

# Reguły umieszczania skryptów audytu site-centric w Badaniu Mediapanel

## Serwisy WWW

Wersja 2.2

## Spis treści

|     |  |    |
|-----|--|----|
| 1.  | Definicje .....  | 3  |
| 2.  | Wstęp .....  | 4  |
| 3.  | Ogólne reguły umieszczania skryptów .....  | 4  |
| 4.  | Definicja odsłony strony internetowej oraz przykłady akcji uznawanych i nieuznawanych za odsłony ..... | 6  |
| 5.  | Informacje wymagające zgłoszenia .....   | 8  |
| 6.  | Skrypty audytowe.....  | 8  |
| 7.  | Podstawowy skrypt audytowy (odsłony pełne) .....   | 8  |
| 8.  | Skrypt zliczający (tzw. „wklejka” do umieszczenia w źródle strony):.....                               | 9  |
| 9.  | Skrypt główny (plik xgemius.js): .....   | 9  |
| 10. | Ładowanie zminifikowanego skryptu głównego. ....   | 9  |
| 11. | Odsłony częściowe.....   | 10 |
| 12. | Wywołanie dla stron z elementami AJAX:.....  | 10 |
| 13. | Wywołanie dla stron z elementami FLASH:.....   | 10 |
| 14. | Oskryptowanie w przypadku automatycznego odświeżania stron internetowych .....                         | 11 |
| 15. | Reguła generowania odsłony na dwa oddzielne projekty .....   | 12 |
| 16. | Pomiar odsłon dla Single-page Applications .....   | 14 |
| 17. | Rekomendacja back-forward cache .....  | 17 |
| 15. | Pomiar czasu.....  | 17 |
| 16. | Zasady dotyczące skryptów .....  | 17 |

## 1. Definicje

**Audyt** - usługa, polegająca na wykorzystaniu danych site-centric do uzupełnienia wyników Badania Mediapanel. Audyt opiera się na pomiarze site-centric realizowanym przez firmę Gemius S.A.

**Firma badawcza** – spółka Gemius S.A. z siedzibą w Warszawie

**Identyfikator skryptu** – unikalny znacznik jednoznacznie identyfikujący dany *skrypt audytowy*. Przykładowym identyfikatorem skryptu jest ciąg znaków:

"nSeVf3eRCXDTjZUCObr3wWaDXteBMgcTyCA5.lwjed..h7".

**Komponent statyczny** – część graficzna i funkcjonalna warstwy prezentacji realizująca funkcję tła, nagłówek, stopki, menu lub paska nawigacji, które nie podlegają zmianie w zależności od dostarczanych treści bądź usług.

**Odsłona** – zdarzenie pełnego przeładowania nową treścią zawartości strony w oknie przeglądarki lub aplikacji z wyłączeniem Komponentów Statycznych, poprzedzone każdorazowo intencjonalną akcją użytkownika wynikłą z intencji pozyskania nowej treści, aktualizacji bieżącej treści lub skorzystania z nowej usługi.

Dla potrzeb klasyfikacji technicznej sposobu powstawania odsłony dzielimy je na dwa typy:

**Odsłona Pełna** – Odsłona, w trakcie której następuje przeładowanie wszystkich Komponentów Statycznych.

**Odsłona Częściowa** – Odsłona, w trakcie której nie następuje przeładowanie wszystkich Komponentów Statycznych.

**Skrypty audytowe** - skrypty odpowiedzialne za zbieranie danych służących do *audytu*.

**Funkcja pp\_gemius\_hit** – funkcja javascript stanowiąca część skryptu audytowanego, służąca do przesyłania informacji o wykonaniu odsłony częściowej.

**Strony automatycznie odświeżane** - strony internetowe, które automatycznie (bez intencjonalnego działania użytkownika internetu) przeładują się w przeglądarce użytkownika (są ponownie renderowane). Przykładowymi stronami tego typu są wykresy giełdowe.

**Strony przekierowujące** – strony internetowe, które automatycznie (bez intencjonalnego działania użytkownika internetu) automatycznie przekierowują na inne strony internetowe (np. poprzez użycie skryptów w nagłówku HTML).

**Strony z ramkami** - strony zbudowane w oparciu o ramki (FRAMESET) lub zawierające ramki (iFrame), składające się z więcej niż jednego dokumentu html.

## 2. Wstęp

Witryny występujące w Badaniu Mediapanel mogą poddać się *audytowi* umieszczając *skrypty audytowe* w kodzie stron internetowych. *Skrypt audytowy* oraz inne (nie będące *skryptami audytowymi*) *skrypty Gemius* mogą występować równocześnie w kodzie jednej strony internetowej. Na podstawie identyfikatora skryptu *firma badawcza* określa czy skrypt jest *skryptem audytowym*. *Firma badawcza* na żądanie udziela informacji, czy dany *identyfikator skryptu* jest przyporządkowany do *skryptu audytowego*, czy do innego (nie będącego *skryptem audytowym*) *skryptu Gemius*. *Skrypty audytowe* podlegają poniższym regułom rozmieszczania.

## 3. Ogólne reguły umieszczania skryptów

- 1) Na każdej stronie internetowej należącej do witryny audytowanej w Badaniu Mediapanel powinien być przez cały miesiąc badany umieszczony *skrypt audytowy* (dopuszcza się jedynie braki oskryptowania opisane poniżej).

Niekompletne oskryptowanie witryny (umieszczone nie na wszystkich podstronach lub nie przez cały miesiąc) powoduje zniżenie wyników *audytu*, za które firma badawcza nie ponosi odpowiedzialności. Niekompletność audytu może stanowić podstawę do zaprzestania audytowania danej strony w wynikach badania.

