**KẾ HOẠCH THỰC HÀNH, KIỂM TRA CUỐI KỲ, THI**

**MÔN ĐẶC TẢ PHẦN MỀM**

**PHẦN 1: ĐẶC TẢ**

**Buổi 1:**

Mỗi sinh viên tự chọn Đề tài một Ứng dụng nhỏ (tương đương với đề tài tốt nghiệp) khoảng 10 chức năng cơ sở và cơ sở dữ liệu gồm khoảng 10 bảng.

Xác định chính xác Tên đề tài, mục tiêu và yêu cầu của Đề tài. Khảo sát thực tế. Phân tích yêu cầu.

Bước đầu viết đặc tả tổng quan yêu cầu cho Đề tài ứng dụng (3 trang) theo mẫu Đặc tả yêu cầu và các Ví dụ như Amazing Lunch Indicator, Đặt phòng khách sạn.

**Buổi 2**

Hoàn thành Đặc tả yêu cầu của Đề tài đã chọn, đưa ra Sơ đồ phân rã chức năng và đặc tả các chức năng. (bổ sung thành 6 trang)

**Buổi 3**

Tiếp tục đặc tả chi tiết từng chức năng cơ sở có trong sơ đồ phân rã chức năng của Đề tài.

Hoàn thiện báo cáo, In nộp báo cáo khoảng 5 Trang với nội dung sau:

**LẦN 1. Mỗi sinh viên In báo cáo (sử dụng 1 mặt giấy): Họ tên sinh viên, Lớp, Tên đề tài, Đặc tả tổng quan yêu cầu phần mềm, Sơ đồ phân rã chức năng, đặc tả các chức năng chính và các chức năng cơ sở. Đem nộp tại buổi học thứ 4. Báo cáo tự viết: chính xác, rõ ràng, đầy đủ, không quá rộng, không quá hẹp, KHÔNG SAO CHÉP, CÓP NHẶT.**

**Buổi 6**

Xác định các thực thể (bộ khóa duy nhất của các thực thể), các thuộc tính, ràng buộc. Đưa ra Sơ đồ quan hệ thực thể (bổ sung thành 8-10 trang)

**Buổi 7-8**

Hoàn thiện Đặc tả cơ sở dữ liệu: đặc tả các thực thể, các thuộc tính, quan hệ, ràng buộc, bộ khóa duy nhất.

**Tiếp tục hoàn thiện báo cáo với nội dung:**

* Họ tên, lớp
* Tên đề tài
* **Đặc tả tổng quan yêu cầu phần mềm**
* Sơ đồ phân ra chức năng
* Đặc tả các chức năng
* Sơ đồ quan hệ thực thể
* Đặc tả các thực thể, unique keys, các thuộc tính khác, quan hệ, kiểu và độ dài dữ liệu.

**LẦN 2. Mỗi sinh viên In báo cáo đem nộp tại buổi học thứ 9 với nội dung:**

* Họ tên, lớp
* Tên đề tài
* **Tóm tắt Đặc tả yêu cầu phần mềm**
* Sơ đồ phân ra chức năng
* Đặc tả các chức năng
* Sơ đồ quan hệ thực thể
* Đặc tả các thực thể, unique keys, các thuộc tính khác, quan hệ, kiểu và độ dài dữ liệu.

Dựa theo góp ý của Thày giáo hoàn thiện sơ bộ tài liệu Đặc tả. Hoàn thiện cài đặt CSDL vật lý với các ràng buộc và chèn 5 bản ghi vào mỗi bảng:

Chỉ rõ các lệnh SQL việc tạo ra các bảng với các ràng buộc: khóa tự tăng, bộ khóa unique keys, nullable hay mandatory, kiểu dữ liệu, độ dài,

**Buổi 10**

Cài đặt cơ sở dữ liệu Vật lý. Dùng lệnh SQL insert khoảng 5 bản ghi vào mỗi bảng trong CSDL. Ví dụ tạo bảng Chi tiết hóa đơn với các ràng buộc unique kyes gồm 2 trường #masanpham và #madonhang như sau:

Create Table TAIKHOAN (

mataikhoan int IDENTITY(1,1) PRIMARY KEY,

tendangnhap nvarchar(30) not null,

matkhau varchar(100) not null,

loaitaikhoan nvarchar(50) not null,

)

