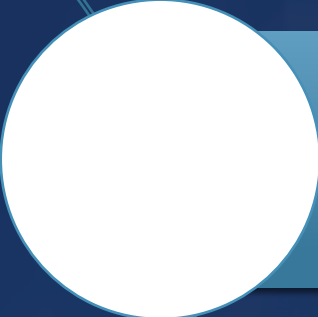





ТЕСТИРОВЩИК
ПРОГРАММНОГО
ОБЕСПЕЧЕНИЯ

КУРС «РУЧНОЕ ТЕСТИРОВАНИЕ»

13.ТЕСТИРОВАНИЕ WEB - ПРИЛОЖЕНИЙ



Web-приложение. Тестирование web-приложений



Инструменты разработчика для тестировщика

ВЕБ-ПРИЛОЖЕНИЕ

- ✓ на данный момент в сети Интернет действует более миллиарда сайтов, и пользуются ими более 3,5 млрд. людей по всему миру (по данным Международного союза электросвязи на июль 2016 года);
- ✓ в России более 70% взрослого населения являются интернет-пользователями, а общий оборот средств на российском рынке интернет-торговли за первое полугодие 2016 года вырос на 26% в сравнении с аналогичным периодом 2015 года и достиг 405 млрд. рублей.

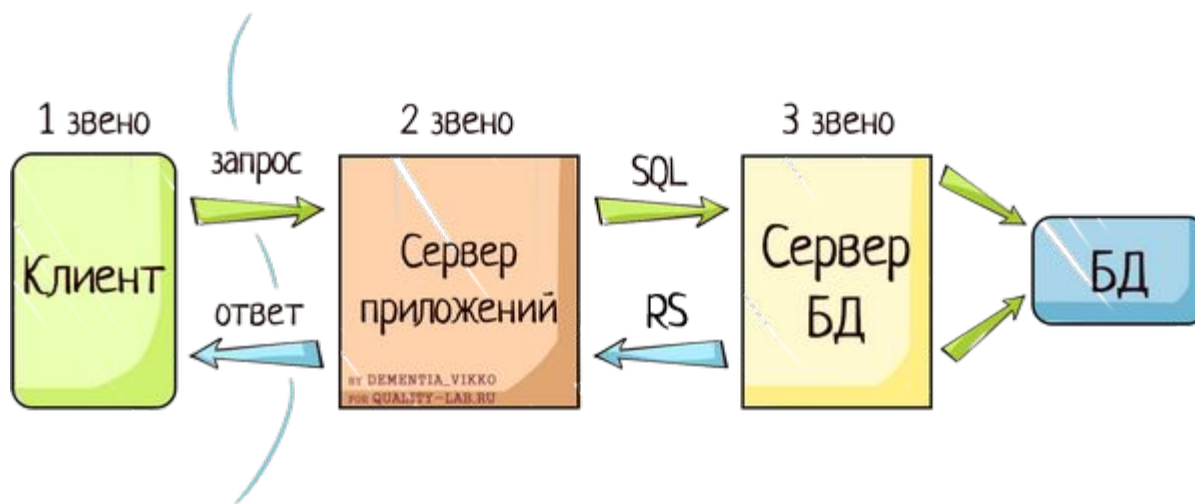
При взгляде на эти баснословные цифры становится понятным, почему в мире разрабатывается так много новых веб-приложений. Этот процесс приводит к необходимости привлечения большого количества специалистов.



ВЕБ-ПРИЛОЖЕНИЕ

Веб-приложение – это (вспоминаем предыдущее занятие)

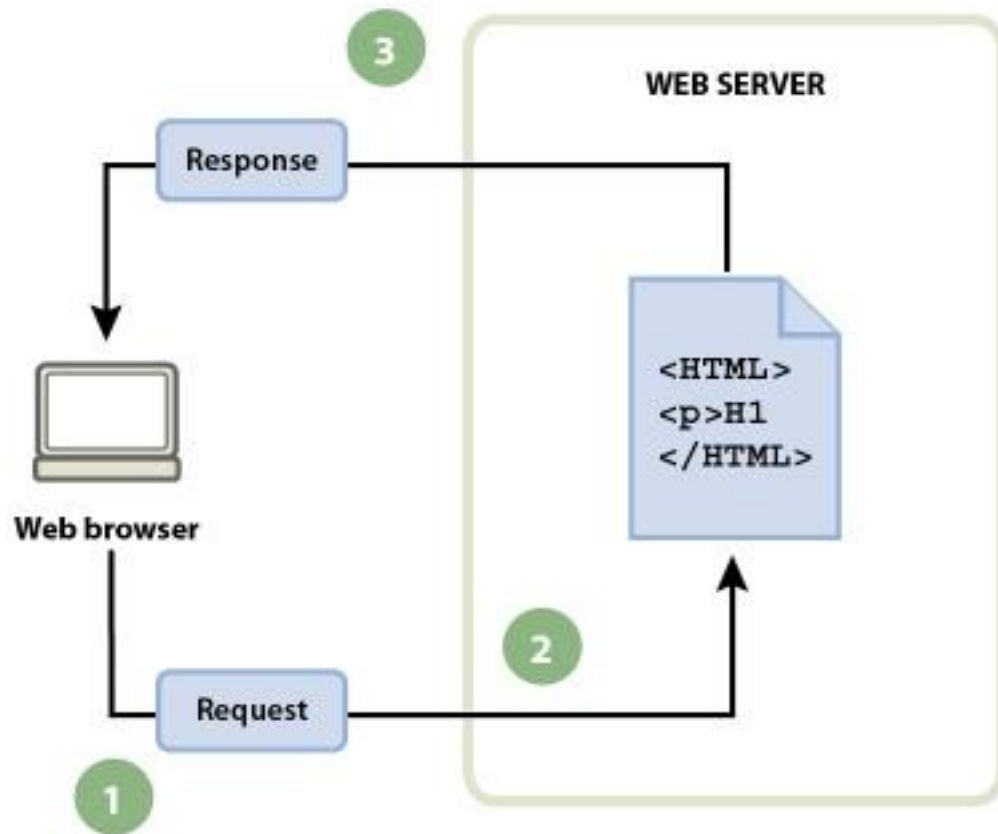
Веб-приложение представляет собой веб-сайт, на котором размещены страницы с частично либо полностью несформированным содержимым. Также можно в отдельный класс выделить мобильные веб-приложения.



