

Setting up Formstack Forms for Creatio

Introduction

Creatio is a No-Code Platform offering significant integration capability with other systems. Formstack's suite of products is a robust toolset to manage the initialization of data collection, generation of forms from that data (and external data), and solicitation for digital signature. Together, they offer a powerful synergy that complements each other significantly.

Formstack offers three primary products that can be integrated with Creatio using our series of Marketplace connectors:

- Forms
- Documents
- Sign

In this document, you'll find step-by-step instructions that will be needed to configure your Creatio instance to receive submissions from Formstack Forms. We will use a templated approach in this guide; however, connecting any Formstack Form with Creatio will follow a similar process.

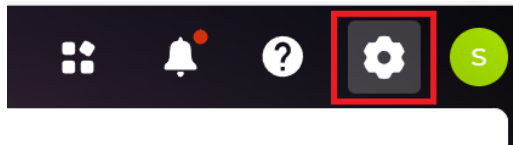
Setup Requirements

Before getting started, there are a couple of minor pre-requisites required to properly use the Formstack Forms Connector for Creatio:

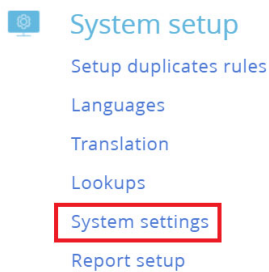
1) **Obtain a Software Selection Advisors Account Number:**

Please contact us at formstack@ss-adv.com to obtain your Software Selection Advisors account number. We will create a number and email it back to you. Placing this in your Creatio instance is not a hard requirement; however, it will allow us to know who uses our product and how we can provide expedited assistance.

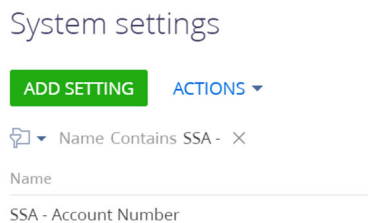
- a. Once obtained, navigate to the System Designer:



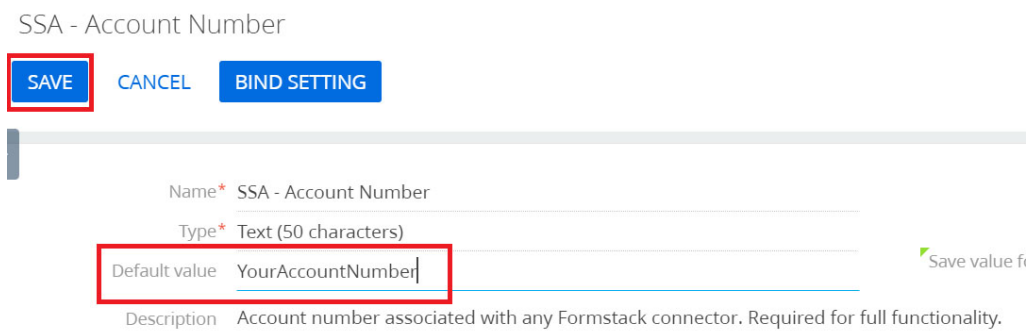
- b. Left-click on "System Settings" from within the "System setup" section:



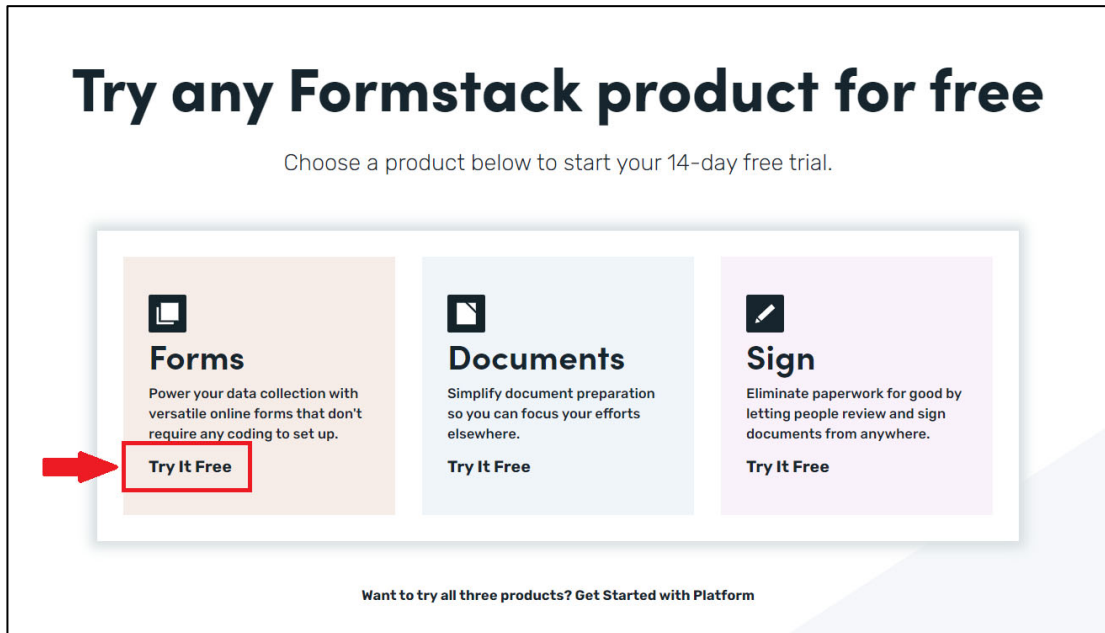
- c. Apply a filter to search by Setting Name where the Name starts with "SSA - ":



- d. Place your account number in the "Default Value" field and click "Save"



- 2) You must have an active [Formstack Forms subscription](#). You may use [this link](#) to obtain a free 14-day trial of Formstack Forms.



Additionally, while Formstack is highly configurable and can connect any form's data and push it into Creatio, we suggest using a template we've created as a "starting place." To obtain these templates, please email formstack@ss-adv.com with the email address used to register for your Formstack 14-day free trial.

If you do not have an active trial or subscription with Formstack, we cannot share the standard templates with your Formstack Forms account.

Before Getting Started

We've attempted to make connecting Creatio and Formstack Forms as easy as possible. Please note that while Creatio is a No-Code Platform, not all integrations can be accomplished without code. Thankfully, our solution provides the code needed and the steps to modify that code in simple copy/paste exercises.

For more complex integrations of Formstack Forms with Creatio (i.e., connecting custom forms with Creatio), please contact us at formstack@ss-adv.com. We will gladly help assess the effort required to accomplish the desired outcome.

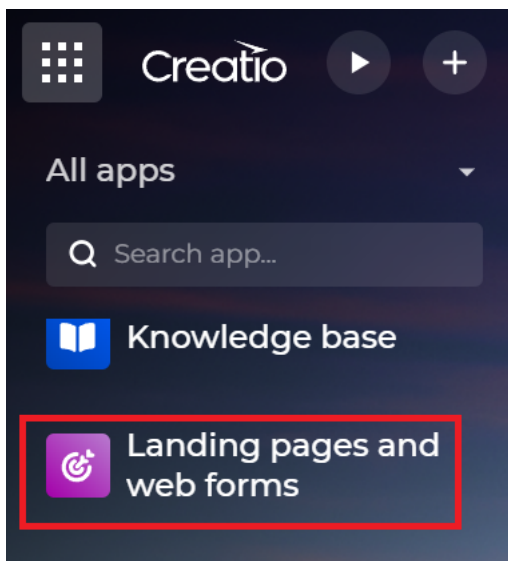
We try to offer this service as a compliment for customers where Software Selection Advisors, LLC is the assigned Formstack partner; however, should any engagement (for our customers and others) require paid participation on our part, we will clarify that upfront.

Setup Instructions

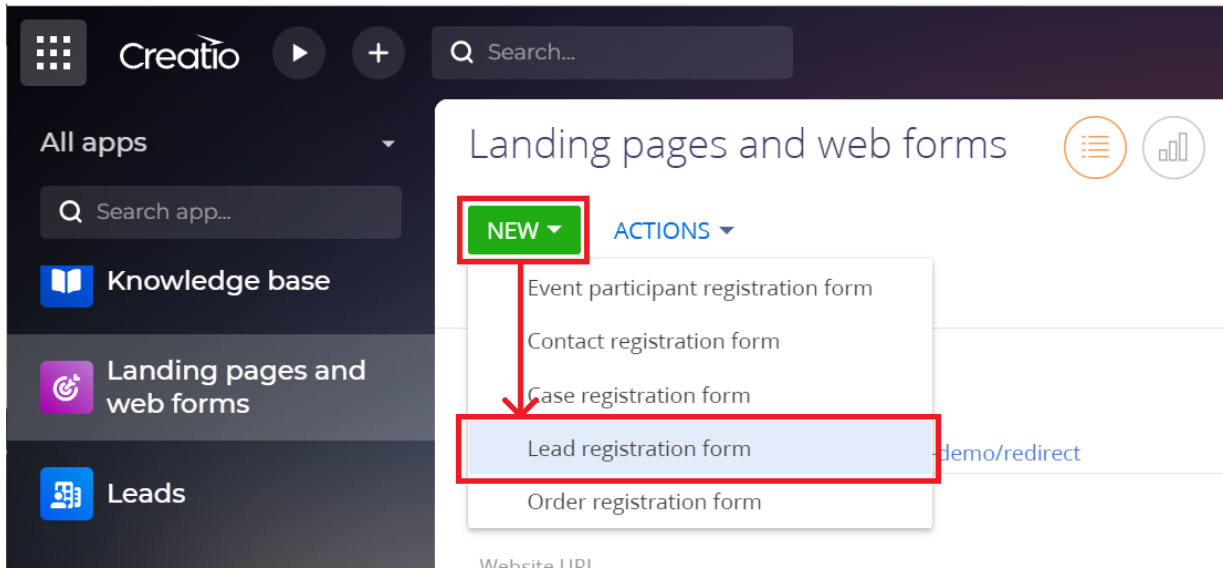
Follow the below instructions to configure our "Creatio Lead" form from Formstack. **The below process should take less than 30 minutes to complete if the "Setup Requirements" have been completed.**

The person performing the tasks within Creatio must have the proper System Administrator role and access to the sections covered in the steps that follow.