*Skrypty audytowe* powinny być umieszczone w kodzie wszystkich stron danej witryny, włącznie ze stronami komunikatów o błędach takimi jak komunikat o nieznalezieniu strony ("Error 404 Page not found"). Formaty reklamowe nie są traktowane jako strony witryny i nie jest dozwolone wklejanie w nie *skryptów audytowych*, w szczególności niedozwolone jest wklejanie *skryptów audytowych* w reklamy typu pop-up i inne formy reklam wyświetlanych w odrębnych oknach lub powłokach.

- 2) *Skrypt audytowy* powinien być umieszczony na początku kodu audytowanej strony, bezpośrednio przed tagiem zamknięcia sekcji HEAD (...</HEAD>).

Miejsce wklejenia *skryptu audytowego* na stronie może wpływać na wyniki *audytu* w zależności od innych elementów strony. Ze względu na spójność i porównywalność wyników *audytu* skrypty powinny być umieszczane na witrynach w jednakowy sposób. W szczególnych przypadkach, np. ze względu na specyficzną budowę witryny, *firma badawcza* może wyrazić zgodę na umieszczenie skryptu w innym miejscu. Wklejony *skrypt audytowy* powinien być wywoływany bezpośrednio z dedykowanego serwera wskazanego przez *firmę badawczą* i wczytywany asynchronicznie.

- 3) *Skrypty audytowe* powinny być wklejone w kod stron w dokładnie takiej postaci, w jakiej zostały dostarczone przez *firmę badawczą*.

Wprowadzanie jakichkolwiek zmian w kodzie *skryptu audytowego* jest niedozwolone. Zmiana kodu może spowodować nieprawidłowe działanie *skryptu audytowego* i wpłynąć na wyniki *audytu*. W szczególnych przypadkach, *firma badawcza* może wyrazić zgodę na zmodyfikowanie kodu *skryptu audytowego*. W żadnym przypadku nie jest dopuszczalne kodowanie lub zastrzeganie dostępu do kodu skryptu audytowanego w nieuzasadniony sposób utrudniające lub uniemożliwiające weryfikacje pomiaru i stojące w sprzeczności z transparentnością Badania Mediapanel.

- 4) W trakcie wyświetlania (renderowania) strony w przeglądarce użytkownika może nastąpić tylko jedno wykonanie *skryptu audytowego*.

*Pełnej odsłonie* strony internetowej może towarzyszyć wyłącznie jedno wykonanie *skryptu audytowego*. W tę samą stronę internetową nie wolno wklejać więcej niż jednego *skryptu audytowego*. Umieszczenie więcej niż jednego *skryptu audytowego* w kodzie strony, lub spowodowanie w inny sposób wykonania *skryptu audytowego* więcej niż raz w trakcie wyświetlania strony internetowej, zaburza wyniki *audytu* zmieniając wskaźniki rejestrowane na audytowanej witrynie, szczególnie wpływając na liczbę *odślon* rejestrowanych na audytowanej witrynie. Podobnie każdej akcji internauty wyczerpującej definicję odsłony częściowej towarzyszyć może dokładnie jedno wygenerowanie odsłony przy użyciu funkcji `pp_gemius_hit`.

Ograniczenie powyższe dotyczy wyłącznie *skryptów audytowych* – dopuszczalne jest umieszczanie na stronie więcej niż jednego (nie będącego *skrypcem audytowym*) innego *skryptu Gemius*. W trakcie wyświetlania strony w przeglądarce użytkownika może nastąpić więcej niż jedno wykonanie (nie będącego *skrypcem audytowym*) *skryptu Gemius*.

- 5) Niedozwolone jest umieszczanie *skryptu audytowego* na stronie internetowej w taki sposób, który powoduje wykonanie skryptu bez uprzedniego przeładowania (ponownego wyświetlenia w przeglądarce użytkownika) strony internetowej. Do przesłania odsłony częściowej wyczerpującej jej definicję, służy wbudowana funkcja *skryptu audytowego* `pp_gemius_hit`.

Wykonanie *skryptu audytowego* bez uprzedniego przeładowania strony internetowej powoduje zliczenie w *audycie pełnej odsłony* pomimo niewykonania *odsłony* przez użytkownika. Umieszczenie *skryptów audytowych* w opisany sposób zaburza wyniki *audytu* zmieniając liczbę odsłon rejestrowaną na audytowanej witrynie i stanowi podstawę dla *firmy badawczej* do nieuwzględnienia tak przesłanych danych audytowych w badaniu. Rygor ten nie dotyczy pomiaru odsłon częściowych.

- 6) Pod warunkiem uprzedniego poinformowania i za zgodą *firmy badawczej* dozwolone jest umieszczanie *skryptów audytowych* w kodach *stron automatycznie odświeżanych*.

*Strony automatycznie odświeżane* wyświetlają (renderują) stronę internetową w przeglądarce użytkownika automatycznie. *Skrypt audytowy* umieszczony na tego typu stronie rejestruje więcej *odślon* niż liczba *odślon* wykonanych intencjonalnie przez użytkownika. Właściciel witryny zawierającej *strony automatycznie odświeżane* ma obowiązek zgłaszania takich stron *firmie badawczej*. Liczba odsłon będących wynikiem automatycznego odświeżania stron jest oszacowywana przez *firmę badawczą* i w ostatecznych wynikach Badania Mediapanel odejmowana jest od łącznej liczby odsłon zarejestrowanych przez skrypty audytowe na danym serwisie.

