



Orientation in Objects

HTML 5, CSS 3 + JavaScript IDE shootout

A comparison of tools for the development of HTML 5 Applications

) Schulung)

AUTOR



Sebastian Damm
Orientation in Objects GmbH

) Beratung)

Veröffentlicht am: 21.4.2013

ABSTRACT

It is quite normal in the IT business that every year one or two new technologies arrive that cause a fundamental hype and that promise to change literally everything. Once the hype wave diminishes it often appears as if the technology could not live up to its hype. With HTML 5 the hype seems to be justified, but for developers a good technology or language is often only as good as their tooling support. In this article we will compare some of the most popular IDEs for HTML 5 development regarding their support for HTML 5, CSS 3 and JavaScript including features like auto-completion, validation and refactoring.

) Entwicklung)

) Artikel)

Orientation in Objects GmbH

Weinheimer Str. 68
D-68309 Mannheim

Tel. +49 (0) 6 21 - 7 18 39 - 0
Fax +49 (0) 6 21 - 7 18 39 - 50

www.oio.de info@oio.de

Java, XML, UML, XSLT, Open Source, JBoss, SOAP, CVS, Spring, JSF, Eclipse

INTRODUCTION

Recent developments and the arrival of HTML5, CSS3 and foremost many new HTML/JavaScript APIs (canvas, offline storage, web sockets, asynchronous worker threads, video/audio, geolocation, drag & drop ...) resulted in a massive HTML5 hype. It is now possible to develop serious sophisticated web frontends only using HTML, CSS and JavaScript. With Microsoft abandoning Silverlight[1] and Adobe officially favoring HTML5 instead of Flash[2] for mobile development it is quite obvious that HTML5 is not just another huge hype bubble that will burst once the next shiny new technology arrives.

The vast spread of mobile devices with different operating systems is another aspect, where HTML5 can shine. The market for smartphones and tablets is heavily divided into Android, iOS, Windows Phone/Windows 8, BlackBerry and webOS, with the first two obviously being the big players. However, this segmented market means for developers that they have to rewrite their applications in several languages and with different technologies in order to target as much customers as possible. Now, instead of building native apps for a specific platform, developers can also develop one single web application using HTML5 and make it available on PCs and all kind of mobile devices without much trouble.

...HTML5 is here, it works and the hype seems justified, so...

THE TECHNOLOGIES ARE READY, BUT ARE THE IDES AS WELL?

The choice of an appropriate IDE or editor for your project is absolutely mandatory as it can greatly ease and speed up your development if you choose the right one. Contrary it can impair the chances of success if you don't.

This is why we decided to look for IDEs that are considered being good by the majority of the community and to compare them.

These are the IDEs that we chose for our comparison:

Name	NetBeans 7.3	Microsoft WebMatrix 2	Aptana Studio	JetBrains WebStorm 6	Visual Studio Express 2012 for Web	Komodo Edit	EclipseEE
Manufacturer	Oracle	Microsoft	Aptana	JetBrains	Microsoft	ActiveState	Eclipse Foundation
License	free (GPL, CDDL)	Microsoft Public Licence	free Aptana Public License, GNU GPL	proprietary 26-89	Microsoft Public Licence	Mozilla Public Licence 1.1	free (EPL)
Latest Release MM-DD-YYYY	2-21-2013	9-6-2012	8-31-2012	3-6-2013	9-7-2012	11-2-2012	6-27-2012

TESTED FEATURES

It is no surprise that almost every good IDE on the market has its unique features. Nevertheless every IDE should provide the core functionalities that a developer expects if he wants to develop a web application. Following criteria were considered for an impartial comparison.

Regarding the **editor in general** it is interesting how the UI looks like in general and how it is structured and arranged because this affects the usability of the IDE. The next point is the **project creation** with intuitivity and the amount and quality of project templates being important aspects. When it comes to **auto-completion for HTML and CSS** it is first of all relevant how fast and accurate it is in general. Furthermore we will test if the auto-completion (hereinafter referred to as AC) includes tags, attributes, attribute-values and pseudo-classes or only parts of this collection. Auto-completion for files, file paths, fonts and colors surely isn't mandatory but will also be tested. Last but not least we will check if there is auto-completion in the HTML for IDs and classes that were specified in an included CSS file and vice versa.

JavaScript auto-completion will be tested regarding the general support for browser objects (like 'window' or 'navigator') and user-created variables and methods. The support for **JavaScript / HTML 5 APIs** will be checked separately. Furthermore we will also take a look at the out-of-the-box assistance for **JQuery**. Afterwards we will take a look at the **validation for HTML, CSS and JavaScript**, where it is important that syntax errors are marked and unknown tags/elements/variables/methods are reported. Next, we will look for **refactoring options for all three technologies** and check if there are e.g. "rename" or "move" options.


Not a must-have per se but nonetheless a really nice feature is **live editing** with the ability to have your editor and a browser window side-by-side and directly see the impacts of your code changes without the need to reload the site. **Code navigation** is the next category where we will check if there are navigable overviews for all file types and options like "go to declaration" or "find usages". Last but not least we will also rate the **performance** in general and the response times for the auto-completion.

As mentioned above, these are only the core functionalities that can ease the development of HTML5 applications.

Regarding the JavaScript support it has to be considered that due to the design of the language and the fact that it is not statically typed, it is quite impossible to achieve an IDE support that comes close to what IDEs can offer for statically typed languages like Java or .NET. Please also note that we just tested the out-of-the-box experience of every IDE and that there are existing plug-ins or workarounds for missing features in some IDEs.

