

Pliki .twig

@modules/basket_addresses/basket_addresses.twig

```
1  {% set userSettings = ObjectApi.getUsersSettings() %}
2
- 3  <basket-addresses {% if ObjectApi.getFeatureFlag("dev_storefront_simple_basket_address").isEnabled %}display-type="visible"{% endif %}
    class="basket-module" {% if ObjectApi.getFeatureFlag("storefront_registration").isEnabled %}address-mode="{{ userSettings.registrationType
    pe ?? '0' }}"{% endif %}>
+ 3  <basket-addresses class="basket-module" {% if ObjectApi.getFeatureFlag("storefront_registration").isEnabled %}address-mode="{{ userSettings
    ings.registrationType ?? '0' }}"{% endif %}>
4      <div slot="loader" class="loader p-all-xs-4">
5          <div class="loader__spinner"></div>
6      </div>
```

@modules/countdown_timer/countdown_timer.twig

```
+ 1  {% from "@macros/header.twig" import header %}
+ 2
+ 3  {% set textAlign = 'center' %}
+ 4  {% if moduleConfig.contentAlignment == 'start' %}
+ 5      {% set textAlign = 'left' %}
+ 6  {% elseif moduleConfig.contentAlignment == 'end' %}
+ 7      {% set textAlign = 'right' %}
+ 8  {% endif %}
+ 9
+ 10 {% set endDate = moduleConfig.endDateTime.iso8601() %}
+ 11
+ 12 <countdown-timer
+ 13     id="countdown-timer-{{ moduleInstance }}"
+ 14     class="countdown-timer countdown-timer_{{ moduleConfig.bannerSize }} countdown-timer__position_{{ moduleConfig.contentPosition }} {
    { moduleConfig.classNames }}"
+ 15     end-date="{{ endDate }}"
+ 16     end-action="{{ moduleConfig.endCountdownAction }}"
+ 17     tabindex="0"
+ 18     role="timer"
+ 19     {% if moduleConfig.backgroundImage.paths.original %}style="--background-url: url('{{ moduleConfig.backgroundImage.paths.original }}
    ');"{% endif %}
+ 20     {% if moduleConfig.alt %}aria-label="{{ moduleConfig.alt }}"{% endif %}
+ 21 >
```

```

+ 22
+ 23     {% if moduleConfig.hasBackgroundOverlay %}
+ 24     <div class="countdown-timer__overlay" aria-hidden="true" style="--overlayBackgroundColor: {{ moduleConfig.backgroundOverlay }};"></div>
+ 25     {% endif %}
+ 26
+ 27     <div class="countdown-timer__active justify-items-{{ moduleConfig.contentAlignment }} text-align-{{ textAlign }}">
+ 28     {% if moduleConfig.hasActiveCounterHeader and moduleConfig.activeCounterHeaderContent and moduleConfig.activeCounterHeaderLevel
+ 29     %}
+ 30         {{
+ 31             header({
+ 32                 level: moduleConfig.activeCounterHeaderLevel|default('h2'),
+ 33                 content: moduleConfig.activeCounterHeaderContent,
+ 34                 classNames: "countdown-timer__header "
+ 35             })
+ 36         }}
+ 37     {% endif %}
+ 38     {% if moduleConfig.hasActiveCounterDescription and moduleConfig.activeCounterDescriptionPlacement == 'above' %}
+ 39     <div class="countdown-timer__description countdown-timer__description_above text-align-{{ textAlign }}">
+ 40         {{ moduleConfig.activeCounterDescription }}
+ 41     </div>
+ 42     {% endif %}
+ 43
+ 44     <div
+ 45         class="countdown-timer__countdown"
+ 46         style="
+ 47             --timeUnitBackgroundColor: {{ moduleConfig.digitsBackgroundColor }};
+ 48             --borderRadius: {{ moduleConfig.borderRadius }};
+ 49             --textColor: {{ moduleConfig.digitsTextColor }};"
+ 50     >
+ 51         <span class="sr-only">{{ translate("Countdown to") }}: {{ endDate }}</span>
+ 52         <div class="countdown-timer__unit countdown-timer__unit_days {% if moduleConfig.timeUnitsPosition == 'inside' %}countdown-timer__unit_with-background{% endif %} resetcss">
+ 53             <span class="countdown-timer__value {% if moduleConfig.timeUnitsPosition == 'under' %}countdown-timer__value_with-background{% endif %}" data-unit="days" aria-hidden="true">00</span>
+ 54             <span class="countdown-timer__label" aria-hidden="true">{{ translate("Days") }}</span>
+ 55         </div>
+ 56         {% if moduleConfig.shouldShowSeparator %}
+ 57         <span class="countdown-timer__separator" aria-hidden="true">:</span>
+ 58         {% endif %}
+ 59         <div class="countdown-timer__unit countdown-timer__unit_hours {% if moduleConfig.timeUnitsPosition == 'inside' %}countdown-timer__unit_with-background{% endif %} resetcss">
+ 60             <span class="countdown-timer__value {% if moduleConfig.timeUnitsPosition == 'under' %}countdown-timer__value_with-background{% endif %}" data-unit="hours" aria-hidden="true">00</span>
+ 61             <span class="countdown-timer__label" aria-hidden="true">{{ translate("Hours") }}</span>
+ 62         </div>
+ 63         {% if moduleConfig.shouldShowSeparator %}
+ 64         <span class="countdown-timer__separator" aria-hidden="true">:</span>
+ 65         {% endif %}
+ 66         <div class="countdown-timer__unit countdown-timer__unit_minutes {% if moduleConfig.timeUnitsPosition == 'inside' %}countdown

