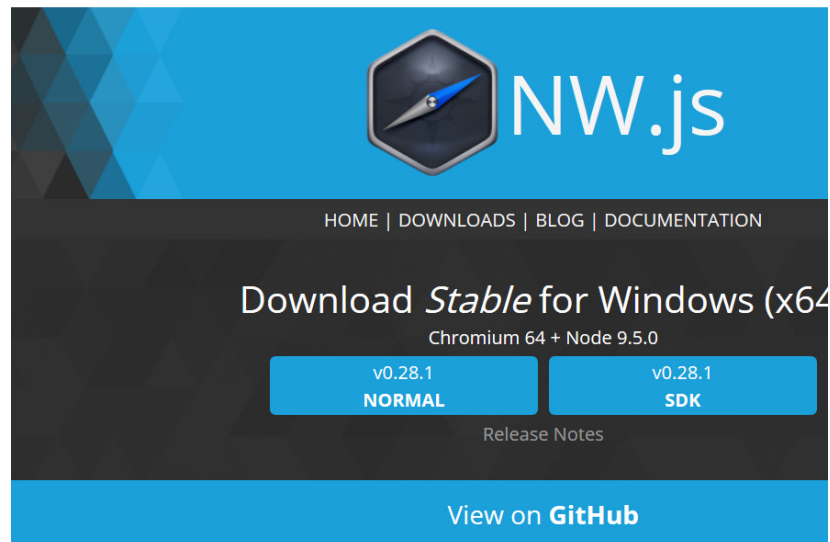


Partie 1 : NW.js



NW.js (previously known as node-webkit) lets you call all I

NW (ou node-webkit) est un environnement de développement basé sur HTML, CSS et javascript qui donne la possibilité de réaliser des applications complètes avec accès aux fonctions proposées par node.js. De base, vous pouvez donc rendre autonome (via un exécutable) une application web, mais également l'étendre (par exemple pour les entrées sorties) en utilisant node.js. L'autre alternative, Electron ou atom-shell propose des fonctionnalités analogues, éventuellement sans partir d'une page Web.

Dans ce TD, vous allez transformer en application autonome, votre application de Candy Crush. Vous allez générer les applicatifs pour Linux, Windows et MacOS.

Prérequis : NW.js utilise node.js. Node.js est déjà installé sur les postes de l'université. Si vous souhaitez utiliser vos laptops, il faut vous-même installer node.js : voir <https://nodejs.org/en/>

La démarche à suivre est ici indiquée dans l'environnement Linux de l'université (NW.js fonctionne très bien sous windows et sous MacOS également).

Vérifiez que vous avez au moins accès à l'interpréteur node.js version 6 : **node --version**

L'installation de node.js ajoute également la commande **npm**

Récupérez le SDK NW.Js sur le site web et décompressez le dans un de vos répertoires (voir <https://nwjs.io/>)

Vérifiez que vous pouvez lancer l'exécutable nw (qui affiche une fenêtre vide). En bas de la fenêtre sont indiquées les versions des interpréteurs et navigateurs intégrés dans cette version de nw.

Fabriquez une page web simple avec un texte (genre 'Hello') dans un répertoire test.

Dans le même répertoire que test.html, vous allez créer un fichier package.json : ce fichier contient des informations pour l'interpréteur nw. Créez ce fichier et un objet javascript à l'intérieur.

```
{  
    "main" : "test.html",  
    "name" : "test",  
    "version" : "0.1.0"  
}
```

Ce fichier sert à indiquer à NW.js comment interpréter le code (quel code interpréter). Lancez NW sur le répertoire courant du projet (si vous êtes dedans, /chemin d'installation de nw/nw .)

Normalement, une fenêtre s'ouvre avec votre page web.

Afin de créer un exécutable, vous allez lancer le compilateur **nw-builder**.

Vous devez d'abord l'installer. Si node.js est installé correctement, vous avez également accès à la commande **npm** qui permet d'installer des modules et des librairies pour node.js.

Placez-vous dans le répertoire de test et tapez :

npm i nw-builder

Cette commande crée un répertoire **node_module** et y télécharge les librairies **nw-builder**. Attention, vérifiez que vous avez de la place sur votre compte.

Vous allez maintenant créer un programme Javascript qui lance la compilation de votre projet.

Ce code javascript ne va pas être exécuté par un navigateur, mais par **node.js**. Créez un fichier **compil.js** :

```
var NwBuilder = require('nw-builder');  
  
var nw = new NwBuilder({  
    files: ['./**'],  
    platforms: ['linux64'] // ['win64','linux32' etc..] les cibles de cross compilation  
});  
  
nw.build().then(function () { // promesse Javascript (execute à la fin de la compilation)  
    console.log('Compilation terminée !');  
}).catch(function (error) {  
    console.error(error);  
});
```

Pour lancer ce code (vérifiez que vous avez de la place sur votre compte) :

node compil.js

Si vous avez un problème de connexion (**EHOSTUNREACH**), il est possible qu'il soit lié au mécanisme de gestion de proxy de l'UBS. Vous pouvez configurer l'utilisation d'un proxy de la manière suivante :

```
export HTTPS_PROXY=http://squidva.univ-ubs.fr:3128
```

```
export HTTP_PROXY=http://squidva.univ-ubs.fr:3128
```

Le système télécharge les fichiers qui dépendent de votre déploiement, puis réalise la compilation de votre projet en exécutable (attention, ça prend de la place et il n'est pas conseillé si vous êtes juste coté espace disque de lancer la compilation sur plusieurs systèmes cibles en même temps).

