

How to implement TimelineJS in your site

TimelineJS is an open-source tool that enables you to build visually-rich interactive timelines. TimelineJS can automatically load in photos, videos from YouTube or Vimeo, tweets, Wikipedia entries, and many other media types to help tell your story.

At Qualcomm, we have used TimelineJS in the implementation of one of our popular web applications called QMuseum, which gives users a virtual tour of Qualcomm's diverse and rich history and technologies, from its inception in 1985 to the present day. Drawing from our experience, this guide helps you understand the fundamental steps and best practices of integrating TimelineJS into a web application or website.

Pre-requisites

Any modern browser that supports HTML 5, JavaScript and CSS 3
Content entries formatted in a chronological sequence to make optimal use of timeline functionality.
Please note that the recommended number of entries supported by a single use of TimelineJS is 30.

Sample code

Example JSON file template (Step 1)

```
{
  "timeline":
  {
    "headline":"The Main Timeline Headline Goes here",
    "type":"default",
    "text":"<p>Intro body text goes here, some HTML is ok</p>",
    "asset": {

"media":"http://yourdomain_or_socialmedialink_goes_here.jpg",
    "credit":"Credit Name Goes Here",
    "caption":"Caption text goes here"
  },
    "date": [
      {
        "startDate":"2011,12,10",
        "endDate":"2011,12,11",
        "headline":"Headline Goes Here",
        "text":"<p>Body text goes here, some HTML is OK</p>",
        "tag":"This is Optional",
        "classname":"optionaluniqueclassnamecanbeaddedhere",
        "asset": {

"media":"http://twitter.com/ArjunaSoriano/status/164181156147900416",
          "thumbnail":"optional-32x32px.jpg",
```

```

    "credit": "Credit Name Goes Here",
    "caption": "Caption text goes here"
  }
},
"era": [
  {
    "startDate": "2011,12,10",
    "endDate": "2011,12,11",
    "headline": "Headline Goes Here",
    "text": "<p>Body text goes here, some HTML is OK</p>",
    "tag": "This is Optional"
  }
]
}
}

```

Sample HTML Page (Steps 2, 3, 4, and 5)

```

<!DOCTYPE html>
<html>
  <head>
    <meta charset="utf-8" />
    <title>The title of the page </title>
    <!-- The default timeline stylesheet -->
    <link rel="stylesheet" href="assets/css/timeline.css" />
    <!-- Our customizations to the theme -->
    <link rel="stylesheet" href="assets/css/styles.css" />
    <!-- Google Fonts -->
    <link rel="stylesheet" href="http://fonts.googleapis.com/css?family=Dancing+Script|Antic+Slab" />
    <!--[if lt IE 9]>
    <script src="http://html5shiv.googlecode.com/svn/trunk/html5.js"></script>
    <![endif]-->
  </head>
  <body>
    <div id="timeline">
      <!-- Timeline will generate additional markup here -->
    </div>
    <script type="text/javascript">
      var timeline_config = {
        width: '100%',
        height: '600',
        source: 'path_to_json/or_link_to_googlespreadsheet',
        embed_id: 'timeline', //OPTIONAL USE A DIFFERENT DIV ID FOR EMBED
        start_at_end: false, //OPTIONAL START AT LATEST DATE
        start_at_slide: '4', //OPTIONAL START AT SPECIFIC SLIDE
        start_zoom_adjust: '3', //OPTIONAL TWEAK THE DEFAULT ZOOM LEVEL
        hash_bookmark: true, //OPTIONAL LOCATION BAR HASHES
        font: 'Bevan-Potanosans', //OPTIONAL FONT
        debug: true, //OPTIONAL DEBUG TO CONSOLE
        lang: 'fr', //OPTIONAL LANGUAGE
        maptype: 'watercolor', //OPTIONAL MAP STYLE
        css: 'path_to_css/timeline.css', //OPTIONAL PATH TO CSS
        js: 'path_to_js/timeline-min.js' //OPTIONAL PATH TO JS
      }
    </script>
    <!-- JavaScript includes - jquery, turn.js and our own script.js -->
    <script src="http://code.jquery.com/jquery-1.7.1.min.js"></script>
    <script src="assets/js/timeline-min.js"></script>
    <script src="assets/js/script.js"></script>
  </body>
</html>

```

Step # 5

Step # 2

Step # 3

Step # 4

Step # 5

Steps

1. Gather your content entries together and create a data file in JSON format that will be used to provide information to TimelineJS.

Note: Alternatively, you can use Google Docs to build the content entries. For more information, please see <http://timeline.knightlab.com>.

2. Identify where you want the timeline to appear in the web page of your site and place the corresponding embed code into the page.

Note: Ensure that the JavaScript snippet is added just above the bottom of the closing body tag (</body>) in the web page so that the user's web browser will load the web page contents before running the JavaScript.

3. Set the required configuration parameters based on your application, parameters like width, height of the timeline, data source, the initial zoom level, starting point of the timeline, embed tag (position of timeline on the page), developers option like debugging are also available. In the configuration parameter the "embed_id" property provides the location of the timeline on the page. It points to the div element where the timeline js will put the timeline element after loading.
4. Include a script tag entry for the jQuery framework to provide the basic functionality required by the TimelineJS library.
5. Include a script tag entry for the TimelineJS-specific JavaScript libraries in the bottom of the body tag (below the jQuery script tag) and add CSS link tags for the CSS stylesheets in the head tag.

Issues and Suggested Remedies

- Due to multiple screen sizes of Android devices, there is a possibility of unwanted white space displayed in a mobile browser.
- Vimeo is not supported on Android and hence there may be a difficulty in video streaming.
 - Suggestions:
 - Host videos on your own server
 - Upload videos to YouTube and embed them in your HTML code
- HTML page loads very slowly
 - Suggestion: Ensure to prune the JSON data / content such that the number of content entries is not more than 30.d