QUESTION PAPER MAKER

Submitted in partial fulfillment of the requirements

of the degree of

Bachelor of Engineering in Information Technology

By

NIKHIL DHANGADA (15101A0047)

TIRTHESH PATIL (20101A2009)

SANCHIT JADHAV (20101A2010)

Under the Guidance of

PROF. DEBARATI GHOSAL

Department of Information Technology

VIT Vidyalankar Institute of Technology
ACCREDITED A+ BY NAAC

(An Autonomous Institute Permanently Affiliated to Mumbai University)

Vidyalankar Institute of Technology

Wadala(E), Mumbai-400437

University of Mumbai

2022-23

CERTIFICATE OF APPROVAL

This is to certify that the project entitled

“QUESTION PAPER MAKER”

is a bonafide work of

NIKHIL DHANGADA (15101A0047)

TIRTHESH PATIL (20101A2009)

SANCHIT JADHAV (20101A2010)

submitted to the University of Mumbai in partial fulfillment of the requirement for the award of the degree of

Undergraduate in **“Information Technology”**.

Guide	Head of Department	Principal
Prof. Debarati Ghosal	Dr. Vipul Dalal	Dr. S. A. Patekar

Project Report Approval for B. E.

This project report entitled **Question Paper Maker** by

1. ***Nikhil Dhangada (15101A0047)***
2. ***Tirthesh Patil (20101A2009)***
3. ***Sanchit Jadhav (20101A2010)***

is approved for the degree of ***Bachelor of Engineering in Information Technology.***

Examiners

1.----------

2.----------

Date:

Place:

Declaration

I declare that this written submission represents my ideas in my own words and where others' ideas or words have been included, I have adequately cited and referenced the original sources. I also declare that I have adhered to all principles of academic honesty and integrity and have not misrepresented or fabricated or falsified any idea/data/fact/source in my submission. I understand that any violation of the above will be cause for disciplinary action by the Institute and can also evoke penal action from the sources which have thus not been properly cited or from whom proper permission has not been taken when needed.

Name of student	Roll No.	Signature
1. Nikhil Dhangada	15101A0047	
2. Tirthesh Patil	20101A2009	
3. Sanchit Jadhav	20101A2010	

Date:

ACKNOWLEDGEMENT

We are honored to present "**Question Paper Maker**" as our B.E Final Year Project. We are using this opportunity to express our profound gratitude to our principal "**Dr. Sunil Patekar**" for providing us with all proper facilities and support.

We express our deepest gratitude towards our HOD "**Dr. Vipul Dalal**" for his valuable and timely advice during the various phases in our project. We would like to thank our project guide "**Prof. Debarati Ghosal**" for support, patience, and faith in our capabilities and for giving us flexibility in terms of working and reporting schedules. Finally, we would like to thank everyone who has helped us directly or indirectly in our project.

We would also like to thank our staff members and lab assistant for permitting us to use computers in the lab when required. We thank our college for providing us with excellent facilities that helped us to complete and present this project.

Nikhil Dhangada

Tirthesh Patil

Sanchit Jadhav

ABSTRACT

The knowledge of a student is examined at every stage from the day he/she starts their journey to acquire it. Continuous examination is required for testing the student's learning ability in a particular domain. Traditional way of generating question papers can't fulfill such continuous examination process. Most of the universities still follow traditional way for generating question papers. In our survey we found that some Questions are frequently repeated in every year's paper which are marked as important and also generating paper manually is a time-consuming task. It's a difficult task to create a paper which can cover all the requirements required for testing the knowledge acquired by the students in a particular subject/field. Most of the time papers which are generated may have same questions as professor's priorities the question based on their importance which help students to find out which question may occur based on previous years questions paper. So, to avoid the prioritization of question and to provide a variety of question we proposed a system named as question paper maker which makes use of natural language processing for generating a series of question papers based on student's previous marks and advancing the quality of paper at each iteration. The proposed system, Question Paper Maker can be used to overcome the drawback of traditional method and helps to generate a better-quality paper which will cover the domain requirements. Question Paper maker can generate theoretical papers based on adaptive evaluation. A student can also use Question paper maker as a practice tool for preparation of competitive exams.

TABLE OF CONTENTS

Source Code Link:

https://github.com/tirthesh2001/ProjectDarwin.git

LIST OF FIGURES

CHAPTER 1

INTRODUCTION

Introduction

The Question Paper Maker is a solution built for generating a series of question papers, each having unique questions in every question set. The proposed system avoids the prioritization of question and to provide a variety of question using natural language processing for generating a series of question papers based on student’s previous marks and advancing the quality of paper at each iteration.

Students can use the answer to practice, and teachers can use it to make different sets of test questions. Teachers can use the solution to generate test questions based on the guidelines that must be followed while creating a test. It automates the process of delivering a collection of question papers to the moderator, from which the moderator chooses the paper that will serve as the university's final exam.

To understand the context of the information given to the software and put it together, the solution uses NLP-based models and concepts. The questions are created here so that the examinee can try their hand at them. The system makes an effort to use textual data as input and derive the context from it in order to generate the questions. If the user provides the URL of the relevant online page, the system can also produce quizzes by scraping questions from websites. Giving the examiners the opportunity to develop questions based on desired content rather than a current topic. The software's examination portion will take on this form. The submitted question set is cleaned by removing the irrelevant or incomplete questions. The categories used to group the questions include difficulty level, logical basis, theoretical basis, numerical basis, etc. With the use of NLP models, classification of questions is carried out taking into account these factors, and each question is given a difficulty rating.

1.1 Problem Statement

Construct a system that can use adaptive evaluation and NLP (Natural Language Processing) to generate questions based on the subjects from user-provided information sets or by scraping questions for a certain website to generate a question set. Utilizing NLP, the solution must categorize the website scraped questions according to the topic and rate the difficulty of each question. The questions must not appear more than once on the created question paper. All of the question sets that are generated based on the subject should be managed by the solution. For additional evaluations, it is necessary to record every user's response from each evaluation. The solution should manage all the information pertaining to the examinee, acting as a fictitious information management system. The suggested remedy for the current issue should thus be an exam instrument that adapts as the exam goes along and handles all the pertinent data at once.

1.2 Scope

The primary objective of the proposed project is to develop software that can serve as a proof of concept. A working prototype that can be used to easily and accurately produce question papers that meet university standards while removing human error. Our goal is to develop a system that can cover the full curriculum for the chosen subject and generate original test questions for each iteration. Another goal is to create a system that allows students to potentially attempt test questions by providing the subject's historical test scores, which determine how difficult the questions are adjusted.

1.3 Motivation

The difficulty of implementing the idea of AE the minute scale of individual questions was the primary driver behind the problem statement selection. It also appeared intriguing that NLP would be a viable remedy for the software's inability to generate original queries for each specific data collection. This enormous task seemed like the right place to challenge we to brainstorm a solution and to learn the necessary skills to implement a solution, if not, a proof-of-concept type prototype of the solution. The proposed solution is not only required to be an examination tool that can generate questions and implement AE but also for it to handle all the examinee related information for each individual examinee.

CHAPTER 2

LITERATURE SURVEY

2.1 Survey of Existing System

"Automatic generation of question papers from user entered specifications using a semantically tagged question repository" is the idea put out by G. Nalawade and R. Ramesh [1]. The question generating in this system is based on a question repository with semantic tags. In the form, which is subsequently transformed into a word document, the person who is going to make the paper must manually enter the parameters for creating question papers.

The "Automatic Question Paper Generator System" that Zalte S.V., Jadhav C.C. et al. [2] suggested was implemented in C#. To avoid asking the same questions repeatedly, the author employs randomization approach. The old system automatically generates the exam questions, which are stored in a database. Its query must be accompanied by a statement about the difficulty and priority.

Shrimangal Rewagad, Pratik Pisat, and others [3] The "Question Paper Generator and Answer Verifier", Both subjective and objective questions are generated by the system. The subject name and level of difficulty must be entered by the user before a paper may be generated. The score for an objective exam is continually computed and stored in the database.

Igor Khokhlov, Akhil Killawala, and others. [4] The system uses a rule-based methodology and the "Computational Intelligence Framework for Automatic Quiz Question Generation" to generate quiz questions that are semantically correct. It produces four different types of questions, including MCQs, True/False, Fill in the Blanks, and "Wh"-style inquiries.

Amit Khairnar, Pramod Patil, Bhagwat Jadhav, and others. [5] The "Automatic Question Paper Generator" was created in Microsoft Visual Studio and stores the questions on Microsoft SQL server. When the question is entered during the paper generating process, the question paper is generated quickly. When creating questions, the system use shuffling algorithms to prevent recurrence and duplication.

Rami Hodrob el at., Ashraf Amria, and Ahmed Ewais. [6] A system that can choose questions based on desired learning objectives has been proposed in "A Framework for Automatic Exam Generation based on Intended Learning Outcomes". The factors of question variety, randomization, mark distribution, and question-to-learning outcome mapping were taken into account when creating the article. The questions and associated desired learning outcomes are stored in a web-based system that was created using PHP and MySQL as the database.

2.2 Drawbacks of Existing Systems

The drawback of the above-mentioned systems is that they required question set from the user with specific parameters set with every question present. So, because of this we cannot clean the question even if any mistakes were present in the question set provided. Question paper generated are based on a static paper pattern. User cannot get a modified paper pattern in the above specified systems. A huge dataset cannot be used as it may affect the performance of the system. Only one paper is generated at a time, and we cannot have multiple papers of subject as well as objective type. They implement AE on a section basis not based on the result of individual questions. These systems are also examination tools who handle information on a larger scale and provide rankings, scores along with other information. They do not provide personalized information for the individual user except for the information about their performance in the said examination. These systems are also inclined towards organizational applications. Which brings up the point of there not being a personalized examination tool.

CHAPTER 3

SYSTEM DESIGN

3.1 Proposed Solution

The proposed system consists of the examination system, a question generation system, a login-registration view for users, an information processing module, a user dashboard for the user to access all the information related to them and an about us page combined with a contact us view. The system will consist of two main modules. The first module will be the questions dataset cleaning and assigning parameters required for generating question paper. And the second is the question paper generation module.

3.1.1 Questions Dataset Module

In this module, the questions can be scrapped from a website or can be taken from the teacher/student for generation of the paper. A single dataset is maintained for all the questions related to the subject. The questions are scrapped and cleaned using NLP techniques. Cleaning includes removing duplication, null values, incomplete questions, etc. Questions are assigned with the difficulty level based on bloom's taxonomy. The questions dataset can be displayed along with its level of difficulty in the system once it is generated. The dataflow of the question set generation is shown in figure 1.

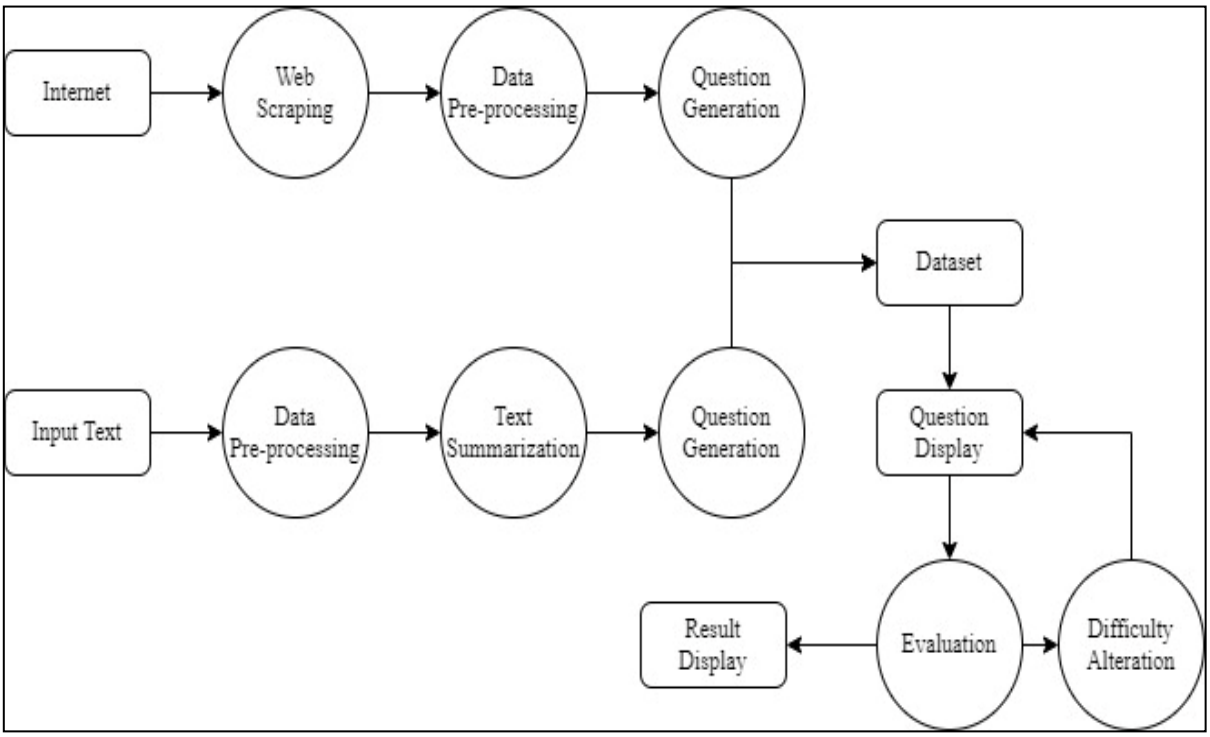


Fig 3.1 - Dataset Generation Flow

3.1.2 Question Paper Generation Module

This module will generate one or multiple questions papers based on the parameter which are considered while generating a final exam paper. Generally multiple question sets are presented in front of the moderator from which the moderator selects one. This module will do the same task automatically. The system will generate question sets considering the constraints and will present these sets in front of the moderator. The question paper sets generated are unique in their composition. The format of the paper will change based on the total marks provided.

