RESEARCH ARTICLE

RSP Science Hub

International Research Journal on Advanced Science Hub
2582-4376
www.rspsciencehub.com
Vol. 06, Issue 07 July

Check for updates

http://dx.doi.org/10.47392/IRJASH.2024.027

Eduassesspro: An Automated Quiz Generator and Exam Creator

Ashwini Khairkar[1], Sanjivani Bulbule[2], Sneha Manchalkar[3], Shweta Phatate[4], Krishna Soni[5]

[1]Professor in Information Technology Department, Bharati Vidyapeeth's College of Engineering for Women (BVCOEW), Savitribai Phule Pune University (SPPU), Pune, India.

[2,3,4,5]UG - Information Technology, Bharati Vidyapeeth's College of Engineering for Women (BVCOEW), Savitribai Phule Pune University (SPPU), Pune, India.

Emails: *ashwini.khairkar@bharatividyapeeth.edu[1], sanjivani0202@gmail.com[2], snehamanchalkar153@gmail.com[3], shwetaphatate22@gmail.com[4], krishna13218@gmail.com[5]*

Article history

Received: 30 June 2024
Accepted: 23 July 2024
Published: 27 July 2024

Keywords:

Assessment, Exam Creation, Grading, MCQ generation, NLP, Quiz Generation.

Abstract

The standard exam creation process is slow and arduous for both teachers and students which lead to delay in result declaration. Multiple-choice questions (MCQs) are recognized as the best way of assessing student understanding. This project try to address these challenges faced in assessments. This project focuses on generating objective questions with correct answers based on input text by the user with the help of quiz generator bot, exam creation bot, interactive quiz bot and grading bot collectively streamlining the entire process which will help to create engaging and challenging assessments for students. Input to this platform will be a random text in the form of pdf, docx or txt given by user based on which quiz will be generated by means of NLP techniques for pre-processing, BERT Algorithm, WordNet, PKE and T5 Algorithm, the system not only facilitates efficient question generation but also stores all questions and student results for future reference. The system stores all the questions generated on a particular topic as well as the results and grades received by the students. This integrated platform offers fixed question tests with immediate feedback, simplifying the assessment process for both educators and students, providing a quick and easy way to assess student understanding on topic.

1. Introduction

Traditional examination procedures present significant challenges for both educators and students, characterized by their time-consuming nature and protracted result waiting periods. Acknowledging the efficacy of Multiple-Choice Questions (MCQs) in gauging student understanding, we propose an innovative solution: the "Automated Quiz Generator and Exam Creator" system, designed to modernize online assessments.

This system integrates specialized modules tailored for distinct tasks within the examination process. The Quiz Generation Module employs NLP techniques and PyPDF2, Python-docx, and pandas for pre-processing, BERT for text summarization, RAKE for keyword extraction, T5 for question-and-answer generation, and WordNet for distractor creation. All generated questions are systematically stored in a Firebase database. The Exam Creation Module empowers users to craft personalized exams, specifying parameters such as duration and question count. The Interactive Quiz Module facilitates educators in assigning exams to students, enriching the overall learning experience. Simultaneously, the Grading Module assumes a central role by automating the grading process, furnishing objective evaluations based on students' performance. This overall system not only streamlines the entire assessment process but also harnesses advanced technologies to refine question generation and grading precision. By amalgamating these functionalities, the "Automated Quiz Generator and Exam Creator" system emerges as a dynamic solution addressing the inefficiencies of traditional examination methods, promising a more efficient and engaging approach to online. [1]

2. Methodology

As shown in Figure 1, The System consists of two Dashboards, one for the admin and the other for students, both are accessible after Authentication. [3] The admin inputs the document in the System and the modules takes over the task further. The system consists of five modules as given below

2.1 Quiz Generation Module

Quiz Generation Module takes any type of file as an input from Admin and extracts the Text. Using NLTK, the module pre-processes the text from the document [12]. Further, BERT model is used to summarize the text [9]. PKE model extracts Keywords [2], T5 model is used to generate questions [4]. WordNet will create distractors for multiple choices [6].

2.2 Question Bank Database

The Multiple-Choice Questions (MCQs) which are generated are stored in the firebase database and these questions can be used at the time of creating exams.

2.3 Exam Creation Module

The admin can give number of questions to be chosen from the database and can create Exam. The admin can also set a timer while creating exam [7].

