**🞟🞟🕮🞟🞟**

**TRƯỜNG ĐẠI HỌC SÀI GÒN**

*KHOA CÔNG NGHỆ THÔNG TIN*



**BÀI TIỂU LUẬN**

XÂY DỰNG PHẦN MỀM THEO MÔ HÌNH PHÂN LỚP

Giảng viên hướng dẫn: Cao Minh Thành

Sinh viên thực hiện:

Lý Xuân Sang

3118410363

DCT1189

TP.HCM , ngày 24 tháng 06 năm 2021

NHẬN XÉT CỦA GIẢNG VIÊN HƯỚNG DẪN \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Hồ Chí Minh, ngày … tháng … năm 2021

Giảng viên hướng dẫn

***LỜI CẢM ƠN***

Với lòng biết ơn sâu sắc và sự kính trọng cho phép em gửi lời cảm ơn chân thành nhất tới:

– Trường Đại học Sài Gòn , khoa Công nghệ thông tin cùng các giảng viên đã tận tình chỉ dạy và tạo điều kiện giúp đỡ em trong quá trình học tập, nghiên cứu và hoàn thành đề tài.

– Đặc biệt nhóm em xin bày tỏ lòng biết ơn sâu sắc đến thầy ***Cao Minh Thành*** – giảng viên hướng dẫn và cũng là người đã luôn tận tình hướng dẫn, chỉ bảo, giúp đỡ và động viên nhóm em trong suốt quá trình nghiên cứu và hoàn thành tiểu luận này.

– Cảm ơn bạn bè và các thành viên trong nhóm đã luôn khích lệ, động viên và giúp đỡ nhau trong quá trình học tập và nghiên cứu,hoàn thành đề tài.

Mặc dù đã cố gắng rất nhiều, nhưng bài luận không tránh khỏi những thiếu sót; nhóm em rất mong nhận được sự thông cảm, chỉ dẫn, giúp đỡ và đóng góp ý kiến của các nhà khoa học, của quý thầy cô, các cán bộ quản lý và các bạn đồng nghiệp.

Em xin chân thành cảm ơn!

***Người thực hiện***

*Lý Xuân Sang*

**Mục lục**

**Chương 1 :Trình bày mô hình MVC**

1. Giới thiệu mô hình
2. Lịch sử phát triển
3. Thành phần của mô hình
4. Quy trình hoạt động của mô hình
5. Ưu điểm,hạn chế của mô hình
6. Code minh họa mô hình

**Chương 2: Ứng dụng quản lý danh bạ nhân viên**

1. Yêu cầu chức năng của ứng dụng
2. Giao diện ứng dụng
3. Source code ứng dụng

***Chương 1 :Trình bày mô hình MVC***

1. **Giới thiệu MVC**

MVC là tên viết tắt của Model-View-Controller là một mô hình kiến trúc ngăn cách một ứng dụng thành 3 phần logic chính: mô hình, quan điểm và bộ điều khiển. Mỗi thành phần sẽ có một nhiệm vụ riêng biệt và độc lập với những thành phần khác.

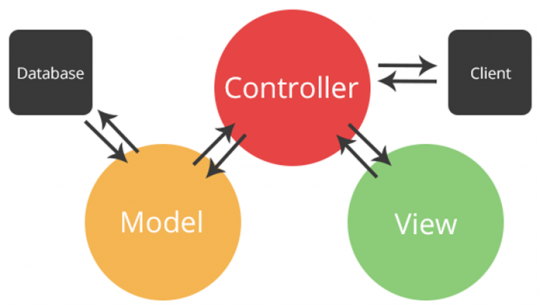
MVC cung cấp các phần cơ bản để thiết kế một chương trình cho máy tính để bàn hoặc thiết bị di động, cũng như các ứng dụng web. Mô hình này hoạt động tốt với lập trình hướng đối tượng, vì những mô hình, khung nhìn và bộ điều khiển khác nhau thì có thể được coi là đối tượng và chúng được sử dụng lại trong một ứng dụng.

1. **Lịch sử phát triển**

Năm 1979 kiến trúc MVC được thảo luận lần đầu tiên bởi Trygve Reenskaug. Đến năm 1987, mô hình MVC được giới thiệu lần đầu bằng ngôn ngữ lập trình Smalltalk.

MVC lần đầu được chấp nhận là một khái niệm chung trong một bài viết vào năm 1988. MVC được sử dụng rộng rãi trong các ứng dụng web hiện đại trong thời gian gần đây.

1. **Thành phần của mô hình MVC**

****

**Model:**

Phần Model của kiến trúc MVC là thành phần chính đảm nhiệm chức năng lưu trữ dữ liệu và các bộ phận logic liên quan của toàn bộ ứng dụng. Model chịu trách nhiệm cho các thao tác dữ liệu giữa Controller hoặc bất kỳ logic nghiệp vụ liên quan nào khác như: cho phép xem, truy xuất dữ liệu, … Ví dụ, Controller sẽ lấy thông tin khách hàng từ cơ sở dữ liệu. Model sẽ thực hiện các thao tác dữ liệu và gửi lại cơ sở dữ liệu hoặc sử dụng nó cho View.

**View:**

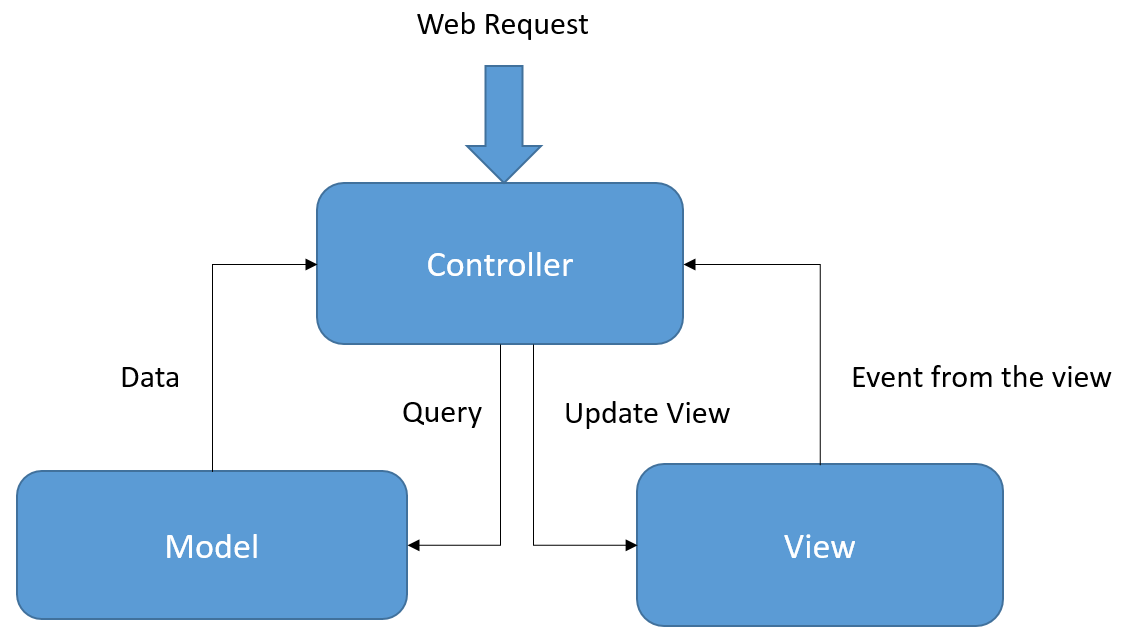
View là một phần của ứng dụng chịu trách nhiệm cho việc trình bày dữ liệu. Thành phần này được tạo bởi dữ liệu thu thập từ dữ liệu mô hình, và giúp người dùng có cái nhìn trực quan về trang web, cũng như ứng dụng.

