

jQuery, jQuery UI et jQuery MOBILE

J. ROMAGNY

I-JQUERY	3
1. SELECTORS.....	4
<i>Cacher les sélecteurs</i>	4
<i>\$(this)</i>	4
<i>Savoir si un élément existe</i>	4
<i>Sélectionner les éléments</i>	4
2. DOM.....	6
<i>Récupérer le contenu d'un élément</i>	6
Dimensions	6
Traversing	6
<i>Modifier un élément</i>	6
Insérer avant, après	7
<i>Supprimer</i>	7
CSS.....	7
Attribut « data ».....	8
3. EVENTS.....	8
4. ANIMATION	10
a. <i>Afficher/ Masquer un élément avec show/hide</i>	10
b. <i>Fondu avec fadeIn, fadeout, fadeToggle</i>	10
c. <i>Slide avec slideDown, slideUp, slideToggle</i>	11
d. <i>Easing</i>	11
e. <i>Custom animations avec Animate</i>	12
Animation couleur	12
Animation de mouvement	13
Transformations.....	13
translate(x,y)	13
scale(x,y)	14
rotate(deg).....	14
Combinées	14
5. AJAX.....	15
a. <i>Load</i>	15
b. <i>\$.get</i>	15
c. <i>\$.post</i>	16
d. <i>\$.getJSON</i>	16
a. <i>\$.ajax</i>	16
serialize	17
Headers setRequestHeader et getRequestHeader	17
<i>Créer un data service</i>	18
a. <i>Promises et \$.when</i>	20
6. JQUERY VALIDATION.....	21
7. JQUERY TEMPLATES.....	22
<i>Installation du plug in:</i>	22
<i>tmpl</i>	22
<i>Template tags</i>	24
<i>\$.item</i>	25
<i>Caching</i>	25

<i>Vue Master/details</i>	25
<i>D'autres solutions</i>	27
8. PLUGIN.....	27
a. <i>Créer un plugin</i>	27
Publier son plugin.....	29
b. <i>Trouver des plugins existants</i>	29
II-JQUERY UI	30
1. <i>Theme roller</i>	30
2. <i>Widgets jQuery UI</i>	31
a. <i>Accordion</i>	32
b. <i>Autocomplete</i>	33
c. <i>Button</i>	34
d. <i>Datepicker</i>	35
e. <i>Dialog</i>	36
f. <i>Menu</i>	37
g. <i>Progressbar</i>	38
h. <i>Selectmenu</i>	39
i. <i>Slider</i>	39
j. <i>Spinner</i>	40
k. <i>Tabs</i>	40
l. <i>Tooltip (visuel amélioré par rapport au tooltip par « défaut »)</i>	41
4. <i>Interactions</i>	42
<i>Draggable</i>	42
<i>Droppable</i>	42
<i>Sortable</i>	42
<i>Resizable</i>	42
5. <i>Effects</i>	43
III-JQUERY MOBILE	44
1. <i>ThemeRoller pour jQuery Mobile</i>	44
2. <i>Page</i>	44
Création de Page jQuery MOBILE.....	44
Mettre en cache.....	44
Header et navbar.....	45
Data-role.....	45
Navigation.....	45
Prefetch.....	46
Transitions.....	46
3. <i>Widgets jQuery Mobile</i>	47
a. <i>Tabs</i>	49
b. <i>Popup</i>	50
c. <i>Boite de dialogue</i>	51
d. <i>Fenêtre superposée</i>	51
e. <i>Panel</i>	51
f. <i>ListView</i>	52
b. <i>Table</i>	54
h. <i>Collapsible</i>	55
i. <i>Grid</i>	57
j. <i>Button</i>	57
k. <i>Bar</i>	58
l. <i>Form</i>	59
m. <i>Icons</i>	60

Pour utiliser jQuery/jQuery UI/JQuery Mobile :

- **Télécharger** les sources sur le [site](#) ou packages **NuGet** avec **Visual Studio**
- **CDN** (Content Delivery Network)
 - Google <https://developers.google.com/speed/libraries/devguide?hl=fr>
 - Microsoft <http://www.asp.net/ajaxlibrary/cdn.ashx>
 - Cdnjs <https://cdnjs.com/>

I-jQuery

IE 6+ : utiliser jQuery 1.x

IE 9+ : utiliser jQuery 2.x

CDN

```
<script
src="https://ajax.googleapis.com/ajax/libs/jquery/2.1.3/jquery.min.js"></script>
```

