**Question Answer Generator Using NLP**

SUBMITTED IN PARTIAL FULFILMENT OF THE REQUIREMENTS FOR THE AWARD OF DEGREE OF

**BACHELOR OF TECHNOLOGY IN**

**INFORMATION TECHNOLOGY**



**Submitted By:**

**Name: Ayush Kumar**

**University Roll No.: 1803821**

**Under the supervision of**

**Guide Name: Mrs. Reeta Bhardwaj**

**Designation: Associate Professor**

**DEPARTMENT OF INFORMATION TECHNOLOGY**

**DAV INSTITUTE OF ENGINEERING & TECHNOLOGY**

**Jalandhar – 144008**

**November 2019**

**ABSTRACT**

Learning through the internet becomes popular that facilitates learners to learn anything, anytime, anywhere from the web resources. Assessment is most important in any learning system. An assessment system can find the self-learning gaps of learners and improve the progress of learning. The manual question generation takes much time and labour. Therefore, automatic question generation from learning resources is the primary task of an automated assessment system. This paper presents a survey of automatic question generation and assessment strategies from textual and pictorial learning resources. The purpose of this survey is to summarize the state-of-the-art techniques for generating questions and evaluating their answers automatically.

**ACKNOWLEDGEMENT**

Achievement of this task would have been impossible without the help and constant guidance of certain people. We hereby take the opportunity of thanking a few people for their support, guidance and encouragement for the successful completion of our project.

I would like to thank Dr. Dinesh Kumar, Hon’ble HOD(IT), DAVIET. It’s due to his efforts that we are able to experience a platform like DAVIET for our undergraduate degree.

I would like to express our sincere thanks to Mrs. Reeta Bhardwaj, who provided us with valuable guidance and support.

I would like to thank our group members Mr. Akhil Kathuria, Mr. Gautam Taneja and Mr. Adarsh Kumar. for making such a valuable effort and making good team spirit among all the members.

Each of us would also like to thank our respective parents who have been a constant motivation for us during the course of this project.

**TABLE OF CONTENTS**

|  |  |
| --- | --- |
| **Contents** | **Page No.** |
|  |  |
| ***Abstract*** | ***i*** |
| ***Acknowledgement*** | ***ii*** |
| ***List of Figures*** | ***iii*** |
| ***List of Tables*** | ***iv*** |
| ***Table of Contents*** | ***v*** |
| **Chapter 1: Introduction** | **1** |
| **Chapter 2: Requirement Analysis** |  |
| **Chapter 3: System Design** |  |
| **Chapter 4: Development, Implementation and Testing** |  |
| **Chapter 5: Results and Discussions** |  |
| **Chapter 6: Conclusion and Future Scope** |  |
| ***References*** |  |
| ***Appendix A: …..*** |  |

**List of Figures**

**Fig. No.** **Figure Title** **Page No.**

**List of Tables**

**Table No.** **Table Caption** **Page No.**

### [Chapter -1] Introduction

### 1.1 Project Overview

### An examination is an effective method for testing one's understanding of any topic. Answering questions helps to improve the process of learning and is an integral part of academics. This project falls under internet-based industry automation. This project aims to build an automated examination system using machine learning, natural language processing (NLP), python environment, flask framework, and web technologies to provide an inexpensive alternative to the current examination system. Automatic question generation for textual inputs is valuable in academics where answering questions helps students to learn and improve their understanding of their field of study. Automatic question generation finds application in dialog systems or virtual assistants where asking questions is an important part of interactions between humans and machines. Apart from the education domain, Quiz Question Generation for the entertainment industry is also gaining importance. There are millions of people playing quiz games as a fun activity and at times serious money-making options. This requires the automated generation of millions of questions each day manually. The Automated Question Generation system aims at automating this process of question generation by providing an end-to-end system that will take the data which may be in the form of comprehension (sentence, chapters) as input and provide a question as output. Conducting examination and answer sheet evaluation are hectic testing tools for assessing academic achievement, integration of ideas, and ability to recall, but are expensive, resourceful, and time-consuming to generate questions and evaluate responses manually. Manual evaluating of answer sheets take up a significant amount of instructors' valuable time and hence is an expensive process. We implement a model to automatically generate questions with their respective answers and assess user responses.

### 1.2 Objectives

The main objective is that this tool will be a great help to the People using AI using NLP to   manage the Test-taking task conveniently.  The manual question generation takes much time and labour. Therefore, automatic question generation from learning resources is the primary objective of this automated assessment system.

1.     Reducing the effort of making questions from a given topic.

2.     Maintaining the integrity of exams by giving random questions to every user.

3.     Conducting secured online tests.

### 4.     Fast evaluation of results.

### 1.3 Problem Formulation

### AI-based examination systems add a step towards creating an effortless conducting of examinations. The working of the AI examination system ensures the workload management of professors and teachers of creating and marking the question paper manually. It also helps in better evaluation of students based on their knowledge.

### It generates random questions for every student which makes the system even more precise and all the users will get different question sets so that the chances of cheating will be minimal. On another hand, it will help to save paper and transportation costs of papers to examination halls. It will help in resource management as this will cut corners on stationery products, it will also have greater efficiency with respect to time as it will produce instantaneous results.

### A few problems that are faced in the current system are:

### Teachers have to set question papers manually.

### We are not able to generate different sets of question papers.

### Manual checking takes a lot of time and has chances of human error too.

### Student Record maintenance is manually done.

### Chances of cheating.

### Paper Wastage.

### 1.4 Existing System

### Offline System:

### The offline examination system relies on manual work from printing to transporting the paper to the examination hall, then invigilation and the most tedious task of checking the answer sheets which is a huge mess for any examiner which sometimes leads to resource loss. Also, we hear news about paper leaks and answer sheets being lost in the transporting process.

### Online System:

### The online examination system does not feature descriptive questions. They are good at evaluating the answers but they have little to no scope for the descriptive ones and the analysis is not well implemented to get meaningful results. Even the most famous of them just have a simple system of storing the correct options in the database and just matching the correct option with it to calculate the result.

### 1.5 Proposed System

### In the proposed model we are taking the online examination system to a new level by enabling the examinee to write descriptive answers which will get evaluated on their own i.e. automating the entire offline examination system with the efficiency of computing having no human error involved, this can be done using NLP or Natural Language Processing. The evaluated answers will be stored in the database and they can be viewed anytime and a particular student profile will be maintained for better evaluation of the student.

### This will be a huge boost to the online examination system as this will allow it to overcome its biggest con and it will also help the online examination system to stretch its paw even in the half-yearly or annual examination conducted by schools or colleges for evaluating the profile of the student. This will have instant benefits like the system will relieve the burden of the teachers and professors of checking copies and in return, they can be more productive with their time in teaching things, this will also eliminate biasing in answer script checking and will have leased space for any human error as copies would not be scanned and the entire marks will be allotted according to the way answers are written by the examinee while he was on and there will be little to no space for acquisition, it will help in resource management as this will cut corners on stationery products, it will also have greater efficiency with respect to time as it will produce instantaneous results and will be more secure and reliable.

### 1.5 Features of the Project

### Secure login portal: The portal will have two logins, one for the admin and one for the students.

### Generating questions and answers: Random questions and answers will be generated automatically for every user using AI.

### Different question sets: All the users will get different question sets so that the chances of cheating will be minimal.