These are the results of our tests:

HTML 5, CSS 3 and JavaScript IDE Shootout



Criterion	IDE	NetBeans 7.3	Microsoft WebMatrix 2	Aptana	JetBrains WebStorm 6	Visual Studio Express 2012 for Web	Komodo Edit	EclipseEE
Editor in general		++	+	+	++	○	+	++
Project creation		+	+	+	++	○	○	○
Auto-completion HTML5		++	+	++	++	+	○	○
Auto-completion CSS3		++	-	+	+	+	+	○
Auto-completion JavaScript		++	+	○	+	+	+	○
Auto-completion JavaScript (HTML5 APIs)		+	++	-	++	++	-	-
Auto-completion JQuery		++	/	/	++	/	/	/
Validation HTML/CSS		+	++	+	++	++	+	++
Validation JavaScript/ JQuery		+	+	○	++	-	○	+
Refactoring HTML/CS		++	/	/	++	/	/	/
Refactoring JavaScript/ JQuery		/	/	/	++	/	/	+
Live Editing / Instant Feedback		++	/	/	++	/	/	/
Code navigation		++	-	+	++	-	○	+
Performance/ response times		+	++	++	++	+	+	+

++/++ (very) good
 ○ average
 -/- below average
 / feature not implemented

Orientation in Objects GmbH Weinheimer Str. 68 68309 Mannheim www.oio.de ©2013

Abbildung 1: HTML 5, CSS 3, JavaScript IDE Shootout

IN-DEPTH LOOK

NETBEANS 7.3

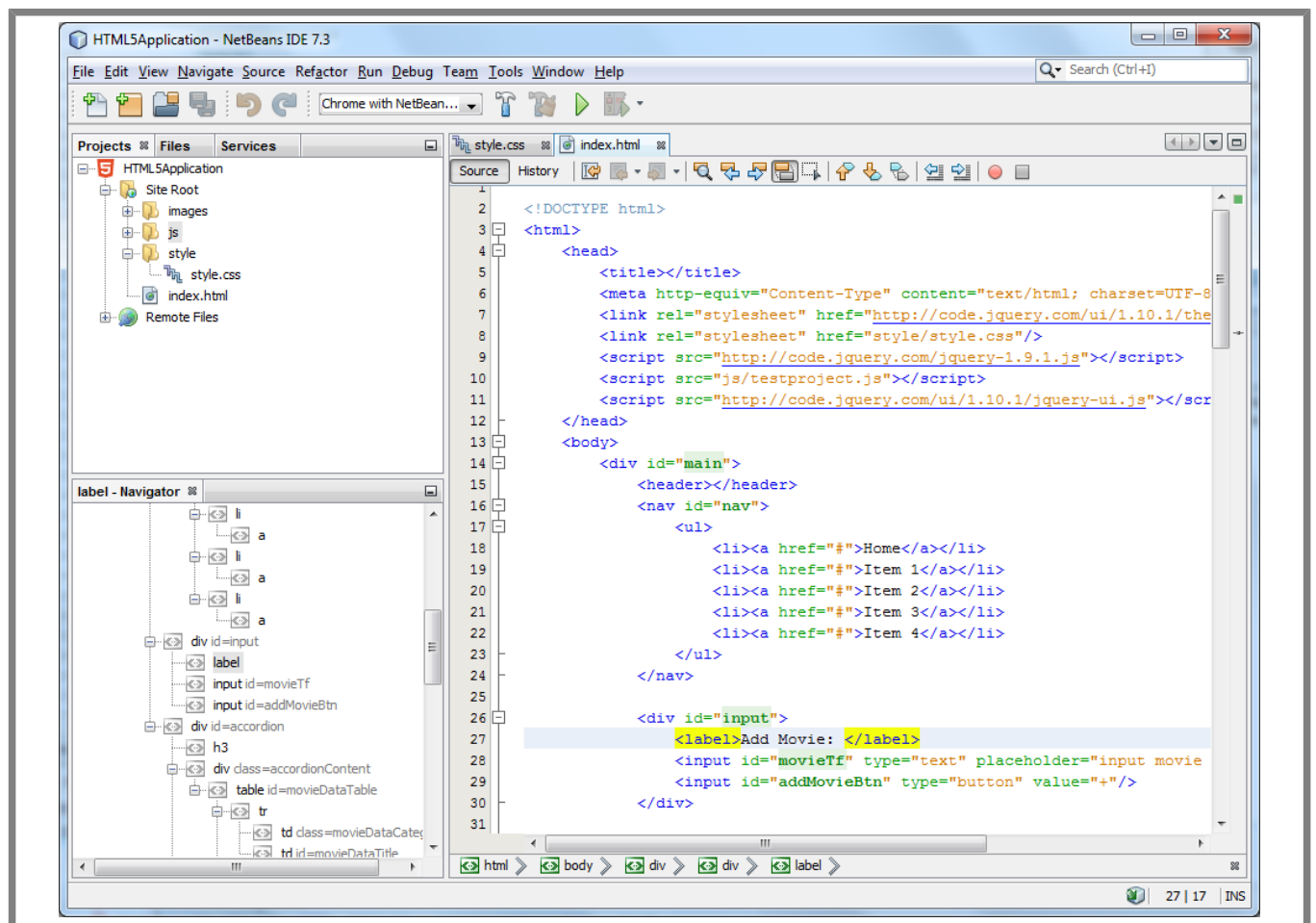


Abbildung 2: NetBeans 7.3- navigable DOM overview

NetBeans[3] as an IDE is known and acclaimed for several years now in the Java community. Since the last official version, the IDE also supports HTML 5, CSS 3 and JavaScript.

The clearly arranged editor appears nice and modern although the amount of menus and submenus may seem a little confusing for beginners. When creating a project, NetBeans offers a wide range of templates, e.g. including Twitter Bootstrap, Boilerplate or Angular JS.

Not only the HTML and CSS auto-completion and validation features per se are very good, but also the interaction between those two is very compelling. If you use IDs and classes in your HTML markup you will have AC for the CSS file that you included in the header. The other way around, if you specify rules for certain IDs and classes in the stylesheet, you will also have AC for them in your HTML. Additionally there is a "Find usages" option for the IDs and classes in your HTML that directly points you to the declaration in the corresponding CSS file.

Another nice feature is that you are able to rename IDs and classes in the HTML/CSS and that the changes automatically affect the corresponding entries in the CSS/HTML. This kind of refactoring can only be matched by WebStorm.

The HTML AC offers support for HTML (5) tags, attributes and even attribute values if possible. It is also context-sensitive, for example in the sense of offering the "li" tag as the first option if your cursor is inside a surrounding "ul" tag.

Last but not least inside an HTML file there is also auto-completion for file names and file paths, which is a nice detail when including stylesheets/script files or embedding pictures. A small point of criticism is that the closing tags aren't inserted automatically as done in most other IDEs.

The support for CSS is almost as good as for HTML. There is AC for HTML tags in CSS selectors, attributes, attribute values (including their CSS3 representatives) and pseudo-classes. As already mentioned above, you also have AC for IDs and classes that you used in the HTML. Unfortunately there is no AC for file names and file paths in CSS.

