Vaadin

Việt Nam

2014   
Vaadin Ltd

Vaadin 7 Edition – v3

Đây là sách cung cấp nhiều về tổng quan Vaadin Framework và đảm bảo những chủ đề quan trong mà khi bạn có thể gặp khi phát triển ứng dụng với nó. Nhiều tại liệu được chi tiết các class, interface và các phương thức được ghi trong Tham khảo Vaadin API.

Phiên bản này chủ yếu bao gồm Vaadin 7.2 phát hành vào tháng 5 năm 2015. Phiên bản bao gồm hỗ trợ cho các biểu tượng chữ và bố trí các layout. Trên Client –side, xử lý các thay đổi dễ dàng hơn với dấu chú thích @OnStateChange. Ngoài ra các chủ đề được viết trong Vaddin 7.3 là các tính năng đặc trưng.  
  
Ngoài những thay đổi trong famework core, phiên bản tài liệu này coati liệu cho các tiện ích (add on) TestBeanch 4 và Touchkit 4, chưa được phát hành tại thời điểm hiện tại. Các chươgn được cập nhật dựa trên các phiên bản tiện ích (add-on) phát hành trước đây, do đó phiên bản cuối cùng có một số thay đổi.

Viết hướng đân này là một công việc đang diễn ra và nó hiếm khi hoàn toàn kịp với các sản phẩm. Một số tính năng có thể không được bao gồm trong sách này được nêu ra. Đói với phiên bạn mới nhất, xin vui lòng xem bản trực tuyến có tại : <http://vaadin.com/book>. Bạn có thể tìm thấy phiên bản PDF và EPUB của cuốn sách đó. Bạn có thể tìm thấy các phiên bản khác dễ dàn hơn với sách in. Các chỉ ố trên sách là không đầy đủ và sẽ được mở rộng sau. Mục đíc của phiên bản rút gọn nhẹ in nhiều hơn đế có một sách hướng dẫn đầy đủ và vãn còn phù hợp trong túi của bạn.

Ngoài ra, nhiều tính nnăg Vaadin 7 được giới thiệu như hướng dẫn nhỏ, trong đó có sẵn Wiki Vaadin tại :.<https://vaadin.com/wiki/-/wiki/Main/Vaadin+7.>

Sách này dành cho ai ?

Đây là cuốn sách dành cho nhà phát triển, ngừoi sử dụng hoặc đang xem xét sử dugj, Vaadin để phát triển ứng dụng web.

Cuốn sách này giả định rặng bạn có một số kinh nghiệp lập trình với Java, nhưng nếu không, nó sẽ dễ dàng để bắt đầu học Java với Vaadin như với bất kỳ một Famework UI nào khác. Không hiểu biết về AJAX là cần thiết vì nó cũng ẩn từ nhà phát triển.

Bạn có thể đã sử dụng một số nền tảng giao diện hướng đối tượng Java cho Desktop . chẳng hạn như AWT, Swing hoặc SWT, hoặc thư viện như Qt cho C++. Kiến thức này rất hữu ích cho việc hiểu ccác phạm vi của Vaadin, mô hình lập trình hướng sự kiện, và các khái niệm của nền tảng giao diện người dùng.

Nếu bạn không có bàn tay cả một nhà thiết kế đồ họa web, hiểu biết cơ bản về HTML, CSS có thể giúp bạn phát triển các chủ đề (THEMES) trình bày cho ứng dụng của bạn. Giới thiệu CSS được sử dụng. Kiến thứ về Google Web Tookit (GWT) co sthể hữu ích nếu bạn phát triển hoặc tích hợp thành phần cho client-side.

Cấu trúc tổ chức của sách.

Sách Vaadin gới thiệu nhiều Vaadin là gì và làm thế nào bạn sử dụng nó để phát triển các ứng dụng web.

1. **Giới thiệu**
2. **Giới thiệu**

Chương này cung cấp thới thiệu về kiến trúc được hỗ trợ bởi Vaadin, ý tưởng code đằng sau nền tảng, và một số bối cảnh lịch sử.

1. **Bắt đầu với Vaadin**

Chương này đưa ra hướng dẫn thực hành để cài đặt Vaadin và toolchain tài liệu tham khảo, trong đó có Vaadin Plugin cho Eclipse, làm thế nào để chạy và gỡ lỗi các chương trình, và làm thế nào để tacọ các dự án riêng trong Eclipse.

1. **Kiến trúc**

Chương này đưa ra một giới thiệu về kiến trúc của Vaadin và công nghệ lớn của nó, bao gồm Ajax, Google Web Toolkit và lập trình hướng sự kiện.

1. **Nền tảng Server-side**
2. **Viết ứng dụng Server-Side**

Chương này cung cấp cho tất cả các kiến thực yêu cầu thực tế để tạo các ứng dụng với Vaadin, chẳng hạn như quản lý của sổ, vòng đời ứng dụng, phát triển trong một Servlet container, và quản lý các sự kiện, các lỗi và tài nguyên.

1. **Sử dụng thành phần giao diện người dùng User Interface**

Chương này chủ yếu cung cấp các tài liệu tham khảo cho tất cả các thành phần của nền tảng giao diện người dùng trong Vaadin và tính năng quan trọng . Các ghi chú đưa ra nhiều ví dụ example cho sử dụng mỗi thành phần.

1. **Quản lý bố cục (layout)**

Chương này mô tả các phầnh phần bố cụ , được sử dụng cho quản lý bố cụ của giao diện ngưuời dùng UI, giống như bất kỳ nền tảng ứng dụng desktop nào.

1. **Thiết kế giao diện ảo với Eclipse**

Chương này đưa ra hướng dẫn các sử dụng trình soạn thảo cho Eclipse, có trong các plugin Vaadin cho Eclipse IDE.

1. **Các chủ đề (Themes)**

Chương này ddưa ra một giới thiệu về CSS Cascading Style Sheets CSS và giải thích làm thế nào có thể xây dựng các chủ đề cho ứng dụng của bạn.

1. **Kết hợp thành phần dữ liệu**

Chương này cung cấp một cách tổng quan về mô hình xây dựng dữ liệu của Vaadin, bao gồm các thuộc tính, các mục, và cách thành phần chứa.

1. **Vaadin SQLConntainner**

Chương này cung cấp tài liệu cho SQLContainner, cho phép Vaadin kết hợp các thành phần để truy vấn SQL.

1. **Chủ đề nâng cao ứng dụng web**

Chương này cung câp nhiều các chủ đề đặc biệt mà thường cần thiết trong các ưng sdụn, chẳng hạn như mở một cửa sổ trình duyệt mới, nhúng các ứng dụng trong các trang web, quản lý tài nguyên ở mức thấp, các phím tắt, gỡ lỗi.,,vvvv

1. **Tương tác Portal**

Chương này mô tả sự phát triển của các ứng dụn Vaadin như Portlet mà bạn có thể triển khai bất kỳ cổng thông nào hỗ trợ Java, Chương này cũng mô tả các hỗ trựo đặc biệt cho Liferay và Controll Pannel, IPC và WSRP và các tiện ích.

1. **Nền tảng Client-side**
2. **Phát triển Vaadin Client Side**

Chương này cung cấp giới thiệu để tạo và phát triển các ứng dụng ở phía client và các widget, bao gồm cài đặt, biên trình và gỡ lỗi.

1. **Các ứng dụng Client –Side**

Chương này mô tả làm thế nào phát triển các ứng dụng phía client và làm thế nào để tương tác chủ đề (THEMES) với dịch vụ Back-end.

1. **Widgets phía client**

Trong chương này mô tả xây dựng widget(thành phần của máy khách) có sẵn để phát triển phía client. Các vật dụng bao gồm GWT Google Web Toolkit cũng như Vaadin widgets.

1. **Tương tác với Server-side**

Chhương này mô tả làm thế nào để tương tác widget phía client với các đối tác phía server cho mục đích tạo mới các thành phần phía server. Chương này bao gồm các thành phần tích hợp JavaScript.

1. **Những tiện ích(add-ons) Vaadin**
2. **Sử dụng Tiện ích (add-ons) Vaadin**

Chương này cũng cấp hướng dẫn để tải về và cài đặt add-on từ thư mục Vaadin.

1. **Các biểu đồ Vaadin**

Chương này đưa ra dẫn chứng việc sử dụng add-on Chart cho tương tác các biểu đồ. Các add-on bao gồm các thành phần Chart và Timeline.

1. **JPA Containner**

Chương này đửa ra tài liệu về add-on JPAContainner, cho phép liên kết các thành phần Vaadin trực tiếp với quan hệ và các cơ sở dữl iệu khác sử dụng JPA Java Persistence.

1. **Các ứng dụng Mobile với TouchKit**

Chương này đưa ra các ví dụ và tài liệu tham khảo để sử dụng Vaadin TouchKit cho việc phát triển các ứng dụng di động.

1. **Vaadin TeshBench**

Chương này đưa ra các tài liệu hướng dẫn đầy đủ về cách sử dụng các công cụ Vaddin testbench bản ghi và thực hiện các bài kiểm tra hồi của ứng dụng Vaadin.

Phụ lục A – Các bài hát Vaadin

**Tài nguyên bổ sung**

Các trang web cung cấp râ nhiều tài nguyên có thể giúp bạn hiểu Vaadin là gì ? và những gì có thể làm việc với nó ? và là như thế nào để có thể làm nó.

Các ứng dụng Demos

Các ứng dụng Demo rất quan trọng cho Vaadin simple, chứng tỏ việc sử dụng của tất cả các thnàh phần cơ bản và tính năng. Bnạ có thể chạy nó trên mạng tại : <http://demo.vaadin.com/> hoặc tải nó như một file War (Web Application aRward)

Hầu hết các vị dụ trong cuốn sách này và nhiều người khác có thể tìm thấy trực truyếtn tại : <http://demo.vaadin.com/book-example-vaadin7/book>

**Cheat Sheet**

Hai trang Cheat sheet minh họa hệ thống cơ bản của giao diện người dùng và phân cấp cơ bản của giao diện người dùng và dữ liệu ràng buộc các class và interface. Bạn có thể tải nó tại <http://vaadin.com/book>

**Refcard**

## 6 trang DZone Refcard đửa ra một cách tổng quan để phát triển ứng dụng với Vaadin. Nó bao gồm một sơ đồ giao diẹn diện người dùng và class kêt hợp dữ liệu. Bạn có thể tìm hiểm thếm thông tinvè nó tại : <https://vaadin.com/refcard>

**Địa chỉ sách hướng dẫn.**

Địa chỉ sách là mọt ứng dụng mẫu đi kèm với một hướng dẫn cung cấp cho bước theo các ứngdụng chi tiết để tạo một dứng dụng web thực tế cuộc sống với Vaadin. Bạn có thể fim hướng dẫn tại website sản phẩm.

**Trang web cho nhà phát triển**

<http://dev.vaadin.com/> cung cấp nguồn tài nguyên online, chẳng hạn như hệ thống vé, và wiki cho phát triển, kho nguồn, thời gian hoạt động, sư liệu quan trọng phát triển và như vậy.