Create Table DONDATHANG (

madondathang int IDENTITY(1,1) PRIMARY KEY,

--user1: khách hàng đặt hàng

mataikhoan1 int FOREIGN KEY

REFERENCES KHACH\_HANG (makhachhang),

-- user2:nhân viên tạo đơn đặt

mataikhoan2 int FOREIGN KEY

REFERENCES TAIKHOAN (mataikhoan),

trangthaitt nvarchar(50) not null,

thoigian datetime null default(GETDATE()) ,

tongtien float null,

mota nvarchar(255) null,

CONSTRAINT dondathang\_unique UNIQUE(mataikhoan1,thoigian)

)

Create Table PHIEUNHAPKHO (

maphieunhapkho int IDENTITY(1,1) PRIMARY KEY,

--user: nhân viên lập phiếu

maNV int FOREIGN KEY

REFERENCES TAIKHOAN (mataikhoan),

-- : nhà cung cấp

maNCC int FOREIGN KEY

REFERENCES NHACUNGCAP (maNCC),

trangthaitt nvarchar(50) not null,

thoigian datetime null default(GETDATE()) ,

tongtien float null,

mota nvarchar(255) null,

CONSTRAINT phieunhap\_unique UNIQUE(maNV,thoigian)

)

create table DACTRUNG(

madactrung int IDENTITY(1,1) PRIMARY KEY,

loaidactrung nvarchar(50) not null,

thutu int not null,

giatri nvarchar(50) null,

donvi nvarchar(50) null,

ten nvarchar(50) not null,

mota nvarchar(255) null,

CONSTRAINT dtrung\_unique UNIQUE(loaidactrung,thutu)

)

create table DACTRUNG\_SANPHAM(

madtsp int IDENTITY(1,1) PRIMARY KEY,

masanpham int FOREIGN KEY

REFERENCES Sanpham (masanpham),

madactrung int FOREIGN KEY

REFERENCES Dactrung (madactrung),

mota nvarchar(255) null,

CONSTRAINT dtrung\_spham\_unique UNIQUE(masanpham,madactrung)

)

create table CHITIETPNK(

machitietpnk int IDENTITY(1,1) PRIMARY KEY,

masanpham int FOREIGN KEY

REFERENCES Sanpham (masanpham),

maphieunhap int FOREIGN KEY

REFERENCES Phieunhapkho (maphieunhapkho),

soluong int not null,

gianhap float,

tongtien float null,

lohang nvarchar(255) null,

ghichu nvarchar(255) null,

CONSTRAINT cttiet\_pnk\_unique UNIQUE(masanpham,maphieunhap)

)

create table CHITIETDDH(

machitietDDH int IDENTITY(1,1) PRIMARY KEY,

masanpham int FOREIGN KEY

REFERENCES sanpham (masanpham),

madondat int FOREIGN KEY

REFERENCES Dondathang (madondathang),

soluong int not null,

tongtien float null,

ghichu nvarchar(255) null,

CONSTRAINT ctiet\_ddh\_unique UNIQUE(masanpham,madondat)

)

create table DANHGIA(

madanhgia int IDENTITY(1,1) PRIMARY KEY,

mataikhoan int FOREIGN KEY

REFERENCES Taikhoan (mataikhoan),

masanpham int FOREIGN KEY

REFERENCES sanpham (masanpham),

thoigian datetime null default(GETDATE()),

xephang nvarchar(255) null,

mota nvarchar(255) null,

tthienthi int null,

CONSTRAINT danhgia\_unique UNIQUE(mataikhoan,thoigian)

)

**Buổi 9 & Buổi 10**

Chỉnh sửa báo cáo theo góp ý của Thày giáo, cài đặt CSDL vật lý, mô tả các lệnh SQL đảm bảo các bộ Unique keys và các ràng buộc. Bổ sung đặc tả chi tiết 2 module, thiết kế giao diện của hai module Form nhập dữ liệu sau:

**1. Form Cập nhật dữ liệu vào một bảng**. Ví dụ như Cập nhật sản phẩm có nhiều trường dữ liệu kiểu khác nhau như:

* Kiểu dữ liệu tự tăng: làm khóa
* Khai báo bộ dữ liệu duy nhất
* Kiểu Date
* Kiểu Image
* Kiểu text: mô tả sản phẩm.
* Bắt buộc hay tùy chọn
* Giá trị mặc định
* …..

