

خلاصه نکاتی که من در تهیه یک برنامه ASP.NET MVC رعایت می‌کنم:

- استفاده از [T4MVC](#) اجباری است. به هیچ عنوان نباید از رشته‌ها برای مشخص سازی نام کنترلرها یا اکشن متدها در قسمت‌های مختلف برنامه استفاده شود.
- تا حد امکان از ViewData ، ViewBag و امثال آن استفاده نشده و به ازای هر View یک مدل متناظر (ViewModel) ایجاد شود.
- فایل پروژه برنامه توسط یک ادیتور متنی ویرایش شده و [MvcBuildViews](#) آن به True تنظیم شود.
- مدل‌های متناظر با جداول بانک اطلاعاتی نباید مستقیماً در Viewهای برنامه استفاده شوند.
- پوشه Models، از پروژه اصلی حذف شود. یک پروژه class library جدید به نام MyProjectName.Models برای نگهداری ViewModels ایجاد گردد.
- یک پروژه Class library دیگر به نام MyProjectName.DomainClasses برای نگهداری کلاس‌های متناظر با جداول بانک اطلاعاتی ایجاد شود.
- از سیستم [bundling و minification](#) ، برای یکی سازی اسکریپت‌ها و CSSهای برنامه استفاده شود.
- قسمت [custom errors](#) فایل web.config برنامه به نحو صحیحی مقدار دهی شود.
- تمام فرم‌های عمومی برنامه باید دارای [AntiForgeryToken](#) باشند.
- تمام فرم‌های عمومی برنامه باید [captcha](#) داشته باشند.
- پوشه‌های Content و Scripts از سیستم مسیریابی تعریف شده در Global.asax خارج شوند.
- Application_Start به MvcHandler.DisableMvcResponseHeader = true اضافه شود.
- اگر فقط از Razor به عنوان ViewEngine استفاده می‌شود، در Application_Start، باید سایر ViewEngineهای مورد استفاده، [حذف شوند](#) .
- فیلتر پیش فرض مدیریت خطاها حذف و بجای آن از [ELMAH](#) استفاده شود.
- در web.config، مقادیر executionTimeout و [maxRequestLength](#) مرتبط با httpRuntime تنظیم شوند. همچنین enableVersionHeader آن نیز خاموش گردد.
- استفاده از سشن‌ها کلاً باید حذف شود. ماژول توکار آن از قسمت httpModules [حذف گردد](#) تا پردازش موازی صفحات فعال گردد. (سشن مربوط است به دوران ASP کلاسیک دهه نود و هیچ نیازی به استفاده از آن در MVC نیست)
- در هیچ کنترلری نباید جزئیات پیاده سازی متدی مشاهده شود. تمام پیاده سازی‌ها باید به لایه سرویس‌های مختلف برنامه منتقل و از طریق تزریق وابستگی‌ها در دسترس باشند.
- اگر نیاز به مشخص سازی آدرسی در سایت است (خصوصاً در اسکریپت‌ها) باید از [Url.Action](#) استفاده شود و نه رشته‌ها.
- بهتر است [بومی سازی](#) برنامه از روز اول آن در نظر گرفته شده و تمام عبارات مورد استفاده در فایل‌های Resource درج شوند.
- برای مدیریت ساده‌تر بسته‌های مورد استفاده (وابستگی‌های برنامه) بهتر است از [NuGet](#) استفاده شود.
- از یک [ماژول HTTP compression](#) مستقل و با کیفیت استفاده شود (برای سازگاری بهتر با نگارش‌های مختلف IIS).
- برای معرفی HTTP modules و سادگی تعریف و فعال سازی آن‌ها در انواع و اقسام IISها بهتر است از کتابخانه [WebActivator](#) استفاده شود.
- امکان دوبار کلیک کردن بر روی تمام دکمه‌ها نباید وجود داشته باشد.
- از هش‌های ترکیبی استفاده شود. مستقیماً از MD5 یا SHA1 استفاده نشود.
- با اسکریپت‌های [anti IE6,7](#) ، این مرورگرها به رحمت ایزدی واصل شوند.
- اگر کاربری JavaScript را در مرورگر خود غیرفعال کرد، [نباید بتواند](#) از سایت استفاده کند.
- کلیه تغییرات تنظیمات و محتوای مهم سایت باید برای مدیر سایت بلافاصله ایمیل شوند.
- یک سری کارهای متداول مانند تهیه فایل‌های favicon.ico, apple-touch-icon-XXY.png, crossdomain.xml, robots.txt, sitemap.xml ([ترجیحاً پویا](#)) فراموش نشود.
- در web.config و در زمان ارائه، compilation debug=false [تنظیم شود](#) .
- در تمام قسمت‌هایی که AllowHtml فعال شده باید از [پاکسازی Html](#) دریافتی جهت مقابله با XSS مطمئن شد.