The proposed system is built on python as the base for the backend and makes use of HTML (Hyper-Text Markup Language), CSS (Cascading Style Sheets) and JavaScript for the frontend. Thus, it being completely open source and having a scope to be extended as whatever and however one desires to.

3.1.3 Web-end User Module

The website end of the module works as an interface between the background working question generation modules and the user who's input highly affects the desired output; which in our case is the question paper itself. The web-end also handles the user login/ logout and other operations which require the user in some degree. The system makes use of SQLite Database for managing whatever data is generated locally.

3.2 Flow of the Solution

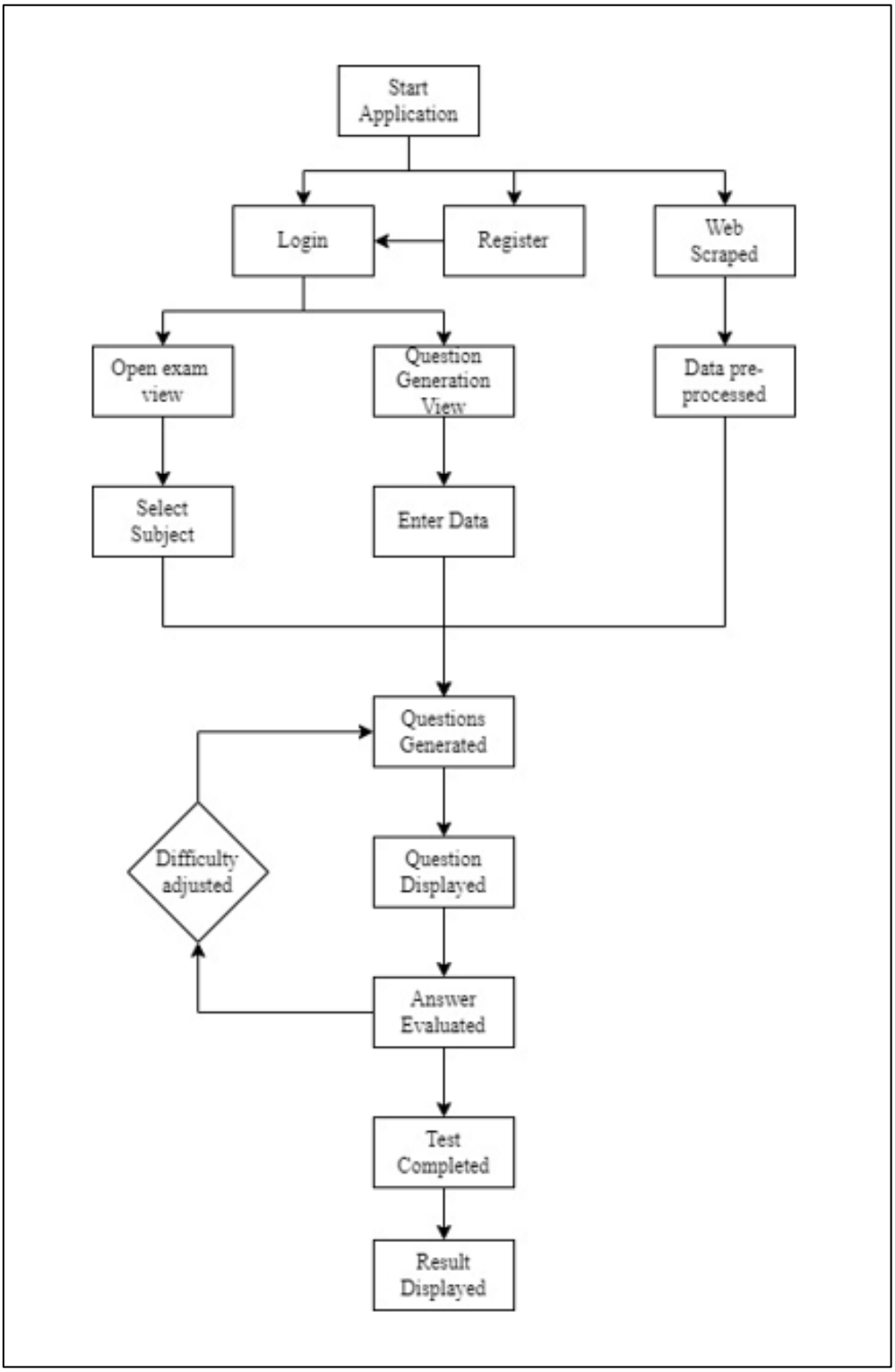


Fig. 3.2 - Flow of the Solution

3.3 Methodology

The proposed system is going to work on a few actions parallelly, these core actions are displayed in the flow of the system as shown below.

The solution has multiple background worker modules each of which is tasked with solving one part of the puzzle. These modules run parallelly to the frontend of the solution which looks after communicating with the user. The process starts up top with the scrapper file reading the links it is provided. The web scrapper goes to each of the given website hyperlinks and extracts the entire HTML source of the said web pages.

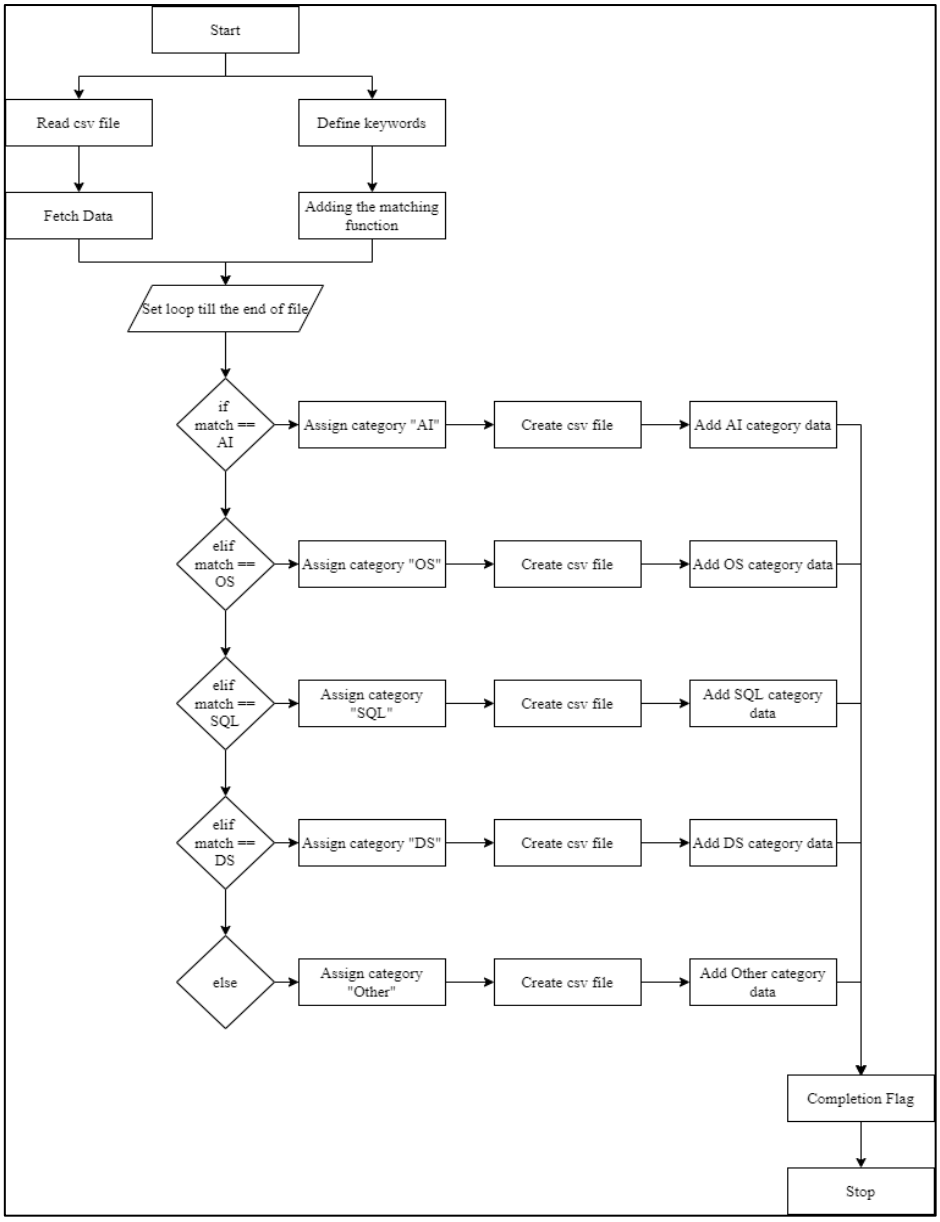


Fig. 3.3 - Sample flow of scraped data processing

This source of the web pages is sorted based on their tags and the relevant data is form there is extracted. The need for web scrapping is the inherent lack of dataset there is either about the contents one could study for a subject or the questions one could expect for a certain subject. For web-scrapping we go through websites which contain a lot a technical data. Once the relevant data is extracted the same data is cleaned. The process of data cleaning is handled by a different background worker module.

Here, cleaning of data entails that the data is check for redundance and for blank spaces. Both of which the discarded. All the textual data is checked for if they contain any remade questions. If they do, these questions are stored in a CSV file as a dataset. The rest of the data is also processed for its context and questions are created out of the same. All of these questions are stored in a CVS file. This dataset as shown in Figure 3.3, undergoes an extensive process of classification based on the difficulty of the question and also the subject of the question. Bloom taxonomy is carried out to classify the questions which is followed by the questions being divided into different datasets. Which makes it convenient for the system to handle the questions moving forward. The textual data in the datasets is used to generate MCQs of varied difficulty in a similar manner.

While this is happening, the user can access the frontend. Which starts with the Home page which navigates to the tests, about us and contact us pages. The Test's page requires a detour of login and if the user doesn't have an account, then registration as well. Once the user has logged in, they are provided with two options based on what the user aims to do with system they can choose any. The first option takes the user to the practice question paper module. This module lets the user generate MCQ type questions or theory-based questions, with variable difficult and a subject of their choosing. The main aim of the module is optimal question generation thus, it doesn't fuss about too many details when generating questions. Here for generating questions another background worker module jumps in. It fetches the question from the datasets as per the user demands. These questions are then used as based used to generate similar questions which are then fed to the user in the form of question papers. This helps the solution in providing unique question papers every time the user demands for one.

The other option in a Tests page navigates the user to a setup which facilitates the user to generate a question paper which is fine tuned to every possible detail. Here, the user can select the subject for the question paper, the number of easy, medium and hard difficulty of questions that the user requires and also is the user requires multiple sets of the question papers to be generated.

If the user requires multiple sets of papers, then it is made sure that the papers are unique in nature while maintaining the same paper pattern and also having certain similar questions.

All background worker classes and the frontend website view together make sure that the solution works fairly in generating optimal questions and abides to the scope and problem statement as well.

CHAPTER 4

ANALYSIS

4.1 Process Model

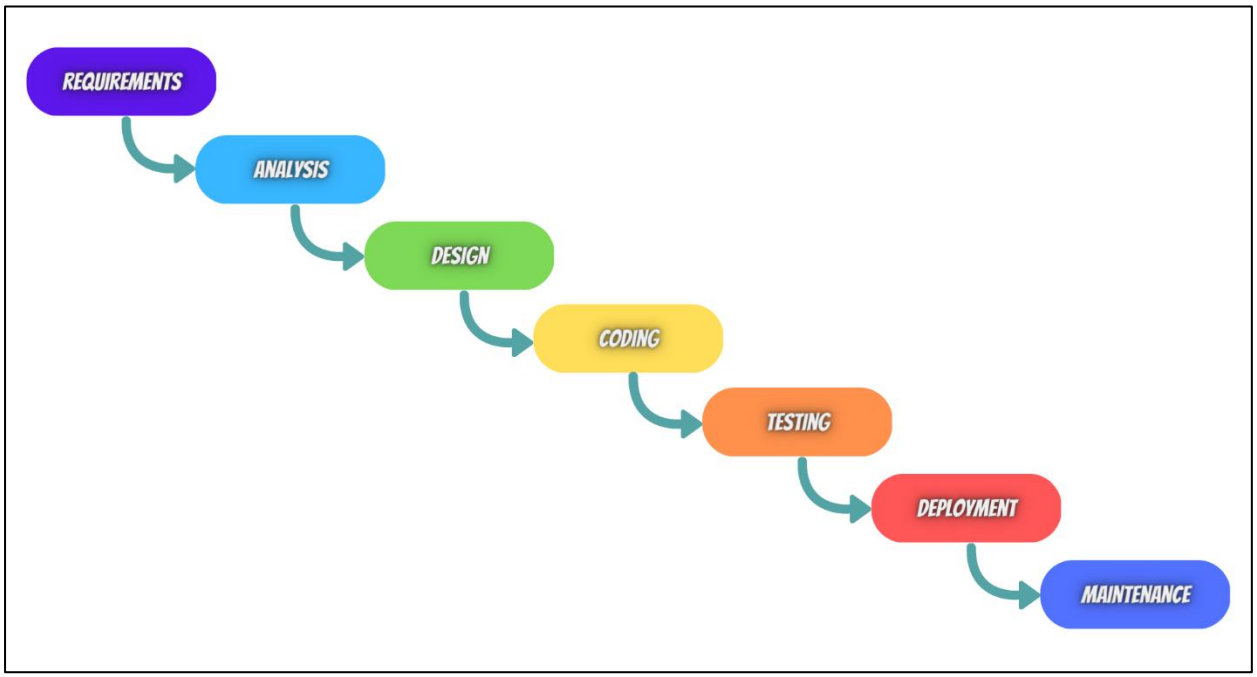


Fig. 4.1 - Waterfall Model

After overall discussion we decided to use simplest yet efficient software engineering model to develop our project.

The waterfall model is a classical model used in system development life cycle to create a system with a linear and sequential approach. It is termed as waterfall because the model develops systematically from one phase to another in a downward fashion.