2.4 Interactive Quiz Module

This module shares the interface with students where students can attempt exams designed by the admin. Students need to complete the Exam under the time limit set by the admin.

2.5 Grading Module

After the Exam is completed, the grades are evaluated by Grading Module [7]. It shows the grades to the admin and the students.

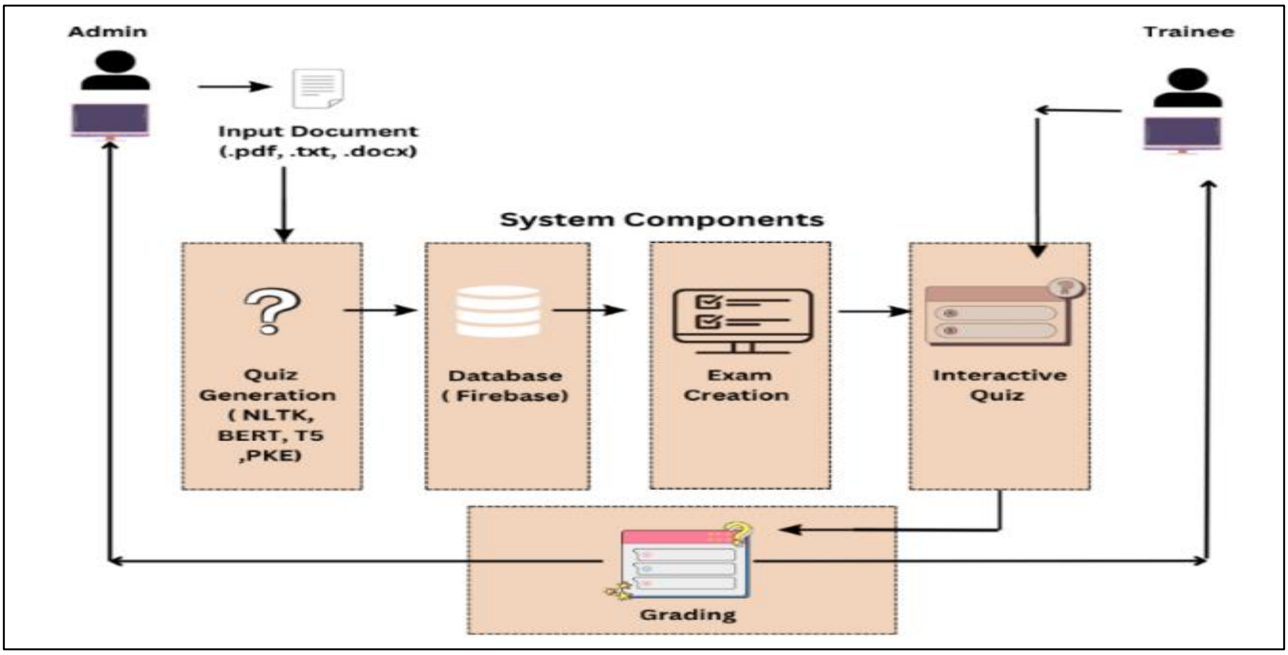


Figure 1 System Architecture

3. Algorithms / Libraries

3.1 PyPDF2, Python-Docx

PyPDF2 is a Python Library which is used extract text from .pdf File. Docx is a Python Library which is used to extract the text from .doc file. The user can give any type of document as an input. For extracting the text from that document, which can be in any format like PDF or DOC file, Libraries like PyPDF2 and Python-docx are used. So, that text from those documents is extracted correctly.

3.2 NLTK (Natural Language Toolkit)

NLTK (Natural Language Toolkit) is a Python library used for Text Preprocessing. Text extracted from the input document is given as an input to get Pre-processed. Firstly, the functions like Tokenizing Sentences and Words is performed then Stop words from the tokenized words are removed and then POS tagging is applied. Also Lemmatization and Stemming of the words is done using the available functions in NLTK library [13].