Ou **LOCAL**

```
<script src="Scripts/jquery-2.1.3.min.js"></script>
```

Détecter quand la page est chargée

```
$(document).ready(function () {
    //
});
```

Ou

```
$(function () {
    //
});
```

Exemple

```
<!DOCTYPE html>
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
<meta http-equiv="Content-Type" content="text/html; charset=utf-8"/>
<title></title>
</head>
<body>

    <script src="Scripts/jquery-2.1.3.min.js"></script>
    <script>
        $(document).ready(function () {

        });
    </script>
</body>
</html>
```

Prévoir un chargement local en cas de problème CDN

```
<script src="//ajax.googleapis.com/ajax/libs/jquery/2.1.1/jquery.min.js"></script>
<script>
    window.jQuery || document.write("<script src='scripts/jquery.js'></script>");
</script>
<script>
    $(document).ready(function () {
        $("#div1").html("Script loaded!");
    });
</script>
```

1. Selectors

Selectors et ref selectors

Cacher les sélecteurs

S'ils doivent être utilisés plus d'une fois.

```
$("#div1").addClass("class1");
$("#div1").fadeIn(500);
```

Avec caching

```
var div1 = $("#div1");
div1.addClass("class1");
div1.fadeIn(500);
```

\$(this)

Élément en cours

Représente div1

```
$("#div1").on("click", function () {
    $(this).append("clac !");
});
```

Savoir si un élément existe

```
var div1 = $("#div1");
if (div1.length) {
}
else {
    // existe pas
}
```

Sélectionner les éléments

1. NOM DE TAG

`$('tag')`

`$('div')` ...sélectionne tous les `<div>`

`$('p,a,span')` ...sélectionne tous les `<p>`, `<a>` et ``

`$('div:contains("value"))`

2. ID

`$('#id')`

3. NOM DE CLASSE

`$('.class')`

4. ATTRIBUT

`$('[attribute]')` ou `$('[attribute="value"]')`

`$('[title]')` ... toutes les balises ayant un attribut « title »

`$('[title="my title"]')` ... les balises ayant un attribut « title » dont la valeur est « my title »

`$('[title*="my"]')` ...valeur contient « my »

`$('[title^="my"]')` ...valeur d'attribut commençant par « my »

`$('[title$="my"]')` ...valeur d'attribut finissant par « my »

Sélection d'éléments de formulaires

`$(':button')`

`$(':input[type="radio"]')` sélectionne uniquement les boutons radio

Plus ...

`$('div[title]')`

`$('div.myclass')`

`$('div img')` ...descendant : image contenue dans une div

`$('div>img')` ...uniquement les images qui sont enfants directs de div

`$('div~img')`

`$('li:nth-child(2)')`

`$('.myclass:nth-child(3n+2)')`

`$('.myclass:nth-child(odd)')`

`$(p :first), $(p :last)`

`$('li:first-child')`

`$(':not(.myclass)')`

`$('p:empty')`

`$('*')` ...Universel

2. DOM

Manipulation, traversing
ref manipulation, ref traversing

Récupérer le contenu d'un élément

- **text(), html(), and val()**
`$("#id").text()`
`$("#id").html()`
`$("#id").val()`
- Attribut : **attr()**
`$("#id").attr("href")`

Dimensions

- `width()` element
- `height()`
- `innerWidth()` element + padding
- `innerHeight()`
- `outerWidth()` element + padding + border
- `outerHeight()`

Traversing

- `Parent(), parents(), Children(), find()`
- *Sibling* : `siblings(), next(), nextAll(), nextUntil(), prev(), prevAll(), prevUntil()`
- *filter* : `first(), last(), eq(), filter(), not()`

```
each $(selector).each(function(index,element))
  $("#div").each(function (index) {
    $(this).append(index);
  });
```

Modifier un élément

- **text(), html(), and val()**
`$("#id").text("Message!")`
`$("#id").html("<h1>Message!!</h1>")`
`$("#id").val("Message");`
- Attribut : **attr()**
`$("#id").attr("href", "http://www.google.com/");`

Modifier plusieurs attributs (json utilisé)

```
$("#id").attr({
  "href": "http://www.google.com/",
  "title": "Google"
});
```

Insérer avant, après

- `append()`, `appendto()` ajout à la fin
- `prepend()`, `prependto()` ajout au début
- `after()` ajout après
- `before()` ajout avant
- *plus ...* `wrap()`, `wrapAll()`, `wrapInner()`, `replaceWith()`, `replaceAll()`

```
$("#id").append("Append text.");
$("#id").prepend("Prepend text.");
$("#id").after("After text.");
$("#id").before("Before text.");
```

Before text.