View cũng đại diện cho dữ liệu từ các cuộc trò chuyện, sơ đồ và bảng. Ví dụ: bất kỳ View nào cũng sẽ bao gồm tất cả các thành phần giao diện người dùng như một nút bấm, khung nhập, menu, hình ảnh, ..

**Controller:**

Đây là phần xử lý tương tác người dùng của ứng dụng. Controller xử lý dữ liệu đầu vào từ bàn phím và chuột của người dùng sau đó thông báo tới View và Model. Controller gửi các lệnh tới Model để thay đổi trạng thái của Model (Ví dụ: lưu một tài liệu cụ thể). Controller cũng gửi các lệnh tương tự tới View để thực hiện các thay đổi về giao diện.

1. **Quy trình hoạt động của mô hình MVC**



•  Khi có yêu cầu từ phía client gửi đến server  
⇒ Controller sẽ tiếp nhận và xử lý yêu cầu đó  
⇒ Nếu có yêu cầu lấy dữ liệu thì nó sẽ điều hướng đến phương thức xử lý ở Model  
⇒ Model nhận thông tin, thực thi các yêu cầu và trả kết quả về lại cho Controller  
⇒ Controller sẽ xử lý kết quả và trả về View tương ứng  
⇒ Khi nhận được dữ liệu từ Controller thì View sẽ xây dựng các thành phần hiển thị và trả về GUI Content để Controller trả kết quả về trình duyệt của client  
⇒ Trình duyệt nhận giá trị trả về và hiển thị lên màn hình cho người dùng.

1. **Ưu điểm, hạn chế của mô hình MVC**

**Ưu điểm**

* Hỗ trợ quá trình phát triển nhanh chóng: Với đặc điểm hoạt động độc lập của từng thành phần, các lập trình viên có thể làm việc đồng thời trên từng bộ phận khác nhau của mô hình này. MVC giúp bạn tiết kiệm rất nhiều thời gian.
* Khả năng cung cấp đồng thời nhiều khung View: Với mô hình MVC, bạn có thể tạo ra đồng thời nhiều khung View cho Model.
* Hỗ trợ các kỹ thuật không đồng bộ: MVC có thể hoạt động trên nền tảng JavaScript. Điều này có nghĩa là các ứng dụng MVC có thể hoạt động với các file PDF, các trình duyệt web cụ thể, và cả các widget máy tính.
* Dễ dàng thao tác chỉnh sửa: Bộ phận Model hoạt động tách biệt với View đồng nghĩa với việc bạn có thể đưa ra các thay đổi, chỉnh sửa hoặc cập nhật dễ dàng ở từng bộ phận.
* Giữ nguyên trạng thái data: Mô hình MVC truyền lại dữ liệu nhưng không định dạng lại dữ liệu. Do đó, các dữ liệu này có thể được dùng lại cho các thay đổi sau này.
* Hỗ trợ các nền tảng phát triển SEO: Với mô hình MVC, bạn có thể dễ dàng tạo ra các mã SEO URL để thu hút lượng truy cập đối với ứng dụng bất kỳ.

**Hạn chế**

* Khó khăn trong quá trình điều hướng code: Điều hướng khung có thể phức tạp vì mô hình này bao gồm nhiều lớp và yêu cầu người dùng thích ứng với các tiêu chí phân tách của MVC.
* Không thích hợp việc phát triển các ứng dụng nhỏ vì mô hình này yêu cầu bạn lưu trữ một số lượng lớn các file.
* Nhiều khung hoạt động đồng thời: Việc phân tách một tính năng thành ba bộ phận khác nhau dễ dẫn đến hiện tượng phân tán. Do đó, đòi hỏi các nhà phát triển phải duy trì tính nhất quán của nhiều bộ phận cùng một lúc.

**VI. Code minh họa mô hình**

Chức năng lọc sản phẩm theo nhóm (LCD, HDD, RAM, CPU…) và theo nhà sản xuất (SamSung, Intel, Kingmax…) theo schema dưới đây:

**Sanpham** (sp\_id, sp\_ten, sp\_nhom, sp\_chitiet, sp\_nhasanxuat)

**Sơ đồ sequence**

|  |
| --- |
| **s** |

**Code minh họa**

*Lớp Model*

|  |
| --- |
| public partial class Sanpham  {  public int SpId { get; set; }  public string SpTen { get; set; }  public string SpNhom { get; set; }  public string SpChitiet { get; set; }  public string SpNhasanxuat { get; set; }  } |

