

```
Program executing template "Designs\ncnielsen\_parsed\pdf-used-machine.parsed.cshtml"
System.NullReferenceException: Object reference not set to an instance of an object.
   at CompiledRazorTemplates.Dynamic.RazorEngine_83a70730fabf43dbbacec26db9f1961a.Execute() in D:\dynamicweb.net\Solutions\Gotcha\ncn.dw9.dynamicweb-cms.com\files\Templates\Designs\ncnielsen\_parsed\pdf-used-machine
   at RazorEngine.Templating.TemplateBase.RazorEngine_83a70730fabf43dbbacec26db9f1961a.Execute() in D:\dynamicweb.net\Solutions\Gotcha\ncn.dw9.dynamicweb-cms.com\files\Templates\Designs\ncnielsen\_parsed\pdf-used-machine
   at RazorEngine.Templating.RazorEngineService.RunCompile(ITemplateKey key, TextWriter writer, Type modelType, Object model, DynamicViewBag viewBag)
   at RazorEngine.Templating.RazorEngineServiceExtensions.<>c__DisplayClass16_0.<RunCompile>b__0(TextWriter writer)
   at RazorEngine.Templating.RazorEngineServiceExtensions.WithWriter(Action`1 withWriter)
   at Dynamicweb.Rendering.RazorTemplateRenderingProvider.Render(Template template)
   at Dynamicweb.Rendering.TemplateRenderingService.Render(Template template)
   at Dynamicweb.Rendering.Template.RenderRazorTemplate()
```

```
1
2 @inherits Dynamicweb.Rendering.RazorTemplateBase<Dynamicweb.Rendering.RazorTemplateModel<Dynamicweb.Rendering.Template>>
3 @using Dynamicweb
4 @using Dynamicweb.Environment
5 @using Gotcha.NCnielsen.API;
6 @using Gotcha.NCnielsen.Business;
7
8 <!DOCTYPE html>
9 <html lang="@GetString("LongLang")">
10 @
11     string istouch = "False";
12
13     string jsEditDate = System.IO.File.GetLastWriteTime(System.Web.HttpContext.Current.Server.MapPath("/files/templates/designs/ncnielsen/js/ncnielsen.min.js")).ToString();
14     jsEditDate = jsEditDate.Replace(":", "").Replace(" ", "").Replace("-", "");
15
16     if (GetGlobalValue("Global:IsTablet").ToString() == "True" || GetGlobalValue("Global:Device:IsMobile").ToString() == "True")
17     {
18         istouch = "True";
19     }
20
21     string templateName = GetGlobalValue("Global:Pageview:LayoutTemplate").Replace("test.", "").Replace(".cshtml", "").Replace(".html", "").Replace(".", "");
22
23     string UMID = "-1";
24     string productID = "-1";
25
26     if (System.Web.HttpContext.Current.Request.QueryString["UMID"] != null)
27     {
28         UMID = System.Web.HttpContext.Current.Request.QueryString["UMID"];
29     }
30
31     if (System.Web.HttpContext.Current.Request.QueryString["ProductID"] != null)
32     {
33         productID = System.Web.HttpContext.Current.Request.QueryString["ProductID"];
34     }
35
36     string language = "dan";
37
38     if (Pageview.Area.ID == 3)
39     {
40         language = "enu";
41     }
42     else if (Pageview.Area.ID == 4)
43     {
44         language = "sve";
45     }
46     else if (Pageview.Area.ID == 5)
47     {
48         language = "nor";
49     }
50
51     ProductModel productModel = ProductFacade.GetProductModel(UMID, language);
52     string pdfDownloadLink = System.Web.HttpContext.Current.Request.Url.PathAndQuery;
53
54     if (pdfDownloadLink.Contains("?"))
