Automatic Question Paper Generator System

Dubey Harish[1], Tamore Hardik[2], Padhi Sagar[3], Prof. Manisha Bharambe[4]

[1,2,3]Students, Dept. of Information Technology Engineering, SSJCOE, Maharashtra, India
[4]Professor, Dept. of Information Technology Engineering, SSJCOE, Maharashtra, India

---***---

Abstract—*The task of question paper generation is very tiresome and time-consuming. In this paper, a fuzzy logic-based model is developed for autonomous paper generation, using Python. This project enables college authorities to generate a fully customized question paper with the help of a large number of questions stored in the database. The software can be used effectively to generate question papers based on the level of the examination which includes unit tests. It allows the authorities to select chapters from the syllabus. The software uses the database to utilize the question paper where the database could consist of thousands of questions. The software produces a random question paper set such that the question does not repeat in the same paper. The software is a powerful tool to generate question papers in a short period thus saving efforts.*

Keywords: Question paper, Python, MySQL.

1. INTRODUCTION

Examination plays a crucial role in the education system. Various examinations are conducted in the institutes worldwide to measure the knowledge and understanding of the students. Though the predominant methodology is used to create question papers in most of the institutes, it is inefficient. The conventional method is very time consuming and needs more effort. It is very challenging for teachers to cover all the aspects of the syllabus. Hence, there is a need for an automated question paper generating system which is fully customized in all the aspects. We have developed a system that can generate question papers that can cover all the syllabus in the curriculum efficiently and with minimum efforts. We have designed a role-based model that restricts access to the users. The system allows the administrator to input a set of questions into the database along with their weightage and complexity. At the time of question paper generation, the user has to specify the difficulty level and the modules/chapters from the syllabus on which they intend to take a test. The system generates question paper in a doc file with a pre-generated institute name on the top of the paper. This system aims to reduce the time consumption and efforts required for generating a question paper.

2. PROBLEM DEFINATION

The traditional method of question paper generation is time-consuming and hectic. It is very challenging for the teachers to cover all aspects of the course objectives and avoid duplication of questions in the subsequent exams. There are no standardized methods and hence the quality of the question paper depends completely on an individual teacher's experience and expertise. It requires going through vast syllabus and picking questions that require a lot of time and employee work. Sometimes there is a need to generate question papers at the eleventh hour. The fact that there is a shortage of experienced teachers makes the situation even worse. At times, all these factors may deteriorate the quality of the question paper. Making a good question paper containing varied questions aligned with the learning objective of the course in terms of contents and cognitive level is very difficult. Hence there is necessity to develop an automatic question paper generating system specifically for colleges which will allow colleges to generate question papers with random questions that cover all the selected chapters.

3. LITERATURE SURVEY

In recent years, many people have done research and proposed various system to generate question papers. The advancement in technology lead to improvement in previously developed systems. We have read some research papers based on this topic that can help us developing improvised system which will be efficient and reliable in all aspects. These papers are listed below in accordance to their year of publication.

[1]"Design of Adaptive Question Bank Development and Management System"

In this paper, they have designed an adaptive question bank management system that is intelligently picking questions from rich database (question bank) and representing the question model according to the inputs or parameters provided by the question paper designer (QPD). The system is using graph for constructing concept map and a database for storing questions.

[2]"Fuzzy Logic Based Intelligent Question Paper Generator"

In this paper, a fuzzy logic based model is proposed for autonomous paper generation, using MATLAB. Comparative analysis with classical method is done and fuzzy model is found to be more reliable, fast and logical. Fuzzy logic based approach is implemented for logical selection of these parameters while framing question paper for every subject irrespective of its discipline.

[3]"Representing Examination Question Knowledge into Genetic Algorithm"

The glaring issue is how to auto-generate question papers that would conform to Outcome Based Education (OBE) specification. This project will investigate multi-constraints genetic algorithm approaches in designing Auto-Generator of Examination Questions (AGEQ).

