CENTRAL UNIVERSITY OF SOUTH BIHAR

PROJECT ON

INTERACTIVE MCQ QUIZ APPLICATION WITH AUTOMATED GRADING

Under The Supervision of

Dr . Nemi Chandra Rathore

Assistant Professor

Department of Computer Science

Submitted by:

Rajat Kumar – CUSB2202312023

Priyanjali Kumari – CUSB2202312022

Mohammed Mubaris B – CUSB2202312017

December 2023

CENTRAL UNIVERSITY OF SOUTH BIHAR

CANDIDATES'S DECLARATION

We hereby certify that the work which is being presented in the project entitled "**Interactive MCQ Quiz Application with Automated Grading**" in the partial fulfillment of the requirements for the completion of the Mini- Project of the **Master of Science in Computer Science** submitted in the Department of Computer Science of Central of Central University of South Bihar, Gaya, is an original work carried out during the period of November, 2023, under the supervision of **Dr. Nemi Chandra Rathore (Assistant Professor),** Department of Computer Science, Central University of South Bihar, Gaya.

The matter presented in the project has not been submitted by us for the award of any other project of this or any other places.

Rajat Kumar – CUSB2202312023

Priyanjali Kumari – CUSB2202312022

Mohammed Mubaris B – CUSB2202312017

This is to certify that the above statement made by the candidates is correct to the best of my knowledge.

Dr . Nemi Chandra Rathore

(Assistant Professor)

AKNOWLEDGEMENT

We take this occasion to thank God, almighty for blessing us with his grace and taking our endeavour to a successful culmination. We extend our sincere and heartfelt thanks to our esteemed guide, Dr. Nemi Chandra Rathore for providing us with the right guidance and advice at the crucial Junctures and for showing us the tight way. We extend our sincere thanks to our respected guide Mr.Purnendu Prabhat, for Guidance and support to do the project work.

We would like to thank the other faculty members also, al this occasion, Last but not the least; we would like to thank to our team members & friends for thie support and encouragement they have given us during the Project Report of our work:

Thank You,

Sincerely,

Rajat Kumar – CUSB2202312023

Priyanjali Kumari – CUSB2202312022

Mohammed Mubaris B – CUSB2202312017

ABSTRACT

The Interactive MCQ Quiz Application is designed as a desktop application, offering interaction with a quiz format. Upon clicking the ‘Start’ button, the quiz initiates, providing user with a structured series of questions and options to choose from. Each question is accompanied by four choices.

The interface guides users through the quiz, allowing them to select an option from the presented choices. After making a selection, user can effortlessly navigate to the subsequent question using the ‘Next’ button. This flow continues until the final question is answered, ensuring user provide a response for each question.

Upon completion of the final question, the application presents a concise and informative summary of the user’s performance. A small message box displays the user’s score, providing clarity on the number of correct responses achieved out of the total attempted questions.

This application emphasizes user-friendliness and efficient interaction, encouraging active participation while providing immediate feedback on the user’s quiz performance. The deliberate design choice to prevent skipping questions is to make a thorough engagement with the quiz content and to make the design more simplify. Overall, the application aims to offer an enjoyable, informative, and seamlessly navigable experience for users engaging in the quiz.

TABLE OF CONTENTS

LIST OF FIGURES

CHAPTER 1

INTRODUCTION

1. INTRODUCTION

The Interactive MCQ Quiz Application is a desktop application in which user can easily interact with questions and also user can see their results. In this project as user clicks the start button the quiz will start. User will be displayed a question followed by 4 options and a button for moving to next question. By clicking any of the choice user can move to next question, in this project we build like this as user cannot skip the questions. So after completing the final question, user can see their result in a small message box.

1.1 SYSTEM REQUIREMENTS

To demonstrate and work with this project a very few hardware and software requirements that are to be satisfied. The major Hardware and Software requirements are listed below.

1.1.1 HARDWARE REQUIREMENTS

- Any Modern Operating System (Preferably 64-bit Architecture)
- 2 GB RAM (Preferably 4 GB)
- 1GB disk space

1.1.2 SOFTWARE REQUIREMENTS

- Python 3.x (Preferably 3.7 or later)

CHAPTER 2

IMPLEMENTATION

2. WORKING ON PROJECT

After satisfying system requirements additionally we are using a python library named Tkinter, its not necessary to use this library. But with this library we can make good GUI also we its very easy to learn. We don't need additional installation of anything as this library is already included in python 3.

Initialization

The script starts by importing the required libraries: tkinter for GUI, simpledialog and

messagebox for dialog boxes.