```

```

n-timer__unit_with-background{% endif %} resetcss">
+ 67         <span class="countdown-timer__value {% if moduleConfig.timeUnitsPosition == 'under' %}countdown-timer__value_with-backg
round{% endif %}" data-unit="minutes" aria-hidden="true">00</span>
+ 68         <span class="countdown-timer__label" aria-hidden="true">{{ translate("Minutes") }}</span>
+ 69     </div>
+ 70     {% if moduleConfig.shouldShowSeparator %}
+ 71         <span class="countdown-timer__separator" aria-hidden="true">:</span>
+ 72     {% endif %}
+ 73     <div class="countdown-timer__unit countdown-timer__unit_seconds {% if moduleConfig.timeUnitsPosition == 'inside' %}countdow
n-timer__unit_with-background{% endif %} resetcss">
+ 74         <span class="countdown-timer__value {% if moduleConfig.timeUnitsPosition == 'under' %}countdown-timer__value_with-backg
round{% endif %}" data-unit="seconds" aria-hidden="true">00</span>
+ 75         <span class="countdown-timer__label" aria-hidden="true">{{ translate("Seconds") }}</span>
+ 76     </div>
+ 77
+ 78
+ 79 </div>
+ 80
+ 81     {% if moduleConfig.hasActiveCounterDescription and moduleConfig.activeCounterDescriptionPlacement == 'below' %}
+ 82         <div class="countdown-timer__description countdown-timer__description_below text-align-{{ textAlign }}">
+ 83             {{ moduleConfig.activeCounterDescription }}
+ 84         </div>
+ 85     {% endif %}
+ 86
+ 87     {% if moduleConfig.hasActiveCounterButton %}
+ 88         <a
+ 89             class="btn btn_{{ moduleConfig.activeCounterButtonSize }} btn_{{ moduleConfig.activeCounterButtonType }} countdown-time
r__button"
+ 90             href="{{ moduleConfig.activeCounterButtonUrl }}"
+ 91             rel="noopener noreferrer"
+ 92             {% if not moduleConfig.shouldOpenActiveCounterLinkInTheSameTab %}target="_blank"{% endif %}
+ 93             >
+ 94                 {{ moduleConfig.activeCounterButtonContent }}
+ 95             </a>
+ 96         {% endif %}
+ 97     </div>
+ 98
+ 99     {% if moduleConfig.endCountdownAction == 'showEndBanner'%}
+ 100         <div class="countdown-timer__ended justify-items-{{ moduleConfig.contentAlignment }} text-align-{{ textAlign }}" hidden>
+ 101             {% if moduleConfig.hasEndBannerHeader and moduleConfig.endBannerHeaderContent and moduleConfig.endBannerHeaderLevel %}
+ 102                 {{
+ 103                     header({
+ 104                         level: moduleConfig.endBannerHeaderLevel|default('h2'),
+ 105                         content: moduleConfig.endBannerHeaderContent,
+ 106                         classNames: "countdown-timer__header"
+ 107                     })
+ 108                 }}
+ 109             {% endif %}
+ 110
+ 111             {% if moduleConfig.hasEndBannerDescription %}
+ 112                 <div class="countdown-timer__description text-align-{{ textAlign }}">

```

```

+ 113         {{ moduleConfig.endBannerDescription }}
+ 114     </div>
+ 115     {% endif %}
+ 116
+ 117     {% if moduleConfig.hasEndBannerButton %}
+ 118     <a
+ 119         class="btn btn_{{ moduleConfig.endBannerButtonSize }} btn_{{ moduleConfig.endBannerButtonType }} countdown-timer_b
button"
+ 120         href="{{ moduleConfig.endBannerButtonUrl }}"
+ 121         rel="noopener noreferrer"
+ 122         {% if not moduleConfig.shouldOpenEndBannerLinkInTheSameTab %}target="_blank"{% endif %}
+ 123     >
+ 124         {{ moduleConfig.endBannerButtonContent }}
+ 125     </a>
+ 126     {% endif %}
+ 127 </div>
+ 128 {% endif %}
+ 129
+ 130 </countdown-timer>

```

@modules/product_shipping_time/product_shipping_time.twig

```

6     {% if product.delivery is not null or product.delivery is null and product.options.count > 0 %}
7     <product-shipping-time
8         class="product-shipping-time property"
- 9         {% if product.delivery is null %}hidden{% endif %}
+ 9         {% if product.delivery is null or not product.availability.isAvailable %}hidden{% endif %}
10         product-id="{{ product.id }}"
11     >
12         {{

```

@macros/icon.twig

```

4     {% set iconNoStroke = options.noStroke ? "icon_no-stroke" : "" %}
5     {% set iconClickable = options.clickable ? "icon_clickable" : "" %}
6
- 7     {% if ObjectApi.getFeatureFlag("dev_system_icons").isEnabled %}
- 8         {% set defaultIconsPath = systemConfig.systemIcons.paths.original ?? '/assets/img/icons/symbol-defs.svg' %}
- 9         {% set src = options.src ?? defaultIconsPath %}
- 10        {% else %}
- 11        {% set src = options.src ?? '/assets/img/icons/symbol-defs.svg' %}
- 12        {% endif %}
+ 7     {% set defaultIconsPath = systemConfig.systemIcons.paths.original ?? '/assets/img/icons/symbol-defs.svg' %}
+ 8     {% set src = options.src ?? defaultIconsPath %}
9
10    <svg class="
11        icon