This model is divided into different phases and the output of one phase is used as the input of the next phase. Every phase must be completed before the next phase starts and there is no overlapping of the phases.

The Phases of Waterfall model are:

- **Requirement Gathering and Analysis:** In this phase, the development team captures all possible requirements of the system to be developed and documents them in a requirement specification document.
- **Design:** The requirement specifications are studied in this phase and the system design is prepared. This system design helps in specifying hardware and system requirements and defining the overall system architecture.
- **Coding:** The actual coding is done in this phase. The system is developed in small programs called units, which are integrated in the next phase. Each unit is developed and tested for its functionality, known as Unit Testing.

- **Testing:** All the units developed in the implementation phase are integrated into a system after testing each unit. Post integration, the entire system is tested for any faults and failures.
- **Deployment:** Once functional and non-functional testing is completed, the product is deployed in the customer environment or released into the market.
- **Maintenance:** Patches are released to fix issues that arise in the client environment, and better versions are released to enhance the product. Maintenance is performed to deliver these changes in the customer environment.

4.2 Feasibility Analysis

- **Technical Feasibility:** By all means it is very possible to build the proposed solution. Technically, there models and libraries present which can aid the solution in making sure it meets the aim and objectives set by the problem statement. The most difficult tasks in hand with the proposed system are:
  - Generation of questions from the extracted textual context.
  - Making sure that the generated questions are valid.
  - Building a model which can vary the difficulty as per the problem statement demands.
  - Making sure that the system has quick processing times.

- **Behavioural Feasibility:** The proposed system will have positive feedback on personal level as it works as a tool which bridges the gap between online examinations and the students. With an intermediatory application in between which helps the students in preparing for the highly unpredictable online examination the solution is bound to be accepted by its target users.

- **Economic Feasibility:** The proposed system is economically feasible as only cost incurred on the solution will be the human effort. The system is proposed to be built completely using python and its libraries which makes it completely open source as well. This helps it being extremely pocket friendly.

4.3 Gantt Chart

Task No.	Task Name
1	Identification of Problem Statement
2	Analysis of existing systems and Literature Review
3	Gathering Resources
4	Project Planning
5	Determining Methodology
6	Preparing project design and Formulating SRS for the said problem statement
7	Analyzing risk and preparing RMM plan
8	Prototyping (Preparing a Proof of Concept)
9	Review and Rework
10	Designing User Interface and User Experience
11	Building Background Worker Modules
12	Building the Website based on the designs
13	Integrating the website and the background worker modules
14	Testing the complete functionality
15	Adding finishing touches

