

# Dodatek C: Instalacja widżetu CHContext

CHContext (<https://github.com/psnc-dl/chcontext/>) to javascriptowy widżet dostarczający listę odnośników do zasobów dziedzictwa kulturowego utworzonych na podstawie wyników wyszukiwania bazującego na zawartości określonego elementu strony WWW, na której widżet jest osadzony.

Widżet wspiera trzy źródła danych:

- [Federacja Bibliotek Cyfrowych](#) (nie wymaga klucza API)
- [Europeana](#) (wymaga klucza API)
- [Digital Public Library of America](#) (wymaga klucza API).

Przykłady użycia:

- Repozytorium Instytucjonalne PCSS: <http://lib.psnec.pl/publication/517>
- Cyfrowe Archiwum Pamiętek: <http://fbc.pionier.net.pl/zbiorki/publication/117>

W przypadku pytań lub problemów prosimy o kontakt: <http://dl.psnec.pl/contact/>.

## Instalacja instancji

Dodaj następujący fragment kodu do pliku `dlibra-webapp/WEB-INF/layout/templates/simple-inv.vm` pomiędzy liniami `#foreach( $comp in $left )` a `#if ( ${comp.RenderedTemplate} != " " )`

**dlibra-webapp/WEB-INF/layout/templates/simple-inv.vm**

```
#if(4 == ${velocityCount} && ${pageId} == "docmetadata")

    <div id="chcontext-widget" class="chcontext-widget-wrapper" data-searchProvider="FBC+" data-resultCount="5"
    data-queryselector="span#Subject a" data-show-img="false">
        <h3 id="chc-widget-header" class="leftDiv"> $res.getProperty("CHContext.SeeAlso") </h3>
        <div class="chcontext-widget-container left_box"></div>
    </div>

    <script type="text/javascript" src="http://cdn.jsdelivr.net/chcontext/1.1.0/chcontext.min.js"></script>
#end
```

Powyższa konfiguracja widżetu zakłada, że:

- wyszukiwanie będzie realizowane poprzez FBC+ (`data-searchProvider="FBC+"`)
- wyświetlanych będzie 5 elementów (`data-resultCount="5"`)
- wyszukiwana będzie zawartość elementu wskazanego poprzez selektor `span#Subject a` (`data-queryselector="span#Subject a"`), czyli słowa kluczowe
- nie będą wyświetlane miniaturki (`data-show-img="false"`)
- w dlibrze zdefiniowane są etykiety tekstowe dla klucza `CHContext.SeeAlso`
- temat kolorystyczny to `theme1`.

Następnie należy zdefiniować etykiety tekstowe dla klucza `CHContext.SeeAlso`. W tym celu w pliku `dlibra-webapp/WEB-INF/components/resources/WEBAPP_pl.xml` dodajemy wpis:

**dlibra-webapp/WEB-INF/components/resources/WEBAPP\_pl.xml**

```
<entry key="CHContext.SeeAlso">Zobacz take</entry>
```

Analogicznie w pliku `dlibra-webapp/WEB-INF/components/resources/WEBAPP_en.xml` dodajemy:

**dlibra-webapp/WEB-INF/components/resources/WEBAPP\_en.xml**

```
<entry key="CHContext.SeeAlso">See also</entry>
```

Chcemy wyszukiwać podobne obiekty bazując na słowach kluczowych. Aby to zrealizować, musimy zmienić kod HTML strony z opisem publikacji w ten sposób, aby umożliwić dostęp do nich. W tym celu musimy zmodyfikować makro `dispNode`.

Do pliku `dlibra-webapp/WEB-INF/components/templates/custom_library.vm` dodajmy następującą treść (treść tego makra można znaleźć w pliku `custom_library.vm`, który znajduje się w archiwum `dlibra-webapp/WEB-INF/lib/dcore-web-components-x.x.x.jar`):

#### `dlibra-webapp/WEB-INF/components/templates/custom_library.vm`

```
#macro ( dispNode $node $info)
  <span class="attributeJustify">
    #if($documentMetadata.getAttributeValues($info.Id, $metadataLanguage, $valueLangType).size() > 0)
      <a href="{homepageUrl}${servletName}/indexsearch?attId=${info.id}" class="attributeInfo">
        #set($t = "#printParentsLabels($node 0)")
        #if($sprintAttParents)
          ${t.trim()}:
        #else
          ${node.Info.Label}:
        #end
      </a>
    <ul>
      <li class="attributeValuesListElement">
        <span class="attribute" style="margin:0;">
          #foreach($value in $documentMetadata.getAttributeValues($info.Id, $metadataLanguage,
$valueLangType))
            #if($velocityCount > 1); #end
            #if($value.toString().startsWith('http://'))
              #set($urlName = ${res.getProperty("element_metadata.link")})
              #set($urlValue = $value)
              #if($value.toString().indexOf(" ") > 0)
                #set($urlValue = $value.toString().substring(0,$value.toString().indexOf(" "))
                #set($urlName = $value.toString().substring($value.toString().indexOf(" "), $value.toString().
length()))
              #end
              <a href="{urlValue}" target="_blank">$urlName</a>
              <a href="{urlValue}" target="_blank">
                </a>
            #elseif($value.toString().startsWith('oai:'))
              #if(${conf.get("use.fbc")})
                <a href="{conf.get("fbc.url")}/id/${value}">${value}</a>
              #else
                <a href="{homepageUrl}${servletName}/docmetadata?id=${value}">${value}</a>
              #end
            #elseif(!$info.getRoleId().toString().equals("description"))
              #set ( $strValue = $value.toString() )
              <script type="text/javascript"><!--
                function submit${info.Id}_${velocityCount}() {
                  submitQuery("${info.Id}','${escapeUtil.escapeForJS($strValue)}\''');
                }
              //--> </script>
              <a href="javascript:submit${info.Id}_${velocityCount}()">${escapeUtil.escapeHtml("${value}")}</a>
            ## perform slice of long data but considering only "description" attributes
            #else
              #set ( $strValue = $value.toString() )
              <script type="text/javascript"><!--
                function submit${info.Id}_${velocityCount}() {
                  return '\${escapeUtil.escapeForJS($strValue)}\'';
                }
              //--> </script>
              <a href="javascript:submit${info.Id}_${velocityCount}()" id="attr_anch_${info.Id}_${velocityCount}"
>${escapeUtil.escapeHtml("${value}")}</a>
            #end
          #end
        </span>
      </li>
    </ul>
  #end
</span>
#end
```

a następnie zamieńmy linie:

```
<span class="attribute" style="margin:0;">
```

na:

```
<span class="attribute" style="margin:0;" id="{info.RDFName}">
```

Pozostało tylko zmodyfikowanie domyślnego wyglądu, poprzez dodanie następującego kodu w pliku `dlibra-webapp/style/dlibra/theme1/simple-inv.css` (przy założeniu, że wykorzystywany temat kolorystyczny to `theme1`):

#### **dlibra-webapp/style/dlibra/theme1/simple-inv.css**

```
#chcontext-widget ol {  
    padding-left: 15px;  
    margin: 0;  
}
```