1. Configure Creatio for Formstack Forms
 - a. Install our "[Formstack Forms Connector for Creatio](#)" Connector from the Marketplace
 - b. Navigate to the "Landing pages and web forms" section.



- c. From the section's list page, click the "New" button, and select "Lead registration form" from the dropdown options.



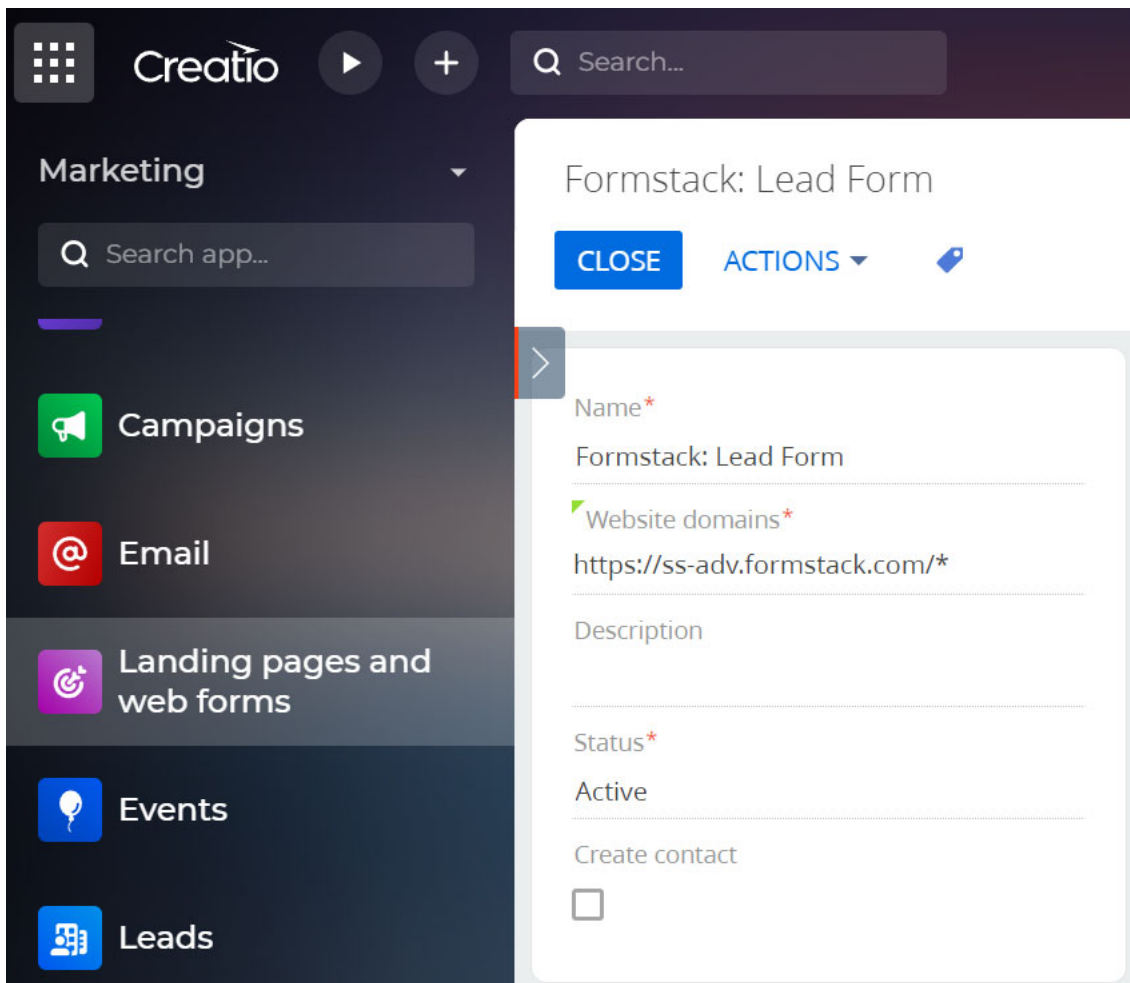
- d. On the new record entry page for Landing Pages, fill out the following parameters using the guidance provided on each field:
- Name:** this can be whatever you prefer. In this example, we've used "Formstack: Lead Form" to illustrate the connection between Formstack and Creatio further.
 - Website domains:** this can vary for each customer. This URL is intended to create a means of access from wherever the Formstack Form will originate.

- Formstack URL: if you do not plan on embedding your Formstack Form onto a website or wish to proceed using this guide, use the domain value that gets assigned to your company for your trial. For our company, our Formstack subdomain is "ss-adv" and we place it into the field in the below screenshot:

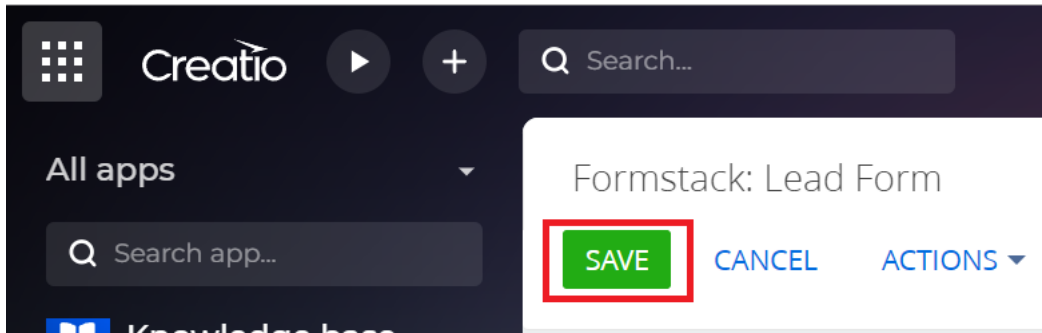
*https://ss-adv.formstack.com/**

For more help identifying this value, refer to the Formstack Forms help article: ["Subdomains, Aliases, and User-Friendly URLs"](#)

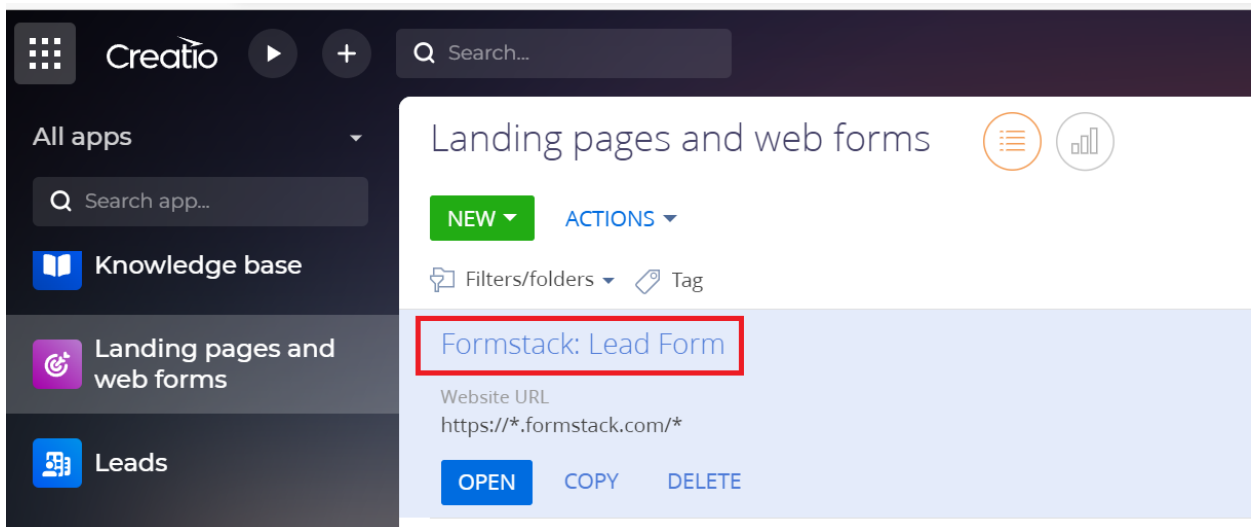
- Custom URL: if you are planning to embed the Formstack Form on a custom web page, then the value should represent that location here. i.e., *https://www.yourdomain.com/**



- e. Click the "Save" button to commit the new landing page to the Creatio database.



- f. You should now see your landing page in the list of landing pages:



- g. Navigate back into the record by clicking on the record's title.

- h. Direct your attention to the “Step 1. Set up a redirection URL (optional)” and “Step 2. Copy the code and configure and map the fields” steps.

▲ STEP 1. Set up a redirection URL (optional) ⓘ

Redirection URL

▲ STEP 2. Copy the code and configure and map the fields ⓘ

```
<script src="https://ajax.googleapis.com/ajax/libs/jquery/1.11.2/jquery.min.js"></script>
<script src="https://webtracking-v01.bpmonline.com/JS/track-cookies.js"></script>
<script src="https://webtracking-v01.bpmonline.com/JS/create-object.js"></script>
<script>

/**
 * Replace the "css-selector" placeholders in the code below with the element selectors on your landing page.
 * You can use #id or any other CSS selector that will define the input field explicitly.
 * Example: "Email": "#MyEmailField".
 * If you don't have a field from the list below placed on your landing, leave the placeholder or remove the line.
 */
var config = {
  fields: {
```

- i. **“Step 1. Set up a redirection URL (optional)”**: This allows you to define where users should get directed upon filling out the form. If you have a specific URL to thank individuals for submitting their information, you may place it here (i.e., “https://www.yourdomain.com/thank-you/”). If you modify this field, repeat steps 1e, 1f, and 1g from above.

For the purposes of this guide, you can disregard this setting.

- ii. **“Step 2. Copy the code and configure and map the fields”**: do not let this step intimidate you. What would normally be an exercise in low-code configuration, we’re turning into a Copy (“Ctrl+C”) and Paste (“Ctrl+V”) exercise.