### Evaluation: Results of student performance will be available just after the test submission.

### Maintaining record: When all the users have submitted their test, a CSV file containing records of users will be generated.

### [Chapter -2] Requirement Analysis

### 2.1 Feasibility Study

The feasibility study includes consideration of all the possible ways to provide a solution to the given problem. The proposed solution should satisfy all the user requirements and should be flexible enough so that future changes can be easily done based on the future upcoming requirements.

I. Economical Feasibility:

This is a very important aspect to be considered while developing a project. We decided on the technology based on the minimum possible cost factor.

* AI hardware and software cost has to be borne by our team.
* Overall we have estimated that the benefits the organization is going to receive from the proposed system will surely overcome the initial costs and the later on running cost for the system.

II. Technical Feasibility:

This included the study of function, performance, and constraints that may affect the ability to achieve an acceptable system. For this feasibility study. We studied complete functionality to be provided in the system, as described in the System Requirement Specification (SRS). and checked if everything was possible using a different type of frontend and backend platforms.

III. Operational Feasibility:

No doubt the proposed system is fully GUI based that is very user friendly and all inputs to be taken all self-explanatory even to a layman.

2.2 Software Required Specification Document

A software requirements specification (SRS) is a description of a software system to be developed. It lays out functional and non-functional requirements and may include a set of use cases that describe user interactions that the software must provide.

Software requirements specification establishes the basis for an agreement between customers and contractors or suppliers (in market-driven projects, these roles may be played by the marketing and development divisions) on what the software product is to do as well as what it is not expected to do. Software requirements specification permits a rigorous assessment of requirements before design can begin and reduces later redesign. It should also provide a realistic basis for estimating product costs, risks, and schedules. Used appropriately, software requirements specifications can help prevent software project failure.

Functional Requirements:

1. This project requires dataset for generating questions and answers.
2. Based on this dataset the software will generate questions using NLP (Using Python and its libraries).
3. Its User Interface design presents a seamless blend of visual design, interaction design, and information architecture.
4. Automatic evaluation of result also reduces effort of teacher to manually evaluate them.
5. Stores result of all students in database for future use.

Non Functional Requirements:

1. User Satisfaction: - The system is such that it stands up to the user expectations.
2. Response Time: - The response of all the operation is good. This has been made possible by careful programming.
3. Error Handling: - Response to user errors and undesired situations has been taken care of to ensure that the system operates without halting.
4. Safety and Robustness: - The system is able to avoid or tackle disastrous action. In other words, it should be foul proof. The system safeguards against undesired events, without human intervention.
5. User friendliness: - The system is easy to learn and understand. A native user can also use the system effectively, without any difficulties.

Hardware Requirements:

* Processor: Intel Quad-core 1.7 GHZ Processor or above.
* HD: Minimum 10 GB of HD.
* RAM: Minimum 8 GB of RAM.

Software Requirements:

* Languages : HTML, CSS, Bootstrap, JavaScript
* Development Environment : VScode
* Backend : Flask, Python
* Database: SQLite
* Other frameworks : JQuery, Numpy, Pandas, JSON, NLTK

2.3 SDLC Model used

The model that has been followed by our team is Spiral Model.

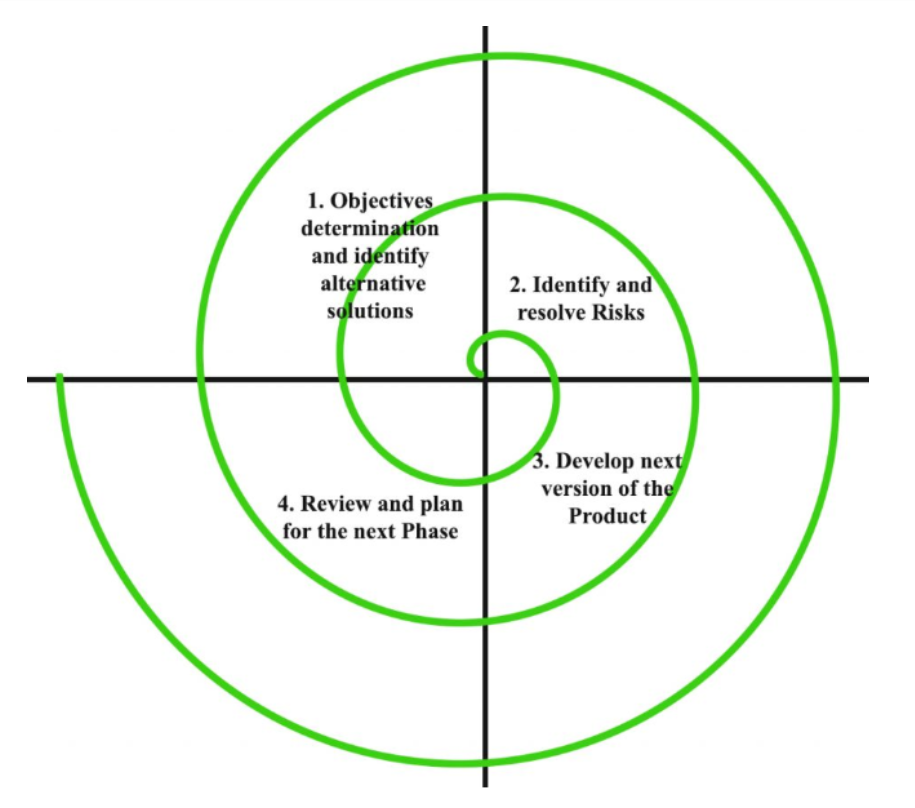


Fig. No. 1

**[Chapter -3] System Design**

3.1 Product perspective:

In the fast-paced society, people prefer to use practical tools to improve work efficiency. As for online exam, convenient operation methods can reduce the workload of teachers and also make it easier for students to study. The convenience of online exams is that teachers can use Question Answer Generation System which is a powerful online exam system to upload test questions in batches and quickly create test papers. It only takes four steps, and the teacher can successfully create a test paper in a few minutes.

3.2 Product functions:-

* Dataset is required for generating question and answers using NLP (Natural Language Processing).
* Different login portals available for teachers and students.
* Objective as well as Subjective questions will be generated and the result will be evaluated using NLP.
* Result is evaluated and a file containing result of all the students is sent to teacher’s portal.

3.3 User characteristics:-

There are two types of users that are going to interact with system, i.e. students and teachers. User interface of the system is easily understandable by every age group. The end users might use the user interface model to get their desired outputs and the backend engineers (developers) that control, fix bugs and change the code or technique of the software program as per the requirement.

3.4 Constraints:-

Few constraints to run the project:-

* The project should have all the libraries installed that are required for Natural Language Processing.
* The project needs to install third-party products.
* The source data provided should be efficient to generate required number of questions.

