**\*Chức năng của ứng dụng web**

Ứng dụng web được thiết kế nhằm hỗ trợ hoạt động đặt tour du lịch Việt Nam, mang lại trải nghiệm liền mạch và hiệu quả cho cả khách hàng, nhân viên và quản trị viên. Các chức năng chính bao gồm:

* **Đối với customer:**
  + Lọc tour du lịch theo vùng miền.
  + Thêm tour du lịch vào giỏ hàng.
  + Thanh toán tiện lợi với nhiều tùy chọn, đảm bảo quy trình đặt tour, search
* **Đối với seller:**
  + Có thẻ chỉnh sửa tên ảnh giá số lượng miêu tả của tour
  + Tạo tour mới cho khách hang , bao gồm thông tin sản phẩm, số lượng và tổng tiền
* **Đối với admin:**
  + Quản lý toàn diện dữ liệu của hệ thống, bao gồm thông tin tour du lịch, danh sách tour , và tài khoản người dùng.
  + Cập nhật, chỉnh sửa hoặc xóa dữ liệu linh hoạt để đảm bảo thông tin luôn chính xác và phù hợp với nhu cầu kinh doanh.

#### 

**\*Mô tả luồng dữ liệu**  
  
Ứng dụng web được thiết kế để quản lý quản lý tour du lịch vòng quanh Việt Nam, bao gồm các chức năng như xem sản phẩm, quản lý giỏ hàng, thanh toán, quản lý tài khoản người dùng, bán hàng cho nhân viên và quản trị hệ thống dành cho admin. Dưới đây là mô tả chi tiết các luồng dữ liệu tương ứng với từng chức năng.

#### ****1. Chức năng khách hàng****

##### **1.1. Xem thông tin tour du lịch**

* **Luồng dữ liệu**:
  1. Người dùng chọn một sản phẩm bất kỳ → Gửi yêu cầu GET tới detail?tourID=x
  2. Hệ thống truy vấn cơ sở dữ liệu (DB) để lấy thông tin sản phẩm dựa trên tourID
  3. Chuyển tiếp (forward) dữ liệu tới giao diện Views/TourDetail.jsp để hiển thị.

##### **1.2. Thêm sản phẩm vào giỏ hàng**

* **Luồng dữ liệu**:
  1. Người dùng nhấn nút "Thêm vào giỏ hàng" trên trang sản phẩm → Gửi yêu cầu GET tới tour.
  2. Servlet doGETproduct xử lý, tạo cookie chứa thông tin giỏ hàng.
  3. Chuyển hướng (sendRedirect) tới doGet CartDetailServlet.
  4. Servlet doGet cart đọc dữ liệu từ cookie, truy vấn DB để lấy chi tiết sản phẩm.
  5. Chuyển tiếp (forward) tới Views/cart.jsp để hiển thị giỏ hàng.

##### **1.3. Xóa sản phẩm khỏi giỏ hàng**

* **Luồng dữ liệu**:
  1. Người dùng nhập yêu cầu xóa sản phẩm trong giỏ hàng → Gửi yêu cầu GET tới cart.
  2. Servlet doGET cart xóa các cookie liên quan đến sản phẩm bị xóa.
  3. Chuyển hướng (sendRedirect) về CartDetailServlet để cập nhật giỏ hàng.
  4. Servlet doGet cart đọc dữ liệu từ cookie, truy vấn DB để lấy chi tiết sản phẩm.
  5. Chuyển tiếp (forward) tới Views/cart.jsp để hiển thị giỏ hàng.

##### **1.4. Thanh toán đơn hàng**

* **Luồng dữ liệu**:
  1. Người dùng nhấn vào **Cart rồi dẫn đến trang**  "cartdetail" lấy ra từ cookie → Gửi yêu cầu POST tới CheckoutServlet.
  2. Servlet doPOST CheckoutServlet xử lý dữ liệu từ cookie giỏ hàng → Chuyển tiếp (forward) tới /thank.

#### ****2. Chức năng tài khoản người dùng****

##### **2.1. Đăng xuất**

* **Luồng dữ liệu**:
  1. Tại trang chủ (home), người dùng nhấn "Đăng xuất" → Gửi yêu cầu GET tới LogoutServlet.
  2. Servlet doGet logout xóa session người dùng.
  3. Chuyển hướng (sendRedirect) tới trang chủ (home).

#### ****3. Chức năng bán hàng (dành cho nhân viên)****

##### **3.1. Trang quản lý tour**

* **Luồng dữ liệu**:
  1. Nhân viên nhấn liên kết " /managetour" → Gửi yêu cầu GET ManageTourServlet.
  2. Servlet doGet ManageTourServlet truy vấn DB lấy thông tin sản phẩm thuộc cơ sở của nhân viên
  3. Chuyển tiếp (forward) tới Views/ ManageTour.jsp.

