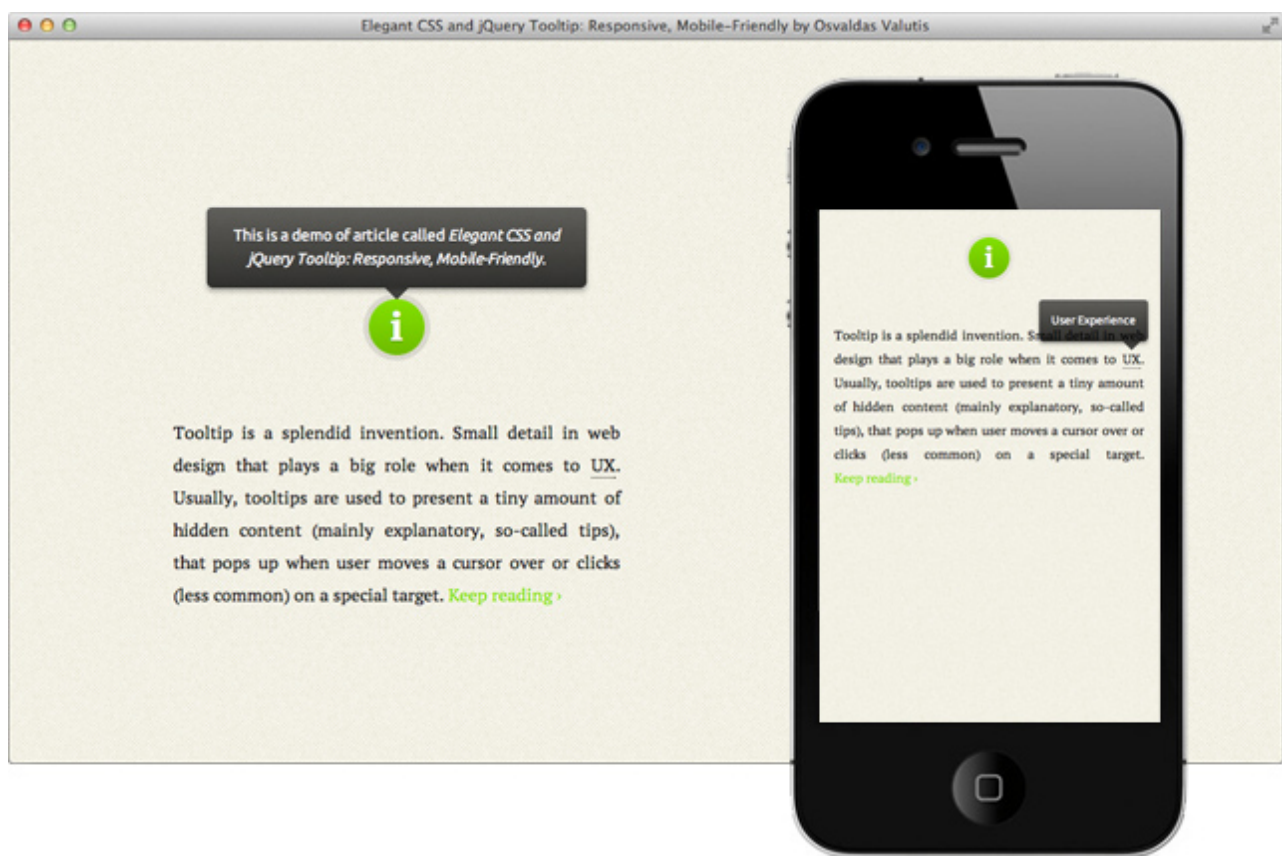# Responsive and Mobile–Friendly Tooltip

16 FEB, 2012

Tooltip is a splendid invention. Small detail in web design that plays a big role when it comes to user experience. Usually, tooltips are used to present a tiny amount of hidden content (mainly explanatory, so-called tips), that pops up when user moves a cursor over or clicks (less common) on a special target.

When working on my personal website, I was in need of a tooltip. I decided not to limit it just on a typical definition of a tooltip, but also make it a better occurrence using CSS and jQuery technologies.

Key Features

It's **responsive**. It relies on a maximum width value when viewed on large screens, adopts to narrow environments and picks the best viewable position relatively to the target (top, bottom; left, center, right);

It's **mobile-friendly**. It pops up when a call-to-action button is tapped
and disappears when tapped on the tooltip itself;

It's **HTML formatting capable**. Need to write some words in italic or
so? No problem, this will work out.