Table 1. – Tasks

Gantt Chart

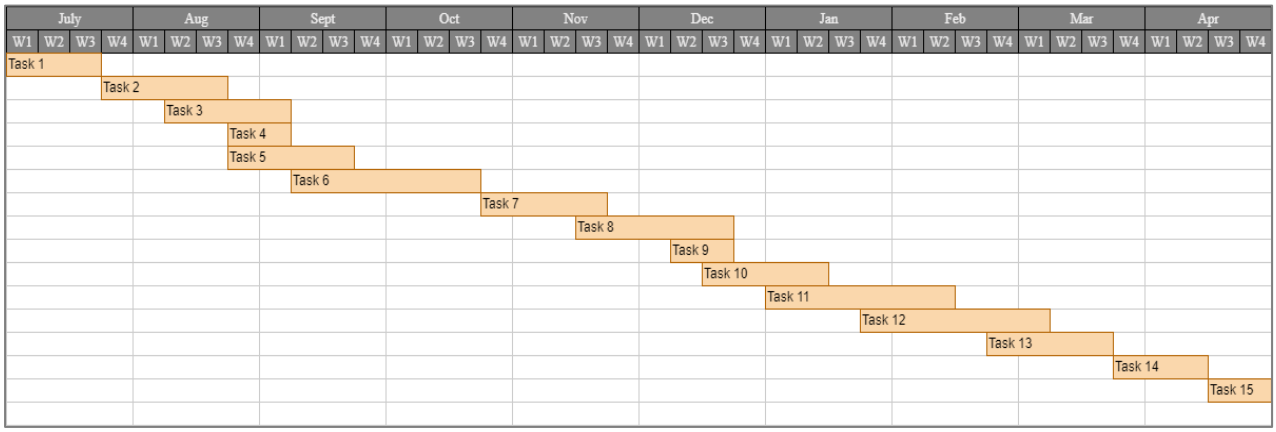


Fig. 4.3 - Gantt Chart

CHAPTER 5

DESIGN

5.1 UML Diagrams

5.1.1 Use Case Diagram for Students

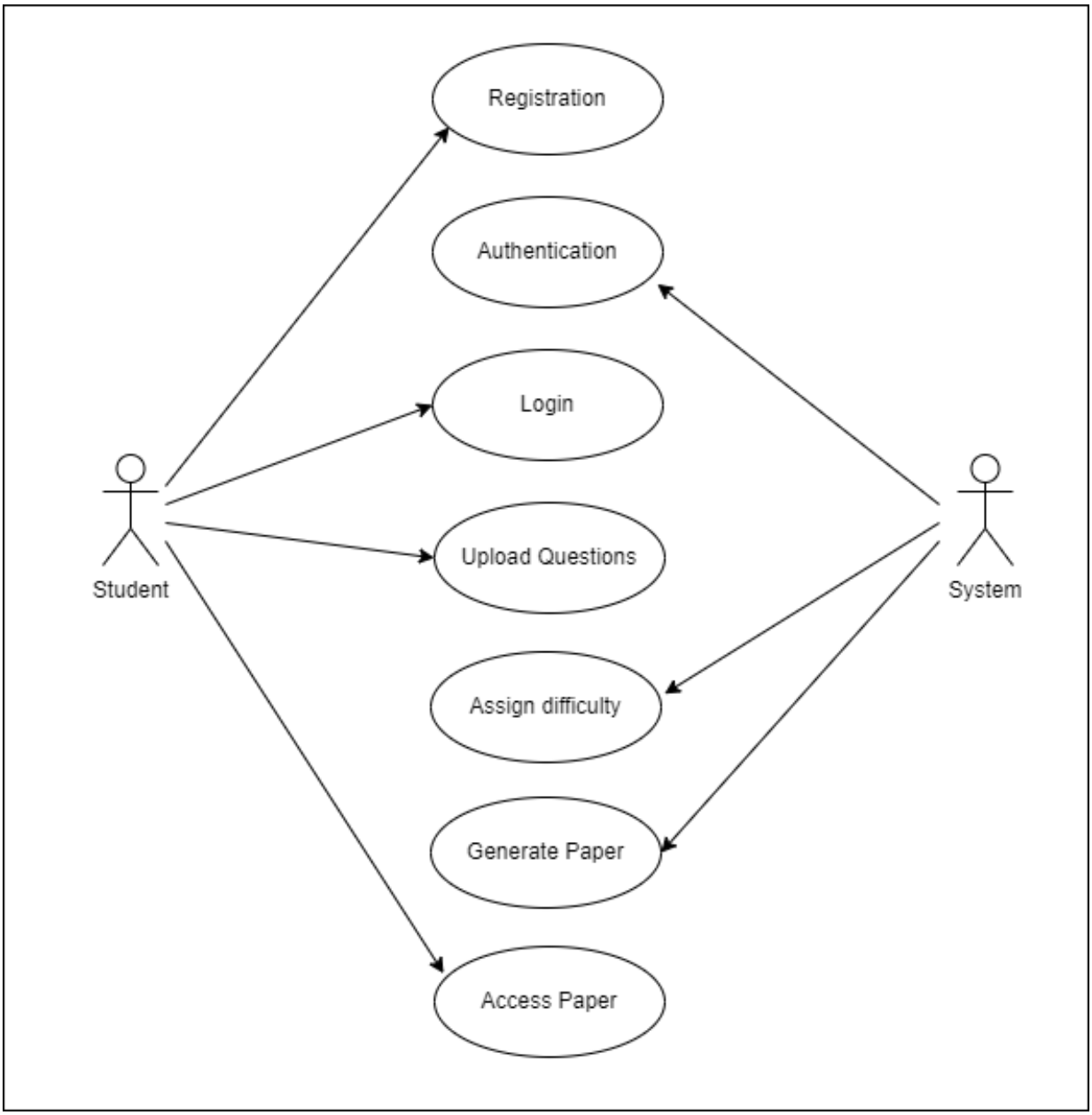


Fig. 5.1.1 - Use case diagram for students

5.1.2 Use Case Diagram for Other Users

Fig. 5.1.2 - Use case diagram for users

5.1.3 Package Diagram

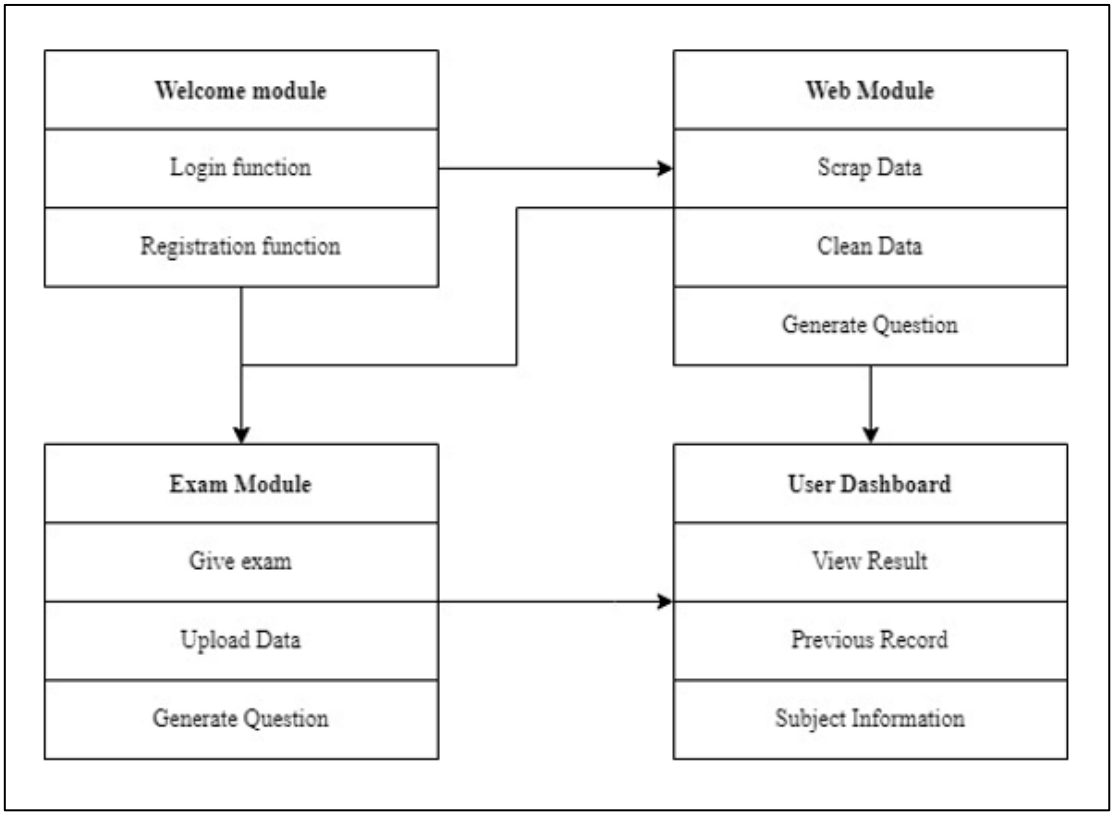


Fig. 5.1.3 - Package Diagram

5.2 Data Flow Diagram

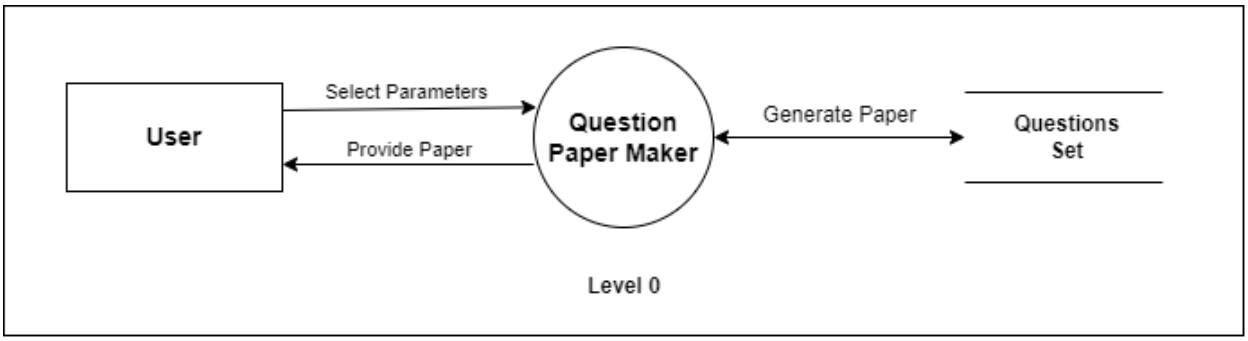


Fig. 5.2.1 - Level 0 Dataflow Daigram

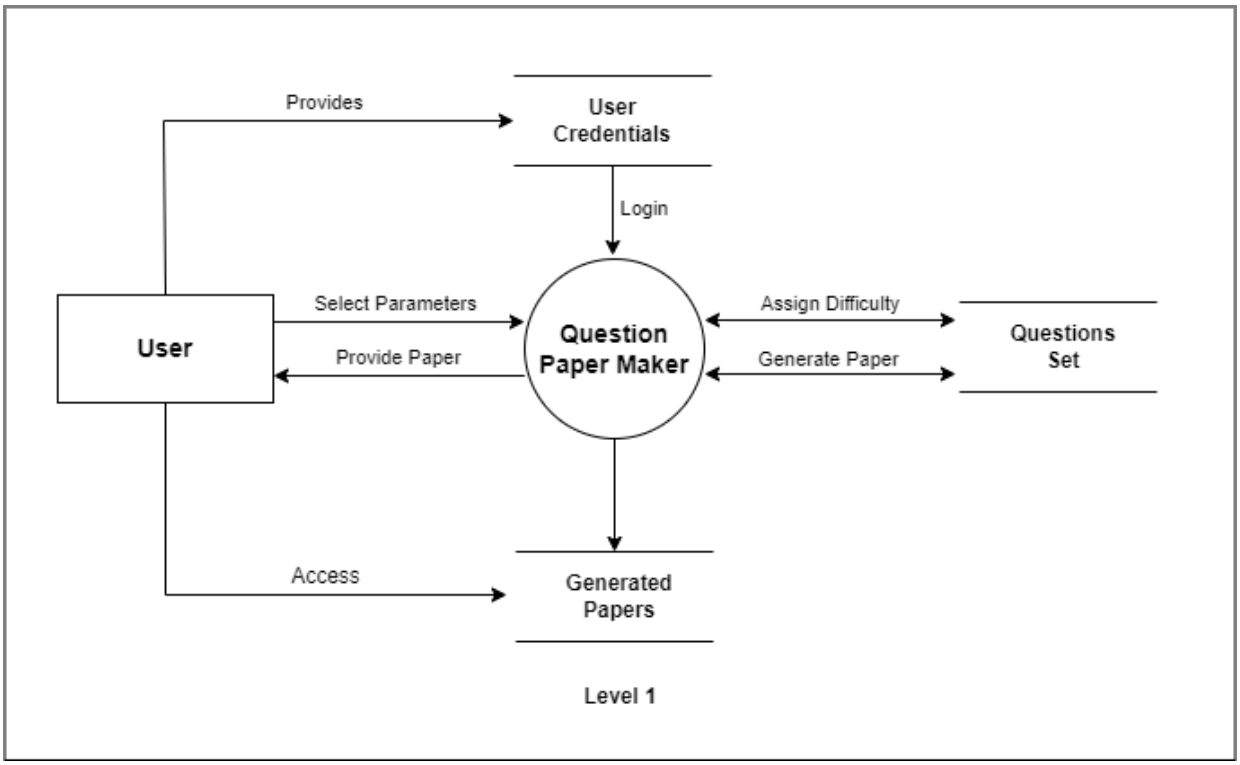


Fig. 5.2.2 - Level 1 Dataflow Daigram

5.3 E-R Diagram

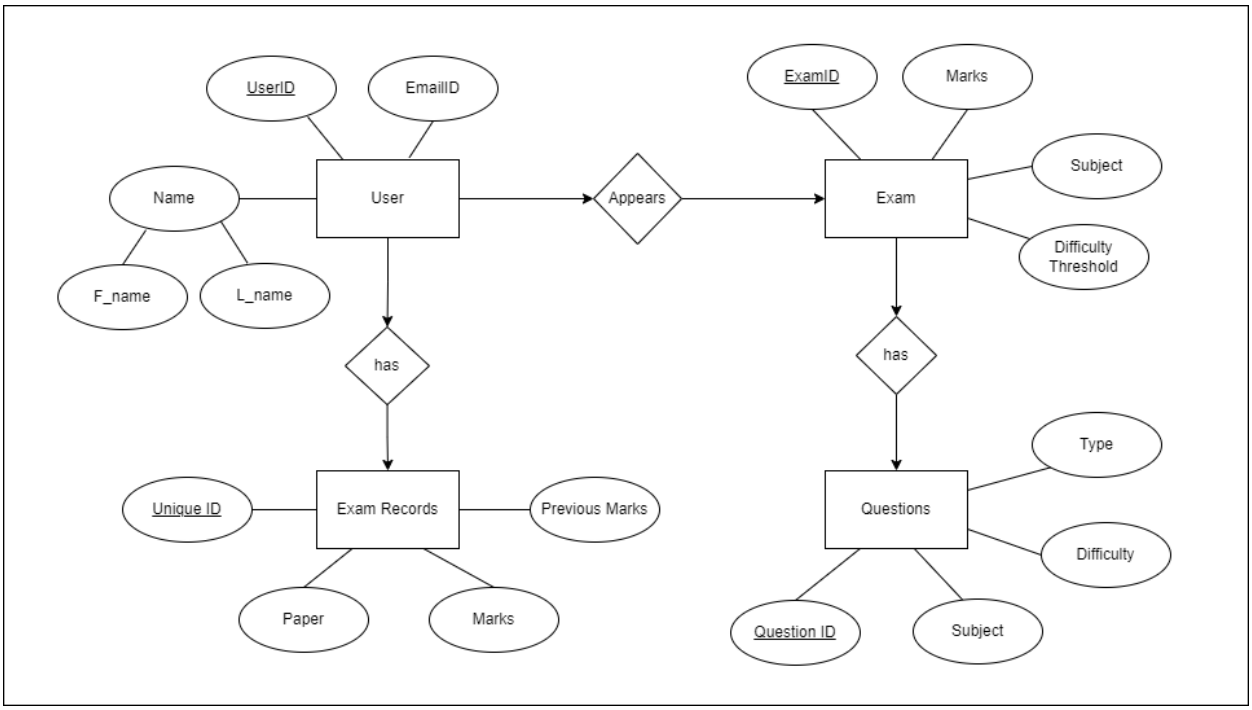


Fig. 5.3 - E-R Diagram

CHAPTER 6

RESULTS AND DISCUSSIONS

RESULTS AND DISCUSSIONS

Below are the output of the home interface and other modules present on the home page before logging into the system.

