Document

Contents

[I. Các nguyên tắc chung 3](#_Toc527905913)

[1. Đặt tên biến 3](#_Toc527905914)

[2. Upload code 3](#_Toc527905915)

[3. Database 3](#_Toc527905916)

[4. Code với các mã lỗi 3](#_Toc527905917)

[5. Các tên viết tắt của các đối tượng 4](#_Toc527905918)

[II. Các yêu cầu khác 5](#_Toc527905919)

[III. Các hàm 6](#_Toc527905920)

[0. Ví dụ: Hàm ….(Để tên chức năng trên đây)….. 6](#_Toc527905921)

[1. Hàm Login dành cho customer 6](#_Toc527905922)

[2. Hàm Logout (Hàm để thoát các loại User đang đăng nhập) 6](#_Toc527905923)

[3. Hàm gọi form đăng ký khách hàng mới 6](#_Toc527905924)

[4. Hàm thực hiện chức năng lưu khách hàng mới 7](#_Toc527905925)

[5. Hàm thực hiện chức năng đăng nhập cho đối tác 7](#_Toc527905926)

[6. Hàm thực hiện chức năng gọi form đăng nhập cho đối tác. 7](#_Toc527905927)

[7. Hàm thực hiện chức năng gọi trang chủ của đối tác. 8](#_Toc527905928)

[8. Hàm thực hiện chức năng gọi form đăng ký cho đối tác. 8](#_Toc527905929)

[9. Hàm thực hiện chắc năng thêm tài khoản mới cho đối tác 8](#_Toc527905930)

[10. Hàm thực hiện chức năng gọi trang chờ check cho đối tác 9](#_Toc527905931)

[IV. Mã lỗi return 10](#_Toc527905932)

# Các nguyên tắc chung

## Đặt tên biến

* Đối với các biến thuộc tính trong các Entity thì giữ nguyên, không thêm các tiền tố loại dữ liệu.
* Đối với các loại biến khác, thêm tiền tố loại dữ liệu ở đầu tên biến.
* Một số loại tiền tố thông dụng:
  + int i
  + long n
  + string str
  + float f
  + boolean b
  + list list

## Upload code

* Code **chỉ được upload** sau khi build **chạy được và không lỗi.**
* Phải comment vào một số chỗ khó hiểu, hay chưa hiểu.
* Nếu có hai hay nhiều người cùng code một hàm, nhớ comment tên mình vào phần code hay chỉnh sửa, kèm lí do nếu đưa code người khác thành comment.
* Khi đã upload code **nhớ phải viết document** về tên hàm, chức năng, tham số truyền vào, tham số trả ra (đối với các hàm chính).

## Database

* Các trường CreateDate sẽ tự động có giá trị ngay lúc add dữ liệu vào database nên khi **INSERT** dữ liệu vào database, không cần đưa thêm giá trị đó. Nếu có, truyền hàm GETDATE() vào cho mysql, **không được nhập tay.** Sau khi đã add dữ liệu xong, **KHÔNG** bao giờ được đụng vô trường CreateDate. Nhớ truyền tham chiếu vào các trường CreateBy,…
* Khi muốn **UPDATE** dữ liệu, **NHỚ** truyền hàm GETDATE vào trường ModifyDate, tham chiếu ModifyBy,…
* **Không nên** xóa dữ liệu. Khi **DELETE**, trong một số bảng có các trường IsActive, phần này có thể hiểu như xóa hay chưa, chỉ cần **UPDATE** trường đó thành không Active, đồng thời nhớ truyền tham chiếu ai là người ở InActiveBy.

## Code với các mã lỗi

* Các mã lỗi được quy định từ
* Mỗi hàm sẽ mặc định chứa 10 số mã lỗi. Nghĩa là người thứ nhất code hàm A, ghi document thì mã lỗi cho hàm đó là 110 cho dù có dùng hết hay không cũng là . Người thứ hai code hàm B thì hàm B sẽ có các mã lỗi là ,…
* **KHI CODE, NHỚ TRY CATCH** để bắt lỗi.