55     {
56         pdfDownloadLink += "&";
57     }
58     else
59     {
60         pdfDownloadLink += "?";
61     }
62
63     if (UMID != "-1" && productModel.HasProduct)
64     {
65         pdfDownloadLink += "layoutTemplate=pdf-used-machine.cshtml&pdf=true&filename=" + productModel.Product.Name.ToLower().Replace(" ", "_").Replace("/", "-") + ".pdf";
66     }
67     else if (productID != "-1")
68     {
69         Dynamicweb.Ecommerce.Products.ProductService ps = new Dynamicweb.Ecommerce.Products.ProductService();
70         Dynamicweb.Ecommerce.Products.Product p = ps.GetProductById(productID, "", true);
71
72         pdfDownloadLink += "layoutTemplate=pdf-used-machine.cshtml&pdf=true&filename=" + p.Name.ToLower().Replace(" ", "_").Replace("/", "-") + ".pdf";
73     }
74
75     string ConvertToBase64Image(string uri)
76     {
77         using var httpClient = new System.Net.Http.HttpClient();
78         var imageBytes = httpClient.GetByteArrayAsync(uri).Result;
79         return $"data:image/png;base64, {Convert.ToBase64String(imageBytes)}";
80     }
81 }
82 </head>
83 <meta charset="@GetString("CharSet")">
84 <title>@GetValue("@Title")</title>
85 @if (System.Web.HttpContext.Current.Request.Url.Host.Contains(".by.gotcha.dk"))
86 {
87     <meta name="robots" content="noindex,nofollow">
88 }
89 <meta name="viewport" content="width=device-width, initial-scale=1.0">
90 @GetString("MetaTags")
91 @if (System.Web.HttpContext.Current.Request.Cookies["ncnielsen-styles"] != null || System.Web.HttpContext.Current.Request.QueryString["oldschoolcss"] != null)
92 {
93     <link rel="stylesheet" href="/Files/Templates/Designs/ncnielsen/css/ncnielsen.min.css" id="style1" media="screen">
94 }
95 else
96 {
97     if (System.IO.File.Exists(System.Web.HttpContext.Current.Server.MapPath("/files/templates/designs/ncnielsen/css/" + templateName + ".critical.css")))
98     {
99         <text>@System.IO.File.ReadAllText(System.Web.HttpContext.Current.Server.MapPath("/files/templates/designs/ncnielsen/css/" + templateName + ".critical.css"))</text>
100     }
101     else
102     {
103         <link rel="stylesheet" href="/Files/Templates/Designs/ncnielsen/css/ncnielsen.min.css" id="style2" media="screen">
104     }
105 }
106 <link rel="preload" href="/Files/Templates/Designs/ncnielsen/fonts/35C9D3_0_0.woff2" as="font" type="font/woff2" crossorigin="">
107 <link rel="preload" href="/Files/Templates/Designs/ncnielsen/fonts/35C9D3_0_0.woff" as="font" type="font/woff" crossorigin="">
108 <link rel="preload" href="/Files/Templates/Designs/ncnielsen/fonts/35C9D3_1_0.woff2" as="font" type="font/woff2" crossorigin="">
109 <link rel="preload" href="/Files/Templates/Designs/ncnielsen/fonts/35C9D3_1_0.woff" as="font" type="font/woff" crossorigin="">
110 <link rel="preload" href="/Files/Templates/Designs/ncnielsen/fonts/24cac6da-66aa-48cc-b950-fcadf2b76ec5.woff2" as="font" type="font/woff2" crossorigin="">
111 <link rel="preload" href="/Files/Templates/Designs/ncnielsen/fonts/bd659a5c-378a-4055-80a1-69bc94044bac.woff" as="font" type="font/woff" crossorigin="">
112 <link rel="preload" href="/Files/Templates/Designs/ncnielsen/fonts/4e27b36d-615c-41eb-baf6-a3095eb585f5.woff2" as="font" type="font/woff2" crossorigin="">
113 <link rel="preload" href="/Files/Templates/Designs/ncnielsen/fonts/9805cfa5-519d-4bfb-82af-f6e07c275f94.woff" as="font" type="font/woff" crossorigin="">
114 @RenderSnippet("csstop")
115 </head>
116 <body data-is-touch="@istouch" data-is-mobile="@GetGlobalValue("Global:Device:IsMobile)" data-is-tablet="@GetGlobalValue("Global:Device:IsTablet)" data-is-desktop="@GetGlobalValue("Global:Device:IsDesktop")">
117 <div id="pageWrap">
118     <nav id="navigation">
119         <a href="#" id="logo">
120             @if (System.IO.File.Exists(System.Web.HttpContext.Current.Server.MapPath("/files/templates/designs/ncnielsen/images/nc_nielsen_logo.svg")))
121             {
122                 <text>@System.IO.File.ReadAllText(System.Web.HttpContext.Current.Server.MapPath("/files/templates/designs/ncnielsen/images/nc_nielsen_logo.svg"))</text>
123             }
124             </a>
125             @GetValue("DwNavigation(primaryNavigation)")
126             <a id="burger">
127                 <span class="icon icon-burger">