- جهت سهولت طراحی table less از یک فریم ورک CSS ایی [استفاده شود](#) .
- در تمام قسمت‌هایی که فایلی آپلود می‌شود باید بررسی شود فایل‌های نا امن (فایل‌های اجرایی ASP.NET) قابل آپلود [نباشند](#) .
- حین کار با بانک‌های اطلاعاتی یا از ORM استفاده شود و یا از [کوئری‌های پارامتری](#) .
- هر برنامه ASP.NET باید داری یک [Application pool](#) مجزا به همراه تنظیمات حافظه مشخصی باشد.
- تمام صفحات باید عنوان داشته باشند. به همین منظور مقدار دهی پیش فرض آن در فایل ViewStart صورت گیرد.
- در صفحه لاگین سایت، autocomplete خاموش شود.
- تمام delete‌های برنامه باید به HttpPost محدود شوند. تمام گزارشات و نمایش اطلاعات غیرعمومی برنامه به HttpGet محدود شوند.
- تعداد رفت و برگشت‌های به بانک اطلاعاتی در یک صفحه توسط پروفایلرها بررسی شده و اطلاعات عمومی پرمصرف [کش شوند](#) .
- در هیچکدام از کلاس‌های ASP.NET MVC نباید از HttpContext مستقیماً استفاده شود. کلاس پایه جدید آن باید مورد استفاده قرار گیرد یا از [Action Result](#) صحیحی استفاده گردد.
- کش کردن [فایل‌های استاتیک](#) در نظر گرفته شود.
- تمام درخواست‌های jQuery Ajax باید بررسی شوند. آیا واقعا مرتبط هستند به سایت جاری و [آیا واقعا Ajax ایی هستند](#) .

یک نکته:

امکان تهیه قالب‌های سفارشی VS.NET و لحاظ موارد فوق در آن جهت استفاده‌های بعدی نیز وجود دارد:

[Create Reusable Project And Item Templates For Your Development Team](#)

[Write Templates for Visual Studio 2010](#)

[Building a Custom Project Wizard in Visual Studio .NET](#)

نظرات خوانندگان

نویسنده: رضا

تاریخ: ۱۱:۱۴ ۱۳۹۱/۰۸/۱۹

یک چک لیست کامل. مثل همیشه عالی بود آقای نصیری. ممنون

نویسنده: محسن.د

تاریخ: ۱۱:۳۷ ۱۳۹۱/۰۸/۱۹

بسیار بسیار عالی بود .
ممنون

نویسنده: سروش ترک زاده

تاریخ: ۱۲:۵۹ ۱۳۹۱/۰۸/۱۹

سلام
به نظر من، شاید این بهترین مطلبی بود که توی وبسایت شما است...
ممنون
شاید جای مورد "تهیه Helperهای سفارشی" توی لیست خالی باشه...

نویسنده: علی

تاریخ: ۱۳:۲۸ ۱۳۹۱/۰۸/۱۹

عالی بود، اگه ممکنه در مورد ضرورت وجود apple-touch-icon-XxY.png بیشتر توضیح بدید.

نویسنده: امیرحسین جلوداری

تاریخ: ۱۳:۵۹ ۱۳۹۱/۰۸/۱۹

خواهش این گزینه‌ها رو بیشتر توضیح بدین :- در web.config، مقادیر executionTimeout و [maxRequestLength](#) مرتبط با httpRuntime تنظیم شوند. همچنین enableVersionHeader آن نیز خاموش گردد.

- true = MvcHandler.DisableMvcResponseHeader به Application_Start اضافه شود.

همچنین :

- استفاده از سشن‌ها کلا باید حذف شود. مازول توکار آن از قسمت httpModules [حذف گردد](#) تا پردازش موازی صفحات فعال گردد. (سشن مربوط است به دوران ASP کلاسیک دهه نود و هیچ نیازی به استفاده از آن در MVC نیست)

اگه قرار باشه یه eshop راه بندازیم چجوری بدون سشن باید سبد خریدو نگه داشت؟! (راه معادل چیه؟!)

نویسنده: محمد کریمی

تاریخ: ۱۵:۰ ۱۳۹۱/۰۸/۱۹

ممنون

بسیار عالی، امیدوارم باز هم از این گونه مطالب (کارهایی که خودتون در پروژه هاتون اعمال می‌کنید و انجام می‌دهید) در وب سایت درج کنید

نویسنده: وحید نصیری
تاریخ: ۱۶:۲۳ ۱۳۹۱/۰۸/۱۹

دستگاه‌های شرکت Apple برای نمایش آیکون زیبای مرتبط با یک سایت [از این نوع آیکون‌ها](#) استفاده می‌کنند. تهیه آن‌ها سبب می‌شود که تعداد خطاهای ثبت شده در لاگ سایت شما، کاهش یابند.

نویسنده: وحید نصیری
تاریخ: ۱۶:۲۵ ۱۳۹۱/۰۸/۱۹

- پیش فرض حجم مجاز قابل آپلود فایل‌ها در ASP.NET حدود 4 مگ است که کافی نیست. این رو باید بیشتر کنید. همچنین حجم فایل که بالا رفت نیاز است سرور زود timeout نزند.
- خاموش کردن enableVersionHeader و همچنین MvcHandler.DisableMvcResponseHeader سبب می‌شود تا روابطی باگ یاب سایت‌ها نتوانند دقیق عمل کنند چون شماره نگارش MVC و ASP.NET مورد استفاده مخفی می‌شود.
- از کوکی استفاده کنید. برای شروع: ([^](#))
در دوران ASP کلاسیک، از سشن عموماً برای مشخص سازی وضعیت لاگین افراد استفاده می‌شد. الان با وجود روش‌های مختلف اعتبارسنجی در ASP.NET هیچ ضرورتی به استفاده از آن نیست.

