**LỜI CẢM ƠN**

*Lời đầu tiên, tôi xin chân thành cảm ơn quý thầy cô trong khoa Tin học cũng như các quý thầy cô đang giảng dạy và công tác tại trường Đại học Sư Phạm - Đại học Đà Nẵng đã truyền đạt những kiến thức quý báu cho tôi trong những năm học vừa qua. Đặc biệt, tôi xin chân thành cảm ơn TS. Nguyễn Hoàng Hải, trường Đại học Sư phạm – Đại học Đà Nẵng đã tận tình hướng dẫn, giúp đỡ tôi trong suốt thời gian thực hiện đề tài.*

*Mặc dù đã cố gắng để hoàn thành dự án trong phạm vi cho phép và khả năng, nhưng không tránh khỏi sai sót và thiếu sót một số chức năng của hệ thống. Vì vậy, tôi mong muốn nhận được sự nhắc nhở và đóng góp ý kiến quý giá từ các thầy cô để hệ thống cũng như đề tài của tôi được hoàn thiện hơn.*

*Một lần nữa xin chân thành cám ơn!*

*Đà Nẵng, ngày 20 tháng 04 năm 2018*

*Sinh viên thực hiện*

**Nguyễn Quang Dũng**

**LỜI CAM ĐOAN**

*Tôi xin cam đoan những nội dung trong luận văn này là do tôi thực hiện dưới sự hướng dẫn trực tiếp của TS. Nguyễn Hoàng Hải.*

*Mọi tham khảo dùng trong luận văn đều được trích dẫn rõ ràng tên tác giả, tên công trình, thời gian, địa điểm công bố.*

*Mọi sao chép không hợp lệ, vi phạm quy chế đào tạo, hay gian trá, tôi xin chịu hoàn toàn trách nhiệm.*

*Sinh viên thực hiện*

**Nguyễn Quang Dũng**

MỤC LỤC

**MỞ ĐẦU**

* 1. **Lý do chọn đề tài**

Hiện tại câu lạc bộ tin học trường Đại học Sư phạm Đà Nẵng vẫn chưa có

trang quản lý hoạt động cụ thể, thông tin các hoạt động của câu lạc bộ được đăng trên nhóm mạng xã hội vẫn chưa được rõ ràng cũng như thông tin của các thành viên trong câu lạc bộ. Để dễ dàng quản lý các hoạt động, quản lý thành viên tham gia hoạt động và đưa thông tin hoạt động đến các thành viên giúp thành viên câu lạc bộ đăng ký tham gia hoạt động một cách linh hoạt nhất, tôi đã chọn đề tài “Ngiên cứu xây dựng website quản lý hoạt động câu lạc bộ tin học” trên nền tảng Java Spring Framework.

* 1. **Mục tiêu nghiên cứu**
* Tạo chức năng quản lý thông tin của các thành viên trong câu lạc bộ tin học.
* Xác định được các hoạt động chính của câu lạc bộ.
* Tạo chức năng chia sẽ thông tin về các hoạt động của câu lạc bộ.
  1. **Nhiệm vụ nghiên cứu**
* Đưa thông tin các hoạt động của câu lạc bộ tin học lên website, giúp cho các thành viên câu lạc bộ tiếp cận tới các hoạt động một cách dễ dàng và linh hoạt. Đảm bảo các hoạt động của câu lạc bộ đề ra được nhiều người biết và số lượng tham gia cao.
* Phát triển chức năng thông tin thành viên giúp cho việc quản lý thành viên dễ dàng hơn.
  1. **Đối tượng nghiên cứu**
* Website quản lý hoạt động câu lạc bộ tin học, cụ thể là câu lạc bộ tin học trường Đại học Sư phạm Đà Nẵng.
* Ngôn ngữ Java
* Java Spring Framework
* Các hoạt động của câu lạc bộ
  1. **Phạm vi nghiên cứu**
* Nền tảng Java.
* Cộng cụ Eclipse.
* Câu lạc bô tin học trường Đại học Sư phạm Đà Nẵng.
  1. **Phương pháp nghiên cứu**
* Tìm hiểu nền tảng Java và Spring Framework để ứng dụng vào đề tài.
* Tham khảo cách tổ chức hoạt động của câu lạc bộ tin học.
* Khảo sát nhu cầu của thành viên trong câu lạc bộ.
  1. **Bố cục đề tài**
* **Mở đầu:**

Giới thiệu lý do chọn đề tài, khách thể và đối tượng nghiên cứu trong đề tài, giới hạn phạm vi nghiên cứu của đề tài, nhiệm vụ và phương pháp nghiên cứu của đề tài.