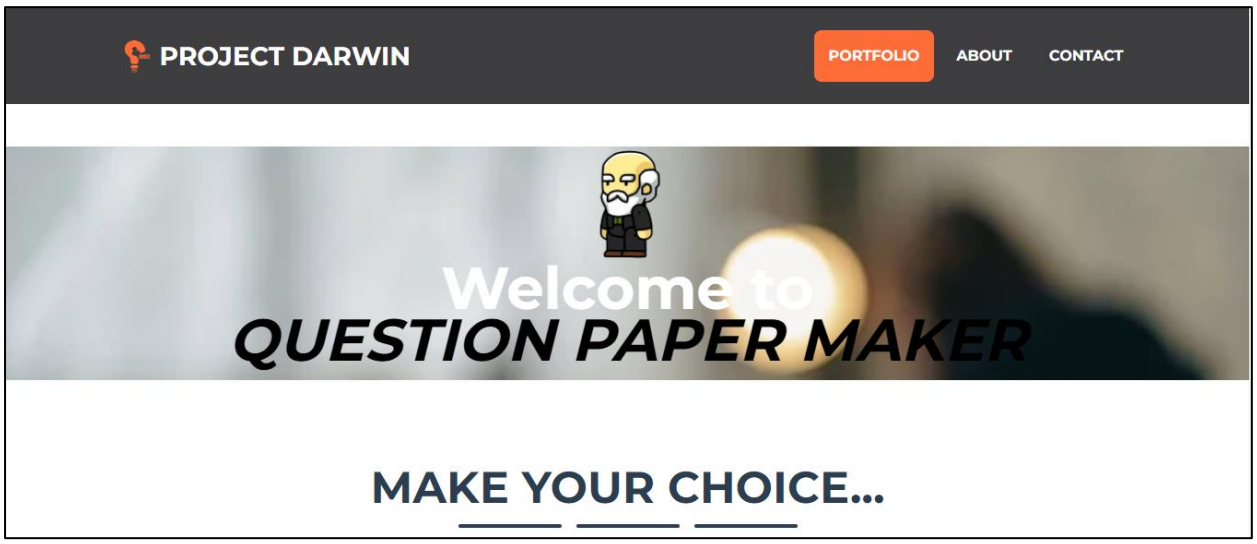


Fig. 6.1 – Home Page 1

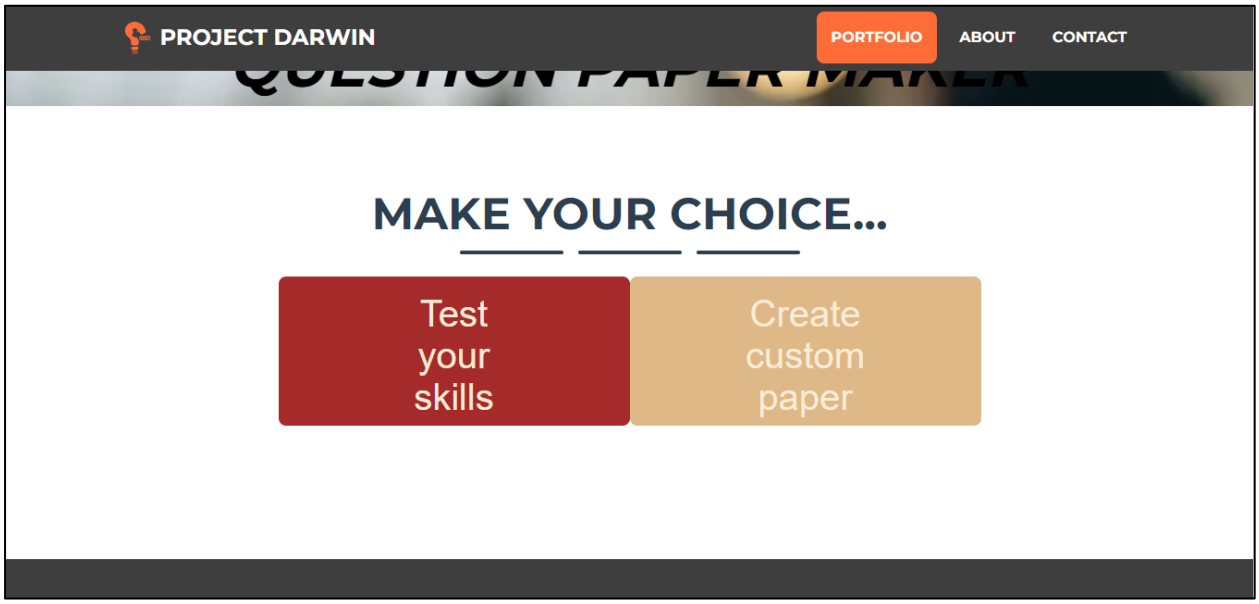


Fig. 6.2 – Home Page 2

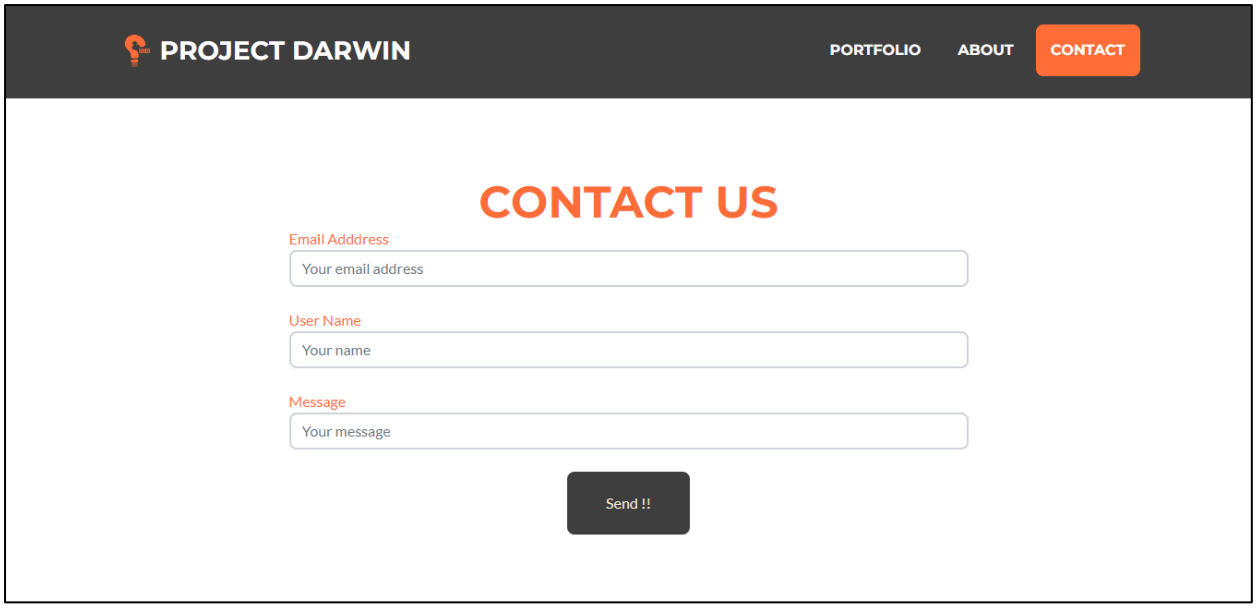


Fig. 6.3 – Contact Us Page

Fig. 6.4 - Footer

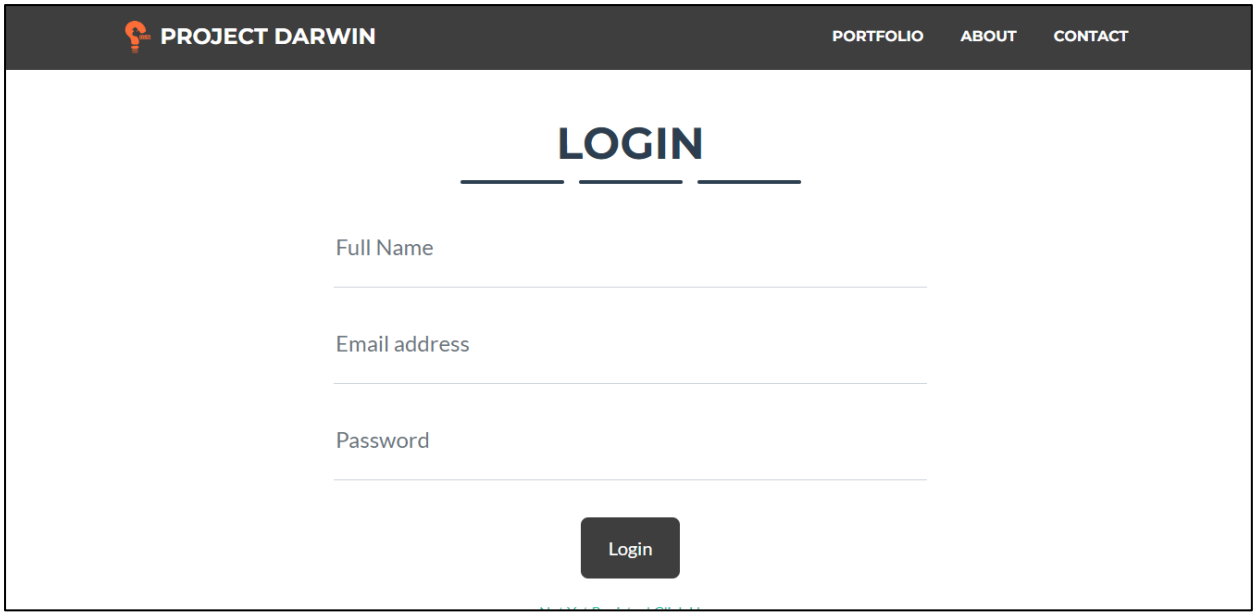


Fig. 6.5 – Login Page

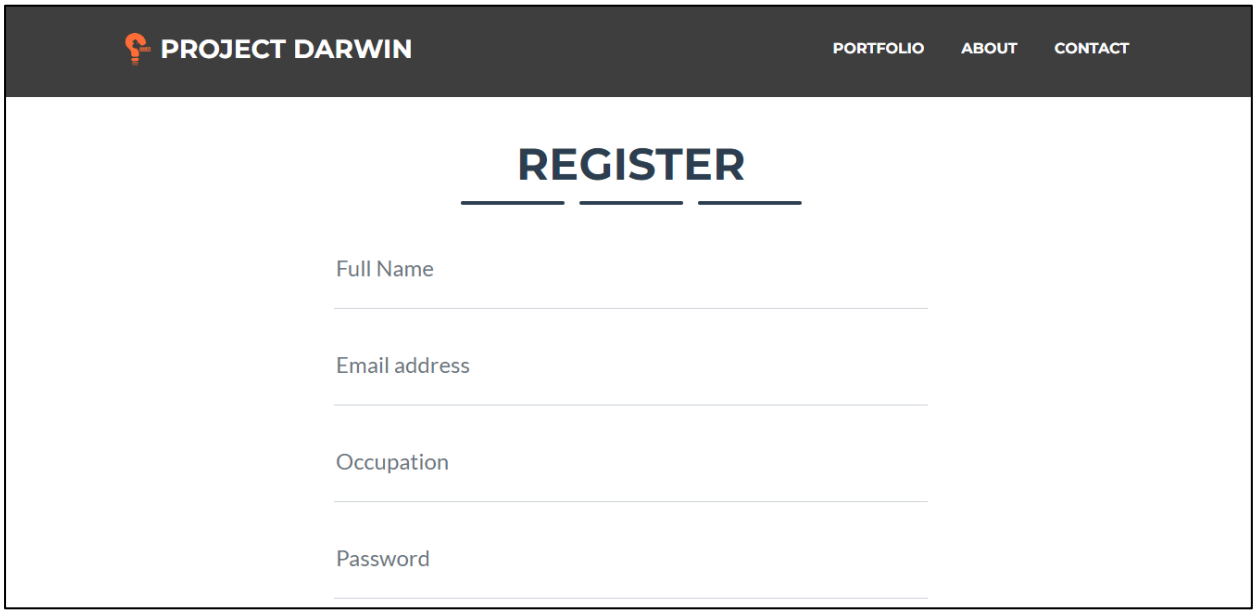


Fig. 6.6 – Registration Page (Part 1)

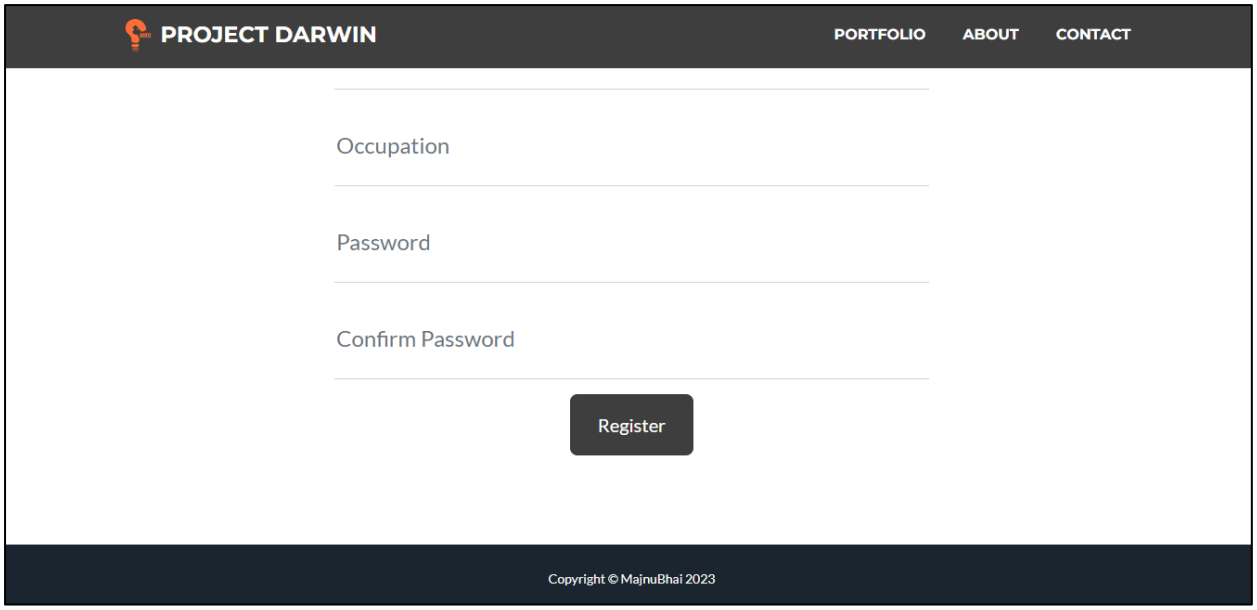


Fig. 6.7 – Registration Page (Part 2)

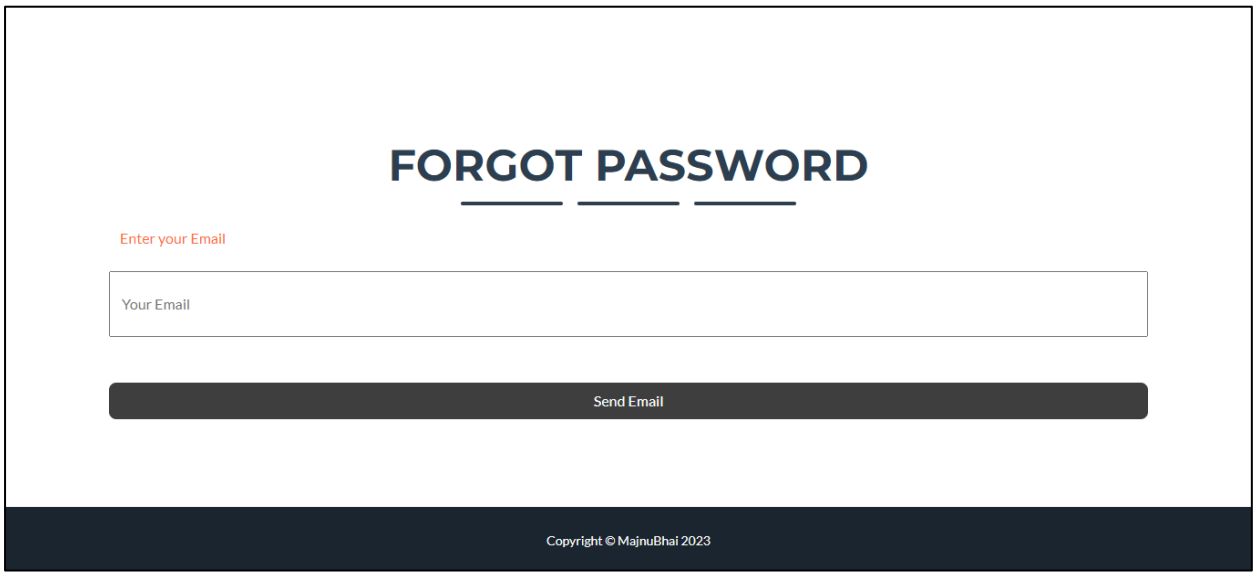


Fig. 6.8 – Forgot Password

- **Quick Test Page**

The question section page does not require external questions. It generates paper by getting questions from the dataset generated by scrapping from a website. This page has two options, one for MCQ paper generation and the other for Theory Paper generation.

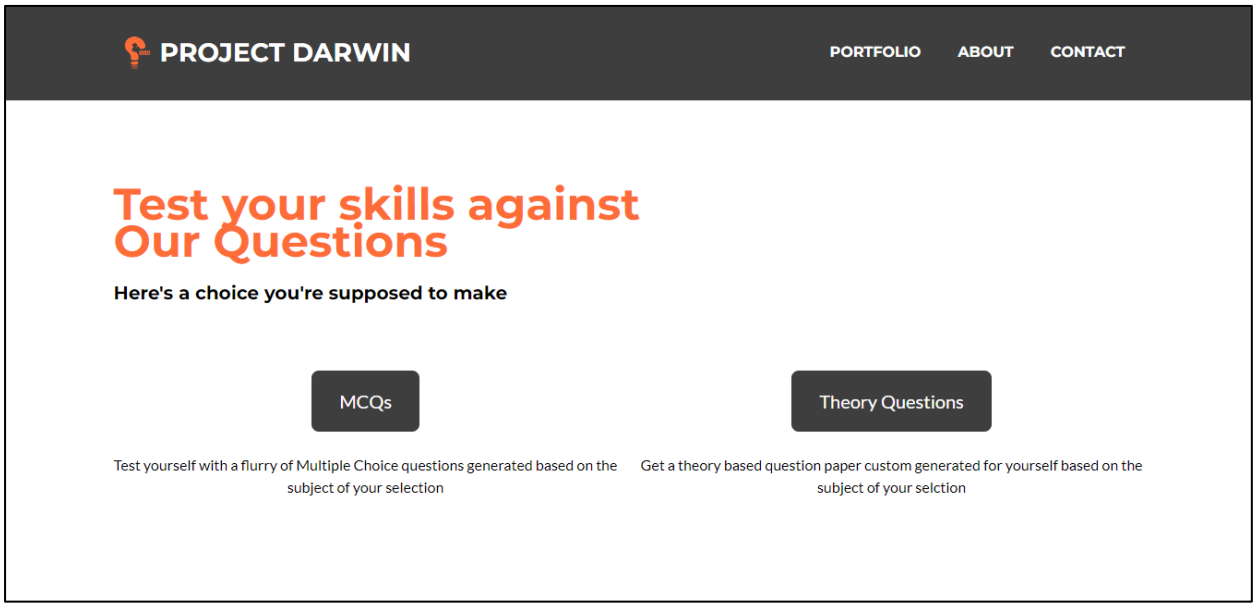


Fig. 6.9 – Quick Test Page

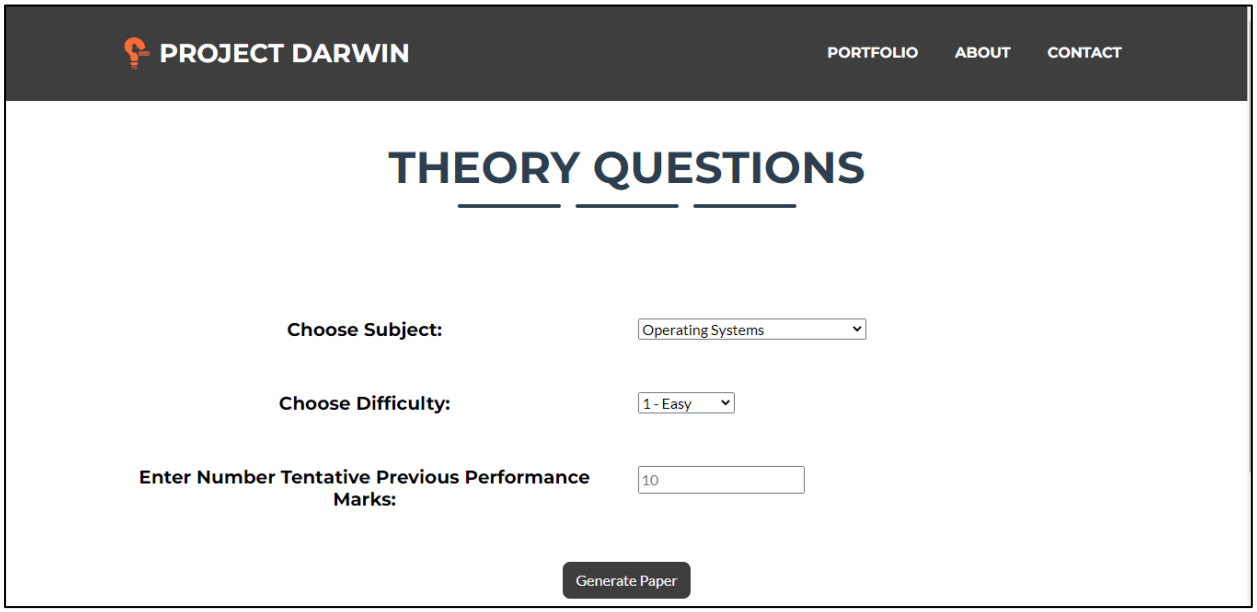


Fig. 6.10 – Theory Section

Question Paper: AI

(1) Attempt any 20 of the following:

(5 marks each)

Q1. What is the difference between agglomerative and divisive hierarchical clustering?