## Các tên viết tắt của các đối tượng

|  |  |
| --- | --- |
| Tên đối tượng | Tên viết tắt |
| Area | AREA |
| City | CITY |
| Comment | CMT |
| DetailOrder | DORD |
| DetailScheduleDate | DSD |
| DetailService | DFS |
| EmployeeAccount | EMP |
| EmployeeManageFService | EMF |
| FService | FSER |
| FunctionProgram | FUNC |
| Image | IMG |
| ImageCategory | IMGC |
| Order | ORD |
| PartnerAccount | PAR |
| QnA | QNA |
| QnACategory | QNAC |
| Role | ROLE |
| RoleFunction | RFUNC |
| ScheduleDate | SD |
| ScheduleManage | SM |
| ServicePost | SP |
| ServicePostDetail | SPD |
| TypeService | TS |
| WebDataContent | WDC |
| WorkingArea | WA |

# Các yêu cầu khác

# Các hàm

## Ví dụ: Hàm ….(Để tên chức năng trên đây)…..

* Tên hàm: Hàm A
* Tham số truyền vào (nhớ tiền tố và nhớ để kiểu dữ liệu phía trước cho người sau nếu cần thì dễ dùng)
  + int iNumberOfFunction: số lượng hàm
* Tham số trả về:
  + string: …….

------- QuyDH Update 20/10/2018------------------

## Hàm Login dành cho customer

* Tên hàm: **public** String customer(String strEmail, String strPass, HttpServletRequest request, Model model)
* Tham số truyền vào:

+ String strEmail: Email khách hàng.

+ String strPass: Password Khách hàng.

+ HttpServletRequest request: request (đối tượng request của website)

+ Model model: Đối tượng của spring.

* Tham số trả về:

+ String: errors => Trả về có đăng nhập được hay không.

## Hàm Logout (Hàm để thoát các loại User đang đăng nhập)

* Tên hàm: public String logout(HttpSession session, HttpServletRequest request)
* Tham số truyền vào:

+ HttpSession session: đối tượng session của java

+ HttpServletRequest request: đối tượng request của java

* Tham số trả về:

+ Chuyển trang và thoát ra nên không có tham số trả về.

## Hàm gọi form đăng ký khách hàng mới

* Tên hàm: public String Create()
* Tham số truyền vào:

+ HttpServletRequest request: request (đối tượng request của website)

+ Model model: Đối tượng của spring.

* Tham số trả về:

+ Điều hướng trang nên không có tham số trả về.

## Hàm thực hiện chức năng lưu khách hàng mới

* Tên hàm: public String save()
* Tham số truyền vào:

+ String customerFullName: Full Name của khách hàng.

+ String customerEmail: Email của khách hàng.

+ String customerPassword: password của khách hàng

+ String customerPassCheck: password nhập lại của khách hàng

+ String customerPhone: Phone của khách hàng

+ String customerAddress: Địa chỉ khách hàng

+ String customerDOB: Ngày sinh khách hàng.

+ String customerBankAccount: Số tài khoản ngân hàng của khách hàng.

+ String customerImage: Ảnh đại diện của khách hàng

+ String customerIdentityNumber: IdentityNumber của khách hàng

+ Model model: Đối tượng của Spring

* Tham số trả về:

+ String errors => “Email đã tồn tại” và “Đăng ký không thành công”.

## Hàm thực hiện chức năng đăng nhập cho đối tác

* Tên hàm: public String partner() – Đường dẫn: login/partner
* Tham số truyền vào:

+ String strEmail: Email của đối tác

+ String strPass: Password của đối tác

+ Model model: Đối tượng của spring

+ HttpServletRequest request: Đối tượng request của java

* Tham số trả về:

+ String redirect:/partner/login – Đường dẫn về form đăng nhập của đối tác.