JavaScript in general is very well supported, regardless if it comes to AC of browser API objects (window, navigator, ...) or user-created variables/methods or syntax validation. Unfortunately there is only partial support for HTML 5 APIs. For example there is AC for "localStorage", but not for "navigator.geolocation". When it comes to refactoring for JavaScript, NetBeans sadly doesn't offer the same quality as with HTML/CSS refactoring because for JS there is no "rename" option for example.

JQuery is an aspect where NetBeans can earn some points again. Out-of-the-box it offers an impressive AC even including IDs and classes used in the HTML in the JQuery selectors `$("#someId")`, respectively `$(".someClass")`.

After installing an extension for Google Chrome, NetBeans offers a very good live editing support and changes you make to the code are immediately displayed in the browser.

The code navigation is also very nice and comes close to what developers know from Java or .NET development. There are navigable overviews for the HTML, CSS and JS, a "go to declaration" for JS variables/methods and the "find usages" option mentioned above. Performance wise it has to be said that the AC is quite quick, but compared to the other candidates the performance is a little worse, especially if you run NetBeans on an older/weaker PC.

If you have a Java backend it is obviously a nice aspect that you could develop your Java backend and the HTML 5 frontend in the same IDE.

MICROSOFT WEBMATRIX 2

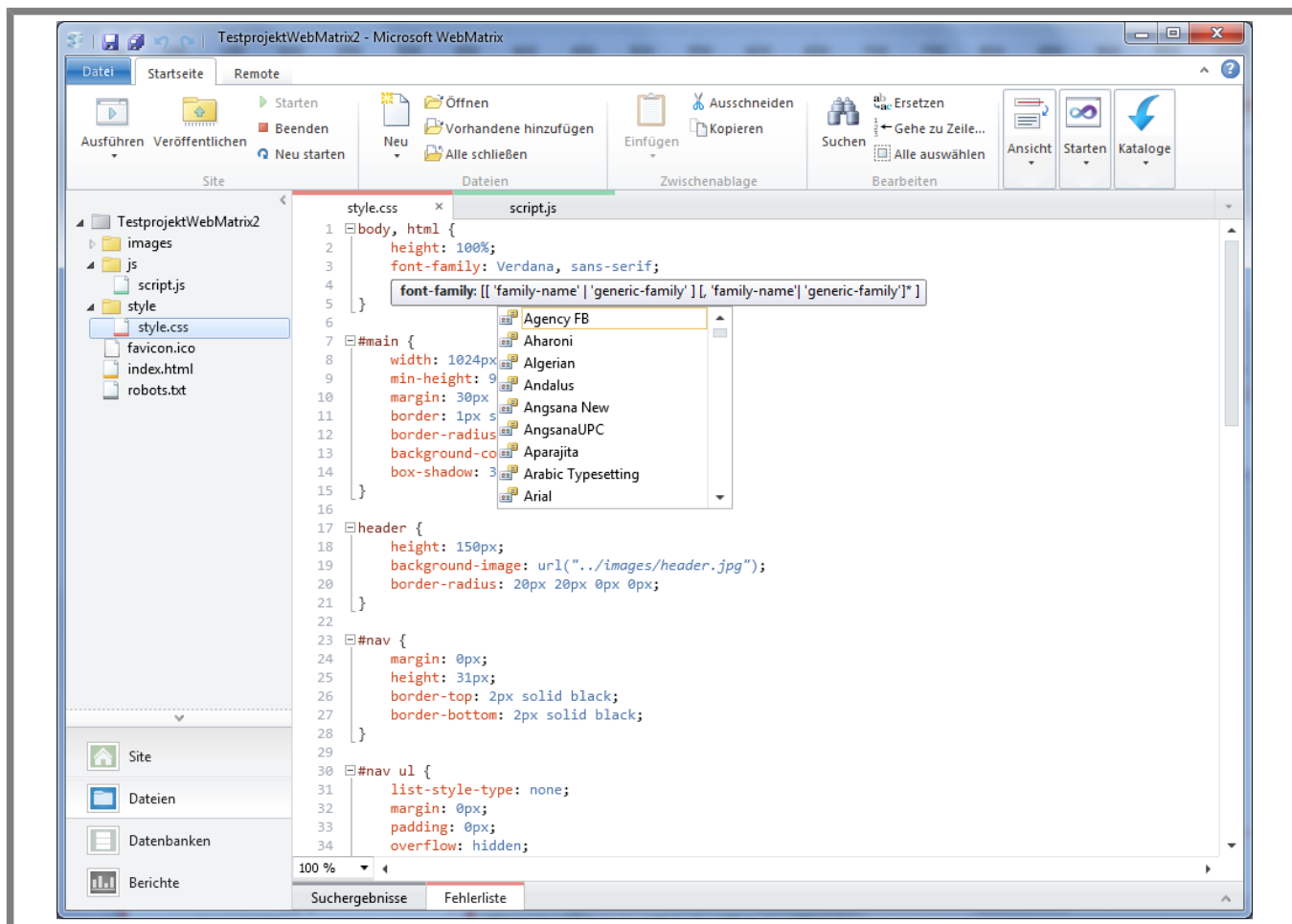


Abbildung 3: WebMatrix 2- font auto-completion

WebMatrix 2[4] is a free IDE from Microsoft for web development featuring templates for CMS, Blog or E-Shop projects and deployment assistance. It supports a huge amount of frameworks with WordPress, Joomla and Drupal probably being the standout candidates. The score in the 'project templates' category for WebMatrix 2 is nevertheless only "good" because this comparison focuses on HTML 5 applications and in this regard the templates can't compete with NetBeans or WebStorm.

The editor is quite minimalistic but nicely arranged and features the same ribbon style menus that were introduced by the Microsoft Office 2007 suite. Unfortunately the editor is a little bit too minimalistic, lacking a feature to split the window and view and edit two source files side-by-side.

The support for HTML 5 is quite good in general and, in contrast to NetBeans, closing tags are inserted automatically. Context-sensitivity (see NetBeans) is also given as well as AC for HTML (5) tags, attributes and attribute values. Missing features are AC for IDs and classes that are defined in the CSS and AC for file names and file paths. Anyhow, these are only minor points of criticism and therefore the HTML 5 support has to be rated as "good", especially because the validation for HTML (and CSS) is also quite nice and unknown tags are immediately marked, which should prevent typos.

