

SIMPLE EXAMINATION SYSTEM

Created By: MUBASHIR

HAMAD

Degree:

BS-AI

OOP PROJECT

Introduction

The program is a FAST Examination system that allows teachers and students to interact with the system based on their respective roles. The system provides functionalities such as managing question banks, creating quizzes, viewing analytics, marking attendance, and updating passwords.

Functionalities

1. Teacher Menu: This menu is accessible to teachers and provides the following options:

- View Question Bank: Allows teachers to view the existing question bank.
- Create Quiz: Enables teachers to create quizzes using questions from the question bank.
- View Quiz Analytics: Allows teachers to view analytics related to quizzes.
- View Attendance of the Students: Provides the ability to view attendance records of students.
- Update Password: Allows teachers to update their password. ● Logout: Logs out the teacher from the system.

2. Student Menu: This menu is accessible to students and provides the following options:

- Mark Your Attendance: Allows students to mark their attendance.
- See the Courses You're Registered in: Enables students to view the courses they are registered in.
- Attempt Quiz: Allows students to attempt quizzes.
- View Result: Provides the ability to view quiz results.
- Update Password: Allows students to update their password.
- Logout: Logs out the student from the system.

Concepts Used in the System

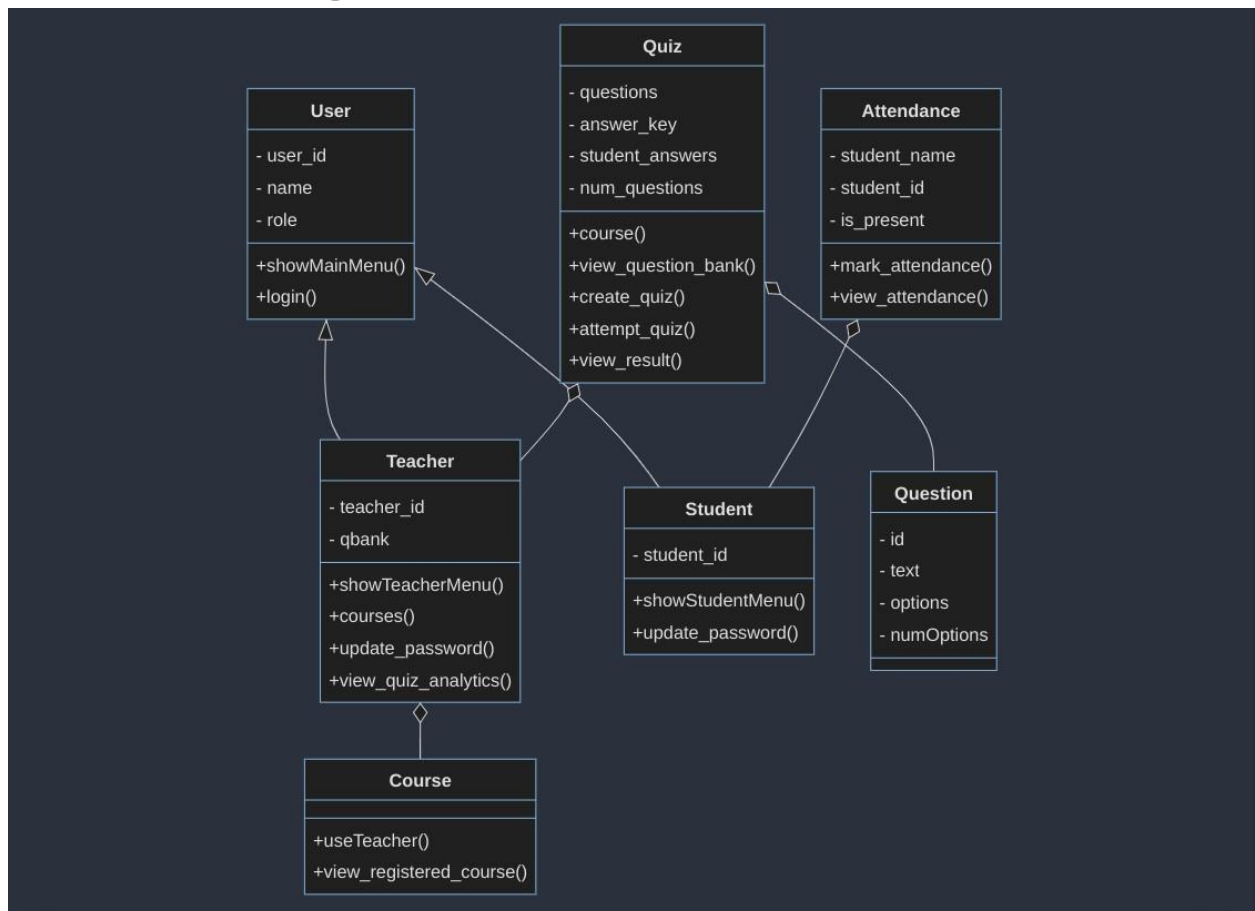
The system utilizes the following object-oriented programming concepts:

Inheritance: The classes Teacher and Student inherit from the base class User. This allows them to inherit common properties and methods from the User class.

Composition: The Quiz class has a composition relationship with the Teacher and Course classes. It uses the courses object to access the Teacher class and the view_registered_course method from the Course class.

Aggregation: The Question class contains an array of options (aggregates) represented by the options array.

Class Diagram



Manual on How to Use the System

To use the FAST Examination system, follow these steps:

- **Run the program.**

You will be presented with the main menu. Enter your choice:

Enter 1 if you are a teacher.

Enter 2 if you are a student.

Enter 3 to exit the program.

If you choose to log in as a teacher, enter your email and password when prompted. If the login is successful, you will see the teacher menu.

Choose an option from the menu by entering the corresponding number.

Follow the prompts and perform the desired actions.

To log out, choose option 6.

If you choose to log in as a student, enter your email and password when prompted. If the login is successful, you will see the student menu.

Choose an option from the menu by entering the corresponding number.

Follow the prompts and perform the desired actions.

To log out, choose option 6.

If you choose to exit the program, it will terminate.

Note: Make sure to enter valid email and password combinations for successful login.

That's the overview of the FAST Examination system. You can use the provided functionalities based on your role as a teacher or a student.

Screenshots from the program

```
class Quiz : public Teacher
{
private:
    Student student;
    Question questions[MAX_QUESTIONS];
    string answer_key[MAX_QUESTIONS];
    string student_answers[MAX_QUESTIONS];
    int num_questions;
    Course courses;
```

```
class User
{
protected:
    int user_id;
    string name;
    int role;

public:
    void showMainMenu()
    {
        cout << "\n\n    Welcome to FAST Examination system\n\n"
              << endl;
        cout << "1. Teacher Menu\n\n";
        cout << "2. Student Menu\n\n";
        cout << "3. Exit\n\n";
        cout << "Enter your choice: \n\n";
    }
```

```
class Question
{
public:
    string id;
    string text;
    string* options[MAX_OPTIONS];
    int numOptions = 0;
```

```
class Course
```

```
{
```

```
public:
```

```
    string course_name;
```

```
    void useTeacher(Teacher &teacher)
```

```
{
```

```
        teacher.courses();
```

```
}
```

```
class Attendance
{

public:
    string student_name;
    string student_id;
    bool is_present;
```

```
class Student : public User
{
private:
    int student_id;

public:

    void showStudentMenu()
    {
        cout << "\n      Student Menu\n";
        cout << "\n1. Please, Mark Your Attendance First\n";
        cout << "\n2. See the courses You're Registered in: \n";
        cout << "\n3. Attempt Quiz\n";
        cout << "\n4. View Result\n";
        cout << "\n5. Update password\n";
        cout << "\n6. Logout\n";
        cout << "\nEnter your choice: ";
    }
}
```