3.3 BERT Model (Bidirectional Encoder Representations from Transformers)

BERT Model is used for Extractive Summarization of the text given as an input. In our project, the extracted text from the document is given as an input to BERT model and then it summarizes that text once, the text is summarized using BERT model, and then the Summarized text is used further for Keyword Extraction Process [2]. Due to this Summarized text, the MCQs will be created based on only important text from file. [5]

3.4 PKE (Python Key Phrase Extractor)

PKE is an open-source Python Library which is used in our project for extracting Keywords from the Summarized text. [8] The text Summarized using BERT model is given as an input to PKE for extracting keywords from the summarized text. The extracted Keywords from the summarized text will help to create questions. The extracted keywords are the answers to the question created for that keyword. [10]

3.5 Wordnet

Wordnet is a large lexical database of words and their semantic relations. [11] In our System, we used Wordnet for Generating Distractors for the Correct answer of the question i.e. Keywords. As our System is going to generate MCQ from the text, we have created the options which can be the distractors of the correct answer in the MCQ Question. It takes keyword as an input and generates distractors for that specific keyword. Using wordnet, distractors for the correct answer is generated [6].

3.6 T5 (Text-To-Text-Transfer-Transformer)

In our system, T5 model is used to generate questions from the summarized text. T5 Model takes summarized text and extracted keywords as an input and generates questions based on that text. T5 Model is used to generate meaningful questions from the summarized text and keywords extracted [4]. The questions generated using T5 model will be based on the text file given as an input. [14]

3.7 Random

Random is a Python library which is used for the purpose of randomization. In our system, Random is used to randomize or change the sequence of questions and options of MCQs. So that we can prevent students from copying. [15-18]

4. Results and Analysis

4.1 Results

Figure (2-11) shows the results of the Proposed System.

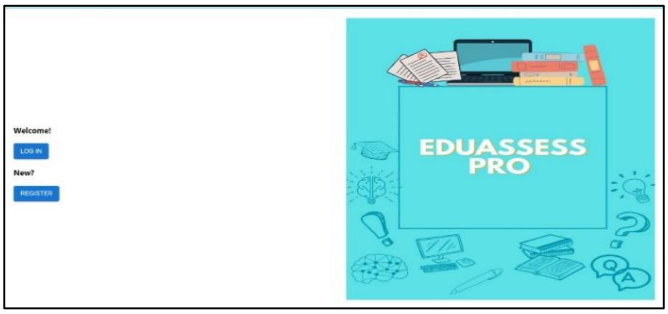


Figure 2 Home Page

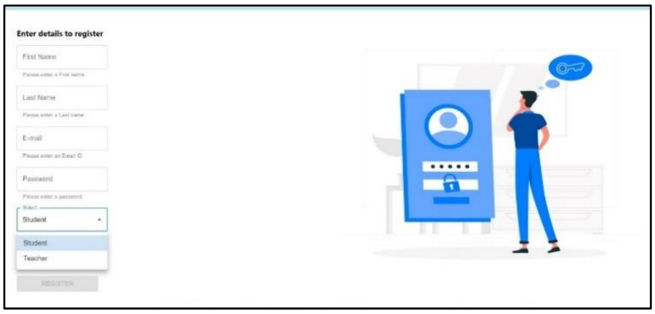


Figure 3 Registration Page

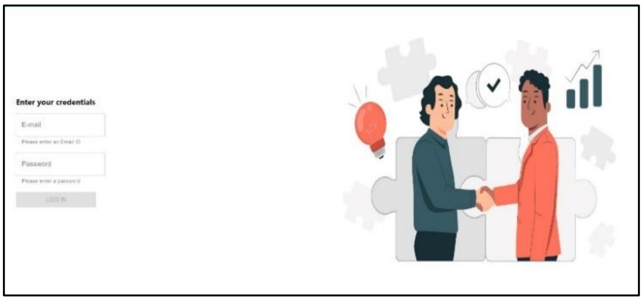


Figure 4 Login Page

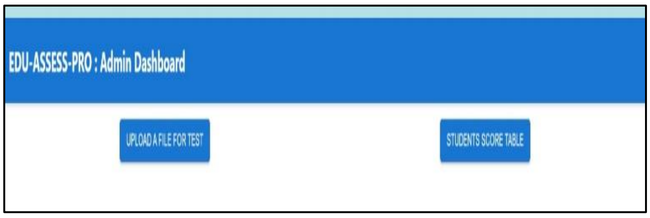
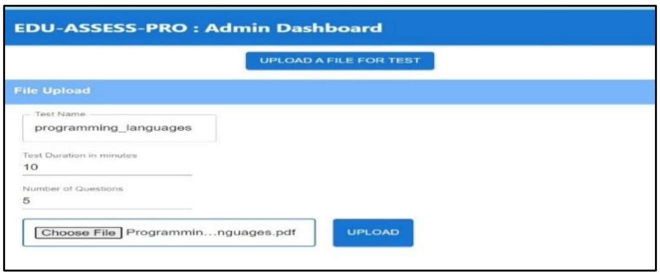