* **Chương 1: Cơ sở lý thuyết.**

Trình bày những lý thuyết liên quan đến đề tài. Những kiến thức lý thuyết này đóng vai trò quan trọng, là cơ sở chính trong việc hình thành và phát triển hệ thống.

* **Chương 2: Phân tích và thiết kế hệ thống.**

Trình bày phân tích thiết kế hệ thống và phân tích cơ sở dữ liệu.

* **Chương 3: Thử nghiệm và đánh giá.**

Trình bày các kết quả đạt được sau thời gian xây dựng website. Từ kết quả đạt được đánh giá hệ thống.

* **Kết luận và hướng phát triển.**

Tổng kết kết quả đạt được.

Nếu hướng phát triển.

CHƯƠNG 1: CƠ SỞ LÝ THUYẾT

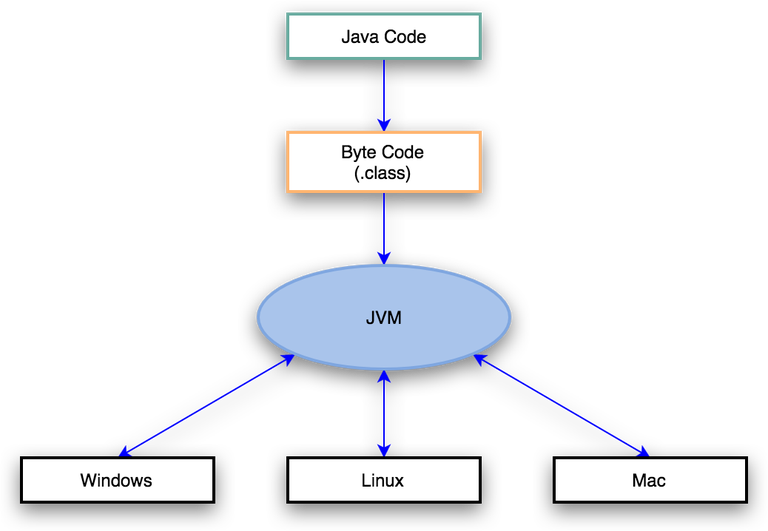
* 1. Ngôn ngữ JAVA
     1. Tổng quan về ngôn ngữ JAVA

Java là một ngôn ngữ lập trình được phát triển bởi Sun Microsystems năm 1995. Ngày nay, Java thuộc quyền sở hữu bởi Oracle và là một trong những ngôn ngữ lập trình được sử dụng rộng rãi nhất thế giới.

|  |  |  |  |
| --- | --- | --- | --- |
| Năm | Tháng | | Sự kiện |
| 1996 | Tháng một | Phát hành JDK 1.0 | |
| 1997 | Tháng hai | Phát hành JDK 1.1 | |
| 1998 | Tháng mười hai | Phát hành SDK 1.2 | |
| 1999 | Tháng tám | Phát hành Java 2 Platform, Standard Edition (J2SE) | |
| Tháng mười hai | Phát hành Java 2 Platform, Enterprise Edition (J2EE) | |
| 2000 | Tháng năm | J2SE và SDK 1.3 | |
| 2002 | Tháng hai | J2SE và SDK 1.4 | |
| 2004 | Tháng chín | J2SE 5.0 và JDK 1.5 | |
| 2006 | Tháng mười hai | Java SE 6 và JDK 1.6 | |
| 2010 | Tháng tư | Oracle mua lại Sun | |
| 2011 | Tháng bảy | Java SE 7 và JDK 1.7 | |
| 2014 | Tháng ba | Java SE 8 và JDK 1.8 | |
| 2017 | Tháng chín | Java SE 9 và JDK 1.9 | |

Bảng 1: Các phiên bản Java đã phát hành

Java là ngôn ngữ lập trình hướng đối tượng và có cú pháp tượng tự C++. Java được đảm bảo « Write Once, Run Anywhere », nghĩa là các chương trình chỉ cần viết một lần và chạy được trên mọi nền tảng khác nhau. Để làm được điều này Java đưa ra khái niệm máy ảo JVM (Java Virtual Machine). Khi biên dịch, mã nguồn sẽ được dịch sang bytecode sau đó máy ảo JVM sẽ dịch tiếp bytecode thành mã máy để máy tính có thể hiểu và chạy được.



