

Publicly shared

Product spec: Integrations

Author: @pbakaus | Updated: 04/17/2020

This document describes the planned third party integrations for the story editor.

Note: Please contact pbakaus@ to get access to a [companion cheat sheet](#) with API keys and the likes. For obvious reasons, this doc is on a need to know basis.

Media Library integrations	1
Coverr	1
Tenor	1
Unsplash	3
Google Photos	4
Other Integrations	4
Google Fonts	4
Google Analytics	4
Google Drive	6

Media Library integrations

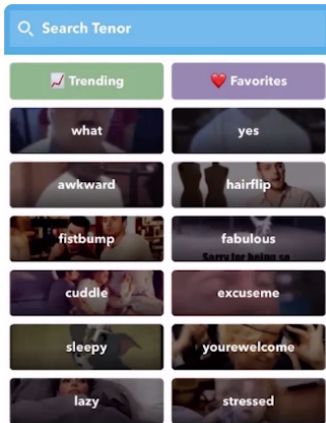
Coverr

Spec TBD. [Coverr](#) provides high quality, free background/cover videos.

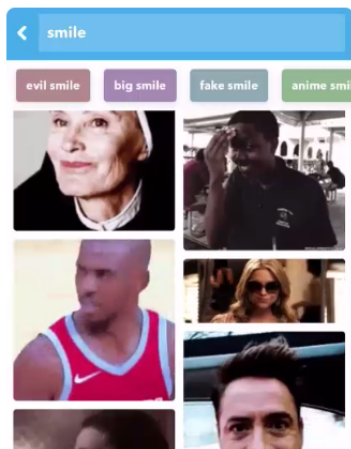
Tenor

Tenor brings millions of animated GIFs right into the editor to every story creator. Integration with [Tenor](#) is built into the story editor and does not require a separate plugin.

The Tenor tab behaves slightly differently to the other tabs and features high level categories to drill into when opening it, as per their documentation's example:



The search bar, in addition to providing basic search, offers rich autocomplete functionality when at least 2 characters are typed in, along with rich live suggestions:



Localized content

Tenor integration will forward the CMS user's locale to the API endpoints to show [locally relevant content](#) whenever possible.



Video, not gif: For performance reasons, we won't actually use animated GIFs in the exported story. We'll use the mp4/webm video instead (on autoplay/loop).

Hotlink only: Tenor content is not allowed to be cached/copied, so we will be embedding from their remote URL endpoints. Features like contrast detection and eyedropper will still work due to the use of CORS.

Unsplash

[Unsplash](#) brings millions of royalty-free, public domain stock imagery to the story editor. Unsplash integration is built into the editor, but can be enriched with the dedicated, official Unsplash plugin.

When first selecting the tab, two options are offered to the user:

1. **Install Unsplash** (recommended)
When choosing to **install** Unsplash, we automatically download and activate the Unsplash plugin from the WP plugin repository and forward to the OAuth authentication screen, then return back to the editor.
2. **Insert API Key**
When choosing to **insert an API key** instead, we show a field to insert an API key and a link/button leading to Unsplash's key registration page. As soon as the API has been inserted by the user, we validate the key and show the first results.

The images can be sorted in the following ways:

- Popular (default)
- Latest
- Oldest

In addition to sorting, the following ways allow the user to filter and explore:

1. Search
2. Collections
3. Tab for Unsplash in the media gallery that features
 - Popular images
 - Explore by tag
 - Search



Hotlink only: Like Tenor, Unsplash requires direct hotlinking to its images. Thankfully, their images send the appropriate cross-origin (CORS) headers, allowing us to use features like contract detection and eyedropper just like with local images.

Other Integrations

Google Fonts

The font picker allows the author to pick any font from the Google Fonts catalogue. Head to [the font picker spec](#) to learn more.

Google Analytics

Integrated Google Analytics integration that gets better with Site Kit installed.

The story editor supports two ways of inserting a Google Analytics tracking code & functionality:

1. Built-in (default)
2. Site-Kit

Built-in

If Site Kit is not installed, the pre-publish checklist and helper warn the user on their first story that Google Analytics is not configured/set. The user can choose one of the following three options then:

1. Setup basic tracking
2. Setup advanced tracking (via Site Kit)
3. Dismiss (I'm not using Google Analytics)

The action “**Setup basic tracking**” allows the user to insert their analytics ID in an input field and confirm. While this is an inline option in the setup flow, this id can always be changed in the global story plugin settings at a later time.

The basic tracking configures Google Analytics for all web stories with a default, non-modifiable optimized amp-analytics configuration, as described in [this blog post](#):

```
<amp-analytics type="gtag" data-credentials="include">
  <script type="application/json">
    {
      "vars": {
        "gtag_id": "YOUR\_GOOGLE\_ANALYTICS\_ID",
        "config": {
          "YOUR\_GOOGLE\_ANALYTICS\_ID": {
            "groups": "default"
          }
        }
      }
    }
  </script>
</amp-analytics>
```

```

    },
    "triggers": {
      "storyProgress": {
        "on": "story-page-visible",
        "vars": {
          "event_name": "custom",
          "event_action": "story_progress",
          "event_category": "${title}",
          "event_label": "${storyPageId}",
          "send_to": ["YOUR_GOOGLE_ANALYTICS_ID"]
        }
      },
      "storyEnd": {
        "on": "story-last-page-visible",
        "vars": {
          "event_name": "custom",
          "event_action": "story_complete",
          "event_category": "${title}",
          "send_to": ["YOUR_GOOGLE_ANALYTICS_ID"]
        }
      }
    }
  }
}
</script>
</amp-analytics>

```

The action **“Setup advanced tracking (via Site Kit)”** will lead to Site Kit installation and activation, and then subsequent configuration of Analytics in Site Kit.

The action **“Dismiss (I’m not using Google Analytics)”** is critical for those who don’t or can’t use Google Analytics. This dismisses the checklist warning indefinitely.



Note: The story editor’s settings page has an advanced option field to insert a custom `<amp-analytics>` configuration, and a built-in API hook to insert arbitrary HTML content into the page for technical users.

Site Kit

When Site Kit is installed, the Site Kit overrides the basic analytics behavior of the story editor and inserts its own snippet via a programmatic hook.

This is the recommended option to set up analytics for web stories, as this will allow users to further customize goals, advanced settings via Site Kit configuration screens.

For future consideration

Google Photos

Google Photos integration enables fluid creation workflows that span mobile, tablet and desktop. This integration is powered by Site Kit, due to the need to authenticate a Google user.

The Google photos tab displays the most recent pictures the user has taken as default, with the most recent pictures at the top. To quickly find the right image, it supports **fuzzy search**, as well as the the following ways to **filter**:

- Tag
- Face
- Location
- Date range

One dragged into the canvas, images get auto-imported into the local media gallery (required for publishing, as well as a technical need for stuff like contrast detection)

Google Drive

TBD.