Wiki cung cấp hướng dẫn cho nhà phát triển, đặc biệt cho những người muốn kiểm tra tài nguyên. Các bìa viêt ký thuật đối với ứng dụng của Vaadin. Với các hệ thống khác nhau chẳng hạn như JSP, Maven, Spring, Hibernate and cổng thông tin Portals. Wiki cũng cung cấp câu trả lời cho các câu hỏi thường gặp FAQ (Frequently Asked Questions)

**Tài liệu trực tuyến(Online Document)**

Bạn có thể đọc cuốn sách này trực tuyêt tại <http://vaadin.com/book> . Rât nhiều tài liệu bổ sung, bao gồm kỹ thuật HOWTOS, và tài liệu hướng dẫn khác cũng có sãn trên trang web Vaadin.

**Hỗ trợ**

Bị mắc kẹt với một vấn đề ? Không cần phải mất mái tóc của bạn trên nó, Nền tảng Vaadin và công ty Vaadin cung câp hỗ trợ cho tất cả các nhu cầu của bạn.

**Cộng đồng hỗ trợ.**

Bạn có thể tìm thấy diễn đànn người sử dụng và phát triển tại : <http://vaadin.com/forum>. Vui lòng ử dụng diễn đàn để thảo luật về bất kỳ các vấn đề bạn gặp phải, muốn cho các tính năng, và như vậy. Câu trả lời cho vấn đề của bạn có thể nằmm trong các bài viết của forum, vì vậy việc tìm kiếm các cuộc thảo luận luôn là cách tốt nhât để bắt đầu.

**Báo lỗi**

Nếu bạn tìm thấy một lỗi có thể trong Vaadin, các ứng dụng DEMO, hoặc tài liệu, xin vui lòng báo cáo nó vằng một vé tại trang web của nhà phát triển tại : <http://dev.vaadin.com/> . Bạn có thể muốn tìm kiếm các thẻ trước khi kiểm tra tạo thẻ mới, bạn có thể làm cho một vé để thể thực hiện một yêu cầu báo cáo mới cho tính năng mới đó là tốt, hoặc đê nghị sử đổi một tính năng hiện có.

Vaadin cung câp dịch vụ thương mại hỗ trợ và đào tạo cho nền tảng Vaadin và các sản phẩm liên quan. Tìm hiểu thêm về các sản phẩm thương mại , xem chi tiết tại : <http://vaadin.com/pro>

**Thông tin về tác giả**

**Marko Gronross** là chuyên gia về viết và phát triển phầm mềm đã làm việc tại Vaadin Ltd tại Turku, Phần Lan( Finland). Ông đã tưng tham gia vào các dự án phát triển web từ năm 1994 và đã làm việc trên một số nền tảng như C, C++, Java. Ông đã hoạt động trong nhiều dự án ngôn ngữ mở và có bằng thạc sĩ về Khoa học máy tính.

**Lời cảm ơn**

Phần lớn của cuốn sách là kết quả của việc đoàn kết chặt chẽ trong nhóm phát triện tại Vaadin Ltd Jonnas Lehtinen, Giám đốc điều hành của Vaadin Ltd, viết đề cương đầu tiên của cuốn sách mà đã chở thành cho hai chương đầu tiên, kể từ đó, Marko Gronross dã trở thành tác giả chính và biên tập. Đội ngũ phát triển đã đóng góp một số đoạn, trả lời rất nheièu câu hỏi kỹ thậut, xem xét nhiều hướng dẫn, và có nhiều chính sửa.

Nhưng người đóng góp theo thứ tự thời gian

Joonas Lehtinen

Jani Laakso

Marko Grönroos

Jouni Koivuviita

Matti Tahvonen

Artur Signell

Marc Englund

Henri Sara

Jonatan Kronqvist

Mikael Grankvist (TestBench)

Teppo Kurki (SQLContainer)

Tomi Virtanen (Calendar)

Risto Yrjänä (Calendar)

John Ahlroos (Timeline)

Petter Holmström (JPAContainer)

Leif Åstrand

**Thông tin về Vaadin Ltd**

**Vaadin** **Ltd** là công ty chuyên về thiết kế và phát triển các ứng dụng Internet phong phú. Công ty cung cấp lập kế hoạch, thực hiện, và hỗ trợ dịch vụ cho các dự án phần mềm của khách hang, cũng như phát triển và các dự án con. Vaadin Framewwork, trước đây gọi là IT Mill Toolki là mã nguồn hàng đầu của công ty, mà nó cung cấp các dịch vụ thương mại phát triển và hỗ trợ.

Mục lục

1. **Giới thiệu**

**Các nội dung**

1 Giới thiệu

2 Bắt đầu với Vaadin

3 Kiến trúc

Đây là phần đầu trong 3 chương, trình bày ý tưởng nền tảng Vaadin và nó bao gồm hai mô hình, Server-side và client-side, mô tả và cung câp tổng quan về kiến trúc.

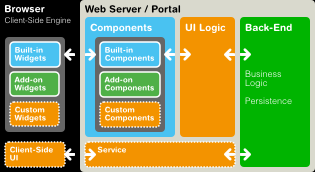
**Chương 1 : Giới thiệu**

Chương này cung cấp hiểu biết thiệu kiến thức về phát triển phầm mềm Vaadin. Chung tôi cung cấp một số thông thiết kế về

1. **Giới thiệu**
   1. Tổng quan

Vaadin Framework là nền tản phát triển ứng dụng web, được thiết kế để tạo và bảo trị web dễ dang. Vaadin hỗ trợ hai mô hình lập trình: Server side và Client Side. Mô hình Lập trình hướng Server mạnh mẽ. Nó giúp bạn tập trung vào web và giao dienẹ ngừoi dùng giống như một số ngôn ngữu lập trình desktop mà bạn đã dùng như AWT, Swing, SWT. Nhưng dễ hơn.

While ngôn ngữ web cách dễ để dành thời gian của bạn học công nghệ web mới, bạn có thể muốn sản phẩm và trên một ứng ứng dụng logic. Nền tảng Vaadin trên Server quản trị giao diện người dùng trên trình duyệt và AJAX giao tiếp giữa trình duyệt và Server. Với Vaadin, bạn không cần học công nghệ trên trình duyệt như HTML và JS



**Hình 1.1. Kiến trúc ứng dụng Vaadin**

**Hình 1.1. Kiến trúc ứng dụng Vaadin là kiến trúc cơ bản của các ứng dụng web với Vaadin. Ứng dung Sever side bao gồm nền tảng Server và kỹ thuật Client Side. Kỹ thuật chạy trên trình duyệt như mã Javascript, tạo giao diện người dùng và phân tán giao diện tương tác với ngừoi dugnf. ÚI logic của một ứng dụng chạy nhiều trên các Java Servlet trong một ứng dụng Java server.**

**Như một phương tiện để thực thi như JavaScript trên trình duyệt, không plugin trình duyệt cần thiết cho sử dụng ứng dụng để tạo với Vaadin. Cung cấp nó với trên nền tảng dựa trên Flash, Java Applét hoặc Pluging. Vaadin dựa trên hỗ trợ GWT Google Web Toolkit cho một hoặc trình duyệt hoặc phát triển không cần quan tâm đến trình duyệt hỗ trợ.**

**Bởi vì HTML, JS và các công nghệ trình duyệt khác được ẩn trên ứng dụng logic, bạn có thể nghĩ trình duyệt web chỉ như một nên tảng mỏng. Một hiển thị mỏng của giao diện người dùng và giao tiếp với sự kiện người dugnf trên máy chủ tại một mức độ thấp. Điều khiểu mưucs logic giao diện người dùng chạy dựa trên kiến trúc với ứng dụng client có thể bao gồm nhiều đặc điểm ứng dụng giao tiếp giưuã client và server. Xóa bỏ tầng giao diện người dùng từ kiến trúc ứng dụng.**

**Bên cạnh mô hình phát triển hướng server, Vaadin tạo cách sử dụng tốt nhất sử dụng AJAX nền tảng (Asynchronous Java Script and XML) tạo nó có thể tạo làm phong phú ứng dụng web trên internet RIA Rich Internet Application như việc phản hồi và tương tác với ứng dụng trên desktop.**

**Ngoài ra, phát triển ứng dụng Serverside, bạn có thể triển khải trên client side bởi tạo thêm một Widget trên Java, và các ứng dụng Client-side nó có thể chạy và trên trình duyệt. Nền tảng Client side Vaadin bao gồm GWT Google Web Toolkit, cung cấp trình biên dịch từ Java để Java chạy trên trình duyệt, như một nền tảng giao giao diện người dugnf xa hơn. Nó được áp dụng Java trên tất cả side( Server-side và Client- side)**

**Vaadin sử dụng như một phương tiện client side cho việc tạo tự động các giao diện người dùng của ứng dụng Serverside trên trình duyệt. Tất cả giao tiếp client –server được ẩn dưới. Vaadin được thiết có khả năng mở rộng và bạn có thể sử dụng Widget thứ 3 bất kỳ một cách dễ dàng. Ngoài ra đê thành phần báo cáo cũng được sử dụng trong Vaadin, trong thực tế bạn có thể tìm hàng tram các tiện ích trên thư mục Vaadin.**

**Nền tảng Vaadin định nghĩa nhiều dải phân cách rõ ràng giữ cấu trúc của giao diện người dùng và nó là giao diện cho phép bạn phát triển chủ đè. Áp dụng chủ đề theme, điều khiển giao diện bới CSS và mẫu HTML . Như nhà cung cấp hoàn hảo cho cá chủ đề mặc định, bạn có thể không thường xuyên sử dụng đê tạo sự cá nhân hóa, nhưng bạn có thể nếu bạn cần nó (Chi tiết xem chương 8 : Chủ đề )**

**Chúng tôi hi vọng đây là kién trúc cơ bản và một số tính ănng của Vaadin cho bạn bây giờ. Bạnc so thể đọc nhiều thông tin hơn sau chương 3 : Kiến trúc, hoặc nhảy để đọc nhiều phần thực hành trong chương 4 : Viết một Ứng dụng web Server.**

* 1. **Ví dụ ứng dụng**

Bây giờ chúng ta thực hiện ví dụ theo truyền thống khi học một nền tảng ngôn ngữ lập trình mới. “Hello world”.

import com.vaadin.server.VaadinRequest;

import com.vaadin.ui.Label;

import com.vaadin.ui.UI;

@Title("Hello Window")

public class HelloWorld extends UI {

    @Override

    protected void init(VaadinRequest request) {

        // Create the content root layout for the UI

// Tạo bố cục gốc cho giao diện người dùng

        VerticalLayout content = new VerticalLayout();

        setContent(content);

        // Hiển thị ra trình duyệt

        content.addComponent(new Label("Hello World!"));

    }  
}

Ứng dùng Vaadin có một hoặc nhiều hơn UI đưuọc kết thừa từ lớp **com.vaadin.ui.UI** . Một giao diện người dùng là một thành phần của trang web khi ứng dụng Vaadin chạy. Một ứng dụngc so thể co snhiều UI trong cùng trang, cổng thông tin hoặc cửa sổ khác hoặc các tab. Một UI liên quan tới phiên làm việc sử dụng và phiên làm việc Session được tạo cho mỗi sử dụng. Trong phần trước của giao diện người dùng UI Helloworld, nó dược ẩn dưới phiên làm viên Seesion được tạo khi người dùng user truy cập lần đầu ứng dụng bằng cách mở trang web và phương thức init() được chạy tại lần này.