CSS 3 however isn't supported as good as HTML. There is support for CSS (3) attributes and attribute values and WebMatrix also features one of the best AC for fonts and font families of all candidates. Hence it is quite a pity that there is no AC for HTML tags in CSS selectors, no AC for IDs and classes used in the HTML, no AC for pseudo-classes and also no AC for file names and file paths. AC for attributes and attribute values may be more important than AC for HTML tags and pseudo-classes, but compared to other IDEs, WebMatrix can't completely keep up in this department.

Regarding support for JavaScript, the AC is quick and most of the times accurate, syntax errors are immediately marked and there is also support for HTML 5 APIs.

In the department of refactorings WebMatrix hasn't really much to offer. There are no refactoring options whatsoever for HTML, CSS or JavaScript. Live editing and code navigation are further categories where WebMatrix can't offer much.

Lastly, regarding the performance, WebMatrix can gain some points again. The overall performance is really good and the AC is always very fast. Furthermore it doesn't consume as much memory as most of its competitors.

APTANA STUDIO

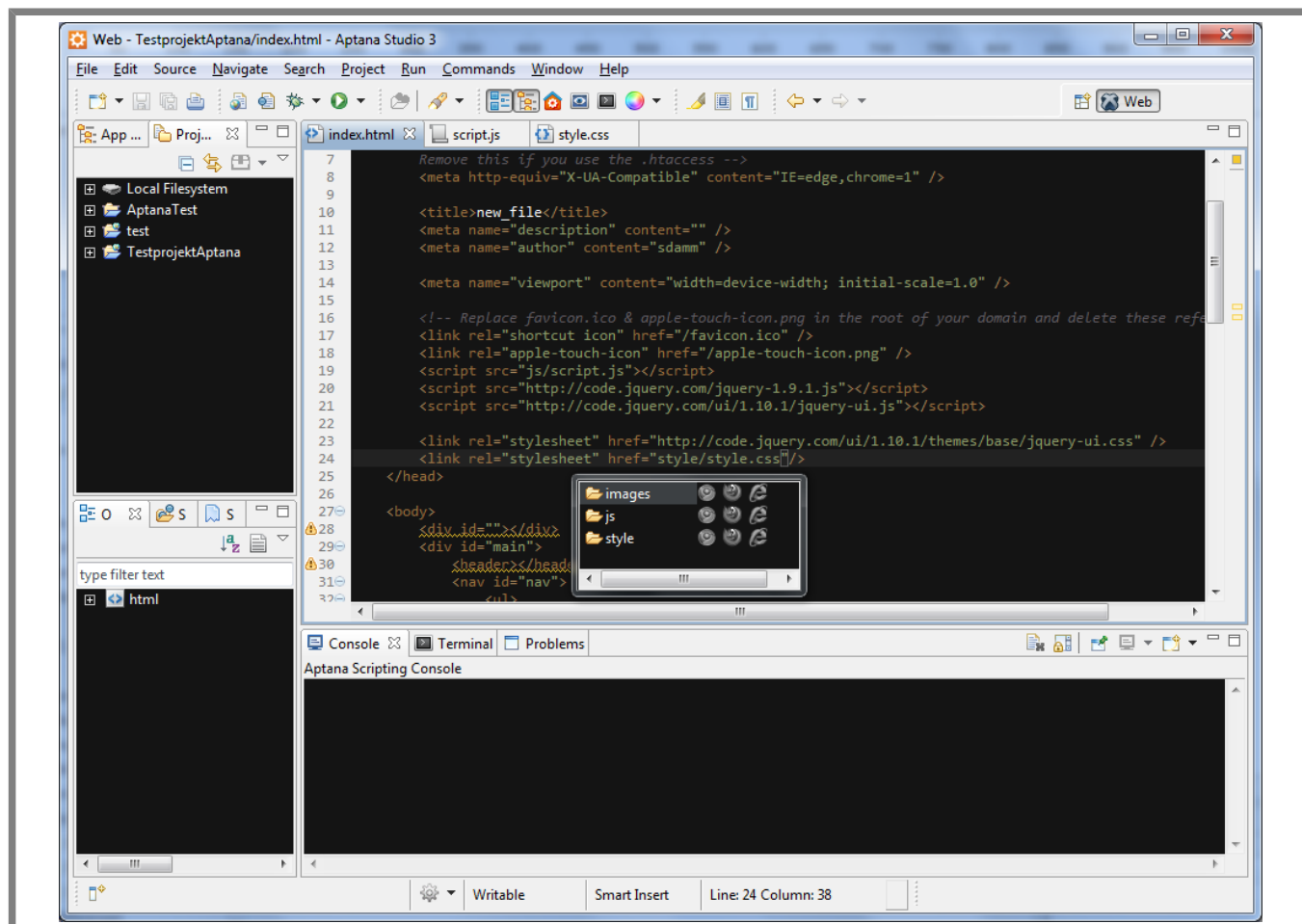


Abbildung 4: Aptana Studio- file names/paths auto-completion

Aptana Studio[5] is based on Eclipse. A fact that it simply can't deny because it basically looks exactly like a darker version of the standard Eclipse IDE. The IDE features a nice overview and in the default settings a dark background and white and colored fonts, which appears a little weird in conjunction with the typical light-gray Eclipse toolbars and menus.

When it comes to project creation, Aptana e.g. offers a template for HTML 5 Boilerplate and various file types.

The HTML support is very good. There is a quick AC for HTML (5) tags, attributes and attribute values, closing tags are inserted immediately and there is AC for file names and file paths. Finally Aptana also offers AC for IDs and classes that are defined in the CSS and the syntax and tag validation is also really good. There is only one small point of criticism because the AC for HTML tags isn't as context-sensitive as the AC that most competitors can offer.

The CSS AC is also fairly good as it covers HTML tags in CSS selectors, CSS (3) attributes, attribute values and pseudo-classes as well as a nice support for fonts and font families. AC for IDs and classes that are used in the HTML and for file names and file paths is sadly missing.

Regarding JavaScript and JQuery Aptana unfortunately can't offer such high quality support as for HTML and CSS. While the AC for browser variables and methods is fast and accurate, there is no reliable (sometimes there is, sometimes there isn't) AC for user-created code. Support for HTML 5 APIs is missing completely as well as out-of-the-box support for JQuery.