3.5 Flow chart/DFDS

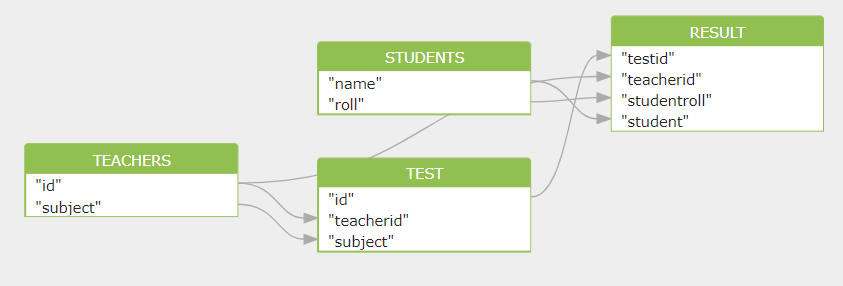


Fig. No.2

3.6 Database Design

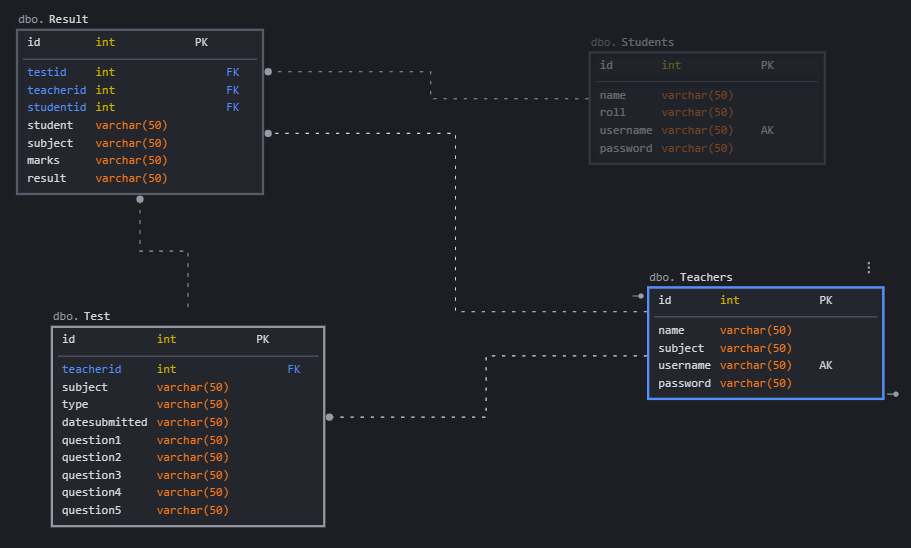


Fig. No. 3

3.7 Table Structure

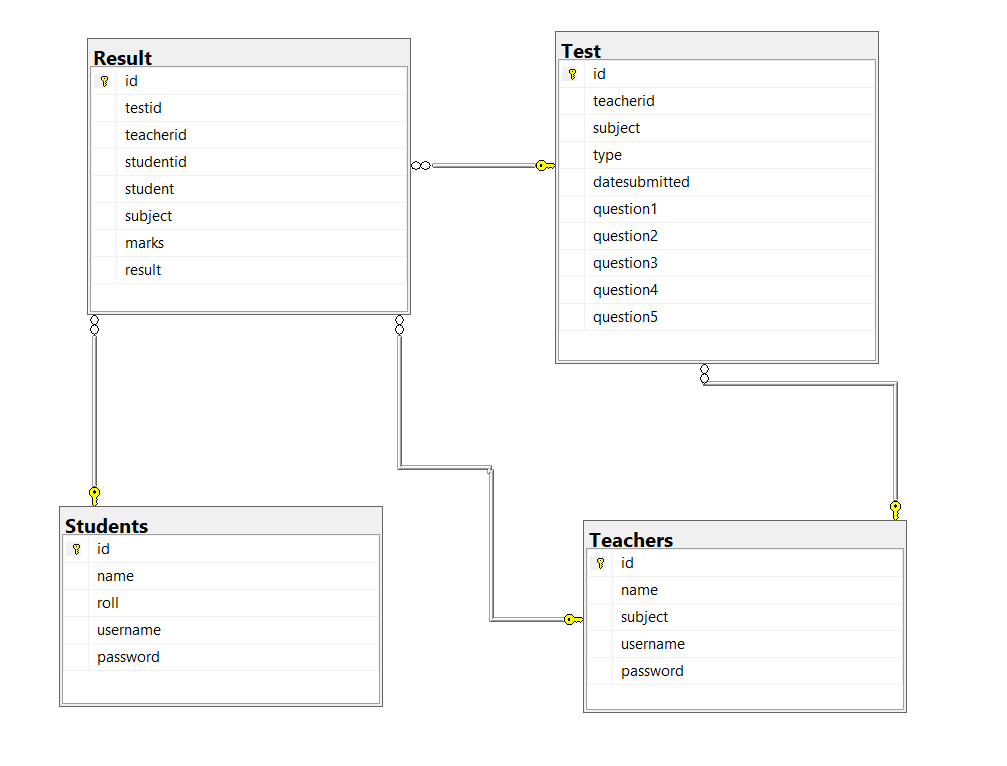


Fig. No. 4

3.8 ER Diagram

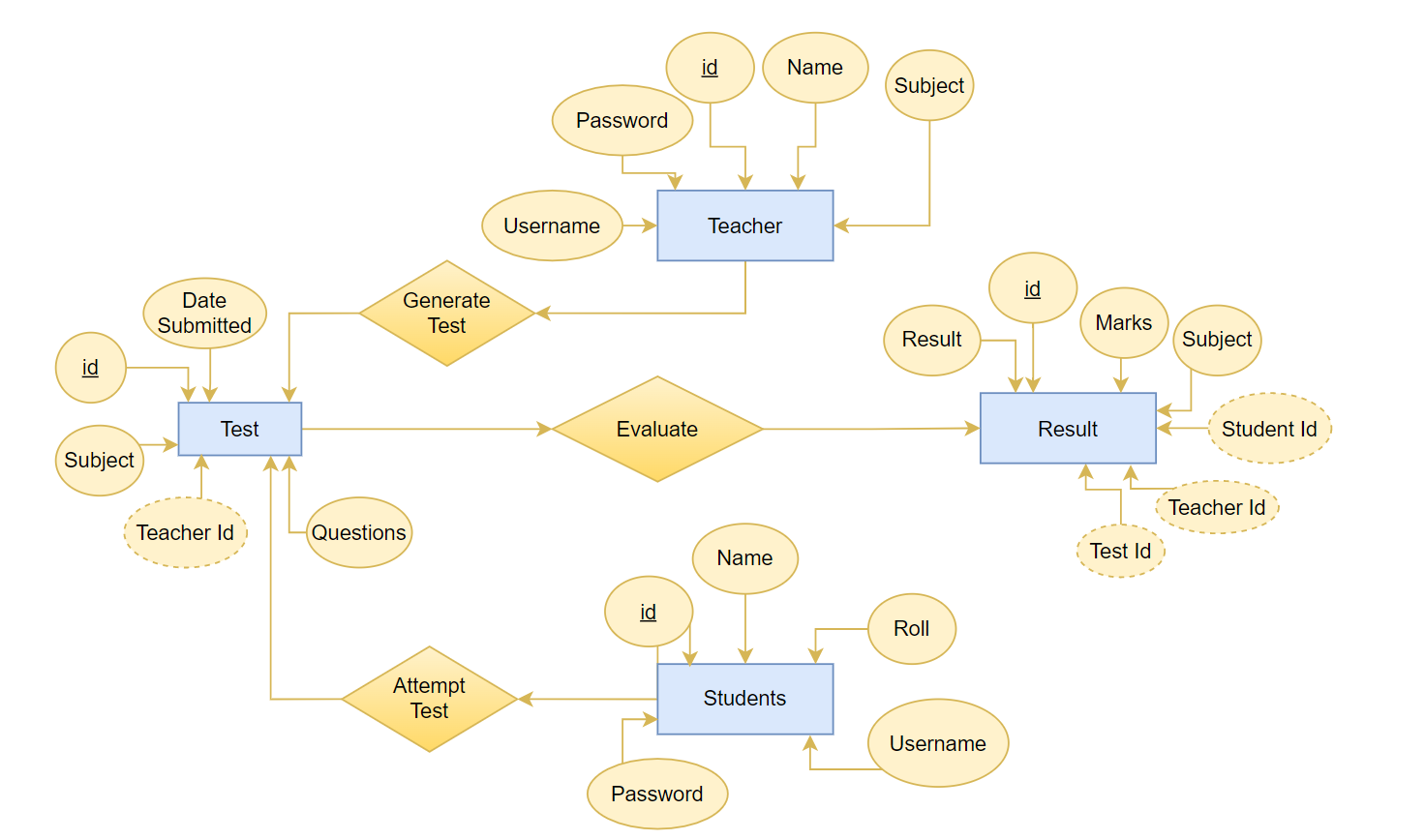


Fig. No. 5

3.9 Assumptions and Dependencies:-

Few assumptions and dependencies that is essential to use the product:

* + User should be aware about the operational functioning of the system.
  + The Source data provided by the user for the generation of the questions-answer must be in meaningful and usable form.
  + User should have enough storage to store the results of students.

3.10 Specific Requirements:

* The project will work only when Python and SQLite is installed on your PC.
* Some libraries use some older version of Python.
* PC with good GPU must be used for faster execution of code.

**[Chapter-4] Development, Implementation and Testing**

4.1 Introduction to development environment (Front End and Back End)

1. Front End

* HTML:-

The HyperText Markup Language, or HTML is the standard markup language for documents designed to be displayed in a web browser. It can be assisted by technologies such as Cascading Style Sheets and scripting languages such as JavaScript.

* CSS :-

Cascading Style Sheets is a style sheet language used for describing the presentation of a document written in a markup language such as HTML.

* Bootstrap :-

Bootstrap is a free and open-source CSS framework directed at responsive, mobile-first front-end web development.

* JavaScript :-

It’s an object-oriented computer programming language commonly used to create interactive effects within web browsers.

* jQuery:-

jQuery is an open-sourced JavaScript library that simplifies creation and navigation of web applications

1. Back End

* Flask :-

It is a micro web framework written in Python. It is classified as a microframework because it does not require particular tools or libraries

* Python :-

It is an interpreted high-level general-purpose programming language.

* SQLite :-

It is a relational database management system contained in a C library.

4.2 Other supporting languages or tools

1. NumPy:-

NumPy is a library for the Python programming language, adding support for large, multi-dimensional arrays and matrices, along with a large collection of high-level mathematical functions to operate on these arrays.

1. Pandas :-

Pandas is a software library written for the Python programming language for data manipulation and analysis.

1. Scikit – Learn :-