Jeżeli jest to tylko możliwe to umieszczanie automatycznie odświeżanych elementów strony (np. wykresów giełdowych) powinno być wykonane w taki sposób, aby ich automatyczne odświeżanie np. co minutę nie powodowało równoczesnego automatycznego wywołania skryptu audytowego. Może to być zrobione poprzez umieszczenie tego elementu w ramce (i odświeżanie tylko ramki a nie całej strony ze skrypcem) lub zastosowanie technologii takich jak np. AJAX.

Jeżeli nie jest to możliwe i musi być odświeżana automatycznie cała strona, to w momencie takiego przeładowania strony, skrypt powinien być wywołany w zmodyfikowany sposób, opisany w rozdziale „Oskryptowanie w przypadku automatycznego odświeżania stron internetowych”.

- 7) W trakcie wyświetlania (renderowania) *stron z ramkami* w przeglądarce użytkownika może nastąpić tylko jedno wykonanie *skryptu audytowego*.

Strona zbudowana w oparciu o ramki (FRAMESET) lub zawierająca ramki typu iFrame może zawierać tylko jeden *skrypt audytowy* (bez względu na liczbę ramek zawartych w tej stronie).

Skrypt audytowy powinien być umieszczony:

w przypadku stron zbudowanych w oparciu o ramki (FRAMESET) w kodzie strony definiującej ramki (FRAMESET) pod koniec sekcji HEAD (...</HEAD>),

w przypadku stron zawierających ramki typu iFrame skrypt audytowy powinien być wklejony w kod strony zawierającej ramkę pod koniec sekcji HEAD (...</HEAD>).

Niedozwolone jest wklejanie *skryptów audytowych* do dokumentu będącego zawartością ramki (FRAME) w stronach zbudowanych w oparciu o ramki (FRAMESET), oraz niedozwolone jest wklejanie *skryptów audytowych* do dokumentu będącego zawartością ramki typu iFrame. W przypadku pomiaru odsłon częściowych generowanych w trakcie zmiany zawartości ramki, elementem generującym wysłanie odsłony dla funkcji pp\_gemius\_hit powinien być ten, który inicjuje tą akcję w wyniku działań internauty, np. użycie przycisku nawigacyjnego na stronie.

Umieszczenie *skryptów audytowych* w inny niż opisany wyżej sposób może spowodować nieprawidłowe działanie *skryptu audytowego* i/lub zaburzenie wyników *audytu* zmieniając wskaźniki rejestrowane na witrynie i stanowi podstawę dla *firmy badawczej* do nieuwzględnienia tak przesłanych danych audytowych w badaniu.

- 8) Niedozwolone jest umieszczanie *skryptów audytowych* na *stronach przekierowujących*.

#### 4. Definicja odsłony strony internetowej oraz przykłady akcji uznawanych i nieuznawanych za odsłony

Odsłona to zdarzenie pełnego przeładowania nową treścią zawartości strony w oknie przeglądarki z wyłączeniem Komponentów Statycznych, poprzedzone każdorazowo intencjonalną akcją użytkownika wynikłą z intencji pozyskania nowej treści, aktualizacji bieżącej treści lub skorzystania z nowej usługi.

Zdarzenie Odsłony nie musi łączyć się ze zmianą adresu URL (np. galerie występują pod jednym adresem URL).

Komponent statyczny to część graficzna i funkcjonalna warstwy prezentacji realizująca funkcję tła, nagłówek, stopki, menu lub paska nawigacji, które nie podlegają zmianie w zależności od dostarczanych treści bądź usług.

Dla potrzeb klasyfikacji technicznej sposobu powstawania odsłony dzielimy je na dwa typy:

Odsłona Pełna – Odsłona, w trakcie której następuje przeładowanie wszystkich Komponentów Statycznych.

Odsłona Częściowa – Odsłona, w trakcie której nie następuje przeładowanie wszystkich Komponentów Statycznych.

**W szczególności za intencjonalną akcję skutkującą odsłoną strony internetowej uznaje się:**

- Odświeżenie, czyli ponowne wczytanie strony w przeglądarce (np. przy pomocy klawisza F5).
- Powrót do poprzedniej strony (np. przyciskami nawigacyjnymi w przeglądarce).
- Przejście pomiędzy zdjęciami w galeriach zdjęć umieszczonych na warstwach.
- Przeciągnięcie strony w bok (tzw. „swipe”) powodujące załadowanie kolejnych treści, np. przechodzenie między kolejnymi artykułami, zmiany kategorii treści.
- Przewinięcie (scrollowanie) strony, jeżeli łączy się z pełnym przeładowaniem nową treścią zawartości strony w oknie przeglądarki z wyłączeniem Komponentów Statycznych. Efektem przewijania strony musi być efekt tożsamy z kliknięciem w link czyli przejście na inną stronę.
- W przypadku poczty elektronicznej:
  - Usunięcie wiadomości.
  - Pobranie nowych wiadomości.
  - Opróżnienie kosza.
  - Filtrowanie poczty – użytkownik decyduje wg. jakiej kategorii w oknie skrzynki pocztowej będą widoczne wiadomości.

- Wyszukiwanie wiadomości w folderze poczty elektronicznej.
- Przechodzenie pomiędzy folderami.