+ String redirect:/partner/ - Đăng nhập thành công dẫn tới trang chủ của đối tác.

## Hàm thực hiện chức năng gọi form đăng nhập cho đối tác.

* Tên hàm: public String login() – Đường dẫn: partner/login
* Tham số truyền vào:

+ HttpServletRequest request: Đối tượng request của java.

+ Model model: đối tượng của spring

* Tham số trả về:

+ String redirect:/partner/home – Đường dẫn tới trang chủ của đối tác ( trường hợp đã có session Đối tác đăng nhập ).

+ String partner/login: gọi trang đăng nhập của đối tác.

## Hàm thực hiện chức năng gọi trang chủ của đối tác.

* Tên hàm: public String home() – Đường dẫn: partner/ , partner/home, partner
* Tham số truyền vào:

+ HttpServletRequest request: Đối tượng request của java.

+ Model model: đối tượng của spring.

* Tham số trả về:

+ String redirect:/partner/login – Chuyển hướng tới trang đăng nhập ( trường hợp chưa đăng nhập ).

+ String partner/waitforcheck: Gọi trang đợi check dành cho tối tác đã tạo tài khoảng nhưng chưa được duyệt.

+ String partner/home: Gọi trang chủ dành cho đối tác.

## Hàm thực hiện chức năng gọi form đăng ký cho đối tác.

* Tên hàm: public String create() – Đường dẫn: partner/create
* Tham số truyền vào:

+ HttpServletRequest request: Đối tượng request của java.

+ Model model: đối tượng của spring.

* Tham số trả về:

+ String redirect:/partner/ - Chuyển tới trang chủ của đối tác khi đối tác đã đăng nhập rồi.

+ String partner/create: Gọi trang đăng ký cho đối tác.

## Hàm thực hiện chắc năng thêm tài khoản mới cho đối tác

* Tên hàm: public String save() – Đường dẫn: partner/save
* Tham số truyền vào:

+ String partnerEmail: Email của đối tác

+ String partnerFullName: Tên đầy đủ của đối tác

+ String password: mật khẩu

+ String passwordcheck: mật khẩu nhập lại

+ String partnerIdentityNumber: cmnd đối tác

+ String partnerImage: đường dẫn ảnh đại diện

+ String partnerPhone: số điện thoại.

+ String partnerDOB: Ngày sinh của đối tác.

+ String partnerAddress: Địa chỉ của đối tác

+ String partnerBankAccount: Số tài khoảng ngân hàng.

+ String partnerType: Loại đối tác (0 – 1 )

+ Model model

+ HttpServletRequest request

* Tham số trả về:

+ String redirect:/partner/ - Chuyển hướng tới trang chủ nếu đã đăng nhập.

+ String redirect:/partner/create – chuyển lại trang đăng ký nếu để trống 1 trường nào đó hoặc email đã tồn tại hoặc password nhập lại không trùng với password hoặc không thêm vào thành công ( lỗi hệ thống).

+ String partner/waitforcheck: gọi trang chờ check khi được thêm thành công.

## Hàm thực hiện chức năng gọi trang chờ check cho đối tác

* Tên hàm: public String wait()

1. Hàm thực hiện chức năng đăng nhập cho nhân viên:

* Tên hàm: public String employee()
* Tham số truyền vào:

+ String strEmail: Email của nhân viên.

+ String strPass: Password của nhân viên

+ Model model

+ HttpServletRequest request

* Tham số trả về:

+ String redirect:/employee/login: chuyển về trang đăng nhập của nhân viên.

+String redirect:/employee/home: chuyển về trang chủ của nhân viên.

-------------QuyDH End Update------------------------

# Mã lỗi return

* Mã lỗi cho hàm A từ

|  |  |
| --- | --- |
| Mã lỗi | Mô tả |
| -9 | Lỗi ABCXYZ |
| -8 | Lỗi KING,… |