```

countdown_timer/countdown_timer.less

```
+ 1  .countdown-timer {
+ 2      display: grid;
+ 3      position: relative;
+ 4      min-width: 100%;
+ 5      max-width: ~'calc(100vw - (@{gridGutterRem} * 2))';
+ 6      background-image: var(--background-url);
+ 7      background-size: cover;
+ 8      background-position: center;
+ 9      background-repeat: no-repeat;
+ 10     padding: @countDownTimerPaddingMobile;
+ 11     overflow: hidden;
+ 12
+ 13     @media (min-width: @breakPointXs) {
+ 14         padding: @countDownTimerPadding;
+ 15     }
+ 16
+ 17     &_small {
+ 18         min-height: @countDownTimerHeightSmall;
+ 19     }
+ 20     &_medium {
+ 21         min-height: @countDownTimerHeightMedium;
+ 22     }
+ 23     &_large {
+ 24         min-height: @countDownTimerHeightLarge;
+ 25     }
+ 26
+ 27     &__position {
+ 28         &_top-left {
+ 29             justify-content: start;
+ 30             align-content: start;
+ 31         }
+ 32         &_top-center {
+ 33             justify-content: center;
+ 34             align-content: start;
+ 35         }
+ 36         &_top-right {
+ 37             justify-content: end;
+ 38             align-content: start;
+ 39         }
+ 40         &_center-left {
+ 41             justify-content: start;
```

```
+ 42         align-content: center;
+ 43     }
+ 44     &_center {
+ 45         justify-content: center;
+ 46         align-content: center;
+ 47     }
+ 48     &_center-right {
+ 49         justify-content: end;
+ 50         align-content: center;
+ 51     }
+ 52     &_bottom-left {
+ 53         justify-content: start;
+ 54         align-content: end;
+ 55     }
+ 56     &_bottom-center {
+ 57         justify-content: center;
+ 58         align-content: end;
+ 59     }
+ 60     &_bottom-right {
+ 61         justify-content: end;
+ 62         align-content: end;
+ 63     }
+ 64 }
+ 65
+ 66 &__active,
+ 67 &__ended {
+ 68     display: grid;
+ 69     z-index: 1;
+ 70 }
+ 71
+ 72 &__ended {
+ 73     .countdown-timer__button {
+ 74         margin-top: @countDownTimerContainerGap;
+ 75     }
+ 76 }
+ 77
+ 78 &__header {
+ 79     word-break: break-word;
+ 80 }
+ 81
+ 82 &__header,
+ 83 &__description,
+ 84 &__countdown {
+ 85     width: 100%;
+ 86     max-width: @countDownTimerContentMaxWidth;
+ 87 }
+ 88
+ 89 &__header {
+ 90     color: inherit;
+ 91     margin: 0;
```

```

+ 92         width: fit-content;
+ 93     .weight-bold();
+ 94 }
+ 95
+ 96     &__description {
+ 97         .font-size(@countDownTimerDescriptionFontSize);
+ 98     &_below {
+ 99         margin-bottom: @countDownTimerContainerGap;
+ 100     }
+ 101 }
+ 102
+ 103     &__button {
+ 104         width: fit-content;
+ 105     }
+ 106
+ 107     &__countdown {
+ 108         display: flex;
+ 109         align-items: flex-start;
+ 110         gap: @countDownTimerUnitGapMobile;
+ 111         margin-inline: inherit;
+ 112         margin-block: @countDownTimerContainerGap;
+ 113         width: fit-content;
+ 114         flex-wrap: wrap;
+ 115
+ 116         &:has(.countdown-timer__separator) {
+ 117             gap: @countDownTimerUnitWithSeparatorGapMobile;
+ 118         }
+ 119
+ 120         @media (min-width: @breakPointXs) {
+ 121             gap: @countDownTimerUnitGap;
+ 122         }
+ 123         &:has(.countdown-timer__separator) {
+ 124             gap: @countDownTimerUnitWithSeparatorGap;
+ 125         }
+ 126     }
+ 127 }
+ 128
+ 129     &__unit {
+ 130         display: flex;
+ 131         flex-direction: column;
+ 132         align-items: center;
+ 133         gap: @globalSpacing * 0.25;
+ 134         min-width: @countDownTimerUnitMinWidthMobile;
+ 135         @media (min-width: @breakPointXs) {
+ 136             min-width: @countDownTimerUnitMinWidth;
+ 137         }
+ 138
+ 139         &_with-background {
+ 140             background-color: var(--timeUnitBackgroundColor);
+ 141             border-radius: ~'calc(var(--borderRadius) * 1%)';

```

```
+ 142 min-width: @countDownTimerUnitMinWidthMobile;
+ 143 padding: @countDownTimerUnitValuePaddingMobile;
+ 144
+ 145 @media (min-width: @breakPointXs) {
+ 146 min-width: @countDownTimerUnitMinWidth;
+ 147 padding: @countDownTimerUnitValuePadding;
+ 148 }
+ 149
+ 150 * {
+ 151 color: var(--textColor, inherit);
+ 152 }
+ 153 }
+ 154 }
+ 155
+ 156 &__value {
+ 157 text-align: center;
+ 158 margin-inline: auto;
+ 159 word-break: break-word;
+ 160 .font-size(@countDownTimerUnitValueFontSizeMobile);
+ 161 @media (min-width: @breakPointXs) {
+ 162 .font-size(@countDownTimerUnitValueFontSize);
+ 163 }
+ 164
+ 165 &__with-background {
+ 166 display: block;
+ 167 background-color: var(--timeUnitBackgroundColor);
+ 168 border-radius: ~'calc(var(--borderRadius) * 1%)';
+ 169 padding: @countDownTimerUnitValuePaddingMobile;
+ 170 color: var(--textColor, inherit);
+ 171
+ 172 @media (min-width: @breakPointXs) {
+ 173 padding: @countDownTimerUnitValuePadding;
+ 174 }
+ 175 }
+ 176 }
+ 177
+ 178 &__label {
+ 179 text-align: center;
+ 180 .font-size(@countDownTimerUnitLabelFontSizeMobile);
+ 181 @media (min-width: @breakPointXs) {
+ 182 .font-size(@countDownTimerUnitLabelFontSize);
+ 183 }
+ 184 }
+ 185
+ 186 &__separator {
+ 187 text-align: center;
+ 188 width: @countDownTimerSeparatorWidth;
+ 189 padding-top: @countDownTimerUnitValuePaddingMobile;
+ 190 .font-size(@countDownTimerUnitValueFontSizeMobile);
+ 191
```

```
+ 192 @media (min-width: @breakPointXs) {
+ 193     padding-top: @countDownTimerUnitValuePadding;
+ 194     .font-size(@countDownTimerUnitValueFontSize);
+ 195 }
+ 196 }
+ 197
+ 198 &__overlay {
+ 199     position: absolute;
+ 200     inset: 0;
+ 201     z-index: 0;
+ 202     pointer-events: none;
+ 203     background-color: var(--overlayBackgroundColor);
+ 204 }
+ 205 }
```

