AI-Powered Question Paper Generation Based On Bloom's Taxonomy

B. Tech. Project Report

Submitted by

Yash Diwane	**112103038**
Yash Bongirwar	**112103026**
Devanshu Gupta	**112103035**

Under the guidance of

Dr. Yashodhara V. Haribhakta

COEP Technological University, Pune

DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

COEP TECHNOLOGICAL UNIVERSITY

(COEP TECH), PUNE - 5

May 2025

DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING,

COEP TECHNOLOGICAL UNIVERSITY, PUNE - 5

CERTIFICATE

Certified that this project titled, “AI-Powered Question Paper Generation Based On Bloom’s Taxonomy” has been successfully completed by

Yash Diwane	**112103038**
Yash Bongirwar	**112103026**
Devanshu Gupta	**112103035**

and is approved for the partial fulfillment of the requirements for the degree of “B. Tech. Computer Engineering”.

Dr. Yashodhara V. Haribhakta	**Dr Pradeep Deshmukh**
Project Guide	**Head**
Department of CSE	**Department of CSE**
COEP Tech Pune,	**COEP Tech Pune,**
Shivajinagar, Pune - 5.	**Shivajinagar, Pune - 5.**

9% Overall Similarity

The combined total of all matches, including overlapping sources, for each database.

Filtered from the Report

- Bibliography

Match Groups

- 55 Not Cited or Quoted 8%
  Matches with neither in-text citation nor quotation marks
- 5 Missing Quotations 1%
  Matches that are still very similar to source material
- 0 Missing Citation 0%
  Matches that have quotation marks, but no in-text citation
- 0 Cited and Quoted 0%
  Matches with in-text citation present, but no quotation marks

Top Sources

5%	Internet sources
6%	Publications
0%	Submitted works (Student Papers)

28/4/25

Abstract

The **AI-Powered Question Paper Generation Based On Bloom's Taxonomy** is a web-based application designed to automate the creation of customized question papers for educational institutions. By leveraging **Google Generative AI (Gemini)**, **Firebase**, and **Flask**, the system generates high-quality questions tailored to specific parameters such as subject, year, branch, topics, subtopics, question types, and Bloom's Taxonomy levels. The application also supports the upload of past question papers in PDF format, extraction of questions using **LlamaParse**, and storage in a hierarchical database for future use. The system's ability to export question papers in multiple formats (PDF, DOCX, TXT) ensures flexibility and convenience. This project aims to reduce the workload on educators, enhance the efficiency of question paper generation, and improve the overall quality of examinations. Preliminary results demonstrate the system's effectiveness in generating unique and balanced question papers, with positive feedback from educators.

Contents

List of Tables

Chapter 1

Introduction

1.1 Problem Overview

The manual process of creating question papers for examinations is time-consuming, labor-intensive, and prone to errors. Educators must ensure that the questions are relevant, balanced, and aligned with the curriculum, which requires significant effort and expertise. Additionally, the need to avoid repetition, maintain question quality, and adhere to specific guidelines (e.g., Bloom's Taxonomy, topic-wise distribution) further complicates the process. Existing tools and technologies for automated question generation are either too simplistic or lack the necessary features to address these challenges comprehensively. This project introduces an **AI-Powered Question Paper Generation Based On Bloom's Taxonomy**, a web-based application designed to automate and streamline the process of creating customized question papers.

1.2 Project Motivation

- **Reduce Workload**: Automate repetitive tasks to reduce the workload on educators.

- **Improve Efficiency**: Enhance the efficiency of question paper generation.
- **Ensure Quality**: Improve the overall quality of examinations by generating unique and balanced question papers.
- **Scalability**: Provide a scalable solution for educational institutions of all sizes.
- **Compliance with Bloom's Taxonomy Distribution**: In many academic institutions, educators are required to follow a specific percentage distribution of questions across Bloom's cognitive levels (e.g., Remembering, Understanding, Applying, Analyzing, Evaluating, Creating) while designing question papers. Manually maintaining this balance is tedious, time-consuming, and prone to inconsistencies. Our project was motivated by the need to automate this process — ensuring that generated question papers systematically meet the desired Bloom's level distributions without manual intervention, thereby improving both quality and compliance.

1.3 Objectives

- **Building the Question Bank**:
  - Extract questions from previous year papers using **LlamaParse**, a robust PDF parsing tool.
  - Organize extracted and generated questions hierarchically (Subject → Topic → Subtopic → Question) in a structured database (Firestore).
  - Capture metadata for each question including topic, subtopic, question type, and Bloom's Taxonomy level.
- **Automated Question Paper Generation**:

- Develop an intuitive web-based UI to collect user parameters such as subject, year, topics, subtopics, question type, difficulty level, and Bloom's level distribution.
- Use **Google Generative AI (Gemini)** to generate high-quality, contextually relevant questions based on selected parameters.
- Ensure even distribution of questions across different topics and Bloom's cognitive levels.
- Integrate a randomization mechanism to prevent repetition and generate unique question papers.

- **Retrieving and Adding Questions from the Question Bank**:
  - Fetch appropriate questions from the structured question bank based on user-defined parameters such as subject, topics, subtopics, question types, difficulty level, and Bloom's Taxonomy distribution.
- **Assessing Cognitive Skills (Bloom's Taxonomy Prediction)**:
  - Classify each generated and extracted question into appropriate Bloom's Taxonomy levels (Remember, Understand, Apply, Analyze, Evaluate, Create).
  - Fine-tune machine learning models (such as DeBERTa) for high-accuracy classification.
  - Ensure the generated paper adheres to the targeted cognitive skill distribution required by institutional guidelines.
- **Validation and Regeneration of Questions**:
  - Allow educators to review, edit, and regenerate questions if they are irrelevant, incorrect, or not matching the intended cognitive level.

  - Provide real-time feedback mechanisms during question generation to improve difficulty and quality dynamically.

