

BỘ GIÁO DỤC VÀ ĐÀO TẠO  
TRƯỜNG ĐẠI HỌC VĂN LANG  
KHOA KỸ THUẬT CƠ – ĐIỆN VÀ MÁY TÍNH

-----a & b-----



**NHÓM 2**

**BÁO CÁO MÔN: CƠ SỞ LẬP TRÌNH**

**Mã LHP: 222\_71ITBS10214\_03**

**Đề tài:** Xây dựng ứng dụng Luyện Thi Trắc Nghiệm

*Thành phố Hồ Chí Minh, ngày 6 tháng 6 năm 2023*

# MỤC LỤC

<b>I. Giới thiệu ứng dụng:</b>	<b>3</b>
<b>II. Phân công công việc các thành viên và thời gian thực hiện:</b>	<b>4</b>
1. Các thành viên trong nhóm	4
2. Bảng thời gian thực hiện:	4
<b>III. Thiết kế giao diện:</b>	<b>5</b>
1. Giao diện đăng nhập của người dùng:	5
2. Giao diện khi đăng nhập dưới quyền Học Viên:	6
a) Hệ thống thi trắc nghiệm:	6
b) Form giao diện vào thi:	7
3. Giao diện khi đăng nhập dưới quyền Admin:	8
a) Các quyền chức năng của Admin:	8
b) Tạo câu hỏi trắc nghiệm:	9
c) Tạo đề thi – Kiểm tra:	10
d) Xem kết quả của học viên:	11
4. Các giao diện của các form chung:	12
a) Form đổi mật khẩu:	12
b) Form xem kết quả:	13
<b>IV. Sơ đồ class:</b>	<b>14</b>
<b>V. Hiện thực chương trình quản lý có kết nối cơ sở dữ liệu:</b>	<b>15</b>
1. Form đăng nhập:	15
2. Form giao diện sinh viên:	17
a) btnThi:	20
b) btnKetQua:	20
c) btnDoiMk:	21
3. Form vào thi của sinh viên:	21
a) btnTiep:	22
b) btnQuayLai:	23
4. Form giao diện của admin:	24
5. Form tạo câu hỏi trắc nghiệm:	25
a) btnThem:	26
b) btnXoa:	27
6. Form tạo đề thi :	28
a) btnTaoDeThi:	29

b) btnTaoDeMoi: .....	30
7. Form chọn học viên để xem kết quả: .....	31
a) btnXemKetQua: .....	32
8. Form xem kết quả:.....	33
a) btnTiep: .....	33
b) btnQuayLai:.....	33
9. Form đổi mật khẩu : .....	34
a) btnDoiMK: .....	35

## **I. Giới thiệu ứng dụng:**

### ***-Nghịệp vụ bài toán:***

+ Yêu cầu ứng dụng phải cho phép học viên luyện thi được trên hệ thống; cho phép học viên xem lại kết quả luyện thi của mình.

+ Một lớp học phải học nhiều môn học và một học viên chỉ phải thi các môn học thuộc lớp học của mình.

+ Một đề thi chỉ dùng cho một lần thi của một môn học.

\*Ứng dụng quản lý thi trắc nghiệm là một phần mềm được thiết kế để hỗ trợ quản lý, tổ chức và tự động hóa quá trình làm bài và chấm điểm trong các kỳ thi trắc nghiệm. Đặc biệt, ứng dụng này có khả năng chuyên biệt và linh hoạt cho nhiều loại kỳ thi và đối tượng thí sinh khác nhau.

- Quản lý đề và câu hỏi: Ứng dụng cần có các chức năng để quản lý một kho dữ liệu câu hỏi, cho phép người dùng tạo đề thi trắc nghiệm từ các câu hỏi trong kho dữ liệu.

- Chấm điểm tự động: Ứng dụng cần cung cấp chức năng để chấm điểm tự động các bài thi trắc nghiệm, đưa ra điểm số và kết quả của từng thí sinh.

- Báo cáo kết quả: Ứng dụng cần cho phép người dùng xem báo cáo kết quả tổng thể của kỳ thi trắc nghiệm, bao gồm thông tin về thí sinh, số điểm của từng thí sinh, số lượng câu hỏi đúng và sai, và các thông tin khác.

- Ứng dụng quản lý thi trắc nghiệm cũng giúp tăng tính thời gian và hiệu quả của quy trình làm bài và chấm điểm, giảm thiểu sai sót con người. Nó cũng giúp giảm thiểu chi phí cho quản lý cuộc thi và tăng tính cạnh tranh của kỳ thi.

## **II. Phân công công việc các thành viên và thời gian thực hiện:**

### **1. Các thành viên trong nhóm**

<b>STT</b>	<b>Họ và tên</b>	<b>Mã sinh viên</b>	<b>Vị trí</b>
1	Mã Trung Hiếu	2274801030049	Trưởng nhóm
2	Trần Hoàng Phúc	2274801030119	Thành viên
3	Nguyễn Chế Anh Hào	2274801030044	Thành viên

### **2. Bảng thời gian thực hiện:**

<b>Thời gian</b>	<b>Họ và tên</b>	<b>Chức năng</b>
22/05/2023 – 03/06/2023	Mã Trung Hiếu	Giao diện chính của chương trình, các chức năng chính của chương trình( học viên thi, học viên xem KQ, admin xem KQ học viên, tạo đề thi, tạo câu hỏi)
22/05/2023 – 03/06/2023	Trần Hoàng Phúc	Quản lý database SQLTK, thiết kế form, xây dựng form cho phép đăng nhập phân quyền (học viên và admin) – đổi mật khẩu, form chức năng admin, học viên, giao diện form khi học viên vào luyện thi
22/05/2023 – 03/06/2023	Nguyễn Chế Anh Hào	Quản lý database SQLdata, lưu thông tin làm bài của học sinh thi xuống database. Truy xuất database cho phép xem kết quả của quyền học viên và admin

### III. Thiết kế giao diện:

#### 1. Giao diện đăng nhập của người dùng:



The screenshot shows a login window titled "Đăng nhập vào chương trình luyện thi trắc nghiệm". The main heading is "ĐĂNG NHẬP HỆ THỐNG". There are two radio buttons for user roles: "Học Viên" (selected) and "Admin". Below these are two input fields: "Mã Học Viên:" with a placeholder "Nhập mã học viên của bạn" and "Mật Khẩu" with a placeholder "Nhập mật khẩu của bạn". A blue "Đăng nhập" button with a right arrow icon is at the bottom.

#### Gồm:

- Nút lựa chọn đăng nhập dưới quyền Học Viên hoặc Admin.
- Hai textbox để nhập thông tin của Học Viên hoặc Admin.
- Nút đăng nhập: Lấy thông tin và đăng nhập vào hệ thống.

## 2. Giao diện khi đăng nhập dưới quyền Học Viên:

### a) Hệ thống thi trắc nghiệm:

**HỆ THỐNG THI TRẮC NGHIỆM**

**Thông tin học viên**

Mã học viên:

Họ lót:

Tên:

Giới tính:

SĐT:

**Lựa chọn**

Mã lớp học:

Môn học:

Chọn mã đề:

### Gồm:

- Một groupbox chứa toàn bộ thông tin về học viên.
- Một groupbox chứa các lựa chọn về Mã lớp, Môn học, Mã đề.
- Một button đổi mật khẩu khi nhấn vào sẽ xuất hiện form để đổi mật khẩu của mã học viên đang được chọn.
- Một button thi sau khi xét các trường hợp nhấn sẽ xuất hiện form giao diện vào thi và nếu mã đề đã được học viên thi rồi, thì sẽ xuất hiện button kết quả, nhấn vào button kết quả sẽ dẫn tới form giao diện xem kết quả.

b) Form giao diện vào thi:

The screenshot shows a web application for an exam. At the top, there are two main sections: 'Thông tin học sinh' (Student Information) and 'Thông tin đề thi' (Exam Information). The student information includes 'Mã học viên: HV02' and 'Họ tên: Trần Thị Lan'. The exam information includes 'Mã lớp: 71TO-VL', 'Môn học: Vật lý', 'Mã đề: 71VL\_009', and 'Thời gian: 5 phút'. To the right of these sections is a blue box with the text 'KIỂM TRA MÔN: Vật lý' and a timer showing 'Thời gian còn lại: 00:04:54'. Below these sections is a question area labeled 'Câu hỏi số 1' with the text '[195] Tất cả cá photon trong chân không có cùng:'. Below the question is a section for answers labeled 'Chọn đáp án' with four radio button options: 'A. Tốc độ', 'B. Bước sóng', 'C. Năng lượng', and 'D. Tần số'. To the right of the answer section is a label 'Số lượng câu hỏi: 1 / 5'. At the bottom of the interface are two buttons: 'Quay lại' (Go back) and 'Tiếp' (Next).

