

## Module 2 WBE

### Ứng Dụng Quản Lý Khu Nghỉ Dưỡng Furama

#### I. Mục đích

Học viên sử dụng những kiến thức đã học ở module 2 để viết ứng dụng quản lý Customer, Employees và các dịch vụ của khu nghỉ dưỡng Furama.

#### II. Công nghệ phải dùng

Sau khi kết thúc khóa bài tập, học viên sẽ nắm được các kiến thức sau:

##### Spring Boot

Spring Security
Spring MVC
Spring JPA
Thymeleaf
Spring Session
Spring Validation

#### III. Mô tả hệ thống

Xây dựng hệ thống quản lý khu nghỉ dưỡng Furama tại thành phố Đà Nẵng. Hệ thống được mô tả như sau:

1. Khu nghỉ dưỡng Furama sẽ cung cấp các dịch vụ cho thuê bao gồm Villa, House, Room.

Tất cả các dịch vụ này sẽ bao có các thông tin: Tên dịch vụ (service\_name), Diện tích sử dụng (service\_area), Chi phí thuê (service\_cost), Số lượng người tối đa

(service\_max\_people), Kiểu thuê (rent\_type\_id, bao gồm thuê theo năm, tháng, ngày, giờ).

- Riêng Villa sẽ có thêm thông tin: Tiêu chuẩn phòng (standard\_room), Mô tả tiện nghi khác (description\_other\_convenience), Diện tích hồ bơi (pool\_area), Số tầng (number\_of\_floors).

- Riêng House sẽ có thêm thông tin: Tiêu chuẩn phòng (standard\_room), Mô tả tiện nghi khác (description\_other\_convenience), Số tầng (number\_of\_floors).

- Riêng Room sẽ có thêm thông tin: Dịch vụ miễn phí đi kèm.

Ngoài ra, resort còn cung cấp các dịch vụ đi kèm như massage, karaoke, thức ăn, nước uống, thuê xe di chuyển tham quan resort.

Dịch vụ đi kèm sẽ bao gồm các thông tin: Tên dịch vụ đi kèm (attach\_service\_name), Đơn vị (attach\_service\_unit), Giá tiền (attach\_service\_cost)

2. Hệ thống quản lý còn cho phép quản lý thêm thông tin của Employee resort. Thông tin Employee sẽ bao gồm: Họ tên Employee (employee\_name), Ngày sinh (employee\_birthday), Số CMND (employee\_id\_card), Số ĐT (employee\_phone), Email (employee\_email), Trình độ (education\_degree\_id), Vị trí (position\_id), Lương (employee\_salary).

Trình độ sẽ lưu trữ các thông tin (education\_degree): Trung cấp, Cao đẳng, Đại học và sau đại học

Vị trí sẽ lưu trữ các thông tin (position): Lễ tân, phục vụ, chuyên viên, giám sát, quản lý, giám đốc.

Mỗi Employee sẽ thuộc một bộ phận làm việc (division): Hiện tại resort bao gồm các bộ phận: Sale – Marketing, Hành Chính, Phục vụ, Quản lý.

3. Hệ thống cho phép quản lý thêm thông tin của các Customer đến thuê và sử dụng các dịch vụ của Furama. Thông tin của Customer bao gồm: Họ tên Customer (customer\_name), Ngày sinh (customer\_birthday), Giới tính (customer\_gender), Số CMND (customer\_id\_card), Số Điện Thoại (customer\_phone), Email (customer\_email), Loại khách (customer\_type\_id), Địa chỉ (customer\_address) (resort sẽ phân loại Customer để áp dụng các chính sách phục vụ cho phù hợp).