When it comes to validation it has to be said that HTML syntax errors and unknown HTML tags are directly marked as errors whereas CSS syntax errors/unknown tags and JavaScript syntax errors aren't covered.

Refactorings and live editing are also features that aren't provided but at least the code navigation (navigable overviews, "go to declaration" for JS) and the performance are really good.

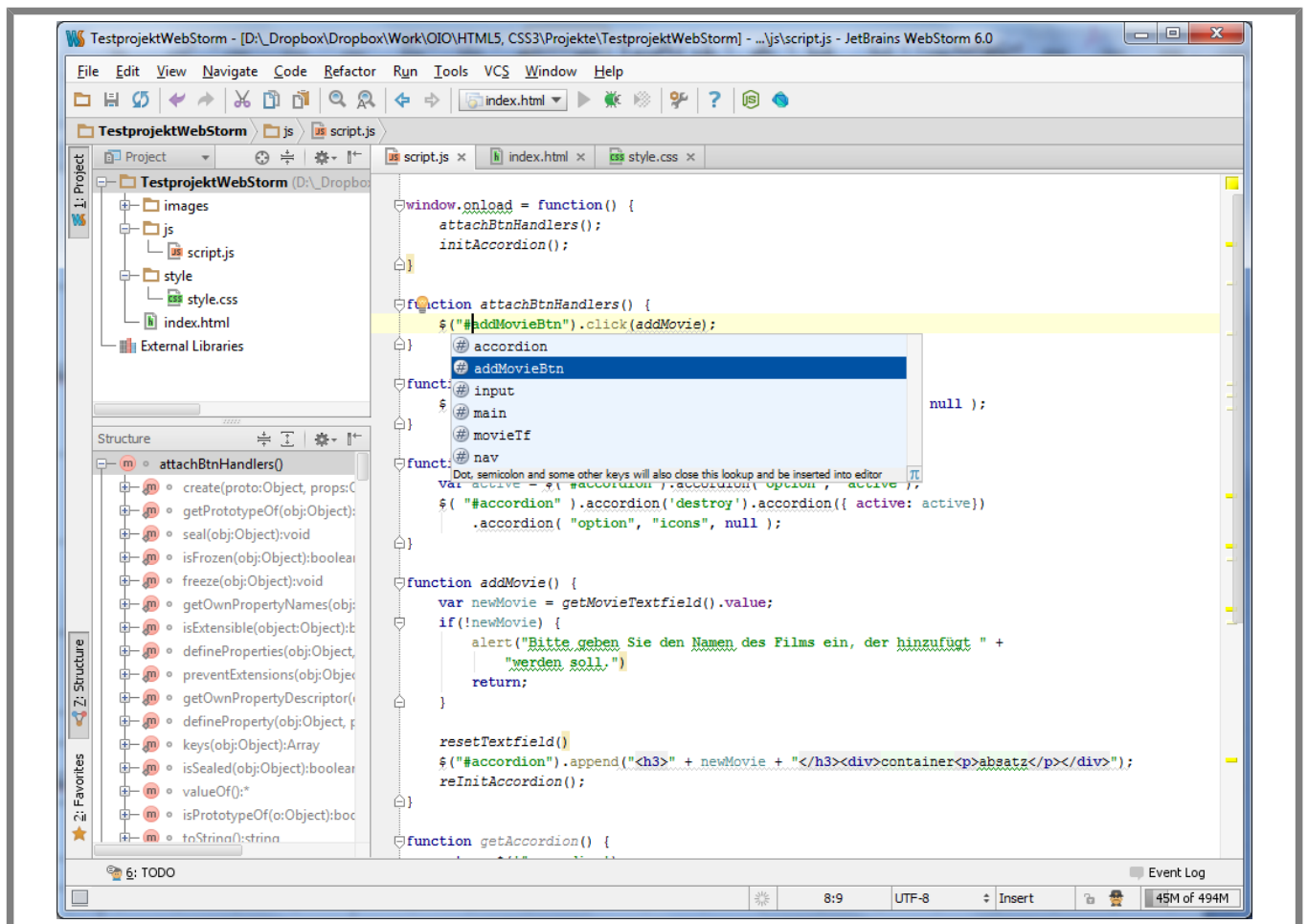


Abbildung 5: WebStorm 6- JQuery auto-completion

JetBrains[6] is the creator of IntelliJ, a popular commercial Java IDE.

Targeted at web developer they just recently released[10] the 6th version of WebStorm, which is also the only proprietary candidate in the comparison.

Right at the beginning, WebStorm can convince developers with a nicely arranged editor window and many project templates, including Twitter Bootstrap, Boilerplate, node.js and even TypeScript and Dart.

It is also the only IDE that can compete with NetBeans when it comes to HTML support and in fact, even surpasses it a little bit. The AC in general is fast, accurate and context-sensitive and it is available for HTML (5) tags, attributes, attribute values, pseudo-classes, file names and file paths and also for IDs and classes that are used in an included stylesheet. Considering that closing tags are inserted automatically and that WebStorm, in contrast to NetBeans, marks unknown HTML tags as errors, the HTML support can only be regarded as outstanding.

A little worse but still really good is the support for CSS. Generally it feels like the fastest AC of all candidates and it is available for HTML tags in CSS selectors, CSS(3) attributes, attribute values, pseudo-classes, file names, file paths and font families (no specific fonts). A small but unique and nice little feature is that there is a little preview for colors in your CSS, even if they are specified via a hex value.

What is missing is AC for IDs and classes that are used in the HTML file. If this is no big deal for you, you can consider the CSS support to be really great, but copy & paste of IDs and classes between the HTML and CSS is quite tedious (especially in larger files). And I guess that every web developer already wondered several times after reloading the web site why the new style wasn't applied when realizing that there was a small typo.

The support for JavaScript and JQuery is altogether the best amongst all candidates. AC for JavaScript is fast and accurate, including HTML 5 APIs, JQuery is supported without the need for a plug-in and it features the same AC as NetBeans regarding the JQuery selectors. There is validation for syntax errors and unknown variables and methods are immediately marked as errors. Unique to WebStorm (at least regarding the quality) are the refactoring options for JavaScript e.g. including "rename", "change signature" or "extract variable/method/parameter".

