

Creating Forms

Creating an email form within the dotCMS platform, all the HTML for the form must be in the Body field of a Content Structure. All names are case sensitive.

In order for the form to process and send correctly the follow objects must be in the “form” tag.

```
<form name="contactForm" method="post" action="/dotCMS/sendEmail" onsubmit="return checkForm()">
```

name="(insert name)"

Identifies the unique form

method="post"

Identifies the method of submittal on the dotCMS

action="/dotCMS/sendEmail"

Is the action called that actually processes the form submittal

onsubmit="return checkForm()"

The Javascript action that verifies field requirements are met.

The action performed by the CMS will require the following hidden fields:

Any email address entered in the hidden form fields can be protect from Spam applications by adding “spamx” within the email address, i.e. ...value="testspamx@example.com". The “spamx” phrase will be removed when the form is processed. This feature, however, is not required for the form to maintain functionality.

from

The email sent by the CMS will be from this email address. Replies to the email will also go to this address.

```
<input type="hidden" name="from" id="from" value="(EMAIL)">
```

order

It determines the order in which the entered information will be displayed in the email.

```
<input type="hidden" name="order" id="order" value="(FIELD NAMES)">
```

to

The email address entered here will receive the information entered by the form submitter.

```
<input type="hidden" name="to" id="to" value="(EMAIL)">
```

subject

This will be the subject of the email received by the address in the “to” field.

```
<input type="hidden" name="subject" id="subject" value="(SUBJECT)">
```

returnUrl

This will be the page that the submitter will be sent to after the form is processed.

```
<input type="hidden" name="returnUrl" id="returnUrl" value="(URL)">
```

errorURL

This will be the page that the submitter will be sent to if the form is processed with error.

```
<input type="hidden" name="errorURL" id="errorURL" value="(URL)">
```

formType

This field will be stored in a database and will then be searchable within a portlet on the CMS. All form results will be displayed with this field as the category. It is also case sensitive; therefore “donations” will be different than “Donations” when it is displayed on the back end.

```
<input type="hidden" name="formType" id="formType" value="(FORM TYPE)">
```

The following hidden fields will also be read by the “sendEmail” action.

cc

A carbon copy of the email will be sent to this email address.

```
<input type="hidden" name="cc" id="cc" value="(EMAIL)">
```

bcc

A blind carbon copy will be sent to this email address

```
<input type="hidden" name="bcc" id="bcc" value="(EMAIL)">
```

html

The CMS will send the email in HTML with the information displayed in a table based on the value of this field. If a value of “0” is given, a plain text email will be sent. If a value of “1” is given, the HTML email will be sent.

```
<input type="hidden" name="html" id="html" value="(0 or 1)">
```

autoReply – When the form process is completed the user can choose to send a unique message populated with the following fields.

autoReplyTo

This is the email address that will receive the unique message.

```
<input type="hidden" name=" autoReplyTo" id=" autoReplyTo" value="(EMAIL)">
```

autoReplyFrom

This will be the reply to address for the unique message.

```
<input type="hidden" name=" autoReplyFrom" id=" autoReplyFrom" value="(EMAIL)">
```

autoReplySubject



This is the subject of the unique message.

```
<input type="hidden" name=" autoReplySubject" id=" autoReplySubject" value="(SUBJECT)">
```

autoReplyText

This will contain the full body of the email to be sent as part of the form submittal process.

```
<input type="hidden" name=" autoReplyText" id=" autoReplyText" value="(MESSAGE)">
```

The following input field names are recognized and stored in a database by field name by dotCMS. Any other field(s) entered as part of the form will be stored as a custom field and separated by pipes with in the column.

i.e. - <input type="text" id="fullname" name="fullname">

- | | |
|-----------------|------------|
| - prefix | - address |
| - title | - address1 |
| - firstName | - address2 |
| - middleInitial | - city |
| - middleName | - state |
| - lastName | - zip |
| - fullName | - country |
| - organization | - email |
| - phone | |

In order to validate a form before processing, the following code must be entered. This script tag must be entered in the Content Body to call the appropriate action to validate the form.

```
<script language="JavaScript" type="text/javascript" src="/global/js/form_validation.js"></script>
```

The following method must also be used in the “form” tag in order to run the validation on submit.

```
onsubmit="return checkForm()"
```

The Validation works based on specific field labels

```
<input type="text" name="LastName" value="" required="text" pretty="Last Name"
requiredLength="9" />
```

required

Format validations: **text, email, date, number**

Requires the values entered to fit the required style

required="radio"

Requires that a radio button or checkbox must be selected

```
<input type="text" name="firstName" value="" required="text" />
<input type="text" name="email" value="" required="email" />
<input type="text" name="DOB" value="" required="date" />
<input type="text" name="IDNumber" value="" required="number" />
```

pretty

When a required field has not been entered, this value will be displayed in the browser alert.

```
<input type="text" id="email" name="email" pretty="Email Address">
```



requiredLength

designates a required length in characters

i.e. Social Security Number, Zip Code, Account Number, etc.

```
<input type="text" name="IDNumber" required="number" requiredLength="9" />
```

requiredNumber

Designates the number of checkboxes that must be selected.

```
<input type="checkbox" name="fruit" value="Banana" required="radio"
requiredNumber="2">
<input type="checkbox" name="fruit" value="Apple">
<input type="checkbox" name="fruit" value="Orange">
<input type="checkbox" name="fruit" value="Mango">
```

validationFunction

Allows to make external JavaScript validations inside the checkForm. You need to include in the page the external JavaScript function then when the checkForm validation run calls your function.