Trong tất cả các ví dụ trên, tiêu đè là trang được hiển thị đầu đề của mỗi cửa sổ trình duyệ hoặc tab, nó đượng định nghĩa bằng sử dụng ký hiệu annotation,. Ví dụ sử dụng một thành phần bố cục Layout như một nội dung gốc của UI, hầu hết được tạo ra trong các ứng dụng Vaadin, thông thường có nhiều thành phần hơn. Nó tạo một thành phần giao diện Label, hiển thị văn bản thường và thiết lập nội dụng “Hello World”. Label được add thêm vào bố cục Layout.

Đây là kết quả của ứng dụng Hello World, khi trình duyệt mở và hiển thị kết quả Hello World.

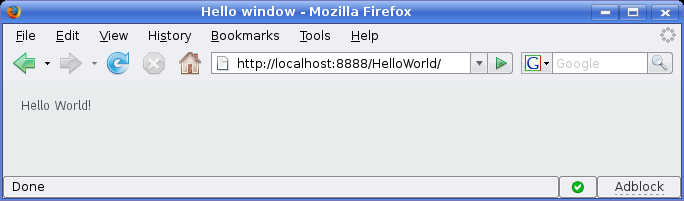


Figure 1.2, Hello World Application.

Để chạy chng trình, bạn có thể tạo thư mục như một file ứng dụng web WAR và triển khai not trên server, xem chi tiết thêm tại mục : 4.8 Triển khai một ứng dụng

Triển khai một ứng dụng client, bạn có thể viết ứng dụng Hello World một cách dễ dàng.

public class HelloWorld implements EntryPoint {

    @Override

    public void onModuleLoad() {

        RootPanel.get().add(new Label("Hello, world!"));

    }  
}

Không cần thiết lập tiêu đề trang đề ở đây, bởi vì nó thường sử dụng trang HTML, được thực thi. Ứng dụng được biên dịch thành JavaScript với trình biên dịch Vaadin Client (or trình biên dịch GWT). Nó nó nhiều kiểu, nhưng để viết Widget phía Client bạn có thể sử dụng nó từ phía Ứng dụng Vaadin Server. Một số thông tin liên quan Chương 13 : PHát triển Vaadin phía Client

* 1. **Hỗ trợ Eclipse IDE**

Trong khi Vaadin không kết hợp với bấn kỳ tính năng nào của IDE và có thể trong không dễ sử dụng trong thực thế với bất kỳ IDE nào khác, chung tôi cung cấp hỗ trợ đặc biệt cho Eclipse IDE, môi trường phổ biến nhất được sử dụng phát triển Java. Plugin Vaaidin cho Eclipse cung cấp hỗ trợ trong sẽ giúp bạn :

+ Tạo mới các dự án Vaadin

+ Tạo mới Chủ đề custom

+ Tạo mới các Widget custom

+ Tạo mới các thành phẩn tổng hợp với biên tập hình ảnh

+ Dễ dàng cập nhật phiên bản mới.

Cách sử dụng Plugin cho Eclipse được khyến cáo để triến khai Vaadin cho phát triển. Tải gói cài đặt có chưa các file Jars hoặc đinh nghĩa các tài nguyên phụ thuộc trên Maven.

Cài đặt và cập nhật plugin Eclipe trình bày lại phần 2.4 Cài đặt Plugin Vaadin cho Eclipse” và tạo mới một ứng dụng Vaadin sử dụng Plugin trong phần 2.5 “ Tạo mới dự án. Xem phần 8.5 Tạo mới chủ đề Them trong Eclipse, phần 16.2 Bắt đầu Vaadin với Eclipse simple” và chương 7. Thiết kế giao diện người dùng cho tạo và sử dụng các tính năng của plugn.

* 1. **Mục tiêu và triết lý**

Đơn giản chỉ cẩn đặt, tham vọng của Vaadin là trở thành công cụ tốt nhất khi cso thể nói đến việc tạo ra giao diện người dùng cho ứng dụng kinh doanh. Nó được dễ dàng để áp dụn cho thiết kế và hỗ trợ cả hai mức độ và lập trình viên tiên tiến. cũng như khả năng sử dụng đồ hòa của các chuyên gia thiết kế.

Khi thiết kế Vaadin, chú tôi đã theo triết lý đưuọc ghi trong cách nguyên tắc dưới đây.

**Công cụ đúng cho đúng mục đích**

Bởi vì mục tiêu của chung tôi cao, tập trung phải rõ ràng, Vaadin được thiết kế cho tạo ứng dụng web. Nếu nó được thiết kế cho viêc tạo mới các website hoặc demo quảng cáo, Bạn có thể tìm, cho ví dụ “ JSP/JSF hoặc Flash cho mục đích đó.

**Tính đơn giản hóa và khả năng bảo trì.**

Chúng tôi đã nhấn mạnh đơn giản hóa và khả năng bảo trì. Điêu này liên quan đến việc thực hiện những điều sau đây được thiết lập thực hành tốt nhất cho các nền tảng giao diện người dùng, và đảm bảo giả pháp lý tưởng cho mục đích của mình không bị hỗn loạn và dư thừa (bloat)

**XML không được thiết kế cho ngôn ngữ lập trình**

Web là trung tâm tài liệu với nhiều liên kết trình bày giao diện người dùng. Nền tảng Vaadin giải phóng lập trình viên từ những hạn chế này. Nó là xa với giao diện tự nhiên và không đủ linh hoạt cho phép người dùng tương tác phức tạp và năng động.

**Các công cụ không nên giới hạn cho công việc của bạn**

Có nhiều thứ không nên giới hạn bất kỳ , cái gì bạn có thể làm với Frame; nếu cho mọt ssô lý do thành phần giao diện người dùng không hỗ trợ bạn cần, nó phải dễ dang để thêm mới cho ứng dụng của bạn. Khi bạn cần tạo mới một thành phần, một chức năng của Frame , nó sẽ rất dễ dàn cho việc sử dụng lại thành phần và bảo trì.

* 1. **Bối cảnh ra đời**

Vaadin không được viết nhanh chong, Sau khi làm việc với web giao diện người sử dụng từ khi bắt đầu với Web, nhóm phát triển bắt đầu lấy ý tưởng từ năm 2000 mẫu IT Mill, Nhóm đã phát triển sơ đồ ngôn ngữ lập trình mới có thể tạo mới và mẫu sử dụng thực hoặc cho ứng dụng thực tế sử dụng ngông ngữ lập trình thực.

Thư viện được tổ chức gọi là thư viên Millstone. Phiên bản đầu tiên được sử dụng lớn hơn ứng dụng web. IT Mill được thiết kế và thưucj hiện cho một công ty Dược quốc tế. IT Mill đã tạo một ứng dụng vào năm 2001 và hoạt động cho đến bây giờ. TỪ đó, công ty có sản xuất nhiều ứng dụng thực tế lớn hơn với thư viện có thể giải quyết nhiều vấn đề hơn và dễ dàng hơn.

**Phiên bản mở số 5**

Gói công cụ IT Mill 5, được phát hành vào cuối năm 2007, công cụ được đánh dấu bước xa hơn với AJAX. Kịch bản phía khách được tự động tạo ra với sự viết lại hoàn hảo giao diện người dùng với GWT, Google Web Toolkit.

Gói công cụ IT Mill được giứoi thiệu và áp dụng cho cả giao diện Serverside. Viết lại kịch bản phía client cho phép người dùng tất cả Java trên Client và phía server. Chuyển đổi từ JS sang GWT, đã tạo các thành phần cá nhân và tương tác vứoi cách thành phần đã tồn tại dễ dàng hơn trước, và snó cho phép dễ dàng tương tác với thành phần GWT. Áp dụng GWT trên phía Client là không gây ra bởi chính nó, nguyên nhân chan đổi bất kỳ trogn phía API máy chủ, bởi vì GWT là một cônng nghệ trên trình duyệt đang ẩn đằng sau API. Ngoài ra chủ đề được sử đổi trong bộ công cụ IT Mill.

Phát hành thứ 5 được xuất bạn dưới Apache 2 và không giới hạn phát hành mã nguồn mở, để tạo chi tiết nhanh hơn được sử dụng nền tảng và tạo thông tin để lập trình viên phát triển.

**Chào đón sự ra mắt mới Vaadin 6**

Bộ công cụ IT Mill được đổi tên thành Vaadin Framework, hoặc ngắn hơn Vaadin, mùa xuân 2009. IT Mill, công ty đã đổi tên thành Vaadin Ltd. Vaaadin có nghĩa người phụ nữ thành tuần lộc ở phần Lan.

Với Vaadin 6, số nhà phát triển sử dụng fame được tang, so với phiên bản phát hành, Vaadin Plugin cho Eclipse được ra mắt, trợ giúp chó tạo dự án Vaadin. Vaadin được giới thiệu sớm hơn trong 2010 cung câp nó, như một thành phần nhiều, nhiều tổ chức chuyên nghiệp được tạo và được sử dụng bởi hàng tram nhà phát triển. Năm 2013, chung tôi đã thấy sự phát triển trong các hệ thống cơ bản trên Vaadin. Số lượng người dung và ít hơn néu đo bằng hoạt động của diễn đàn đã đia qua tất cả các phía máy chủ và thậm chí GWT.

**Các sửa đổi lớn với Vaadin 7**

Vaadin 7 tập trung sử đổi các thay đổi Vaadin nhiều hơn Vaadin 6 đã làm. Nó chắc chắn hướng đối tượng web nhiều hơn Vaadin 6. Chung ta đang làm mọi thứu có thể giúp Vaain tăng cao hơn trong thế giới web. MỘt số công việc này dễ dang và hầu hết sử và gỡ lỗi, triển khai các tính năng nhưng dự định cao hơn và đứng vững chắc. Đó cũng là mục tiêu của Vaadin 7 để thiết kế lại sản phẩm để ngày càng cho phép kiến trúc Vaadin mới. Nhiều ththay đổi yêu cầu phá vỡ API vơi Vaadin 6, đặc biệt trên kịch bản phía client. Nhưng họ đã tạo thực hiến một số mong muốn mạnh mẽ đế tránh mang gánh nặng không cần thiến cho các tính năng sắp tới. Vaadin bao gồm nhiều các lớp cho việc tạo các bố cụ và tương thích các bố cục tồn tại dễ dàng hơn.