Prepend text.[BASE TEXT]Append text.

After text.

Exemple `appendto` (idem pour `prependto`)

```
$("<p>A paragraph.</p>").appendTo("#div1");
```

Supprimer

- `remove()`
- `detach()`
- `empty()`

```
$("#id").remove();
$("p").remove(".class");
$(".class1,.class2").remove();
$("#id").empty();
```

CSS

- `addClass()`
- `hasClass()`
- `removeClass()`
- `toggleClass()` switch
- `css()`

```
$("#id").addClass("cssClass");
$("#id").addClass("class1 class2");
```

```
if ($("#div1").hasClass("class1")) { }
```

```
$("#div1").toggleClass("class1");
```

```

$("#id").removeClass("cssClass");

$("#id").css("background-color", "red");
$("#id").css({
    "background-color": "red",
    "font-size": "24pt"
});

```

Attribut « data »

- **Stocker et lire** une valeur

```

$("#div1").data("mykey", "my value");
var value = $("#div1").data("mykey");

```

- Lire un attribut HTML5 **data-***

Exemple dans la page html on a

```
<div id="div1" data-demo="data demo">my div</div>
```

Obtenir la valeur de l'attribut « data-demo » avec la méthode data()

```
var value = $("#div1").data("demo");
```

3. Events

Attachement d'événements en JavaScript (différence IE et autres browsers)

```

var button = document.getElementById('SubmitButton');
if (document.addEventListener) {
    button.addEventListener('click', function () { alert('Clicked Button'); }, false);
}
else {
    button.attachEvent('onclick', function () { alert('Clicked IE Button'); });
}

```

jQuery fournit un modèle d'événement « **cross-browser** » qui fonctionne aussi bien sous IE, Chrome, Safari, etc .

Liste des événements

Mouse	Keyboard	Form	Document/Window
click	keypress	submit	load
dblclick	keydown	change	resize
mouseenter	keyup	focus	scroll
mouseleave		blur	unload

Les événements peuvent être initialisés avant « ready »

```

<script>
    $(document).on("click", "#div1", function () { });

    $(document).ready(function () {
        //
    });
</script>

```


User input

```
// key
$("input").on("keypress", function (e) {
  console.log(e.which);
  console.log(e.offsetX, e.offsetY);
  console.log(e.altKey, e.ctrlKey);
});
// mouse
$("img").on("click", function (e) {
  console.log(e.pageX, e.pageY);
  console.log(e.offsetX, e.offsetY);
  console.log(e.screenX, e.screenY);
});
```

Events et ref events

On() et off()

`$(selector).on(event,childSelector,data,function,map)`

On() remplace Live(),delegate()et ,bind()

```
$("#btn").bind("click", function () { });
$("#btn").click(function () { });
// On
$("#btn").on("click", function () { });

$(".members").delegate("img", "click", function () { });
// live avant jQuery 1.9
$(document).live("click", ".members img", function () { });
// On
$(".members img").on("click", function () { });
```

```
$("#id").on("click", function () {
});
// replace live
$("#mytable").on("click", "tr", function (event) {
});
// Map
$("#div1").on({
  mouseenter: function () { alert("enter") },
  mouseleave: function () { alert("leave") }
})
```

Plusieurs événements

```
$("#id").on("mouseenter mouseleave", function () {
});
```

```
$("#id").off() ;
$("#id").off("mouseenter") ;
```

Hover()

Est l'équivalent de mouseenter/mouseleave

`$(selector).hover(handlerin,handlerout)`

```
$("#div1").hover(
  function () { alert("enter") },
  function () { alert("leave") }
);
```

4. Animation

ref effects

a. Afficher/ Masquer un élément avec show/hide

<http://api.jquery.com/category/effects/basics/>

```
<head>
  <meta http-equiv="Content-Type" content="text/html; charset=utf-8" />
  <title></title>
  <style>
    #box {
      position: relative;
      height: 400px;
      width: 400px;
      background-color: red;
    }
  </style>
</head>
<body>
  <button id="animateButton">Click me!</button>
  <div id="box"></div>

  <script src="Scripts/jquery-2.1.3.min.js"></script>
  <script>
    $(document).ready(function () {
      $("#animateButton").on("click", function () {
        var box = $("#box");
        if(box.is(":hidden")){
          box.show(3000);
        }
        else {
          box.hide(3000);
        }
      });
    });
  </script>
</body>
</html>
```

