

The **Kentico.Web.Mvc integration package** includes a feature that allows you to return the same '404 Not Found' page in the MVC application in the following scenarios:

- A suitable controller is not found
- A suitable controller action is not found
- A controller action returns *System.Web.Mvc.HttpNotFoundResult* (usually returned by calling the controller's *HttpNotFound* method)

## Prerequisite

The `<modules>` element in the `<system.webServer>` section of your MVC project's main *web.config* file must have the **runAllManagedModulesForAllRequests** attribute set to *true*. The attribute is automatically added by installing the Kentico.Web.Mvc [integration package](#).

```
<system.webServer>
...
  <modules runAllManagedModulesForAllRequests="true"></modules>
...
</system.webServer>
```

## Enabling the global Not Found handler in MVC applications

1. If necessary, set up your Kentico instance and MVC application (see [Starting with MVC development](#)).
2. Verify that the Not Found handler feature is enabled in the MVC project's **ApplicationConfig.cs** file:

```
using Kentico.Web.Mvc;

...

public static void RegisterFeatures(ApplicationBuilder builder)
{
    ...
    builder.UseNotFoundHandler();
    ...
}
```

3. Create a **NotFound.cshtml** view under the *Views\Shared* path in your MVC project.
4. Implement the content of the view to design your site's Not found page.

Now whenever a request for a page that does not match any controller or controller action is made, the system returns the *NotFound.cshtml* page. The system also returns the page whenever a controller action returns the *System.Web.Mvc.HttpNotFoundResult* class (for example, when calling the *System.Web.Mvc.Controller.HttpNotFound* method).

## Handling requests not matched by any route

You can also return the *NotFound.cshtml* view in other cases in which the MVC application responds with a 'Not Found' status. To cover a scenario when a request that does not match any configured route is made, you need to implement a catch-all route in the *RouteConfig.cs* file.

1. Create a new catch-all route in the MVC application's **RouteConfig.cs** file. Make sure the route is the last route that you register in the file:



```
public static void RegisterRoutes(RouteCollection routes)
{
    ...
    routes.MapRoute(
        name: "NotFound",
        url: "{*url}",
        defaults: new { controller = "HttpErrors", action = "NotFound" }
    );
}
```

2. Create a controller and action for the catch-all route.

```
public class HttpErrorsController : Controller
{
    public ActionResult NotFound()
    {
        Response.StatusCode = 404;
        Response.TrySkipIisCustomErrors = true;

        return View();
    }
}
```

The system returns the *NotFound.cshtml* page whenever a request for an undefined route is made on the MVC site.