Bao gồm các bộ công cụ Web của GWT trong Vaadin 7 là phát triển quan trọng vì cnó có nghãi là bây giờ cung tôi cung cấp hỗ trợ tốt cho GWT. Khi Googl mở ra sự phát triển vào mùa hè 2012, Vaadin đã tham gia vào chỉ đạo mới GWT. Là một thành viên trong ủy ban, trong việc hướng tới thành công của GWT như làmột nền tảng phát triển Java Web.

1. **Bắt đầu với Vaadin**

Chương này cung cấp thực thành và cài đặt các công cụ và thư viện Vaadin và tạo mới một Project Vaadin.

* 1. Tổng quan

Bạn có thể triển khai ứng dụng Vaadin chủ yếu và môi trường phát triển trong Java SDK và bộ chứa Java Servlet. Vaadin có nhiêu đặc điểm hỗ trợ cho công cụ Eclipse nhưng giao tiếp với hỗ trợ với các project đã tồn tại với Netbean ID và IntellJ IDEA và bạn có thể sử dụng bất kỳ IDE JAVA nào

Quản lý Vaadin và thư viện Java có thể lấy bình thường hoặc sự dùng lời khuyên sử dụng một số sự phụ thuộc Maven. Vaadin là phân tán sự phụ thuộc trên maven, chẳng hạn như Ivy hoặc Gradle có trong Maven.

Vaadin có vô số các tùy chọn cài đặc ho IDE khách nhau và các quản lý sự phụ thuộc và bạn có thể cài đặt nó từ gói cài đặt.

Với công cụ Eclipse, sử dụng plugin Vaadin cho Eclipse được mô tả trong phần 2.4 Cài đặt Plugin cho Eclipse

Với công cụ Netbean : phần 2.7 Tạo mới dự án với Netbean IDE hoặc IntellJ IDEA.

Với Maven, Ivy, Gradle hoặc thành phần Maven , quản lý sự phụ thuôcj, dưới Eclipse, Netbeans IDE hoặc sử cụng dòng lệnh được mô tả trong phần 2.6 Sử dụng Vaadin với Maven.

Từ gói cài đặt với sự quản lý phụ thuộc được mô tả trong phần 2.9 Gói cài đặt Vaadin.

* 1. Cài đặt môi trường phát triển

Đây là phần hướng dẫn bạn từng bước thiết lập môi trường phát triển. Vaadin hỗ trợ nhiều công cụ, bạn có thể sử dụng bất kỳ IDE cho việc code và, chủ yếu bất gì máy chủ Java web nào cho việc triển khai ứng dụng, hầu hết các trình duyệt đều cho sử dụng nó, và bất kỳ nền tảng hệ điều hành nào cũng đều hỗ trợ Java.

Trong ví dụ dưới đây sử dung mộ chuỗi công cụ.

+ Windown, Linux hoặc Mac OS X

+ Sun J2SE Java 2 Standard Edition 6. ( JDK6).

+ Eclipse cho J2EE.

+ Apache Tomcat 7.

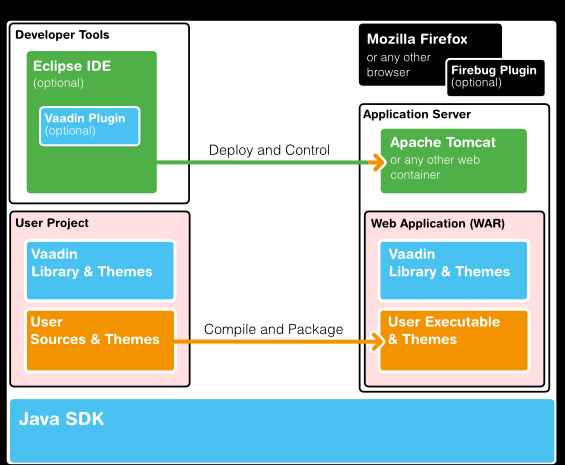
+ Trình duyệt Mozilla.

+ Công cụ gỡ lỗi Firebug.

+ Vaadin Framework.

Trên tất cả các chuỗi công cụ tốt để lựa chọn, nhưng bạn có thê sử dụng hầu hết bất kỳ công cụ nào một cách thoải mái.

Nếu bạn có ý định sử dụng máy chủ push, bạn cần phải sử dụng một máy chủ tương thích J2EE hỗ trợ WebSocket chẳng hạn như Glassfish, TomEE,..



**2.1. Development Toolchain and Process**

**Hình 2.1. Development Toolchain and Process minh họa chuỗi phát triển. Bạn có thể triển khai ứng dụng của bạn giôgs như dự án Eclipse. Dự án phải báo gồm mã nguồn của bạn, thư viện Vaadin, và có cũng bao gồm các chủ đề dự án cụ thể.**

**Bạn cần biên dịch và triển khai dự án để bộ chứa web trước khi bạn có thê sử dụng nó. Bạn có thể triển khai một dự án thông qua WTP Web Tool Plaform cho Eclipse nó dự động triển khai ứng dụng web từ Eclipse. Bạn cũng co sthể triển khai dự án bằng cách tạo file WAR (file lưu ứng dụng web – Web Application Archive) và triển khai nó trên bộ chứa web.**

**Cài đặt Java SDK**

Java SDK yêu cầu Vaadin và một số bởi Vaadin IDE. Vaadin kêt hợp với JDK 1.6 hoặc phiên bản sau. Java EE 7 được hiêu cầu server hỗ trợ Websocket.

**Window**1.Dowload J2SE từ <http://www.oracle.com/technetwork/java/javase/downloads/index.html>

2.Cài đặt Java SDK – chạy chương trình.

**Linux**

Hầu hế hệ thống Linux cài đặt lại JDK hoặc cho phép cài đặt nó thông qua hệ thống quản lý gói. Chú ý tuy nhiên điều đó, họ có OpenJDK như một thực thi Java mặc định. Trong khi nó biến đã làm việc với Vaadin và có thể cũng với một só chuỗi công cụ phát triển khác, chung ta một số đặc điểm hỗ trợ.

Về OX X, thông báo rằng JDK 1.6 hoặc mới hơn bao gồm OS X hoặc mới hơn.

1. Tải J2SE.

2. Giải nét với thưu mục con ví dụ /option.

# cd /option

# sh (duong-dan-thu-muc-cai-dat/jdk-7u1-linux-i586.bin)

3. Thiết lập sẵn sàng biến môi trường JAVA-HOME là điểm để cài đặt thư mục Java. Ngoài ra, bao gồm $JAVA\_HOME/bin trong đường dẫn Path. Làm thế nàobạn có thể bởi chó ví dụ Linux sử dụng Bash shell, bạn có thể thêm dòng chẳng hạn như dưới đây **.bashrc** hoặc **.profile** script trong thư mục của bạn.

**export JAVA\_HOME=/option/jdk1.7.0\_01**

**export PATH=$PATH:$HOME\_/bin:$JAVÂ\_HOME/bin**

Bạn có thể cung cấp tạo mọt số hệ thống trong file chẳng hạn như /etc/bash.bashrc, /etc/profile. Nếu bạn cài đặt Apache Ant hoặc Maven, bạn có thể cung cấp một số bạn muốn thiết lập trên đường dẫn.

Thiết lập hoàn thành trên file bashrc, bạn mở file cửa sổ shell. Thiết lập hoàn thành trong file profile, bạn đăng nhập vào hệ thống. Bạn có thể , dĩ nhiên cung cấp câu lệnh trong shell hiện tại.

**Cài đặt Eclipse**

**Windown**

1.Tải công cụ Eclipse cho phát triển Java EE tại : <http://www.eclipse.org/downloads/>

2.Giải nén công cụ Eclipse, Bạn có thể chọn giải phóng thư mục và sự dụng bất kỳ trình giải nén, nhưng trong ví dụ, cần giải nén cần click double vào thư mục và chọn “Giải nén all file” từ cửa sổ giải nén, trong cài đặt ví dụ : C:\\dev như một thư mục đích.

Thư mục cài đặt giờ : C:\dev\eclipse và có thể bắt đàu từ đây, click double vào eclipse.exe

**Linux/OS X/Unix**

Chúng tôi gợi ý, đó bạn cài đặt Eclipse bình thường trên Linux hoặc Unix dưới đây.

1.Tải công cụ Eclipse cho phát triển Java EE tại : <http://www.eclipse.org/downloads/>

2.Giải nén

3. Eclipse nên được cài đặt với user thông thường, như tạo mtọ cài đặt mới dễ hơn. Eclipse cũng được lưu trữ user cài đặt

**Cài đặt Apache Tomcat**

Apache Tomcat là máy chủ web Java nhẹ và cho sản phẩm và phát triên. Có nhiều cách để cài đặt nó, nhưng ở đây chung tôi trình bày cách đơn giản để cài đặt.

Apache Tomcat nên được cài đặt với quyền user. Do phát triển, bạn sẽ chạy Eclipse hoặc một số IDE khác với quyền user, nhưng triển khai ứng dụng web với máy chủ Tomcat đó được cài đạt với yêu cầu quyền quản trị hoặc quền gốc.

1.Tải thư mục cài đặt. tại <http://tomcat.apache.org/>

2. Giải nén tại thư mục chẳng hạn như C:\\Programfiles\Apache Foundation

**Firefox và firebug**

Vaadin hỗ trợ nhiều trình duyệt web và có thể sử dụng bất kỳ của chúng để phát triển. Nếu bạn lên kế hoạc để tạo chủ để cá nhân, bố cục cá nhân hóa, hoặc tạo thêm ột thành phần, chung tôi nhắc lại rằng bạn sử dụng Firefox khác với Firebug hoặc Google Chrrome, cái để xây dung công cụ đơn giản hóa để gỡ lỗi Firebug.

**Sử dụng Firebug với Vaadin**

Sau khi cài đặt Firefox, sử dụng để mở : <http://www.firebug.com/> .

Hầu hết tính năng quan trọng trong Firebug là phân tách các thẻ HTML. Click chuột phải vào lệnh hoặc mở thẻ với Firebug để phân tách. Nó sẽ hiển thị CSS, quy tắc ghép thẻ, các thẻ để bạn có thể sử dụng để xây dựng chủ đề. Bạn có thể chỉnh sửa CSS.

* 1. **Tông quan về các thư viện Vaadin**

Chào đón Vaadin như cài đặt các thư viện JARs, thư viện một số là tùy chọn và một số thay thế, và phụ thuộc vào sự phát triển cho ứng ụng kịch bản server-side hoặc client side, các thư việc bạn sử dụng để add thêm vào thành phần hoặc sử dụng CSSS hoặc Sass.

**vaadin-server-7.x.x.jar**

Đây là thư viện chính để phát triển kịch bản ứng dụng phía Server, được mô tả trong chương 4 : Viết một ứng dụng phía Server. Nó yêu cầu thư viện vaadin-shared và vaadin-themes. Bạn có thể sử dụng để xây dựng trước khi biên dịch vaadin client cho phát triển kịch bản server, mặc dù bạn cần nhiều hơn thành phần hoặc widgets cá nhân.

