

Vaadin }>

Vaadin is an open source Java Web application framework for rich Internet applications. In contrast to JavaScript libraries and browser-plugin based solutions, it features a server-side architecture, which means that the majority of the logic runs on the servers. Ajax technology is used at the browser-side to ensure a rich and interactive user experience. On client-side Vaadin is built on top of and can be extended with Google Web Toolkit.

Best Practices }>

Vaadin is a pure MVC based framework. The nice thing about Vaadin is that it uses a lot of web technologies, yet on the other hand it does not bother the developer with all those web technologies it uses like XHTML, CSS, javascript, AJAX. In simple words, the developer of using Vaadin can develop the application with only Java knowledge; However, knowledge about those technologies can be used since not only can a developer easily customize the UI components using CSS or whatever web technology he wishes, but he can also effortlessly add new UI components. One of the strongest points about Vaadin is its Server-side state management. The application code mainly runs on the server, which makes applications much more secure and harder to break into, while adding unnoticeable delay. Vaadin also supports Mobile Touch events. To sum up, developing a minimal good looking Vaadin application is no more than writing Java and dragging and dropping for GUI components and can be extended to anything the developer wants to do.

Authentication, Authorizaiton and Security }>

Vaadin supplies the developers with techniques for authentication since it's used in 99% of the applications. Developers can find [here](#) how to perfectly deal with authentication when using Vaadin. The link above also discusses how to get over URL hacking, if user will start application by invoking a particular window URL instead of main application URL? That's pretty easy - simply postpone creation of all but Login window until the authentication is done. The link also shows code snippets of a simple application and with authentication.

Considering the authorization, there exists an authorization module in Vaadin called the Role Based Access Control, usually it's something like a visitor (someone with no authentication), Customer/User (someone with authentication), Administrator, Special User

(Customer with more permissions than usual customers, this role is assigned by the admin).the RBAC module provides 3 interfaces, PermissionManager, Role, Resource. PermissionManager is an object where you define if a role is granted or denied the access to a resource.Role is used in cases when you directly assign permissions to the user. Resource is application specific, it can be something like the views or data. This [link](#) also discusses the authorization of Vaadin in more details and supplies code snippets about the RBAC module and how to use it and implement the interfaces. For more information, developers can check the [API](#) page of Vaadin, or the official Vaadin [book](#).

Having most of the work done on the server-side, this makes a Vaadin application having a strong security. Also Vaadin supports Secure parameter and request validation and contains a built-in input validation framework, which prevents a lot of different website injection attacks, most importantly SQL injection, and also saves a lot of time and coding effort. Vaadin is also secure against CSRT attacks. It also supports the SSL protocol, which encrypts the data on it's way between the server and client. There have also been noticeably very low number of security vulnerabilities in Vaadin since it's first release and they were all fixed. Of course security issues can always arise from bad coding or bad application design so be very careful. With the correct tools and a good authentication and authorization it is possible to build a secure application.

Documentation }>

The Vaadin framework is very well documented and encouraging for developers to start using it. The Vaadin book, on chapter 2, contains a detailed tutorial about "getting started" explaining where to download the required software and how to set them up, and start a project and it also contains a video tutorial that can be checked [here](#). Documentation is often updated and improved by the community in an attempt to reach the best possible documentation. There, also, exists an official Vaadin guide book [here](#), which is a great guide for first time Vaadin users. Vaadin also provides its developers with a good documentation for the Vaadin API , check it [here](#) considering all classes and packages that is descriptive. The API is put in a very similar manner to that of the famous Java Docs.

Society }>

The Vaadin society is not the biggest society out there; however, the Vaadin society is very active and helpful. The latest version of Vaadin has already been downloaded more than 15,000 times, which reveals the number of users ONLY for the latest version. Active official forum can be found [here](#); people on forum looks friendly and willing to help as much as they can

and you usually get fast responses, actually it is only a matter of days. There following statistics about the forum taken on Thursday, March 08 2012

of Categories: 11

of Posts: 34,645

of Participants: 3,366

The Vaadin community is also active on [twitter](#) and on a lot of other forums, which some of them are popular like [stackoverflow](#).

Descriptive errors }>

“Error handling refers to the anticipation, detection, and resolution of programming, application, and communications errors. Specialized programs, called error handlers, are available for some applications. The best programs of this type forestall errors if possible, recover from them when they occur without terminating the application, or (if all else fails) gracefully terminate an affected application and save the error information to a log file.”
(Excerpt from “error handling” taken from <http://searchsoftwarequality.techtarget.com/definition/error-handling>)

Vaadin framework successfully fulfills this definition and is among the best error handlers. When an error occurs, the application continues to work, without even the need to refresh the page. In fact the only difference that will appear on the page is the error indicator that is built in all components. Hovering the mouse pointer over the field displays the error message. This makes error handling much easier for the developer. Since no normal person (that is, other than a software developer) understands what this means and knows what to do, Vaadin allows developers to change this default message to a more user friendly one. To the developers on the other hand Vaadin exceptions couldn't be more descriptive and easy to handle. They show the line numbers where the exception occurred, the related component, and the cause of the error.