- **Measuring and Visualizing Question Repetition Score from Previous Years**:
  - Calculate the semantic similarity between newly generated questions and previously stored questions in the past year paper (PYP) database.
  - Compute a "Question Repetition Score" that quantifies how many generated questions are semantically similar to historical questions.
  - Dynamically visualize the repetition score by changing the color of the associated text box:
    * **Green**: Very low similarity (mostly new questions).
    * **Yellow**: Moderate similarity (some overlap with previous papers).
    * **Red**: High similarity (significant reuse from past papers).
  - Provide real-time feedback to educators, helping them ensure sufficient novelty in generated question papers.

1.4 Scope

- **Educational Institutions**: Designed for educational institutions to streamline the question paper creation process.
- **Multiple Subjects**: Supports multiple subjects, branches, and question types.
- **Scalability**: The application is scalable and can handle large question banks.

- **Fairness and Consistency**: Ensures fairness and consistency in assessments by generating unique and balanced question papers.

1.5 Introduction to Bloom's Taxonomy and Its Importance in Question Paper Design

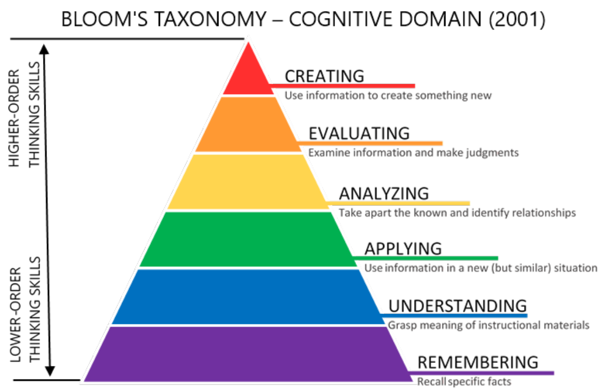

Figure 1.1: Bloom's Taxonomy Pyramid: Hierarchical Levels of Cognitive Skills

Bloom's Taxonomy, introduced by Benjamin Bloom and collaborators in 1956 [4], is a hierarchical model used to classify educational learning objectives into levels of complexity and specificity. It divides cognitive skills into six ascending levels:

- **Remember**: Recall facts and basic concepts.
- **Understand**: Explain ideas or concepts.
- **Apply**: Use information in new situations.
- **Analyze**: Draw connections among ideas.
- **Evaluate**: Justify a decision or course of action.
- **Create**: Produce new or original work.

In many academic institutions, examination guidelines require educators to design question papers with a specific distribution of questions across these Bloom's levels. For instance, 30% of the paper might assess 'Remember' and 'Understand', 40% might target 'Apply' and 'Analyze', and 30% could focus on higher-order thinking skills like 'Evaluate' and 'Create'.

Manually ensuring this cognitive balance during paper setting is labor-intensive, error-prone, and subjective. It often requires multiple iterations and careful cross-verification.

Our system automates this entire process by classifying each question into the correct Bloom's Taxonomy level and assembling question papers that meet institution-specified Bloom's level distributions. This ensures balanced assessments, enhances exam quality, and saves considerable educator time.

1.6 Significance

- **Empower Educators**: Empowers educators to focus on higher-value activities like curriculum design and student engagement.
- **Fair Assessments**: Ensures fairness and consistency in assessments by generating unique and balanced question papers.
- **Research Contribution**: Contributes to the growing body of research on AI-driven educational tools.
- **Scalable Solution**: Provides a scalable and customizable solution for educational institutions.

Chapter 2

Literature Review and Research Gaps

The advent of the Transformer architecture by Vaswani et al. [1] marked a significant shift in natural language processing (NLP). By introducing self-attention mechanisms, the model eliminated the need for recurrent neural networks, enabling greater parallelization and efficiency in training, which led to state-of-the-art results in machine translation.

Building upon this foundation, Devlin et al. [2] developed BERT, a deep bidirectional Transformer model pre-trained using Masked Language Modeling (MLM) and Next Sentence Prediction (NSP). BERT's ability to consider context from both directions allowed it to achieve superior performance across multiple NLP tasks, establishing transfer learning as a dominant approach in the field.

He et al. [3] further enhanced Transformer-based models by introducing DeBERTa, which incorporated disentangled attention mechanisms and an improved mask decoder. This architecture separated content and positional information, leading to better dependency modeling and achieving top scores on benchmarks like GLUE and SuperGLUE.

In the educational domain, Bloom et al. [4] proposed a taxonomy categorizing cognitive skills into a hierarchical structure ranging from basic recall to critical evaluation. This framework has been instrumental in curriculum development and assessment design. Bloom's Taxonomy continues to serve as a cornerstone for designing and evaluating educational assessments, ensuring a structured progression from simple recall to complex creation tasks. Krathwohl [5] later revised this taxonomy, introducing a two-dimensional framework that combined cognitive processes with knowledge types, thereby enhancing its applicability in instructional design.

Recent studies have explored the integration of large language models (LLMs) into educational contexts. Scaria et al. [6] evaluated LLMs like PaLM 2 and Gemini Pro for automated question generation across different Bloom's taxonomy levels. Their findings indicated that prompt engineering significantly affects question quality, and LLMs can generate valid questions across various cognitive levels.

Chindukuri and Sivanesan [7] applied transfer learning techniques to classify educational questions based on Bloom's taxonomy using the NCERT dataset. Their approach effectively addressed class imbalance issues and improved classification accuracy, contributing to scalable educational content analysis.

Sharma et al. [8] proposed a deep learning pipeline for automatic question generation and classification aligned with Bloom's taxonomy. Their method facilitated seamless integration with educational platforms, ensuring that generated questions were consistent with learning objectives.

In evaluating LLM capabilities, Akter et al. [9] benchmarked Gemini Pro against GPT-3.5 Turbo across various NLP tasks. While Gemini Pro performed competitively overall, limitations were noted in mathematical reasoning, suggesting areas for further model refinement.

