An Online Self-Assessment Feature for An Enhanced CrsMgr System

Reethu Navale

A Thesis in the Department of

Computer Science and Software Engineering

Presented in Partial Fulfillment of the Requirements

for the Degree of Master of Computer Science at

Concordia University

Montréal, Québec, Canada

April 2022

CONCORDIA UNIVERSITY

School of Graduate Studies

This is to certify that the thesis prepared

By: **Reethu Navale**

Entitled: **An Online Self-Assessment Feature for An Enhanced CrsMgr System**

and submitted in partial fulfillment of the requirements for the degree of

Master of Computer Science

complies with the regulations of this University and meets the accepted standards with respect to originality and quality.

Signed by the Final Examining Committee:

_____ Chair
Dr. Gosta K. Grahne

_____ Examiner
Dr. Dhrubajyoti Goswami

_____ Supervisor
Dr. Bipin C Desai

Approved by _____
Dr. Lata Narayanan,
Chair of Department or Graduate Program Director

_____ 2022

_____
Dr. Mourad Debbadi,
Dean,
Gina Cody School of Engineering and Computer Science

Abstract

An Online Self-Assessment Feature for An Enhanced CrsMgr System

Reethu Navale

The current pandemic has led to the use of online teaching and calls for the introduction of innovative self-learning techniques. Since contacts with educators are limited, students are required to become more self-reliant, and this includes the use of a self-assessment quiz to provide a measure of the learning progress and discover the area for further study. In this project, we focus on adding a feature for a course instructor to generate, using AI techniques, various types of questions for self-assessment: the system uses various types of course-related material, including text chapters in PDF format. The system is designed to be guided by the instructor and would help students to educate themselves in a typical remote teaching/learning environment.

Acknowledgments

To make something come true is always challenging. It will need a lot of people's effort. I would like to express my deepest gratitude to my thesis supervisor Dr. Bipin C. Desai for guidance, support, care, and great patience. I cannot finish this thesis without his help.

I would like to thank the previous authors of CrsMgr, who provided an excellent web application to help the university to improve class management. It is a great pleasure to improve such a web application and allow us to contribute to the open-source code pool.

I would like to say thank you to all students, who gave me suggestions to improve the open browser and new CrsMgr's backend, also took part in the usability testing, and gave me feedback.

I am thankful to all faculty members and staff at Concordia University, who gave me a great environment to do my study and research.

Finally, I would like to thank my parents, my sister, my brother, and my brother-in-law for their patience, encouragement, endless and selfless support.

Contents

List of Figures

List of Tables

Chapter 1: Introduction

CrsMgr [1] is an online system used for over a dozen years for supporting the administrative and pedagogical issues of teaching at the university level. It can be used by the users (students, teaching assistants, instructors, course coordinators, and administrators at the department, and faculty levels). It can be used by the faculty level administrator to add new departments and the administrators at the department level to add new courses and offerings of existing courses. The course coordinator for a given offering of a course could add online quiz topics and questions to be used by any instructor to create online quizzes. In addition to online quizzes, it supports normally written questions the file of the written answer could be uploaded to CrsMgr by the student.

CrsMgr has also been used to replace traditional 'mid-term' and 'final' exams with multiple online 'quizzes' using a combination of multiple-choice questions (MCQ) and 'written' questions. The rationale for this was the goal of objective testing as soon as possible and to avoid the practice of delaying learning till the end of the course which requires cramming and hoping for leniency of the examiners to give marks for 'partial' answers. Online assessments or examinations have become popular. Many educational institutions, proficiency examinations, and corporations' online examinations have adopted online multiple-choice testing methods to evaluate a person's knowledge and ability to solve problems in a limited time.

As a result of the current COVID pandemic, most university courses have moved online. CrsMgr feature of online quizzes can easily be increased in frequency to measure the synchronous nature of the online classes. The topics for the tests could be limited to the most recent lessons. Students also had access to the recording of the lecture during the quiz. One of the issues we discovered was that many students were not used to online quizzes and did not farewell.

The motivation of this project is to create, using the required texts and class notes, on-demand

practice quizzes for the students. The project is to create not only MCQs but also those involving fill-in blanks and true or false types of questions. The application uses the technique in Natural Language Processing (NLP) and Artificial Intelligence (AI).

Furthermore, this system evaluates, and provides feedback for these on-demand self-assessment quizzes (SAQ), and thus help prepare the students for the real quizzes and thus supplement the task of the educator. These questions would be an amalgamation of multiple-choice with one or several correct answers, True or False, and choosing appropriate values for blanks from the learning material.

Another objective of this project is to improve the security of the web application and fix issues of the previous version of CrsMgr which was completely updated in 2016. The other objective of this project is to introduce a feature to replace worst quiz or missed quiz with average quiz score, this helps student not to feel victimized and thus improving the overall score. The course instructor can decide whether to avail the worst by average substitution and select how many worst quizzes to be replaced by average score. CrsMgr displays the raw total that shows the actual score of the student and bonus total that has the substitution score total.

Since many of university courses moved online so does the assessments. The other objective of the thesis is to improve monitoring and logging of the assessments, CrsMgr is logging the quiz start time and time that student has answered the quiz question. This feature helps instructor to understand and solve problems that student face during the quiz. However, logging will not help in identifying if the student is violating academic code of conduct during the online assessment.

Other Learning Management Systems alternatives are present in the market. One among them is Moodle. Moodle is a free and open-source learning management system (LMS). Moodle was originally developed by Martin Dougiamas to help educators create online courses with a focus on interaction and collaborative construction of content, and it is in continual evolution. The first version of Moodle was released on 20 August 2002 (19 years ago) [2]. Many academic institutions are using Moodle.

Canvas LMS is an open-source system that started in the year 2008. Instructure is the home of Canvas LMS [3]. Instructure hit the $1 billion market cap and acquired Practice, the peer-feedback-driven, video-based professional learning tool [3]. Canvas reached more than 30 million global users

[3]. CrsMgr is not available as an open-source yet, soon CrsMgr will be published as open-source in the Git repository for users to download and install in their system.

1.1 System Requirement and Analysis

Online question and answer generation and assessment place a crucial role in the assessment of a learner. Manual generation of question and answer and assessment is time-consuming, expensive, and needs more manpower. Many researchers and top organizations are attracted to build an online question and answering and assessment application. Question types can be broadly classified into Multiple Choice Questions (MCQ), True/False, Fill up the blanks, and Interrogative based questions (What/Who/When/Why/How). An interrogative-based question can require short or long answers. Most online major examinations are generally do not include essay-type written questions due to the need for human interaction and delay in the feedback.

1.2 The Course Manager System (CrsMgr)

CrsMgr is a web application designed for managing course material, quizzes, and other academic tasks. It also provides other necessary functionalities for professors and students. From a functionality perspective, it is an excellent web application. However, the current version of CrsMgr does not have an interface to generate questions nor have self-study quizzes for students to evaluate or increase their knowledge.

1.3 Structure of the thesis

The thesis is organized as follows:

Chapter 1 is introduction.

Chapter 2 discuss the state of art of question generating applications and Learning Management Systems (LMS) and limitations of current version of CrsMgr.

Chapter 3 presents the solution to overcome the problems discussed in Chapter 2.

Chapter 4 includes unit testing and integration testing of the CrsMgr.

Chapter 5 presents the features updated for better security and some enhancement features for CrsMgr. It includes the logging of online quizzes to address the concern of students about the online quiz timing issues. We have also added the feature to allow the instructor to replace the worst grading component with the average of graded work.

Chapter 6 provides the conclusion and future work.

Appendix provides the statistics of questions that are generated from the CrsMgr application.

Chapter 2: State of the Art

2.1 Learning Management System

Learning management systems (LMS) are web applications used for teaching and learning. LMSs are widely used in academic institutions to enhance traditional learning. These web applications are also popular in corporate enterprises for training, employee development programs, and onboarding activities. There are many LMS systems available, some paid and some free.

2.1.1 Comparison of existing applications

There are many LMS applications available in the market either as open-source or as commercial products such as Adobe Captivate Prime[42] and SAP Litmos[43]. However, comparing CrsMgr with these products with commercial products is not fair as CrsMgr is not a commercial product and does not have a team of developers dedicated to development. For a fair comparison, we will compare CrsMgr with open source LMS applications like Moodle and Canvas LMS. Moodle and Canvas LMS are among the famous widely used LMS web application used in academic institutions. Both Moodle and Canvas LMS are supported by a large group of developers, unlike CrsMgr.

Moodle is a learning platform designed to provide educators, administrators, and learners with a single robust, secure, and integrated system to create personalized learning environments [44]. Moodle was started by Martin Dougiamas in 1999. In 2020, registered users passed over 190 million [44].

Canvas LMS is an open and reliable web-based software that allows institutions to manage digital learning, educators to create and present online learning materials and assess student learning, and

students to engage in courses and receive feedback about skill development and learning achievement [45]. The creator of Canvas LMS was founded in 2008 by two graduate students from Brigham Young University, Brian Whitmer and Devlin Daley (Instructure, 2018). In 2011, Canvas was developed as a new generation learning management system built to work on cloud computing and virtualization environments by Instructure [46].

Hardware/Software Requirement:

	Moodle	**Canvas LMS**	**CrsMgr**
Operating system	Windows, Linux/Unix, Mac OS.	Windows, Linux/Unix, Mac OS.	Windows, Linux/Unix, Mac OS.
Database	Oracle, MySQL, MS SQL Server, PostGreSQL.	PostGreSQL	MariaDB

Table 2.1: Comparison on Software/Hardware Requirements

Administrative features:

	Moodle	**Canvas LMS**	**CrsMgr**
Self-Registration	Yes	Yes	No
Batch Registration	Yes	Yes	Yes
Role Based Authorization	Yes	Yes	Yes
Course Marker Role	No. However, there is TA role which can do course marking.	No. However, there is TA role which can do course marking.	CrsMgr has Course Marker, Lab Tutor and Course Tutor role.

Table 2.2: Comparison on Administrative Features

All three-systems support batch registration and access based on the roles, and a user could have multiple access roles. However, self-registration is not supported in CrsMgr since it is designed for traditional registered course management. In Moodle and Canvas LMS there are no separate roles for course marker, lab tutor, and course tutor. However, in CrsMgr we have separate roles for course marker, lab tutor, and course tutor. Course marker has the access to mark the assignments. Both Lab Tutor and Course Tutor do not have access to mark the assignments.

Availability/Access:

	Moodle	**Canvas LMS**	**CrsMgr**
Open Source	Yes	Yes	Not yet published as open source. This updated version is to be released on git along with the documentation.
Cloud Based Application	Yes	Yes	No
Code Customization	Yes	Not easily customizable.	Yes

Table 2.3: Comparison on Availability

Both Moodle and Canvas LMS are open source and are available in the cloud. However, CrsMgr is built as an open-source application but not yet published as open source. Soon CrsMgr will be released on Git with the documentation on setting up the system. Although all three systems allow customization, Canvas LMS is not fully customizable.

Pricing:

	Moodle	**Canvas LMS**	**CrsMgr**
Download Charges	Free but MoodleCloud is paid service	Both free and paid version available.	Free, and the system is very light.
Installation/Set-up charges	Moodle for school does not charge any implementation or set up fee.	One-time implementation fee.	Free
Maintenance/Technical Support	Free Moodle community technical support for free user and experts to help for paid users.	Support page available with issues and solutions for free users and dedicated support through email and phone.	Currently support is ad-hoc. When release on the git, it is hoped that the community would rally to volunteer support.

Table 2.4: Comparison on Prices

All three systems are free to download. However, both Moodle and Canvas LMS have paid versions for better customer experience and usage. Moodle does not charge schools for any setup, and there is a one-time fee for installation in Canvas LMS. However, CrsMgr is a free and very light installation. Both Moodle and Canvas LMS have a team of developers to provide any technical support with the issues. However, in CrsMgr, we do not have a dedicated team of developers to give any

maintenance. The user who downloads CrsMgr has free access to modify the code. There are no restrictions.

Communication Features:

	Moodle	**Canvas LMS**	**CrsMgr**
Discussion Forum	Yes	Yes	No
Announcement from Instructor	Yes	Yes	Yes
Real Time Chat	Yes	Yes	No
Instructor can send an email to students	Yes	Yes	Yes
Instructor can upload files	Yes	Yes	Yes
Student can upload files	Yes	Yes	Yes
Video Files Upload	Can attach video files up to 100 mb.	Can attach video files up to 500 mb.	The size of file upload is decided by the instructor.
Record Video Message	Record video message using Moodle's Atto editor.	Yes	Upload the recorded video file.

Table 2.5: Comparison on Communication Features

All three system allows the instructor to upload files such as lecture notes, assignment, and students are allowed to upload submission files for assignments/projects. All three system allows the course instructor to send emails or announcement to the course students. In CrsMgr, there is no Discussion Form and Real-Time chat feature available.

Course Student Group Features:

	Moodle	**Canvas LMS**	**CrsMgr**
Course Group Creation and Marking	Yes	Yes	Yes
Group Leader Vote	No	No	Yes
Peer Review	No	No	Yes

Table 2.6: Comparison of Course Student Group Features

All three systems allow the creation of student groups and marking. CrsMgr allows the students to vote for their group leaders. Group Leader has access to upload the assignment/project submission. As part of its grading system for group assignments, CrsMgr uses a peer review feature to determine the contribution of an individual group member. However, in Moodle and Canvas LMS does not

have peer review and assign the same marks for all the members of the group even if all the group members have not contributed to the group assignment.

Quiz/Assessment Features:

	Moodle	**Canvas LMS**	**CrsMgr**
Online Quiz Creation by Instructor	Yes	Yes	Yes
Question Bank	Yes	Yes	Yes
Online Grading/Marking	Yes	Yes	Yes
Shuffling of Quiz Questions	Yes	Yes	Random ordering of on-line assessment questions and have multiple versions of the same question and for MCQ, the ordering of the answer is random.
Multiple Attempts for Online Quiz	Yes	Yes	No, however, a question can be banked for later attempt.
Dry Run of the quiz by Instructor	No	No	Yes
Multiple versions of the Quiz Questions	No	No	One question could have multiple versions. The number of versions of each question is decided by the instructor.
Automation on question generation	No	No	Not available for current version. However, new version will generate MCQ and True/False from course material or input by the instructor.
Self-Study Quiz	No	No	No. However, new version has self-study quiz.

Table 2.7: Comparison on Quiz/Assessment Features -1

	Moodle	**Canvas LMS**	**CrsMgr**
Question Type	• Multiple choice • Matching • Ordering • Fill in the Blank • Short answer • Survey questions • Essay • Questions can contain other media elements (images, videos, audio) • Custom question types can be defined • File Upload	• Multiple Choice • True/False • Fill-in-the-Blank • Fill-in-Multiple-Blanks • Multiple Answers • Matching • Numerical Answer • Formula (simple formula and single variable) • Essay • File Upload	• Multiple choice • Short answer • Long answer • Essay • Questions can contain other media elements (images, videos, audio) • File Upload

Table 2.8: Comparison on Quiz/Assessment Features - 2

All three systems have a feature to create online quizzes/assessments. In CrsMgr, the course Instructor can generate self-study quiz questions from the course material or inputs submitted by the instructor. Students can take self-study quizzes, and these quizzes do not contribute to the total score. It is for students to learn or self-assess. However, both Moodle and Canvas LMS do not have features to generate questions and have self-study quizzes for students to try. Multiple attempts on an assessment are not allowed in CrsMgr, and the instructor can take a "dry run" of an online assessment before enabling the quiz to students. The assessment in CrsMgr can shuffle both questions and answers, and the question can have multiple versions. A different question version is chosen randomly for each student to prevent cheating. CrsMgr allows students to bank questions that they find challenging and try later.

Other Features:

	Moodle	**Canvas LMS**	**CrsMgr**
Used for	Both academic and corporate domains.	More suitable for academic purpose.	More suitable for academic purpose.
Student's Course History Information	Yes	Yes	The data for each student in courses administered by the system is accessible to the administrators and the instructor indefinitely.
Schedule Meeting	Yes	Yes	Yes
Calendar	Yes	Yes	Yes

Table 2.9: Comparison on Other Features

All three systems provide a course calendar for users to post events and allow the instructor or TA to schedule a meeting with a student. In CrsMgr, the student's previous history of courses taken is not available. However, the administrator who has access to the database can retrieve the course history.

Based on the above comparisons, we can conclude that both Moodle and Canvas LMS are supported by a large group of system developers. These are better in some features than CrsMgr. However, CrsMgr provides most of the features that are available in these two mature learning management systems and have additional capabilities such as course group management like voting for the group

leader, peer-review feature, and grading based on peer review. The current version of CrsMgr has the facility to create self-assessment quiz questions (self-study quiz) to improve the student's knowledge on the course topics and the instructor has the feature to increase the total score of the student by replacing the few worst scored quizzes with the average quiz score.

2.2 Question generation applications

Online question and answer generation and assessment place a crucial role in the assessment of a learner. Manual generation of question and answer and assessment is time-consuming, expensive, and needs more manpower. Many researchers and top organizations are attracted to build an online question and answering and assessment application. Question types can be broadly classified into Multiple Choice Questions (MCQ), True/False, Fill up the blanks, and Interrogative based questions (What/Who/When/Why/How). An interrogative-based question can require short or long answers. Most online major examinations are generally do not include essay-type written questions due to the need for human interaction and delay in the feedback.

Some online assessment tools are either commercial or free. Socrative is an intelligent student response system that empowers teachers to engage their classrooms through a series of educational games and exercises via smartphones and tablets [26][27]. ProProfs (Quiz Maker) is another online quiz creation app that has both free and paid versions [28]. Quibblo is a free online quiz application [26]. Some organizations have created apps for online quizzes. Google Forms generates different types of questions, and we can also use a tool called Flubaroo for grading [29]. Google Forms has many public templates with some quizzes that can be used. Google Forms works with other Google Apps; you can either send the quiz to students via their mail or embed them in Google Site. Class Marker's secure, professional web-based 'Quiz maker' is an easy-to-use, customizable online testing solution for business, training & educational assessments with Tests & Quizzes graded instantly, saving hours of paperwork [30].

There are some pre-built quiz questions online applications. ThatQuiz [31] is a free online non-commercial resource for teachers and students. There are built-in quizzes for math, science, language arts, and social studies, which are adjustable in both difficulty and length. The interface allows educators to make their tests cover any subject and set of questions, and it even allows multiple languages [31]. ExamTime Quizzes [32], Testmoz [33], Gnowledge [34], Online Quiz Creator [35], GoToQuiz [36], QuizStar [37], Survey Anyplace [38] (also have a mobile app), Mentimeter [39], and Edmodo [40] are some more online quiz creation application is either freely or commercially available for students and educators.

2.2.1 Multiple Choice Questions (MCQs)

MCQ is a form of objective assessment in which respondents are asked to select only correct answers from the choices offered. MCQ is associated with four fields: question sentence, the correct answer(s), distractors, and the number of answers and distractors. The multiple-choice format is most frequently used in educational testing. E. L. Thorndike developed an early scientific approach to testing students. It was his assistant Benjamin D. Wood who developed the multiple-choice test [16][17]. In the mid-20th century, MCQ's popularity increased. Christopher P. Sole created the first multiple-choice examination to aid people with dyslexia in coping with agricultural subjects [17]. At St Edwards School in Romsey, England first MCQ type exam was developed [17][19].

Multiple choice question test types are chosen because they are affordable for testing many students regardless of the topic being assessed. Students can take a guess and answer the question rather than understand the concept to determine the correct answer. Despite all the flaws, MCQs are popular because they are easy to create, score and analyze [18][19].

2.2.2 True/False

A True or False question is known as a polar question or a general question [21]. True or False types of questions have only two possible answers, either true or false. However, it could also be 'Yes' or 'No', 'Agree' or 'Disagree', or any other suitable pair of mutually exclusive responses [20].

One of the huge drawbacks of the True or False types of question is that the learner has a 50% chance of choosing the correct answer, which could be inadequate for testing knowledge. However,

many educational institutions and organizations use these types of questions during the assessment. They can ask tricky questions and confuse the learner so that they can test their knowledge.

2.2.3 Fill in the Blank

A Fill in the Blank question consists of a sentence with a blank space where a student answers the missing word. This type of question can also be clubbed with multiple choice and are easy to evaluate automatically. One of the famous tests that use Fill in the Blank types of question is a cloze test, also called cloze deletion test or occlusion test. A cloze test is an exercise, test, or assessment consisting of a portion of language with certain items, words, or signs removed, where the participant is asked to replace the missing language item [22].

2.2.4 Interrogative

Interrogative questions are of open question that admits indefinitely many possible answers. An open question is also called a variable question, non-polar question, or special question [23]. When, who, where, why, or what words are called interrogative words. These are also called wh-words and these forms open questions. Wh-word is sometimes also referred to as Five Ws, Five Ws and How, 5W1H, or Six Ws [24]. By 1917, the Five Ws were being taught in high-school journalism classes [25]. Wh-questions could also include imperative statements that start with the name, tell, find, define, or describe.

2.3 Brief introduction of CrsMgr

CrsMgr (Course Manager System) is a web-based software system to manage almost all aspects of an academic credit course. It has evolved from a PL/I version developed in the 1980s, through the use of an early database using motif as the graphical interface and thence into the present version with the web as the front end and a MariaDB database as the back end with PHP as the scripting language. CrsMgr has features for use by the various parties involved. These parties are students, professors, teaching assistants, and administrators at various levels including the course coordinators, department chairs, and deans (or their representatives). It provides functions including setting

up courses, course offerings through terms and sections, assigning co-coordinator and various personnel for each newly created course section. The personnel for a course would involve assigning an instructor, tutors, lab instructors, and markers for each new section. In the case of a course having multiple sections during the same term, a course coordinator may also be assigned. CrsMgr provides a facility to create question banks for quizzes; each question would have multiple correct and incorrect answers, and the system would create a multitude of versions for the same question with different choices of answers to select: the question bank could be used to create online quizzes. CrsMgr has features to send email notifications and provides course material sharing among multiple sections of a coordinated course. CrsMgr allows the uploading of files representing the student or group submissions for course assignments, posting of course announcements, auto-grading of online multiple-choice quizzes. CrsMgr facilitates student group management, provides a facility for grading by markers among other functionalities. The various versions of CrsMgr have been used since the 1980s for several courses at Concordia University [15].

However, CrsMgr lacks a feature for automating question generation from the course material and self-study quizzes. One of the objectives of this project is to allow an instructor to create a self-study quiz from the course material. The instructor can modify the generated questions. Students can take the quiz several times to improve their understanding and score. Such self-study quizzes would prepare the student for a quiz that is graded.

2.4 Enhancement of CrsMgr

The above feature implementation is one of the objectives of this project. However, CrsMgr does have a feature where students can practice quiz questions or assess their knowledge of the course. It does not have an interface for the instructor to automatically generate quiz questions. There are many online applications available for question generation however, in this case, student needs to log in to different web application. The implementation of this feature is presented in the next chapter.

Chapter 3: Self-Study Quiz Generation for CrsMgr

3.1 System Design

In this subsystem extension of CrsMgr, questions are generated from the passage that an instructor uploads to CrsMgr. The MCQ, Fill up the Blanks, and True/False types of questions are generated. The questions are generated from the summary of the text passage that the user has submitted in the web application. The application uses the training dataset to generate the questions and the incorrect answers. Spacy training data to identify the part of speech, wordNet and sense2vec word grouping for incorrect answer generation and GPT2(Generative Pre-trained Transformer 2) and bert-base-nli-mean-tokens pretrained model to generate sentences. These questions and answers set can be stored in a database for future use. Just like some systems mentioned in above section 2.2, CrsMgr is free for the students who have registered to the system, unlike other systems that are paid. However, CrsMgr generates questions based on the course material unlike some of the systems that are mentioned above have pre-built questions. The instructor can Add/Delete/Modify the question and the multiple-choice options and submit the questions in the Database.

The process of generating True or False or Fill Up the blanks with multiple-choice questions is loosely based on the blogs and publications of Ramsi Goutham Golla's work [47][48][49].

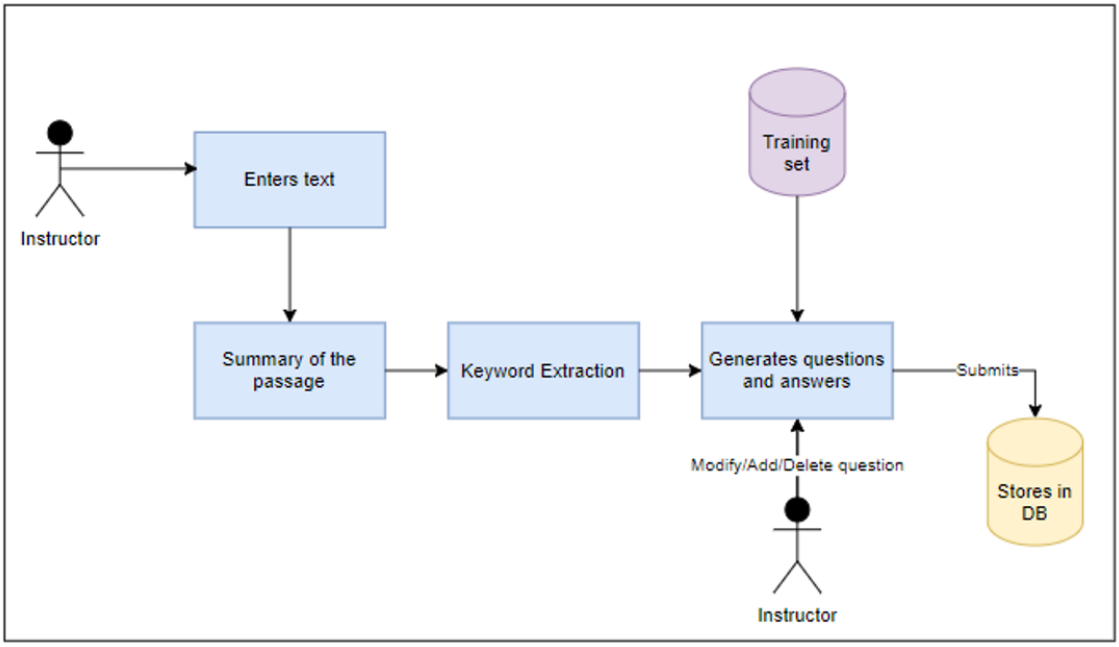


Figure 3.1: Architecture of CrsMgr for generation of practice quiz from User text

In this subsystem extension of CrsMgr, questions are generated from the PDF file that the user uploads. The system will read each page of the PDF file and try to find the paragraph text from the PDF page by considering the font size. The count of texts of the same font size that has a maximum count is considered as the paragraph/passage for generating a summary [50]. Such a summary is generated for each paragraph on the page and questions are generated from it. MCQ, Fill up the Blank and True/False types of questions are generated. The instructor can Add/Delete/Modify the question and the multiple-choice options and submit the questions to the Database.

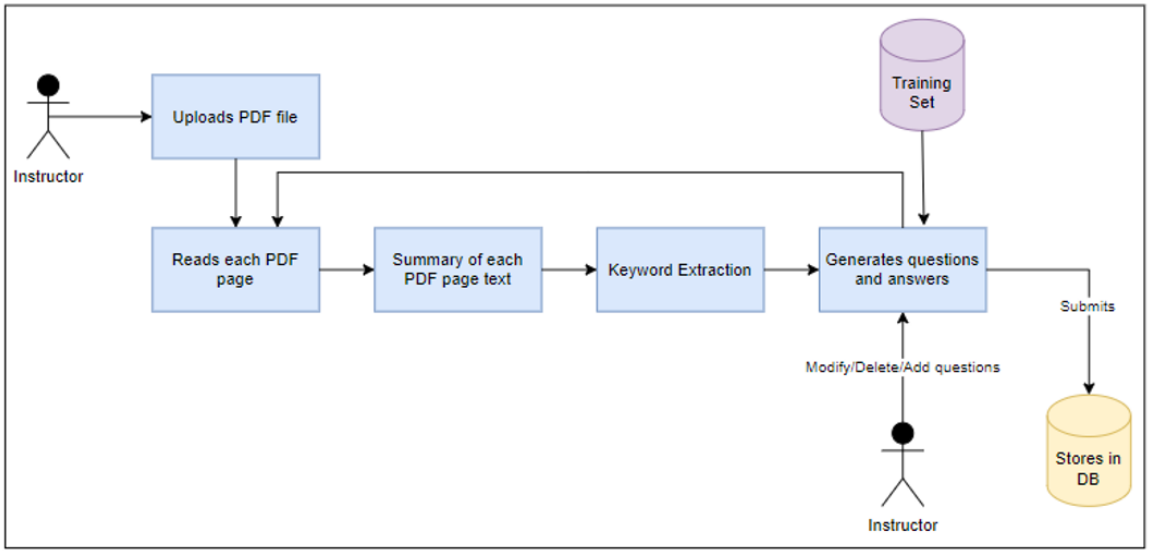


Figure 3.2: Architecture of CrsMgr for generation of practice quiz from PDF file

In CrsMgr, students could do self-evaluation quizzes without having to change portals. Students can also check their scores and the correct answers to their questions. Students can take repeat the quizzes any number of times, and without any time limit. They can take self-study quizzes on any of the topics from a list accessible to them.

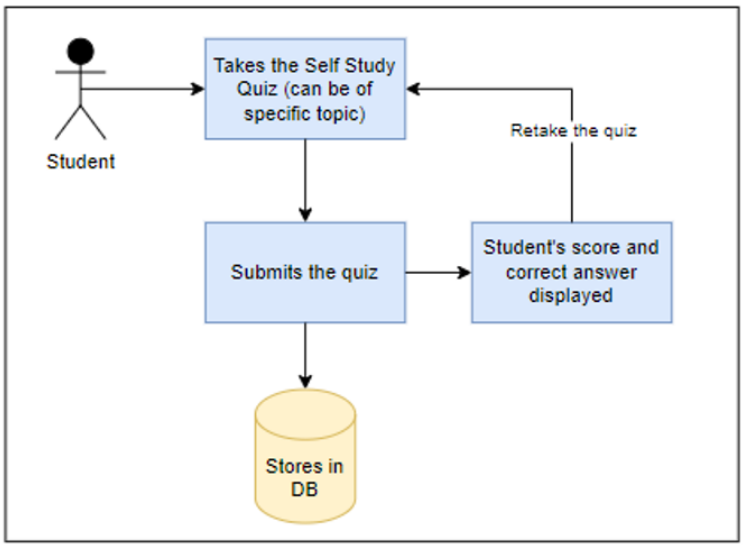


Figure 3.3: Architecture of CrsMgr for Self-Study Quiz

3.2 Implementation

This section describes the approaches that has been used in CrsMgr system.