**W szczególności za intencjonalną akcją skutkującą odsłoną NIE uznaje się:**

- Przewijanie (scrolowanie) strony o ile nie spełnia warunku wywołania pełnego przeładowania strony opisanego dokładnie powyżej - nawet jeżeli łączy się z doładowywaniem nowych treści w momencie przewijania strony ( Przykład: scrolowanie/przewijanie tzw. rolek czyli załadowanie kolejnych krótkich materiałów wideo. Bez znaczenia czy akcja jest intencjonalna czy z autoplay ).
- Filtrowanie treści w obrębie bieżącej treści, powodującej jedynie jej zawężenie.
- Wyszukiwanie fragmentu w obrębie bieżącej treści.
- Zmiany widoku poprzez np. umożliwienie oznaczania różnymi kolorami listów na poczcie.
- Wyświetlanie komunikatów i „dymków” – (np. „Czy opróżnić kosz” – „tak”, „nie”), nawet, jeśli są na warstwie.
- Wypełnianie poszczególnych pól formularza.
- Doładowanie i odtworzenie filmu (nie jest to liczone jako odsłona strony – jest to tylko odtworzenie streamu video).
- Dodanie/pojawienie się nowego komentarza.
- W przypadku czatów, stron serwisów randkowych i stron komunikatorów pierwszego otwarcia pokoju czatu lub rozmowy z innym użytkownikiem oraz kolejnych przejść do otwartych już wcześniej poszczególnych pokoi lub rozmów, o ile nie powoduje to zmiany wszystkich widocznych treści na stronie na nowe.
- Zmiana wyglądu strony poprzez przeciąganie poszczególnych bloków treści – zwykle pogrupowanych wg kategorii – w celu zmiany ich położenia na stronie – dostosowanie wyglądu strony do użytkownika.
- Rozwijanie poszczególnych menu danej kategorii w bardziej szczegółowe zarówno przy najechaniu wskaźnikiem myszki jak i przy kliknięciu.
- Wysuwanie się/Chowanie menu – zakładka z opcjami do wyboru pojawia się lub znika po najechaniu lub kliknięciu wskaźnikiem myszki na określony punkt.
- Dostosowanie wyglądu strony poprzez personalizację – wybór czy dany blok tematyczny ma być widoczny/niewidoczny.
- Dodawanie/usuwanie poszczególnych tematów lub np. programów TV do kategorii „ulubione”.
- Zmiana wyglądu strony poprzez wybór kolorów, motywów graficznych, czcionki, itp.
- Zamknięcie reklamy widocznej na stronie umożliwiające zobaczenie treści, które reklama zakrywała
- Zmiana wielkości czcionki na stronie.
- W poczcie dodawanie/usuwanie folderów (nie dodaje nowych treści – konfiguruje jedynie usługę, w celu dostosowania do potrzeb użytkownika).
- Wybór opcji dostosowujących działanie strony do użytkownika, np. zaznaczenie opcji „śledź bieżące wydarzenia” powoduje automatyczne doładowywanie informacji o nowych artykułach na stronie.
- Automatycznego doładowywania treści, np. w relacjach „na żywo”.
- Zamknięcie okna informacyjnego pojawiającego się po wejściu na stronę.
- Wybór miasta, z jakiego będą wyświetlać się informacje lokalne.
- Wybór, jakie elementy strony mają być widoczne, np. w przypadku map: budynki, punkty usługowe, natężenie ruchu.
- Zmiana skali mapy, zarówno przez użycie suwaka, jak i poprzez przycisk przewijania myszki.
- Oznaczenie artykułu/komentarza lubię/nie lubię.
- Odświeżenie przy przeciągnięciu ekranu z góry na dół tzw. pull to refresh.
- Obrócenie ekranu z pozycji poziomej na pionową i odwrotnie.
- Przywołanie na nowo na ekran okienka przeglądarki, która była cały czas wczytana do pamięci urządzenia, ale nie była widoczna na ekranie.
- Zmiana sposobu wyświetlania treści, np. od najnowszej itp.
- Jeśli po zakończeniu odtwarzania materiału video użytkownik zostaje automatycznie odesłany do artykułu to nie jest to odsłona.

- Oznaczanie listów w poczcie elektronicznej kategoriami, np. „przeczytane”, „ważne”, „spam”.
- Korzystanie z funkcji poczty – pokazywanie/ukrywanie kontaktów, dodawanie/usuwanie folderów, zmiana nazwy folderu, włączanie/wyłączanie filtrów automatycznie przenoszących listy do poszczególnych folderów.
- Zmiana sposobu pokazywania wiadomości - lista lub wątek.
- Umieszczanie wiadomości w poszczególnych folderach – zarówno poprzez przeciąganie, jak i przy pomocy wyboru z menu kontekstowego. (uznawane za konfigurację dostosowującą usługę poczty do potrzeb użytkownika).

## 5. Informacje wymagające zgłoszenia

Właściciele witryn zobowiązani są zgłaszać *firmie badawczej*:

- 1) Adresy URL *stron automatycznie odświeżanych* (np. wykresy giełdowe) wraz z *identyfikatorami skryptów*, którymi zostały oskryptowane. Zgłoszenie powinno nastąpić niezwłocznie po umieszczeniu *skryptów audytowych*.
- 2) Wszelkie odstępstwa od reguł oskryptowania umieszczonych w punkcie „Ogólne reguły umieszczania skryptów”.
- 3) Wszystkie adresy URL stron, na których właściciel witryny zamierza prowadzić pomiar *odślon częściowych*. Implementacja i rozpoczęcie pomiaru odślon częściowych może odbyć się tylko po weryfikacji zgłoszenia przez *firmę badawczą* i wyłącznie w ramach przesłanych, pisemnych wytycznych.

## 6. Skrypty audytowe

Na podstawie *identyfikatora skryptu firma badawcza* może określić, czy skrypt jest *skryptem audytowym*, czy też innym *skryptem Gemius*.

Poniżej znajdują się wzory *skryptów audytowych*. Skryptem rekomendowanym do oskryptowania wszystkich stron jest *podstawowy skrypt audytowy*.