Within the box of code, use your center mouse wheel or the navigation sliders that appear when you hover over the section to navigate to the highlighted code below:

STEP 2. Copy the code and configure and map the fields ⓘ

```

},
contactFields: {
  "FullName": "css-selector", // Name of a contact
  "Phone": "css-selector", // Contact's mobile phone
  "Email": "css-selector" // Contact's email
},
customFields: {},
landingId: "309a5b04-244d-437f-acd5-6ff2dd12d779",
serviceUrl: "https://1202257-demo.creatio.com/0/ServiceModel/GeneratedObjectWebFormService.svc/SaveWebFormObjectData",
redirectUrl: ""
};
/**
 * The function below creates a object from the submitted data.
 * This function will be the "Submit" event of the form.

```

Note: if you placed a value in “Step 1.” for a redirect URL, you should see it in the screenshot sections where “redirectUrl” is located. If you don’t, repeat steps 1e, 1f, and 1g above to save changes and regenerate the code that is provided in the “Step 2” area.

STEP 1. Set up a redirection URL (optional) ⓘ

Redirection URL

STEP 2. Copy the code and configure and map the fields ⓘ

```

    "Phone": "css-selector", // Contact's mobile phone
    "Email": "css-selector" // Contact's email
  },
  customFields: {},
  landingId: "309a5b04-244d-437f-acd5-6ff2dd12d779",
  serviceUrl: "https://1202257-demo.creatio.com/0/ServiceM
  redirectUrl: "https://www.yourdomain.com/thank-you"
};

```

- iii. Copy these three lines of code from the box and place them into your preferred code editor or Windows Notepad. We prefer to use [Notepad++](#).

```

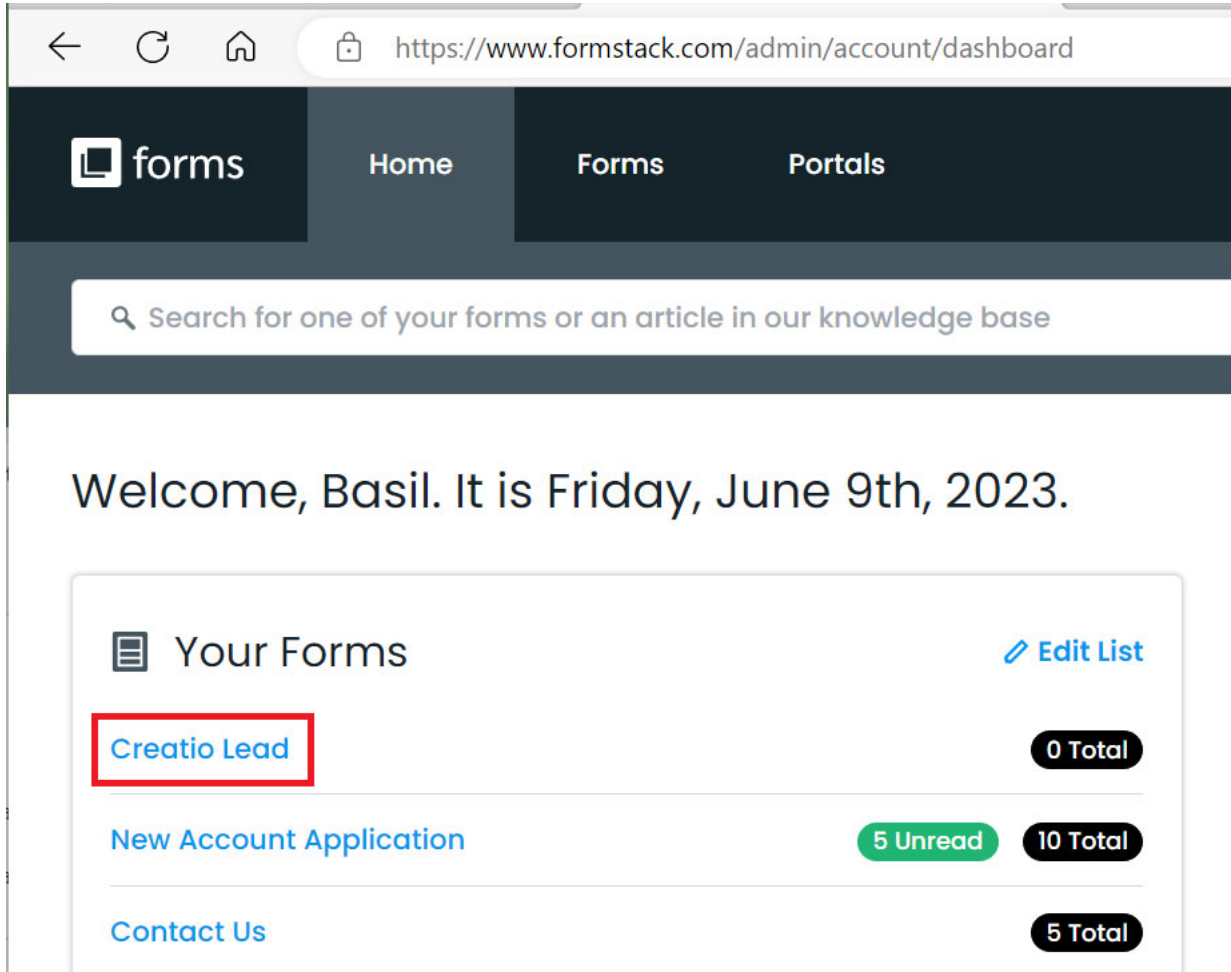
1 landingId: "309a5b04-244d-437f-acd5-6ff2dd12d779",
2 serviceUrl:
3 "https://1202257-demo.creatio.com/0/ServiceModel/GeneratedObjectWebFormService.svc/SaveWebFormObjectData",
4 redirectUrl: ""
5

```

- iv. With that, you are done configuring Creatio.

2. Configure Formstack Forms

- a. Log into your Formstack Forms account.
- b. Locate the "Creatio Lead" form from within the "Your Forms" section of the Home page.



- c. Left-click the "Creatio Lead" title to navigate to the form.

d. The form will automatically open in the “Build” mode (shown below):

The screenshot displays the 'Build' mode of a form creation tool. The top navigation bar includes 'forms', 'Home', 'Forms', and 'Portals'. The main toolbar shows 'Build' (highlighted), 'Style', 'Changes saved', and 'View live form'. The left sidebar is divided into 'Basic Fields' (Short Answer, Long Answer, Name, Address, Email Address, Phone Number, 123 Number, Dropdown List, Radio Button, Checkbox) and 'Advanced Fields' (Credit Card, Date/Time, File Upload, Matrix, Description Area, Embed Code, Event/Product, Signature, Rating). Below this is 'Saved Fields' (Primary Interest) and 'Layout and Sections' (Section). The main canvas shows a 'Creatio Lead Form' with the following layout:

- Section header: Creatio Lead Form
- Row 1: Your Name (First Name, Last Name), Your Job Title, Mobile Phone Number, Email Address
- Row 2: Your Company, Website
- Row 3: Address (Address Line 1, City, State, ZIP Code)
- Row 4: How can we help you? (Additional service dropdown)
- Row 5: Embed Code (</>)
- Row 6: hidden_NameValue, hidden_StateValue, hidden_HowCanWeHelpYouValue

e. Click the “View live form” button in the top-right corner, ensuring that you do not click on the dropdown arrow.

This close-up shows the top-right corner of the interface. It includes the user profile 'Basil Malik' and a navigation bar with 'Settings', 'Share', 'Submissions', and 'Analytics'. The 'View live form' button is highlighted with a red box, and a red arrow points to the dropdown arrow on its right side, indicating that the user should click the button itself, not the arrow.

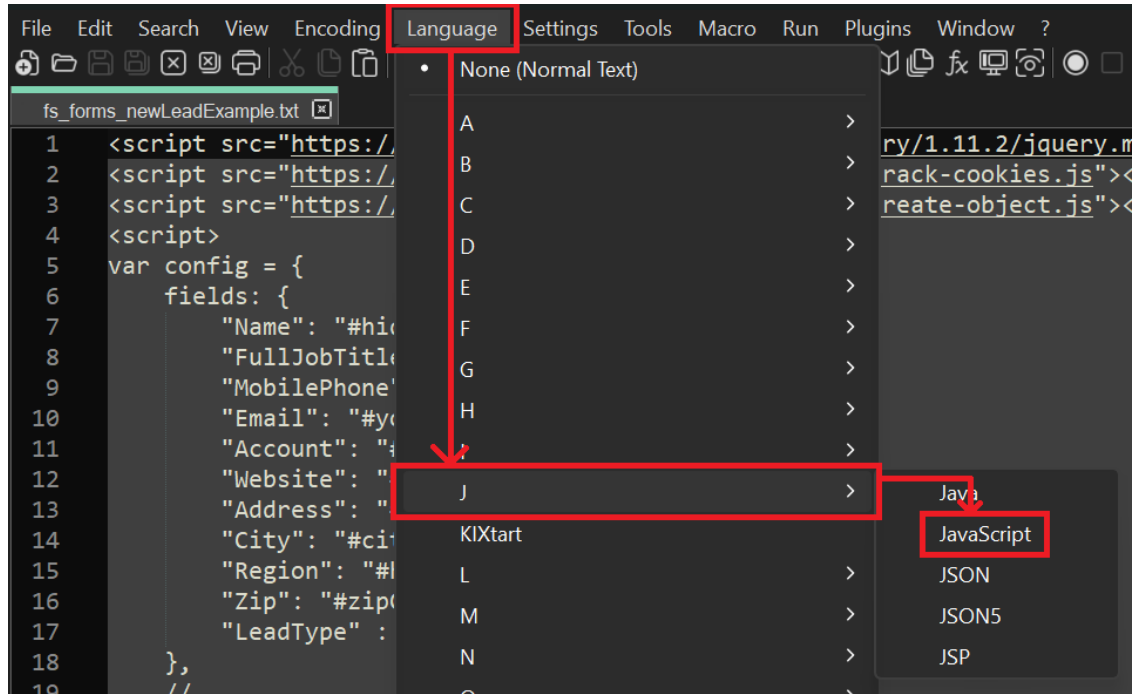
- f. You should now be viewing the form in its raw format if it weren't embedded on your website. Note that it contains the following fields, noting the fields in **bold blue** as the only required fields:
- i. **Your Name**
 - **First Name**
 - **Last Name**
 - ii. Your Job Title
 - iii. Mobile Phone Number
 - iv. Email Address
 - v. Your Company
 - vi. Website
 - vii. Address
 - Address Line 1
 - City
 - State
 - ZIP Code
 - viii. **How can we help you?**
 - ix. hidden_NameValue
 - x. hidden_StateValue
 - xi. hidden_HowCanWeHelpYouValue

Each field will map to a field within Creatio by the end of the next few steps. Leaving the Live View of the form open, proceed to the next step.

