GEMIUS

gemiusPrism Tracking Code

Installation Guide for DK market

2020.11.10

www.gemius.com

List of contents

1. Basic Script Installation

Installation of the gemiusPrism system scripts on a website must be performed by a person who has the ability to edit the project's code. To install the scripts, you need a project identifier in the gemiusPrism system and **Tracking Script**.

1.1 Overview

The basic configuration of the **Tracking Script** lets you analyze your project through URL address structure, referrers, keywords, as well as create basic statistics (geolocation, web browsers, mobile devices etc.).

The basic form of the **Tracking Script** placed in the source code of your website looks as follows:

```
<script type="text/javascript">
<!--//--><![CDATA[//><!--
var gemius_identifier = 'IDENTIFIER';
// lines below shouldn't be edited
function gemius_pending(i) { window[i] = window[i] || function() {var x =
window[i+'_pdata'] = window[i+'_pdata'] || []; x[x.length]=arguments;};};
gemius_pending('gemius_hit'); gemius_pending('gemius_event');
gemius_pending('pp_gemius_hit'); gemius_pending('pp_gemius_event');
(function(d,t) {try {var
gt=d.createElement(t),s=d.getElementsByTagName(t)[0],l='http'+((location.protocol=='ht
tps:')?'s':''); gt.setAttribute('async','async');gt.setAttribute('defer','defer');
gt.src=l+'://PREFIX.hit.gemius.pl/gemius.js'; s.parentNode.insertBefore(gt,s);} catch
(e) {}})(document,'script');
//--><!]]>
</script>
```

IDENTIFIER – this is a project identifier in the gemiusPrism system. It is a sequence of letters and numbers thanks to which the analyzing server recognizes the project from which data is coming.

Identifier example: '**zZCQBO8qo8FpfdK3lsjyMXZHjwzBsFs1vQ9YPinPCAX.M7**'.

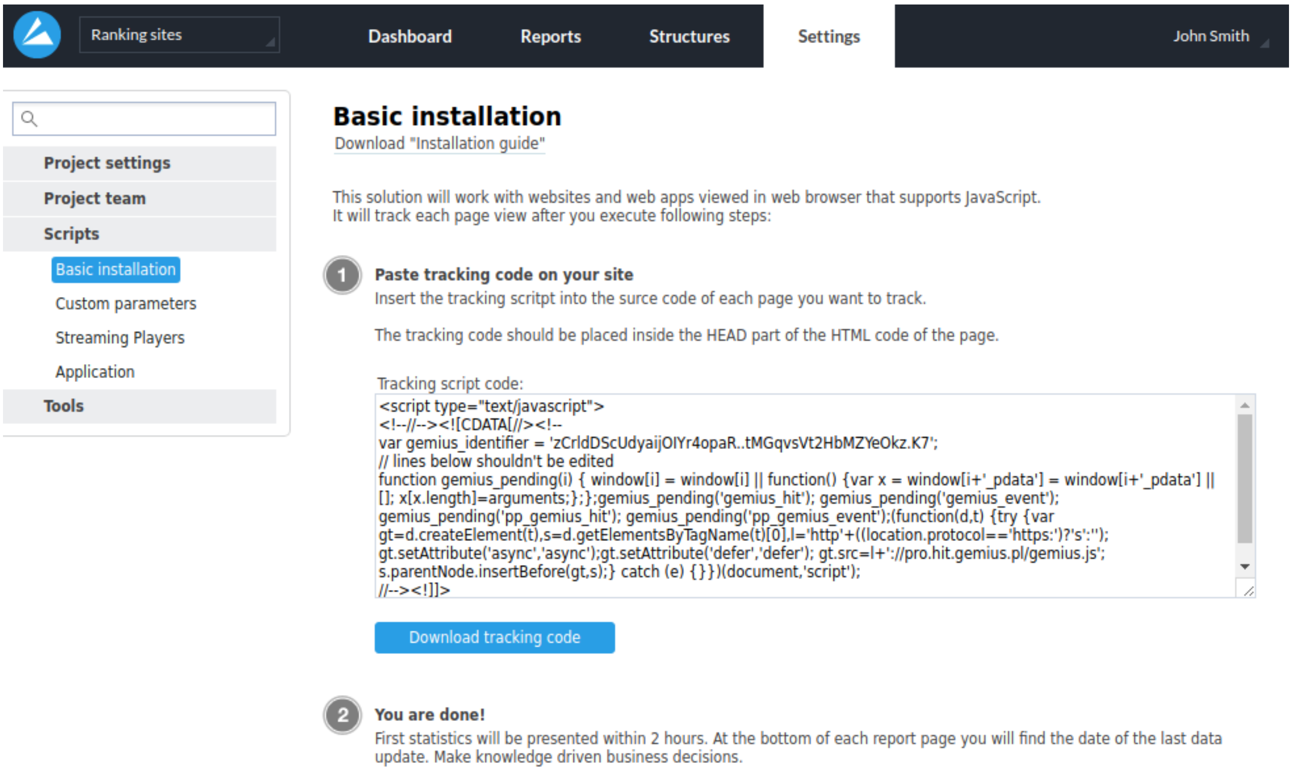
PREFIX – this is the name of the Gemius analyzing server.