*Lớp View*

|  |
| --- |
| @using Sanpham.Models;  @model IEnumerable<Sanpham>  @{  ViewData["Title"] = "Index";  }  <h1>Index</h1>  @using (Html.BeginForm())  {  <form action="Index">  <div class="form\_\_item">  <label for="\_nhom">Nhóm</label>  <select name="\_nhom">  <option **value**="">All</option>  <option **value**="LCD">LCD</option>  <option **value**="RAM">RAM</option>  <option **value**="HDD">HDD</option>  <option **value**="CPU">CPU</option>  </select>  </div>  <div class="form\_\_item">  <label for="\_nhasanxuat">Nhà sản xuất</label>  <select name="\_nhasanxuat">  <option **value**="">All</option>  <option **value**="Sony">Sony</option>  <option **value**="Samsung">Samsung</option>  <option **value**="Kingmax">Kingmax</option>  <option **value**="Intel">Intel</option>  </select>  </div>  <div class="form\_\_item">  <input type="submit" value="Filter">  </div>  </form>  }  <table class="table">  <thead>  <tr>  <th>  Tên Sản Phẩm  </th>  <th>  Nhóm  </th>  <th>  Chi Tiết  </th>  <th>  Nhà Sản xuất  </th>  </tr>  </thead>  <tbody>  @foreach (var item in Model)  {  <tr>  <td>  @Html.DisplayFor(modelItem => item.SpTen)  </td>  <td>  @Html.DisplayFor(modelItem => item.SpNhom)  </td>  <td>  @Html.DisplayFor(modelItem => item.SpChitiet)  </td>  <td>  @Html.DisplayFor(modelItem => item.SpNhasanxuat)  </td>  </tr>  }  </tbody>  </table> |

*Lớp Controllers*

|  |
| --- |
| public class SanphamController : Controller  {  private readonly MHPL\_HKContext context = new MHPL\_HKContext();  //GET: SanphamController  public ActionResult Index(string \_nhom, string \_nhasanxuat)  {  var listSanpham = context.Sanphams.ToList(); // Lấy tất cả  if (!String.IsNullOrEmpty(\_nhom)) // Lọc 1 điều kiện  {  listSanpham = context.Sanphams.Where(s => s.SpNhom == \_nhom).ToList();  }  if (!String.IsNullOrEmpty(\_nhasanxuat)) // Lọc 1 điều kiện  {  listSanpham = context.Sanphams.Where(s => s.SpNhasanxuat == \_nhasanxuat).ToList();  }  if(!String.IsNullOrEmpty(\_nhom) && !String.IsNullOrEmpty(\_nhasanxuat)) //Lọc 2 điều kiện  {  listSanpham = context.Sanphams.Where(s => s.SpNhom == \_nhom && s.SpNhasanxuat == \_nhasanxuat).ToList();  }  return View(listSanpham);  }  } |

***Chương 2: Ứng dụng quản lý danh bạ nhân viên***

**I. Yêu cầu chức năng của ứng dụng**

Một nhân viên được tuyển dụng sẽ làm việc trong một phòng nào đó như Kinh doanh, Kế toán, Nhân sự, Lễ tân, Bảo trì. Trong quá trình làm việc, nhân viên có thể được chuyển sang phòng khác để làm tiếp hoặc có thể nghỉ việc.

Lãnh đạo công ty muốn xây dựng ứng dụng web để quản lý danh bạ của nhân viên, đồng thời có thể gửi email cho một hoặc nhiều nhân viên cùng lúc một cách thuận lợi, nhanh chóng.

Anh (chị) hãy xây dựng ứng dụng trên với các chức năng tối thiểu như sau:

1. Quản lý thông tin nhân viên: xem danh sách, thêm mới, sửa, xóa, gán trạng thái [đang làm, đã nghỉ], tìm kiếm nhân viên theo tên.

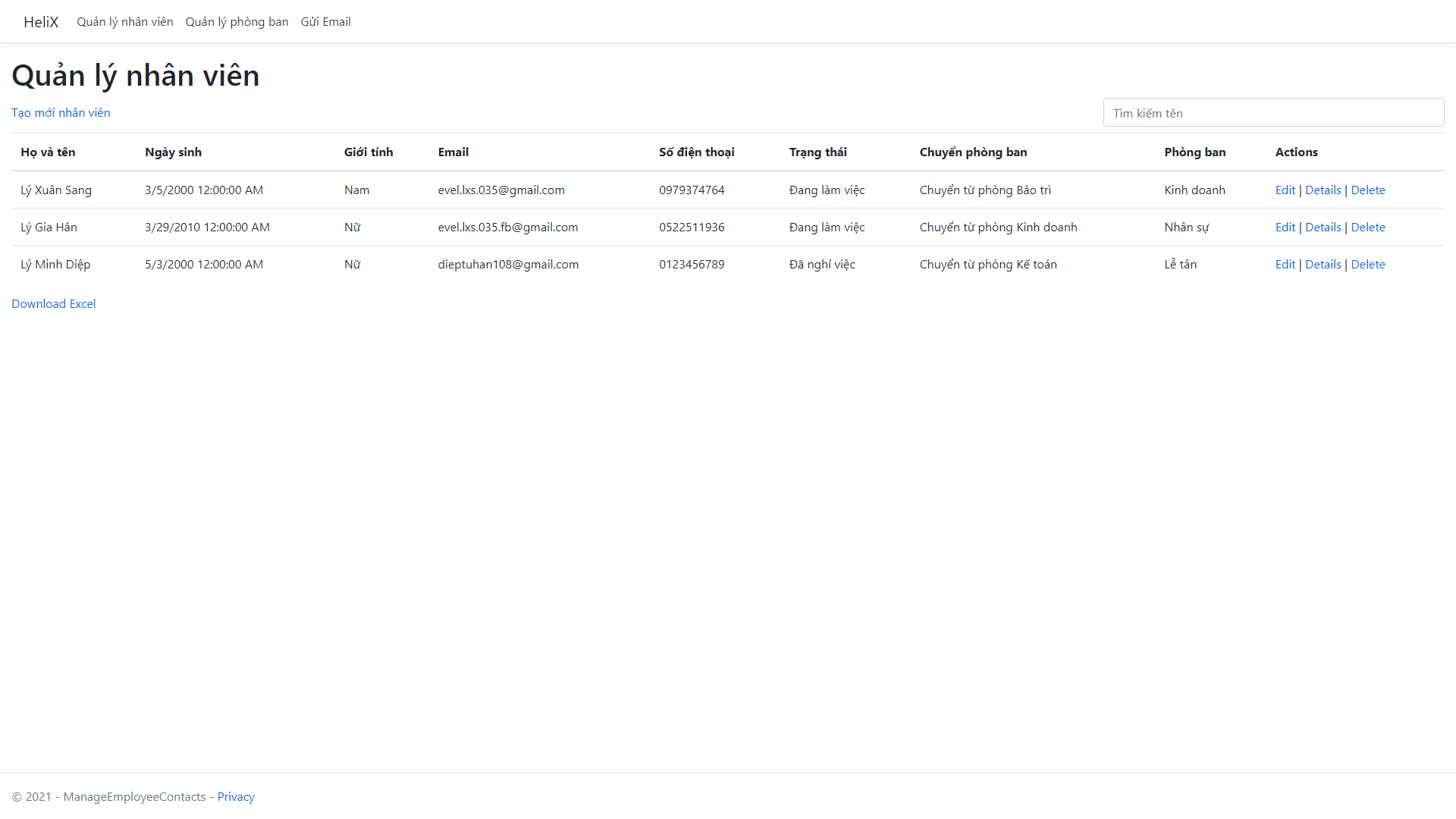
2. Quản lý thông tin phòng làm việc: xem danh sách các phòng, thêm mới, sửa, xóa, xem nhân viên trong một phòng, xuất danh sách thành tập tin xlsx.