3.2.1 Summary Generation

In this sub-system of CrsMgr, we use the Summa library for extractive-based summarization. The library selects the most important sentences in a document or text. Summa library uses TextRank[5][51]. In this web application, we define the length of the summary as a proportion of the text. Later the summarized sentences are split into individual sentences. The program considers only mid-length sentences for generating the questions.

The PDF (Portable Document Format) files are made up of text, vector graphics, raster graphics, and multimedia [6]. The user uploads the PDF file and submits it. In CrsMgr, the PyMuPDF python library is used to read the PDF file. Unlike HTML, where each text has tags like header or paragraph, PyMuPDF reads the text but does not have tags or structure to identify the passage text. The program extracts font style information like font size most used fonts sorted by count and color. The HTML tags are decided based on the count of words of the same font size. The highest

number of words of the same font size is assigned paragraph text. The next highest count font text is considered header text. Each sentence is then marked with its corresponding HTML tags that are identified by the process mentioned above. For the extractive-based summarization, the program considers the sentences that are marked as paragraph text. The program opens PDF documents page by page, identifies the HTML tags, and generates the summary. However, this approach does not give a 100% result. If the PDF page contains any program or table, then the program will fail to identify the paragraph text.

3.2.2 Generation of True/False Question

In this sub-system of CrsMgr, the True and False types of question are generated by identifying part-of-speech tags such as Verb, Adjective, Noun, Named Entity, and Negation words from the summarized text. Replace the identified part-of-speech tagged words with the incorrect word (or distractor). Spacy library is used to identify the part of speech tags from the summary sentences. If the selected mid-size summarized sentence does not have any negation words, then the program will generate a negation sentence. To generate the negation sentence, the program identifies the VERB part-of-speech tag and adds the "not" word in front of the VERB word.

True Sentence: A database is a place where data is stored.

Here **"stored"** word is a VERB. The program will add **"not"** word in front of the VERB word.

Program generated Negative/False sentence: A database is a place where data is ***not*** stored.

If the sentence is a negation sentence, then generating a non-negation sentence will create a False question.

True Sentence: The shell program, mysql, is not case sensitive meaning that instructions can be typed in uppercase or lowercase or a combination of both.

The program identifies the negation sentence and removes the negated word, thus making the sentence a non-negation sentence i.e., a False sentence.

Program generated Negative/False sentence: The shell program, mysql, is case sensitive meaning that instructions can be typed in uppercase or lowercase or a combination of both.

Apart from identifying the part-of-speech, the program uses a sentence transformer to generate the questions. The program takes the summarized mid sentences and passes the partial sentence to the sentence transformer model. The model will generate a list of sentences, then the sentence that has the highest cosine similarity is considered. The program uses a "bert-base-nli-mean-tokens" pre-trained model for sentence transformation.

True Sentence: A relational database consists of one or more tables which are containers for the data.

Generated False Sentence by the Sentence Transformer: A relational database consists of one or more tables which are containers for data that can be stored on a single host system (such as SQL Server and Oracle Database).

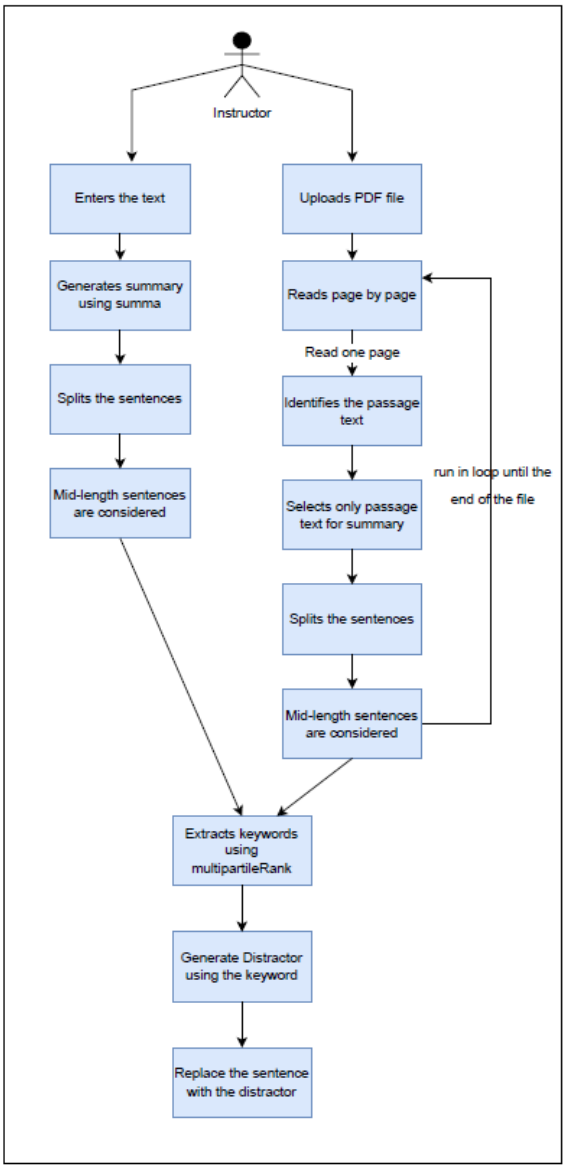


Figure 3.4: Generation of True/False Questions

3.2.3 Extracting Keywords

Keyword extraction helps in extracting the most important words from the text and the sentence that contains the keywords can be used to generate quiz questions. In the sub-system of the CrsMgr web application, the program extracts the phrases or words that are NOUN. The program uses MultipartiteRank unsupervised model to extract the keywords from the text. This model selects the

longest sequence of nouns that do not contain punctuation marks or stopwords as keywords. Later build the Multipartile graph and rank the keywords.

3.2.4 Generation of Distractors

Distractors are the wrong answer or keyword for a given question. The quality of the question depends heavily on the quality of the distractors. The distractors/incorrect option for a given question should be similar to the given question but not the answer. If the system fails to generate a good distractor, then students can easily guess, and it would be difficult for instructor to evaluate the students.

WordNet, and Sense2vec are used in this sub-system of CrsMgr to generate distractors. WordNet is a large lexical database of English [9]. Nouns, verbs, and adverbs are grouped into sets of cognitive synonyms (synsets). If the answer keyword belongs to any synsets then any other keywords that are part of the same synsets can be considered as the candidate for the distractor. However not all words have synsets. For example, the keyword "MySQL" does not have any synsets. Sense2vec helps in learning detailed word vectors [10]. Sense2vec is an extension of the Word2vec algorithm. Sense2vec creates embeddings for the token/word along combined with a label. The label can be a POS Tag, Polarity, Entity Name, Dependency Tag, etc. The topmost similar words can be used as distractors. Phrases can also be used in sense2vec.

To generate a better-quality distractor, the program uses WordNet, and sense2vec and creates a list of words from the output of each function. Each word in the list is then checked if they belong to the same synsets using WordNet. The final list of words is then considered as the list of distractors for a given keyword.

Keyword: machine learning

Generated list of Keywords: ['Data Analysis', 'Computer Vision', 'Natural Language Processing', 'Deep Learning', 'Data Science']

If both i.e., WordNet, and Sense2vec does not give any results then a random word from the text that is not the keyword or stopword or any other keyword from the keyword generated list is considered as the distractor for the question.

Using of less similar word from the concatenated list of wordNet and sense2vec or using random word or using another word/phrase from the keyword extracted list from the text is my contribution to the thesis.

3.2.5 Generation of Fill Up the Blanks with Multiple-choice Question

MCQ is a form of objective assessment in which respondents are asked to select only correct answers from the choices offered. MCQ is associated with four fields: question sentence, the correct answer(s), distractors, and the number of answers and distractors. The multiple-choice format is most frequently used in educational testing. E. L. Thorndike developed an early scientific approach to testing students. It was his assistant Benjamin D. Wood who developed the multiple-choice test [11][12]. In the mid-20th century, MCQ's popularity increased. Christopher P. Sole created the first multiple-choice examination to aid people with dyslexia in coping with agricultural subjects [12]. At St Edwards School in Romsey, England first MCQ type exam was developed [13][12].

Multiple choice question test types are chosen because they are affordable for testing many students regardless of the topic being assessed. Students can take a guess and answer the question rather than understand the concept to determine the correct answer. Despite all the flaws, MCQs are popular because they are easy to create, score and analyze [14][12].

In the sub-system of CrsMgr, the summarized text from section 3.2.1 above is taken for the keyword extraction process. Only the sentences of mid-size that contain the keywords are considered. Then the question is generated by replacing the keyword with blank. The same keyword is then used for generating the distractors or incorrect answer option for the question. Along with the distractor the program also selects the other keywords that are extracted from the text as an option for incorrect answers.

Question generated from the model: Conceptually, a _____ is a container, and the data is the contents of the container.

Options generated from the models:

- Database [Correct Answer/Keyword]
- Software components [Model generated distractor]
- Database model [Model generated distractor]
- Query [Other keywords]

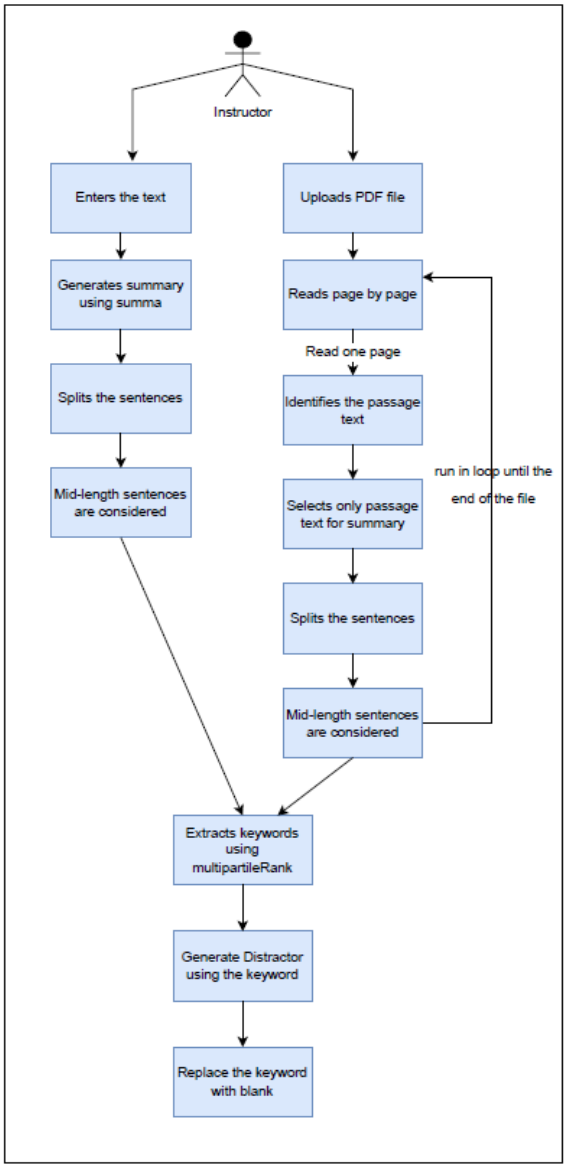


Figure 3.5: Generation of Fill Up the Blanks MCQ

3.3 Illustration of the system

This section illustrates the process of generating questions in CrsMgr and self-study quiz options.

3.3.1 Generation of questions from user text

Only the Course Instructor/Coordinator can generate the questions. The instructor will select Create Quiz Questions from the main menu option once they login in CrsMgr. The instructor needs to type, or copy-paste the text. CrsMgr will generate questions using this text. Along with the text, the instructor needs to select an appropriate topic name for the given text. To generate the questions Instructor needs to click on submit button. The topics are created by the department administrator.

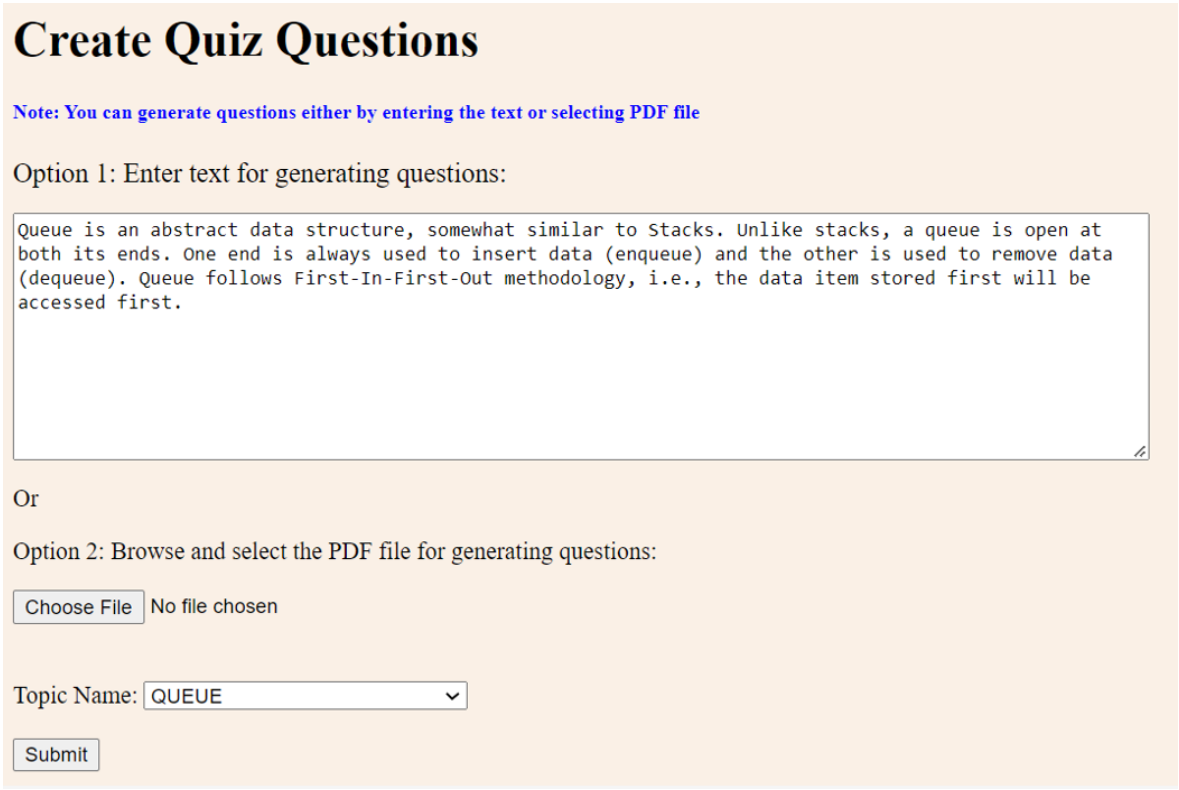


Figure 3.6: Course Instructor to generate questions from text

The CrsMgr system will generate both True/False and Fill Up Blanks with MCQ using the text that Instructor has entered. CrsMgr will display the system-generated questions on the screen. However, these questions are not the final set of questions that will be used for the self-study quiz.
CrsMgr has the option for the instructor to validate the questions manually that are generated by the program. The instructor can also edit or delete or add a new question.

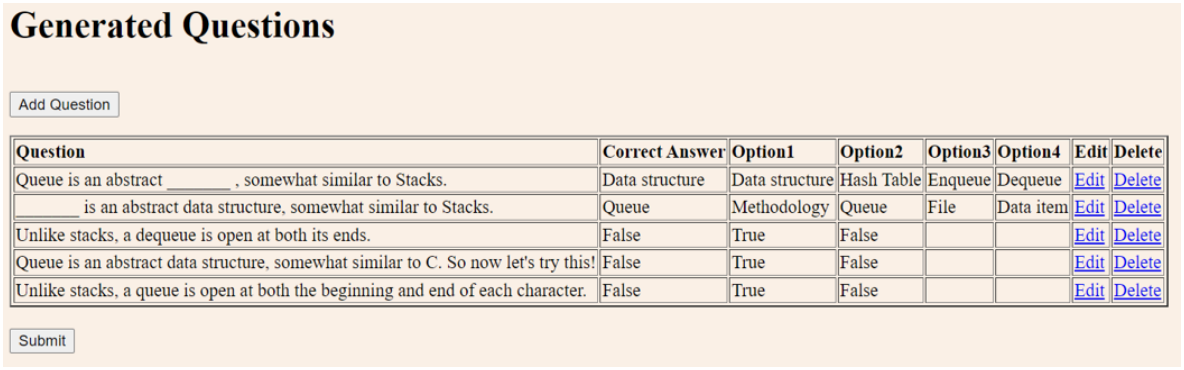


Figure 3.7: CrsMgr generated questions from text

Once the instructor validates the questions and decides to commit, the instructor clicks on the "Submit" button. Only the committed/submitted manually validated questions are used for the self-study quiz.

On successful commit of manually validated questions to the database by the instructor, a message is displayed. The below figure 3.8 shows the message that is displayed after a successful commit.

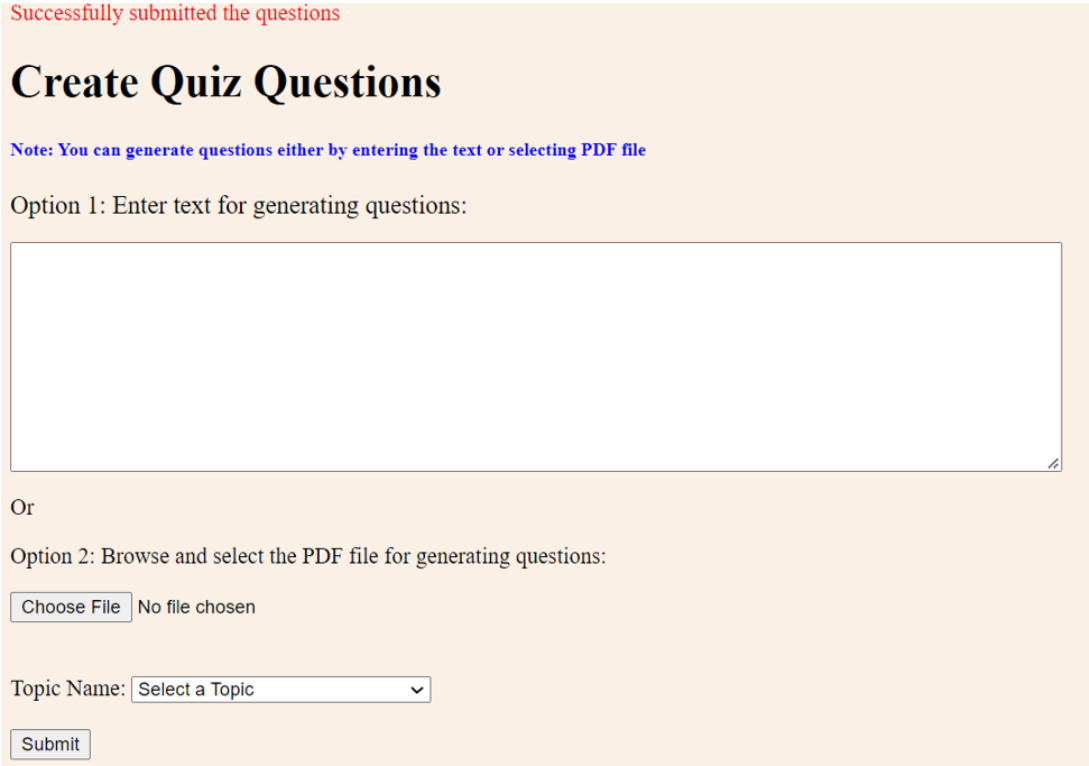


Figure 3.8: Message on successful submission

3.3.2 Generation of questions from a PDF file

CrMgr has a feature to generate questions not only using text but also by using a PDF file. The instructor will select Create Quiz Questions from the main menu option once they login in CrsMgr. The instructor can upload the PDF document for question generation. Currently, CrsMgr only supports PDF files. To browse and select the file from the local device, the instructor needs to click on the “Choose File” button. Below figure 3.9 shows the screen that has the feature.

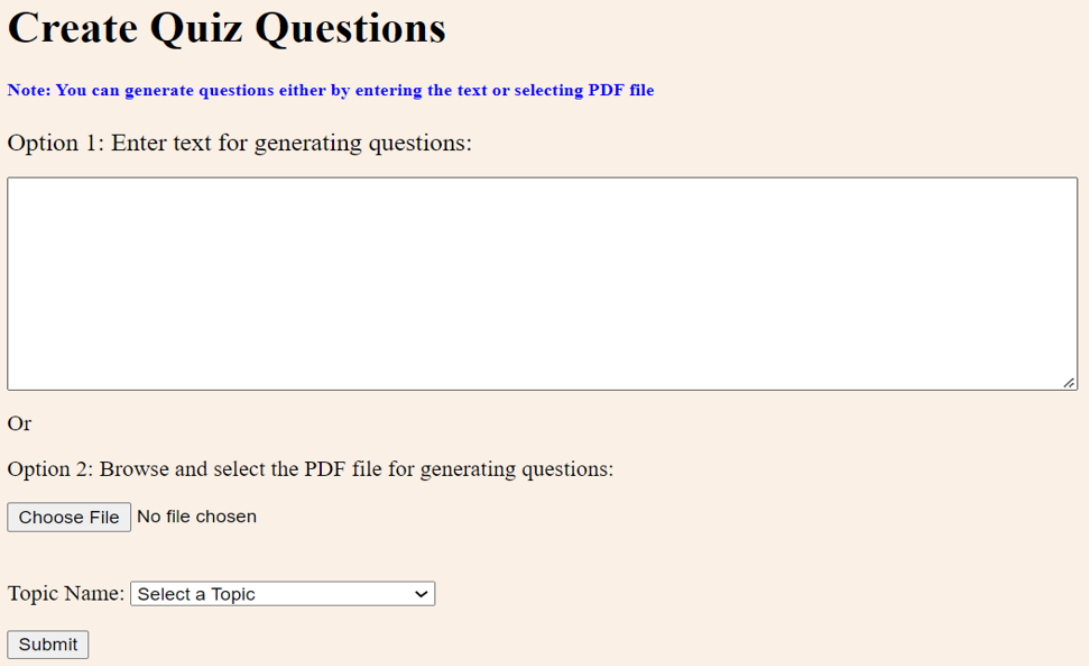


Figure 3.9: CrsMgr has option to generate questions from PDF file

Instructor needs to browse for the file in local storage and select the appropriate PDF document for upload.

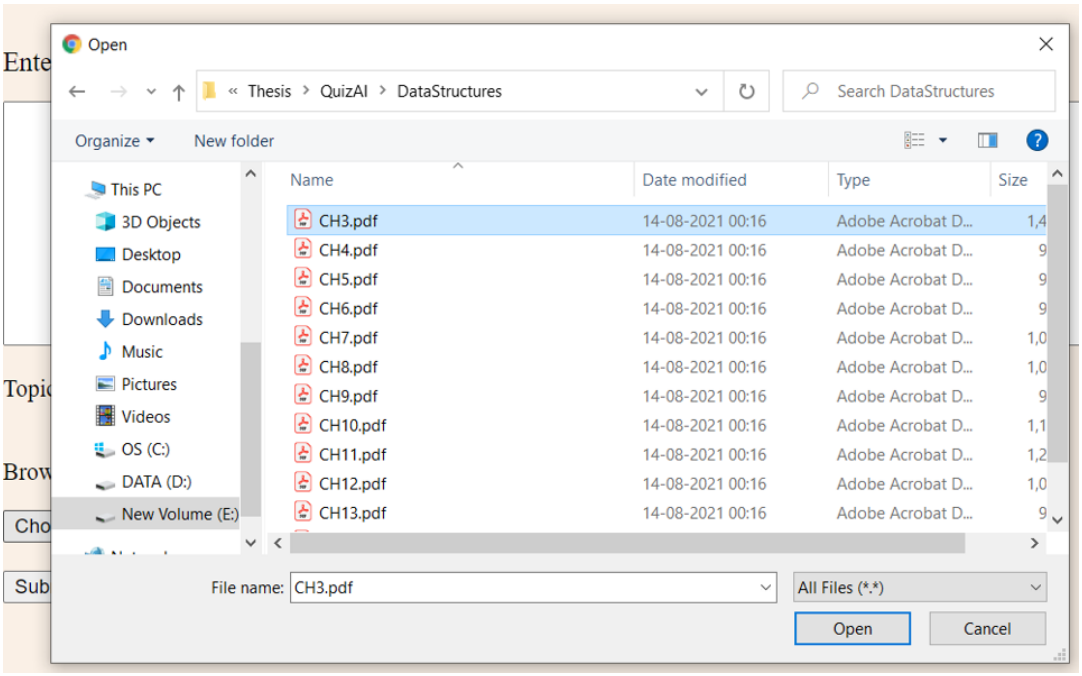


Figure 3.10: Instructor browse file from local storage

The instructor selected file name is displayed on the screen. Instructor needs to select proper topic name and click the “Submit” button to generate questions.

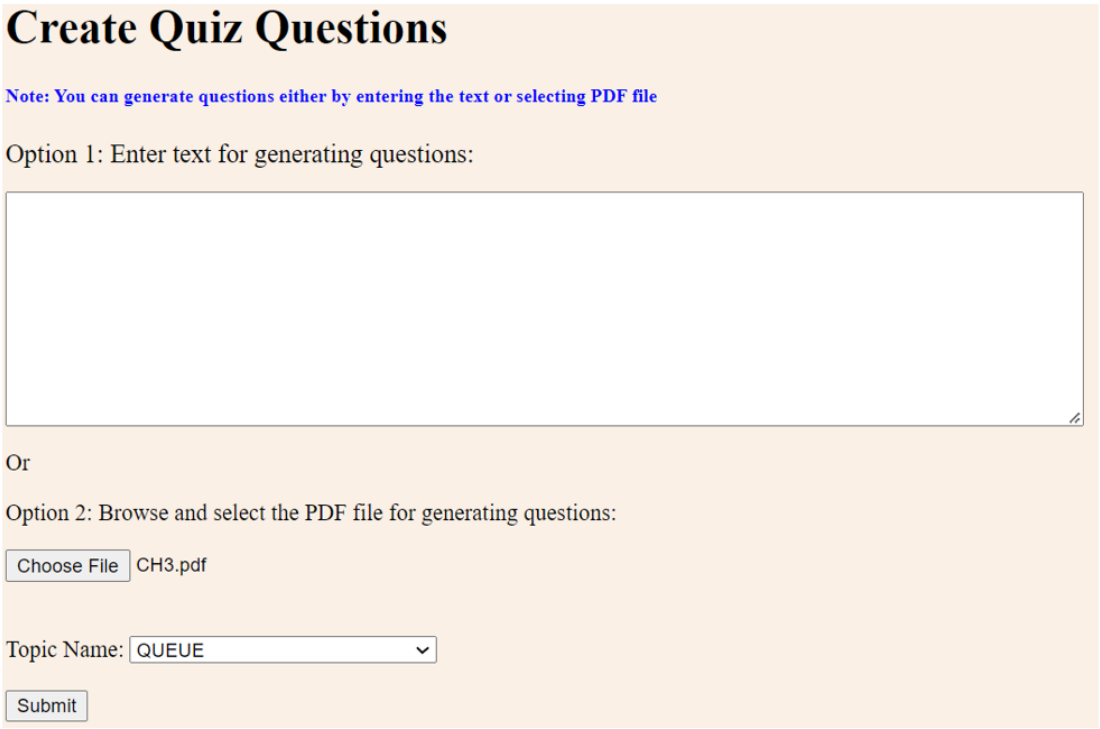


Figure 3.11: Instructor selecting topicName and PDF file

3.3.3 Modify/Delete/Add Questions

3.3.3.1 Delete the Question

CrsMgr has an option to delete the questions from the generated list of questions. To delete the question, the instructor needs to click on the Delete hyperlink for each question row.

Below figure 3.12 shows the list of questions before deletion of the question by the course Instructor.

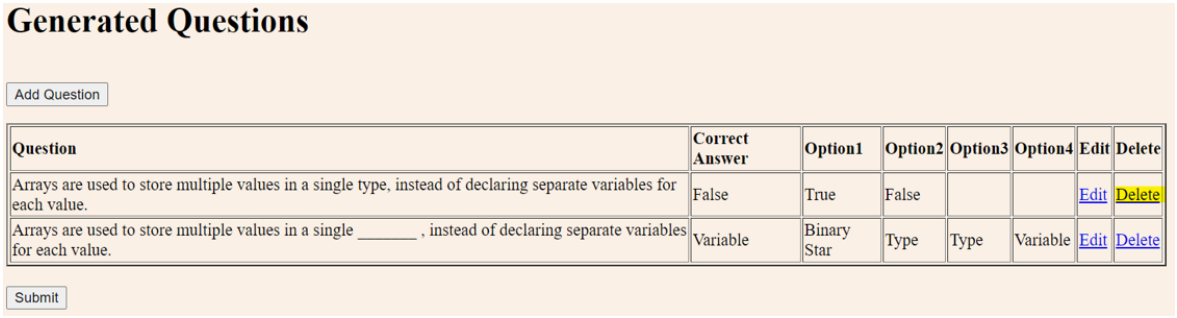


Figure 3.12: Instructor has option to delete the generated question

Below figure 3.13 shows the question lists after deletion of the question by the course instructor.

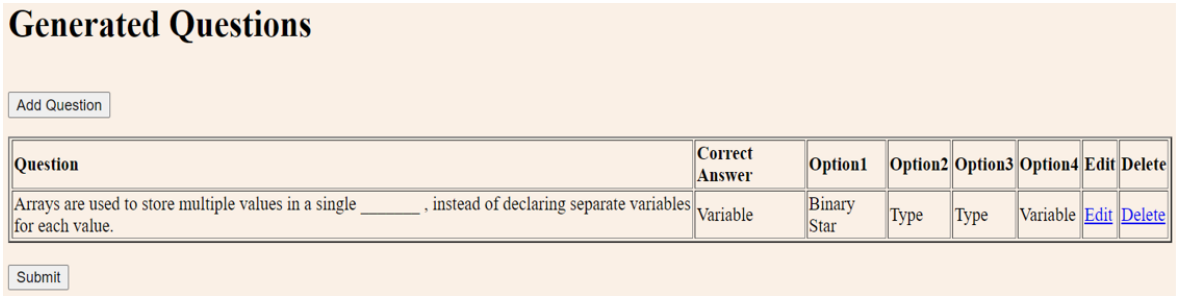


Figure 3.13: Generated question list after deletion

3.3.3.2 Modify/Edit the Question

The instructor can modify the question/answer/options for a given question. To modify the question the instructor needs to click on the “Edit” hyperlink for the question.