Important: The **'gemius_identifier'** variable must be defined before the **Main Script** is loaded. If this condition is not fulfilled, data will not be gathered correctly.

In the case of a web project based on frames (**FRAME**), the **Tracking Script** should not be placed in the frame defining (**FRAMESET**) page code.

1.2 Tracking Script

You can download the **Tracking Script** file directly from the gemiusPrism interface. In order to do that, you must first choose **Settings** from the top menu of the selected project, then choose **Scripts** from the side menu and **Basic installation**, and follow the steps shown on the screen.



Basic installation

Download "Installation guide"

This solution will work with websites and web apps viewed in web browser that supports JavaScript.
It will track each page view after you execute following steps:

1 **Paste tracking code on your site**

Insert the tracking scritpt into the surce code of each page you want to track.

The tracking code should be placed inside the HEAD part of the HTML code of the page.

Tracking script code:

```
<script type="text/javascript">
<!--//--><![CDATA[//><!--
var gemius_identifier = 'zCrldDScUdyaijOlYr4opaR..tMGqvsVt2HbMZYeOkz.K7';
// lines below shouldn't be edited
function gemius_pending(i) { window[i] = window[i] || function() {var x = window[i+'_pdata'] = window[i+'_pdata'] ||
[]; x[x.length]=arguments;};};gemius_pending('gemius_hit'); gemius_pending('gemius_event');
gemius_pending('pp_gemius_hit'); gemius_pending('pp_gemius_event');(function(d,t) {try {var
gt=d.createElement(t),s=d.getElementsByTagName(t)[0],l='http'+((location.protocol=='https:')?'s':'');
gt.setAttribute('async','async');gt.setAttribute('defer','defer'); gt.src=l+'://pro.hit.gemius.pl/gemius.js';
s.parentNode.insertBefore(gt,s);} catch (e) {}})(document,'script');
//--><!]]>
```

Download tracking code

2 **You are done!**

First statistics will be presented within 2 hours. At the bottom of each report page you will find the date of the last data update. Make knowledge driven business decisions.

2. Scripting guidelines

Implementation of the **Tracking Script** and the functionality has to be in accordance to rules provided by DMR that can be found at: https://danskonlineindex.dk/definitioner/regelsaet/

2.1 Code placement

There can only be one **Tracking Script** per page impression (page view) (Afsnit 2B. HVORDAN DER SKAL TÆLLES - 2B.1.).

Important: Participants of the study that do not comply with the applicable rules may be excluded from the public gemiusAudience data site and all other statistical publications until their practice once again adheres to the applicable rules.

It is recommended to place the **Tracking Script** at the top of the <BODY> section.

Important: The **Tracking Script** file is loaded asynchronously and bares no impact on content and services load times, their performance, and user experience.

2.2 Tag Management platforms

It is **allowed** to implement the **Tracking Script** using Tag Management systems. The guidelines for this type implementation state that the Tracking Script execution must match the execution sequence provided in section **3 – Scripting guidelines for GDPR compliance**, part **3.3 – Execution sequence**.