Figure 5 Admin Dashboard



EDU-ASSESS-PRO : Admin Dashboard

UPLOAD A FILE FOR TEST

File Upload

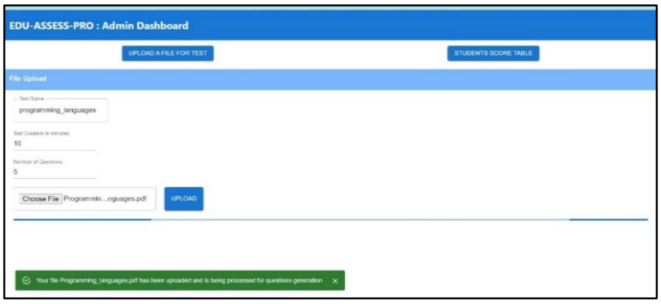
Test Name: programming_languages

Test Duration in minutes: 10

Number of Questions: 5

Choose File Programmin…nguages.pdf UPLOAD

Figure 6 Admin Creating Exam



EDU-ASSESS-PRO : Admin Dashboard

UPLOAD A FILE FOR TEST STUDENTS SCORE TABLE

File Upload

Test Name: programming_languages

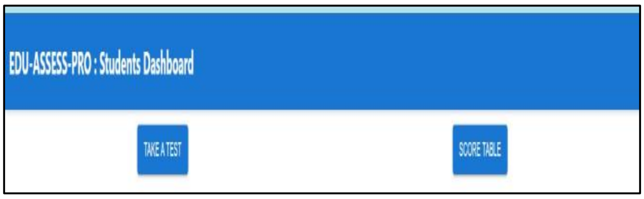
Test Duration in minutes: 10

Number of Questions: 5

Choose File Programmin…nguages.pdf UPLOAD

Your file Programming_languages.pdf has been uploaded and is being processed for questions generation ×

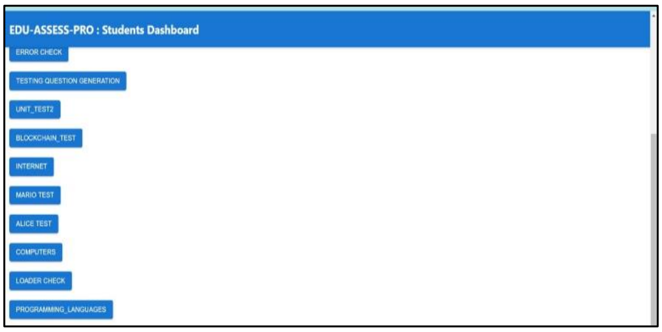
Figure 7 Document Uploaded Successfully and Given for MCQ Generation and Exam Creation



EDU-ASSESS-PRO : Students Dashboard

TAKE A TEST SCORE TABLE

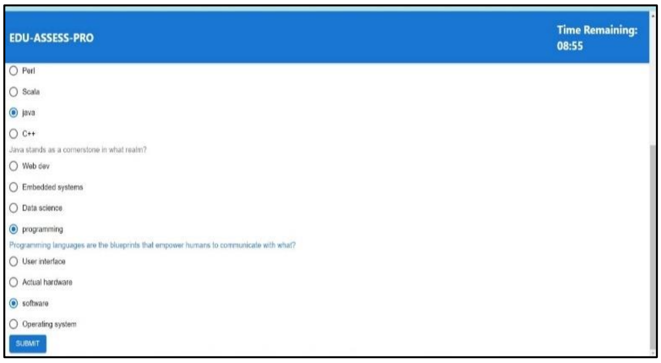
Figure 8 Student Dashboard



EDU-ASSESS-PRO : Students Dashboard

- ERROR CHECK
- TESTING QUESTION GENERATION
- UNIT_TEST2
- BLOCKCHAIN_TEST
- INTERNET
- MARIO TEST
- ALICE TEST
- COMPUTERS
- LOADER CHECK
- PROGRAMMING_LANGUAGES

Figure 9 Displaying Available Exams to Student



EDU-ASSESS-PRO Time Remaining: 08:55

○ Perl
○ Scala
● java
○ C++

Java stands as a cornerstone in what realm?