Wang and Zhao [10] assessed Gemini's commonsense reasoning using multimodal datasets. Their research revealed strong performance in language-only tasks but identified challenges in multimodal emotional reasoning and integrating visual and textual information.

Anil et al. [11] introduced the Gemini model family (Ultra, Pro, Nano), showcasing models capable of processing text, image, audio, and video modalities. These models outperformed existing baselines on multimodal benchmarks and were optimized for diverse computational environments, setting new standards in multimodal AI.

Research Gaps

Despite significant advancements in NLP and educational technologies, several research gaps persist:

- **Alignment with Educational Objectives**: While LLMs have demonstrated capabilities in question generation, ensuring that these questions align accurately with specific educational objectives and Bloom's taxonomy levels remains a challenge.
- **Quality and Validity of Generated Questions**: The quality and pedagogical validity of automatically generated questions need further evaluation to ensure they meet educational standards.
- **Integration into Educational Systems**: There is a lack of comprehensive systems that seamlessly integrate automated question generation and classification into existing educational platforms, facilitating ease of use for educators.
- **Handling of Multimodal Inputs**: Current models show limitations in processing and reasoning over multimodal inputs, which is essential for

comprehensive educational assessments that include visual and textual information.

Addressing these gaps is crucial for developing an AI-Powered Question Paper Generation Based On Bloom's Taxonomy that not only automates the creation of examination papers but also ensures their relevance, quality, and alignment with educational standards.

Chapter 3

Problem Statement

The manual process of creating question papers for examinations is time-consuming, labor-intensive, and prone to errors. Educators must ensure that the questions are relevant, balanced, and aligned with the curriculum, which requires significant effort and expertise. Additionally, the need to avoid repetition, maintain question quality, and adhere to specific guidelines (e.g., Bloom's Taxonomy, topic-wise distribution) further complicates the process. Existing tools and technologies for automated question generation are either too simplistic or lack the necessary features to address these challenges comprehensively. This project aims to develop an **AI-Powered Question Paper Generation Based On Bloom's Taxonomy** that automates the entire process of question paper creation, from question generation to final export, while ensuring customization, scalability, and ease of use.

3.1 Manual Process Challenges

- **Time-Consuming**: Manual question paper generation requires significant time and effort.
- **Error-Prone**: The process is prone to errors and inconsistencies.
- **Question Relevance**: Ensuring that questions are relevant and aligned

with the curriculum is challenging.

- **Balance and Fairness**: Maintaining a balanced distribution of questions across topics and difficulty levels is difficult.

3.2 Existing Tools Limitations

- **Rigid Systems**: Rule-based systems are rigid and lack adaptability.
- **Limited Features**: AI-powered tools focus on individual question generation, not end-to-end solutions.
- **PDF Extraction**: Tools that support PDF uploads struggle with accurate text extraction and formatting.
- **User-Friendliness**: Most systems lack user-friendly interfaces and customization options.

3.3 Need for a Comprehensive Solution

- **Automation**: Automate the entire process of question paper creation.
- **AI Integration**: Integrate advanced AI capabilities for question classification and difficulty adjustment.
- **User-Friendly Interface**: Provide a user-friendly interface for customization and ease of use.
- **Scalability**: Ensure scalability to handle large question banks and multiple users.

Chapter 4

Proposed Methodology/ Solution

4.1 Core Features

4.1.1 AI-Powered Question Generation

- **Google Generative AI (Gemini)**: The system leverages Google Generative AI (Gemini) to generate high-quality questions based on user-defined parameters. These parameters include subject, year, branch, topics, subtopics, question types (MCQ, Short Answer, Long Answer), and Bloom's Taxonomy levels. The AI model is trained to understand the context and generate questions that are relevant, balanced, and aligned with the curriculum. This feature significantly reduces the time and effort required for question paper creation, as educators no longer need to manually draft questions.

- **Multiple Question Types**: The system supports a variety of question types, including Multiple Choice Questions (MCQs), Short Answer Questions, and Long Answer Questions. This ensures that the generated question papers are comprehensive and cater to different assessment needs. The AI model is capable of generating questions that test various cognitive skills, from basic recall to higher-order thinking.

- **Even Distribution**: The system ensures an even distribution of questions across selected topics and subtopics. This is achieved by analyzing the syllabus and dividing the questions proportionally based on the weightage of each topic. This feature ensures that the question paper is balanced and covers all important areas of the curriculum.

- **Randomization**: To ensure fairness and prevent repetition, the system incorporates randomization in question generation. Each time a question paper is generated, the system selects questions randomly from the question bank, ensuring that no two question papers are identical. This feature is particularly useful for institutions that conduct multiple examinations throughout the year.

4.1.2 Question Bank Management

- **Hierarchical Database**: The system uses a hierarchical database (Firestore) to store questions in a structured manner. The database is organized under the structure: **Subject** → **Topics** → **Subtopics** → **Questions**. This hierarchical structure allows for easy navigation and retrieval of questions, making it simple for educators to manage large question banks.

- **Edit and Regenerate**: Educators have the flexibility to edit, regenerate, or delete questions in the question bank. If a question is found to be irrelevant or incorrect, it can be easily modified or replaced. The system also allows educators to regenerate questions based on updated parameters, ensuring that the question bank remains up-to-date and relevant.

- **Search and Filter**: The system provides robust search and filter functionality, allowing educators to quickly find specific questions based on criteria such as topic, subtopic, question type, and Bloom's Taxonomy level. This feature saves time and ensures that educators can easily locate the questions they need for their question papers.

- **Version Control**: The system tracks version control for questions, enabling educators to track changes over time. Each time a question is modified, the system creates a new version while retaining the previous version. This feature is particularly useful for auditing purposes and ensures that educators can revert to earlier versions if needed.

4.1.3 Syllabus Management

- **Syllabus Upload**: Educators can upload syllabi for different subjects, years, and branches. The system supports various file formats, including PDF and DOCX, making it easy for educators to upload their syllabi. Once uploaded, the syllabus is stored in the system and can be accessed at any time.

