Confirmit Digital Feedback User Guide

Confirmit Horizons v24

This is document revision 4 of the Confirmit Horizons v24 Digital Feedback User Guide published in April 2019. The information herein describes Confirmit Horizons Digital Feedback and its features as of Build nr. 24.0.952. New features may be introduced into the product after this date. Go to www.confirmit.com or check “News” on the Customer Extranet for the latest updates.

Confirmit makes no representations or warranties regarding the contents of this manual, and specifically disclaims any implied warranties of merchantability or fitness for any particular purpose. The information in this manual is subject to change without notice.

The companies, names and data used or described in the examples herein are fictitious.

Table of Contents

What's New in this Revision?

Note: Only the latest changes to this documentation are listed here. Changes made to earlier revisions are listed in the "Changes to the User Documentation" document which can be downloaded from the Confirmit Extranet at https://extranet.confirmit.com. Note that you will need to log in to the extranet to download this document.

The following changes have been made in revision 4 of the Confirmit Horizons v24 Digital Feedback User Guide:

- An Important note is added to the Adding Scenarios section (see Adding Scenarios on page 11 for more information).
- The Using Templates section is added to the How to Create a New Program chapter (see Using Templates on page 23 for more information).
- The Throttling the Number of Completes... section is added to the How to Create a New Program chapter (see Throttling the Number of Completes in a Timeframe on page 29 for more information).
- The Testing the Program chapter is changed to Deploying and Testing... and the deployment text is added (see Deploying and Testing the Program on page 33 for more information).
- Appendix B: User Experience Summary is added (see Appendix B: User Experience Summary on page 47 for more information).

Note: The general layout and language in this document is continually being corrected, adjusted and improved to ensure the user has the best possible source of information. Only NEW information and details of functionality that has changed since the previous revision are listed here - minor corrections to the text and document layout are not listed.

Important
We need your feedback so we can improve this document and provide you with the information you require. If you have any comments or constructive criticism concerning the content or layout of this documentation, please send an email to documentation@confirmit.com. Please include in your email the section number and/or heading text of the section to which your comment applies.

1. Introduction

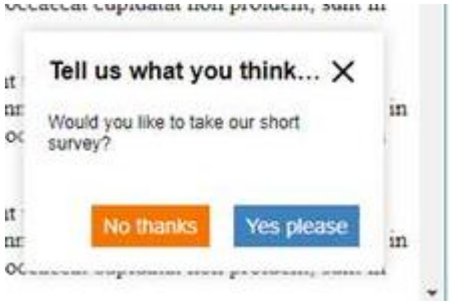
This user guide describes how to use the Confirmit Digital Feedback application. The application is currently under development and is continually being improved and added to, and this documentation will be updated to keep pace with that development.

Digital Feedback is part of the next generation of survey creation software from Confirmit. The application enables you to define multiple intercept scenarios in a single program, using a modern and intuitive interface.

The application provides you with the ability to demonstrate intercept targeting, for example surveys in overlays, without having to continually deploy script links to the website. The application provides a single script hook for one-time integration with the website, and all scenario management is performed via Confirmit Horizons. You have full control over the triggering criteria, and complete customization of the look-and-feel of the invitations and the surveys.

For example, imagine your client has a website to which they want to add overlay surveys to elicit users' feedback. Your sales team wants to demonstrate the possibilities to the client, and the client wants to be able to see how the interactions will look on their website. However at this stage of the project you don't want to make any changes to the website that would effect the presentation as that would disturb the users' experience. Digital Feedback allows you to temporarily add a number different types of surveys to different pages in the website, whilst ensuring the changes are only visible to specified users, for example the web designers or testers.

Each “program” has a snippet of code which users can place onto their web sites. Once properly configured and deployed, the code displays a prompt followed by a survey if the invitation is accepted, or a survey directly, to the site's viewers. This prompt is fully configurable – it could be an overlay dialog, a button or a sidebar item etc.



Tell us what you think...

Would you like to take our short survey?

Figure 1 Example of an invitation pop-up placed on a webpage

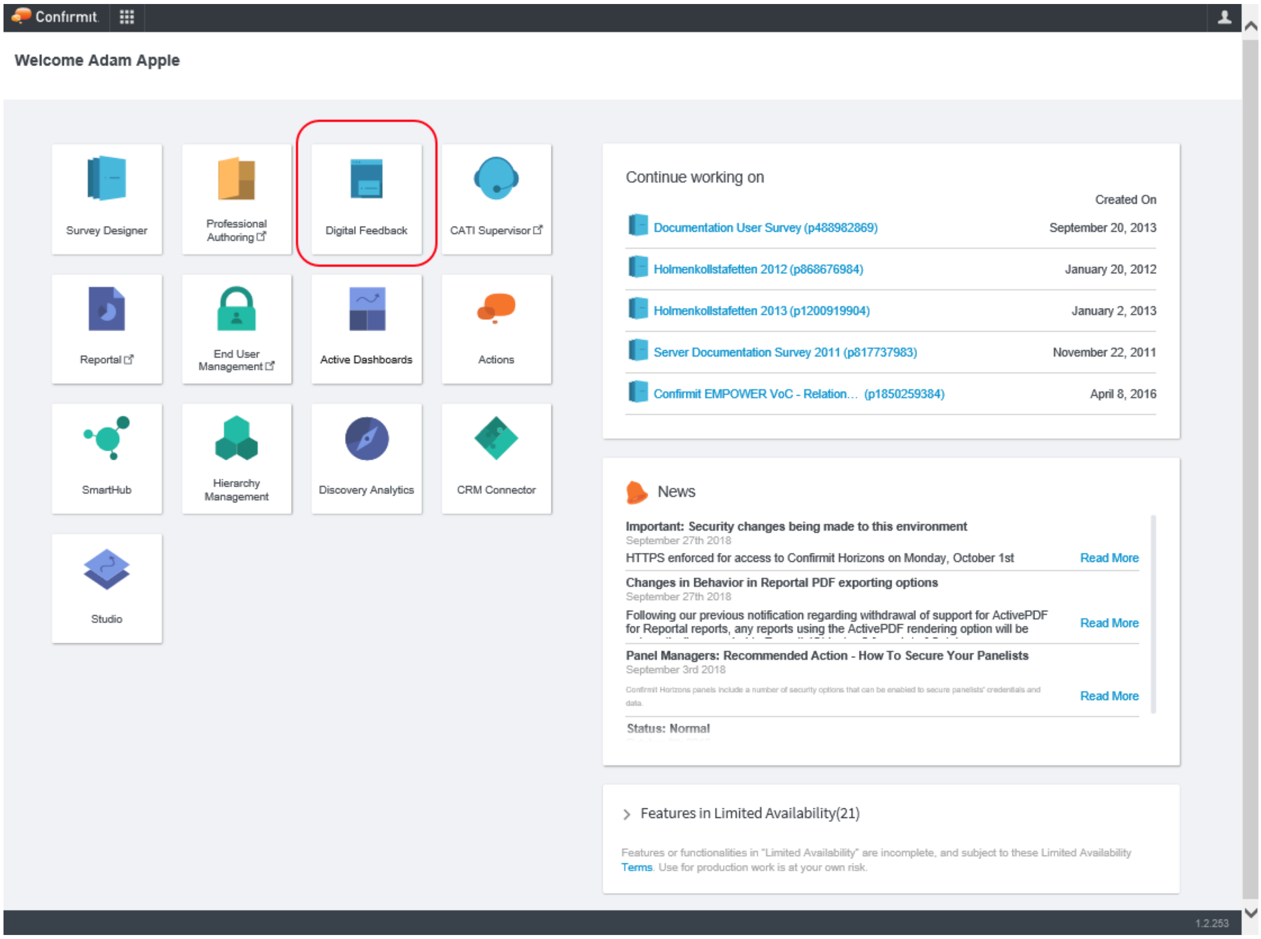
Note that the prompt or invitation is optional; the survey can appear without the respondent being asked first. But the survey must be part of the program and will always appear unless the respondent specifically declines the invitation. If the prompt is used and the invitation is accepted, or if the survey is to open directly, a "normal" Confirmit survey is presented to the viewer as an overlay to the web page.

You can fully customize the look and feel of both the invitation and the "container" that the survey is presented in - you can re-write the entire HTML/CSS code. There are however a few minor syntax conventions that must be followed.

The Digital Feedback module can be used to design programs on Chrome, Firefox, IE 11 and Edge browsers. The targeting experiences created can be used on PCs, tablets and mobile devices (IE 10 and below are not supported).

1.1. How to Access Digital Feedback

The Horizons Homepage is common to the majority of Confirmit's applications. When you first log in to Horizons this page opens, and it provides access to all the Confirmit applications that you are licensed to use. Click the Digital Feedback tile icon to go to that application. Note that if you are not licensed to use an application, its icon will not be displayed in the page.



Welcome Adam Apple

Continue working on

	Created On
Documentation User Survey (p488982869)	September 20, 2013
Holmenkollstafetten 2012 (p868676984)	January 20, 2012
Holmenkollstafetten 2013 (p1200919904)	January 2, 2013
Server Documentation Survey 2011 (p817737983)	November 22, 2011
Confirmit EMPOWER VoC - Relation… (p1850259384)	April 8, 2016

News

Important: Security changes being made to this environment
September 27th 2018
HTTPS enforced for access to Confirmit Horizons on Monday, October 1st Read More

Changes in Behavior in Reportal PDF exporting options
September 27th 2018
Following our previous notification regarding withdrawal of support for ActivePDF for Reportal reports, any reports using the ActivePDF rendering option will be Read More

Panel Managers: Recommended Action - How To Secure Your Panelists
September 3rd 2018
Confirmit Horizons panels include a number of security options that can be enabled to secure panelists' credentials and data. Read More

Status: Normal

Features in Limited Availability(21)

Features or functionalities in "Limited Availability" are incomplete, and subject to these Limited Availability Terms. Use for production work is at your own risk.

Figure 2 Example of the Horizons Homepage

Note: Applications that do not yet have a waffle icon have ☐ beside the application name. These will open in a new window, allowing this Homepage to remain open while you are working in the selected application.

Towards the right side of the Homepage is a list of projects you have recently been working on. This provides easy access to those projects - click on a project to open it.

Below the recently used list is a list of news items. These news items provide information on the latest changes, and other features and functionality that have recently become available.

Towards the bottom of the page is a list of new features and functionality that is currently accessible to you under "Limited availability". These items are still undergoing development, but are made available for testing purposes if your company has agreed to trial them. Please note that use of this new functionality for production work is at your own risk.

1.2. Saving Changes

When you make changes within Digital Feedback, your changes are saved immediately and automatically, so there is no **Save** button in the system.

1.3. The Browser Back Button

The **Back** button in your browser saves the history of your movement through Digital Feedback for the current session. This means that if you wish you can go directly back to a page that you were working with earlier.

To open the history overlay, either right-click on the browser **Back** button or place the pointer onto the **Back** button then click and hold the left mouse button.

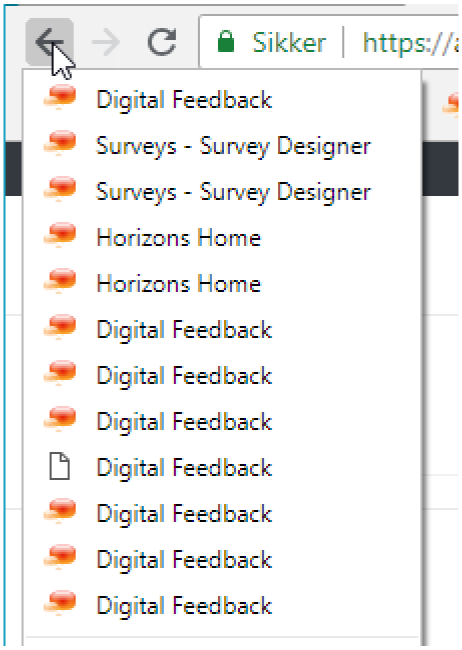


Figure 3 Accessing the history list

Click on the desired page to go directly to that page.

1.4. The Waffle Icon

The **Waffle** ▦ icon in the application's main toolbar is common to the majority of Confirmit's applications. Click the icon to open a selection overlay providing access to all the other Confirmit applications that you are licensed to use, and to the Horizons Homepage (see How to Access Digital Feedback on page 1 for more information). If you are logged in to Horizons and you wish to change to a different application, click the waffle icon and select the appropriate application. The current application is underlined and is also named in the toolbar. Note that if you are not licensed to use an application, its icon will not be displayed in the overlay.

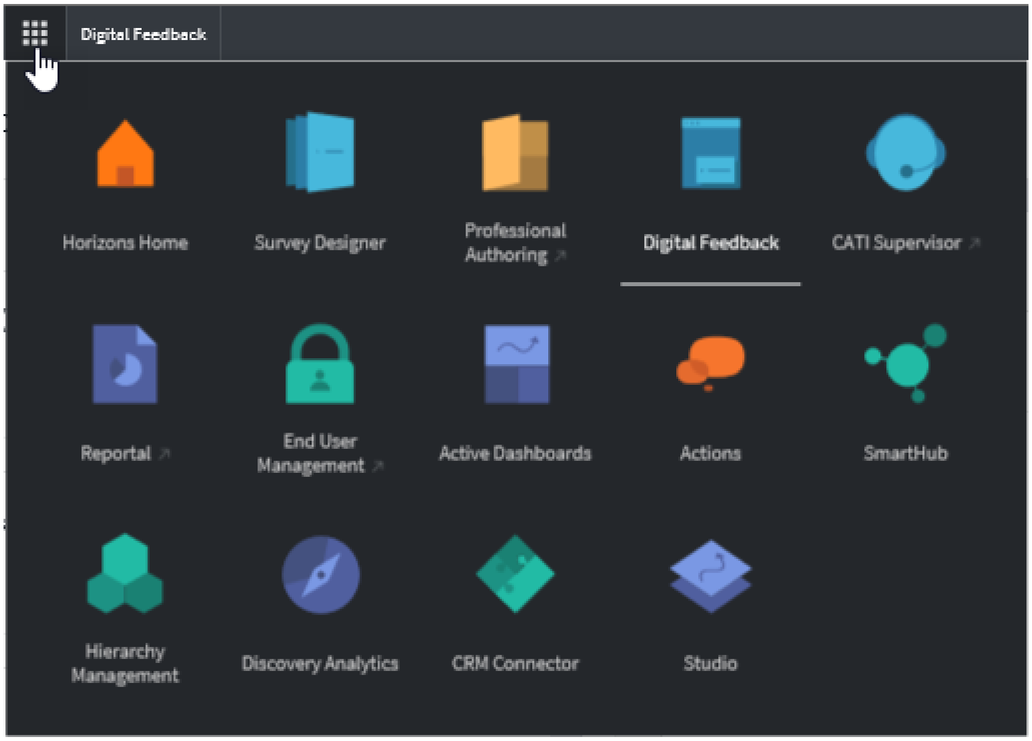


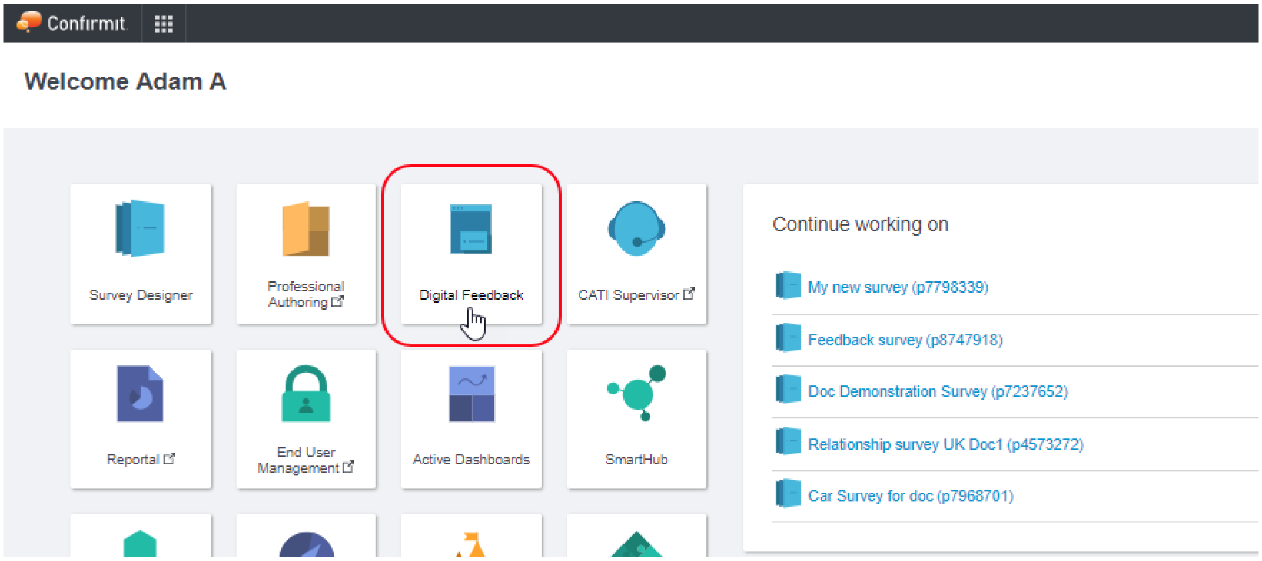
Figure 4 The application selection overlay

Icons with an “arrow” ↗ beside the product name open Authoring or Reportal, icons without the arrow take you directly to the selected application. Refer to the separate user guides for further information on the various applications.

The **Digital Feedback** button beside the waffle icon is always available while you are in Digital Feedback, and returns you directly to the Surveys list (see The Programs List on page 7 for more information).

1.5. Starting Digital Feedback

To start Digital feedback, sign in to Horizons, then on the Horizons Home page or in another application's selection overlay (see The Waffle Icon on page 3 for more information), click the **Digital Feedback** tile.



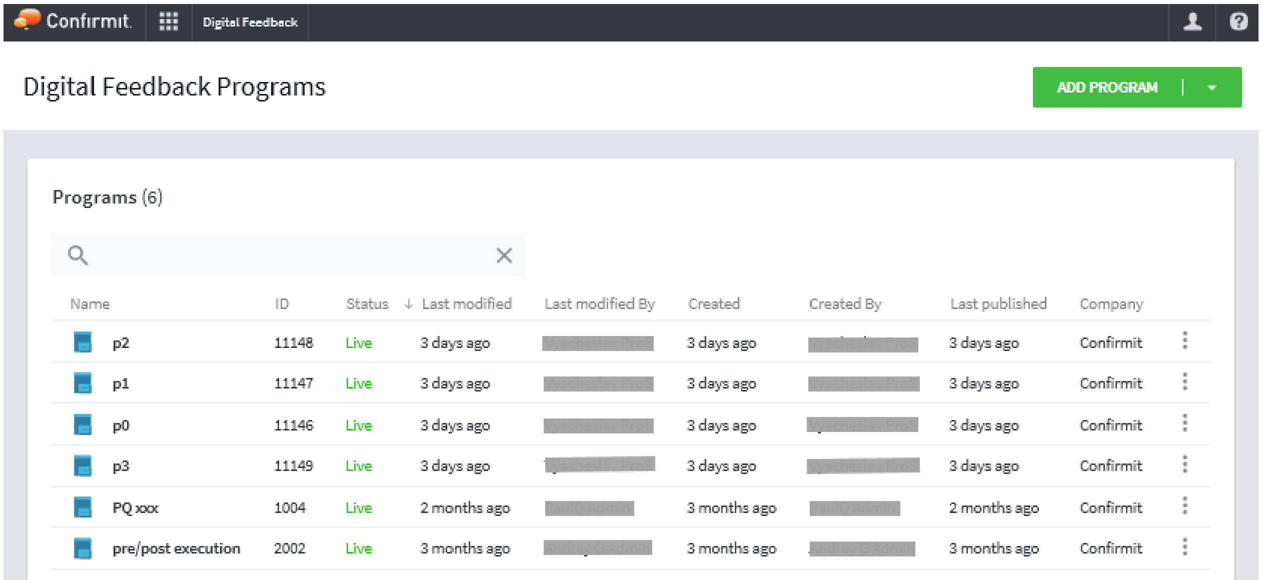
Continue working on

- My new survey (p7798339)
- Feedback survey (p8747918)
- Doc Demonstration Survey (p7237652)
- Relationship survey UK Doc1 (p4573272)
- Car Survey for doc (p7968701)

Figure 5 The Digital Feedback tile in the Horizons Home page

The Digital Feedback Programs List opens (see The Programs List on page 7 for more information).