Un répertoire **build** est créé avec les répertoires des binaires.

Lancez le binaire 'test' et vérifiez que vous avez bien le résultat souhaité. Quelle est la taille occupée par le répertoire ?

Vérifiez que vous pouvez faire de la compilation croisée : si vous êtes sous linux, vérifiez que vous pouvez compiler un exécutable pour Windows, puis exécuter cet exécutable sous Windows – et l'inverse coté linux (évidemment si vous utilisez MacOS, testez plutôt sur MacOS).

Vous pourrez trouver les options à placer dans package.js sur cette page :

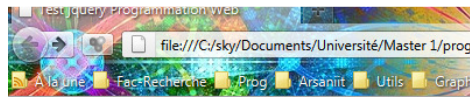
<https://github.com/nwjs/nw.js/wiki/manifest-format>

Il est possible de fixer la taille de la fenêtre, enlever les bordures écran, mettre en plein écran, changer l'icône de l'application etc.

APRES avoir vérifié que la compilation donne des résultats prévus pour le fichier de test, utilisez NW.js pour fabriquer un exécutable linux + un exécutable Windows de votre candy crush. Vous rendrez une archive compressée avec les deux répertoires d'exécution.

Partie 2 : Introduction jQuery

L'objet de ce TD est de vous faire réaliser une petite console directement dans une page Web pour expérimenter de manière interactive l'utilisation de JQuery.



ceci est du texte

voici le second paragraphe

et voici le troisieme

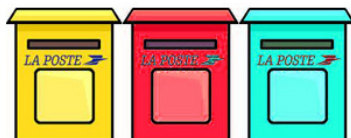
rock'n roll

La liste ul d'id #laliste :

- un
- deux
- trois
- quatre

hello ici la div d'id #ladiv

hello ici la div d'id
#ladiv2



```
$("#ladiv").show(4000);
```

run

Les images sont fournies dans l'archive jointe jquery1.zip.

L'idée ici est de réaliser une page avec des identifiants associés aux objets. Le champ texte en bas, associé au bouton 'run' permet de taper directement du JavaScript et de voir le résultat de l'exécution dans la page.

L'exécution se fait localement par le navigateur (laissez tomber les post/get etc.. il s'agit bien de JavaScript exécuté par le navigateur courant en local). Pour chaque image de lettre, choisissez les ID suivants :

lettre jaune
lettre rouge
lettre bleu

Chaque ligne à partir de 'ceci est du texte' est un paragraphe différent.

Les id sont :

un
deux
trois
quatre

La liste UL a pour id 'laliste'.

Les deux div ont pour id respectivement 'ladiv' et 'ladiv2'

Les boites aux lettres ont pour id :

boite jaune

boite rouge

boite bleu

La page se termine par un formulaire pour rentrer le JavaScript. Il est composé d'un *textarea* d'id 'montexte' et d'un *input* de type *button*.

Rajoutez quelques classes en début de fichier :

```
<style>

.red {
    color: #FF0000;
}
.blue {
    color: #0000FF;
}
.black {
    color: #000000;
}

</style>
```

Le principe ici est de récupérer le contenu du champ 'montexte' et de l'interpréter. Vous utiliserez la fonction JavaScript 'eval' appliquée sur le contenu texte du champ. Toutes les modifications de la page se feront par la console exclusivement. Vous indiquerez pour chaque question la ou les commandes à taper (rendu TD = source + réponses aux questions)

1. Indiquez deux manières (au moins) d'accéder au contenu du champ 'montexte' en utilisant JavaScript et en utilisant JQuery.
2. Testez le fonctionnement de votre page en affichant une boîte d'alerte en tapant la commande dans le champ et en appuyant sur Run.
3. Testez 'document.write("coucou")'. Proposez une autre méthode pour remplacer directement tout le contenu de la page (utilisant les sélecteurs JQuery)
4. Idem : modifiez le contenu d'une des div par la console. Comment ?
5. Est-il possible d'injecter du texte en gras dans une div ? Comment ?
6. Est-il possible d'injecter 'test' dans une div (et voir les). Comment ?
7. Faites disparaître puis réapparaître les div (par show / hide). Testez différents paramètres numériques en argument de show et hide pour voir.
8. Utilisez le changement de classes par JQuery pour changer les couleurs des textes des différents paragraphes.
9. Proposez une/des commande(s) JQuery pour inverser l'ordre des divs texte dans la page.
10. Proposez les commandes JQuery qui permettent de faire un glisser/deposer des enveloppes sur les boîtes aux lettres.
11. Proposez les commandes JQuery qui permettent aux boîtes d'afficher un message dans la console quand elles reçoivent une lettre de leur couleur.