QUIZ APPLICATION

CS814 Course Project Report

Submitted By:

Ayushi Jain (202CS003)
Harendra Singh Kushwaha (202CS010)

Department of Computer Science and Engineering
National Institute of Technology Karnataka
P.O. Srinivasnagar, Surathkal,
Mangalore-575025 Karnataka,
India January 2021

TABLE OF CONTENT

1. INTRODUCTION OF APPLICATION

Online Quiz examination system is a web based application for technical evaluation. Online Quiz examination system not only replaces paper work but also releases the workload of faculty. It fulfills the requirements of the institute to conduct the exam online. Students can give exams without the need of going to any physical destination.They can view the result at the same time.Thus the purpose of the site is to provide a system that saves the efforts and time of faculty**.**

The aim of this project is to computerize the existing manual system and help the examiners to save their valuable time and important data. Apart from this, data which exists in this system, will exist for a long period of time and will be easily accessible. This project helps the examiners to manage their services in a good way and provide a better service to their users.

Faculty has a privilege to create, modify and delete the test papers and its particular questions. Users can register, login and give the test with his specific id, and can see the results as well. Admin can remove users, add faculty, remove faculty.

Roles: user and faculty

Administrative role: admin

Functionalities of the project will be as following:

For faculty

- Able the examiners to punch the MCQ questions online;

- Able the examiners view users and their rankings;

- Able the examiners to see and delete quizzes uploaded by them while hiding quizzes uploaded by other faculties.

- Examiners can manage the information regarding exam;

- Correct answers will be evaluated by system (First it should be determining by examiner);

For user

- users can register and attempt quizzes online.
- Able the users to solve the questions online;
- Users can see their result after submitting the test.
- Users can see the history of the quizzes attempted by them.
- Users can see their rankings.

For admin

- admin can add examiner.
- admin can remove users.
- admin can remove examiner.

Project walk through

To design and implement this project we plan that the project support to different types of users apart from its administrative part.

Login Page :

As we start the application we enter a login page. Users, faculty and admin can use the same page to login and will be directed towards their respective page.

Register user page:

A user can also register using the register user page.

Modules available for user:

Home: here users can see the list of quizzes available and can attempt any quiz.

History: here users can see the history of all the previous attempts.

Rankings : Here user can see his/her current ranking.

Modules available for faculty:

Home: welcome page for faculty.

Users: here faculty can see all the users.

Ranking : Here faculty can see rankings of all the users.

Add quiz: here faculty can add a quiz.

Remove quiz: here faculty can see all the quizzes added by them and can remove any quiz.

Modules available for admin:

Home: welcome page for admin.

Remove Users: here admin can view all users and remove any user.

Remove Faculty: here admin can view all faculties and remove any faculty.

Add faculty: here admin can add a faculty.

2. Authorization

Need for RBAC based authorization:

In our application we have two different users, students and faculty. We have admin as administrative users. Each user has their different roles.

We need RBAC based authorization for:

-Access can be granted on a **need**-to-know basis by the admin.

-Security is more easily maintained by limiting unnecessary access to sensitive information **based** on each user's established role.

-With **Role-Based Access Control** (**RBAC**), access decisions are based on the roles.

- Data abstraction can be achieved.

-With RBAC we can implement administrative services like adding and removing roles,

assigning roles to users.

Components of RBAC:

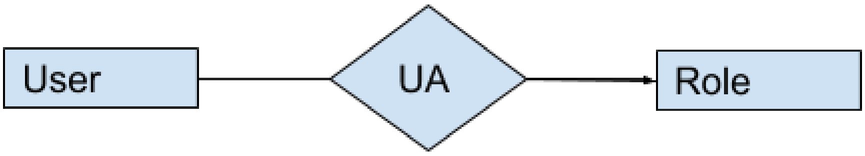
R(Roles) : Student, Teacher

P(Permissions) : Add quiz, delete quiz, view students, attempt quiz, view ranking, view history of quiz attempted,

U(Users): abc, xyz, pqr.