Gồm:

- Một groupbox chứa thông tin cơ bản về học viên.
- Một groupbox chứa thông tin về đề thi.
- Một button tiếp để load lên câu hỏi tiếp theo, đồng thời sau mỗi lần nhấn cũng load lên các lựa chọn đáp án của mã câu hỏi đó, sau khi nhấn button tiếp tới câu hỏi cuối cùng nó sẽ trở thành button nộp bài.
- Một button quay lại để quay lại câu hỏi trước đó, đồng thời sau mỗi lần nhấn cũng load lên các lựa chọn đáp án của mã câu hỏi đó.
- Một button nộp bài sau khi nhấn sẽ lưu thông tin và kết thúc bài thi.



-

### 3. Giao diện khi đăng nhập dưới quyền Admin:

#### a) Các quyền chức năng của Admin:



**CÁC QUYỀN CHỨC NĂNG CỦA ADMIN**

Tên TK Giảng Viên: tranhoangphuc

**TẠO CÂU HỎI**   **TẠO ĐỀ THI**   **XEM KẾT QUẢ CỦA HỌC VIÊN**

**Đổi mật khẩu**   **Thoát**

#### Gồm:

- Một textbox thể hiện tên tài khoản đăng nhập của giảng viên.
- Một button đổi mật khẩu khi nhấn vào sẽ xuất hiện form để đổi mật khẩu của mã học viên đang được chọn.
- Button tạo câu hỏi: nhấn vào sẽ xuất hiện form tạo câu hỏi.
- Button tạo đề thi: nhấn vào sẽ xuất hiện form tạo đề thi.
- Button xem kết quả của học viên: nhấn vào sẽ xuất hiện form xem kết quả.

**b) Tạo câu hỏi trắc nghiệm:**

**Tạo câu hỏi trắc nghiệm**

Chọn mã lớp:  Chọn chương:

Chọn môn:

**Nội dung câu hỏi**

A:

B:

C:

D:

**Chọn đáp án đúng**

☐ Đáp án A ☐ Đáp án B

☐ Đáp án C ☐ Đáp án D

**Thêm câu trắc nghiệm** **Xóa câu trắc nghiệm**

**Danh sách câu trắc nghiệm**

Mã Câu Hỏi	Nội dung câu hỏi	Đáp án A	Đáp án B	Đáp án C	Đáp án D	Đáp án đúng
------------	------------------	----------	----------	----------	----------	-------------

**Gồm:**

- Một groupbox lựa chọn: chứa các combobox lựa chọn về Mã lớp, Môn học, Chương để giáo viên có thể chọn và tạo câu hỏi cho từng chương.
- Sau khi chọn đầy đủ thông tin của groupbox lựa chọn, groupbox danh sách câu trắc nghiệm sẽ xuất hiện list những câu hỏi đã có sẵn trong chương được chọn từ trước.
- Một textbox nội dung câu hỏi, dùng để điền nội dung câu hỏi.
- Có 4 textbox của 4 đáp án A, B, C, D dùng để điền đáp án cho từng lựa chọn câu hỏi.
- Một button thêm câu trắc nghiệm: Sau khi điền đầy đủ tất cả các thông tin ràng buộc, nhấn button này sẽ thêm câu trắc nghiệm mới vào chương đang được chọn.
- Một button xóa câu trắc nghiệm: chúng ta có thể lựa chọn câu hỏi của chương được chọn được thể hiện dưới bảng listview bên dưới và xóa nó ra khỏi chương.

c) **Tạo đề thi – Kiểm tra:**

**Tạo đề thi - kiểm tra**

Chọn mã lớp:  Tên bài kiểm tra:

Chọn môn:  Số lượng câu:  Thời gian làm bài:  phút

**Chọn hình thức thi**

☐ Đề tổng hợp ☐ Đề 1 chương

**Tạo đề thi**

**Chọn chương**

**Danh sách đề thi**

Mã Đề Thi	Tên đề thi	Số lượng câu	TG làm bài	Mã lớp	Môn thi
-----------	------------	--------------	------------	--------	---------

**Gồm:**

- Một groupbox lựa chọn: chứa các combobox lựa chọn về Mã lớp, Môn học và những textbox để điền tên bài kiểm tra, số lượng câu, thời gian làm bài cho đề thi được tạo.
- Một groupbox chọn hình thức thi: chứa 2 lựa chọn để tạo dạng thi cho đề thi được tạo.
- Sau khi chọn được hình thức thi, groupbox chọn chương sẽ xuất hiện và đồng thời cũng thể hiện toàn bộ các chương thuộc môn học đó.
- Một groupbox danh sách đề thi: chứa listview thể hiện đầy đủ tất cả các thông tin về đề thi đã có sẵn của môn học đó, cũng như thông tin về đề thi vừa được tạo.
- Một button tạo đề thi: sau khi xét tất cả các thông tin được ràng buộc, nhấn vào button tạo đề thi sẽ thực hiện tạo đề mới cho môn học được chọn.
- Một button tạo đề mới: sẽ xuất hiện sau khi nhấn button tạo đề thi thành công.

**d) Xem kết quả của học viên:**

**CÁC QUYỀN CHỨC NĂNG CỦA ADMIN > Xem Kết Quả Học Viên**

**Lựa chọn**

Chọn Mã Lớp

Chọn Môn Học

Chọn mã đề:

Chọn mã học viên:

**Thông tin học viên**

Họ tên học viên:

Giới tính:

Số điện thoại:

**Xem kết quả**

**Gồm:**

- Một groupbox lựa chọn: chứa các combobox để giáo viên có thể lựa chọn các thông tin về Mã lớp, Môn học, Mã đề, Mã học viên mà giáo viên muốn xem kết quả.
- Một groupbox thông tin học viên: chứa các thông tin cơ bản của học viên được chọn bên grbox lựa chọn về Họ tên, Giới tính, Số điện thoại.
- Button xem kết quả: sau khi xét đầy đủ tất cả các ràng buộc, nhấn button này sẽ xuất hiện lên form xem kết quả cho giáo viên.

-

#### 4. Các giao diện của các form chung:

##### a) Form đổi mật khẩu:

**ĐỔI MẬT KHẨU**

Mã học viên:  
HV01

Mật khẩu cũ: \*

Mật khẩu mới: \*

☐ Hiển thị mật khẩu

Xác nhận mật khẩu mới: \*

**Đổi mật khẩu** **Thoát**

##### Gồm:

- Một textbox mã học viên: Thể hiện sẵn mã học viên đã được đăng nhập.
- Một textbox nhập mật khẩu cũ.
- Một textbox nhập mật khẩu mới.
- Một textbox nhập xác nhận mật khẩu mới.
- Một button đổi mật khẩu: Xét các trường hợp của quá trình đổi mật khẩu, sau đó truy xuất và thực hiện quá trình đổi mật khẩu.