- **AI-Based Extraction**: The system uses Google Generative AI to extract topics and subtopics from the syllabus text. The AI model analyzes the syllabus and identifies key areas that need to be covered in the question paper. This feature ensures that the generated question papers are aligned with the curriculum and cover all important topics.

- **Hierarchical Storage**: Similar to the question bank, the syllabi are stored in a hierarchical structure for easy reference. The syllabi are organized under the structure: **Subject** → **Year** → **Branch** → **Syllabus**. This hierarchical storage ensures that educators can easily navigate and retrieve syllabi based on specific criteria.

- **View and Edit**: Educators can view and edit syllabi directly within the application. If any changes are made to the syllabus, the system automatically updates the question generation parameters to reflect the new syllabus. This feature ensures that the question papers remain aligned with the latest curriculum.

4.1.4 PDF Upload and Extraction

- **PDF Upload**: Educators can upload past question papers in PDF format. The system supports batch processing, allowing multiple PDFs to be uploaded simultaneously. This feature is particularly useful for institutions that have a large repository of past question papers.
- **LlamaParse**: The system uses LlamaParse to extract text from PDFs. LlamaParse is a powerful tool that ensures accurate extraction of text, even from complex PDFs with tables, images, and other formatting elements. Once the text is extracted, the system processes it using Google Generative AI to classify and organize the questions.
- **Automatic Storage**: The extracted questions are automatically added to the question bank for future use. The system ensures that the questions are stored in the correct hierarchical structure, making it easy for educators to retrieve them later. This feature eliminates the need for manual data entry and ensures that the question bank is continuously updated with new questions.
- **Batch Processing**: The system supports batch processing for multiple PDFs. This feature is particularly useful for institutions that have a large repository of past question papers. Educators can upload multiple PDFs at once, and the system will automatically extract and process the questions in the background.

4.1.5 Export Functionality

- **Multiple Formats**: The system allows educators to export question papers in multiple formats, including PDF, DOCX, and TXT. This ensures that the question papers can be easily shared and printed in the desired format. The system also provides customizable layouts, allowing educators to choose the format that best suits their needs.

- **Customizable Layouts**: Educators can customize the layout of the question paper before exporting. The system provides various templates and formatting options, including font size, line spacing, and margin settings. This feature ensures that the exported question papers are professional and easy to read.

- **Preview**: Before exporting, educators can preview the question paper to ensure that it meets their requirements. The preview feature allows educators to make any last-minute changes before finalizing the question paper. This feature ensures that the final output is error-free and meets the desired standards.

- **Headers and Footers**: The system includes headers, footers, and page numbers in the exported documents. This feature ensures that the question papers are well-organized and easy to navigate. Educators can also customize the headers and footers to include additional information, such as the institution's name or the examination date.

4.2 Advanced Features

4.2.1 Transfer Learning for Question Classification

- **Fine-Tuning**: The system fine-tunes pre-trained models on a custom dataset to improve the accuracy of question classification. The fine-tuning process involves training the model on a dataset of questions that have been manually classified according to Bloom's Taxonomy levels. This ensures that the model can accurately classify questions into the appropriate cognitive levels.

- **Bloom's Taxonomy**: The system classifies questions into Bloom's Taxonomy levels (Remember, Understand, Apply, Analyze, Evaluate, Create). This classification ensures that the question papers are balanced and test a range of cognitive skills. The system also provides educators with insights into the distribution of questions across different Bloom's Taxonomy levels.

- **Real-Time Classification**: The system supports real-time classification during question generation. As questions are generated, the system automatically classifies them into the appropriate Bloom's Taxonomy levels. This feature ensures that the question papers are balanced and aligned with the curriculum.

- **High Accuracy**: The system achieves high accuracy in classifying questions into cognitive levels. This is achieved through the use of advanced AI models and fine-tuning techniques. The high accuracy ensures that the generated question papers are of high quality and meet the desired standards.

4.2.2 Adaptive Difficulty Adjustment

- **Dynamic Prompting**: The system uses dynamic prompting to generate more challenging or creative questions. The AI model adjusts the difficulty level of the questions based on the user's input. For example, if the user requests a more challenging question paper, the system will generate questions that test higher-order thinking skills.

- **Regeneration Count**: The system adjusts the difficulty of questions based on the regeneration count and user feedback. If a question is regenerated multiple times, the system will adjust the difficulty level to ensure that the question is appropriate for the intended audience. This feature ensures that the question papers are balanced and meet the desired difficulty level.

- **Diversity**: The system ensures diversity in question content and context. This is achieved by generating questions that cover a wide range of topics and subtopics. The system also ensures that the questions are varied in terms of format and difficulty level. This feature ensures that the question papers are comprehensive and test a range of skills.

- **Real-Time Feedback**: The system provides real-time difficulty feedback to users. As questions are generated, the system provides feedback on the difficulty level and suggests adjustments if needed. This feature ensures that the question papers are balanced and meet the desired standards.

4.3 Technical Specifications

4.3.1 AI and Machine Learning

- **Google Generative AI (Gemini)**: The system uses Google Generative AI (Gemini) for question generation, classification, and extraction. Gemini is a state-of-the-art AI model that is capable of generating high-quality questions based on user-defined parameters. The model is trained on a large dataset of educational content, ensuring that the generated questions are relevant and aligned with the curriculum.

- **LlamaParse**: The system uses LlamaParse to extract structured text from PDFs. LlamaParse is a powerful tool that ensures accurate extraction of text, even from complex PDFs with tables, images, and other formatting elements. Once the text is extracted, the system processes it using Google Generative AI to classify and organize the questions.

- **Transfer Learning**: The system fine-tunes pre-trained models on a custom dataset for Bloom's Taxonomy classification. The fine-tuning process involves training the model on a dataset of questions that have been manually classified according to Bloom's Taxonomy levels. This ensures that the model can accurately classify questions into the appropriate cognitive levels.