**2. Form Cập nhật dữ liệu vào nhiều bảng**. Ví dụ Lập hóa đơn bán:

* Tạo tiêu đề Hóa đơn bán, insert bản ghi mới vào Bảng Hóa đơn bán
* Tạo các dòng chi tiết Hóa đơn, mỗi dòng ứng với một sản phẩm
* Cập nhật chi tiết hóa đơn
* Cập nhật tiêu đề Hóa đơn, ví dụ tính tổng tiền, …
* Như vậy trong Form này có 2 cửa sổ: một của sổ cho tiêu đề và các trường của Bảng Master và một cửa sổ dành cho các bản ghi chi tiết của Bảng Detail.

Việc cài đặt 2 form này có thể thực hiện sau khi đã đặc tả và thiết kế giao diện chi tiết, trên bất cứ ngôn ngữ lập trình nào, mà sinh viên mong muốn và được hoàn thiện trong quá trình ôn thi kết thúc học phần. Cùng với Tài liệu Đặc tả, chất lượng cài đặt cơ sở dữ liệu vật lý, việc cài đặt 2 Form này sẽ được đánh giá trong buổi thi kết thúc học phần theo lịch thi của nhà Trường.



**Trước buổi học 13, mỗi sinh viên hoàn thiện, chỉnh sửa báo cáo với nội dung:**

* Họ tên, lớp
* Tên đề tài
* **Tóm tắt Đặc tả yêu cầu phần mềm**
* Sơ đồ phân ra chức năng
* Đặc tả các chức năng
* Sơ đồ quan hệ thực thể
* Đặc tả các thực thể, unique keys, các thuộc tính khác, quan hệ, kiểu và độ dài dữ liệu.
* Cài đặt CSDL vật lý
* Chỉ rõ các lệnh SQL việc tạo ra các bảng với các ràng buộc: khóa tự tăng, bộ khóa unique keys, nullable hay mandatory, kiểu dữ liệu, độ dài, …
* Mỗi bảng dữ liệu dùng lệnh Insert của SQL chèn 5 bản ghi có đầy đủ dữ liệu bắt buộc và các quan hệ khóa ngoài
* Đặc tả 2 Form dự kiến cài đặt

**PHẦN 2: MẪU THIẾT KẾ**

Mỗi mẫu thiết kế được học trong Giáo trình đi kèm với chương trình demo trên Java. Sinh viên tải bộ chương trình mẫu về, sử dụng môi trường cài đặt Java tùy thích, tạo Project cho mẫu, vận hành chạy các chương trình mẫu:

* Đọc hiểu yêu cầu bài toán đặt ra
* Đọc hiểu mẫu thiết kế sử dụng, các thành phần và mối liên hệ của chúng
* Hiểu cài đặt các giao diện các lớp
* Hiểu ứng dụng client sử dụng mẫu và giải thích được kết quả nhận dược
* Thực hiện yêu cầu cài đặt bổ sung trong mô tả đối với từng mẫu thiết kế sau đây. Chỉnh sửa các lớp, giao diện và đưa ra kết quả Demo
* Có thể thay vì cài đặt theo bài toán yêu cầu trong mô tả thực hành bằng việc tự sử dụng mẫu thiết kế để giải quyết một bài toán thực tế của riêng mình.

**Buổi 11**

Thực hành 3 mẫu

* **Addapter:** Kiểu cũ XPay: có trường Tên và trường Họ đệm riêng, trong khi đó kiểu mới PayD chỉ có trường chung cả Họ tên. Hãy bổ sung điều chỉnh vào Addapter và ứng dụng client để hệ thống trả tiền PayD hoạt động được.
* **Façade**: Giả sử máy chủ của chúng ta là máy chủ Cơ sở dữ liệu. Bạn có thể tạo façade cho người dùng vào sử dụng một phiên làm việc của máy chủ Cơ sở dữ liệu với cơ sở dữ liệu mặc định trước dbA như thế nào?
* **Composite**: Liên lạc nhóm.

Một cá nhân có một địa chỉ liên lạc.

Một địa chỉ liên lạc nhóm là danh sách các địa chỉ liên lạc (cá nhân) hoặc địa chỉ liên lạc nhóm. Cài đặt ứng dụng gửi thư cho một địa chỉ liên lạc (cá nhân hoặc nhóm).