CSS

To begin with, let's build the look of our tooltip. No fancy styling, just major
properties so you can clearly see how it works (all of the sexy things will take
part in the demo below).

```css
1   #tooltip
2   {
3       text-align: center;
4       color: #fff;
5       background: #111;
6       position: absolute;
7       z-index: 100;
8       padding: 15px;
9   }
10
11      #tooltip:after /* triangle decoration */
12      {
13          width: 0;
14          height: 0;
15          border-left: 10px solid transparent;
16          border-right: 10px solid transparent;
17          border-top: 10px solid #111;
18          content: '';
19          position: absolute;
20          left: 50%;
21          bottom: -10px;
22          margin-left: -10px;
23      }
24
25          #tooltip.top:after
26          {
27              border-top-color: transparent;
28              border-bottom: 10px solid #111;
29              top: -20px;
30              bottom: auto;
31          }
32
33          #tooltip.left:after
34          {
35              left: 10px;
36              margin: 0;
37          }
38
39          #tooltip.right:after
40          {
41              right: 10px;
```

```
42                  left: auto;
43                  margin: 0;
44              }
```

JavaScript

```
1   $( function()
2   {
3       var targets = $( '[rel~=tooltip]' ),
4           target  = false,
5           tooltip = false,
6           title   = false;
7
8       targets.bind( 'mouseenter', function()
9       {
10          target  = $( this );
11          tip     = target.attr( 'title' );
12          tooltip = $( '<div id="tooltip"></div>' );
13
14          if( !tip || tip == '' )
15              return false;
16
17          target.removeAttr( 'title' );
18          tooltip.css( 'opacity', 0 )
19                 .html( tip )
20                 .appendTo( 'body' );
21
22          var init_tooltip = function()
23          {
24              if( $( window ).width() < tooltip.outerWidth() * 1.5 )
25                  tooltip.css( 'max-width', $( window ).width() / 2
26              else
27                  tooltip.css( 'max-width', 340 );
28
29              var pos_left = target.offset().left + ( target.outerWi
30                  pos_top  = target.offset().top - tooltip.outerHeig
31
32              if( pos_left < 0 )
33              {
34                  pos_left = target.offset().left + target.outerWidt
35                  tooltip.addClass( 'left' );
36              }
37              else
38                  tooltip.removeClass( 'left' );
39
40              if( pos_left + tooltip.outerWidth() > $( window ).widt
41              {
42                  pos_left = target.offset().left - tooltip.outerWid
43                  tooltip.addClass( 'right' );
44              }
45              else
46                  tooltip.removeClass( 'right' );
47
48              if( pos_top < 0 )
49              {
50                  var pos_top  = target.offset().top + target.outerH
51                  tooltip.addClass( 'top' );
```

```
52                }
53                else
54                    tooltip.removeClass( 'top' );
55
56                tooltip.css( { left: pos_left, top: pos_top } )
57                    .animate( { top: '+=10', opacity: 1 }, 50 );
58            };
59
60            init_tooltip();
61            $( window ).resize( init_tooltip );
62
63            var remove_tooltip = function()
64            {
65                tooltip.animate( { top: '-=10', opacity: 0 }, 50, func
66                {
67                    $( this ).remove();
68                });
69
70                target.attr( 'title', tip );
71            };
72
73            target.bind( 'mouseleave', remove_tooltip );
74            tooltip.bind( 'click', remove_tooltip );
75        });
76    });
```

HTML Example

```
1   <abbr title="User Experience" rel="tooltip">UX</abbr>
```

DEMO

How To Implement

1. Put the CSS code anywhere in existing CSS file or create a new one.
    You can also insert it in HTML file in *head* by wrapping the code with
    `<style>` and `</style>` tags.

2. Put the JavaScript code in existing JS file or create a new one. You
    can also insert it in HTML file in *head* or, better, *body* by wrapping the
    code with `<script>` and `</script>` tags.

3. Assign the attribute `rel="tooltip"` and `title="Enter your
    tip here"` to any of *body* tags in HTML file where you want the
    tooltip to pop up when called. Set *title* value with your tip (use

`<strong>`, `<em>` etc. to distinguish text fragments, but avoid *block* elements).

Also, be sure that you already have jQuery library included.

That's it! Simple enough. If you've got any suggestions or insights, feel free to share them: leave a comment or send me a tweet.

Tweet   Like 6   G+   Follow @osvaldas   2,813 followers

NEWSLETTER

✉ Email

In case you want to be informed about my blog posts.

RSS FEED

You can also subscribe to my **RSS feed** in order to follow my blog. Not using any RSS reader? You can try Readerrr.

READ NEXT
Dribbble and Forrst Invitation Giveaway