Scikit-learn is a free software machine learning library for the Python programming language. It features various classification, regression and clustering algorithms including support vector machines

4.3 Implementation of problem

4.3.1 Pseudo code of Tokenization

Define : List of unwanted\_character

Define : String\_article

For i=1 to number\_of\_character in String Article

If (String\_article[i]==“unwanted character”)

Remove String\_article[i];

End If

End For

String\_split (String\_article)

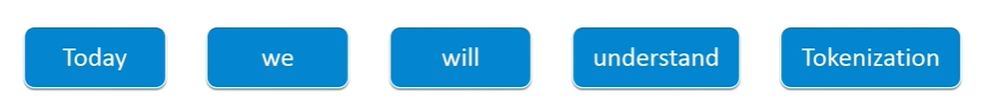


Fig. No. 6

4.3.2 Pseudo code of Keyword extraction

sort sentences by weignt

while (desiredSumLength is not met and there are unused sentences)

for (all sentences x )

if (sentence x not already in summary)

if (segment of sentece x has the lowest or equally low use)

set sum\_sentence = x

break out of for loop

end if

end if

end for

add sum\_sentence to the summary

record sum\_sentence as having been used

increment sum\_sentence's segment use

increment currentSumLength

end while

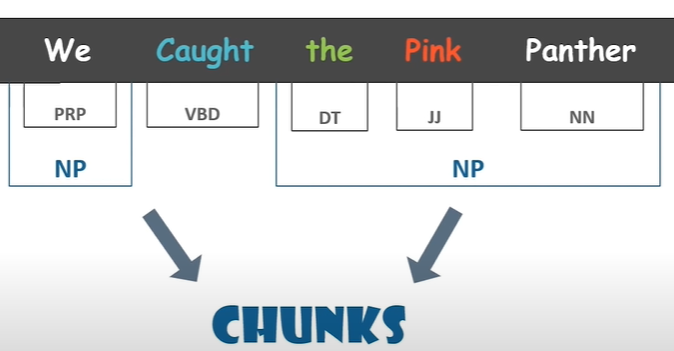


Fig. No. 7

4.3.3 Pseudo Code of Part of speech tagging

Algorithm Noun<n>

for each word in the list n >= 1do

if the word=noun then mark it up with <n>

else mark it up with <a>

return <n>

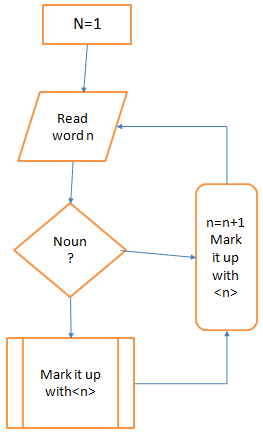


Fig. No. 8

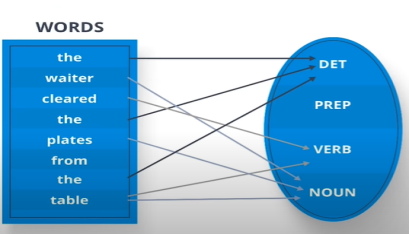


Fig. No. 9

**[Chapter - 5] Results and Discussions**

5.1 User interface representation

|  |  |
| --- | --- |
| **User Interface Module** | Is the module user directly interacts with. |
| **NLP Module** | This module is acting as backbone for project. Keyword extraction is done using various processes like tokenization, morphology, part of speech tagging, distributional similarity , chunking , name entity recognition. |
| **Database Module** | All students’ records will be saved in database. |

1. User login

* Teacher Login:-

Teacher can generate the test by providing some data. Test will be available to every student and after the student completes the test, teacher can either see the results on the website or can download the CSV file.

* Student Login:-

Each student will receive subjective or objective test with random questions and after they submit the test, result is evaluated and sent to teacher’s portal.

1. Online generated test questions

* Objective type questions:-

Five objective questions will be generated.

* Subjective type question:-

Two objective questions will be generated.

1. Result declaration

* CSV file format:-

Teacher can download the result of all the students.

* Online Portal:-

Teacher can view the results online.