Q2. What are the differences between Supervised and Unsupervised Machine Learning?

Q3. What is Linear Regression in Machine Learning?

Q4. Mention some machine learning algorithms exposed by Mahout?

Q5. In a scatter diagram, what is the line that is drawn above or below the regression line called?

Q6. What is classification?

Q7. What is Clustering in Machine Learning?

Q8. What is the meaning of model capacity in Deep Learning?

Q9. Explain how Machine Learning is different from Deep Learning.

Q10. What is the difference between Search Head Pooling and Search Head Clustering?

Q11. What is a Decision Tree in Machine Learning?

Q12. How to calculate the accuracy of a binary classification algorithm using its confusion matrix?

Q13. Implement simple linear regression in Python on this 'Boston' dataset where the dependent variable is 'medv' and the independent variable is 'lstat.'

Q14. : What is Hierarchical Clustering?

Q15. Differentiate between supervised, unsupervised, and reinforcement learning.

Q16. Build a logistic regression model on the 'customer_churn' dataset in Python. The dependent variable is 'Churn' and the independent variable is 'MonthlyCharges.' Find the log_loss of the model.

Q17. What are the assumptions required for linear regression?

Q18. What is the meaning of bagging and boosting in Deep Learning?

Q19. What is logistic regression?

Q20. What do you understand about linear regression?

Q21. What are some of the Deep Learning frameworks or tools that you have used?

Q22. What is PCA in Machine Learning?

Fig. 6.11 – Generated Paper

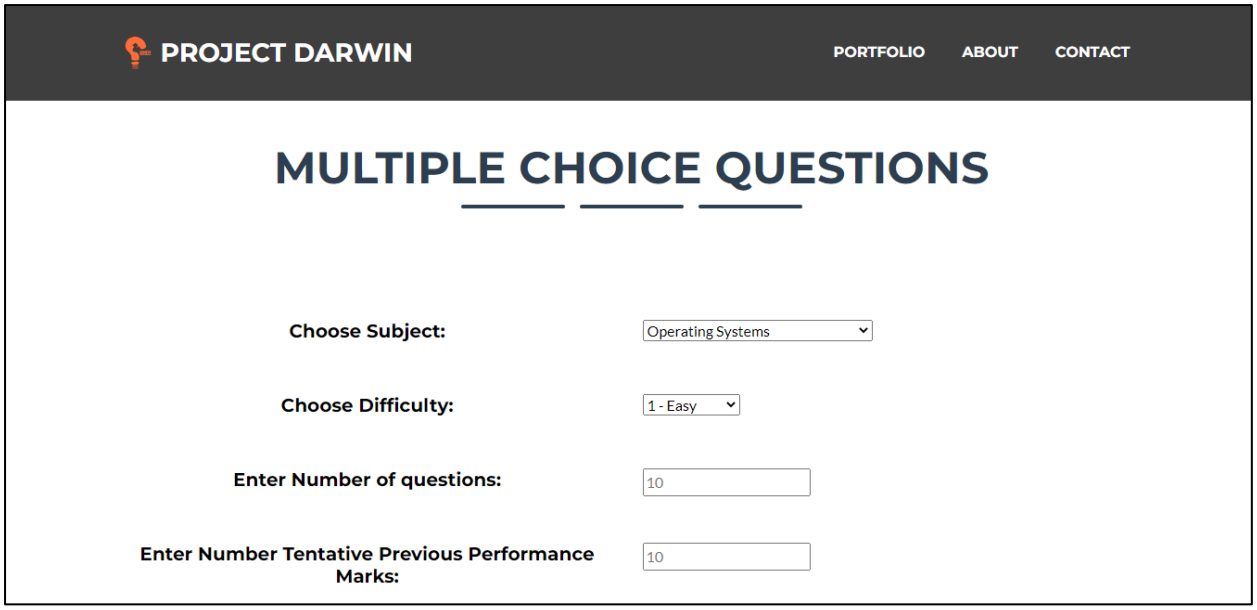


Fig. 6.12 – MCQ Section

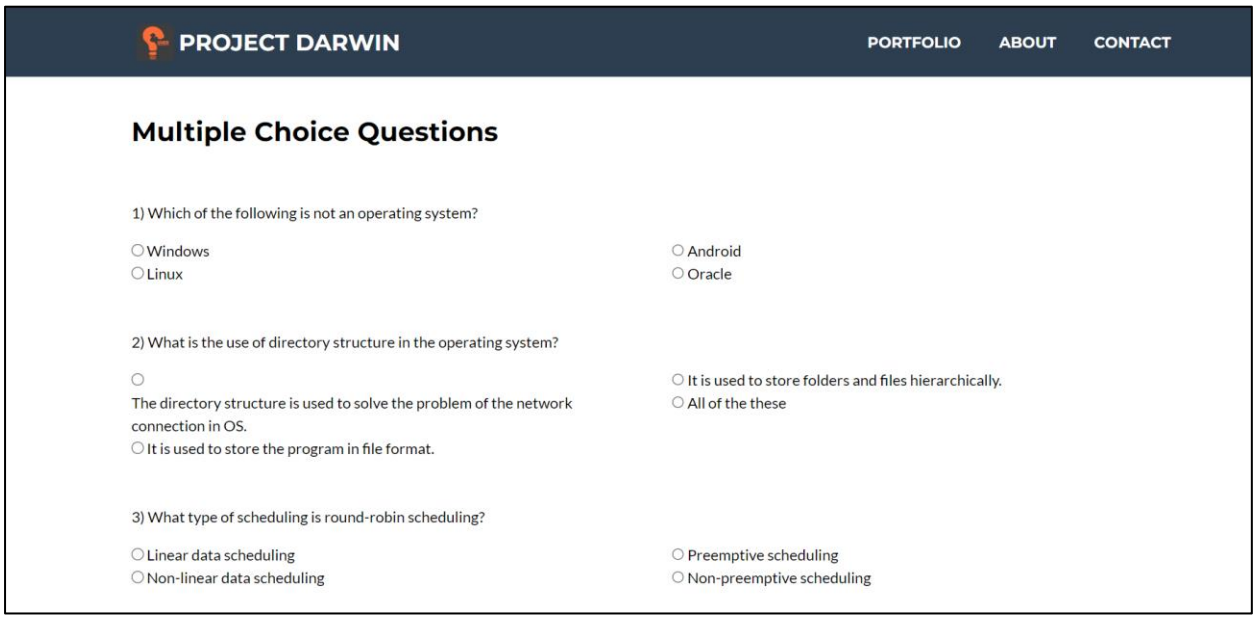


Fig 6.13 – MCQ Paper (Part 1)

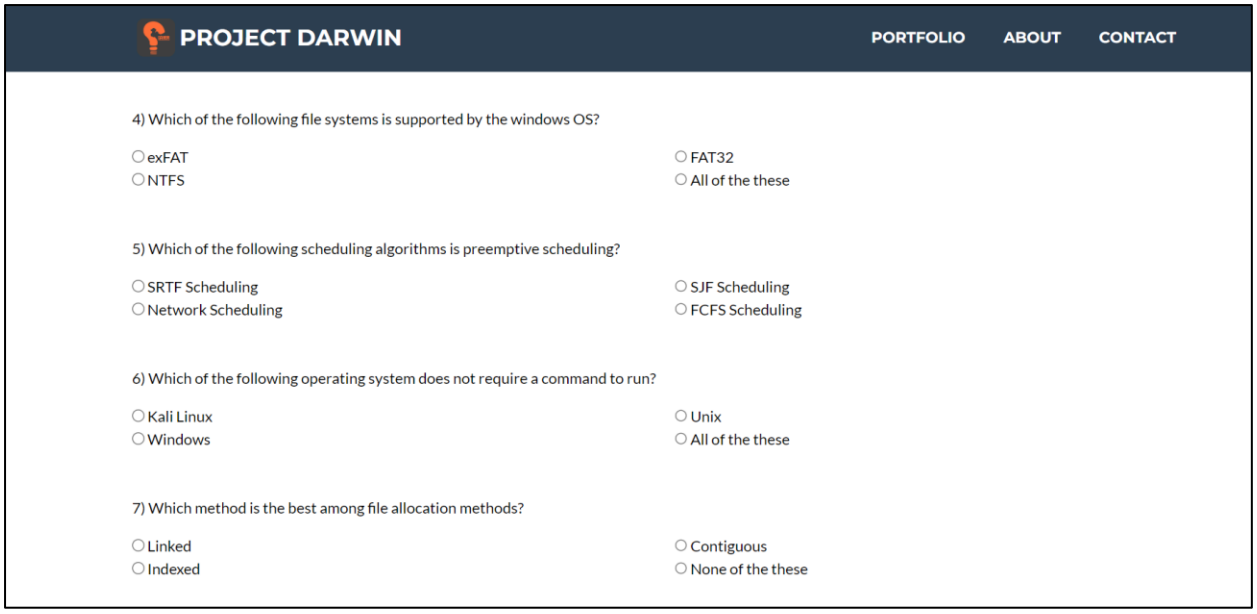


Fig. 6.14 – MCQ Paper (Part 2)

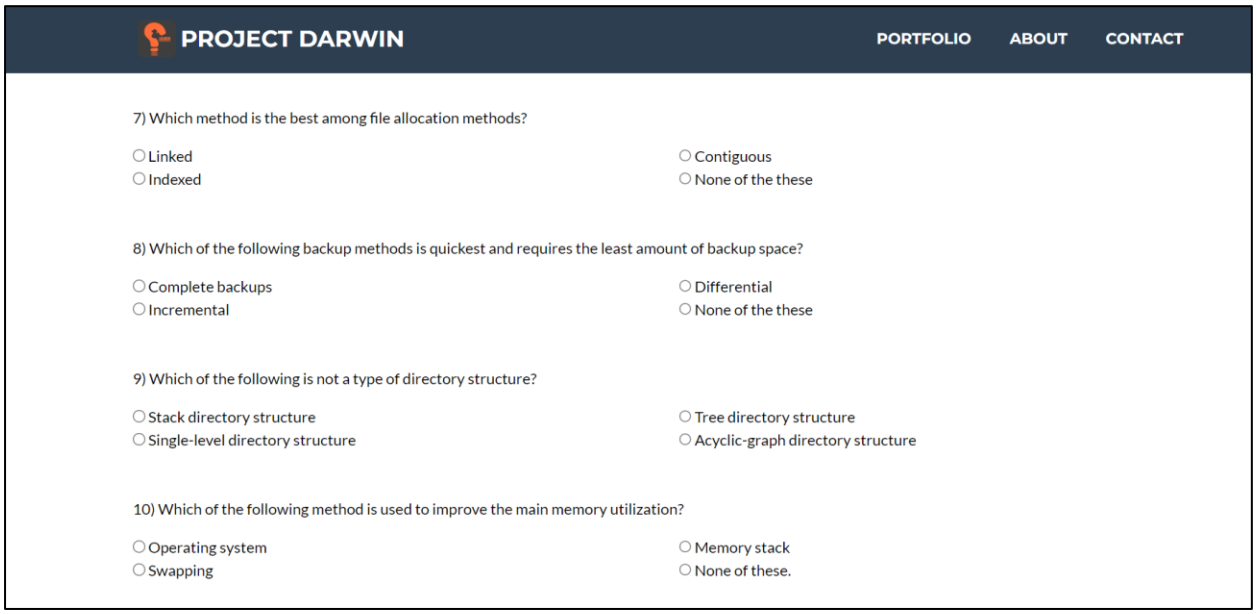


Fig. 6.15 - MCQ Paper (Part 3)

- **Custom Paper Generation Page**

The question paper is generated by getting requirements from the user. It keeps on changing the module of papers based on the previous marks of the user.

