An RPA Based Question Paper Generator using UiPath

K Soumya, D Ashwitha, B Sriram, D Naveen, J Poorana Chandar

Assistant Professor, UG Scholars, Department of Information Technology, Vignana Bharathi Institute of Technology, Aushapur, Hyderabad

Abstract: This article proposes an overview of an innovative Question Paper Generator developed using Robotic Process Automation (RPA) technology with UI Path as the platform. In this application the main aim is to generate question paper template which satisfy the condition of Bloom's Taxonomy. This application offers an innovative and efficient tool to enhance the learning experience of students. Traditional methods of manually generating questions can be time- consuming, error prone and lack consistency. The automated question paper generator is an innovative solution designed to streamline and revolutionalize the process of creating question papers for educational assessment. This application involves the selection of random questions satisfying the condition of Bloom's Taxonomy.

Keywords: Automatic Question Paper Generator, Question Bank, Robotic Process Automation, Blooms Taxanomy, Xampp control panel.

1 Introduction

The conventional method of manually creating question sheets has long been a labor-intensive and challenging task for instructors. In response to this, an innovative solution is proposed in the form of an Intelligent Automatic Question Paper Generator system. This system seeks to alleviate the burden on teachers by providing a user-friendly interface that facilitates data storage, the selection of random questions, and expeditious generation of question papers.

The existing system lacks a robust mechanism for the efficient generation of question papers, resulting in a tedious and time-consuming process that lacks consistency. While various applications have been developed to create question paper templates through programming knowledge, individuals with limited programming skills find it challenging to either develop such applications or comprehend existing ones. Additionally, managing large question banks within the current system is a cumbersome task.

To confront the aforementioned challenges associated with manual question paper generation, the envisioned system strategically harnesses the potent capabilities of Robotic Process Automation (RPA) with the integration of UiPath. UiPath stands out as a preeminent RPA tool, celebrated for its intuitive drag-and-drop functionality, which facilitates a seamless and accessible user experience. RPA, as a cutting-edge technology, is purposefully crafted to streamline and automate repetitive, rule-based tasks entrenched in corporate processes. At the core of this innovation lies the deployment of software robots or bots, intelligent entities designed to emulate human interactions with digital systems.

This transformative technology unfolds its prowess across a spectrum of functionalities, encompassing intricate tasks such as data entry, extraction, transaction processing, and seamless communication across diverse systems. The software robots exhibit a remarkable ability to replicate human actions, executing these tasks with precision and efficiency. By mimicking human behavior in digital interfaces, RPA not only expedites processes but also ensures a high degree of accuracy and consistency in the execution of routine functions.

Within the context of the proposed system, the integration of UiPath and RPA technology emerges as a strategic response to the challenges ingrained in the conventional question paper generation process. The drag-and-drop feature of UiPath enhances the accessibility of the system, making it user-friendly and adaptable for individuals with varying levels of technical proficiency. The intelligent automation

capabilities of RPA, on the other hand, empower the system to autonomously carry out tasks that were traditionally labor-intensive and time-consuming.

The envisioned system aims to streamline the question paper generation process by utilizing an automated question paper generator built with UiPath. This approach promises to save significant time and effort for teachers and test coordinators. The project scope encompasses the development of an intuitive user interface, integration of question banks into databases, and implementation of algorithms to generate balanced and randomly selected question papers. The system may also include features allowing users to select difficulty levels and classify questions.

In alignment with Bloom's Taxonomy, the application employs various problem-solving techniques and assesses the difficulty levels of questions. By doing so, it ensures a comprehensive approach to testing that spans different cognitive skills. Overall, the proposed system represents a significant advancement in the realm of question paper generation, offering a user-friendly and efficient solution to the challenges faced by educators. The incorporation of RPA technology not only simplifies the process but also enhances the overall quality and consistency of generated question papers.
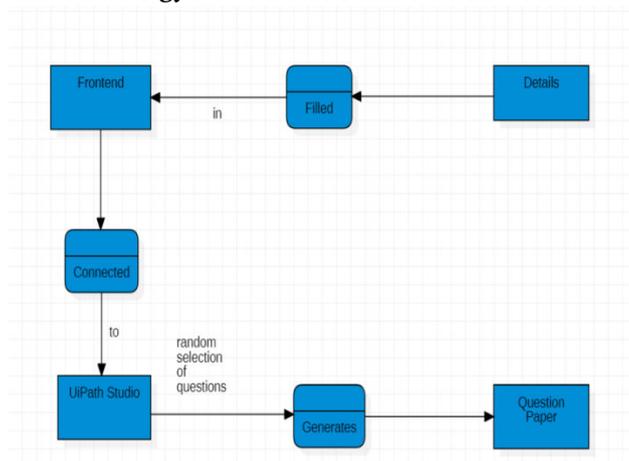
2 Methodology