When it comes to live editing, WebStorm is the only IDE that can compete with what NetBeans has to offer. Like NetBeans, this feature requires a Google Chrome plug-in.

Very comparable to NetBeans is also the code navigation as there are navigable overviews for all file types, a "go to declaration" option for JavaScript and a "find usages" option pointing from IDs and classes in the CSS to all occurrences in the HTML and vice versa.

As mentioned in the beginning, JetBrains is also the creator of the Java IDE IntelliJ. This IDE comes with most of the features built-in that WebStorm has to offer and with the additional supply of plug-ins, IntelliJ is basically a superset of WebStorm. This is very interesting because if you want to back your HTML 5 application with a Java server application, you can code your backend and frontend in the same IDE, just like with NetBeans and Eclipse.

VISUAL STUDIO EXPRESS 2012 FOR WEB

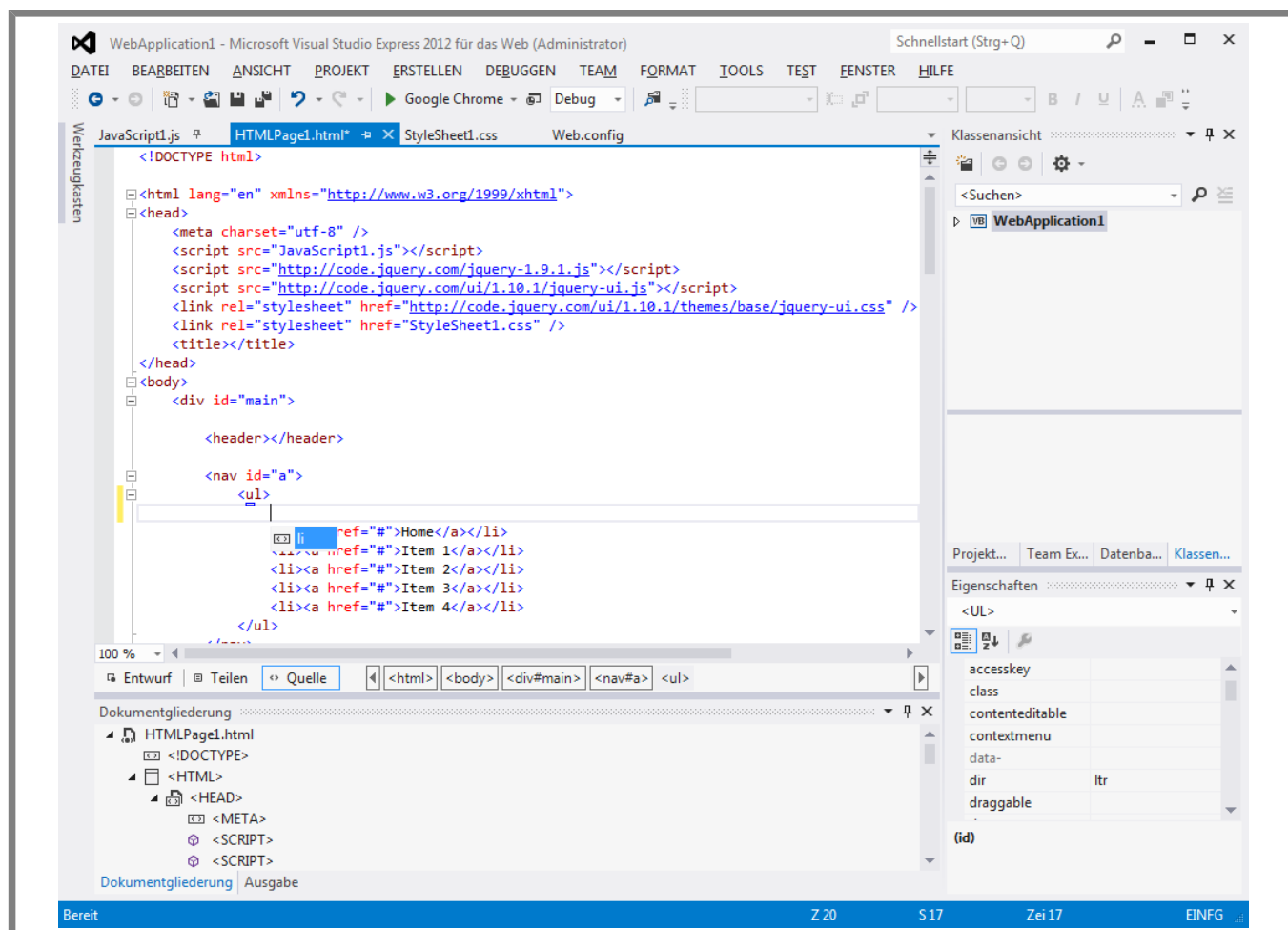


Abbildung 6: Visual Studio Express 2012 for Web- context sensitive HTML auto-completion

Microsoft's IDE flagship Visual Studio[7] is also available as an Express version especially designed for web development.

Generally the editor, its toolbar and little explorer windows seem a little crowded compared to other IDEs and also the project and file creation isn't equally intuitive.

The support for HTML (5) is really good. There is AC for HTML (5) tags, attributes and attribute values, closing tags are inserted automatically and there is also AC for file names and file paths. Compared to NetBeans and WebStorm there is only one feature that is missing and that is AC for IDs and classes that are defined in the CSS. This is only one feature, but a pretty important one in my opinion as you can read above in the paragraph about WebStorm.

CSS 3 is also well supported as there is AC for attributes, attribute values and pseudo-classes. The AC for fonts and font families is also quite nice and can only be matched by Aptana and WebMatrix. Strangely missing are AC for the HTML tags in CSS selectors and unfortunately also for IDs and classes that are used in the markup.

JavaScript support has its good and its bad sides with an accurate and quick AC for browser and user-created variables and methods and full support for the HTML 5 APIs but on the other side no support for JQuery and no sort of validation or refactoring options. When it comes to live editing and code navigation, Visual Studio Express 2012 for Web sadly can't offer much, but performance wise it is surprisingly fast and the AC options are always displayed without any delay.