**b) Form xem kết quả:**

Thông tin học sinh

Mã học viên: **HV01**

Họ tên: **Nguyễn Văn Anh**

Thông tin đề thi

Mã lớp: **71TO-VL**

Mã đề: **71VL\_010**

Môn học: **Vật lý**

Thời gian: **5 phút**

KIỂM TRA MÔN: Vật lý

Tổng điểm: **0**

Số câu đúng: **0/5**

Câu hỏi số 1

[195] Tất cả các photon trong chân không có cùng:

Chọn đáp án

☐ A.Tốc độ

☐ B. Bước sóng

☐ C.Năng lượng

☐ D.Tần số

Câu hỏi số: 1 / 5

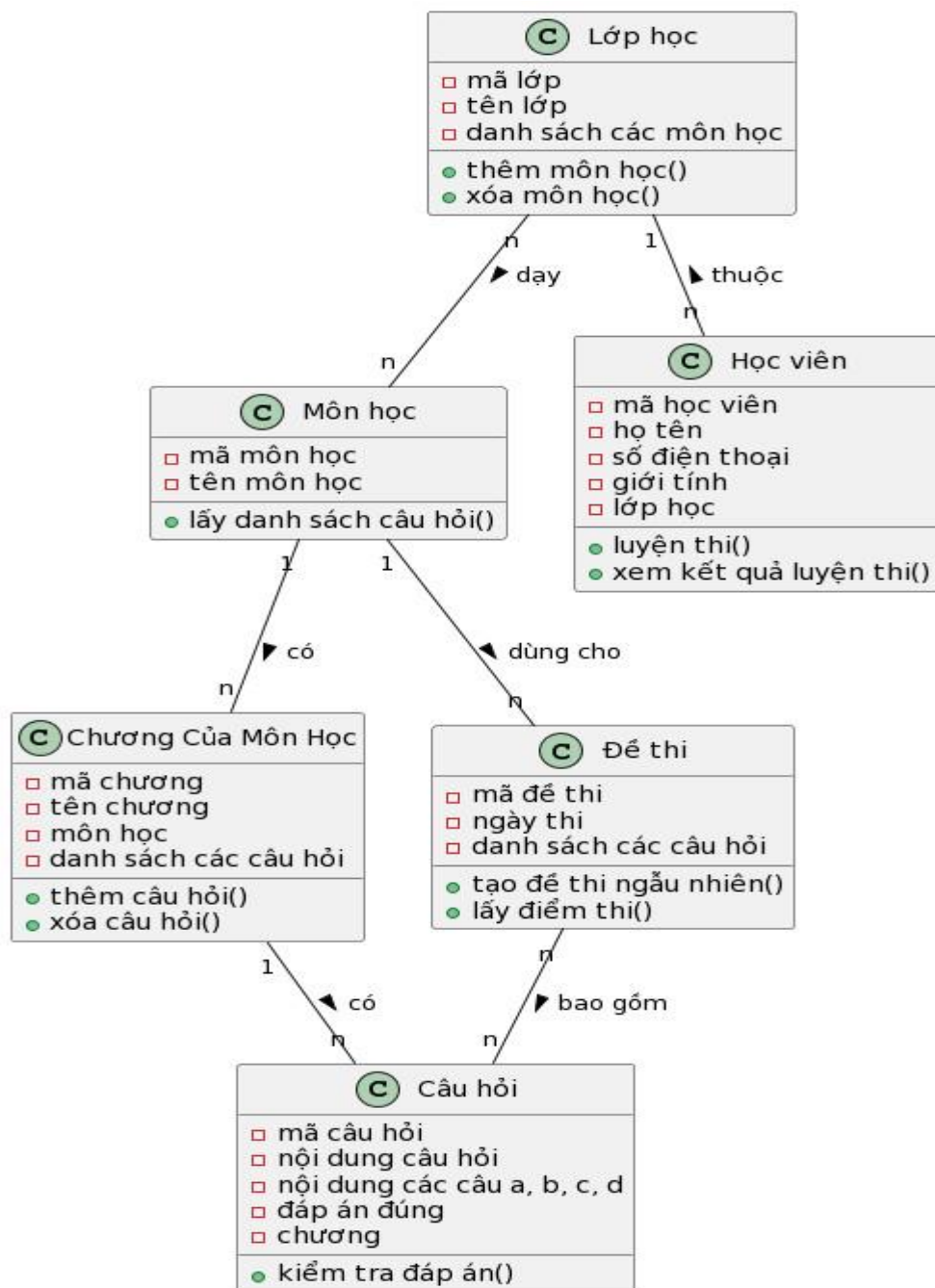
Ghi chú: Học viên không chọn đáp án cho câu này

Quay lại

Tiếp

**\*Có các chức năng giống với form vào thi.**

#### IV. Sơ đồ class:



## V. Hiện thực chương trình quản lý có kết nối cơ sở dữ liệu:

### 1. Form đăng nhập:

Đăng nhập vào chương trình luyện thi trắc nghiệm

## ĐĂNG NHẬP HỆ THỐNG

☒ Học Viên ☐ Admin

Mã Học Viên:

Mật Khẩu

- Khi nhấn vào btnLogin thì nó sẽ truy xuất dữ liệu từ database, sau đó thì sẽ xét điều kiện để thông báo đăng nhập thành công hoặc thất bại.

**\*Chú ý: Tài khoản này đã được cấp từ trước**



```

private void btnLogin_Click(object sender, EventArgs e)
{
    var username = txtUsername.Text.Trim();
    var password = txtPassword.Text.Trim();

    if (string.IsNullOrEmpty(username))
    {
        MessageBox.Show("Hãy nhập tài khoản/mã HV của bạn");
        txtUsername.Focus();
        return;
    }

    if (string.IsNullOrEmpty(password))
    {
        MessageBox.Show("Hãy nhập mật khẩu của bạn");
        txtPassword.Focus();
        return;
    }

    if (rdHocVien.Checked)
    {
        SqlConnection conn = new SqlConnection(@"Data Source=LAPTOP-SJB31CGH\SQLEXPRESS;Initial Catalog=QUANLYTK;Integrated Security=True");
        conn.Open();
        string mhV = txtUsername.Text.Trim();
        string mk = txtPassword.Text.Trim();
        string sql = "Select * from NguoIDung where MaHV = '" + mhV + "' and MatKhau = '" + mk + "' ";
        SqlCommand cmd = new SqlCommand(sql, conn);
        SqlDataReader data = cmd.ExecuteReader();
        if (data.Read() == true)
        {
            MessageBox.Show("đăng nhập thành công");

            HocVienForm hvf = new HocVienForm();
            hvf.temp = txtUsername.Text;
            hvf.ShowDialog();
            txtPassword.Clear();
        }
        else
        {
            MessageBox.Show("đăng nhập thất bại");
            txtUsername.Focus();
            txtUsername.Clear();
            txtPassword.Clear();
        }
    }
    else
    {
        SqlConnection conn = new SqlConnection(@"Data Source=LAPTOP-SJB31CGH\SQLEXPRESS;Initial Catalog=QUANLYTK;Integrated Security=True");
        conn.Open();
        string tk = txtUsername.Text.Trim();
        string mk = txtPassword.Text.Trim();
        string sql = "Select * from GiaoVien where TaiKhoan = '" + tk + "' and MatKhau = '" + mk + "' ";
        SqlCommand cmd = new SqlCommand(sql, conn);
        SqlDataReader data = cmd.ExecuteReader();
        if (data.Read() == true)
        {
            MessageBox.Show("đăng nhập thành công");

            Admin admin = new Admin();
            admin.temp = txtUsername.Text.Trim();
            admin.ShowDialog();
            txtPassword.Clear();
        }
        else
        {
            MessageBox.Show("đăng nhập thất bại");
            txtUsername.Focus();
            txtUsername.Clear();
            txtPassword.Clear();
        }
    }
}

```

## 2. Form giao diện sinh viên:

**HỆ THỐNG THI TRẮC NGHIỆM**

**Thông tin học viên**

Mã học viên:

Họ lót:

Tên:

Giới tính:

SĐT:

**Lựa chọn**

Mã lớp học:

Môn học:

Chọn mã đề:

