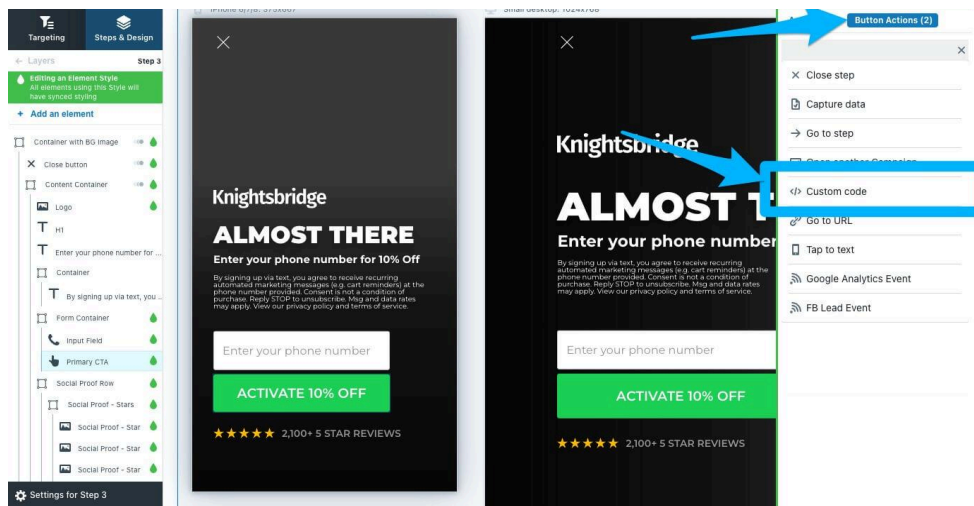
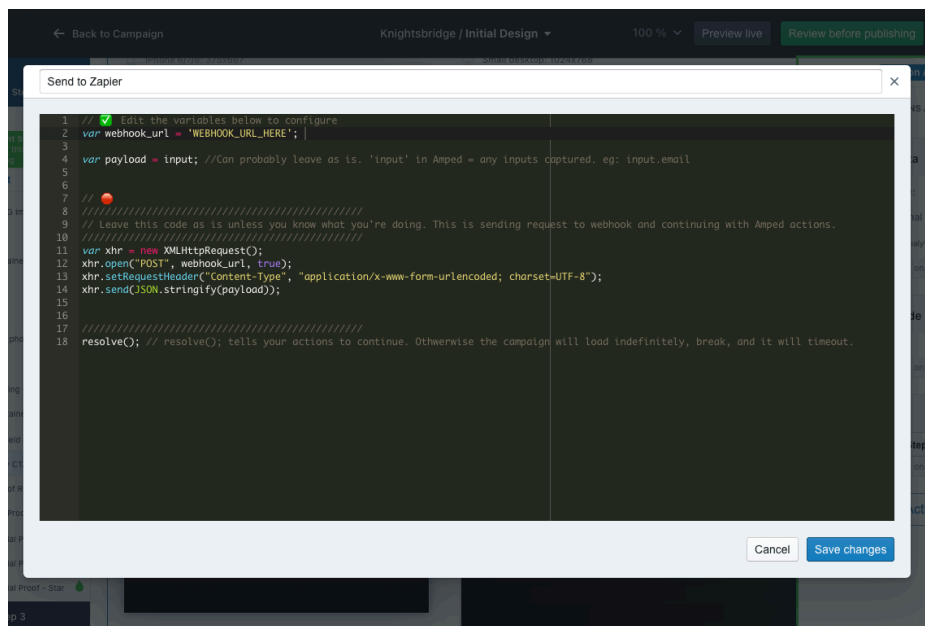


Zapier and Custom Webhooks

If you need to send a service like Zapier, you can simply add a “custom code” action to your buttons when you are capturing data in Mailchimp Forms.



Go to Button Actions -> Custom Code



This will run in the order it is in the Button Actions list



Built in "input" variable available from Custom Code

From a Custom Code block on Button Actions, you can use the already established variable 'input' which is an array for inputs that have been captured up to this point.

e.g. input.email or input.phone_number or any inputs you have set by their name.

You can also access extra information about the current Campaign through the predefined variable of "metadata"

metadata contains

- metadata.campaign_id
- metadata.campaign_title
- metadata.variation_id
- metadata.variation_name

Copy and Paste-able Code for Custom Code Action

```
var webhook_url = "WEBHOOK_URL_HERE"; // replace with Zapier webhook URL
```

```
// //////////////////////////////////////
```

```
// You don't need to edit anything below this line.
```

```
////////////////////////////////////
```

```
// payload is what will be sent to the webhook. We can fill it in with data that needs to be sent to the webhook.
```

```
var payload = {};
```

```
// 'input' contains any inputs captured. eg: the email input can be accessed through input.email or input.phone_number
```

```
payload = input;
```

```
// If we also have user properties and we want to include them in the payload, we can access them from
```

```
var userProperties =
```

```
JSON.parse(window.localStorage.getItem('mcforms-${mailchimpFormsSettings.accountId}-userProperties'));
```



```
if (userProperties){
  payload.userProperties = userProperties;
}
payload.sms_optin_date = new Date().toISOString();

// //////////////////////////////////////
// Keep this and don't edit - This is sending request to webhook and continuing with Amped
// actions.
// //////////////////////////////////////
fetch(webhook_url, {
  method: 'POST',
  mode: "no-cors",
  headers: {
    'Accept': 'application/json',
    'Content-Type': 'application/json'
  },
  body: JSON.stringify(payload),
})
.then(response => {
  if (response.ok) {
    return response.json();
  } else {
    console.log('MC Forms: Custom Webhook Error: Network issue');
  }
})
.then(data => {
  console.log('MC Forms: Custom Webhook sent:', data);
})
.catch((error) => {
  console.log('MC Forms: Custom Webhook error:', error);
});

resolve(); //tells your actions to continue. Otherwise the campaign will load indefinitely, break,
and it will timeout.
```