**vaadin-shared-7.x.x.jar**

Thư viện được chia sẻ phát triển phía Server hoặc client. Điều đó luôn là cần thiết.

**vaadin-client-7.x.x.jar**

Nền tảng Vaadin, bao gồm GWT API cơ bản và Vaadin widget và một số năng cao. Nó yêu cầu khi sử dụng vaadin\_client\_compiler để

biên dịch client\_side\_modules. Nó là cần thiết nếu bạn sử dụng nền tảng server\_side với biên dịch lại Client Side. Bạn không nên triển khai nó với ứng dụng web.

**vaadin-client-compiler-7.x.x.jar**

Trình biên dịch Vaadin Client là được biên dịch Java-to-JavaScript đó cho phép xây dựng các module client-side, chẳng hạn như Client-side yêu cầu cho ứng dụng server-side. Trình biên dịch cần thiết cho ứng dụng để biên dịch các thành phần để thiết lập các widget ứng dựng, được mô tả tại Chươgn 17 : sử dụng Vaadin add-on. BIết thêm chi tiết về trình biên dịch , xem phần 13.4 : Biên dịch module Client-side. THông báo rằng bạn không nên triển khai thư viên với một ứng dụng web.

**vaadin-client-compiled-7.x.x.jar**

Một trình biên dịch lại phía khách (widget set) nó bao gồm tất cả cách widget cơ bản để xây dựng Widget trong Vaadin. Thư viện này không cần thiết nếu ạn biên dịch widget với Vaadin Client Compiler.

**vaadin-themes-7.x.x.jar**

Vaadin tất cả các chủ đề mã nguồn file SCSS và biên dịch lại các file CSS. Thư viện đáp ứng tất cả mọi sử dụng cơ bản với CSS và trình biên dịch chủ đề Sass.

**vaadin-theme-compiler7.x.x.jar**

Trình biên dịch Chủ đề Vaadin biên dịch chủ đề SASS sang CSS, được mô tả trong phần 8.2” Systactically Awsome Stylesheet. Nó yêu cầu file jar vaadin-themes-7-xx-xj.jar. Thư viện cần thiét để phát triển và cho phép phép biên dịch THEMES, nhưng không cần để triển khai sản phẩm, khi chủ đề đã được triển khai trước đó.

Một số thư viện phụ thuộc vào như việc phụ thuộc vào sự phụ thuộc cung cấp của các thư viên trong thư mục **lib** của thư mục cài đặt, đặc biệt như **lib/vaadin-shared-deps.jar.**

Chi chú rằng **vaadin-client-compiler** và **vaadin-client** Jars không nên được triển khai trên ứng dụng web vào trong **WEB-INF/lib**. Một số các thư viẹn khác, chẳng hạn như **vaadin-them-compiler**, không cần thiết khi triển khai sản phẩm.

* 1. **Triển khai Plugin Vaadin cho Eclipse**

Nếu bạn sử dụng công cụ Eclipse, sử dụng **Plugin for Eclipse**, thông báo rằng bạn có thể tạo dự án Vaadin với Maven trong Eclipse.

Plugin bao gồm :

+ Wizards cho việc tạo dự án Vaadin cơ bản, chủ đề **Themes**, và widget client, widget set.

+ Một trình soạt thảo cho tạo các thành phần giao diện người dùng trong **SWSIWYG**. Với hỗ trợ từ mã nguồn , tương tác với quá trình sử lý phát triển.

* + 1. **Cài đặt IvyDE plugin.**

**Vaadin Plugin** cho **Eclipse** yêu cầu Apache IvyDE plugin, nó cần thiết để cài đặt trước khi sử dụng Vaadin plugin.

* + - 1. Chọn Help->Install New Software
      2. Thêm IvyDE bằng cách click vào nút “**Add”**

Nhập tên chẳng hạn như “Apache Ivy Update Site” và URL

<http://www.apache.org/dist/ant/ivyde/updatesite>

Sau đó click **OK**. Vaadin site sau đó hiển thị cửa sổ sẵn sàng.

3. Chọn mới “Apache Ivy Update Site” từ danh sách “**Work list”**

4. Chọn **Apache Ivy, Apache Ant Tasks** và **Apache IvyDe** plugin.

Tùy chọn Apache IvyDE Resolve Visualizer,

Click **Next**

5. Xem lại chi tiết cài đặt và click **Next**

6. Chấp nhận phiên bản, cuối cùng click **Finish**

7. Eclipse muốn cảnh báo thông tin Nếu bạn cảm thấy không an toàn, click **OK**.

8 . Sauk hi cài đặt plugin, Eclipse sẽ hỏi bạn Reset lại IDE. Bạn có thể xử lý cài đặt plugin trước khi restart, được mô tả theo các phần sau, nếu bạn có thể trả lời câu hỏi **Apply Changes Now**

* + 1. **Cài đặt Vaadin plugin.**

Bạn có thể cài đặt plug theo các bước :

1. Chọn Help->Install New Software
2. Thêm Vaadin bằng cách click vào nút “**Add”**

Nhập tên chẳng hạn như “Vaadin Update Site” và URL

<http://www.vaadin.com/eclipse>

Nhập tên chẳng hạn như”Vaadin Update Site” và URL cho update site,

<http://www.vaadin.com/eclipse>. Nếu bạn muốn hoặc cần sử dụng phiên bản cuối cùng thì sử dụng nhiều hơn cho phát triển và ra mắt bản Vaadin Beta. Bạn có thể sử dụng : <http://vaadin.com/eclipse/experimental> và cho tên chẳng hạn như “Vaadin Exprimental Site”. Sau đó click **OK**. Vaadin site chọn Sẵn sàn cập nhật.

3. Hiện tại, nếu bạn sử dụng plugin, Nhóm các mục trong danh sach nên được cho phép . nếu bạn sử dung experimental plugn, not nên bị giới hạn. và có thể thay đôi trong tương lai.

4. Chọn tất cả plugin Vaadin trong cây.sau đó click **Next**

5. Xem lại chi tiết cài đặt và click **Next**

6. Chấp nhận phiên bản, cuối cùng click **Finish**

7. Eclipse muốn cảnh báo thông tin Nếu bạn cảm thấy không an toàn, click **OK**.

8 . Sau khi cài đặt plugin, Eclipse sẽ hỏi bạn Reset lại IDE. Bạn có thể xử lý cài đặt plugin trước khi restart, được mô tả theo các phần sau, nếu bạn có thể trả lời câu hỏi **Apply Changes Now.**

Nếu bạn sử dụng bộ soạn thảo, Eclipse phải cho phép trình duyệt internal. Hầu hếu hệ thống giới hạn bao gồm kỹ thuật trên trình duyệt, nhưng nếu không phải, bạn có thể cần để cài đặt nó có trong mô tả trong Chương 7 **: Thiết kế giao diện người dùng với Eclipse.**

Nhiều thông tin cài đặt được giới thiệu tại <http://vaadin.com/eclipse>

* + 1. **Cập nhật Plugins.**

Nếu bạn tự động cho phép Eclipse update ( xem Windown -> References-> Instal/Update –Automatic Update), Vaadin sẽ tự động update một số plugin khác, ngoài ra bạn có thể update plugin theo cách thông thường như sau.

1.CHọn **Help-> Check for Update**. Eclipse sẽ liên lạc site update để cài đặt phầm mềm.

2. Sau khi update đưuọc cài đặt, Eclipse sẽ hỏi bạn Reset lại. Click **Restart**.

* + 1. **Cập nhật thư viện Vaadin.**

Cập nhật Plugin Vaadin không cập nhật thư viện Vaadin. Thư viện Vaadin đặc biệt trong dự án, giống như phiên bản khác nhau có thể đáp ứng các dự án khác nhau, vì thế bạn đã cập nhật chung khác nhau tại mỗi project.

1. Mở ivy.xml bằng trình soạn thảo Eclipse
2. Sử entity tại dòng đầu và thiết lập phiên bản Vaadin

<!ENTITY vaadin.version "**7.0.1**">

Bạn có thể sử số phiên bản, giống như việc ví dụ trên hoặc xem lại thông tin phiên bản cuối.Bạn có thể tìm nhiều thông tin mô tả sự phụ thuộc trong tài liệu IVY.

1. Click chuột phải vào dự án chọn IVY –**Resolve**

Cập nhật thư viện có thể mất vài phút, Bnạ có thể thấy trình sử lý trên thanh trạng thai Eclipse. Bạn có thể lấy nhiều thông tin bằng cách click vào indicator.

1. Nếu bạn đã biên dịch widget cho dự án của bạn, biên dịch lại nó bằng cách click vào “ **Compiler Vaadin Widget**”.
2. Dừng tương tác Tomcat hoặc Server khác trên Eclipse, xóa bộ nhớ đệm bằng chách clicks chuột phải vào Server chọn **Clean**  như một lựa chọn khác **Clean Tomcat Directory** và restart lại.

Nếu bạn có nhiều kinh nghiệm về các vấn đề sau khi cập trinhậ thư viện, bạn có thể thử làm sạch bộ nhớ đêm Ivy bằng cách click chuột phải vào project chọn “**Ivy** – **Clearn all cache**. Sau đó , chọn tiếp **Ivy**-**Resovle** và làm lại các tác vụ khác.

* 1. Tạo mới và chạy dự án với Eclipse

Trong phần này cung cấp giới thiệu cho taoọ mới dự án Eclipse sử dụng Plugin Vaadin. Các bước bao gồm :

1. Tạo mới project
2. Viết mã nguồn
3. Cấu hình và start Tomcat ( hoặc web server khác)
4. Mở mọt trình duyệt web để sử dụng ứng dụng web.

Chúng ta cung hiển thi làm thế nào bạn cso thể gỡ lỗi ứng dụng trong chế độ gỡ lỗi của Eclipse.

Đây bước tóm tắt rằng bạn đã sẵn sàng cài đặt Vaadin plugin cho Eclipse và thiếu lập môi trường cho phát triển, như đã giứoi thiệu trong phần 2.2 : **Thiết lập môi trường phát triển.**

* + 1. **Tạo mới Project**

Bắt đầu tạo project application đầu tiện bạn cần làm các phần đầu tiên. Mở Eclipse và làm theo các bước sau:

1. Bắt đàu tạo mới project chọn **File – New – Project**.

2. Trong **New Project**  được mở, chọn **Web – Vaadin 7 Project** và click **Next**.

3. Trong bước tiếp theo Vaadin Project, bạn cần thiết lập thông tin cở bản về dự án. Bạn cần cung cấp Project Name và thông tin ứng dụng, một số thông tin được mặc định.

**Project name**

Cung cấp tên project, tên nên là tên không trùng lặp được sử dụng như tên file và URL, chỉ sử dụng ký tự thường, dấu gạch dưới và mọt số ký tự nhắc nhở.