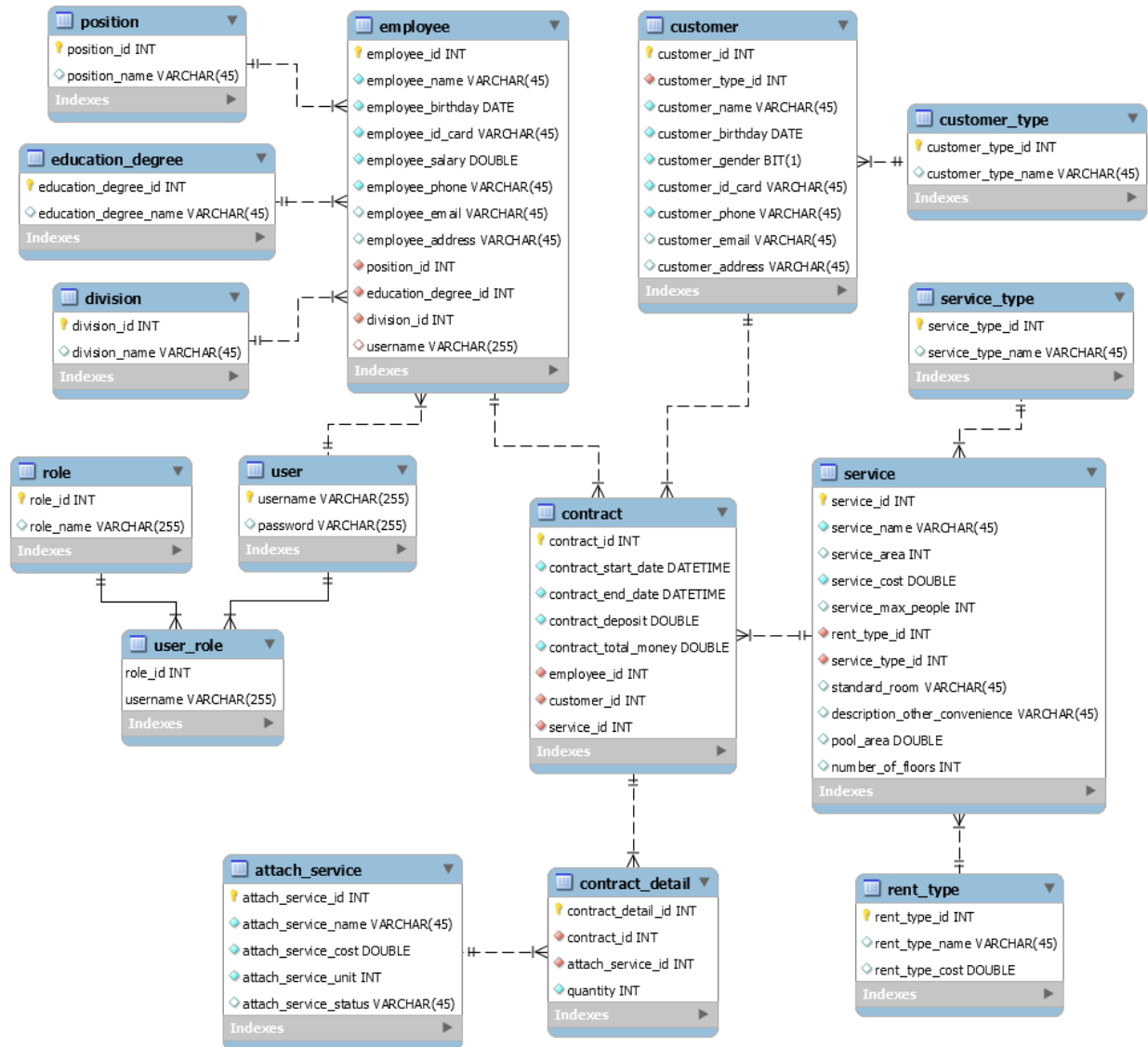
Loại Customer bao gồm (customer\_type): Diamond, Platinum, Gold, Silver, Member.

Customer có thể sử dụng các dịch vụ thuê Villa, House, Phòng và các dịch vụ đi kèm.

Đối với các dịch vụ thuê Villa và House thì Customer cần phải làm hợp đồng thuê với Furama.

