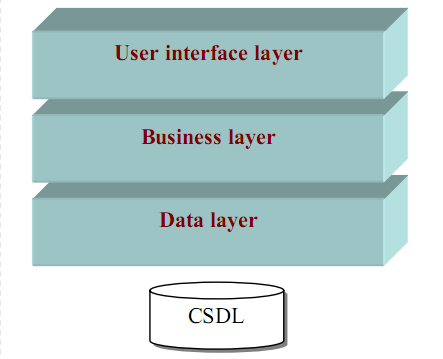
**Thiết kế kiến trúc**

**1.Giới thiệu**

Phần mềm được thiết kế theo kiến trúc ba tầng (three-layers), tạo ra sự độc lập giữa giao diện và người sử dụng bằng cách cô lập các chức năng của giao diện với các chức năng tác nghiệp (business), và cô lập các chức năng tác nghiệp với các chi tiết về truy cập CSDL. Do đó, ba tầng trong cách tiếp cận này là: tầng giao diện người dùng (user interface layer), tầng tác nghiệp (business layer), và tầng truy cập dữ liệu (data layer).



* 1. **Tầng giao diện người dùng:**

Xử lý các sự kiện khi người dùng tương tác với hệ thống, kiểm tra các dữ liệu nhập

* 1. **Tầng xử lý thao tác nghiệp vụ**

Xử lý các thao tác nghiệp vụ

* 1. **Tầng truy cập dữ liệu**

Xử lý việc truy xuất dữ liệu

* 1. **Đối tượng trao đổi liên lạc giữa các layer (DTO)**

**//** Chuyển phần DTO lên trên

**2. Thiết kế đối tượng truyền tải dữ liệu (DTO)**

Các thuộc tính :

2*.1 Lớp BaiHocDTO*

* MaBH (Int)
* STT (Int)
* DoanVan (string[]) //

*2.2 Lớp VanBanDTO*

* MaVB (int)
* MaBH (int)
* TieuDe(int)
* TacGia (string) //
* PhuDe (int)
* NoiDung (string)
* AmThanh (string)
* LoaiVanBan (int)

*2.3 Lớp BaiTapDTO*

* MaBT (int)
* MaVB (int)
* NoiDung (string) //

*2.4 Lớp TuLuanDTO*

* MaTL (int)
* CauHoi (string)
* MucDo (int)
* MaBT (int)

*2.5 Lớp TracNghiemDTO*

* MaTN (int)
* CauHoi (string)
* MucDo (int)
* MaBT (int)
* DapAn (string[]) //

*2.6 Lớp LoaiVanBanDTO*

* MaLoai (Int)
* TenLoai (string)

*2.7 Lớp LoaiCauHoiDTO*

* MaLoai (Int)
* TenLoai (String)

**3. Thiết kế tầng thao tác nghiệp vụ (business layer)**

*2.1 Lớp BaiHocBus*

Các phương thức:

BaiHocDTO Lay\_BaiHoc(int SoThuTu);

Tham số ; int SoThuTu : số thứ tự của bài học

Giá trị trả về : 1 đối tượng BaiHocDTO

Chức năng: lấy 1 đối tượng BaiHoc dựa vào số thứ tự của bài học

*2.2 Lớp VanBanBus*

VanBanDTO Lay\_VanBan(int MaVanBan);

Tham số: int MaVanBan : mã của văn bản cần lấy

Giá trị trả về: 1 đối tượng thuộc kiểu VanBanDTO

Chức năng: lấy 1 đối tượng VanBan dựa vào số thứ tự của văn bản

String[] Lay\_NoiDungVanBan(int MaVanBan)

Tham số: int MaVanBan: mã của văn bản cần lấy nội dung

Giá trị trả về: 1 mảng các chuỗi, mỗi phần tử của mảng là 1 đoạn văn trong văn bản

Chức năng: lấy nội dung của 1 văn bản

String Lay\_TieuDeVanBan(int MaVanBan);

Tham số: int MaVanBan: mã của văn bản cần lấy tiêu đề

Giá trị trả về: chuỗi chứa tiêu đề văn bản

Chức năng: lấy tiêu đề của 1 văn bản

String Lay\_PhuDeVanBan(int MaVanBan);

