologies for sustainable and socially relevant applications can promote Programming and Software Development Skills.
100003/ CS722U.3-PSO3	H	Working in a team can result in the effective development of Professional Skills.
100003/ CS722U.4-PSO3	H	Planning and scheduling can result in the effective development of Professional Skills.
100003/ CS722U.5-PSO1	H	Students are able to develop Computer Science Specific Skills by creating innovative solutions to problems.
100003/ CS722U.6-PSO3	H	Organizing and communicating technical and scientific findings can help in the effective development of Professional Skills.