3. Quản lý nhân viên được tuyển vào/chuyển sang làm việc trong một phòng.

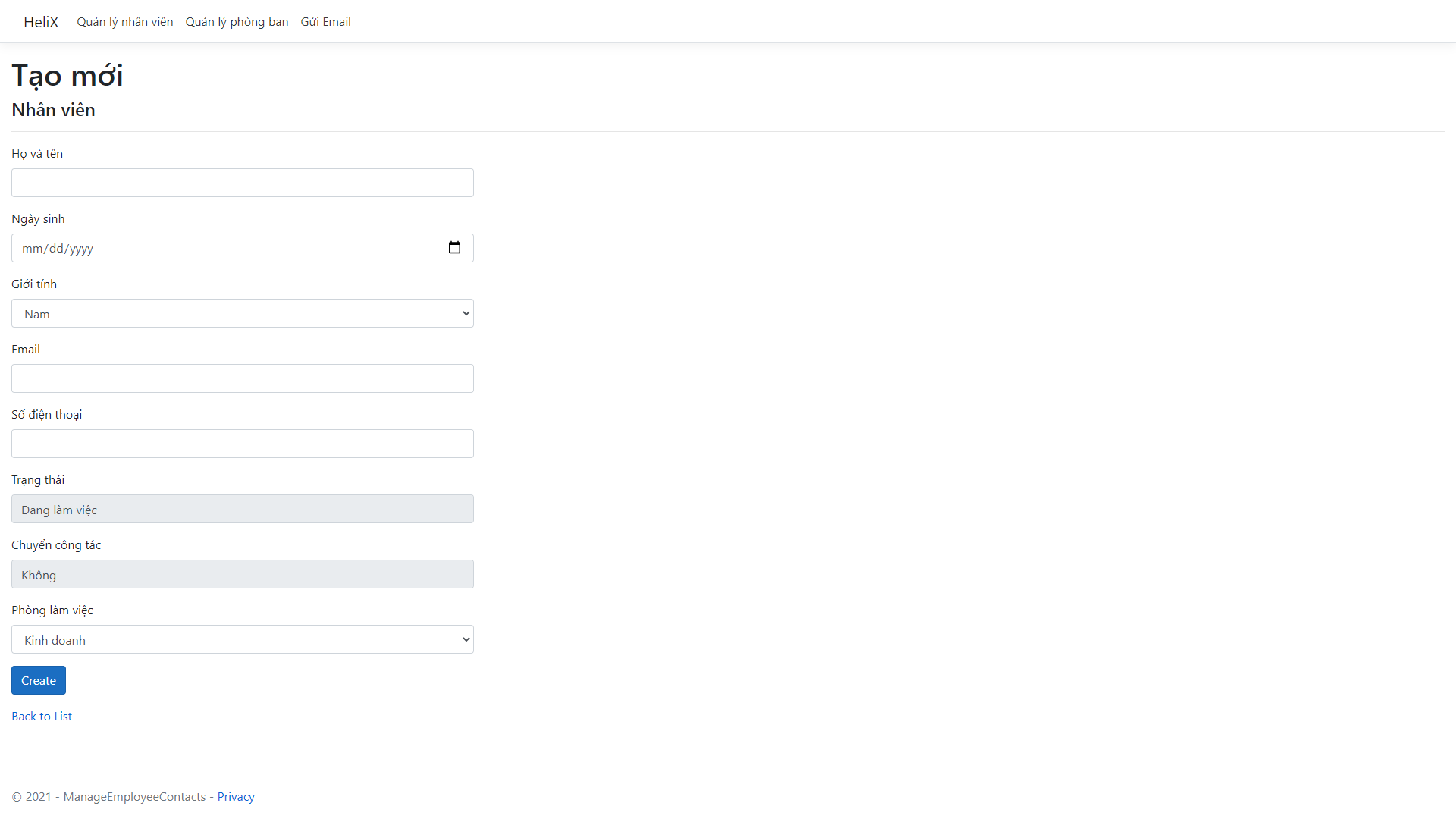
4. Gửi email cho một hoặc nhiều nhân viên cùng lúc (dùng cấu hình gmail để gửi online)