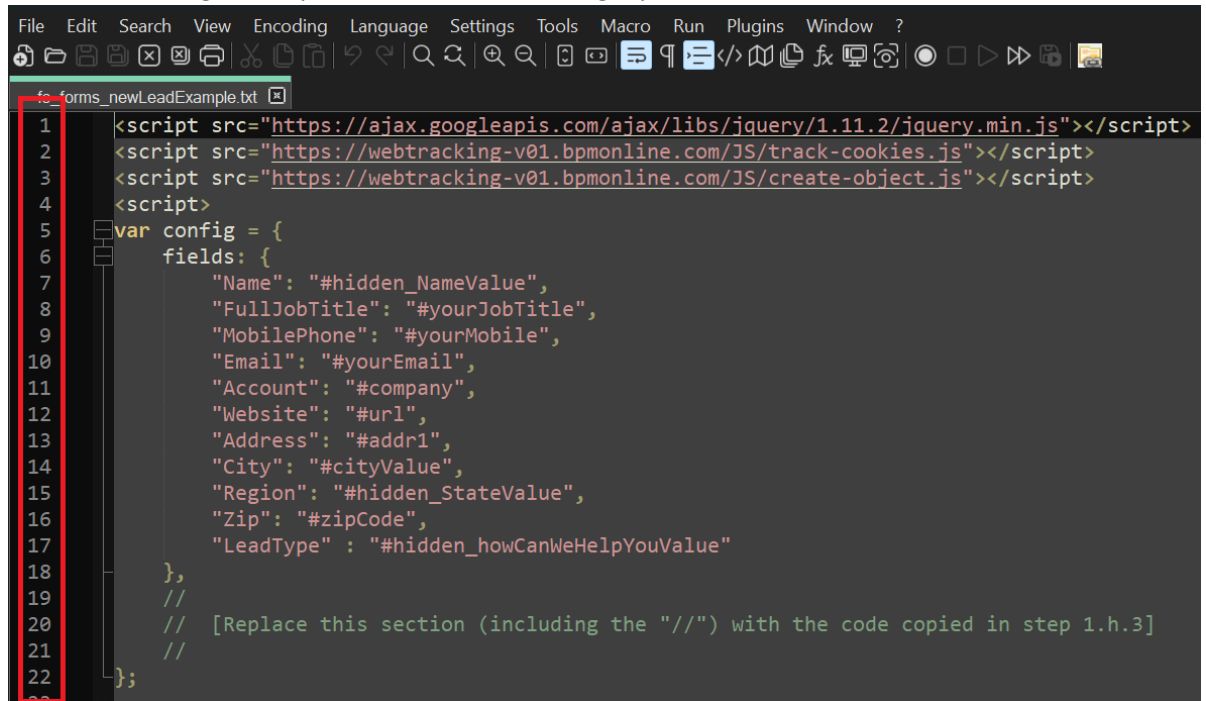
- g. You'll need to open two text files in your preferred text editor. We prefer Notepad++ here as what we'll do in the next steps will create a color-coding effect on the text; making it easier to use.
- i. From our "[Formstack Forms Connector for Creatio](#)" Connector from the Marketplace, there will be a "fs_forms_newLeadExample.txt" file you'll need.
 - ii. From step 1.h.iii., you'll need the text you copied from your Creatio landing page:

```
1  
2 landingId: "309a5b04-244d-437f-acd5-6ff2dd12d779",  
3 serviceUrl:  
4 "https://1202257-demo.creatio.com/0/ServiceModel/GeneratedObjectWebFormService.svc/SaveWebFormObjectData",  
5 redirectUrl: ""
```

- h. Open the “fs_forms_newLeadExample.txt” file. We are using Notepad++ in the following steps:
 - i. Change the “Language” value to be “Javascript”



The text should go from plain black/white to being stylized with some added color:

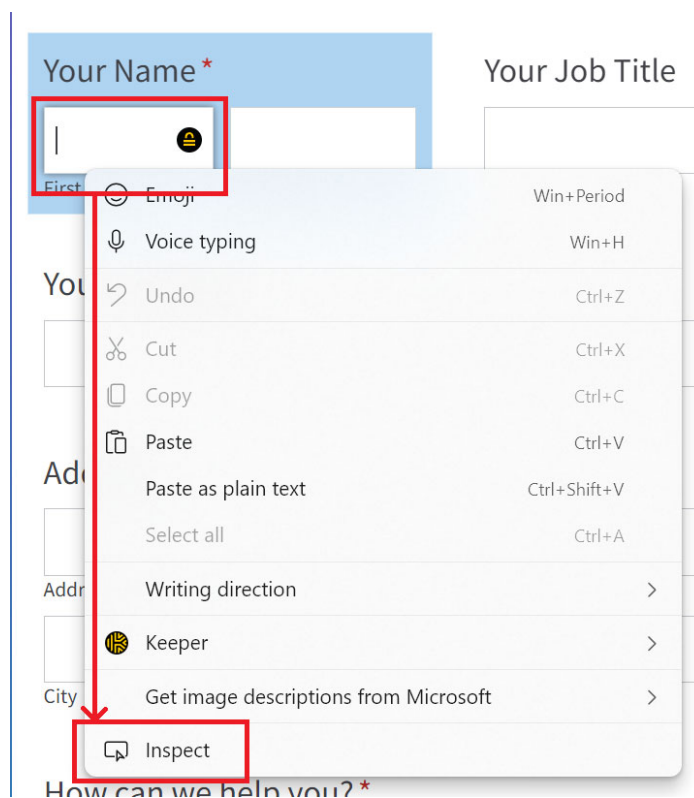


Note: the line numbers referenced in the next steps are located along the left-hand side of the text editor.

- ii. Replace lines 19, 20, and 21 entirely with the code that was copied in step 1.h.3.

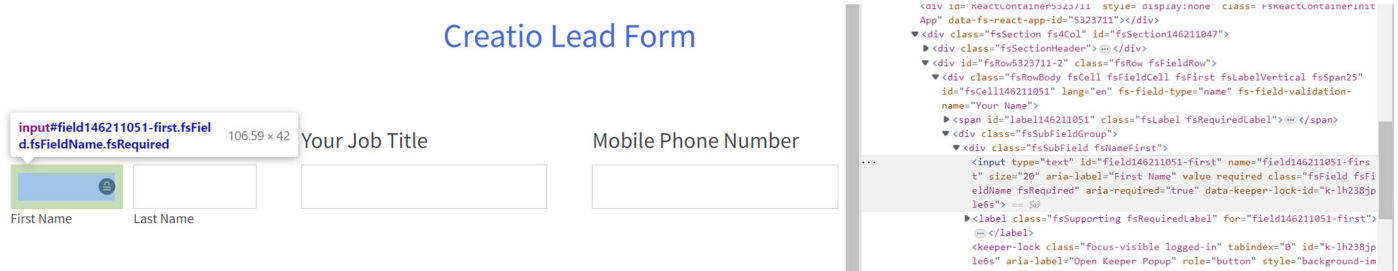
```
18 },
19 landingId: "309a5b04-244d-437f-acd5-6ff2dd12d779",
20 serviceUrl: "https://1202257-demo.creatio.com/0/ServiceModel/GeneratedObjectWebFormService.svc/SaveWebFormObjectData",
21 redirectUrl: ""
22 }
```

- iii. Now, we need to perform a copy/paste exercise using the Live Form view of your Formstack Form and the code. We will be replacing each of the “#fieldname” values throughout the text document. These values can be found throughout the document at:
- Lines 7 – 17: i.e., “#hidden_NameValue” and “#zipCode”
 - Lines 27 – 29
 - Lines 33, 34, 37, and 38
- iv. Left-click on the web browser window where your Formstack Form is still showing in its “Live View” mode.
- v. Right-click on the “First Name” field, and click on the “Inspect” option.



This feature of any modern web browser will allow you to see the more technical aspects of any web page you view.

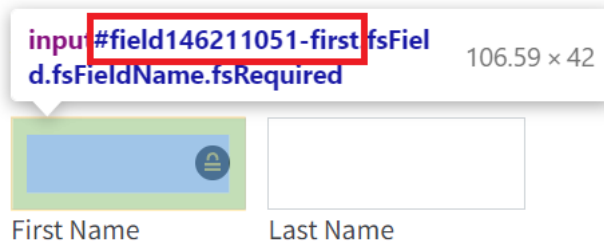
By right-clicking the “First Name” field and selecting the “Inspect” option, your screen should look something similar to the below:



What is needed for the next several steps is the unique “id” field. These values are unique to each Formstack Form created, and, unfortunately, can’t be re-used from one landing page to another within Creatio.

The unique “id” field can be found in two places in the above screenshot:

- 1) In the caption that shows above the field. You want the “#field146211051-first” value and nothing else from this example.



viii. We must now repeat the same exercise in the previous steps for each field on the form so that our code is fully tailored to include the unique fields from the used form.