Fig: Data Flow Diagram

Creating an Automatic Question Paper Generator is a nuanced process that traverses several essential phases, spanning from the initial requirements analysis to the iterative enhancements post-deployment. Let's delve into a comprehensive

methodology for the development of this innovative tool:

1. Vision and Scope Definition:

Clearly articulate goals: Define the overarching objectives of the Question Paper Generator, emphasizing time efficiency, consistency, and adaptability across diverse subjects and difficulty levels.
Engage stakeholders: Establish a collaborative dialogue with educators, teachers, and administrators to capture their unique perspectives, requirements, and expectations.

2. In-Depth Exploration:

Survey existing landscape: Undertake a thorough exploration of comparable applications and systems, discerning their strengths and weaknesses.
Scrutinize question paper formats: Conduct a detailed analysis of various question formats and educational standards to ensure the generator caters comprehensively to diverse educational needs.

3. Architectural Blueprint:

Component identification: Enumerate the pivotal components, encompassing the user interface, question bank, randomization algorithms, and database.
Technology stack selection: Carefully curate a technology stack that aligns with the project's goals, taking into account factors like scalability, maintainability, and user-friendliness.

4. Intuitive User Interface:

User-centric design: Craft an intuitive and user-friendly interface, empowering teachers to effortlessly navigate the system and input their preferences.
Customization features: Integrate customizable options, allowing users to tailor the system to their specific requirements, including subject selection and difficulty levels.

5. Robust Question Bank Management:

Database schema development: Engineer a robust database structure capable of efficiently storing a diverse array of questions categorized by subject,

topic, and difficulty.

CRUD operations implementation: Enable seamless addition, modification, and deletion of questions, ensuring the question bank remains dynamic and up-to-date.

6. Smart Randomization Algorithms:

Logic for randomness: Implement sophisticated algorithms for question selection, ensuring a judicious balance across topics and difficulty levels.

Consider constraints: Factor in constraints such as section-wise question limits or specific distributions of question types for a tailored and balanced approach.

7. Synergizing with RPA (UiPath):

Workflow automation: Develop UiPath workflows that automate intricate tasks like data entry, extraction, and seamless communication with the question bank.

Seamless integration: Ensure harmonious integration of the RPA component with the main application, fostering a cohesive and user-centric experience.

8. Rigorous Testing:

Component validation: Conduct thorough unit testing to validate the functionality of individual components, including question bank management, randomization algorithms, and RPA workflows.

Integration testing: Assure the seamless collaboration of all system components when integrated into the holistic application.

User acceptance testing: Engage teachers and administrators to assess the system's usability, ensuring it aligns effectively with educational objectives.

9. Deployment and Documentation:

Application packaging: Prepare a deployment-ready package for widespread use in educational institutions.

Comprehensive documentation: Create user manuals and guides to facilitate teachers in leveraging the system effectively.

10. Training, Support, and Evolution:

Training sessions: Conduct training programs to empower teachers and administrators with the necessary skills for optimal utilization of the Automatic Question Paper Generator.

Ongoing support: Establish robust support mechanisms and channels for issue resolution and feedback reception.

Continuous improvement: Encourage user feedback, leveraging it for iterative enhancements that introduce new features and optimize system performance over time.

By meticulously adhering to this multifaceted methodology, developers can forge a sophisticated and user-centric Automatic Question Paper Generator that not only meets but exceeds the challenges encountered by educators in the intricate landscape of exam preparation.

2.1 Frontend Setup

The development of Automatic Question Paper Generator first involves the designing of Front-End. The front end part involves the registering and logging of user to begin the process of generating question paper. The front end is developed using HTML, CSS and Javascript. The front end part of this application deals with the entry of user details and their verification. In front end there is aoption to enter the required subject name and with the number of questions to be printed on the question paper. The user is first authenticated in order to provide security.

2.2 Backend Setup and Storage

The development of the application involves setup of backend. The details entered in the front end are stored on to the back end. The backend is developed using xampp control panel.Developed by Apache Friends, XAMPP is a well-liked cross-platform web server solution stack package that is free and open-source. It contains parts like the MySQL database, PHP, Perl, and Apache HTTP Server and is used for web development and testing. The front end of the application is connected to the backend using PHP Language. MySql Database is used to store the information entered in the front end.