**II. Giao diện của ứng dụng**

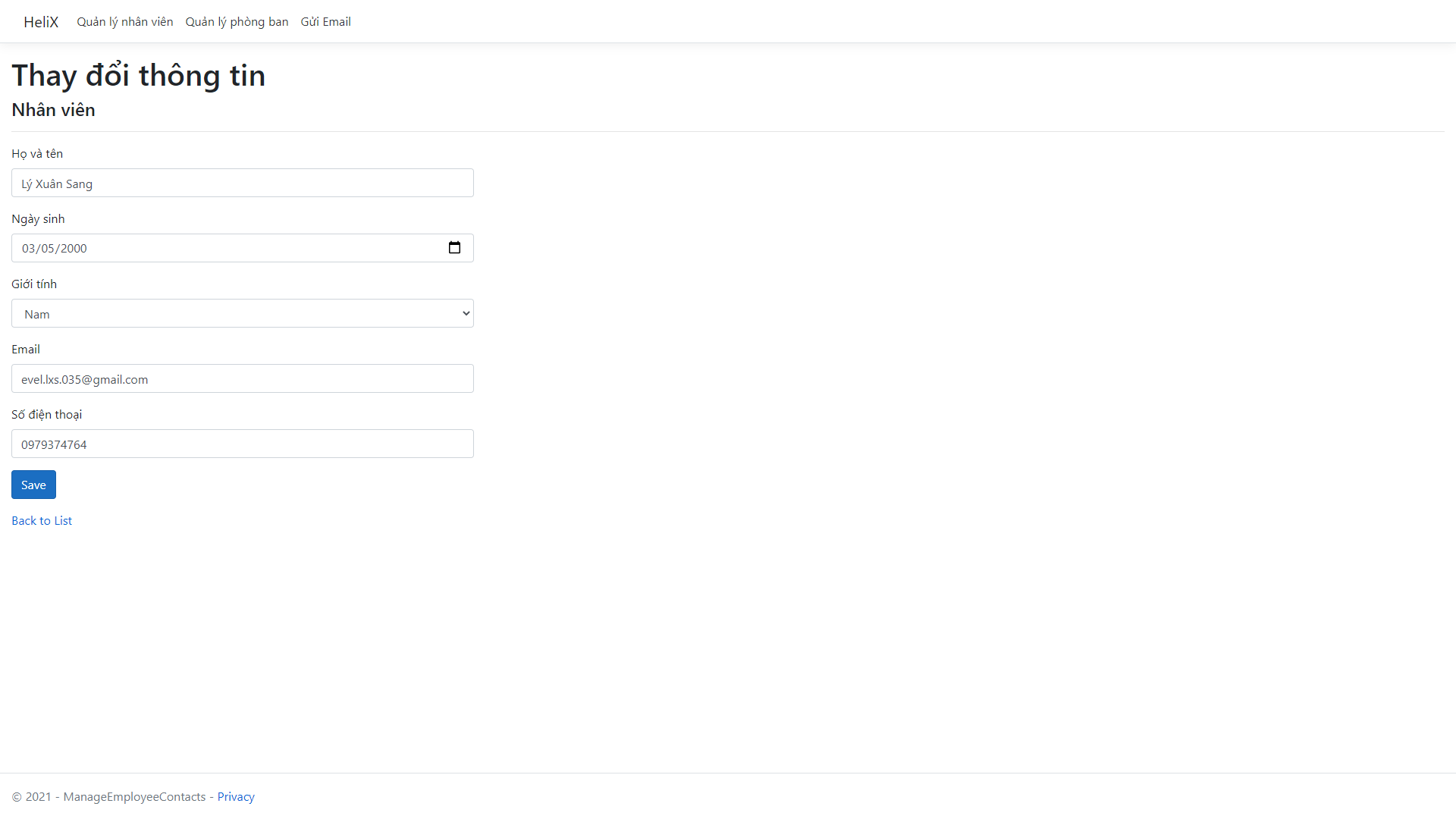
*Hiển thị danh sách nhân viên*



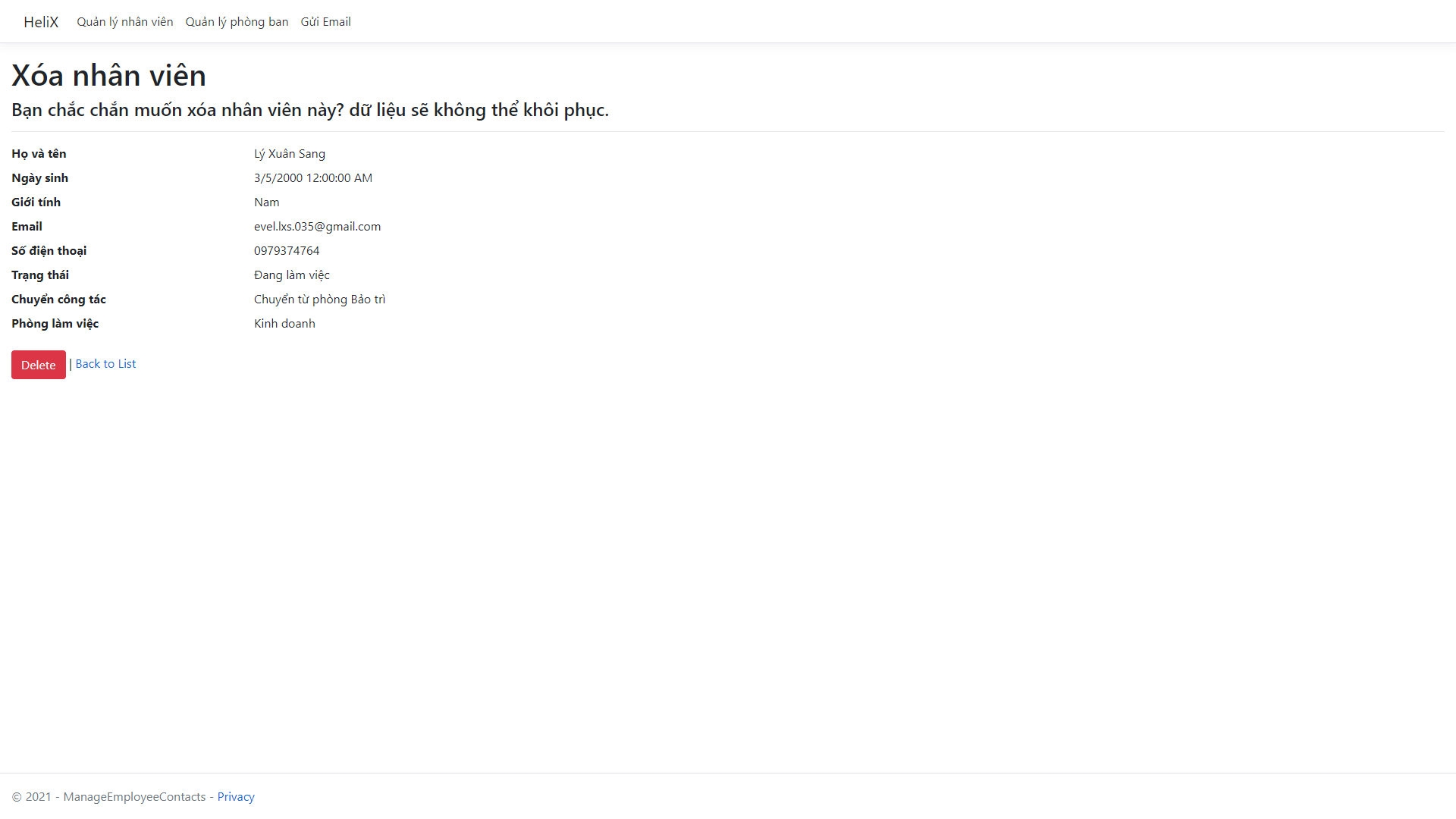
*Thêm mới nhân viên*