128                     @if (System.IO.File.Exists(System.Web.HttpContext.Current.Server.MapPath("/files/templates/designs/ncnielsen/images/icon-burger.svg")))
129                     {
130                         <text>@System.IO.File.ReadAllText(System.Web.HttpContext.Current.Server.MapPath("/files/templates/designs/ncnielsen/images/icon-burger.svg"))</text>
131                     }
132                 </span>
133             </a>
134         </nav>
135     </div>
136 </body>
```



```

279         if (counter == 0)
280         {
281             <div class="gallery-large">
282                 <div class="image js-active">
283                     <img src=@imagePath alt=@img.Key'>
284                 </div>
285                 <div class="image second-image"></div>
286                 @if (approvedValue != "-")
287                 {
288                     <span class="approved">@productModel.Product.Make @approvedValue</span>
289                 }
290             </div>
291             <div class="gallery-small js-active">
292                 <img src=@imagePath alt=@img.Key'>
293             </div>
294         }
295         else
296         {
297             <div class="gallery-small">
298                 <img src=@imagePath alt=@img.Key'>
299             </div>
300         }
301         counter++;
302     }
303 }
304 </div>
305 }
306 }
307 else if (productModel.Product.ImageList.Count() == 1)
308 {
309     foreach (var img in productModel.Product.ImageList)
310     {
311         string imagePath = ConvertToBase64Image($"https://{img.Value}");
312
313         <div>
314             <img src=@imagePath alt=@img.Key'>
315             @if (approvedValue != "-")
316             {
317                 <span class="approved">@productModel.Product.Make @approvedValue</span>
318             }
319         </div>
320     }
321 }
322 else if (productModel.Product.ImageList.Count() == 0)
323 {
324     <div>
325         
326         @if (approvedValue != "-")
327         {
328             <span class="approved">@productModel.Product.Make @approvedValue</span>
329         }
330     </div>
331 }
332 }
333 else
334 {
335     <h1>@Translate("Translate_Used_machine_not_found")</h1>
336     <div>
337         
338     </div>
339 }
340 }
341 else
342 {
343     <text>@RenderSnippet("ChosenUsedMachineGallery")</text>
344 }
345 </div>
346 <div class="content">
347     @if (productModel.HasProduct)
348     {
349         <a href=@pdfDownloadLink target="blank" id="usedMachineSavePdf">
350             @if (System.IO.File.Exists(System.Web.HttpContext.Current.Server.MapPath("/files/templates/designs/ncnielsen/images/icon-pdf.svg")))
351             {
352                 <span class="icon icon-pdf">
353                     @System.IO.File.ReadAllText(System.Web.HttpContext.Current.Server.MapPath("/files/templates/designs/ncnielsen/images/icon-pdf.svg"))
354                 </span>
355             }
356             <span class="text">
357                 @Translate("Translate_Gem_som_PDF")
358             </span>
359         </a>
360     }
361     @if (UMID != "-1")
362     {
363         if (productModel.HasProduct)
364         {
365             <div class="specifications">
366                 <span class="header">@Translate("Translate_Specifikationer")</span>
367                 <ul>
368                     <li class="specification">
369                         <span class="label">@Translate("Translate_Intern_nummer")</span>
370                         <span class="value">@productModel.Product.Id</span>
371                     </li>
372                     <li class="specification">
373                         <span class="label">@Translate("Translate_Aargang")</span>
374                         <span class="value">@productModel.Product.Year</span>
375                     </li>
376                     @if (productModel.Product.SpecificationList.Count() > 10)
377                     {
378                         for (var a = 0; a < 10; a++)
379                         {
380                             if (
381                                 @productModel.Product.SpecificationList[a].Id != "M-0090570" &&
382                                 @productModel.Product.SpecificationList[a].Id != "M-0090565"
383                             )