KOMODO EDIT

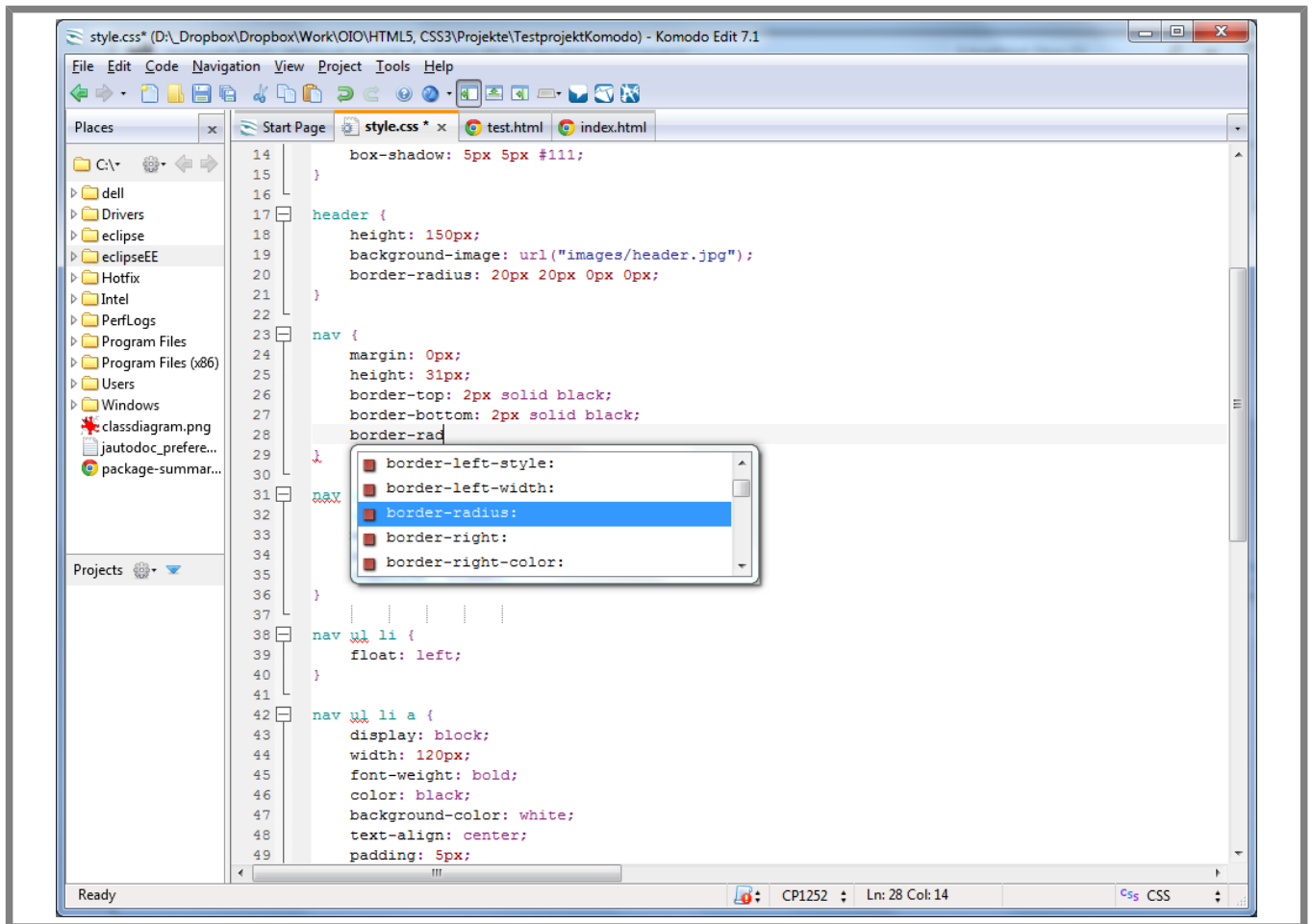


Abbildung 7: Komodo Edit- CSS 3 auto-completion

Komodo Edit[8] is the free alternative to ActiveState’s proprietary IDE Komodo[11].

The editor itself is quite minimalistic but well arranged. There are many templates for file types but no real “web projects” or templates for Bootstrap or HTML 5 Boilerplate.

HTML support is only mediocre with a context-sensitive AC for HTML tags, attributes, attribute values and auto-inserted closing tags. Unfortunately this AC is not available for HTML 5 tags and attributes. Furthermore missing is AC for file names and file paths and for IDs and classes that are defined in the CSS.

The support for CSS is better because there is also full-blown support for CSS 3 including HTML tags in selectors, attributes, a mediocre AC for attribute values (not as accurate as most competitors), pseudo-classes and font families.

Syntax errors are marked both for HTML and for CSS and the validation mechanism also identifies unknown tags in your HTML.

Regarding JavaScript the general AC is fast and accurate, but unfortunately there is no support for the HTML 5 APIs at all. JQuery support is missing as well as any form of refactoring or live editing. The code navigation cannot compete with most competitors but at least offers a “go to declaration” method for JavaScript.