- **State-of-the-Art Performance**: The system achieves state-of-the-art performance on educational datasets. This is achieved through the use of advanced AI models and fine-tuning techniques. The high accuracy ensures that the generated question papers are of high quality and meet the desired standards.

4.3.2 Database and Storage

- **Firestore**: The system uses Firestore, a hierarchical NoSQL database, for storing questions, syllabi, and user data. Firestore ensures real-time synchronization and scalability, making it ideal for handling large datasets. The hierarchical structure of Firestore allows for easy navigation and retrieval of data.

- **Cloud Storage**: The system uses cloud storage to store uploaded PDFs and generated question papers. Cloud storage ensures that the data is securely stored and can be accessed from anywhere. The system also provides backup and recovery options, ensuring that the data is safe and can be restored in case of any issues.

- **Real-Time Sync**: The system ensures real-time synchronization across devices. Any changes made to the database are immediately reflected across all devices, ensuring that the data is always up-to-date. This feature is particularly useful for institutions that have multiple users accessing the system simultaneously.

- **Data Security**: The system provides data security with Firebase Authentication. Firebase Authentication ensures that only authorized users can access the system. The system also provides role-based access control, ensuring that users can only access the data that is relevant to them.

4.3.3 APIs and Integrations

- **Firebase REST API**: The system uses the Firebase REST API for user authentication and session management. The Firebase REST API ensures that user sessions are securely managed and that only authorized users can access the system. The API also provides rate limiting, ensuring that the system is not overwhelmed by excessive requests.

- **Google OAuth**: The system integrates Google OAuth for seamless user authentication. Google OAuth allows users to sign in to the system using their Google accounts, eliminating the need for separate login credentials. This feature ensures a smooth user experience and reduces the risk of password-related issues.

- **API Key Management**: The system implements API key management to ensure secure access to AI services. API keys are used to authenticate requests to the AI services, ensuring that only authorized applications can access the system. The system also provides rate limiting, ensuring that the AI services are not overwhelmed by excessive requests.

- **Input Validation**: The system validates user inputs to prevent SQL injection and XSS attacks. Input validation ensures that the system is protected against common security vulnerabilities. The system also provides error handling, ensuring that any invalid inputs are flagged and corrected before processing.

Chapter 5

Experimental Setup

5.1 System Architecture

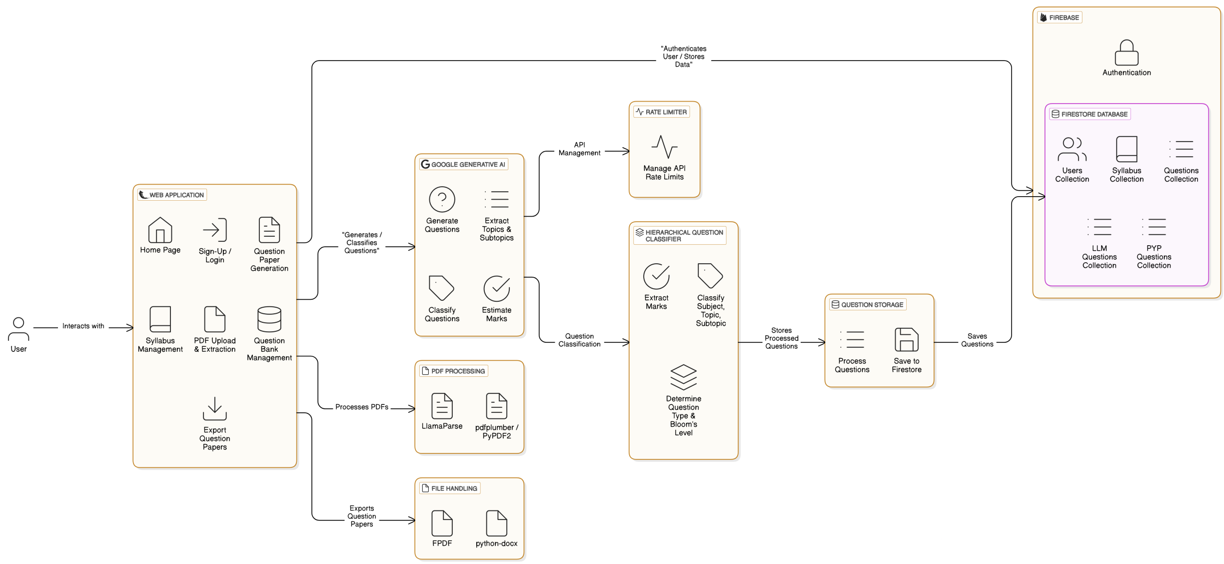

Figure 5.1: Overall System Architecture of the AI-Powered Question Paper Generation System

The system architecture follows a modular and scalable design, integrating cloud services, AI models, and database components to automate the question paper generation workflow efficiently.

5.1.1 Three-Tier Architecture

- **Presentation Layer (Frontend)**: Built using HTML, CSS, and JavaScript. The frontend offers modules for user authentication, syllabus management, question paper generation, PDF upload, question bank management, and paper export functionalities.
- **Application Layer (Backend)**: Developed using Flask, the backend handles user interactions, business logic, and communication with AI services and the database. It includes components for AI-based question generation and classification, PDF processing (using LlamaParse and PyPDF2), API rate limiting for external services, and file handling for export generation.
- **Data Layer (Database)**: Uses Firestore for hierarchical data storage. Separate collections are maintained for Users, Syllabi, LLM-Generated Questions, and Previous Year Paper (PYP) Questions. Structured storage ensures fast retrieval and easy scalability.

5.1.2 Client-Server Model

- **Frontend**: Interacts with the backend via RESTful APIs. It allows users to input paper generation parameters, manage syllabi, upload PDFs, and download the generated papers.
- **Backend**: Processes requests, invokes Google Generative AI for question generation and Bloom's classification, processes PDFs through LlamaParse and PDFPlumber, classifies extracted questions into topics, subtopics, and Bloom's levels, and stores results into Firestore. Also handles API rate limiting to manage service stability.
- **Separation of Concerns**: The architecture follows strict separation be-

tween UI, business logic, AI interaction, and database access, ensuring modularity, easy maintenance, and scalability.