384                             {
385                                 <li class="specification"><span class="label">@productModel.Product.SpecificationList[a].Label</span> <span class="value">@productModel.Product.SpecificationList[a]
386                             }
387                             }
388                         }
389                     }
390                     else
391                     {
392                         foreach (var spec in productModel.Product.SpecificationList)
393                         {
394                             if (
395                                 spec.Id != "M-0090570" &&
396                                 spec.Id != "M-0090565"
397                             )
398                             {
399                                 <li class="specification"><span class="label">@spec.Label</span> <span class="value">@spec.Value</span></li>
400                             }
401                         }
402                     }
403                 </ul>
404                 @if (productModel.Product.SpecificationList.Count() > 10)
405                 {
406                     <div>
407                         <a id="allSpecificationsLink" class="button button-inline">@Translate("Translate_Se_alle_specifikationer")</a>
408                     </div>
409                 }
410                 <a id="usedMachineRequestButton" class="button button-primary">
411                     <span class="text">@Translate("Translate_Prisfoerespoergsel")</span>
412                 </a>
413             </div>
414             <div id="usedMachinesRequestForm">
415                 @({
416                     int formParagraphID = -1;
417                     try
418                     {
419                         formParagraphID = Convert.ToInt32(Pageview.Area.Item["UsedMachinesRequestParagraph"]);
420                     }
421                     catch
422                     {

```

```
224
225
226         if (formParagraphID != -1)
227         {
228             <text>@RenderParagraphContent(formParagraphID)</text>
229         }
230     }
231     @if (System.IO.File.Exists(System.Web.HttpContext.Current.Server.MapPath("/files/templates/designs/ncnielsen/images/icon-close.svg")))
232     {
233         <span class="icon icon-close">
234             @System.IO.File.ReadAllText(System.Web.HttpContext.Current.Server.MapPath("/files/templates/designs/ncnielsen/images/icon-close.svg"))
235         </span>
236     }
237 </div>
238 }
239 else
240 {
241     <div class="specifications">
242
243     </div>
244 }
245 }
246 else
247 {
248     <text>@RenderSnippet("ChosenUsedMachineHeaderSpecifications")</text>
249     <div id="usedMachinesRequestForm">
250         @{
251             int formParagraphID = -1;
252             try
253             {
254                 formParagraphID = Convert.ToInt32(Pageview.Area.Item["UsedMachinesRequestParagraph"]);
255             }
256             catch
257             {
258             }
259         }
260         if (formParagraphID != -1)
261         {
262             <text>@RenderParagraphContent(formParagraphID)</text>
263         }
264     }
265     @if (System.IO.File.Exists(System.Web.HttpContext.Current.Server.MapPath("/files/templates/designs/ncnielsen/images/icon-close.svg")))
266     {
267         <span class="icon icon-close">
268             @System.IO.File.ReadAllText(System.Web.HttpContext.Current.Server.MapPath("/files/templates/designs/ncnielsen/images/icon-close.svg"))
269         </span>
270     }
271 </div>
272 }
273 </div>
274 <div class="social">
275 <a href="tel:@Translate("Translate_Brugte_maskiner_kontakt_telefonnr")" class="button button-secondary">
276     <span class="text">@Translate("Translate_Kontakt_os_paa")</span>
277 </a>
278 <span class="header">Følg os</span>
279 @if (System.IO.File.Exists(System.Web.HttpContext.Current.Server.MapPath("/files/templates/designs/ncnielsen/images/icon-facebook.svg"))) && !string.IsNullOrEmpty(Pageview.Area.Item["
280 {
281     <a href="@Pageview.Area.Item["Facebook"]" class="icon icon-facebook" target="_blank">
282         @System.IO.File.ReadAllText(System.Web.HttpContext.Current.Server.MapPath("/files/templates/designs/ncnielsen/images/icon-facebook.svg"))
283     </a>
284 }