Below figure 3.14 shows the list of questions before modification/edit of question.

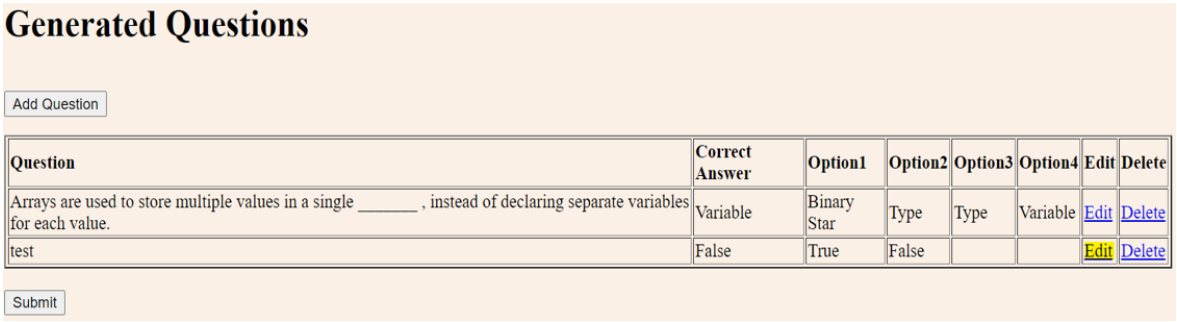


Figure 3.14: Generated question list before modification

CrsMgr has an option for the instructor to modify the question, its answer, and its options. To commit the changes made for the questions instructor needs to click on the Update button.

Below figure 3.15 shows the frontend web page where the instructor can provide the question details.

Figure 3.15: CrsMgr having option for Instructor to modify the question

Below figure 3.16 shows the course instructor typed the modified question. To commit the changes in the database course instructor needs to click on the “Update” button.

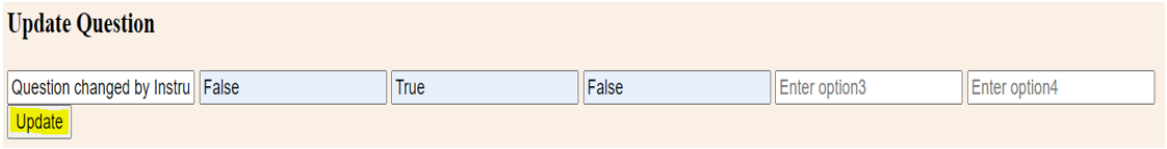


Figure 3.16: Instructor commits the changes

Below figure 3.17 shows the highlighted question that is modified by the course instructor.

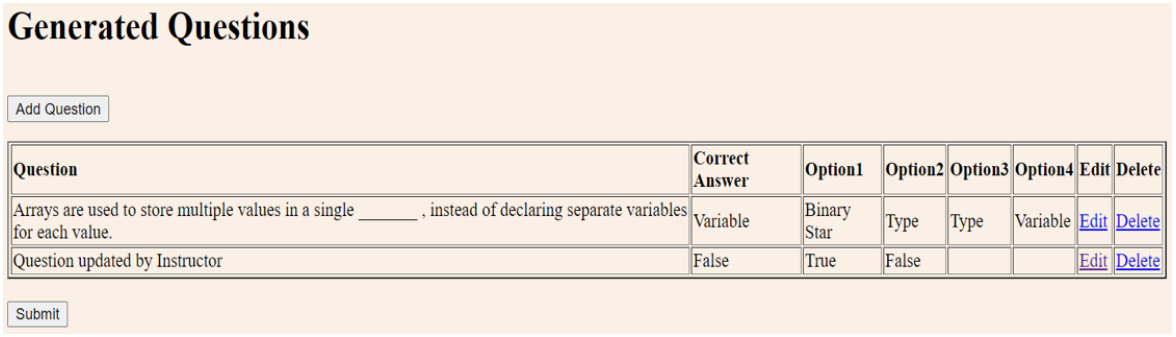


Figure 3.17: Generated question list after modification

3.3.3.3 Add New Question

CrsMgr has a feature to add a new question to the generated question list. The instructor can either add True/False or Fill Up the Blanks with multiple-choice options. To add a new question, the instructor needs to click on the "Add Question" button as highlighted in the below figure 3.18.

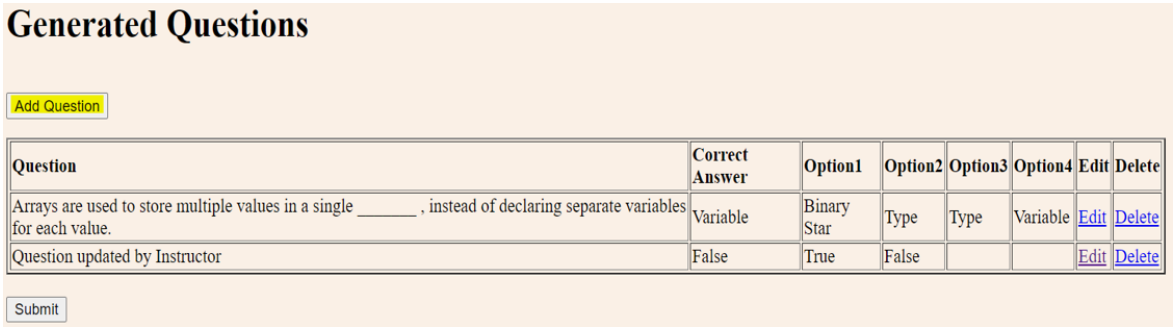


Figure 3.18: Instructor has an option to add new question

The instructor enters the question, answer, answer options (includes both correct and incorrect answers) and its question type ("TF" for True/False and "MCQ" for Fill Up The Blanks with multiple choice option). Later click on the "Add" button to add the new question to the question list.

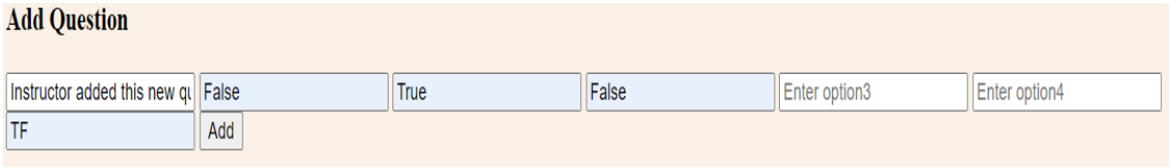


Figure 3.19: Instructor has an option to add new question

Newly added question will be displayed in the question generated list. In the figure 3.20, shows the new question that course instructor has added, the new question is highlighted.

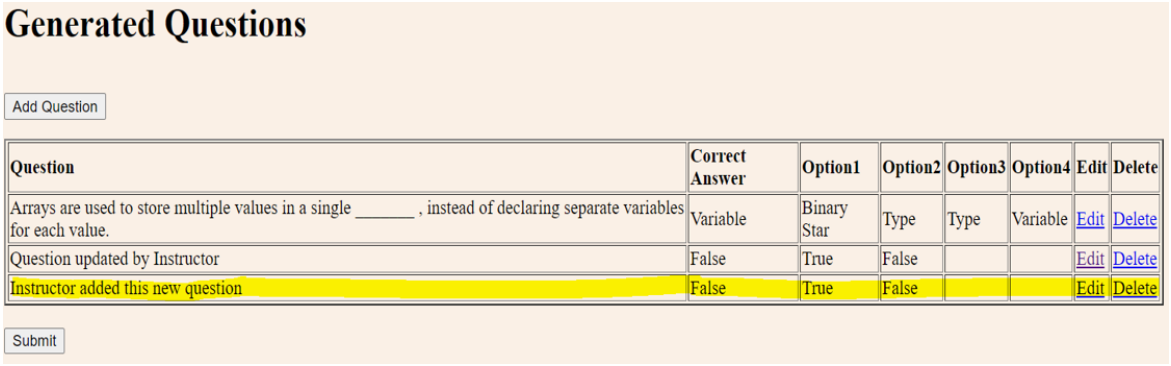


Figure 3.20: Screenshot showing new question added

3.3.4 Student taking self-study quiz

Students can take a quiz on specific topics or quiz on overall topics present for the course the student has enrolled in. This self-study quiz does not have any weight for the overall course score. Students can take these quizzes any number of times and there is no time limit for these quizzes.

3.3.4.1 Self-Study quiz

To take the self-study quiz student should log in to the CrsMgr portal and select the option “Self-Study Quiz” from the main menu.

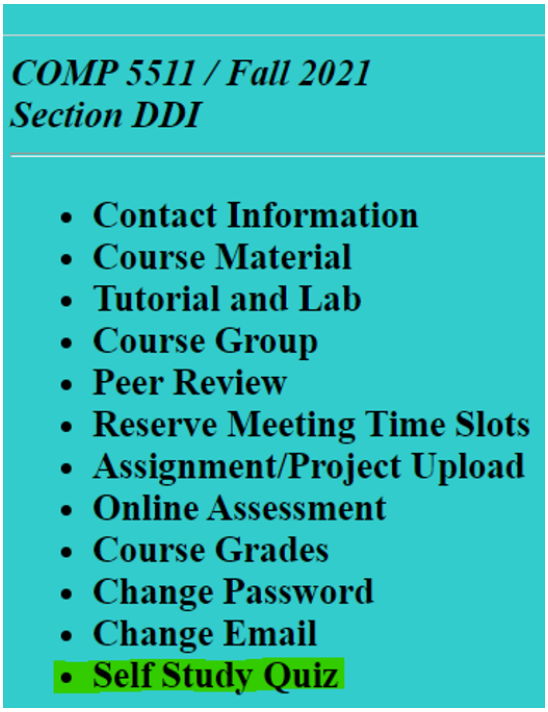


Figure 3.21: Options available for Students

To take a self-study quiz on all topics related to the course. The student needs to click on the "TakeQuiz" button highlighted in figure 3.22.

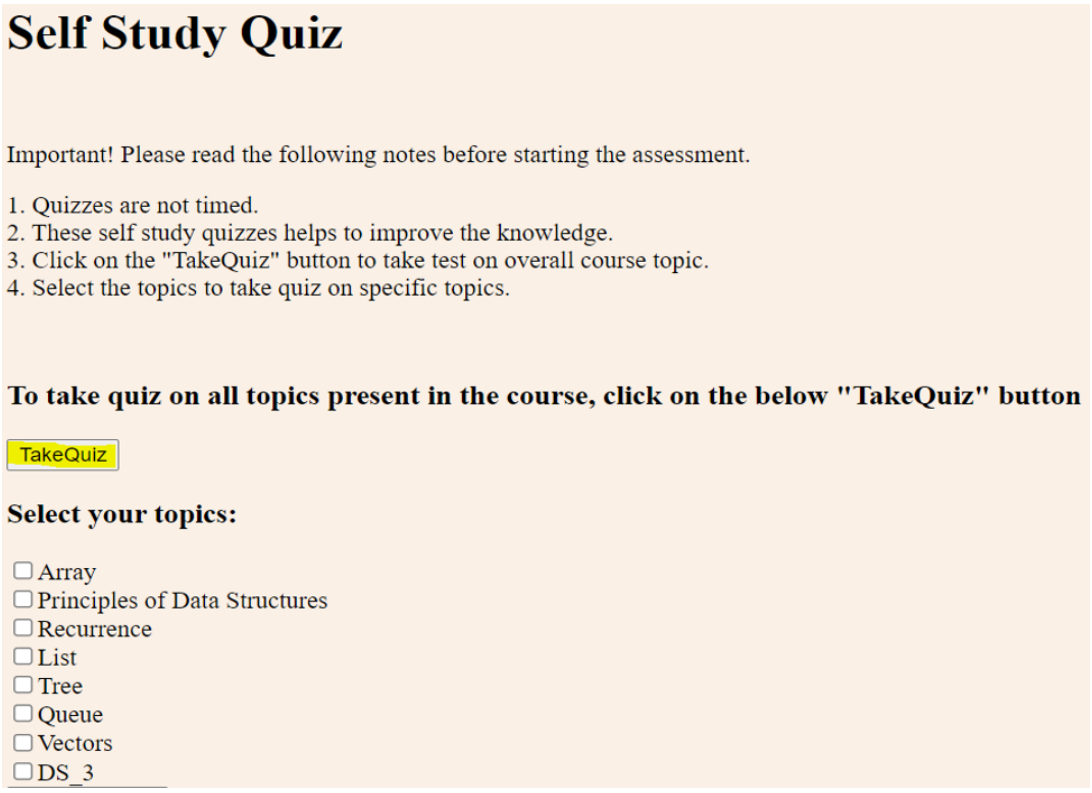


Figure 3.22: Self Study Quiz (TakeQuiz) - All course topics

Twenty questions are selected randomly from the self-assessment question bank for the topic and presented to the student. Student can take quizzes any number of times and does not have any time limit. These quiz questions are not given any weightage for the overall quiz/course grade. The student needs to click on the “Submit” button to submit the quiz.

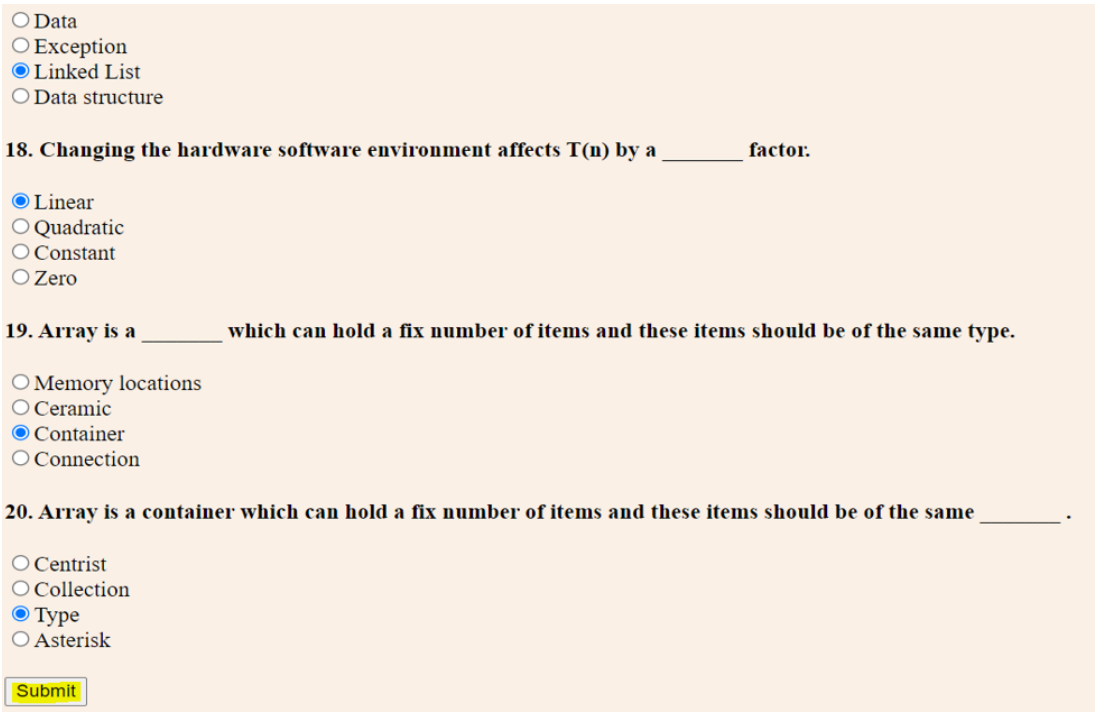


Figure 3.23: Self Study Quiz questions (All topics)

The student will immediately receive the score and can review the quiz questions. There is no negative marking in the quiz. CrsMgr shows the student's answer and the correct answer for the given question in the quiz.

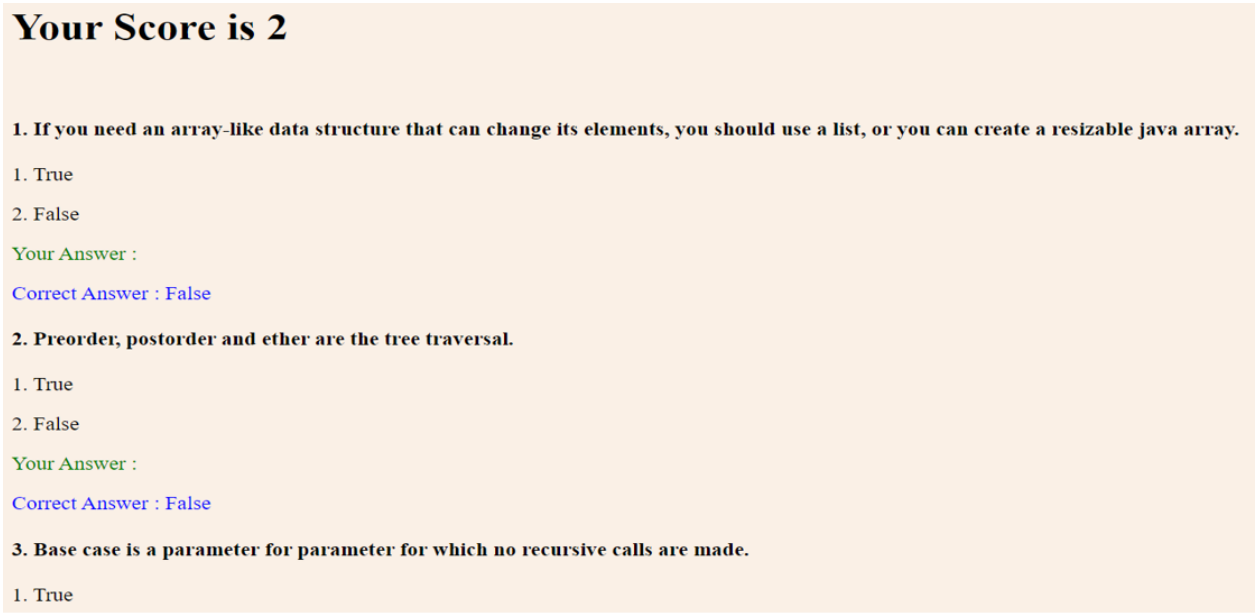


Figure 3.24: Self Study Quiz Result

Students can re-take a self-study quiz on any topic more than once. Each of these self-assessments has been randomly selected from the database. To re-take a quiz, student needs to click on the "Retake Quiz" button.

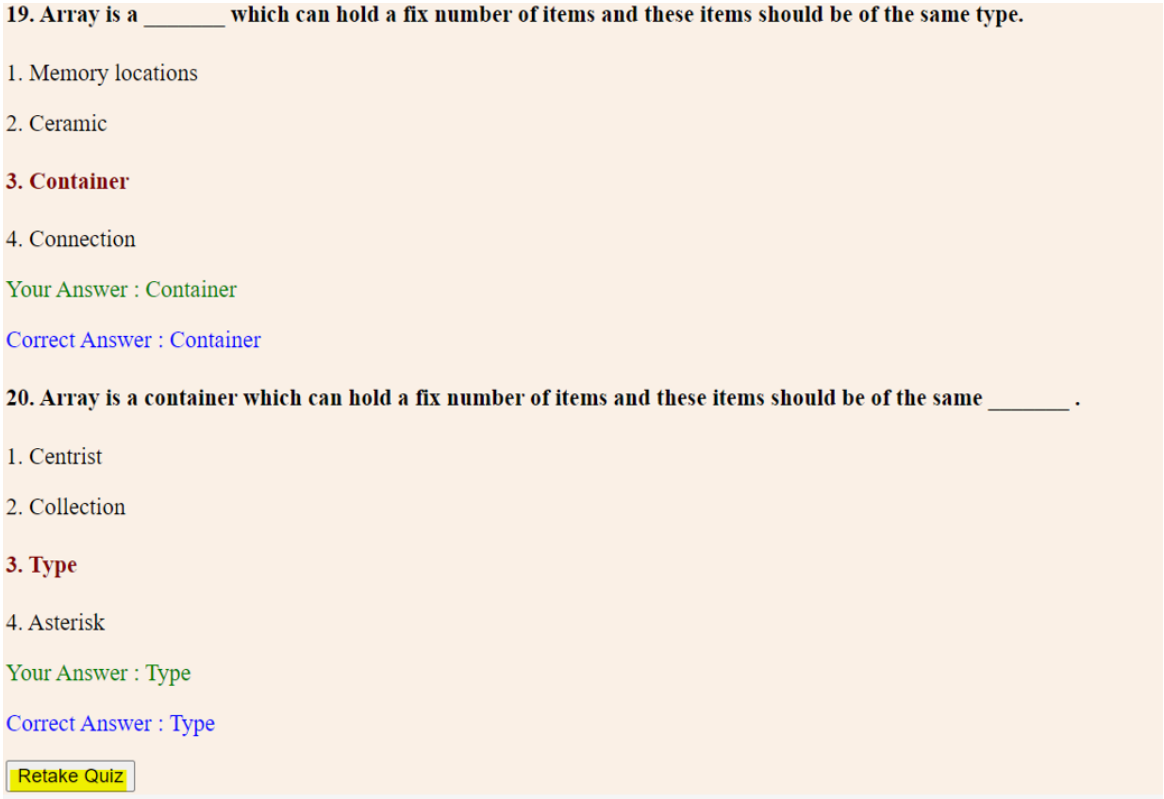


Figure 3.25: CrsMgr has options for students to retake self-study quiz

CrsMgr logs the self-study quiz in the 'self_study_log' table. Below figure 3.26 illustrates the self-study log entries. The log stores user_id, randomly picked question id lists, score, and quiz start and end date and time.

login.encs.concordia.ca - PuTTY

```
MariaDB [crsmgrtwnt]> select * from self_study_log;
```

<table>
<tr><th>ss_quiz_id</th><th>user_id</th><th>quest_ids</th><th>score</th><th>quiz_start_time</th><th>quiz_end_time</th></tr>
<tr><td>1</td><td>2013</td><td>|1348|1346|1363|1376|1321|1316|1327|1361|1323|1373|</td><td>0</td><td>2022-02-07 20:00:29</td><td>2022-02-07 20:01:00</td></tr>
<tr><td>2</td><td>2013</td><td>|1506|1463|1503|1494|1496|1501|1509|1499|1502|1454|</td><td>0</td><td>2022-02-07 20:01:21</td><td>2022-02-07 20:01:39</td></tr>
<tr><td>3</td><td>2013</td><td>|1147|966|985|963|1013|1150|1145|976|1148|1146|</td><td>1</td><td>2022-02-07 23:22:39</td><td>2022-02-07 23:22:44</td></tr>
<tr><td>4</td><td>1964</td><td>|972|991|1022|1145|989|998|1146|1148|1147|1154|</td><td>9</td><td>2022-02-09 09:55:48</td><td>2022-02-09 09:58:46</td></tr>
<tr><td>5</td><td>1964</td><td>|999|989|1149|1016|996|1003|1148|1145|1147|1146|</td><td>10</td><td>2022-02-09 09:59:06</td><td>2022-02-09 10:00:22</td></tr>
<tr><td>6</td><td>1964</td><td>|997|1152|1153|992|976|958|1148|1146|996|1149|</td><td>9</td><td>2022-02-09 10:00:26</td><td>2022-02-09 10:01:49</td></tr>
<tr><td>7</td><td>1964</td><td>|1122|1113|1129|1134|1141|1196|1140|1176|1194|1139|</td><td>7</td><td>2022-02-09 10:02:06</td><td>2022-02-09 10:04:44</td></tr>
<tr><td>8</td><td>1964</td><td>|1136|1124|1173|1082|1196|1109|1177|1143|1190|1059|</td><td>6</td><td>2022-02-09 10:06:10</td><td>2022-02-09 10:07:42</td></tr>
<tr><td>9</td><td>1964</td><td>|1305|1250|1255|1208|1210|1233|1297|1294|1295|1262|</td><td>7</td><td>2022-02-09 10:08:58</td><td>2022-02-09 10:11:44</td></tr>
<tr><td>10</td><td>1964</td><td>|1257|1231|1250|1219|1246|1285|1298|1244|1262|1310|</td><td>6</td><td>2022-02-09 10:13:23</td><td>2022-02-09 10:15:09</td></tr>
<tr><td>11</td><td>1964</td><td>|1305|1210|1267|1258|1226|1249|1296|1262|1238|1212|</td><td>7</td><td>2022-02-09 10:16:46</td><td>2022-02-09 10:20:00</td></tr>
<tr><td>12</td><td>1964</td><td>|1329|1385|1327|1338|1319|1373|1331|1359|1356|1361|</td><td>9</td><td>2022-02-09 10:20:20</td><td>2022-02-09 10:23:26</td></tr>
<tr><td>13</td><td>1964</td><td>|1382|1358|1331|1375|1361|1347|1350|1365|1323|1348|</td><td>9</td><td>2022-02-09 10:23:42</td><td>2022-02-09 10:25:23</td></tr>
<tr><td>14</td><td>1964</td><td>|1358|1359|1377|1390|1347|1334|1376|1327|1388|1363|</td><td>10</td><td>2022-02-09 10:26:11</td><td>2022-02-09 10:27:37</td></tr>
<tr><td>15</td><td>1964</td><td>|1356|1337|1368|1360|1327|1317|1349|1344|1312|1359|</td><td>10</td><td>2022-02-09 10:27:41</td><td>2022-02-09 10:28:43</td></tr>
<tr><td>16</td><td>1964</td><td>|1343|1356|1364|1318|1349|1348|1376|1321|1367|1346|</td><td>10</td><td>2022-02-09 10:28:48</td><td>2022-02-09 10:30:16</td></tr>
<tr><td>17</td><td>1964</td><td>|1359|1382|1340|1358|1351|1353|1335|1339|1327|1332|</td><td>9</td><td>2022-02-09 10:30:19</td><td>2022-02-09 10:31:57</td></tr>
<tr><td>18</td><td>1964</td><td>|1356|1312|1350|1329|1319|1348|1331|1316|1340|1381|</td><td>8</td><td>2022-02-09 10:32:12</td><td>2022-02-09 10:33:25</td></tr>
<tr><td>19</td><td>1964</td><td>|1250|1020|962|1084|1571|1236|969|1527|1563|1363|1542|1404|1382|1179|1627|1233|1135|1540|1040|1144|</td><td>13</td><td>2022-02-09 10:34:00</td><td>2022-02-09 10:37:43</td></tr>
<tr><td>20</td><td>2005</td><td>|1385|1338|1356|1343|1359|1366|1342|1355|1367|1349|</td><td>9</td><td>2022-02-09 16:19:37</td><td>2022-02-09 16:33:56</td></tr>
<tr><td>21</td><td>1968</td><td>|973|1151|1154|1148|991|995|1153|998|1149|978|</td><td>8</td><td>2022-02-09 17:04:29</td><td>2022-02-09 17:06:29</td></tr>
<tr><td>22</td><td>1968</td><td>|1085|1043|1177|1044|1184|1179|1103|1198|1083|1197|</td><td>7</td><td>2022-02-09 17:06:56</td><td>2022-02-09 17:09:33</td></tr>
<tr><td>23</td><td>1968</td><td>|1286|1216|1244|1233|1205|1225|1262|1261|1294|1296|</td><td>6</td><td>2022-02-09 17:09:58</td><td>2022-02-09 17:12:51</td></tr>
<tr><td>24</td><td>2005</td><td>|1402|1397|1410|1414|1408|1440|1416|1417|1421|1429|</td><td>6</td><td>2022-02-09 17:11:14</td><td>2022-02-09 17:15:10</td></tr>
<tr><td>25</td><td>1968</td><td>|1390|1356|1331|1329|1335|1342|1358|1348|1388|1357|</td><td>9</td><td>2022-02-09 17:13:16</td><td>2022-02-09 17:16:36</td></tr>
<tr><td>26</td><td>2005</td><td>|1365|1327|1356|1347|1340|1346|1381|1350|1370|1387|</td><td>7</td><td>2022-02-09 17:30:06</td><td>2022-02-09 17:34:32</td></tr>
<tr><td>27</td><td>2005</td><td>|1350|1363|1346|1356|1315|1382|1340|1380|1327|1319|</td><td>7</td><td>2022-02-09 17:45:08</td><td>2022-02-09 17:47:23</td></tr>
<tr><td>28</td><td>1945</td><td>|1344|1337|1376|1390|1356|1332|1365|1359|1386|1387|</td><td>4</td><td>2022-02-09 17:51:31</td><td>2022-02-09 17:53:20</td></tr>
<tr><td>29</td><td>1951</td><td>|1366|1307|1362|1325|1306|1204|1359|1288|1292|1354|</td><td>8</td><td>2022-02-09 18:36:04</td><td>2022-02-09 18:39:57</td></tr>
<tr><td>30</td><td>1945</td><td>|1359|1341|1327|1368|1318|1348|1349|1390|1360|1350|</td><td>7</td><td>2022-02-09 18:38:39</td><td>2022-02-09 18:40:37</td></tr>
<tr><td>31</td><td>1951</td><td>|1361|1340|1382|1358|1377|1332|1364|1389|1335|1349|</td><td>7</td><td>2022-02-09 18:40:36</td><td>2022-02-09 18:44:44</td></tr>
<tr><td>32</td><td>1945</td><td>|1207|1227|1208|1242|1206|1309|1236|1308|1211|1298|</td><td>10</td><td>2022-02-09 18:45:31</td><td>2022-02-09 18:47:22</td></tr>
<tr><td>33</td><td>1945</td><td>|1216|1284|1257|1272|1234|1323|1322|1291|1249|1364|</td><td>0</td><td>2022-02-09 19:52:26</td><td>2022-02-09 19:52:29</td></tr>
<tr><td>34</td><td>2005</td><td>|1227|1253|1249|1264|1203|1232|1291|1282|1259|1294|</td><td>10</td><td>2022-02-09 19:50:23</td><td>2022-02-09 19:59:15</td></tr>
</table>

Figure 3.26: Self Study Quiz logs

3.3.4.2 Self Study Quiz based on Topic

Students can select any specific topic to take a self-study quiz from the topic lists in the CrsMgr. The student chooses the topic by selecting the checkbox as illustrated in figure 3.27. Students can pick more than one topic for taking the self-assessment quiz.