5.2 Parameters used for evaluation

5.3 Project screenshots

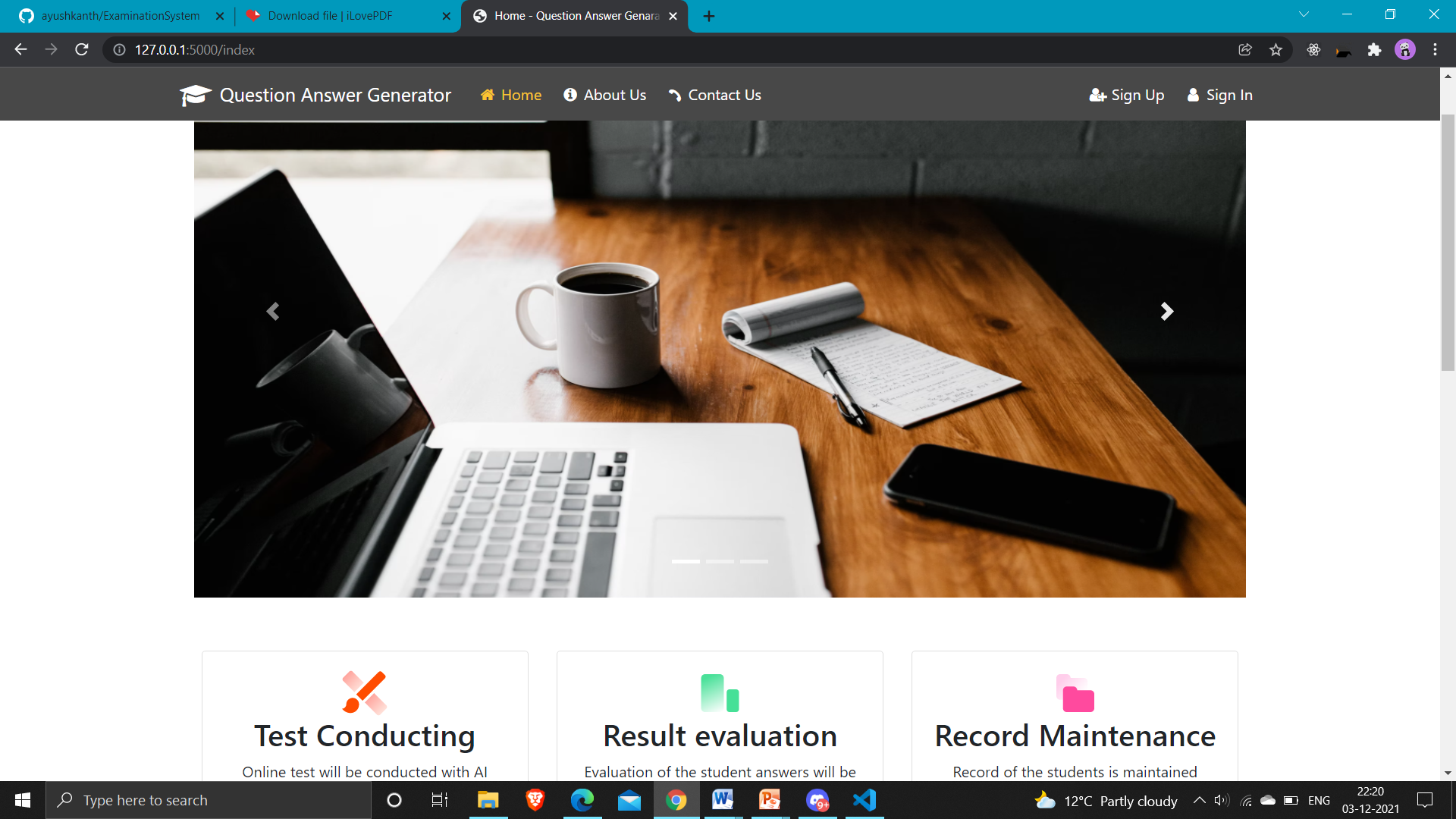


Fig. No. 10

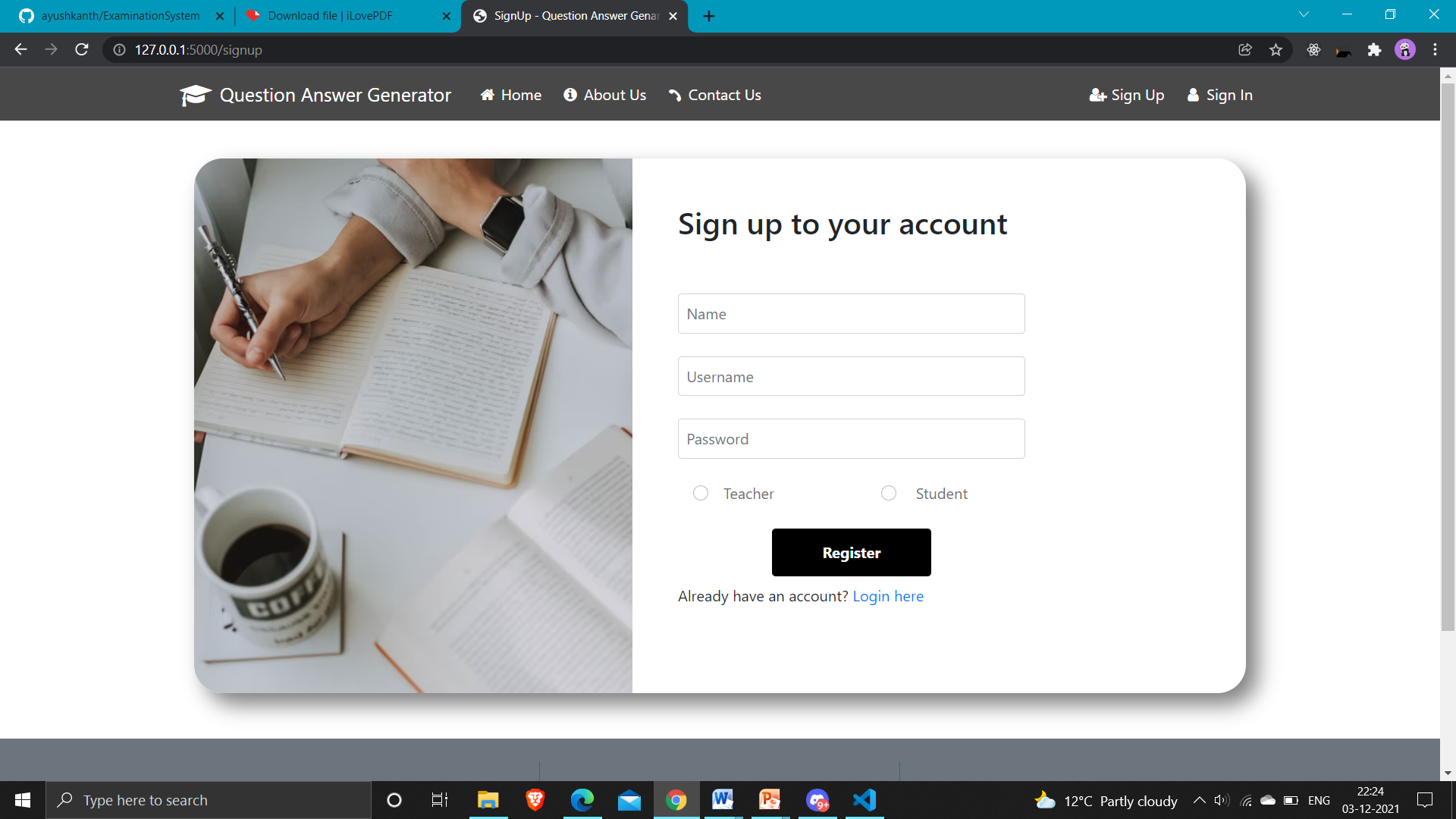


Fig. No.11

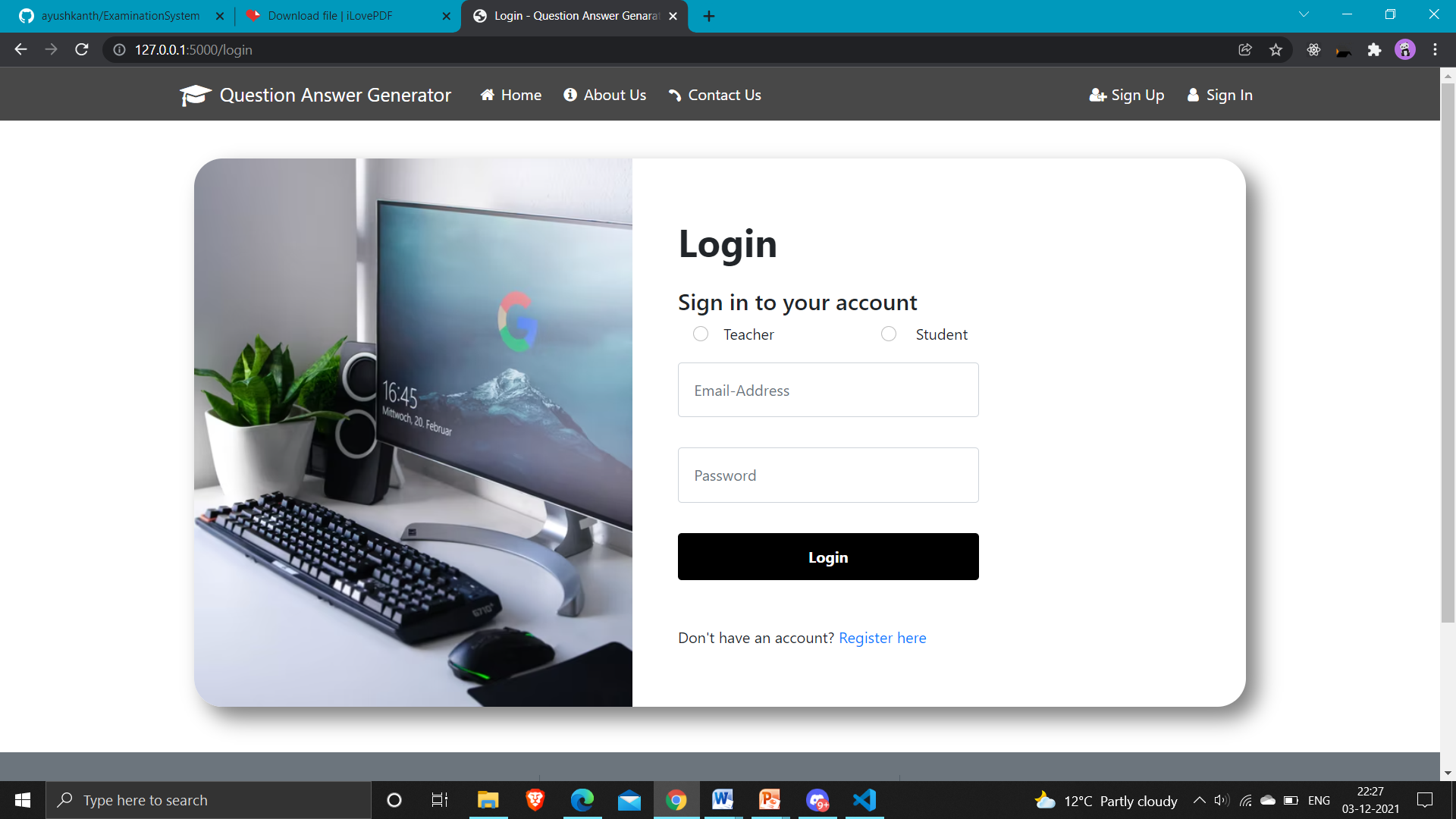


Fig. No. 12

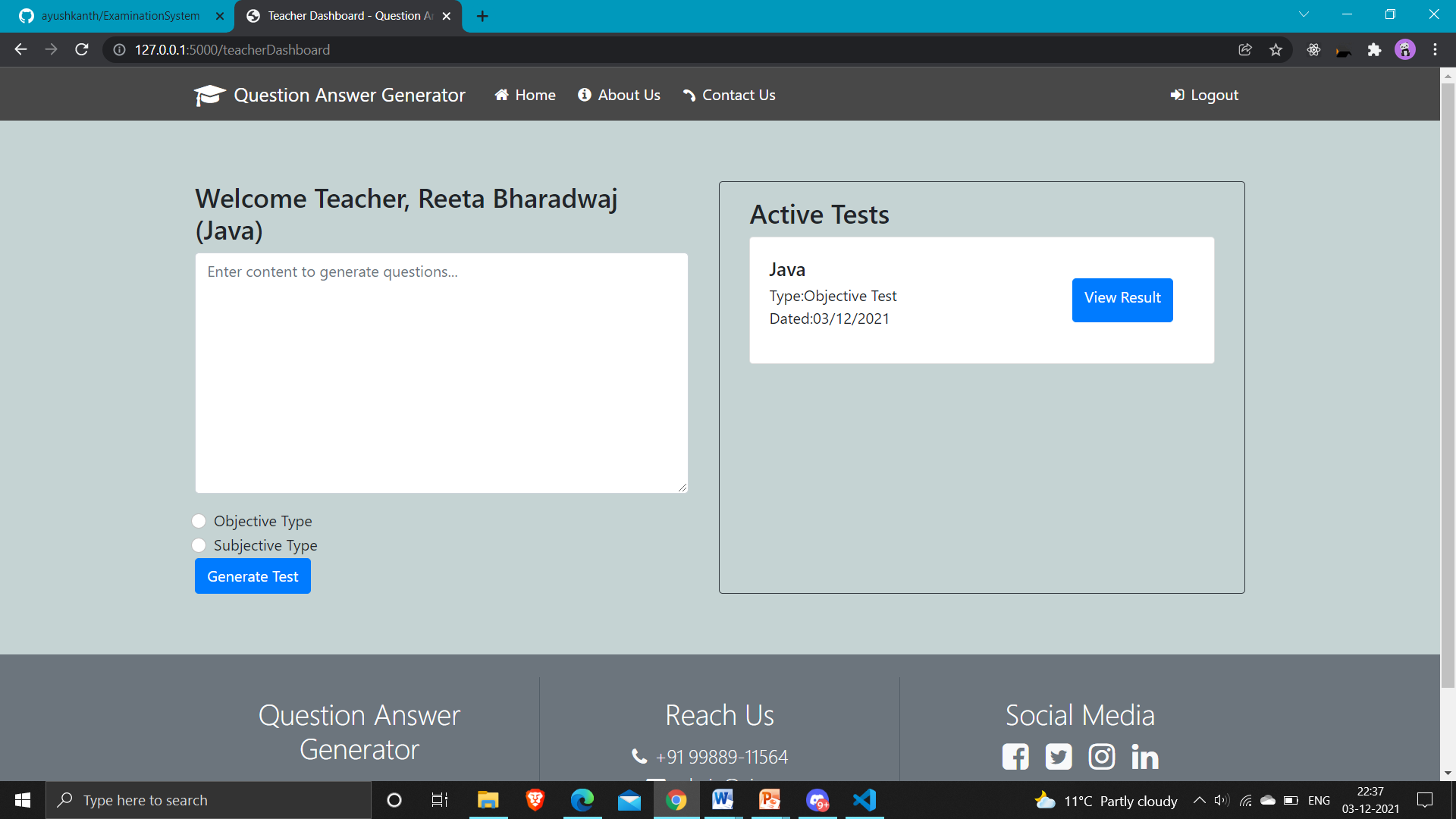


Fig. No. 13

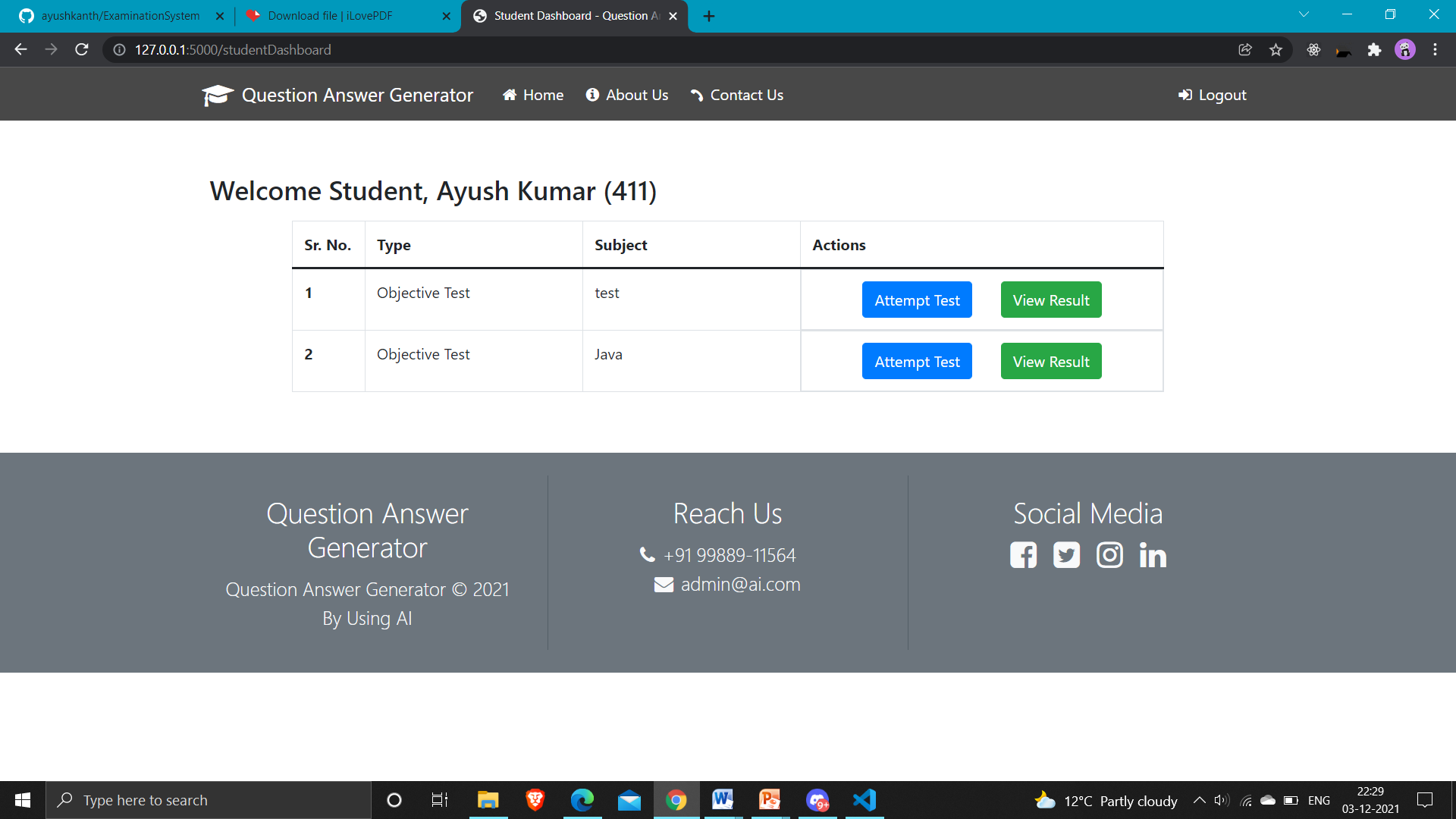


Fig. No. 14

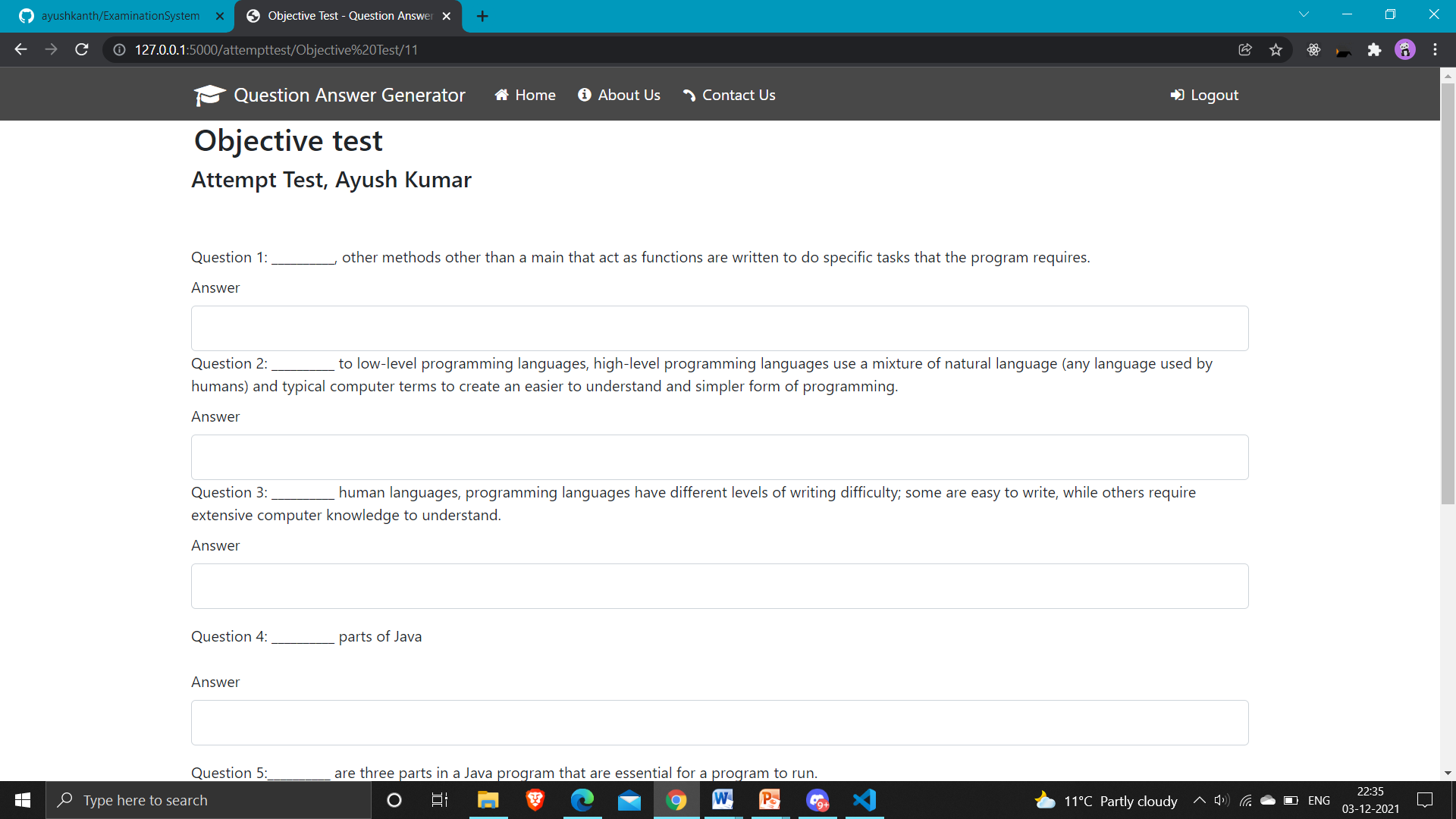


Fig. No. 15

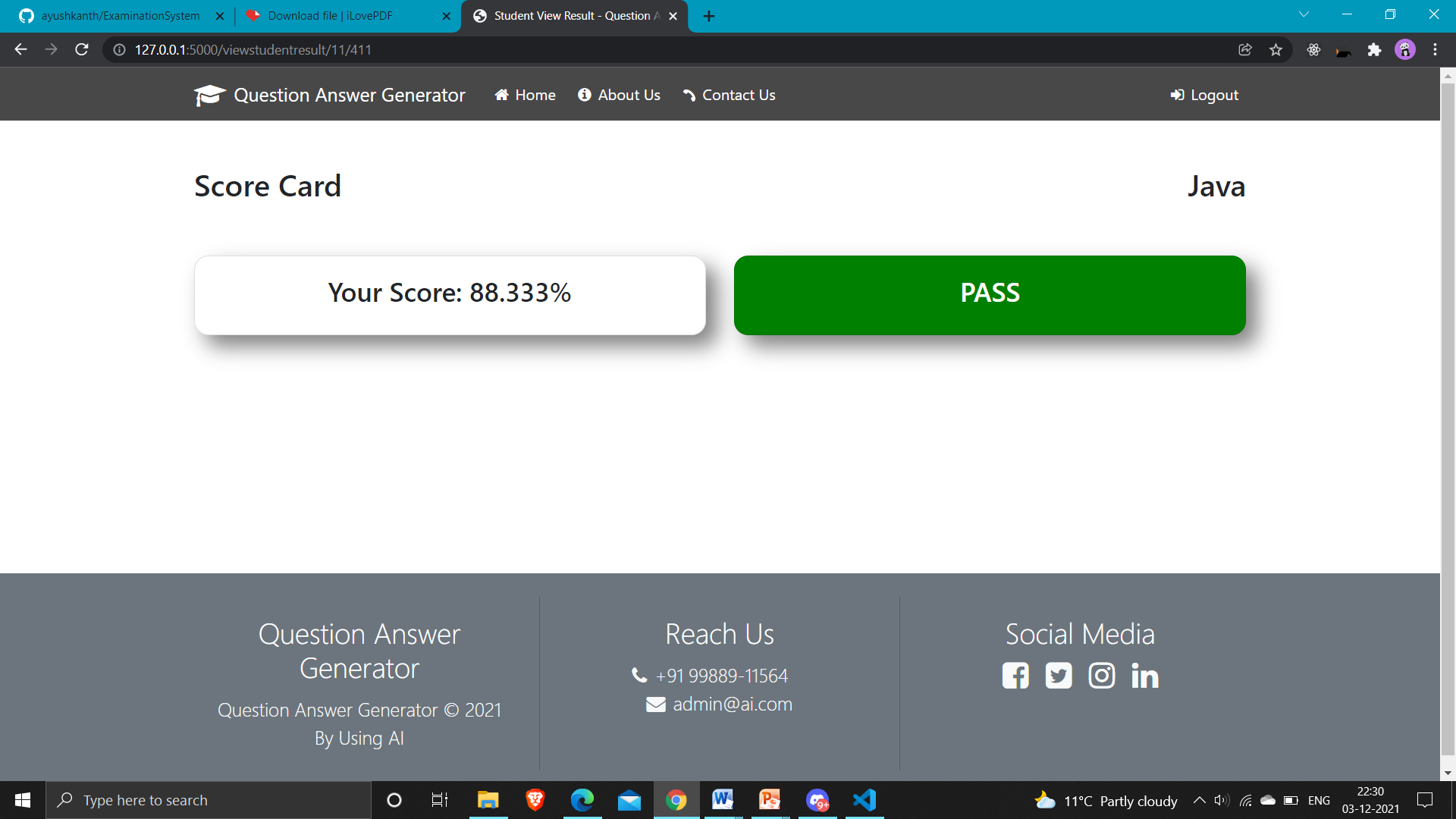


Fig. No. 16

5.4 Comparative Analysis

**[Chapter - 6] Conclusion and Future Scope**

 6.1 Conclusion

Our project is only a humble venture to satisfy the needs of educational institutes to manage their Test-taking task conveniently. Several user-friendly coding has also been adopted. This package shall prove to be a powerful package in satisfying all the requirements of the educational institutes. The