# Về ngôn ngữ lập trình

Lập trình ứng dụng Web hiện tại có thường dùng ngôn ngữ sau : .Net, Java, PHP. Sơ lược về các ngôn ngữ trên :

**Mô hình MVC**

MVC là viết tắt chữ cái đầu của Models, Views, Controllers. MVC chia giao diện UI (User Interface) thành 3 phần tương ứng, đầu vào của các controller là các điều khiển thông qua HTTP request, model chứa các miền logic, view là những thứ được sinh ra trả về cho trình duyệt. Sau đây là một vài chi tiết trong 3 thành phần của MVC:

-  Model:  Được giao nhiệm vụ cung cấp dữ liệu cho cơ sở dữ liệu và lưu dữ liệu vào các kho chứa dữ liệu. Tất cả các nghiệp vụ logic được thực thi ở Model. Dữ liệu vào từ người dùng sẽ thông qua View để kiểm tra ở Model trước khi lưu vào cơ sở dữ liệu. Việc truy xuất, xác nhận và lưu dữ liệu là một phần của Model.

-  View: Hiển thị các thông tin cho người dùng của ứng dụng và được giao nhiệm vụ cho việc nhận các dữ liệu vào từ người dùng, gởi đi các yêu cầu đến bộ điều khiển, sau đó là nhận lại các phản hồi từ bộ điề khiển và hiển thị kết quả cho người dùng. Các trang HTML, ASPX, ASCX các thư viện thẻ và các file nguồn là một phần của View.

-  Controller: là tầng trung gian giữa Model và View. Controller được giao nhiệm vụ nhận các yêu cầu từ phía máy khách. Một yêu cầu được nhận từ máy khách được thực hiện bởi một chức năng logic thích hợp từ thành phần Model và sau đó sinh ra các kết quả cho người dùng và được thành phần View hiển thị. ActionServlet, Action, ActionForm, struts-config.xml là các thành phần của Controller.

**JAVA**

* **Giới thiệu về Java**

Java là ngôn ngữ lập trình hướng đối tượng – OOP (Object-Oriented Programming). Tháng 1/2010, Oracle mua lại Java tử Sun và phiên bản hiện tại JavaSE 7.

* **Đặc điểm của Java**

Java là ngôn ngữ hỗ trợ mạnh hướng đối tượng tương tự như C++, C#.

Java là được thiết kế để biên dịch mã nguồn thành bytecode, bytecode sau đó sẽ được môi trường thực thi (runtime environment) chạy. Vì Java không biên dịch trực tiếp thành mã máy nên sẽ chạy được trên bất kỳ hệ điều hành, phần cứng cụ thể nào với điều kiện có máy ảo. Do đó các ứng dụng Java có mặt trên nhiều thiết bị như máy tính để bàn, smart-phone, thiết bị mạng….

Chưa phải đóng phí bản quyền. Có cộng đồng phát triển mạnh. Nguồn tài liệu tham khảo dồi dào.

* **Yêu cầu phần mềm và phần cứng khi lập trình Java**

Để lập trình Java thì cần có các phần mềm sau nền tảng Java là Java 2 Platform, Standard Edition (J2SE) và môi trường phát triển ứng dụng tích hợp IDE như Eclipse hoặc NetBean.

Để lập trình một ứng dụng Web bằng Java thì phải dựa trên công nghệ JavaServer Pages (JSP)/Servlet. Ngoài ra để lập trình nâng cao thì cần tìm hiểu thêm về Hibernate để tạo kết nối cơ sở dữ liệu, Spring MVC để hỗ trợ lập trình theo mô hình MVC (Model-View-Controller).

Sử dụng Web Server thường là Tomcat, Glassfish.

Điểm bất tiện nhưng cũng là điểm mạnh của Java đó là thư viện, người lập trình phải tự tìm các thư viện phù hợp với yêu cầu sử dụng để thêm vào chương trình, điều này mang lại cho chương trình được gọn nhẹ, lập trình viên hiểu rõ về chương trình mình làm. Nhưng việc lựa chọn cũng là một khó khăn vì đôi khi có rất nhiều thư viện thực hiện cùng một nhiệm vụ và nhiều phiên bản khác nhau.