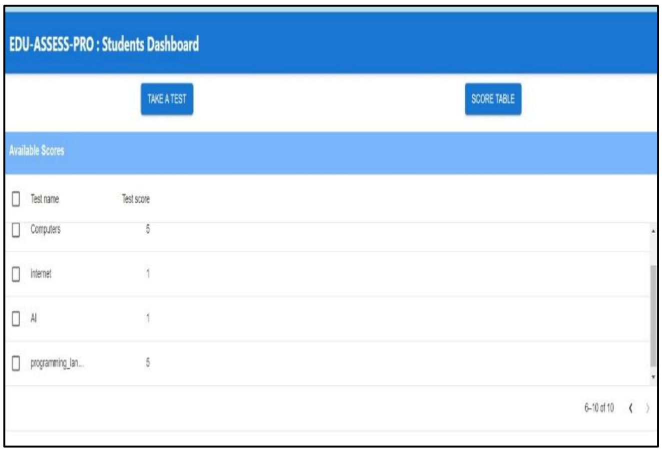
○ Web dev
○ Embedded systems
○ Data science
● programming

Programming languages are the blueprints that empower humans to communicate with what?

○ User interface
○ Actual hardware
● software
○ Operating system

SUBMIT

Figure 10 Student Attempting Assigned Exam



EDU-ASSESS-PRO : Students Dashboard

TAKE A TEST SCORE TABLE

Available Scores

□	Test name	Test score
□	Computers	5
□	internet	1
□	AI	1
□	programming_lan...	5

6–10 of 10 ‹ ›

Figure 11 Displaying Results to Students

4.2 Analysis

Table 1 Comparison of Features

Systems Features	Quest gen	Quizly	OpExam	Our System (EduAssess Pro)
MCQ Generation	Yes	Yes	Yes	Yes
Exam Creation	No	No	Yes	Yes
Grading	No	No	No	Yes
Accepts document with unlimited word count	No	No	No	Yes
Takes Number of questions to generate as an input	Yes	Yes	No	Yes
Any type of text file is accepted	Yes	No	No	Yes
Customize Questions	Yes	No	Yes	Yes
Time limit to attempt exam	No	No	No	Yes
Randomization of Questions and Options	No	No	No	Yes

As shown in Table 1, Questgen tool provides only 4 features, Quizly provides only 2 features and OpExam provides 3 features but our system (EduAssessPro) provides 8 most important features which are more than twice as compared to all other systems. The Features provided by our system are shown in Figure 12,

1. MCQ generation,
2. Exam Creation and Grading
3. Accepting document with unlimited word count
4. Taking Number of questions to generate parameter as an input
5. Accepting any type of text file
6. Customized Questions
7. Taking time limit parameter as an input and setting time limit for exam
8. Randomization of Questions and Options

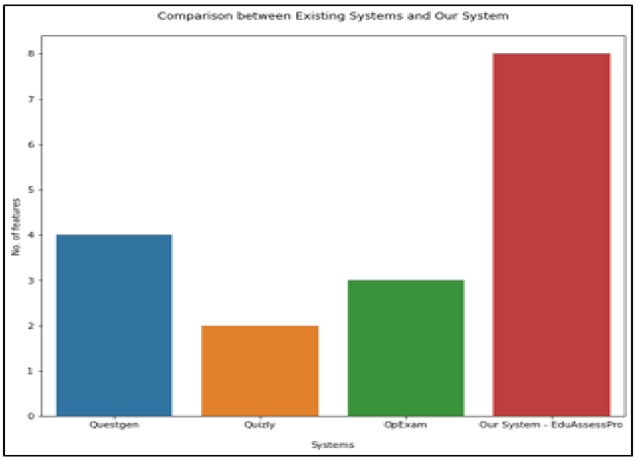


Figure 12 Comparison of Features of Various Existing Systems with Our System

Conclusion

We have implemented Automated Quiz Generator and Exam Creator, which takes any text document as an input and generates MCQs from the text extracted from document and creates an exam for the students. [19] In this Project, several types of algorithms and Libraries to generate MCQs are used. We have studied the BERT Algorithm for Summarization, NLTK library for Pre-processing of text, PKE library for Keyword Extraction, Wordnet library for generating Distractors and T5 Model for generating questions. [20] This project helps to create a quick responsive quiz generation as well as exam creation module. In this project, we are going to implement a quiz generation module in which we fulfil the user requirement that is to create a MCQ quiz from the document given as input by user. The System will support Quiz generation as well as Exam creation along with a grading module which will grade student. After grading is done, each student receives result of the Exam attempted.