countdown_timer/countdown_timer_variables.less

```
+ 1 @countDownTimerHeightSmall: 356px;
+ 2 @countDownTimerHeightMedium: 400px;
+ 3 @countDownTimerHeightLarge: 516px;
+ 4 @countDownTimerContainerGap: @globalSpacing * 2;
+ 5
+ 6 @countDownTimerPaddingMobile: @globalSpacing;
+ 7 @countDownTimerPadding: @globalSpacing * 2.5;
+ 8
+ 9 @countDownTimerContentMaxWidth: 720px;
+ 10
+ 11 @countDownTimerUnitMinWidthMobile: 45px;
+ 12 @countDownTimerUnitMinWidth: 76px;
+ 13 @countDownTimerUnitValuePaddingMobile: @globalSpacing * 0.75;
+ 14 @countDownTimerUnitValuePadding: @globalSpacing * 1.25;
+ 15
+ 16 @countDownTimerUnitValueFontSizeMobile: @baseFontSize;
+ 17 @countDownTimerUnitValueFontSize: @baseFontSize * 1.75;
+ 18
+ 19 @countDownTimerUnitLabelFontSizeMobile: @baseFontSize * 0.625;
+ 20 @countDownTimerUnitLabelFontSize: @baseFontSize * 0.875;
+ 21
+ 22 @countDownTimerUnitGapMobile: @globalSpacing * 0.5;
+ 23 @countDownTimerUnitGap: @globalSpacing * 1.5;
+ 24
+ 25 @countDownTimerUnitWithSeparatorGapMobile: @countDownTimerUnitGapMobile * 0.5;
+ 26 @countDownTimerUnitWithSeparatorGap: @countDownTimerUnitGap * 0.5;
+ 27
+ 28 @countDownTimerSeparatorWidth: 0.375rem;
+ 29 @countDownTimerDescriptionFontSize: @baseFontSize;
```

styles.less

```
174 @import 'visual_subcategories/visual_subcategories_variables.less';
175 @import 'icon_with_text/icon_with_text_variables.less';
176 @import 'faq/faq_variables.less';
+ 177 @import 'countdown_timer/countdown_timer_variables.less';
178
179 /**
180  * Import actual styles
```

```
360 @import 'featured_product/featured_product.less';
361 @import 'sso/sso.less';
362 @import 'faq/faq.less';
+ 363 @import 'countdown_timer/countdown_timer.less';
364
365 @import 'consents/consents.less';
366 @import 'blik0/blik0.less';
```

utils/utils.less

```
23 }
24
25 .resetcss {
- 26
27     em {
28         font-style: italic;
29     }
```

```
106     align-items: end;
107 }
108
+ 109 .justify-items-center {
+ 110     justify-items: center;
+ 111 }
+ 112
+ 113 .justify-items-start {
+ 114     justify-items: start;
+ 115 }
+ 116
+ 117 .justify-items-end {
+ 118     justify-items: end;
+ 119 }
+ 120
121 .text-align-center {
122     text-align: center;
123 }
```

Pliki .json

@modules/countdown_timer/config/schema.json

```
+ 1  [
+ 2  {
+ 3    "label": "General settings",
+ 4    "state": "unfolded",
+ 5    "elements": [
+ 6      {
+ 7        "type": "datePicker",
+ 8        "name": "endDateTime",
+ 9        "label": "End date and time",
+10        "isRequired": true,
+11        "options": {
+12          "withTime": true
+13        }
+14      },
+15      {
+16        "name": "endCountdownAction",
+17        "type": "radio",
+18        "label": "Action after countdown ends",
+19        "defaultValue": "hide",
+20        "options": {
+21          "radioOptions": [
+22            {
+23              "key": "hide",
+24              "label": "hide module"
+25            },
+26            {
+27              "key": "showEndBanner",
+28              "label": "display final banner"
+29            }
+30          ]
+31        }
+32      },
+33      {
+34        "name": "backgroundImage",
+35        "type": "imageUpload",
+36        "label": "Background image",
+37        "options": {
+38          "requireImageSize": false,
+39          "allowedExtensions": ["webp", "svg", "jpg", "png", "gif", "jpeg"]
+40        }
+41      },
+42      {
+43        "type": "text",
+44        "name": "alt",
+45        "label": "Alternative image description (\"alt\")",
+46        "supportsTranslations": true,
+47        "hint": "Alternative text (\"alt\" attribute) should contain a description of what the graphic presents. This descripti
```

```

on is read by software for the blind and analyzed when search engines index the page."
+ 48     },
+ 49     {
+ 50         "name": "hasBackgroundOverlay",
+ 51         "type": "checkbox",
+ 52         "label": "Add color overlay",
+ 53         "hint": "Use this if you're placing text over an image. A solid or semi-transparent overlay will improve text readability.",
+ 54         "defaultValue": 0,
+ 55         "children": [
+ 56             {
+ 57                 "name": "backgroundOverlay",
+ 58                 "type": "colorPicker",
+ 59                 "label": "Overlay color",
+ 60                 "defaultValue": "#FFFFFF4D",
+ 61                 "relations": [
+ 62                     {
+ 63                         "parentName": "hasBackgroundOverlay",
+ 64                         "parentValueToActionsMap": [
+ 65                             {
+ 66                                 "value": 0,
+ 67                                 "actions": ["setHidden", "setDisabled"]
+ 68                             },
+ 69                             {
+ 70                                 "value": 1,
+ 71                                 "actions": ["setVisible", "setAvailable"]
+ 72                             }
+ 73                         ]
+ 74                     }
+ 75                 ],
+ 76                 "options": {
+ 77                     "allowVariables": true
+ 78                 }
+ 79             }
+ 80         ]
+ 81     },
+ 82     {
+ 83         "name": "bannerSize",
+ 84         "type": "select",
+ 85         "label": "Banner size",
+ 86         "defaultValue": "small",
+ 87         "options": {
+ 88             "selectOptions": [
+ 89                 {
+ 90                     "key": "large",
+ 91                     "label": "Large <<height>>"
+ 92                 },
+ 93                 {
+ 94                     "key": "medium",
+ 95                     "label": "Medium"