نویسنده: ایلیا اکبری فرد
تاریخ: ۸:۳۸ ۱۳۹۱/۰۸/۲۰

سلام. بهترین مقاله ای که تا به حال جایی دیدم. نمره من 100 از 5. همه چی تموم. چرا شما با بقیه فرق دارید؟

نویسنده: رضا
تاریخ: ۱۹:۵۱ ۱۳۹۱/۰۸/۲۰

سلام. میشه درباره خارج کردن فولدرهای Content و Script از سیستم مسیریابی راهنمایی بفرمایید؟ من که هر کار کردم نشد.

نویسنده: وحید نصیری
تاریخ: ۲۱:۲ ۱۳۹۱/۰۸/۲۰

به این صورت:

```
routes.IgnoreRoute("Content/{*pathInfo}");
routes.IgnoreRoute("Scripts/{*pathInfo}");
```

نویسنده: ایلیا اکبری فرد
تاریخ: ۲۳:۶ ۱۳۹۱/۰۸/۲۰

آقای نصیری با سلام.
برای MVC ورژن 4 کدام دیتاگرید رایگان را پیشنهاد میکنید که قابلیت سورت و صفحه بندی نیز داشته باشد؟ باتشکر فراوان.

نویسنده: وحید نصیری
تاریخ: ۲۳:۱۲ ۱۳۹۱/۰۸/۲۰

WebGrid توکار آن خوب است: ([^](#))
پروژه MVC Contrib هم گرید دارد: ([^](#))
به علاوه دست شما در MVC باز است. مثلاً از jqGrid هم می‌تونید استفاده کنید: ([^](#))

نویسنده: ایلیا اکبری فرد

تاریخ: ۲۳:۱۵ ۱۳۹۱/۰۸/۲۰

سپاس فراوان.
شب خوش.

نویسنده: امیرحسین مرجانی
تاریخ: ۱۹:۰۰ ۱۳۹۱/۰۸/۲۲

سلام آقای نصیری

با این مطلب دیگه حجت رو بر ما تموم کردی.

واقعا ممنونم خیلی عالی بود

اگر منابع انگلیسی از نوع همین چک لیست می‌شناسید لطفا معرفی کنید؟

نویسنده: کیارش سلیمان زاده
تاریخ: ۲۳:۲۳ ۱۳۹۱/۰۸/۲۲

سلام

" استفاده از سشن‌ها کلا باید حذف شود. ماژول توکار آن از قسمت `httpModules` [حذف گردد](#) تا پردازش موازی صفحات فعال گردد. (سشن مربوط است به دوران ASP کلاسیک دهه نود و هیچ نیازی به استفاده از آن در MVC نیست) "

در این صورت:

- Captcha ها که از سشن استفاده میکنن دیگه کار نمیکنن! راه حل منطقی چیه؟

- TempData کار میکنه یا این امکان هم از کار میفته؟

ممنون.

نویسنده: وحید نصیری
تاریخ: ۲۳:۵۸ ۱۳۹۱/۰۸/۲۲

- از Captcha ای استفاده کنید که طراحی به روزی داشته باشد و نیازی به سشن نداشته باشد. مانند همان نمونه‌ای که در متن فوق لینک داده شده. هم `Cookie StorageProvider` دارد و هم `SessionStorageProvider`.
- نیازی به TempData نیست و با استفاده صحیح از امکانات Routing می‌شود آنرا پوشش داد. خیلی از عناصر جهت اطلاع هستند، نه جهت استفاده.

کسانی که از سشن استفاده می‌کنند:

- نمی‌دانند ASP.NET دارای سیستم اعتبار سنجی پیشرفته‌ای است.

- نمی‌دانند روش‌های دیگری نیز برای مدیریت حالت وجود دارد، مانند استفاده از کوکی‌های رمزنگاری شده، استفاده از کوئری استرینگ‌های رمزنگاری شده و کلا در مورد استفاده از امکانات پیشرفته سیستم Routing بی‌خبرند.

نویسنده: صابر فتح الهی
تاریخ: ۱:۸ ۱۳۹۱/۰۸/۲۳

سلام

می‌تونن سبد خرید توی Profile نگهداری کنی، با قطع برق و اتصال اینترنت و ... راحت قابل بازیابی هست.

نویسنده: محمد صاحب
تاریخ: ۱۴:۳۳ ۱۳۹۱/۰۸/۲۳

[راه معادل](#) پیاده سازی سبد خرید.

نویسنده: sdg

تاریخ: ۱۵:۲۵ ۱۳۹۱/۰۸/۲۴

مورد سوم لینک [MvcBuildViews](#) درسته آیا ؟!

نویسنده: وحید نصیری

تاریخ: ۱۶:۱۲ ۱۳۹۱/۰۸/۲۴

بله. مطلب انتهایی لینک داده شده را تحت عنوان «نکته‌ای تکمیلی در مورد Strongly typed view ها»، مطالعه کنید:
 «... پاسخ: بله. باید فایل پروژه را اندکی ویرایش کرده و مقدار `MvcBuildViews` را که به صورت پیش فرض `false` هست، `true` نمود...»