UA (User Role assignment) UxR :

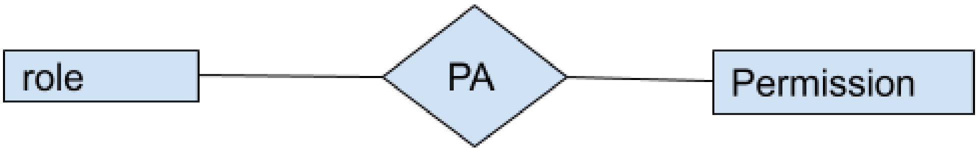
Relation UA defines User role assignment. In our implementation it is many to one relation,i.e, each user can have only one role.



User	**Role**
abc	student
xyz	student
pqr	teacher

PA(Role Permission assignment) PxR :

Relation PA defines role permission assignment.It is many to many relation. Each role can have many permissions and a permission can be assigned to many roles.



Role	**Permission**
Student	attempt quiz
Student	view ranking
Student	view attempt summary
Teacher	add quiz
Teacher	remove quiz
Teacher	view ranking
Teacher	view students

Components of administrative RBAC:

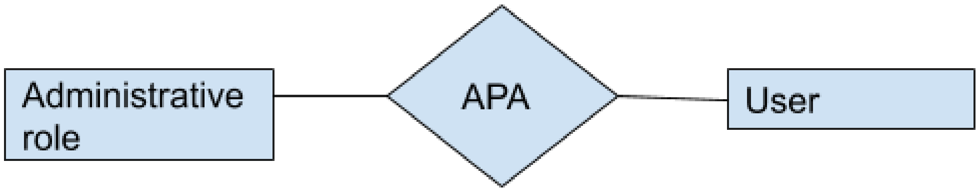
U(Users): qwe

AR(Administrative roles): admin

AP(Administrative Permission): remove student, remove teacher, add teacher.

AUA (Administrative user assignment) (U X AR):

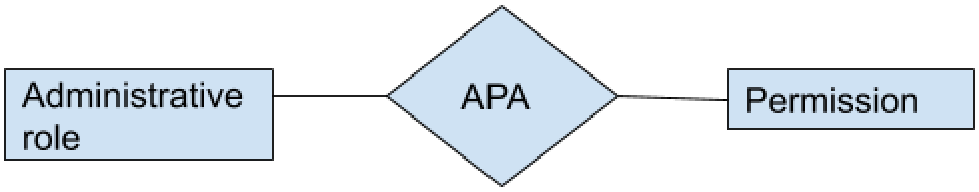
Relation AUA defines Administrative User role assignment. In our implementation it is many to many relation,i.e, each user can have only one role.



U	AR
qwe	admin

Administrative Role Permission Assignment: APA (AP x AR)

Relation APA defines administrative permission assignment. This relation specifies the relation between permission and role.It is many to many.



Administrative role	Permission
admin	remove student
admin	add faculty
admin	remove faculty

Implementation of RBAC:

We have implemented RBAC by specifying various roles as:-

1) student
2) Teacher
3) Admin

The student role is public, any user can register himself on application and get access to the permission of attempt quiz, view ranking, view attempt summary. Admin roles have the main authority of application. Admin can give a teacher role to a user and also can remove a user from the teacher role. Admin can remove a student role from a user also. If a user has a teacher role he/she can add quiz, remove quiz, view ranking, view student. And permission to one user can not be accessed by any other users. That's the way our project is based on rbac.

3. CONCLUSION:

As mentioned the project is on a PHP platform which is coded in Visual Studio Code IDE with help HTML, CSS, JavaScript and PHP and running as a web page by Apache XAMPP web server. But this project is only for MCQ test but in the future we have plans to extend it to support subjective types of questions with more functionality. We will add some more functionality in the Administrative part to record time stamps of users.

To conclude, this is a simple Online MCQ Quiz which enables a teacher to punch MCQ questions to a system which will be stored in SQL Server database and allow the student to attempt any test for once. The marks of students will be calculated according to questions they attempt and will be displayed by the system to teachers and students.

4. REFERENCES:

- https://www.tutorialspoint.com/index.htm
- https://html.com
- https://www.w3schools.com
- https://www.javatpoint.com
- https://devdocs.io/
- https://developer.mozilla.org/en-US/docs/Web/Reference
- https://stackoverflow.com/
- https://jquery.com/