ВЕБ-ПРИЛОЖЕНИЕ

Работа со статическими страницами

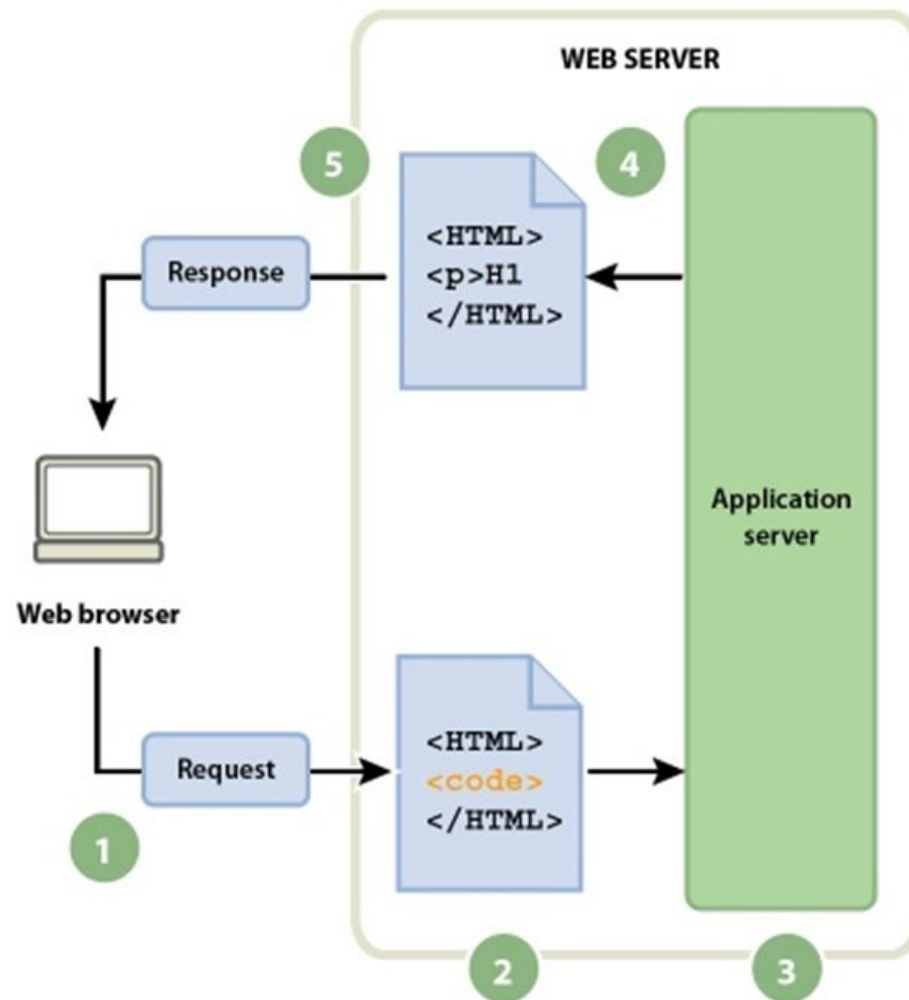
1. Веб-браузер запрашивает статическую страницу.
2. Веб-сервер находит страницу.
3. Веб-сервер отправляет страницу запросившему ее браузеру.



ВЕБ-ПРИЛОЖЕНИЕ

Работа с динамическими страницами

1. Веб-браузер запрашивает динамическую страницу.
2. Веб-сервер находит страницу и передает ее серверу приложений.
3. Сервер приложений просматривает страницу на наличие инструкций и выполняет ее создание.
4. Сервер приложений возвращает подготовленную страницу на веб-сервер.
5. Веб-сервер отправляет подготовленную страницу запросившему ее браузеру.



ВЕБ-ПРИЛОЖЕНИЕ

Стандарты для веб приложений могут делиться на:

- **функциональные** — основанные на выполнении основной функции приложения;
- **вспомогательные** — основанные на критериях качества:
 - ✓ *Надежность.* Приложение должно работать с заданными характеристиками и установленной скоростью вне зависимости от количества пользователей.
 - ✓ *Быстродействие.* В любых условиях среднее время обработки запроса системой не должно превышать заданных параметров.
 - ✓ *Безопасность.* Включает уровни прав доступа, авторизацию и аутентификацию.
 - ✓ *Масштабируемость.* Если в будущем будет принято решение добавить компоненты, система должны быть способна увеличить производительность с учетом новых условий.

ВЕБ-ПРИЛОЖЕНИЕ

Тестирование веб приложений включает:

- Функциональное тестирование.
- Тестирование безопасности.
- Нагрузочное тестирование.
- Кроссбраузерное тестирование (рассмотрим его подробнее!)

ВЕБ-ПРИЛОЖЕНИЕ

Кросс-браузерное тестирование представляет собой процесс тестирования веб-приложений в нескольких браузерах.

В настоящее время пользователи имеют широкий выбор браузеров для доступа к веб-приложениям, поэтому в современных условиях стало крайне важным выполнять кросс-браузерное тестирование. В разных браузерах компоненты приложений могут вести себя по-разному.

Такое поведение приложения может быть вызвано рядом факторов:

- Разработанное веб-приложение может быть не адаптивно под тот или иной браузер или его версию.
- Неверно были применены стандарты, по которым разрабатывается сам браузер.
- Были допущены ошибки при разработке браузера.
- У пользователя установлен какой-либо плагин или надстройка, вызывающие ошибки веб-приложения.

ВЕБ-ПРИЛОЖЕНИЕ

Из-за разной работы браузеров или ошибок, допущенных в нем, могут возникать дефекты в Вашем продукте. Часто встречающиеся дефекты:

- **Верстка.**

Наиболее распространенная ошибка в различных браузерах. Разработчики часто создают приложение и проверяют его в одном, наиболее удобном для них, браузере. Но у пользователей может быть установлена другая версия браузера, в котором «красивая картинка» разработчика может выглядеть совсем некрасиво у пользователя. Такие ошибки чаще всего не являются критичными, но неприятное впечатление о вашем продукте могут оставить.

