

How-to: Widget-Einbindung

Am Beispiel von Wordpress

13.03.2023



VisitorApp by IT-Kompass GmbH

Hagenbucher Mühle 1

73072 Donzdorf

Mail: hello@visitorapp.io

Web: about.visitorapp.io

1 Einleitung

Es gibt zwei Varianten, um das VisitorApp-Widget auf einer Website einzubinden:

1. Mit einer **statischen** Angebotsübersicht:
Man hat nur wenige Angebote und fügt diese einzeln in eine Übersichtsseite ein.
2. Mit einer **dynamischen** Angebotsübersicht:
Man hat viele Angebote und die Angebotsliste auf der Übersichtsseite wird dynamisch aufgebaut.

2 Vorbereitung und Voraussetzungen

2.1 Wordpress-Plugins

Möchte man das VisitorApp-Widget mit Wordpress verwenden, dann ist die Installation von zusätzlichen Plugins erforderlich:

1. Ein Plugin zur Einbindung von HTML-Code-Snippets in den <head>-Bereich einer Seite, wie z.B. „[Header Footer Code Manager](#)“
2. Ein Plugin zur Einbindung von PHP-Code-Snippets in eine Seite, wie z.B. „[Insert PHP Code Snippet](#)“

Die Anleitung zur Verwendung dieser Plugins kann auf den verlinkten Seiten nachgelesen werden und ist nicht Teil dieser Dokumentation.

2.2 Seiten

Es sollten außerdem zwei Seiten angelegt werden:

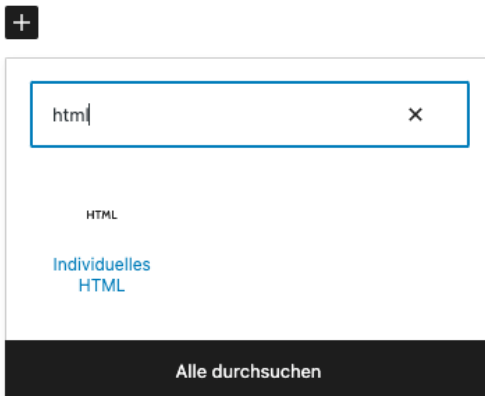
1. Angebotsübersicht: Enthält die Listenansicht mit allen Angeboten
2. Angebotsseite: Enthält das eigentliche Widget mit der Einzelansicht eines Angebots
Die URL zu dieser Seite wird später benötigt, daher sollte man sich diese notieren.

3 Aufbau der Seiten

3.1 Angebotsübersicht

3.1.1 Statische Variante

Um die Angebote darzustellen, wird das Wordpress-Inhaltselement für „Individuelles HTML“ benötigt:



Hier fügt man für jedes Angebot das folgende Code-Snippet ein. Der HTML-Code in folgendem Beispiel kann beliebig verändert und an die eigenen Anforderungen angepasst werden. Wichtig ist nur der rot markierte Teil mit dem Button:

```
<div class="offer">
  <div class="offer-inner">
    <div class="offer-details ">
      [Name und Beschreibung des Angebots]
    </div>
    <div class="offer-tickets">
      <button type="button" onclick="showOffer('[Angebots-ID]')">
        Tickets
      </button>
    </div>
  </div>
</div>
```

Die jeweilige Angebots-ID erhalten Sie vom VisitorApp-Team nach Anlage Ihrer Veranstaltungen und muss ohne die eckigen Klammern eingefügt werden.

Darunter ergänzt man dann nur noch folgenden Javascript-Code, damit der Klick auf den Ticket-Button nicht ins Leere läuft:

```
<script>
  window.showOffer = (offer_id) => {
    window.open("[URL der Detailseite, siehe 2.2]?offer=" + offer_id,
'_blank');
  }
</script>
```

```
</script>
```

3.1.2 Dynamische Variante

Die VisitorApp enthält eine Schnittstelle (API), von der die Angebote einfach vom VisitorApp-Server geladen werden können. Hierfür benötigen Sie Ihre Anbieter-ID. Diese erhalten Sie vom VisitorApp-Team nach Anlage Ihrer Veranstaltungen. Sie muss immer ohne die eckigen Klammern eingefügt werden.

Die URL zur VisitorApp-API lautet:

```
https://www.visitorapp.io/channel/offers/organisations/[Ihre Anbieter-ID]
```

Greift man darauf per GET-Abfrage zu, muss der folgende Header mitgegeben werden:

```
juice-channel: marketplace:channel:default
```

Dies kann mit der PHP-Bibliothek cURL erreicht werden. Die einzelnen Schritte werden nachfolgend erklärt.

An dieser Stelle kommt das unter 2.1 erwähnte Plugin zur Einbindung von PHP-Code zum Einsatz. Dort müssen die folgenden Code-Snippets eingefügt werden, um die Angebote von der VisitorApp-API zu laden und mit HTML darzustellen:

```
<?php
$host = 'www.visitorapp.io';

$curl = curl_init();

curl_setopt_array($curl, array(
    CURLOPT_URL => 'https://'.$host.'/channel/offers/organisations/[Ihre Anbieter-ID]',
    CURLOPT_RETURNTRANSFER => true,
    CURLOPT_ENCODING => '',
    CURLOPT_MAXREDIRS => 10,
    CURLOPT_TIMEOUT => 0,
    CURLOPT_FOLLOWLOCATION => true,
    CURLOPT_HTTP_VERSION => CURL_HTTP_VERSION_1_1,
    CURLOPT_CUSTOMREQUEST => 'GET',
    CURLOPT_HTTPHEADER => array(
        'juice-channel: marketplace:channel:default'
    ),
));

$response = curl_exec($curl);
curl_close($curl);

$offers = json_decode($response, true);
```

`$offers` enthält nun alle Angebote als Array. Diese können in einer Schleife ausgegeben werden.