Każdy ze skryptów odwołuje się do specjalnego serwera badawczego, tzw. hitCollectora, który znajduje się pod adresem <http://PREFIX.hit.gemius.pl/>, gdzie *PREFIX* jest subdomeną hitCollectora (np. <http://gapl.hit.gemius.pl/>).

## 7. Podstawowy skrypt audytowy (odśloni pełne)

*Podstawowy skrypt audytowy* odpowiada zarówno za zbieranie danych o ruchu na witrynie internetowej jak i emisję kwestionariuszy Badania Mediapanel. Składa się ze skryptu zliczającego umieszczonego bezpośrednio w źródle strony (tzw. „wklejka”) oraz skryptu głównego, do którego odwołuje się ten skrypt zliczający. W miejsce parametru „IDENTYFIKATOR” należy wpisać odpowiedni dla konta danego podmiotu *identyfikator skryptu* pobrany np. z interfejsu PRISM interfejsu gacc lub dostarczony przez firmę badawczą. Jeżeli skrypt główny jest pobierany z hitcollectora to należy jedynie w skrypcie zliczającym określić nazwę hitcollectora wstawiając odpowiedni PREFIX (np. gapl) w wyrażeniu PREFIX.hit.gemius.pl.

Skrypt główny xgemius.js jest przystosowany zarówno do obsługi protokołu http jak i https. Jeżeli skrypt główny jest pobierany z hitcollectora to na stronach po protokole https odwołanie również musi być wpisane z https czyli src="https://PREFIX.hit.gemius.pl/xgemius.js". Dynamiczna obsługa zmiany protokołu wykonywana jest automatycznie przez skrypt audytowy. Podstawowy skrypt audytowy i odśloną przesyłaną w momencie wykonania skryptu służy do pomiaru odślon pełnych.



## 8. Skrypt zliczający (tzw. „wklejka” do umieszczenia w źródle strony):

```
<script type="text/javascript">
var pp_gemius_identifier = 'IDENTYFIKATOR';
// lines below shouldn't be edited
function gemius_pending(i) { window[i] = window[i] || function() {var x = window[+'_pdata'] = window[+'_pdata']
|| []; x[x.length]=Array.prototype.slice.call(arguments, 0);}};
(function(cmds) { var c; while(c = cmds.pop()) gemius_pending(c)}(['gemius_cmd', 'gemius_hit', 'gemius_event',
'gemius_init', 'pp_gemius_hit', 'pp_gemius_event', 'pp_gemius_init']);
window.pp_gemius_cmd = window.pp_gemius_cmd || window.gemius_cmd;
(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+'://PREFIX.hit.gemius.pl/xgemius.js'; s.parentNode.insertBefore(gt,s);}
catch (e) {}})(document,'script');
</script>
```

### Uwagi:

Wyrażenie "IDENTYFIKATOR" należy zastąpić odpowiednim identyfikatorem skryptu pobranym z interfejsu PRISM, interfejsu gacc lub dostarczonym przez Firmę Badawczą.

Wyrażenie "PREFIX" w wyrażeniu PREFIX.hit.gemius.pl należy zastąpić odpowiednią dla danego podmiotu nazwą hitcollector'a (np. gapl.hit.gemius.pl).

## 9. Skrypt główny (plik *xgemius.js*):

Aktualna, obowiązująca w danym momencie jako oficjalna wersja skryptu głównego w wersji służącej do audytu site-centric jest zawsze umieszczona na hitcollectorach firmy badawczej w lokalizacji <http://PREFIX.hit.gemius.pl/xgemius.js> (gdzie PREFIX oznacza nazwę hitcollector'a np. gapl).

## 10. Ładowanie zminifikowanego skryptu głównego.

W celu poprawy wydajności witryny, umożliwiamy przejście na zminifikowany skrypt główny. W tym celu należy zmienić ścieżkę źródłową skryptu w skrypcie zliczającym (tzw. „wklejce”), umieszczonym w kodzie witryny na *xgemius.min.js* zamiast *xgemius.js*:

```
<script type="text/javascript">
var pp_gemius_identifier = 'IDENTYFIKATOR';
// 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]=Array.prototype.slice.call(arguments, 0);};};
(function(cmds) { var c; while(c = cmds.pop()) gemius_pending(c)})(['gemius_cmd', 'gemius_hit',
'gemius_event', 'gemius_init', 'pp_gemius_hit', 'pp_gemius_event', 'pp_gemius_init']);
window.pp_gemius_cmd = window.pp_gemius_cmd || window.gemius_cmd;
(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+'://PREFIX.hit.gemius.pl/xgemius.min.js';
s.parentNode.insertBefore(gt,s);} catch (e) {}})(document,'script');
</script>
```

## 11. Odśłony częściowe

Skrypt audytowy zawiera wbudowaną funkcję `pp_gemius_hit` służącą do przesyłania odślon wyczerpujących definicję odślon częściowej. Wywołanie funkcji dokonuje się poprzez umieszczenie w kodzie zapisu postaci `pp_gemius_hit('IDENTYFIKATOR');`

Odśłony częściowe na stronie zawierającej elementy flashowe bądź ajaxowe, na której został wczytany skrypt główny można przesłać odpowiednio.