Hình 1: Quá trình biên dịch và chạy chương trình

* + 1. Môi trường và công cụ phát triển

Java Development Kit (JDK), đây là bộ công cụ gồm môi trường lập trình, phát triển ứng dụng bằng ngôn ngữ Java bao gồm bộ biên dịch và thực thi chương trình.

Để lập trình Java, bạn có thể sử dụng những IDE hiện đại hỗ trợ riêng cho việc lập trình Java như Netbean, Eclipse,…Đây là những IDE mã nguồn mở, miễn phí, mạnh mẽ.

* 1. Spring Framework

Phần đầu tiên của Spring Framework ban đầu nguyên được [Rod Johnson](https://vi.wikipedia.org/w/index.php?title=Rod_Johnson&action=edit&redlink=1) viết vào năm 2000. Ngày nay, Spring đã trở thành framework mã nguồn mở phổ biến nhất để xây dựng các ứng dụng doanh nghiệp. Theo một số nguồn, trên 50% các ứng dụng web Java hiện nay đang sử dụng Spring.

Spring Framework là một [bộ khung ứng dụng](https://vi.wikipedia.org/w/index.php?title=B%E1%BB%99_khung_%E1%BB%A9ng_d%E1%BB%A5ng&action=edit&redlink=1) và [bộ chứa](https://en.wikipedia.org/wiki/Servlet_container) [đảo ngược điều khiển](https://en.wikipedia.org/wiki/inversion_of_control) cho [nền tảng Java](https://vi.wikipedia.org/wiki/Java_(c%C3%B4ng_ngh%E1%BB%87)). Chức năng tính của bộ khung này có thể áp dụng cho bất kỳ ứng dụng Java nào, tuy vậy, nhiều bản mở rộng dành cho việc xây dựng ứng dụng nền web cũng được phát triển trên nền tảng [Java EE](https://vi.wikipedia.org/w/index.php?title=Java_EE&action=edit&redlink=1). Tuy bộ khung không ấn định lên một cụ thể nào, Spring Framework đã trở nên phổ biến trong cộng đồng [Java](https://vi.wikipedia.org/wiki/Java_(ng%C3%B4n_ng%E1%BB%AF_l%E1%BA%ADp_tr%C3%ACnh)) như một phương án song hành, thay thế, hoặc thậm chí bổ sung cho mô hình [Enterprise JavaBeans](https://vi.wikipedia.org/w/index.php?title=Enterprise_JavaBeans&action=edit&redlink=1) (EJB). Đây là bộ khung [mã nguồn mở](https://en.wikipedia.org/wiki/open_source).

Để ngăn chặn sự phức tạp trong phát triển các ứng dụng, Spring Framework thường dựa trên các quan điểm như sau:

* + - Đơn giản hóa công việc phát triển thông qua việc sử dụng các đối tượng Java đơn giản hay còn được gọi là POJO (Plain Old Java Object)
    - Nới lỏng ràng buộc giữa các thành phần thông qua việc sử dụng Dependency Injection và viết các interface
    - Tiếp cận lập trình khai báo bằng cách sử dụng các quy tắc (convention) và các khía cạnh (aspect) chung

Giảm thiểu các mã nghi thức và soạn sẵn (boilerplate) thông qua việc sử dụng các khuôn mẫu (template) và các khía cạnh.

* 1. Mô hình MVC

MVC (Model – View – Controller) là một mẫu kiến trúc phần mềm trong kỹ thuật kỹ sư phần mềm. Khi sử dụng đúng cách, mẫu MVC giúp cho người phát triển phần mềm cô lập các nguyên tắc nghiệp vụ và giao diện người dùng một cách rõ ràng hơn. Phần mềm phát triển theo mẫu MVC tạo nhiều thuận lợi cho việc bảo trì vì các nguyên tắc nghề nghiệp và giao diện ít liên quan với nhau.