5.1.3 Key Functional Components

- **Google Generative AI Integration**: Used for generating new questions, extracting syllabus topics, and classifying questions based on Bloom's Taxonomy levels.
- **PDF Processing Layer**: Incorporates tools like LlamaParse, pdfplumber, and PyPDF2 to extract structured questions from uploaded PDFs.
- **Hierarchical Question Classifier**: Classifies each question into Bloom's level, type (MCQ, Short, Long), and estimates marks if missing.
- **Question Storage**: Organized in Firestore with distinct collections for LLM-generated and PYP questions for efficient management and retrieval.
- **File Export Handling**: Uses libraries like FPDF and python-docx to export generated question papers in user-preferred formats (PDF, DOCX, TXT).
- **Rate Limiter Module**: Ensures controlled access to AI APIs to maintain service reliability under high request loads.

5.2 Hardware Specifications

5.2.1 Development Environment

- **Processor**: Intel Core i5 or higher. A powerful processor ensures smooth performance during development and testing.

- **RAM**: 8 GB or higher. Sufficient RAM is required to handle large datasets and AI models.
- **Storage**: 256 GB SSD or higher. Adequate storage is necessary for storing code, datasets, and generated question papers.

5.2.2 Deployment Environment

- **Cloud Infrastructure**: Hosted on Google Cloud Platform (GCP) or Amazon Web Services (AWS). Cloud hosting ensures scalability, reliability, and high availability.
- **Server Specifications**: 4 vCPUs, 16 GB RAM, 100 GB SSD. These specifications ensure that the system can handle multiple users and large question banks efficiently.

5.3 Software Specifications

5.3.1 Frontend

- **HTML5, CSS3, JavaScript**: Used for structuring, styling, and interactivity. These technologies ensure a responsive and user-friendly interface.
- **Bootstrap**: Provides pre-designed UI components and layouts. Bootstrap simplifies the development process and ensures consistency across different devices.
- **User Authentication**: Handles login, sign-up, and Google Sign-In. Secure authentication ensures that only authorized users can access the system.

- **Question Paper Generation**: Customizable parameters and preview. Educators can specify parameters like subject, year, branch, and question types to generate customized question papers.

5.3.2 Backend

- **Flask**: Python web framework for handling requests and business logic. Flask is lightweight and easy to use, making it ideal for developing the backend of the system.

- **Firebase SDK**: Integrates Firebase for authentication and Firestore. Firebase provides a robust backend infrastructure, ensuring secure and scalable data storage.

- **Google Generative AI API**: Used for question generation and classification. The API leverages advanced AI models to generate high-quality questions tailored to specific parameters.

- **LlamaParse API**: Extracts text from PDFs. LlamaParse ensures accurate extraction of questions from past question papers, making it easier to build the question bank.

5.3.3 Database

- **Firestore**: Hierarchical NoSQL database for storing questions, syllabi, and user data. Firestore ensures real-time synchronization and scalability, making it ideal for handling large datasets.

- **Real-Time Sync**: Ensures data consistency across devices. Real-time synchronization ensures that any changes made to the database are immediately reflected across all devices.

- **Scalability**: Handles large datasets and multiple users. Firestore is designed to scale seamlessly, ensuring that the system can handle growing data and user demands.
- **Data Security**: Provides role-based access control and data encryption. Firestore ensures that only authorized users can access sensitive data, ensuring data security and privacy.

5.3.4 AI Services

- **Google Generative AI (Gemini)**: Used for question generation and classification. Gemini leverages advanced AI models to generate high-quality questions tailored to specific parameters.
- **LlamaParse**: Extracts structured text from PDFs. LlamaParse ensures accurate extraction of questions from past question papers, making it easier to build the question bank.
- **Transfer Learning**: Fine-tunes pre-trained models for Bloom's Taxonomy classification. Transfer learning ensures that the system can classify questions into cognitive levels with high accuracy.
- **State-of-the-Art Performance**: Achieves state-of-the-art performance on educational datasets. The system leverages the latest advancements in AI to ensure high-quality question generation and classification.

5.4 Security and Cryptographic Mechanisms

5.4.1 User Authentication

- **Firebase Authentication**: Provides secure login and sign-up. Firebase Authentication ensures that only authorized users can access the system.

- **Google Sign-In**: Enables seamless user authentication. Google Sign-In simplifies the authentication process, ensuring a smooth user experience.

- **Session Management**: Manages user sessions with Flask session management. Session management ensures that users remain authenticated during their session, enhancing security.

5.4.2 Data Encryption

- **HTTPS**: Encrypts data transmitted between the client and server. HTTPS ensures that data is transmitted securely, preventing unauthorized access.

- **AES-256**: Encrypts data at rest in Firestore. AES-256 ensures that data stored in the database is secure and cannot be accessed without proper authorization.

- **Password Hashing**: Uses bcrypt for securely hashing user passwords. Password hashing ensures that user passwords are stored securely, preventing unauthorized access.

5.4.3 Access Control

- **Role-Based Access Control (RBAC)**: Restricts access based on user roles (Admin, Educator). RBAC ensures that only authorized users can access specific features and data.

- **Firestore Security Rules**: Ensures users can only access their own data. Firestore Security Rules provide an additional layer of security, ensuring data privacy and integrity.

5.4.4 API Security

- **API Key Management**: Uses Google API keys for accessing AI services. API key management ensures that only authorized applications can access the system's APIs.
- **Rate Limiting**: Implements rate limiting to prevent abuse. Rate limiting ensures that the system is not overwhelmed by excessive requests, ensuring smooth performance.
- **Input Validation**: Validates user inputs to prevent SQL injection and XSS attacks. Input validation ensures that the system is protected against common security vulnerabilities.