ВЕБ-ПРИЛОЖЕНИЕ

- **Навигация.**

Бывают ситуации, когда в одном из браузеров ссылка не работает, как было запланировано, либо не работает вовсе. Такие ошибки могут негативно отразиться на Вашем продукте. Когда клиент не находит нужный раздел или не может перейти на другую страницу, чтобы завершить действие, это доставляет неудобства и раздражает. Как результат – потеря клиента.

- **Ошибки JavaScript.**

Такие ошибки имеют высокий приоритет. Неработоспособность JavaScript в одном из браузеров может привести к потере заказа, клиента, или к потере документа, например, если у вас система электронного документооборота; к невозможности создания заявки, если у вас система заявок и т.д.

ВЕБ-ПРИЛОЖЕНИЕ

Десктопные браузеры:

- Chrome
- Firefox
- IE
- Safari
- Edge
- Opera
-

Мобильные браузеры:

- Chrome
- Safari
- UC Browser
- Opera
- Samsung Internet
- Android

Движки браузеров:

- Trident
- Gecko
- KHTML
- WebKit
- Presto
- Blink.
- Edge

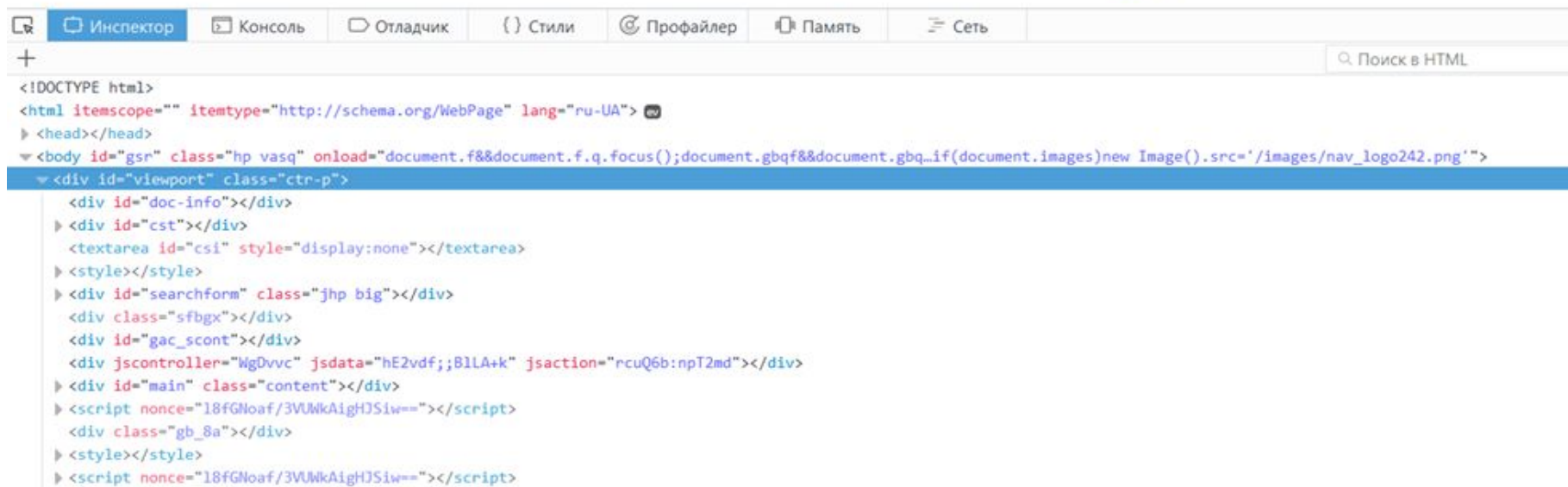
ИНСТРУМЕНТЫ РАЗРАБОТЧИКА

Инструменты разработки позволяют:

- Просматривать элементы, соответствующие определённому HTML коду (например, какой-нибудь заголовок).
- Получить общий CSS используемый на странице и какой CSS применяется к элементу.
- Модифицировать CSS в реальном времени и визуально увидеть ваши изменения в браузере.
- Увидеть HTTP запросы, производимые браузером.
- Запускать JavaScript код в середине содержимого страницы.
- Определять узкие места в производительности и производить её измерение.
- Изменять ресурсы оффлайн, чтобы понять какие данные, что запрашивает страница, хранятся локально.

ИНСТРУМЕНТЫ РАЗРАБОТЧИКА

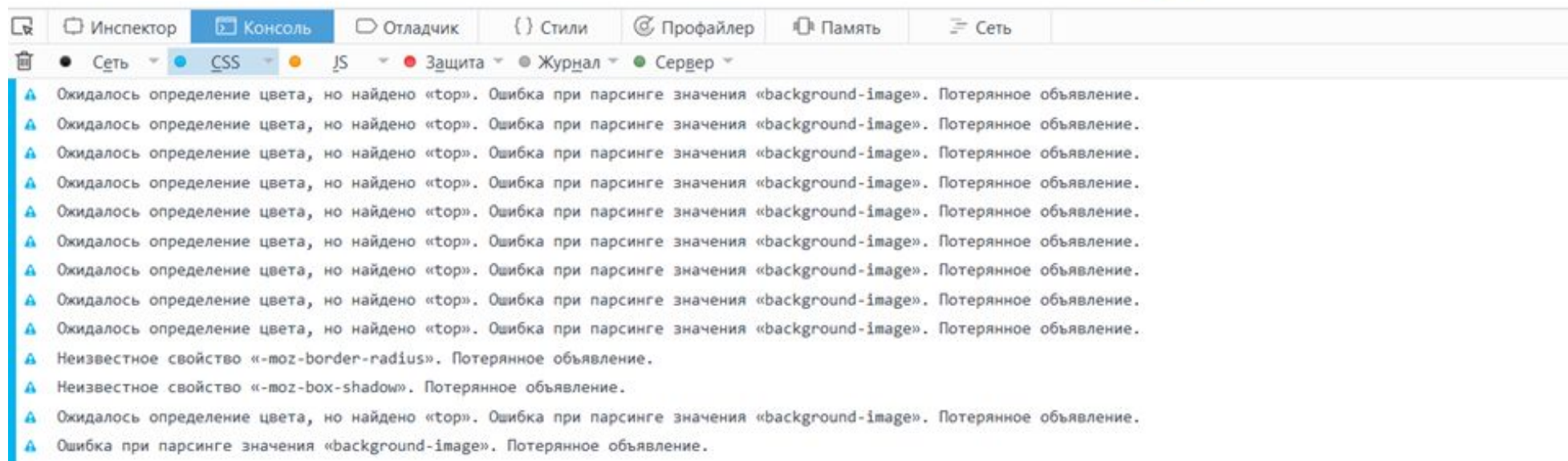
Инспектор - позволяет видеть HTML-код и CSS, который применён к каждому элементу на странице. Также позволяет в реальном времени редактировать как HTML, так и CSS. Внесённые изменения можно увидеть непосредственно в окне браузера.

