

#### Software Development Project - Dự án Phát triển Phần mềm

# Object Oriented Analysis & Design &

# Database Design In Harmony

**SE TEAM** 



# **User Requirements...**

User Requirements: Các câu phát biểu ngắn gọn, tập trung vào mục tiêu mà người dùng muốn đạt được khi sử dụng hệ thống, chứ không đi sâu vào chi tiết kỹ thuật. Hệ thống cung cấp điều gì – what cho họ?

#### Ví dụ ứng dụng Chăm sóc sắc đẹp:

- Người dùng muốn tìm kiếm salon gần nhất dựa trên vị trí hiện tại của họ.
- Người dùng muốn xem dịch vụ cụ thể như làm tóc, làm móng, hoặc chăm sóc da mà các salon cung cấp.
- Người dùng muốn đặt lịch hẹn trực tiếp qua ứng dụng và nhận xác nhận lịch hẹn qua thông báo hoặc email.
- Người dùng muốn nhận tư vấn cá nhân hóa về các sản phẩm chăm sóc sắc đẹp phù hợp với tình trạng da hoặc tóc của họ.



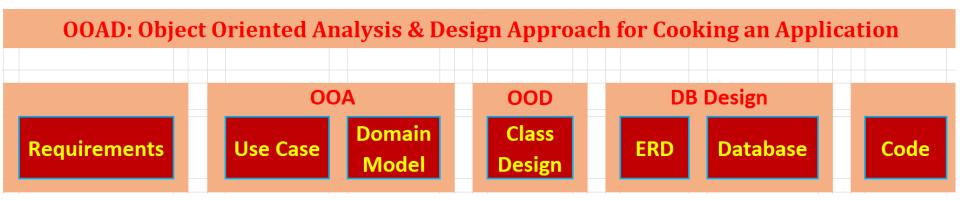
# **User Requirements...**

User Requirements: Các câu phát biểu ngắn gọn, tập trung vào mục tiêu mà người dùng muốn đạt được khi sử dụng hệ thống, chứ không đi sâu vào chi tiết kỹ thuật. Hệ thống cung cấp điều gì – what cho họ?

Ví dụ phần mềm Quản lý bán xe điện thông qua kênh đại lý:

- Khách hàng muốn xem danh mục xe, thông tin chi tiết về xe bao gồm thông tin thông số kĩ thuật và giá bán.
- Khách hàng có thể đọc các review xe, so sánh các mẫu xe theo nhiều tiêu chí khác nhau.
- Dealer có thế tạo đơn hàng, hợp đồng bán hàng.
- Dealer muốn xem danh mục xe, thông tin chi tiết về xe bao gồm thông tin thông số kĩ thuật và giá bán để hỗ trợ tư vấn bán hàng.

## Bí kíp thực chiến SWP391



#### OOAD thực chiến

# **OOAD** = Object-Oriented Analysis (**OOA**) + Object-Oriented Design (**OOD**)

- 1. BƯỚC 1: Yêu cầu người dùng / user requirements
- 2. **BƯỚC 2:** OOA Nhận diện nghiệp vụ (business processes), actor, use case, domain model
- 3. **BƯỚC 3:** OOD Thiết kế lớp, quan hệ, kiến trúc (MVC, DAO, Repository, Singleton...)
- 4. Bước 4: Database Design: ERD, chuẩn hóa (1NF, 2NF, 3NF), PK, FK
- **5. Bước 5:** Code

#### **Bước 1: Requirements**

# Đã done trong webinar #1 ©

#### **Bước 2: Requirements → 00A**

- Là gì?: Bước chuyển từ yêu cầu người dùng, nghiệp vụ sang mô hình hóa hệ thống ở mức chức năng.
- Mục đích: Làm rõ 'ai làm gì' với hệ thống trước khi đi vào đối tượng và dữ liệu.
- Làm gì?: Trích actor, nhóm chức năng, viết use case brief, vẽ Use Case Diagram.
- Tool: Tài liệu đặc tả yêu cầu SRS, FRS, UML Use Case Diagram.
- Ví dụ: Dealer Staff: xem xe, tạo đơn; Dealer Manager: báo cáo, khuyến mãi.

#### Bước 2: Requirements → OOA | Use Case

- Là gì?: Use Case mô tả tương tác Actor ↔ Hệ thống để đạt mục tiêu.
- Mục đích: Định nghĩa biên hệ thống và danh mục chức năng ưu tiên.
- Làm gì?: Xác định actor, use case, mối liên hệ include/extend...; viết kịch bản cơ bản/ngoại lệ.
- Tool: UML Use Case Diagram, Use Case Specification.
- Ví dụ: Use Case: 'Tạo đơn hàng':

Actor: Dealer Staff

**Dòng sự kiện**: chọn xe  $\rightarrow$  áp khuyến mãi  $\rightarrow$  báo giá  $\rightarrow$  xác nhân.

#### Bước 2: Requirements → OOA | Data Domain

- Là gì?: Domain Model (conceptual) biểu diễn thực thể nghiệp vụ và quan hệ, chưa ràng buộc kỹ thuật (field, trường dữ liệu, data type).
- Mục đích: Tạo từ vựng dùng chung giữa nghiệp vụ và kỹ thuật;
   tránh thiếu sót khái niệm chính.
- Làm gì?: Từ danh từ trong requirements → lớp khái niệm; xác định quan hệ 1–N, N–N, ràng buộc.
- Tool: UML Class Diagram (conceptual), ERD conceptual, Glossary.
- **Ví dụ:** Vehicle, Customer, Order, Quotation, Contract, Payment, Promotion, TestDriveSchedule, Feedback.