## 12. Wywołanie dla stron z elementami AJAX:

```
<a href="TARGET_ADDRESS" onclick= "pp_gemius_hit ('IDENTYFIKATOR') ">LINK</a>
```

## 13. Wywołanie dla stron z elementami FLASH:

```
on (release){
getURL("javascript: pp_gemius_hit(IDENTYFIKATOR)");}
```

W obydwu przypadkach wyrażenie `IDENTYFIKATOR` należy zastąpić odpowiednim identyfikatorem skryptu audytowego. Natomiast wyrażenia `TARGET_ADDRESS` i `LINK` należy zastąpić odpowiednimi adresami i opisem w zgodzie z wytycznymi przesłanymi przez firmę badawczą w odpowiedzi na zgłoszenie przez właściciela witryny potrzeby pomiaru odślon częściowych. Jednocześnie w obydwu tych przypadkach musi nastąpić wcześniejsze wczytanie skryptu głównego i przesłanie odślon pełnej związanej z załadowaniem treści witryny.

W przypadku, gdy witryna wczytuje jednocześnie elementy ajaxowe lub flashowe, które przesyłają odśłony częściowe, uczestnik badania powinien usunąć `IDENTYFIKATOR` z wklejki skryptu zliczającego i nie przysłać odślon pełnej

```
<script type="text/javascript">
// lines below shouldn't be edited
function gemius_pending(i) { window[i] = window[i] || function() {var x = window[+'_pdata'] =
window[+'_pdata'] || []; x[x.length]=Array.prototype.slice.call(arguments, 0)};};
(function(cmds) { var c; while(c = cmds.pop()) gemius_pending(c)})(['gemius_cmd', 'gemius_hit',
'gemius_event', 'gemius_init', 'pp_gemius_hit', 'pp_gemius_event', 'pp_gemius_init']);
window.pp_gemius_cmd = window.pp_gemius_cmd || window.gemius_cmd;
(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+'://PREFIX.hit.gemius.pl/xgemius.js';
s.parentNode.insertBefore(gt,s);} catch (e) {}})(document,'script');
</script>
```

## 14. Oskryptowanie w przypadku automatycznego odświeżania stron internetowych

Wydawcy, którzy na swoich witrynach korzystają z mechanizmu odświeżania strony przy powrocie do wczytanej już wcześniej strony, przechowywanej w nieaktywnej zakładce w przeglądarce, zobowiązani są do wywoływania *skryptu audytowego* w sposób opisany w tym rozdziale. Inne oskryptowanie spowoduje zaniżenie czasu lub zawyżenie liczby odstępów w badaniu Mediapanel dla tej witryny.

Automatyczne odświeżenie strony, po powrocie do nieaktywnej zakładki, zgodnie z regułami oskryptowania nie może być uwzględniane w liczbie odstępów witryny w wynikach badania Mediapanel. W takiej sytuacji również nie można odświeżyć strony z całkiem usuniętym skrypcem audytowym, ponieważ zatrzymane zostałyby liczenie czasu na potrzeby badania. Aby odstęp, która w takim momencie zostanie wysłana, nie była uwzględniona w wynikach badania, musi być więc wysłana w specjalny sposób - za pomocą funkcji `pp_gemius_event`, która jest zawarta w skrypcie `xgemius.js`

Przy odświeżeniu strony, należy więc załadować skrypt główny bez zdefiniowanego identyfikatora `pp_gemius_identifier` - oznacza to całkowite usunięcie linijki kodu, który tę zmienną definiuje.

Konieczne jest też dodanie linii z wywołaniem funkcji `pp_gemius_event` wraz z określeniem dowolnego identyfikatora skryptu z projektu audytowego dla danej witryny. Może to być ten sam identyfikator na który są wysyłane standardowe odstępy lub może to być inny, dedykowany identyfikator, tak aby łatwo było odróżnić te odstępy od odstępów uwzględnianych w wynikach badania Mediapanel. Odstępy takie muszą być wysyłane jako hity typu `view` (`et=view`) a nie `action` (`et=action`), Typ `view` jest domyślnym rodzajem hita wysyłanego przez `pp_gemius_event`, więc przy użyciu skryptu przedstawionego poniżej ten warunek będzie spełniony.

Skrypt zliczający na stronie powinien wyglądać następująco:

```
<script type="text/javascript">
// lines below shouldn't be edited
function gemius_pending(i) { window[i] = window[i] || function() {var x = window['+_pdata'] =
window['+_pdata'] || []; x[x.length]=Array.prototype.slice.call(arguments, 0);}};
(function(cmds) { var c; while(c = cmds.pop()) gemius_pending(c)}(['gemius_cmd', 'gemius_hit',
'gemius_event', 'gemius_init', 'pp_gemius_hit', 'pp_gemius_event', 'pp_gemius_init']);
window.pp_gemius_cmd = window.pp_gemius_cmd || window.gemius_cmd;
(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+'://PREFIX.hit.gemius.pl/xgemius.js';
s.parentNode.insertBefore(gt,s);} catch (e) {}})(document,'script');
pp_gemius_event('IDENTYFIKATOR');
</script>
```

Jeżeli automatyczne odświeżanie strony występuje też w innych przypadkach niż w momencie powrotu do nieaktywnej zakładki oraz nie jest możliwe odświeżanie tylko jej elementów np. samego wykresu giełdowego lub tabelki wyników sportowych umieszczonych w ramce osadzonej na stronie to w takim przypadku w momencie automatycznego wczytania strony, ona również powinna być oskryptowana w powyższy sposób.

## 15. Reguła generowania odsłony na dwa oddzielne projekty

Rekomendowaną przez Gemius opcją skryptowania witryny jest standardowa wersja w której wydawca korzysta z jednego projektu audytowego w załączonym skrypcie zliczającym (tzw. „wklejce”) czyli 1 zdarzenie generowane jest tylko na 1 projekt.

Jeżeli jednak wydawca uzna za konieczne oskryptowanie witryny dwoma projektami – audytowym i nieaudytowym powinien skorzystać ze sposobu opisanego poniżej. Zagwarantuje to prawidłowe zliczanie się czasu w badaniu Mediapanel.

