




alvineForm

- Historie
- Installation
 - Bibliotheken
- Konfigurieren
 - Parameter
 - Methoden
 - Events
 - CSS-Klassen
 - Lokalisierung
- HTML-Template
- Anwendungsbeispiele
 - Löschen in einer Tabelle
 - Login-Formular

Dieses Plugin wandelt ein normales HTML-Formular, das seinen Inhalt per POST sendet, in ein Formular um, dessen Inhalt per Ajax gesendet wird.

Historie

Version	Änderungen	Anmerkungen
0.1.0	 ALEJS-72 - Abrufen der Vorgangsdetails... <input type="button" value="STATUS"/>	Neues Plugin löst alvineFormAjax ab <ul style="list-style-type: none">• Plugin umbenannt• neues Flag "ajaxsend" um zu entscheiden zu können, ob das Formular per Ajax gesendet werden soll• neues Event formBeforeSend, wird getriggert bevor das Formular per Ajax abgeschickt wird
0.1.1	 ALEJS-72 - Abrufen der Vorgangsdetails... <input type="button" value="STATUS"/>	.bugfix in der Methode beforeSend wurde die defaults nicht korrekt definiert
0.1.2	 ALEJS-72 - Abrufen der Vorgangsdetails... <input type="button" value="STATUS"/>	bugfix in der Methode setResponseData. Wenn der ResponseSelector nicht mit find() gefunden wird. Wird mit closest() gesucht

Frameworks & Plugins

jQuery	> 1.6
Bootstrap	> 3
Bootstrap-notify	> 3 (optional)

Installation

Javascript- und CSS-Dateien können vom Alvine-CDN geladen werden.

```
//cdn.alvine.io/libs/alvine/form/jquery.alvineform-0.1.2.min.js
```

Bibliotheken

```

<script
src="//cdn.alvine.io/libs/alvine/alvinenotify/jquery.alvinenotify-0.0.1.
min.js"></script>
<script
src="//cdn.alvine.io/libs/alvine/form/jquery.alvineform-0.1.2.min.js"></
script>

```

Konfigurieren

Über die angegebenen Parameter kann das Plugin konfiguriert werden. Die Werte können entweder per Script, als Json im Tag oder als einzelne data-Attribute (data-) im Tag notiert werden. Weitere Details siehe die Seite [Frontend](#).

```

<div id="meinFormular">
  <form class="alvineForm" method="post" action="/index.php"
target="_self"
  data-alvineform-datatype="html"
  data-alvineform-targetselector="#meinFormular"
  data-alvineform-responsesselector="#meinFormular" >

  ...

  </form>
</div>

```

Parameter

Schlüssel	Type	Beschreibung	Standardwert
ajaxsend	boolean	Formular per Ajax senden	true
method	String	Alternatives HTTP-Verb (POST, PUT, DELETE, ...), wenn das Verb im Formular überschrieben werden soll	null
datatype	String	Datentyp nach jquery Standard	json
validation	String	Wird der Wert auf 'ajax' gesetzt, so wird das Formular bei Änderungen zur Validierung an die URL des Formulars gesendet (an die URL wird der Parameter whatif angehängt)	false
notifications	array	Für diese Status-Codes wird eine Notification ausgegeben. Die Werte müssen Zeichenketten sein (keine Integer)	['500', '404', '403']
alertboxTemplate	string	Template für die Fehlermeldungen. Der Platzhalter \${message} wird mit der Meldung ersetzt.	<div....
spinnerTemplate	string	Template für den Spinner	<i class="fa fa-refresh fa-spin"></i>
targetselector	string	Selektor um die Antwort in dieses Element zu ersetzen	
responsesselector	string	Selektor um Elemente in der Antwort auszuschneiden um ausschließlich diesen Inhalt in den targetselector zu schreiben	
beforeseendevent	string	Event Trigger vor dem abgeschickten des Formulars	formBeforeSend
succeessevent	string	Event Trigger für ein erfolgreich abgeschicktes Formular	formSuccess
completedevent	string	Event Trigger wenn eine Ajax-Anfrage abgeschlossen wurde	formCompleted
errorevent	string	Event Trigger wenn beim abschicken des Formulars ein Fehler aufgetreten ist	formError