- Thông qua Form\_Load, form sẽ truy xuất dữ liệu từ database và load thông tin lên form.

```

private void HocVien_Load(object sender, EventArgs e)
{
    cbMonHoc.Items.Clear();
    cbMonHoc.ResetText();
    if (connection == null)
    {
        connection = new SqlConnection(strConnect);
        if (connection.State == ConnectionState.Closed)
            connection.Open();
        string sql = "SELECT MALOP, TENLOP FROM LopHoc";

        SqlCommand sqlCmd = new SqlCommand(sql);
        sqlCmd.Connection = connection;
        SqlDataReader read = sqlCmd.ExecuteReader();
        while (read.Read())
        {
            LopHoc s = new LopHoc();
            s.MaLop = read.GetString(0).Trim();
            s.TenLop = read.GetString(1).Trim();
        }
        read.Close();
        txtMaHV.Text = temp;

        string sqls = "SELECT MAHV, HOLOT, TEN, PHAI, SODT, MALOP " +
            "FROM HOCVIEN WHERE MAHV = @maHV";
        SqlParameter prMaHV = new SqlParameter("@maHV", SqlDbType.Char);
        prMaHV.Value = txtMaHV.Text;

        SqlCommand sqlCmds = new SqlCommand(sqls);
        sqlCmds.Parameters.Add(prMaHV);
        sqlCmds.Connection = connection;
        SqlDataReader readers = sqlCmds.ExecuteReader();
        while (readers.Read())
        {
            string maHV = readers.GetString(0);
            string hoLot = readers.GetString(1);
            string ten = readers.GetString(2);
            string phai = readers.GetString(3);
            string soDT = readers.GetString(4);
            string maLop = readers.GetString(5);

            txtMaHV.Text = maHV.Trim();
            txtHoLot.Text = hoLot.Trim();
            txtTen.Text = ten;
            txtGioiTinh.Text = phai.Trim();
            txtSĐT.Text = soDT.Trim();
            txtMaLop.Text = maLop.Trim();
        }
        readers.Close();

        string sqlHV = $"select * from LOPHOC where Malop = '{txtMaLop.Text}'";
        SqlCommand sqlHVCmd = new SqlCommand(sql);
        sqlHVCmd.Connection = connection;
        SqlDataReader reader = sqlHVCmd.ExecuteReader();
        while (reader.Read())
        {
            string maLop = reader.GetString(0).Trim();
            string tenLop = reader.GetString(1).Trim();
            lopHoc = new LopHoc(maLop, tenLop);
        }
    }
}

```

```

reader.Close();
//
string Count = $"SELECT COUNT (MaMonHoc) FROM LOP_MONHOC where LOP_MONHOC.MALOP = '{txtMaLop.Text}'";
int countNum = 0;
SqlCommand sqlCount = new SqlCommand(Count);
sqlCount.Connection = connection;
SqlDataReader readerCount = sqlCount.ExecuteReader();
while (readerCount.Read())
{
    countNum = readerCount.GetInt32(0);
}
readerCount.Close();
for (int i = 1; i <= countNum; i++)
{
    string getMonHoc =
        $"SELECT MaMonHoc FROM" +
        $"\\n(SELECT MaMonHoc, ROW_NUMBER() OVER (ORDER BY MaMonHoc) AS RowNum FROM LOP_MONHOC WHERE LOP_MONHOC.MALOP = '{txtMaLop.Text}') AS T" +
        $"\\nWHERE RowNum = {i}";
    SqlCommand GETsqlCmdMonHoc = new SqlCommand(getMonHoc);
    GETsqlCmdMonHoc.Connection = connection;
    SqlDataReader GETreaderMonHoc = GETsqlCmdMonHoc.ExecuteReader();
    string GetTenMonHoc = "";
    while (GETreaderMonHoc.Read())
    {
        GetTenMonHoc = GETreaderMonHoc.GetString(0);
    }
    GETreaderMonHoc.Close();
    string sqlTenMonHoc = $"select * from monhoc where MaMonHoc = '{GetTenMonHoc}'";
    SqlCommand sqlCmdTenMonHoc = new SqlCommand(sqlTenMonHoc);
    sqlCmdTenMonHoc.Connection = connection;
    SqlDataReader readerTenMonHoc = sqlCmdTenMonHoc.ExecuteReader();
    while (readerTenMonHoc.Read())
    {
        string maMonHoc = readerTenMonHoc.GetString(0);
        string tenMonHoc = readerTenMonHoc.GetString(1);

        MonHoc = new MonHoc(maMonHoc, tenMonHoc);
        cbMonHoc.Items.Add(MonHoc);
    }
    readerTenMonHoc.Close();
}
}

```

a) **btnThi:**

```
private void btnThi_Click(object sender, EventArgs e)
{
    if (cbMonHoc.Items == null || cbMonHoc.SelectedIndex == -1)
    {
        MessageBox.Show("Chưa chọn môn thi", "Thông báo");
        return;
    }
    if (cbMaDe.Items == null || cbMaDe.SelectedIndex == -1)
    {
        MessageBox.Show("Chưa chọn mã đề", "Thông báo");
        return;
    }
    //Còn tạo điều kiện cho phép thi
    bool checkFlag = false;
    string sqlCheck = $"select IDKETQUA from KETQUA where MAHV = '{txtMaHV.Text.Trim()}' and MADETHI = '{cbMaDe.Text.Trim()}'";
    SqlCommand sqlCheckCmd = new SqlCommand(sqlCheck);
    sqlCheckCmd.Connection = connection;
    SqlDataReader readerCheck = sqlCheckCmd.ExecuteReader();
    while (readerCheck.Read())
    {
        string check = readerCheck.GetString(0).Trim();
        if (!(String.IsNullOrEmpty(check)) && !(String.IsNullOrEmpty(check)))
        {
            checkFlag = true;
        }
    }
    readerCheck.Close();
    if (checkFlag)
    {
        MessageBox.Show("Bạn đã làm đề thi này trước đó", "Thông báo");
        btnXemKetQua.Visible = true;
        return;
    }

    MessageBox.Show("Bắt đầu thi");
    VaoThi vt = new VaoThi();
    vt.maHV = txtMaHV.Text.Trim();
    vt.hoTen = txtHoTen.Text.Trim() + " " + txtTen.Text.Trim();
    vt.maLop = txtMaLop.Text.Trim();
    MonHoc = (MonHoc)cbMonHoc.SelectedItem;
    vt.monHoc = MonHoc.TenMonHoc.Trim();
    vt.maDe = cbMaDe.Text.Trim();
    vt.ShowDialog();
}
```

b) **btnKetQua:**

```
private void btnXemKetQua_Click(object sender, EventArgs e)
{
    XemKetQuaHV xemKetQua = new XemKetQuaHV();
    xemKetQua.maHV = txtMaHV.Text.Trim();
    xemKetQua.hoTen = txtHoTen.Text.Trim() + " " + txtTen.Text.Trim();
    xemKetQua.maLop = txtMaLop.Text.Trim();
    MonHoc = (MonHoc)cbMonHoc.SelectedItem;
    xemKetQua.monHoc = MonHoc.TenMonHoc.Trim();
    xemKetQua.maDe = cbMaDe.Text.Trim();
    xemKetQua.ShowDialog();
}
```

c) btnDoiMk:

```
private void btnDoiMK_Click(object sender, EventArgs e)
{
    DoiMatKhau dmk = new DoiMatKhau();
    dmk.maHV = txtMaHV.Text;
    dmk.ShowDialog();
}
```

### 3. Form vào thi của sinh viên:

**Thông tin học sinh**

Mã học viên: HV02  
Họ tên: Trần Thị Lan

**Thông tin đề thi**

Mã lớp: 71TO-VL Môn học: Vật lý  
Mã đề: 71VL\_010 Thời gian: 5 phút

**KIỂM TRA MÔN: Vật lý**

Thời gian còn lại: 00:04:54

**Câu hỏi số 1**

[195] Tất cả các photon trong chân không có cùng:

**Chọn đáp án**

☐ A. Tốc độ  
☐ B. Bước sóng  
☐ C. Năng lượng  
☐ D. Tần số

Số lượng câu hỏi: 1 / 5

Quay lại Tiếp

- Thông qua Form\_Load, form sẽ truy xuất dữ liệu từ database và load thông tin lên form.

a) **btnTiep:**

```
//Xử lý xuống database kết quả
string sqlInsertKetQua = "INSERT INTO KETQUA(IDKETQUA, MADETHI, MAHV, TONGSODIEM) VALUES (@idketqua, @madethi, @mahv, @diem)";

SqlParameter prID = new SqlParameter("@idketqua", SqlDbType.Char);
SqlParameter prMaDeThi = new SqlParameter("@madethi", SqlDbType.Char);
SqlParameter prMaHV = new SqlParameter("@mahv", SqlDbType.Char);
SqlParameter prDiem = new SqlParameter("@diem", SqlDbType.Float);
prID.Value = txtMaDe.Text.Trim() + "_" + txtMaHV.Text;
prMaDeThi.Value = txtMaDe.Text.Trim();
prMaHV.Value = txtMaHV.Text.Trim();
prDiem.Value = tongDiem;
SqlCommand sqlCmd = new SqlCommand(sqlInsertKetQua);
sqlCmd.Connection = connection;
sqlCmd.Parameters.Add(prID);
sqlCmd.Parameters.Add(prMaDeThi);
sqlCmd.Parameters.Add(prMaHV);
sqlCmd.Parameters.Add(prDiem);
if (sqlCmd.ExecuteNonQuery() > 0)
{
    MessageBox.Show("Nộp bài thành công!", "Thông báo");
}
else
{
    MessageBox.Show("Nộp bài thất bại!", "Thông báo");
}

string[] maCauHoi = new string[countSoCauHoi];
string[] cauTraLoi = new string[countSoCauHoi];
for (int i = 1; i <= countSoCauHoi; i++)
{
    string sqlGetCauHoi = $"select MaCauHoi,CauTraLoi from cacCauHoi where MaCauHoi = ((SELECT MaCauHoi FROM (SELECT MaCauHoi, ROW_NUMBER() OVER
    SqlCommand sqlGetCauHoiCmd = new SqlCommand(sqlGetCauHoi);
    sqlGetCauHoiCmd.Connection = connection;
    SqlDataReader readerGetCH = sqlGetCauHoiCmd.ExecuteReader();
    while (readerGetCH.Read())
    {
        maCauHoi[i - 1] = readerGetCH.GetString(0);
        cauTraLoi[i - 1] = readerGetCH.GetString(1);
    }
    readerGetCH.Close();

    string str = "insert into CT_KETQUA(IDKETQUA, MACAUHOI, CAUTRALOI, CauTraLoiCuaHV) values(@idketqua_CT, @macauhoi, @cautraloi, @HVTraLoi)";
    SqlParameter prID_CT = new SqlParameter("@idketqua_CT", SqlDbType.Char, 20); // -OK
    SqlParameter prMaCauHoi = new SqlParameter("@macauhoi", SqlDbType.Char, 10);
    SqlParameter prCauTraLoi = new SqlParameter("@cautraloi", SqlDbType.NVarChar, 200);
    SqlParameter prHVTraLoi = new SqlParameter("@HVTraLoi", SqlDbType.Char, 10);

    prID_CT.Value = txtMaDe.Text.Trim() + " " + txtMaHV.Text; // - OK
    prMaCauHoi.Value = maCauHoi[i - 1].Trim();
    prCauTraLoi.Value = cauTraLoi[i - 1].Trim();
    prHVTraLoi.Value = SelectedAnswer[1, i].ToString().Trim();
    SqlCommand sqlCmdCTKQ = new SqlCommand(str);
    sqlCmdCTKQ.Connection = connection;
    sqlCmdCTKQ.Parameters.Add(prID_CT);
    sqlCmdCTKQ.Parameters.Add(prMaCauHoi);
    sqlCmdCTKQ.Parameters.Add(prCauTraLoi);
    sqlCmdCTKQ.Parameters.Add(prHVTraLoi);

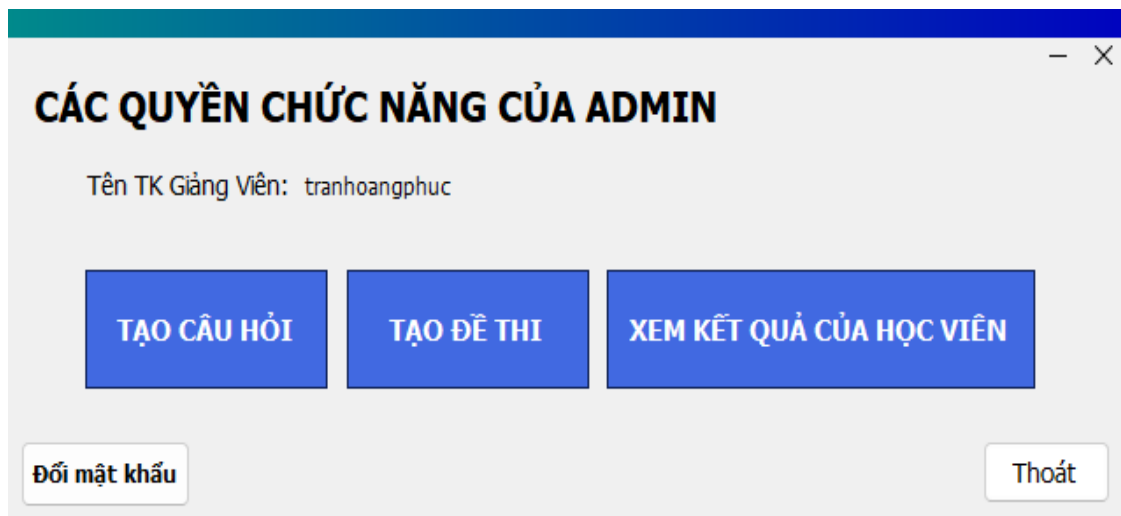
    if (sqlCmdCTKQ.ExecuteNonQuery() > 0)
    {
        ;
    }
}
this.Close();
```

## b) btnQuayLai:

```
string sqlShowCauHoi = $"select * from cacCauHoi where MaCauHoi = ((SELECT MaCauHoi FROM (SELECT MaCauHoi, ROW_NUMBER() * +  
$ "OVER (ORDER BY MaCauHoi) AS RowNum FROM CHITIET_DETHI WHERE CHITIET_DETHI.MADETHI = '{txtMaDe.Text.Trim()}'") AS T WHERE RowNum = {countButton}))";  
  
SqlCommand sqlCmdShowCauHoi = new SqlCommand(sqlShowCauHoi);  
sqlCmdShowCauHoi.Connection = connection;  
SqlDataReader readerShowCauHoi = sqlCmdShowCauHoi.ExecuteReader();  
while (readerShowCauHoi.Read())  
{  
    cauHoi.MaCauHoi = readerShowCauHoi.GetString(0).Trim();  
    cauHoi.NoiDungCauHoi = readerShowCauHoi.GetString(1);  
    cauHoi.PanA = readerShowCauHoi.GetString(2);  
    cauHoi.PanB = readerShowCauHoi.GetString(3);  
    cauHoi.PanC = readerShowCauHoi.GetString(4);  
    cauHoi.PanD = readerShowCauHoi.GetString(5);  
    cauHoi.Answer = readerShowCauHoi.GetString(6).Trim();  
  
    cauHoi.MaChuong = readerShowCauHoi.GetString(7).Trim();  
    //MessageBox.Show(cauHoi.MaChuong);  
    //MessageBox.Show(cauHoi.Answer.Substring(cauHoi.Answer.Length - 1, 1));  
    if (CorrectAnswer[i, countButton] < 1)  
    {  
        if (cauHoi.Answer.Substring(cauHoi.Answer.Length - 1, 1) == "A")  
        {  
            CorrectAnswer[i, countButton] = 1;  
        }  
        if (cauHoi.Answer.Substring(cauHoi.Answer.Length - 1, 1) == "B")  
        {  
            CorrectAnswer[i, countButton] = 2;  
        }  
        if (cauHoi.Answer.Substring(cauHoi.Answer.Length - 1, 1) == "C")  
        {  
            CorrectAnswer[i, countButton] = 3;  
        }  
        if (cauHoi.Answer.Substring(cauHoi.Answer.Length - 1, 1) == "D")  
        {  
            CorrectAnswer[i, countButton] = 4;  
        }  
        //txtCheck.Text += "[" + cauHoi.MaCauHoi + " ] " + CorrectAnswer[i, countButton];  
    }  
    txtCauHoi.Text = "[" + cauHoi.MaCauHoi + " ] " + cauHoi.NoiDungCauHoi;  
    rdCauA.Text = cauHoi.PanA;  
    rdCauB.Text = cauHoi.PanB;  
    rdCauC.Text = cauHoi.PanC;  
    rdCauD.Text = cauHoi.PanD;  
}  
readerShowCauHoi.Close();
```