Do skryptu głównego należy odwołać się tylko raz, przy czym ma to być skrypt główny w wersji audytowej czyli *xgemius.js*. Odwołać się do niego należy z identyfikatorem *pp\_gemius\_idenfifier* odpowiednim dla projektu audytowego. Do skryptu głównego w wersji nieaudytowej (*gemius.js*) w ogóle nie należy się odwoływać.

Odsłona wygenerowana na projekt nieaudytowy przesłana zostaje za pomocą wbudowanej w skrypt *xgemius.js* funkcji *pp\_gemius\_hit* lub *pp\_gemius\_event* (z parametrem „action”, dla zdarzeń innych niż odsłony). Funkcja ta powinna zostać dodana na końcu skryptu zliczającego.

Skrypt zliczający powinien wyglądać następująco:

```
<script type="text/javascript">
var pp_gemius_identifier = 'IDENTYFIKATOR_PROJEKTU_AUDYTOWEGO';
// 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]=Array.prototype.slice.call(arguments, 0)};};
(function(cmds) { var c; while(c = cmds.pop()) gemius_pending(c)}(['gemius_cmd', 'gemius_hit',
'gemius_event', 'gemius_init', 'pp_gemius_hit', 'pp_gemius_event', 'pp_gemius_init']);
window.pp_gemius_cmd = window.pp_gemius_cmd || window.gemius_cmd;
(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+'://PREFIX.hit.gemius.pl/xgemius.js';
s.parentNode.insertBefore(gt,s);} catch (e) {}})(document,'script');
pp_gemius_hit('IDENTYFIKATOR_PROJEKTU_NIEAUDYTOWEGO');
</script>
```

## 16. Pomiar odstępów dla Single-page Applications

Skrypt główny Gemius: „xgemius.js” zawiera funkcję „pp\_gemius\_hit”, która jeśli zostanie wywołana, generuje hit traktowany jako odstęp strony. Rozwiązanie to ma zastosowanie w pomiarze aplikacji webowych oraz stron, które wchodzi w interakcję z użytkownikiem dynamicznie przepisyując bieżącą stronę internetową nowymi danymi z serwera WWW, zamiast stosowania domyślnej metody ładowania przez przeglądarkę internetową całych nowych stron.

Aby umożliwić pomiar za pomocą tej funkcji, należy najpierw wykonać skrypt główny:

```
<script type="text/javascript">
var pp_gemius_identifier = 'IDENTYFIKATOR';
// 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]=Array.prototype.slice.call(arguments, 0);}};
(function(cmds) { var c; while(c=cmds.pop()) gemius_pending(c)}(['gemius_cmd',
'gemius_hit', 'gemius_event', 'gemius_init', 'pp_gemius_hit', 'pp_gemius_event',
'pp_gemius_init']);
window.pp_gemius_cmd = window.pp_gemius_cmd || window.gemius_cmd;
(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+'://PREFIX.hit.gemius.pl/xgemius.js';
s.parentNode.insertBefore(gt,s);} catch (e) {}})(document,'script');
</script>
```

### Wywołanie funkcji pp\_gemius\_hit

Wywołanie funkcji powinno zawierać identyfikator Gemius oraz wszystkie niezbędne extraparametry:

```
gemius_hit('IDENTYFIKATOR', 'ekstraparametry');
```

Identyfikator można dołączyć do konkretnej akcji użytkownika - poniżej przedstawiamy przykład wywołania funkcji pp\_gemius\_hit wraz z kliknięciem wykonanym przez użytkownika:

```
<button onclick= "gemius_hit ('nLuQbYi8G0G3.6QAI21YEJew.ohgBu7_mtmDRiEvdFP.C7',
'lan=NL') ">Link testowy</a>;
```

## Usuwanie zdublowanych odsłon

Jeśli funkcja `pp_gemius_hit` jest wywoływana jednocześnie z ładowaniem skryptu głównego (np. podczas ładowania określonego elementu strony na początku jej przeglądania), może być konieczne usunięcie definicji identyfikatora z kodu skryptu mierzącego, aby uniknąć podwójnego zliczania odsłon.

Oznacza to usunięcie fragmentu kodu `var pp_gemius_identififier = 'IDENTYFIKATOR'`; i pozwala na prawidłowy pomiar pojedynczej odsłony strony.

```
<script type="text/javascript">
// 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]=Array.prototype.slice.call(arguments, 0);}};

(function(cmds) { var c; while(c=cmds.pop()) gemius_pending(c)}(['gemius_cmd',
'gemius_hit', 'gemius_event', 'gemius_init', 'pp_gemius_hit', 'pp_gemius_event',
'pp_gemius_init']);

window.pp_gemius_cmd = window.pp_gemius_cmd || window.gemius_cmd;

(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+'://PREFIX.hit.gemius.pl/xgemius.js';
s.parentNode.insertBefore(gt,s);} catch (e) {}})(document,'script');
```

W rezultacie odsłona strony jest generowana tylko z funkcji `hit`, bez dodatkowej odsłony wykonywanej przy standardowym ładowaniu Skryptu Głównego. Uczestnik badania musi zadbać o to, aby wysyłać odsłonę za każdym razem, kiedy powinna zostać wysłana.

## Przekazywanie odsyłacza

Aby dokładnie zmierzyć Źródło Ruchu z jakiego pochodzi odsłona w SPA, należy postępować według poniższej instrukcji.