##### **3.2. Thêm tour**

* **Luồng dữ liệu**:
  1. Nhân viên nhấn liên kết " /managetour" → Gửi yêu cầu GET ManageTourServlet.
  2. **Click vào bution Add new Tour gửi yêu cầu doPost tới AddTourServlet nhân viên sẽ nhập đầy đủ thông tin vào form** xử lý form theo method doPost rồi sendirect tới trang / managetour

##### **3.3. Edit tour**

* **Luồng dữ liệu**:
  1. Nhân viên nhấn liên kết " /managetour" → Gửi yêu cầu GET ManageTourServlet.
  2. Nhân viên click chọn tour cần chỉnh sửa sẽ xuất hiện đường dẫn edittour?tourID=x gửi yêu cầu doGet tới EditTourServlet Chuyển tiếp (forward) /Views/EditTour.jsp form có thể chỉnh sửa gưi yêu cầu doPost ManageTourServlet sendirect tới trang / managetour

##### **3.4. quản lý đặt tour**

* **Luồng dữ liệu**:
  1. Nhân viên nhấn liên kết " /manageorder " → Gửi yêu cầu GET ManageOrderServlet.
  2. Servlet doGet ManageOrderServlet truy vấn DB lấy thông tin sản phẩm thuộc cơ sở của nhân viên
  3. Chuyển tiếp (forward) tới Views/ ManageOrder.jsp.

##### **3.5. quản lý đặt tour**

* **Luồng dữ liệu**:
  1. Nhân viên nhấn liên kết " /manageorder " → Gửi yêu cầu GET ManageOrderServlet.
  2. Nhân viên click chọn tour cần chỉnh sửa sẽ xuất hiện đường dẫn processorder?id=x gửi yêu cầu doGet tới EditTourServlet Chuyển tiếp (forward) Chuyển tiếp (forward) tới Views/ ProcessOrder.jsp xử lý form theo phương thức Post sendirect tới trang / manageorder.

#### ****4. Chức năng quản trị (dành cho admin)****

##### **4.1. Trang quản trị**

* **Luồng dữ liệu**:
  1. Tại trang chủ, admin nhấn "dashboard→ Gửi yêu cầu admin/control.
  2. Chuyển tiếp (forward) tới Views/Dashboard.jsp.

##### **4.2. Quản lý thông báo**

* **Luồng dữ liệu**:
  1. Admin chọn " Notification" → Gửi yêu cầu GET tới VịewLogServlet
  2. Servlet truy vấn DB lấy danh sách nhân viên → Chuyển tiếp (forward) tới Views/ Notification.jsp.

##### **4.3. Quản lý đặt tour**

* **Luồng dữ liệu**:
  1. Admin chọn "Order" → Gửi yêu cầu GET tới /manageorder.
  2. Servlet truy vấn DB lấy danh sách khách hàng → Chuyển tiếp (forward) tới Views/adminControl.jsp.
  3. **Xóa khách hàng**: Gửi yêu cầu GET tới /manageorder**4.4. Quản lý sản phẩm**
* **Luồng dữ liệu**:
  1. Admin chọn "Manage Tour
  2. " → Gửi yêu cầu GET tới /managetour Servlet truy vấn DB lấy danh sách sản phẩm → Chuyển tiếp (forward) tới Views/ManageTour.jsp.
  3. **Chỉnh sửa tour**:
     + Gửi yêu cầu ?edittour?tourID=x →xử lý với method POST response sendirect tới Views/EditTour.jsp.
     + Nhập dữ liệu → manage tour (như seller)
  4. **Thêm tour**: như seller

##### **4.5. Quản lý vùng miền**

* **Luồng dữ liệu**:
  1. Admin chọn "Manager Category
  2. " → Gửi yêu cầu GET tới /managecategory.
  3. Servlet truy vấn DB lấy danh sách → Chuyển tiếp (forward) tới Views/adminControl.jsp.
  4. **Xóa đơn hàng**: Gửi yêu cầu GET tới admin/control?action=deleteOrder → Xóa dữ liệu trong DB → Chuyển hướng (sendRedirect) về admin/control?action=orders.
  5. **Xem chi tiết**: Gửi yêu cầu ?action=viewOrder → Lấy dữ liệu → Chuyển tiếp (forward) tới Views/adminControl.jsp.

##### **4.6. Quản lý tài khoản**

* **Luồng dữ liệu**:
  1. Admin chọn "Categories" → Gửi yêu cầu GET tới /manageaccount
  2. Servlet truy vấn DB lấy danh sách danh mục → Chuyển tiếp (forward) tới Views/ManageAccount.jsp.
  3. **Sửa danh mục**:
     + Gửi yêu cầu /editaccount?accountID → Chuyển tiếp (forward) tới Views/EditAccount jsp.
     + Nhập dữ liệu → Gửi POST tới EditAccountServelet →chỉnh sửa quyền.
     + Gửi POST tới admin/control → Cập nhật DB.
  + **Cột**:
    - warehouseID (int, NOT NULL): Mã kho (FK tham chiếu Warehouse).
    - productID (int, NOT NULL): Mã sản phẩm (FK tham chiếu Products).
    - quantity (int, NOT NULL): Số lượng tồn kho.
  + **Ràng buộc**:
    - Khóa chính: (productID, warehouseID).
    - Khóa ngoại: warehouseID, productID (xóa theo CASCADE).

1. **PaymentMethods (Phương thức thanh toán)**
   * **Mục đích**: Lưu trữ các phương thức thanh toán.
   * **Cột**:
     + paymentMethodID (int, PK, IDENTITY): Mã phương thức.
     + methodName (nvarchar(50), NOT NULL, UNIQUE): Tên phương thức.
   * **Ràng buộc**: Khóa chính là paymentMethodID.
2. **Orders (Đơn hàng)**
   * **Mục đích**: Quản lý thông tin đơn hàng.
   * **Cột**:
     + orderID (int, PK, IDENTITY): Mã đơn hàng.
     + orderDate (date, NOT NULL): Ngày đặt hàng.
     + total (int, NOT NULL): Tổng tiền.
     + orderType (bit, NOT NULL): Loại đơn hàng (0: bán, 1: nhập).

