**ĐẠI HỌC PHENIKAA**

TRƯỜNG CÔNG NGHỆ THÔNG TIN

==============



**BÀI TẬP LỚN KẾT THÚC HỌC PHẦN LẬP TRÌNH HƯỚNG ĐỐI TƯỢNG**

**ĐỀ TÀI: QUẢN LÍ NHÂN SỰ**

**Lớp: OOP-N02**

**Giáo viên giảng dạy: Hà Thị Kim Dung**

**TRƯỜNG ĐẠI HỌC PHENIKAA**

TRƯỜNG CÔNG NGHỆ THÔNG TIN

**BÁO CÁO BÀI TẬP LỚN**

**MÔN HỌC: LẬP TRÌNH HƯỚNG ĐỐI TƯỢNG**

**ĐỀ TÀI: QUẢN LÍ NHÂN SỰ**

**Giáo viên giảng dạy: Hà Thị Kim Dung**

**Nhóm thực hiện: Nhóm 3**

**Lớp: N02**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Họ và tên** | **MSV** | **Đóng góp** | **Điểm bằng số** | **Điểm bằng chữ** | **Ký tên** |
| Trần Tuấn Anh | 23017118 |  |  |  |  |
| Ngô Quang Trung | 23017195 |  |  |  |  |

**Link github:** [trantuananh1803/OOP-NO6-NHOM3-PhanMemQuanLyNhanSu](https://github.com/trantuananh1803/OOP-NO6-NHOM3-PhanMemQuanLyNhanSu)

# MỤC LỤC

1. **GIỚI THIỆU BÀI TOÁN**
2. **PHÂN TÍCH BÀI TOÁN**
   * 1. Mô tả hệ thống...............................................................................................4
     2. Phân tích chức năng.......................................................................................6

2.1. Quản lý thông tin nhân viên……………………………………………6

2.2. Tìm kiếm nhân viên……………………………………………………6

2.3. Thống kê và báo cáo…………………………………………………...7

* + 1. Phân tích cơ sở dữ liệu

3.1. Lý do lựa chọn XML…………………………………………………..8

3.2. Cấu trúc dữ liệu………………………………………………………...8

* + 1. Cài đặt và sử dụng………………………………………………………….9

4.1. Hướng dẫn cài đặt và khởi động……………………….………………9

4.2. Hướng dẫn sử dụng các chức năng…………………………………….9

1. **TỔNG KẾT**

1. Nhận xét và đánh giá……………………………………………………..12

2. Kết luận và Hướng phát triển…………………………………………….12

3. Tài liệu tham khảo………………………………………………………..13

1. **Giới thiệu về bài toán**

* Trong môi trường doanh nghiệp hiện đại, việc quản lý thông tin nhân sự một cách hiệu quả và chính xác là yếu tố then chốt quyết định sự thành công của tổ chức. Tuy nhiên, việc quản lý thủ công bằng giấy tờ hoặc các công cụ không chuyên dụng thường tốn nhiều thời gian, dễ gây sai sót và khó khăn trong việc tra cứu, thống kê.
* Để giải quyết vấn đề này, đề tài "Phần mềm Quản lý Nhân sự" được thực hiện nhằm xây dựng một ứng dụng desktop giúp tự động hóa các nghiệp vụ cơ bản trong quản lý nhân sự. Phần mềm được phát triển bằng ngôn ngữ lập trình Java, sử dụng thư viện Swing để xây dựng giao diện đồ họa thân thiện và áp dụng các nguyên tắc của lập trình hướng đối tượng để đảm bảo mã nguồn có cấu trúc rõ ràng, dễ bảo trì và mở rộng. Dữ liệu của hệ thống được lưu trữ dưới dạng tệp XML, giúp ứng dụng trở nên gọn nhẹ và linh hoạt.