```

```
+ 96     },
+ 97     {
+ 98         "key": "small",
+ 99         "label": "Small <<height>>"
+ 100     }
+ 101 ]
+ 102 }
+ 103 },
+ 104 {
+ 105     "name": "contentPosition",
+ 106     "type": "select",
+ 107     "label": "Content position",
+ 108     "defaultValue": "center",
+ 109     "options": {
+ 110         "selectOptions": [
+ 111             {
+ 112                 "key": "bottom-right",
+ 113                 "label": "Bottom right"
+ 114             },
+ 115             {
+ 116                 "key": "bottom-center",
+ 117                 "label": "Bottom center"
+ 118             },
+ 119             {
+ 120                 "key": "bottom-left",
+ 121                 "label": "Bottom left"
+ 122             },
+ 123             {
+ 124                 "key": "center-right",
+ 125                 "label": "Middle right"
+ 126             },
+ 127             {
+ 128                 "key": "center",
+ 129                 "label": "Center <<in_the_middle>>"
+ 130             },
+ 131             {
+ 132                 "key": "center-left",
+ 133                 "label": "Middle left"
+ 134             },
+ 135             {
+ 136                 "key": "top-right",
+ 137                 "label": "Top right"
+ 138             },
+ 139             {
+ 140                 "key": "top-center",
+ 141                 "label": "Top center"
+ 142             },
+ 143             {
+ 144                 "key": "top-left",
+ 145                 "label": "Top left"
```

```
+ 146 }
+ 147 ]
+ 148 }
+ 149 },
+ 150 {
+ 151     "name": "contentAlignment",
+ 152     "type": "select",
+ 153     "options": {
+ 154         "selectOptions": [
+ 155             {
+ 156                 "key": "start",
+ 157                 "label": "Left <<to_left>>"
+ 158             },
+ 159             {
+ 160                 "key": "center",
+ 161                 "label": "Center"
+ 162             },
+ 163             {
+ 164                 "key": "end",
+ 165                 "label": "Right <<to_right>>"
+ 166             }
+ 167         ]
+ 168     },
+ 169     "label": "Content alignment",
+ 170     "defaultValue": "center"
+ 171 }
+ 172 ]
+ 173 },
+ 174 {
+ 175     "label": "Counter",
+ 176     "state": "unfolded",
+ 177     "elements": [
+ 178         {
+ 179             "name": "digitsTextColor",
+ 180             "type": "colorPicker",
+ 181             "label": "Digit color",
+ 182             "defaultValue": "@globalFontColor",
+ 183             "options": {
+ 184                 "allowVariables": true
+ 185             }
+ 186         },
+ 187         {
+ 188             "name": "digitsBackgroundColor",
+ 189             "type": "colorPicker",
+ 190             "label": "Digit background color",
+ 191             "defaultValue": "@globalBodyBackgroundColor",
+ 192             "options": {
+ 193                 "allowVariables": true
+ 194             }
+ 195         },
```

```

+ 196 {
+ 197     "type": "number",
+ 198     "name": "borderRadius",
+ 199     "label": "Background rounding",
+ 200     "labelDescription": "0% - rectangular, 100% - round background",
+ 201     "defaultValue": 0,
+ 202     "isRequired": true,
+ 203     "validators": [
+ 204         {
+ 205             "type": "int"
+ 206         },
+ 207         {
+ 208             "type": "greaterEqThan",
+ 209             "options": {
+ 210                 "min": 0
+ 211             }
+ 212         },
+ 213         {
+ 214             "type": "lessEqThan",
+ 215             "options": {
+ 216                 "max": 100
+ 217             }
+ 218         }
+ 219     ],
+ 220     "options": {
+ 221         "postfix": "%"
+ 222     }
+ 223 },
+ 224 {
+ 225     "name": "timeUnitsPosition",
+ 226     "type": "radio",
+ 227     "label": "Time unit display",
+ 228     "defaultValue": "inside",
+ 229     "options": {
+ 230         "radioOptions": [
+ 231             {
+ 232                 "key": "inside",
+ 233                 "label": "inside digit tile"
+ 234             },
+ 235             {
+ 236                 "key": "under",
+ 237                 "label": "below digit tile"
+ 238             }
+ 239         ]
+ 240     }
+ 241 },
+ 242 {
+ 243     "name": "shouldShowSeparator",
+ 244     "type": "checkbox",
+ 245     "label": "Display separator between digits",

```

```

+ 246         "defaultValue": 1
+ 247     }
+ 248 ]
+ 249 },
+ 250 {
+ 251     "label": "Countdown banner",
+ 252     "state": "unfolded",
+ 253     "elements": [
+ 254         {
+ 255             "type": "checkbox",
+ 256             "name": "hasActiveCounterHeader",
+ 257             "label": "Display heading",
+ 258             "defaultValue": 1,
+ 259             "children": [
+ 260                 {
+ 261                     "type": "text",
+ 262                     "name": "activeCounterHeaderContent",
+ 263                     "label": "Module heading content",
+ 264                     "defaultValue": "",
+ 265                     "supportsTranslations": 1,
+ 266                     "relations": [
+ 267                         {
+ 268                             "parentName": "hasActiveCounterHeader",
+ 269                             "parentValueToActionsMap": [
+ 270                                 {
+ 271                                     "value": 0,
+ 272                                     "actions": ["setHidden", "setDisabled", "setOptional"]
+ 273                                 },
+ 274                                 {
+ 275                                     "value": 1,
+ 276                                     "actions": ["setVisible", "setAvailable", "setRequired"]
+ 277                                 }
+ 278                             ]
+ 279                         }
+ 280                     ]
+ 281                 },
+ 282                 {
+ 283                     "type": "select",
+ 284                     "name": "activeCounterHeaderLevel",
+ 285                     "label": "Heading level",
+ 286                     "hint": "The numbers 2 to 6 indicate the hierarchy of headings, with H2 being the most important and H6 being t
he least. H1 is reserved for the page title. If you need to add a page title, use the \"Page title\" module.",
+ 287                     "defaultValue": "h2",
+ 288                     "options": {
+ 289                         "selectOptions": [
+ 290                             {
+ 291                                 "key": "h2",
+ 292                                 "label": "H2"
+ 293                             },
+ 294                             {

