Google Maps

- Screenshots
- Settings
- Data
- Template usage
  - Displaying a map
  - Getting the raw field value
  - Displaying a map inside a group
- Filters
  - rwmb_google_maps_url

import Screenshots from '@site/src/components/Screenshots';

The **Google Maps field** lets you select and save a location directly on a map. It works together with an **address text field**, which supports **autocomplete** powered by the Google Maps Geocoding service.

You can set a location in two ways:

- Type an address into the text field (autocomplete helps you quickly find the place).
- Click on the map or drag the marker to a new spot. The field will then save the coordinates automatically.

Screenshots

<Screenshots name="map" col1={[['https://imgur.elightup.com/zyKqRTD.png', 'The Google Maps field interface']]} />

Settings

In addition to the common field settings, this field has the following options:

Name	Key	Description
Google Maps API key	`api_key`	Your Google Maps API key. Get one here. **Required.**
Default location	`std`	The map's default location when it first loads. Format: `'53.346881,-6.258860'` (latitude, longitude). If not set, the map defaults to Dublin, Ireland.
Address field	`address_field`	The ID of the address field. For multiple fields, separate IDs with commas. **Required.**
Language	`language`	Set the map language. See the full list of language codes.
Region	`region`	A country code top-level domain. This helps autocomplete suggest results based on a specific region. Learn more.
Marker draggable	`marker_draggable`	Determines whether the marker can be dragged. Boolean, default is `true`. When set to `false`, the marker cannot be moved on the map — the location can only be set by searching for an address (Google will place the marker automatically).

:::warning Address field required You must create a text field for the address input, and pass its ID to the map's **Address field** setting. :::

The `address_field` can also be a list of multiple field IDs, separated by commas (e.g. `street,city,state`). In this case:

- You can use both text and select fields.
- Autocomplete will not work. Instead, the plugin combines the values from those fields to search for the location and updates the map whenever any field changes.

:::warning Don't forget to enable the **Geocoding API** in your Google Cloud project - it's required for the autocomplete feature! :::

Example field setup in code:

```
// Address field
[
    'id'   => 'my_address',
```

```
    'name' => 'Address',
    'type' => 'text',
],
// Map field
[
    'id'            => 'map',
    'name'          => 'Location',
    'type'          => 'map',
    'std'           => '-6.233406,-35.049906,15',
    'address_field' => 'my_address',
    'api_key'       => 'XXXXXXXXX',
],
```

Data

The field saves the location as:

```
latitude,longitude,zoom
```

Example: `53.346881,-6.258860,14`

Template usage

Displaying a map

To show the map on the frontend, use `rwmb_the_value()` with extra parameters:

```
<h2>Google Maps</h2>
<?php
$args = [
    'width'        => '640px',
    'height'       => '480px',
    'zoom'         => 14,
    'marker'       => true,
    'marker_icon'  => 'https://url_to_icon.png',
    'marker_title' => 'Click me',
```

```
    'info_window'  => '<h3>Title</h3><p>Content</p>.',
];
rwmb_the_value( 'my_field_id', $args );
?>
```

Available parameters:

Name	Description
`width`	Map width. Default: `640px`. Accepts `%` or `px`.
`height`	Map height. Default: `480px`. Accepts `%` or `px`.
`zoom`	Map zoom level. Default: value set in the admin, or `14` if not set.
`marker`	Show marker? `true` (default) or `false`.
`marker_icon`	URL to a custom marker icon. Optional.
`marker_title`	Marker tooltip (shows on hover).
`info_window`	Content shown when clicking the marker. HTML allowed. Avoid quotes inside to prevent JavaScript issues.
`js_options`	Pass custom JavaScript options. Reference.

The code below shows how to use `js_options` for advanced control how the map is displayed:

```
$args = [
    'width'      => '640px',
    'height'     => '480px',
    'js_options' => [
        'mapTypeId'   => 'HYBRID',
        'zoomControl' => false,
    ],
];
rwmb_the_value( 'my_field_id', $args );
```

Getting the raw field value

If you don't need to render the map but just want the coordinates:

```
$location = rwmb_get_value( $field_id );
echo $location['latitude'];
echo $location['longitude'];
```

```
echo $location['zoom'];
```

See more about `rwmb_get_value()`.

Displaying a map inside a group

If the map is part of a group field, the helper functions above won't work. Instead, use the built-in `RWMB_Map_Field::render_map()` helper:

```
$args = [
    'width'      => '640px',
    'height'     => '480px',
    'js_options' => [
        'mapTypeId'   => 'HYBRID',
        'zoomControl' => false,
    ],
];
$group_values = rwmb_meta( 'group_id' );
$args = [
    'api_key' => 'your-API-key'
];
// If the group is cloneable
foreach ( $group_values as $group_value ) {
    echo RWMB_Map_Field::render_map( $group_value['map_id'], $args );
}
```

The helper function accepts:

Name	Description
`$location`	Map location in the format `latitude,longitude[,zoom]` (zoom is optional). Same as the map field value.
`$args`	Additional display options. Same as in `rwmb_the_value()`.

Filters

`rwmb_google_maps_url`

This filter lets you modify the Google Maps script URL - for example, to add extra libraries. It receives the script URL as a parameter and should return the modified URL.

For example, the code below adds "geometry" library to the Google Maps script:

```
add_filter( 'rwmb_google_maps_url', function ( $url ) {
    return add_query_arg( 'libraries', 'geometry', $url );
} );
```

□□□ #2
□ □□□ □□□ 2026-08-29 15:36:17 UTC
□ □□□ □□□ 2026-08-29 15:36:17 UTC