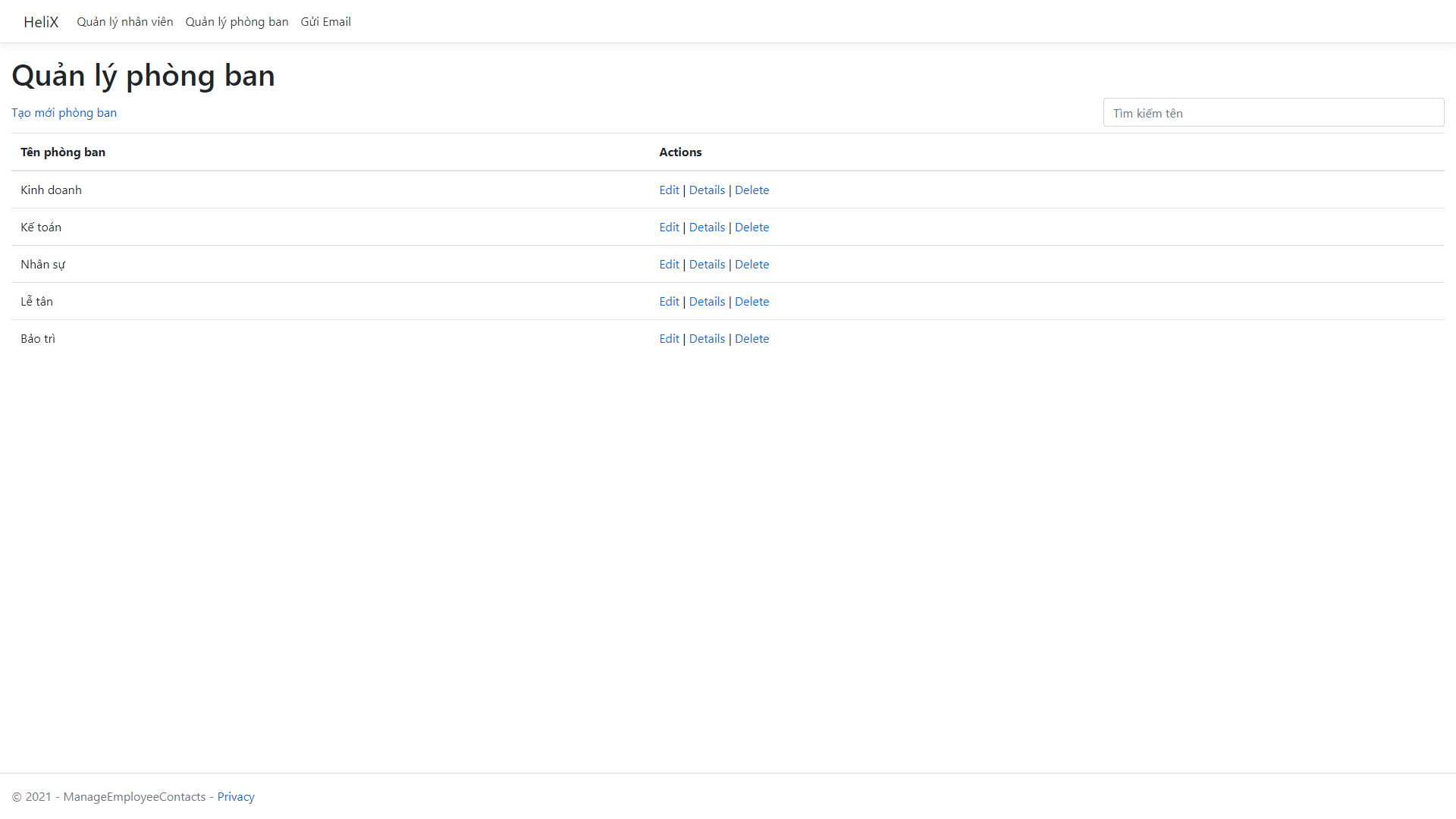
*Sửa thông tin nhân viên*



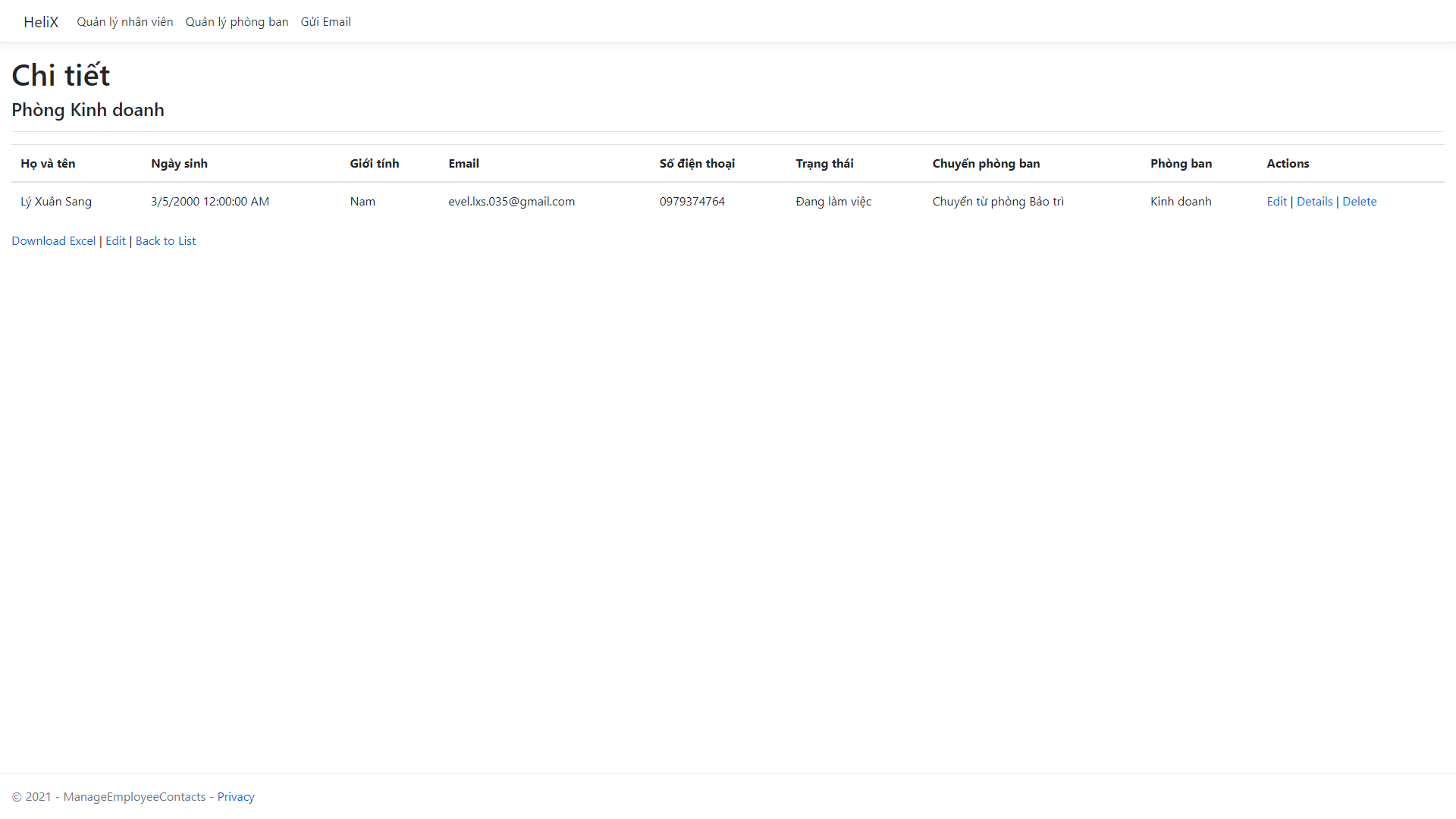
*Xóa nhân viên*