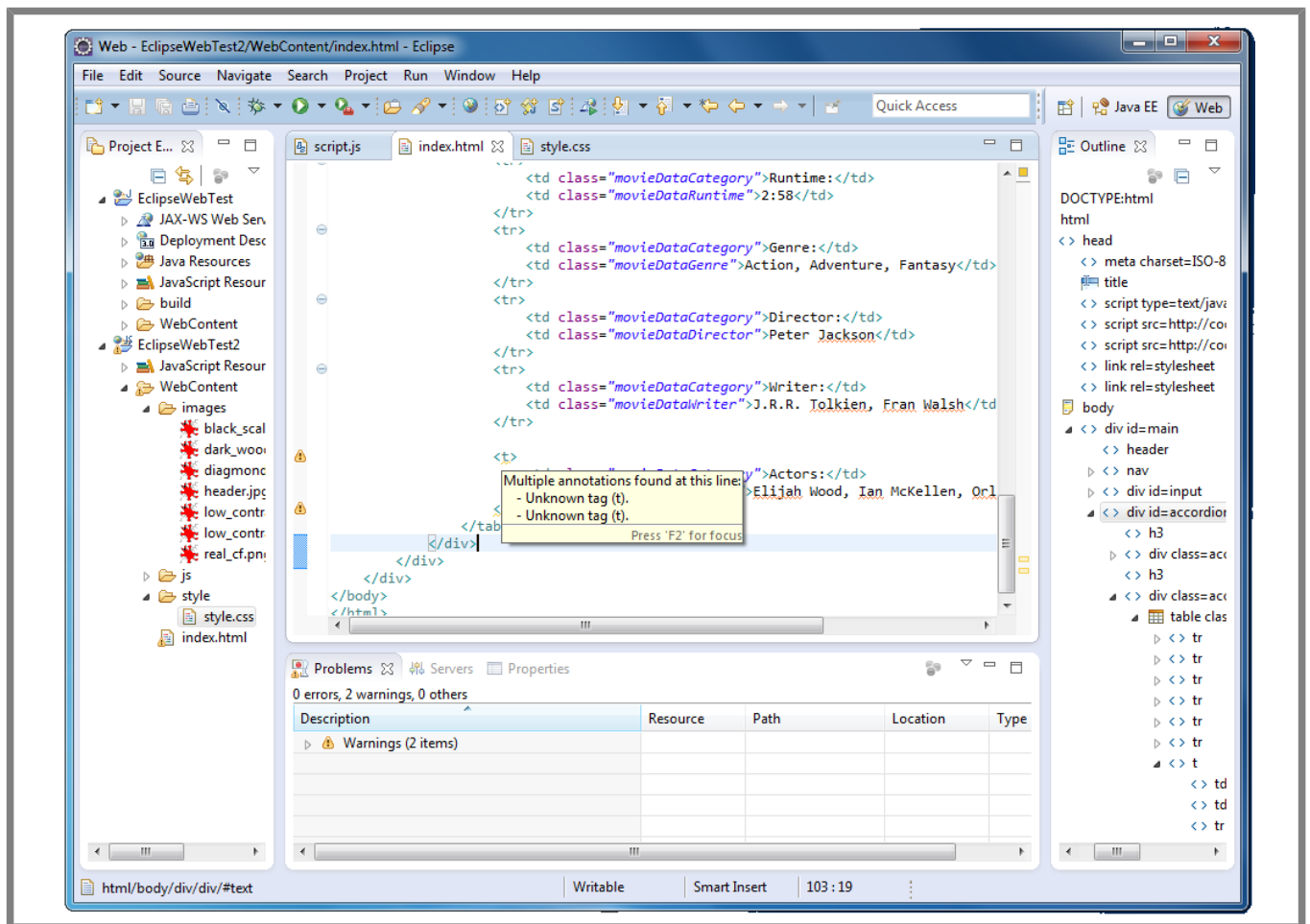


Abbildung 8: EclipseEE- HTML validation

The enterprise version of the Eclipse IDE[9] also offers quite good support regarding web development thanks to the WTP (WebToolsPlatform) project.

The components of the IDE window are nicely arranged. For developers that want to develop HTML 5 applications however, the toolbar and the amount of menus may seem a little confusing at first, mainly because they won't need most features for their purposes.

Project creation is easy and intuitive but there are no project templates like those that some of the other candidates offer.

The HTML support is quite average with AC for HTML (5) tags, attributes and attribute values. Sadly the latter two do not cover all new HTML 5 elements. Also missing is AC for file names and file paths as well as AC for IDs and classes that were defined in the stylesheet.

Pretty similar is the CSS support with an AC for HTML tags in selectors, attributes, attributes values and pseudo-classes that is generally fast and accurate, but lacks any kind of support for CSS 3. AC for file names and file paths is also missing alongside the AC feature for in the CSS defined IDs and classes.

The JavaScript support is only average. The AC for user-created code is most of the times accurate but although there is AC for browser objects (like 'window' or 'navigator') you sometimes miss a certain method of these objects. More importantly there is unfortunately no support for the JS/HTML 5 APIs and JQuery. Validation is pretty mixed as in HTML syntax errors and unknown tags are directly marked but in CSS syntax errors and incorrect attributes/attribute values are not covered.

When it comes to refactorings, Eclipse hasn't much to offer for HTML/CSS but for JavaScript there are "rename" and "change signature" options with the last one unfortunately often producing errors. The code navigation offers navigable overviews for all three file types and the performance is comparable to NetBeans.

Also comparable to NetBeans is the fact that you could write your Java backend code and the HTML 5 frontend in the same IDE.

CONCLUSION

It is quite obvious that a good HTML and CSS auto-completion and validation can almost be considered as a standard amongst good IDEs today, which doesn't mean that there are no IDEs that are outstanding in those aspects nevertheless. But where most IDEs really differ is the support for JavaScript, JQuery and "more advanced" features like refactorings, the interaction between HTML and CSS, live editing or a quick and intuitive code navigation.

An absolute recommendation cannot be given, because every IDE has its pros and cons and personal preferences and project needs therefore decide which IDE is the best for you. Regarding the plain results, NetBeans 7.3 and WebStorm 6 probably have to be considered as the most complete IDEs for HTML5 development right now.

Again it has to be said, that every tested IDE will help you to develop HTML5 applications. Mediocre ratings in some categories therefore mostly result from the difference with IDEs that are truly great in these aspects.

To answer the question that occurred at the beginning: The technologies are ready and yes, the IDEs are as well. Have fun coding HTML 5 applications.

REFERENZEN

- [1] Microsoft abandoning silverlight
http://www.osnews.com/story/25165/Microsoft_Abandons_Flash_Silverlight_in_Windows_8
- [2] Adobe officially favoring HTML5 instead of Flash
<http://www.wired.com/gadgetlab/2011/11/adobe-kills-mobile-flash>
- [3] NetBeans 7.3
<http://netbeans.org/>
- [4] Microsoft WebMatrix 2
<http://www.microsoft.com/web/webmatrix/v2/>
- [5] Aptana Studio
<http://www.aptana.org/>
- [6] JetBrains WebStorm 6
<http://www.jetbrains.com/webstorm/>
- [7] Visual Studio Express 2012 for Web
<http://www.microsoft.com/visualstudio/deu/products/visual-studio-express-for-web>
- [8] Komodo Edit
<http://www.activestate.com/komodo-edit>
- [9] EclipseEE
<http://www.eclipse.org/downloads/>
- [10] WebStorm 6 – HTML5/CSS3/JavaScript IDE now with TypeScript, CoffeeScript and Dart support (OIO-Blog)
<http://blog.oio.de/2013/03/08/webstorm-6-html5css3javascript-ide-now-with-typescript-coffeescript-and-dart-support/>
- [11] ActiveState´s proprietary IDE Komodo
<http://www.activestate.com/komodo-ide>