نویسنده: امیرحسین مرجانی

تاریخ: ۲:۲ ۱۳۹۱/۰۸/۳۰

سلام

برای مدیریت سبد خرید حتما به پروژه MusicStore رو ببینید.

خیلی می‌تونه بهتون کمک کنه.

نویسنده: حسین

تاریخ: ۱۱:۳۵ ۱۳۹۱/۰۹/۱۱

سلام.

در مورد نکته سوم " مدل‌های متناظر با جداول بانک اطلاعاتی نباید مستقیماً در View های برنامه استفاده شوند. " می‌خواستم بدونم این کار حتی برای نشان دادن یک لیست (مثلاً در Index) یا نشان دادن جزئیات (صفحه Details) هم لازم هستش یا اینکه فقط در فرم‌هایی که کاربر اطلاعات وارد میکنه مثل Create و Edit و ... ممنون.

نویسنده: وحید نصیری

تاریخ: ۱۱:۴۵ ۱۳۹۱/۰۹/۱۱

- در فرم‌هایی که اطلاعاتی را به سرور Post می‌کنند الزامی است (خصوصاً از لحاظ مسایل امنیتی)
 - در گزارشات ... «بهتر» است اعمال شود. مثلاً با کم کردن تعداد فیلدها به تعدادی که نمایش داده می‌شوند، می‌توان مصرف حافظه برنامه را کاهش داد. شاید یک جدول شما 20 خاصیت داشته باشد و در یک گزارش فقط 5 مورد آن نمایش داده شود. 15 مورد دیگر صرفاً مصرف کننده حافظه خواهند شد اگر از `viewModel` استفاده نکنید. (ولی ... در کل بهتر است اینگونه باشد)

نویسنده: کیارش سلیمان زاده

تاریخ: ۱۶:۵ ۱۳۹۱/۱۱/۰۹

چطور میشه از UI تعیین کرد که Service Layer چه فیلدهایی رو از دیتابیس واکشی کنه (جلوگیری از واکشی فیلدهایی که نیاز نیستن)؟ راه حل پیشنهادی شما چیه؟

نویسنده: وحید نصیری

تاریخ: ۱۶:۳۸ ۱۳۹۱/۱۱/۰۹

لایه UI مصرف کننده است؛ تعیین کننده نیست و نباید حاوی منطقی بجز نمایش اطلاعات باشد. طراحی و بازگشت فیلدهای مورد نیاز باید در لایه سرویس به صورت مقید و واضحی انجام شود.

نویسنده: کیارش سلیمان زاده
تاریخ: ۱۸:۹ ۱۳۹۱/۱۱/۰۹

فرض کنید که در UI فقط به نام کاربر و آدرس ایمیل کاربر احتیاج است، کلاس کاربر در در Domain به شرح زیر است:

```
public class User
{
    public int UserId {get;set;}
    public string Name {get;set;}
    public string Family {get;set;}
    public string Web {get;set;}
    public string Email {get;set;}
    public DateTime RegisterDate {get;set;}
}
```

حالا در یک کنترلر فقط به نام کاربر و آدرس ایمیل کاربر احتیاج است، حالا میایم یک کلاس تعریف میکنیم که شامل فیلدهای مورد نیاز است:

```
public class UserInfoViewModel
{
    public string Name {get;set;}
    public string Family {get;set;}
    public string Email {get;set;}
}
```

و لایه سرویس یک متد منحصر بفرد مختص این کار می نویسیم، مثلاً:

```
public UserInfoViewModel GetMemberByUserName(string username)
{
    var result = from u in _users
                  where u.UserName == username
                  select new UserInfoViewModel() {Name = u.Name, Family=u.Family, Email=u.Email};

    return result;
}
```

1-اگر این روش درست، حالا:

- برای هر متد که نیاز به سری فیلد مورد نیاز رو برگردونه، **یه کلاس جداگانه** باید تعریف کرد؟ (در اینجا UserInfoViewModel)

- این کلاس UserInfoViewModel باید جز ViewModel های لایه UI باشه و در لایه Models قرار بگیره؟
در اینصورت (کلاس UserInfoViewModel باید جز ViewModel های لایه UI باشه)، لایه سرویس وابسته به لایه UI نمیشه؟

اگر جواب منفیه،

کلاس UserInfoViewModel تو کدوم لایه باید قرار بگیره؟

دیگه نباید پسوند ViewModel رو به این کلاس اضافه کرد، درسته؟

2-اگر این روش کلاً اشتباهه، راه حل شما دقیقاً چیه؟

نویسنده: وحید نصیری
تاریخ: ۱۸:۴۳ ۱۳۹۱/۱۱/۰۹

- بله. کار اصولی انجام دادن نیاز به کدنویسی بیشتری داره. اما در دراز مدت نگهداری ساده‌تری خواهد داشت چون حد و مرزها مشخص است و تغییرات سیستم جزئی‌تر خواهند بود با به هم ریختن حداقل قسمت‌های دیگر که به آن گره نخورده‌اند.

- ViewModel تغذیه کننده لایه UI است و نه اینکه با آن یکی باشد. اگر علاقمند هستید که وابستگی به UI رو بدونید مراجعه کنید به Web form ها و جاییکه ارجاع مستقیمی از یک TextBox داخل کدهای برنامه (code behind) است. به این وابستگی می‌گن و نه حالت دیگری.