**

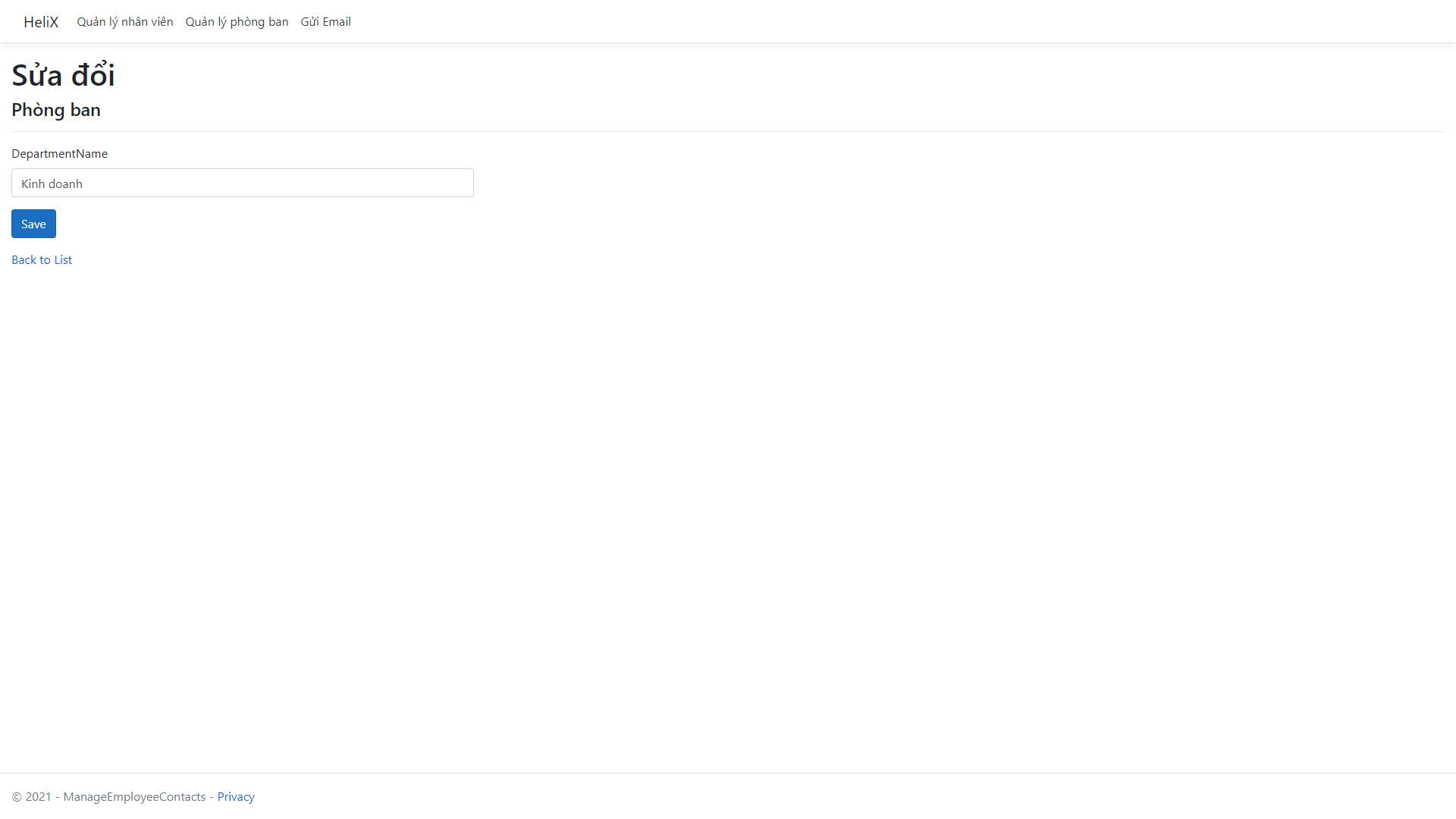
*Hiển thị danh sach phòng ban*

**

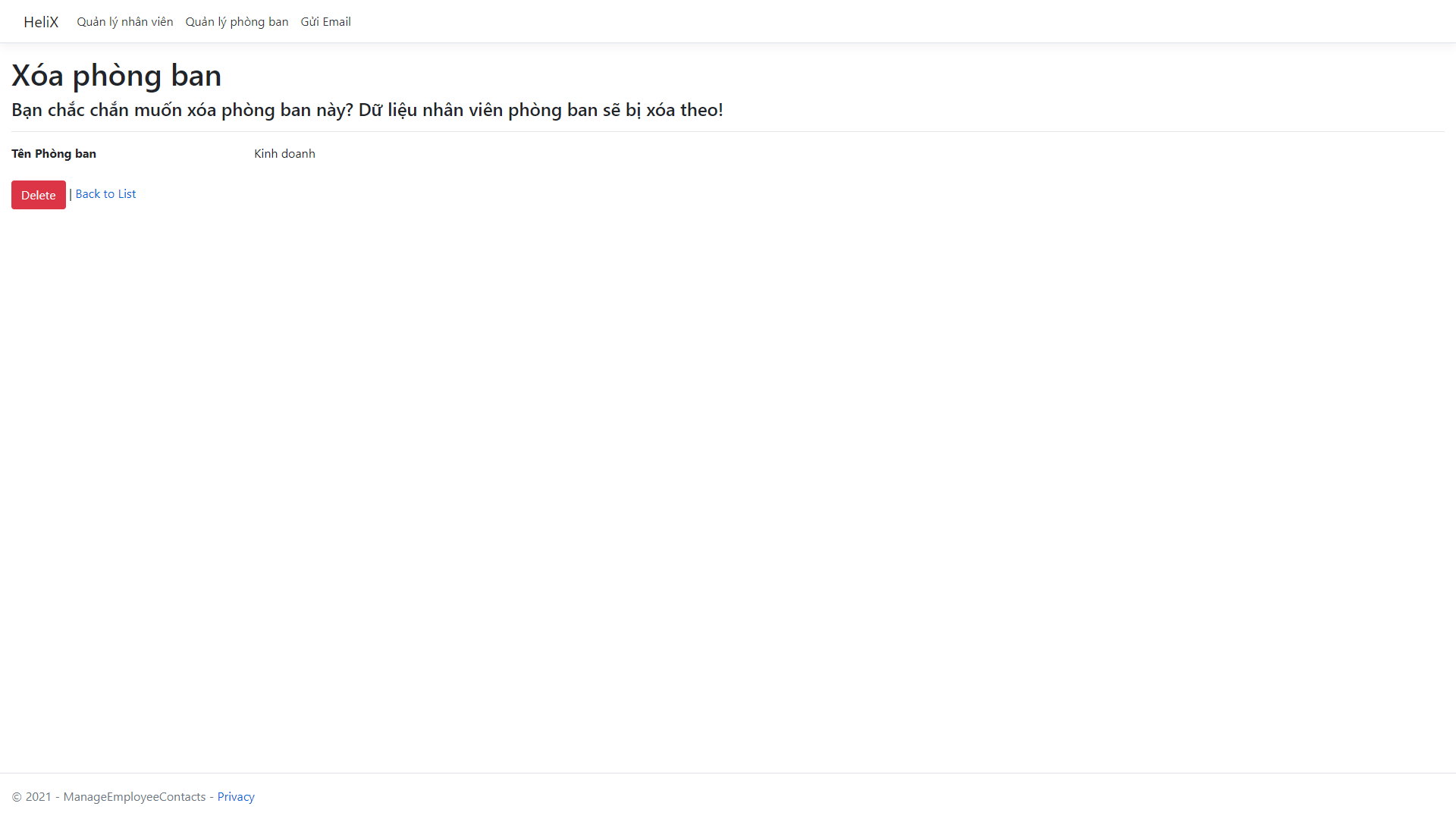
*Hiển thị thông tin phòng ban*

**

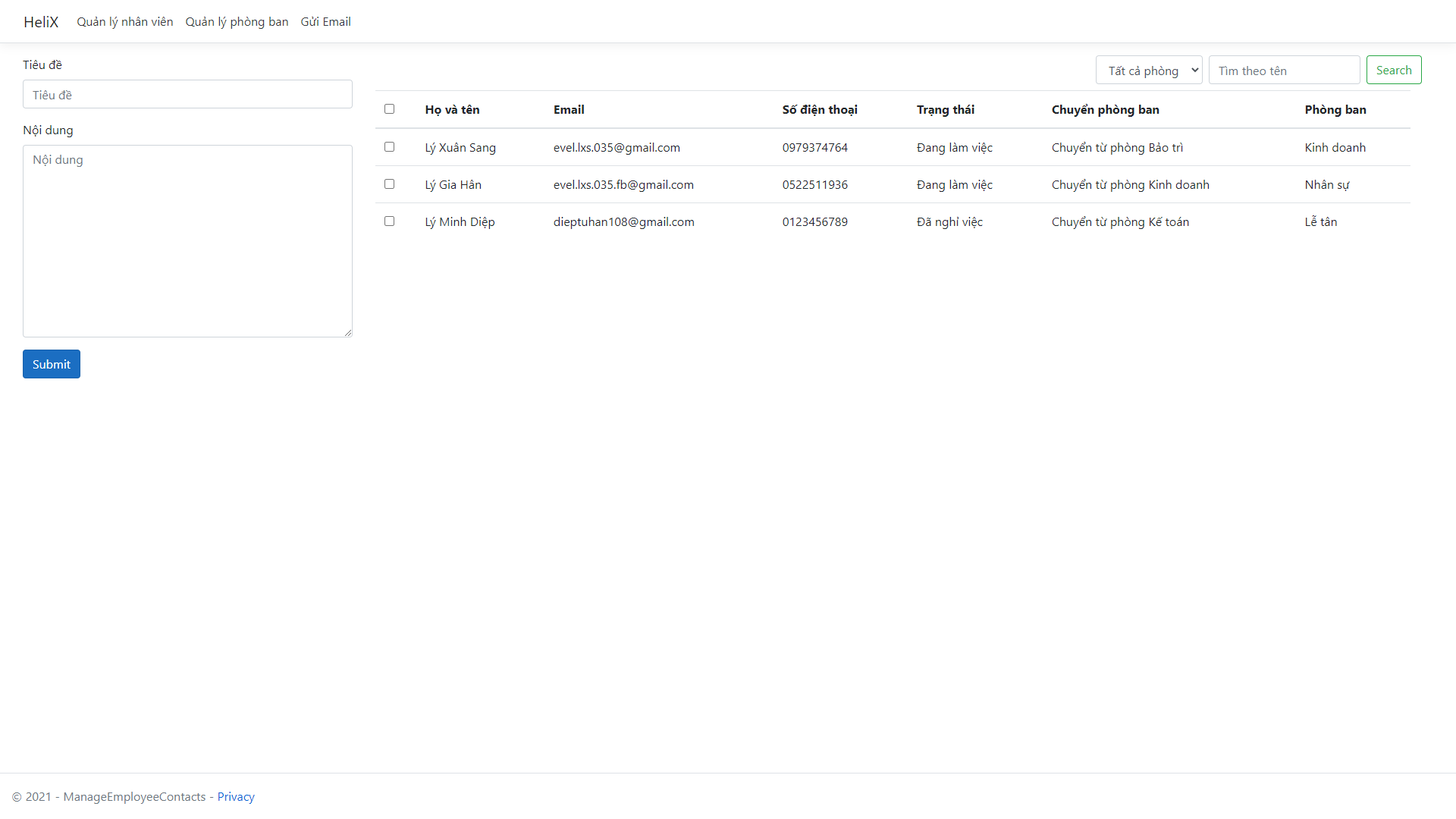
*Sửa đổi phòng ban*

**

*Xóa phòng ban*

**

*Màn hình gửi mail*

**

**III. Source code ứng dụng**

Github:<https://github.com/evel2903/MHPL>