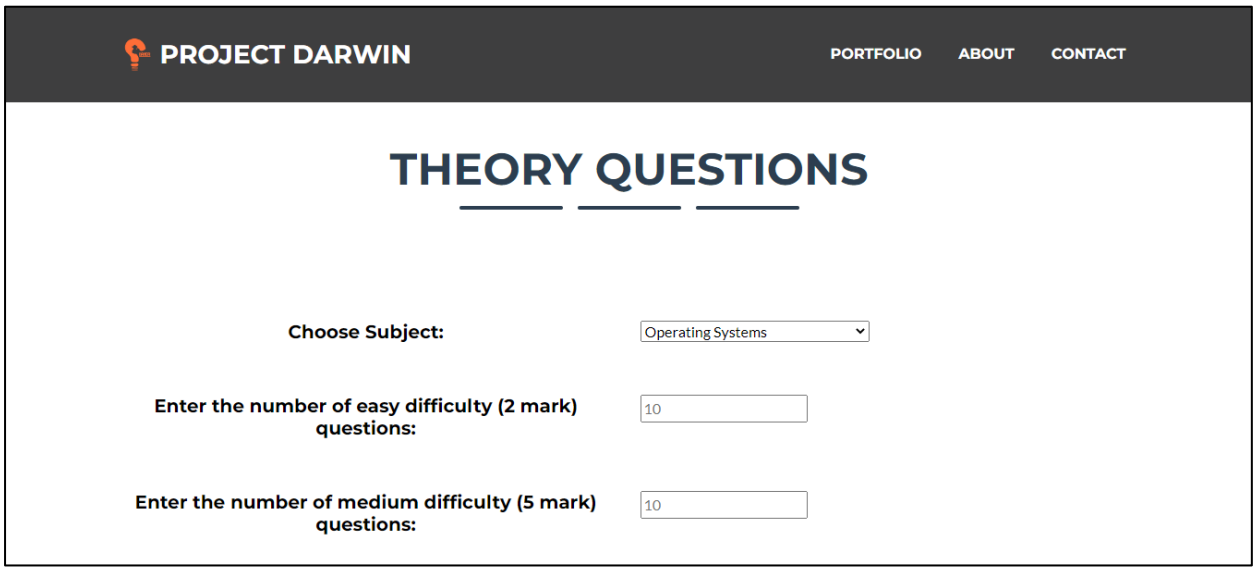


Fig. 6.16 – Custom Theory Section (Part 1)

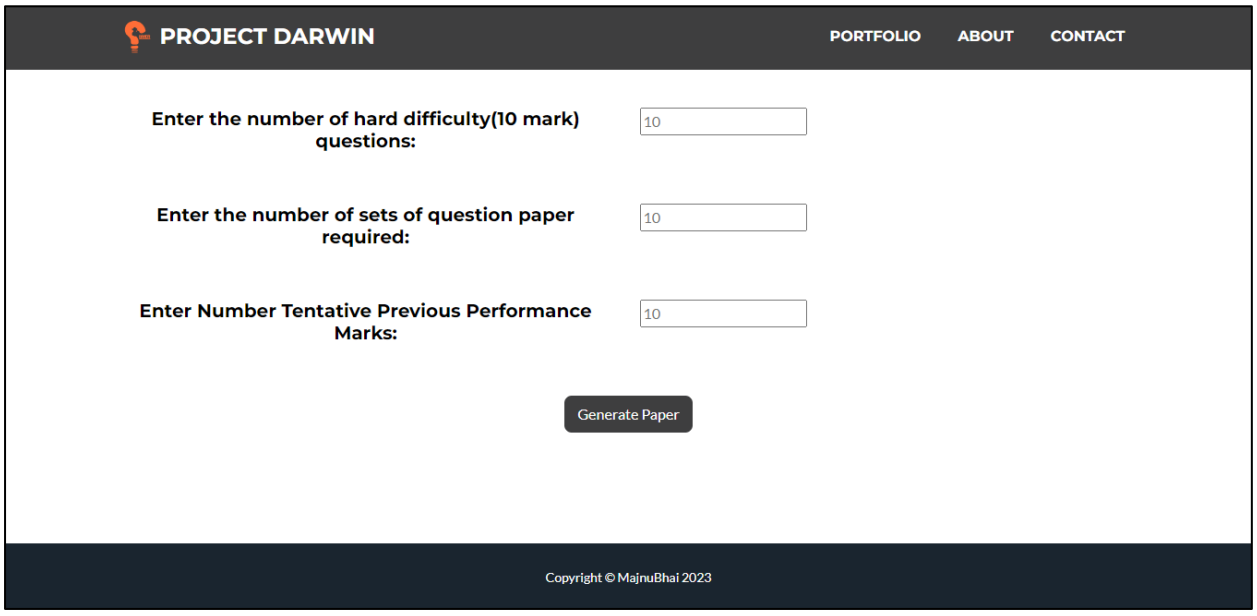


Fig. 6.17 – Custom Theory Section (Part 2)

Question Paper: AI

(1) Attempt any eight of the following:

(5 marks each)

Q1. Is KNN different from K-means Clustering?

Q2. What is PCA in Machine Learning?

Q3. What is a Decision Tree in Machine Learning?

Q4. How to calculate the accuracy of a binary classification algorithm using its confusion matrix?

Q5. : What are some of the properties of clustering algorithms?

Q6. Differentiate between Classification and Regression in Machine Learning

Q7. What is the difference between Search Head Pooling and Search Head Clustering?

Q8. What do you understand by a decision tree?

Q9. What are the assumptions required for linear regression?

(2) Attempt any four of the following:

(10 marks each)

Q1. When should Classification be used over Regression?

Q2. How is it different from doing machine learning in R or SAS?

Q3. Find Online Machine Learning course in Other Regions

Q4. What is the difference between inductive, deductive, and abductive Machine Learning?

Q5. Explain Dependency Parsing in NLP.

(3) Attempt any one of the following:

(20 marks each)

Q1. If you want to learn Natural Language Processing then go through the following tutorial:

Q2. What do you understand by logistic regression?

Fig. 6.18 – Generated Paper (Iteration-1)

Question Paper: AI

(1) Attempt any six of the following:

(5 marks each)

Q1. Implement simple linear regression in R on this 'mtcars' dataset, where the dependent variable is 'mpg' and the independent variable is 'disp.'

Q2. What is Linear Regression in Machine Learning?

Q3. What are some of the Deep Learning frameworks or tools that you have used?

Q4. Mention some machine learning algorithms exposed by Mahout?

Q5. What is F1 score in NLP?

Q6. What is the difference between agglomerative and divisive hierarchical clustering?

Q7. Is KNN different from K-means Clustering?

(2) Attempt any three of the following:

(10 marks each)

Q1. What are some of the examples of unsupervised learning algorithms in Deep Learning?

Q2. What is Bayes's Theorem in Machine Learning?

Q3. Why is Fourier transform used in Deep Learning?

Q4. What are unigrams, bigrams, trigrams, and n-grams in NLP?

(3) Attempt any two of the following:

(20 marks each)

Q1. What is Hypothesis in Machine Learning?

Q2. Explain Machine Learning, Artificial Intelligence, and Deep Learning

Q3. How is the transformer architecture better than RNNs in Deep Learning?

Fig. 6.19 - Generated Paper (Iteration-2)

(1) Attempt any four of the following:

(10 marks each)

Q1. What is Bayes's Theorem in Machine Learning?

Q2. What is Entropy in Machine Learning?

Q3. Name a few Machine Learning algorithms you know.

Q4. List the applications of Machine Learning.

Q5. List any two real-life applications of Natural Language Processing.

(2) Attempt any three of the following:

(20 marks each)

Q1. What is entropy in a decision tree algorithm?

Q2. What is Support Vector Machine (SVM) in Machine Learning?

Q3. What are some of the most used applications of Deep Learning?

Q4. What is an RNN in Deep Learning?

Fig. 6.20 - Generated Paper (Iteration-3)

Question Paper : AI

(1) Attempt any five of the following:

(20 marks each)

Q1. A regression analysis between apples (y) and oranges (x) resulted in the followi

Q2. What is Deep Learning?

Q3. How is the transformer architecture better than RNNs in Deep Learning?

Q4. Explain Machine Learning, Artificial Intelligence, and Deep Learning

Q5. If you want to learn Natural Language Processing then go through the following t

Q6. What is an RNN in Deep Learning?

Fig. 6.21 - Generated Paper (Iteration-4)

CHAPTER 8

CONCLUSION AND FUTURE SCOPE

8.1 Conclusion

The System allows user to have a customize generation of papers based on their requirements. The questions collected from the website are cleaned using NLP techniques and are used in quick question generation module. Setting parameters to the questions extracted is completely automated. The system follows the concept of adaptive evaluation which helps users to increase their ability to learn based on the question paper set. The system provides different sets of question papers, each with 10-20% repetitive questions which can be used by the teacher for generating Final examination papers. The algorithm used for paper generation avoids repetition of questions in every set. The format of the paper keeps on changing depending on the marks provided. The User can have a better experience about competitive examination by using the solution as practice tool.

8.2 Future Scope

- With rise in self operating AIs the scope for the system vastly increases. The major point of which is the addition of a custom AI which can help in the generation of questions.
- The system can be imporved further by adding the option to check the theory based answers provied by the user and scoring them as well.
- Another addtion based on the above mentioned points is the addition of a section which not only checks the theory-based answers given by the users for correctness but also provides feecback on the same.
- The effeciency of the system can be a scope of improvement as well. The current system is clunky and consumes more than ideal amount of time in its execution.

REFERENCES

REFERNECES

[1] G. Nalawade, and R. Ramesh. “Automatic generation of question paper from user entered specifications using a semantically tagged question repository,” 2016 IEEE Eighth International Conference on Technology for Education (T4E). IEEE, 2016.

[2] Zalte S.V., Jadhav C.C., Mangire A.A., Hole A.D., Tulshi A.R. “Automatic Question Paper Generator System” March 2018, International Journal of Advanced Research in Computer and Communication Engineering (IJARCCE) Vol. 7.

[3] Pratik Pisat, Shrimangal Rewagad, Devansh Modi, Ganesh Sawant, Prof. Deepshikha Chaturvedi “Question Paper Generator and Answer Verifier” International Conference on Energy, Communication, Data Analytics and Soft Computing (ICECDS-2017).

[4] Akhil Killawala, Igor Khokhlov, Leon Reznik “Computational Intelligence Framework for Automatic Quiz Question Generation” 978-1-5090-6020- 7/18/c 2018 IEEE International Conference on Fuzzy Systems (FUZZ).

[5] Mr. Amit Sanjay Khairnar, Mr. Bhagwat Chintaman Jadhav, Mr. Rahul Birhade, Mr. Pramod Patil “Automatic Question Paper Generator” International Journal for Technological Research In Engineering Volume 4, Issue 9, May-2017

[6] Ashraf Amria, Ahmed Ewais, Rami Hodrob “A Framework for Automatic Exam Generation based on Intended Learning Outcomes” CSEDU 2018.

PUBLISHED PAPER

Published by :
http://www.ijert.org

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 12 Issue 03, March-2023

Question Paper Maker using Natural Language Processing

Sanchit P. Jadhav
Information Technology
Vidyalankar Institute of Technology
Mumbai, India

Tirthesh J. Patil
Information Technology
Vidyalankar Institute of Technology
Mumbai, India

Nikhil Dhangada
Information Technology
Vidyalankar Institute of Technology
Mumbai, India

Prof. Debarati Ghosal
Information Technology
Vidyalankar Institute of Technology
Mumbai, India

Abstract **- The knowledge of a student is examined at every stage from the day he/she starts their journey to acquire it. Continuous examination is required for testing the student's learning ability in a particular domain. Traditional way of generating question papers can't fulfill such continuous examination process. The proposed system, Question Paper Maker can be used to overcome the drawback of traditional method and helps to generate a better-quality paper which will cover the domain requirements. Question Paper maker can generate theoretical papers based on adaptive evaluation. A student can also use Question paper maker as a practice tool for preparation of competitive exams.**

Keywords - QPM – Question Paper Maker, NLP – Natural Language Processing, AE – Adaptive Evaluation, ML – Machine Learning

I. INTRODUCTION

Generating Question paper manually is a time-consuming work. It's a difficult task to create a paper which can cover all the requirements required for testing the knowledge acquired by the students in a particular subject/field. Most of the time papers which are generated may have same questions as professor's priorities the question based on their importance which help students to find out which question may occur based on previous years questions paper. So, to avoid the prioritization of question and to provide a variety of question we proposed a system named as question paper maker which makes use of natural language processing for generating a series of question papers based on student's previous marks and advancing the quality of paper at each iteration.

The solution can be used by students for practicing as well as by the teacher for creating multiple sets of question papers. Teachers can use the solution for generating question papers based on the constraints which need to be followed while designing a question paper. It automates the task for providing a set of question papers in front of moderator, from which the moderator selects the paper which will be the final one for the university. As every subject is divided into modules, teachers can generate paper based on selecting the modules from which the question must be selected. The solution can also be useful for generating mid-semester examination by selecting module wise questions.

The solution makes use of NLP based models and concepts to grasp the context of the input given to the software and compile it together. It is here that the questions are generated for the examinee to have a crack at it. The system attempts to take textual data as input and extract the context from it for generating the questions. The system can also generate questions set by scrapping questions from a website, where the user must provide the URL of that particular web page. Thus, providing a way for the examiners to create questions based on the material they desire rather than a present subject. That's how the examination side of the software will shape up. The question set provided is cleaned by extracting the unwanted/incomplete questions. The questions are categorized based on the parameters like difficulty level, logical based, theoretical based, numerical, etc. Considering such parameters, classification of questions is done with the help of NLP models and a difficulty level is assigned to each question.

II. LITERATURE SURVEY

G. Nalawade and R. Ramesh [1] has proposed "Automatic generation of question paper from user entered specifications using a semantically tagged question repository". In this system the question generation is based on semantically tagged question repository. The person going to generate paper must manually enter specifications for generating question paper in the form, which is then converted into word document.

Zalte S.V., Jadhav C.C et al. [2] proposed "Automatic Question Paper Generator System" was implemented in C#. The author uses randomization technique for avoiding repetitive questions. The question paper is automatically generated by the traditional method and the questions were kept in the database. The difficulty and priority must be mentioned along with its question.

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 12 Issue 03, March-2023

Pratik Pisat, Shrimangal Rewagad et al. [3] "Question Paper Generator and Answer Verifier", The system generates subjective as well as objective question. The user must enter the subject name and difficulty, based on that paper is generated. For objective paper, score is continuously calculated and saved in the database.

Akhil Killawala, Igor Khokhlov, et al. [4] "Computational Intelligence Framework for Automatic Quiz Question Generation", the system generates quiz questions based on semantic correctness and follows a rule-based approach. It generates 4 kinds of questions like MCQ, True/False, Fill in the Blanks and "Wh" type questions.

Amit Khairnar, Bhagwat Jadhav, Pramod Patil, el at. [5] "Automatic Question Paper Generator" was built in Microsoft visual studio and uses Microsoft SQL server for storing questions. The user must enter the question at the time of paper generation and within a short time the question paper is generated. The system makes use of shuffling algorithms to avoid duplication and repetition while generating questions.

Ashraf Amria, Ahmed Ewais, Rami Hodrob el at. [6] "A Framework for Automatic Exam Generation based on Intended Learning Outcomes" has proposed system which can select questions based on intended learning outcomes. For generating paper, parameters such as variety of questions, randomization, marks distribution, and mapping questions with learning outcomes were considered. It is a web-based system, built using PHP and MySQL as Database for storing questions and related intended learning outcomes.

III. OBJECTIVE

The major objective of the project is to avoid human error while manually generating question sets for final exams. Every question paper generated must be unique and should be generated following the constraints. Another objective is to develop a system for students, through which they can theoretically attempt question papers by providing the previous marks for the subject based on which the difficulty of the papers keeps on changing.

