BML Interactive Video Integration - iFrame

Pre-requisites

- **BML_url** - Your BML interactive video link

Process

- The integration is really simple. Simply use the *BML_url* as the src-attribute value for the iFrame

```
<iframe id="why-us-video" src="<BML_url>" frameborder="0" allow="autoplay; fullscree
```

iFrames and the CORS-policy

Cross-Origin Resource Sharing (CORS) is a security policy, that handles the sharing of external ressources on a given website, for instance externally hosted JavaScript libraries, but also iFrames. If this policy is not followed in a correct manner, it can happen, that the iFrame is blocked by the respective browser. When integrating Saferpay into an iFrame, please always consider this policy.

Size of the iFrame

The iframe can be dynamically adapted to the content.

Example :

```
<iframe src="<BML_url>" height="500px" width="500px"></iframe>
```

Attributes you need to know

Some additional attributes to help customize your iFrame

```
<iframe

  src="<BML_url>"     <!-- Sets the address of the document to embed -->

  height="100px"             <!-- Sets the iframe height in pixels -->

  width="100px"              <!-- Sets the iframe width in pixels -->

  name="my-iframe"           <!-- Sets the name of the iframe (mainly used to refe

  allow="fullscreen"         <!-- Sets the feature policy for the iframe. -->

  referrerpolicy="no-referrer" <!-- Set the referrer to send when fetching the ifram

  sandbox="allow-same-origin"  <!-- Sets the restrictions of the iframe -->

></iframe>
```

iFrame Events

Loading and Errors

Because an iframe is a document, you can use most global event handlers.

When you are initiating the iframe, two of them come in handy to improve the experience, like displaying a spinner or a specific message to assist the user:

The load event. It is triggered when the iframe is fully loaded. In other words, all static assets have been downloaded, and all the elements in the DOM tree have fired their load event. The error event that is triggered when the loading failed.

You can listen to them with the onload and onerror attribute respectively:

```
<iframe src="<BML_url>" onload="onLoad()" onerror="onError()"></iframe>
```

Or you can add the listeners to your iFrame programatically :

```javascript
// For a new iframe
const iframe = document.createElement("iframe");

iframe.onload = function() {
  console.log("The iframe is loaded");
};
iframe.onerror = function() {
  console.log("Something wrong happened");
};

iframe.src = "<BML_url>";
document.body.appendChild(iframe);

// For an existing iframe
const iframe = document.querySelector('.my-iframe');

iframe.onload = function() {
  console.log("The iframe is loaded");
}
iframe.onerror = function() {
  console.log("Something wrong happened");
}
```

Compatibility and Fallbacks

We sustain our interactive videos to be compatible with all major Web technologies.

A customized deployment will be offered in cases of no iFrame support

Deal with incompatible browsers

If a browser does not support an iframe, it will display the content included between the opening `<iframe>` tag and the closing `</iframe>` tag.

Thus, you should always think about placing a warning message as a fallback.

```html
<iframe>
  <p>Your browser does not support iframes.</p>
</iframe>
```

FAQs

Does it affect the SEO of my site/webapp ?

It is best to assume that the content displayed via iframes may not be indexed or available to appear in Google's search results. A workaround would be to made sure to provide additional text-based links to the content they display so that Googlebot can crawl and index this content.

Note: You should also not worry about duplicate content issues since today's web crawlers usually recognize them.

Will it affect the loading speed of my website?