Configuration Speed }>

Vaadin is very easy to setup as it needs no configuration to be used. All that is needed is JDK, eclipse IDE (among other IDEs such as Netbeans) and Tomcat (or another servlet container) and of course the framework itself, which can be downloaded as an Eclipse plugin. With Vaadin, no browser plug-in or JavaScript programming is required, framework extensions are done in Java, and no HTML or XML configuration is required. A [visual editor](#) is available in the framework's eclipse plugin enabling developers to create GUI by dragging and dropping components, without the need to write any code.

Active Releases }>

Development and releases are active, and new versions are released quarterly with a descriptive [roadmap](#) on the website, showing new features and future updates. An Up-to-the-minute list of things that the development team is working on is maintained on the [Vaadin timeline](#). Bugs can easily be reported on the [website](#), along with suggestions of new features. These are handled in the form of tickets and are displayed in the timeline and the developers are active in fixing bugs and replying to tickets.

Compatibility }>

Updates to the framework are well documented, and a [migration guide](#) exists which tells the differences in the new version allowing you to be able to migrate to it easily. Vaadin is based on Java making it compatible with most operating system supporting Java 5 or later. Vaadin is a single JAR file without external dependencies. Add the JAR to your web application project and you are ready to start creating the user interface. It also exists as plugins for famous IDEs such as eclipse. Vaadin is compatible with Windows (XP or later), Linux and Mac OSX and supports all newer versions of browsers such as IE (6+), Firefox (3+), Safari (4+), Google chrome and Opera. Vaadin also works on Android browser (2+) and it even works on IOS Safari (4+) although IOS doesn't support Java. Vaadin also supports touch controls and allows IOS applications to be made using [Vaadin TouchKit](#). Vaadin has complete add-on architecture. Add-ons, such as new components can be packaged as single JARs, which can then be used in other projects just by adding them to the project's classpath. Using add-ons from [Vaadin Directory](#) into a project is as easy as downloading them and recompiling the widgetset. Vaadin is also compatible with the Google App Engine (GAE).

Frequency of questions and response rate }>

Vaadin has a high frequency of asked questions and a high response rate to the questions asked...tens of questions are asked and answered daily, the framework is also updated frequently as shown here <http://dev.vaadin.com/timeline> the forums show the frequency of questions and answers to problems facing developers and the recent activity by developers on the forums <https://vaadin.com/home>

Popularity }>

Thousands of companies all over the world are creating compelling and user friendly web applications using Vaadin such as ctiviti explorer, Aktia forms generator, Aperte workflow,

Axioma Onlyconnect, Event tracker, healthcare portal, logica XQS service and many others that can be found here <https://vaadin.com/who-is-using-vaadin>

Scalability }>

“Does it scale?” This is one of the most common questions we get after one figures out how Vaadin works. Vaadin runs all UI logic on the server-side and user interaction in the client-side UI results in a large number of request-response round trips to the server. An experiment was made that showed that a single server could handle 20,622 AJAX requests / minute before exceeding 1% of errors (rejecting connections). To put that number into perspective, the actual real world movie ticket sales of 2009 were 1.4 billion tickets in US and Canada (source: MPAA - 2009 Theatrical Market Statistics). That's around 2600 sold tickets / minute, which translates to some 1300 purchase scenarios per minute assuming two tickets per transaction. So basically a single server could easily handle the load of selling all tickets for US and Canada! according to [this](#) experiment.

Learning curve }>

If you are a Java developer the learning curve is almost non-existent as all the work is done purely on the server side and Vaadin takes care of all the client side stuff for you.

Who is using Vaadin? }>

<http://www.tnt.com/>

<http://www.wapice.com/>

<http://www.xqs-service.de/>

<http://www.fedhaltero.qc.ca/>

References }>

<https://www.vaadin.com/forum>

<https://vaadin.com/book>

<https://vaadin.com/api/>

<https://vaadin.com/who-is-using-vaadin>

https://vaadin.com/forum/-/message_boards/view_message/137252

<https://demo.vaadin.com>

<https://vaadin.com/blog/-/blogs/vaadin-scalability-study-quicktickets>

https://vaadin.com/wiki/-/wiki/Main/Authenticating+Vaadin-based+applications/pop_up

<https://vaadin.com/home>

<http://dev.vaadin.com/timeline>

<http://code.google.com/p/vaadin-appfoundation/wiki/Authorization>

<http://stackoverflow.com/questions/1183801/should-i-use-vaadin-framework>

<https://vaadin.com/tutorial>

<http://blue-walrus.com/?p=187>

<http://dev.vaadin.com/wiki/Vaadin7/MigrationGuide>

<https://vaadin.com/roadmap>

<http://dev.vaadin.com/timeline>

<https://vaadin.com/bug>

<https://vaadin.com/wiki/-/wiki/Main/Vaadin%20TouchKit%20-%20Create%20iPhone%20applications%20using%20Vaadin>

<http://dev.vaadin.com/wiki/Addons/VisualEditor>

<http://vaadin.com/eclipse>

<https://vaadin.com/who-is-using-vaadin>