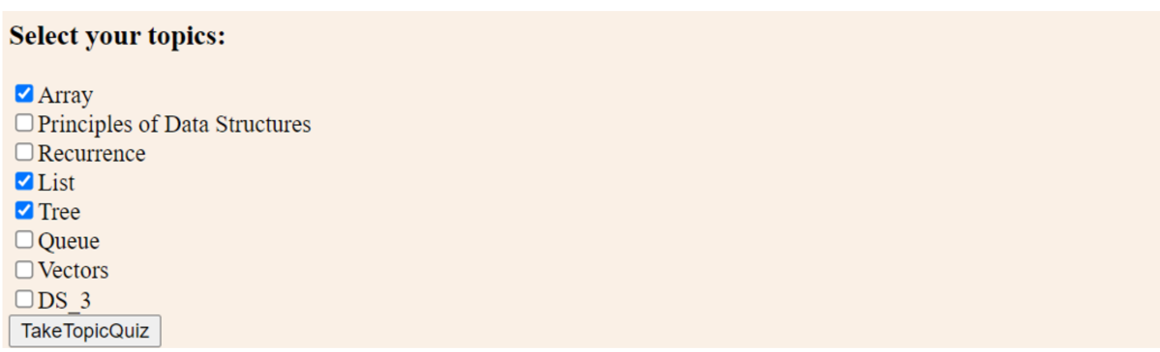


Figure 3.27: Self Study Quiz based on selected topics

The questions based on the topic are selected from the SAQ question bank for the picked topic(s). These self-study quizzes are not timed and do not have any weightage on course marks. Students can submit the quiz and review the quiz.

Figure 3.28: Self Study Quiz based on topics result page

Students can re-take the same topic(s) questions by clicking the “Retake Quiz” button as highlighted in the below screenshot. The questions will be from the same topic(s) that the student has selected before. If a student wants to pick a different topic for the quiz, then the student needs to click the “Self-Study Quiz” hyperlink present in the main menu.

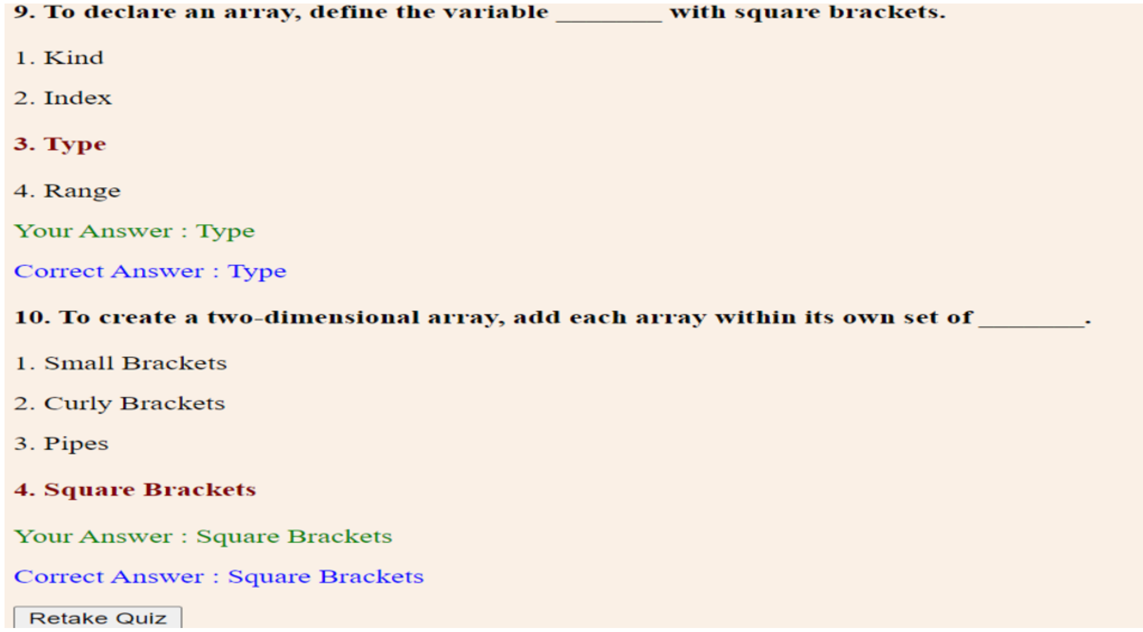
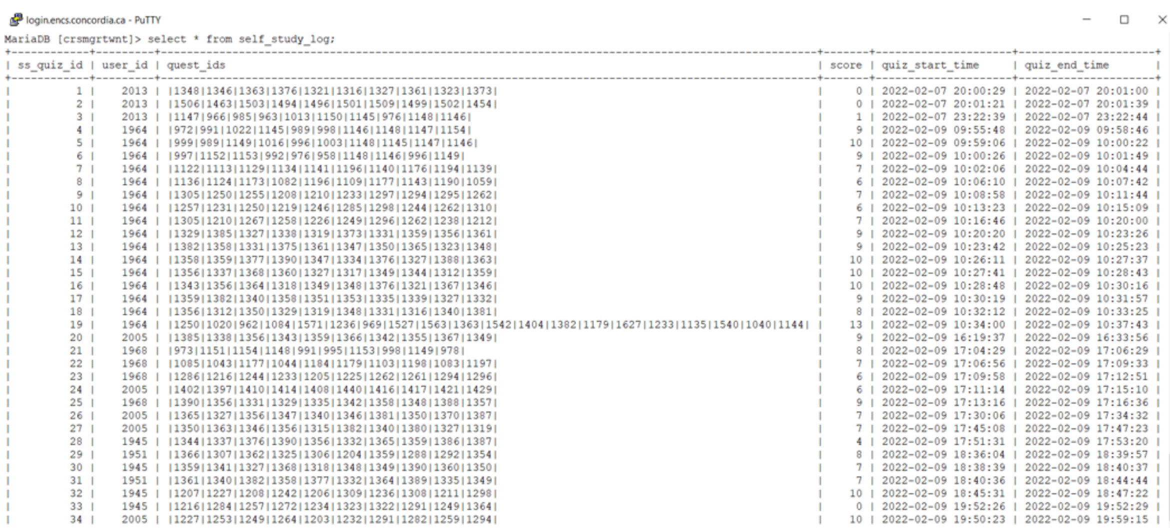


Figure 3.29: Retake Self Study Quiz based on selected topics

CrsMgr logs the self-study quiz in the ‘self_study_log’ table. Below figure 3.30 illustrates the self-study log entries. The log stores the user_id, a list of randomly picked question ids, score, and quiz start and end date and time.



login.encs.concordia.ca - PuTTY

```
MariaDB [crsmgrtwnt]> select * from self_study_log;
+------------+---------+-------------------------------------------------------------------------------------------------+-------+---------------------+---------------------+
| ss_quiz_id | user_id | quest_ids                                                                                       | score | quiz_start_time     | quiz_end_time       |
+------------+---------+-------------------------------------------------------------------------------------------------+-------+---------------------+---------------------+
|          1 |    2013 | |1348|1346|1363|1376|1321|1316|1327|1361|1323|1373|                                             |     0 | 2022-02-07 20:00:29 | 2022-02-07 20:01:00 |
|          2 |    2013 | |1506|1463|1503|1494|1496|1501|1509|1499|1502|1454|                                             |     0 | 2022-02-07 20:01:21 | 2022-02-07 20:01:39 |
|          3 |    2013 | |1147|966|985|963|1013|1150|1145|976|1148|1146|                                                 |     1 | 2022-02-07 23:22:39 | 2022-02-07 23:22:44 |
|          4 |    1964 | |972|991|1022|1145|989|998|1146|1148|1147|1154|                                                 |     9 | 2022-02-09 09:55:48 | 2022-02-09 09:58:46 |
|          5 |    1964 | |999|989|1149|1016|996|1003|1148|1145|1147|1146|                                                |    10 | 2022-02-09 09:59:06 | 2022-02-09 10:00:22 |
|          6 |    1964 | |997|1152|1153|992|976|958|1148|1146|996|1149|                                                  |     9 | 2022-02-09 10:00:26 | 2022-02-09 10:01:49 |
|          7 |    1964 | |1122|1113|1129|1134|1141|1196|1140|1176|1194|1139|                                             |     7 | 2022-02-09 10:02:06 | 2022-02-09 10:04:44 |
|          8 |    1964 | |1136|1124|1173|1082|1196|1109|1177|1143|1190|1059|                                             |     6 | 2022-02-09 10:06:10 | 2022-02-09 10:07:42 |
|          9 |    1964 | |1305|1250|1255|1208|1210|1233|1297|1294|1295|1262|                                             |     7 | 2022-02-09 10:08:58 | 2022-02-09 10:11:44 |
|         10 |    1964 | |1257|1231|1250|1219|1246|1285|1298|1244|1262|1310|                                             |     6 | 2022-02-09 10:13:23 | 2022-02-09 10:15:09 |
|         11 |    1964 | |1305|1210|1267|1258|1226|1249|1296|1262|1238|1212|                                             |     7 | 2022-02-09 10:16:46 | 2022-02-09 10:20:00 |
|         12 |    1964 | |1329|1385|1327|1338|1319|1373|1331|1359|1356|1361|                                             |     9 | 2022-02-09 10:20:20 | 2022-02-09 10:23:26 |
|         13 |    1964 | |1382|1358|1331|1375|1361|1347|1350|1365|1323|1348|                                             |     9 | 2022-02-09 10:23:42 | 2022-02-09 10:25:23 |
|         14 |    1964 | |1358|1359|1377|1390|1347|1334|1376|1327|1388|1363|                                             |    10 | 2022-02-09 10:26:11 | 2022-02-09 10:27:37 |
|         15 |    1964 | |1356|1337|1368|1360|1327|1317|1349|1344|1312|1359|                                             |    10 | 2022-02-09 10:27:41 | 2022-02-09 10:28:43 |
|         16 |    1964 | |1343|1356|1364|1318|1349|1348|1376|1321|1367|1346|                                             |    10 | 2022-02-09 10:28:48 | 2022-02-09 10:30:16 |
|         17 |    1964 | |1359|1382|1340|1358|1351|1353|1335|1339|1327|1332|                                             |     9 | 2022-02-09 10:30:19 | 2022-02-09 10:31:57 |
|         18 |    1964 | |1356|1312|1350|1329|1319|1348|1331|1316|1340|1381|                                             |     8 | 2022-02-09 10:32:12 | 2022-02-09 10:33:25 |
|         19 |    1964 | |1250|1020|962|1084|1571|1236|969|1527|1563|1363|1542|1404|1382|1179|1627|1233|1135|1540|1040|1144| |    13 | 2022-02-09 10:34:00 | 2022-02-09 10:37:43 |
|         20 |    2005 | |1385|1338|1356|1343|1359|1366|1342|1355|1367|1349|                                             |     9 | 2022-02-09 16:19:37 | 2022-02-09 16:33:56 |
|         21 |    1968 | |973|1151|1154|1148|991|995|1153|998|1149|978|                                                  |     8 | 2022-02-09 17:04:29 | 2022-02-09 17:06:29 |
|         22 |    1968 | |1085|1043|1177|1044|1184|1179|1103|1198|1083|1197|                                             |     7 | 2022-02-09 17:06:56 | 2022-02-09 17:09:33 |
|         23 |    1968 | |1286|1216|1244|1233|1205|1225|1262|1261|1294|1296|                                             |     6 | 2022-02-09 17:09:58 | 2022-02-09 17:12:51 |
|         24 |    2005 | |1402|1397|1410|1414|1408|1440|1416|1417|1421|1429|                                             |     6 | 2022-02-09 17:11:14 | 2022-02-09 17:15:10 |
|         25 |    1968 | |1390|1356|1331|1329|1335|1342|1358|1348|1388|1357|                                             |     9 | 2022-02-09 17:13:16 | 2022-02-09 17:16:36 |
|         26 |    2005 | |1365|1327|1356|1347|1340|1346|1381|1350|1370|1387|                                             |     7 | 2022-02-09 17:30:06 | 2022-02-09 17:34:32 |
|         27 |    2005 | |1350|1363|1346|1356|1315|1382|1340|1380|1327|1319|                                             |     7 | 2022-02-09 17:45:08 | 2022-02-09 17:47:23 |
|         28 |    1945 | |1344|1337|1376|1390|1356|1332|1365|1359|1386|1387|                                             |     4 | 2022-02-09 17:51:31 | 2022-02-09 17:53:20 |
|         29 |    1951 | |1366|1307|1362|1325|1306|1204|1359|1288|1292|1354|                                             |     8 | 2022-02-09 18:36:04 | 2022-02-09 18:39:57 |
|         30 |    1945 | |1359|1341|1327|1368|1318|1348|1349|1390|1360|1350|                                             |     7 | 2022-02-09 18:38:39 | 2022-02-09 18:40:37 |
|         31 |    1951 | |1361|1340|1382|1358|1377|1332|1364|1389|1335|1349|                                             |     7 | 2022-02-09 18:40:36 | 2022-02-09 18:44:44 |
|         32 |    1945 | |1207|1227|1208|1242|1206|1309|1236|1308|1211|1298|                                             |    10 | 2022-02-09 18:45:31 | 2022-02-09 18:47:22 |
|         33 |    1945 | |1216|1284|1257|1272|1234|1323|1322|1291|1249|1364|                                             |     0 | 2022-02-09 19:52:26 | 2022-02-09 19:52:29 |
|         34 |    2005 | |1227|1253|1249|1264|1203|1232|1291|1282|1259|1294|                                             |    10 | 2022-02-09 19:50:23 | 2022-02-09 19:59:15 |
+------------+---------+-------------------------------------------------------------------------------------------------+-------+---------------------+---------------------+
```

Figure 3.30: Self Study Quiz logs

Chapter 4: Software Testing

This chapter presents the tests conducted on CrsMgr. The test was conducted on Windows 10 and Linux (https://ideas.encs.concordia.ca/IDSCrsMgr) machines. Unit testing and Integration testing were carried out.

Following testing methodologies were carried out:

1. Unit test involves testing of new functions added to generate questions.
2. Integration testing involves testing the frontend and backend of the application.

4.1 Unit Testing

The developer executes the unit testing. In this type, the individual components of the application are tested and validated that it performs as expected. Unit testing was carried out for each component manually.

4.1.1 Backend Program Testing

In this web application, the Python program is used to generate the quiz questions. To perform this testing python program was tested manually in the command prompt.

4.1.1.1 Python Program to generate questions from user text

Figure 4.1 illustrates the manual testing of generating questions from user text. ge_questions.py python file generates both True/False and Fill In the Blanks with Multiple-Choice options questions.

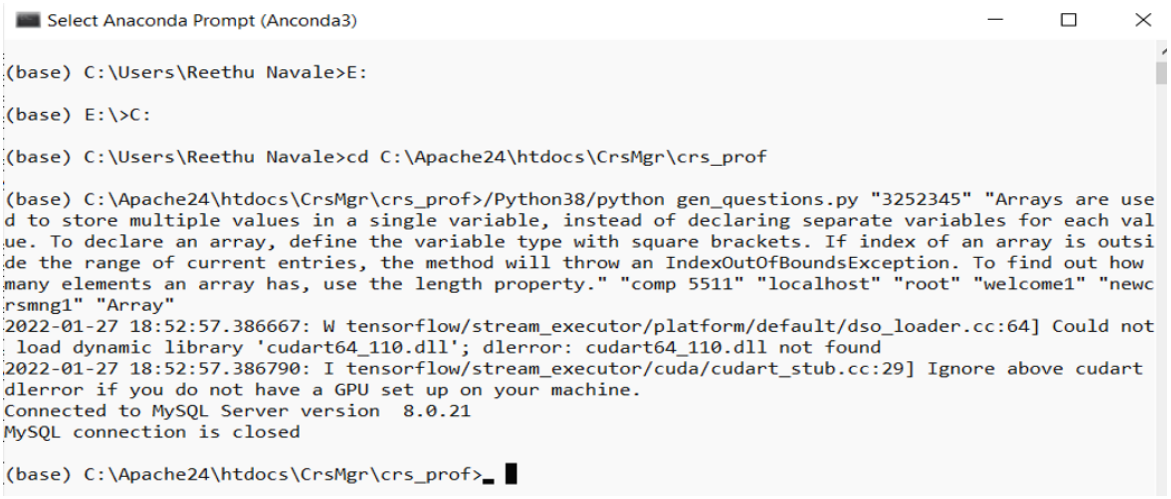


Figure 4.1: Unit Testing of backend program - gen_questions.py

Python program, gen_questions.py is executed in the back to generate quiz questions from the user text. 8 parameters are passed to this program. Session ID, User Text, Course Name, database host, database username, database password, database name, topic name in the corresponding sequence is passed as the parameters.

The generated quiz questions will be stored in the temp_question table.

Figure 4.2: Questions generated from user text stored in temp_question table in database

4.1.1.2 Python Program to generate questions from PDF

Figure 4.3 illustrates the manual testing of generating questions from a PDF file. gen_questions_file.py python file generates both True/False and Fill In the Blanks with Multiple-Choice options questions.

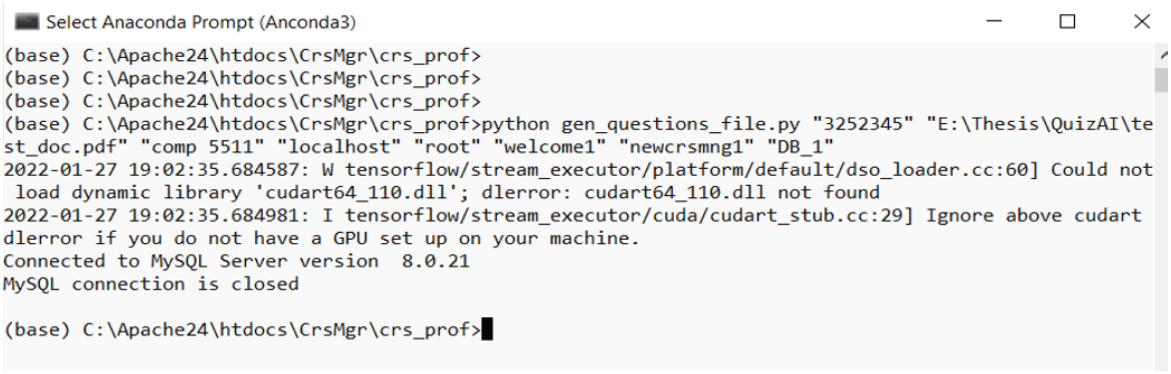


Figure 4.3: Unit testing of background program - gen_questions_file.py

Python program, gen_questions_file.py is executed in the back to generate quiz questions from the PDF file. 8 parameters are passed to this program. Session ID, PDF File Path, Course Name, database host, database username, database password, database name, topic name in the corresponding sequence is passed as the parameters.

The generated quiz questions will be stored in the temp_question table.

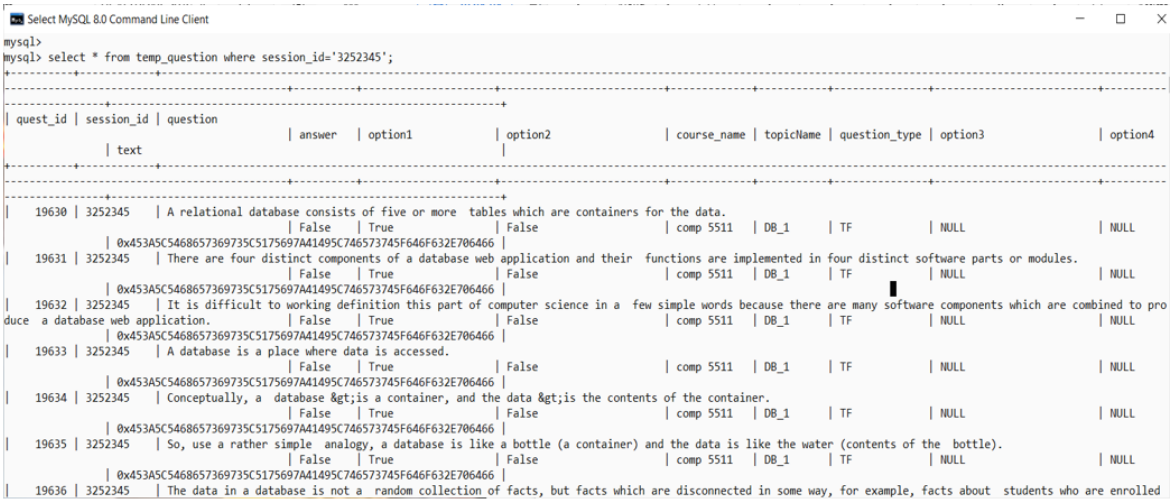


Figure 4.4: Questions generated from PDF stored in temp_question table in database

4.1.2 Frontend Program

In this web application, PHP is used for frontend development. Manual testing is done to validate the frontend program.

4.1.2.1 Coordinator/Instructor has the access to generate questions

There are various roles that the CrsMgr have, namely System Administrator, Department Administrator, Course Coordinator, Course Instructor, Course Student, Course Marker, Course Tutor, Lab Tutor, Thesis Supervisor, and Thesis Graduate Student. Only the Course Coordinator/Course Instructor has access to generate quiz questions. Other user roles should not have access to generate quiz questions.

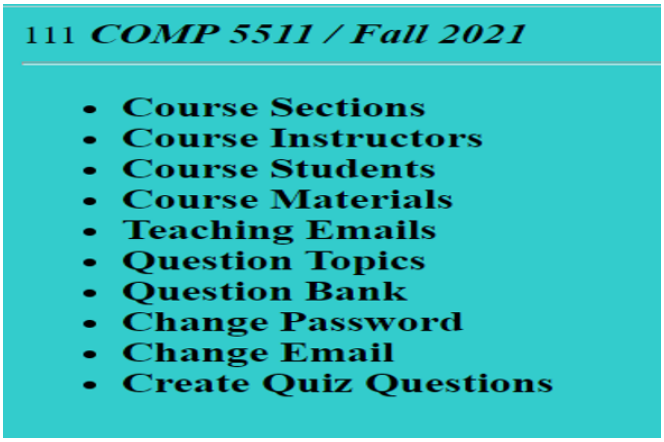


Figure 4.5: Course coordinator options

Below figure 4.6 shows the options for Course Instructor present in CrsMgr.

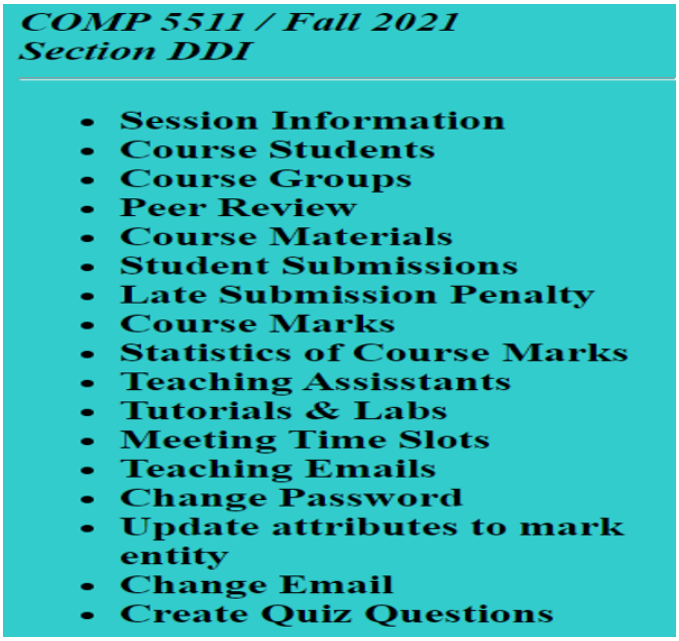


Figure 4.6: Course Instructor options

Verifying if other CrsMgr roles do not have access to generate questions. The below figure 4.7 shows the options available for the Thesis Supervisor role user.

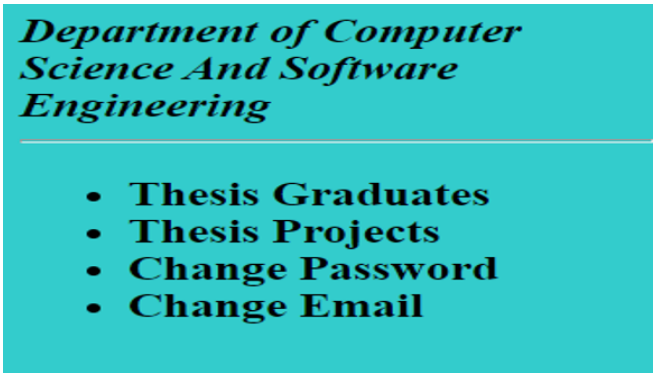


Figure 4.7: Thesis Supervisor Menu

4.1.2.2 Student has the access to take quiz

Only active students who have enrolled in the course have access to take the quiz. The quiz topics are based on the enrolled course. The student should not get the quiz topic that is not part of the enrolled course.

The below figure 4.8 shows the test user 1994 is enrolled in course COMP 5511 Fall 2021. COMP 5511 course is a data structure course at Concordia University. The topics related to this course should be available for the student.

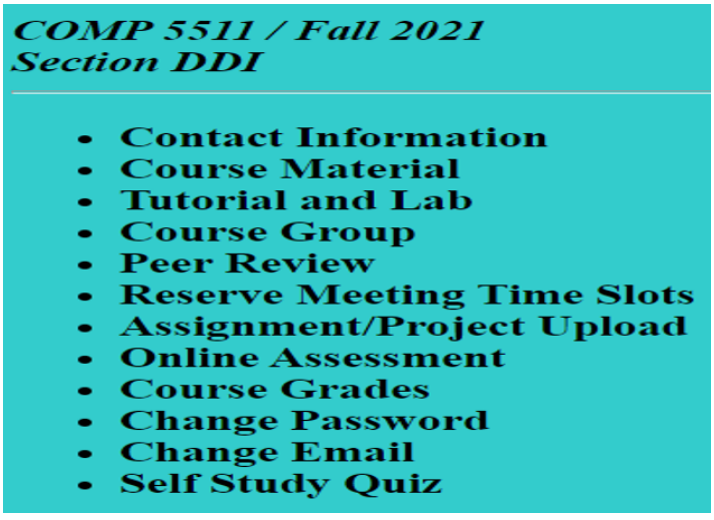


Figure 4.8: Student menu for self-study quiz

Below, figure 4.9 shows the list of topics that the course instructor/coordinator has created for the enrolled students.

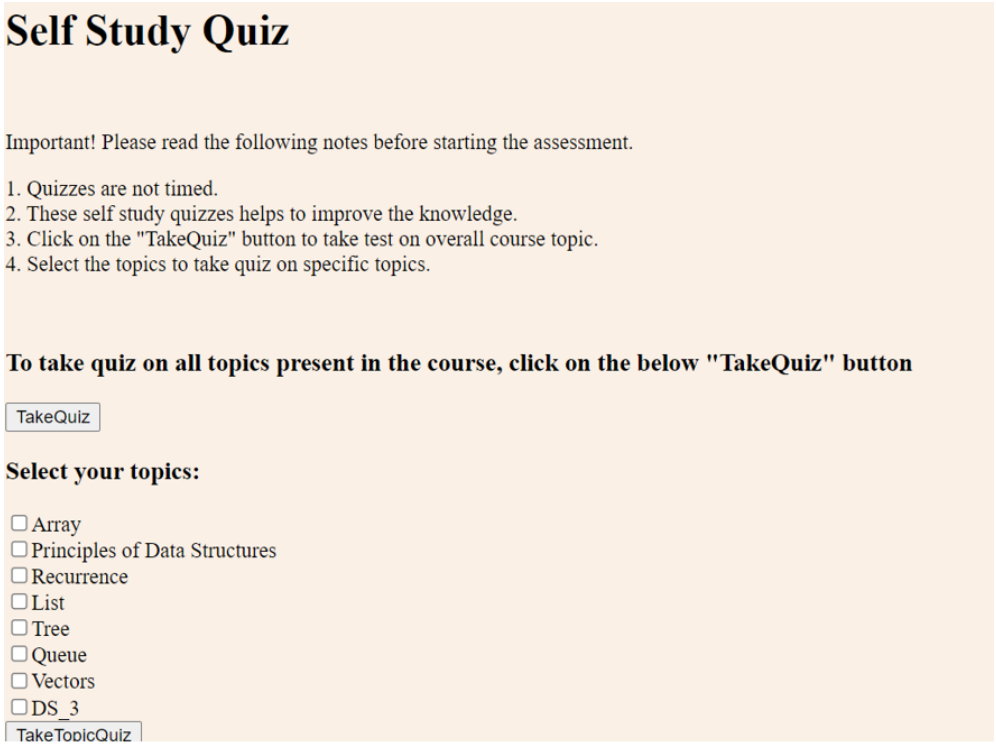


Figure 4.9: Self Study Quiz Topic List available for Students

Verifying if the topics mentioned above in figure 4.9 belong to COMP 5511 course in the database and manually validating.

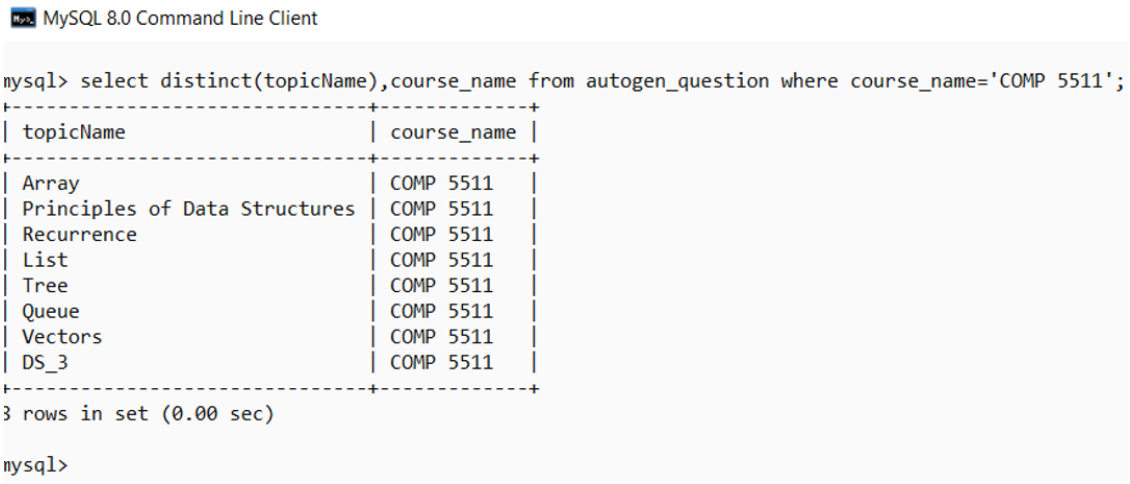


Figure 4.10: Verification of topics in database

Manually verified the topic list in the database and the list displayed in the CrsMgr are the same for the course COMP 5511. Similar testing was done for other courses in CrsMgr. The CrsMgr is correctly displaying the topics with respect to the course that the student is enrolled in.

4.2 Integration Testing

In integration testing, testing is carried out on the integrated modules i.e., the frontend and backend of the applications are connected as a group. The main purpose of this testing is to validate the interaction between the modules is working as expected. This section illustrates the integration testing done on the user scenarios.