- **Last Name:** from the below, we'll place "field146211051-last" into the "#lastName" value from line 28 of our code.

```

27 let fName = document.getElementById("field146211051-first");
28 let lName = document.getElementById("field146211051-last");
29 let fullName = document.getElementById("#hidden_NameValue");

```

- **Your Job Title:** from the below, we'll place "#field146211208" into the "#yourJobTitle" value from line 8 of our code. Notice that we are keeping the "#" in the value. We are modifying a line between line 7 and 17 – therefore, this part of the code requires that we keep the "#" in this section of code.

```

7 "Name": "#hidden_NameValue",
8 "FullJobTitle": "#field146211208",
9 "MobilePhone": "#yourMobile",

```

- **Mobile Phone:** from the below, we'll place "#field146211282" into the "#yourMobile" value from line 9 of our code. Again, note that we are replacing a value within lines 7 and 17 – therefore, we keep the "#" symbol.

The screenshot shows a form with a 'Your Job Title' field and a 'Website' field. A tooltip for the 'Mobile Phone' field is visible, showing the ID 'field146211282'. Below the form, a code editor snippet shows the following JSON structure:

```

8      "FullJobTitle": "#field146211208",
9      "MobilePhone": "#field146211282",
10     "Email": "#yourEmail",

```

The value "#field146211282" in line 9 is highlighted with a red box.

- **Email Address:** from the below, we'll place "#field146211284" into the "#yourEmail" value from line 10 of our code. Again, note that we are replacing a value within lines 7 and 17 – therefore, we keep the "#" symbol.

The screenshot shows a form with a 'Mobile Phone Number' field. A tooltip for the 'Email' field is visible, showing the ID 'field146211284'. Below the form, a code editor snippet shows the following JSON structure:

```

9      "MobilePhone": "#field146211282",
10     "Email": "#field146211284",
11     "Account": "#company",

```

The value "#field146211284" in line 10 is highlighted with a red box.

- **Your Company:** from the below, we'll place "#field146211232" into the "#company" value from line 11 of our code. Again, note that we are replacing a value within lines 7 and 17 – therefore, we keep the "#" symbol.

The screenshot shows a form with a 'Your Company' field. A tooltip for the 'Company' field is visible, showing the ID 'field146211232'. Below the form, a code editor snippet shows the following JSON structure:

```

10     "Email": "#field146211284",
11     "Account": "#field146211232",
12     "Website": "#url",

```

The value "#field146211232" in line 11 is highlighted with a red box.

- **Website:** from the below, we'll place "#field146211245" into the "#url" value from line 12 of our code. Again, note that we are replacing a value within lines 7 and 17 – therefore, we keep the "#" symbol.

```

<input type="text" id="field146211245" />

```

```

11 "Account": "#field146211232",
12 "Website": "#field146211245",
13 "Address": "#addr1",
14 "City": "#cityValue"

```

- **Address Line 1:** from the below, we'll place "#field146211360-address" into the "#addr1" value from line 13 of our code. Again, note that we are replacing a value within lines 7 and 17 – therefore, we keep the "#" symbol.

```

<input type="text" id="field146211360-address" />

```

```

11 "Account": "#field146211232",
12 "Website": "#field146211245",
13 "Address": "#field146211360-address",
14 "City": "#cityValue",
15 "Region": "#hidden_StateValue"

```

- **City:** from the below, we'll place "#field146211360-city" into the "#cityValue" value from line 14 of our code. Again, note that we are replacing a value within lines 7 and 17 – therefore, we keep the "#" symbol.

```

<input type="text" id="field146211360-city" />

```

```

13 "Address": "#field146211360-address",
14 "City": "#field146211360-city",
15 "Region": "#hidden_StateValue",
16 "Zip": "#zipCode"

```

- **State:** from the below, we'll place "field146211360-state" into the "#state" value from line 33 of our code – leaving out the "#" from the value.

The screenshot shows a dropdown menu labeled 'State' with a value of '222'. The HTML code for the dropdown is as follows:

```
<div class="fsSubField fsFieldState">
  <select id="field146211360-state" aria-label="State">
    <option value=" " >&nbsp;</option>
    <option value="AL">Alabama</option>
```

```
32
33 | let stateVar = document.getElementById("field146211360-state");
34 | let stateStr = document.getElementById("#hidden_StateValue");
35 | stateVar.onchange = function(){stateStr.value = stateVar.selected
```

- **ZIP Code:** from the below, we'll place "#y" into the "#zipCode" value from line 16 of our code. Again, note that we are replacing a value within lines 7 and 17 – therefore, we keep the "#" symbol.

The screenshot shows a text input field labeled 'ZIP Code' with a value of '77.3'. The HTML code for the input is as follows:

```
<div class="fsSubField fsFieldZip">
  <input type="text" id="field146211360-zip" aria-label="ZIP Code" value="77.3" />
```

```
15 | "Region": "#hidden_StateValue",
16 | "Zip": "#field146211360-zip",
17 | "LeadType": "#hidden_howCanWeHelpYou"
```

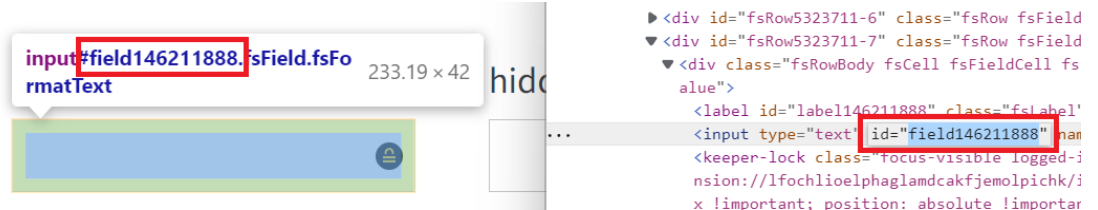
- **How can we help you?:** from the below, we'll place "field146211369" into the "#howCanWeHelpYou" value from line 37 of our code – leaving out the "#" from the value.

The screenshot shows a dropdown menu labeled 'Additional service' with a value of '508.8 x 42'. The HTML code for the dropdown is as follows:

```
<div id="field146211369" class="fsField">
  <select id="field146211369" name="fsField146211369">
    <option value="Additional service">Additional service</option>
    <option value="Bulk email management">Bulk email management</option>
    <option value="Hardware">Hardware</option>
```

```
36
37 | let customerNeedVar = document.getElementById("field146211369");
38 | let customerNeedStr = document.getElementById("#hidden_howCanWeHelpYou");
39 | customerNeedVar.onchange = function(){customerNeedStr.value = cus
```

- **hidden_NameValue:** from the below, we'll place "#field146211888" into the "#hidden_NameValue" value from line 7 of our code. Again, note that we are replacing a value within lines 7 and 17 – therefore, we keep the "#" symbol.



```

6   fields: {
7     "Name": "#field146211888",
8     "FullJobTitle": "#field146211888",
9     "MobilePhone": "#field146211888"

```

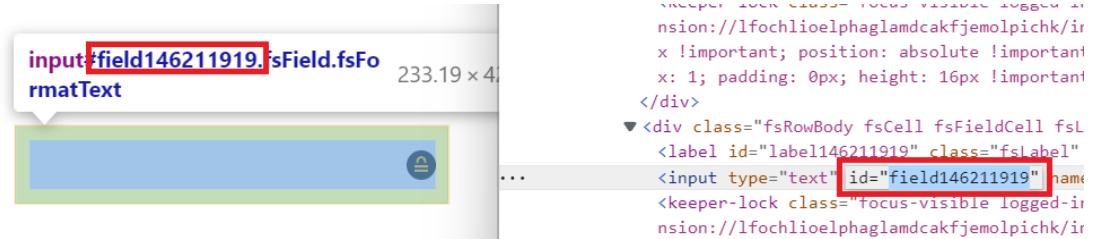
We will also alter line 29 of our code, placing "field146211888" into the second "#hidden_NameValue" location. Since this line is outside of lines 7 through 17, we drop the "#" from this spot.

```

28  let lName = document.getElementById("field146211051-last");
29  let fullName = document.getElementById("field146211888");
30  fName.onChange = function(){fullName.value = fName.value +

```

- **hidden_StateValue:** from the below, we'll place "#field146211919" into the "#hidden_StateValue" value from line 15 of our code. Again, note that we are replacing a value within lines 7 and 17 – therefore, we keep the "#" symbol.



```

14 "City": "#field146211360-city",
15 "Region": "#field146211919",
16 "Zip": "#field146211360-zip",

```

We will also alter line 34 of our code, placing "field146211919" into the second "#hidden_StateValue" location. Since this line is outside of lines 7 through 17, we drop the "#" from this spot.

```

33 let stateVar = document.getElementById("field146211360-state")
34 let stateStr = document.getElementById("field146211919");
35 stateVar.onchange = function(){stateStr.value = stateVar.sele

```


- **hidden_howCanWeHelpYouValue:** from the below, we'll place "#field146211924" into the "#hidden_howCanWeHelpYouValue" value from line 17 of our code. Again, note that we are replacing a value within lines 7 and 17 – therefore, we keep the "#" symbol.



```

16      "Zip": "#field146211360-zip",
17      "LeadType": "#field146211924"
18    },

```

We will also alter line 38 of our code, placing "field146211924" into the second "#hidden_StateValue" location. Since this line is outside of lines 7 through 17, we drop the "#" from this spot.

```

36
37      let customerNeedVar = document.getElementById("field146211369");
38      let customerNeedStr = document.getElementById("field146211924");
39      customerNeedVar.onchange = function(){customerNeedStr.value = custo
40