Par défaut la vitesse est de 400ms
. On peut indiquer une vitesse et
une fonction de retour

Ou plus simplement avec toggle

```
$("#box").toggle(3000);
```

b. Fondu avec fadeIn, fadeout, fadeToggle

<http://api.jquery.com/category/effects/fading/>

```
var box = $("#box");
if(box.is(":hidden")){
  box.fadeIn(1000);
}
else {
  box.fadeOut(1000);
}
```

Ou

```
$("#box").fadeToggle(1000);
```

c. Slide avec `slideDown`, `slideUp`, `slideToggle`

<http://api.jquery.com/category/effects/sliding/>

```
var box = $("#box");
if(box.is(":hidden")){
  box.slideDown(1000);
}
else {
  box.slideUp(1000);
}
```

Ou

```
$("#box").slideToggle(1000);
```

De gauche à droite

```
box.animate({ width: 'toggle' });
```

Fermeture vers la gauche, ouverture
vers la droite

d. Easing

jQuery n'a que « **swing** » (défaut) et « **linear** ». Sinon il faut utiliser **jQuery UI** ou **jQuery Easing plugin**

<http://easings.net/fr>

```
<head>
  <meta http-equiv="Content-Type" content="text/html; charset=utf-8" />
  <title></title>
  <style>
    #box {
      position: relative;
      height: 400px;
      width: 400px;
      background-color: red;
    }
  </style>
</head>
<body>

  <button id="animateButton">Click me!</button>
  <div id="box"></div>

  <script src="../Scripts/jquery-2.1.3.min.js"></script>
  <script src="../Scripts/jquery-ui-1.11.4.min.js"></script>
  <script>
    $(document).ready(function () {
      $("#animateButton").on("click", function () {
        $("#box").animate({
          "left": "250px",
        }, 2000, "easeOutElastic");
      });
    });
  </script>
</body>
</html>
```

Propriétés au format « JSON »

400ms par défaut. « slow » (600ms) « fast » (200ms) ou valeur en ms

« linear » ou « swing » sinon nécessite jQuery UI

e. Custom animations avec Animate

Animate a deux signatures

`.animate(properties [, duration] [, easing] [, complete])`

Et

`.animate(properties, options)`

Fonction callback

<http://api.jquery.com/category/effects/custom-effects/>

Toutes les propriétés CSS sont animables.

Animation couleur

Requiert le **plugin jQuery color**

```
<!DOCTYPE html>
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
  <meta http-equiv="Content-Type" content="text/html; charset=utf-8" />
  <title></title>
  <style>
    #box{
      position: relative;
      height:400px;
      width:400px;
      background-color:red;
    }
  </style>
</head>
<body>
  <button id="animateButton">Click me!</button>
  <div id="box"></div>

  <script src="Scripts/jquery-2.1.3.min.js"></script>
  <script src="Scripts/jquery.color-2.1.2.min.js"></script>
  <script>
    $(document).ready(function () {
      $("#animateButton").on("click", function () {
        $("#box").animate({
          backgroundColor: "blue"
        }, 2000);
      });
    });
  </script>
</body>
</html>
```

Animation de mouvement

Sur X

```
$("#box").animate({
  "left": "250px"
}, 2000);
```

```
$("#box").animate({
  "left": "+=250px"
}, 2000);
```

La boîte sera décalée à chaque de 250px sur la gauche. On peut utiliser « += » ou « -= »

Sur X et Y

```
$("#box").animate({
  "left": "250px",
  "top": "250px"
}, 2000);
```

Transformations

translate(x,y)

```
<head>
  <meta http-equiv="Content-Type" content="text/html; charset=utf-8" />
  <title></title>
  <style>
    #box {
      height: 200px;
      width: 200px;
      background-color: red;
    }
  </style>
</head>
<body>
  <button id="animateButton">Click me!</button>
  <div id="box"></div>

  <script src="../Scripts/jquery-2.1.3.min.js"></script>
  <script>
    $(document).ready(function () {
      $("#animateButton").on("click", function () {
        $({ t: 0 }).animate({ t: 400 },
          {
            duration: 3000,
            step: function (now) {
              $("#box").css('transform', 'translate(0px,' + now + 'px)');
            }
          }
        );
      });
    });
  </script>
</body>
</html>
```

Déplacement de 0 à 400px vers le bas pendant 3 secondes

scale(x,y)

Scale x2

```

$({ s: 1 }).animate({ s: 2 },
{
  duration: 3000,
  step: function (now) {
    $("#box").css('transform', 'scale(' + now + ')');
  }
});