- در مورد محل قرارگیری Models و ViewModels در ابتدای متن فوق توضیح داده شده.

نویسنده: کیارش سلیمان زاده
تاریخ: ۱۳۹۱/۱۱/۰۹ ۱۹:۳۰

محل قرارگیری UserInfoViewModel در قسمت ViewModel درست‌ه؟ یا تمام کلاس‌های بازگشتی از لایه سرویس باید در قسمت دیگری در لایه Models، و یا لایه دیگری قرار بگیرند؟

نویسنده: وحید نصیری
تاریخ: ۱۳۹۱/۱۱/۰۹ ۲۳:۳۰

بله. لایه سرویس هم می‌تونه domain model بازگشت بده و هم view model. در عموم کارهای میکروسافت این دو بجای هم بکار رفتن و یکی هستند. ما اینجا جداشون کردیم.

نویسنده: ابراهیم
تاریخ: ۱۳۹۱/۱۱/۱۰ ۹:۳۶

سلام؛

لایه سرویس باید مستقل از presentation باشد. یکی از اهداف از طراحی سرویس گرا اینست که سرویس ارایه شده قابل استفاده در clientها متفاوت باشد. لذا نباید به viewmodel وابستگی داشته باشد. بنظر می‌رسد که راه درست این است که سرویس domain model یا DTO برگرداند و در کنترلر با استفاده از automapper یا مپینگ دستی viewmodel ساخته شود و به view ارسال شود.

هرچند طراحی کاملا سلیقه ای و مربوط به دامنه کاربرد است و نمی‌توان گفت که غلط است

با تشکر از سایت بسیار مفیدی که ایجاد کرده اید. م

نویسنده: وحید نصیری
تاریخ: ۱۳۹۱/۱۱/۱۰ ۹:۵۳

- لایه سرویس وابستگی به View نداره. هیچ ارجاع مستقیمی از عناصر بصری را در لایه سرویس نخواهید یافت. ViewModelها در MVC یک سری کلاس ساده دارای خاصیت ... خاصیت ... خاصیت بیشتر نیستید. این سرویس ارائه شده چون با کلاس‌های مدل یا ViewModel ساده‌ای کار می‌کند که ارجاعی از عناصر بصری در آن نیست، در همه جا قابل استفاده است. چه View وجود داشته باشد یا خیر. کاملا مستقل از آن است و در نهایت View را تغذیه خواهد کرد؛ اما ارتباط یک طرفه است و نه دو طرفه. عدم وابستگی به لایه بصری را در Web forms بهتر می‌شود توضیح داد. زمانیکه شما TextBox1.Text را در کدهای خود دارید، این به معنای وابستگی مستقیم است به عناصر بصری؛ نه حالتیکه یک کلاس ساده ViewModel دارای خاصیت ... خاصیت ... خاصیت را دارید که روح آن از وجود TextBox1 بی‌خبر است. - در مورد بازگشت کلی domain model هم بحث شد؛ کمی بالاتر، خصوصا در گزارشات. این مساله سبب می‌شود که شما 20 فیلد را بازگشت داده و سپس در سمت کلاینت 3 مورد آن را نمایش دهید. به این ترتیب مصرف حافظه بالاتری را خواهید داشت. چون در ابتدای کار باید select تهیه شده در پشت صحنه هر چه هست و نیست را واکنشی کند. سپس قصد دارید آنرا به ViewModel نگاشت کنید. این نگاشت بهتر است همان ابتدای کار در select تهیه شده صورت گیرد؛ به این ترتیب واکنشی از دیتابیس مثلا فقط به سه فیلد مورد نیاز محدود خواهد شد (با سرعت بیشتر و مصرف حافظه کمتر).

نویسنده: مسعود مشهدی
تاریخ: ۱۳۹۱/۱۱/۱۲ ۱۷:۴۰

با سلام جناب نصیری

عموما برای سادگی کار و تفکیک پذیری یک برنامه MVC پروژه به لایه هایی تقسیم بندی می‌شود لطفا در مورد لایه‌های زیر یک توضیح کلی بدهید و اینکه معمولا در این لایه‌ها چه قسمت هایی از پروژه قرار می‌گیرند :

Application.Web
Application.Manager

Application.Repository

Application.DAL

Application.DTO

Application.Core

Application.Common

و اینکه لایه‌های DomainClasses و DataLayer در دل این لایه‌ها قرار می‌گیرند یا بصورت لایه‌های مجزا هستند. با تشکر

نویسنده: وحید نصیری
تاریخ: ۱۷:۴۶ ۱۳۹۱/۱۱/۱۲

سلام؛ لطفاً برای توضیحات بیشتر مراجعه کنید به قسمت 12 سری EF Code first. یک مثال کامل به همراه ارائه معماری قابل استفاده در MVC در آن مطرح شده است و کدهای آن نیز قابل دریافت است.