[4]"Automated Question Paper Generator System"

In this paper a new fuzzy logic based IQPGS system for autonomous paper generation has been proposed. Comparison with classical method shows that the proposed system is more reliable in terms of duplicity removal, uncompromised issues, and lesser man power, logical in terms of unbiased selection and faster as the use of fuzzy logics in machines both approximate and precise reasoning are considered very well.

[5] "Automatic Generation of Question Paper from User Entered Specifications using a Semantically Tagged Question Repository"

The existing tools are rigid and support very basic or limited tags. The system will automatically generate a question paper from semantically tagged question repository.

By implementing this system it was found that the accuracy with respect to questions selection is 95%. The usability and user friendliness was checked by 5 Computer Science instructors with an experience of at least 5 years. In the first phase of user testing, they were given simple set of instructions, viz. "explore each components of the system", "Validate each entry on every screen with positive and negative inputs "and "Check the correctness of generated question paper in XML format and word format".

[6]"Question Paper Generator and Answer Verifier"

They have incepted a software through which students can appear for assignments, get them corrected via the software and he marks will be automatically be updated in the college database.

The parameters provided to the person generating the assignment. Based on the parameters selected by, the system will form a dataset containing all the questions that satisfy all the constraints and form those data, 5 question will be randomly selected, and added to the assignment. The result given by this system is very significant. It provide high accuracy rate for checking papers

[7]"Automatic Question Paper Generator System"

In this system they have used Fuzzy logic and randomization in Asp.net for generating the question paper. The algorithm is illustrated in the paper by using four constraints namely question paper format, coverage of syllabus, coverage of difficulty levels and coverage of cognitive level. The algorithm presented is extensible to support any number of user defined constraints.

[8]"Automatic Question Paper Generator"

The system has capability to process different unique sets of papers very automatically. This system utilizes the SQL queries to solve the problem of randomizing the questions every time. Automatic Question Paper Generator system is .Net based application to provide robustness and feasibility in making question papers

[9]"Android Based Exam Paper Generator"

Exam Paper Generator provides a solution to choose challenging, well framed questions and make it easy for the instructor to generate it within a short period of time. This can be done in a few taps of the hand as it is an Android application, therefore accessible at any time and place. In this android application they have implemented a system in which random questions will be picked by mapping it with the conditions provided. Prior to this, administrators can enter the formatted questions in the database (also called as a specification table

4. SYSTEM ARCHITECTURE

A. Flowchart

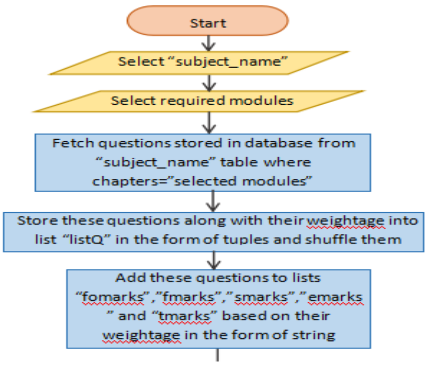

Start
Select "subject_name"
Select required modules
Fetch questions stored in database from "subject_name" table where chapters="selected modules"
Store these questions along with their weightage into list "listQ" in the form of tuples and shuffle them
Add these questions to lists "fomarks","fmarks","smarks","emarks" and "tmarks" based on their weightage in the form of string

Fig 1. Flowchart of QPGS (Part 1)