Following are the user scenarios for this thesis:

a. Course coordinator has the access to generate questions in CrsMgr

b. Course coordinator can add new question

c. Course coordinator can edit the question that are generated by the application

d. Course coordinator can delete the question that are generated by the application

e. Course coordinator can submit the generated questions by the application into the database

f. Student can take self-study quiz for the enrolled course

g. Student can take specific topic of the course enrolled

4.2.1 Course Coordinator generates the questions from user text

Only the course coordinator has access to generate the questions. The coordinator selects the "Create Quiz Questions" option from the menu. Enters the text and provides appropriate topic name and clicks on Submit button in the front end. The front-end application needs to fetch the database details from the config file and call the backend script (gen_questions.py) with the correct parameters. The python program accepts the following eight parameters in the same sequence: Session ID, User Text, Course Name, database host, database username, database password, database name, and topic name. The python program generates the questions and commits the questions in the temp_question table. Once the python program is executed the front-end application needs to display the questions that are generated by the python program. If the backend application fails to generate the questions, then the list should be empty and logged in the Apache server log.

Below screenshots illustrates the integration testing.

4.2.1.1 Testing for success scenario

Coordinator/Instructor logins to CrsMgr and clicks on the "Create Quiz Question" option for the course. Enters the text and selects the topic name and clicks the submit button to initiate the process.

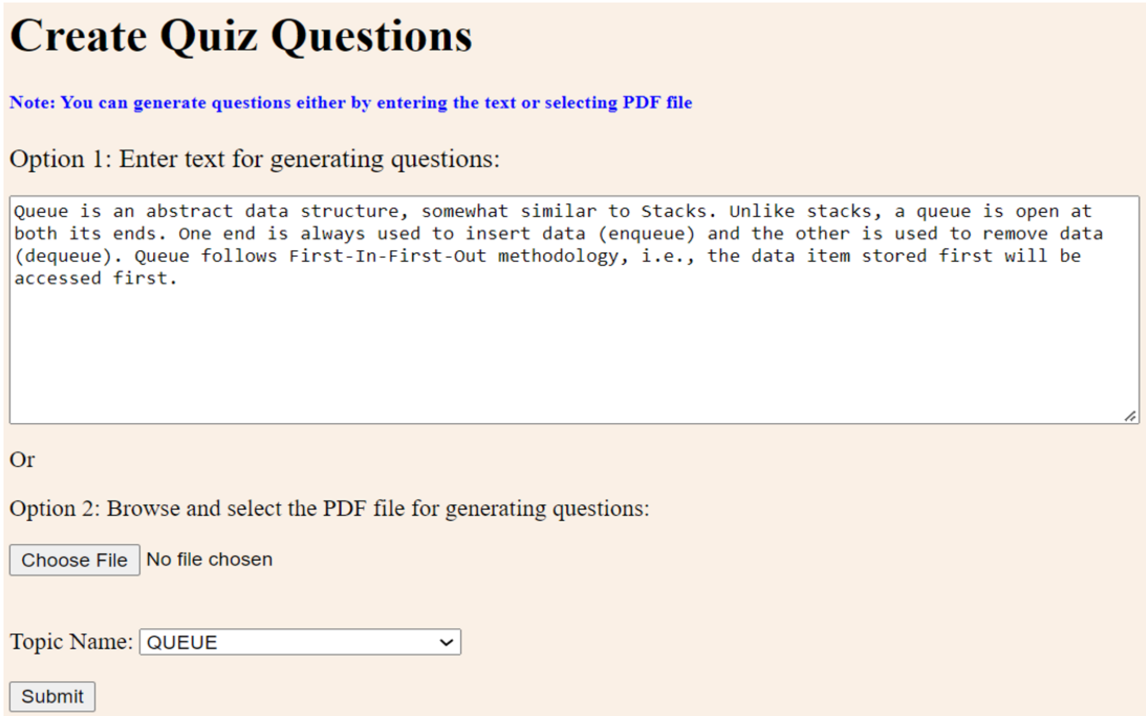
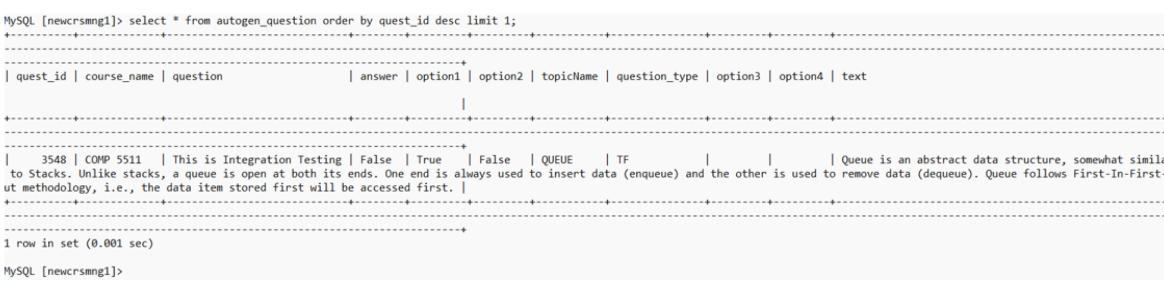


Figure 4.11: Testing of Instructor/Coordinator to generate questions from text

The application will generate the questions and commit them to the database. The generated questions will be committed in a temporary table. After the course, the instructor/coordinator manually validates the question list and commits it into the autogen_question table.



```
MySQL [newcrsmng1]> select * from autogen_question order by quest_id desc limit 1;
| quest_id | course_name | question                    | answer | option1 | option2 | topicName | question_type | option3 | option4 | text
                                                                 |
|     3548 | COMP 5511   | This is Integration Testing | False  | True    | False   | QUEUE     | TF            |         |         | Queue is an abstract data structure, somewhat similar
 to Stacks. Unlike stacks, a queue is open at both its ends. One end is always used to insert data (enqueue) and the other is used to remove data (dequeue). Queue follows First-In-First-O
ut methodology, i.e., the data item stored first will be accessed first. |
1 row in set (0.001 sec)

MySQL [newcrsmng1]>
```

Figure 4.12: Testing of generated questions from CrsMgr that are stored in database

The questions that are generated from the current session need to be displayed in the front-end of CrsMgr.

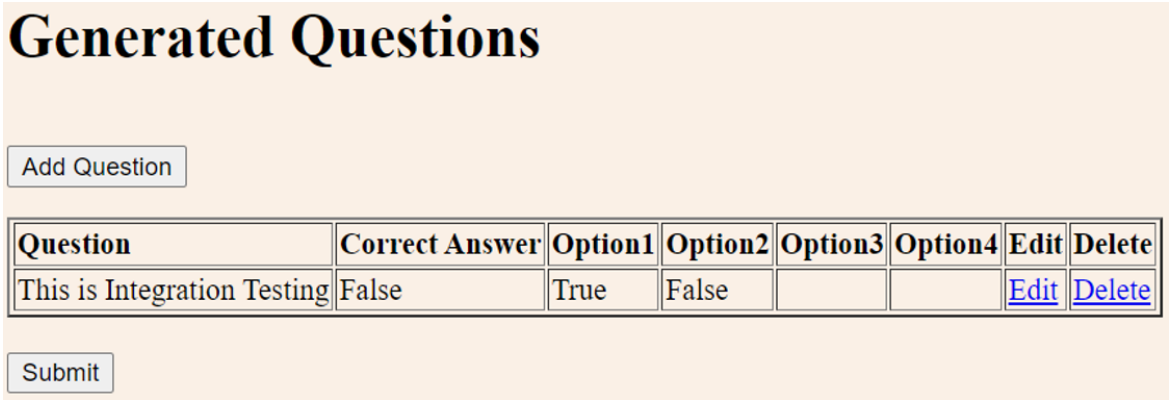


Figure 4.13: Question list in CrsMgr

4.2.1.2 Testing Negative scenario

If the coordinator does not give any text or if there are any failure in the python program, then the error message is displayed.

Below figure 4.14 shows the error message. The error will be logged in the Apache log table.

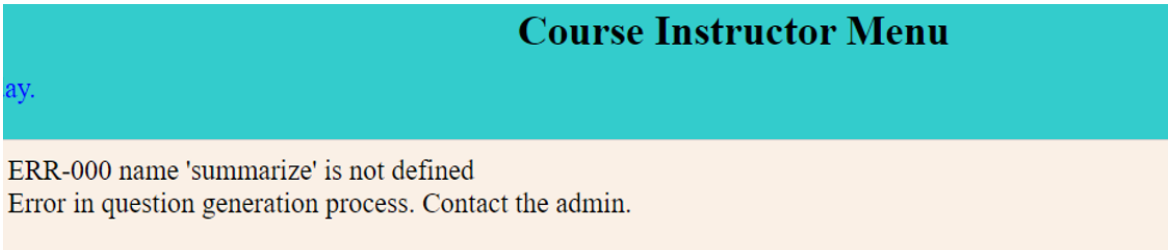


Figure 4.14: Error Message is displayed on error

Below figure 4.15 of Apache error logs.

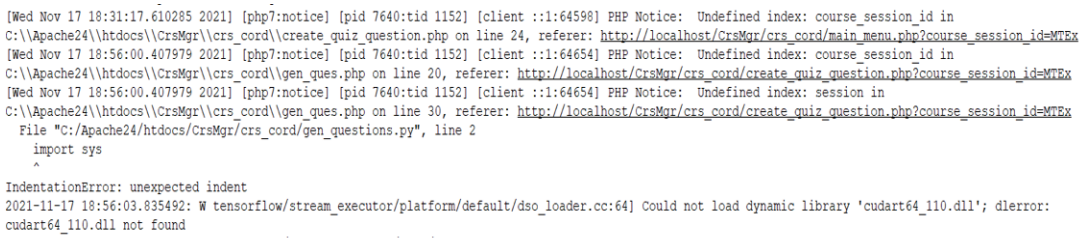


Figure 4.15: Apache error log

4.2.2 Course Instructor generates the questions from PDF file

Only the course coordinator/course instructor has the access to generate the questions. The instructor selects the “Create Quiz Questions” option from the menu. Browse the file present in the local device and upload it with the appropriate topic name and click on Submit button. The front-end application needs to fetch the database details from the config file and call the backend script (gen_questions_file.py) with the correct parameters. The python program accepts the following 8 parameters in the same sequence: Session ID, File Path, Course Name, database host, database username, database password, database name, and topic name. The python program generates the questions and commits the questions in the temp_question table. Once the python program is executed the front-end application needs to display the questions that are generated by the python program. If the backend application fails to generate the questions, then the list should be empty and logged in the Apache server log.

Below screenshots illustrates the integration testing screenshots.

4.2.2.1 Testing for success scenario

Coordinator/Instructor logins to CrsMgr and clicks on the “Create Quiz Question” option for the course. Browse the file present in the local device.

Select the course PDF file that would be used to generate questions from the local device.

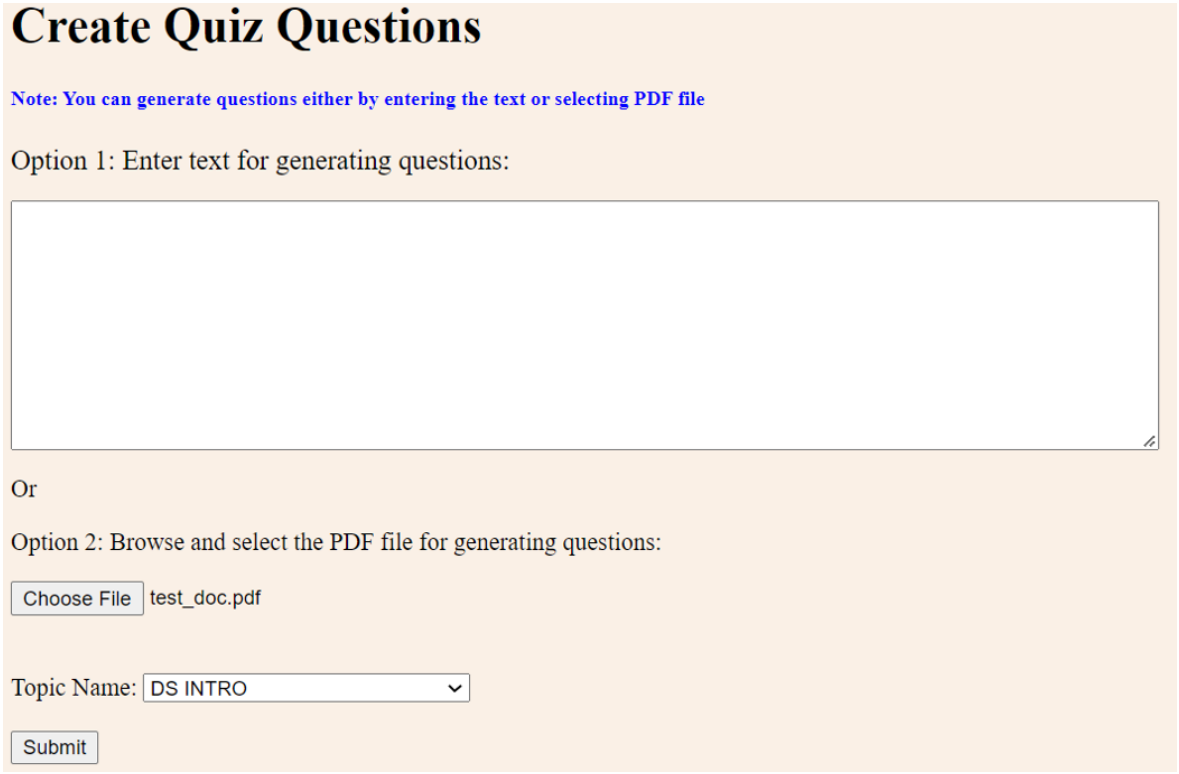


Figure 4.16: Testing generation of questions from PDF file

Questions that are generated from the session will be displayed in the CrsMgr.

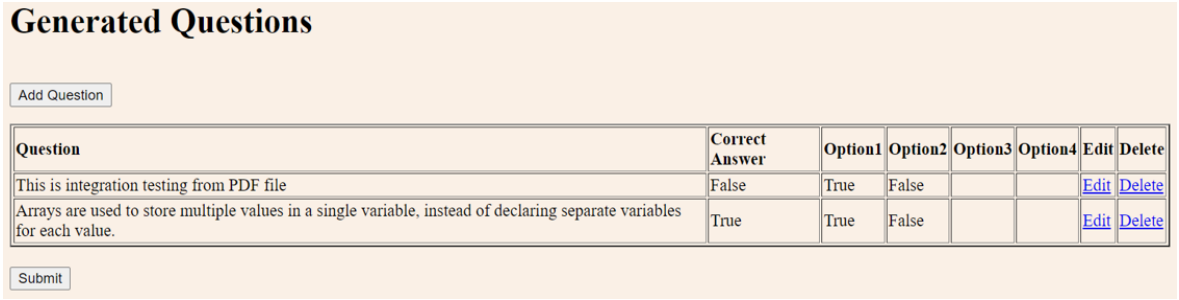


Figure 4.17: Testing of questions generated by PDF file

4.2.2.2 Testing Negative scenario

Coordinator/Instructor logins to CrsMgr and clicks on the “Create Quiz Question” option for the course. Browse the file present in the local device.

If the user selects a corrupt PDF file or if the PDF file does not have any text or if the user selects any non-PDF file, then the questions will not be generated and logged in Apache error log.

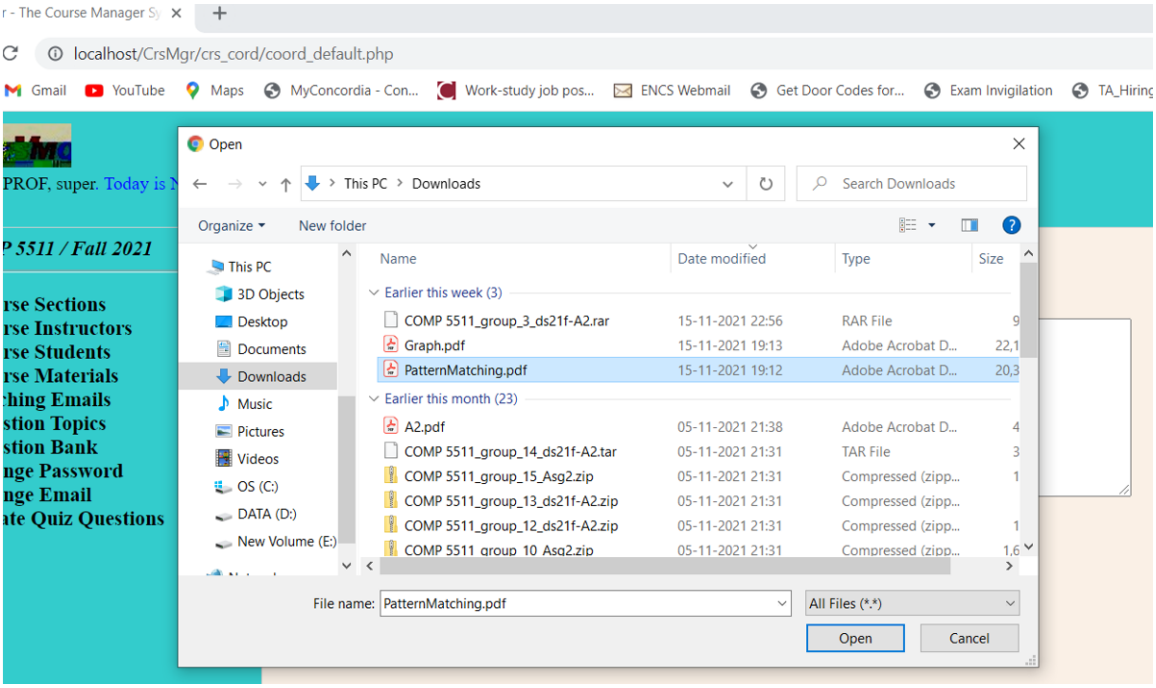


Figure 4.18: Testing the negative scenario with corrupt PDF file

The question generation failed, and error message is shown in frontend. Below figure 4.19 illustrates the scenario of CrsMgr displaying the error message.

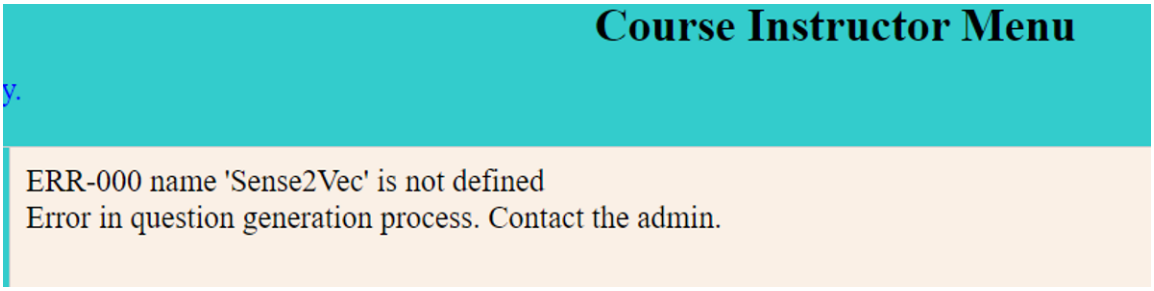


Figure 4.19: Error Message Displayed on Error Scenario

4.2.3 Course Coordinator/Instructor can add new question

The coordinator/instructor can add a question in CrsMgr. The coordinator has an option to add either True/False or MCQ type of question. The new question added by the coordinator is inserted into temp_question.

The front has a button named “Add Question”. This option is available once the questions are generated from the text or PDF file.

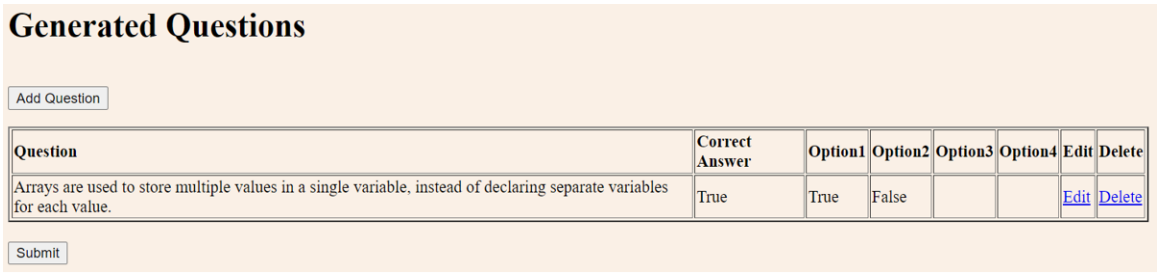


Figure 4.20: Testing of questions that are generated by the system

Once the instructor/coordinator clicks on the “Add Question” button, the CrsMgr should call “add_new_question.php” to accept the input from the coordinator.

Figure 4.21: Testing of CrsMgr having feature to add new question

After entering the question details the coordinator will click on the “Add” button. The CrsMgr will execute SQL commands to insert the question in “temp_question”. The below figure 4.22 shows the question inserted in the database.

```
mysql> select * from temp_question;
+----------+--------------+----------------------------------------------------------------------------------------------------------+--------+---------+---------+-------
------+-----------+--------------+---------+---------+-----------+
| quest_id | session_id   | question                                                                                                 | answer | option1 | option2 | course
_name | topicName | question_type | option3 | option4 | text      |
+----------+--------------+----------------------------------------------------------------------------------------------------------+--------+---------+---------+-------
------+-----------+--------------+---------+---------+-----------+
|    19913 | 61f34a9f5ac0e | Arrays are used to store multiple values in a single variable, instead of declaring separate variables for each value. | True   | True    | False   | COMP 5
511   | Array     | TF            |         |         | 0x        |
|    19914 | 61f34a9f5ac0e | testing the Add New question scenario                                                                   | True   | True    | False   | COMP 5
511   | Array     | TF            |         |         | 0x        |
+----------+--------------+----------------------------------------------------------------------------------------------------------+--------+---------+---------+-------
------+-----------+--------------+---------+---------+-----------+
2 rows in set (0.00 sec)

mysql>
```

Figure 4.22: Testing of questions in database

Coordinator/Instructor can check the inserted question in the CrsMgr question list page.

Generated Questions

Add Question

Question	Correct Answer	Option1	Option2	Option3	Option4	Edit	Delete
Arrays are used to store multiple values in a single variable, instead of declaring separate variables for each value.	True	True	False			Edit	Delete
testing the Add New question scenario	True	True	False			Edit	Delete

Submit

Figure 4.23: Testing of question list after adding new question

The CrsMgr will not allow the user to submit until the required information is entered. The question list present in the database and CrsMgr is the same. CrsMgr is correctly displaying the questions that are generated for the current session.

4.2.4 Course Coordinator?Instructor can edit the question

The coordinator/instructor can modify the generated question. The CrsMgr should provide access to the coordinator to edit the question. Each record of the question has a hyperlink to edit the question.

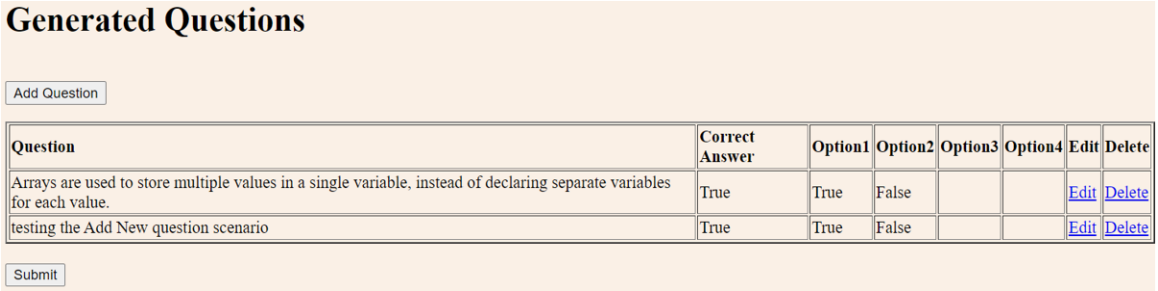


Figure 4.24: Testing of question list before update

When the coordinator clicks on the “Edit” hyperlink for the specific question, the editquestion.php web application is called. The coordinator will enter the question details.

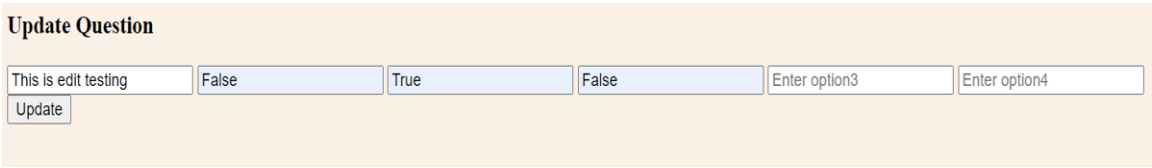
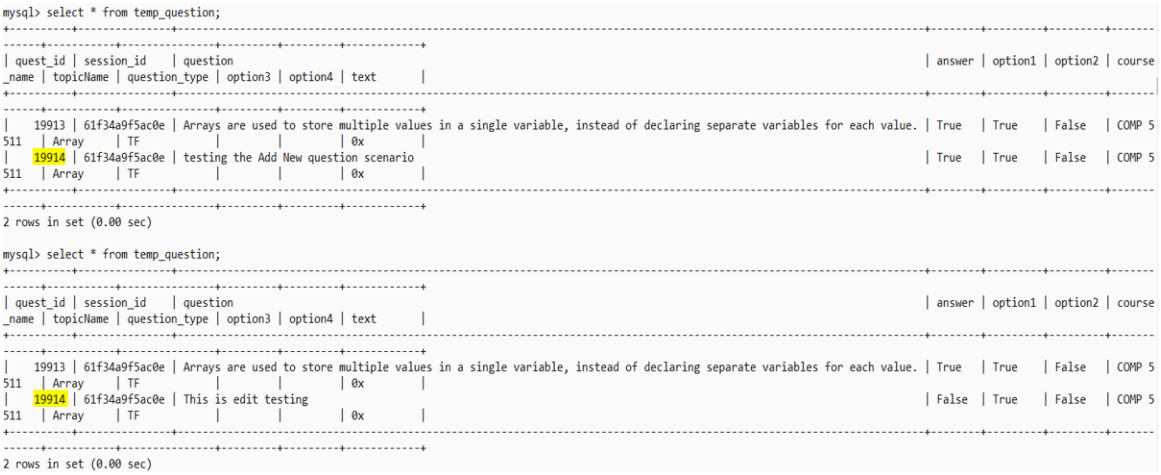


Figure 4.25: Testing of edit page to edit the generated question

Once the coordinator/instructor clicks on the “Update” button, SQL commands are executed, and the data is updated in the database.

Below, figure 4.26 shows that question Id 19914 was updated by the coordinator/instructor.

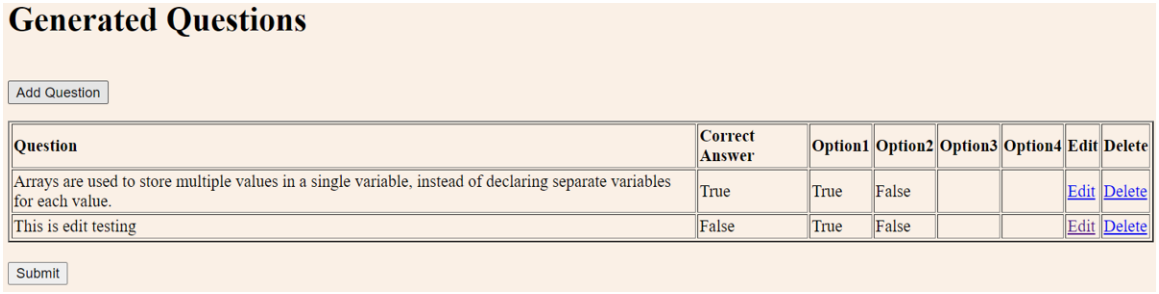


```
mysql> select * from temp_question;
+----------+--------------+-----------------------------------------------------------------------------------------------------------------------------------------------------------------+--------+---------+---------+------
------+-----------+---------------+---------+---------+------------+
| quest_id | session_id   | question                                                                                                                                                        | answer | option1 | option2 | course
_name | topicName | question_type | option3 | option4 | text       |
+----------+--------------+-----------------------------------------------------------------------------------------------------------------------------------------------------------------+--------+---------+---------+------
------+-----------+---------------+---------+---------+------------+
|    19913 | 61f34a9f5ac0e | Arrays are used to store multiple values in a single variable, instead of declaring separate variables for each value. | True   | True    | False   | COMP 5
511   | Array     | TF            |         |         | 0x         |
|    19914 | 61f34a9f5ac0e | testing the Add New question scenario                                                                                                                           | True   | True    | False   | COMP 5
511   | Array     | TF            |         |         | 0x         |
+----------+--------------+-----------------------------------------------------------------------------------------------------------------------------------------------------------------+--------+---------+---------+------
------+-----------+---------------+---------+---------+------------+
2 rows in set (0.00 sec)

mysql> select * from temp_question;
+----------+--------------+-----------------------------------------------------------------------------------------------------------------------------------------------------------------+--------+---------+---------+------
------+-----------+---------------+---------+---------+------------+
| quest_id | session_id   | question                                                                                                                                                        | answer | option1 | option2 | course
_name | topicName | question_type | option3 | option4 | text       |
+----------+--------------+-----------------------------------------------------------------------------------------------------------------------------------------------------------------+--------+---------+---------+------
------+-----------+---------------+---------+---------+------------+
|    19913 | 61f34a9f5ac0e | Arrays are used to store multiple values in a single variable, instead of declaring separate variables for each value. | True   | True    | False   | COMP 5
511   | Array     | TF            |         |         | 0x         |
|    19914 | 61f34a9f5ac0e | This is edit testing                                                                                                                                            | False  | True    | False   | COMP 5
511   | Array     | TF            |         |         | 0x         |
+----------+--------------+-----------------------------------------------------------------------------------------------------------------------------------------------------------------+--------+---------+---------+------
------+-----------+---------------+---------+---------+------------+
2 rows in set (0.00 sec)
```

Figure 4.26: Testing of question list before and after update of question in database

The updated question is displayed in the CrsMgr.