2.3 What is to be counted

- Counting can only encompass publicly-accessible online pages which have been legally created.
- Pages not generally and publicly accessible (e.g. intranet pages) cannot be counted.
- In order for a page to be counted, it is necessary for the user to have performed an action (e.g. clicked on a link or entered an URL address in Internet browser) so as to gain access to the page.
- Pages with built-in auto-forwarding, the measurement code cannot be placed in. If a user has taken action to access the transmitted page, the counting code can be placed on the last page the user enters, i.e., not on pages that the user automatically skips.
- Pop-up windows can only be inserted a measurement code if the user has to open such windows independently, i.e., an action is performed on the portal (for example, a click on a link) to open it.
- Only user-generated page views can be counted.
- One has to inform Gemius in case auto refresh is necessary for the website to be able to provide good quality service to their visitors. By default, page views which are generated automatically are deducted from page views, presented in gemiusAudience monthly data.

2.4 Special cases

- As a rule, in pages which are developed with frames (including AJAX, pop-ups and similar solutions), page impressions (page views) can only be counted provided that a user-initiated page impression (page view) corresponds to a renewal of at least 40% of the area.
- Publications that use frames must place the counting tag in the content frame – not in menus, top frames, hidden frames or other areas.
- Pages with functions that differ from those of "ordinary" HTML pages, e.g. various applications, games, chat or streaming, can be tagged if approved by Gemius and if they are in compliance with rules stated in section
- “FB Instant Articles", “Google Accelerated Mobile Pages” - web page statistics generated using Facebook IA or Google AMP solutions are counted if the codes are located in accordance with Gemius certified documentation.
- Use of server-side or other solutions that change the User Agent values of site visitors or modify hits in any way is prohibited.

Disclaimer

Gemius does not intend to obstruct the automated code deployment provided by numerous content management systems (CMS). We fully support the use of such tools as long as the end result is a measurement script delivered to the webpage's code in an unmodified and unveiled form. The majority of tag management tools offered on the market does not fulfill these two simple conditions.

Our recommendation is to refrain from using tag managers in the measurement of code deployment, and it originates from a significant investment of our resources to maintain a high quality of the study and aims to eliminate looming threats instigated by the use of 3rd party code to control the study. Gemius is able to guarantee the quality of its services only if scripts are pasted according to our recommendations.

The application of a not recommended method of tag embedding might influence both site-centric data and panel recruitment process.

3. Scripting guidelines for GDPR compliance

3.1 Introduction

It is a condition for participation in the measurement that the publisher is in compliance with relevant legislation and guidelines from the **Danish Data Protection Agency** and the **Danish Business Authority**. This includes that the publisher must gather active consent from users entering a website – without prior consent – for the websites use of cookies for statistical purposes.

3.2 Cookies categorization

Gemius uses cookies to collect data about the use of our website such as the number of visitors, the time users spend on a webpage and composing visitor statistics.

Name	Domain	Expiration	Functionality
__gfp_64b	used as cookie first party in Client's domain	1000 days	Unique User ID. Collects information about how the visitor uses the website.
Gdyn	hit.gemius.pl; used as cookie third party (gemius.pl domain)	5 years	Unique User ID. Collects information about how the visitor uses the website.
gstorage	local storage in ls.hit.gemius.pl domain	valid till deletion	Unique User ID. Collects information about how the visitor uses the website.
gaddstorage	local storage in ls.hit.gemius.pl domain	valid till deletion	Unique User ID. Collects information about how the visitor uses the website.
Gtest	hit.gemius.pl	max 5 years	Checks whether the users' browser has third-party-cookies enabled. Temporary cookie files.
OPT-OUT	hit.gemius.pl	1-Jan-2030	Refusal to participate in researches conducted by Gemius.

All Gemius cookies have to be classified as "Statistical" cookies.

3.3 Execution sequence

Tracking Script should, in connection with Consent Management Solution, only be executed after the event of consent statement is completed. Depending on the nature of the consent the user choice of rejecting cookies must be respected.