**Buổi 12**

Thực hành 3 mẫu

* **Bridge**: Mở rộng ứng dụng cho một mẫu ôtô Tải và cũng tạo ra 2 sản phẩm khóa Trung tâm và khóa bánh răng cho loại oto mới đó. Thử ứng dụng client minh họa.
* **Singleton**: Giả sử cần thiết tạo chỉ một Database Object cho một ứng dụng của Công ty (vì nếu tạo nhiều đối tượng như vậy sẽ rất tốn kém). Bạn đề xuất giải pháp cài đặt cụ thể như thế nào.
* **Obsever**: Thông báo điểm thi môn. Chủ đề Subject ở đây là một lớp Học phần, thỉnh thoảng gửi thông báo dạng text về các hoạt động của học phần. Các sinh viên học phần đó sẽ đăng ký và nhận được thông báo của Học phần.

**Buổi 13**

Thực hành 3 mẫu

* **Mediator**: Thêm một sensor cân trọng lượng quần áo để tính toán mức nước cho lần giặt đó. Cập nhật trao đổi với Mediator để thực hiện nhiệm vụ trên và ứng dụng client minh họa.
* **Proxy**: Giả sử Báo cáo Doanh thu chạy trên máy cục bộ. Tuy nhiên chỉ người chủ Owner mới được quyền xem. Điều chỉnh ứng dụng để thể hiện điều đó.
* **Chain of Responsibility**: Cài đặt ứng dụng Tiếp nhận Dịch vụ Công một cửa. Các yêu cầu được gửi đến và xếp vào một Hàng đợi. Sau đó truyền qua các bộ phận xử lý, đến phần trách nhiệm của bộ phận nào, bộ phần đó xử lý và trả về kết quả, nếu không thuộc trách nhiệm xử lý của mình thì chuyển tiếp cho bộ phận tiếp theo sau. Các bộ phận nối tiếp nhau được thiết kế linh hoạt tùy tình hình thực tế.

**Buổi 14**

Thực hành 3 mẫu

* **Flighweight**: Thêm một nền tảng Python. Và cập nhật ứng dụng client để minh họa sử dụng dịch code Python 2 lần.
* **Builder**: Giả sử việc tạo Website môn học là phức tạp gồm các phần như: Giới thiệu nội dung môn (Đề cương, Tài liệu), Giảng dạy lý thuyết (các bài giảng), Hướng dẫn thực hành (các bài thực hành), Luyện tập Bài tập (các bài tập), Đánh giá (Cách cấm Điểm thành phần và Hình thức Thi). Sử dụng mẫu Builder để trợ giúp xây dựng Website môn học cho Khoa CNTT và Khoa Mác lê và cài đặt ứng dụng client xây dựng Website 2 môn cụ thể: Đặc tả phần mềm và Tư tưởng Hồ Chí Minh.
* **Prototype:** Giả sử trước mỗi học kỳ, sinh viên các lớp Đăng ký các học phần sẽ học theo chương trình đào tạo. Đây là thao tác đắt giá, vì phải kiểm tra các môn được đăng ký theo chương trình đào tạo, một môn có thể có nhiều học phần và số lượng sinh viên vào Đăng ký đông. Do đó việc tạo ra Danh sách các học phần mặc định cho một lớp là một việc làm hữu ích tiết kiệm nhiều công sức, dùng chung cho mỗi sinh viên của lớp đó đăng ký. Sau đó sinh viên này có thể cập nhật một số học phần tự chọn. Áp dụng mẫu Prototype giải quyết bài toán này.
* **Prototype:** thay đổi trường access trong Ví dụ mẫu thành kiểu HashTable có khóa là đối tượng tài nguyên Subject và giá trị là các kiểu thao tác. Ở đây cần thực hiện Deep Clone, tức là thực hiện sao mọi thành phần bên trong của đối tượng.

**Buổi 15**

Thực hành 3 mẫu

* **Factory Method**: Giả sử khi học các Học phần, hàng tuần sinh viên phải nộp các báo cáo gồm nhiều loại khác nhau: Các bài test trắc nghiệm, các bài code, các bài luận. Mỗi loại bài được xử lý đánh giá chấm điểm khác nhau. Tạo phần code tiếp nhận báo cáo của sinh viên và cho kết quả đánh giá. Sử dụng mẫu Factory Method để cài đặt, sau đó viết ứng dụng client minh họa
* **Abstract Factory**: Giả sử việc Đánh giá báo cáo nộp hàng tuần của các sinh viên trong các học phần khác nhau là khác nhau. Sử dụng mẫu Abstract Factory để mở rộng code phần trước đáp ứng yêu cầu này cho các môn học khác nhau. Chỉnh sửa và viết ứng dụng client minh họa.
* **Memento:** Giả sử trong quá trình cài đặt một chương trình, lập trình viên luôn tìm tòi thử nghiệm các phương án khác nhau. Và muốn lưu giữ các phương án code đã thử nghiệm để quay trở lại một phương án khi cần thiết. Hãy sử dụng mẫu thiết kế Memento để cài đặt bài toán đó và viết ứng dụng client minh họa.