1. **Phân tích bài toán**
2. **Mô tả hệ thống**

* Hệ thống được thiết kế và xây dựng dựa trên mô hình kiến trúc **Model-View-Controller (MVC)**, một biến thể của kiến trúc 3 lớp. Việc lựa chọn mô hình này nhằm mục đích tách biệt rõ ràng giữa logic xử lý dữ liệu (Model), giao diện người dùng (View), và luồng điều khiển ứng dụng (Controller). Cách tiếp cận này mang lại nhiều lợi ích như mã nguồn trở nên sạch sẽ, dễ bảo trì, dễ dàng thay đổi một thành phần mà không ảnh hưởng lớn đến các thành phần khác.
* Các thành phần chính trong hệ thống bao gồm:
* **Model (Mô hình dữ liệu và Logic nghiệp vụ):**
  + **Entity (NhanVien.java):** Là một lớp đối tượng Java đơn giản đại diện cho cấu trúc dữ liệu của một nhân viên, bao gồm các thuộc tính như mã, họ tên, địa chỉ, lương...
  + **Service (NhanVienService.java):** Chứa các logic nghiệp vụ chính của ứng dụng như tính toán thống kê, kiểm tra dữ liệu đầu vào. Lớp này đóng vai trò trung gian giữa Controller và lớp truy cập dữ liệu.
  + **Data Access (XMLDOMParser.java):** Là lớp tầng thấp nhất, chịu trách nhiệm duy nhất cho việc giao tiếp với nguồn dữ liệu, cụ thể là đọc và ghi file nhanvien.xml. Nó che giấu sự phức tạp của việc phân tích XML khỏi các tầng cao hơn.
* **View (Giao diện người dùng):**
  + Bao gồm các lớp kế thừa từ JFrame như NhanVienView.java, ThongKeView.java.
  + Nhiệm vụ của View là hiển thị dữ liệu cho người dùng và thu thập các tương tác (nhấn nút, nhập liệu). View không chứa bất kỳ logic nghiệp vụ nào; nó chỉ đơn thuần gửi yêu cầu đến Controller khi có sự kiện xảy ra.
* **Controller (Bộ điều khiển):**
  + Bao gồm các lớp như NhanVienController.java, ThongKeController.java.
  + Controller đóng vai trò là "bộ não" của ứng dụng. Nó lắng nghe sự kiện từ View, quyết định hành động cần thực hiện, gọi các phương thức tương ứng từ Service để xử lý, và cuối cùng nhận kết quả trả về để cập nhật lại cho View.
* ***Sơ đồ hệ thống:***

Sơ đồ luồng hoạt động của hệ thống

**A diagram of a diagram

AI-generated content may be incorrect.**

Giải thích sơ đồ:

* Người dùng nhấn vào nút "Chức Vụ" trên giao diện ThongKeView.
* ThongKeView bắt được sự kiện và thông báo cho ThongKeController.
* ThongKeController gọi phương thức tương ứng trong NhanVienService để yêu cầu dữ liệu thống kê.
* NhanVienService gọi đến XMLDOMParser để đọc toàn bộ danh sách nhân viên từ file nhanvien.xml.
* NhanVienService nhận danh sách, thực hiện logic gộp và đếm, sau đó trả về kết quả đã xử lý cho ThongKeController.
* ThongKeController nhận kết quả, tạo TableModel và ChartPanel, sau đó ra lệnh cho ThongKeView cập nhật lại giao diện Bảng và Biểu đồ.

1. **Phân tích chức năng**

Phân tích chức năng là quá trình xác định và mô tả chi tiết các nhiệm vụ mà hệ thống phải thực hiện. Dựa trên yêu cầu của bài toán, phần mềm Quản lý Nhân sự được chia thành ba nhóm chức năng chính.

* 1. **.Quản lý thông tin nhân viên (CRUD)** Đây là nhóm chức năng cốt lõi, cho phép người dùng thực hiện các thao tác cơ bản trên dữ liệu nhân viên (Create, Read, Update, Delete).
* **Thêm nhân viên:** Người dùng nhập thông tin vào các trường trên giao diện NhanVienView và nhấn nút "Thêm". Hệ thống sẽ kiểm tra tính hợp lệ của dữ liệu và lưu thông tin nhân viên mới vào file XML.
* **Sửa thông tin nhân viên:** Người dùng chọn một nhân viên từ bảng, thông tin sẽ được tải lên các ô nhập liệu. Sau khi chỉnh sửa, người dùng nhấn nút "Sửa" để cập nhật lại thông tin trong file XML.
* **Xóa nhân viên:** Người dùng chọn một nhân viên từ bảng và nhấn nút "Xóa". Hệ thống sẽ yêu cầu xác nhận trước khi xóa vĩnh viễn thông tin nhân viên khỏi file XML.
* **Hiển thị danh sách:** Khi khởi động, hệ thống tự động đọc và hiển thị toàn bộ danh sách nhân viên lên bảng.
  1. **.Tìm kiếm nhân viên:** Chức năng này giúp người dùng nhanh chóng lọc và tìm kiếm thông tin nhân viên từ một danh sách lớn.
* **Tìm kiếm theo tiêu chí:** Người dùng có thể tìm kiếm dựa trên một trong ba tiêu chí: Tên, Quê quán (Địa chỉ), hoặc Năm sinh. Chức năng tìm kiếm không phân biệt chữ hoa/thường và chỉ cần chứa một phần của chuỗi để tăng tính linh hoạt.
  1. **.Thống kê và báo cáo:** Chức năng này cung cấp một cái nhìn tổng quan về dữ liệu nhân sự thông qua các bảng biểu và biểu đồ trực quan.
* **Thống kê theo Chức vụ:** Hệ thống sẽ đếm số lượng nhân viên trong mỗi phòng ban/chức vụ và hiển thị kết quả dưới dạng bảng và biểu đồ cột.
* **Thống kê theo Tuổi:** Hệ thống sẽ phân loại nhân viên vào các nhóm tuổi (ví dụ: Dưới 25, Từ 25-35, Trên 35) và hiển thị số lượng tương ứng dưới dạng bảng và biểu đồ cột.