```


- ix. Presuming all changes have been completed successfully, your final code should look something similar to the below:

```

1 <script src="https://ajax.googleapis.com/ajax/libs/jquery/1.11.2/jquery.min.js"></script>
2 <script src="https://webtracking-v01.bpmonline.com/JS/track-cookies.js"></script>
3 <script src="https://webtracking-v01.bpmonline.com/JS/create-object.js"></script>
4 <script>
5     var config = {
6         fields: {
7             "Name": "#field146211888",
8             "FullJobTitle": "#field146211208",
9             "MobilePhone": "#field146211282",
10            "Email": "#field146211284",
11            "Account": "#field146211232",
12            "Website": "#field146211245",
13            "Address": "#field146211360-address",
14            "City": "#field146211360-city",
15            "Region": "#field146211919",
16            "Zip": "#field146211360-zip",
17            "LeadType": "#field146211924"
18        },
19        landingId: "309a5b04-244d-437f-acd5-6ff2dd12d779",
20        serviceUrl: "https://1202257-demo.creatio.com/0/ServiceModel/GeneratedObjectWebFormService.svc/SaveWebFormObjectData",
21        redirectUrl: ""
22    };
23
24    // Fill in the below fields with "#" with the corresponding Formstack Form unique input IDs.
25    // If you do not replace these values with your form's unique field names, this will not work.
26
27    let fName = document.getElementById("field146211051-first");
28    let lName = document.getElementById("field146211051-last");
29    let fullName = document.getElementById("field146211888");
30    fName.onchange = function(){fullName.value = fName.value + ' ' + lName.value};
31    lName.onchange = function(){fullName.value = fName.value + ' ' + lName.value};
32
33    let stateVar = document.getElementById("field146211360-state");
34    let stateStr = document.getElementById("field146211919");
35    stateVar.onchange = function(){stateStr.value = stateVar.selectedOptions[0].innerText};
36
37    let customerNeedVar = document.getElementById("field146211369");
38    let customerNeedStr = document.getElementById("field146211924");
39    customerNeedVar.onchange = function(){customerNeedStr.value = customerNeedVar.selectedOptions[0].innerText};
40
41    function createObject() {
42        landing.createObjectFromLanding(config);
43    }
44
45    function submitToCreatio(){
46        createObject();
47        return false;
48    }
49
50    function initLanding() {
51        landing.initLanding(config);
52    }
53    jQuery(document).ready(initLanding);
54
55    let form = document.querySelector('form.fsForm');
56    form.addEventListener('submit', submitToCreatio);
57
58 </script>

```

Note: if you are using a text editor like Notepad++, the document does a wonderful job tracking changes in the code. In the above example, all lines that are modified from our "" document have an orange indicator to the left of the line that was modified. You should see something similar if you're using Notepad++. These orange indicators may be green if you saved your progress while applying the changes.

- i. With the required code to connect the Formstack Form to Creatio, we need to modify the “Embed Code” block from our form.

The screenshot shows a form builder interface. At the top, there is a dropdown menu with the text "How can we help you?" and a selected option "Additional service". Below this is a large empty rectangular area. To the right of this area is a button labeled "</> Embed Code", which is highlighted with a red rectangular border. Below the main form area are three separate input fields, each with a label: "hidden_NameValue", "hidden_StateValue", and "hidden_HowCanWeHelpYouValue".

Single left-click on the “Embed Code” block to highlight it.

- j. If the template copied correctly, you should see a preview of it in the small window within the “Field-Specific” section:

The screenshot shows a configuration panel with two sections: "General" and "Field-Specific". The "Field-Specific" section is expanded and contains a code editor with the following JavaScript code:

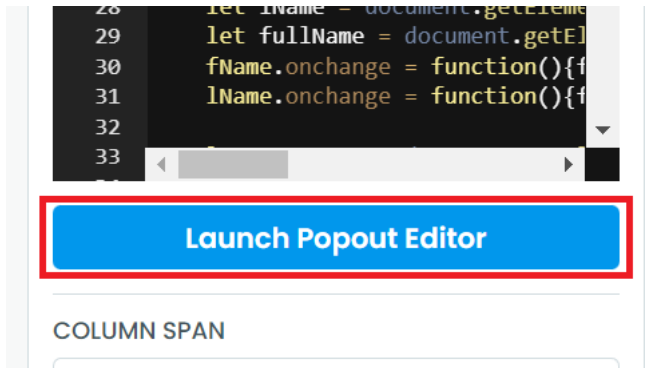
```

1 <script src="https://ajax.googlea
2 <script src="https://webtracking-
3 <script src="https://webtracking-
4 </script>
5 var config = {
6   fields: {
7     "Name": "#hidden_NameValu
8     "FullJobTitle": "#yourJob
9     "MobilePhone": "#yourMobi
10    "Email": "#yourEmail",
11    "Account": "#company",
12    "Website": "#url",
13    "Address": "#addr1",
14    "City": "#cityValue",

```

The code editor is highlighted with a red rectangular border.

- k. Single left-click the “Launch Popout Editor” button:



- l. Replace the entire code present here with the modified in steps 2.h.ii, 2.h.vii, and 2.h.viii (a simple copy/paste exercise). Similar to 2.h.ix, your code should look like the below within Formstack Forms’s Popout Editor:

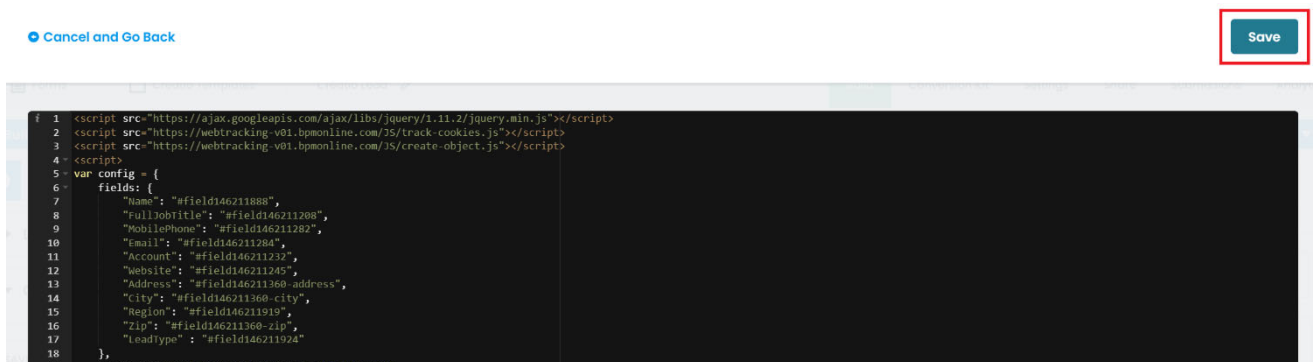
[Cancel and Go Back](#)

```

1 <script src="https://ajax.googleapis.com/ajax/libs/jquery/1.11.2/jquery.min.js"></script>
2 <script src="https://webtracking-v01.bpmonline.com/JS/track-cookies.js"></script>
3 <script src="https://webtracking-v01.bpmonline.com/JS/create-object.js"></script>
4 <script>
5 var config = {
6   fields: {
7     "Name": "#field146211888",
8     "FullJobTitle": "#field146211208",
9     "MobilePhone": "#field146211282",
10    "Email": "#field146211284",
11    "Account": "#field146211232",
12    "Website": "#field146211245",
13    "Address": "#field146211369-address",
14    "City": "#field146211369-city",
15    "Region": "#field146211919",
16    "Zip": "#field146211369-zip",
17    "Leadtype": "#field146211924"
18  },
19  landingId: "309a5b04-244d-437f-acd5-6ff2dd12d779",
20  serviceUrl: "https://1202257-demo.creatio.com/ServiceModel/GeneratedObjectWebFormService.svc/SaveWebFormObjectData",
21  redirectUrl: ""
22 };
23
24 // Fill in the below fields with "#" with the corresponding Formstack Form unique input IDs.
25 // If you do not replace these values with your form's unique field names, this will not work.
26
27 let fName = document.getElementById("field146211051-first");
28 let lName = document.getElementById("field146211051-last");
29 let fullName = document.getElementById("field146211888");
30 fName.onChange = function(){fullName.value = fName.value + ' ' + lName.value;
31 lName.onChange = function(){fullName.value = fName.value + ' ' + lName.value;
32
33 let stateVar = document.getElementById("field146211369-state");
34 let stateStr = document.getElementById("field146211919");
35 stateVar.onChange = function(){stateStr.value = stateVar.selectedOptions[0].innerText;
36
37 let customerNeedVar = document.getElementById("field146211369");
38 let customerNeedStr = document.getElementById("field146211924");
39 customerNeedVar.onChange = function(){customerNeedStr.value = customerNeedVar.selectedOptions[0].innerText;
40
41 function createObject() {
42   landing.createObjectFromLanding(config);
43 }
44
45 function submitToCreatio(){
46   createObject();
47   return false;
48 }
49
50 function initLanding() {
51   landing.initLanding(config);
52 }
53 jQuery(document).ready(initLanding);
54
55 let form = document.querySelector('form.fsForm');
56 form.addEventListener('submit', submitToCreatio);
57
58 </script>

```

- m. Click the “Save” button in the top-right of the Popout Editor:



Upon saving, the changes made will automatically get applied to your Formstack Form.

- n. Return to your Formstack Form’s Live View and refresh the page.
- o. Test that the code works by first making changes to the following fields:
- i. First Name
 - ii. Last Name
 - iii. State
 - iv. How can we help you?

Note: as you make individual changes to the above fields, you’ll notice their linked counterparts will update. Making changes to the “First Name” or “Last Name” fields will alter the “hidden_NameValue” field. Altering the dropdown values for either “State” or “How can we help you?” will alter their respective “hidden_” counterparts.

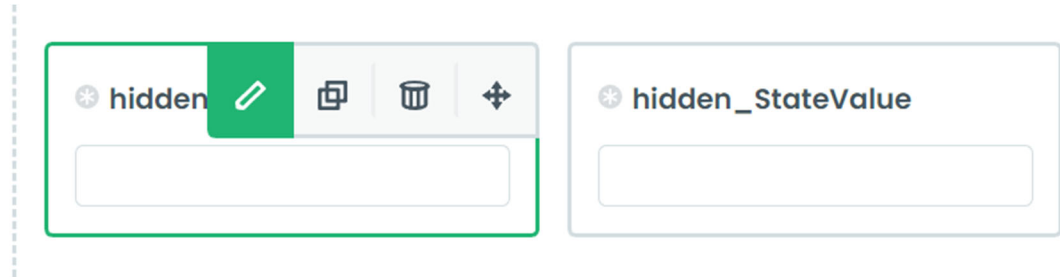
Note: these four fields require some manipulation to ensure that the values obtained from the form arrive into Creatio correctly. The “First Name” and “Last Name” fields must be catenated into one value, and dropdown values from the “State” and “How can we help you?” boxes must be text values. That is the purpose of the “hidden_” fields – to capture the data as it’s needed to be transmitted to Creatio.