Chapter 6

Results and Discussion

The goal of the project was to build an end-to-end system capable of generating high-quality, customized question papers aligned with Bloom's Taxonomy cognitive skill levels. In this section, we discuss how the project objectives were fulfilled and present the corresponding evaluation results.

6.1 Building the Question Bank

The system successfully built a rich, structured question bank through two major pipelines:

- **Extraction from Previous Year Papers (PYPs)**: Using **LlamaParse** combined with PDF processing libraries (pdfplumber, PyPDF2), questions were accurately extracted from scanned or structured PDFs. Metadata such as subject, topic, subtopic, question type, and estimated marks were also extracted.

- **Hierarchical Organization**: Extracted and LLM-generated questions were organized into Firestore's hierarchical structure: Subject $\rightarrow$ Topic $\rightarrow$ Subtopic $\rightarrow$ Question. Metadata tagging including Bloom's Taxonomy level ensured efficient retrieval for future use.

6.2 Automated Question Paper Generation

An intuitive web application frontend was developed where users could specify parameters such as:

- Subject, Branch, Year
- Topics and Subtopics
- Desired Question Types (MCQ, Short, Long)
- Difficulty Level
- Bloom's Level Distribution (customizable percentage splits)

The backend, powered by **Google Generative AI (Gemini)**, generated high-quality questions dynamically based on these parameters. To ensure paper diversity and balance:

- Randomization mechanisms were incorporated to prevent repetitive question patterns.
- Dynamic distribution logic maintained topic balance and Bloom's level proportion in each generated paper.
- Export functionality supported multiple formats (PDF, DOCX) via `FPDF` and `python-docx`.

6.3 Retrieving and Adding Questions from the Question Bank

The system also allowed integration of pre-existing questions into newly generated papers:

- Questions were fetched intelligently based on user-defined parameters (subject, topics, Bloom's levels, difficulty).
- The system merged LLM-generated questions with PYPs to ensure a balanced mix of new and traditional questions.
- An optional setting allowed users to control the percentage of reused (previously extracted) questions.

6.4 Assessing Cognitive Skills (Bloom's Taxonomy Prediction)

A major contribution of the system is the accurate classification of questions into Bloom's Taxonomy levels.

Using a fine-tuned `Microsoft/deberta-v3-base` model:

- **Dataset Size**: 11,976 questions annotated with Bloom's levels.
- **Training Strategy**: 5-fold cross-validation was employed to ensure robustness.
- **Performance Metrics**:
  - **Accuracy:** 0.8798
  - **Precision:** 0.8793
  - **Recall:** 0.8798
  - **F1-Score:** 0.8789

6.4.1 Detailed Classification Report

The classification report reveals insightful patterns in the model's performance across different Bloom's Taxonomy levels:

Class	Precision	Recall	F1-Score	Support
0 (Remember)	0.80	0.86	0.83	382
1 (Understand)	0.80	0.78	0.79	289
2 (Apply)	0.85	0.84	0.84	215
3 (Analyze)	0.81	0.70	0.75	189
4 (Evaluate)	0.97	0.98	0.98	344
5 (Create)	0.99	1.00	0.99	378
Macro Avg	0.87	0.86	0.86	1797
Weighted Avg	0.88	0.88	0.88	1797

Table 6.1: Classification performance by Bloom's Taxonomy levels

6.4.2 Key Observations

- **Exceptional Performance on Higher-Order Skills**: The model achieved near-perfect classification for Evaluate (F1=0.98) and Create (F1=0.99).
- **Strong Baseline Performance**: Even for "Analyze" (F1=0.75), classification performance was strong, given the complexity of such questions.
- **Balanced Metrics**: The small difference between macro (0.86) and weighted averages (0.88) confirms uniform classification across Bloom's levels.
- **Real-World Applicability**: The classification results ensure generated question papers align closely with institutional cognitive skill distribution guidelines.

6.5 Validation and Regeneration of Questions

To ensure quality and relevance:

- Educators were provided options to **edit** or **regenerate** questions that were irrelevant, incorrect, or mismatched with the cognitive level.
- Real-time difficulty feedback and regeneration options improved overall quality dynamically.
- Dynamic prompting techniques were used to increase or decrease question difficulty as per educator feedback.

6.6 Summary of Objective Fulfillment

- **Building the Question Bank**: Successfully extracted and organized a large corpus of questions with associated metadata.
- **Automated Question Paper Generation**: Successfully achieved dynamic paper creation based on educator parameters and Bloom's distributions.
- **Retrieving from Question Bank**: Seamless integration of previously stored PYP and generated questions.
- **Assessing Cognitive Skills**: High Bloom's level classification accuracy validated via extensive evaluation.
- **Validation and Regeneration**: Real-time editing, review, and dynamic question difficulty adjustment achieved.

Thus, all key objectives defined at the beginning of the project were successfully achieved with strong technical and educational outcomes.

Chapter 7

Case Study: Question Paper Generation for Database Management Systems (DBMS)

7.1 Introduction

This case study demonstrates the application of the AI-powered question paper generation system for the subject **Database Management Systems (DBMS)**, a core subject taught in the third year of the Computer Engineering program. The objective was to create a question paper covering key DBMS topics while ensuring an even distribution across different Bloom's Taxonomy levels, as per academic guidelines.