***Biểu đồ phân rã chức năng:***

* Biểu đồ phân rã chức năng (Function Decomposition Diagram) là một công cụ để trực quan hóa việc chia nhỏ các chức năng lớn của hệ thống thành các chức năng con, dễ quản lý và thực hiện hơn.

Biểu đồ phân rã chức năng của phần mềm

A screenshot of a computer

AI-generated content may be incorrect.

1. **Phân tích cơ sở dữ liệu**

Đối với một ứng dụng desktop có quy mô nhỏ và yêu cầu tính linh hoạt, việc sử dụng một hệ quản trị cơ sở dữ liệu (CSDL) đầy đủ như SQL Server hay MySQL có thể không cần thiết và làm tăng sự phức tạp trong cài đặt và triển khai. Do đó, giải pháp lưu trữ dữ liệu bằng tệp XML (eXtensible Markup Language) đã được lựa chọn.

* 1. **.Lý do lựa chọn XML**
* Đơn giản và gọn nhẹ: Hệ thống không yêu cầu cài đặt thêm bất kỳ phần mềm CSDL nào. Toàn bộ dữ liệu nằm trong một file nhanvien.xml, giúp việc sao chép, di chuyển và triển khai ứng dụng trở nên dễ dàng.
* Cấu trúc rõ ràng: XML lưu trữ dữ liệu dưới dạng cây phân cấp với các thẻ (tags) tự mô tả, giúp con người có thể đọc và hiểu được cấu trúc dữ liệu một cách trực quan.
* Hỗ trợ mạnh mẽ trong Java: Java cung cấp các thư viện mạnh mẽ như JAXB và DOM để làm việc với XML, giúp việc đọc, ghi và truy vấn dữ liệu trở nên thuận tiện.
  1. **.Cấu trúc dữ liệu**
* Dữ liệu của toàn bộ nhân viên được lưu trong file data/nhanvien.xml. File này có một thẻ gốc là <nhanviens> chứa một danh sách các thẻ <nhanvien>. Mỗi thẻ <nhanvien> đại diện cho một đối tượng nhân viên và có cấu trúc như sau:

|  |  |  |  |
| --- | --- | --- | --- |
| **Tên trường** | **Tên thẻ XML** | **Kiểu dữ liệu** | **Mô tả** |
| Mã nhân viên | ma | Integer | Mã số định danh duy nhất, tự động tăng. |
| Họ và tên | hoTen | String | Họ và tên đầy đủ của nhân viên. |
| Địa chỉ | diaChi | String | Địa chỉ hoặc quê quán. |
| Giới tính | gioiTinh | String | Giới tính ("Nam" hoặc "Nữ"). |
| Lương | Luong | Double | Mức lương. |
| Ngày sinh | ngaySinh | Date | Ngày sinh (định dạng dd/MM/yyyy). |
| Phòng ban | phongBan | String | Phòng ban hoặc chức vụ. |
| Số điện thoại | phoneNumber | String | Số điện thoại liên hệ. |