Example:

```
<script language="javascript">
function myValidation () {
//Return true if validation sucecced
//return false if not
}
</script>
<input type="text" name="lastName" id="lastName" validationFunction="myValidation()"
/>
< textarea name="lastName" id="lastName" validationFunction="myValidation()"> <
/textarea>
<input type="checkbox" name="fruit" value="Banana" required="radio"
validationFunction="myValidation()">
```

Credit Card Validation

When adding Credit Card verification to a form, the following fields are required in order for the CMS to process the fields correctly.

cardType

This must be used as the field name when offering selections to the user for Credit Card choices.

The Credit Card verification process is only run when this field name is used.

```
<input type="radio" name="cardType" value="AMERICAN EXPRESS" />
```

ccNumber

This field name is used as the Credit Card number in the verification process.

```
<input type="text" size="35" name="ccNumber" />
```

cardName

This field name is used for the exact name on the Credit Card.

```
<input type="text" size="35" name="cardName" />
```



expMonth

The expiration month is processed using this field name.

```
<input type="text" size="10" name="expMonth" />
```

expYear

The expiration year is processed using this field name.

```
<input type="text" size="10" name="expYear" />
```

total

The total is processed using this field name. This field is required if Credit Card is used.

```
<input type="text" size="10" name="total" />
```

Creating User Accounts

In order to create a user account for each user that submits the form, the following field needs to be added:

```
<input type="hidden" name="createAccount" id="createAccount"
value="true" />
```

This will create a new user account with the user's email address as username and random password. Also a cookie will be set on the user's computer to begin tracking their visits throughout the site.

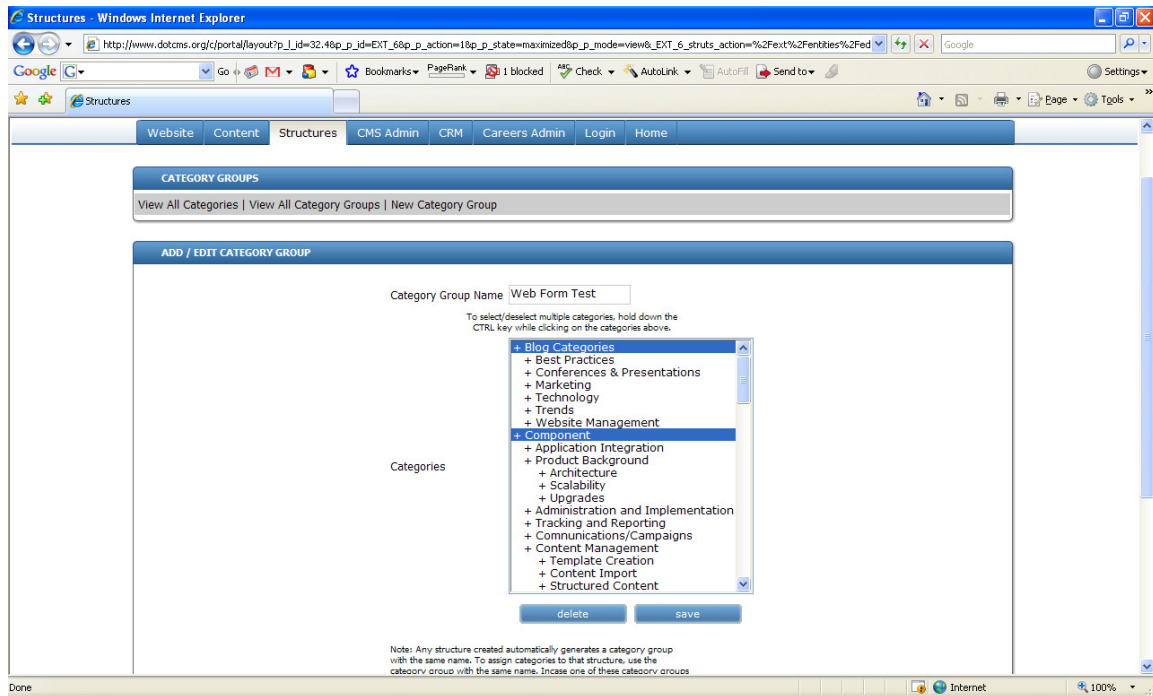
An account note is created everytime a web form is submitted

If the user already exists in dotCMS, the user only gets added a new account note.