A screenshot of a web browser's developer tools interface. The 'Инспектор' (Inspector) tab is active, showing the HTML structure of the page. The selected element is a <div id="viewport" class="ctr-p">. The code is displayed in a light blue background with syntax highlighting. The toolbar at the top includes icons for 'Инспектор', 'Консоль', 'Отладчик', 'Стили', 'Профайлер', 'Память', and 'Сеть'. A search bar for HTML is visible on the right side of the toolbar.

```
+
<!DOCTYPE html>
<html itemscope="" itemtype="http://schema.org/WebPage" lang="ru-UA">
  <head></head>
  <body id="gsr" class="hp vasq" onload="document.f&&document.f.q.focus();document.gbqf&&document.gbq.if(document.images)new Image().src='/images/nav_logo242.png'">
    <div id="viewport" class="ctr-p">
      <div id="doc-info"></div>
      <div id="cst"></div>
      <textarea id="csi" style="display:none"></textarea>
      <style></style>
      <div id="searchform" class="jhp big"></div>
      <div class="sfbgx"></div>
      <div id="gac_scont"></div>
      <div jscontroller="WgDvvc" jsdata="hE2vdf;B1LA+k" jsaction="ncuQ6b:npT2md"></div>
      <div id="main" class="content"></div>
      <script nonce="18fGNoaf/3VUNkAighJ5iw=="></script>
      <div class="gb_8a"></div>
      <style></style>
      <script nonce="18fGNoaf/3VUNkAighJ5iw=="></script>
```

ИНСТРУМЕНТЫ РАЗРАБОТЧИКА

Консоль - инструмент, где выводятся сообщения и ошибки JavaScript, CSS и другие. Она позволяет загружать JavaScript вопреки порядку загрузки скрипта в браузере и докладывает об ошибках, как только браузер пытается выполнить Ваш код.



ИНСТРУМЕНТЫ РАЗРАБОТЧИКА

Отладчик JavaScript - инструмент для отладки JavaScript, если он не работает, как ожидалось. Он позволяет Вам загружать JavaScript вопреки порядку загрузки скрипта в браузере и докладывает об ошибках, как только браузер пытается выполнить код.



```
1 /* JS */ gapi.loaded_0(function(_){var window=this;
2 var ha,ia,ja,ma,sa,na,ta,ya,Ja;_.ea=function(a){return function(){return _.da[a].apply(this,arguments)}};_.DumpException=function(a){
3 ma=function(){var a=0;return function(b){return"jscomp_symbol_"+(b||"")+a++}}();sa=function(){ja();var a=ia.Symbol.iterator;a||(a=ia.S
4 _.wa=function(a){sa();var b=a[window.Symbol.iterator];return b?b.call(a):na(a)};_.xa="function"!==typeof Object.create?Object.create:fu
5 Ja=function(a,b){if(b){var c=ia;a=a.split(".");for(var d=0;d<a.length-1;d++){var e=a[d];e in c||(c[e]={});c=c[e]}a=a[a.length-1];d=c[a
6 Ja("WeakMap",function(a){function b(a){Ka(a,d)||ha(a,d,{value:{}})}function c(a){var c=Object[a];c&&(Object[a]=function(a){b(a);return
7 1).toString();if(a){ja();sa();a=_.wa(a);for(var b;! (b=a.next()).done;)b=b.value,this.set(b[0],b[1])};f.prototype.set=function(a,c){b(
8 Ja("Map",function(a){if(function(){if(!a||"function"!==typeof a||!a.prototype.entries||"function"!==typeof Object.seal)return!1;try{var
9 {}};this.Pe=f();this.size=0;if(a){a=_.wa(a);for(var b;! (b=a.next()).done;)b=b.value,this.set(b[0],b[1])};c.prototype.set=function(a,b)
10 a.ke.Pi,a.ke.head=null,this.size--,!0):!1};c.prototype.clear=function(){this.lf={};this.Pe=this.Pe.Pi=f();this.size=0};c.prototype.has
11 d;!(d=c.next()).done;)d=d.value,a.call(b,d[1],d[0],this)};c.prototype[window.Symbol.iterator]=c.prototype.entries;var d=function(a,c){
12 c.next,{done:!1,value:b(c)};c=null}return{done:!0,value:void 0}});f=function(){var a={};return a.Pi=a.next=a.head=a,h=0;return c});
13 Ja("Set",function(a){if(function(){if(!a||"function"!==typeof a||!a.prototype.entries||"function"!==typeof Object.seal)return!1;try{var
14 new window.Map;if(a){a=_.wa(a);for(var b;! (b=a.next()).done;)this.add(b.value)}this.size=this.V.size};b.prototype.add=function(a){this
15 b.prototype.values;b.prototype[window.Symbol.iterator]=b.prototype.values;b.prototype.forEach=function(a,b){var c=this;this.V.forEach(
16 _.Ma=function(a){var b=typeof a;if("object"==b)if(a){if(a instanceof Array)return"array";if(a instanceof Object)return b;var c=Object.
17 else if("function"==b&&"undefined"!==typeof a.call)return"object";return b};_.0a=function(a){return"array"==_.Ma(a)};_.Pa="closure_uid_
18 _.z=function(a,b){function c(){c.prototype=b.prototype;a.H=b.prototype;a.prototype=new c;a.prototype.constructor=a;a.bp=function(a,c,
```


ИНСТРУМЕНТЫ РАЗРАБОТЧИКА

Сеть - записывает и отображает сетевые запросы в любое время, когда панель инструментов открыта, даже если сам монитор сети не выбран. Отображает запросы, методы, статус коды, объем данных.