IV. EXISTING SYSTEM

In the existing system, the questions were added to the database along with its difficulty manually assigned to it by the user. The generation of question papers was based on the level of difficulty and priority assigned by the generator to the questions. Randomization algorithms, NLP algorithms, shuffling algorithms and random function techniques were followed to avoid repetition of questions while generating papers. The aim of the system was to only generate a single question paper using randomization algorithm listed above. The papers which were generated do not cover the modules of the selected subject. The format used for paper generation was static or similar. Once the exam has been conducted, the outcome/result of the exam is not considered for the next iteration. The existing system uses various relational databases for storing questions, such as MySQL, Microsoft SQL Server, SQLite, etc.

V. SYSTEM ARCHITECTURE

The system will consist of two main modules. The first module will be the questions dataset cleaning and assigning parameters required for generating question paper. And the second is the question paper generation module.

A. Questions Dataset Module

In this module, the questions can be scrapped from a website or can be taken from the teacher/student for generation of the paper. A single dataset is maintained for all the questions related to the subject. The questions are scrapped and cleaned using NLP techniques. Cleaning includes removing duplication, null values, incomplete questions, etc. Questions are assigned with the difficulty level based on bloom's taxonomy. The questions dataset can be displayed along with its level of difficulty in the system once it is generated. The dataflow of the question set generation is shown in figure 1.

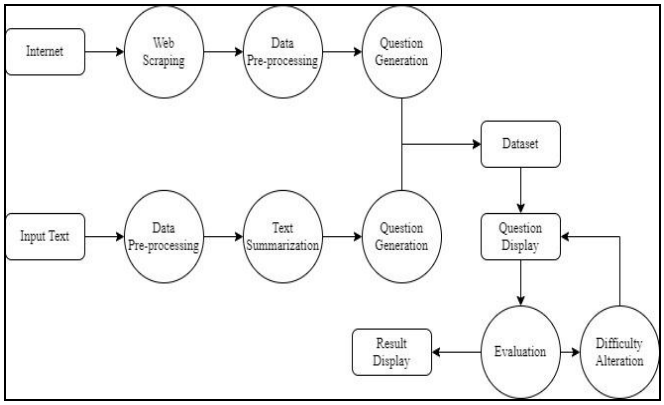


Figure 1. Dataset Generation Flow

B. Question Paper Generation Module

This module will generate one or multiple questions papers based on the parameter which are considered while generating a final exam paper. Generally multiple question sets are presented in front of the moderator from which the moderator selects one. This module will do the same task automatically. The system will generate question sets considering the constraints and will present these sets in front of the moderator. The format of the paper will change based on the total marks provided.

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 12 Issue 03, March-2023

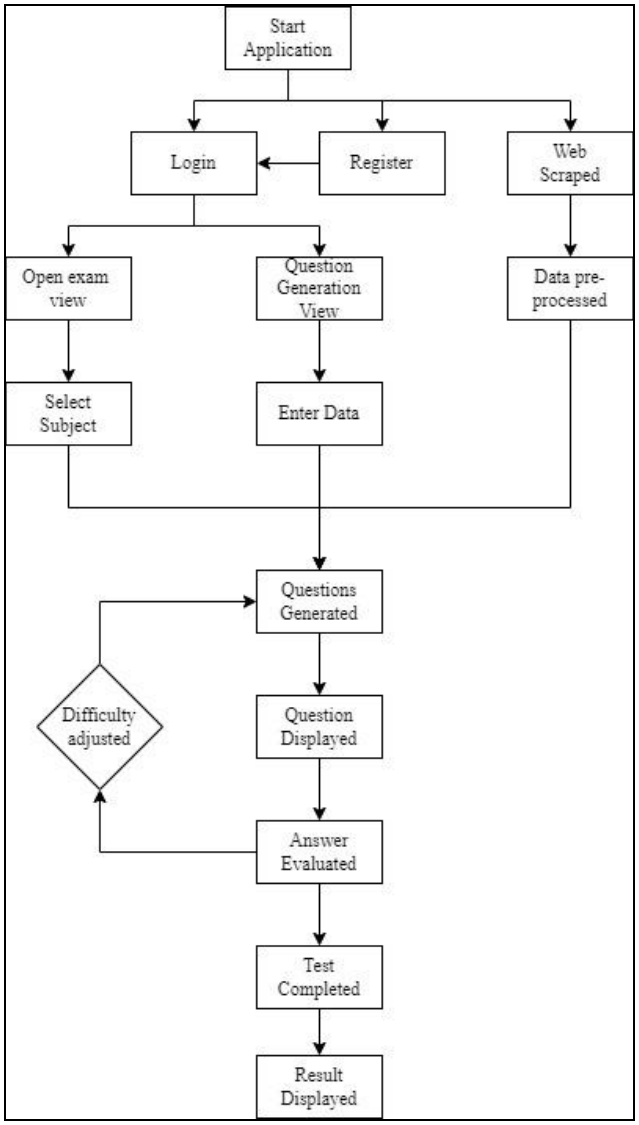


Figure 2. Paper Generation Flowchart

VI. SYSTEM USERS

The system will be used by three users' group, they are Student, Teacher, and Moderator.

1. Student

The students will have their separate database. They need to register and login into the system. Questions will be provided by them and are classified based on the subject he/she may enter. Multiple questions papers can also be generated based on the marks provided. The question dataset and question paper generated by the student is stored into the database.

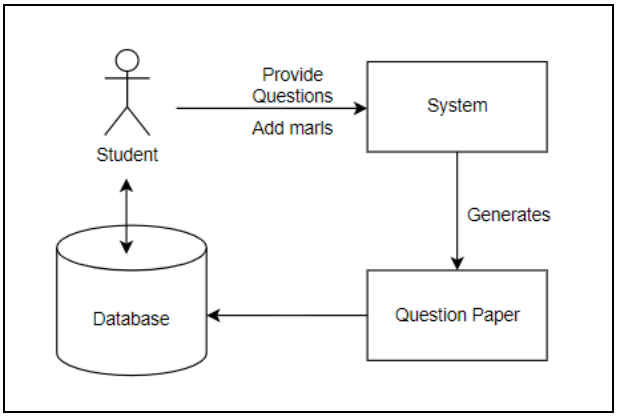


Figure 3. Student System Architecture

2. Teacher

The teachers can create several sets required for final selection of the question paper. For that they must provide the question bank which states questions related to each module of the subject. They question papers are stored in the database.

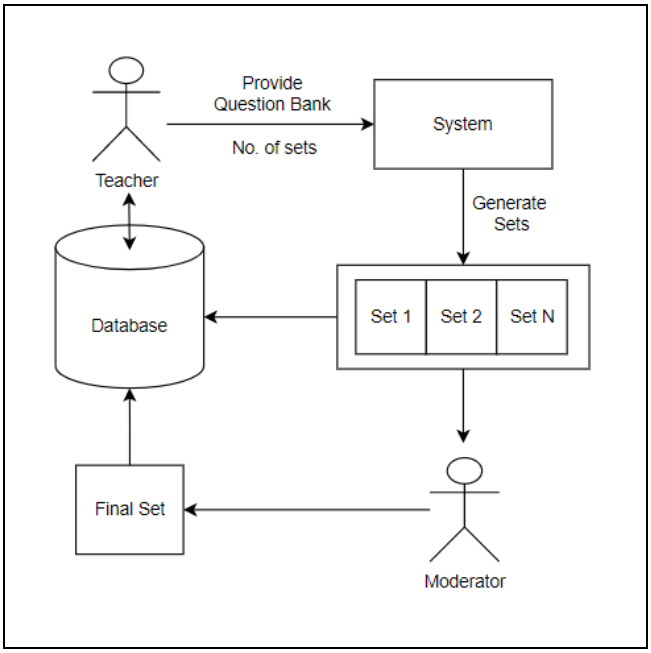


Figure 4. Teacher-Moderator System Architecture

3. Moderator

The Moderator acts as an admin in the system. The sets generated by the teachers are passed to the moderator where he/she has the right to select the question paper form that sets. Only Moderator has the right to modify the paper. No one can access the papers which are selected by the moderator. The paper which is finalized is stored securely into the database.

VI. CONCLUSION

The main purpose of this application is to generate question papers considering the constraints required for generating final exam papers. Questions in the paper generated cover all the modules which are present in the subject. The

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 12 Issue 03, March-2023

algorithm used for paper generation avoids repetition of questions in every set. The format of the paper keeps on changing depending on the marks provided.

REFERENCES

[1] G. Nalawade, and R. Ramesh. "Automatic generation of question paper from user entered specifications using a semantically tagged question repository," 2016 IEEE Eighth International Conference on Technology for Education (T4E). IEEE, 2016.

[2] Zalte S.V., Jadhav C.C., Mangire A.A., Hole A.D., Tulshi A.R. "Automatic Question Paper Generator System" March 2018, International Journal of Advanced Research in Computer and Communication Engineering (IJARCCE) Vol. 7.

[3] Pratik Pisat, Shrimangal Rewagad, Devansh Modi, Ganesh Sawant, Prof. Deepshikha Chaturvedi "Question Paper Generator and Answer Verifier" International Conference on Energy, Communication, Data Analytics and Soft Computing (ICECDS-2017).

[4] Akhil Killawala, Igor Khokhlov, Leon Reznik "Computational Intelligence Framework for Automatic Quiz Question Generation" 978-1-5090-6020- 7/18/c 2018 IEEE International Conference on Fuzzy Systems (FUZZ).

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

[6] Ashraf Amria, Ahmed Ewais, Rami Hodrob "A Framework for Automatic Exam Generation based on Intended Learning Outcomes" CSEDU 2018.

CERTIFICATES

International Journal of
Engineering Research & Technology
ISSN : 2278 - 0181, www.ijert.org
(Published by : ESRSA Publications)

CERTIFICATE
OF PUBLICATION

This is to certify that

Sanchit P. Jadhav

Has published a research paper entitled

Question Paper Maker using Natural Language Processing

In IJERT, Volume 12, Issue 3 , March - 2023

Registration No: IJERTV12IS030156 **Date:** 05-04-2023 **Chief Editor,IJERT**


International Journal of
Engineering Research & Technology
ISSN : 2278 - 0181, www.ijert.org
(Published by : ESRSA Publications)
CERTIFICATE
OF PUBLICATION
This is to certify that
Tirthesh J. Patil
Has published a research paper entitled
Question Paper Maker using Natural Language Processing
In IJERT, Volume 12, Issue 3 , March - 2023
Registration No: IJERTV12IS030156
Date: 05-04-2023
Chief Editor,IJERT


International Journal of
Engineering Research & Technology
ISSN : 2278 - 0181, www.ijert.org
(Published by : ESRSA Publications)
CERTIFICATE
OF PUBLICATION
This is to certify that
Nikhil Dhangada
Has published a research paper entitled
Question Paper Maker using Natural Language Processing
In IJERT, Volume 12, Issue 3 , March - 2023
Registration No: IJERTV12IS030156
Date: 05-04-2023
Chief Editor,IJERT

International Journal of
Engineering Research & Technology
ISSN : 2278 - 0181, www.ijert.org
(Published by : ESRSA Publications)

CERTIFICATE
OF PUBLICATION

This is to certify that

Debarati Ghosal

Has published a research paper entitled

Question Paper Maker using Natural Language Processing

In IJERT, Volume 12, Issue 3 , March - 2023

Registration No: IJERTV12IS030156 **Date:** 05-04-2023 **Chief Editor,IJERT**

DrillBit

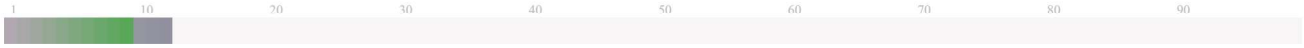
The Report is Generated by DrillBit Plagiarism Detection Software

Submission Information

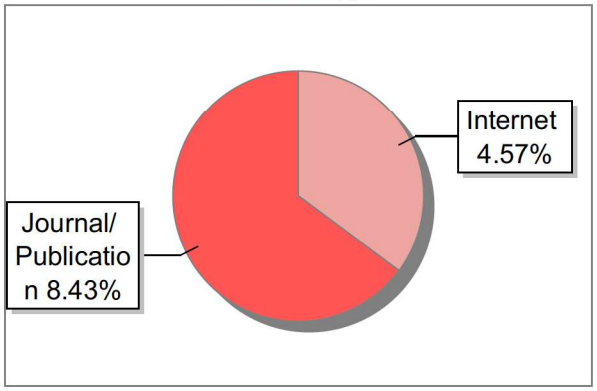
Author Name	Nikhil Dhnagada, Tirthesh Patil, Sanchit..
Title	QUESTION PAPER MAKER
Paper/Submission ID	728039
Submission Date	2023-04-27 22:39:41
Total Pages	59
Document type	Thesis

Result Information

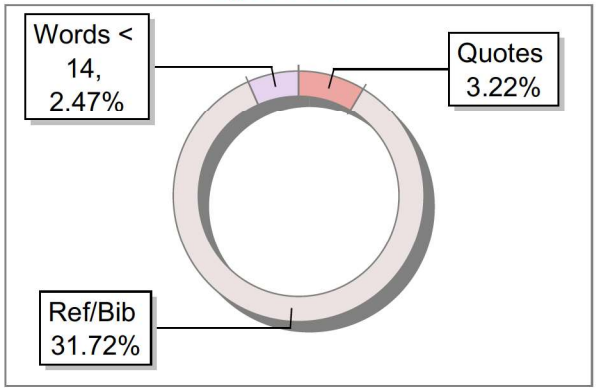
Similarity **13 %**



Sources Type



Report Content



Exclude Information

Quotes	Excluded
References/Bibliography	Excluded
Sources: Less than 14 Words Similarity	Not Excluded
Excluded Source	**0 %**
Excluded Phrases	Not Excluded

A Unique QR Code use to View/Download/Share Pdf File