Hợp đồng thuê sẽ bao gồm các thông tin: Số hợp đồng (contract\_id), Ngày bắt đầu (contract\_start\_date), Ngày kết thúc (contract\_end\_date), Số tiền cọc trước (contract\_deposit), Tổng số tiền thanh toán (contract\_total\_money).

Với lược đồ cơ sở dữ liệu như bên dưới:



## IV. Yêu cầu hệ thống

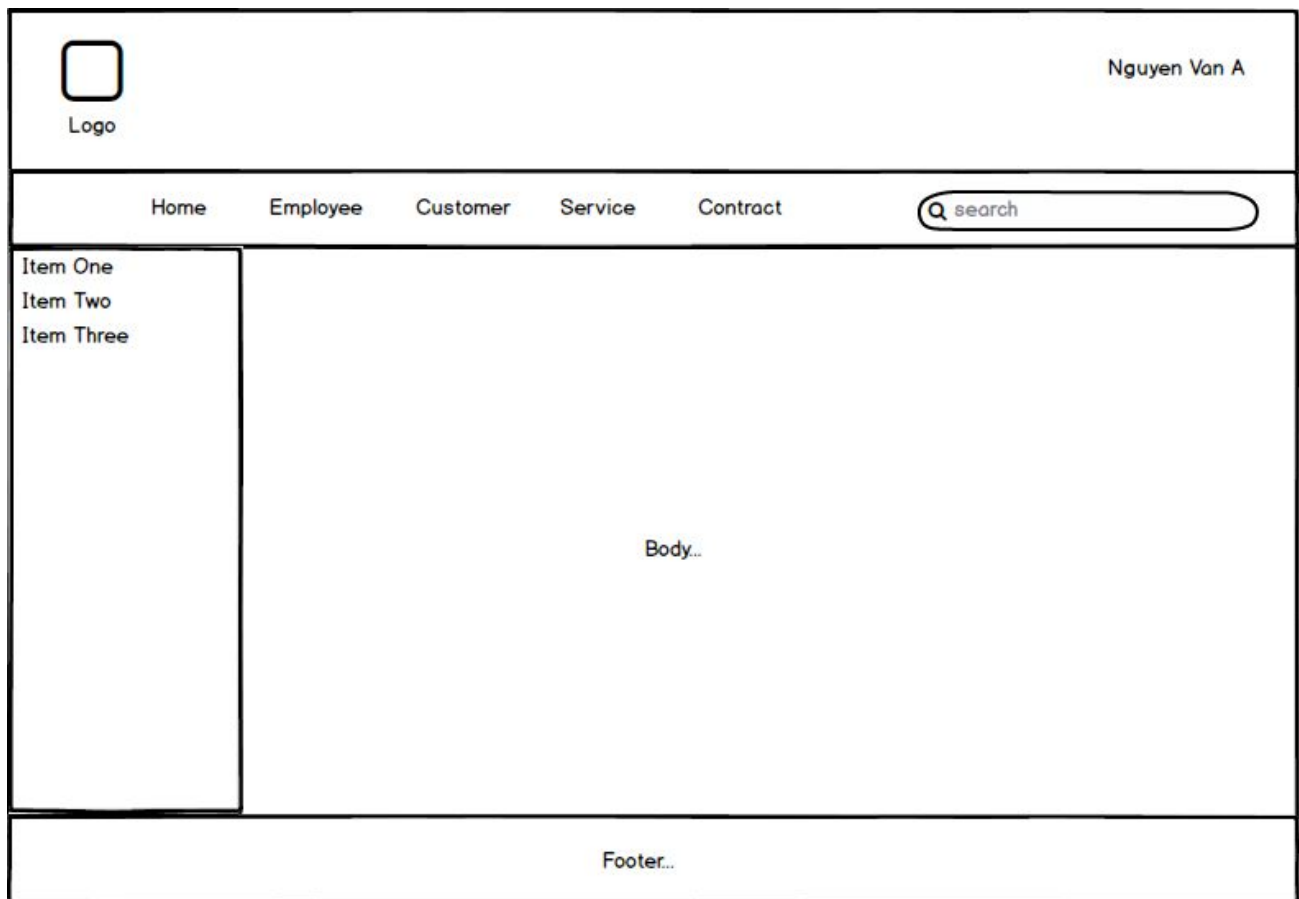
### Yêu cầu 1 (#1. Tạo dự án Spring Boot MVC)

