

How to send SMS 2-way session

This article describes the flows related to creating a 2-way SMS survey where Customer Insight Surveys can be sent to customers and they have the ability to respond via SMS to the questions in the survey.

Overview

This document describes the **functionality** and **flow** of a **2-way SMS survey**.

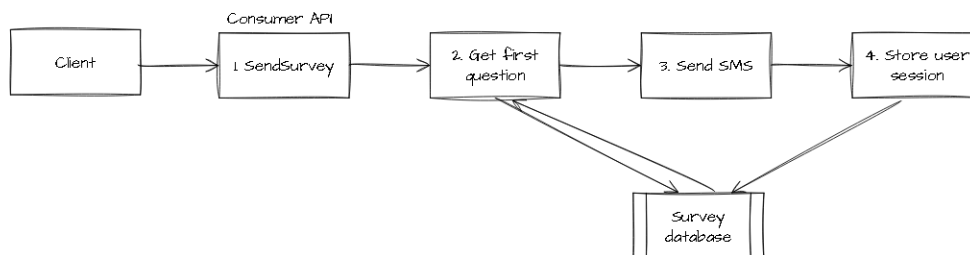
The pre-requisites are to have:

- SMS service configured
- a survey created in our web interface

The flow is as follows:

1. Begins by calling the SendSurvey method in the consumer API
2. This triggers a SMS to the end-user with the first survey question
3. The 2-way SMS handler will process incoming responses from the end-user
4. The 2-way SMS handler will send the correct responses back until the survey is completed.

Flow when initiating a 2-way SMS survey



1. Invoke the SendSurvey method in the consumer API

Example code using rest tag and json body:

```

<assignlocal variable="surveyId" value="2b156ca0-7562-4f2e-b851-e51f0bc27647"/>
<!-- Define survey_expiry if you want it to not be the default 3 days -->
<!-- <assignlocal notexpression="$exist{survey_expiry}" variable="survey_expiry" value="$dateadd(system_datetime,day,3)" /> -->
<include src="/Common/CustomerInsight/SendSurvey.xml"/>

```

Here is an example of the request payload sent through SendSurvey.xml:

```

{
  "clientReference": "${system_session_id}",
  "expiryTime": "${survey_expiry}",
  "userSessionVariables": [
    {
      "key": "agentid",
      "value": "${system_last_user_id}"
    },
    {
      "key": "queueKey",
      "value": "${system_queue_key}"
    }
  ]
}

```

```

    "key": "consumerChannel",
    "value": "Phone"
  },
  {
    "key": "consumerSession",
    "value": "${system_session_id}"
  },
  {
    "key": "consumerPhone",
    "value": "${e164_phone_number}"
  }
],
"identifier": "${e164_phone_number}",
"identifierType": "Phone",
"channel": "SMS",
"content": "On a scale from 1-10 how much did you enjoy the shopping experience with us. Please respond by replying to this message."
}

```

The userSessionVariables can contain **any key/value information** which can later be used in **reporting** and **visualisation** of the results.

Note

We strongly recommend the use of the following standard key/values at a minimum for the results to be useful especially for performance management (agentId and queueKey are essential).

All keys are optional.

| Key name | Description |
|-----------------|---|
| consumerEmail | Email address of consumer |
| consumerPhone | Phone number of consumer |
| consumerChannel | Channel used in the request (SMS, Phone, Web, Email, Social etc.) |
| consumerSession | Identifier for session that handled the request |
| queueKey | Identifier for contact centre queue that handled the request |
| agentId | Identifier for contact centre agent that handled the request |
| userGroupId | Identifier for contact centre user group that handled the request |

Relevant properties:

| Property | Description |
|----------------|--|
| Identifier | Must be a valid E.164 formatted mobile phone number |
| IdentifierType | Must be Phone |
| Channel | Must be SMS |
| Content | For SMS, we advice not to use the Content property but instead configure a text element in the survey itself before the first question element. If used, this value will be prepended to the first SMS. |

| Property | Description |
|------------|--|
| Language | Will be used in preference to survey's default language in 2-way SMS scenarios. Strongly advice to use this so that the end-user gets the survey in his preferred language. Currently these language codes are supported: en - no |
| ExpiryTime | We advice a short expiry time for SMS (max 24 hours) to avoid complications if an end-user receives more than one survey from the same solutions. |

Note

The ExpiryTime, which determines the validity of a survey session sent to a user, if undefined, is 3 days. The ExpiryTime information will need to be defined for a different duration.