**User default location**

Định nghĩa thư mục dưới khi project được tạo. Mặc định là thư mục namespace và nếu bạn có thể thay thế với tên khác. Bạnc ó thể cần thiết lập thư mục cho ví dụ, nếu bạn tạo mới Project trên thư mục cây điều khiển.

**Taget runtime**

Định nghĩa máy chủ ứng dụng để sử dụng để triển khai ứng dụng. Máy chủ đó bạn đã được cài đặt ví dụ Apache Tomcat, ncó thể được chọn một cách tự động. Nếu không, clicks **New** để cấu hình một **Server** dưới Eclipse.

**Configuration**

Chọn cấu hình để sử dụng, bạn có thể sử dụng mặt định thông thượng cho máy chủ ứng dụng. Nếu bạn cần thay đổi thông tin dự án click **Modify**. Gợi ý Servlet 3.0 cấu hình sử dụng **@WebServlet** triển khai, trong khi Servlet 2.4 sử dụng **web.xml** để triển khai.

**Deployment configuration**

Đây lầ phần định nghĩa môi trường cho việc ứng dụng có thể được triển khai, để tạo ra ứng dụng và file cấu hình. Chọn

+ **Servlet** mặc định

+ **Google App Engine Servlet.**

**+ Genneric Porlet (Portlet 2.0).**

Nhiều bước hơn trong tạo một project mới phụ thuộ c vào sự lựa chọn cấu hình triển khai , các bước được xử lý trong phần này và mặc định cấu hình Serverlet, Xem phần 11.7 :Tương tác **Google App Engine** và Chương 12 **: Tương tác cổng thông tin Portlet**  trong phần giới thiệu của Vaadin và trong thay thế môi trường.

**Vaadin version**

Chọn phiên bản sử dụng Vaadin. Trong menu hiển thị , bằng mặc định hoặc chọn các phiên bản Vaadin đã sẵn sàng. Chọn bao gồm **SNAPSHOOT** , nếu bạn muốn giữ nó liên bản mới nhất, bạn có thể thay đổi phiên bản tại **ivy.xml.**

4. Thiết lập **Web Module** các bước cơ bản để triển khai ứng dụng web, thiết lập cấu truc ứng dụng web. Tất cả ác thiết lập được thiết lập đầy đủ và bạn có thể chấp nhận chúng như một số mặc định.

**Context Root**

Context root được định nghĩa trong ứng dụng trong URL được sử dụng để truy cập nó. Ví dụ nếu dự án là **myproject** và một UI đơn tại context root, URL có thể là : <http://example.com/myproject>. Điều đó sẽ đồng rằng tên cung câp trong bước như contextname. Bạn có thể chay đổi nó trong thuộc tính của dự án Project.

**Context Directory**

Thư mục chưa tất cả các nôi dung để bao gôm trong ứng dụng web (**WAR)** điều đó được triển khai trên máy chủ web. Thu mục liên quan đến thư mục project

5. **Vaadin project** các bước đặc biệt thiết lập ứng dụng. Nếu bạn thử Vaadin lần đàu, bạn không nên thay đổi bất kỳ thứ gì. Nếu có thể thiết lập hầu hết thông tin, chấp nhận tạo Portlet.

**Create project template**

Tạo với class UI

**Application Name**

Một tên cho ứng dụng UI, hiển thị trên tiêu đề cửa sổ trình duyệt

**Base Package name**

Tên của thư mục Java dưới, chưa các class của ứng dụng và nó có thể được thay thế.

**Application/UI class name**

Tên của class UI cho ứng dụng, khi giao diện được triển khai.

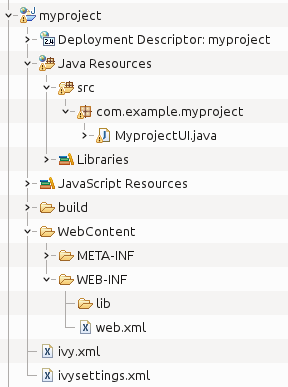
**Portlet Version**

Khi một phiên bản Portlet được lựa chọn (Portlett 2.0 được hỗ trợ), tạo mới các file cần thiết để chạy các ứng dụng portal, Xem **Chương 12. Tích hợp Portal**

Cuối cùng click Finish để tạo project.

* + 1. **Mở rộng Project**

Sau khi **New Project** đã được tạo, nó đã được chọn tất cả như : một class UI đã được tạo trong thư mục **src** và **WebContent/WEB-INF/web.xml** chứa mô tả triển khai. Cấu trúc cây Project.



Thư viện Vaadin hoặc sự phụ thuộc được quản lý bởi **ivy.xml**. Thông báo rằng các thư viện không lưu trữ thư mục dưới project, **Java Resources-Libraries –ivy.xml**

**UI Class**

package com.example.myproject;

import com.vaadin.ui.UI;

...

@SuppressWarnings("serial")

@Theme("myproject")

public class MyprojectUI extends UI {

    @WebServlet(value = "/\*", asyncSupported = true)

    @VaadinServletConfiguration(

            productionMode = false,

            ui = MyprojectUI.class)

    public static class Servlet extends VaadinServlet {

    }

    @Override

    protected void init(VaadinRequest request) {

        final VerticalLayout layout = new VerticalLayout();

        layout.setMargin(true);

        setContent(layout);

        Button button = new Button("Click Me");

        button.addClickListener(new Button.ClickListener() {

            public void buttonClick(ClickEvent event) {

                layout.addComponent(

                    new Label("Thank you for clicking"));

            }

        });

        layout.addComponent(button);

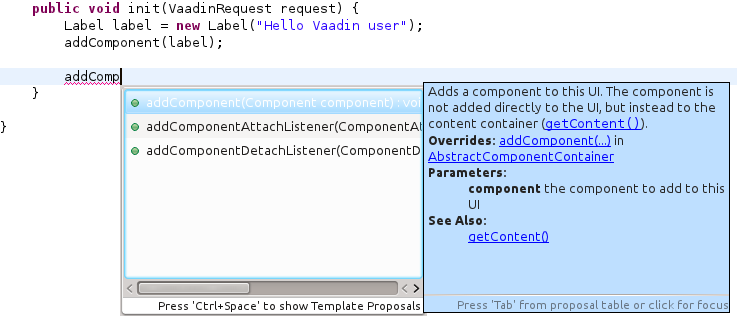
    }  
}

Trong project Servlet 3.0, triển khai phai cấu hình với class servlet và ký tự **@WebServlet**. Bao gồm các class Servlet như một class static lồng. Bạn có thể muốn **Reflactor** nó để phân cách các class thông thường.

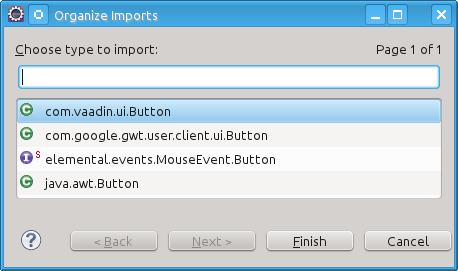
Trong Servlet 3.0, bạn có file mô tả triển khai **web.xml**.

* + 1. **Mẹo code cho Eclipse**

Hầu hết sử dụng tính năng trong Eclipse là hoàn thiện mã (code completion). Ấn **Ctrl+Space** trong trình soạn thảo sẽ hiển thị danh sách popup có thể là class name, method name như hình 2.4. Java Code Completion in Eclipse



Để **import** câu lệnh cho class, chẳng hạn như **Button,** đơn giản ấn **Ctrl+Shift+O** hoặc click vào báo lỗi đỏ ở bên trái trình soạn thảo. Nếu class sẵn sàng có trong các thư mục, một danh sách thay thể được hiển thị như hình 2.5 : 2.5. Importing Classes Automatically



* + 1. **Thiết lập và Start với Websserver**

Công cụ Eclipse cho Java Enterprice Editor có thư mục công cụ web chuẩn được cài đặt, hỗ trợ điều khiển máy chủ web và tự động triển khai nội dung web để máy chủ khi thay đổi trên project đã tạo.

Hãy chắc chắn rằng Tomcat đã được cài đặt với quyền user. Cấu hình máy chủ web trong Eclipse sẽ thất bại nếu người dung không có quyền cấu hình và triển khai thư mục dưới thư mục cài đặt TOMCAT.

Theo dõi các bước dưới đây:

1. Thay đổi tab **Servers**  ở phía thấp Eclipse. Danh sách server có thể trống sau khi cài đặt Eclipse. Click phải vào khoảng trống và chọn **New –Server.**

2. Chọn  **Apache->Tomcat v7 Server** và Server’s host name as localhost, có thể đưuọc để mặc định. Nếu bạn chỉ cài đặt Tomcat, **Server runtime**  có duy nhất một lựa chọn, click **Next.**

3. Để thêm Project vào server bằng cách chọn nó từ bên trái và click Add để thêm nó vào cấu hình project bên phải. Click **Finish**

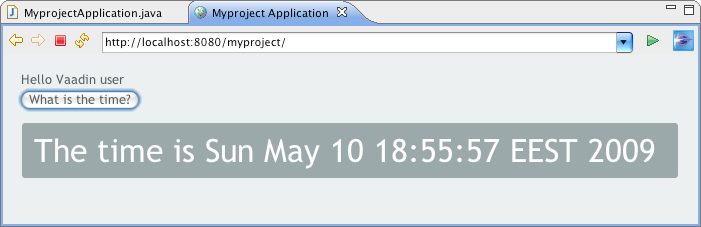
**4. Server** và project được cài đặt trong Eclipse và được hiển thị trong tab Server, Để bắt đầu Server, Click phải vào **Server** chọn **Debug**. Để bắt đầu server chọn **Start.**

5.Máy chủ đã được start với thư mục Webcontent xuất bản :

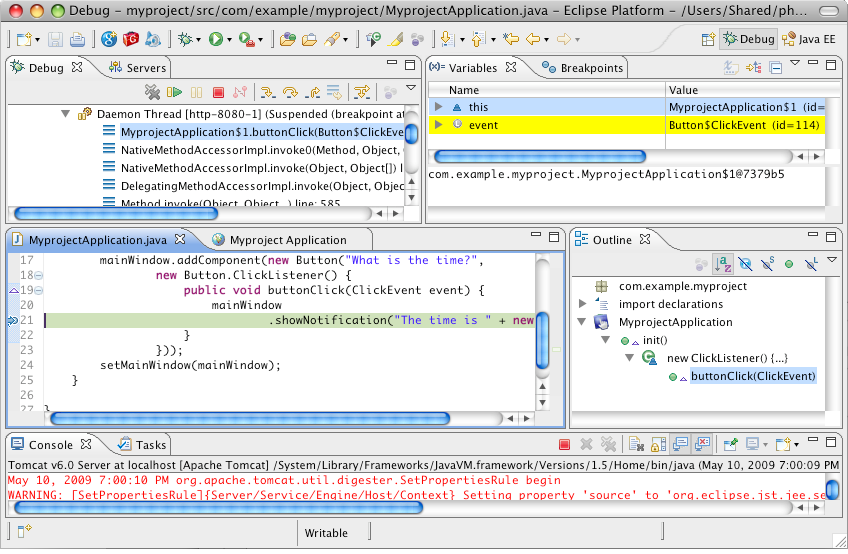
http://localhost:8080/myproject/.