#### 4. Form giao diện của admin:



**CÁC QUYỀN CHỨC NĂNG CỦA ADMIN**

Tên TK Giảng Viên: tranhoangphuc

**TẠO CÂU HỎI**   **TẠO ĐỀ THI**   **XEM KẾT QUẢ CỦA HỌC VIÊN**

**Đổi mật khẩu**   **Thoát**

- Form này có chức năng để lựa chọn và bật các form chức năng khác.

```
private void btnTaoDeThi_Click(object sender, EventArgs e)
{
    TaoDeThi_AdminForm taoDeThi = new TaoDeThi_AdminForm();
    taoDeThi.ShowDialog();
}

1 reference
private void btnTaoCauHoi_Click(object sender, EventArgs e)
{
    AdminForm adminForm = new AdminForm();
    adminForm.ShowDialog();
}

1 reference
private void btnXemKQAdmin_Click(object sender, EventArgs e)
{
    ChonMaHV_XemKQAdmin beforeKQ = new ChonMaHV_XemKQAdmin();
    beforeKQ.ShowDialog();
}

1 reference
private void btnDoiMK_Click(object sender, EventArgs e)
{
    DoiMatKhau dmk = new DoiMatKhau();
    dmk.maHV = lblTenGV.Text;
    dmk.ShowDialog();
}
```

## 5. Form tạo câu hỏi trắc nghiệm:

Tạo câu hỏi trắc nghiệm

Chọn mã lớp:

Chọn chương:

Chọn môn:

Nội dung câu hỏi

A:

B:

C:

D:

Chọn đáp án đúng

☐ Đáp án A

☐ Đáp án B

☐ Đáp án C

☐ Đáp án D

Thêm câu trắc nghiệm

Xóa câu trắc nghiệm

Danh sách câu trắc nghiệm

a) btnThem:

```
string sql = "insert into CacCauHoi(MaCauHoi,NoiDungCauHoi,DapAnA,DapAnB,DapAnC,DapAnD,CauTraLoi,MaChuong) values"
"(@macauhoi, @NDCH, @PANa, @PANb, @PANc, @PANd, @cautraloi, @machuong)";
SqlParameter prMaCauHoi = new SqlParameter("@macauhoi", SqlDbType.Char);
SqlParameter prNoiDungCauHoi = new SqlParameter("@NDCH", SqlDbType.NVarChar);
SqlParameter prPANa = new SqlParameter("@PANa", SqlDbType.NVarChar);
SqlParameter prPANb = new SqlParameter("@PANb", SqlDbType.NVarChar);
SqlParameter prPANc = new SqlParameter("@PANc", SqlDbType.NVarChar);
SqlParameter prPANd = new SqlParameter("@PANd", SqlDbType.NVarChar);
SqlParameter prCauTraLoi = new SqlParameter("@cautraloi", SqlDbType.NVarChar);
SqlParameter prMaChuong = new SqlParameter("@machuong", SqlDbType.Char);

//Thêm giá trị cho tham số
int test = int.Parse(maCH.ToString());
prMaCauHoi.Value = (test + 1).ToString();
prNoiDungCauHoi.Value = txtNDCauHoi.Text;
prPANa.Value = txtPANa.Text;
prPANb.Value = txtPANb.Text;
prPANc.Value = txtPANc.Text;
prPANd.Value = txtPANd.Text;
if (rdPANa.Checked)
{
    prCauTraLoi.Value = "A";
}
if (rdPANb.Checked)
{
    prCauTraLoi.Value = "B";
}
if (rdPANc.Checked)
{
    prCauTraLoi.Value = "C";
}
if (rdPANd.Checked)
{
    prCauTraLoi.Value = "D";
}
prMaChuong.Value = ChuongMonHoc[cbChuongMonhoc.SelectedIndex + 1];
SqlCommand sqlCmd = new SqlCommand(sql);
sqlCmd.Connection = connection;
sqlCmd.Parameters.Add(prMaCauHoi);
sqlCmd.Parameters.Add(prNoiDungCauHoi);
sqlCmd.Parameters.Add(prPANa);
sqlCmd.Parameters.Add(prPANb);
sqlCmd.Parameters.Add(prPANc);
sqlCmd.Parameters.Add(prPANd);
sqlCmd.Parameters.Add(prCauTraLoi);
sqlCmd.Parameters.Add(prMaChuong);

int n = sqlCmd.ExecuteNonQuery();
string MaCauHoi = (test + 1).ToString();
string NoiDungCauHoi = txtNDCauHoi.Text;
string PanA = txtPANa.Text;
string PanB = txtPANb.Text;
string PanC = txtPANc.Text;
string PanD = txtPANd.Text;

string Answer = prCauTraLoi.Value.ToString();
string MaChuong = ChuongMonHoc[cbChuongMonhoc.SelectedIndex + 1];
CauHoi cauHoi = new CauHoi(MaCauHoi, NoiDungCauHoi, PanA, PanB, PanC, PanD, Answer, MaChuong);
string[] obj = new string[] {cauHoi.MaCauHoi, cauHoi.NoIDungCauHoi, cauHoi.PanA,
    cauHoi.PanB, cauHoi.PanC, cauHoi.PanD, cauHoi.Answer.Substring(cauHoi.Answer.Length-1,1)};
ListViewItem item = new ListViewItem(obj);
if (n > 0)
{
    // ...
}
```

b) btnXoa:

```
private void btnXoa_Click(object sender, EventArgs e)
{
    if (cbMaLopHoc.Items == null || cbMaLopHoc.SelectedIndex == -1)
    {
        MessageBox.Show("Chưa chọn mã lớp", "Thông báo");
        cbMaLopHoc.Focus();
        return;
    }
    if (cbMonHoc.Items == null || cbMonHoc.SelectedIndex == -1)
    {
        MessageBox.Show("Chưa chọn môn thi", "Thông báo");
        cbMonHoc.Focus();
        return;
    }
    if (cbChuongMonhoc.Items == null || cbChuongMonhoc.SelectedIndex == -1)
    {
        MessageBox.Show("Chưa chọn chương", "Thông báo");
        cbMonHoc.Focus();
        return;
    }
    if (lsvDSCauTracNghiem.SelectedItems == null || lsvDSCauTracNghiem.TabIndex < 0)
        return;

    try
    {
        if (connection == null)
            connection = new SqlConnection(strConnect);
        if (connection.State == ConnectionState.Closed)
            connection.Open();
        string sql = " DELETE FROM CacCauHoi WHERE MaCauHoi = @MACAUHOI";

        //khai bao cac tham so
        SqlParameter prMaCauHoi = new SqlParameter("@MACAUHOI", SqlDbType.Int);

        //gan gia tri cho tham so
        ListViewItem item = lsvDSCauTracNghiem.SelectedItems[0];
        CauHoi cauHoi = new CauHoi();
        cauHoi.MaCauHoi = item.SubItems[0].Text;
        int index = lsvDSCauTracNghiem.SelectedItems[0].Index;
        prMaCauHoi.Value = cauHoi.MaCauHoi;

        //thuc thi query
        SqlCommand sqlCmd = new SqlCommand(sql);
        sqlCmd.Parameters.Add(prMaCauHoi);

        sqlCmd.Connection = connection;

        int n = sqlCmd.ExecuteNonQuery();
        if (n > 0)
        {
            MessageBox.Show("Xóa thành công", "Thông báo");
            lsvDSCauTracNghiem.Items.RemoveAt(index);
        }
        else
        {
            MessageBox.Show("Chưa thêm được dữ liệu");
        }
    }
    catch (Exception ex)
    {
    }
}
```