نویسنده: ایلینا اکبری فرد
تاریخ: ۱۷:۳ ۱۳۹۱/۱۱/۱۵

با سلام.
اگر اطلاعات کاربری که وارد سیستم شده، علاوه بر کد کاربری، شامل موارد دیگری مانند کد مرکز، کد ترم و ... نیز نیاز باشد و نخواهیم از session استفاده کنیم، بهترین روش برای نگهداری این اطلاعات در طول یک session به نظر شما چیست؟ متشکرم.

نویسنده: وحید نصیری
تاریخ: ۱۷:۷ ۱۳۹۱/۱۱/۱۵

از کوکی رمزنگاری شده.

نویسنده: سعید
تاریخ: ۱۲:۲۵ ۱۳۹۱/۱۱/۱۹

"پوشه Models، از پروژه اصلی حذف شود. یک پروژه class library جدید به نام MyProjectName.Models برای نگهداری ViewModels ایجاد گردد."

پروژه class library جدید که برای Model درست می‌کنیم [مورد پنجم]، همیشه اسمبلی System.Web.Mvc و خیلی رفرنسهای دیگه رو بهش اضافه کرد و میننده به خطا! فرضاً می‌خواهم AccountModels.cs پیش فرض رو منتقل کنم به class library جدید؛ به مشکل رفرنس‌ها برمخورم؛ ویژوال استودیو 2012 و پروژه MVC 4

ضمناً به سوال دیگه هم از خدمت شما داشتم، بنظرتون درسته بریم سمت MVC 4 یا فعلاً برامون در انتشار پروژه مشکل ایجاد میکنه؟ مثلاً هاست‌های کمی ممکنه دات نت 4.5 یا MVC 4 رو ساپورت کنن و ...
ممنون از زحماتی که میکشید، مخصوصاً از این چک لیست کامل ...

نویسنده:

وحید نصیری

تاریخ:

۱۲:۳۱ ۱۳۹۱/۱۱/۱۹

- حالت پیش فرض Target Framework Version پروژه‌های دات نت 4 بر مبنای Client profile است. یعنی دسترسی به امکانات وب در آن وجود ندارد. برای اصلاح آن به خواص پروژه مراجعه کرده و Client profile یاد شده را به Full profile تغییر بدید تا بتوانید ارجاعات لازم به اسمبلی‌های مرتبط با پروژه‌های وب را اضافه کنید. چیزی شبیه به این در فایل پروژه برنامه (csproj) با تمام مداخل لازم دیگر که بیشتر جهت تعاریف DataAnnotations مورد نیاز هستند تا هر کاربرد دیگری:

```
<ItemGroup>
  <Reference Include="EntityFramework, Version=4.4.0.0, Culture=neutral,
  PublicKeyToken=b77a5c561934e089, processorArchitecture=MSIL">
    <SpecificVersion>False</SpecificVersion>
    <HintPath>..\packages\EntityFramework.5.0.0\lib\net40\EntityFramework.dll</HintPath>
  </Reference>
  <Reference Include="System" />
  <Reference Include="System.ComponentModel.DataAnnotations" />
  <Reference Include="System.Core" />
  <Reference Include="System.Web" />
  <Reference Include="System.Web.Mvc, Version=4.0.0.0, Culture=neutral,
  PublicKeyToken=31bf3856ad364e35, processorArchitecture=MSIL">
    <SpecificVersion>False</SpecificVersion>
    <HintPath>C:\Program Files\Microsoft ASP.NET\ASP.NET MVC
  4\Assemblies\System.Web.Mvc.dll</HintPath>
  </Reference>
  <Reference Include="System.Xml.Linq" />
  <Reference Include="System.Data.DataSetExtensions" />
  <Reference Include="Microsoft.CSharp" />
  <Reference Include="System.Data" />
  <Reference Include="System.Xml" />
</ItemGroup>
```

- MVC4 بر مبنای دات نت 4 تهیه شده و نه دات نت 4 و نیم. به این معنا که به راحتی حتی بر روی ویندوز سرور 2003 سرویس پک (و ویندوزهای بعدی) قابل نصب است.

نویسنده:

ایلیا اکبری فرد

تاریخ:

۱۶:۵۴ ۱۳۹۱/۱۱/۲۵

با سلام.

امکات دارد که برای قسمت " امکان دوبار کلیک کردن بر روی تمام دکمه‌ها نباید وجود داشته باشد " مثالی بزنید؟
باتشکر.

نویسنده:

وحید نصیری

تاریخ:

۱۸:۳ ۱۳۹۱/۱۱/۲۵

» [استفاده از دکمه‌های CSS توئیت در ASP.NET MVC - قسمت دوم](#) «

نویسنده:

حسام محمدی

تاریخ:

۱۸:۷ ۱۳۹۱/۱۱/۲۷

سلام. می‌خواستم بدونم کار درستی که کلاس‌ها و اینترفیس‌های مربوط به Service Layer (برای کار با دیتابیس) رو داخل DomainClasses بذاریم یا بهتره که یک پروژه دیگه برای لایه سرویس درست کرد؟

نویسنده:

میهمان

تاریخ:

۲۰:۹ ۱۳۹۱/۱۱/۲۷

با سلام

خیر ، فکر نمیکنم کار درستی باشد . لایه سرویس خود باید به صورت لایه ای مجزا باشد
باید توجه داشته باشیم که ما میتوانیم تمام کلاس‌های سرویس ، DomainClasses و ... را در همان پروژه اصلی داشته
باشیم(منظور همان پروژه‌های است که UI درون آن میباشد) ولی اگر بخواهیم در آینده آن را گسترش دهیم ، با حجم عظیمی از