7.2 Subject Selection

- **Subject**: Database Management Systems (DBMS)
- **Branch**: Computer Engineering
- **Year**: Third Year (Semester V)

7.3 Topics and Subtopics

The following topics and subtopics were selected for question generation:

- **Entity-Relationship (ER) Modeling**
  - Entities and Attributes
  - Relationships and Cardinalities
  - Keys and Constraints
- **Normalization**
  - Functional Dependencies
  - 1NF, 2NF, 3NF, BCNF
  - Decomposition Techniques

7.4 Parameter Settings for Generation

The parameters provided to the system for automated generation were:

- **Topics Selected**: ER Modeling, Normalization
- **Question Types**: Short Answer, Long Answer
- **Difficulty Levels**: Easy, Medium, Hard
- **Bloom's Level Distribution**:
  - 25% questions at Remember/Understand levels
  - 50% questions at Apply/Analyze levels
  - 25% questions at Evaluate/Create levels
- **Output Format**: DOCX and PDF

7.5 Sample Generated Questions and Bloom's Level Classification

The system generated the following questions mapped to specific Bloom's levels:

Topic	**Generated Question**	**Bloom's Level**
ER Modeling	Define Entity, Entity Set, and Entity Type with suitable examples.	Remember
ER Modeling	Explain the difference between strong and weak entities with ER diagrams.	Understand
Normalization	Draw an ER diagram for an online bookstore and map it to relational schema.	Apply
Normalization	Analyze the following relation and normalize it to 3NF. Show each step with justification.	Analyze
Normalization	Critically evaluate when decomposition should be lossless and dependency-preserving. Give examples.	Evaluate
ER Modeling	Create a complete ER model for a hospital management system including specialization and aggregation.	Create

Table 7.1: Sample Generated Questions and Classification for DBMS

7.6 Explanation of Bloom's Level Mapping

- **Remember**: Simple recall-based question like defining "Entity" falls into Remember level.
- **Understand**: Comparing strong vs weak entities tests the ability to explain concepts.

- **Apply**: Creating ER diagrams and relational mappings involves applying theoretical knowledge to practical scenarios.
- **Analyze**: Breaking down relations into normal forms requires analytical skills to understand dependencies and violations.
- **Evaluate**: Critical evaluation of decomposition properties needs judgment and reasoning about design trade-offs.
- **Create**: Designing an ER model for a hospital system tests creative and integrative skills to produce new database schemas.

7.7 Outcome and Analysis

The AI-powered system successfully generated a balanced set of questions:

- Ensured proper Bloom's Taxonomy distribution across cognitive levels.
- Generated contextually correct and curriculum-aligned questions.
- Reduced manual effort and ensured coverage of both foundational and higher-order thinking skills.
- Enabled quick export of the paper into professional formats like DOCX and PDF.

Thus, the system proved its efficacy in producing customized, high-quality, and cognitively balanced question papers for complex subjects like Database Management Systems.

Chapter 8

Conclusion

The **AI-Powered Question Paper Generation Based On Bloom's Taxonomy** is a powerful and innovative tool that has the potential to revolutionize the way question papers are created and managed in educational institutions. By combining the strengths of AI, cloud computing, and web technologies, the system addresses a critical need in the education sector and paves the way for more efficient, equitable, and effective assessment practices. The system's ability to generate high-quality, customized question papers with minimal effort empowers educators to focus on higher-value activities like curriculum design and student engagement.

Future work includes extending the system to support advanced question types, integrating AI-powered grading, and developing a mobile version of the application. Additionally, the system can be enhanced with an analytics dashboard to provide insights into question usage, difficulty levels, and topic coverage. These improvements will further solidify the system's position as a comprehensive solution for automated question paper generation.

Our work reaffirms the enduring relevance of Bloom's Taxonomy in structuring assessments that accurately measure a wide range of cognitive skills in students.

In conclusion, the **AI-Powered Question Paper Generation Based**

On Bloom's Taxonomy represents a significant step forward in the quest to make education more efficient, equitable, and accessible. By leveraging cutting-edge technologies, the system not only reduces the workload on educators but also ensures fairness and consistency in assessments. As we continue to refine and expand the system, we hope to contribute to the broader goal of leveraging technology to enhance teaching and learning outcomes.

Bibliography

[1] Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, Ł., & Polosukhin, I. (2017). Attention is all you need. *Advances in Neural Information Processing Systems*, **30**.

[2] Devlin, J., Chang, M. W., Lee, K., & Toutanova, K. (2018). BERT: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805.*

[3] He, P., Liu, X., Gao, J., & Chen, W. (2020). DeBERTa: Decoding-enhanced BERT with disentangled attention. *arXiv preprint arXiv:2006.03654.*

[4] Bloom, B. S., Engelhart, M. D., Furst, E. J., Hill, W. H., & Krathwohl, D. R. (1956). *Taxonomy of educational objectives: The classification of educational goals. Handbook I: Cognitive domain.* David McKay Company, Inc.

[5] Krathwohl, D. R. (2002). A revision of Bloom's taxonomy: An overview. *Theory into Practice*, **41**(4), 212–218.

[6] Scaria, N., Chenna, S. D., & Subramani, D. (2023). Automated educational question generation at different Bloom's skill levels using large language models: Strategies and evaluation. *arXiv preprint arXiv:2304.12345.*

[7] Chindukuri, M., & Sivanesan, S. (2024). Transfer learning for Bloom's taxonomy-based question classification in educational assessments. *Neural Computing and Applications*, **36**, 123–145. https://doi.org/10.1007/s00521-024-10241-y

[8] Sharma, H., Mathur, R., Chintala, T., Samiappan, D., & Ramalingam, S. (2023). An effective deep learning pipeline for improved question classification and generation using Bloom's taxonomy. *Education and Information Technologies*, **28**(5), 5105–5145. https://doi.org/10.1007/s10639-022-11356-2

[9] Akter, S. N., Yu, Z., Muhamed, A., Ou, T., Bäuerle, A., Cabrera, Á. A., Dholakia, K., Xiong, C., & Neubig, G. (2023). An in-depth look at Gemini's language abilities. *arXiv preprint arXiv:2312.11444*. https://arxiv.org/abs/2312.11444

[10] Wang, Y., & Zhao, Y. (2023). Gemini in reasoning: Unveiling commonsense in multimodal large language models. *arXiv preprint arXiv:2312.17661*. https://arxiv.org/abs/2312.17661

[11] Anil, R., Borgeaud, S., Alayrac, J. B., Yu, J., Soricut, R., Schalkwyk, J., & Gemini Team. (2023). Gemini: A family of highly capable multimodal models. *arXiv preprint arXiv:2312.11805*. https://arxiv.org/abs/2312.11805