Note: Click on Digital Feedback in the Confirmit toolbar at any time to return to this list.



Digital Feedback Programs

Programs (6)

Name	ID	Status	↓ Last modified	Last modified By	Created	Created By	Last published	Company
p2	11148	Live	3 days ago	[redacted]	3 days ago	[redacted]	3 days ago	Confirmit
p1	11147	Live	3 days ago	[redacted]	3 days ago	[redacted]	3 days ago	Confirmit
p0	11146	Live	3 days ago	[redacted]	3 days ago	[redacted]	3 days ago	Confirmit
p3	11149	Live	3 days ago	[redacted]	3 days ago	[redacted]	3 days ago	Confirmit
PQ xxx	1004	Live	2 months ago	[redacted]	3 months ago	[redacted]	2 months ago	Confirmit
pre/post execution	2002	Live	3 months ago	[redacted]	3 months ago	[redacted]	3 months ago	Confirmit

Figure 6 Example of the Digital Feedback Programs List

1.6. Logging Out

When you are finished using Digital Feedback, the normal procedure is to log out. This closes Digital Feedback in the correct sequence, ensuring all work and any changes to settings etc. are saved. To log out, click the **User Settings and Logout** button in the upper-right corner of the Horizons window and select **Logout**. The Digital Feedback window closes and you are taken to the Confirmit Login dialog.

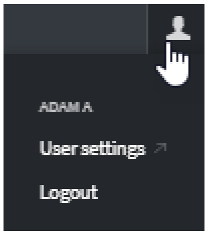
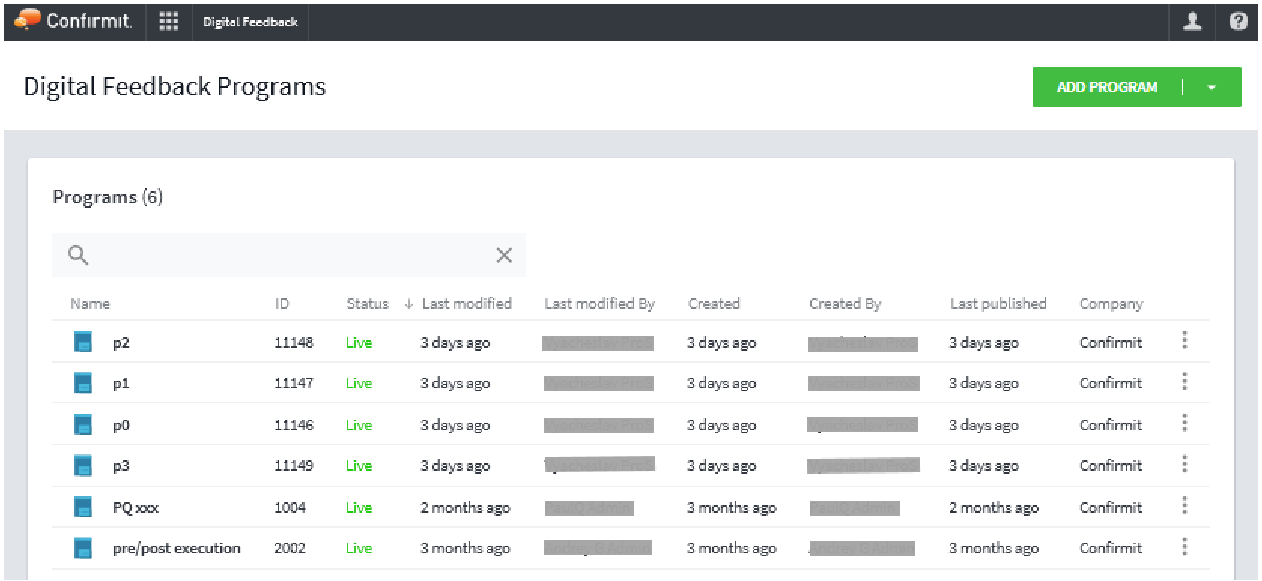


Figure 7 The User Settings icon and menu

2. The Programs List

When you click the **Digital Feedback** tile in the Horizons Homepage (see How to Access Digital Feedback on page 1 for more information), or in the waffle overlay (see The Waffle Icon on page 3 for more information) from another application, the Digital Feedback application opens and you are taken to the Digital Feedback Programs List. This page lists all the programs to which you have access. You can filter the list and search for specific programs (see Searching the List on page 7 for more information), and you can sort the list as you wish (see Sorting the List on page 7 for more information).



Digital Feedback Programs

Programs (6)

Name	ID	Status	↓ Last modified	Last modified By	Created	Created By	Last published	Company	
p2	11148	Live	3 days ago	[redacted]	3 days ago	[redacted]	3 days ago	Confirmit	⋮
p1	11147	Live	3 days ago	[redacted]	3 days ago	[redacted]	3 days ago	Confirmit	⋮
p0	11146	Live	3 days ago	[redacted]	3 days ago	[redacted]	3 days ago	Confirmit	⋮
p3	11149	Live	3 days ago	[redacted]	3 days ago	[redacted]	3 days ago	Confirmit	⋮
PQ xxx	1004	Live	2 months ago	[redacted]	3 months ago	[redacted]	2 months ago	Confirmit	⋮
pre/post execution	2002	Live	3 months ago	[redacted]	3 months ago	[redacted]	3 months ago	Confirmit	⋮

Figure 8 Example of the Digital Feedback Programs list

Click in a program row to open that program.

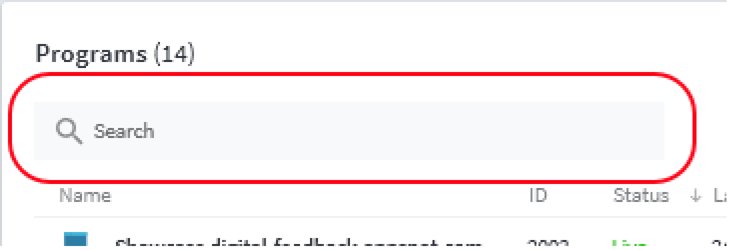
To delete a program, click the **Actions** icon (⋮) for that program, select **Delete** and confirm the deletion task.

2.1. Sorting the List

You can sort the Programs list by any column, up or down. To sort the list, click the appropriate column header; to toggle the sort direction click again.

2.2. Searching the List

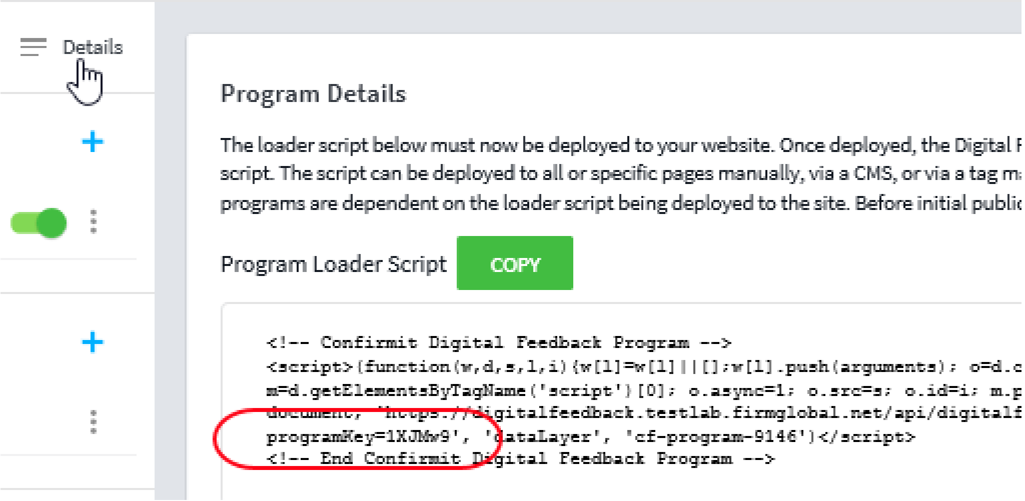
If you have access to a large number of programs, the Search field at the top of the Program list allows you to more easily find a specific program.



Programs (14)

Figure 9 The Search field

Start typing a text string into the Search field. As you type, the Programs list is filtered such that it only includes the programs where the Program name starts with that text string. You can also search by the program key, given in the program loader script in the Program Details field.



Program Details

The loader script below must now be deployed to your website. Once deployed, the Digital F
script. The script can be deployed to all or specific pages manually, via a CMS, or via a tag m
programs are dependent on the loader script being deployed to the site. Before initial public

Program Loader Script

```
<!-- Confirmit Digital Feedback Program -->
<script>(function(w,d,s,l,i){w[l]=w[l]||[];w[l].push(arguments); o=d.c
m=d.getElementsByTagName('script')[0]; o.async=1; o.src=s; o.id=i; m.p
document, 'https://digitalfeedback.testlab.firmglobal.net/api/digitalf
programKey=1XJMw9', 'dataLayer', 'cf-program-9146')</script>
<!-- End Confirmit Digital Feedback Program -->
```

Figure 10 Example of a program key

The search field is not case-sensitive.

2.3. Exporting a Program Definition

You may need to transfer a Digital Feedback program to another Confirmit system, or you may wish to make for example a backup copy. To do this you can export the program.

1. In the Digital feedback Programs list, find the program you wish to export, click its **Actions** icon (⋮), and select **Export** from the drop-down menu.

   The file is zipped and downloaded to your computer's default download folder. The exported file is a zipped JSON file, and this is the only type of file that can be imported into Digital Feedback. The file name will be **programxxxx.zip**, where xxxx is the program id number. Note that if a file of the same name already exists in the download folder, subsequent exports of that program will be denoted with (1), (2) etc. after the program id.

   The file can now be moved or copied to another system as necessary, and imported into Digital feedback (see Importing a Program Definition on page 8 for more information).

2.4. Importing a Program Definition

Once a program has been exported (see Exporting a Program Definition on page 8 for more information), it can be imported into another system. The exported file is a zipped JSON file, and this is the only type of file that can be imported. To import a program:

1. In the Digital Feedback Programs page, click the down-arrow button beside the **Add Program** button, and select **Import Program Definition**.

   A standard file selection dialog opens.

2. Browse to and select the definition file you wish to import, and click **Open**.

   The program definition is added to the Programs list with the same name as the original program. The new program will be given the next consecutive program id number and the publication code snippet will therefore be different compared to the original.

   You can now open the program and edit it as required.

Note: If surveys that are registered in the program do not exist or if the current user does not have access to them, they will be listed as “undefined” and attempts to publish the program will fail. You will in this case need to register a valid survey and update the reference to it.

3. How to Create a New Program

Invitations, containers and surveys can be considered as the “resources” used by different scenarios. These resources can be switched on and off depending on the requirements of a use case. For example, different invitations could be shown on different pages of the same site where exactly the same program is deployed, or you can choose a survey to show based on the viewer's language and preferences etc.

To create a new program:

1. In the Programs List page, click **Add program** towards the upper-right corner of the Horizons window.

   The New program overlay opens.

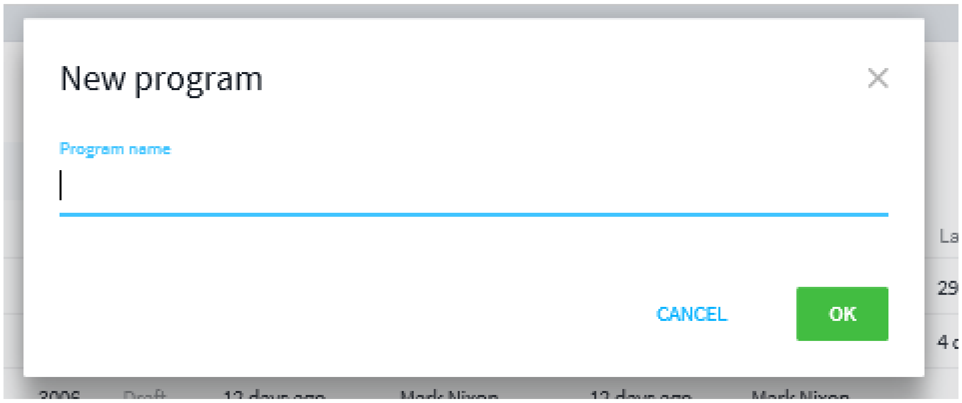
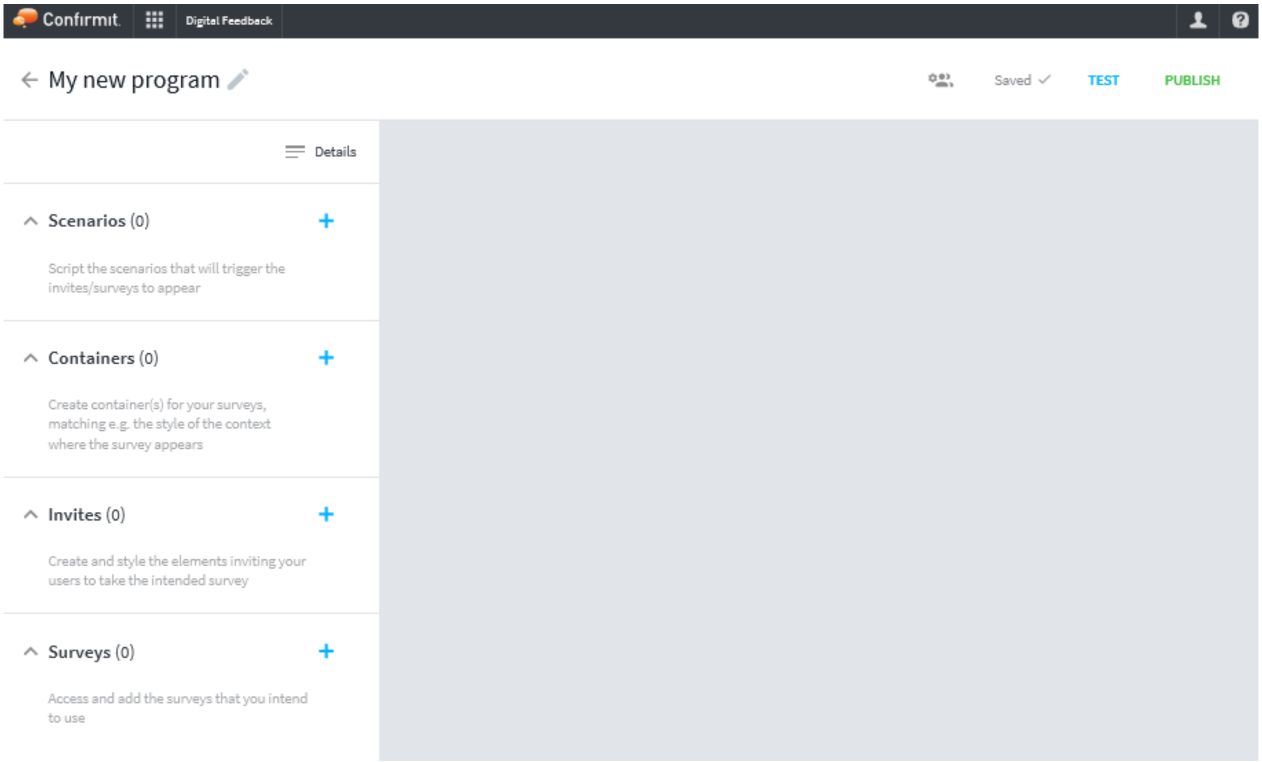


Figure 11 The New program overlay

2. Type in a name for your program, then click **OK**.

   The program's Details list opens.



Confirmit
Digital Feedback
My new program
Saved ✓
TEST
PUBLISH
Details
Scenarios (0)
Script the scenarios that will trigger the invites/surveys to appear
Containers (0)
Create container(s) for your surveys, matching e.g. the style of the context where the survey appears
Invites (0)
Create and style the elements inviting your users to take the intended survey
Surveys (0)
Access and add the surveys that you intend to use

Figure 12 The new program's Details List

You can now add the required content to the new program.

To rename a program, click on the **Edit** icon ✎ to the right of the program name to open the editing field, then edit as required. Note that on completion you must click **Enter** to save your changes; if you click **Esc** or click outside the field the changes will not be saved.

3.1. Adding Scenarios

A core part of an intercept program is the scenario. This is a JavaScript that is executed in the user's browser when he/she visits the web site. It contains a specified set of conditions which, if met by the user, triggers the survey or invitation. A program can have many scenarios, each defining a different set of conditions to be met - you must have one scenario for each situation, and there must be at least one scenario in a program. The JavaScript script controls the display of the invitations and surveys, and can check additional conditions such as the web page url, cookies, the date and time etc. It can pass custom data from the web site to a survey, it can add delays and it can implement behaviors. For example it could show the survey when the user is about to leave the page and moves the cursor to the address bar, or it could trigger an invitation when the user searches for something on the site but doesn't find what they are looking for.

Each scenario can check for the specified conditions, select one invitation, container and survey from those linked to the program, and then display the invitation and/or survey as required. It can also include logic to create different combinations of invitations and containers, or cross-scenario interactions for example “if scenario 1 has been triggered, don’t trigger scenario 2” etc.

Important
When multiple scenarios are to be used in a script, the combined scenario script is created in the order that the scenarios are listed in the program. Any global variables that may be needed in these scenarios must therefore be created at the top of the list.

Note: If you want to pass data from the web page into the feedback survey, delay the loading of the survey invitation button until after the page load is complete.

In this example we will create a scenario where the survey will be triggered when the user intends to leave the page and moves the mouse pointer to the top of the document.

1. Click **+** beside the Scenarios header.

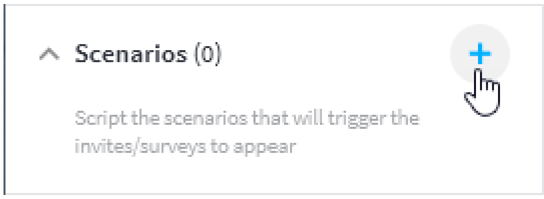

Scenarios (0)
+
Script the scenarios that will trigger the invites/surveys to appear

Figure 13 Creating a new scenario

The New scenario overlay opens.

