Code & Tutorials

Toggle your local video and participants bar in Daily's prebuilt UI

Customize your prebuilt call UI with Daily's latest methods and properties



Jess Mitchell

Published: 29 Jan 2021 (Updated: 1 Nov 2021) • 6 min read

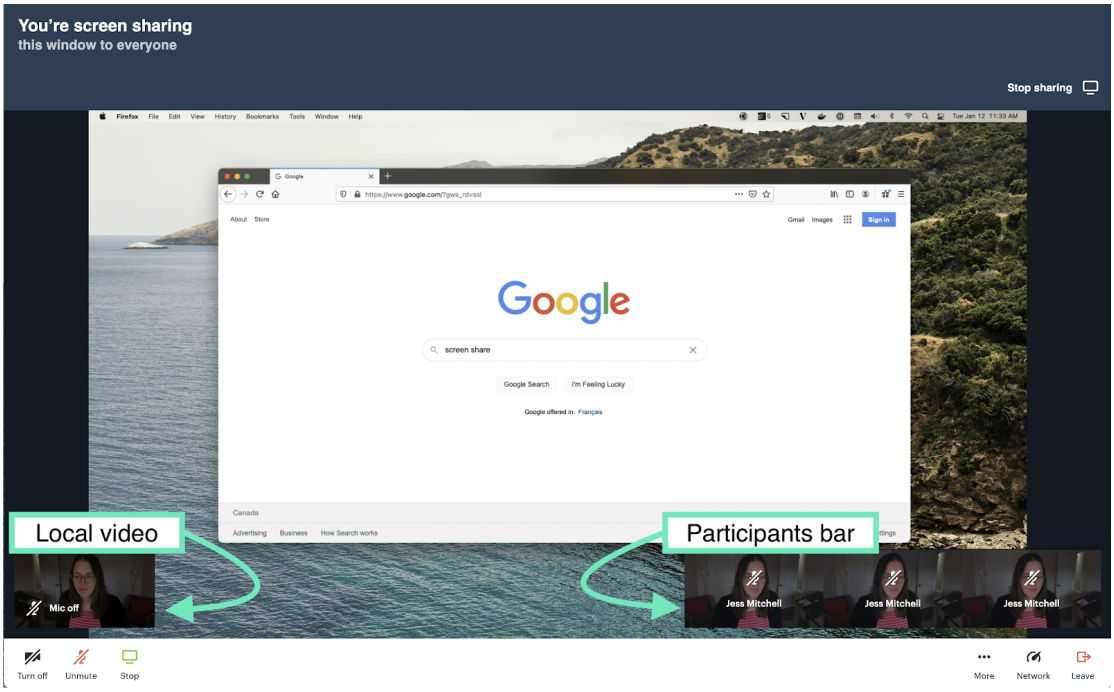


Daily's prebuilt UI offers one of the fastest ways to add embedded video calls to your app. Compared to the Daily call object, which lets you completely customize a video call from scratch, the prebuilt UI comes with an out-of-the-box interface that still incorporates all of Daily's video call functionality. By using the prebuilt UI instead of call object mode

you ultimately sacrifice some customization, but you can save yourself quite a bit of time.

One aspect of video calls that Daily has been focused on is accommodating use cases that have recently become more popular. One such use case even came up internally for our customer experience team: hosting effective webinars.

Our prebuilt UI seemed like the obvious decision for building a webinar app since we want customers to get familiar with Daily's default appearance. There was one issue, however; we knew only the presenter's video and microphone would be on, which meant we didn't need to show the local video or participants bar. Both of these UI elements show by default and can't be hidden in the prebuilt UI. Until now.



Daily's prebuilt UI with the local video and participants bar showing

TL;DR: Daily's new properties and instance methods

If you prefer to cut to the chase, check out our new properties and instance methods in our docs.