Generated Questions

Add Question

Question	Correct Answer	Option1	Option2	Option3	Option4	Edit	Delete
Arrays are used to store multiple values in a single variable, instead of declaring separate variables for each value.	True	True	False			Edit	Delete
This is edit testing	False	True	False			Edit	Delete

Submit

Figure 4.27: Testing of question list after update

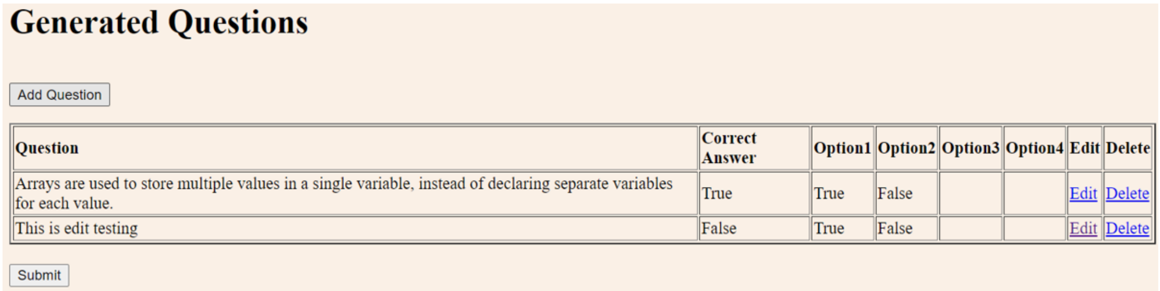
The question list present in the database and CrsMgr is the same. CrsMgr is correctly displaying the questions that are generated for the current session.

4.2.5 Course Coordinator/Instructor can delete the question

The coordinator can delete the generated question. The web application should provide access to the coordinator to delete the question. Each record of the question has a hyperlink to delete the question.

Once the coordinator clicks on the delete hyperlink, the SQL command is executed, and the question

record is deleted from the temp_question table.



Generated Questions

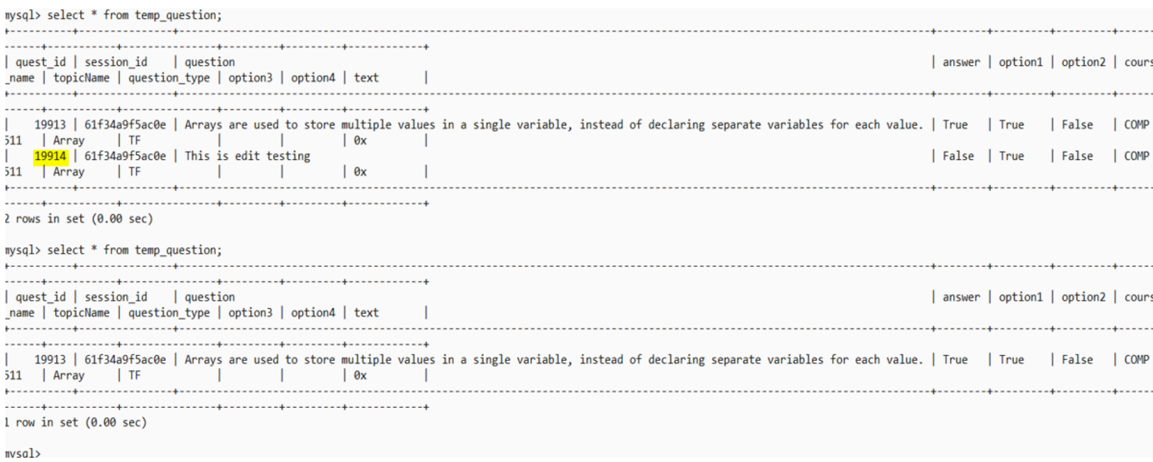
Add Question

Question	Correct Answer	Option1	Option2	Option3	Option4	Edit	Delete
Arrays are used to store multiple values in a single variable, instead of declaring separate variables for each value.	True	True	False			Edit	Delete
This is edit testing	False	True	False			Edit	Delete

Submit

Figure 4.28: Testing of question list before delete

Below, figure 4.29 shows that the question id 19914 is deleted from the table.



```
mysql> select * from temp_question;
| quest_id | session_id     | question                                                                                                    | answer | option1 | option2 | course
_name | topicName | question_type | option3 | option4 | text       |
|    19913 | 61f34a9f5ac0e | Arrays are used to store multiple values in a single variable, instead of declaring separate variables for each value. | True   | True    | False   | COMP 5
511   | Array     | TF            |         |         | 0x         |
|    19914 | 61f34a9f5ac0e | This is edit testing                                                                                        | False  | True    | False   | COMP 5
511   | Array     | TF            |         |         | 0x         |
2 rows in set (0.00 sec)

mysql> select * from temp_question;
| quest_id | session_id     | question                                                                                                    | answer | option1 | option2 | course
_name | topicName | question_type | option3 | option4 | text       |
|    19913 | 61f34a9f5ac0e | Arrays are used to store multiple values in a single variable, instead of declaring separate variables for each value. | True   | True    | False   | COMP 5
511   | Array     | TF            |         |         | 0x         |
1 row in set (0.00 sec)

mysql>
```

Figure 4.29: Testing of question list before and after of deletion of question in database

The updated list of questions should be displayed.

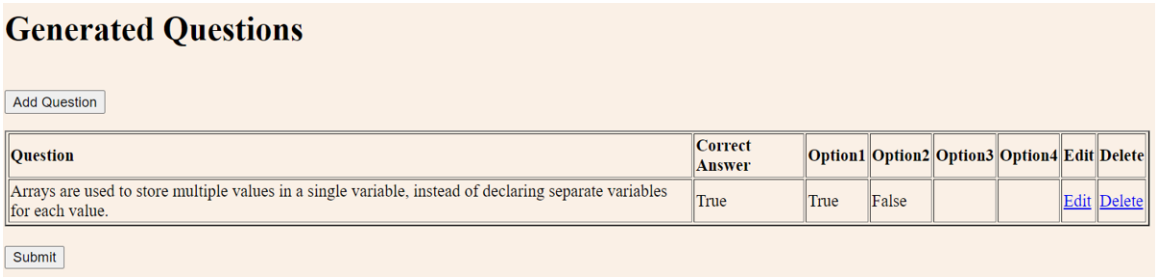


Figure 4.30: Testing of question List after delete

The question list present in the database and CrsMgr is the same. CrsMgr is correctly displaying the questions that are generated for the current session.

4.2.6 Course Coordinator/Instructor can submit the questions to database

The coordinator/instructor verifies the generated questions and submits the questions into the database. Once the coordinator/instructor clicks on the "Submit" button, the SQL command commits the displayed questions in the CrsMgr. The verified questions are submitted to the autogen_question table. This table contains the questions that are used for generating the SAQ quiz.

CrsMgr needs to provide the message of whether the questions are successfully submitted into the database.

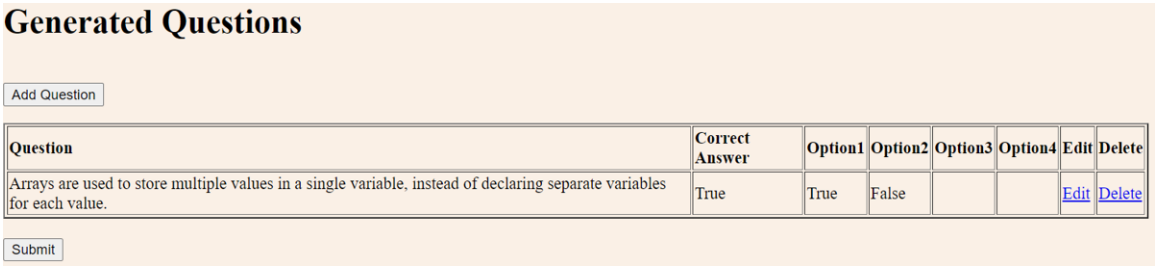


Figure 4.31: Testing of verified question list that will be submitted into database once instructor/coordinator submits

Below, figure 4.32 shows that the questions are submitted into the database.

```
mysql> select * from autogen_question order by quest_id desc limit 2;
+----------+-------------+------------------------------------------------------------------------------------------------------------+--------+---------+---------+--------
--+---------------+---------+---------+-----------+
| quest_id | course_name | question                                                                                                   | answer | option1 | option2 | topicNam
e | question_type | option3 | option4 | text      |
+----------+-------------+------------------------------------------------------------------------------------------------------------+--------+---------+---------+--------
--+---------------+---------+---------+-----------+
|     3541 | COMP 5511   | Arrays are used to store multiple values in a single variable, instead of declaring separate variables for each value. | True   | True    | False   | Array
  | TF            |         |         | 0x        |
|     3540 | COMP 5511   | This is Integration Testing                                                                                | False  | True    | False   | Array
  | TF            |         |         | 0x        |
+----------+-------------+------------------------------------------------------------------------------------------------------------+--------+---------+---------+--------
--+---------------+---------+---------+-----------+
2 rows in set (0.00 sec)

mysql>
```

Figure 4.32: Testing of verified questions that are submitted in database

The verified question list in the CrsMgr is submitted by the course instructor or coordinator. The questions are correctly submitted in the database.

4.2.7 Student can take self-study quiz on the topic

CrsMgr should provide a web page for the student to take a quiz. The quiz questions should be from the course that the student is enrolled in. Once the student submits the quiz, the CrsMgr should show the answer to the quiz questions.

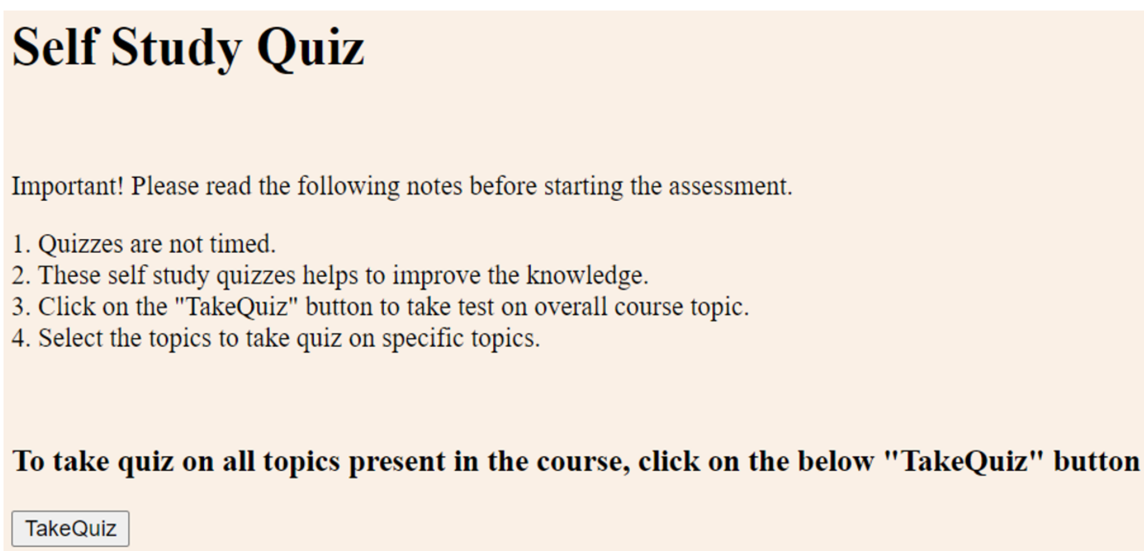


Figure 4.33: Testing of student having the option for Self-Study Quiz

When a student clicks on the “take quiz”, SQL commands should be executed to display the quiz

questions. Random questions from the database are picked and displayed for the students to take up the quiz.

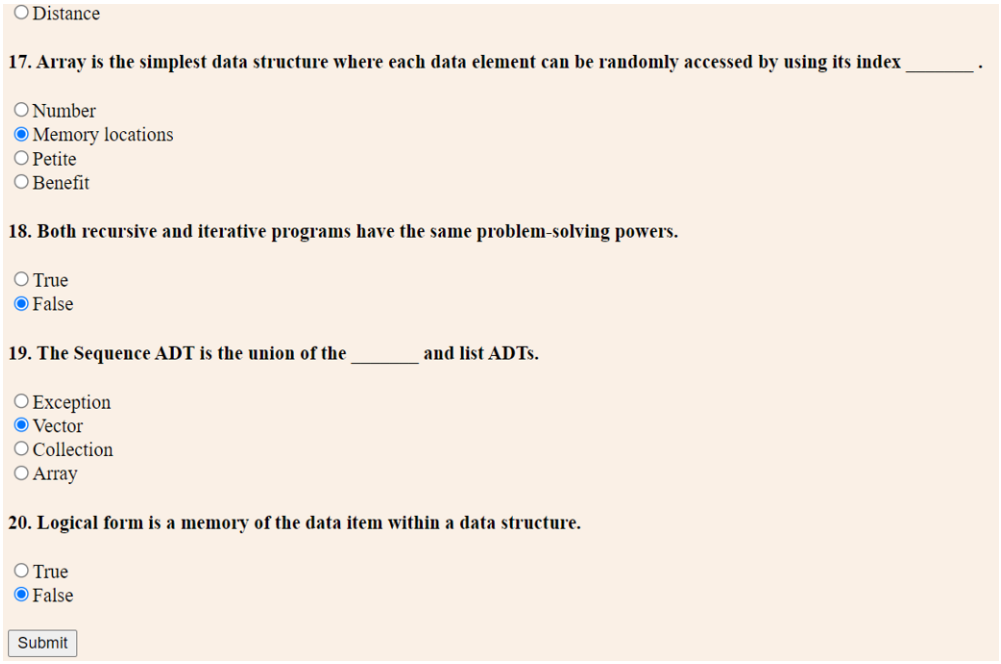


Figure 4.34: Testing of Self-study quiz questions

Once a student submits the quiz, CrsMgr should give the score and the correct answer in the front end. The question displayed should be of the same sequence as the question displayed in the quiz.

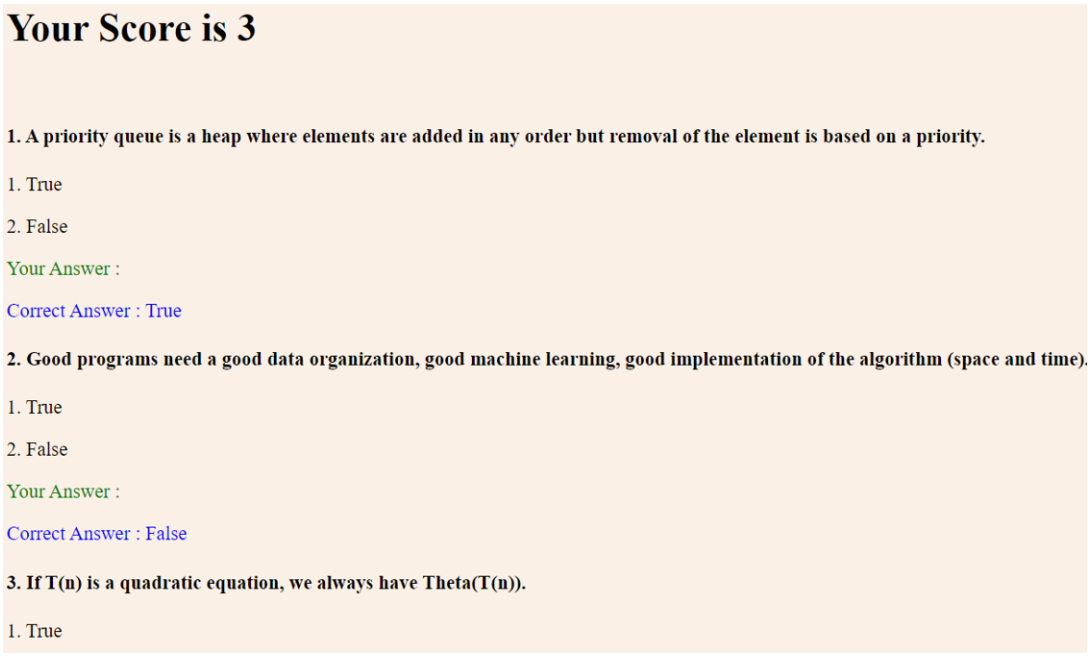


Figure 4.35: Testing of result of Self-study quiz

On clicking the retake button on the quiz result page, the CrsMgr should give questions for the student to take up the quiz. The questions should be from the same course.

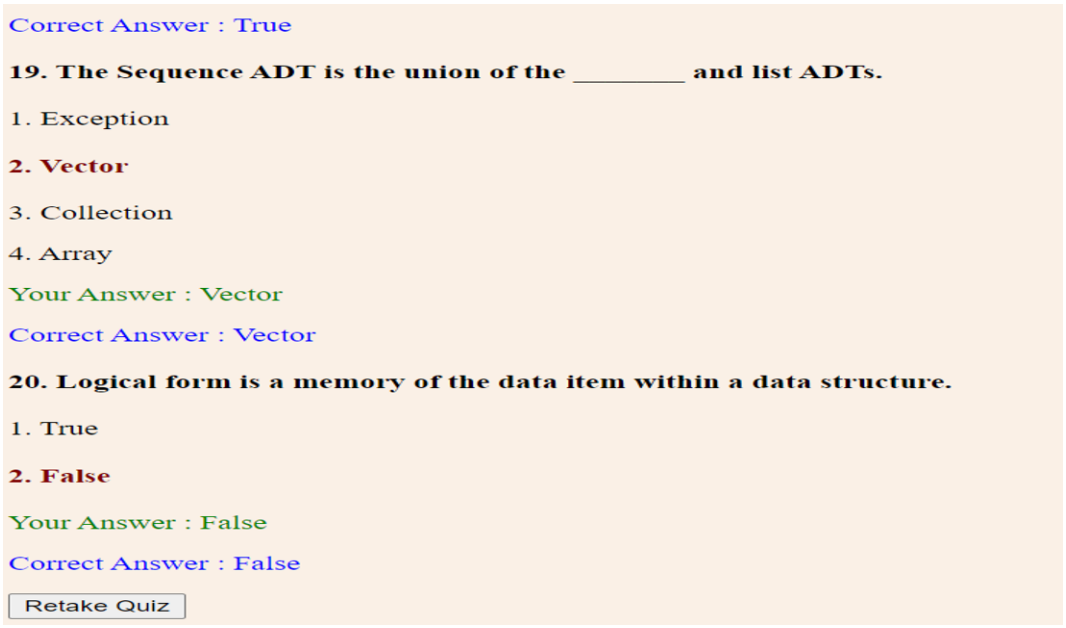


Figure 4.36: Testing of option to re-take quiz

Random questions picked from the database for the course that student has enrolled need to be displayed.

We manually verified the displayed questions to the student belonging to the course that the student has enrolled in. After verification, we found that all the questions were part of the enrolled course.

4.2.8 Student can take self-study quiz on specific topic(s)

Student who has enrolled to a credit course should be able to access to take quiz on the topics that are available for the course. List of topics should be displayed to the students to select. SQL commands is executed in the backend to fetch the topic names that are available for the course.

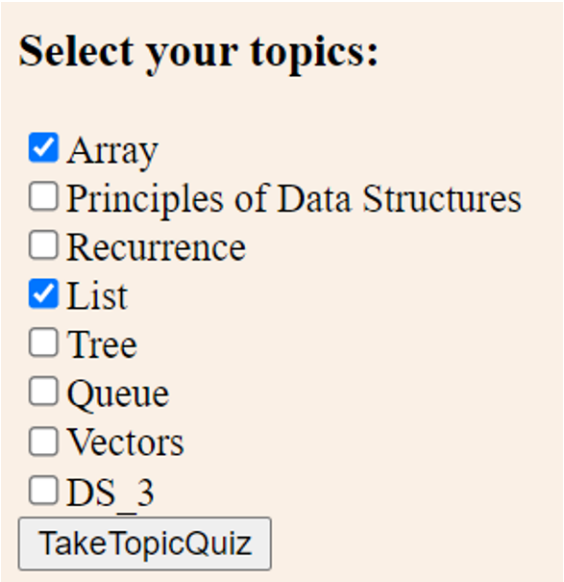


Figure 4.37: Testing of topic list available for the student

On checking the course topic checkboxes, CrsMgr picks the questions from the database, and the questions should be from the selected topic(s) only. Students can pick more than one topic. Manual verification is done to check whether the generated questions belong to the selected topic. The student should be able to take the quiz and submit the quiz.

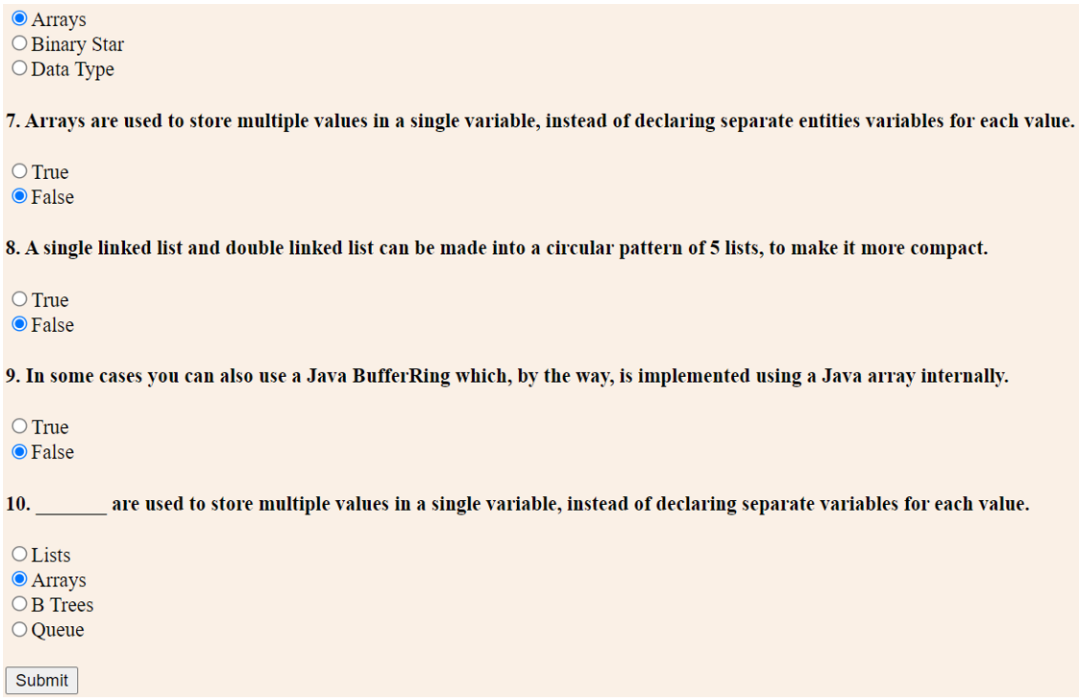


Figure 4.38: Testing of self-study with selected topic

On submitting the quiz, CrsMgr should give the score and the correct answer for the quiz questions.

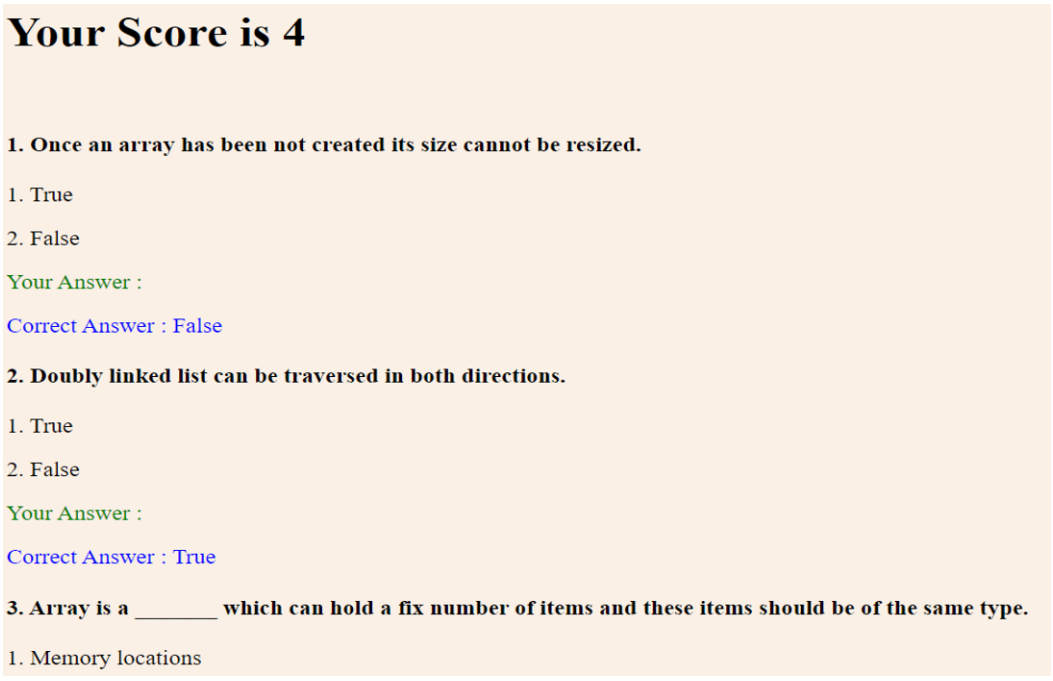


Figure 4.39: Testing of result of self-study quiz on selected topic

The student should have the option to re-take the quiz. The re-take quiz questions should be the same as what the student has selected before.

Figure 4.40: Testing of retake quiz for selected topic

On clicking the ”Retake Quiz” button, the CrsMgr should randomly pick questions from the database and display them to the student. To display a new set of questions, the CrsMgr should execute the SQL command.

We manually verified that the randomly picked questions from the database are from the same topic that the student has selected.

4.3 Conclusion

All the manual unit and integration tests are passed, and the above sections have the test case evidence screenshots. We verified each generated question by the system front end (i.e., in the CrsMgr portal) and the database. All the scenarios of generation of quiz questions or self-study quizzes are working as expected. The appendix section has the statistics of the quality of the questions.

In addition, we have used the latest version of CrsMgr to the COMP 5511 Fall 2021 and COMP 5531 Winter 2022 enrolled students at Concordia University to try out self-study quiz web application. All the enrolled students have access to the system and can try self-study quizzes.

Chapter 5: Update for security and enhancement

5.1 Password Encryption

The previous version of CrsMgr i.e., CrsMgr2016 suffers from security problems such as the plain text password stored in the database. This allowed for the plain text password to be seen by the instructor and show it to the student who had lost/forgotten it. The current practice is to harden the system against hacking and encrypt all sensitive information including the answers to challenge questions. The objective of this work is to harden the system while continuing to provide its functionalities.

Password encryption helps in protecting users' credentials. Without password encryption, hackers or any other user that has access to the database can get the user credentials easily. This makes the web application unsecure. Even if the password is strong enough and does not have an encrypted password then the system is highly unsecure. Storing users' passwords in plain text is a recipe for disaster; any hacker that manages to access the database would discover a treasure of unprotected passwords.

To strengthen the CrsMgr version, the SHA256 hashing algorithm to store the password in the database. The SHA-256 algorithm is one flavor of SHA-2 (Secure Hash Algorithm 2), which was created by the National Security Agency in 2001 as a successor to SHA-1 [41]. SHA-256 is a patented cryptographic hash function that outputs a value that is 256 bits long [41]. In SHA 256 encryption algorithm, data is hashed and not readable unless the recipient has a key [41]. Storing hash values of the password instead of raw text is more secure. When a user enters the password to log in to the system, the hashing value for the entered password checks in the database for authenticating the user credentials for login to the system.

In addition, whenever the course Instructor enters student detail, the SHA256 hashed password is generated and stored in the database, and the raw password is shared with the user. Furthermore, the system always stores the hashed password even when a user tries to modify or change the password. The current version of CrsMgr allows the course Instructor to view the course students' plain text passwords. This is a security breach, and the personal data of the student is not protected. To protect the student's critical data, we have removed the display of the username and password to the course instructor when the instructor views the course students list.

Below is the figure 5.1 of previous version of CrsMgr (i.e., CrsMgr2016) that has the plain text password and username visible to the course instructor.

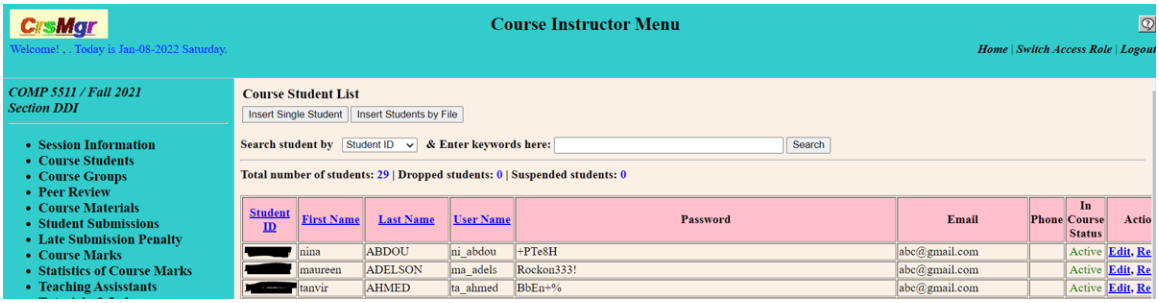
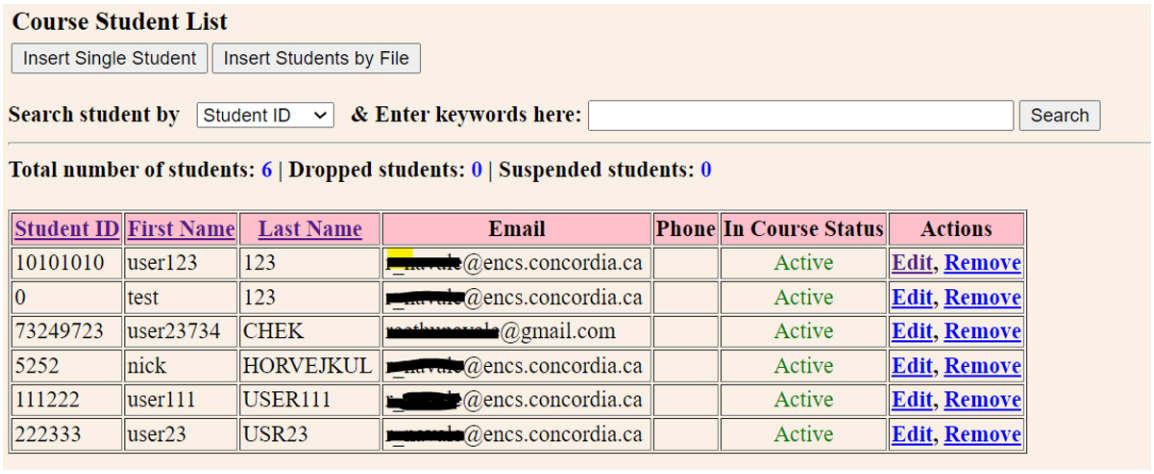


Figure 5.1: CrsMgr2016 version displaying clear text

The current version of CrsMgr has removed the username and password column from the course instructor display for the course students list. Below figure 5.2 is the updated version of CrsMgr.