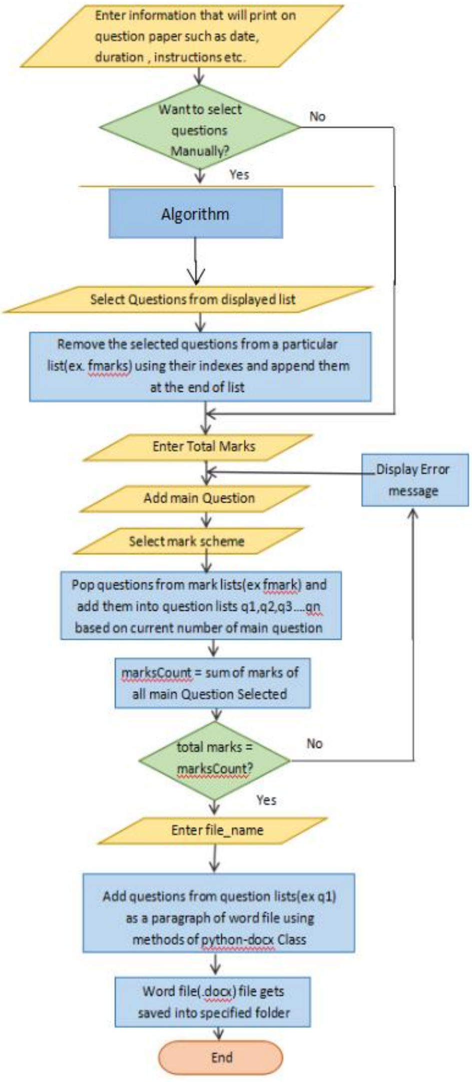

Enter information that will print on question paper such as date, duration , instructions etc.
Want to select questions Manually?
No
Yes
Algorithm
Select Questions from displayed list
Remove the selected questions from a particular list(ex. fmarks) using their indexes and append them at the end of list
Enter Total Marks
Display Error message
Add main Question
Select mark scheme
Pop questions from mark lists(ex fmark) and add them into question lists q1,q2,q3....qn based on current number of main question
marksCount = sum of marks of all main Question Selected
total marks = marksCount?
No
Yes
Enter file_name
Add questions from question lists(ex q1) as a paragraph of word file using methods of python-docx Class
Word file(.docx) file gets saved into specified folder
End

Fig 1.2. Flowchart of QPGS(Part 2)

The flow chart illustrates the flow of process of generating question paper from adding questions to the database to generating printable question paper. The administrator creates users and specific roles for each of them. The authorized user has to log into the system first. The authorized users are restricted to access database and add/delete/update questions to/from it. When user has a successful login. The user has to select any one option between generating the question paper and creating the database. The user has to define subject from a predefined course. For generating a question paper user has to input total marks and select chapters. The system automatically generates a skeleton. In the skeleton, the user can change the difficulty level and distribution of marks. After the selection difficulty level of question, paper user has to click on generate the question paper button. And the format of question paper will appear to the user in which all the questions that were selected while question the paper generation process. For creating database .there are some steps that authorized user has to follow such as course selection, subject selection and for adding or removing the questions in database user has to select in which chapter user wants change, update or delete.

5. IMPLEMENTATION

A. Working

The system has two portals i.e. one for adding questions to database and other for generating question paper. For both processes, the first user has to select semester and subject from comboboxes. While generating question paper user has to specify some information such as date, duration, instructions, etc. The system will then fetch all the questions from selected subject table from database where the chapter column is specified modules. These questions are stored along with their weightage in the list "listQ" in the form of tuple eg. listQ = [(question1, marks),(question2, marks)....]. Now the system will segregate these questions into lists fomarks, fmarks, smarks, emarks and tmarks based on their weightage. For example, all the questions with 5 marks will go into list fmarks. The system will then ask the user if they want to generate paper manually. If yes, a new window opens which contains a list of all the questions loaded into pyQt5's Qlist element. From here user can add questions that he/she wants to give the most priority. These questions are then removed from the lists based on their indexes and appended at the end of lists. If user chooses to "generate automatically" the above-described step is skipped. The user now specifies the total marks and add the main question. After adding main question user has to select mark scheme for that particular main question for example if user selects 10 then he/she has to choose between (5,5) and(10) marks implying the question will contain either two sub questions with 5 marks weightage or one sub question with 10 marks weightage. Now the system pops questions from lists(for eg, two times from fmarks if (5,5) is selected) and adds them into a list named Q1. System checks if the count of marks of the main question selected is equal to the total marks defined earlier. Users can add more questions based on requirement and the same process is repeated. When user is satisfied with the skeleton generated he/she can proceed to define filename. Word document is created by adding questions from lists Q1,Q2....Qn using methods from python-docx class. Users can then view and print the document.