Статус	Метод	Файл	Домен	Причина	Тип	Передано	Размер	0 мс
● 200	GET	/	www.google.com	JS document	html	58,23 КБ	193,39 КБ	
● 200	GET	rs=ACT90oFxLYDwuDJ4rYxoeOUZJB4009szDg	www.google.com	JS script	js	141,03 КБ	407,82 КБ	
● 204	POST	gen_204?s=webaft&atyp=csi&ei=2fOSWtqDK4fWkwWDwJi...	www.google.com	JS beacon	html	—	0 6	
● 200	GET	rs=AA2YrTtMoJJMGQfOfYzY7reaiava99OQ	www.gstatic.com	JS script	js	46,40 КБ	135,49 КБ	
● 200	GET	rs=ACT90oEkGc7BMGCJQLR_QLFnTrvnrATVag?xjs=s1	www.google.com	JS script	js	28,58 КБ	88,78 КБ	
● 200	GET	rs=ACT90oEkGc7BMGCJQLR_QLFnTrvnrATVag	www.google.com	JS script	js	1,60 КБ	3,77 КБ	
● 200	GET	cb=gapi.loaded_0	apis.google.com	JS script	js	46,75 КБ	135,88 КБ	
● 200	GET	read	www.google.com.ua	JS xhr	html	—	0 6	
● 204	GET	write?data=&xsrif=ALAmJdEDTFXJk63_4iKDuoGbrBt5t2GQ9Q...	www.google.com	JS xhr	html	—	0 6	
● 204	POST	gen_204?atyp=csi&ei=2fOSWtqDK4fWkwWDwJioDQ&s=we...	www.google.com	JS beacon	html	—	0 6	

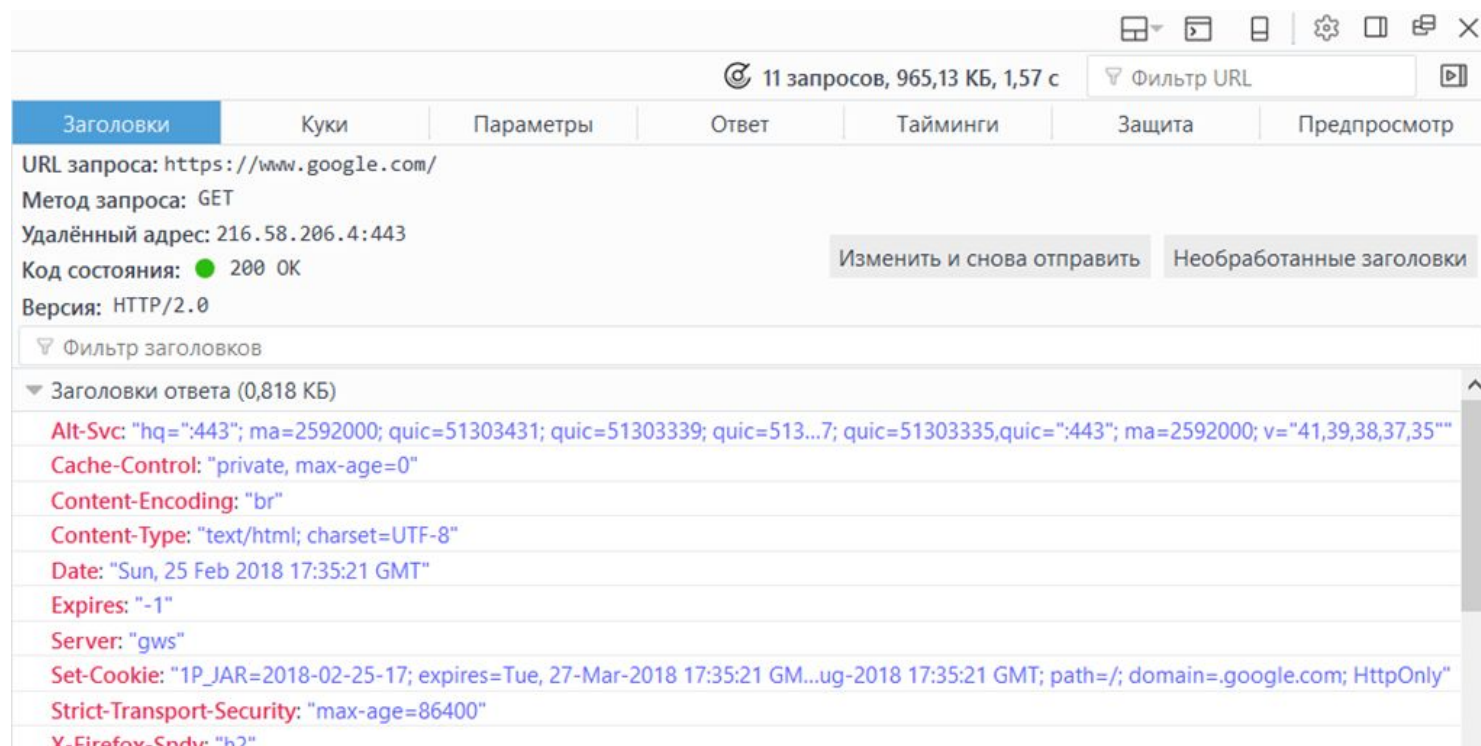
ИНСТРУМЕНТЫ РАЗРАБОТЧИКА

Заголовки - перечислены

основные сведения о запросе, в

том числе:

