Collecting Customer data with Booking forms

Last Modified on Jul 1, 2019

Booking forms allow you to collect information from your Customers when they make a booking. After selecting a date and time, Customers reach the Booking form step and provide the information required to complete the booking.

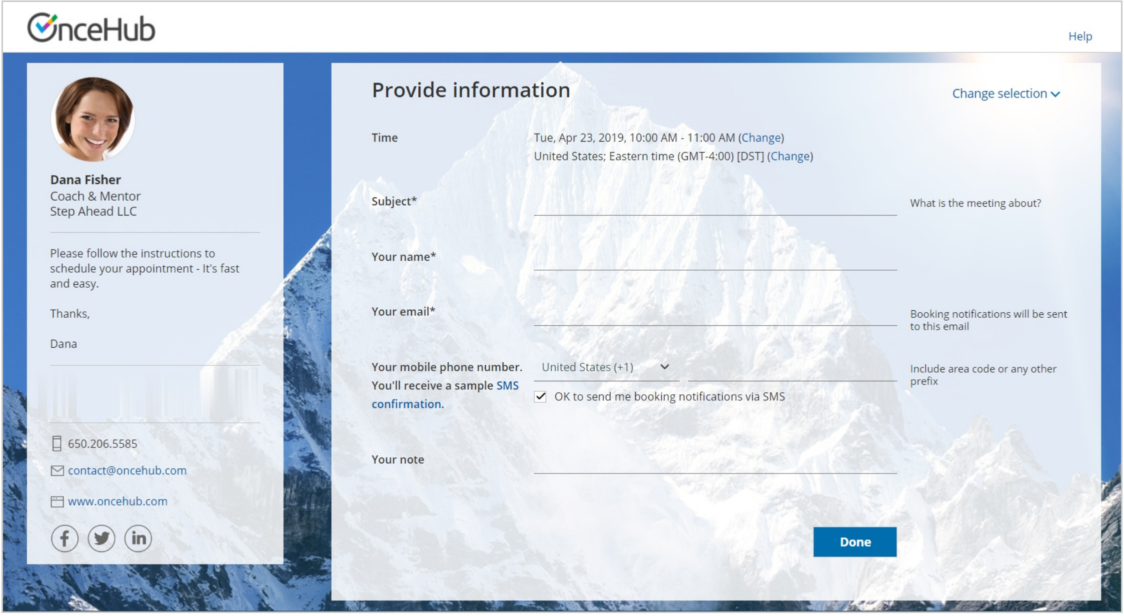

Figure 1: Sample Booking form

In this article, you'll learn about about collecting and utilizing Customer data using Booking forms.

Booking form fields

You can create and customize as many Booking forms as you need using the Booking forms editor. Each form can have its own set of fields in a specified order.

The Booking form always contains the name and email fields, which are ScheduleOnce mandatory fields, plus any System fields or Custom fields you choose to add. You can choose to make these additional fields either mandatory or not.

Some **System fields** will be conditionally added to the form. For example: if you choose to let the Customer set the meeting subject, this field will be automatically added to the Booking form.

The graphic style of the fields is determined by the theme you set in the Theme designer. The graphic style can be either Modern (fields as underlines) or Classic (fields as boxes). By default, the graphic style is set to Modern.

Finding Customer data

All Customer data collected in the Booking form is available to you via the following methods:

- Email notifications, reminders and calendar events (unless set otherwise by creating custom notification templates)

- The Activity stream
- Reports

Selecting a Booking form

When you have multiple Booking forms, you can choose which Booking forms will be used for different scenarios. Booking forms can be related either to Booking pages or Event types.

By default, the location of the Booking form section depends on whether or not your Booking page is associated with any Event types.

- Booking pages **with Event types**: Each Event type has its own Booking form, set in the **Event types's** Booking form section.
- Booking pages **without Event types**: The Booking form is selected in the **Booking page's** Booking form section.

However, you can customize the location of the Booking form section.

Skipping or prepopulating Booking forms

It's not always necessary to collect Customer information as part of the scheduling process. For example, you may already have the data in your CRM.

In these cases, the Booking form is typically skipped completely if you have all the data needed, or prepopulated if you require additional data. Prepopulated Booking forms work in private mode or public mode depending on the source of the prepopulation data. Learn more about prepopulated forms

The Booking form is typically skipped or prepopulated in the following scenarios:

- **Personalized links:** When you invite an existing prospect or Customer to schedule, their information is already known to you. You can send them a Personalized link from Salesforce or Infusionsoft to save them the time of filling out redundant information. If you are not using our native CRM integration, you can create a Personalized link using URL parameters.
- **Web form integration:** The Customer filled out your web form, so their data is already available to you or stored in your CRM. This data can be passed to ScheduleOnce as parameters in the URL or as a CRM record ID (Salesforce or Infusionsoft).
- **Login integration:** When the Customer is authenticated in your app or portal, their identity is known to you. You can pass their CRM record ID (Salesforce or Infusionsoft) or parameters like name or email, and the Booking form will be skipped or prepopulated for them.
- **Landing page:** You send your Customer an email (possibly as part of a campaign) with a link to a landing page (any web page that offers scheduling). That email link is personalized with the Customer data (using URL parameters, Salesforce ID or Infusionsoft ID), and the data is passed to the landing page. The landing page allows the Customer to schedule and all their data is seamlessly passed on to ScheduleOnce, so they don't need to refill known values.