B. Algorithm

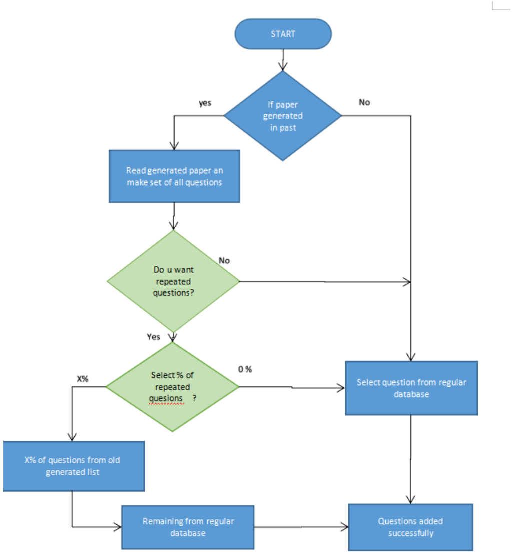
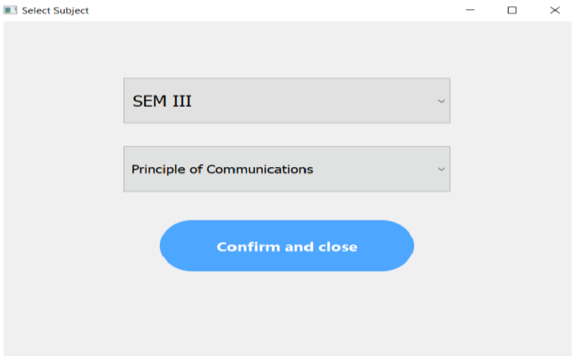

START
If paper generated in past
yes
No
Read generated paper an make set of all questions
Do u want repeated questions?
No
Yes
Select % of repeated quesions ?
X%
0 %
Select question from regular database
X% of questions from old generated list
Remaining from regular database
Questions added successfully

Fig 2. Algorithm of QPGS

The algorithm that we have created takes all the question papers that are previously generated into consideration. This algorithm is applied after selection of modules of particular subject.

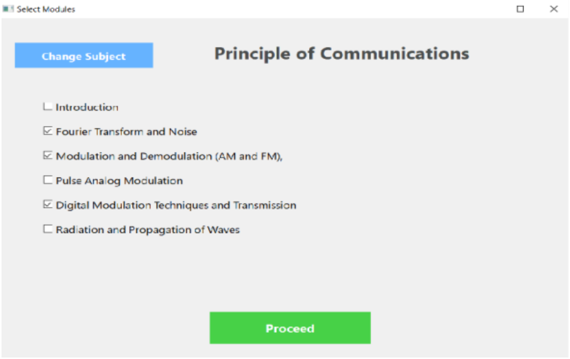
The system checks for previously generated question papers from the unique folder given to the subject. If there exists such file/files in that folder, they are scanned and questions are procured which are stored in a set (let's say A) in the form of tuples which contains question and their weightage at index 0 and 1 respectively. In the next procedure the user has to insert percentage of questions that are to be repeated from previously generated papers. Lets take this value X. Then system select X% of questions from A set and store them into a new list B avoiding duplicates. System fetches questions from existing database and store them into list C. While generating question paper, if it requires N number questions then new list is created which contains all the questions from List B and (N - n(B)) questions from list C where n(B) represents number of questions in list B.



SEM III

Principle of Communications

Fig 3. Select Subject and sem window.



Principle of Communications

☐ Introduction
☑ Fourier Transform and Noise
☑ Modulation and Demodulation (AM and FM),
☐ Pulse Analog Modulation
☑ Digital Modulation Techniques and Transmission
☐ Radiation and Propagation of Waves