2. Get first question element

The API will loop through all survey elements in sequence until it finds an element that requires user input (ie some sort of question, scale, etc).

3. Send SMS

If more than 1 element is found, send one SMS per survey element with a 2 second delay. In cases where the survey has been configured with text elements before the first question, these will be sent first.

If the SendSurvey request contains any Content, the content will be pre-pended to the text in the first SMS.

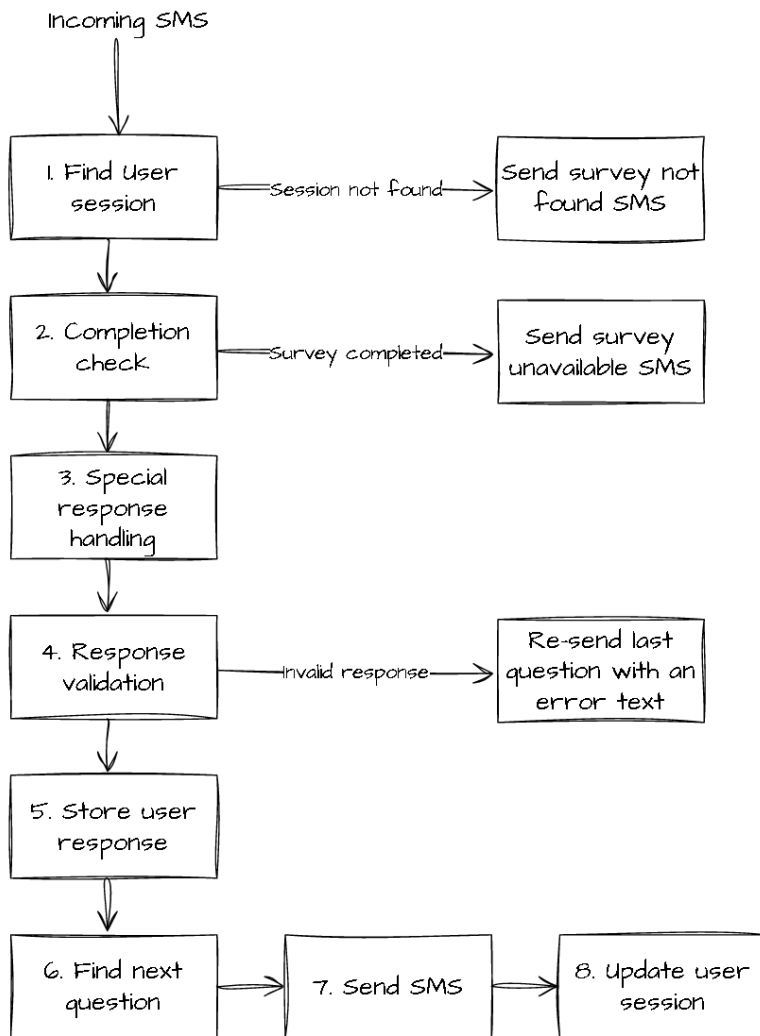
Convert the element into sms text based on the element type.

For Norwegian numbers, subnumbering (sms session) will be used to ease identifying a specific user session.

4. Store user session

Store the user session with currentElementId = <id of the question element sent to end-user>

Flow when handling end-user responses



1. Find user session

Find relevant end-user session either based on SMS session (for Norwegian numbers) or based on end-user's phone number.

If a user session is not found, send a generic survey not found SMS.

2. Survey completion check

If user session.currentElement is the completion element (ie user has already completed the survey), send the text defined for Survey Unavailable.

3. Special handling of dropdown and multiple choice responses

If the currentElement is of type *drop-down list* or *multiple choice*, subtract 1 from the response.

(This is because the responses of these element types are stored with the item number with a starting point of 0, whilst the sms text will number the first item as number 1 when asking for user response as that's deemed a better user experience).

4. Validate the response

Check if user response is of valid data type and within the defined range if a scale is used.

If validation fails, re-send the last question with a pre-pended error text.

5. Register user response

Store the user response in the database.

6. Find next question element

- Check if the currentElement is part of a rule condition, and if it matches the condition criteria, the rule action will determine the next element to send.
- If currentElement does not match any rules, find the next element in the survey sequence
- If no further elements are found, send the text defined in the Survey Completion element

7. Send SMS

Convert the survey element to a sms text based on the element type and send to the end-user.

8. Update user session

Update user session with the new elementId sent to the end-user to keep track of progress.