Tham số: int MaVanBan: mã của văn bản cần lấy phụ đề

Giá trị trả về: chuỗi chứa phụ đề văn bản

Chức năng: lấy phụ đề của 1 văn bản

//

String Lay\_TacGia(int MaVanBan);

Tham số: int MaVanBan : mã của văn bản cần lấy

Giá trị trả về: string chứa tên tác giả

Chức năng: lấy tác giả của tác phẩm

String Lay\_AmThanh(int MaVanBan);

Tham số: int MaVanBan: mã của văn bản cần lấy âm thanh

Giá trị trả về: chuỗi chứa file âm thanh văn bản

Chức năng: lấy file âm thanh của 1 văn bản

*2.3 Lớp TuLuanBus*

String Lay\_NoiDungCauHoi(int MaCauHoi);

Tham số: int MaCauHoi: mã của câu hỏi cần lấy nội dung

Giá trị trả về: chuỗi chứa nội dung câu hỏi

Chức năng: lấy nội dung của 1 câu hỏi

TuLuanDTO Lay\_BaiHoc(int MaCauHoi);

Tham số ; int MaCauHoi: mã câu hõi

Giá trị trả về : 1 đối tượng TuLuanDTO

Chức năng: lấy 1 đối tượng TuLuan dựa vào mã câu hỏi.

*2.4 Lớp TracNghiemBus*

String Lay\_NoiDungCauHoi(int MaCauHoi);

Tham số: int MaCauHoi: mã của câu hỏi cần lấy nội dung

Giá trị trả về: chuỗi chứa nội dung câu hỏi

Chức năng: lấy nội dung của 1 câu hỏi

String[] Lay\_CacPhuongAnTraLoi(int MaCauHoi);

Tham số: int MaCauHoi: mã của câu hỏi cần lấy.

Giá trị trả về: mảng các chuỗi, mỗi chuỗi chứa nội dung 1 lựa chọn

Chức năng: lấy nội dung của 1 câu hỏi

Bool DanhGiaCauTraLoi(int MaCauHoi, String pa);

Tham số : int MaCauHoi: mã của câu hỏi cần đánh giá

String pa: phương án mà người dùng lựa chọn

Giá trị trả về: True 🡪 nếu người dùng chọn đúng

False 🡪 nếu người dùng chọn sai

Chức năng: đánh giá xem người dùng có trả lời đúng câu trắc nghiệm hay không

TracNghiemDTO Lay\_BaiHoc(int MaCauHoi);

Tham số ; int MaCauHoi: mã câu hõi

Giá trị trả về : 1 đối tượng TracNghiemDTO

Chức năng: lấy 1 đối tượng TracNghiem dựa vào mã câu hỏi.

*2.5 Lớp BaiTapBus*

BaiTapDTO Lay\_BaiTap(int MaBaiTap);

Tham số ; int MaBaiTap: mã bài tập

Giá trị trả về : 1 đối tượng BaiTapDTO

Chức năng: lấy 1 đối tượng BaiTap dựa vào mã bài tập.

TracNgiemDTO[] Lay\_CauHoiTracNghiem(int MaBaiTap);

Tham số ; int MaBaiTap: mã bài tập

Giá trị trả về : 1 mảng đối tượng TracNghiemDTO

Chức năng: lấy tất cả các câu hỏi trắc nghiệm có trong 1 bài tập

TuLuanDTO[] Lay\_CauHoiTracNghiem(int MaBaiTap);

Tham số ; int MaBaiTap: mã bài tập

Giá trị trả về : 1 mảng đối tượng TuLuanDTO

Chức năng: lấy tất cả các câu hỏi tự luận có trong 1 bài tập

**4. Thiết kế tầng truy xuất dữ liệu (Data layer)**

*3.1 Lớp BaiHocDAO*

Các phương thức:

BaiHocDTO Lay\_BaiHoc(int SoThuTu);

Tham số ; int SoThuTu : số thứ tự của bài học

Giá trị trả về : 1 đối tượng BaiHocDTO

Chức năng: lấy 1 đối tượng BaiHoc dựa vào số thứ tự của bài học

*3.2 Lớp VanBanDAO*

VanBanDTO Lay\_VanBan(int MaVanBan);