- p. If your tests are successful, proceed to the next step. If there are any errors, or the form isn’t behaving as expected – reach out to us at formstack@ss-adv.com, and we’ll schedule a support call to troubleshoot any issues with the sample template example.

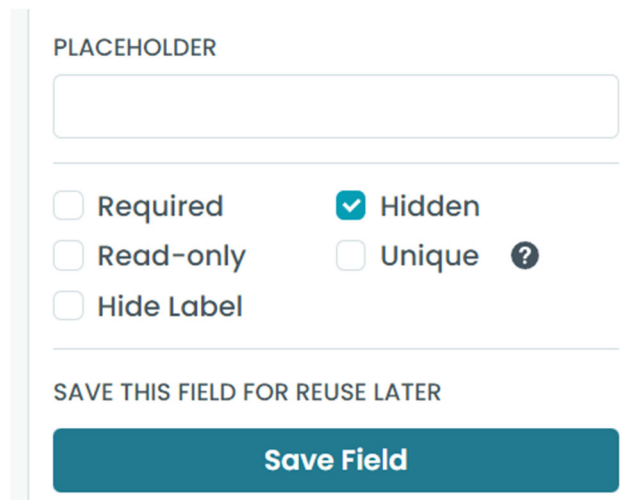
Support for this connector does not include customizations made to the template or the code that does not align with the preceding walkthrough.

- q. Return to the Formstack Form codeless editor page.

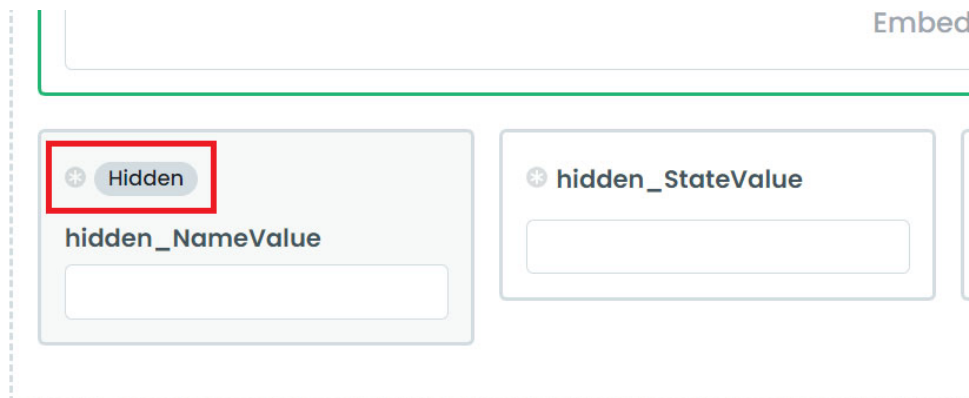
- r. We now need to hide the Repeat the below steps to keep the fields present on the page, but hide them from view:
 - i. Single left-click on the field you wish to hide (“hidden_NameValue” in the below example):



- ii. In the “General” section of the field editor, ensure the “Hidden” value is selected:



- iii. Click on any other field within the editor to ensure that “Hidden” was appended to the field in the form layout view:



- iv. Repeat the above steps for the “hidden_StateValue” and “hidden_HowCanWeHelpYouValue” fields to hide them from your form.

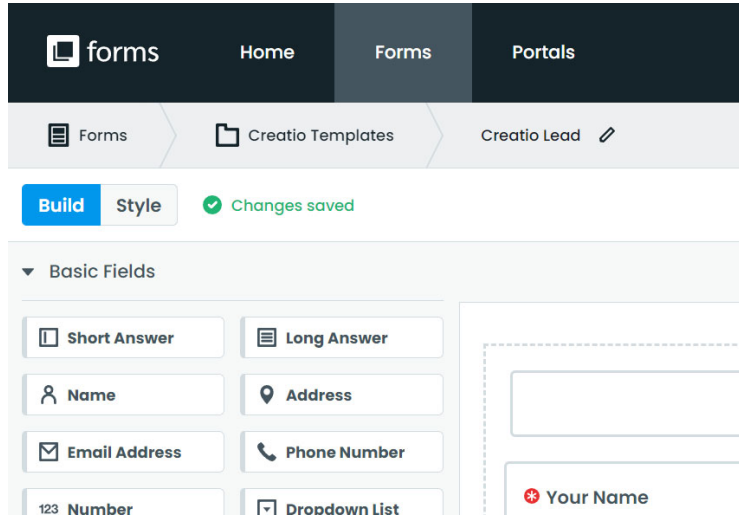
v. Once completed, your form should look the same as below:

Creatio Lead Form

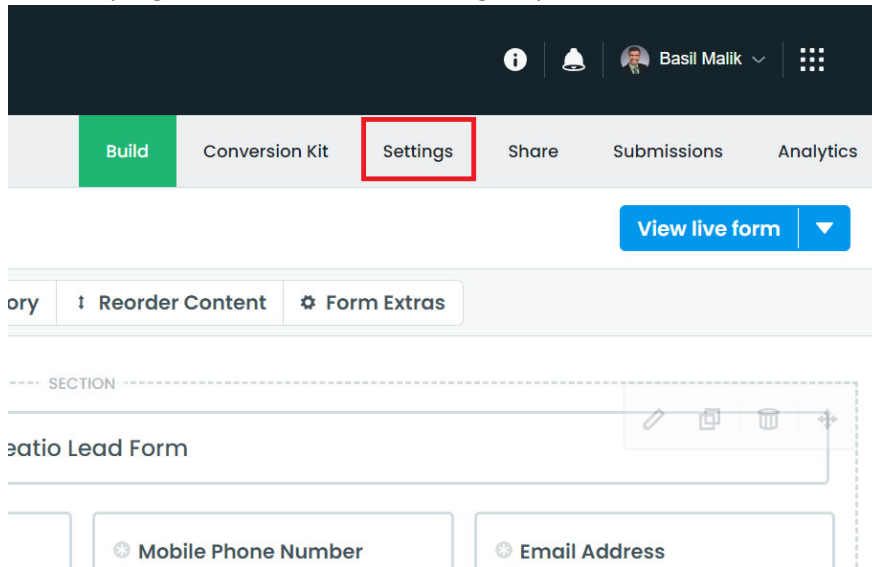
<p>Your Name *</p> <table style="width: 100%; border-collapse: collapse;"><tr><td style="width: 50%; border: 1px solid #ccc; height: 25px;"></td><td style="width: 50%; border: 1px solid #ccc; height: 25px;"></td></tr><tr><td style="font-size: 8px; text-align: center;">First Name</td><td style="font-size: 8px; text-align: center;">Last Name</td></tr></table>			First Name	Last Name	<p>Your Job Title</p> <div style="border: 1px solid #ccc; height: 25px;"></div>	<p>Mobile Phone Number</p> <div style="border: 1px solid #ccc; height: 25px;"></div>	<p>Email Address</p> <div style="border: 1px solid #ccc; height: 25px;"></div>		
First Name	Last Name								
<p>Your Company</p> <div style="border: 1px solid #ccc; height: 25px;"></div>	<p>Website</p> <div style="border: 1px solid #ccc; height: 25px;"></div>								
<p>Address</p> <div style="border: 1px solid #ccc; height: 25px; margin-bottom: 5px;"></div> <p style="font-size: 8px; margin-bottom: 5px;">Address Line 1</p> <table style="width: 100%; border-collapse: collapse;"><tr><td style="width: 40%; border: 1px solid #ccc; height: 25px;"></td><td style="width: 20%; border: 1px solid #ccc; height: 25px; text-align: center;">▼</td><td style="width: 40%; border: 1px solid #ccc; height: 25px;"></td></tr><tr><td style="font-size: 8px;">City</td><td style="font-size: 8px;">State</td><td style="font-size: 8px;">ZIP Code</td></tr></table>					▼		City	State	ZIP Code
	▼								
City	State	ZIP Code							
<p>How can we help you? *</p> <div style="border: 1px solid #ccc; padding: 2px; height: 25px; text-align: center;">Additional service ▼</div>									
<div style="background-color: #0070C0; color: white; padding: 5px 15px; border: none; cursor: pointer;">Submit Form</div>									

- s. Before testing the form, define settings related to email delivery to ensure all test cases and live form submissions get directed to the proper location.