1st Page Load			**2nd Page Load**	
CMP shown	Consent Granted	Consent Rejected	Consent Granted	Consent Rejected
No PageView	PageView	PageView with DNT	PageView	PageView with DNT

PageView	An event reflecting execution of a tracking script on a visited page of a given website.
PageView with DNT	An event reflecting execution of a tracking script on a visited page of a given website counted as non-cookie/non-Browser ID.
No PageView	Tracking script was not executed.

Tracking Script execution sequence and the usage of extra parameters can be inspected with Gemius Debugger. More information can be found at section **4. Debugging Script with Gemius Debugger**.

3.4 Tracking script execution in a custom JavaScript function

In order to achieve execution sequence that is described in section **3.3 Execution sequence** there is a need to place **Tracking script** in a custom JavaScript function. A JavaScript function is a block of code designed to perform a particular task. A JavaScript function is executed when "something" invokes it (calls it). In order to properly execute **Tracking script** inside of a custom JavaScript function, please meet the following conditions:

- When gemiusPrism tracking script is used in a JavaScript function - all gemiusPrism extra parameters (example: pp_gemius_identifier, pp_gemius_dnt and etc…) have to have a global scope.

- The 'pp_gemius_identifier' variable must be defined before the Main Script is loaded.

If one of the conditions is not met, the data will not be gathered correctly.

Example:

```
<script type='text/javascript'>
window.addEventListener('CookieInformationConsentGiven', function (event) {
  if (CookieInformation.getConsentGivenFor('cookie_cat_statistic')) {
<!--//--><![CDATA[//><!--
window.pp_gemius_identifier = 'IDENTIFIER';
// lines below shouldn't be edited
function  gemius_pending(i)  {  window[i]  =  window[i]  ||  function()  {var  x  =
window[i+'_pdata']          =          window[i+'_pdata']          ||          [];
x[x.length]=arguments;};};gemius_pending('gemius_hit');
gemius_pending('gemius_event');                        gemius_pending('pp_gemius_hit');
gemius_pending('pp_gemius_event');(function(d,t)               {try              {var
gt=d.createElement(t),s=d.getElementsByTagName(t)[0],l='http'+((location.protocol=='ht
tps:')?'s':'');     gt.setAttribute('async','async');gt.setAttribute('defer','defer');
gt.src=l+'://PREFIX.hit.gemius.pl/xgemius.js'; s.parentNode.insertBefore(gt,s);} catch
(e) {}})(document,'script');
//--><!]]>
  }
  }, false);
</script>
```

In this example the variable **pp_gemius_identifier** is declared at the global scope by attaching the variable to the "global object", which in a browser environment is aliased to **window** object.

A variable is a global variable if it's declared outside of a function (with or without "var"), or if it's declared inside a function without using "var", or if it's declared as window.myVariable or window["myVariable"].

3.5 Usage of Do Not Track Parameter

If the visitor objects to processing of his data on your website, you are obliged to change the script by adding the **var pp_gemius_dnt** parameter set to **1**:

```
<script type="text/javascript">
<!--//--><![CDATA[//><!--
var gemius_identifier = 'IDENTIFIER';
var pp_gemius_dnt = 1;
// lines below shouldn't be edited
function gemius_pending(i) { window[i] = window[i] || function() {var x =
window[i+'_pdata'] = window[i+'_pdata'] || []; x[x.length]=arguments;};};
gemius_pending('gemius_hit'); gemius_pending('gemius_event');
gemius_pending('pp_gemius_hit'); gemius_pending('pp_gemius_event');
(function(d,t) {try {var
gt=d.createElement(t),s=d.getElementsByTagName(t)[0],l='http'+((location.protocol=='ht
tps:')?'s':''); gt.setAttribute('async','async');gt.setAttribute('defer','defer');
gt.src=l+'://PREFIX.hit.gemius.pl/gemius.js'; s.parentNode.insertBefore(gt,s);} catch
(e) {}})(document,'script');
//--><!]]>
```

Events generated by such visitors will be counted as non-cookie/non-Browser ID events.