285 @if (System.IO.File.Exists(System.Web.HttpContext.Current.Server.MapPath("/files/templates/designs/ncnielsen/images/icon-linkedin.svg"))) && !string.IsNullOrEmpty(Pageview.Area.Item["
286 {
287     <a href="@Pageview.Area.Item["LinkedIn"]" class="icon icon-linkedin" target="_blank">
288         @System.IO.File.ReadAllText(System.Web.HttpContext.Current.Server.MapPath("/files/templates/designs/ncnielsen/images/icon-linkedin.svg"))
289     </a>
290 }
291 @if (System.IO.File.Exists(System.Web.HttpContext.Current.Server.MapPath("/files/templates/designs/ncnielsen/images/icon-youtube.svg"))) && !string.IsNullOrEmpty(Pageview.Area.Item["Y
292 {
293     <a href="@Pageview.Area.Item["YouTube"]" class="icon icon-youtube" target="_blank">
294         @System.IO.File.ReadAllText(System.Web.HttpContext.Current.Server.MapPath("/files/templates/designs/ncnielsen/images/icon-youtube.svg"))
295     </a>
296 }
297 </div>
298 </div>
299 </header>
300 @if (UMID != "-1")
301 {
302     if (productModel.HasProduct)
303     {
304         if (productModel.Product.SpecificationList.Count() > 10)
305         {
306             <section id="allSpecifications" class="section section-white">
307                 <div class="content">
308                     <div class="specifications">
309                         <h2>@Translate("Translate_Alle_specifikationer")</h2>
310                         <ul>
311                             <li class="specification">
312                                 <span class="label">@Translate("Translate_Intern_nummer")</span>
313                                 <span class="value">@productModel.Product.Id</span>
314                             </li>
315                             <li class="specification">
316                                 <span class="label">@Translate("Translate_Aargang")</span>
317                                 <span class="value">@productModel.Product.Year</span>
318                             </li>
319                             @foreach (var spec in productModel.Product.SpecificationList)
320                             {
321                                 if (
322                                     spec.Id != "M-0090570" &&
323                                     spec.Id != "M-0090565"
324                                 )
325                                 {
326                                     <li class="specification"><span class="label">@spec.Label</span> <span class="value">@spec.Value</span></li>
327                                 }
328                             }
329                         </ul>
330                     <a id="usedMachineRequestButton" class="button button-primary">
331                         <span class="text">@Translate("Translate_Prisforespoergsel")</span>
332                     </a>
333                 </div>
334             </section>
335         }
336     }
337 }
338 string templatename = GetGlobalValue("Global:Pageview.LayoutTemplate").Replace(".cshtml", "").ToLower();
339
340 if (templatename == "pdf-used-machine")
341 {
342     double dHalfWay = Convert.ToDouble(productModel.Product.SpecificationList.Count() - 2) / 2;
343     int halfWay = (int)Math.Ceiling(dHalfWay);
344     halfWay = halfWay;
345
346     <section id="pdfSpecifications" class="section section-white">
347         <div class="content">
348             <div class="specifications">
349                 <ul>
350                     <li class="specification">
351                         <span class="label">@Translate("Translate_Intern_nummer")</span>
352                         <span class="value">@productModel.Product.Id</span>
353                     </li>
354                     <li class="specification">
355                         <span class="label">@Translate("Translate_Aargang")</span>
356                         <span class="value">@productModel.Product.Year</span>
357                     </li>
358                     @for (int a = 0; a < halfWay; a++)
359                     {
360                         <li class="specification">
361                             <span class="label">@productModel.Product.SpecificationList[a].Label</span>
362                             <span class="value">@productModel.Product.SpecificationList[a].Value</span>
363                         </li>