2.3 UiPath Studio connection

The main motive of the application is to generate a random number of questions from a question bank. The number of questions mentioned in the front end is taken as input .The input selects random number of questions from the question bank and generates the question paper.One of the top Robotic Process

Automation (RPA) tools is UiPath Studio, which was created by UiPath, a business that specializes in automation software. By automating repetitive and rule-based processes with software robots, or "bots," RPA enables businesses to increase production and efficiency. One essential part of the UiPath RPA platform is UiPath Studio, which gives customers the ability to develop, test, and implement automation workflows.

2.4 Selection of Random Questions

In UiPath Studio, you can select random questions from a collection using a combination of activities to achieve this.

Set up the variables:

Establish variables to hold the total number of questions as well as the desired number of randomly chosen questions.

Assign Task:

To assign the total number of questions to a variable, use the Assign activity.

Dialog Activity for Input:

Prompt the user to enter the number of random questions they wish to choose by using the Input Dialog activity. Put the outcome in a variable (like number Of Random Questions) and save it.

Decision-Making Flow:

To determine whether the number of random questions entered is less than or equal to the total number of questions, use a Flow Decision activity. If so, move forward; if not, address the error or request that the user submit a genuine number.

Assign a task for arbitrary inquiries:

Make a list of random indexes using another Assign activity inside the "True" branch of the Flow Decision.

For Each Task:

To extract the matching random questions from your initial list, go over the random Indexes list using a For Each activity.

2.5 Generation of Question Paper

The Question Paper is generated based on given input such given subject name, number of questions and Blooms taxanomy. The Question Paper is generated in a format which is already mentioned in Uipath in an activity. The question paper is generated in the given format. The Question Bank used for generation of question paper is also taken as input in the Uipath Studio. The Question Paper generated is saved at a location which is specified in the body of the activity.

The generated Question Paper is available at the desired location in two formats. The Question paper is in word document format and PDF format.

3 Discussion and Results

The primary outcome of developing this application is to generate an automated question paper.In this application the UiPath studio plays a major role as it is responsible for the selection of random questions and generation of question paper.UiPath studio uses its drag and drop functionality to connect to the database and implement the randomiser function. The question paper generated contains unique questions which follows the Bloom's Taxonomy.

User Experience and Interface:

The Automatic Question Paper Generator's user interface was created with an emphasis on ease of use and intuitiveness.
Users appreciated that UiPath Studio's drag-and-drop feature made it easier for people with little technical experience to navigate and make efficient use of the system.
The user-centric design appears to have served its intended aim, based on the good feedback received about the user interface.
The system's accessibility supports the objective of lowering instructors' and test coordinators' learning curves.
f randomization techniques, guaranteeing an equitable and varied collection of inquiries.

Effectiveness in Formulating Question Papers:

When compared to human approaches, the automated question paper generation utilizing UiPath showed considerable time savings.
The question papers produced by the randomization algorithms were evenly distributed among various subjects and skill levels.
The observed time efficiency demonstrates how well robotic process automation (RPA) works to automate repetitive processes.
A fair and varied set of questions is ensured by the balanced distribution of questions, which shows that randomization techniques were successfully implemented.

Results of the integration with UiPath Orchestrator:

The central administration of automation processes was made easier by the interface with UiPath Orchestrator.

Simplified deployments and updates helped to create a more ordered workflow.

The Orchestrator connection improved the system's scalability and maintainability by enabling smooth automation control and monitoring.

The Automatic Question Paper Generator is consistent across several instances thanks to the centralized deployment approach.

Comments and Iterative Enhancements:

Users' constant input revealed areas that needed work, which prompted further revisions and upgrades.

Iterative improvements were implemented in response to concerns and recommendations from users.

An important factor in the system's improvement and optimization was user feedback.

The project's commitment to flexibility and responsiveness to user needs is shown in the iterative methodology.

Instruction and User Acceptance:

To acquaint educators and administrators with the Automatic Question Paper Generator, training sessions were held.

Constant support systems were set up to handle questions and issues from users.

By equipping users with the necessary skills to use the system efficiently, the training sessions helped to facilitate a seamless onboarding experience.

Continuous support makes sure that the system is adaptable and keeps up with the changing needs of its users.

4 Conclusion:

The question selection difficulty has been characterized as a multi-constraint optimization problem, where the goal is to produce question papers that satisfy numerous constraints given by the paper setter. The Intelligent Question Paper Generator saves a great deal of time over traditional systems by producing a well-formatted question paper in a couple of seconds.Since the question paper is prepared right before the examination, there is no danger of it being leaked when this technique is used to generate the questions. Teachers will save a great deal of time as a result, increasing productivity. The mechanism in place aims to find constructive solutions to the issues mentioned above.

