You can configure the system to display custom pages instead of standard error messages. Custom pages help reduce the inconvenience caused to visitors if they run into an error while browsing your website, and also improves the security of the site by hiding potentially sensitive internal data (such as code in stack traces). You can create custom pages for this purpose with any kind of content, such as an apology or additional instructions, and then configure the system to display the pages in the appropriate situations.

Adding custom Page not found error pages

The *Page not found* error (404 HTTP status code) is one of the most common problems encountered by visitors. Kentico provides several features that allow you to conveniently set up a custom page as a response. For *page not found* errors, the error page can either be a physical *.aspx* file placed under the web project or a dedicated page created in a specific website's content tree.

To assign your custom page to a particular website (or globally):

1. Go to **Settings -> Content**.
2. Enter the URL of the given page as the value of the **Page not found URL** setting, for example: *~/SpecialPages/PageNotFound.aspx*

   > Since there are two possible types of error pages, the system interprets the URL in two different ways. The sample URL value above specifies either:
   >
   > - The URL of a physical page named *PageNotFound.aspx* located in the web project under a folder called *SpecialPages*.
   > - If such a file does not exist, the system attempts to select a Kentico page under the current website, with an alias path equal to */SpecialPages/PageNotFound*.

3. Click **Save**.

It is recommended to use Kentico pages for page not found error pages. With this approach, you can define the error page's content using the portal engine and leverage all of its features. For instance, you can translate the page not found page on a multilingual website and Kentico automatically displays the culture version that matches the language selected by the user.

You do not need to manually handle the HTTP response code of the page specified by the setting. The page automatically returns a 404 status code when accessed (applies to both pages and physical pages). This allows applications, services and web crawlers to find out that a *page not found* error has occurred.

> **Handling 404 errors for general content**
>
> Perform the following steps to ensure that the system returns your custom *Page not found* error page for invalid requests that target all types of site content, not just the pages processed by the Kentico engine:
>
> 1. Edit your application's **web.config** file.
> 2. Find the **system.webServer** section directly under the root (i.e. not under a specific <*location*> element).
> 3. Set the **runAllManagedModulesForAllRequests** attribute to *true* for the opening tag of the **<modules>** element:
>
> ```
> <system.webServer>
>   <modules runAllManagedModulesForAllRequests="true">
>     ...
>   </modules>
> ```

Handling general errors

Kentico is a standard ASP.NET application, so you can configure the handling of all types of errors and exceptions via the **<customErrors>** element under the <*system.web*> section of your web.config file. See the customErrors MSDN article for more information.

By default, the Kentico web.config sets a general error page for errors with the 500 HTTP status code. The error page is displayed only for remote clients, requests from local development machines return errors with full details.

You can add any number of child **<error>** elements representing individual types of HTTP errors that you wish to handle.

- You need to enter the HTTP response code of the given error into the **statusCode** attribute and the URL of the appropriate error page as the **redirect** value.
- It is recommended to add general error pages directly into your web project as *.aspx* files, for example under the **CMSMessages** folder.
- You can define the content of the error page to match your specific requirements.
- Keep in mind that your custom error pages should always return the appropriate *HTTPResponse* status code.

Example

```
<system.web>
...
        <customErrors mode="RemoteOnly">
                <error statusCode="500" redirect="~/CMSMessages/CustomError.aspx" />
                <error statusCode="503" redirect="~/CMSMessages/CustomError.aspx" />
        </customErrors>
...
</system.web>
```

When using *<customErrors>*, it is also recommended to disable handling of errors on the level of IIS by adding a **httpErrors** element under the application's main *<system.webServer>* element, with the **existingResponse** attribute set to *PassThrough*:

```
<system.webServer>
...
        <httpErrors existingResponse="PassThrough" />
...
</system.webServer>
```

Using <httpErrors> for error handling

Alternatively, handling of errors can also be configured on the IIS level via the **<httpErrors>** element under your web.config's main **<system.webServer>** element. See the HTTP Errors article for more information.

Choose and test which approach works best, depending on the environment in which your application is running.

Before deployment

Set custom error messages as described in this procedure *before* deploying your website and going live. Also, remember to disable debugging and tracing for your ASP.NET application.

Page not found error conflicts

If you set the **defaultRedirect** attribute of the **customErrors** element in your web.config file, the default redirect may override the page assigned for 404 page not found errors in the Kentico settings. This applies to URLs ending with the *.ashx*, *.asmx* or *.svc* extensions.

To work around this behavior, you can set the **redirectMode** attribute to *ResponseRewrite:*

```
<customErrors defaultRedirect="~/CMSMessages/CustomError.aspx" mode="RemoteOnly"
redirectMode="ResponseRewrite">
```