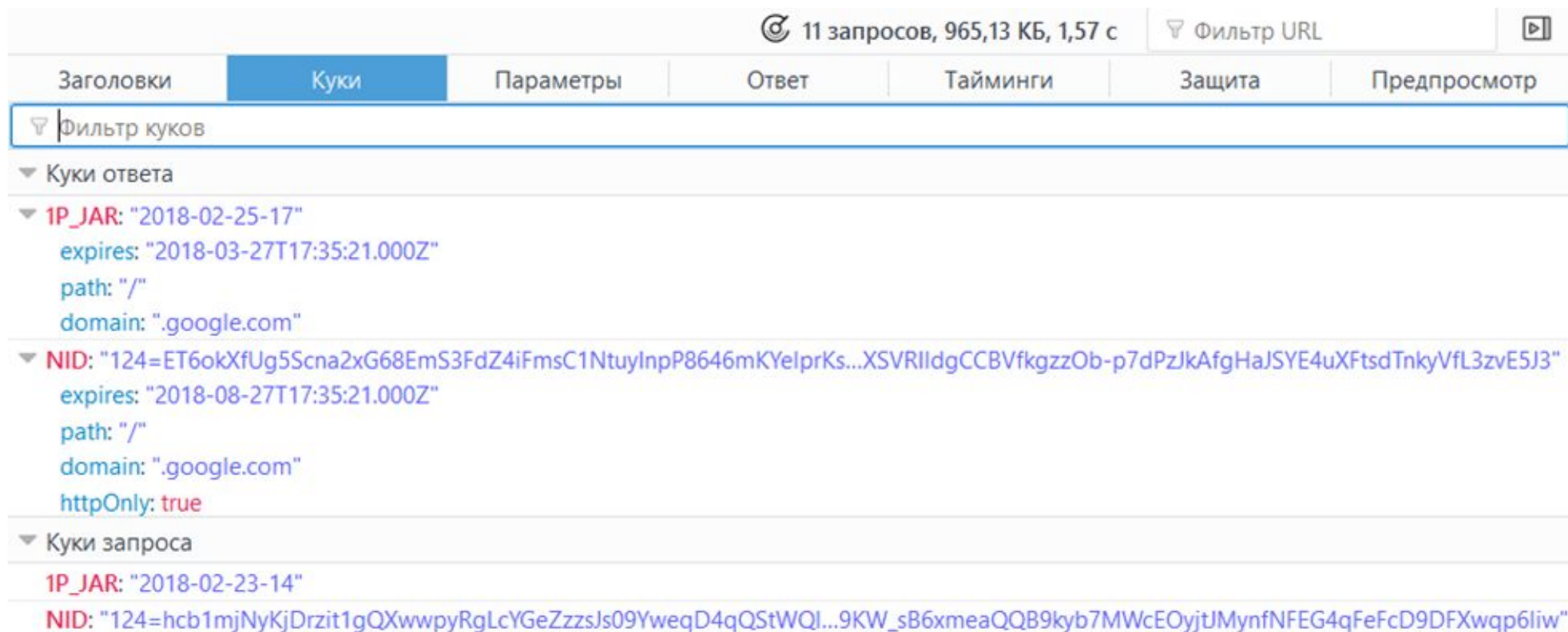
- URL-адрес запроса.
- Метод запроса.
- Удаленный IP-адрес и порт.
- Код состояния.
- HTTP-запросы и заголовки ответов, которые были отправлены.



The screenshot shows the 'Network' tab in a browser's developer tools. The selected request is for 'https://www.google.com/'. The status is '200 OK' (HTTP/2.0). The 'Headers' sub-tab is active, displaying the response headers. The headers include: 'Alt-Svc', 'Cache-Control: private, max-age=0', 'Content-Encoding: br', 'Content-Type: text/html; charset=UTF-8', 'Date: Sun, 25 Feb 2018 17:35:21 GMT', 'Expires: -1', 'Server: gws', 'Set-Cookie: 1P_JAR=2018-02-25-17; expires=Tue, 27-Mar-2018 17:35:21 GMT...', and 'Strict-Transport-Security: max-age=86400'. There are also buttons for 'Изменить и снова отправить' and 'Необработанные заголовки'.

ИНСТРУМЕНТЫ РАЗРАБОТЧИКА

Куки - перечислены все сведения о любых файлах cookie, отправленных с запросом или ответом.



The screenshot shows the Chrome DevTools Network tab with the 'Cookies' sub-tab selected. The top bar indicates 11 requests, 965.13 KB, and 1.57 seconds. A search filter for 'Фильтр URL' is present. The main area is divided into tabs: 'Заголовки', 'Куки', 'Параметры', 'Ответ', 'Тайминги', 'Защита', and 'Предпросмотр'. A search filter for 'Фильтр куков' is active. The 'Куки ответа' section is expanded, showing two cookies:

- 1P_JAR:** "2018-02-25-17"
expires: "2018-03-27T17:35:21.000Z"
path: "/"
domain: ".google.com"
- NID:** "124=ET6okXfUg5Scna2xG68EmS3FdZ4iFmsC1NtuyInpP8646mKYelprKs...XSVRIldgCCBVfkgzzOb-p7dPzJkAfgHaJSYE4uXFtsdTnkyVfL3zvE5J3"
expires: "2018-08-27T17:35:21.000Z"
path: "/"
domain: ".google.com"
httpOnly: true

The 'Куки запроса' section is also expanded, showing two cookies:

- 1P_JAR:** "2018-02-23-14"
- NID:** "124=hcb1mjNyKjDrzit1gQXwwpyRgLcYGeZzssJs09YweqD4qQStWQI...9KW_sB6xmeaQQB9kyb7MWcEOyjtJMynfNFEG4qFeFcD9DFXwqp6liw"