* + 1. **Chạy và gỡ lỗi**

Bắt đầu ứng dụng của bạn như một sự lựa chọn dễ dàng **myproject** từ **Project Explorer** và **Run->Debug As - >Debug on Server.** Eclipse sau đó mở ứng dụng xây dựng nó trên trình duyệt

****

Bạn có thê chèn một điểm ngắt (“**Break point**) trong code java bằng cách click đúp vào bên trái cửa sổ mã nguồn. Cho ví dụ, nếu bạn chèn breakpoint trong phương thức buttonclick() và click “**What is the time?** Button ? Eclipse sẽ hỏi trước hợp Debug. Debug sẽ hiển thị thực thi ở đâu sẽ dừng ở điểmm breackpoint đó. Bạn có thể kiêm tra các trạng thái thay đổi ưng sdụng để tiếp tục chọn **Resume** từ **menu Run.**



Trên tất cả, chúng tôi đã mô tả làm thế nào để gỡ lỗi ứng dụng phía Server, Gỡ lỗi ứng dụng phía client và widget được mô tả trong phần **13.6 Gỡ lỗi Code phía Client.**

* 1. Sử dụng Vaadin với Maven

**Maven** được sử dụng để quản lý sự phụ thuộc của hệ thống. Nền tảng thư viện Vaadin và tất cả các tiện ích add-ons được sẵn sàng thông qua **Maven.** Bạn có thể sử dụng **Maven** trong Eclipse hoặc Netbeans hoặc sử dụng các dòng lệnh commands line như đã mô tả trong phần này.

Ngoài ra,Maven bạn có thể sử dụng bất kỳ xây dựng tương thích hợp hệ thống quản ly sự phụ thuộc nào, chẳng hạn như Ivy hoặc Gradle. Cho Gradle xem Gradle Vaadin Plugin. Vaadin Plugin cho Eclipse sử dụng Ivy cho giải quyết sự phụ thuộc trên dự án Vaadin, và nó nên cung câp bạn với cấu hình Ivy cơ bản.

**2.6.1. Làm việc với Command line**

Bạn có thể tạo mới project Maven với các dòng câu lệnh dưới đây:

$ **mvn** archetype:generate

-DarchetypeGroupId=com.vaadin

-DarchetypeArtifactId=*vaadin-archetype-application*

-DarchetypeVersion=*7.x.x*

-DgroupId=*your.company*

-DartifactId=*project-name*

-Dversion=*1.0*

-Dpackaging=war

Các tham số :

**-archetypeGroupId=com.vaadin : : Nhóm Id là** archetypes

**-DarchetypeArtifactId=*vaadin-archetype-application***

**Là** archetype Id , Vaadin 7 hỗ trựo **vaadin-archetype-application** archetype cho ứng dụng phía server, và **vaadin-archetype-widget** cho ứng dụng phía client.

**-DarchetypeVersion=*7.x.x***

***Phiên bản Archetype sử dụng. nên là phiên bản Vaadin mới nhất.(7.0.0.beta3)***

**-DgroupId=*your.company***

***Maven group Id, nó được sử dụng cho nên gói Java và thông thường là tên domain, chẳng hạn như com.example.myproject. GoupId cũng được sử dụng cho tên thư mục mã nguồn Java.***

**-DartifactId=*project-name***

***Được định nghĩa của artifacet, đó là project của bnạ được đinh nghĩa chưa các ký tự cơ bản.***

**-Dversion=*1.0***

***Tên phiên bản ứng dụng.***

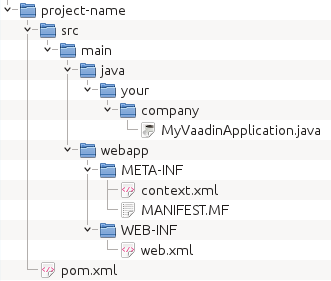
**-Dpackaging=war**

**Định dạng dự án sẽ được đóng gói ? War : Web Application Archive**

**2.6.2. Biên dịch và chạy ứng dụng**

Tạo mới project cần một s

Hình **2.8. A New Vaadin Project with Maven**



Trước khi ứng dụng có thể được triển khai, nó phải được biên dịch thành và đóng gọi thành file WAR, bạn có thể làm điều này với đích **package**

$mvn package

Thư mục cục bộ trả về kết quả đóng gói của Warr được hiển thị trong câu lệnh thông báo. Bạn có thể triển khai nó lên ứng dụng server.

Dễ dàng nhất để chạy Vaadin application với Maven là sử dụng máy chủ web nhẹ Jetty, sau khi biên dịch và đóng gói, tất cả bạn cần làm là   
$mvn jett:run

Đặc điểm chính bắt đầu máy chủ Jetty trong cổng 8080 và triển khai ứng dụng , bạn có thể mở nó trong trình duyệt web với đường dẫn “: http://localhost:8080/project-name

**2.6.3 Sử dụng tiện ích add-ons và Custom Widget Set**

Nếu bạn sư dụng tiện íh Vaadin bao gồm Widget hoặc tạo Widget cá nhân, bạn cần cho phép Wideget thiết lập cả khả năng biên dịch trong **POM .** Yêu cầu câu hình mô tả trong phần 17.4 **Sử dụng tiện ích –add-** trong Maven Project

* 1. Tạo mới Project với Netbean IDE
  2. Tạo mới Project với Intellij IDEA
  3. Thư mục triển khai Vaadin
  4. Sử dụng Vaadin với Scala

1. **Kiến trúc**
   1. Tổng quan
   2. Nền tảng công nghệ
   3. Kỹ thuật phía client
   4. Sự kiện và nắm bắt sự kiện
2. **Nền tảng phía Server**
3. **Viết một ứng dụng Web phía Server**
   1. Tổng quan
   2. Xây dựng giao diện người dùng
   3. Xử lý sự kiện với nắm bắt sự kiện
   4. Hình ảnh và tài nguyên khác
   5. Quản lý lỗi
   6. Thông báo
   7. Vòng đời ứng dụng
   8. Triển khai một ứng dụng
4. **Thành phần giao diện người dùng**
   1. Tổng quan
   2. Interface và abstractions
   3. Các thành phần phố biến
   4. Các Kiểu thành phần
   5. Nhãn
   6. Đường dẫn
   7. Textfield
   8. TextArea
   9. PasswordField
   10. RichTextArea
   11. Date và Nhập thời gian với DateField
   12. Button
   13. Checkbox
   14. Selection Components
   15. Bảng
   16. Tree
   17. Menu Bar
   18. Nhúng dữ liệu (Embedded Resouce)
   19. Tải dữ liệu
   20. ProgressBar
   21. Slider
   22. Calender
   23. Thành phần kết hợp với thành phần cá nhân
   24. Tổng hợp các trường với thành phần cá nhân