Using Categories in Web Forms

In order to use categories in Web Forms, the following needs to be done:

- Create Category Group with the same name as the Web Form Type



- Add categories fields to the form using categories api like this

```
$categories.getCategoriesByEntityName('categories', 'Enter Web Form Type Here', true, true, 5)
```

This will display all the categories associated with the Category Group on the HTML form.

Viewing Web Form results

In order to view the results of the web form submissions, go into dotCMS and click in CMS Admin → Web Form Reports

By clicking on Remove Data all data from the form submissions will be removed. Make sure you save a backup of the report before clicking this button.

dotCMS Admin - Windows Internet Explorer

http://dotcms.dotmarketing.net/c/porta/layout?p_id=32.2&p_id=EXT_WEBFORMS&p_action=1&p_state=maximized&EXT_WEBFORMS_struts_action=%2Fext%2Fwel

Google

Structures CMS Admin

dotCMS by dotmarketing

Welcome, Maria Ahues
Home - My Account - Personalize Pages - Sign Out
Community Page: CMS Admin

Home Web Content Structures CMS Admin CRM Careers Admin Login Home

WEB FORM REPORTS

Contact Us Form

Generate Report

Remove Data

About | Disclaimer 2007 © Dotmarketing Inc. All rights reserved. powered by: dotcms.org

Select the Form from the drop down and click on Generate Report.

On the report all the fields from the form are displayed including all the categories. Each main category is created as one column with its selected values for each form displayed underneath.

MartinFormReport-9-13-2007[1].xls - Microsoft Excel

Submit Date/Time	First Name	Last Name	Email	Createaccount	Blog Categories	Toemail	Organization Type	Submit
9/12/2007 4:57PM	First Name	Last Name	email2@dm.com	TRUE	Conferences & Presentations, Marketing, Technology	martin@dotmarketing.com	Nonprofit, Education	Submit
9/12/2007 4:57PM	First Name	Last Name	email2@dm.com	TRUE	Conferences & Presentations, Marketing, Technology	martin@dotmarketing.com	Nonprofit, Education	Submit
9/12/2007 4:07PM	First Name	Last Name	email2@dm.com	TRUE	Conferences & Presentations, Marketing, Technology	martin@dotmarketing.com	Nonprofit, Education	Submit
9/12/2007 4:05PM	First Name	Last Name	email1@dm.com	TRUE	Conferences & Presentations, Marketing, Technology	martin@dotmarketing.com	Nonprofit	Submit
9/6/2007 5:23PM	First Name	Last Name	joaquin@nomail.nomail	TRUE	Conferences & Presentations, Marketing, Technology	martin@dotmarketing.com	Association, Nonprofit, Corporation	Submit
9/6/2007 4:50PM	First Name	Last Name	martin12@cantv.net	TRUE	Marketing, Technology	martin@dotmarketing.com	Corporation	Submit

MartinFormReport-9-13-2007 1

Select destination and press ENTER or choose Paste

Count: 7



Sample Code

Here is an example of a basic web form with user information and categories.

```
<form name="contactForm" method="post" action="/dotCMS/submitWebForm">

<input type="hidden" name="fromEmail" id="from"
value="from@mysite.com">
<input type="hidden" name="from" id="from" value="From Name Here" />
<input type="hidden" name="to" id="to" value="To Name Here" />
<input type="hidden" name="toEmail" id="to" value="to@mysite.com" />
<input type="hidden" name="subject" id="subject" value="Subject Here"
/>
<input type="hidden" name="returnUrl" id="returnUrl"
value="thank_you.dot" />
<input type="hidden" name="html" id="html" value="1" />
<input type="hidden" name="order" id="order"
value="firstName,lastName,title,organization,email,phone,message" />
<input type="hidden" name="formType" id="formType" value="Web Form Type
Here" />
<input type="hidden" name="createAccount" id="createAccount"
value="true" />

<div>
    <div class="row"><label for="firstName">*First Name:</label>
<input type="text" id="firstName" name="firstName" required="text"
pretty="First Name" style="width: 200px;" /></div>

    <div class="row"><label for="lastName">*Last Name:</label> <input
type="text" id="lastName" name="lastName" required="text" pretty="Last
Name" style="width: 200px;" /></div>

    <div class="row"><label for="email">*Email:</label> <input
type="text" id="email" name="email" required="email" pretty="Email
Address" style="width: 200px;" /></div>

    <div class="row"><label for="phone">Phone:</label> <input
type="text" id="phone" name="phone" style="width: 200px;" /></div>

    <div class="row"><label for="message">Comments:</label> <textarea
id="message" name="message" style="width: 200px; height:
100px;"></textarea></div>

</div>
<div>
    <div class="row">
        $categories.getCategoriesByEntityName('categories', 'Web Form
Type Here', true, true, 5)
    </div>
</div>

<div class="row"><label for="submit">&nbsp;</label>
```



```
<input style="height:20px;" type="submit" name="submit" value="Submit"
/></div>
</form>
```