

بله! همانطور که حدس زده می‌شد بالاخره مایکروسافت تکلیف خودش را با کتابخانه‌های Ajax ایی تولید شده در طی این چند سال مشخص کرد و از این پس انتخاب اصلی جهت تولید برنامه‌های ASP.NET مبتنی بر Ajax ، تنها jQuery است. اصل مطلب رو می‌تونید اینجا مطالعه کنید:

[Javascript Libraries and ASP.NET: A Guide to jQuery, AJAX and Microsoft](#)

[The Murky Future of ASP.NET AJAX](#)

خلاصه آن:

ASP.NET AJAX در آینده نیز کاملاً پشتیبانی می‌شود، اما شهبوند درجه یک محسوب نخواهد شد؛ مانند استفاده از ScriptManager و UpdatePanel

Ajax Control Toolkit که اکنون به صورت سورس باز در سایت CodePlex نیز قابل دریافت است، منسوخ شده در نظر گرفته شده و تنها در آینده شاید هر از چندگاهی به رفع باگ‌های گزارش شده در آن پرداخته شود (اما دیگر به صورت فعال توسعه داده نخواهد شد).

Microsoft Ajax Library / ASP.NET Ajax Library هر چند تشابه اسمی با ASP.NET Ajax دارند اما جزئی از کتابخانه‌ی AJAX Control Toolkit بوده و این مورد نیز از این پس منسوخ شده در نظر گرفته خواهد شد (مانند استفاده از DataView یا Sys.require).

نظرات خوانندگان

نویسنده: farbod
تاریخ: ۱۳۸۹/۰۷/۲۶ ۲۲:۵۱:۴۴

سلام آقای نصیری. من، هم با این کتابخانه و updatepanel کار کردم و هم jquery. درسته که خروجی کار با jquery بسیار با کیفیت تر و انعطاف پذیری هم بیشتره، ولی حقیقتش هر موقع از این روش استفاده می کنم احساس خوبی ندارم! به نظرم کدها بسیار نا مرتب و به قول معروف اسپاگتی هستن. من تمام تلاشم رو میکنم که در داخل فایل جاوااسکریپت بسیار با قاعده و طبقه بندی شده بنویسم، ولی باز هم .. به خصوص وقتی لازم میشه html رو داخل جاوااسکریپت و به صورت رشته ای تولید کنم. می خواستم بدونم واقعا به همین صورته یا من از روش های نادرست استفاده میکنم.

نویسنده: محمد امین شریفی
تاریخ: ۱۳۸۹/۰۷/۲۶ ۲۳:۱۱:۳۰

سلام.

درست مثل linq to sql، که اولش تبلیغ می کرد از ADO.NET باین سمت لinq to SQL و بعدش گفت از Linq to SQL باین به سمت ADO.net Entity
MFC هم همینطور محجور مانده، ASP کلاسیک

بدی محصولات ماکروسافت این هست که opensource نیست که توسط third party همیشه پشتیبانی و به مرور بهنگام بشه و خوبیش هم این هست که یک فناوری خوش دست و راحتی میده.

مهندس نصیری، نظرت درباره جاوا چی هست؟، پستی را برای مقایسه این دو می نویسید؟

نویسنده: وحید نصیری
تاریخ: ۱۳۸۹/۰۷/۲۷ ۰۰:۱۰:۲۳

farbod@

برای کلا جاوا اسکریپت هم کتابخانه های Unit test وجود دارد تا از این دلچرکین بودن خارج شوید:

[QUnit](#)

@محمد امین شریفی

اینبار انتخاب مایکروسافت درست بوده:

- 1- هزینه توسعه رو کاهش دادند. برای مثال هدف از Ajax Control Toolkit چیزی بوده شبیه به پلاگین های jQuery اما با هزینه بیشتر. اینبار شما هزاران پلاگین جی کوئری در اختیارتون هست و نه صرفا مجموعه ای Ajax Control Toolkit.
- 2- موردی رو که در مقاله بالا بهش اشاره شده و دلیل اصلی کنار گذاشتن ASP.NET Ajax بوده پایین بودن کارایی ASP.NET Ajax است. هم حجم بالایی دارد و هم تبادلات شبکه ای حجیمی را به همراه دارد که در هر دو مورد جی کوئری بهینه تر است. و یک سری موارد دیگر که قبلا بهش اشاره کرده بودم: (+)
- 3- انتخاب اصلی کتابخانه Ajax در ASP.NET MVC الان همین جی کوئری است و در مقاله فوق به یک سری افزونه و کتابخانه کمکی برای آن اشاره شده (که در حال توسعه و تکمیل است).

نویسنده: وحید نصیری
تاریخ: ۱۳۸۹/۰۷/۲۷ ۰۰:۲۸:۳۵

ضمنا یک مورد رو در باره ی LINQ to SQL و ASP کلاسیک اضافه کنم. جایگزین شدن entity framework بجای L2S یا ASP.NET به جای ASP کلاسیک، یک روند سالم و سلامت توسعه است. LINQ to SQL فقط محدود است به SQL Server اما الان اکثر بانک های اطلاعاتی موجود پروایدر EF دارند و مدل توسعه ای آن بسته نیست. ASP کلاسیک رو نمی دونم باهاش کار کرده بودید یا نه؟ رسما

یک فاجعه بود! مخلوطی از کدهای برنامه داخل کدهای HTML و وابستگی آن به اشیاء COM و غیره (اگر می‌خواستید مثلاً رمزنگاری را به آن اضافه کنید باید Active-X می‌نوشتید و در سرور رجیستر می‌کردید!). این مورد اصلاً قابل قیاس نیست با ASP.NET و امکانات دات نت فریم ورک.