```

```
+ 295         "key": "h3",
+ 296         "label": "H3"
+ 297     },
+ 298     {
+ 299         "key": "h4",
+ 300         "label": "H4"
+ 301     },
+ 302     {
+ 303         "key": "h5",
+ 304         "label": "H5"
+ 305     },
+ 306     {
+ 307         "key": "h6",
+ 308         "label": "H6"
+ 309     }
+ 310 ]
+ 311 },
+ 312     "relations": [
+ 313     {
+ 314         "parentName": "hasActiveCounterHeader",
+ 315         "parentValueToActionsMap": [
+ 316         {
+ 317             "value": 0,
+ 318             "actions": ["setHidden", "setDisabled", "setOptional"]
+ 319         },
+ 320         {
+ 321             "value": 1,
+ 322             "actions": ["setVisible", "setAvailable", "setRequired"]
+ 323         }
+ 324     ]
+ 325     }
+ 326 ]
+ 327 }
+ 328 ]
+ 329 },
+ 330 {
+ 331     "name": "hasActiveCounterDescription",
+ 332     "type": "checkbox",
+ 333     "label": "Display additional description",
+ 334     "defaultValue": 0,
+ 335     "children": [
+ 336     {
+ 337         "type": "textarea",
+ 338         "name": "activeCounterDescription",
+ 339         "label": "Description",
+ 340         "supportsTranslations": true,
+ 341         "relations": [
+ 342         {
+ 343             "parentName": "hasActiveCounterDescription",
+ 344             "parentValueToActionsMap": [
```

```
+ 345 {
+ 346     "value": 0,
+ 347     "actions": ["setHidden", "setDisabled", "setOptional"]
+ 348 },
+ 349 {
+ 350     "value": 1,
+ 351     "actions": ["setVisible", "setAvailable", "setRequired"]
+ 352 }
+ 353 ]
+ 354 }
+ 355 ]
+ 356 },
+ 357 {
+ 358     "name": "activeCounterDescriptionPlacement",
+ 359     "type": "select",
+ 360     "label": "Description position",
+ 361     "defaultValue": "above",
+ 362     "options": {
+ 363         "selectOptions": [
+ 364             {
+ 365                 "key": "above",
+ 366                 "label": "Above counter"
+ 367             },
+ 368             {
+ 369                 "key": "below",
+ 370                 "label": "Below counter"
+ 371             }
+ 372         ]
+ 373     },
+ 374     "relations": [
+ 375         {
+ 376             "parentName": "hasActiveCounterDescription",
+ 377             "parentValueToActionsMap": [
+ 378                 {
+ 379                     "value": 0,
+ 380                     "actions": ["setHidden", "setDisabled", "setOptional"]
+ 381                 },
+ 382                 {
+ 383                     "value": 1,
+ 384                     "actions": ["setVisible", "setAvailable", "setRequired"]
+ 385                 }
+ 386             ]
+ 387         }
+ 388     ]
+ 389 }
+ 390 ]
+ 391 },
+ 392 {
+ 393     "type": "checkbox",
+ 394     "name": "hasActiveCounterButton",
```

```
+ 395     "label": "Add button",
+ 396     "defaultValue": 0,
+ 397     "children": [
+ 398       {
+ 399         "type": "text",
+ 400         "name": "activeCounterButtonContent",
+ 401         "label": "Button content",
+ 402         "supportsTranslations": 1,
+ 403         "relations": [
+ 404           {
+ 405             "parentName": "hasActiveCounterButton",
+ 406             "parentValueToActionsMap": [
+ 407               {
+ 408                 "value": 0,
+ 409                 "actions": ["setHidden", "setDisabled", "setOptional"]
+ 410               },
+ 411               {
+ 412                 "value": 1,
+ 413                 "actions": ["setVisible", "setAvailable", "setRequired"]
+ 414               }
+ 415             ]
+ 416           }
+ 417         ]
+ 418       },
+ 419       {
+ 420         "type": "select",
+ 421         "name": "activeCounterButtonType",
+ 422         "label": "Button type",
+ 423         "defaultValue": "primary",
+ 424         "options": {
+ 425           "selectOptions": [
+ 426             {
+ 427               "key": "primary",
+ 428               "label": "Primary <<button>>"
+ 429             },
+ 430             {
+ 431               "key": "secondary",
+ 432               "label": "Secondary"
+ 433             },
+ 434             {
+ 435               "key": "outline",
+ 436               "label": "Outline"
+ 437             },
+ 438             {
+ 439               "key": "ghost",
+ 440               "label": "Ghost (text only)"
+ 441             }
+ 442           ]
+ 443         },
+ 444         "relations": [
```

```

+ 445     {
+ 446         "parentName": "hasActiveCounterButton",
+ 447         "parentValueToActionsMap": [
+ 448             {
+ 449                 "value": 0,
+ 450                 "actions": ["setHidden", "setDisabled", "setOptional"]
+ 451             },
+ 452             {
+ 453                 "value": 1,
+ 454                 "actions": ["setVisible", "setAvailable", "setRequired"]
+ 455             }
+ 456         ]
+ 457     }
+ 458 ]
+ 459 },
+ 460 {
+ 461     "type": "select",
+ 462     "name": "activeCounterButtonSize",
+ 463     "label": "Button size",
+ 464     "defaultValue": "m",
+ 465     "options": {
+ 466         "selectOptions": [
+ 467             {
+ 468                 "key": "xs",
+ 469                 "label": "XS"
+ 470             },
+ 471             {
+ 472                 "key": "s",
+ 473                 "label": "S"
+ 474             },
+ 475             {
+ 476                 "key": "m",
+ 477                 "label": "M"
+ 478             },
+ 479             {
+ 480                 "key": "l",
+ 481                 "label": "L"
+ 482             }
+ 483         ]
+ 484     },
+ 485     "relations": [
+ 486         {
+ 487             "parentName": "hasActiveCounterButton",
+ 488             "parentValueToActionsMap": [
+ 489                 {
+ 490                     "value": 0,
+ 491                     "actions": ["setHidden", "setDisabled", "setOptional"]
+ 492                 },
+ 493                 {
+ 494                     "value": 1,