Course Student List

Insert Single Student | Insert Students by File

Search student by Student ID **& Enter keywords here:** Search

Total number of students: 6 | Dropped students: 0 | Suspended students: 0

Student ID	First Name	Last Name	Email	Phone	In Course Status	Actions
10101010	user123	123	[redacted]@encs.concordia.ca		Active	Edit, Remove
0	test	123	[redacted]@encs.concordia.ca		Active	Edit, Remove
73249723	user23734	CHEK	[redacted]@gmail.com		Active	Edit, Remove
5252	nick	HORVEJKUL	[redacted]@encs.concordia.ca		Active	Edit, Remove
111222	user111	USER111	[redacted]@encs.concordia.ca		Active	Edit, Remove
222333	user23	USR23	[redacted]@encs.concordia.ca		Active	Edit, Remove

Figure 5.2: Current version of CrsMgr displaying list of students

Below figure 5.3 depicts that the CrsMgr stores the SHA256 hashed password value in the database.

```
mysql> select user_id, first_name, last_name, password from user where user_id=1759;
+---------+------------+-----------+------------------------------------------------------------------+
| user_id | first_name | last_name | password                                                         |
+---------+------------+-----------+------------------------------------------------------------------+
|    1759 | user123    | 123       | 26800edc81fd31a4c9954a61ecd69c9a92f5bc12cf7fa3346995b63a8b509e38 |
+---------+------------+-----------+------------------------------------------------------------------+
1 row in set (0.00 sec)

mysql>
```

Figure 5.3: Encrypted password stored in database

If the user did not receive the credential detail email, then the system administrator can change the password of the user in CrsMgr.

5.2 Worst by Average of Quizzes

CrsMgr has a feature available for the course instructor to replace bad score quiz(s) or missed ones with the average quiz score. This helps students not to feel victimized and allows students to increase their overall scores. Sometimes, students have network issues while taking the test or have anxiety issues or forgot to take the test or performed poorly on the quiz. Due to this, the student

score is not up to the mark. To motivate the students and boost their overall score course instructor can decide whether to replace the worst quiz(s) with the average score of the quizzes. By doing this, the overall score of the student can be increased. This would help the student to get a better grade.

If a student performed better on the average than in one or more of the marked entities within a chosen set, then that entity's mark is replaced by the prorated mark based on the student's average performance in this set including the one that is to be replaced by the average. The course instructor has an option to choose how many worst quizzes score to be replaced by the average score of quizzes. This calculated/replaced average bonus score is added to the total score and displayed on the CrsMgr course marks web page. The total bonus score is displayed with the column name "Earned Total with Median".

The course Instructor needs to keep in mind that the substitution should be performed at the end of the term. This "worst by average" substitution is done only for quizzes. Before doing the substitution, make sure that all marked entities in the set for replacement should be marked.

Below figure 5.4 shows the Course Marks page for course instructor users having the button "Set Worst Quizzes By Avg of Quizzes" for substituting the worst quizzes with an average score.

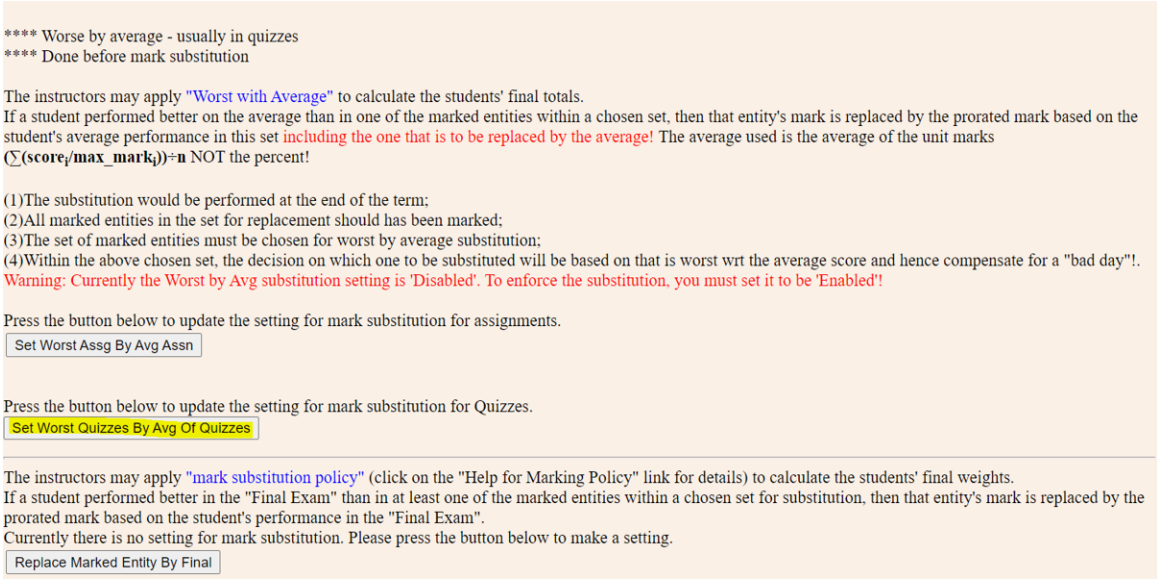


Figure 5.4: Course instructor have option to set worst quizzes by average score of quizzes

To enable the substitution, the course instructor will click on the "Set Worst Quizzes By Avg of

Quizzes" button and select how many worst-performed quizzes will be replaced by the average score. The instructor needs to enable the substitution by selecting the radio button. If the disabled radio button is selected, then average score substitution will not be done. Below, figure 5.5 shows the "Replace worst quizzes with an average of all quizzes" page has options for the instructor to choose how many least marked entities to be changed with average substitution.

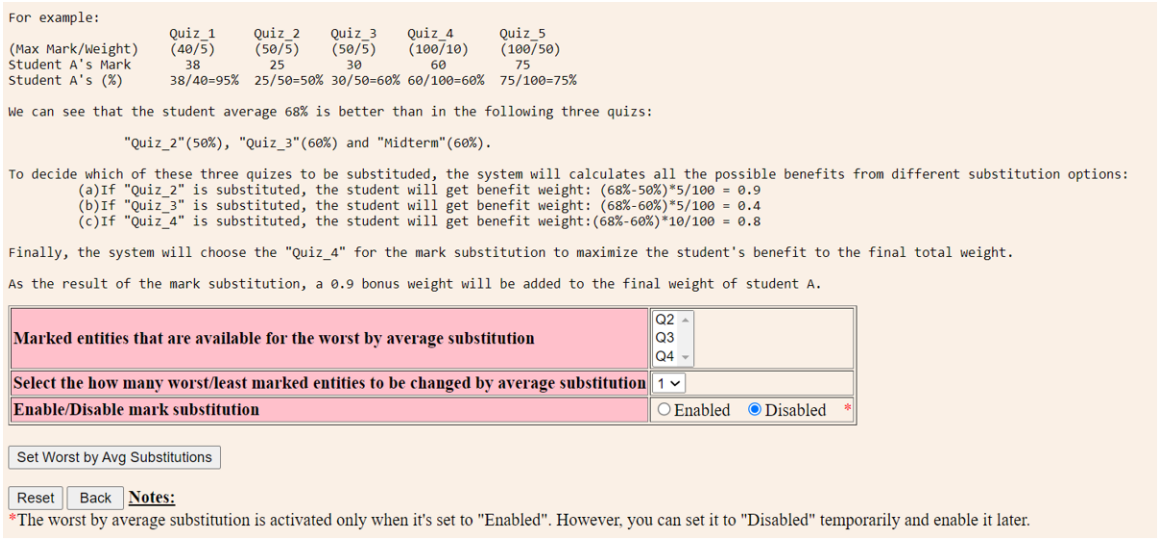


Figure 5.5: List of Quizzes and option to select how many substitutions to be done

Below, figure 5.6 shows the message when the course instructor selects the disabled button.

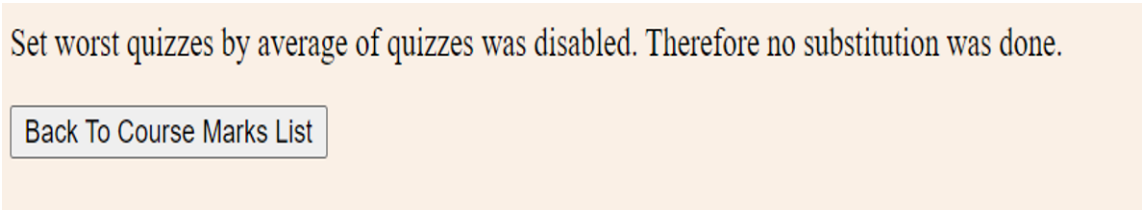
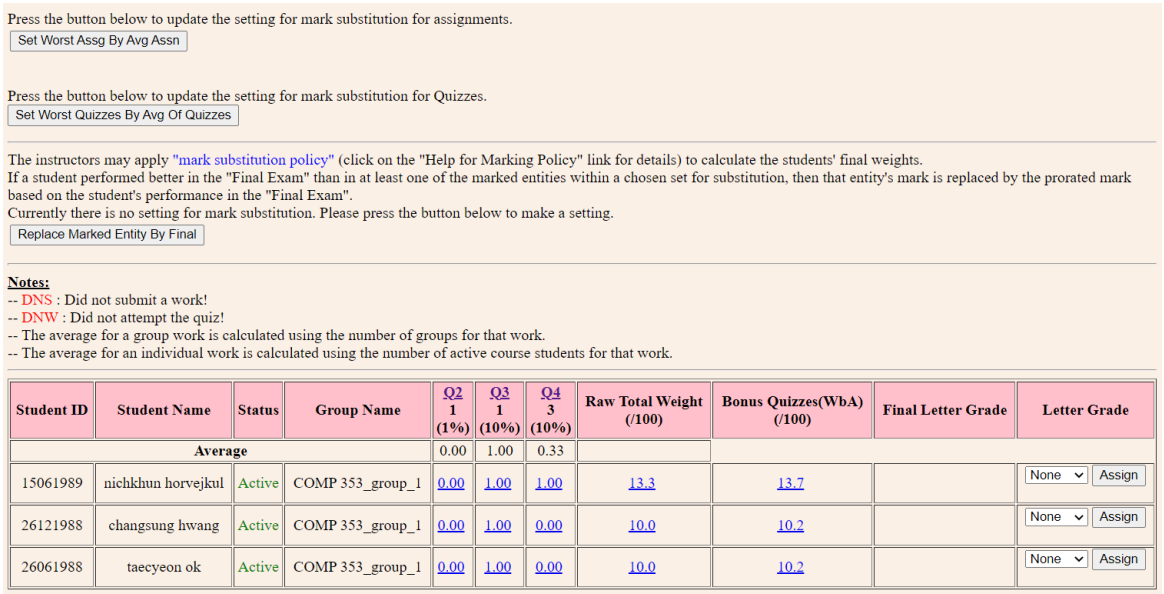


Figure 5.6: Message displayed when instructor disable the substitution process

After the substitution, the prorated marks are updated on the course marks page for the course instructor to check and validate. Below, figure 5.7 shows the column that has the prorated score calculated.


Press the button below to update the setting for mark substitution for assignments.
Set Worst Assg By Avg Assn

Press the button below to update the setting for mark substitution for Quizzes.
Set Worst Quizzes By Avg Of Quizzes

The instructors may apply "mark substitution policy" (click on the "Help for Marking Policy" link for details) to calculate the students' final weights.
If a student performed better in the "Final Exam" than in at least one of the marked entities within a chosen set for substitution, then that entity's mark is replaced by the prorated mark based on the student's performance in the "Final Exam".
Currently there is no setting for mark substitution. Please press the button below to make a setting.
Replace Marked Entity By Final

Notes:
-- DNS : Did not submit a work!
-- DNW : Did not attempt the quiz!
-- The average for a group work is calculated using the number of groups for that work.
-- The average for an individual work is calculated using the number of active course students for that work.

Student ID	Student Name	Status	Group Name	Q2 1 (1%)	Q3 1 (10%)	Q4 3 (10%)	Raw Total Weight (/100)	Bonus Quizzes(WbA) (/100)	Final Letter Grade	Letter Grade
Average				0.00	1.00	0.33				
15061989	nichkhun horvejkul	Active	COMP 353_group_1	0.00	1.00	1.00	13.3	13.7		None Assign
26121988	changsung hwang	Active	COMP 353_group_1	0.00	1.00	0.00	10.0	10.2		None Assign
26061988	taecyeon ok	Active	COMP 353_group_1	0.00	1.00	0.00	10.0	10.2		None Assign

Figure 5.7: Course marks list with bonus score substitution

5.3 Other Enhancements to CrsMgr

Some features listed below was either not present or was not working as expected in the current version of CrsMgr.

5.3.1 Forgot Password

When a user forgets their login password, CrsMgr asks for security questions that the user had entered during the first login. Only if the user enters the correct password, then only CrsMgr will generate a new password and send an email to the user with the new password. The previous version of CrsMgr (i.e., CrsMgr2016) did not have a 24-hour time limit for using the newly generated password.

The current version of CrsMgr has the feature of not accepting the newly generated password if it was generated more than 24 hours ago. That means if the user tries to log in with a new password after 24 hours, that password will not work for login. The user needs to generate a new password by clicking the ”forget password” link and answering the security questions. If a user tries to log

in to CrsMgr within a 24-hour window, then CrsMgr will re-direct the user to change the password window before proceeding to the user options. Changing a password is an obligatory task when a new password is generated through forgot password scenario. After changing the password, the CrsMgr re-directs to the login page. This time user needs to enter the changed password to log in.

The below figures illustrate the forgot password scenario in the current version of CrsMgr.

The login page of CrsMgr has a hyperlink “Forgot Password”. If the user forgot his/her password, then the system can generate a new password after the authentication. Below figure 5.8 illustrates the CrsMgr login page.

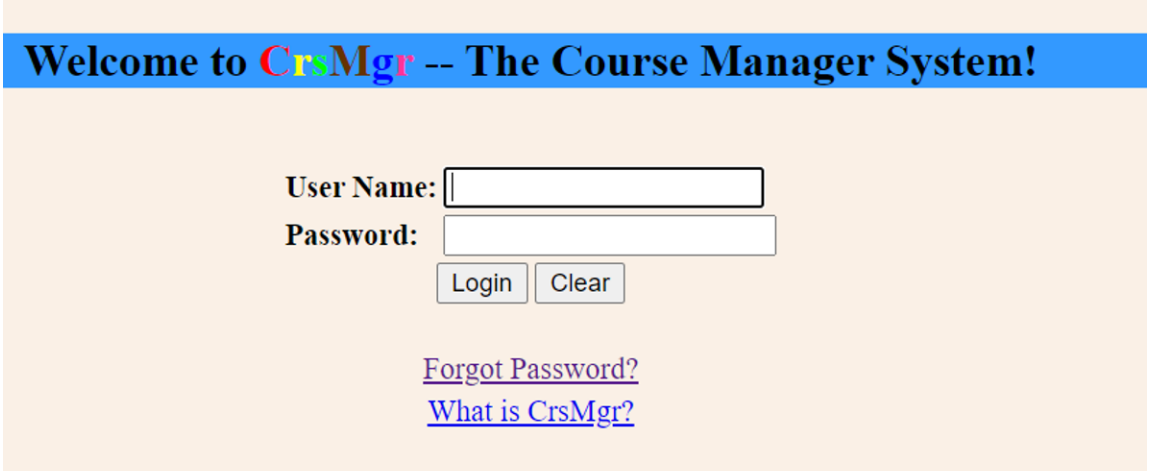


Figure 5.8 CrsMgr Login Page

Figure 5.8: CrsMgr Login Page

User needs to identify whether they are Student or Professor. The user needs to enter either Student ID or Professor ID registered with CrsMgr, first name, and last name. Clicks on Submit button. CrsMgr will validate the information entered by the user. If details are valid, then CrsMgr will ask the security question that the user answered during the first login scenario.

Figure 5.9 illustrates the CrsMgr retrieve password user information page.

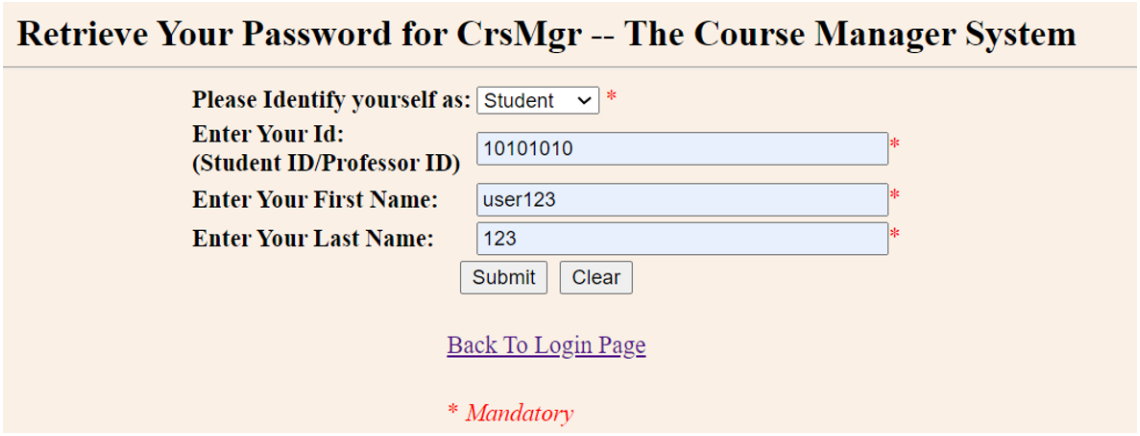


Figure 5.9: Retrieve Password Scenario - Retrieve User Information

CrsMgr will randomly pick a security question from the database. Once the user enters the correct value, an email with credentials is sent. If the user enters the wrong value, then an error message will be displayed, and another random question is displayed for the user to answer.

Figure 5.10 illustrates the security question page for the retrieve password scenario.

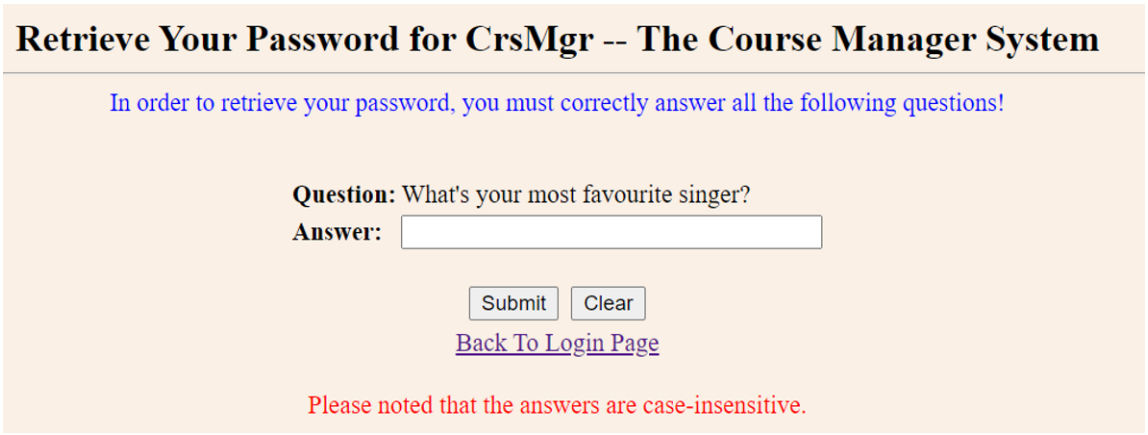


Figure 5.10: Retrieve Password - Security question page

CrsMgr verifies the answer that the user submits. If the answer is wrong, then an error message is displayed.

Figure 5.11 illustrates the scenario when a user enters the wrong answer for the given security question.

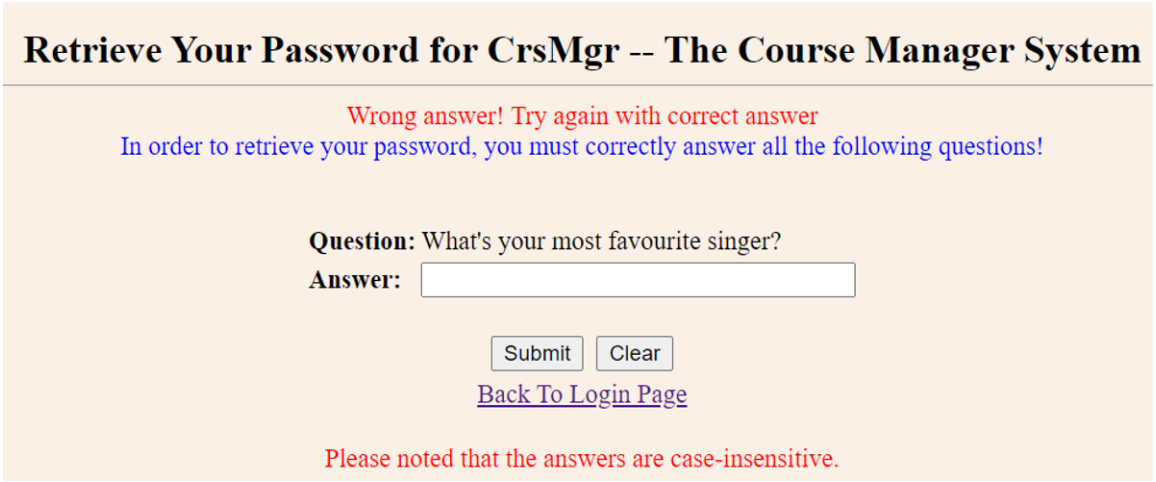


Figure 5.11: Retrieve Password – Negative test scenario

Once the user enters the correct password, a message is displayed on the CrsMgr page stating that the new password is sent to the registered mail address with the CrsMgr, and the user has 24-hour to login and changes the password. If the user does not re-login with the new password, the account will be locked, and the system admin needs to unlock the user.

Below figure 5.12 shows the message that is displayed by the CrsMgr on successful validation.

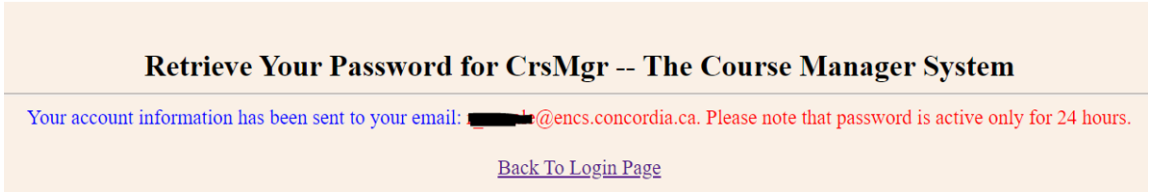


Figure 5.12: Reset Password Message on CrsMgr

If the user reset the password and try to login with new password after 24 hours, then following error message is displayed to the user.

Below figure 5.13 shows the error message if user logins in after 24-hour window.

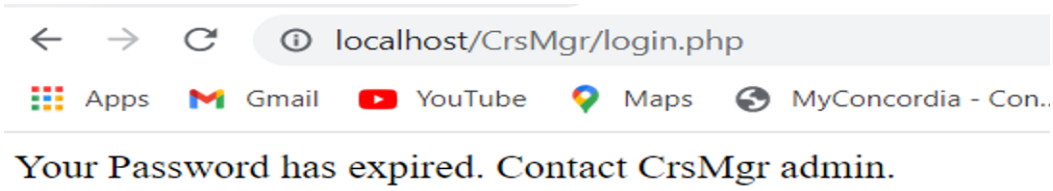


Figure 5.13: Error Message after 24-hour window on reset password

If the user login with a new reset password within a 24-hour window, then CrsMgr will verify the username, password, and reset time (this should be within 24 hours). On successful validation, the CrsMgr will re-direct the user to change the password before proceeding to the main menu of CrsMgr.

Below 5.14 shows the change password page.

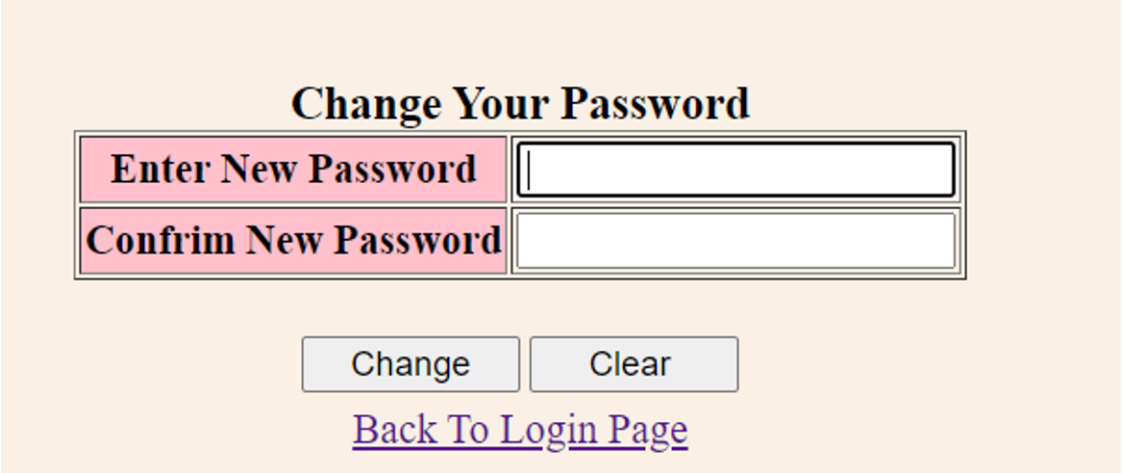


Figure 5.14: Change Password page

Below figure 5.15 illustrates the message on successful password update.

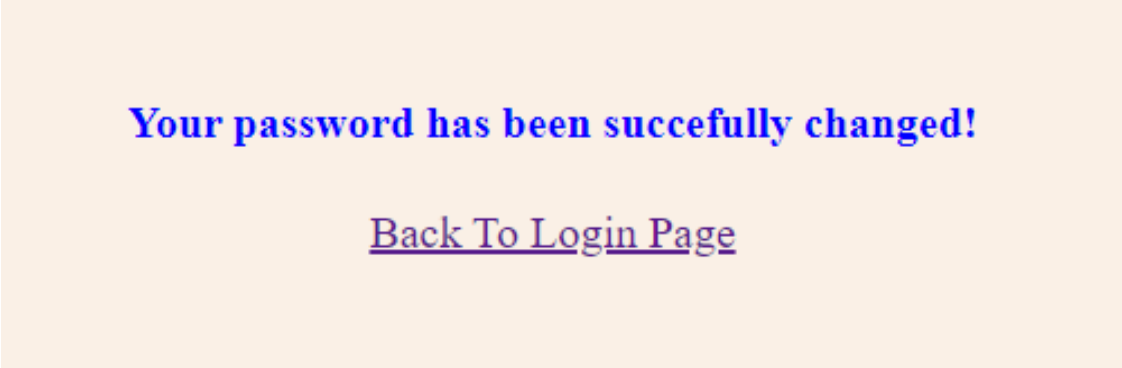


Figure 5.15: Message on password change

User needs to click on hyperlink “Back To Login Page” to login with new changed password.

5.3.2 Updating the final grade

CrsMgr has a feature to assign a grade based on the total score. The total can contain bonus marks based on whether the worst by average substitution was enabled. The previous CrsMgr version also had a bug and was assigning ‘F’ to all the students. The grading was not done based on the grading schema. The current version of CrsMgr has fixed this issue and now assigning the grade based on the grading schema. The course instructor sets the grading schema.

5.3.3 Sending email to group members on group assignment upload and reminder for peer review

CrsMgr has a feature for the student group leader to upload the group work. CrsMgr sends an email to the group leader regarding the group work submission details. However, the CrsMgr does not send an email to other members of the group.

The current version of CrsMgr has fixed this feature of sending emails to the other members of the group along with the group leader. In addition to this, CrsMgr also sends a peer review reminder email to group members (including the leader) for the group work.

5.3.4 Quiz logging

During the period of the COVID pandemic in 2020-2022, courses moved online. The features of CrsMgr were useful in moving the traditional evaluation methods such as mid-terms and final to regularly scheduled remote online quizzes. From its inception, CrsMgr was used to administer these online quizzes, but the students took them online using CrsMgr; however, these students were supervised by invigilators and were on hand to monitor any issues. During the pandemic, it called for the need to offer the quizzes as remote online as well: This was facilitated by CrsMgr. However, since the students were not supervised, and use of 'monitoring' tools was found to be a privacy violation. Under these circumstances, it was felt that the CrsMgr should log the progress of the online assessment more closely, and hence we introduced a logging system to record the progress of the assessment without resorting to privacy-violating monitoring tools.

The quiz logging feature helps the course instructor to investigate problems that students may have in the quiz and provide detail of the quiz questions. These logs are not intended to identify cheating for a quiz as these logs do not provide the student behavior during the quiz. However, the logs show the pattern if the student takes long stops to answer the question or numerous long pauses between questions. Even though the logs provide useful information, they cannot be considered as crucial evidence of cheating.

The previous version was logging the start time of the question to calculate the remaining time left for the quiz. CrsMgr did not log the quiz records. To improve the monitoring of online assessments, the current version of CrsMgr logs the start time of the quiz, IP address of the client machine and the start time of each question. If the question was deferred, then that question will have two entries in the log table. One entry during the deferred time and the other while answering the deferred question.

This logging system helps the instructor to double-check the record entries if there were any issues or concerns raised by the students in the future. Below figure 5.16 shows the entries in the quiz log tables.

```
MariaDB [crsmgrtwnt]> select * from qlog;
+---------------+---------+---------------------+
| assessment_id | user_id | quiz_start_time     |
+---------------+---------+---------------------+
|           202 |    2016 | 2022-02-07 23:02:25 |
|           202 |    2016 | 2022-02-07 23:02:31 |
|           202 |    2016 | 2022-02-07 23:02:37 |
|           202 |    2016 | 2022-02-07 23:02:43 |
+---------------+---------+---------------------+
4 rows in set (0.001 sec)

MariaDB [crsmgrtwnt]> select * from qqlog;
+---------------+---------+---------+---------------------+
| assessment_id | user_id | ques_id | ques_start_time     |
+---------------+---------+---------+---------------------+
|           202 |    2015 |    6350 | 2022-02-07 22:56:35 |
|           202 |    2015 |    6347 | 2022-02-07 22:56:40 |
|           202 |    2015 |    6351 | 2022-02-07 22:56:44 |
|           202 |    2016 |    6347 | 2022-02-07 23:02:31 |
|           202 |    2016 |    6352 | 2022-02-07 23:02:37 |
|           202 |    2016 |    6350 | 2022-02-07 23:02:43 |
+---------------+---------+---------+---------------------+
6 rows in set (0.001 sec)
```

Figure 5.16: Quiz logging tables

5.3.5 Display Changes

In the current version of was designed when the information system of the university did not support it: it still does not! The instructor was obliged to download a csv file of the student list from the university IT and upload it to CrsMgr for the given course. The students are inserted in the order of the names in the student list displayed to the instructor. Students are allowed to be admitted for a few weeks following the start of classes. Such student, inserted, one at a time were put in the end of the existing list in the order of insertion. This anomaly has been rectified in this enhanced version and the student list is maintained in ascending order by the last name.