Fig 4. Module selection window

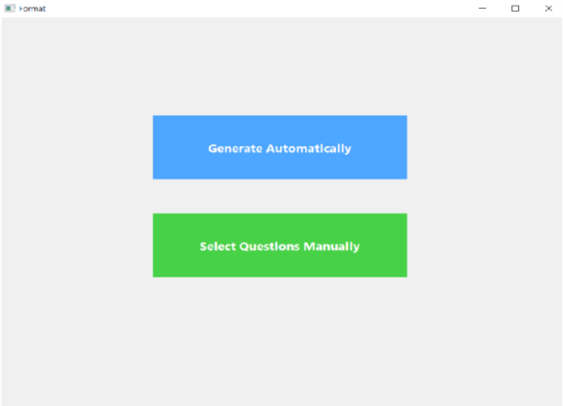
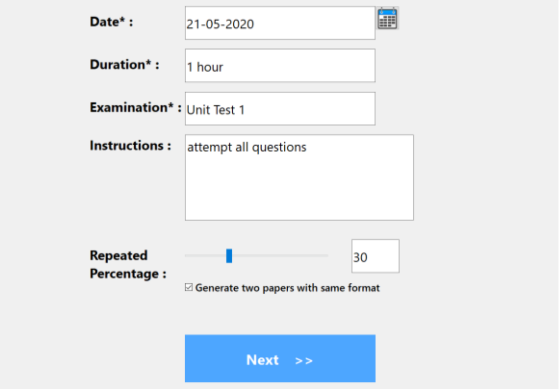


Fig 5. Selection of manual and automation.



Date* : 21-05-2020

Duration* : 1 hour

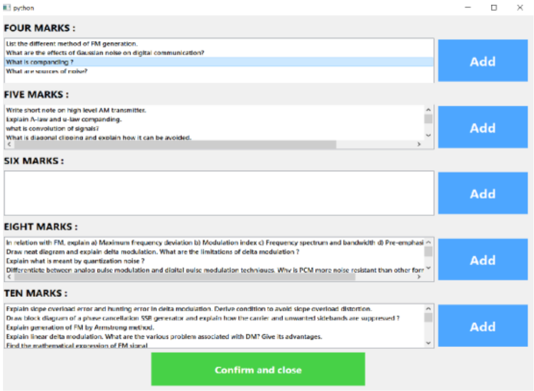
Examination* : Unit Test 1

Instructions : attempt all questions

Repeated Percentage : 30

☑ Generate two papers with same format

Fig 6. Selection of QP date and time window.



FOUR MARKS :

List the different method of FM generation.
What are the effects of Gaussian noise on digital communication?
What is companding ?
What are sources of noise?

FIVE MARKS :

Write short note on high level AM transmitter.
Explain A-law and u-law companding.
what is convolution of signals?
What is diagonal clipping and explain how it can be avoided.

SIX MARKS :

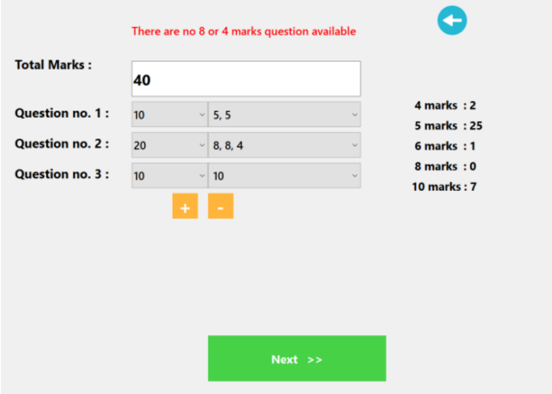
EIGHT MARKS :

In relation with FM, explain a) Maximum frequency deviation b) Modulation index c) Frequency spectrum and bandwidth d) Pre-emphasi
Draw neat diagram and explain delta modulation. What are the limitations of delta modulation ?
Explain what is meant by quantization noise ?
Differentiate between analog pulse modulation and digital pulse modulation techniques. Why is PCM more noise resistant than other forr

TEN MARKS :

Explain slope overload error and hunting error in delta modulation. Derive condition to avoid slope overload distortion.
Draw block diagram of a phase cancellation SSB generator and explain how the carrier and unwanted sidebands are suppressed ?
Explain generation of FM by Armstrong method.
Explain linear delta modulation. What are the various problem associated with DM? Give its advantages.
Find the mathematical expression of FM signal

Fig 7. Manual Question selection window.