کدها و کلاسها مواجه هستیم . پس بهتر است تمامی قسمت‌ها به صورت مجزا (در لایه‌های مجزا) تعریف شود

نویسنده: مهدی سعیدی فر
تاریخ: ۸:۴۱ ۱۳۹۲/۰۲/۰۹

سلام

می‌خواستم ببینم این مازول httpcompression را چگونه تو وب سایت خود فعال کردید؟
من همه‌ی کارهایی که تو سایت معرفی شده گفته بود کردم ولی باز کار نکرد. (نکته خاصی داره؟)
من از iis 7.5 استفاده می‌کنم ولی gzip فعال نیست و وقتی با وب سرور تماس می‌گیرم می‌گه فقط برای سرورهای لینوکس به
طور پیش فرض فعال هست!.
ممنون میشم کمک کنید.

نویسنده: وحید نصیری
تاریخ: ۹:۳۷ ۱۳۹۲/۰۲/۰۹

من یک مقدار [کدهاش رو](#) برای اعمال تنظیمات تغییر دادم: [HttpCompress.zip](#)

نویسنده: حسین
تاریخ: ۱۷:۳۵ ۱۳۹۲/۰۲/۱۵

سلام. در مورد " تمام گزارشات و نمایش اطلاعات غیرعمومی برنامه به HttpGet محدود شوند " یعنی برای actionها حتماً
[HttpGet] رو ذکر کنیم؟ چون به صورت پیش فرض اگر ذکر نکنیم هم HttpGet در نظر گرفته خواهد شد.
ممنون.

نویسنده: وحید نصیری
تاریخ: ۱۷:۵۳ ۱۳۹۲/۰۲/۱۵

نه. اگر ذکر نشه به درخواست‌های از نوع Head هم مثلاً جواب می‌ده. [اطلاعات بیشتر](#)

نویسنده: hossein101211
تاریخ: ۱۶:۱۷ ۱۳۹۲/۰۳/۰۷

با سلام همونطوری که فرمودید که کمتر از VIEW BAG یا VIEW DATA استفاده کنیم
من قبلاً برای ایجاد صفحه محصول شاخه‌های محصول رو در view bag قرار میدادم و میخوندم
آیا با view model میشه اینکار رو کرد؟

نویسنده: وحید نصیری
تاریخ: ۱۶:۲۵ ۱۳۹۲/۰۳/۰۷

بله. مراجعه کنید به مطلب « [نحوه استفاده از ViewModel در ASP.NET MVC](#) »

نویسنده: علی پائیزی
تاریخ: ۱۳:۲۶ ۱۳۹۲/۰۴/۱۸

با سلام و خسته نباشید
آقای نصیری بی زحمت در مورد:
در هیچ کنترلری نباید جزئیات پیاده سازی متدی مشاهده شود. تمام پیاده سازیها باید به لایه سرویسهای مختلف برنامه منتقل و
از طریق تزریق وابستگیها در دسترس باشند.
بیشتر توضیح دهید.
متشکرم

نویسنده: وحید نصیری
تاریخ: ۱۳۹۲/۰۴/۱۸ ۱۳:۳۳

مراجعه کنید به دوره « [بررسی مفاهیم معکوس سازی وابستگی ها و ابزارهای مرتبط با آن](#) »

نویسنده: ایلینا اکبری فرد
تاریخ: ۱۳۹۲/۰۴/۲۷ ۲۰:۵۵

با سلام.

آیا می شود قابلیت این کتابخانه را در کل پروژه بصورت خودکار فعال کرد؟ سپاس.

نویسنده: وحید نصیری
تاریخ: ۱۳۹۲/۰۴/۲۷ ۲۰:۲۱

HTTP Module استاندارد ASP.NET است و بر روی کل سایت تاثیر دارد. روش فعال سازی آن در وب فرم ها یا MVC تفاوتی نمی کند و یکی است.

مراجعه کنید به سایت اصلی [سازنده آن](#) ، مثال به همراه وب کانفیگ تنظیم شده دارد؛ [یا پروژه IRIS](#) هم از این ماژول استفاده می کند.

نویسنده: اکبر
تاریخ: ۱۳۹۲/۰۷/۰۷ ۷:۳۹

با سلام.

من طبق راهنمایی شما ماژول سشن را از وب کانفیگ حذف کردم، حالا رویداد سشن استارت من کار نمی کند، چون محاسبه آمار افراد آنلاین را در این رویداد قرار دارد، در مورد محاسبه تعداد افراد آنلاین کمی راهنمایی میکنید؟

نویسنده: وحید نصیری
تاریخ: ۱۳۹۲/۰۷/۰۷ ۱۰:۴۵

کاربری آنلاین هست که حداقل یک EndRequest داشته باشد. در اینجا اطلاعات IP او را به همراه زمان جاری، در یک ConcurrentDictionary استاتیک ثبت کنید. در یک تایمر استاتیک، کسانی را که در 5 دقیقه قبل رکورد به روز شده ای نداشته اند از این دیکشنری حذف کنید.