نویسنده: shahin

تاریخ: ۱۷:۵۹:۳۷ ۱۳۸۹/۰۷/۲۷

سلام جناب نصیری یکی از مشکلاتی که برنامه نویسان تازه کار را اذیت می‌کند اجرای JQuery از سمت سرور هست. برای حل این ضعف چه توصیه ای دارید
مثلاً Ajax پارامترها را سمت سرور ست می‌کردیم و استفاده می‌کردیم.
اما JQuery باید سمت HTML همه کار را انجام داد.

نویسنده: وحید نصیری

تاریخ: ۱۸:۳۴:۵۱ ۱۳۸۹/۰۷/۲۷

من در مورد JQuery در این سایت مطلب زیاد منتشر کردم. به عمد هم تمامشون با دید ASP.NET بوده.
به تگ‌های JavaScript و یا ASP.NET در کنار صفحه مراجعه کنید یا کلا فایل CHM خلاصه سایت رو جهت سهولت مطالعه دریافت کنید.
و بله. میکروسافت اومده تمام این‌ها رو برای شما در ASP.NET Ajax کپسوله و از دید شما مخفی کرده. همین کپسوله سازی در پلاگین‌های JQuery هم با کیفیت بالا موجود است مثلاً: [\(+\)](#)

نویسنده: Mehran Zand

تاریخ: ۲۱:۴۲:۱۵ ۱۳۸۹/۰۷/۲۷

با سلام،
تمام این موارد که مبفرمایید درست، من خودم از طرفداران استفاده از JQuery هستم اما یک مسئله برای من هنوز وجود دارد
اون هم RAD است. در یکی از کتابخانه‌های آژاکسی برای asp.net مثل Anthem.Net
این مسئله خیلی خوب رعایت شده و کار توسعه نرم افزار مبتنی بر استفاده از آژاکس به سرعت انجام میشه. این برای من هنوز
سواله با JQuery چطور می‌توان به این سرعت نرم افزارهای Full-Ajax را توسعه داد.

نویسنده: وحید نصیری

تاریخ: ۲۲:۴۷:۵۶ ۱۳۸۹/۰۷/۲۷

میکروسافت ASP.NET Ajax را با اون عظمت و زحمتی که براش کشیده بودند، در مقابل JQuery بازنده اعلام کرده الان شما
صحبت از Anthem.Net می‌کنید که فقط منحصر است به ASP.NET web forms آن هم نگارش‌های قبل از سه آن + پشتیبانی از
مرورگرهای مختلف هم در آن ضعیف است.
هدف میکروسافت از اینکار مدیریت هر دو پروژه MVC و Web forms است آن هم با هزینه کم، کیفیت بالا و سازگار با تمام
مرورگرها.
ضمناً این رو در نظر باشید که یکی از توانمندی‌های Ajax، JQuery است (از کار با DOM گرفته تا دستکاری CSS نمایش داده شده، تا
Animation، مدیریت ساده‌تر رخدادها، اعمال قالب به سایت و غیره) و این مورد مزیت مهمی است نسبت به تمام کتابخانه‌هایی
که فقط برای یک کار و آن هم سهولت تولید برنامه‌های مبتنی بر Ajax ایجاد شده‌اند و از چند مشکل مهم رنج می‌برند:
- تک کاره‌اند. فقط Ajax.
- مشکل سازگاری با مرورگرهای مختلف را دارند.
- به صورت فعال توسعه داده نمی‌شوند؛ رفع باگ نمی‌شوند و غیره. برای مثال فواصل به روز رسانی همان Anthem.Net را بررسی
کنید.
- توسعه پذیر نیستند. برای مثال آیا می‌توان برای Anthem.Net افزونه نوشت؟
- حجم بالایی دارند.
- سرعت پایینی دارند.

jQuery در مورد تمام موارد عنوان شده حرف برای گفتن دارد از حجم کم تا سرعت بالاتر نسبت به اکثر کتابخانه‌های جاوا اسکریپتی دیگر تا توسعه‌ی منظم، سازگاری عالی با مرورگرها، توسعه پذیری و صدها و هزاران افزونه‌ی مهیا برای آن و غیره.

و RAD هزینه بر است. یعنی چی؟ یعنی حجم بالای کدهای اسکریپتی که باید به برنامه‌ی شما مثلا توسط ASP.NET Ajax تزریق شود که مبادا شما بخواهید دست خودتان را به نوشتن چند سطر کد جاوا اسکریپتی آلوده کنید. همچنین حجم تبادل اطلاعات ASP.NET Ajax را هم که مبتنی است بر RAD را هم با حجم اطلاعات مبادله شده توسط jQuery مقایسه کنید (با پلاگین فایرباگ مربوط به فایرفاکس). این حجم واقعا زیاد است و قابل مقایسه نیست. (تمام این‌ها هزینه‌های RAD است) ضمنا وجود 100 ها افزونه و پلاگین نوشته شده برای jQuery کار شما را بسیار بسیار ساده می‌کنند، مانند پاسخ قبلی من در این مطلب.

فقط باید کمی وقت بگذارید و چیزی را بیاموزید که واقعا ارزش دارد. گیرم فردا نخواستید با ASP.NET کار کنید. تمام این اطلاعات در PHP هم به درد شما می‌خورد، چون قسمت سمت کلاینت آنچنان تفاوتی نمی‌کند و jQuery یک کتابخانه‌ی سمت کلاینت است.

نویسنده: وحید نصیری
تاریخ: ۱۳۸۹/۱۲/۱۹ ۰۱:۳۷:۵۳

[\(+\)](#) در ادامه ...