364                     }
365                 </ul>
366             </div>
367         </div>
368     </section>
369 }
```

```

569         @for (int a = halfWay; a < productModel.Product.SpecificationList.Count(); a++)
570         {
571             <li class="specification">
572                 <span class="label">@productModel.Product.SpecificationList[a].Label</span>
573                 <span class="value">@productModel.Product.SpecificationList[a].Value</span>
574             </li>
575         }
576     </ul>
577 </div>
578 </div>
579 </section>
580 }
581 }
582 else
583 {
584     <text>@RenderSnippet("UsedMachineAllSpecifications")</text>
585 }
586 <div id="ecomContent">@GetValue("DwContent(ecomContent)")</div>
587 @inherits Dynamicweb.Rendering.RazorTemplateBase<Dynamicweb.Rendering.RazorTemplateModel<Dynamicweb.Rendering.Template>>
588 @using Dynamicweb
589 @using Dynamicweb.Environment
590
591
592
593 @SnippetStart("cstop")
594 <style>
595     @page {
596         size: A4;
597     }
598
599     * {
600         box-sizing: border-box;
601     }
602
603     html {
604         margin: 0;
605         min-height: 100%;
606         position: relative;
607         background: transparent;
608     }
609
610     body {
611         width: 100%;
612         height: 100%;
613         background: transparent;
614         margin: 0;
615         color: #000;
616         font-family: Arial, Helvetica, sans-serif;
617         position: relative;
618     }
619
620     body:before {
621         content: none;
622     }
623
624     #pageWrap {
625         overflow: hidden;
626         width: 100%;
627         height: 100%;
628         position: relative;
629         padding: 2cm 2cm;
630         background: transparent;
631     }
632
633     .content .specifications .header {
634         line-height: 1.5em;
635         background: transparent;
636     }
637
638     #navigation,
639     #modallanguages,
640     #modalSearch,
641     #header:before,
642     #header:after,
643     #header .specifications *,
644     #usedMachineRequestButton,
645     .gallery .gallery-small,
646     #usedMachineRequestButton,
647     #usedMachineRequestForm,
648     #usedMachineSavePdf,
649     #allSpecifications h2,
650     .social,
651     #footerNavigation,
652     #footer:before,
653     #footer br,
654     #footer .social,
655     #allSpecifications,
656     #ecomContent {
657         display: none;
658     }
659
660     #burgerNav:after,
661     #burgerNavigation ul,
662     .section.section-white {
663         background: transparent;
664     }
665
666     #header {
667         height: auto;
668         /*margin: 2cm 2cm 0.5cm 2cm;*/
669         padding: 0;
670         background: transparent;
671     }
672
673     .section {
674         /*margin: 0 2cm;*/
675         background: transparent;
676     }
677
678     #header .specifications .header {
679         display: block;
680         background: transparent;
681     }
682
683     #header .header,
684     h1 {
685         color: #000;
686         font-size: 24pt;
687         display: block;
688         font-family: Arial, Helvetica, sans-serif;
689         background: transparent;
690     }
691
692     #header .content,
693     .section {
694         text-align: left;
695         background: transparent;
696     }
697
698     .section,
699     .gallery {
700         padding: 0;
701         background: transparent;
702     }
703
704     #productGalleryImageMain {
705         width: calc(100% - 4cm);
706         margin-bottom: 5mm;
707         background: transparent;
708     }
709
710     #productGalleryImageMain h1 {
711         margin-bottom: 5mm;
712         background: transparent;

```

```
714
715 #productGalleryImageMain {
716     width: calc(100% - 4cm);
717     background: transparent;
718 }
719
720 #productGalleryImageMain > div {
721     padding: 0;
722     background: transparent;
723     height: auto;
724     width: auto;
725     max-width: 100%;
726     max-height: 142mm;
727 }
728
729 .gallery .gallery-large {
730     /*width: 100%;*/
731     height: auto;
732     width: auto;
733     max-width: 100%;
734     max-height: 142mm;
735 }
736
737 #productGalleryImageMain img,
738 .gallery .gallery-large img {
739     height: auto;
740     width: auto;
