1. Cách triển khai 1 chương trình khi clear Requirement và Business

Thu đc DB diagram, document….

* Tạo môi trường làm việc

Team leader lập coding convention, solution và SVN(networking…)

Xác định mô hình làm việc

Tạo thư mục làm việc, giải thích hệ thống thư mục có document lại, sau đó phải lưu mô tả thư mục lại.

Kick off dự án

**Lợi ích của coding rule và coding convention:**

*Thống nhất về dl và source code*

*Thống nhất cách xử lý khi bị conflict*

*Giúp dễ review*

*Hạn chế rủi ro*

*Dễ maintain*

*Tiết kiệm thời gian khi phải làm code mới*

*Dễ quản lý source code*

*Dễ xây dựng core*

**Business:**

* Xuất phát từ nhu cầu thực tế

Cần lưu có thời gian, mẫu biểu

Giao diện hiển thị cho người dung trực tiếp(phân quyền)

Back up? Tg bao lâu, server nào hay lưu ra đĩa

**Tạo ERD diagram**

**Tạo Database**

**Tạo Model**

**Tạo kết nối DB**

Làm thế nào để kết nối model và sql?????????????????????

* API – connect – library – query/nonquery – datatable – list<T>(dl rất nhỏ, phải viết nhiều)
* Entity Framework(dl tương đối nhỏ, vòng đời nhỏ và rất chậm, phát triển trên .net cho fep C# hết nối sql server và 1 số csdl, cách hoạt động khá giống hvit Framework)
* LinQ to SQL(viết lệnh truy vấn trực tiếp tại c# và đổ trực tiếp ra list, cx là 1 framework chậm).

***Lựa chọn: xây dựng framework riêng dựa trên nền tảng đang học(DataAccess)***

DataAccess cho phép truy cập và tương tác vào CSDL mà không cần biết đó là CSDL nào.

Truy vấn:

Có thể truy vấn trực tiếp

Store procedule: có tham số hoặc k có tham số truyền

Phi truy vấn:

Có thể truy vấn trực tiếp

Store procedule: có tham số hoặc k có tham số truyền

* *Detail design cho DataAccess*

K nên mở kết nối từ đầu ct cho đến khi ct tự đóng vì khi open ta phải bỏ ra 1 lượng tài nguyên bộ nhớ để lưu trữ kết nối của C# vào sql. Khi sd ct thao tác chủ yếu của ng dung là croll chuôt, vì vậy nếu giữ kết nối sẽ mất thời gian.

* *Đổ dữ liệu ra lis<T> và trả lại dữ liệu từ đối tượng vào DB*

Xây dựng controller chung cho chương trình

Vấn đề:

+ Mỗi model khác nhau

+ Action khác nhau

+ lib không biết đến object đưa vào là gì

Phương pháp giải quyết vấn đề: Generic (viết core)

* Tao ModelBase đọc mọi loại đối tượng

List<T>,Dictionary<T,k>: T cho phép truyền kiểu vào

Where T: new() là đối tượng đc tạo ra từ toán tử new, nếu k đc tạo từ toán tử new thì là *abstract, interface* hoặc *hàm tạo private*.

*Reflection* sử lý các vấn đề liên quan đến kiểu(rất chậm, là core của linQ)

* Tạo ControllerBase để xử lý nghiệp vụ chung
* Sau khi có core

Khai báo Model cụ thể

Tùy sử dụng Controller cụ thể theo 1 hướng nào đó.