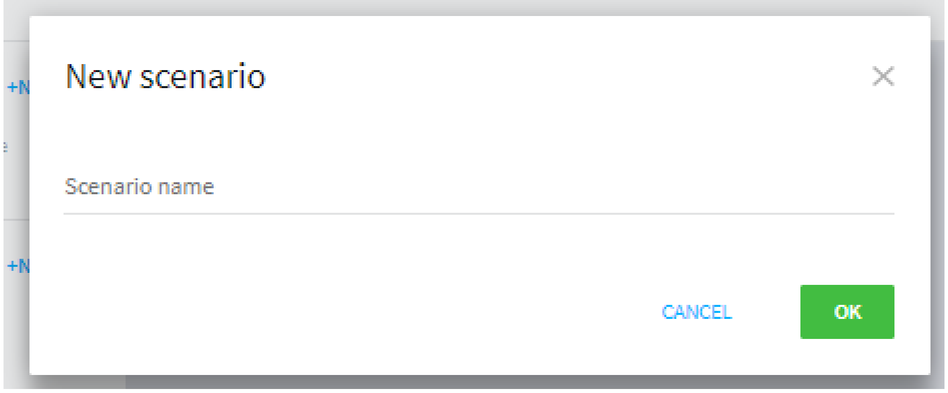


Figure 14 The New scenario overlay

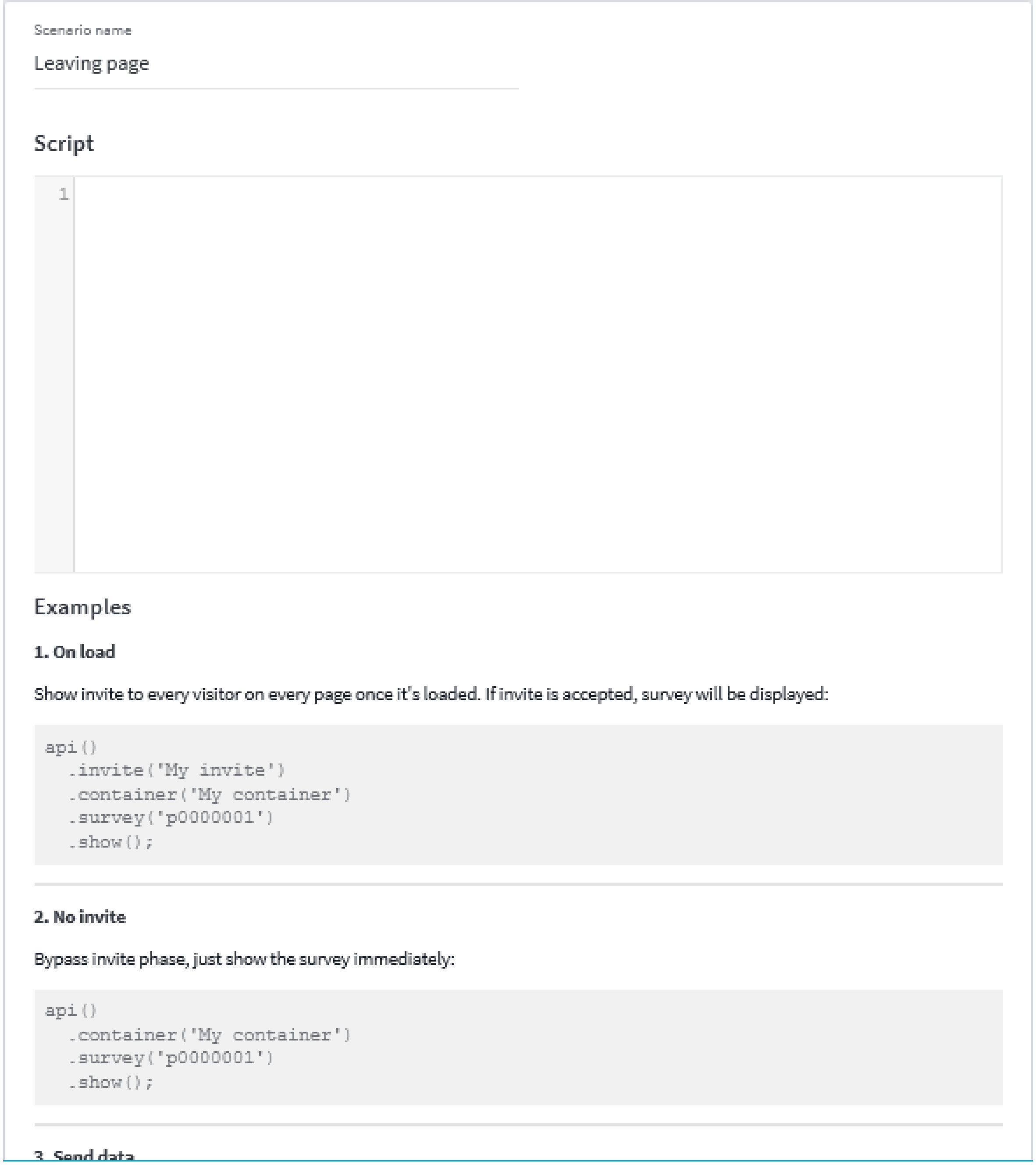
Note: If the program includes templates then the overlay will include a template selection field (see Creating a Scenario Based on a Template on page 28 for more information).

2. Type in a logical name for your scenario, perhaps based on the scenario's function.

Note: To rename a scenario, open the scenario, click into the Scenario name field at the top of the Script page, and edit the name as required. Click out of the field on completion.

3. Click **OK**.

   The scenario creation page opens.



Scenario name

Leaving page

Script

Examples

1. On load

Show invite to every visitor on every page once it's loaded. If invite is accepted, survey will be displayed:

```
api()
  .invite('My invite')
  .container('My container')
  .survey('p0000001')
  .show();
```

2. No invite

Bypass invite phase, just show the survey immediately:

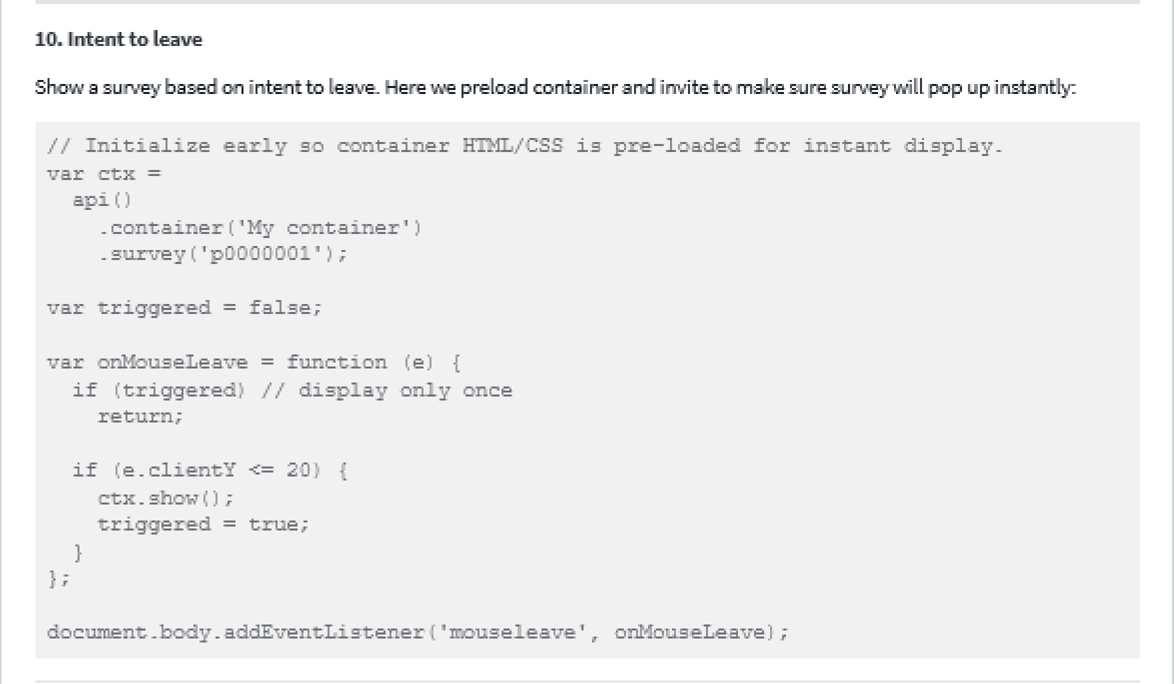
```
api()
  .container('My container')
  .survey('p0000001')
  .show();
```

3. Send data

Figure 15 The scenario creation page

This page lists a substantial number of code examples covering many of the more common scenarios (see List of Scenario Code Examples on page 15 for more information).

4. Scroll through the list to see if your scenario is provided - in this case Intent to leave.



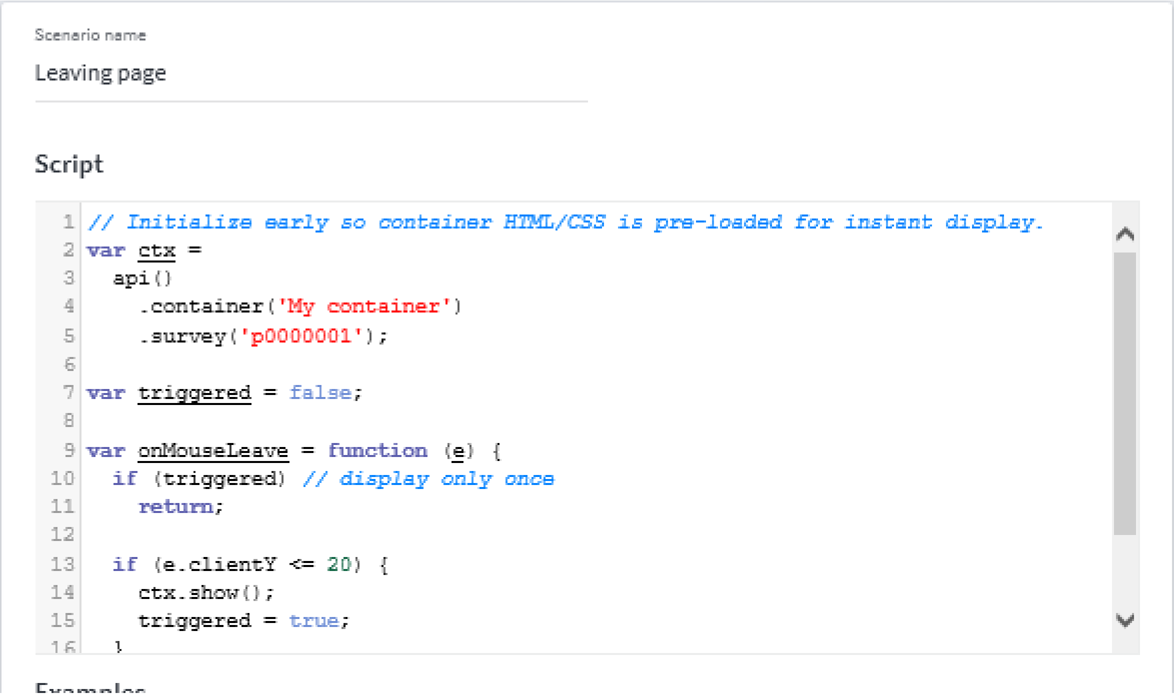
10. Intent to leave

Show a survey based on intent to leave. Here we preload container and invite to make sure survey will pop up instantly:

```
// Initialize early so container HTML/CSS is pre-loaded for instant display.
var ctx =
  api()
    .container('My container')
    .survey('p0000001');

var triggered = false;

var onMouseLeave = function (e) {
  if (triggered) // display only once
    return;

  if (e.clientY <= 20) {
    ctx.show();
    triggered = true;
  }
};

document.body.addEventListener('mouseleave', onMouseLeave);
```

Figure 16 Example of the pre-written code

5. Copy and paste the code for the required template into the Script field.



Scenario name

Leaving page

Script

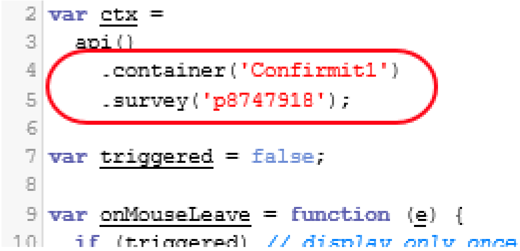
```
// Initialize early so container HTML/CSS is pre-loaded for instant display.
var ctx =
  api()
    .container('My container')
    .survey('p0000001');

var triggered = false;

var onMouseLeave = function (e) {
  if (triggered) // display only once
    return;

  if (e.clientY <= 20) {
    ctx.show();
    triggered = true;
  }
```

Examples

Figure 17 The code copied into the Script field

You now need to link in the invitation you wish to use (if appropriate) (see Adding Invitations on page 19 for more information), the container you wish to use (see Adding Containers on page 17 for more information), and the survey that is to be triggered (see Adding Surveys on page 21 for more information). In this example the invitation will not be used, but the overlay name and the survey id must be placed into rows 4 and 5 of the code respectively, replacing the red place-holder text (keep the **'** characters).

6. Once you have the container you wish to use for the scenario linked in to the program, copy the container name from the Container field (see The Containers Action Menu on page 18 for more information) and paste it over the red **My container** text (keep the **'** characters).
7. Do the same for the survey id for the survey you wish to use for the scenario (see The Surveys Action Menu on page 22 for more information).



```
 2 var ctx =
 3   api()
 4     .container('Confirmit1')
 5     .survey('p8747918');
 6
 7 var triggered = false;
 8
 9 var onMouseLeave = function (e) {
10   if (triggered) // display only once
```

Figure 18 The overlay and survey ids copied into the scenario script

Once a scenario is ready for use, a toggle will be presented in the Program pane to allow you to activate and deactivate the scenario as required.

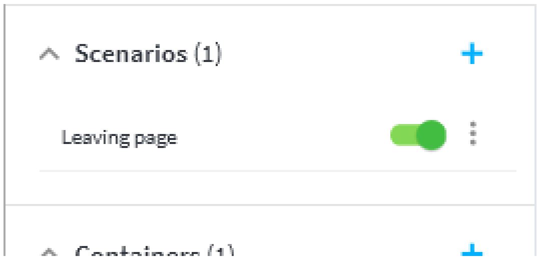


Figure 19 Example of a program's scenario with toggle

The **Actions** icon (⋮) for the scenario opens the **Actions** menu (see The Scenarios Action Menu on page 16 for more information).

3.1.1. List of Scenario Code Examples

A number of code examples for commonly used scenarios are provided to simplify the scenario creation process. Copy the example that provides the result closest to what you are looking for, into the Scenario field. Note that the code examples can be edited as required to give exactly the scenario you want. The examples are:

1. **On load** - presents the invitation to every visitor to the website, on every page they access. If the invitation is accepted the survey is displayed. The visitor can respond to the survey for every page in the website that they visit.
2. **No invite -** for this scenario no invitation is presented, the survey is opened immediately for every visitor to the website, on every page they access, as soon as the page is loaded. The visitor can provide a new response to the survey for every page in the website that they visit.
3. **Send data -** sends custom data from the website to the survey using a query string. This can be used to personalize the survey, for example with the visitor's name if this is known from a log-in.

4. **Nth visitor -** presents the invitation to every Nth visitor. The initial value is every 10th visitor, though this can easily be edited.
5. **Respondent status -**only presents the invitation to respondents of a specified status, for example first-time visitors to the website.
6. **Visited page count -** presents the invitation when the visitor has viewed a specified number of pages in the website.
7. **Delay -** presents the invitation after a preset time delay. This may be useful for example if you want to give the visitor time to view and absorb the information on the web page before you ask him/her their opinion regarding the page.
8. **Resources switching -** can be used to change the resources used in the invitation and survey. For example you may wish to change the language used depending on the visitor's browser language setting.
9. **URL based condition -** presents the invitation only when the visitor opens a specific page in the website. This is achieved by checking the URL of the page.
10. **Intent to leave -** the survey is presented without an invitation when the system senses the visitor is about to leave the website. The trigger is that the mouse cursor moves into the browser's toolbar.
11. **Expiration -** prevents the invitation and/or survey from being presented after a specified date and time, or if the visitor has previously declined the invitation and thereby indicated they do not wish to respond to the survey.
12. **Complex example -** this is a combination of several of the previous examples. This is currently set to present the invitation on a specific page, for every 10th visitor, after a 20-second delay, and only if they have already visited three or more pages in the website. In addition, the template presented is selected based on the user agent language.
13. **Events -** overrides the default behavior and opens the survey in a new browser window.

Further code examples may be added to the list as development and usage progresses.

Note: If you want to pass data from the web page into the feedback survey, delay the loading of the survey invitation button until after the page load is complete.

3.1.2. The Scenarios Action Menu

Click the **Action** icon (⋮) for the scenario to open the **Action** menu.

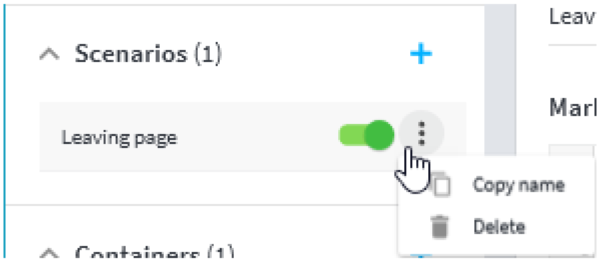


Figure 20 Opening the Scenarios Action menu

- Click **Copy name** to copy the scenario name onto your clip-board. You can then paste it elsewhere into the program code where it may be needed.
- Click **Delete**to the scenario from the program.

3.2. Adding Containers

A "container" is what the survey is displayed in; effectively a style sheet for the survey overlay. Set up a container to specify the size of the survey overlay, it's location in the website window, and its look-and-feel, so you can for example make the survey match the style of the website where the survey is to appear. You can set up any number of containers for a project, perhaps setting them up to position the survey overlay in different locations in the website window for different contexts, or giving them different colors depending on which website page they are opened from.

1. Click **New**.

   The New container overlay opens.