1. **Cài đặt và sử dụng**

**Yêu cầu hệ thống**

* Hệ điều hành: Windows, macOS, hoặc Linux.
* Phần mềm: Cần cài đặt sẵn Môi trường thực thi Java (Java Runtime Environment - JRE) phiên bản 11 trở lên.
  1. **.Hướng dẫn cài đặt và khởi động**
* Ứng dụng được đóng gói dưới dạng một file .jar duy nhất, không yêu cầu cài đặt phức tạp.
* Tải về: Tải file ***PhanMemQuanLyNhanSu-1.0-SNAPSHOT.jar*** và thư mục data (chứa file nhanvien.xml) về máy.
* Tổ chức thư mục: Đặt file .jar và thư mục data trong cùng một thư mục để ứng dụng có thể tìm thấy file dữ liệu.
* Khởi động: Nháy đúp chuột vào file .jar để chạy ứng dụng. Nếu không thành công, mở Command Prompt hoặc Terminal, di chuyển đến thư mục chứa file và chạy lệnh: ***java -jar PhanMemQuanLyNhanSu-1.0-SNAPSHOT.jar.***
  1. **.Hướng dẫn sử dụng các chức năng**
* **Giao diện chính:** Khi khởi động, giao diện chính (NhanVienView) sẽ hiển thị danh sách toàn bộ nhân viên có trong hệ thống. Giao diện bao gồm khu vực nhập liệu ở trên, các nút chức năng ở bên cạnh và bảng dữ liệu ở dưới.

A screenshot of a computer

AI-generated content may be incorrect.

* **Thêm nhân viên mới**
  + Nhập đầy đủ thông tin của nhân viên vào các ô tương ứng: Họ và Tên, Địa chỉ, Ngày sinh, Lương, Phòng ban, Số điện thoại.
  + Chọn giới tính (Nam/Nữ).
  + Nhấn nút **"Thêm"**. Nếu thông tin hợp lệ, một thông báo thành công sẽ hiện ra và nhân viên mới sẽ xuất hiện ở cuối bảng.
* **Sửa thông tin nhân viên**
  + Dùng chuột nhấn vào một dòng bất kỳ trong bảng để chọn nhân viên cần sửa.
  + Toàn bộ thông tin của nhân viên đó sẽ tự động được điền lên các ô nhập liệu ở trên.
  + Thay đổi thông tin cần thiết trong các ô.
  + Nhấn nút **"Sửa"**. Thông tin của nhân viên trong bảng sẽ được cập nhật.
* **Xóa nhân viên**
  + Chọn nhân viên cần xóa từ bảng.
  + Nhấn nút **"Xóa"**.
  + Một hộp thoại xác nhận sẽ hiện ra. Chọn **"Yes"** để xác nhận xóa. Nhân viên sẽ biến mất khỏi bảng.
* **Tìm kiếm nhân viên**
  + Nhấn nút **"Tìm kiếm"** trên giao diện chính.
  + Một cửa sổ tìm kiếm sẽ xuất hiện.
  + Nhập từ khóa cần tìm vào ô "Nhập Nội Dung Tìm Kiếm".
  + Chọn một trong ba tiêu chí tìm kiếm: **Tên**, **Quê Quán**, hoặc **Năm Sinh**.
  + Nhấn nút **"Tìm kiếm"** trong cửa sổ này. Cửa sổ sẽ đóng lại và bảng trên giao diện chính sẽ chỉ hiển thị các kết quả phù hợp.
  + Để hiển thị lại toàn bộ danh sách, nhấn nút **"Hủy Tìm Kiếm"**.

A screenshot of a computer

AI-generated content may be incorrect.

* **Thống kê**
  + Từ giao diện chính, nhấn nút **"Thống kê"**.
  + Cửa sổ thống kê (ThongKeView) sẽ mở ra.
  + Mặc định, hệ thống sẽ hiển thị thống kê theo Chức vụ.
  + Nhấn vào nút **"Tuổi"** để xem thống kê theo các nhóm tuổi.
  + Nhấn vào nút **"Chức Vụ"** để quay lại thống kê theo chức vụ.
  + Cả bảng dữ liệu và biểu đồ cột sẽ được cập nhật tương ứng với lựa chọn của bạn.

A screenshot of a computer

AI-generated content may be incorrect.A screenshot of a computer

AI-generated content may be incorrect.