#### Objectives

- Xây dựng được cấu trúc dự án bằng SpringMVC sử dụng SpringBoot
- Biết sử dụng @Controller, @Service
- Biết cấu hình được thymeleaf

Tạo project có tên là Module2\_AccountName\_WBE để thực hiện các yêu cầu sau:

Khi người dùng gõ địa chỉ trên trình duyệt <http://localhost:8080> thì hiển thị ra trang home theo thiết kế dưới đây:



## Yêu cầu 2 (#2. Sử Dụng Spring data repository)

### Objectives

- Sử dụng Spring data (Creation Query , @Query)
- Biết sử dụng @Repository
- Biết sử dụng JpaRepository
- Kết nối database
- Query như tìm kiếm, thêm mới , xóa , sửa , delete

Thiết kế các chức năng với yêu cầu sau:

1. Tạo màn hình thêm mới 1 Khách hàng (sử dụng bảng **customer**)
2. Tạo màn hình liệt kê danh sách Khách hàng (Dạng bảng, có phân trang, nhảy trang, tìm kiếm, liên kết đến các chức năng xóa và chỉnh sửa thông tin của Khách hàng).
3. Tạo màn hình thêm mới 1 Dịch vụ (sử dụng bảng **service**)
4. Tạo màn hình thêm mới 1 Nhân viên (sử dụng bảng **employee**)
5. Tạo màn hình liệt kê danh sách Nhân viên (Dạng bảng, có phân trang, nhảy trang, tìm kiếm, liên kết đến các chức năng xóa và chỉnh sửa thông tin của Nhân viên).
6. Tạo màn hình thêm mới 1 Hợp đồng (sử dụng bảng **contract**)
7. Tạo màn hình thêm mới 1 Hợp đồng chi tiết (sử dụng bảng **contract\_detail**)
8. Tạo màn hình liệt kê danh sách các khách hàng đang sử dụng dịch vụ tại resort (Dạng bảng, có phân trang, nhảy trang, tìm kiếm, liên kết đến các chức năng xóa và chỉnh sửa thông tin của Dịch vụ). Nếu hợp đồng có sử dụng dịch vụ đính kèm thì hãy liệt kê các dịch vụ đính kèm đó ra.  
(sử dụng bảng **customer**, **contract**, **contract\_detail**, **attach\_service**)