* **Lập trình ArcGIS với Java**

Xây dựng ứng dụng GIS với Arcgis Server sẽ sử dụng Java Web Application Developper Framework (Java Web ADF). Java Web ADF là một framework có hỗ trợ AJAX được xây dựng trên JSF (Java Server Face) và J2EE nhằm mục đích tạo ra các control điều khiển trên server , xử lý yêu cầu từ phía client với sự giúp đỡ của thư viện javascript.

Người lập trình Java xây dựng các ứng dụng Web của mình sử dụng các SOAP Services (Simple Object Access Protocol) để thực hiện các chức năng cơ bản do ArcGIS Server cung cấp. SOAP Services là cách giao tiếp dữ liệu bằng XML (eXtensible Markup Language).

.NET

* **Giới thiệu về .NET Framework:**

Là một nền tảng ứng dụng web (*web application framework*) được phát triển và cung cấp bởi Microsoft. Cái tên đã nói lên tất cả .NET là nền tảng mà mọi ứng dụng .NET phải chạy trên đó, Phiên bản của .NET framework hiện tại là .NET4. Ngôn ngữ thường được được sử dụng để lập trình trên .NET là C#, Visual Basic.

* **Đặc điểm của .NET là gì ?**

Nền tảng .NET hỗ trợ rất nhiều ngôn ngữ lập trình, nhưng tiểu biểu nhất có lẽ là C#, và Visual Basic, là những ngôn ngữ lập trình hướng đối tượng mạnh.

Lập trình ứng dụng Web trên .NET thì có thể lựa chọn sử dụng Web Form hoặc Web MVC.

Lập trình với Web Form là phong cách lập trình Web gần giống với lập trình kéo thả trên desktop. Đặc điểm đáng lưu ý của Web Form đó là các sự kiện và control(button, selectbox…) đều do server quản lý. Do ứng dụng Web là môi trường không lưu trạng thái (stateless) nên WebForm sử dụng biếnViewState để quản lý lưu tình trạng các Web Control thực hiện các hành động cho UI (User Interface). Khi trang web có số control lớn đồng nghĩa với kích thước trang lớn, tốc độ load các trang sẽ rất chậm. Ưu điểm là phát triển ứng dụng nhanh do các control đã có sẵn.

So sánh với WebForm là lập trình với mô hình MVC, là mô hình xây dựng ứng dụng Web thông dụng hiện nay. Nội dung của mô hình này thì như đã trình bày ở trên, với ASP.NET thì các sự kiện sẽ được Controllers quản lý, còn control không do Server quản lý nữa, do đó tốc độ sẽ được cải thiện. Một điều nữa là lập trình với mô hình đối với người đã lập trình Web bằng ngôn ngữ khác sẽ dễ tiếp cận hơn.

|  |  |  |
| --- | --- | --- |
| Các tính năng | ASP.NET WebForm | ASP.NET MVC |
| Kiến trúc chương trình | Kiến trúc mô hình WebForm 🡪Bussiness 🡪 Database | Database Kiến trúc sử dụng việc phân chia chương trình thành: Models, Views, Controllers |
| Cú pháp chương trình | Sử dụng cú pháp của WebForm, tất cả các sự kiện và controls do server quản lý | Các sự kiện được điều khiển bởi controllers, các controls không do server quản lý. |
| Truy cập dữ liệu | Sử dụng hầu hết các công nghệ truy cập dữ liệu trong ứng dụng | Phần lớn dùng LINQ và SQL class để tạo mô hình truy cập đối tượng. |
| Debug | Debug phải thực hiện tất cả bao gồm các lớp truy cập dữ liệu, sự hiển thị, điều khiển các controls. | Debug có thể sử dụng các unit test để kiểm tra các phương thức trong controllers. |
| Tốc độ phân tải | Tốc độ phân tải chậm khi trong trang có quá nhiều các controls vì ViewState quá lớn | Phân tải nhanh hơn do không phải quản lý ViewState để quản lý các controls trong trang. |
| Tương tác với JavaScript | Tương tác với JavaScript khó khăn vì các controls được điều khiển bởi server | Tương tác với JavaScript dễ dàng vì các đối tượng không do server quản lý điều khiển không khó |
| URL address | Cấu trúc địa chỉ URL có dạng:  <filename>.aspx?&<các tham số> | Cấu trúc địa chỉ rành mạch theo dạng Controllers/Action/ID |

THAM KHẢO

<http://asp.net.vn> (thuộc ASP.NET VIETNAM)