2014-05-03 02:40

Icon Standardization For Web Applications

L2/14-107

Abstract

This research is derived from my talk at ArabNet conference held in Beirut on 2013-03-20.

The aim of this research is to develop standards for representating icons in web applications. This means giving proposed icons in this research addresses in Miscellaneous Symbols and Pictographs unicode section. i.e.:

- 1F54B = RIGHT-POINTING MAGNIFYING GLASS WITH PLUS SIGN = Zoom in
- 1F54C = RIGHT-POINTING MAGNIFYING GLASS WITH MINUS SIGN = Zoom out
- 1F54D = FOUR ANGULAR ARROWS FACING OUT = Full-screen
- 1F54E = FOUR ANGULAR ARROWS FACING IN = Exit full-screen
-
- ... and so on.

Methedology

- First of all, I will show the solutions currently being used to display icons in web applications:
  - Whether using the usual image files,
  - Or using the embeded-fonts technology.
- Then I will form a model for a proposed web site, in order to summarize the icons needed for most web applications.
  In light of this model, I will go to propose a set of recomendations to:
  - Unicode organization to determine Unicode icons missing in the current version of Unicode,
  - and to W3C - The World Wide Web Consortium, to visualize a view for represnting icons using CSS.

Using Regular Image Files

The most popular solution used to display icons in web applications is called “CSS Sprites”, which is to collect the various needed icons -of all sizes and colors- in a single regular image file. Here is a sample of Google icons stored in a single regular image file. Notice that icons are spreaded over the image canvas randomly.

Google icons are stored in a one single image file

Using The Embed Fonts Technology

The embed fonts technology has paved a new way for web developers to represent icons using characters in the embedded fonts instead of the regular image formats used usually in the web, such as PNG, GIF… etc, which represents an important step for icon addressing in software industry generally, and in web development world particulaly.

This step coincided with another important step as important as its previous one, which is the adoption of Unicode 6.0 for a broad spectrum of visual symbols (Emoji) and embedding it in its tables.

> Emoji are the ideograms or smileys used in Japanese electronic messages and webpages, whose use is spreading outside Japan. Originally meaning pictograph, the word emoji literally means “picture” + “character”. Wikipedia, 2014

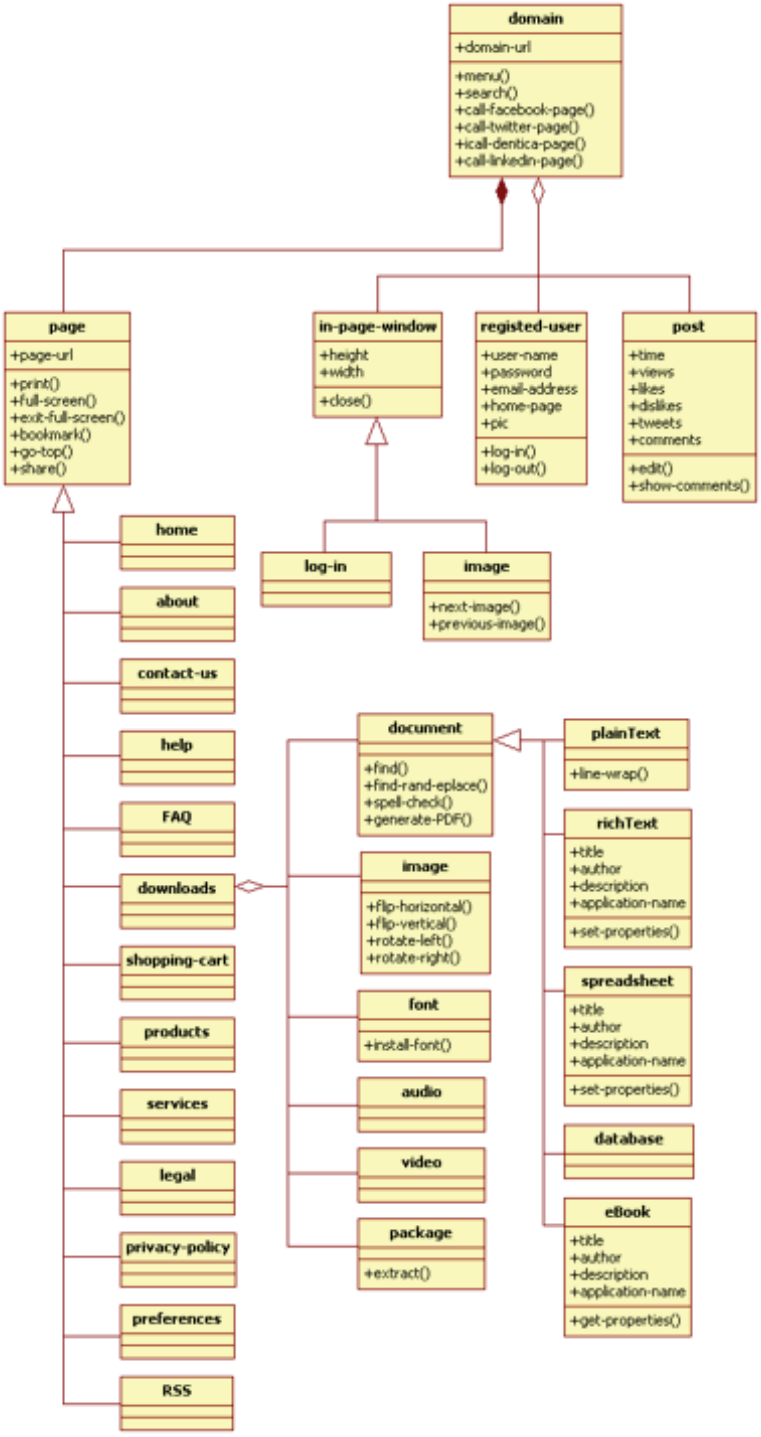
This step might lead us to a completely **different usage of Emoji**; providing a stable standard for addressing UI icons in software engineering. According to this version of Unicode, each popular UI icon had a stable address in Unicode pages. For example; the muted speaker has the address `1F507`, and the magnifying glass has the address `1F50D`... and so on.