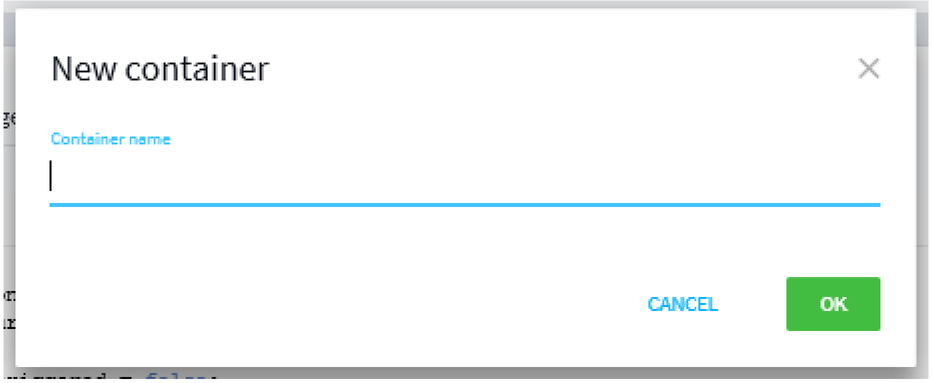


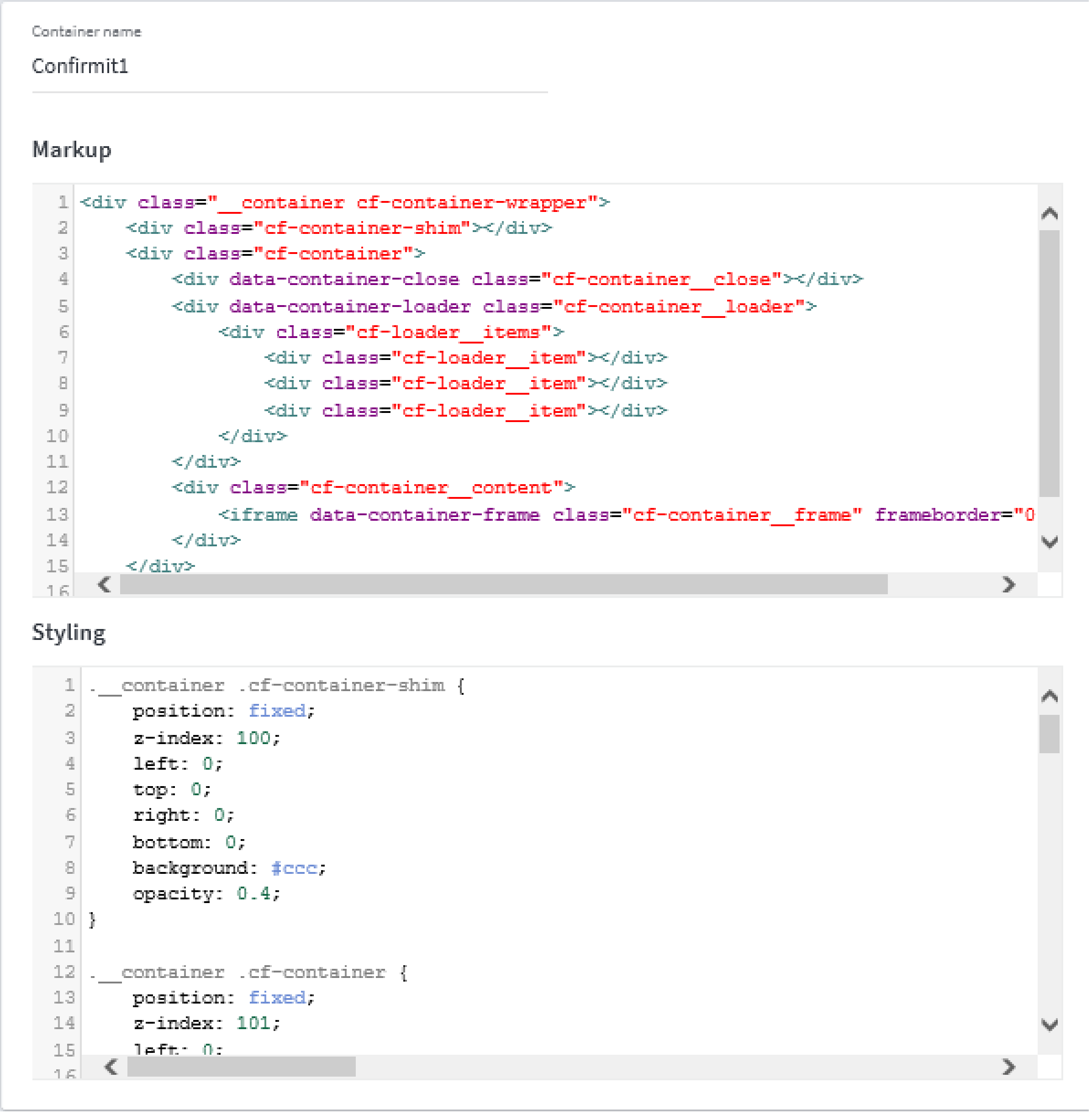
Figure 21 The New container overlay

2. Type in a logical name for your container, perhaps based on the container's styling. Note that you can add as many containers to a program as you wish, but container names must be unique for the current program; duplicates are not allowed.

Note: To rename a container, open the container, click into the Container name field at the top of the Script page, and edit the name as required. Click out of the field on completion.

3. Click **OK**.

   The Container creation page opens.



Container name

Confirmit1

Markup

```
<div class="__container cf-container-wrapper">
    <div class="cf-container-shim"></div>
    <div class="cf-container">
        <div data-container-close class="cf-container__close"></div>
        <div data-container-loader class="cf-container__loader">
            <div class="cf-loader__items">
                <div class="cf-loader__item"></div>
                <div class="cf-loader__item"></div>
                <div class="cf-loader__item"></div>
            </div>
        </div>
        <div class="cf-container__content">
            <iframe data-container-frame class="cf-container__frame" frameborder="0
        </div>
    </div>
```

Styling

```
.__container .cf-container-shim {
    position: fixed;
    z-index: 100;
    left: 0;
    top: 0;
    right: 0;
    bottom: 0;
    background: #ccc;
    opacity: 0.4;
}

.__container .cf-container {
    position: fixed;
    z-index: 101;
    left: 0;
```

Figure 22 The scenario creation page

Default HTML/CSS markup is provided but this is just an example. You can edit this code as necessary to achieve the desired effects or write your own.

The **Actions** icon (⋮) for the container opens the **Actions** menu (see The Containers Action Menu on page 18 for more information).

3.2.1. The Containers Action Menu

Click the **Action** icon (⋮) for the container to open its **Action** menu.

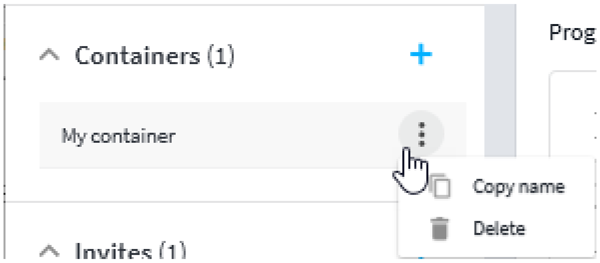


Figure 23 Opening the Containers Action menu

- Click **Copy name** to copy the container name onto your clip-board. You can then paste it elsewhere into the program code where it may be needed, for example into the Scenario script (see Adding Scenarios on page 11 for more information).
- Click **Delete**to remove the container from the program.

3.3. Adding Invitations

When a specific set of conditions (the Scenario) is met by a visitor to the website, for example he/she has visited a preset number of pages in the site, you can display an invitation asking them to take a survey. Note that an invitation is not mandatory; the survey can open as soon as the conditions are met. It's up to you to decide whether an invitation is necessary or appropriate to the situation, but if you wish to present an invitation before the survey opens then you create and style the elements here.

1. Click **New**.

   The New invite overlay opens.

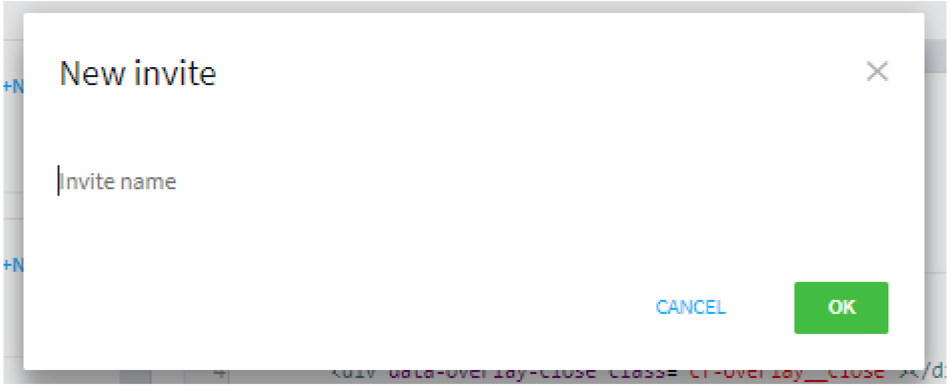


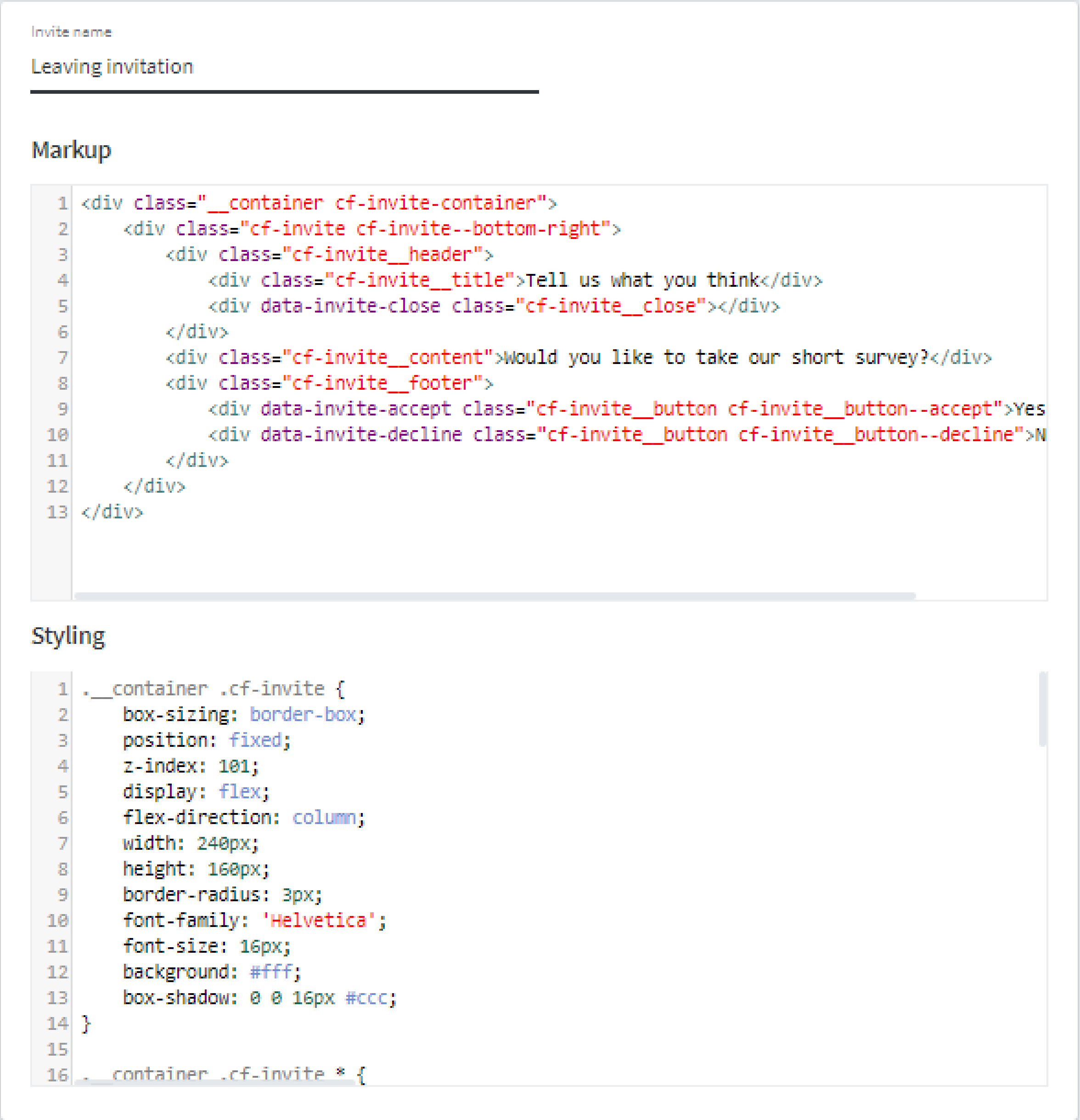
Figure 24 The New invite overlay

2. Type in a logical name for your invitation, perhaps based on the invitation's function. Note that you can add as many invitations to a program as you wish, but invitation names must be unique for the current program; duplicates are not allowed.

Note: To rename an invitation, open the invitation, click into the Invite name field at the top of the Script page, and edit the name as required. Click out of the field on completion.

3. Click **OK**.

   The Invite creation page opens with default code.



Invite name

Leaving invitation

Markup

```
<div class="__container cf-invite-container">
    <div class="cf-invite cf-invite--bottom-right">
        <div class="cf-invite__header">
            <div class="cf-invite__title">Tell us what you think</div>
            <div data-invite-close class="cf-invite__close"></div>
        </div>
        <div class="cf-invite__content">Would you like to take our short survey?</div>
        <div class="cf-invite__footer">
            <div data-invite-accept class="cf-invite__button cf-invite__button--accept">Yes
            <div data-invite-decline class="cf-invite__button cf-invite__button--decline">N
        </div>
    </div>
</div>
```

Styling

```
.__container .cf-invite {
    box-sizing: border-box;
    position: fixed;
    z-index: 101;
    display: flex;
    flex-direction: column;
    width: 240px;
    height: 160px;
    border-radius: 3px;
    font-family: 'Helvetica';
    font-size: 16px;
    background: #fff;
    box-shadow: 0 0 16px #ccc;
}

.__container .cf-invite * {
```

Figure 25 The Invite creation page

Default HTML/CSS markup is provided but this is just an example. A code example is provided in Appendix A (see Appendix A: Code Examples on page 42 for more information). You can edit this code as necessary to achieve the desired effects, or write your own.

The **Actions** icon (⋮) for the invitation opens the **Actions** menu (see The Invites Action Menu on page 20 for more information).

3.3.1. The Invites Action Menu

Click the **Action** icon (⋮) for the invitation to open its **Action** menu.

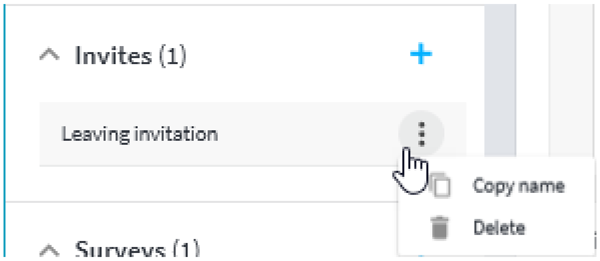


Figure 26 Opening the Invites Action menu

- Click **Copy name** to copy the invitation name onto your clip-board. You can then paste it elsewhere into the program code where it may be needed, for example into the Scenario script (see Adding Scenarios on page 11 for more information).
- Click **Delete**to remove the invitation from the program.

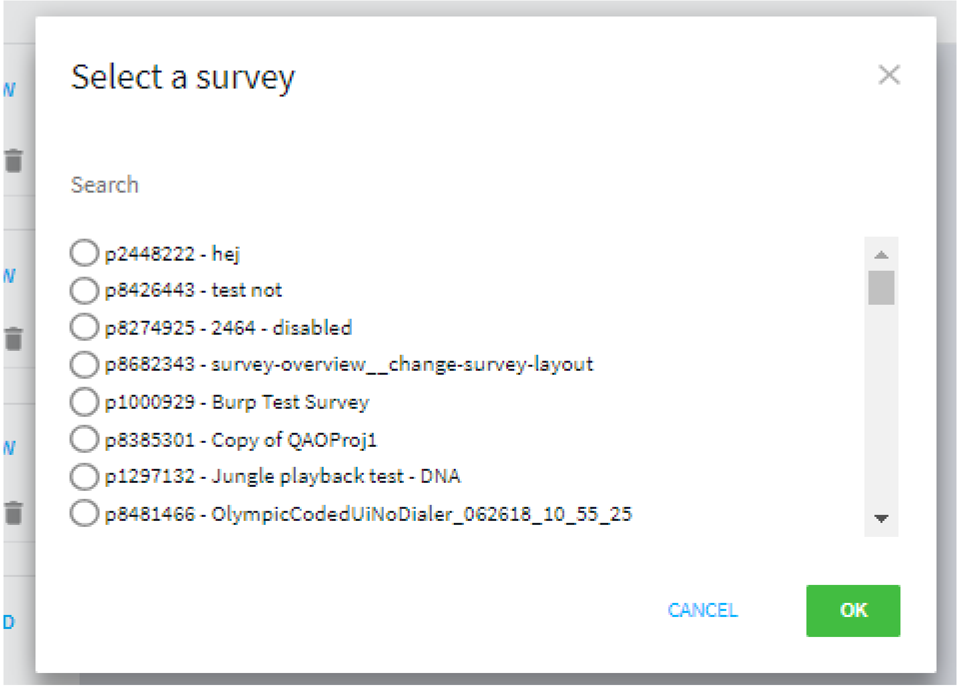
3.4. Adding Surveys

Here you add the surveys that you intend to present to the users, and you can also access those surveys. You can register any number of surveys with a program, and you can register surveys in multiple programs. Note that you must first create the survey(s) using Survey Designer or Professional Authoring.

Important
- The survey to be used MUST NOT have the setting "Prevent survey page from being displayed within a frame" set in Survey Settings > Web.
- It is recommended that the setting "Do not create row until first submission" is enabled .
- The survey must be launched for Production in Survey Designer or Professional Authoring.

1. Click **+Add**.

   The Select a survey overlay opens, with a list of the surveys to which you have access.



Select a survey

Search

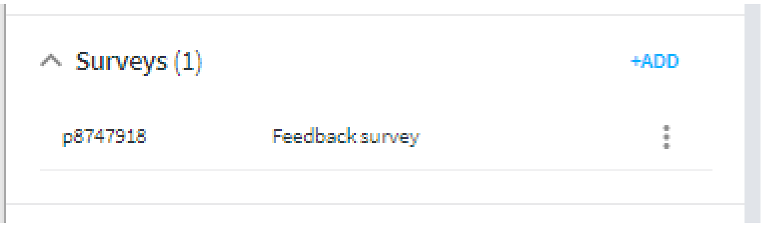
○ p2448222 - hej
○ p8426443 - test not
○ p8274925 - 2464 - disabled
○ p8682343 - survey-overview__change-survey-layout
○ p1000929 - Burp Test Survey
○ p8385301 - Copy of QAOProj1
○ p1297132 - Jungle playback test - DNA
○ p8481466 - OlympicCodedUiNoDialer_062618_10_55_25

Figure 27 The Select a survey overlay

2. Scroll through the list or type search criteria into the Search field to find the survey you wish to use, then select it.

3. Click **OK**.

   The selected survey is added to the list.



Surveys (1) +ADD

p8747918	Feedback survey	⋮

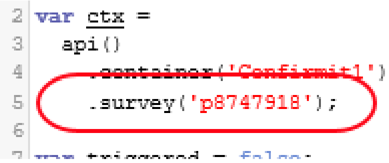
Figure 28 The selected survey is added to the list

Note: You can add as many surveys to a program as you wish.

You must now copy the survey name (the pxxxxxx number) into the scenario script.

4. Select the survey name or click the **Action** icon (⋮) for the survey and select **Copy name** (see The Surveys Action Menu on page 22 for more information)

5. Click on the Scenario name to open the Scenario script, then paste the survey id into the **.survey('pxxxxxx');** line.



```
var ctx =
  api()
    .container('Confirmit1')
    .survey('p8747918');

var triggered = false;
```

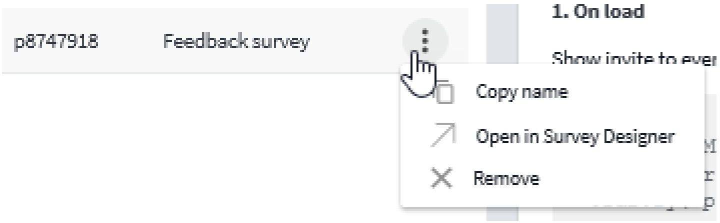
Figure 29 The survey name in the scenario script

Be sure to include the pairs of parentheses and apostrophes.

The survey is now linked into the program, and once the program is published it will be presented to the respondent when the conditions specified in the program are met. You can change the survey at any time by adding a different survey to the Surveys list and copying the new survey's name into the scenario. Until the program is publish, any changes made will only be visible in test mode.

3.4.1. The Surveys Action Menu

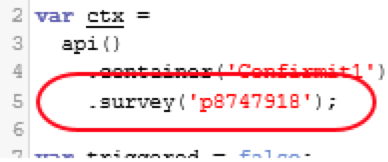
From Digital Feedback you can go to Survey Designer to edit your survey, and you can remove the survey from the program. Click the **Action** icon (⋮) for the survey to open the **Action** menu.



p8747918 Feedback survey

Figure 30 Opening the survey's Action menu

- Click **Copy name** to copy the survey id onto your clip-board. You can then paste it into the scenario script.



```
var ctx =
  api()
    .container('Confirmit1')
    .survey('p8747918');

var triggered = false;
```

Figure 31 The survey id in the scenario script

- Click **Open in Survey Designer** to open a new window in your browser and run Survey Designer with the survey. You can now edit the survey as required.
- To remove the survey from the program, select **Remove**, then confirm the removal operation.