Hình 2: Mô hình MVC

Controller giữ nhiệm vụ nhận điều hướng các yêu cầu từ người dùng và gọi đúng những phương thức xử lý chúng… Chẳng hạn thành phần này sẽ nhận request từ url và form để thao tác trực tiếp với Model.

Model là thành phần chứa tất cả các nghiệp vụ logic, phương thức xử lý, truy xuất database, đối tượng mô tả dữ liệu như các Class, hàm xử lý…

View đảm nhận việc hiển thị thông tin, tương tác với người dùng, nơi chứa tất cả các đối tượng GUI như textbox, images… Hiểu một cách đơn giản, nó là tập hợp các form hoặc các file HTML.

+ Luồng đi: Khi có một yêu cầu từ phía client gửi đến server, Bộ phận controller có nhiệm vụ nhận yêu cầu, xử lý yêu cầu đó. Và nếu cần, nó sẽ gọi đến phần model, vốn là bộ phần làm việc với Database.. Sau khi xử lý xong, toàn bộ kết quả được đẩy về phần View. Tại View, sẽ gen ra mã Html tạo nên giao diện, và trả toàn bộ html về trình duyệt để hiển thị.

* 1. HTML, CSS, JAVASCRIPT, MYSQL
     1. HTML

HTML là chữ viết tắt của cụm từ **H**yper**T**ext **M**arkup **L**anguage được sử dụng để tạo một trang web, trên một website có thể sẽ chứa nhiều trang và mỗi trang được quy ra là một tập tin HTML.  Cha đẻ của HTML là **Tim Berners-Lee**, cũng là người khai sinh ra World Wide Web và chủ tịch của **World Wide Web Consortium.**

Một tài liệu HTML được hình thành bởi các phần tử HTML (HTML Elements) được quy định bằng các cặp thẻ (tag), các cặp thẻ này được bao bọc bởi một dấu ngoặc ngọn (ví dụ <html>) và thường là sẽ được khai báo thành một cặp, bao gồm thẻ mở và thẻ đóng (ví <strong> dụ </strong> và ). Các văn bản muốn được đánh dấu bằng HTML sẽ được khai báo bên trong cặp thẻ (ví dụ <strong>Đây là chữ in đậm</strong>). Nhưng một số thẻ đặc biệt lại không có thẻ đóng và dữ liệu được khai báo sẽ nằm trong các thuộc tính (ví dụ như thẻ <img>).

Khi một tập tin HTML được hình thành, việc xử lý nó sẽ do trình duyệt web đảm nhận. Trình duyệt sẽ đóng vai trò đọc hiểu nội dung HTML từ các thẻ bên trong và sẽ chuyển sang dạng văn bản đã được đánh dấu để đọc, nghe hoặc hiểu (do các bot máy tính hiểu).

HTML sẽ được khai báo bằng các phần tử bởi các từ khóa. Nội dung nằm bên trong cặp từ khóa sẽ là nội dung bạn cần định dạng với HTML. Ví dụ dưới đây là một đoạn HTML khai báo một đoạn văn bản.

|  |  |
| --- | --- |
|  | <p>Đây là một đoạn văn bản trong HTML.</p> |

Ngoài ra, trong các thẻ còn có **các thuộc tính**, thuộc tính sẽ đặt bên trong thẻ mở đầu, mỗi thuộc tính sẽ có giá trị được đặt trong dấu ngoặc kép và cách nhau bởi dấu bằng (=) với tên thuộc tính. Ví dụ dưới đây là một thẻ có sử dụng thuộc tính

|  |  |
| --- | --- |
|  | <form action="google.com"> </form> |

Một thẻ có thể sử dụng nhiều thuộc tính.

HTML là một ngôn ngữ đánh dấu siêu văn bản nên nó sẽ có vai trò xây dựng cấu trúc siêu văn bản trên một website. Điều đó không có nghĩa là chỉ sử dụng HTML để tạo ra một website mà HTML chỉ đóng một vai trò hình thành trên website.