Some of the new icons supported by Unicode 6 and implemented using RichStyle font

RichStyle font is an attempt to implement the idea of icons-addressing in the web pages through the Unicode 6.0, but this project revealed a lack of some icons and visual symbols that are not addressed yet by Unicode. These icons are the ones that have a label in the preview page of RichStyle font. The most important are: printer, image, tag (label), FAQ... and others.

Recomendations for Unicode: Missing Icons

In order to clarify this limitation I will try to develop a model for a generic web site covers the most popular requirements for a modern web site.



A simple structure for a web site

Notice that:

- Every single class in this model needs an icon.
- Every single opertation in each class needs an icon too.
- Attributes do not need icons.

In the light of this model, and comparing to the icons supported in Unicode 6, you will notice that the most imprtant icons missing are: ~~printer~~, ~~image~~, full-screen, exit full-screen... etc, as it's

displayed in the following figure:

Propsed web Emoji icons for Unicode

Printer, image, and label icons are now supported in Unicode 7.0 Beta.

Recomendations for W3C: Representing Icons Using CSS

W3C standards stated that icons should be represnted using the following syntax:

```
<menu>
        <menuitem icon='page-about.png'/>
        <menuitem icon='domain-settings.png'/>
</menu>
```

Unfortunately, this standard will not allow you to implement the popular ways used nowadays to represent icons, whether through CSS Sprites or embedded fonts technology, it's a backward movement, and it means representing each icon using an independent image file, and therefore fatiguing the network with too many request for downloading a large amount of images.

As an alternative, I suggest representing icons using CSS pseudo element, not as an HTML element's attribute. The reason for this is that icon -even though it represents an attribute- but:

- First of all; it's a presentation attribute, and therefore must be represented using CSS rather than HTML,
- Secondly, it's a rich attribute; an attribute needs attributes. These kind of attributes are usually represented using something called Pseudo Element.

practically; in the HTML page, we'll use the element `MenuItem` with a class name, and in CSS file we'll use a pseudo element called `:icon`, to be able to adjust all the presentation attributes of this icon, just like content, font name, font size, and position.

```
<menu>
        <menuitem class='page-about'/>
        <menuitem class='domain-settings'/>
</menu>
```

```
.page-about:icon {
        content: char(𝐢);
        font: 10pt RichStyle;
        icon-position: top;
```

```
}

.domain-settings:icon {
        content: char(⚙);
        font: 10pt RichStyle;
        icon-position: top;
}
```

Icon Standardization for Desktop Applications

A few years ago, Tango Icons project have tried to set a standard for icons naming for Linux, in order to be addressed in a standard way. However, I think that Unicode 6.0 now represents a great alternative to be adopted as a standard for addressing the icons within the software process.

Unicode 6.0 icons naming vs. Tango icons naming

Here is an express view of proposed icons for web, desktop, and mobile icons, inspired from Unicode 7 Beta itself.

Proposed Emoji icons for web, desktop, and mobile environments