Note: The survey is not deleted; it is only removed from the program. The survey will still be available in the survey list to be used in other programs. If you wish to delete the survey, this operation must be performed in Survey Designer or Professional Authoring.

3.5. Using Templates

Digital Feedback generally uses scripting to set up the scenarios, containers etc. However not all users have experience with scripting. Templates can therefore be used to allow someone who does know scripting to create a program, then users with less scripting knowledge can adjust values and constraints by merely inputting text and numerical values into a form.

A template consists of a series of parameters that are referenced by the template's script.

A program can include any number of templates, and these templates can be simple, perhaps allowing just one parameter to be adjusted, or complex with multiple parameters on the input form.

3.5.1. Creating a Template

Note that the template can be as complex as necessary, and can include scripting and any number of input forms. In this example we will create a simple template that allows the user to input a delay, in seconds, between when the visitor opens the web page and when the invitation opens. To create the template:

1. Click the + button beside the Scenario Templates header.

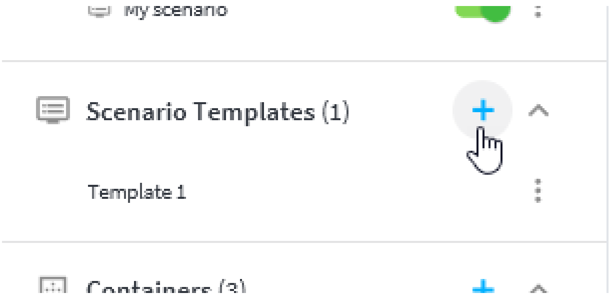


Figure 32 Creating a new template

The New scenario template overlay opens.

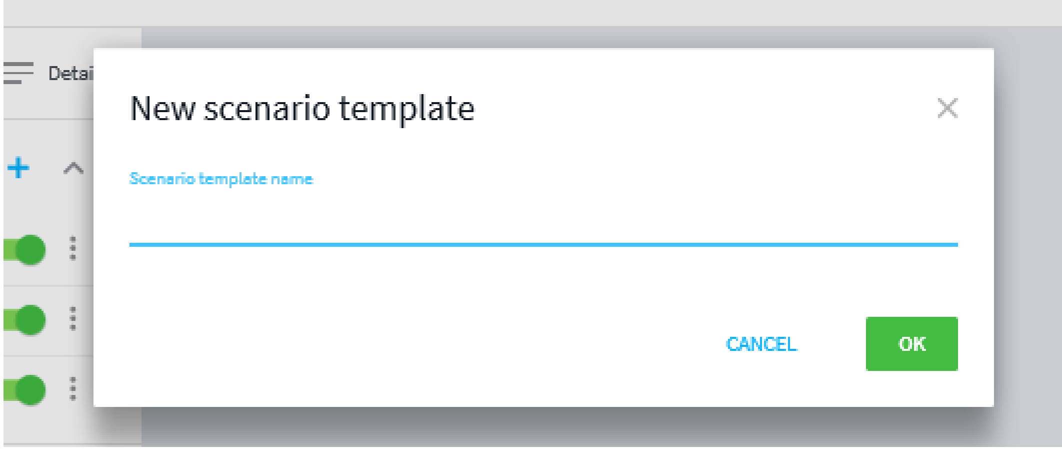


Figure 33 The New scenario template overlay

2. Type in a name for your template and click **OK**.

   The template scripting overlay opens.

3. Click the blue + button in the lower-right corner of the page to create a new parameter.

   The New scenario template parameter overlay opens.

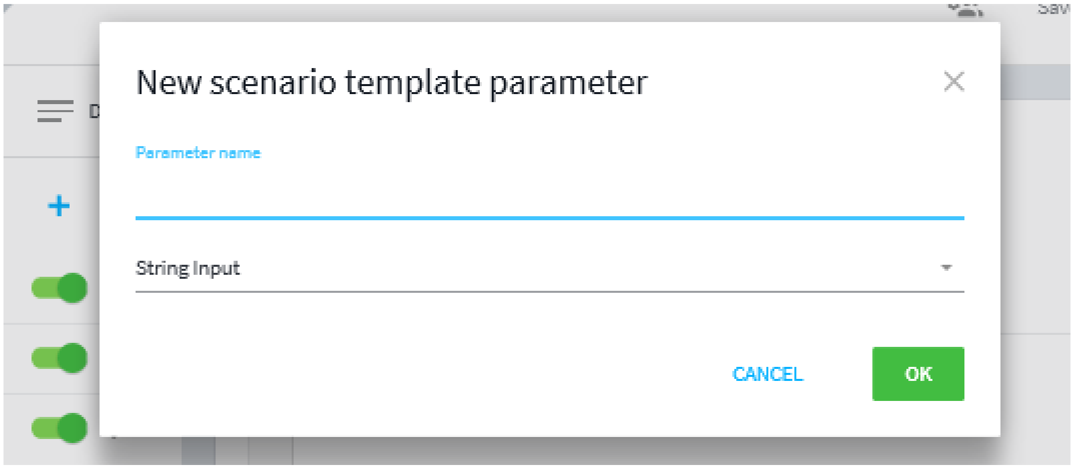
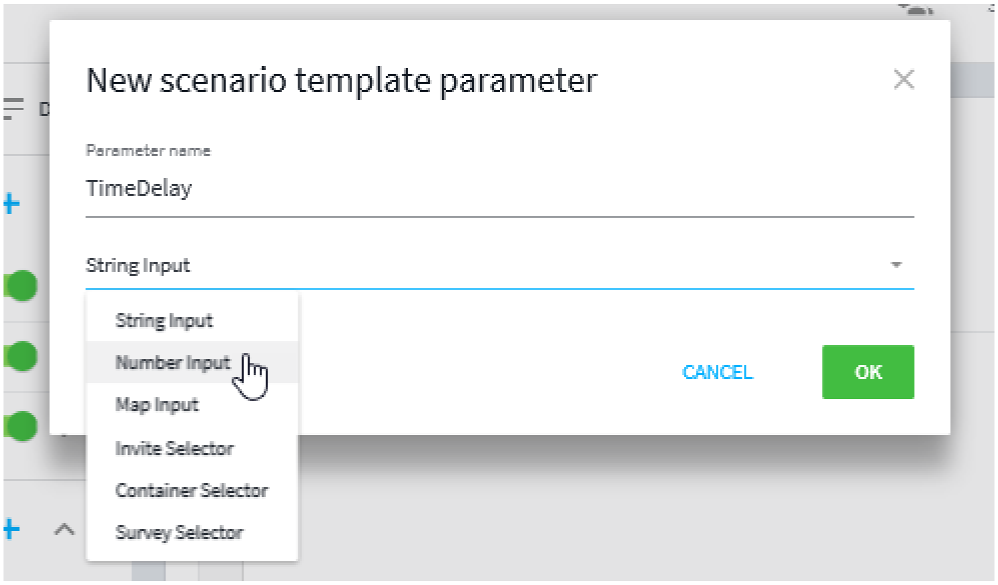


Figure 34 The New scenario template parameter overlay

4. Click into the Parameter name field and type in a name for the parameter (spaces are not allowed).
5. Click into the Input type field and select from the drop-down list the type of input you wish to have for this parameter.



New scenario template parameter

Parameter name

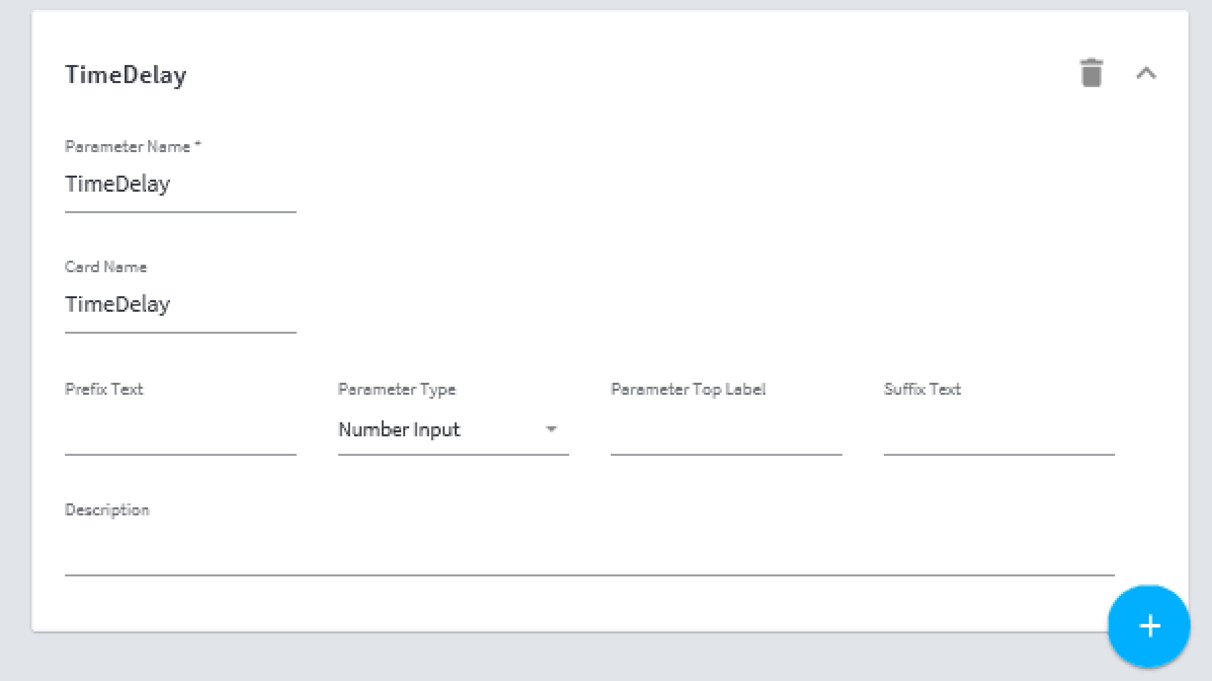
TimeDelay

String Input

Figure 35 Selecting the type of input

6. Click **OK**.

   The parameter form opens.



TimeDelay

Parameter Name *

TimeDelay

Card Name

TimeDelay

Prefix Text

Parameter Type

Number Input

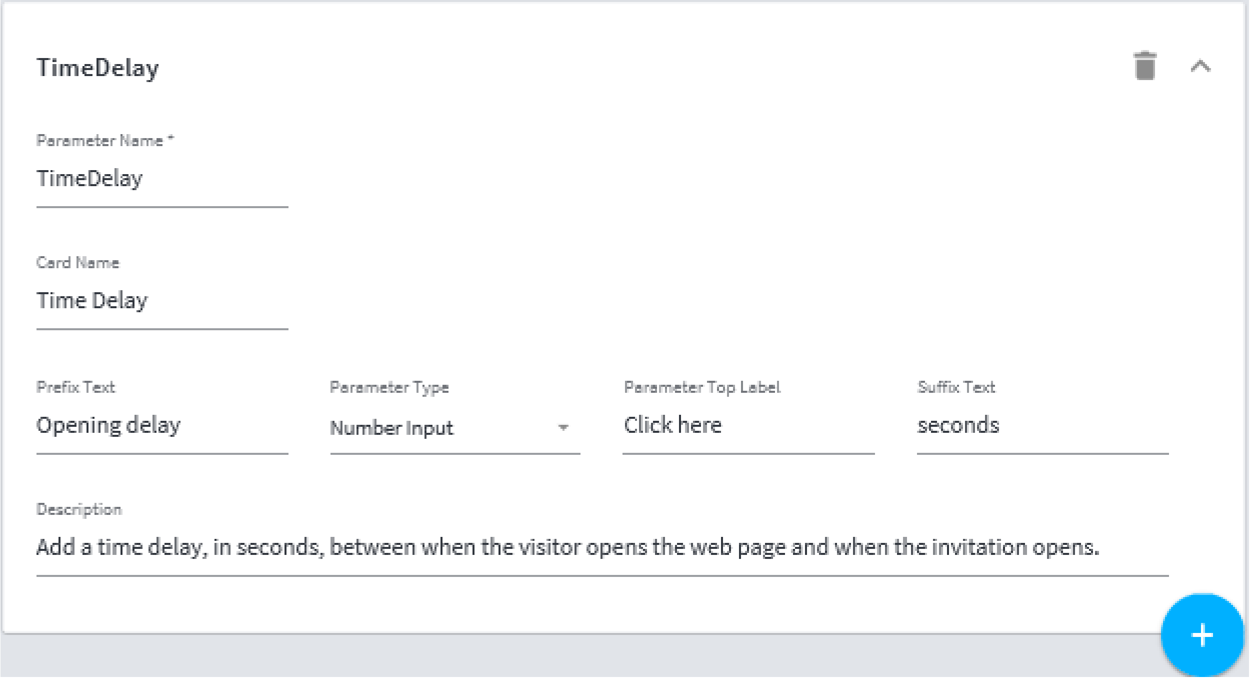
Parameter Top Label

Suffix Text

Description

Figure 36 Example of the parameter form

7. You can now edit the form to include prefix and suffix text, a description etc..



TimeDelay

Parameter Name *
TimeDelay

Card Name
Time Delay

Prefix Text
Opening delay

Parameter Type
Number Input

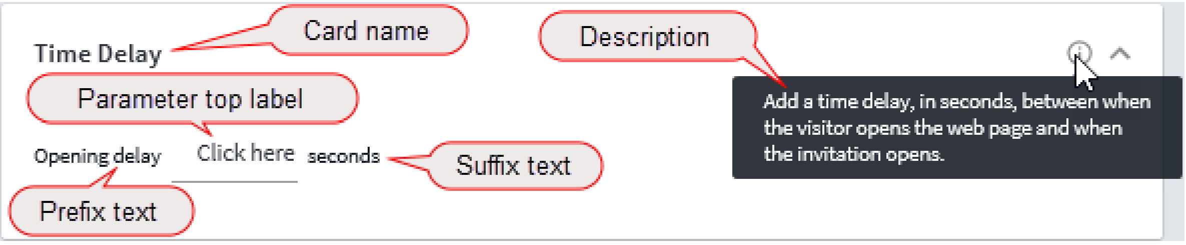
Parameter Top Label
Click here

Suffix Text
seconds

Description
Add a time delay, in seconds, between when the visitor opens the web page and when the invitation opens.

Figure 37 The template form with texts

The various texts you can add to the template appear in the template-based scenario as below.



Time Delay

Opening delay Click here seconds

Add a time delay, in seconds, between when the visitor opens the web page and when the invitation opens.

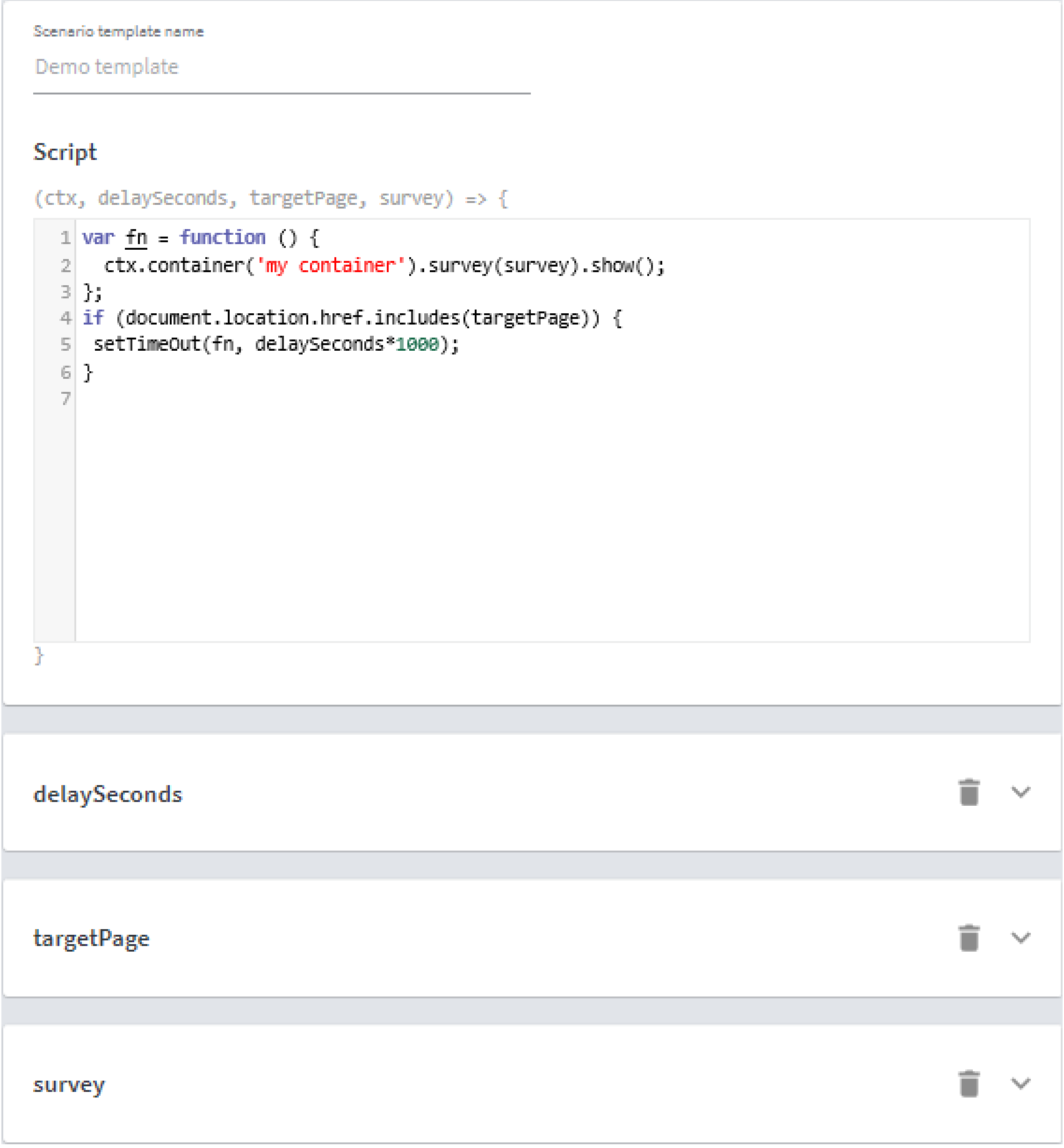
Figure 38 The scenario based on the template

You can now use the template to create a scenario (see Creating a Scenario Based on a Template on page 28 for more information).

3.5.2. Scripting in Templates

The "scripter" creates the template and includes parameters. The script in the template references the parameters and shows these in the Scenario, and a "non-scripter" can then change the parameters to adjust the settings instead of having to edit the script.

In the template example below, the script references the three parameters delaySeconds, targetPage, and survey.