1. **Tổng kết**

**1.Nhận xét và Đánh giá**

* **Kết quả đạt được:** Dự án đã hoàn thành việc xây dựng một phần mềm quản lý nhân sự desktop bằng Java Swing, đáp ứng đầy đủ các yêu cầu cơ bản của bài toán đặt ra. Cụ thể:
* Xây dựng thành công các chức năng quản lý thông tin nhân viên (thêm, sửa, xóa).
* Triển khai chức năng tìm kiếm linh hoạt theo nhiều tiêu chí.
* Tích hợp chức năng thống kê dữ liệu trực quan bằng bảng và biểu đồ cột.
* Áp dụng thành công mô hình MVC, giúp cấu trúc mã nguồn rõ ràng, tách biệt giữa các thành phần giao diện, logic và dữ liệu.
* Sử dụng tệp XML làm cơ sở dữ liệu, đảm bảo tính gọn nhẹ và dễ triển khai của ứng dụng.
* **Ưu điểm:**
* **Giao diện thân thiện:** Giao diện được thiết kế đơn giản, trực quan, giúp người dùng dễ dàng làm quen và sử dụng.
* **Đáp ứng đủ chức năng:** Phần mềm thực hiện tốt các chức năng quản lý cơ bản cần có của một hệ thống quản lý nhân sự quy mô nhỏ.
* **Cấu trúc tốt:** Việc áp dụng mô hình MVC giúp mã nguồn dễ đọc, dễ bảo trì và có khả năng mở rộng trong tương lai.
* **Tính linh hoạt:** Việc sử dụng file XML giúp ứng dụng không phụ thuộc vào bất kỳ hệ quản trị CSDL nào, dễ dàng sao chép và chạy trên nhiều máy tính khác nhau.
* **Hạn chế:**
* **Hiệu năng với dữ liệu lớn:** Việc lưu trữ bằng file XML sẽ gặp hạn chế về hiệu năng (tốc độ đọc/ghi) khi số lượng nhân viên tăng lên đáng kể.
* **Tính năng hạn chế:** Phần mềm còn thiếu các tính năng nâng cao như phân quyền người dùng, quản lý chấm công, tính lương, xuất báo cáo ra file PDF/Excel.
* **Giao diện:** Giao diện Swing tuy ổn định nhưng có thể chưa thực sự hiện đại so với các framework giao diện mới.

**2. Kết luận và Hướng phát triển**

**- Kết luận:** Qua quá trình thực hiện bài tập lớn, nhóm đã áp dụng thành công các kiến thức về lập trình hướng đối tượng, thiết kế giao diện Swing và xử lý dữ liệu để xây dựng một ứng dụng hoàn chỉnh. Dự án "Phần mềm Quản lý Nhân sự" đã đạt được mục tiêu đề ra, là một công cụ hữu ích cho việc quản lý nhân sự ở quy mô nhỏ và vừa. Đồng thời, đây cũng là một cơ hội quý báu để nhóm củng cố kiến thức và rèn luyện kỹ năng làm việc nhóm, phân tích và giải quyết vấn đề.

**- Hướng phát triển trong tương lai:** Để phần mềm trở nên hoàn thiện và có tính ứng dụng cao hơn, nhóm đề xuất một số hướng phát triển sau:

* **Nâng cấp cơ sở dữ liệu:** Chuyển đổi từ việc sử dụng file XML sang một hệ quản trị cơ sở dữ liệu thực thụ như MySQL hoặc PostgreSQL để cải thiện hiệu năng và khả năng quản lý dữ liệu lớn.
* **Phát triển phiên bản Web:** Xây dựng một phiên bản ứng dụng web dựa trên Java (ví dụ: Spring Boot) để người dùng có thể truy cập và quản lý từ bất kỳ đâu.
* **Bổ sung tính năng nâng cao:**
  + Xây dựng hệ thống đăng nhập, phân quyền người dùng (quản trị viên, nhân viên).
  + Thêm module quản lý chấm công, tính lương tự động.
  + Tích hợp chức năng xuất báo cáo, thống kê ra các định dạng phổ biến như PDF, Excel.
* **Cải thiện giao diện người dùng (UI/UX):** Sử dụng các thư viện hoặc framework giao diện hiện đại hơn (như JavaFX) để mang lại trải nghiệm tốt hơn cho người dùng.

<div style="page-break-after: always;"></div>

**3. Tài liệu tham khảo**

1. Oracle. *The Java™ Tutorials - Creating a GUI With Swing*: <https://docs.oracle.com/javase/tutorial/uiswing/>
2. JFree. *JFreeChart Developer Guide*.

<https://www.jfree.org/jfreechart/>

1. Baeldung. *Introduction to Java Streams*.

<https://www.baeldung.com/java-8-streams-introduction>

1. Jenkov, Jakob. *Java XML - DOM*:

<http://tutorials.jenkov.com/java-xml/dom.html>

1. File QuanLiDoiTuong:

[BellaHa/DTDBv1](https://github.com/BellaHa/DTDBv1)