Methoden

Handler	Beschreibung	Parameter
beforeSend	Vor abschicken des Requests	jqXHR, settings
onSuccess	Im Erfolgsfall	data, textStatus, jqXHR
onCompleted	Request ist abgeschlossen	jqXHR, textStatus
onError	Fehlerfall	jqXHR, textStatus, errorThrown
notify	Benachrichtigung des Benutzers	title, message, CSS-Icon Klasse
handleFailed	Fehlerbehandlung eines Ajax Requests	jqXHR, textStatus, errorThrown
showFailSpinner	Ruft die Methode \$.fn.alvineSpinner('fail', this); auf	
showSuccessSpinner	Ruft die Methode \$.fn.alvineSpinner('success', this); auf	

Events

Handler	Beschreibung	Daten
formBeforeSend	Wird getriggert vor dem abgeschickten des Formulars	jqXHR Objekt
formSuccess	Wird getriggert bei Ajax onSuccess	jqXHR Objekt
formCompleted	Wird getriggert bei Ajax onCompleted	jqXHR Objekt
formError	Wird getriggert bei Ajax onError	jqXHR Objekt

CSS-Klassen

Klasse	Tag	Funktion
alvineFormAjax	form	Aktiviert die Funktion des Plugins

Lokalisierung

Schlüssel	Standardwert
form_request_500_headline	Server Error
form_request_500_message	During request to the server, there has been an error.
form_request_500_icon	fa fa-exclamation-triangle
form_request_404_headline	Not Found
form_request_404_message	The server reported that the record is not available.
form_request_404_icon	fa fa-exclamation-triangle
form_request_403_headline	Forbidden
form_request_403_message	You do not have permission to call the url.
form_request_403_icon	fa fa-exclamation-triangle

HTML-Template

Im folgenden Beispiel wird ein Formular per Ajax übertragen. Das Ziel wird aus action genommen.



```

<!DOCTYPE html>
<html lang="en">
  <head>
    <meta charset="utf-8">
    <meta http-equiv="X-UA-Compatible" content="IE=edge">
    <meta name="viewport" content="width=device-width,
initial-scale=1">
    <title>Mein titel</title>
    <link rel="stylesheet"
href="//cdnjs.cloudflare.com/ajax/libs/bootswatch/3.3.4/cosmo/bootstrap.
min.css">
    <!--[if lt IE 9]>
      <script
src="https://oss.maxcdn.com/html5shiv/3.7.2/html5shiv.min.js"></script>
      <script
src="https://oss.maxcdn.com/respond/1.4.2/respond.min.js"></script>
    <![endif]-->
    <link rel="stylesheet"
href="//maxcdn.bootstrapcdn.com/font-awesome/4.4.0/css/font-awesome.min.
css">
  </head>
  <body>

    <div class="container">
      <div class="row">
        <div class="col-12"><h1>Formular</h1></div>
      </div>
      <div class="row">
        <div class="col-12">
          <form class="form-horizontal alvineForm"
action="/login" method="post" data-alvineform-validation="ajax">
            <fieldset>
              <!-- Form Name -->
              <legend>Form Name</legend>
              <!-- Text input-->
              <div class="form-group">
                <label class="col-md-4 control-label"
for="textinput">Text Input</label>
                <div class="col-md-4">
                  <input id="textinput"
name="textinput" type="text" placeholder="placeholder"
class="form-control input-md">
                  <span class="help-block">help</span>
                </div>
              </div>
              <!-- Select Basic -->
              <div class="form-group">
                <label class="col-md-4 control-label"
for="selectbasic">Select Basic</label>
                <div class="col-md-4">
                  <select id="selectbasic"
name="selectbasic" class="form-control">
                    <option value="1">Option
one</option>
                    <option value="2">Option
two</option>