Scenario template name

Demo template

Script

```
(ctx, delaySeconds, targetPage, survey) => {
var fn = function () {
  ctx.container('my container').survey(survey).show();
};
if (document.location.href.includes(targetPage)) {
 setTimeOut(fn, delaySeconds*1000);
}

}
```

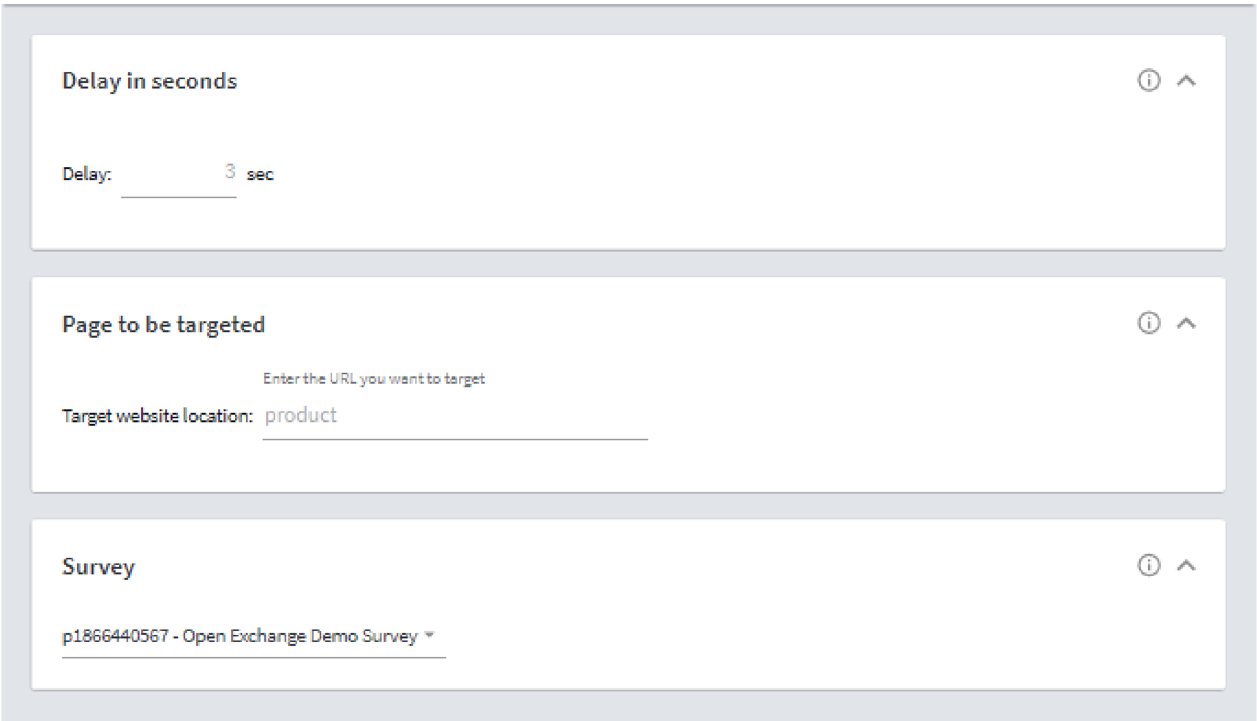
delaySeconds

targetPage

survey

Figure 39 Example of the script in a template referencing the parameters

The non-scripter can then open the scenario to set the time delay between when the visitor opens the web page and when the invitation is displayed, to set the page that the visitor must open to trigger the invitation, and to set the survey that will be opened if the visitor agrees to respond



Delay in seconds

Delay: 3 sec

Page to be targeted

Target website location: product

Survey

p1866440567 - Open Exchange Demo Survey

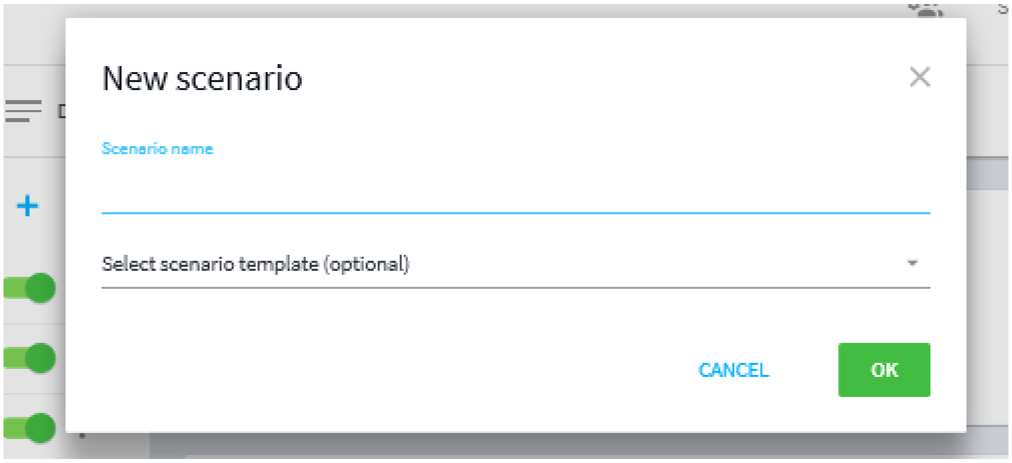
Figure 40 The scenario as the non-scripter will see it

3.5.3. Creating a Scenario Based on a Template

Once you have a template, you can then create a scenario based on that template.

1. Click + beside the Scenarios header.

   The New scenario overlay opens, now including a template selection field.



New scenario

Scenario name

Select scenario template (optional)

Figure 41 The New scenario overlay with the template selection field

2. Type in a name for your scenario.
3. Click into the Select scenario template field and select the template you wish to use.

Note that using a template is optional. You do not need to select one; you can just create a "normal" template (see Adding Scenarios on page 11 for more information).

4. Click **OK**.

The scenario opens with the layout based on the template you have selected rather than the scripting field.

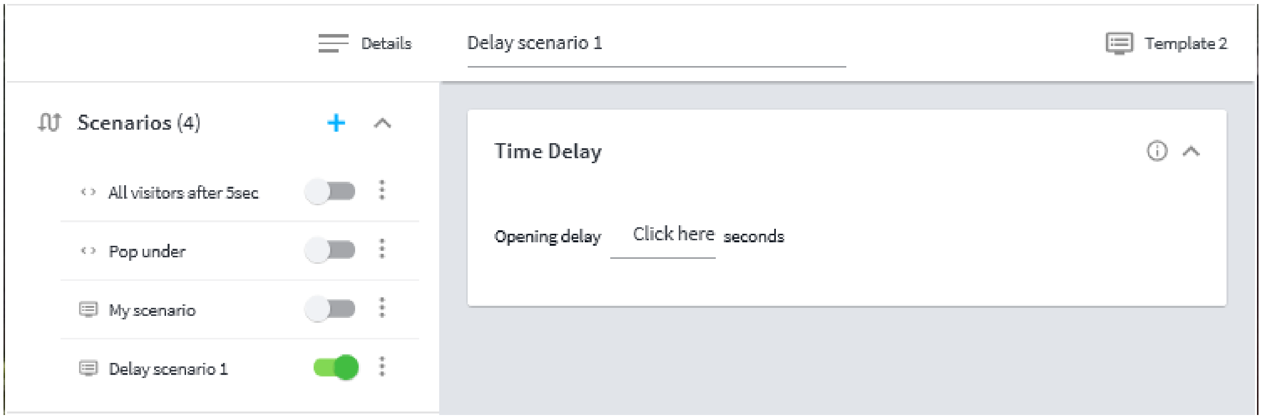


Figure 42 The scenario based on the template

Now, if a user wants to adjust the time delay between when the visitor opens the web page and when the invitation opens, he/she can click into the number field (in this example denoted "Click here") and type in a numerical value. Note that the template can be as complex as necessary, and can include scripting and any number of input forms.

You can edit the name of the scenario by clicking into the name field at the top of the page, in this example Delay scenario 1, and if you wish to edit the template you can click the template icon in the upper-right corner, in this example Template 2.

3.6. Throttling the Number of Completes in a Timeframe

You can throttle, or limit, the number of completed surveys that are allowed by a program in a specific time-frame. This allows you to control the through-put of completed interviews for a particular survey in the program. Note that a counter applies to a specified survey, so if for example your program has three surveys that will open under different circumstances, then each survey can have its own counters that can have different limits.

The time-frames you can set limits for are hour, day or week. For example for one survey you could set a limit of maximum 50 completes per hour, with 500 per day and 2000 per week, and for another survey in the same program you could set a limit of maximum 100 completes per hour, with 1000 per day and no upper limit for a week. Once the limit is reached for a time-frame, the invitation (if used) or the survey will not be displayed again until the start of the next time-frame. Note that the counters register completed surveys, so if several visitors are responding to a survey simultaneously then you can exceed the counter limits.

The counters are based on the time used by the server that is running your Digital Feedback application, with the new week starting on the Monday.

Note that you can set up the counters using scripting or by creating a template (see Creating a Template on page 23 for more information).

Use the code below to reference the counters in the script:

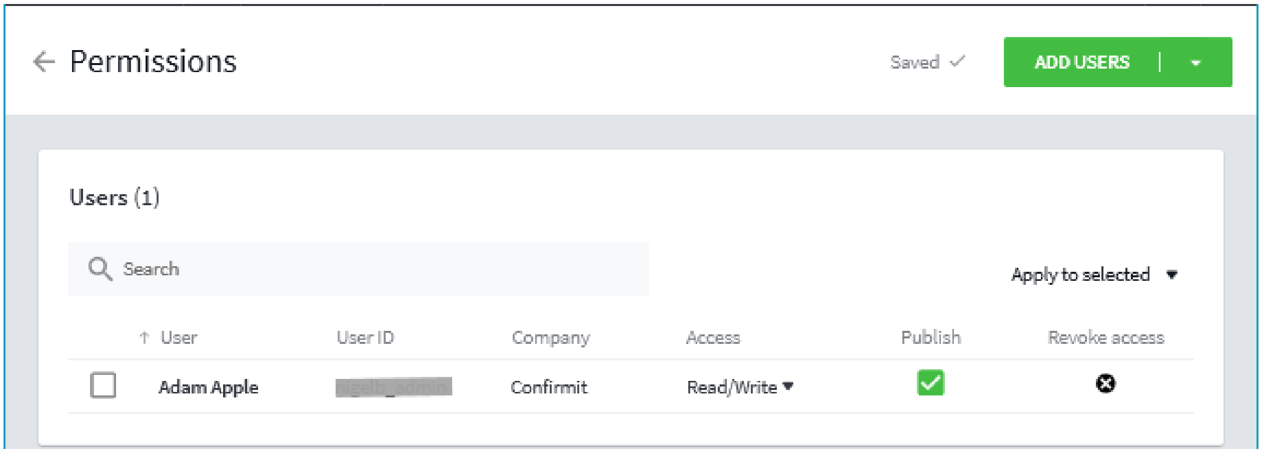
```
api().scenarioCounters('<projectID>').completesCurrentHour
api().scenarioCounters('<projectID>').completesCurrentDay
api().scenarioCounters('<projectID>').completesCurrentWeek
```

3.7. Setting Access Permissions

By default, you as the program owner have full control over the program. However you may want others to be able to view and maybe edit and publish your program. To allow other users access to your program:

1. Click the **Permissions** icon towards the right end of the Program toolbar.

   The Permissions page opens listing all the users who currently have an access permission for this program.



	↑ User	User ID	Company	Access	Publish	Revoke access
□	Adam Apple	[redacted]	Confirmit	Read/Write ▾	☑	☒

Figure 43 Example of the Permissions page

In the event the user list is extensive you can search for the desired user. Start typing characters into the **Search** field; the list is filtered as you type to present only those users whose user name or user id starts with the typed characters. If the user name comprises multiple words, then each word is checked.

2. To add a user to the list, click **Add Users**.

   By default, a list of the users in your company opens.

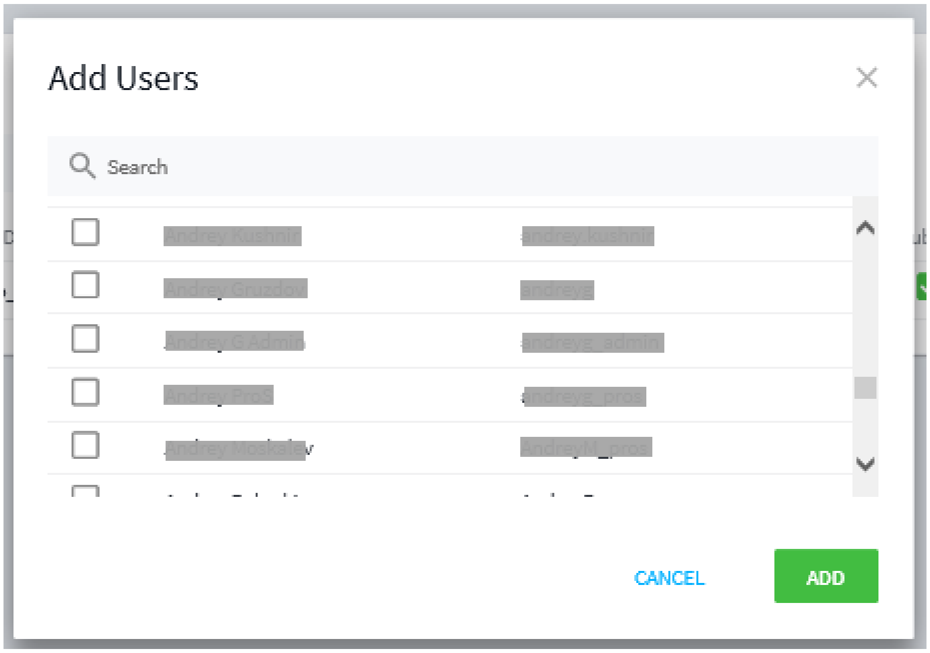


Figure 44 Example of the Add Users overlay

3. Select the user or users you wish to allow access to your program, then click **Add**.

   Again, if the user list is extensive you can search for the desired user.

   The overlay closes and you are returned to the Permissions page. The selected users are added to the list with the default access of Read only, and you can now change this access and/or provide Publish permissions as required. You can apply permissions individually to specific users, or if you have a number of users who require the same permissions you can perform a bulk application. Note that only users in the list will be able to access the program; all other users will not see the program listed in their Program list.

To change the permission for a specific user:

1. Click the down-arrow beside the current permission in the Access column to open the drop-down list.

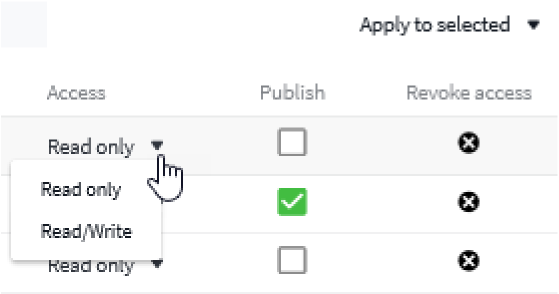


Figure 45 Changing the access permission for a user

2. Click on the desired permission to select it.

   The drop-down list closes and the selected permission is applied to that user.

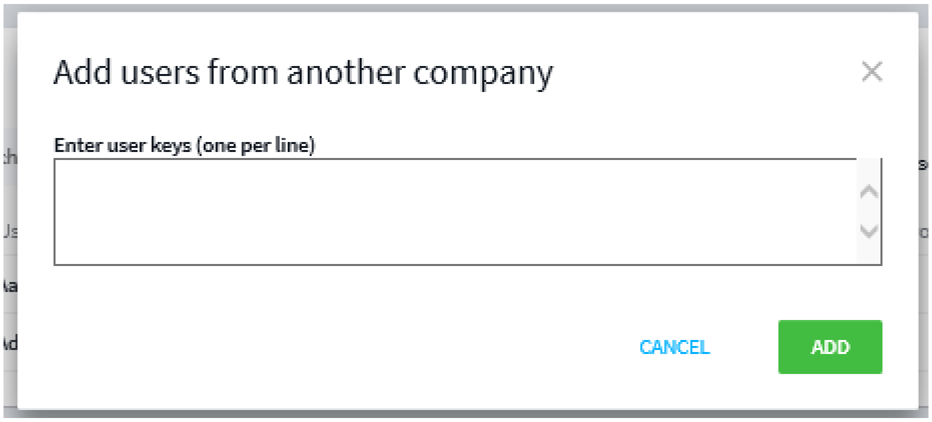
- To allow a user to publish the program, check the relevant box in the Publish column.
- To revoke all access to the program for a user, remove the user from the list by clicking the **X** button.

- To change the access permissions for a number of users simultaneously, select the users by clicking in the boxes at the left end of the applicable rows, then click **Apply to selected** and select the desired permission.

If you want to add users who are registered with a different company:

1. Click the down-arrow beside the **Add users** button and select **Add users from another company**.

   The Add users... overlay opens.



Add users from another company

Enter user keys (one per line)

Figure 46 The Add users from another company overlay

2. Copy the user key for the user into the field and click **Add**.

   To add multiple users, copy their keys into the field onto separate lines.

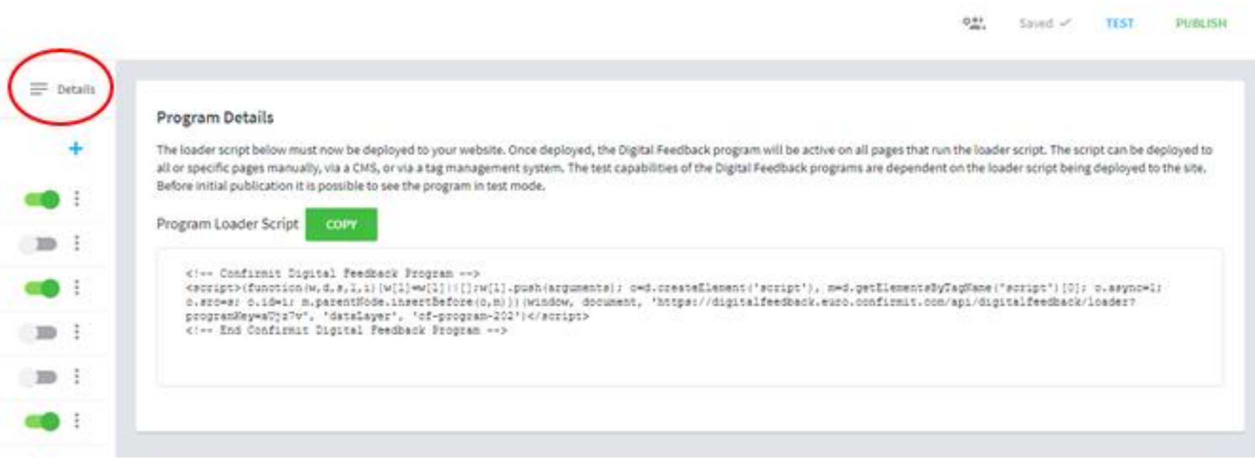
4. Deploying and Testing the Program

Before any testing can be performed, your webmaster must place the snippet of code on your website. This can be done using a code management system or via a tag manager. Once the code is deployed your program can be tested using your actual website.

To deploy your program

1. Once you have created your program, click **Details** (ringed below) to open the Program Details page.

   This page contains the details and code required to link your program to the website.



Program Details

The loader script below must now be deployed to your website. Once deployed, the Digital Feedback program will be active on all pages that run the loader script. The script can be deployed to all or specific pages manually, via a CMS, or via a tag management system. The test capabilities of the Digital Feedback programs are dependent on the loader script being deployed to the site. Before initial publication it is possible to see the program in test mode.

Program Loader Script COPY

```
<!-- Confirmit Digital Feedback Program -->
<script>(function(w,d,s,l,i){w[l]=w[l]||[];w[l].push(arguments); o=d.createElement('script'), m=d.getElementsByTagName('script')[0]; o.async=1;
o.src=s; o.id=i; m.parentNode.insertBefore(o,m)})(window, document, 'https://digitalfeedback.euro.confirmit.com/api/digitalfeedback/loader?
programKey=aUjz7v', 'dataLayer', 'cf-program-202')</script>
<!-- End Confirmit Digital Feedback Program -->
```

Figure 47 Example of the Program Details page

2. Click the green **Copy** button to copy the Program Loader Script snippet to your clipboard (you may need to confirm this action).
3. Forward this snippet to your webmaster for deployment to your website.