Graphical user interface (GUI) setup

The Tkinter library is utilized to create the graphical user interface. Beside that here we uses 3 images. One for the logo (QUIZ Image), second for Start the quiz and third one for the next button.

Button functions

The script defines various functions to handle button clicks, including:

StartButton(): To Start the Quiz

NextButton(): To go to next question

r1(): To select 1st option of the question

r2(): To select 2nd option of the question

r3(): To select 3rd option of the question

r4(): To select 4th option of the question

Main loop

The script enters the Tkinter main event loop (root. mainloop(), allowing the GUI tocontinuously handle user inputs and events.

Exception handling

The script uses if else blocks to catch exceptions like if user doesn't press any options it display informative message.

Resizing window

The application window has a fixed minimum size but can be resized within specified limits.

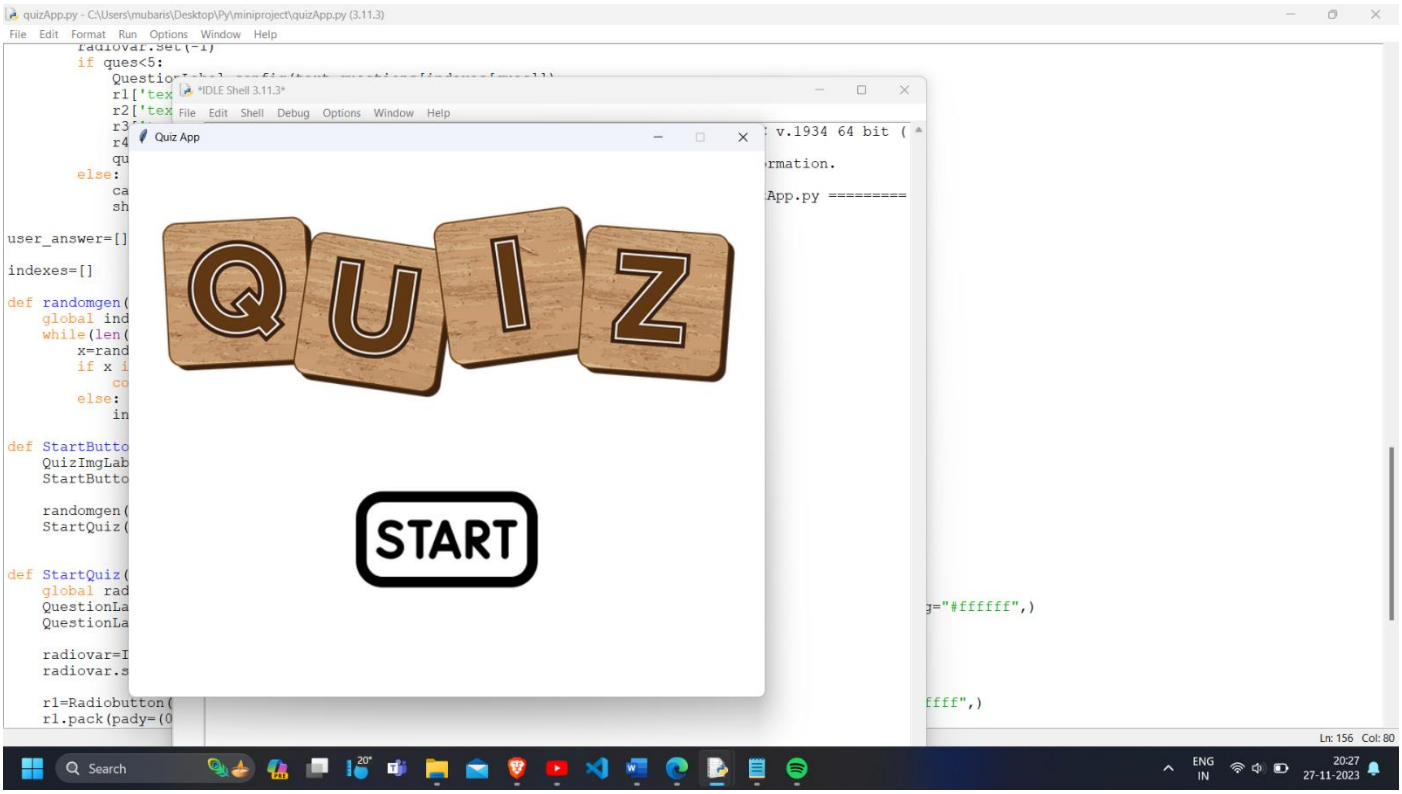
Visual feedback

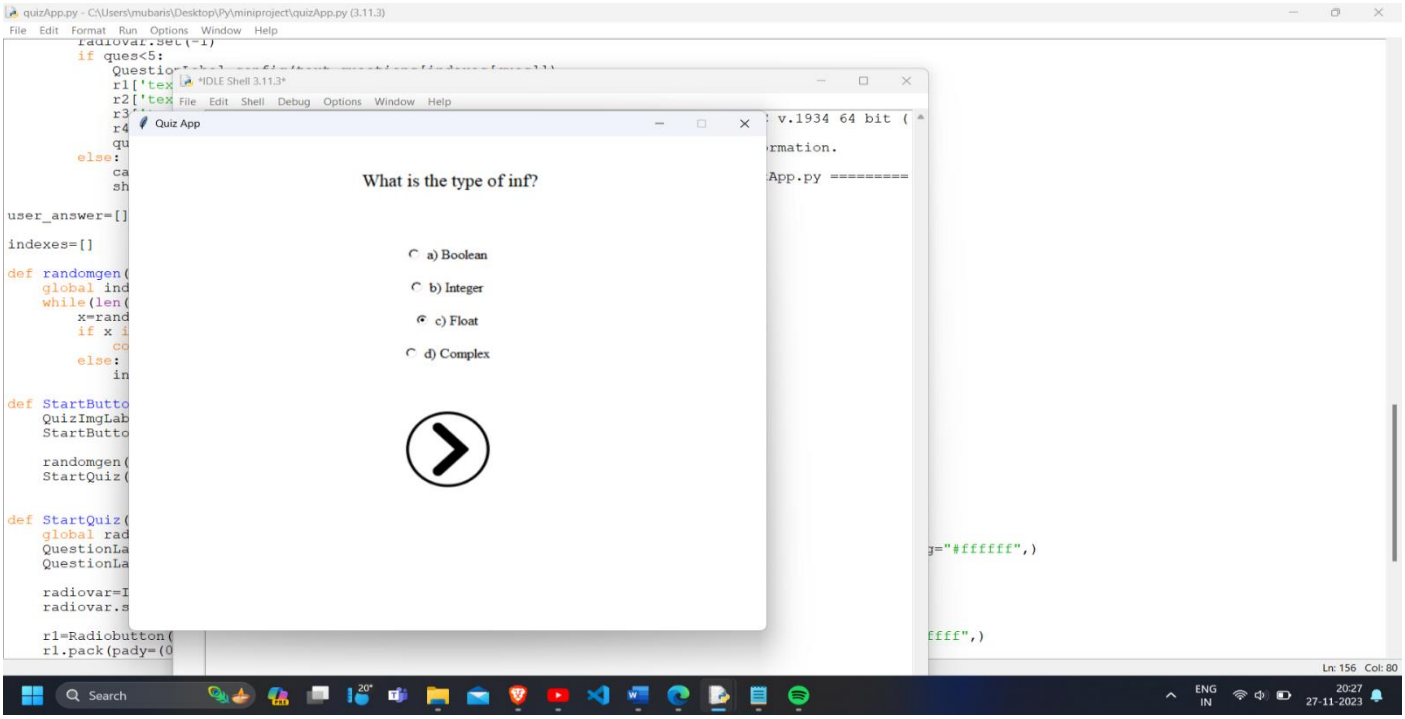
The GUI provides visual feedback to user. After completing the Quiz the application display their result in a messagebox.