Properties:

- `showLocalVideo`
- `showParticipantsBar`

Instance methods (setters)

- `setShowLocalVideo(boolean)`
- `setShowParticipantsBar(boolean)`

Instance methods (getters)

- `showLocalVideo()`
- `showParticipantsBar()`

Introducing showLocalVideo and showParticipantsBar ✨

To help customers and Daily's own team quickly build a webinar app, we are now offering two new properties and two new sets of instance methods.
To see how these new features work, let's take a quick look at how to get started with the DailyIframe class.

Calling all properties

Adding the embedded prebuilt UI to your app can be done a few ways. Let's start with our current prebuilt demo code, which uses the factory method `createFrame()` to initialize the DailyIframe class. It accepts two parameters: the parent element to embed the iframe in and an

object for customizing the Daily iframe that will be rendered. This object

```
callFrame = await window.DailyIframe.createFrame(
 document.getElementById('callframe'),
 {
   iframeStyle: {
     position: 'absolute',
     top: '0',
     left: '0',
```

```
      width: '100%',
      height: '90%',
      border: '0',
    },
  }
);
```

Initializing this call with no local video shown is as simple as specifying our new property: `showLocalVideo`. This value defaults to `true`, so if you don't specify it in the properties the local video will be shown.

```
callFrame = await window.DailyIframe.createFrame(
 document.getElementById('callframe'),
 {
   iframeStyle: {
     // custom styles
   },
   showLocalVideo: false,
 }
);
```

Similarly, if you want to hide the participants bar, you can specify this feature with our new property `showParticipantsBar`. Let's add it to the previous code example to see how it looks when both are added:

```
callFrame = await window.DailyIframe.createFrame(
 document.getElementById('callframe'),
 {
   iframeStyle: {
     // custom styles
   },
   showLocalVideo: false,

   showParticipantsBar: false,
 }
);
```

These two new properties are independent of each other and can be added individually or together depending on your use case. For our team and many of our customers who are building a webinar app, it makes sense to hide both UI elements. However, you may only need to

hide one! We'll leave that to you to decide.

createFrame, but make it React

In case you're more comfortable with React, here's a simplified version of a video call component to see how to include these properties.

```
import DailyIframe from "@daily-co/daily-js";

const CALL_OPTIONS = {
 iframeStyle: {
   // add styles here
 },
 showLocalVideo: false,
 showParticipantsBar: false,
};


const VideoCall = () => {
   const videoRef = useRef(null);

   useEffect(() => {
     if (!videoRef || !videoRef?.current) return;
     CALL_OPTIONS.url = 'https://your-team.daily.co/room-name';

       const callFrame = DailyIframe.createFrame(
         videoRef.current,
         CALL_OPTIONS
       );

       callFrame.join();
   }, [videoRef]);


   return <div ref={videoRef} />;
};
```

Similarly to the first code block example, the parent element and Daily properties are passed to the `DailyIframe` factory method, `createFrame()`.

Wrapping up Daily's factory method properties

You may have noticed there are two other factory methods available for

the prebuilt UI: `wrap()` and `createTransparentFrame()`. Both of these will accept the same properties as `createFrame()`, so you can set `showLocalVideo` and `showParticipantsBar` the same way as demonstrated above.

Hiding in plain sight

In the React code block example above, you may also have noticed we call `callFrame.join()` once the DailyIframe class has been initialized. `join()` — a `DailyIframe` instance method— will also accept the same properties object. This means you can set `showLocalVideo` and `showParticipantsBar` in `join()` instead of in the factory method you're using.