## Bước 3: 00A → 00D | Class Design

- Là gì?: Thiết kế kỹ thuật chi tiết cho lớp, interface, gói (package) và kiến trúc tương tác.
- Mục đích: Làm cho mô hình sẵn sàng implement, rõ luồng tương tác, tách mối quan tâm (separation of concerns).
- Làm gì?: Thiết kế thuộc tính/phương thức, quan hệ; Sequence Diagram; áp dụng patterns (MVC, DAO, Repository, Factory).
- **Tool:** UML Design Class Diagram, Sequence/Communication, Package/Component/Deployment.
- **Ví dụ:** Package MVC: Model (Order, Customer), Controller (OrderController), View (OrderPage). Method Order.tinhTongTien() tính sau khi áp Promotion.

#### **Bước 4: Database Design**

- Là gì?: Thiết kế lược đồ CSDL lưu trữ bền vững và đảm bảo toàn vẹn dữ liệu.
- Mục đích: Giảm dư thừa, tránh anomaly, tối ưu truy vấn nghiệp vụ chính.
- Tool: ERD tool, SQL DDL, Migration.
- **Ví dụ:** Order(order\_id PK, customer\_id FK, vehicle\_id FK, status); Order\_Promotion(order\_id, promotion\_id).

#### **Harmony: 00AD ↔ Database**

- Là gì?: Sự đồng bộ giữa mô hình hướng đối tượng và lược đồ dữ liệu. Quy tắc ánh xạ mô hình đối tượng sang mô hình quan hệ.
- Mục đích: Giảm mismatch object-relational, nâng tốc độ phát triển và chất lượng. Bảo toàn nghĩa nghiệp vụ khi lưu trữ; tạo nền tảng cho ORM.
- Làm gì?: Review song song Class Diagram & ERD; checklist quy tắc mapping; thử nghiệm CRUD sớm. Class → Table; Attribute → Column; Association → FK; Inheritance → chiến lược (single/joined/table-per-class).
- Tool: UML, ERD mapping, ORM (Hibernate annotations/JPA, EF Core), seed data.
- **Ví dụ:** Customer ↔ Order ↔ Vehicle đồng nhất ở cả Class Diagram và ERD; test 'tạo đơn, áp khuyến mãi, thanh toán'.

Order.customerId (object ref)  $\leftrightarrow$  FK ORDER.CUSTOMER\_ID; Staff kế thừa  $\rightarrow$  Joined Inheritance.

## **Harmony: 00AD ← Database**

OOAD (Object)	Database (Table)	Ví dụ case study
Class	Table	Customer → Customer
Object	Record (Row)	Một khách hàng cụ thể
Attribute	Column	email → email
Association	Foreign Key	Order.customerId FK → Customer.customerId
Inheritance	Table Mapping	Staff (Manager/Staff) → strategy mapping

#### Com thêm: 00A → 00D

#### Tạo mới sản phẩm (Admin)

- 1. Đăng nhập vào hệ thống quản lý.
- Chọn mục 'Quản lý sản phẩm' → Nhấn nút 'Thêm sản phẩm mới'.
- 3. Nhập các thông tin cần thiết (tên sản phẩm, mô tả, giá, tồn kho, danh mục).
- Kiểm tra lại thông tin đã nhập → Nhấn 'Lưu'.
- Hệ thống xác nhận sản phẩm đã được thêm thành công → Cập nhật vào danh sách sản phẩm.

#### Tạo mới sản phẩm (Admin)

- Class: Admin, Product, System.
- Method của Admin: addProduct().
- Method của System: validateProductData(), saveProduct().
- Attributes của Product: name, description, price, stock, category

#### Com thêm: OOA → DB

#### Tạo mới sản phẩm (Admin)

- 1. Đăng nhập vào hệ thống quản lý.
- 2. Chọn mục 'Quản lý sản phẩm'  $\rightarrow$  Nhấn nút 'Thêm sản phẩm mới'.
- 3. Nhập các thông tin cần thiết (tên sản phẩm, mô tả, giá, tồn kho, danh mục).
- 4. Kiểm tra lại thông tin đã nhập  $\rightarrow$  Nhấn 'Lưu'.
- Hệ thống xác nhận sản phẩm đã được thêm thành công → Cập nhật vào danh sách sản phẩm.
- Admin thêm sản phẩm mới với các thông tin: tên, mô tả, giá, tồn kho, danh mục.
- Hệ thống lưu sản phẩm vào cơ sở dữ liệu.

#### Tạo mới sản phẩm (Admin)

- Bảng:
  - Product (id, name, description, price, stock, category\_id)
  - Admin? (id, name, email)
- Khóa ngoại:
  - category\_id liên kết với bảng Category.

# Com thêm: Quy tắc chuyển đổi OOA → DB

- 1. Actor → Thực thể (Entities), Table: Tác nhân hoặc đối tượng chính (như Product, Order, Customer, Admin) sẽ trở thành các thực thể trong ERD.
- 2. Thuộc tính (Attributes), Column: Các thông tin mà Business Flow xử lý (như tên sản phẩm, giá, tồn kho) sẽ là thuộc tính của thực thể, cột của table.
- **3. Quan hệ (Relationships) thành FK:** Các tương tác hoặc liên kết giữa tác nhân và hệ thống được chuyển thành các quan hệ trong ERD (1-N, N-N...).

### Cơm thêm: Nhận diện đối tượng, thực thể

- 1. Entity, Table, Class: Danh từ đại diện cho 1 thứ/object nào đó, nó được mô tả "dài dòng" qua các từ khác
- 2. Relationship: Động từ
- 3. Chen's Notation vs. Crow's Foot Notation



Trường Đại học FPT campus TP. HCM hướng tới sự giao hoà giữa thiên nhiên và con người, giúp sinh viên có môi trường học tập, sáng tạo trong lành và hiệu quả.

# THANK YOU

Thanks to the Internet for the slide template and images