

Mise en œuvre d'API JavaScript de production de codes QR

Comme pour les API de génération de documents au format PDF (cf. chapitre précédent), il existe de nombreuses API permettant la production de codes QR.

Nous allons au travers de deux exemples voir quelques-unes des possibilités des codes QR :

- Exemple 1 : Code QR mémorisant le numéro de téléphone d'un correspondant à appeler téléphoniquement,
- Exemple 2 : Code QR mémorisant l'URL d'un site Web.

1. Exemple 1 : Code QR embarquant un numéro de téléphone

Commentaire du script

Pour ce premier exemple, nous allons utiliser le générateur de codes QR de la société IWWWIT : <http://qr-code-generator.iwwit.com/generateur-qr-code.php>

L'inconvénient de la solution est qu'il faut transmettre (poster au sens HTML du terme) les paramètres du code QR à créer à un script PHP hébergé par la société IWWWIT. Ceci pénalise quelque peu l'intégration de cette solution dans votre propre développement Web. Il s'agit du choix technique de la majorité des prestataires proposant ce type de dispositif.

Le code du script est très simple. Il suffit de transmettre, via un formulaire installé dans la section HTML <body>, d'un jeu de paramètres à destination d'un script PHP mis à notre disposition par le prestataire. Une fois les informations prises en compte par celui-ci, il génère le code QR sur une page Web (c'est justement ceci qui est un peu gênant du point de vue ergonomique).

Voici le code du formulaire :

```
<!-- Formulaire de demande de création du code QR correspondant
à un numéro de téléphone -->
<!-- NB : Ne pas modifier les attributs name des différents input -->
<h2>QR code permettant de lancer un appel téléphonique</h2>
<form method="post"
action="http://qr-code-generator.iwwit.com/createur-qr-code.php">
    Numéro de téléphone : <input type="text" name="tel-num" id="tel-num" />
    <input type="hidden" name="couleurf" class="couleurf" value="#ffffff" />
    <input type="hidden" name="couleurp" class="couleurp" value="#000000" />
    <input type="hidden" name="err" class="err-l" value="L" />
    <input type="hidden" name="qrcode" class="qrcode" value="300" />
    <input type="submit" name="sub" value="Génération du code QR" />
</form>
```

Vous noterez que les informations (hormis le numéro de téléphone qu'il vous est demandé de saisir) sont transmises au script serveur en mode caché (type="hidden").

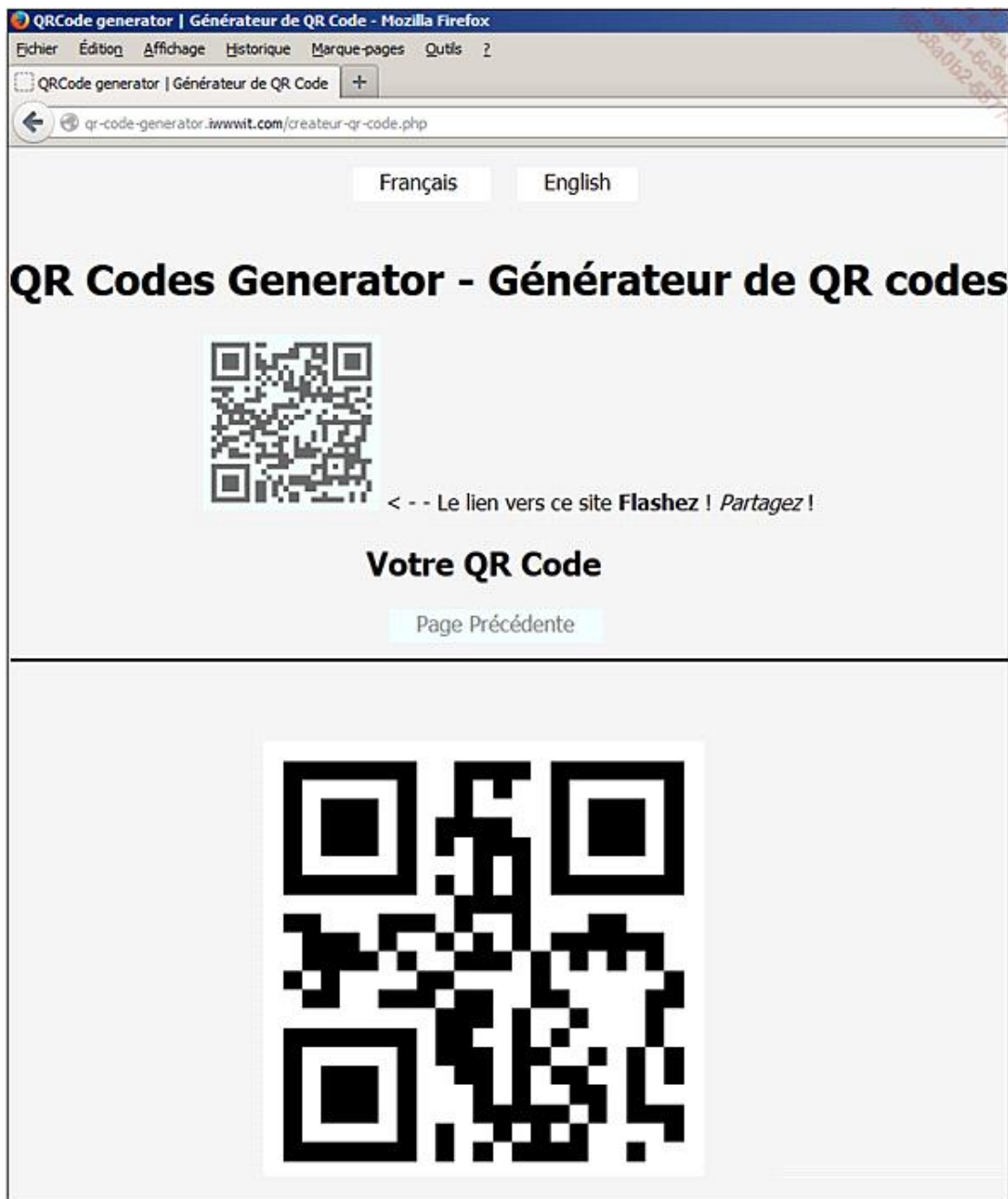
Les valeurs des attributs name et id sont bien sûr imposées par le script serveur. Ne modifiez pas non plus la clause value du bouton de soumission !