If the visitor consents to processing of his data on your website, you are obliged to change the script by adding the **var pp_gemius_dnt** parameter set to **0**:

```
<script type="text/javascript">
<!--//--><![CDATA[//><!--
var gemius_identifier = 'IDENTIFIER';
var pp_gemius_dnt = 0;
// lines below shouldn't be edited
function gemius_pending(i) { window[i] = window[i] || function() {var x =
window[i+'_pdata'] = window[i+'_pdata'] || []; x[x.length]=arguments;};};
gemius_pending('gemius_hit'); gemius_pending('gemius_event');
gemius_pending('pp_gemius_hit'); gemius_pending('pp_gemius_event');
(function(d,t) {try {var
gt=d.createElement(t),s=d.getElementsByTagName(t)[0],l='http'+((location.protocol=='ht
tps:')?'s':''); gt.setAttribute('async','async');gt.setAttribute('defer','defer');
gt.src=l+'://PREFIX.hit.gemius.pl/gemius.js'; s.parentNode.insertBefore(gt,s);} catch
(e) {}})(document,'script');
//--><!]]>
```

Events generated by such visitors will be counted as cookie/Browser ID events.

3.6 Consent Management Providers (CMP)

A CMP is a solution implemented on the publisher's website which allows the user to decide on sharing his personal data and helps to pass this information to vendors. The IAB holds a list of registered Consent Management Providers (CMPs), which is available at https://iabeurope.eu/cmp-list.

In order to guarantee all study participants, the most precise results, comparable measurement environment and equal treatment, every participant is required to use Consent Management Platform (CMP) that is compliant with IAB Europe's GDPR Transparency & Consent Framework 2.0.

In order to provide support for Consent Management Platforms (CMP) that are compliant with IAB Europe's GDPR Transparency & Consent Framework, we share the following parameters that help pass the consent information to our servers.

Important: Please note that all variables are required.

Set **var gemius_use_cmp** to **true**. Consent is collected via the CMP. If the user does not give his consent to the processing of his personal data by Gemius, hits will not contain information about cookieID and browserID.

Set **var pp_gemius_cmp_timeout** to **infinity**. The script detects the presence of CMP supporting TCF 2.0 on the website and will not send any hits until the CMP interface exits. More precisely, execution of the **Tracking script** will be paused and will wait for CMP to inform it about granted consents or that the user has closed the CMP after it was displayed in given session.

If you use a CMP that is not compliant with the IAB Framework please contact our technical support (audience-dk@gemius.com) for further instructions.