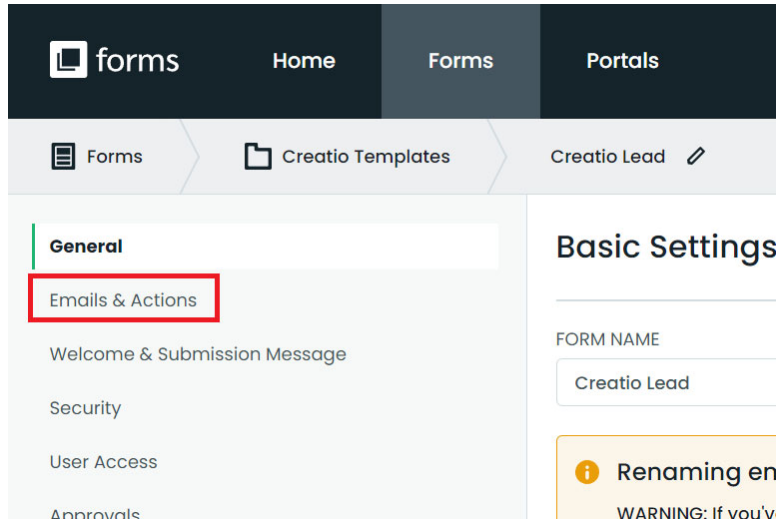
- i. Return to the Formstack Forms Builder:



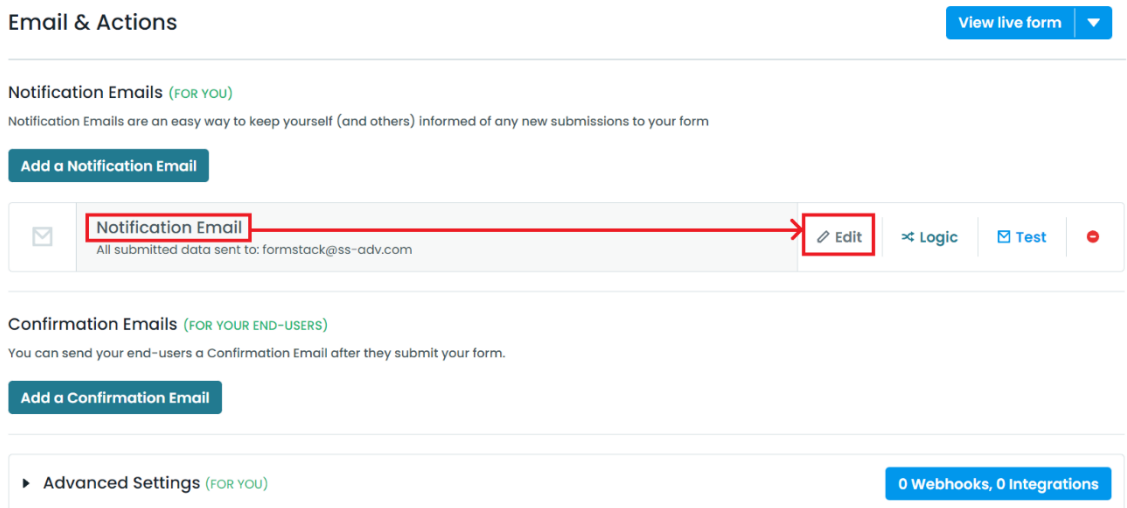
- ii. In the top-right corner, click the "Settings" option:



iii. Single-left click on the “Emails & Actions” option from the settings screen:



iv. Location the default “Notification Email” option and click the “Edit” option:



- v. Define the email addresses you wish to use for your testing purposes (this can always be changed again later):

Notification Email Settings

EMAIL NAME (OPTIONAL)
Notification Email

EMAIL FROM: Custom Address | NAME: Software Selection Advisors, LLC | EMAIL ADDRESS: formstack@ss-adv.com

Important Note
Emails are sent from [noreply@formstack.com](#) (or the default email address set in your account's [SMTP settings](#)). Mapping the "From" address to the email address field on the form will allow you to reply to that email address.

EMAIL TO: formstack@ss-adv.com
Separate email addresses by commas. Maximum of 50 recipients.

SUBJECT: Creatio Lead

- vi. Click the "Save Notification Email" option in the top-right corner of the screen.

Save Notification Email ✓

Form Fields Panel

You can add any of your form's fields or extra submission information to a Notification Email by dragging and dropping them into the Custom Message area.

Search for a field by label

Form Order

All Fields

Form Fields

- vii. You will be returned to the Formstack Forms Builder. Click on the "View live form" button in the top-right of the screen:

Settings | Share | Submissions | Analytics

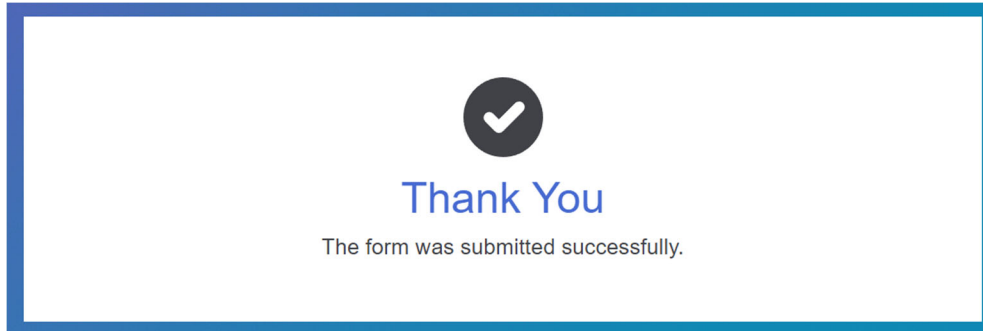
View live form ▼

- t. Test that your form seamlessly transmits the contents of the form to your instance of Creatio by filling out the form and clicking “Submit Form” – below is our test example:

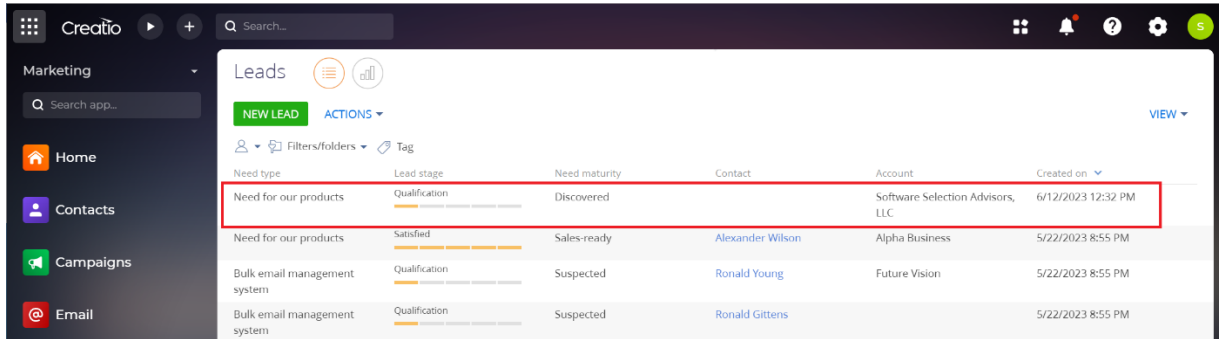
Creatio Lead Form

<p>Your Name *</p> <table style="width: 100%;"><tr><td style="width: 50%;"><input type="text" value="Test"/> <small>First Name</small></td><td style="width: 50%;"><input type="text" value="McTest"/> <small>Last Name</small></td></tr></table>	<input type="text" value="Test"/> <small>First Name</small>	<input type="text" value="McTest"/> <small>Last Name</small>	<p>Your Job Title</p> <input type="text" value="Quality Assurance Tester"/>	<p>Mobile Phone Number</p> <input type="text" value="(918) 600-8700"/>	<p>Email Address</p> <input type="text" value="formstack@ss-adv.com"/>
<input type="text" value="Test"/> <small>First Name</small>	<input type="text" value="McTest"/> <small>Last Name</small>				
<p>Your Company</p> <input type="text" value="Software Selection Advisors, LLC"/>	<p>Website</p> <input type="text" value="https://ss-adv.com"/>				
<p>Address</p> <input type="text" value="10026A S. Mingo Rd. #472"/> <small>Address Line 1</small>					
<input type="text" value="Tulsa"/> <small>City</small>	<input type="text" value="Oklahoma"/> <small>State</small>	<input type="text" value="74133"/> <small>ZIP Code</small>			
<p>How can we help you? *</p> <input type="text" value="Need for our products"/>					
<input type="button" value="Submit Form"/>					

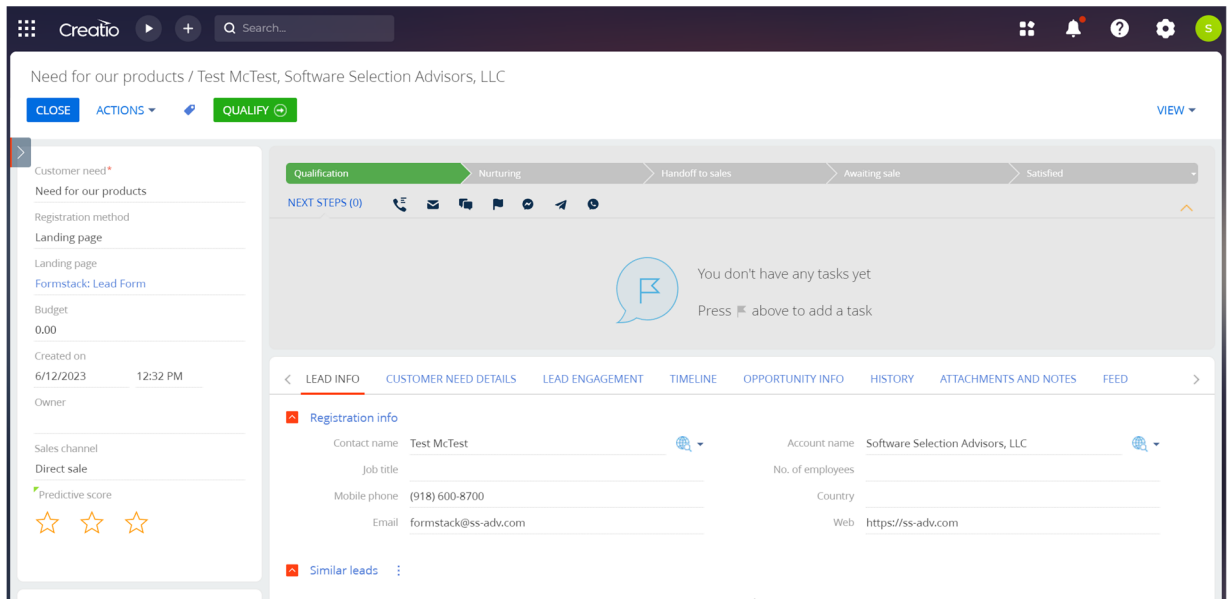
- u. Clicking “Submit Form” should elicit three actions:
 - i. You’ll receive a confirmation in your web browser:



- ii. An email will be received at the email provided in step 2.s.vi.
- iii. A new lead will get created within Creatio.



Clicking into the lead, we can see more details from the form fill:



Conclusion and Future Support

Presuming the above steps were followed correctly, you should now have a fully functioning landing page that creates a new Lead record in your Creatio instance.

For more examples or to obtain help using Formstack Forms to capture information relevant to your business, don't hesitate to contact us at formstack@ss-adv.com.