**Buổi 16**

Thực hành 3 mẫu

* **Template**: Giả sử qui trình tạo khung cho một Form nhập một bản ghi mới cho một Bảng đã được xác định: Tạo một đối tượng cơ sở dữ liệu, xác thực người sử dụng, tạo một phiên làm việc, mở một bảng, tạo một bản ghi mới, insert vào bảng, Commit, đóng phiên, đóng cơ sở dữ liệu. Hãy sử dụng mẫu Template cho bài toán này.
* **State**: Giả sử yêu cầu Robot nấu ăn có thêm một trạng thái giữ nóng Warm. Nếu Robot đang ở trạng thái Cook có thể chuyển sang trạng thái giữ nóng Warm. Nếu Robot đang ở trạng thái giữ nóng Warm, nó có thể cook và off để chuyển sang trạng thái Cook hoặc Off tương ứng, nhưng không thể walk. Chỉnh sửa và viết ứng dụng client minh họa.
* **Strategy:** Giả sử chúng ta có 3 chiến lượcThiết kế để giải một Bài toán: Thiết kế trên xuống, Thiết kế dưới lên và Thiết kế kết hợp trên xuống với dưới lên. Thiết kế trên xuống một bài toán là Thiết kế trừu tượng đích trước khi thiết kế các bổ trợ của nó (có ưu điểm là kiểm soát tốt các mục tiêu); ngược lại Thiết kế dưới lên là Thiết kế các use-case trước khi Thiết kế các module sử dụng chúng. Giả sử mỗi đội lập trình có một chiến lược thiết kế, nó được quyết định động tùy thuộc nhiều yếu tố. Cài đặt chương trình hỗ trợ Thiết kế cho các đội lập trình sử dụng mẫu Strategy.

**Buổi 17**

Thực hành 3 mẫu

* **Command**: Giả sử trong Ví dụ minh họa mẫu Command, chúng ta thêm một kiểu công việc là Đăng ký Tài Khoản (gồm Tên người sử dụng và mật khẩu) – RegistrationJob có Private thuộc tính là Registration và phương thức Registry(). Bổ sung công việc vào chương trình và xây dựng ứng dụng client minh họa.
* **Command**: Phòng đào tạo xây dựng chương trình tiếp nhận các yêu cầu của sinh viên để xử lý. Các yêu cầu gồm: Xin hoãn thi, Xin dừng học một học kỳ, Xin vay vốn ngân hàng, Xin ở ký túc xá, … Thiết kế chương trình tự động xử lý các yêu cầu của sinh viên sử dụng mẫu Command.
* **Interpreter**: Cài đặt thêm phép chia và phép lũy thừa cho Ví dụ minh họa của mẫu thiết kế này.
* **Decorator:** Giả sử chúng ta thiết kế chương trình bán điện thoại di động. Khách hàng có thể yêu cầu lắp thêm nhiều loại vỏ bọc bảo vệ, các loại dán màn hình, các loại thẻ nhớ, giá đỡ, túi đựng, … Hãy sử dụng mẫu Decorator thiết kế chương trình đáp ứng yêu cầu bổ sung của Khách hàng khi mua một chiếc điện thoại di động.

**Buổi 18**

Thực hành 2 mẫu

* **Iterator**: Cài đặt bộ duyệt Iterator đưa ra danh sách các phần tử số nguyên từ nhỏ đến lớn đã xét cuối Chương 1.
* **Visitor**: Giả sử chúng ta có các họ các phần tử là các số với các bộ duyệt của chúng như: Mảng, Danh sách, Vecto, cây nhị phân, … Các kiểu họ này có thể được bổ sung thêm trong quá trình sử dụng. Đối với từng loại họ các phần tử như trên chúng ta có các phương thức xử lý đồng nhất cho từng phần tử và cả họ như: Đếm số phần tử, Tính tổng các phần tử, Tính tích các phần tử, … Các phương thức này có thể được bổ sung trong quá trình khai thác. Hãy sử dụng mẫu Visitor thiết kế chương trình giải quyết bài toán trên. Xây dựng ứng dụng client minh họa việc dễ dàng bổ sung kiểu họ các phần tử như cây đống, hay thêm phương thức cho họ Tính tỷ lệ các số không âm (thương số các số không âm trên tổng số các phần tử).