To test your program

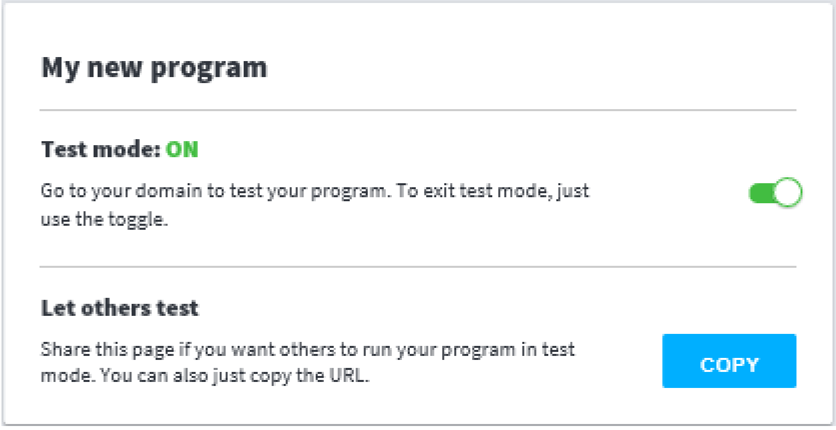
Testing is an important part of the process for creating a new Digital Feedback program. There are several key features for testing programs using Digital Feedback:

- The ability to test before initial publication.
- The ability to test changes made before they are published.
- The ability to allow other, non-Professional users to test programs or changes to programs before they are published.

If any changes are later made to the program, you must re-publish the program to make those changes visible to the users - unpublished changes will not be visible to the users.

The Test mode applies only to the current session in your browser.

To run the program in Test Mode, click **Test** towards the right end of the Program toolbar. The Test dialog opens.



My new program

Test mode: ON

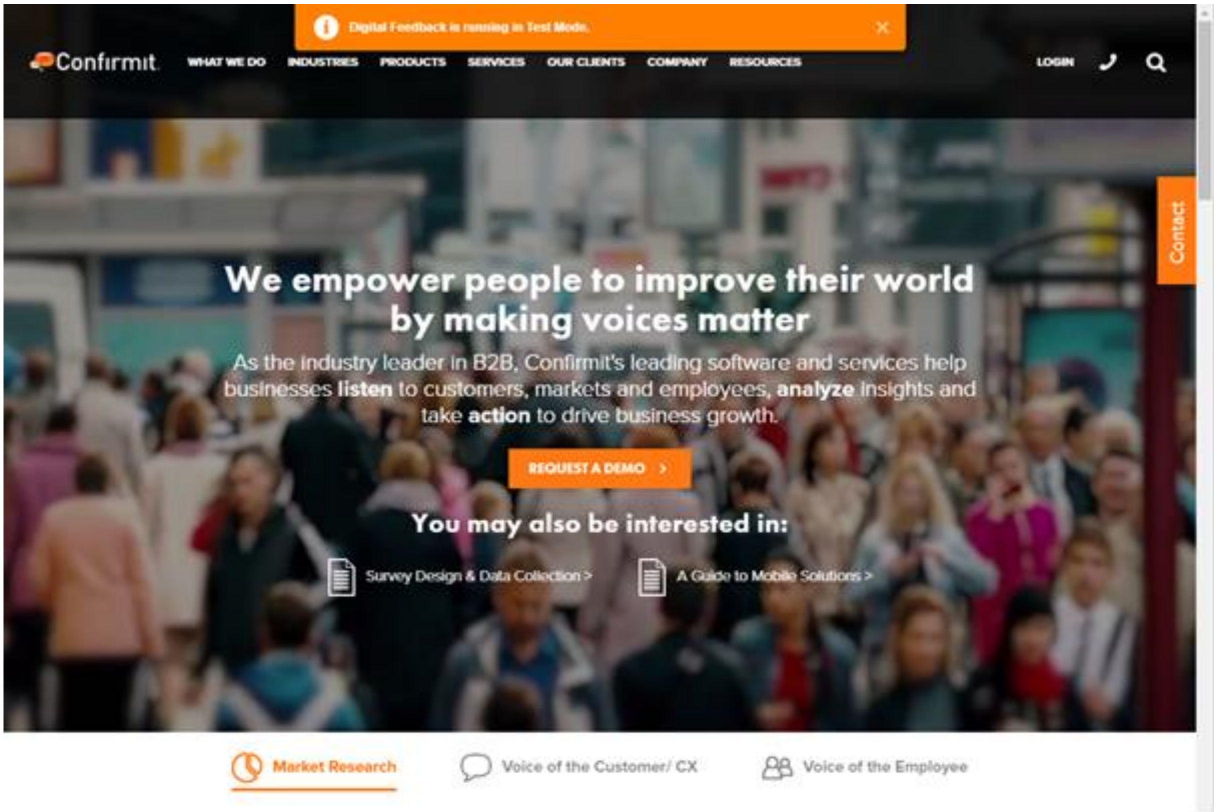
Go to your domain to test your program. To exit test mode, just use the toggle.

Let others test

Share this page if you want others to run your program in test mode. You can also just copy the URL.

Figure 48 The Test dialog

While the Test mode is toggled on, when you go to the website where the code snippet is deployed you will see the website and program along with any un-published changes. A tab is displayed across the top of the page to remind you that the program is in Test mode.



Digital Feedback is running in Test Mode.

We empower people to improve their world by making voices matter

As the industry leader in B2B, Confirmit's leading software and services help businesses **listen** to customers, markets and employees, **analyze** insights and take **action** to drive business growth.

You may also be interested in:

Survey Design & Data Collection >
A Guide to Mobile Solutions >
Market Research
Voice of the Customer/ CX
Voice of the Employee

Figure 49 Example of a website with a program in Test mode

To switch off the test mode, toggle the switch in the dialog to **Off**, or close your browser and load a new browser.

You can allow others to view your program in test mode. To do this, copy the URL (click **Copy** in the dialog), paste the URL into an email and send it to the users. When the user applies the URL to their browser, the page they are presented with will be the website with the program in Test mode. This will remain available to them during the current browser session. To switch off the test they only need to close the browser.

5. Editing the Markup

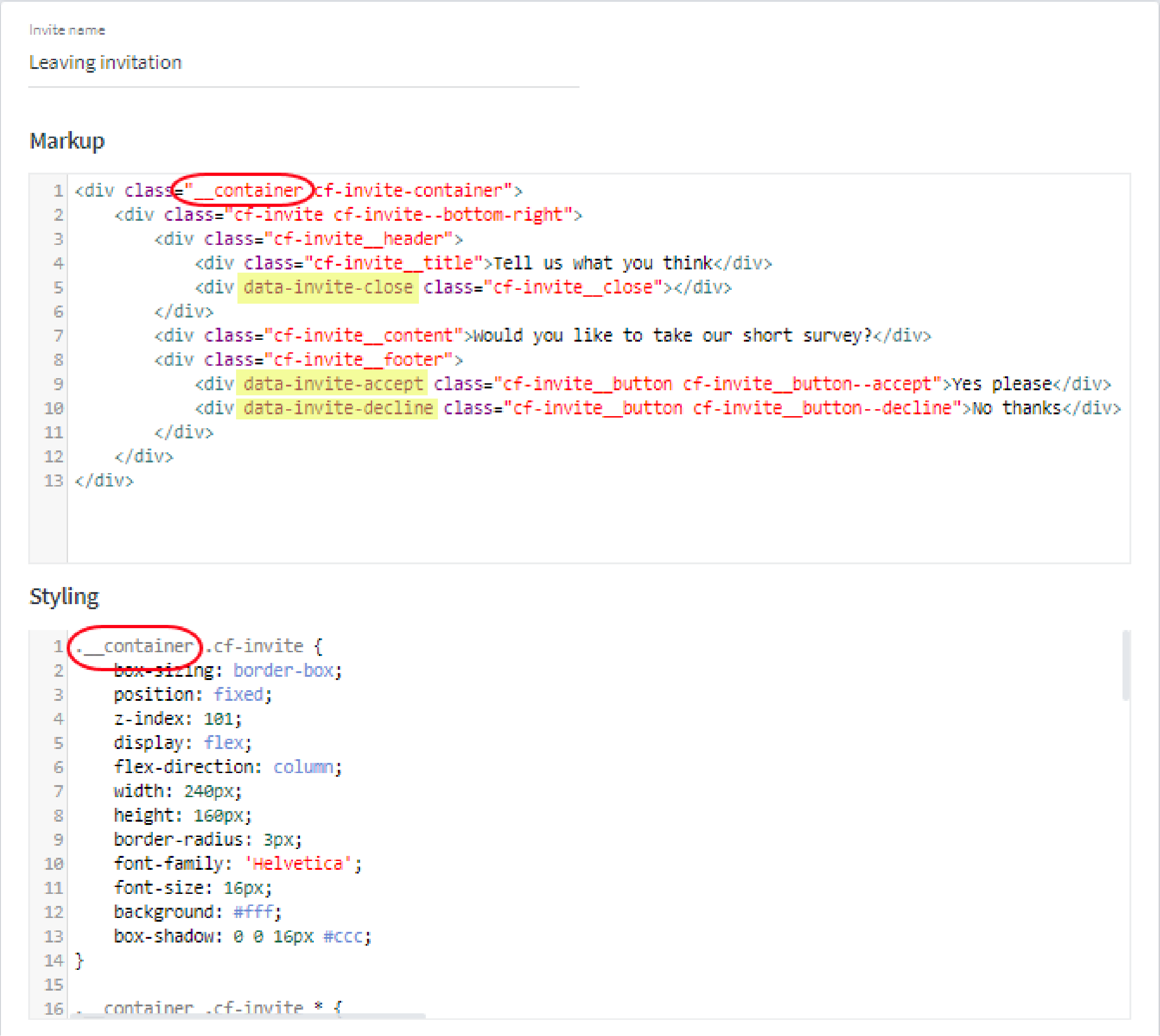
You can edit the markup of any of the scenarios, overlays and invites as necessary. You can copy code from another application and paste it into the desired field, you can select and delete code, or click into the field and type it as required. The script in the fields is color-coded to simplify editing.

5.1. The Invitation Markup

An invitation will normally include some presentation parts (colors, texts, styles etc) and some functional parts (buttons).

The look and feel of an invitation is heavily dependent on styles, and since multiple invitations could appear on the same web page, the invitations' styles must be isolated from each other so they do not collide. Additionally the Digital Feedback styles must be isolated from the target website’s styles, again to avoid collisions. To achieve the isolation, a special “__container” CSS class name is supported, which can be added to the top level container and then used as a prefix for every CSS style. This “__container” will be replaced in runtime with a unique identifier that ensures styles from different invitations don’t overlap.

However this container is not mandatory; a user can ignore this convention and not use “__container”. In this case the application will still work, but there might be collisions between the styles of different invitations, and also the host site’s styles. So using “__container” is highly recommended.



Invite name

Leaving invitation

Markup

```
<div class="__container cf-invite-container">
    <div class="cf-invite cf-invite--bottom-right">
        <div class="cf-invite__header">
            <div class="cf-invite__title">Tell us what you think</div>
            <div data-invite-close class="cf-invite__close"></div>
        </div>
        <div class="cf-invite__content">Would you like to take our short survey?</div>
        <div class="cf-invite__footer">
            <div data-invite-accept class="cf-invite__button cf-invite__button--accept">Yes please</div>
            <div data-invite-decline class="cf-invite__button cf-invite__button--decline">No thanks</div>
        </div>
    </div>
</div>
```

Styling

```
.__container.cf-invite {
    box-sizing: border-box;
    position: fixed;
    z-index: 101;
    display: flex;
    flex-direction: column;
    width: 240px;
    height: 160px;
    border-radius: 3px;
    font-family: 'Helvetica';
    font-size: 16px;
    background: #fff;
    box-shadow: 0 0 16px #ccc;
}

.__container .cf-invite * {
```

Figure 50 Example of the Invitation markup

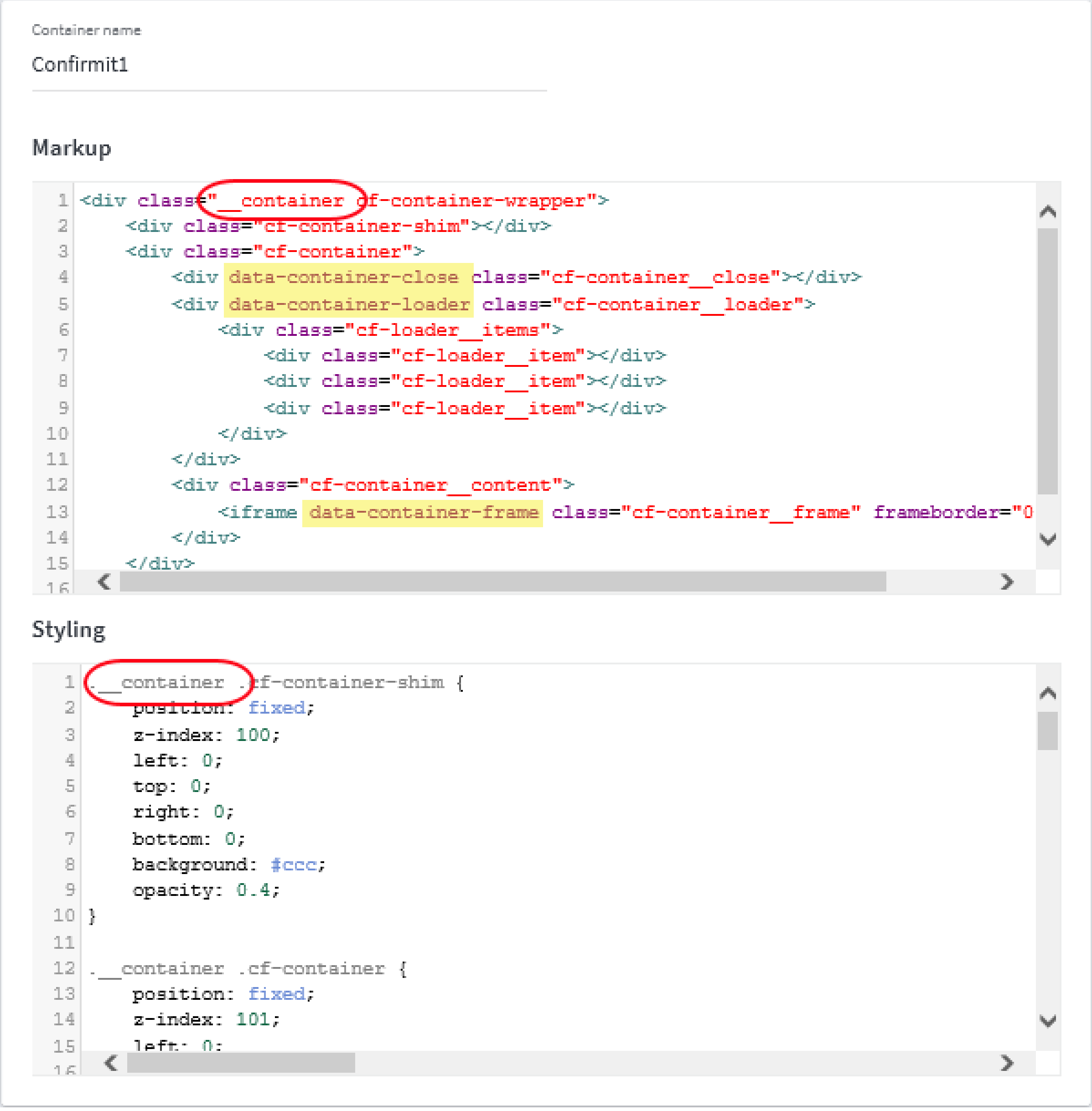
The behavior of invitations (mostly button click handlers) is controlled by special data attributes. The code is constructed so the user can totally rewrite the HTML and CSS; there are no “magical” IDs or CSS classes. If you want an element to behave as an accept-button, simply add the conventional data attribute “data-invite-accept” to it. There are currently four such attributes supported:

- data-invite-accept - the invitation closes and the survey is run.
- data-invite-decline - the invitation closes and the survey is not run.
- data-invite-close - the invitation closes, the survey is not run, but the invitation can re-open later if the conditions are correct.
- data-invite-persistent - will cause the invitation to remain visible after it is accepted. This behavior is useful for certain scenarios such as a feedback button that you want to be permanently visible.

5.2. The Container Markup

The survey Container markup functions in the same way as the Invitation (see The Invitation Markup on page 36 for more information). It uses a container, and it has some data-attributes. The attributes are:

- data-container-close – self-explanatory
- data-container-loader – this element displays loading animation while the survey is loading.
- data-container-frame – this is the iframe element within which the survey is loaded.



Container name

Confirmit1

Markup

```
<div class="__container cf-container-wrapper">
    <div class="cf-container-shim"></div>
    <div class="cf-container">
        <div data-container-close class="cf-container__close"></div>
        <div data-container-loader class="cf-container__loader">
            <div class="cf-loader__items">
                <div class="cf-loader__item"></div>
                <div class="cf-loader__item"></div>
                <div class="cf-loader__item"></div>
            </div>
        </div>
        <div class="cf-container__content">
            <iframe data-container-frame class="cf-container__frame" frameborder="0
        </div>
    </div>
```

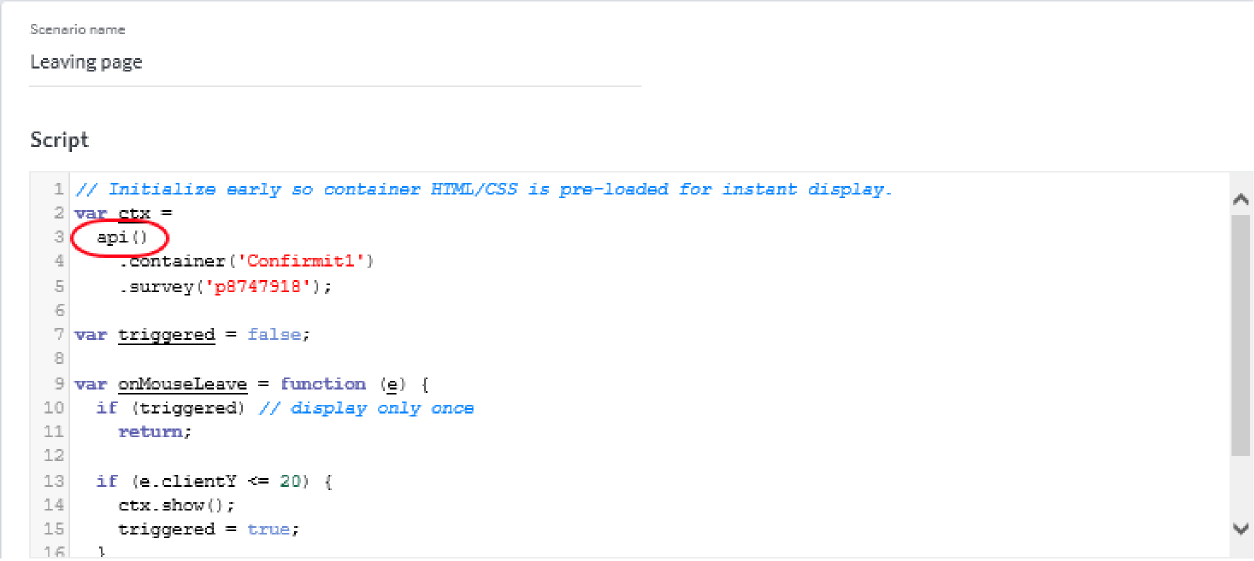
Styling

```
.__container .cf-container-shim {
    position: fixed;
    z-index: 100;
    left: 0;
    top: 0;
    right: 0;
    bottom: 0;
    background: #ccc;
    opacity: 0.4;
}

.__container .cf-container {
    position: fixed;
    z-index: 101;
    left: 0;
```

Figure 51 Example of the Overlay markup

5.3. The Scenario Script

The scenario script is a normal JavaScript code which is executed in the browser on a target site. All standard JavaScript can be used. For Digital Feedback, a special “api” function is made available.