```

```

        </select>
    </div>
</div>
<!-- Prepended checkbox -->
<div class="form-group">
    <label class="col-md-4 control-label"
for="prependedcheckbox">Prepended Checkbox</label>
    <div class="col-md-4">
        <div class="input-group">
            <span class="input-group-addon">

                <input type="checkbox">
            </span>
            <input id="prependedcheckbox"
name="prependedcheckbox" class="form-control" type="text"
placeholder="placeholder">
        </div>
        <p class="help-block">help</p>
    </div>
</div>
<!-- Button -->
<div class="form-group">
    <label class="col-md-4 control-label"
for="singlebutton">Single Button</label>
    <div class="col-md-4">
        <button type="submit"
id="singlebutton" name="singlebutton" class="btn btn-primary
has-spinner">Button <span class="spinner"><i class="fa fa-refresh
fa-spin"></i></span></button>
    </div>
</div>
</fieldset>
</form>
</div>
</div>
<script
src="//ajax.googleapis.com/ajax/libs/jquery/1.11.2/jquery.min.js"></scri
pt>
<script
src="//maxcdn.bootstrapcdn.com/bootstrap/3.3.4/js/bootstrap.min.js"></sc
ript>
<script
src="//cdn.alvine.io/libs/alvine/formajax/jquery.alvineform-0.0.3.min.js
"></script>

</body>
</html>

```

Anwendungsbeispiele

Löschbutton in einer Tabelle

Beim erfolgreichen absenden des Löschbutton-Formulars die entsprechende Zeile ausblenden.

```
<table>
  <tbody>
    <tr><td>1</td><td><form class="alvineFormAjax"
action="/delete/xxxxxxx"
methode="DELETE"><button>delete</button></td></tr>
    <tr><td>1</td><td><form class="alvineFormAjax"
action="/delete/xxxxxxx"
methode="DELETE"><button>delete</button></td></tr>
  </tbody>
</table>
```

Das passende Javascript

```
(function($, window, document, undefined) {
  $(function() {

    /** Beim Löschen die gelöschte Zeile ausblenden */
    $('.alvineFormAjax[method="DELETE"]').each(function(index,
element) {
      if(!element.hasOwnProperty('alvineFormAjaxSettings')) {
        return;
      }
      element.alvineFormAjaxSettings.onSuccess = function() {
        var jqElement = $(element);
        var jqTBody = jqElement.closest('tbody');
        /** Zeile löschen */
        jqElement.closest('tr').remove();
        /** Wenn keine Zeile mehr vorhanden, dann die Zeile neu
laden */
        if(jqTBody.find('tr').length===0) {
          jqTBody.html('<tr><td
colspan="100">'+$.fn.editableform.loading+'</td></tr>');
          location.reload();
        }
      };
    });
  });
})(jQuery, window, document);
```

Login-Formular

Bei einem Login-Formular sollte die Fehlermeldung 403 Forbidden nicht als notify eingeblendet werden. Um dies einzustellen, erhält die Form-Klasse neben der Klasse alvineFormAjax auch die Klasse alvineFormLogin.

```
<form id="loginForm" method="POST" action="/login"
novalidate="novalidate" class="alvineFormAjax alvineFormLogin">
```

```
(function($, window, document, undefined) {
    $(function() {

        /** Beim Löschen die gelöschte Zeile ausblenden */
        $('.alvineFormAjax.alvineFormLogin').each(function(index,
element) {
            if(!element.hasOwnProperty('alvineFormAjaxSettings')) {
                return;
            }
            // Notify nur bei 500 und 404
            element.alvineFormAjaxSettings.notifications = ['500',
'404'];
        });
    });
})(jQuery, window, document);
```