**PHẦN 3: NGHIỆM THU THỰC HÀNH – ĐIỂM THÀNH PHẦN**

**Buổi nghiệm thu thực hành**

Vào 2 tuần trước khi môn học kết thúc Giáo viên sẽ Tổ chức Buổi nghiệm thu Kết quả Thực hành.

Từng sinh viên sẽ báo cáo việc cài đặt các chương trình qua các bài tập thực hành trên: có thể thay các bài toán yêu cầu trong các tuần thực hành bằng các bài toán của riêng mình và sử dụng các mẫu thiết kế để cài đặt.

Mỗi sinh viên tự cài đặt ít nhất 9 mẫu thiết kế riêng của mình (3 mẫu khởi tạo, 3 mẫu cấu trúc và 3 mẫu hành vi).

Viết đặc tả 9 mẫu thiết kế đã cài đặt, kèm theo giao diện kết quả cài đặt các mẫu đó, đưa vào **Báo cáo nghiệm thu thực hành** (khoảng 5 trang)**.**

Kết quả **Báo cáo nghiệm thu thực hành và demo cài đặt các mẫu đó** tại Buổi nghiệm thu thực hành, cùng với kết quả điểm danh các buổi học và thái độ tích cực trong các buổi học sẽ góp phần đánh giá điểm thành phần của sinh viên.

**THI KẾT THÚC HỌC PHẦN**

Buổi thi sẽ được tổ chức theo Lịch thi của Trường.

Đến buổi thi, từng sinh viên in và nộp báo Đặc tả chương trình của mình như mô tả trong Phần 1, khoảng 15 trang, gồm

1. Tên Đề tài
2. Đặc tả yêu cầu
3. Đặc tả thiết kế, gồm
   1. Sơ đồ module phụ thuộc (Phân tích thiết kế hướng đối tượng), đặc tả từng module hoặc
   2. Sơ đồ phân ra chức năng (đặc tả từng chức năng) và Mô hình quan hệ thực thể (đặc tả từng thực thể, thuộc tính, bộ thuộc tinh duy nhất và các ràng buộc)
4. Cài đặt cơ sở dữ liệu vật lý với các ràng buộc: Chỉ rõ các lệnh SQL việc tạo ra các bảng với các ràng buộc: khóa tự tăng, bộ khóa unique keys, nullable hay mandatory, kiểu dữ liệu, độ dài. Mỗi bảng dữ liệu dùng lệnh Insert của SQL chèn 5 bản ghi có đầy đủ dữ liệu bắt buộc và các quan hệ khóa ngoài
5. Đặc tả 2 Form dự kiến cài đặt
6. Cài đặt 2 module là Form nhập dữ liệu như đã mô tả ở Bài thực hành của Tuần 9-10. Một Form nhập dữ liệu trực tiếp vào 1 Bảng và một Form nhập dữ liệu đồng thời trên một số bảng. Hai Form này được cài đặt trên ngôn ngữ lập trình bất kỳ, tùy theo mong muốn của sinh viên
7. Các Form được đánh giá tốt nếu việc nhập dữ liệu là thuận tiện, cho phép nhập nhiều kiểu dữ liệu, giao diện gần với thực tế sử dụng, nhiều tiện ích hỗ trợ như chọn lịch, chọn file hình ảnh, đảm bảo tính đúng đắn của các trường dữ liệu và các ràng buộc như cảnh báo các lỗi xảy ra.

Tại buổi thi từng sinh viên lên trình bày: nộp báo cáo, vận hành cơ sở dữ liệu vật lý và demo hai Form nhập dữ liệu. Sinh viên sẽ có thêm thời gian thực hiện yêu cầu bổ sung - chỉnh sửa nhỏ của Giáo viên tại buổi thi.

Điểm thi sẽ được đánh giá qua báo cáo đặc tả và các kết quả demo cơ sở dữ liệu vật lý, 2 module Form nhập dữ liệu và kết quả thực hiện yêu cầu bổ sung của giáo viên tại buổi thi

Giảng viên: Trần Văn Dũng

Số điện thoại: 0904588833

Email: [zungtv@yahoo.com](mailto:zungtv@yahoo.com)