

Verint Digital Experience Comment Card API POST

This document details how to replicate an existing comment card POST from a native environment. The comment card must exist and be active in the Digital Experience Portal before replicating it natively.

Overview

What is a native comment card?

A native comment card is one hosted created natively in the environment in which it is collecting feedback. It is not hosted or served from the secure.opinionlab.com domain.

When should a native comment card be used?

The most prolific use case for a native comment card would be within a mobile app. However, any environment where you want to host the comment card may make the native comment card a viable option.

What are the benefits of a native comment card?

A native comment card gives you full control over the user experience as opposed to a hybrid App which frames a Verint web comment card in a webview. This allows you to control the following:

- Comment card trigger
- Comment card rendering
 - Styles
 - Layout
- Comment card validation
- Post-submission message
- Post-submission experience/behavior

If the above items are priorities for your user experience team, then a native comment card is most likely the best solution.

How to Accomplish

Per the instructions below, the configurations are completed on the client side with assistance from your Verint team.

Step 1:

Create a comment card in the Digital Experience Portal under the domain/segment where you want to collect the data.

Step 2:

Add all ratings, questions and other custom fields to the comment card that you'd like to use in your native comment card. It should be an exact replication of the form fields you plan to use natively all the way down to the answer types.

Note: It is not necessary to add a banner or apply any custom styling to this comment card, as the end user will never see it.

Step 3:

Activate the comment card as a **static activation**. Once you have statically activated the comment card, copy the direct link and paste it in your browser of choice to load the card.

Step 4:

Within the browser’s developer tools, open the Network tab. ***DO NOT*** view the source code or attempt to submit from the preview mode in the Digital Experience Portal.

Step 5:

Fill out the comment card and submit your answers. Locate rate40.asp and find the form data being submitted via POST.

This will give you several of the dynamic elements needed to populate the POST. Here are the parameters defined:

Field Name	Required?	Data Description/Value	Example
ccid	required	The comment card id; use the value from your sample POST	
time1	optional	Value represents initial timestamp when user entered “page”. Must be less than time2. Expressed in the format of an epoch (Unix) timestamp	1617747680929 (Corresponds to Tuesday, April 6, 2021 10:21:20.929 PM GMT)
time2	optional	Value represents timestamp when user triggered comment card. Must be greater than time1. Expressed in the format of an epoch (Unix) timestamp	1617747682204 (Corresponds to Tuesday, April 6, 2021 10:21:22.204 PM GMT)
prev	required	Full URL (including protocol) that represents previous “page” before the “page” where the user triggered the comment card. If this is not called from the web, set it to “direct_call”	https://www.company.com/cart or direct_call
referer	required	Full URL (including protocol) of the trigger “page” where the data	https://app.company.com/checkout

		<p>will be recorded in the portal. Note, this is <i>not</i> the document.referrer, which is more closely akin to the prev element, above.</p> <p>If using a fake domain (e.g. "app.company.com") use it here.</p>	
currentURL	optional	<p>The actual URL (including protocol) of the "page" the user was on. Recorded as Actual URL in the portal</p> <p>If excluded, the system will use the referer value as the Actual URL</p>	https://www.company.com/checkout
width	optional	Integer value of user's screen width; can be 0 if unknown	1024
height	optional	Integer value of user's screen height; can be 0 if unknown	768
comment_card	required	1	1
thank_you	required	1 or null; 1 will include thank you message HTML content in the form response	1 or null
ip_address_optout	optional	1 or null; 1 will not capture the IP address	1 or null
custom_var	optional	<p>Legacy custom variables; up to 1000 characters. Separate multiple values with vertical pipes ()</p> <p>We recommend using the V5 customVars below unless you have a specific need for legacy variables</p>	12345 gold 1.0.1
customVars	optional	V5 custom variables. Pass as string key/value pairs (see below)	{clientId: '12345', campaign: 'campaign_x', appVersion: '1.0.1'}



comments	optional	Comment field; up to 1000 characters	
overall	required	1, 2, 3, 4, 5	
question<1,2,3,4,5,6,7,8,9,10,11,12>	optional	The unique identifier for each question on the comment card; use values from your sample POST	Examples: question1: 34343 question2: 34344
answer_<INTEGER>	optional	Integer in name should be same as question <i>value</i> . This value should correspond to user response; up to 100 characters in length	Examples: answer_34343: Make a purchase answer_34344: Yes

Any other parameters can be safely ignored.

Step 5:

Construct your native comment card with field names that replicate the exact names from the comment card view source. The native comment card action should POST to <https://gen3.opinionlab.com/v1/response>.

Form data should be URL-encoded.

Include the following header: Content-Type: application/x-www-form-urlencoded

Step 6:

Submit test feedback and confirm that the data was properly captured in the Digital Experience Portal. Once data has been confirmed in the Digital Experience Portal, your native comment card is ready to release.

Additional Information

V5 Custom Variables

Determine which custom variables you will pass in with the feedback, and how they will be named. Variables an app version (for native app feedback), a user or client identifier, or any other data that would be used for segmenting, prioritizing or acting on the feedback.

New variables can be added at any time by simply including them in the URL; there is no need to preconfigure variables in the Verint Digital Experience system.

Assume the following variables are used:

Variable name	Variable value
---------------	----------------



clientId	12345
campaign	campaign_x
appVersion	1.0.1

Variables are written in the format:

```
{variable_name1: 'variable value 1', variable_name2: 'variable value 2', etc.}
```

For this example:

```
{clientId: '12345', campaign: 'campaign_x', appVersion: '1.0.1'}
```

Do not include commas in variable names or values. Place the value in the customVars parameter of the POST. Remember to URL-encode the form data

Keeping Comment Cards in Sync

Your native comment card must always remain in perfect alignment with its web counterpart in the Digital Experience Portal. Any changes to questions or activation in the Digital Experience Portal will require an update to your native comment card and a new release of your App.

If flexibility is a concern for your team, consider framing the Verint mobile web comment card into your app as a webview. All customization capabilities will then be maintained.

Whitelisting

If you need to whitelist domains for your native mobile app to access them, whitelist “gen3.opinionlab.com”.

Verint. Powering Actionable Intelligence.®

Verint® is a global leader in Actionable Intelligence® solutions with a focus on customer engagement optimization and cyber intelligence. Today, more than 10,000 organizations in 180 countries—including over 80 percent of the Fortune 100—count on Verint solutions to make more informed, effective, and timely decisions.