ИНСТРУМЕНТЫ РАЗРАБОТЧИКА

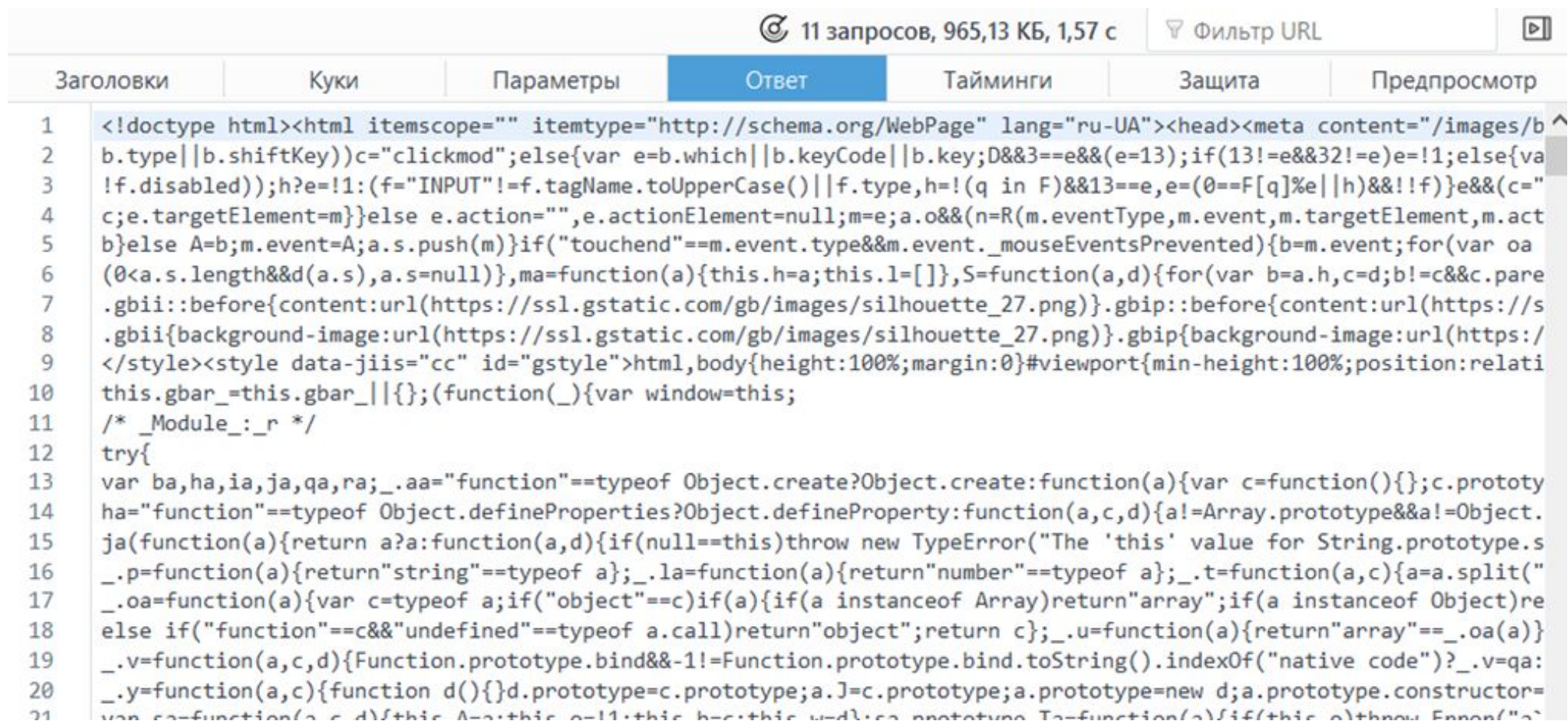
Параметры - перечислены все параметры, отправленные с запросом.

The screenshot shows a web developer tool interface with the following elements:

- Top status bar: 11 запросов, 965,13 КБ, 1,57 с
- Filter URL: Фильтр URL
- Navigation tabs: Заголовки, Куки, **Параметры**, Ответ, Тайминги, Защита, Предпросмотр
- Filter parameters: Фильтр параметров запроса
- Request line: Строка запроса
- Parameters list:
 - s: "webaft"
 - atyp: "csi"
 - ei: "2fOSWtqDK4fWkwWDwJioDQ"
 - rt: "wsrt.268,aft.159,prt.84"

ИНСТРУМЕНТЫ РАЗРАБОТЧИКА

Ответ - отображается ответ, пришедший на запрос в определенном формате данных.