The previous version of CrsMgr was displaying the oldest course year for a given course as the drop-down menu box. Each login time the course Instructor had to scroll through down and select the year for the given course. To save time and for convenience, the course year display is sorted in descending order. Now the course instructor will always have current or latest year displayed and scrolling and selecting is not required.

5.4 Conclusion

Enhancing the features and fixing problems in CrsMgr greatly helps in enhancing usability and reliability. Replacing the worst quiz or not attempted quiz with an average quiz score will help in students feeling not victimized. Enhancing the reset password feature of keeping a 24-hour active window to reset the password will greatly help in the enhancement of security. Due to the increase in online assessments, improving the logging mechanism of online assessments helps the course instructor to monitor the quiz. The course Instructor gets an insight into the student behavior pattern of the quiz. However, these logs cannot be considered to evaluate students' academic integrity.

All these feature enhancements help CrsMgr to level up the game in the current market for learning management systems and help CrsMgr to become a one-stop destination for all academic-related management system tools.

Chapter 6: Conclusion

6.1 Contribution of the thesis

Our goal in this thesis is to develop features such as generating self-study quiz questions, a self-study quiz for students, providing an option for the course instructor to substitute the worst quiz by average quiz, enhancing security by encrypting the password and applying 24 hours active window for reset password feature, logging the marked quiz entity, and making CrsMgr more accessible based on the feedback in CrsMgr, a web-based learning management system for managing academic task.

Online assessment has become a new norm. Many academic institutions and learners are using the remote learning method either by choice or due to circumstances like the COVID pandemic. Creating a quiz question needs more time, cost, and manpower. The course instructor does not have much time to create sample quiz questions for students to learn. The previous version of CrsMgr i.e., CrsMgr2016 does not have a feature where students can have access to self-quiz to evaluate their learning nor have an interface where the course instructor can generate the questions as discussed in Chapter 2.

The implementation of the generation of quiz questions and self-study quizzes is the main contribution of this thesis. Another objective of the thesis is to add new features and improvements to the existing features of CrsMgr. To save time, cost, and manpower the current version of CrsMgr has a feature to generate quiz questions. The application generates True/False and Fill in the Blanks with multiple-choice options. My main contribution to the thesis is to generate incorrect answer. To ease the jobs of the instructor the display changes were made as discussed in section 5.3.5. The older version of CrsMgr (i.e., CrsMgr2016) was storing the raw password in the database. The raw password was visible for the course instructor and system administrator on the CrsMgr web page.

To protect users' sensitive information and protect CrsMgr from security threats the current version of CrsMgr is storing the encrypted password in the database and removing the display of password information from the course instructor webpage. In the present version of CrsMgr, a new feature is added to the course instructor role user to replace the worst quiz score or not attempted quiz with an average quiz score. This worst by average substitution feature is an option for course instructors, and the instructor can decide whether to enable this feature. In the older version of CrsMgr only the start time of the quiz was being logged there, was no logging done for each question that the user answered. The current version of CrsMgr is logging both the start time of the quiz and its corresponding questions that the user has attempted.

We have tested the systems as demonstrated in Chapter 4 and all the unit and integration testing were passed. All the testing was done manually to verify the quality of the questions that were generated were done manually. Statistics of the questions that were generated are discussed in the appendix section. In addition, we have published the latest version of CrsMgr to COMP 5511 Fall 2021 and COMP 5531 Winter 2022 Concordia University enrolled students for using the application and providing feedback. The self-study quiz has been used successfully many times to support around 100 concurrent student users in a lab environment and in open access.

Apart from the enhancements that are the main objective of this thesis, CrsMgr will be released in GitHub as a free open source for any user to download and install in their system. Thus, making CrsMgr an open-source web application. Furthermore, an instruction document for users to set up the application and user manual to help the user on all pages of the system. In conclusion, the new version of CrsMgr has been improved significantly on security and usability.

6.2 Future Work

In the future, CrsMgr could be more powerful and unique from other learning management systems that are available in the market by adding long and short answer questions (essay questions). CrsMgr could use these generated questions for quizzes. The quality of the questions can be improved by adding more data and having more powerful processing units. As a future extension, we will add a feature to pre-process the PDF file automatically and generate the quiz questions. Thus, improving

the quality of questions. As future work, this research can be extended to other languages like Germany, French, Dutch, Hindi, etc. CrsMgr could be more user-friendly by adding a feature like real-time chat or discussion forum within the course students for the communication and online forum for the communication purpose are common feature for a web-based application. Due to the resources limit of this thesis project, we are unable to fill all these gaps to further improve the CrsMgr system, and we leave as the possible future projects for CrsMgr.

References

[1] Jianhui Zhu, Secure CrsMgr: a course manager system. Master's thesis, Concordia University, 2016. C. H. Chen, "CrsMgr - the course manager system.," 2 12 2013. [Online]. Available: http://spectrum.library.concordia.ca/975541/. [Accessed 1 8 2016].

[2] Moodle Wiki Page, https://en.wikipedia.org/wiki/Moodle

[3] Instructure webpage, https://www.instructure.com/about/our-story

[4] Google Classroom webpage, https://edu.google.com/products/classroom/

[5] TextRank for Automated Summarization, https://arxiv.org/pdf/1602.03606.pdf

[6] PDF Wikipedia, https://en.wikipedia.org/wiki/PDF

[7] Extracting headers and paragraphs from pdf using PyMuPDF, https://ichi.pro/fr/extraction-d-en-tetes-et-de-paragraphes-a-partir-de-pdf-a-l-aide-de-pymupdf-113724360868967

[8] ConceptNet Webpage, https://conceptnet.io/

[9] WordNet Princeton University, https://wordnet.princeton.edu/

[10] sense2vec reloaded: contextually keyed word vectors, Matthew Honnibal & Ines Montani in 2019, https://explosion.ai/blog/sense2vec-reloaded

[11] The Alcalde: The University of Texas at Austin Alumni Magazine - https://books.google.ca/books?id=6M8DAAAAMBAJ&redir_esc=y

[12] Multiple Choice Wikipedia - https://en.wikipedia.org/wiki/Multiple_choice

[13] Origins and Purposes of Multiple-Choice Tests, San Francisco State University - http://ctfd.sfsu.edu/teaching-practices/origins-and-purposes-of-multiple-choice-tests

[14] "Multiple-Choice Tests: Revisiting the Pros and"Cons". Faculty Focus — Higher Ed Teaching & Learning - https://www.facultyfocus.com/articles/educational-assessment/multiple-choice-tests-pros-cons/

[15] C. H. Chen, ”CrsMgr - the course manager system.,” 2 12 2013. [Online]. Available: http://spectrum.library.concordia.ca/975541/. [Accessed 1 8 2016].

[16] The Alcalde: The University of Texas at Austin Alumni Magazine - https://books.google.ca/books?id=6M8DAAAAMBAJ&redir_esc=y

[17] Multiple Choice Wikipedia - https://en.wikipedia.org/wiki/Multiple_choice

[18] ”Multiple-Choice Tests: Revisiting the Pros and Cons”. Faculty Focus — Higher Ed Teaching & Learning - https://www.facultyfocus.com/articles/educational-assessment/multiple-choice-tests-pros-cons/

[19] Origins and Purposes of Multiple-Choice Tests, San Francisco State University - http://ctfd.sfsu.edu/teaching-practices/origins-and-purposes-of-multiple-choice-tests

[20] Question types, True or False - http://www.technicalauthoring.com/wiki/index.php/ Question_types:_True_or_False

[21] Yes-no question Wikipedia - https://en.wikipedia.org/wiki/Yes%E2%80%93no_question

[22] Cloze test Wikipedia - https://en.wikipedia.org/wiki/Cloze_test

[23] Question Wikipedia - https://en.wikipedia.org/wiki/Question#Alternative_questions

[24] Five Ws Wikipedia - https://en.wikipedia.org/wiki/Five_Ws

[25] Leon Nelson Flint, Newspaper Writing in High Schools, Containing an Outline for the Use of Teachers, University of Kansas, 1917, p. 47. - https://archive.org/details/newspaperwritin00flingoog/page/n7

[26] 10 Useful Web Tools for Creating Online Quizzes and Polls - https://www.educatorstechnology.com/2014/02/10-useful-web-tools-for-creating-online.html

[27] Socrative, Student Response System - https://www.socrative.com/

[28] ProProfs, Quiz Maker - https://www.proprofs.com/quiz-school/?utm_campaign=Listly&utm_medium=list&utm_source=listly

[29] Flubaroo, online tool for grading - http://www.flubaroo.com/

[30] Class Maker – Quiz Maker for Business and Education - https://www.classmarker.com/?utm_campaign=Listly&utm_medium=list&utm_source=listly

[31] ThatQuiz - https://www.thatquiz.org/?utm_campaign=Listly&utm_medium=list&utm_source=listly

[32] ExamTime Quizzes - https://www.goconqr.com/en/quiz-maker/?utm_campaign=Listly&utm_medium=list&utm_source=listly

[33] Testmoz - https://testmoz.com/?utm_campaign=Listly&utm_medium=list&utm_source=listly

[34] Gnowledge - https://www.gnowledge.com/

[35] Online Quiz Creator: Play or Make a Quiz for Free - https://www.onlinequizcreator.com/?utm_campaign=Listly&utm_medium=list&utm_source=listly

[36] GoToQuiz - https://www.gotoquiz.com/create.html?errs=1&utm_campaign=Listly&utm_medium=list&utm_source=listly

[37] QuizStar - http://quizstar.4teachers.org/?utm_campaign=Listly&utm_medium=list&utm_source=listly

[38] Survey Anyplace - Mobile Quizzes and Surveys - https://surveyanyplace.com/?utm_campaign=Listly&utm_medium=list&utm_source=listly

[39] Mentimeter - https://www.mentimeter.com/?utm_campaign=Listly&utm_medium=list&utm_source=listly

[40] Edmodo - https://new.edmodo.com/

[41] SHA-256 Algorithm Overview Blog, https://www.n-able.com/blog/sha-256-encryption

[42] Adobe Captivate Prime, https://www.adobe.com/

[43] SAP Litmos, https://www.litmos.com/

[44] Moodle, https://moodle.org/

[45] Canvas LMS, https://www.instructure.com/canvas

[46] The Evolution and Diffusion of Learning Management Systems: The case of Canvas LMS, https://ohiostate.pressbooks.pub/drivechange/chapter/the-evolution-and-diffusion-of-learning-management-systems-the-case-of-canvas-lms/#: :text=Instructure%2C%20the%20creator%20of%20Canvas,Daley%20(Instructure%2C%202018).

[47] Ramsi Goutham Golla's GitHub repository, https://github.com/ramsrigouthamg

[48] Practical AI: Automatically Generate Multiple Choice Questions (MCQs) from any content

with BERT Summarizer, Wordnet and Conceptnet, https://towardsdatascience.com/practical-ai-automatically-generate-multiple-choice-questions-mcqs-from-any-content-with-bert-2140d53a9bf5

[49] Practical AI: Automatically Generate True or False questions from any content with OpenAI GPT2, Sentence BERT and Berkley Constituency parser, https://medium.com/swlh/practical-ai-automatically-generate-true-or-false-questions-from-any-content-with-openai-gpt2-9081ffe4d4c9

[50] Extracting headers and paragraphs from pdf using PyMuPDF, Louis de Bruijn, Apr 9, 2020, https://towardsdatascience.com/extracting-headers-and-paragraphs-from-pdf-using-pymupdf-676e8421c467

[51] Summa - a python Library, https://pypi.org/project/summa/

Appendix A:

A.1 Statistics and Analysis of questions generated

Below is the statistics for the text-based question generation. We have used some text from the database course textbook. 126 out of 152 questions that are generated are either good or ok quality with or without some edition. Around 82% of the question are considered. 101 questions out of 152 questions does not need any update which is somewhere around 66%.

Text 1 Statistics		
Quality	Changes Required	No Changes
Good	2	84
Ok	23	17
Not Good	26	0
Count	51	101
Total Question Generated		**152**

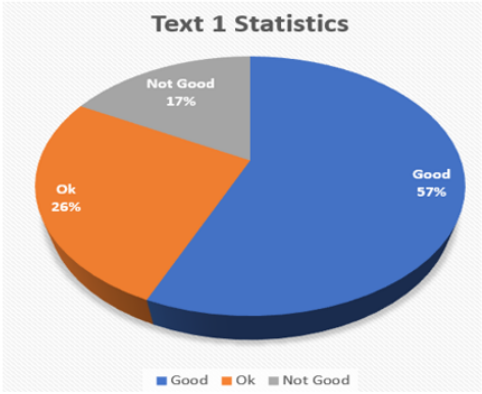
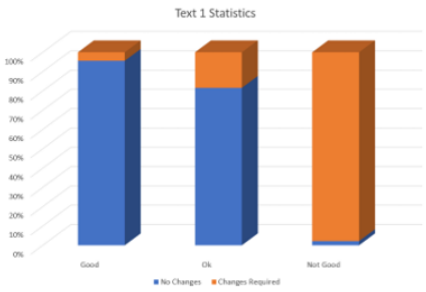


Figure A.1: Text 1 Statistics – Run1

The same text is used again to check if same results are produced. However, the distractor or incorrect keyword/answer is generated different for each session. 106 out of 152 questions that are generated are of good or ok quality with or without some changes i.e., 69%. 97 out of 152 questions does not need any update i.e., around 63%. The accuracy deteriorated as the distractor keywords of not good quality or not relevant to the question topic.



Text 1 Statistics		
Quality	Changes Required	No Changes
Good	3	65
Ok	7	31
Not Good	45	1
Count	55	97
Total Question Generated		**152**

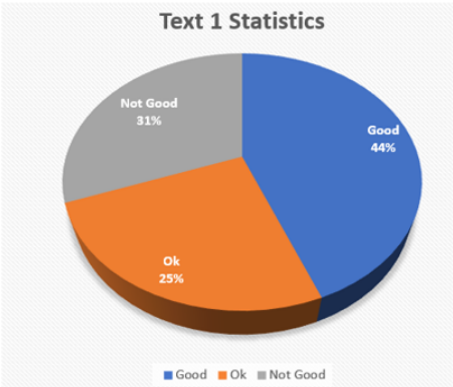


Figure A.2: Text 1 Statistics – Run2

Similar tests were carried on different type/topic data.

Text2: The data text had many SQL commands, and the distractor gave lessor accuracy results.

111/168 questions are of good or ok quality. (66% accuracy)

89/168 questions do not need any modifications. (52% accuracy)

Text 2 Statistics		
Quality	Changes Required	No Changes
Good	3	43
Ok	19	46
Not Good	57	0
Count	79	89
Total Question Generated		**168**

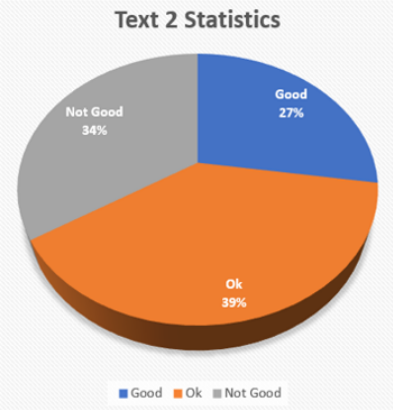
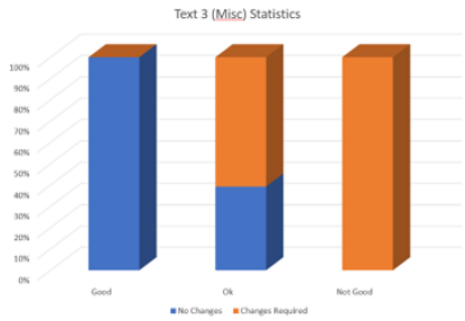


Figure A.3: Text 2 Statistics

Text 3: This data text has general topic (non-technical/non computer science based).

101/117: Questions are of good or ok quality (86% accuracy)

84/117: Questions that does not need any modification/update. (72% accuracy)



Text 3 (Misc) Statistics		
Quality	Changes Required	No Changes
Good	0	73
Ok	17	11
Not Good	16	0
Count	33	84
Total Question Generated		**117**

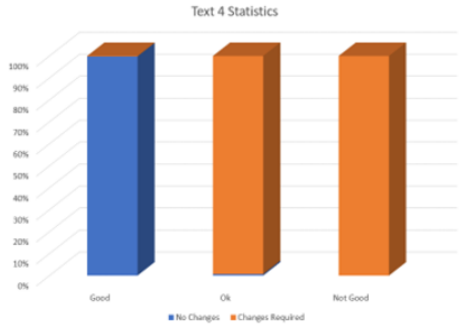
Figure A.4: Text 3 (Misc) Statistics

Text 4: The PDF file test output. The test file had java program examples explained in the chapter. This testing shows how the identification of passage or summary plays an important role in achieving higher accuracy.

735/2810: Questions are of good or ok quality (26% accuracy)

448/2810: Questions that does not need any modification/update. (16% accuracy)

In this negative scenario the program fails to identify passage from the PDF file. Since the chapter has java programs that occupies most number of words that are of same font size. Therefore, the program considers the java program for summarization. The generated summary has impartial java program therefore, the question that are generated from this summary have very low accuracy.



Text 4 Statistics		
Quality	Changes Required	No Changes
Good	1	446
Ok	286	2
Not Good	2075	0
Count	2362	448
Total Question Generated		**2810**

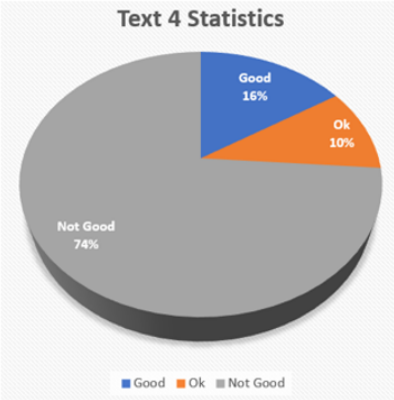


Figure A.5: Text 4 Statistics

The quality of the question depends on the quality of the distractor i.e., it should be more relevant to the topic. Another factor on which the quality of question depends on is the summary sentences. Identification of important sentences from the text will help in improving the quality. Good quality of questions is generated when the text or PDF file has text-based sentences nor any program or any computer language syntax.

To generate good quality of questions, you can pre-process the PDF file by removing the unwanted passages or text or figures or tables. Currently CrsMgr does not have a mechanism to pre-process the text.

A.2 Database Tables for MCQ Sub-system and for the Enhanced Features

The **temp_question** table stores the self-evaluation questions that are generated by MCQ sub-system of CrsMgr system temporarily until the course instructor validates the questions. This is a temporary table to store the questions. The attribute **'quest_id'** is a primary key.

Field	Type	Null	Key	Default	Extra
quest_id	bigint	NO	PRI	NULL	auto_increment
session_id	varchar(255)	YES		NULL	
question	varchar(500)	YES		NULL	
answer	varchar(100)	YES		NULL	
option1	varchar(50)	YES		NULL	
option2	varchar(50)	YES		NULL	
option3	varchar(50)	YES		NULL	
option4	varchar(50)	YES		NULL	
course_name	varchar(20)	YES		NULL	
topicName	varchar(255)	YES		NULL	
question_type	varchar(255)	YES		NULL	
text	blob	YES		NULL	

Table A.1: temp_question

The **autogen_question** table stores the validated questions that instructor has committed. These questions are generated by CrsMgr or modified by course instructor. The attribute **'quest_id'** is a primary key.

Field	Type	Null	Key	Default	Extra
quest_id	bigint	NO	PRI	NULL	auto_increment
session_id	varchar(255)	YES		NULL	
question	varchar(500)	YES		NULL	
answer	varchar(100)	YES		NULL	
option1	varchar(50)	YES		NULL	
option2	varchar(50)	YES		NULL	
option3	varchar(50)	YES		NULL	
option4	varchar(50)	YES		NULL	
course_name	varchar(20)	YES		NULL	
topicName	varchar(255)	YES		NULL	
question_type	varchar(255)	YES		NULL	
text	blob	YES		NULL	

Table A.2: autogen_question

The **user** table is modified to accommodate the password encryption and reset password features. The user table contains the user account information for the CrsMgr System. Each user has unique user_name to login, even though the user might have multiple access roles. The attribute **'active'** indicates whether the user is in good condition or has been suspended. When a user forgets his password, three challenge questions will be used for validating the users. The attribute **'user_id'** is the primary key. The attribute **'fp_flag'** and **'fp_dateTime'**. The **'fp_flag'** indicates if 24-hour limit is applied and **'fp_dateTime'** stores the time when user request to reset the password.

Field	Type	Null	Key	Default	Extra
user_id	int(11)	NO	PRI	NULL	auto_increment
user_name	varchar(20)	NO	UNI	NULL	
password	varchar(65)	NO		NULL	
first_name	varchar(30)	YES		NULL	
last_name	varchar(30)	YES		NULL	
phone	varchar(20)	YES		NULL	
extension	varchar(6)	YES		NULL	
office	varchar(50)	YES		NULL	
email	varchar(50)	YES		NULL	
home_page	varchar(100)	YES		NULL	
question_1	varchar(100)	YES		NULL	
answer_1	varchar(30)	YES		NULL	
question_2	varchar(100)	YES		NULL	
answer_2	varchar(30)	YES		NULL	
question_3	varchar(100)	YES		NULL	
answer_3	varchar(30)	YES		NULL	
active	tinyint(1)	YES		1	
identity	tinyint(1)	YES		0	
office	varchar(50)	YES		NULL	
fp_flag	varchar(1)	YES		NULL	
fp_dateTime	int	YES		NULL	

Table A.3: user

The table **qlog** contains the quiz start time information of each student access user. The attribute **'assessment_id'** is primary key of assessment table and foreign key in qlog table. The attribute **'user_id'** is primary key of user table and foreign key in qlog table. The attribute **'quiz_start_time'** records the quiz start time. The attribute **'ip'** records the IP address of the client machine.

Field	Type	Null	Key	Default	Extra
assessment_id	int(11)	NO	MUL	NULL	
user_id	int(11)	NO	MUL	NULL	
quiz_start_time	datetime	YES		NULL	
ip	varchar(100)	YES		NULL	

Table A.4: qlog

The **qqlog** table stores the time of the quiz question that the student attempts. The attribute **'assessment_id'** is primary key of assessment table and foreign key of qqlog table. The attribute **'user_id'** is primary key of user table and foreign key of qqlog table. The attribute **'ques_id'** is the primary key in the assessment_question table. The attribute **'ques_start_time'** records the time for each question that student attempts.

Field	Type	Null	Key	Default	Extra
assessment_id	int(11)	NO	MUL	NULL	
user_id	int(11)	NO	MUL	NULL	
ques_id	int(11)	NO		NULL	
ques_start_time	datetime	YES		NULL	

Table A.5: qqlog

The **ALLQSCORE** table stores the details of all quiz scores of each user. Whether student have attempted the quiz or not. The table ALLQSCORE is used for substituting worst quiz by average score feature. The attribute **'UID'** records user_id, **'SID'** attribute records student_id, **'Q'** records assessment_id, **'MAX'** attribute records max score and total attribute records score of a student.

Field	Type	Null	Key	Default	Extra
UID	int(11)	NO		0	
course_id	int(11)	NO		0	
course_session_id	int(11)	YES		NULL	
SID	int	NO		0	
Lastname	varchar(30)	YES		NULL	
Q	int	NO		0	
MAX	float(10,2)	NO		000	
total	int	NO		0	

Table A.6: ALLQSCORE

The **AvgQmark** view used to store the average score of the quiz. The AvgQmark is used for substituting worst quiz by average score feature. The attribute **'UID'** stores user_id information and **'SID'** stores student_id.

Field	Type	Null	Key	Default	Extra
UID	int(11)	NO		0	
SID	int	NO		0	
Lastname	varchar(30)	YES		NULL	
Avg	double(22,3)	YES		NULL	

Table A.7: AvgQmark

The **HaveDoneScore** view stores the details of quiz that are attempted by the student. The HaveDoneScore view is used for substituting worst quiz by average score feature. The attribute '**UID'** records user_id, **'SID'** records student_id, **'Q'** records assessment_id, **'MAX'** records max score and **'total'** records the quiz score.

Field	Type	Null	Key	Default	Extra
UID	int(11)	NO		0	
course_id	int(11)	NO		0	
course_session_id	int(11)	YES		NULL	
SID	int	NO		0	
Lastname	varchar(30)	YES		NULL	
Q	int	NO		0	
MAX	float(10,2)	NO		000	
total	int	NO		0	

Table A.8: HaveDoneScore

The **NotDoneScore** view stores the details of quiz that are not attempted by the student. The NotDoneScore view is used for substituting worst quiz by average score feature. The attribute **'UID'** records user_id, **'SID'** records student_id, **'Q'** records assessment_id, **'MAX'** records max score and **'total'** records the quiz score.

Field	Type	Null	Key	Default	Extra
UID	int(11)	NO		0	
course_id	int(11)	NO		0	
course_session_id	int(11)	YES		NULL	
SID	int	NO		0	
Lastname	varchar(30)	YES		NULL	
Q	int	NO		0	
MAX	float(10,2)	NO		000	
total	int	NO		0	

Table A.9: NotDoneScore

The **QBONUS** table stores the calculated bonus score for each student. The QBONUS table is used for substituting worst quiz by average score feature. The attribute **'ScoreDiff'** records difference score value after the worst by average computation and the attribute **'PerCentDiff'** records the difference based on the weights of the assessments. The PerCentDiff value will be used to compute the Bonus Total. Based on this Bonus Total final grades will be assigned.

Field	Type	Null	Key	Default	Extra
course_id	int	NO	MUL	NULL	
course_session_id	int	NO	MUL	NULL	
user_id	int	NO	MUL	NULL	
student_id	int	NO		NULL	
ScoreDiff	decimal(4,2)	YES		NULL	
PerCentDiff	decimal(4,2)	YES		NULL	

Table A.10: QBONUS

The **QDetails** view has the details of quiz. The details like max score, weight, and course details. The QDetails view is used for substituting worst quiz by average score feature. The attribute **'QID'** records the assessment_id, **'Max'** records the max score for the given assessment, **'wt'** records the weight of the assessment.

Field	Type	Null	Key	Default	Extra
course_id	int	NO	MUL	0	
QID	int	NO	MUL	0	
Max	float(10,2)	NO		NULL	
wt	float(5,2)	NO		NULL	
course_session_id	int	YES		NULL	

Table A.11: QDetails

The **QSCORETEMP** table stores the average quiz calculation details for each student. The QSCORETEMP table is used for substituting worst quiz by average score feature. The attribute **'UID'** records user_id, **'SID'** records student_id, **'Q'** records worst assessment id, **'Qwt'** records quiz weight, **'NewScore'** stores the Average score that is calculated, **'Qmax'** records the quiz max score, **'OldTotal'** stores the raw total, attribute **'ScoreDiff'** records difference score value after the worst by average computation and the attribute **'PerCentDiff'** records the difference based on the weights of the assessments.

Field	Type	Null	Key	Default	Extra
UID	int	NO		0	auto_increment
course_id	int	NO		0	
course_session_id	int	YES		NULL	
SID	int	NO		0	
Lastname	varchar(30)	YES		NULL	
Q	text	YES		NULL	
Qwt	float(5,2)	NO		0.00	
Newscore	double(22,0)	YES		NULL	
Qmax	float(10,2)	NO		0.00	
OldTotal	int	NO		0	
ScoreDiff	double(22,2)	YES		NULL	
PerCentDiff	double(22,2)	YES		NULL	

Table A.12: QSCORETEMP

The **SecStudents** view stores the details of students that enrolled for the course. The SecStudents view is used for substituting worst quiz by average score feature. The attribute **'UID'** records user_id and **'SID'** records the student Id.

Field	Type	Null	Key	Default	Extra
UID	int	NO		0	
SID	int	NO		0	
Lastname	varchar(30)	YES		NULL	

Table A.13: SecStudents

The **WorstQ** view stores the details of the worst score quiz for each student for the course that student has enrolled. The WorstQ view is used for substituting worst quiz by average score feature. The attribute **'UID'** stores user_id and **'WorstQ'** stores the worst quiz id.

Field	Type	Null	Key	Default	Extra
UID	int	NO		0	
WorstQ	text	YES		NULL	

Table A.14: WorstQ

The **self_study_log table** logs the self-study quiz details. The attribute **'user_id'** stores the user id information of the student and is a foreign key reference to user table. The attribute **'quest_ids'** stores all the question ids that are picked for the self-study quiz. The attribute **'score'** stores the self-study quiz score of the corresponding self-study quiz that a student takes. The attribute **'quiz_start_time'** records the self-study quiz start date and time. The attribute **'quiz_end_time'** records the self-study quiz submitted date and time.

Field	Type	Null	Key	Default	Extra
ss_quiz_id	int	NO	PRI	NULL	auto_increment
user_id	int	NO		NULL	
quest_ids	varchar(255)	YES		NULL	
score	int	YES		NULL	
quiz_start_time	datetime	YES		NULL	
quiz_end_time	datetime	YES		NULL	

Table A.15: self_study_log

A.3 Git Release

CrsMgr is now available in Git with deployment steps and software details. The link to the GIT repository is present in Dr. Bipin C Desai's webpage, https://users.encs.concordia.ca/ bcdesai/CrsMgr.

A.4 Python Libraries

It is recommended to use Python 3.8 version. Following python libraries needs to be installed.

- **summa**: Version higher than 1.2.0
- **nltk**: Version higher than 3.5
- **spacy**: Version higher than 3.0.6
- **sense2vec**: Version higher than 2.0.0
- **pke**: Version higher than 1.8.1
- **flashtext**: Version higher than 2.7
- **transformers**: Version higher than 4.9.0
- **sentence_transformers**: Version higher than 2.0.0
- **scipy**: Version higher than 1.5.2
- **torch**: Version higher than 1.8.1+cpu
- **PyMuPDF**: Version higher than 1.18.16
- **mysql-connector-python**: Version higher than 8.0.25