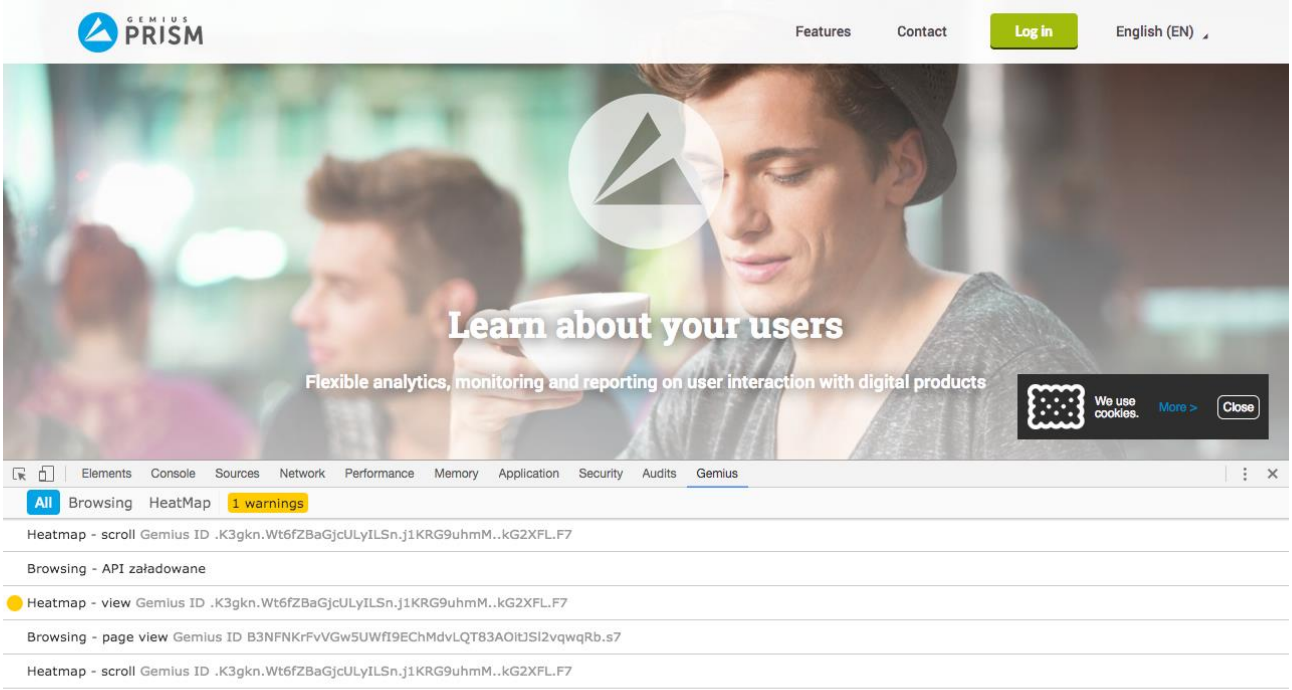
```
<script type="text/javascript">
<!--//--><![CDATA[//><!--
var gemius_identifier = 'IDENTIFIER';
var gemius_use_cmp = true;
var pp_gemius_cmp_timeout = 'infinity';
// lines below shouldn't be edited
function gemius_pending(i) { window[i] = window[i] || function() {var x =
window[i+'_pdata'] = window[i+'_pdata'] || []; x[x.length]=arguments;};};
gemius_pending('gemius_hit'); gemius_pending('gemius_event');
gemius_pending('pp_gemius_hit'); gemius_pending('pp_gemius_event');
(function(d,t) {try {var
gt=d.createElement(t),s=d.getElementsByTagName(t)[0],l='http'+((location.protocol=='ht
tps:')?'s':''); gt.setAttribute('async','async');gt.setAttribute('defer','defer');
gt.src=l+'://PREFIX.hit.gemius.pl/gemius.js'; s.parentNode.insertBefore(gt,s);} catch
(e) {}})(document,'script');
```

4. Debugging Script with Gemius Debugger

Gemius Debugger is a Chrome browser extension that helps to verify Gemius script installation.

The extension displays information whether or not Prism scripts are properly installed and working correctly on your website and all its subpages. When the debugging mode is enabled, a list of standard, e-commerce and stream hits is presented with names, comments and warnings/errors counter. A solution to each warning/error found is also provided.