References

1. M. Mohandas, A. Chavan, R. Manjarekar, and D. Karekar, "Automated Question Paper Generator System," International Journal of Advanced Research in Computer and Communication Engineering, Vol. 4, number 4, December 12, 2015, ISSN 2278- 1021.

2. B. Buvisuriya, Shuffling Algorithm-Based Automatic Question Paper Generator (June 24, 2022). The International Journal of Innovative Research in Computer and Communication Engineering, Volume 10, Issue 6, June 2022, eISSN: 2320-9801, p-ISSN: 2320-9798.

3. In the International Journal of Computer Science Trends and Technology, volume 6, number 1, January - March 2015, Ashok Immanuel and Tulasi B. published "Framework for Automatic Examination Paper Generation System".

4. Adaptive Online Exam Questions Based on Systematic Analysis and Design, Yang Yu and Hongyan Wang, Wuhan University of Technology, vol. 4, 2008, p. 30. Sat. 29/10/2011; whatis.techtarget.com.

5. "Automatic Question paper Generation System using Randomization Algorithm," by Kapil Naik, Shreyas Sule, Shruti Jadhav, and Surya Pandey Issue 12, pages 1-3, IJETR, Vol. 2, Dec.

6. Acropolis Course Management System's 2011 Software Design Specification (SDS)Ibrahim Eldesoky Fattoh published Automatic Multiple Choice Question Generation System for Semantic Attributes Using String Similarity Measures in the Computer Engineering and Intelligent Systems journal (www.iiste.org). 2014; Vol.5, No.8, ISSN 2222-1719 (Paper) ISSN 2222-2863

(Online).

7. The IEEE Computer Society's Software Engineering Standards Committee published IEEE Recommended Practice for Software Requirements Specifications in 1998.

8. Yang Yu, Hongyan Wang, Systematic Analysis and Design- Based Adaptive Online Exam Questions, Wuhan University of Technology, vol. 4, 2008, p. 3. Scholar on Google .

9. China's Information Technology Education, vol. 08, Binghua Chen B/S Model of Building a Test Server, 2008, p. 87 Examine the Record in Scopus

10. Gui Wang JSP-Based Platform for Remote Examination of The Design and Implementation, vol. 10, Mobile office (2008), p. 41CrossRefView Record in Scopus .

11. Developers Surprised by Bing Liu Beijing's Java WebPublishing House of Electronics Industry (2008) Google Academic.

12. R. Johnson, J. HoellerExpert One- on-one J2EE Development Without EJB[M] Wiley Publishing Inc., Indianapolis (2004) Google Scholar.

13. Himanshu Jethwani, Mohd Shahid Husain, Mohd Akbar, ISSN 2347 - 7911 "International Journal for Innovations in Engineering, Science and Management" Available online at: www.ijiesm.com Volume 3, Issue 4, April 2015.

14. Isha Sood, 7 Characteristics Of A Good Question. Online available at: elearningindustry.com.

15. Fenil Kiran Gangar, Hital Gopal Gori, Ashwini Dalvi, "International Journal of Computer Applications" (0975 – 8887) Volume 166 – No.10, May 2017 42 Automatic Question Paper Generator System.

16. "Design of Adaptive Question Bank Development and Management System," 2nd IEEE International Conference on Parallel, Distributed and Grid Computing, 2012, Vijay Krishan Purohit, Abhijeet Kumar, Asma Jabeen, Saurabh Srivastava, R H Goudar, Shivanagowda.

17. Fuzzy Logic Based Intelligent Question Paper Generator, Suraj Kamya, Madhuri Sachdeva, Navdeep Dhaliwal, and Sonit Singh, IEEE International Advance Computing Conference (IACC), 2014.

18. "Representing Examination Question Knowledge into Genetic Algorithm," by Nordin Abu Bakar, Moamed Rezduan Abd Rashid, and Noor Hasimah Ibrahim Teo, IEEE Global Engineering Education Conference (EDUCON), 2014. [19] "Automated Question Paper Generator System," International Journal of Advanced Research in Computer and Communication Engineering, Vol. 4, Issue 12, December 2015, Mohandas, Aishwarya Chavan, Rasika Manjarekar, and Divya Karekar. [20] In the 2016 IEEE 8th International Conference on Technology for Education, Gauri Nalawade and Rekha Ramesh presented their paper, "Automatic Generation of Question Paper from User Entered Specifications using a Semantically Tagged Question Repository"