نویسنده: ایمان اسلامی
تاریخ: ۱۳۹۲/۰۷/۲۹ ۹:۴۵

با سلام؛ میشه راجع به این مطلب توضیح بیشتری بدید. برای کاربران آنلاین عضو و تفکیک آنها از کاربران آنلاین مهمان چکار کنیم و اینکه میشه برای تایمر استاتیک ، از quartz استفاده کرد؟ به عنوان یه جاب با 5 interval دقیقه ای.

نویسنده: وحید نصیری
تاریخ: ۱۳۹۲/۰۷/۲۹ ۱۰:۳۶

- Request.IsAuthenticated یک IP را در روال رخداد گردان AuthenticateRequest و یا PostAuthenticateRequest بررسی کنید.

- از هر روشی که علاقمندید برای تایمر استفاده کنید. من به روش های ساده و توکار برای انجام کارهای ساده در یک HTTP Module نصف صفحه ای علاقمندم.

نویسنده: ایمان اسلامی

تاریخ: ۱۳۹۲/۰۷/۲۹ ۱۲:۱۵

با تشکر از پاسخ شما ، فقط یک مطلب
با توجه به دینامیک بودن IP ، آیا Authenticate کردن با IP معتبر است؟

نویسنده: وحید نصیری
تاریخ: ۱۳۹۲/۰۷/۲۹ ۱۲:۴۸

- Forms authentication با IP کار نمی‌کند. بر اساس کوکی‌های رمزنگاری شده خاص خودش (مثلا در حالت بخاطر سپاری کلمه عبور) کاربر را بلافاصله اعتبارسنجی می‌کند.
- نکته Request.IsAuthenticated رو گرفتی؟ پخته‌اش کن. مثلا بجای IP از نام یوزر که در Request کاربر اعتبارسنجی شده موجود است، استفاده کن. هدف ذکر یک نکته بود. هدف مطلب جاری ارائه راه حلی جامع برای شمارش مساله خطیر تعداد کاربران آنلاین نبود که مثلا بین 25 با 30 آن واقعا تفاوتی وجود دارد یا نه. اصلا مهم است یا نه.
من در این سایت این تعداد عدد رو برای 15 دقیقه کش می‌کنم. یعنی اصلا به روز نیست و لحظه‌ای محاسبه نمیشه و نمایش داده نخواهد شد. چرا؟ چون یک کاربرد مهم و اساسی نیست در سایت. فقط یک نمایش کلی است. فقط یک حس حضور در سایت هست؛ یعنی هستند چند نفر خواننده این دور و اطراف.

نویسنده: حسن دهیاری
تاریخ: ۱۳۹۲/۰۷/۳۰ ۱۷:۱۱

ضمن تشکر از مطالب خوبتون
آقای نصیری من به ASP.NET MVC علاقه پیدا کرده ام و در نظر دارم پروژه ای را با این تکنولوژی شروع کنم. قبلا هم دو تا وب سایت با ASP.NET Web Form نوشته ام اما از وقتی با این تکنولوژی آشنا شده ام هدفم پیشرفت در این زمینه می‌باشد. اکثر مطالب شما را خوانده ام. اما اگر می‌شد مواردی را که در این چک لیست ذکر کرده اید با یک پروژه عملی توضیح داده شود از نظر یادگیری و در ک مطلب خیلی مفید واقع خواهد شد. ضمنا ممنون می‌شم موارد زیر را با مثال توضیح دهید
«مدل‌های متناظر با جداول بانک اطلاعاتی نباید مستقیما در Viewهای برنامه استفاده شوند.
- پوشه Models، از پروژه اصلی حذف شود. یک پروژه class library جدید به نام MyProjectName.Models برای نگهداری ViewModels ایجاد گردد.
- یک پروژه Class library دیگر به نام MyProjectName.DomainClasses برای نگهداری کلاس‌های متناظر با جداول بانک اطلاعاتی ایجاد شود. در هیچ کنترلی نباید جزئیات پیاده سازی متدی مشاهده شود. تمام پیاده سازی‌ها باید به لایه سرویس‌های مختلف برنامه منتقل و از طریق تزریق وابستگی‌ها در دسترس باشند.
- در صفحه لاگین سایت، autocomplete خاموش شود.
- تعداد رفت و برگشت‌های به بانک اطلاعاتی در یک صفحه توسط پروفایلرها بررسی شده و اطلاعات عمومی پرمصرف [کش شوند](#).
- هیچکدام از کلاس‌های ASP.NET MVC نباید از HttpContext مستقیما استفاده شود.
- تمام درخواست‌های jQuery Ajax باید بررسی شوند»

نویسنده: وحید نصیری
تاریخ: ۱۳۹۲/۰۷/۳۰ ۱۷:۵۹

- مراجعه کنید به [پروژه IRIS](#) . اغلب این best practices در آن لحاظ شده است.
- در مورد تزریق وابستگی‌ها و لایه بندی، مراجعه کنید به [دوره مخصوص آن در سایت](#) .
- در مورد پروفایلرها [مطلب هست در سایت](#) به اندازه کافی.
- در مورد مسایل امنیتی، در لابلای [دوره MVC](#) سایت این مباحث مطرح شدند. باید وقت بگذارید یکبار تمام قسمت‌ها را مطالعه کنید.