There are no 8 or 4 marks question available

Total Marks : 40

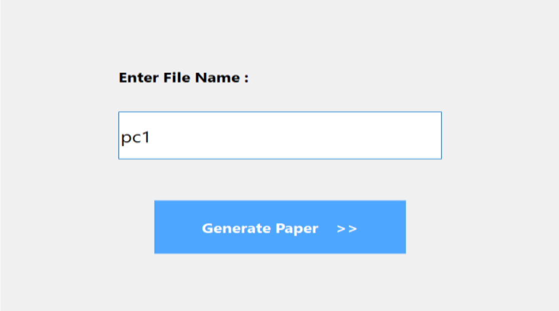
Question no. 1 : 10 | 5, 5

Question no. 2 : 20 | 8, 8, 4

Question no. 3 : 10 | 10

4 marks : 2
5 marks : 25
6 marks : 1
8 marks : 0
10 marks : 7

Fig 8. Selection of marks



Enter File Name : pc1

Fig 9. Entering of File name

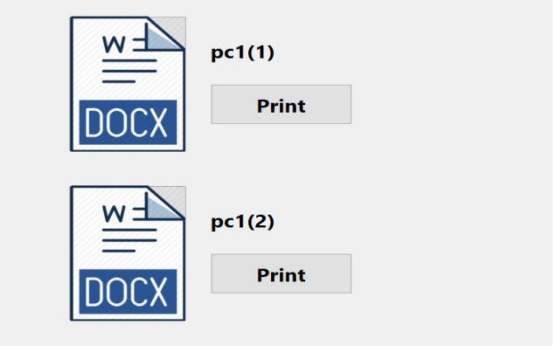


Fig 10. Printing window

6. CONCLUSION

We have develop a python based desktop application for automatically generating question papers. This system will be able to generate question papers based on skeleton of question paper which user have to decide. The total marks are to given as an input and system will automatically divide these marks into questions which are also to be described by the person who is generating. For developing this system we used scikit-fuzzy package of python for "fuzzy logic" algorithm. This system is more reliable in terms of duplicity removal, compromise issues, and security.

7. REFERENCES

[1] Vijay KrishanPurohit', Abhijeet Kumar', Asma Jabeen, Saurabh Srivastava, R H Goudar, Shivanagowda, "*Design of Adaptive Question Bank Development and Management System*", 2nd IEEE International Conference on Parallel, Distributed and Grid Computing, 2012.

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

[3] Noor Hasimah Ibrahim Teo, Nordin Abu Bakar and Moamed RezduanAbd Rashid, "*Representing Examination Question Knowledge into Genetic Algorithm*", IEEE Global Engineering Education Conference (EDUCON), 2014.

[4] Mojitha Mohandas , Aishwarya Chavan , Rasika Manjarekar and Divya Karekar, "*Automated Question Paper Generator System*", International Journal of Advanced Research in Computer and Communication Engineering, Vol. 4, Issue 12, December 2015

[5] Gauri Nalawade and Rekha Ramesh, "*Automatic Generation of Question Paper from User Entered Specifications using a Semantically Tagged Question Repository*", 2016 IEEE 8th International Conference on Technology for Education

[6] Prateek Pisat, Shrimangal Rewagad, Devansh Modi and Ganesh Sawan, "*Question Paper Generator and Answer Verifier*", International Conference on Energy, Communication, Data Analytics and Soft Computing (ICECDS-2017)

[7] Tejas Barot and Poornima Salunke "*Automatic Question Paper Generator System*", International Journal of Scientific Research Engineering & Technology (IJSRET), Volume 6, Issue 4 , April 2017

[8] Amit Sanjay Khairnar, Bhagwat Chintaman Jadhav, Rahul Birhade, Pramod Patil, "*Automatic Question Paper Generator*", International Journal For Technological Research In Engineering Volume 4, Issue 9, May-2017

[9] Prof. Mrunal Fatangare, Rushikesh Pangare, Shreyas Dorle, Uday Biradar, Kaustubh Kale,"*Android Based Exam Paper Generator*", Proceedings of the Second International Conference on Inventive Systems and Control (ICISC 2018)