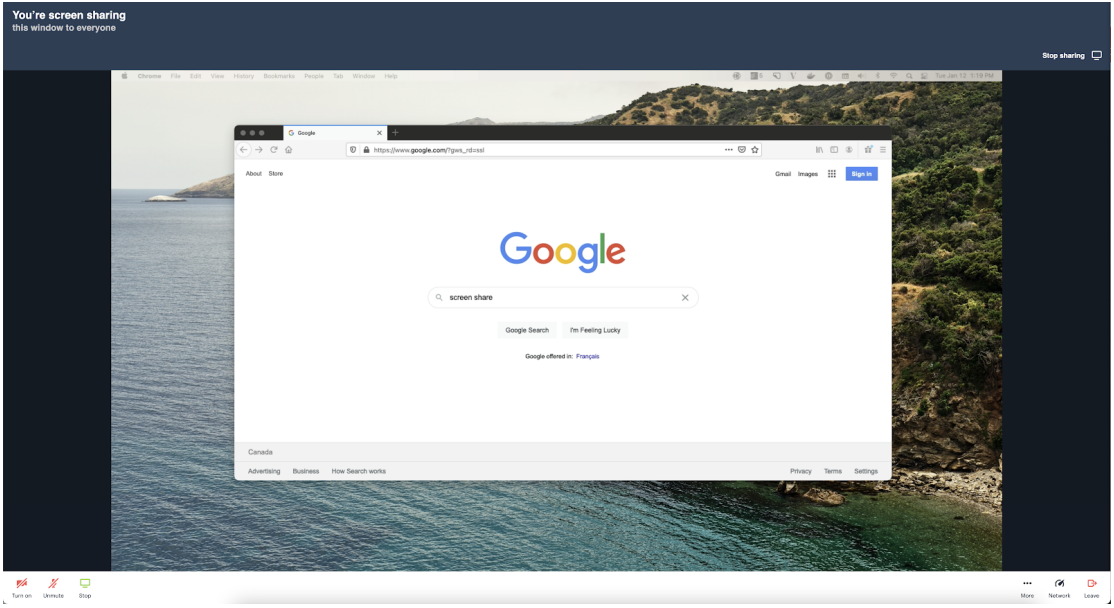
The only downside to be aware of when setting `showLocalVideo` and `showParticipantsBar` in `join()` is that you may notice the local video and participants bar flash into view before being hidden. This flash represents the time it takes in between the iframe being created and the join method being called.

Going back to the prebuilt demo code, adding these properties in the `join()` method would look like this:

```
async function joinCall() {
 const roomURL = document.getElementById('room-url');
 await callFrame.join({
   url: roomURL.value,
   showLeaveButton: true,
   showLocalVideo: false,
   showParticipantsBar: false,
 });
}
```

Seeing our end result

The end result will be the same prebuilt video call experience as before but without the local video or participants bar rendered.



Daily's prebuilt UI with the local video and participants bar excluded

To show or not to show: Daily's newest instance methods

While considering use cases for hiding the local video and participants bar, we decided there may be some applications where these two UI elements don't need to always be visible or hidden.

For example, we've heard of users wanting the participants bar to only be hidden specifically when a screen share is happening to help focus attention on the presentation. In this case, you'd need to be able to dynamically hide and show the participants bar.

There are likely many more cases where dynamically setting the display would be preferable. To keep things flexible, we decided to let our users be in charge of when to hide and show the local video and participants bar.

To accommodate this, we've added two new sets of instance methods.

Toggling the local video display with setShowLocalVideo()

For anyone who would like to choose when to show or hide the local video, we've added `setShowLocalVideo()` to our instance methods. It accepts one parameter: a boolean value (`true` or `false`).

Going back to our demo code, we can create a couple new methods that can each be called on an event (e.g. a button click).

```
function hideLocalVideo() {
   callFrame.setShowLocalVideo(false);
}

function showLocalVideo() {
   callFrame.setShowLocalVideo(true);
}
```

With all this toggling, it can be difficult to keep track of what the current state is programmatically. To help stay organized, we've added a "getter" method to the DailyIframe class instance to tell you the current state: `showLocalVideo()`.