## 6. Form tạo đề thi :

Tạo đề thi - kiểm tra

Tạo đề mới

Chọn mã lớp:

Tên bài kiểm tra:

Chọn môn:

Số lượng câu:

Thời gian làm bài:  phút

Chọn hình thức thi

☐ Đề tổng hợp ☐ Đề 1 chương

Tạo đề thi

Chọn chương

Chương 1

Chương 2

Chương 3

Chương 4

Chương 5

Chương 6

Chương 7

Chương 8

Chương 9

Chương 10

Chương 11

Chương 12

Chương 13

Chương 14

Chương 15

Chương 16

Chương 17

Chương 18

Danh sách đề thi

a) **btnTaoDeThi:**

```
        if (sqlCTDECmd.ExecuteNonQuery() > 0)
        {
            //MessageBox.Show("Nạp dữ liệu vào để thi thành công!", "Thông báo");
        }
        else
        {
            MessageBox.Show("Nạp dữ liệu vào để thi thất bại! Thử lại.", "Thông báo");
        }
    }

    string tenDeThi = txtTenBaiKiemTra.Text;
    string thoiGianLamBai = txtThoiGian.Text;
    string maLop = maLopSelected.Trim();
    string monThi = monhoc.TenMonHoc.Trim();
    string[] obj = new string[] { maDeThi, tenDeThi, soLuongCauHoi + "", thoiGianLamBai + " phút", maLop, monThi };
    ListViewItem item = new ListViewItem(obj);

    string sql = "INSERT INTO DETHI_QLADMIN(MADETHI, TENDETHI, SOLUONGCAUHOI, TGLAMBAI, MALOP, MONTHI) VALUES " +
        "(@madethi, @tendethi, @soluongcauhoi, @tglambai, @malop, @monthi)";
    SqlParameter prMaDeThi_QL = new SqlParameter("@madethi", SqlDbType.Char, 10);
    SqlParameter prTenDeThi = new SqlParameter("@tendethi", SqlDbType.NVarChar);
    SqlParameter prSLCH = new SqlParameter("@soluongcauhoi", SqlDbType.Char);
    SqlParameter prTGLambai = new SqlParameter("@tglambai", SqlDbType.Char);
    SqlParameter prMaLop = new SqlParameter("@malop", SqlDbType.Char, 12);
    SqlParameter prMonThi_QL = new SqlParameter("@monthi", SqlDbType.NVarChar);
    prMaDeThi_QL.Value = maDeThi.Trim();
    prTenDeThi.Value = tenDeThi.Trim();
    prSLCH.Value = soLuongCauHoi + "";
    prTGLambai.Value = thoiGianLamBai.Trim();
    prMaLop.Value = maLop.Trim();
    prMonThi_QL.Value = monThi.Trim();
    SqlCommand sqlCmd = new SqlCommand(sql);
    sqlCmd.Connection = connection;
    sqlCmd.Parameters.Add(prMaDeThi_QL);
    sqlCmd.Parameters.Add(prTenDeThi);
    sqlCmd.Parameters.Add(prSLCH);
    sqlCmd.Parameters.Add(prTGLambai);
    sqlCmd.Parameters.Add(prMaLop);
    sqlCmd.Parameters.Add(prMonThi_QL);

    if (sqlCmd.ExecuteNonQuery() > 0)
    {
        MessageBox.Show("Thêm vào DS để thi thành công", "Thông báo");
        lsvDSDeThi.Items.Add(item);
        btnTaoDeThi.Enabled = false;
        btnTaoDeThi.BackColor = Color.CadetBlue;
        cbMaLopHoc.Enabled = false;
        cbMonHoc.Enabled = false;
        groupBox1.Enabled = false;
        groupBox2.Enabled = false;
        txtSLCauHoi.Enabled = false;
        txtThoiGian.Enabled = false;
        txtTenBaiKiemTra.Enabled = false;
        button1.Visible = true;
    }
}
```

**b) btnTaoDeMoi:**

```
{
    button1.Visible = false;
    btnTaoDeThi.Enabled = true;
    btnTaoDeThi.BackColor = Color.Teal;
    cbMaLopHoc.Enabled = true;
    cbMonHoc.Enabled = true;
    cbMonHoc.ResetText();
    cbMaLopHoc.ResetText();
    lsvDSDeThi.Items.Clear();

    groupBox1.Enabled = false;
    rdDeTH.Checked = false;
    DelChuong.Checked = false;
    groupBox2.Enabled = true;
    txtSLCauHoi.Enabled = true;
    txtSLCauHoi.Text = "";
    txtThoiGian.Enabled = true;
    txtThoiGian.Text = "";
    txtTenBaiKiemTra.Enabled = true;
    txtTenBaiKiemTra.Text = "";

    Button[] buttons = new Button[] { btnChuong1, btnChuong2, btnChuong3, btnChuong4, btnChuong5, btnChuong6, btnChuong7,
                                     btnChuong8, btnChuong9, btnChuong10, btnChuong11, btnChuong12, btnChuong13, btnChuong14, btnChuong15, btnChuong16, btnChuong17,
                                     btnChuong18 };
    for (int k = 0; k < countSoChuong; k++)
    {
        buttons[k].Enabled = true;
        buttons[k].ForeColor = Color.Black;
    }
    if (prevChuong < 1)
    {
        cauHoiChuongSelected[prevChuong] = 0;
        soCauHoiTrongChuong[prevChuong] = 0;
    }
    else
    {
        cauHoiChuongSelected[prevChuong - 1] = 0;
        soCauHoiTrongChuong[prevChuong - 1] = 0;
    }
}
```

## 7. Form chọn học viên để xem kết quả:

CÁC QUYỀN CHỨC NĂNG CỦA ADMIN > Xem Kết Quả Học Viên

Lựa chọn

Chọn Mã Lớp

Chọn Môn Học

Chọn mã đề:

Tên Đề Thi:

Chọn mã học viên:

Tên Lớp:

Thông tin học viên

Họ tên học viên:

Giới tính:

Số điện thoại:

Xem kết quả

31



a) btnXemKetQua:

```
private void btnXemKetQua_Click(object sender, EventArgs e)
{
    //Check Điều kiện nhấn nút
    if (cbMaLopHoc.Items == null || cbMaLopHoc.SelectedIndex == -1)
    {
        MessageBox.Show("Chưa chọn mã lớp", "Thông báo");
        cbMaLopHoc.Focus();
        return;
    }
    if (cbMonHoc.Items == null || cbMonHoc.SelectedIndex == -1)
    {
        MessageBox.Show("Chưa chọn môn thi", "Thông báo");
        cbMonHoc.Focus();
        return;
    }
    if (cbMaHV.Items == null || cbMaHV.SelectedIndex == -1)
    {
        MessageBox.Show("Chưa chọn mã học viên", "Thông báo");
        cbMaHV.Focus();
        return;
    }

    //Load Data HV được chọn
    string holot_ = "";
    string ten_ = "";
    string sql = $"select HOLOT, TEN from HOCVIEN where MAHV = '{cbMaHV.Text.Trim()}'";
    SqlCommand sqlCommand = new SqlCommand(sql);
    sqlCommand.Connection = connection;
    SqlDataReader reader = sqlCommand.ExecuteReader();
    while(reader.Read())
    {
        holot_ = reader.GetString(0).Trim();
        ten_ = reader.GetString(1).Trim();
    }
    reader.Close();

    XemKetQuaHV xemKetQua = new XemKetQuaHV();
    xemKetQua.maHV = cbMaHV.Text.Trim();
    xemKetQua.hoTen = holot_ + " " + ten_;
    xemKetQua.maLop = cbMaLopHoc.SelectedItem.ToString().Trim();
    MonHoc = (MonHoc)cbMonHoc.SelectedItem;
    xemKetQua.monHoc = MonHoc.TenMonHoc.Trim();
    xemKetQua.maDe = cbMaDe.Text.Trim();
    xemKetQua.ShowDialog();
}
```