References

[1]. Harshada Patil, Aarti Gaikwad, Mitali Dalave, Sneha Suryawanshi, Prof. Pallavi Patil “An automatic MCQ and summary generation by using NLP”, May 2023.

[2]. Mr. Krishna Reddy, B. Harshavardhan Reddy, M Dheeraj, G Vishal. “MCQ generation using Natural language processing” (April 2023).

[3]. Kommey, Benjamin & Keelson, Eliel & Frimpong, & Twum-Asare, Seth & Konadu, & Akuffo, Konadu & Address, Present. Automatic Multiple Choice Examination Questions Marking and Grade Generator Software. IPTEK The Journal for Technology and Science. 33. 853-4098. 10.12962/j20882033.v33i3.14522. (2023).

[4]. Puneeth Thotad, Shanta Kallur, Sukanya Amminabhavi. “Automatic Question Generator Using Natural Language Processing” (2022).

[5]. Maheen, Farah, Muhammad Asif, Haseeb Ahmad, Shahbaz Ahmad, Fahad Alturise, Othman Asiry, and Yazeed Yasin Ghadi. "Automatic computer science domain multiple-choice questions generation based on informative sentences." (2022).

[6]. Da Silva, Elton; Aires da Silva, Fernando; Womg, Kim Jang; and Ho, Tai Teei, "QUERAI – A Smart Quiz Generator" (2022).

[7]. Pritam Kumar Mehta, Prachi Jain, Chetan Makwana, Dr. C M Raut “Automated MCQ Generator using Natural Language Processing”. e-ISSN: 2395-0056 (2021).

[8]. Elshafey, Mohamed Abdelmoneim Dr. and Ghoniemy, Tarek Said Dr. "The Development of QMMS: A Case Study for Reliable Online Quiz Maker and Management System" (2021).

[9]. Ramon Dijkstra, Zulkuf Gen, Subhradeep Kayal, Jaap Kamps. “Reading Comprehension Quiz Generation Using Generative Pre-Trained Transformers”

(2021).

[10]. Das, B., Majumder, M., Phadikar, S. et al. Automatic question generation and answer assessment: a survey. RPTEL 16, 5 https://doi.org/10.1186/s41039-021-00151-1. (2021).

[11]. Xinying Qiu, Haiwei Xue, Lingfeng Liang, Zexin Xie, Shuxuan Liao, Guofeng Shi. Automatic Generation of Multiple-choice Cloze-test Questions for Lao Language Learning. (2021).

[12]. Patil, N., K. Kumari, D. Ingale, P. Patil, and A. R. Uttarkar. "A Survey on Automatic Multiple Choice Questions Generation from Text" (2021).

[13]. S. Maniar, J. Parmani, M. Bodke and K. Saxena, "Generation and grading of arduous MCQs using NLP and OMR detection using OpenCV," 12th International Conference on Computing Communication and Networking Technologies (ICCCNT), Kharagpur, India, 2021, pp. 1-7, doi: 10.1109/ICCCNT51525.2021.9580089. (2021).

[14]. Ch, Dhawaleswar Rao, and Sujan Kumar Saha. "Automatic multiple choice question generation from text: A survey." (2020).

[15]. Naveed Afzal. “Automatic Generation of Multiple-Choice Questions using Surface-based Semantic Relations.” (2019).

[16]. Bongir, Amit & Attar, Vahida & Janardhanan, Ramanand. (2018). Automated Quiz Generator. 174-188. 10.1007/978-3-319-68385-0_15. (2018).

[17]. A. Killawala, I. Khokhlov and L. Reznik, "Computational Intelligence Framework for Automatic Quiz Question Generation," (2018).

[18]. Kumar, S. & Prasad, V. & Santhanavijayan, A. & S.R., Balasundaram & Narayanan, S. “Automatic generation of multiple-choice questions for e-assessment.” (2017).

[19]. Lin, C., Liu, D., Pang, W. et al.” Sherlock: A Semi-Automatic Framework for Quiz Generation Using a Hybrid Semantic Similarity Measure.” (2015).

[20]. Pandey, Shivank, and K. C. Rajeswari. "Automatic question generation using software agents for technical institutions." (2013).