manual question generation takes much time and labour. Therefore, automatic question generation from learning resources is the primary task of an automated assessment system. This project presents a survey of automatic question generation and assessment strategies from textual learning resources. The purpose of this project is to summarize the state-of-the-art techniques for generating questions and evaluating their answers automatically.

6.2 Future Scope

Nowadays the fastest developing field in NLP is one of the quickest developing fields from recent years. By the evolution of the period, its significance will increase day by day because online data is present in an enormous amount due to digitalization. Extracting useful information from text has a long way to go. By developing the benefit of connecting content mining to other fields such as machine learning, perception, normal dialect preparing, it could be conceivable to sketch more effective and helpful content mining frameworks. NLP is very useful for the industry to utilize and develop a way of learning that can’t be devoured by people. In this project, we attempted to introduce the application of content mining i.e. the extraction of keywords from text with the use of ensembles approach, NLP, instruments, and applications. Few improvements that can be made are:

* Difficulty level of the question can be improved in the advance version of this project in future.
* User interface can be enhanced.
* Grammatically errors can be improved.

**References**

[1] Indrashis Das, Bharat Sharma, Manjusha Pandey and Siddhart S. Rautaray“An Examination System Automation Using Natural Language Processing”, Fourth International Conference on Communication and Electronics Systems, 978-1-7281-1261-9,2019

[2] Cristley, “Natural Language Processing (NLP) Tutorial with Python & NLTK”, freeCodeCamp.org, 2018 (https://www.youtube.com/watch?v=X2vAabgKiuM)

[3] Catherine Henry, “Natural Language Processing (NLP): Automatic generation of questions and answers from Wikipedia”, Clearwater DevCon - Clearwater Analytics,2017 (https://www.youtube.com/watch?v=3vRZ6dBPL2A)

[4] Zolzaya Luvsandorj , “Introduction to NLP : Preprocessing text in Python”, Towards Data Science, 2020 (https://towardsdatascience.com/introduction-to-nlp-part-1-preprocessing-text-in-python-8f007d44ca96)

[5] B. Kaur, and S. Jain “Keyword extraction using machine learning approaches”, IEEE 3rd International Conference on Advances in Computing, Communication & Automation (ICACCA) (Fall), 2017