Exécution du script

Voici à titre indicatif l'interface de saisie du numéro de téléphone :



Le résultat est fourni sans délai par l'application serveur sous cette forme :



2. Exemple 2 : Code QR embarquant une URL

La solution retenue dans ce deuxième exemple est un peu plus classique. Des paramètres techniques concernant le code QR sont également à transmettre à l'API Google, notamment :

- la taille du code QR (dimension du carré exprimée en pixels),
- le texte de l'URL.

Les paramètres sont transmis à l'API via une URL (et non pas via un formulaire comme dans l'exemple 1).

L'avantage de la solution Google est la maîtrise de la présentation du résultat qui peut être restitué directement sur la page Web ayant servi à saisir les paramètres.

Étudions rapidement le code du script.

Section HTML <body>

Cette section comprend essentiellement le formulaire de saisie des deux paramètres évoqués ci-avant :

```
<!-- Formulaire de saisie des paramètres du code QR -->
<form>
  <table>
    <tr>
      <td>
        Texte QR Code
      </td>
      <td>
        <input type="text" id="texteQRCode" size="50"
        value="http://christian.vigouroux.online.fr">
      </td>
    </tr>
    <tr>
      <td>
        Taille QR Code (Hauteur=Largeur)
      </td>
      <td>
        <input type="text" id="tailleQRCode" size="5"
        value="150">
      </td>
    </tr>
  </table>
  <br />
  <input type="button" id="boutonGenerationQRCode"
  value="Créer le QR Code" onclick="genererQRCode()">
</form>
```

Le formulaire est suivi par :

```
<!-- Image QR Code -->
<img id="imageQRCode">
```

Il s'agit de la position de l'image du code QR qui sera générée par la fonction `genererQRCode`, appelée par le bouton `boutonGenerationQRCode` se trouvant en fin de formulaire.

Section HTML <head>

Après la balise meta habituelle ([`<meta HTTP-equiv="Content-Type" [content="text/html; charset=utf-8" />]`]) et le titre de la page Web, cette section contient la fonction `genererQRCode`.

```
/* Fonction genererQRCode */
function genererQRCode()
{
```

```

/* Préparation de l'URL de création du code QR */
/* Exemple URL :
https://chart.googleapis.com/chart?chs=150x150&cht=qr&chl=
Hello%20world&choe=UTF-8 */
var urlQRCode="https://chart.googleapis.com/chart?";
urlQRCode = urlQRCode + "&chs="
+ document.getElementById('tailleQRCode').value + "x"
+ document.getElementById('tailleQRCode').value;
urlQRCode = urlQRCode + "&cht=qr";
urlQRCode = urlQRCode + "&chl="
+ document.getElementById('texteQRCode').value;
urlQRCode = urlQRCode + "&choe=UTF-8";

/* Affichage de contrôle de l'URL */
// alert("URL du génération du code QR : " + urlQRCode);

/* Affichage du code QR */
document.getElementById('imageQRCode').src = urlQRCode;
}

```

Trois des quatre paramètres transmis via l'URL à l'API Google sont obligatoires :

- le paramètre `chs` signalant la taille du code QR (largeur x hauteur),
- le paramètre `cht` signalant que l'API Google Chart doit produire un code QR (valeur du paramètre `qr`),
- le paramètre `chl` contenant l'URL à laquelle le code QR donnera accès.

Le paramètre `choe` est facultatif, il sert à préciser l'encodage des données dans le code QR (valeurs à choisir entre UTF-8 (valeur par défaut), Shift_JIS et ISO-8859-1).

Le script de la fonction se termine par l'affichage du code QR (cf. image annoncée dans la section HTML), comme suit :

```

/* Affichage du code QR */
document.getElementById('imageQRCode').src = urlQRCode;

```

Exécution du script