To toggle the local video, you could, for example, set it to the opposite of the current state value.

```
function toggleLocalVideo() {
   const currentlyShown = callFrame.showLocalVideo();
   callFrame.setShowLocalVideo(!currentlyShown);
}
```

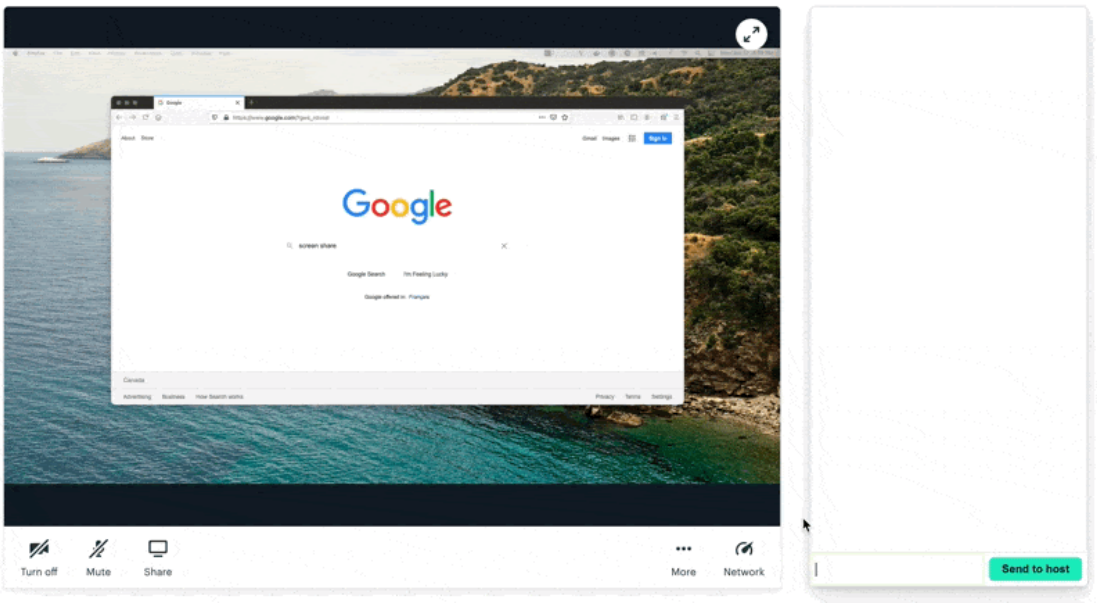
Participation points: Dynamically set the participants bar display

Toggling the participant bar follows the exact same structure as the local video instance methods. To set the display for the participants bar, use `setShowParticipantsBar()`. To get the current setting, use `showParticipantsBar()`.

```
function toggleParticipantsBar() {
   const currentlyShown = callFrame.showParticipantsBar();
   callFrame.setShowParticipantsBar(!currentlyShown);
}
```

Putting these new instance methods into practice in the example below,

we can hide or show the local video and participants bar depending on the string submitted in the chat box. It may be a silly example, but it shows how much customization you can now accomplish. The rest is up to you!



Example of toggling the local video and participants bar display

Introducing Daily's new webinar blog series: Build a webinar app with Daily

At Daily, we've been hearing more interest in creating webinar platforms to help companies provide information to their users faster. Webinar formats can vary but, generally speaking, they allow for one (or more) participants to act as the presenter, while the other participants join to watch the presentation.

As webinars become even more popular, Daily wants to help you get up

and running as fast as possible. We've created a new webinar blog series that will cover:

- Learning how to structure a webinar app in React
- Handling account types and differentiating between admins and participants with Daily meeting tokens
- Building a chat with direct and broadcast messages
- Managing admin access with an admin app

And more! By the end you'll have a fully functional webinar app.

Keep an eye on our blog for our webinar series coming out soon. In the meantime, let us know if we can help you get started!

Get updates from Daily

Don't miss a thing! Subscribe to our monthly newsletter and we'll keep you up-to-date with the latest product releases, tutorials, and more from Daily.

Email*

Country*

Join peerConnection - our video/WebRTC developer community.

By clicking submit below, you consent to allow Daily.co to store and process the personal information submitted above to provide you the content requested.