1. **Quản lý giao diện**
   1. Tổng quan
   2. Giao diện người dùng, UI, Windown và nội dung Pannel
   3. Bố cục ngang , bố cục dọc
   4. Bố cục ô lưới
   5. Bố cục Form
   6. Pannel
   7. Window con
   8. HorizontalSplitPannel và VerticalSplitPannel
   9. Tabsheet
   10. Accordion
   11. AbsoluteLayout
   12. CssLayout
   13. Định dạng Layout
   14. Tùy chỉnh Layout
2. **Thiết kế giao diện ảo với Eclipse**
   1. Tổng quan
   2. Tạo một hỗn hợp mới
   3. Sử dụng Visual Editor
   4. Cấu trúc của một thành phần Visual
3. **Chủ đề**
   1. Tổng quan
   2. Giới thiệu Casecading Style Sheet
   3. Style Sheet cú pháp tuyệt vời
   4. Tạo và sử dụng Themes
   5. Tạo một chủ đề trong Eclipse
   6. Tạo một chủ đề trong Eclipse
   7. Chủ đề Valo
   8. Tùy chỉnh Font
   9. Đáp ứng Theme
4. **Thành phần kết hợp với Data**
   1. Tổng quan
   2. Thuộc tính
   3. Lưu giữ liệu trong Items
   4. Tạo hình thức ràng buộc giữa các control
   5. Tập hợp dữ liệu trong Containner
5. **Vaadin SQLcontainner**
   1. Kiến trúc
   2. Bắt đầu với SQLContainner
   3. Lọc và sắp xếp
   4. Chỉnh sửa
   5. Bộ nhớ đệm cache, Phân trang, và Refreshing
   6. Sử dụng SQLContainer khác
   7. Sử dụng FreeformQuery và FreeformStatêmntDelegate
   8. Phương thức triển khai thưucj hiện gaio diện không chứa Vaadin
   9. Các vấn đề và hạn chế được biết đến
6. **Nâng cáo chủ đề ứng dựng Web**
   1. Xử lý Windowns Brower
   2. Nhung các UI vào trang web
   3. Chế độ gỡ lỗi và Window
   4. Bộ xử lý yêu cầu
   5. Các phím tắt
   6. In
   7. Tích hợp Google App Engine
   8. Vấn đề an ninh
   9. Điều hướng trong ứng dụng
   10. Kiến trúc ứng dụng nâng cao
   11. Quản lý các mảnh URI
   12. Kéo và thả
   13. Log
   14. Tương tác JS
   15. Truy cập Session – Và dữ liệu toàn cục
   16. Máy chủ đẩy
   17. Các kỹ tự chữ số
7. **Tích hợp cổng thông tin**
   1. Tổng quan
   2. Tạo một dự án cổng thôngtin trong eclipse
   3. Giao diện người dùng Portlet
   4. Triển khai một Portlet
   5. Cài đặt Vaaddin trong Liferay
   6. Xử lý yêu cầu trong Portlet
   7. Xử lý portlet thay đổi chế độ
   8. Chế độ Portlet không Vaadin
   9. Vaadin IPC cho Liferay
8. **Nền tảng phía Client**
9. **Phát triển Vaadin phía khách**
   1. Tổng quan
   2. Cài đặt môi trường phía khách
   3. Biên dịch module Client-side
   4. Tạo mới một Widget cá nhân
   5. Gỡ lỗi mã client-side
10. **Ứng dụng phía khách**
    1. Tổng quan
    2. Module Client-side
    3. Biên dịch và chạy một ứng dụng client-side
    4. Tải một ứng dụng client-side
11. Widget phía khách
    1. Tổng quan
    2. Google Web Toolkit
    3. Vaadin Widgets
12. Tương tác phía Server
    1. Tổng quan
    2. Bắt đầu từ sự đơn giản với Eclipse
    3. Tạo một thành phần serverside
    4. Lồng ghép 2 side với một Connnector
    5. Shared State
    6. RPC call giữa Client và Serverside
    7. Các thành phần và UI mở rộng
    8. Kiểu cho Widget
    9. Các thành phần chứa
    10. Các chủ đề Client-Side nâng cao
    11. Tạo mới Add-ons
    12. Chuyển từ Vaadin 6
    13. Tích hợp cá thành phần JS Và mở rộng
13. **Tiện ích Vaadin add-on**
14. **Sử dụng Vaadin Add-ons**
    1. Tổng quan
    2. Tải các add-on từ thư mục Vaadin
    3. Cài đặt add-on trong Eclipse với Ivy
    4. Sử dụng add-on trong một dự án Maven
    5. Xử lý sự cố - troubleshooting
15. **Các biểu đồ Vaadin**
    1. Tổng quan
    2. Cài đặt Biểu đồ Vaadin
    3. Sử dụng cơ bản
    4. Các kiểu Biểu đồ
    5. Cấu hình Biểu đồ
    6. Dữ liệu Biểu đồ
    7. Sử dụng nâng cao
    8. Timeline
16. **Vaadin JPA Containner**
    1. Tổng quan
    2. Cài đặt/ triển khai
    3. Xác định mô hình Domain
    4. Sử dụng cơ bản cả JPAContainner
    5. Cung cấp Entity
    6. Lọc JPAContainner
    7. Truy vấn với các Critaria API
    8. Thế hệ mẫu tự động
    9. Sử dụng JPAContainner với Hibernate
17. **Ứng dụng di động với TouchKit**
    1. Tổng quan
    2. Xem xet về Trình duyệt trên điện thoại
    3. Cài đặt Vaadin Touchkit
    4. Import các Mornitologist Demo
    5. Tạo một dự án mới Touchkit
    6. Các yếu tố của một ứng dụng TouchKit
    7. Các thành phần giao diện người dùng
    8. Tính năng nâng cáo điện thoại
    9. Chế độ ẩn
    10. Xây dựng một Optimized Widget
    11. Kiểm tra và gỡ lỗi trên thiết bị Di động
18. **Vaadin Testbench**
    1. Tổng quan
    2. Cài đặt Vaadin TestBench
    3. Chuẩn bị một ứng dụng cho Testing
    4. Phát triển JUnit Test
    5. Tạo một TestCase
    6. Truy vấn các yếu tố
    7. Chọn lọc nhiều yếu tố
    8. Kiểm tra các tính năng đặc biệt
    9. Tạo bảo trì test
    10. Chụp và so sánh màn hình
    11. Chạy thử nghiệm
    12. Kiểm tra không đồng
    13. Các vẫn đề
19. **Giới thiệu**
20. **Giới thiệu**
    1. Tổng quan
    2. Ví dụ ứng dụng
    3. Hỗ trợ Eclipse IDE
    4. Mục tiêu và lẽ sống
    5. Bối cảnh ra đời
21. **Bắt đầu với Vaadin**
    1. Tổng quan
    2. Cài đặt môi trường phát triển
    3. Tông quang về các thư viện Vaadin
    4. Triển khai Plugin Vaadin cho Eclipse
    5. Tạo mới và chạy dự án với Eclipse
    6. Sử dụng Vaadin với Maven
    7. Tạo mới Project với Netbean IDE
    8. Tạo mới Project với Intellij IDEA
    9. Thư mục triển khai Vaadin
    10. Sử dụng Vaadin với Scala
22. **Kiến trúc**
    1. Tổng quan
    2. Nền tảng công nghệ
    3. Kỹ thuật phía client
    4. Sự kiện và nắm bắt sự kiện
23. **Nền tảng phía Server**
24. **Viết một ứng dụng Web phía Server**
    1. Tổng quan
    2. Xây dựng giao diện người dùng
    3. Xử lý sự kiện với nắm bắt sự kiện
    4. Hình ảnh và tài nguyên khác
    5. Quản lý lỗi
    6. Thông báo
    7. Vòng đời ứng dụng
    8. Triển khai một ứng dụng
25. **Thành phần giao diện người dùng**
    1. Tổng quan
    2. Interface và abstractions
    3. Các thành phần phố biến
    4. Các Kiểu thành phần
    5. Nhãn
    6. Đường dẫn
    7. Textfield
    8. TextArea
    9. PasswordField
    10. RichTextArea
    11. Date và Nhập thời gian với DateField
    12. Button
    13. Checkbox
    14. Selection Components
    15. Bảng
    16. Tree
    17. Menu Bar
    18. Nhúng dữ liệu (Embedded Resouce)
    19. Tải dữ liệu
    20. ProgressBar
    21. Slider
    22. Calender
    23. Thành phần kết hợp với thành phần cá nhân
    24. Tổng hợp cá trường với thành phần cá nhân
26. **Quản lý giao diện**
    1. Tổng quan
    2. Giao diện người dùng, UI, Windown và nội dung Pannel
    3. Bố cục ngang , bố cục dọc
    4. Bố cục ô lưới
    5. Bố cục Form
    6. Pannel
    7. Window con
    8. HorizontalSplitPannel và VerticalSplitPannel
    9. Tabsheet
    10. Accordion
    11. AbsoluteLayout
    12. CssLayout
    13. Định dạng Layout
    14. Tùy chỉnh Layout
27. **Thiết kế giao diện ảo với Eclipse**
    1. Tổng quan
    2. Tạo một hỗn hợp mới
    3. Sử dụng Visual Editor
    4. Cấu trúc của một thành phần Visual
28. **Chủ đề**
    1. Tổng quan
    2. Giới thiệu Casecading Style Sheet
    3. Style Sheet cú pháp tuyệt vời
    4. Tạo và sử dụng Themes
    5. Tạo một chủ đề trong Eclipse
    6. Tạo một chủ đề trong Eclipse
    7. Chủ đề Valo
    8. Tùy chỉnh Font
    9. Đáp ứng Theme
29. **Thành phần kết hợp với Data**
    1. Tổng quan
    2. Thuộc tính
    3. Lưu giữ liệu trong Items
    4. Tạo hình thức ràng buộc giữa các control
    5. Tập hợp dữ liệu trong Containner
30. **Vaadin SQLcontainner**
    1. Kiến trúc
    2. Bắt đầu với SQLContainner
    3. Lọc và sắp xếp
    4. Chỉnh sửa
    5. Bộ nhớ đệm cache, Phân trang, và Refreshing
    6. Sử dụng SQLContainer khác
    7. Sử dụng FreeformQuery và FreeformStatêmntDelegate
    8. Phương thức triển khai thưucj hiện gaio diện không chứa Vaadin
    9. Các vấn đề và hạn chế được biết đến
31. **Nâng cáo chủ đề ứng dựng Web**
    1. Xử lý Windowns Brower
    2. Nhung các UI vào trang web
    3. Chế độ gỡ lỗi và Window
    4. Bộ xử lý yêu cầu
    5. Các phím tắt
    6. In
    7. Tích hợp Google App Engine
    8. Vấn đề an ninh
    9. Điều hướng trong ứng dụng
    10. Kiến trúc ứng dụng nâng cao
    11. Quản lý các mảnh URI
    12. Kéo và thả
    13. Log
    14. Tương tác JS
    15. Truy cập Session – Và dữ liệu toàn cục
    16. Máy chủ đẩy
    17. Các kỹ tự chữ số
32. **Tích hợp cổng thông tin**
    1. Tổng quan
    2. Tạo một dự án cổng thôngtin trong eclipse
    3. Giao diện người dùng Portlet
    4. Triển khai một Portlet
    5. Cài đặt Vaaddin trong Liferay
    6. Xử lý yêu cầu trong Portlet
    7. Xử lý portlet thay đổi chế độ
    8. Chế độ Portlet không Vaadin
    9. Vaadin IPC cho Liferay
33. **Nền tảng phía Client**
34. **Phát triển Vaadin phía khách**
    1. Tổng quan
    2. Cài đặt môi trường phía khách
    3. Biên dịch module Client-side
    4. Tạo mới một Widget cá nhân
    5. Gỡ lỗi mã client-side
35. **Ứng dụng phía khách**
    1. Tổng quan
    2. Module Client-side
    3. Biên dịch và chạy một ứng dụng client-side
    4. Tải một ứng dụng client-side
36. Widget phía khách
    1. Tổng quan
    2. Google Web Toolkit
    3. Vaadin Widgets
37. Tương tác phía Server
    1. Tổng quan
    2. Bắt đầu từ sự đơn giản với Eclipse
    3. Tạo một thành phần serverside
    4. Lồng ghép 2 side với một Connnector
    5. Shared State
    6. RPC call giữa Client và Serverside
    7. Các thành phần và UI mở rộng
    8. Kiểu cho Widget
    9. Các thành phần chứa
    10. Các chủ đề Client-Side nâng cao
    11. Tạo mới Add-ons
    12. Chuyển từ Vaadin 6
    13. Tích hợp cá thành phần JS Và mở rộng
38. **Tiện ích Vaadin add-on**
39. Sử dụng Vaadin Add-ons
    1. Tổng quan
    2. Tải các add-on từ thư mục Vaadin
    3. Cài đặt add-on trong Eclipse với Ivy
    4. Sử dụng add-on trong một dự án Maven
    5. Xử lý sự cố - troubleshooting
40. Các biểu đồ Vaadin
    1. Tổng quan
    2. Cài đặt Biểu đồ Vaadin
    3. Sử dụng cơ bản
    4. Các kiểu Biểu đồ
    5. Cấu hình Biểu đồ
    6. Dữ liệu Biểu đồ
    7. Sử dụng nâng cao
    8. Timeline
41. Vaadin JPA Containner
    1. Tổng quan
    2. Cài đặt/ triển khai
    3. Xác định mô hình Domain
    4. Sử dụng cơ bản cả JPAContainner
    5. Cung cấp Entity
    6. Lọc JPAContainner
    7. Truy vấn với các Critaria API
    8. Thế hệ mẫu tự động
    9. Sử dụng JPAContainner với Hibernate
42. Ứng dụng di động với TouchKit
    1. Tổng quan
    2. Xem xet về Trình duyệt trên điện thoại
    3. Cài đặt Vaadin Touchkit
    4. Import các Mornitologist Demo
    5. Tạo một dự án mới Touchkit
    6. Các yếu tố của một ứng dụng TouchKit
    7. Các thành phần giao diện người dùng
    8. Tính năng nâng cáo điện thoại
    9. Chế độ ẩn
    10. Xây dựng một Optimized Widget
    11. Kiểm tra và gỡ lỗi trên thiết bị Di động
43. Vaadin Testbench
    1. Tổng quan
    2. Cài đặt Vaadin TestBench
    3. Chuẩn bị một ứng dụng cho Testing
    4. Phát triển JUnit Test
    5. Tạo một TestCase
    6. Truy vấn các yếu tố
    7. Chọn lọc nhiều yếu tố
    8. Kiểm tra các tính năng đặc biệt
    9. Tạo bảo trì test
    10. Chụp và so sánh màn hình
    11. Chạy thử nghiệm
    12. Kiểm tra không đồng
    13. Các vẫn đề