Tham số: int MaVanBan : mã của văn bản cần lấy

Giá trị trả về: 1 đối tượng thuộc kiểu VanBanDTO

Chức năng: lấy 1 đối tượng VanBan dựa vào số thứ tự của văn bản

String[] Lay\_NoiDungVanBan(int MaVanBan)

Tham số: int MaVanBan: mã của văn bản cần lấy nội dung

Giá trị trả về: 1 mảng các chuỗi, mỗi phần tử của mảng là 1 đoạn văn trong văn bản

Chức năng: lấy nội dung của 1 văn bản

String Lay\_TieuDeVanBan(int MaVanBan);

Tham số: int MaVanBan: mã của văn bản cần lấy tiêu đề

Giá trị trả về: chuỗi chứa tiêu đề văn bản

Chức năng: lấy tiêu đề của 1 văn bản

String Lay\_PhuDeVanBan(int MaVanBan);

Tham số: int MaVanBan: mã của văn bản cần lấy phụ đề

Giá trị trả về: chuỗi chứa phụ đề văn bản

Chức năng: lấy phụ đề của 1 văn bản

//

String Lay\_TacGia(int MaVanBan);

Tham số: int MaVanBan : mã của văn bản cần lấy

Giá trị trả về: string chứa tên tác giả

Chức năng: lấy tác giả của tác phẩm

String Lay\_AmThanh(int MaVanBan);

Tham số: int MaVanBan: mã của văn bản cần lấy âm thanh

Giá trị trả về: chuỗi chứa file âm thanh văn bản

Chức năng: lấy file âm thanh của 1 văn bản

*3.3 Lớp TuLuanDAO*

String Lay\_NoiDungCauHoi(int MaCauHoi);

Tham số: int MaCauHoi: mã của câu hỏi cần lấy nội dung

Giá trị trả về: chuỗi chứa nội dung câu hỏi

Chức năng: lấy nội dung của 1 câu hỏi

TuLuanDTO Lay\_BaiHoc(int MaCauHoi);

Tham số ; int MaCauHoi: mã câu hõi

Giá trị trả về : 1 đối tượng TuLuanDTO

Chức năng: lấy 1 đối tượng TuLuan dựa vào mã câu hỏi.

*3.4 Lớp TracNghiemDAO*

String Lay\_NoiDungCauHoi(int MaCauHoi);

Tham số: int MaCauHoi: mã của câu hỏi cần lấy nội dung

Giá trị trả về: chuỗi chứa nội dung câu hỏi

Chức năng: lấy nội dung của 1 câu hỏi

String[] Lay\_CacPhuongAnTraLoi(int MaCauHoi);

Tham số: int MaCauHoi: mã của câu hỏi cần lấy.

Giá trị trả về: mảng các chuỗi, mỗi chuỗi chứa nội dung 1 lựa chọn

Chức năng: lấy nội dung của 1 câu hỏi

TracNghiemDTO Lay\_BaiHoc(int MaCauHoi);

Tham số ; int MaCauHoi: mã câu hõi

Giá trị trả về : 1 đối tượng TracNghiemDTO

Chức năng: lấy 1 đối tượng TracNghiem dựa vào mã câu hỏi.

*3.5 Lớp BaiTapDAO*

BaiTapDTO Lay\_BaiTap(int MaBaiTap);

Tham số ; int MaBaiTap: mã bài tập

Giá trị trả về : 1 đối tượng BaiTapDTO

Chức năng: lấy 1 đối tượng BaiTap dựa vào mã bài tập.

TracNgiemDTO[] Lay\_CauHoiTracNghiem(int MaBaiTap);

Tham số ; int MaBaiTap: mã bài tập

Giá trị trả về : 1 mảng đối tượng TracNghiemDTO

Chức năng: lấy tất cả các câu hỏi trắc nghiệm có trong 1 bài tập

TuLuanDTO[] Lay\_CauHoiTracNghiem(int MaBaiTap);

Tham số ; int MaBaiTap: mã bài tập

Giá trị trả về : 1 mảng đối tượng TuLuanDTO

Chức năng: lấy tất cả các câu hỏi tự luận có trong 1 bài tập

**3. Thiết kế tầng giao diện**