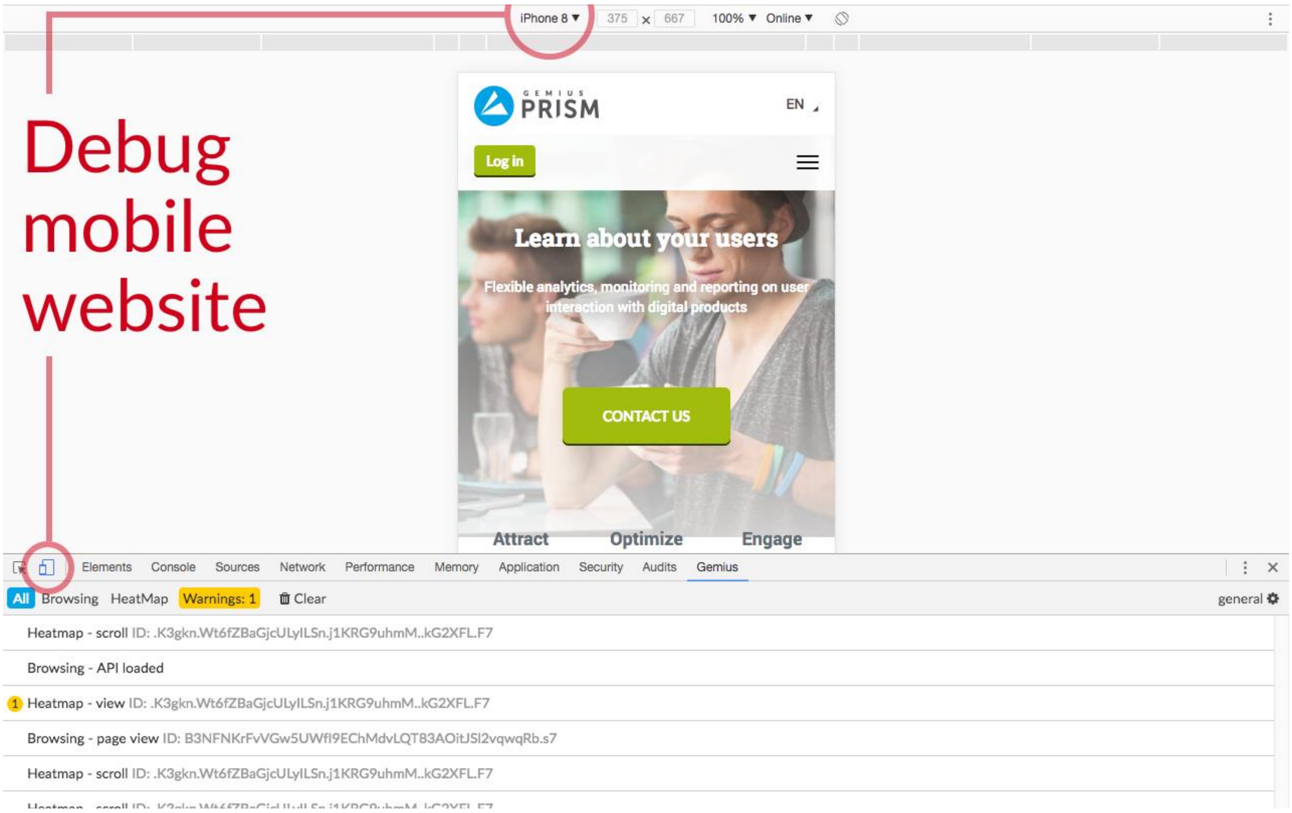
To start the verification process, install Gemius Debugger from https://debugger.gemius.com.



```
Heatmap - scroll Gemius ID .K3gkn.Wt6fZBaGjcULyILSn.j1KRG9uhmM..kG2XFL.F7
Browsing - API załadowane
● Heatmap - view Gemius ID .K3gkn.Wt6fZBaGjcULyILSn.j1KRG9uhmM..kG2XFL.F7
Browsing - page view Gemius ID B3NFNKrFvVGw5UWfI9EChMdvLQT83AOitJSl2vqwqRb.s7
Heatmap - scroll Gemius ID .K3gkn.Wt6fZBaGjcULyILSn.j1KRG9uhmM..kG2XFL.F7
```

After Gemius Debugger is installed, go to Developer Tools (Menu > More Tools > Developer Tools), choose the Gemius tab, click general ⚙ and select options that match your project (you can also do it by going to More Tools > Extensions: chrome://extensions/)

.



```
Heatmap - scroll ID: .K3gkn.Wt6fZBaGjcULyILSn.j1KRG9uhmM..kG2XFL.F7
Browsing - API loaded
1 Heatmap - view ID: .K3gkn.Wt6fZBaGjcULyILSn.j1KRG9uhmM..kG2XFL.F7
Browsing - page view ID: B3NFNKrFvVGw5UWfI9EChMdvLQT83AOitJSI2vqwqRb.s7
Heatmap - scroll ID: .K3gkn.Wt6fZBaGjcULyILSn.j1KRG9uhmM..kG2XFL.F7
```

Gemius Debugger can also be used to debug mobile versions of websites. In order to do so, click the Toggle device toolbar icon in the top-left corner and use the viewport controls to test your website on a specific device.

Important: Tracking Script execution must match the execution sequence provided in section **3 – Scripting guidelines for GDPR compliance**, part **3.3 – Execution sequence**.

5. Tracking verification

It is a condition for participation in the measurement that the publisher is in compliance with relevant legislation and guidelines from Gemius. This includes an obligation for Publishers to inform Gemius (audience-dk@gemius.com) of **Tracking Script** implementation.

The implementation of **Tracking Script** must be **verified** and **confirmed** by Gemius.