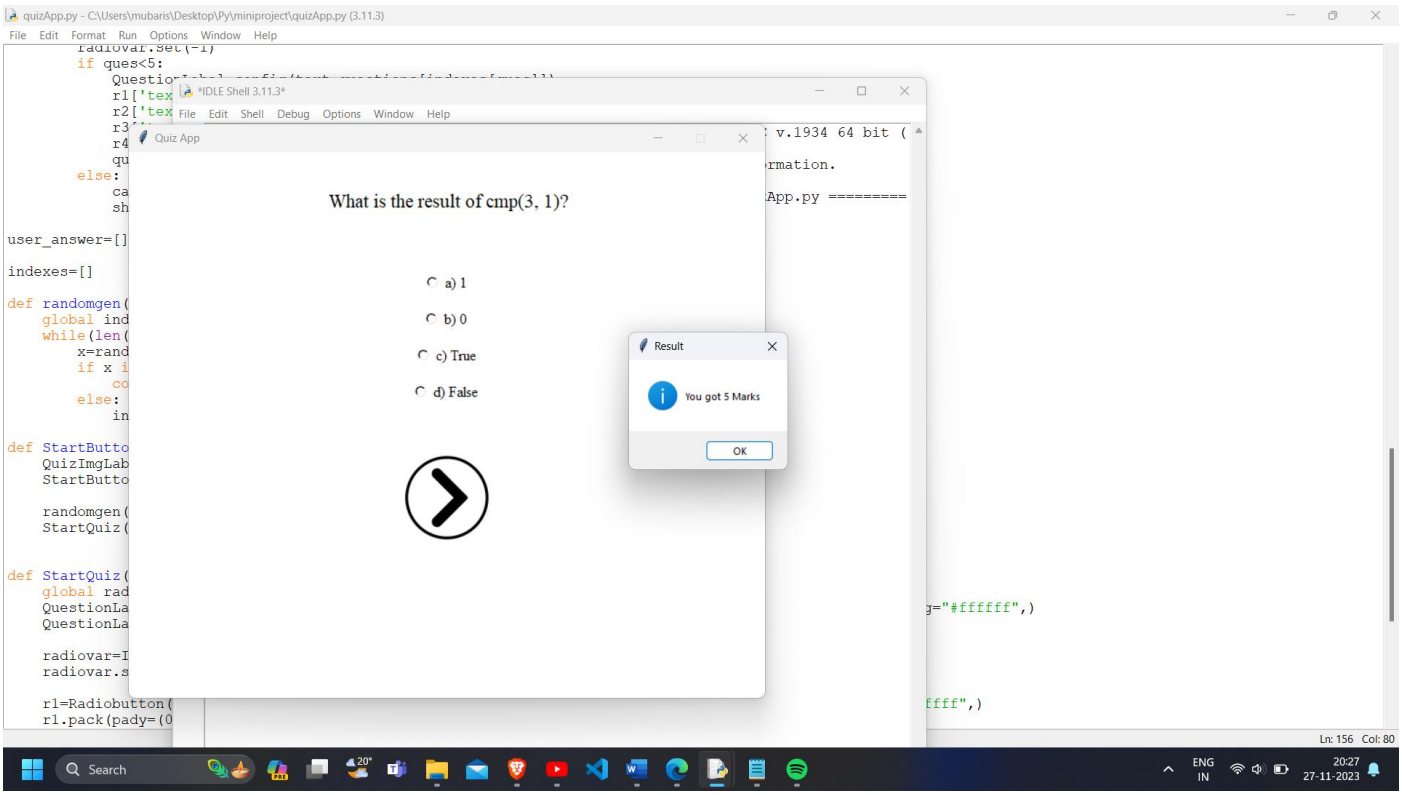
Link to the repository: https://github.com/momubaris/python_miniproject.git

CHAPTER 3

3. SCREENSHOTS






quizApp.py - C:\Users\mubaris\Desktop\Py\miniproject\quizApp.py (3.11.3)
File Edit Format Run Options Window Help
IDLE Shell 3.11.3
File Edit Shell Debug Options Window Help
Quiz App
What is the result of cmp(3, 1)?
a) 1
b) 0
c) True
d) False
Result
You got 5 Marks
OK
Ln: 156 Col: 80
Search
ENG
IN
20:27
27-11-2023

CHAPTER 4

4. CONCLUSION

A multiple-choice question (MCQ) Python application is a versatile and user-friendly tool designed to facilitate learning and assessment. Through a user-centric interface, it accommodates a wide range of subjects, allowing learners to engage with content across various disciplines, from mathematics and science to language and history. The app's dynamic functionality lies in its ability to generate randomized sets of questions, ensuring that each user's experience is unique and diverse.

One of its significant advantages is the instant feedback mechanism it offers. Users receive immediate responses upon answering questions, enabling them to understand concepts better and rectify misconceptions promptly. This quick feedback loop fosters an environment conducive to learning through trial and error, empowering users to learn at their own pace and adapt their approach based on real-time insights.

Moreover, the app's capability to track performance provides users with comprehensive progress reports. This feature allows learners to monitor their growth over time, identifying areas of strength and areas that require further attention. Such insights enable personalized learning paths, where individuals can focus on specific topics or concepts to enhance their understanding and mastery.

The app's interactive nature sparks engagement and sustains interest by presenting knowledge in an engaging format. Incorporating multimedia elements like images, videos, or diagrams can further enrich the learning experience, catering to various learning styles and preferences.

Furthermore, educators can leverage this app as an auxiliary teaching tool. They can create custom question sets tailored to their curriculum, assess students' comprehension levels, and gather valuable data to refine their teaching strategies.

In conclusion, a well-crafted MCQ Python application serves as a powerful educational instrument. It not only cultivates a deeper understanding of subjects but also promotes self-directed learning, adaptability, and continuous improvement, making it an invaluable asset for both learners and educators alike.

References

Tkinter Documentation: https://docs.python.org/3/library/tkinter.html