Submit

More in **Code & Tutorials**

Working with video call participants' media tracks

10 Oct 2022 - 9 min read

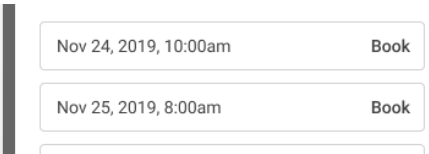
Stream with HLS while recording to a custom S3 bucket using Daily

30 Aug 2022 - 5 min read

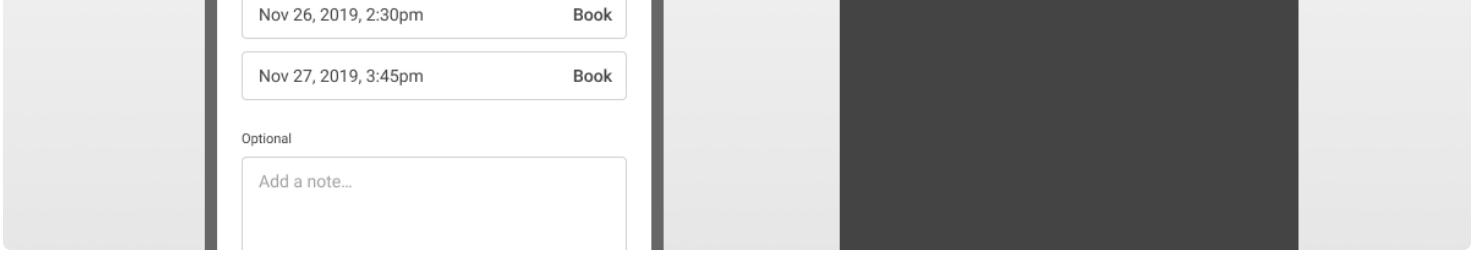
Tutorial: Integrate CometChat's chat widget into a Daily video app for an enriched chat experience

22 Aug 2022 - 13 min read

See all 99 posts →





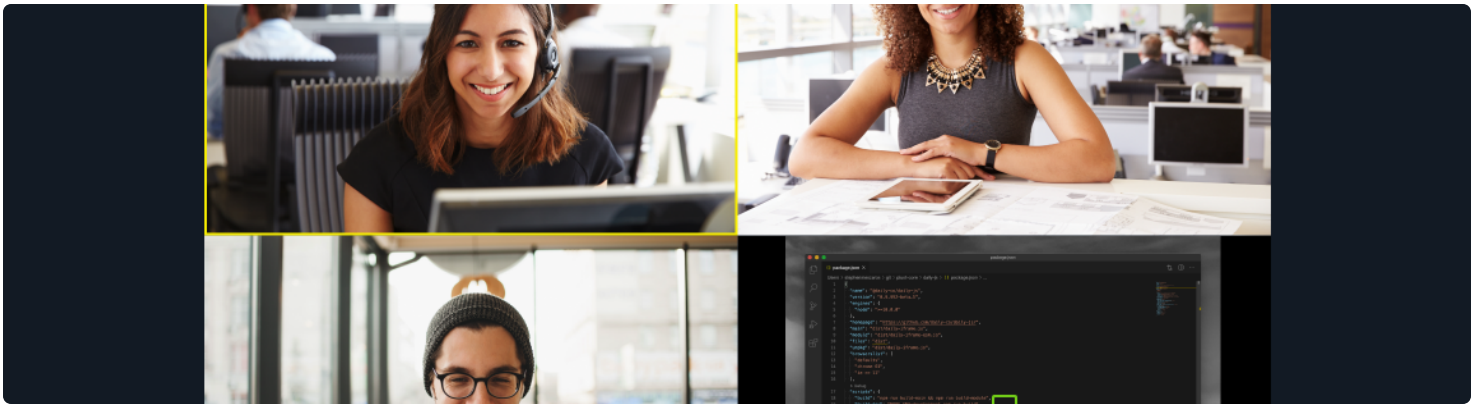


Why native in-app video chat trumps video conferencing every single time

The Daily API makes it easy to add video calling to any site or app — meaning you never have to send customers to an external video call platform again.



Francine Boulanger
5 Feb 2021 – 6 min read



Use Cases

Video as an essential part of a B2B customer engagement strategy

Where video fits into the customer funnel



Francine Boulanger
28 Jan 2021 – 5 min read