```

```
+ 495         "actions": ["setVisible", "setAvailable", "setRequired"]
+ 496     }
+ 497 ]
+ 498 }
+ 499 ],
+ 500 },
+ 501 {
+ 502     "type": "text",
+ 503     "name": "activeCounterButtonUrl",
+ 504     "label": "Webpage URL",
+ 505     "options": {
+ 506         "placeholder": "https:// or /"
+ 507     },
+ 508     "supportsTranslations": 1,
+ 509     "validators": [
+ 510         {
+ 511             "type": "url",
+ 512             "options": {
+ 513                 "allowRelativePath": true
+ 514             }
+ 515         }
+ 516     ],
+ 517     "relations": [
+ 518         {
+ 519             "parentName": "hasActiveCounterButton",
+ 520             "parentValueToActionsMap": [
+ 521                 {
+ 522                     "value": 0,
+ 523                     "actions": ["setHidden", "setDisabled", "setOptional"]
+ 524                 },
+ 525                 {
+ 526                     "value": 1,
+ 527                     "actions": ["setVisible", "setAvailable", "setRequired"]
+ 528                 }
+ 529             ]
+ 530         }
+ 531     ]
+ 532 },
+ 533 {
+ 534     "type": "checkbox",
+ 535     "name": "shouldOpenActiveCounterLinkInTheSameTab",
+ 536     "label": "Open the link in the same tab",
+ 537     "hint": "Opening links in the same tab makes navigation easier: using the \"back\" button is possible. The customer can decide to open the page in a new tab.",
+ 538     "defaultValue": 1,
+ 539     "relations": [
+ 540         {
+ 541             "parentName": "hasActiveCounterButton",
+ 542             "parentValueToActionsMap": [
+ 543                 {
```

```
+ 544         "value": 0,
+ 545         "actions": ["setHidden", "setDisabled"]
+ 546     },
+ 547     {
+ 548         "value": 1,
+ 549         "actions": ["setVisible", "setAvailable"]
+ 550     }
+ 551 ]
+ 552 }
+ 553 ]
+ 554 }
+ 555 ]
+ 556 }
+ 557 ]
+ 558 },
+ 559 {
+ 560     "label": "Final banner",
+ 561     "state": "folded",
+ 562     "elements": [
+ 563     {
+ 564         "type": "checkbox",
+ 565         "name": "hasEndBannerHeader",
+ 566         "label": "Display heading",
+ 567         "defaultValue": 0,
+ 568         "children": [
+ 569         {
+ 570             "type": "text",
+ 571             "name": "endBannerHeaderContent",
+ 572             "label": "Module heading content",
+ 573             "defaultValue": "",
+ 574             "supportsTranslations": 1,
+ 575             "relations": [
+ 576             {
+ 577                 "parentName": "hasEndBannerHeader",
+ 578                 "parentValueToActionsMap": [
+ 579                 {
+ 580                     "value": 0,
+ 581                     "actions": ["setHidden", "setDisabled", "setOptional"]
+ 582                 },
+ 583                 {
+ 584                     "value": 1,
+ 585                     "actions": ["setVisible", "setAvailable", "setRequired"]
+ 586                 }
+ 587             ]
+ 588             }
+ 589             ]
+ 590             },
+ 591             {
+ 592                 "type": "select",
+ 593                 "name": "endBannerHeaderLevel",
```

```
+ 594         "label": "Heading level",
+ 595         "hint": "The numbers 2 to 6 indicate the hierarchy of headings, with H2 being the most important and H6 being t
he least. H1 is reserved for the page title. If you need to add a page title, use the \"Page title\" module.",
+ 596         "defaultValue": "h2",
+ 597         "options": {
+ 598             "selectOptions": [
+ 599                 {
+ 600                     "key": "h2",
+ 601                     "label": "H2"
+ 602                 },
+ 603                 {
+ 604                     "key": "h3",
+ 605                     "label": "H3"
+ 606                 },
+ 607                 {
+ 608                     "key": "h4",
+ 609                     "label": "H4"
+ 610                 },
+ 611                 {
+ 612                     "key": "h5",
+ 613                     "label": "H5"
+ 614                 },
+ 615                 {
+ 616                     "key": "h6",
+ 617                     "label": "H6"
+ 618                 }
+ 619             ]
+ 620         },
+ 621         "relations": [
+ 622             {
+ 623                 "parentName": "hasEndBannerHeader",
+ 624                 "parentValueToActionsMap": [
+ 625                     {
+ 626                         "value": 0,
+ 627                         "actions": ["setHidden", "setDisabled", "setOptional"]
+ 628                     },
+ 629                     {
+ 630                         "value": 1,
+ 631                         "actions": ["setVisible", "setAvailable", "setRequired"]
+ 632                     }
+ 633                 ]
+ 634             }
+ 635         ]
+ 636     }
+ 637 ]
+ 638 },
+ 639 {
+ 640     "name": "hasEndBannerDescription",
+ 641     "type": "checkbox",
+ 642     "label": "Display additional description below the heading",
```

```
+ 643     "defaultValue": 0,
+ 644     "children": [
+ 645       {
+ 646         "type": "textarea",
+ 647         "name": "endBannerDescription",
+ 648         "label": "Description",
+ 649         "supportsTranslations": true,
+ 650         "relations": [
+ 651           {
+ 652             "parentName": "hasEndBannerDescription",
+ 653             "parentValueToActionsMap": [
+ 654               {
+ 655                 "value": 0,
+ 656                 "actions": ["setHidden", "setDisabled", "setOptional"]
+ 657               },
+ 658               {
+ 659                 "value": 1,