Der HTML-Code in folgendem Beispiel kann beliebig verändert und an die eigenen Anforderungen angepasst werden:

```
<?php foreach ($offers["payload"]["items"] as &$item) { ?>
  <div class="offer">
    <div class="offer-inner">
      <div class="offer-details ">
        <div class="offer-img">
          <?php if (isset($item['media'][0])) { ?>
            "
              onclick="showOffer('<?= $item["_id"] ?>')">
          <?php } ?>
        </div>
        <div class="offer-description">
          <b><?= $item["name"]["de_DE"] ?></b>
          <?php if (isset($item["shortDescription"]["de_DE"])) { ?>
            <br><?= $item["shortDescription"]["de_DE"] ?>
          <?php } ?>
        </div>
      </div>
      <div class="offer-tickets">
        <button type="button onclick="showOffer('<?= $item["_id"] ?>')">
          Tickets
        </button>
      </div>
    </div>
  </div>
<?php } ?>
```

Zum Schluss wird nur noch ein bisschen Javascript benötigt, damit der Klick auf den Ticket-Button nicht ins Leere läuft:

```
<script>
  window.showOffer = (offer_id) => {
    window.open("[URL der Detailseite, siehe 2.2]?offer=" + offer_id,
'_blank');
  }
</script>
```

Jetzt kann das Ganze in die Angebotsübersichts-Seite eingefügt werden.

3.2 Detailseite

Auf dieser Seite wird das eigentliche Angebots-Widget eingebunden.

Dafür müssen zunächst verschiedene Javascript- und Stylesheet-Dateien in den <head>-Bereich dieser Seite eingefügt werden. Dies erreicht man am einfachsten mit dem unter 2.1 erwähnten Plugin zur Einbindung von HTML-Code-Snippets.

```
<!-- SCRIPTS -->
<script src="https://s3.eu-central-1.amazonaws.com/cdn.visitorapp.io/js/vue.js"></script>
<script src="https://s3.eu-central-1.amazonaws.com/cdn.visitorapp.io/js/vue-router.js"></script>
<script src="https://s3.eu-central-1.amazonaws.com/cdn.visitorapp.io/js/bootstrap.min.js"></script>
<script src="https://s3.eu-central-1.amazonaws.com/cdn.visitorapp.io/js/httpVueLoader.js"></script>
<script src="https://js.stripe.com/v3"></script>
<script src="https://s3.eu-central-1.amazonaws.com/cdn.visitorapp.io/js/flatpickr.min.js"></script>
<script src="https://cdn.jsdelivr.net/npm/vue-flatpickr-component@8"></script>
<script src="https://npmcdn.com/flatpickr/dist/l10n/de.js"></script>
<script src="https://s3.eu-central-1.amazonaws.com/cdn.visitorapp.io/js/vue-promise-btn.umd.js"></script>
<script
src="https://cdnjs.cloudflare.com/ajax/libs/jquery/3.6.3/jquery.min.js" integrity="sha512-
STof4xmlwgkfm7heWqFJVn58Hm3EtS31XFaagaa8VMReCXAkQnJZ+jEy8PCC/iT18dFy95WcExNH
FTqLyp72eQ==" crossorigin="anonymous" referrerpolicy="no-referrer"></script>
<script
src="https://cdnjs.cloudflare.com/ajax/libs/toastr.js/2.1.4/toastr.min.js"><
/script>

<!-- STYLES -->
<link rel="stylesheet" href="https://s3.eu-central-1.amazonaws.com/cdn.visitorapp.io/style/bootstrap.min.css">
<link rel="stylesheet" href="https://cdnjs.cloudflare.com/ajax/libs/font-awesome/6.2.0/css/all.min.css" integrity="sha512-
xh6O/CkQoPOWddYTDqerRdPCVd1SpvCA9XXcUnZS2FmJNp1coAFzvtCN9BmamE+4aHK8yyUHUSCcJ
HgXloTyT2A==" crossorigin="anonymous" referrerpolicy="no-referrer" />
<link rel="stylesheet" href="https://s3.eu-central-1.amazonaws.com/cdn.visitorapp.io/style/style.css">
<link rel="stylesheet" href="https://s3.eu-central-1.amazonaws.com/cdn.visitorapp.io/style/vue-promise-btn.css">
<link rel="stylesheet" href="https://s3.eu-central-1.amazonaws.com/cdn.visitorapp.io/style/flatpickr.min.css">
<link rel="stylesheet"
href="https://cdnjs.cloudflare.com/ajax/libs/toastr.js/2.1.4/toastr.min.css"
>
```

Das Farbschema des Widgets kann jederzeit beliebig mit einer eigenen Stylesheet-Datei überschrieben werden. Möchte man die vorgegebenen Styles gar nicht erst verwenden, entfernt man einfach die eingefärbte Zeile und lädt stattdessen eine eigene CSS-Datei.

Anschließend wird noch ein letztes Mal das unter 2.1 erwähnte Plugin zur Einbindung von PHP-Code benötigt. Dort muss folgender Code eingefügt werden, um das Angebots-Widget zu laden:

```
<div id="widget">
  <widget-component :offer-id="'<?=$_GET["offer"]; ?>'"
:host="'www.visitorapp.io'"></widget-component>
</div>
<script>
  new Vue({
    el: "#widget",
    components: {
      widgetComponent: httpVueLoader('https://s3.eu-central-
1.amazonaws.com/cdn.visitorapp.io/widget.vue'),
    }
  })
</script>
```