## 8. Form xem kết quả:

The form is titled "Form xem kết quả" and is divided into several sections:

- Thông tin học sinh** (Student Information):
  - Mã học viên:
  - Họ tên:
- Thông tin đề thi** (Exam Information):
  - Mã lớp:
  - Môn học:
  - Mã đề:
  - Thời gian:
- KIỂM TRA MÔN:** (Exam Subject):
  - Tổng điểm:
  - Số câu đúng:
- Câu hỏi số 1** (Question 1):
  - Câu hỏi:
  - Chọn đáp án (Choose the correct answer):
    - ☐ A.
    - ☐ B.
    - ☐ C.
    - ☐ D.
  - Số lượng câu hỏi: 1 /
- Ghi chú:** (Note):
  -

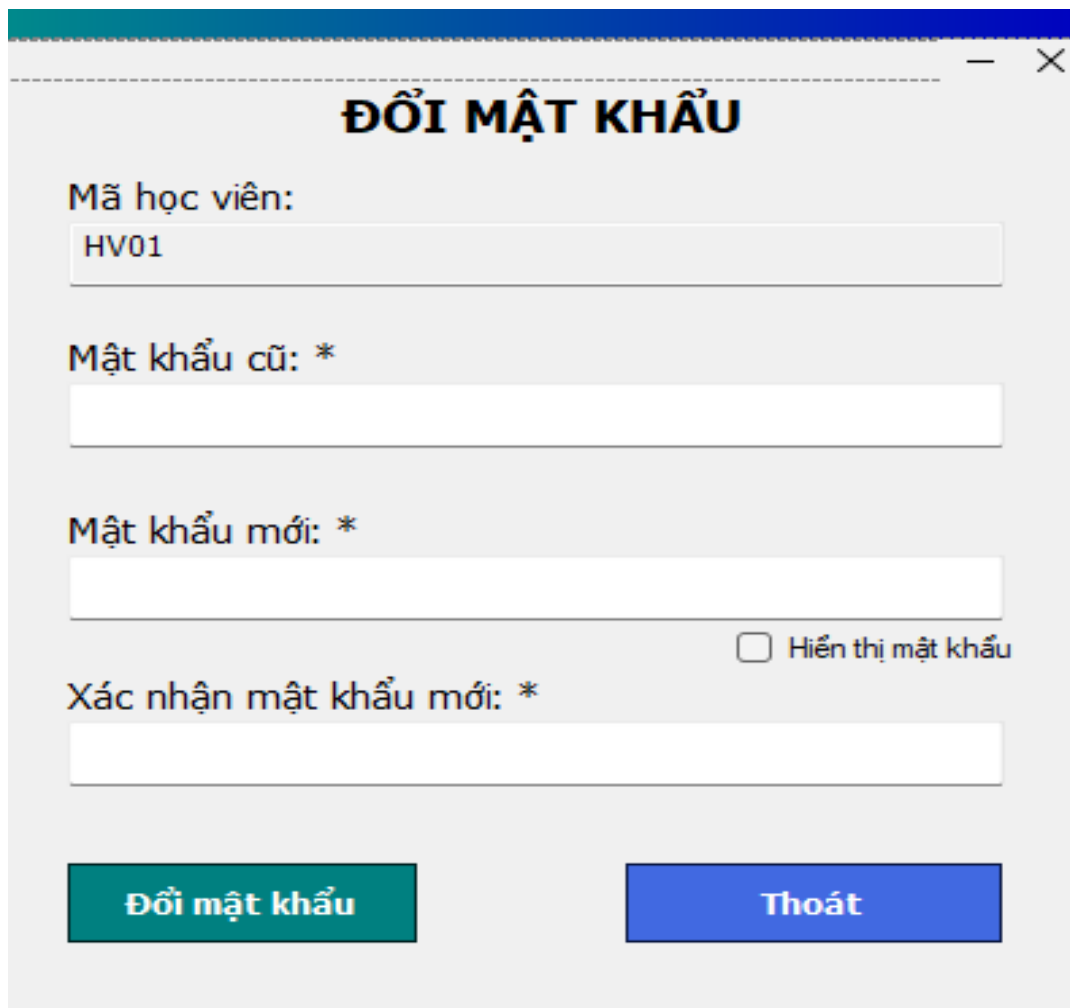
a) **btnTiep:**

- Tương tự như form vào thi.

b) **btnQuayLai:**

- Tương tự như form vào thi.

**9. Form đổi mật khẩu :**



The image shows a web form for changing a password. The form has a title bar with a blue header and a close button. The main title is "ĐỔI MẬT KHẨU" in bold black text. Below the title, there are four input fields: "Mã học viên:" with the value "HV01", "Mật khẩu cũ: \*" (empty), "Mật khẩu mới: \*" (empty), and "Xác nhận mật khẩu mới: \*" (empty). To the right of the "Mật khẩu mới" field is a checkbox labeled "Hiển thị mật khẩu". At the bottom, there are two buttons: a green "Đổi mật khẩu" button and a blue "Thoát" button.

**ĐỔI MẬT KHẨU**

Mã học viên:  
HV01

Mật khẩu cũ: \*

Mật khẩu mới: \*

☐ Hiển thị mật khẩu

Xác nhận mật khẩu mới: \*

**Đổi mật khẩu** **Thoát**

a) btnDoiMK:

```
if (Application.OpenForms.OfType<Admin>().Any())
{
    string sqls = "SELECT * " + $"FROM GIAOVIEN WHERE TAIKHOAN = '{txtMaHV.Text}'";

    SqlCommand sqlCmds = new SqlCommand(sqls);
    sqlCmds.Connection = connection;
    SqlDataReader reader = sqlCmds.ExecuteReader();
    while (reader.Read())
    {
        string matKhai = reader.GetString(1);
        matKhaiHT = matKhai;
    }
    reader.Close();

    if (txtMatKhaiCu.Text != matKhaiHT)
    {
        MessageBox.Show("Mật khẩu cũ không đúng");
        return;
    }
    if (txtMatKhaiCu.Text == txtMatKhaiMoi.Text)
    {
        MessageBox.Show("Mật khẩu bạn muốn đổi giống với mật khẩu hiện tại");
        return;
    }

    string sql = "UPDATE GIAOVIEN SET MATKHAU = @matkhai where TAIKHOAN = @taikhoan";
    //khai bao cac tham so
    SqlParameter prTaiKhoan = new SqlParameter("@taikhoan", SqlDbType.NVarChar);
    SqlParameter prMatKhai = new SqlParameter("@matkhai", SqlDbType.NVarChar);
    //gan gia tri cho tham so
    prTaiKhoan.Value = maHV;
    prMatKhai.Value = txtMatKhaiMoi.Text;
    //thuc thi query
    SqlCommand sqlCmd = new SqlCommand(sql);
    sqlCmd.Parameters.Add(prTaiKhoan);
    sqlCmd.Parameters.Add(prMatKhai);

    sqlCmd.Connection = connection;

    int n = sqlCmd.ExecuteNonQuery();
    if (n > 0)
    {
        MessageBox.Show("Đổi mật khẩu thành công!!");
        this.Close();
    }
    else
    {
        MessageBox.Show("Chưa đổi được mật khẩu");
    }
}
```