+ 660                 "actions": ["setVisible", "setAvailable", "setRequired"]
+ 661               }
+ 662             ]
+ 663           }
+ 664         ]
+ 665       }
+ 666     ],
+ 667   },
+ 668   {
+ 669     "type": "checkbox",
+ 670     "name": "hasEndBannerButton",
+ 671     "label": "Add button",
+ 672     "defaultValue": 0,
+ 673     "children": [
+ 674       {
+ 675         "type": "text",
+ 676         "name": "endBannerButtonContent",
+ 677         "label": "Button content",
+ 678         "supportsTranslations": 1,
+ 679         "relations": [
+ 680           {
+ 681             "parentName": "hasEndBannerButton",
+ 682             "parentValueToActionsMap": [
+ 683               {
+ 684                 "value": 0,
+ 685                 "actions": ["setHidden", "setDisabled", "setOptional"]
+ 686               },
+ 687               {
+ 688                 "value": 1,
+ 689                 "actions": ["setVisible", "setAvailable", "setRequired"]
+ 690               }
+ 691             ]
+ 692           }
+ 692         ]
+ 692       }
+ 692     ]
+ 692   }
+ 692 }
```

```
+ 693 ]
+ 694 },
+ 695 {
+ 696     "type": "select",
+ 697     "name": "endBannerButtonType",
+ 698     "label": "Button type",
+ 699     "defaultValue": "primary",
+ 700     "options": {
+ 701         "selectOptions": [
+ 702             {
+ 703                 "key": "primary",
+ 704                 "label": "Primary <<button>>"
+ 705             },
+ 706             {
+ 707                 "key": "secondary",
+ 708                 "label": "Secondary"
+ 709             },
+ 710             {
+ 711                 "key": "outline",
+ 712                 "label": "Outline"
+ 713             },
+ 714             {
+ 715                 "key": "ghost",
+ 716                 "label": "Ghost (text only)"
+ 717             }
+ 718         ]
+ 719     },
+ 720     "relations": [
+ 721         {
+ 722             "parentName": "hasEndBannerButton",
+ 723             "parentValueToActionsMap": [
+ 724                 {
+ 725                     "value": 0,
+ 726                     "actions": ["setHidden", "setDisabled", "setOptional"]
+ 727                 },
+ 728                 {
+ 729                     "value": 1,
+ 730                     "actions": ["setVisible", "setAvailable", "setRequired"]
+ 731                 }
+ 732             ]
+ 733         }
+ 734     ]
+ 735 },
+ 736 {
+ 737     "type": "select",
+ 738     "name": "endBannerButtonSize",
+ 739     "label": "Button size",
+ 740     "defaultValue": "m",
+ 741     "options": {
+ 742         "selectOptions": [
```

```
+ 743 {
+ 744     "key": "xs",
+ 745     "label": "XS"
+ 746 },
+ 747 {
+ 748     "key": "s",
+ 749     "label": "S"
+ 750 },
+ 751 {
+ 752     "key": "m",
+ 753     "label": "M"
+ 754 },
+ 755 {
+ 756     "key": "l",
+ 757     "label": "L"
+ 758 }
+ 759 ]
+ 760 },
+ 761     "relations": [
+ 762     {
+ 763         "parentName": "hasEndBannerButton",
+ 764         "parentValueToActionsMap": [
+ 765         {
+ 766             "value": 0,
+ 767             "actions": ["setHidden", "setDisabled", "setOptional"]
+ 768         },
+ 769         {
+ 770             "value": 1,
+ 771             "actions": ["setVisible", "setAvailable", "setRequired"]
+ 772         }
+ 773     ]
+ 774     }
+ 775 ]
+ 776 },
+ 777 {
+ 778     "type": "text",
+ 779     "name": "endBannerButtonUrl",
+ 780     "label": "Webpage URL",
+ 781     "options": {
+ 782         "placeholder": "https:// or /"
+ 783     },
+ 784     "supportsTranslations": 1,
+ 785     "validators": [
+ 786     {
+ 787         "type": "url",
+ 788         "options": {
+ 789             "allowRelativePath": true
+ 790         }
+ 791     }
+ 792 ],
```

```

+ 793     "relations": [
+ 794         {
+ 795             "parentName": "hasEndBannerButton",
+ 796             "parentValueToActionsMap": [
+ 797                 {
+ 798                     "value": 0,
+ 799                     "actions": ["setHidden", "setDisabled", "setOptional"]
+ 800                 },
+ 801                 {
+ 802                     "value": 1,
+ 803                     "actions": ["setVisible", "setAvailable", "setRequired"]
+ 804                 }
+ 805             ]
+ 806         }
+ 807     ]
+ 808 },
+ 809 {
+ 810     "type": "checkbox",
+ 811     "name": "shouldOpenEndBannerLinkInTheSameTab",
+ 812     "label": "Open the link in the same tab",
+ 813     "hint": "Opening links in the same tab makes navigation easier: using the \"back\" button is possible. The cust
omer can decide to open the page in a new tab.",
+ 814     "defaultValue": 1,
+ 815     "relations": [
+ 816         {
+ 817             "parentName": "hasEndBannerButton",
+ 818             "parentValueToActionsMap": [
+ 819                 {
+ 820                     "value": 0,
+ 821                     "actions": ["setHidden", "setDisabled"]
+ 822                 },
+ 823                 {
+ 824                     "value": 1,
+ 825                     "actions": ["setVisible", "setAvailable"]
+ 826                 }
+ 827             ]
+ 828         }
+ 829     ]
+ 830 }
+ 831 ]
+ 832 }
+ 833 ]
+ 834 },
+ 835 {
+ 836     "label": "CSS",
+ 837     "state": "folded",
+ 838     "elements": [
+ 839         {
+ 840             "name": "classNames",
+ 841             "type": "text",

```

```
+ 842     "label": "CSS class",
+ 843     "labelDescription": "Enter the class name without a dot at the beginning. You can add multiple classes by separating th
em with spaces."
+ 844   }
+ 845 ]
+ 846 }
+ 847 ]
```