Za każdym razem, gdy wysyłane są odsłony strony lub zdarzenia, skrypt Gemius pobiera wartość `document.referrer` i przekazuje ją w odpowiednim parametrze. Parametr ten jest kluczowy dla poprawnego rozpoznawania Źródeł Ruchu w badaniu Prism i Mediapanel.

W przypadku Single-page applications, należy samemu zmienić adres URL, a przed wywołaniem zmiany adresu, użyć funkcji `gemius_cmd`. Najczęściej, wywołania zmieniające URL są związane z użyciem **History API**:

- `history.pushState`
- `history.replaceState`
- `history.back`
- `history.forward`
- `history.go`

`window.location`, przy zmianie fragmentu adresu URL(hash):

- `location.hash = "..."`
- `location.href = "#..."`
- `location = "#..."`
- `location.assign("#...")`
- `location.replace("#...")`

Aby wysłać zdarzenie z prawidłowym odsyłaczem, należy dodać do skryptu dodatkowy kod.

Przed zmianą adresu URL:

```
gemius_cmd('referrer_override', window.location.href);
```

Po zmianie adresu URL:

```
gemius_hit('IDENTYFIKATOR', 'ekstraparametry');
```

W celu ustawienia poprawnego odsyłacza, podczas korzystania z przycisków nawigacji przeglądarki (wstecz, dalej), należy dodać kod wykrywających ich użycie.

Przykład linku SPA zmieniającego adres URL za pomocą **History API** i wykrywania użycia przycisku **back/forward**:

```
// Link SPA zmieniający adres URL za pomocą History API
history.replaceState({referrer:document.referrer}, "");

someLink.onClick = function() {
    pp gemius_cmd('referrer_override', window.location.href);
    history.pushState({q: query, referrer: window.location.href}, "", "?" + query);
    pp gemius_hit('IDENTYFIKATOR', 'type=simple');
}

// detekcja przycisku back/forward
window.addEventListener("popstate", function(ev) {
    pp gemius_cmd('referrer_override', ev.state.referrer);
});
```

### Szczególne przypadki ignorowania `referrer_override`

1. Zmiana odsyłacza przed opóźnioną inicjalizacją skryptu nie jest możliwa. Wszystkie zdarzenia które oczekują w kolejce do wysłania, otrzymują odsyłacz z momentu inicjalizacji skryptu.
2. Wywołanie `referrer_override` podczas zdarzeń `pageshow` i `pagehide` nie jest możliwe. Obecne rozwiązanie `back-forward cache` używa tych zdarzeń, a po stronie przeglądarki nie jest gwarantowana prawidłowa kolejność ich wykonywania.
3. Wartość `referrer_override` musi posiadać taką samą wartość `origin` jak strona internetowa (`window.location`). W innym przypadku zostanie zignorowana.



## 17.Rekomendacja back-forward cache

Skrypt Gemius nasłuchuje zdarzenia pageshow. Jeśli witryna jest ładowana z pamięci podręcznej, automatycznie wysyłana jest **jedna** odsłona strony (odpowiadająca **ostatniemu** hitowi odsłonowemu (PV) z danej witryny).

Jeśli na stronie wykorzystywana jest funkcja pp\_gemius\_hit w **niestandardowy sposób**, np. dla **jednego wyświetlenia strony** w serwisie wysyłany jest **więcej niż jeden hit odsłonowy** (zliczenia kierowane do osobnych projektów - dla różnych identyfikatorów Gemius), należy:

- użyć **gemius\_disable\_back\_cache\_hits = true**, aby wyłączyć wysyłanie automatycznych hitów ze strony Gemius, gdy strona jest ładowana z pamięci podręcznej
- obsłużyć wysyłanie hitów odsłonowych po swojej stronie, gdy strona jest ładowana z pamięci podręcznej

## 15.Pomiar czasu

W badaniu jednoźródłowym do pomiaru czasu służą Sonary. Skrypt xgemius.js od wersji 3.8 przesyła hity sonarowe umożliwiające dokładny pomiar czasu (et=smpsonar). W każdej sekundzie dla aktywnego okna przeglądarki skrypt dokonuje losowania ze stałym prawdopodobieństwem, czy przesłać hit sonarowy. Wartość prawdopodobieństwa w Polsce w Badaniu Jednoźródłowym wynosi 1/40.

## 16.Zasady dotyczące skryptów

1. Wymagamy korzystania z asynchronicznej wersji skryptu. Podane wyżej „wklejki” mają postać asynchroniczną.
2. Zalecamy, aby skrypt zliczający odwoływał się do skryptu głównego umieszczonego na naszym hitcollectorze, a nie do skryptu głównego przechowywanego lokalnie.
3. Klienci, którzy z jakiegoś ważnego powodu nie mogą odwoływać się do skryptu głównego z naszego hitcollectorza powinni regularnie, synchronizować swój lokalny skrypt z wersją dostępną na hitcollectorze.
4. Zalecamy, aby skrypt główny nie był integrowany z innymi skryptami w bibliotekach skryptowych lub wczytywany przy pomocy użycia tag managerów mogących ingerować w sposób działania skryptów lub utrudniać dostęp do jego kodu.
5. Modyfikacja skryptu głównego i skryptu zliczającego (poza identyfikatorami i prefixem hitcollectorza) jest zabroniona.
6. Testy wewnętrzne skryptów audytowych wykonywane przez klientów powinny być prowadzone wyłącznie z użyciem odrębnych, nie audytowych kont.

**Ważne:** Instrukcja na temat przekazywania zgód użytkownika (CMP) znajduje się w dokumencie „**Instrukcja przekazywania zgód użytkownika**”. Dokument jest do pobrania na stronie <https://media-panel.pl/pl/dolacz-do-badania/>