

اگر پیشتر سایتی را در آدرس مشخصی در اینترنت داشته‌اید و اکنون تنها نرم افزار آن تغییر کرده است، اما نحوه ارائه خدمات آن خیر، لازم است بتوانید شرایط ذیل را مدیریت کنید:

- موتورهای جستجو مدام اطلاعات قبلی خود را به روز می‌کنند. اگر آدرس قبلی مقاله‌ای در سایت شما `http://site/year/month/day/title` بوده، برای نمونه گوگل هر از چندگاهی مجدداً به این آدرس مراجعه می‌کند تا حداقل مطمئن شود وجود خارجی دارد یا خیر (این نکته را از لاگ‌های خطای سایت استخراج کردم).
- سایت‌های زیادی هستند که پیشتر به سایت شما و مطالب آن لینک داده‌اند. نمی‌توانید از آن‌ها درخواست کنید لطفاً بانک اطلاعاتی خود را به روز کنید.
- اگر فید قبلی سایت شما `http://site/feeds/posts` بوده و اکنون چیز دیگری است، باز هم نمی‌توانید از همه درخواست کنید اطلاعات خود را به روز کنند. عده‌ای اینکار را خواهند کرد و تعداد زیادی هم خیر.

برای مدیریت یک چنین مواردی می‌توان از امکانات مسیریابی موجود در ASP.NET MVC استفاده کرد؛ که نمونه‌ای عملی از آن‌را جهت سازگاری سایت جاری با هاست قبلی آن (بلاگر) در ادامه مطالعه خواهید نمود:

الف) سازگار سازی لینک‌های قدیمی برچسب‌های سایت با ساختار جدید آن

در بلاگر آدرس‌های برچسب‌ها، به صورت `http://site/search/label/name` تعریف شده است. در سایت جاری برچسب‌ها توسط کنترلر Tag مدیریت می‌شوند. برای هدایت آدرس‌های قدیمی (موجود در موتورهای جستجو یا ثبت شده در سایت‌هایی که به ما لینک داده‌اند) می‌توان از تعریف مسیریابی ذیل در فایل `global.asax` استفاده کرد:

```
routes.MapRoute(
    "old_blogger_tags_list", // Route name
    "search/label/{name}", // URL with parameters
    new { controller = "Tag", action = "Index", name = UrlParameter.Optional, area = "" }
    // Parameter defaults
);
```

به این ترتیب به صورت خودکار تمامی آدرس‌های شروع شده با `http://site/search/label` پالایش شده و سپس قسمت `name` آن‌ها جدا سازی می‌شود. این نام به متدی به نام `Index` در کنترلر `Tag` که دارای پارامتری به نام `name` است ارسال خواهد شد.

ب) از دست ندادن خوانندگان قدیمی فیدهای سایت

دو نوع فید کلی در بلاگر وجود دارد: `http://site/feeds/posts/default` و `http://site/feeds/comments/default`؛ اما در سایت جاری فیدها توسط کنترلری به نام `Feed` ارائه می‌شوند. برای سازگار سازی آدرس‌های قدیمی و هدایت آن‌ها به صورت خودکار به کنترلر فید می‌توان از دو تعریف مسیریابی ذیل استفاده کرد:

```
routes.MapRoute(
    "old_blogger_posts_feeds_list", // Route name
    "feeds/posts/default", // URL with parameters
    new { controller = "Feed", action = "Posts", name = UrlParameter.Optional, area = "" }
    // Parameter defaults
);

routes.MapRoute(
    "old_blogger_comments_feeds_list", // Route name
    "feeds/comments/default", // URL with parameters
    new { controller = "Feed", action = "Comments", name = UrlParameter.Optional, area = "" }
    // Parameter defaults
);
```

در اینجا دو آدرس ذکر شده به کنترلر `Feed` و متدهای `Posts` و `Comments` آن هدایت خواهند شد و به این نحو کاربران قدیمی سایت هیچگونه تغییری را احساس نکرده و باز هم فیدخوان‌های آن‌ها، بدون مشکل کار خواهند کرد.

ج) پردازش لینک‌های قدیمی مطابق سایت و هدایت آن‌ها به آدرس‌های جدید

این مورد اندکی مشکل‌تر از موارد قبلی است:

```
routes.MapRoute(
    "old_blogger_post_urls",
    "{yyyy}/{mm}/{title}",
    new { controller = "Post", action = "OldBloggerLinks" },
    new { yyyy = @"\d{4}", mm = @"\d{1,2}" }
);
```

برای نمونه آدرس مقاله‌ای مانند <http://site/2012/05/ef-code-first-15.html> را در نظر بگیرید. سه قسمت سال، ماه و عنوان آن، حائز اهمیت هستند. این‌ها را در اینجا به کنترلر Post و متد OldBloggerLinks هدایت خواهیم کرد. همچنین برای سال و ماه آن نیز قید تعریف شده است. سال عددی 4 رقمی است و ماه عددی یک تا دو رقمی. کدهای متد OldBloggerLinks را در اینجا مشاهده می‌کنید:

```
public virtual ActionResult OldBloggerLinks(int yyyy, int mm, string title)
{
    var oldUrl = string.Format(CultureInfo.InvariantCulture,
        "http://www.dotnettips.info/{0}/{1}/{2}", yyyy, mm.ToString("00"), title);
    var blogPost = _blogPostsService.FindBlogPost(oldUrl);
    if (blogPost != null)
        return RedirectToActionPermanent(actionName: ActionNames.Index, controllerName:
            MVC.Post.Name, routeValues: new { id = blogPost.Id, name =
                blogPost.Title.GetPostSlug() });
    return this.Redirect("/");
}
```

در اینجا چون ساختار لینک‌ها کلاً تغییر کرده است، ابتدا بر اساس پارامترهای دریافت شده، لینک قدیمی بازسازی می‌شود. سپس به بانک اطلاعاتی مراجعه شده و لینک قدیمی به همراه شماره مطلب مرتبط با آن یافت می‌شود (یک فیلد oldUrl برای مطالب قدیمی در بانک اطلاعاتی وجود دارد). در آخر هم به کمک متد RedirectToActionPermanent آدرس رسیده به آدرس جدید مطلب در سایت ترجمه و هدایت خواهد شد. Permanent بودن آن برای به روز رسانی خودکار اطلاعات موتورهای جستجو مفید است.

نتیجه گیری

به کمک امکانات مسیریابی توکار ASP.NET MVC می‌توان ساختار قدیمی یک سایت را به ساختار جدید آن ترجمه کرد. به این ترتیب لینک‌های قدیمی ثبت شده در صدها سایت اینترنتی که به سایت ما اشاره می‌کنند، مجدداً بدون مشکل قابل استفاده بوده و همچنین موتورهای جستجو نیز امکان به روز رسانی اطلاعات خود را خواهند یافت.