



WATCH ▾

ABOUT ▾



SUPPORT

[GUIDE](#)[HELP](#)[ABOUT](#)[CONTACT US](#)[FAQ](#)

Embedding a Video


This video is unavailable.

Play Pause

Status: Unknown.

See the *Video Events* section down below for more information on how we listen/update the status above.

Embed Snippet

To get the embed code for a video, visit the video's page, and look for the share icon, , and click on Embed. The embed code will be copied to your browser's clipboard. You can now use this code anywhere you wish to embed to the video.

Below is an example of the code for an embedded video:

```
<iframe src="https://samsungvr.com/watch/75PSPOMrMmz?autoplay=true&st=01:30" frameborder="0" allowfullscreen width="640" height="360"></iframe>
```

Embed Options

Param	Type	Description
autoplay	URL Param	When present, this value tells the embedded video to autoplay (where available).
st	URL Param	The starting time. This query parameter tells the video what time to start playing the video. The format is <code>hh:mm:ss</code> , but you can omit sections if there are not applicable. For example, <code>01:30</code> indicates 1 minute and 30 seconds, and <code>30</code> indicates 30 seconds.
width	Iframe Attribute	How wide the video should be.
height	Iframe Attribute	How tall the video should be.

Video Events

Samsung XR is ending service for its XR apps on September ... [MORE INFO \(/PORTAL/WHATSNEW/A_MESSAGE_FROM_SAMSUNGXR\)](#) ✕

The following `message` events are available:

SAMSUNG XR **WATCH** **ABOUT** ✓ The video has finished playing.



- `samsungvr_message_playing` - The video is playing.
- `samsungvr_message_paused` - The video has been paused.
- `samsungvr_message_video_ready` - The video is ready to play.

The following snippet is an example for how to listen for the video events:

```

window.addEventListener('message', function(e) {
  // Video has ended.
  if (e && e.data === 'samsungvr_message_ended') {
    // Do something...
  }

  // Video has started/resumed.
  if (e && e.data === 'samsungvr_message_playing') {
    // Do something...
  }

  // Video is paused.
  if (e && e.data === 'samsungvr_message_paused') {
    // Do something...
  }

  // Video is ready to play.
  if (e && e.data === 'samsungvr_message_video_ready') {
    // Do something...
  }
}, false);

```

Triggering Play/Pause Through JavaScript (Advanced)

In the embedded video at the top of this page, there are two buttons, one for `Play`, and one for `Pause`. These buttons exist *outside* the embed iframe, and are an example of how one would trigger a `play` or `pause` command to the iframe (as detailed in the snippets below).

There are 2 `message` events that can trigger `play` and `pause` into an embedded iframe:

- `samsungvr_message_play` - Triggers "play" on the video.
- `samsungvr_message_pause` - Triggers "pause" on the video.

The `message` strings above must be preceded with the video ID of the video the action should be performed on (and joined with a colon `:`). So, when posting a `play` command, your string might look like this: `jkIez0nxM0o:samsungvr_message_play`, where `jkIez0nxM0o` is the ID of the video. Passing the ID lets the embedded video know what video to perform the action on. This is useful if you have *multiple* embeds on one page. In the case of a page with multiple embeds, you don't want a user passing "play" to all embedded videos at once, since this could cause confusion, as well as potentially crashing the user's browser (**playing multiple video streams at once require lots of bandwidth and memory!**).

In order to trigger a `play` or `pause` event from the parent to the embed (iframe), you will need a get a reference to the `<iframe>` element, and you will need to extract the source URL, and the video ID.

Here's a simple example:

```

iframeWindow.postMessage('75PSPOMrMmz:samsungvr_message_play', 'https://samsungvr.com');

```

The above assumes that you are triggering `play` on a video with the ID of `75PSPOMrMmz`, and that `iframeWindow` is a reference to the iframe's `contentWindow` property.

The following example illustrates, in more detail, one way you can do this (Note: since browser support varies, you might want to adjust the code so that it works in your application).

```

// (1) Get a reference to the iframe, and iframe's window element
var iframe = document.querySelector('iframe');
var iframeWindow = iframe.contentWindow;

// (2) Parse the Video ID from the iFrame
var parser = document.createElement('a');
parser.href = iframe.src;
var videoID = parser.pathname.split('/').pop();
var samsungTarget = parser.origin;

// (3) Attach events to trigger play INTO the iframe.
var playButton = document.querySelector('#play');
playButton.addEventListener('click', function() {
  var message = [videoID, 'samsungvr_message_play'].join(':');
  iframeWindow.postMessage(message, samsungTarget);
});

// (4) Attach events to trigger pause INTO the iframe.
var pauseButton = document.querySelector('#pause');
pauseButton.addEventListener('click', function() {
  var message = [videoID, 'samsungvr_message_pause'].join(':');
  iframeWindow.postMessage(message, samsungTarget);
});

```

Notes on the example code above:

1. We need to get a reference to the DOM element for the iframe. You can do this by traversing the DOM for an iframe element, or by attaching an ID to the iframe and selecting the iframe by ID.
2. We need both the Samsung XR URL, and the video ID. The URL is important for security reasons, since helps to prevent any malicious scripts from posting messages. The ID is necessary so that the embedded player can determine if the request (play or pause) is for this particular video or not.
3. We attach an event to trigger `play` on the click of a button with an id of "play".
4. We attach an event to trigger `pause` on the click of a button with an id of "pause".

EXPERIENCE IN VIRTUAL REALITY

(<https://www.microsoft.com/store/apps/9N7J19GTLQDG>) (<https://www.oculus.com/experiences/gear-360/>) (<https://www.oculus.com/experiences/quest/2346574095401736/>)

(<https://www.oculus.com/experiences/quest/2346574095401736/>) (<https://www.oculus.com/experiences/quest/2346574095401736/>) (<https://www.oculus.com/experiences/quest/2346574095401736/>)

ALSO ON MOBILE

(<http://apps.samsung.com/appquery/appDetail.as?appId=com.samsung.android.video360>) (<https://play.google.com/store/apps/details?id=com.samsung.android.video360>)

SUPPORT (PORTAL/CONTENT/EOS_FAQ)

FAQ ([HTTPS://HELP.CONTENT.SAMSUNG.COM/CSWEB/AUTH/GOSUPPORT.DO?SERVICECD=SAMUNGVR&CHNLCD=WEB&TARGETURL=/FAQ/SEARCHFAQ.DO&COMMON_LANG=EN](https://help.content.samsung.com/csweb/auth/gosupport.do?SERVICECD=SAMUNGVR&CHNLCD=WEB&TARGETURL=/FAQ/SEARCHFAQ.DO&COMMON_LANG=EN))

© 2020 Samsung Electronics Co., Ltd. All rights reserved. | [Terms of Service \(portal/content/tos\)](#) | [Privacy Policy \(https://account.samsung.com/membership/pp\)](#)

[Copyright Policy \(portal/content/copyright_policy\)](#)