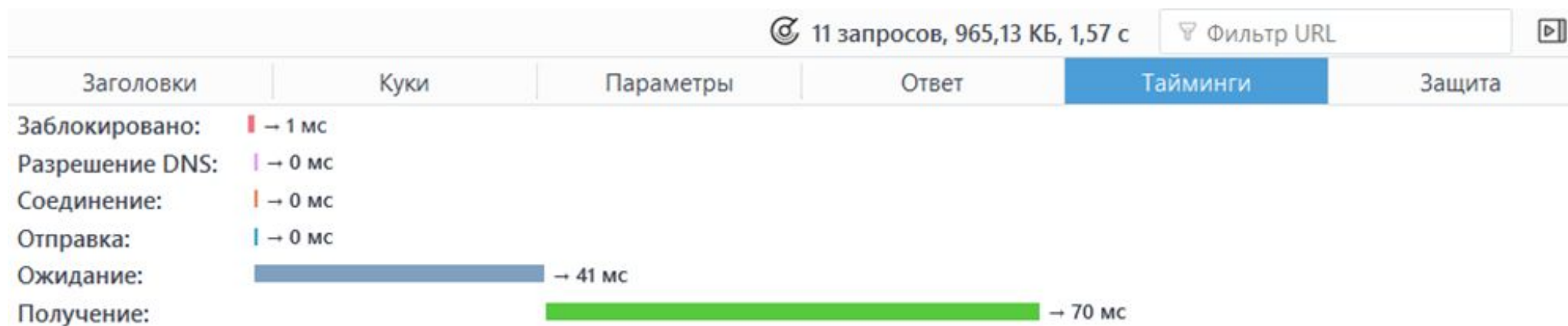


The screenshot shows a browser's developer tools network tab. At the top, it displays '11 запросов, 965,13 КБ, 1,57 с' and a 'Фильтр URL' button. Below this is a table with tabs: 'Заголовки', 'Куки', 'Параметры', 'Ответ' (selected), 'Тайминги', 'Защита', and 'Предпросмотр'. The 'Ответ' tab is active, showing a large block of HTML and JavaScript code. The code starts with a DOCTYPE declaration and a meta tag, followed by a large JavaScript function definition. The function appears to be a jQuery-like utility function for handling events and DOM manipulation. The code is partially obscured by a vertical scrollbar on the right side.

```
1 <!doctype html><html itemscope="" itemtype="http://schema.org/WebPage" lang="ru-UA"><head><meta content="/images/b
2 b.type||b.shiftKey))c="clickmod";else{var e=b.which||b.keyCode||b.key;D&&3==e&&(e=13);if(13!=e&&32!=e)e=!1;else{va
3 !f.disabled));h?e=!1:(f="INPUT"!f.tagName.toUpperCase())||f.type,h=!(q in F)&&13==e,e=(0==F[q]%e||h)&&!f)}e&&(c="
4 c;e.targetElement=m}}else e.action="",e.actionElement=null;m=e;a.o&&(n=R(m.eventType,m.event,m.targetElement,m.act
5 b)else A=b;m.event=A;a.s.push(m)}if("touchend"==m.event.type&&m.event._mouseEventsPrevented){b=m.event;for(var oa
6 (0<a.s.length&&d(a.s),a.s=null)},ma=function(a){this.h=a;this.l=[]},S=function(a,d){for(var b=a.h,c=d;b!=c&&c.pare
7 .gbii::before{content:url(https://ssl.gstatic.com/gb/images/silhouette_27.png)}.gbip::before{content:url(https://s
8 .gbii{background-image:url(https://ssl.gstatic.com/gb/images/silhouette_27.png)}.gbip{background-image:url(https:/
9 </style><style data-jiis="cc" id="gstyle">html,body{height:100%;margin:0}#viewport{min-height:100%;position:relati
10 this.gbar_=this.gbar_||{};(function(_){var window=this;
11 /* _Module_:_r */
12 try{
13 var ba,ha,ia,ja,qa,ra;_ .aa="function"==typeof Object.create?Object.create:function(a){var c=function(){};c.prototy
14 ha="function"==typeof Object.defineProperty?Object.defineProperty:function(a,c,d){a!=Array.prototype&&a!=Object.
15 ja(function(a){return a?a:function(a,d){if(null==this)throw new TypeError("The 'this' value for String.prototype.s
16 _ .p=function(a){return"string"==typeof a};_ .la=function(a){return"number"==typeof a};_ .t=function(a,c){a=a.split("
17 _ .oa=function(a){var c=typeof a;if("object"==c)if(a){if(a instanceof Array)return"array";if(a instanceof Object)re
18 else if("function"==c&&"undefined"==typeof a.call)return"object";return c};_ .u=function(a){return"array"==_ .oa(a)}
19 _ .v=function(a,c,d){Function.prototype.bind&&-1!=Function.prototype.bind.toString().indexOf("native code")?_ .v=qa:
20 _ .y=function(a,c){function d(){}d.prototype=c.prototype;a.J=c.prototype;a.prototype=new d;a.prototype.constructor=
21 var ca=function(a,c,d){this.A=a;this.o=1;this.h=this.u;d.prototype.J=function(a){if(this.o)throw Error("a
```

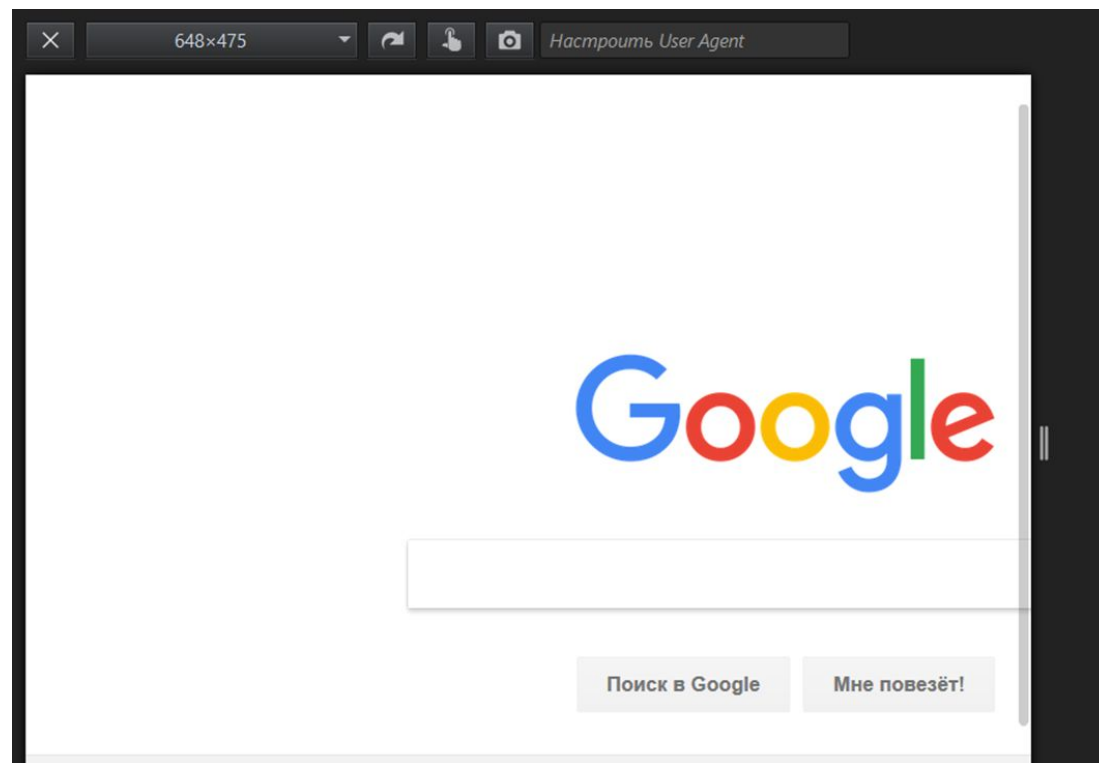
ИНСТРУМЕНТЫ РАЗРАБОТЧИКА

Тайминги - представлен более подробный аннотированный вид временной шкалы для каждого запроса, показывающий время выполнения.



ИНСТРУМЕНТЫ РАЗРАБОТЧИКА


Режим адаптивного дизайна - позволяет проверить сайт при разных разрешениях и сделать скриншот.





СПАСИБО ЗА ВНИМАНИЕ






ТЕСТИРОВЩИК
ПРОГРАММНОГО
ОБЕСПЕЧЕНИЯ

КУРС «РУЧНОЕ ТЕСТИРОВАНИЕ»

13. ПРАКТИЧЕСКОЕ ЗАНЯТИЕ



*Тестирование гиперссылок,
верстки, пользовательского
интерфейса*