Scenario name

Leaving page

Script

```
1  // Initialize early so container HTML/CSS is pre-loaded for instant display.
2  var ctx =
3    api()
4      .container('Confirmit1')
5      .survey('p8747918');
6
7  var triggered = false;
8
9  var onMouseLeave = function (e) {
10   if (triggered) // display only once
11     return;
12
13   if (e.clientY <= 20) {
14     ctx.show();
15     triggered = true;
16   }
```

Figure 52 Example of the Scenario script

API methods are used to select the invitation and overlay (by name) and the survey (by surveyId). If desired you can also pass the site’s data to a survey, and then show the survey configured as required (see API Details on page 39 for more information).

API calls can be chained as:

```
api().invite(‘invitename’).overlay(‘overlayname’).survey(‘surveyid’)
```

or invoked separately as:

```
var ctx = api()
ctx.invite(‘invitename’);
ctx.overlay(‘overlayname’);
ctx.survey(‘surveyid’);
```

Functionally this is the same; chaining is just for convenience. The order of the calls does not matter, except that the “show” method (see line 14 in the image above) would usually appear last because this triggers the actual display.

To link the desired invitation, overlay and survey into the scenario you merely copy the relevant item's name into the appropriate location ('_______') in the scenario. Note that JavaScript logic can be used in scenarios to specify a different invitation / container / survey combination as desired.

5.3.1. API Details

The entry point is the “api” function which, when called, returns the api instance. The api instance has the following methods:

- invite
- container
- survey
- data
- show

Each method configures the api instance (with the invite name, survey id etc. specified), and returns the instance, so methods could be chained, for example api().invite(‘my invite’).survey(‘pxxx’) etc. The ‘show’ method on the api instance triggers the actual display.

The api instance also has the “events” property, which can be used to take actions after specific scenario stages. The following events are currently available:

- 'showInvite’
- ‘acceptInvite’
- ‘declineInvite’
- ‘closeInvite’
- ‘showContainer’
- ‘completeSurvey’
- ‘closeContainer’

Some examples using events are provided below the scenario script editor.

This configuration contains no mandatory parts; you can have an invite without a container, a container without a survey etc. and there are no predefined restrictions. Also, the order does not matter, so “api().invite(‘i1’).container(‘o1’)“ is the same as “api().container(‘o1’).invite(‘i1’)”.

The api function can be called multiple times to create multiple separate api instances within the same scenario. For example, the 'show' function can be called several times on the same api instance to execute the same behavior repeatedly.

The scenario script can access all global JavaScript objects on the page and DOM. For example, if someone wants to access objects scoped in the scenario script (such as the api) from external code, they need to expose the objects to the external code explicitly once the scenario script is loaded. However this is not recommended.

6. Troubleshooting

This section lists some problems that may be encountered, and the possible reasons behind those problems,

Site set-up note:

- When deploying the Digital Feedback script code on your web site, the website must be configured to allow insertion of external JavaScripts. Without this the program will not be effective.

Why can’t other users see the programs?

- By default, only the program creator and company administrators have access.
- If any other users need access, you must add explicit permission to view, edit and/or publish the program (see Setting Access Permissions on page 30 for more information).

When I go to the site I can’t see my latest changes, why?

- Have they been published? Until they are published they will only be viewable in test mode.

The program doesn’t seem to be running at all. Why is this?

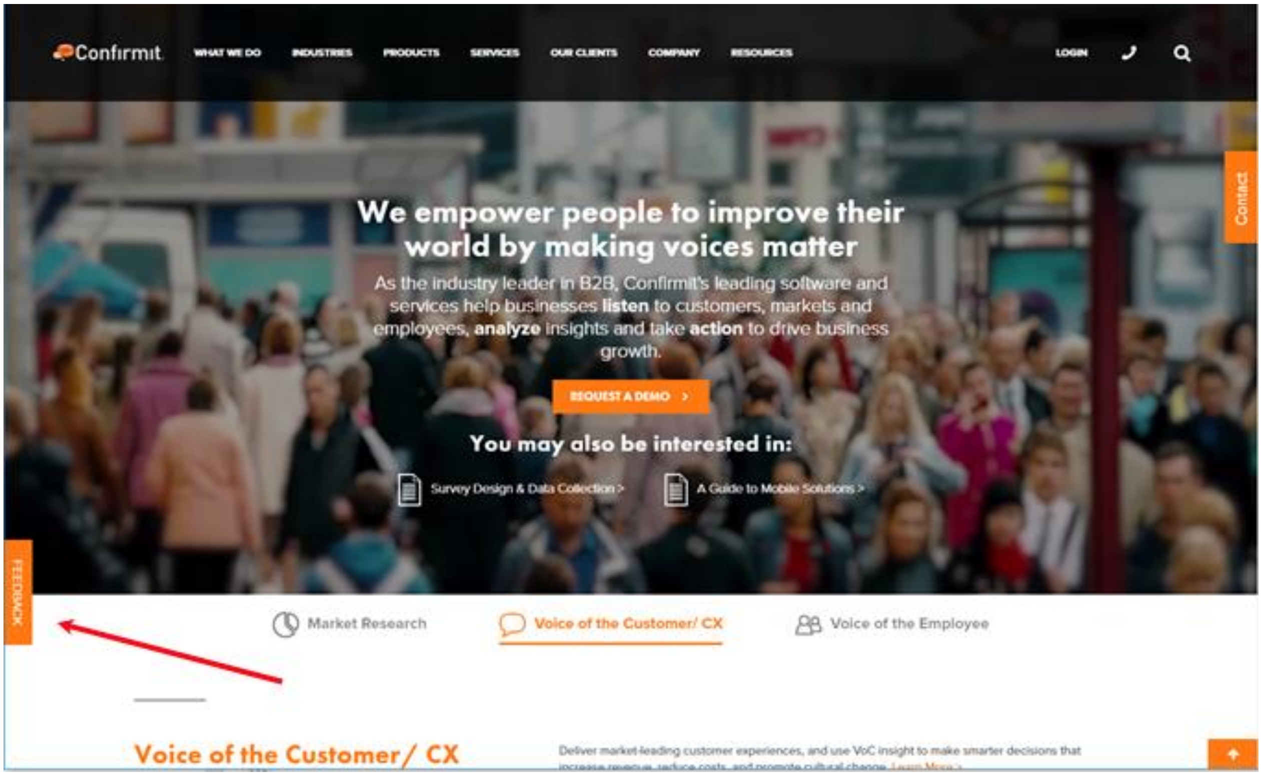
- Has the snipet of code been deployed to your site?
- Does your site allow the insertion of external JavaScript’s?
- As the program relies on running the client-side JavaScript, if there are any errors in the script the program will not function. You will need to troubleshoot any script errors. This can be done using the browser dev tools (press F12 from the browser).

7. Appendix A: Code Examples

This appendix contains code examples to assist you with achieving specific results. Copy the code and paste it into the appropriate field.

7.1. Invite Markup and Styling for Feedback Button

This code creates a **Feedback** button that will appear towards the lower-left corner of the web page, rotated at 90 degrees so it covers as little of the web page as possible.



We empower people to improve their world by making voices matter

As the industry leader in B2B, Confirmit's leading software and services help businesses **listen** to customers, markets and employees, **analyze** insights and take **action** to drive business growth.

You may also be interested in:

Deliver market-leading customer experiences, and use VoC insight to make smarter decisions that

Figure 53 Example of a Feedback button

The button will be visible on every page in the website, and will remain visible after it has been clicked so the user can provide feedback multiple times.

Scenario

Copy the code below into the Scenario field. Note that you must change the invite, container and survey names as appropriate.

```
var fn = function () {
api()
.invite('My feedback button')
.container('My overlay')
.survey('p1862987068')
.show();
};
setTimeout(fn, 1*1000);
```

Markup

Copy the code below into the Markup field:

```
<div data-invite-persistent class="__container cf-prompt-container">
<div class="cf-survey-prompt cf-survey-prompt--bottom-left">
<button data-invite-accept class="cf-survey-prompt--
accept">FEEDBACK</button>
</div>
</div>
```

Styling

Copy the code below into the Styling field:

```
.__container .cf-survey-prompt {
box-sizing: border-box;
transform: rotate(90deg) translateY(123px);
transform-origin: left bottom;
}

.__container .cf-survey-prompt {
position: fixed;
z-index: 101;
font: 14px Arial;
background: rgb(246, 247, 248);
border-radius: 2px;
}

.__container .cf-survey-prompt--title,
.__container .cf-survey-prompt--text,
.__container .cf-survey-prompt--buttons {
padding: 10px 20px;
}

.__container .cf-survey-prompt--title {
background: #ff7813;
font-weight: bold;
color: #fff;
}

.__container .cf-survey-prompt--accept:hover {
```

```
color: rgb(230, 230, 230);
}

.__container .cf-survey-prompt--accept,
.__container .cf-survey-prompt--decline {
-webkit-appearance: none;
border: none;
display: inline-block;
border-radius: 2px;
color: white;
text-align: center;
padding: 4px 23px;
line-height: normal;
height: 34px;
cursor: pointer;
font-family: helvetica, arial;
font-size: 14px;
}

.__container .cf-survey-prompt--accept {
background-color: #ff7813;
border: 1px solid #ff7813;
}

.__container .cf-survey-prompt--decline {
background-color: #ff7813;
border-bottom: 1px solid #ff7813;
}

.__container .cf-survey-prompt--close {
position: absolute;
top: 0;
width: 24px;
height: 24px;
cursor: pointer;
}

.__container .cf-survey-prompt--close circle {
fill: #fff;
stroke:#ff7813;
```

```
stroke-width:2px;
stroke-linecap: round;
}



.__container .cf-survey-prompt--close path {
stroke-width:2px;
stroke-linecap: round;
fill:none;
stroke: #ff7813;
}

.__container .cf-survey-prompt--bottom-left {
left: 120px;
top: 50%;
}

.__container .cf-survey-prompt--top-right {
top: 20px;
right: 20px;
}

.__container .cf-survey-prompt--top-left {
top: 20px;
left: 20px;
}

.__container .cf-survey-prompt--centered .cf-survey-prompt--close,
.__container .cf-survey-prompt--top-left .cf-survey-prompt--close,
.__container .cf-survey-prompt--bottom-left .cf-survey-prompt--close {
left: auto;
right: -30px;
}

.__container .cf-survey-prompt--top-right .cf-survey-prompt--close,
.__container .cf-survey-prompt--bottom-right .cf-survey-prompt--close {
left: -30px;
right: auto;
}
```

```
.__container .cf-survey-prompt--centered {
left: 0;
right: 0;
top: 0;
bottom: 0;
width: 305px;
height: 130px;
margin: auto;
}
```

7.2. Delay with Nth Visitor

This code shows the survey to every 5th visitor after they have visited 3 pages in the website, with a delay of 10 seconds between them opening the 5th page and the survey opening. Some data, in this case the userName and the jobTitle, is also passed to the survey.

```
/* Example: show after 3 pages, to every 5th visitor, with 10 sec delay
and pass some data to the survey. */
// Very basic visited page tracking.
var pageCount = +sessionStorage.getItem('my_visited_page_count');
sessionStorage.setItem('my_visited_page_count', ++pageCount);

// Very basic Nth visitor check.
var is5thVisitor = Math.random() < 1 / 5;
var ok = pageCount > 3 && is5thVisitor;
if (!ok) {
return;
}

// Create and configure api instance
var ctx = api()
.invite('My invite')
.container('My container')
.survey('p0000001')
.data({
userName: 'Peter Parker',
jobTitle: 'Spider-Man'
});

// Show with delay
setTimeout(function() { ctx.show(); }, 10 * 1000);
```

8. Appendix B: User Experience Summary

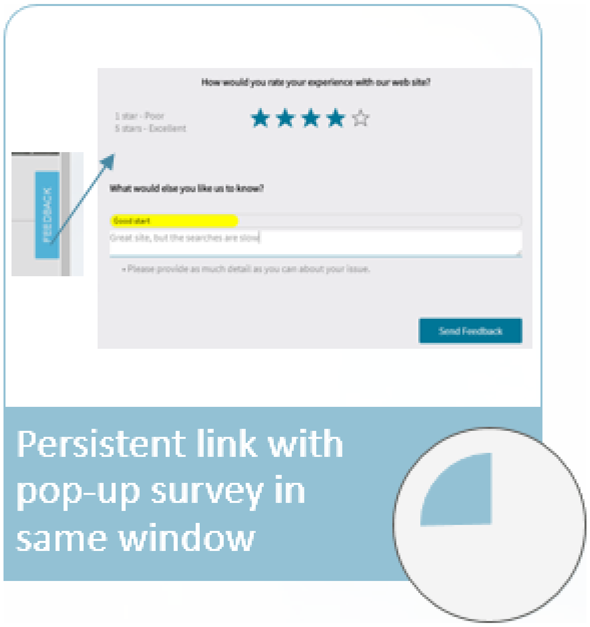
Scenarios or Triggers (see Adding Scenarios on page 11 for more information) are used to control when an invitation or survey pop-up is to appear. Common triggers include:

- Time spent on site (regardless of page).
- Time spent on a specific page.
- Intent to leave a specific page.
- The viewer has visited x number of pages.
- The viewer has completed a specific transaction on a page.

Note: If no triggers are used with the options listed below, the invitation or survey pop-up will appear immediately the viewer opens the web page. This is not recommended as the viewer has no options, will feel "ambushed", and will probably just leave your website.

The segmented circle in the lower-right corner of each image below indicates the level of intrusiveness (and thereby irritation) to the user for the method.

1. A link that is always visible but is located "out of the way", for example on the edge of the web page, is the least intrusive method. The viewer can click the link if and when he/she wants to reply to the survey. In this case, the link itself is the "invitation", so no other invitation is used.



How would you rate your experience with our web site?

1 star - Poor
5 stars - Excellent

★★★★☆

What would else you like us to know?

Good start

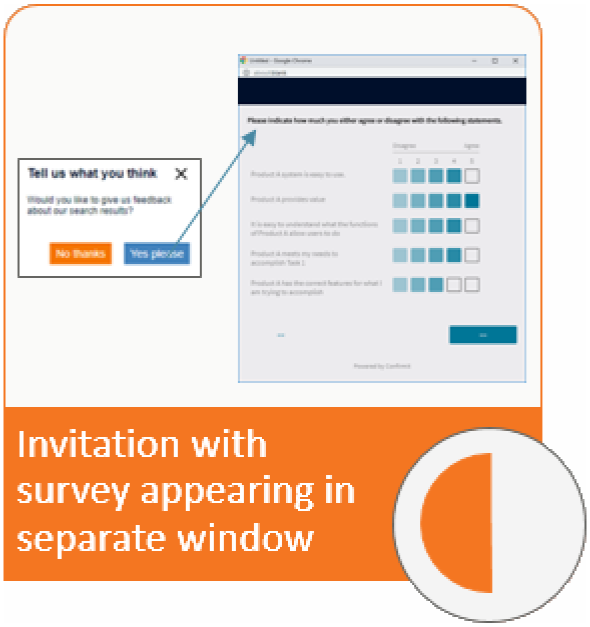
Great site, but the searches are slow

• Please provide as much detail as you can about your issue.

Persistent link with pop-up survey in same window

Figure 54 Persistent link

2. An invitation that appears after a predefined set of circumstances (triggers) are met, that allows the viewer to open the survey in a separate window if he/she wants to respond, is next on the list. If the user does not want to respond to the survey they can close the invitation, and if the user does respond they can then switch between the web page and the survey, giving them "freedom to decide".



Tell us what you think

Would you like to give us feedback about our search results?

Please indicate how much you either agree or disagree with the following statements.

Product A system is easy to use.

Product A provides value

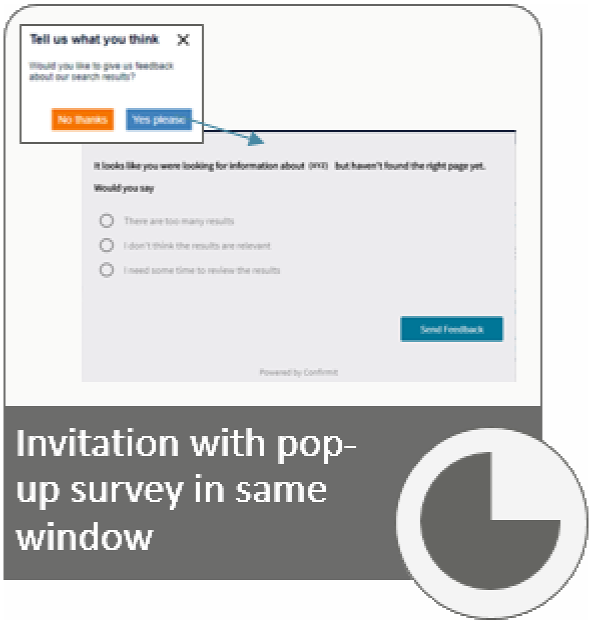
It is easy to understand what the functions of Product A allow users to do

Product A meets my needs to accomplish Task 1

Product A has the correct features for what I am trying to accomplish

Figure 55 Invitation then separate window

3. An invitation that appears after a predefined set of circumstances (triggers) are met, that opens the survey in the same window if he/she wants to respond, is getting more intrusive. If the user does not want to respond to the survey they can close the invitation, but if the user does agree to respond they are then "locked" into the survey until they have completed it.



Tell us what you think

Would you like to give us feedback about our search results?

It looks like you were looking for information about (XYZ) but haven't found the right page yet.

Would you say

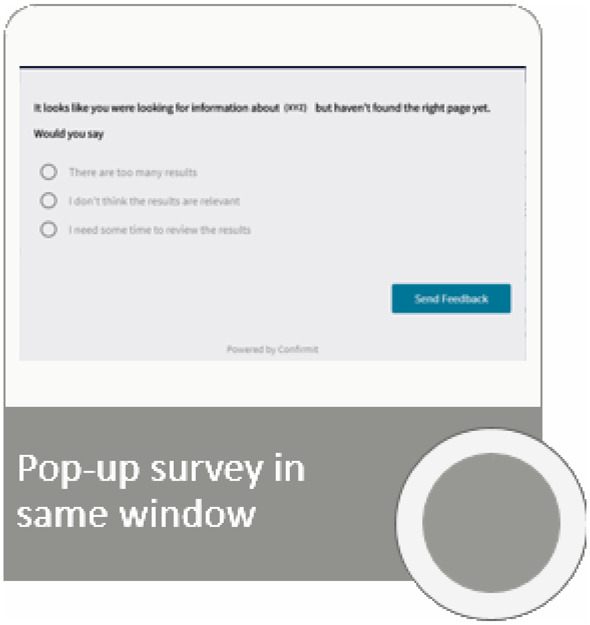
○ There are too many results

○ I don't think the results are relevant

○ I need some time to review the results

Figure 56 Invitation then same window

4. A survey that merely appears and takes over the web page window after a predefined set of circumstances (triggers) are met, without giving the viewer a choice, is most intrusive. In this case if the user does not wish to respond to your survey then they have no option except to leave your web page.



It looks like you were looking for information about (XYZ) but haven't found the right page yet.

Would you say

○ There are too many results

○ I don't think the results are relevant

○ I need some time to review the results

Figure 57 No invitation, same window

Index