## Yêu cầu 3 (#3. Spring Validation)

## **Objectives**

### ● *Sử dụng được Validation bằng Annotation và Custom Validation*

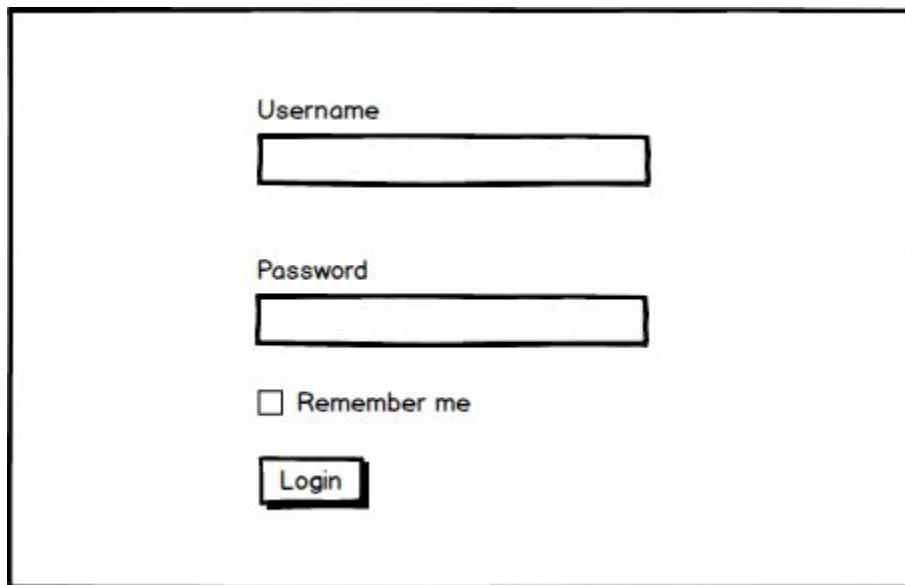
1. Mã Khách hàng có định dạng là KH-XXXX (X là số từ 0-9)
2. Mã Dịch vụ có định dạng là DV-XXXX (X là số 0-9)
3. Số điện thoại phải đúng định dạng 090xxxxxxx hoặc 091xxxxxxx hoặc (84)+90xxxxxxx hoặc (84)+91xxxxxxx
4. Số CMND phải đúng định dạng XXXXXXXXXX hoặc XXXXXXXXXXXX (X là số 0-9)
5. Địa chỉ email phải đúng định dạng email
6. Ngày sinh, Ngày làm hợp đồng, Ngày kết thúc sử dụng datepicker để người dùng có thể chọn ngày thuận tiện. Phải validate xem ngày giờ có hợp lệ hay không (đúng định dạng ngày giờ theo format DD/MM/YYYY).
7. Số lượng, Số tầng phải là số nguyên dương
8. Lương, Giá, Tiền đặt cọc, Tổng tiền phải là số dương
9. Tính tổng tiền sử dụng dịch vụ (trong màn hình 9) dựa vào các dữ liệu cố định (là các dữ liệu giả định của học viên, đang để cố định trên trang HTML, không liên quan đến server).
10. Khi click vào nút xóa (hoặc biểu tượng xóa) trên các dòng ở các màn hình liệt kê thì phải hiển thị thông báo confirm việc người dùng có chắc chắn muốn xóa dữ liệu hay không. Nếu người dùng chọn Yes thì xóa dữ liệu ở dòng đó (Ở đây, chúng ta chỉ giả lập thao tác xóa dữ liệu ở phía client, tức là xóa dữ liệu cứng mà học viên đang để cố định trên trang HTML, không liên quan đến dữ liệu phía server).

## **Yêu cầu 4 (#4. Spring Security, Session và Cookie)**

## Objectives

- Sử dụng được Spring Security để login
- Sử dụng được Spring Session
- Sử dụng được Cookie

1. Tạo màn hình login cho nhân viên, với giao diện mô phỏng như bên dưới:



The image shows a simple login form within a rectangular border. It contains the following elements from top to bottom: a label 'Username' above a text input field; a label 'Password' above a text input field; a checkbox followed by the text 'Remember me'; and a button labeled 'Login'.

Sau khi login thành công thì sẽ chuyển hướng sang home. Sử dụng Session để lưu họ tên nhân viên sau khi đăng nhập và hiển thị tại góc trên bên phải màn hình ở header của tất cả các trang (như giao diện ở Yêu cầu 1).

2. Sử dụng Cookie để lưu tên người dùng (username), password (mật khẩu) khi người dùng chọn checkbox “Remember me”, ngược lại thì sẽ không lưu ở Cookie.

## Yêu cầu 5 (#5. Spring Security)



## **Objectives**

- *Sử dụng được spring security với mysql để phân quyền*

Nếu người dùng là nhân viên “quản lý” hoặc “giám đốc” thì sẽ truy cập được vào trang quản lý Nhân viên, được quyền thêm, sửa, xóa Nhân viên. Ngược lại, các nhân viên khác thì không có quyền truy cập.