* + 1. CSS

**CSS** là chữ viết tắt của Cascading Style Sheets, nó là một ngôn ngữ được sử dụng để **tìm và định dạng** lại các phần tử được tạo ra bởi các ngôn ngữ đánh dấu (ví dụ như HTML).

Phương thức hoạt động của CSS là nó sẽ tìm dựa vào các vùng chọn, vùng chọn có thể là tên một thẻ HTML, tên một ID, class hay nhiều kiểu khác. Sau đó là nó sẽ áp dụng các thuộc tính cần thay đổi lên vùng chọn đó.

Một đoạn CSS bao gồm 4 phần như thế này:

|  |  |
| --- | --- |
|  | vùng chọn {     thuộc tính : giá trị;     thuộc tính: giá trị;     .....  } |

Nghĩa là nó sẽ được khai báo bằng vùng chọn, sau đó các thuộc tính và giá trị sẽ nằm bên trong cặp dấu ngoặc nhọn {}. Mỗi thuộc tính sẽ luôn có một giá trị riêng, giá trị có thể là dạng số, hoặc các tên giá trị trong danh sách có sẵn của CSS. Phần giá trị và thuộc tính phải được cách nhau bằng dấu hai chấm, và mỗi một dòng khai báo thuộc tính sẽ luôn có dấu chấm phẩy ở cuối. Một vùng chọn có thể sử dụng không giới hạn thuộc tính.

* + 1. JAVASCRIPT

**Javascript** là một ngôn ngữ thông dịch, chương trình nguồn của nó được nhúng hoặc tích hợp vào tập tin HTML. Khi trang web được tải trong trình duyệt hỗ trợ javascript, Trình duyệt sẽ thông dịch và thực hiện các lệnh Javascipt.

Javascript là một ngôn ngữ có đặc tính:

* + - Đơn giản.
    - Động (Dynamic).
    - Hướng đối tượng (Object Oriented).

Có 2 cách để nhúng Javascript vào trong tập tin HTML

Cách 1: Viết chưong trình Javascript trực tiếp trong file HTML

|  |  |
| --- | --- |
|  | <script type=”text/javascript”>      //Các lệnh Javascript  </script> |
|  |  |

Cách 2: Sử dụng tập tin javascript bên ngoài :

Cũng giống như CSS ngoại tuyến, chúng ta cũng có thể nhúng Javascript vào tập tin HTML bằng cách liên kết đến một tập tin bên ngoài, đây cũng là phương thức được sử dụng nhiều nhất.

Với phương pháp này, các lệnh Javascrip sẽ được viết trong một file riêng biệt có phần mở rộng là **.js**(Ví dụ ta có tập tin my.js)

Để nhúng tập tin **demo.js** vào tập tin HTML ta sử dụng đoạn mã sau:

|  |  |
| --- | --- |
|  | <script type="javascript" src="demo.js" type="text/javascript"></script> |

* + 1. MYSQL

MySQL là [hệ quản trị cơ sở dữ liệu](https://vi.wikipedia.org/wiki/H%E1%BB%87_qu%E1%BA%A3n_tr%E1%BB%8B_c%C6%A1_s%E1%BB%9F_d%E1%BB%AF_li%E1%BB%87u) [tự do nguồn mở](https://vi.wikipedia.org/wiki/Ph%E1%BA%A7n_m%E1%BB%81m_ngu%E1%BB%93n_m%E1%BB%9F) phổ biến nhất thế giới và được các nhà phát triển rất ưa chuộng trong quá trình phát triển ứng dụng. Vì MySQL là cơ sở dữ liệu tốc độ cao, ổn định và dễ sử dụng, có tính khả chuyển, hoạt động trên nhiều hệ điều hành cung cấp một hệ thống lớn các hàm tiện ích rất mạnh. Với tốc độ và tính bảo mật cao, MySQL rất thích hợp cho các ứng dụng có truy cập CSDL trên internet. MySQL miễn phí hoàn toàn cho nên bạn có thể tải về MySQL từ trang chủ.

CHƯƠNG 2: PHÂN TÍCH VÀ THIẾT KẾ HỆ THỐNG

2.1. Yêu cầu cơ bản của hệ thống