741     max-width: 100%;
742     max-height: 142mm;
743 }
744
745 .image {
746     display: block;
747 }
748
749 #pdfSpecifications ul {
750     width: 50%;
751     float: left;
752     padding: 0;
753     font-size: 11pt;
754     line-height: 1.08;
755     margin-left: -5mm;
756     margin-right: -5mm;
757     background: transparent;
758 }
759
760 #pdfSpecifications ul + ul {
761     margin-left: 10mm;
762     background: transparent;
763 }
764
765 #pdfSpecifications li {
766     display: block;
767     width: 100%;
768     float: left;
769     padding-bottom: 2mm;
770     padding-left: 5mm;
771     padding-right: 5mm;
772     overflow: hidden;
773     background: transparent;
774 }
775
776 #pdfSpecifications .label {
777     display: inline-block;
778     float: left;
779     background: transparent;
780 }
781
782 #pdfSpecifications .value {
783     display: inline-block;
784     float: right;
785     background: transparent;
786 }
787
788 #footer {
789     background: none;
790     font-size: 7pt;
791     height: 7mm;
792     position: absolute;
793     top: 230mm;
794     /*bottom: 1cm;*/
795     left: auto;
796     right: 2cm;
797     overflow: hidden;
798     line-height: 1.3;
799     color: #000;
800     background: transparent;
801 }
802
803 #footer .content {
804     display: block;
805     margin: 0;
806     width: 100%;
807     max-width: none;
808     background: transparent;
809 }
810
811 #footer .header {
812     display: inline;
813     text-transform: none;
814     font-size: 7pt;
815     padding: 0;
816     margin: 0;
817     line-height: 1.3;
818     color: #000;
819     font-family: Arial, Helvetica, sans-serif;
820     font-weight: bold;
821     background: transparent;
822 }
823
824 #footer .header:after {
825     content: " ";
826     display: inline;
827     position: static;
828     height: auto;
829     width: auto;
830     background: none;
831     bottom: auto;
832     left: auto;
833     font-family: Arial, Helvetica, sans-serif;
834     font-weight: bold;
835     background: transparent;
836 }
837
838 #footer p {
839     display: inline;
840     background: transparent;
841 }
842
843 #footer-logo {
844     position: absolute;
845     top: 230mm;
846     /*bottom: 1cm;*/
847     left: 2cm;
848     width: 39.2mm;
849     height: 7mm;
850     z-index: 999999;
851     background: transparent;
852 }
853
854 #footer-logo img {
855     width: 100%;
856     height: 100%;
857     background: transparent;
```



```
1004     }
1005     @RenderSnippet("OtherLanguages")
1006   </div>
1007   @if (System.IO.File.Exists(System.Web.HttpContext.Current.Server.MapPath("/files/templates/designs/ncnielsen/images/icon-close.svg")))
1008   {
1009     <span class="icon icon-close button-close">
1010       @System.IO.File.ReadAllText(System.Web.HttpContext.Current.Server.MapPath("/files/templates/designs/ncnielsen/images/icon-close.svg"))
1011     </span>
1012   }
1013 </div>
1014 </div>
1015 @RenderSnippet("contentbottom")
1016 <script src="/Files/Templates/Designs/ncnielsen/js/ncnielsen.min.js?v=@jsEditDate" async="" defer=""></script>
1017 @if (System.Web.HttpContext.Current.Request.Cookies["ncnielsen-styles"] == null)
1018 {
1019   System.Web.HttpContext.Current.Response.Cookies["ncnielsen-styles"].Value = "1";
1020   System.Web.HttpContext.Current.Response.Cookies["ncnielsen-styles"].Expires = DateTime.Now.AddDays(100);
1021 }
1022 <noscript id="deferred-styles">
1023   <link rel="stylesheet" href="/files/templates/designs/ncnielsen/css/ncnielsen.min.css" media="screen">
1024 </noscript>
1025 <script>
1026   var loadDeferredStyles = function () {
1027     var addStylesNode = document.getElementById("deferred-styles");
1028     var replacement = document.createElement("div");
1029
1030     replacement.innerHTML = addStylesNode.textContent;
1031     document.body.appendChild(replacement)
1032     addStylesNode.parentElement.removeChild(addStylesNode);
1033   };
1034
1035   var raf = requestAnimationFrame || mozRequestAnimationFrame || webkitRequestAnimationFrame || msRequestAnimationFrame;
1036
1037   if (raf) raf(function () { window.setTimeout(loadDeferredStyles, 0); });
1038   else window.addEventListener('load', loadDeferredStyles);
1039 </script>
1040 }
1041 </body>
1042 </html>
1043
```