```

rotate(deg)

Rotation de 90 degrés

```

$({ r: 0 }).animate({ r: 90 },
{
  duration: 3000,
  step: function (now) {
    $("#box").css('transform', 'rotate(' + now + 'deg)');
  }
});

```

Combinées

```

<head>
  <meta http-equiv="Content-Type" content="text/html; charset=utf-8" />
  <title></title>
  <style>
    h1 { text-align:center; }
  </style>
</head>
<body>
  <button id="animateButton">Click me!</button>
  <h1>Mon titre</h1>

  <script src="../Scripts/jquery-2.1.3.min.js"></script>
  <script>
    $(document).ready(function () {
      $("#animateButton").on("click", function () {
        var s = 0;
        var r = 0;

        $({ s: 0, r: 0 }).animate({ s: 2, r: 360 },
        {
          duration: 3000,
          step: function (now, fx) {
            if (fx.prop == 's')
              s = now;
            else if (fx.prop == 'r')
              r = now;

            $("h1").css('transform', 'scale(' + s + ') rotate(' + r +
'deg)');
          }
        });
      });
    });
  </script>
</body>
</html>

```

5. Ajax

<http://api.jquery.com/category/ajax/>

a. Load

Permet de charger une page HTML dans un conteneur de la page.

Ex

```
$("#div1").load("demo.html");
$("#div1").load("demo.html #div1");
$("#div1").load("demo.aspx", { PageSize:10 });
$("#div1").load("demo.aspx", function (response, status, xhr) {
    if (status == "error")
    { }
});
```

b. \$.get

`$.get(url,data,function(data,status,xhr){},datatype)`

Callback:

- data : résultat de la requête
- status - "success", "notmodified", "error", "timeout", "parsererror"
- xhr - XMLHttpRequest object

Datatype : "xml", "html", "text", "script", "json", "jsonp"

➔ Page html

```
$.get('../demo.html', function (data) {
    $('#div1').html(data);
});
```

La page html très simple

```
<div>
  <p>Some text and a <a href="#">link</a>.</p>
</div>
```

➔ Page Aspx

```
$.get('../demo.aspx', { id: 1 }, function (data) {
    $('#div1').html("<p>" + data.LastName + ", " + data.FirstName + "</p>");
}, 'json');
```

Exemple de page aspx (code-behind)

```
using System;
using System.Runtime.Serialization.Json;
using System.Web.UI;
namespace jQueryBasics
{
    public partial class demo : Page
    {
        protected void Page_Load(object sender, EventArgs e)
        {
            // on pourrait récupérer le paramètre pour filtrer Request["id"]
            Response.ContentType = "application/json";
            Person person = new Person() { FirstName = "Marie", LastName = "Bellin" };
            DataContractJsonSerializer serializer = new
            DataContractJsonSerializer(typeof(Person));
```

```

        serializer.SerializeObject(Response.OutputStream, person);
    }
}
public class Person
{
    public string FirstName { get; set; }
    public string LastName { get; set; }
}
}

```

c. \$.post

\$.post(url,data,callback,datatype);

Exemple avec un contrôleur Web Api

```

var Person = function(fname,lname)
{
    this.FirstName = fname;
    this.LastName = lname;
}
var patrick = new Person("Patrick","Prenium")

$.post('http://localhost:20820/api/People', patrick);

```

d. \$.getJSON

\$.getJSON(url,data,success(data,status,xhr))

La même chose avec \$.getJSON

```

$.getJSON('people.json', { id: 1 },function (data) {
    $("#div1").html("<p>" + data.LastName + ", " + data.FirstName + "</p>");
});

```

Avec Asp.Net indiquer dans web.config

```

<system.webServer>
  <staticContent>
    <mimeMap fileExtension=".json" mimeType="application/json; charset=UTF-8" />
  </staticContent>
</system.webServer>

```

a. \$.ajax

\$.ajax(options)

Options

- contentType
- data
- dataType
- error
- success
- type (GET ou POST)

```

var options = {
    url: "../api/People",
    type: "POST",
    contentType: 'application/json; charset=utf-8',
    data: JSON.stringify(patrick),

```



```

        success: function (data, status, xhr) {
            $('#div1').html('Insert status: ' +
                data.d.Status + '<br />' +
                data.d.Message);
        },
        error: function (xhr, status, error) {
            alert('Error occurred: ' + status);
        }
    }
};

$.ajax(options);

$.ajax({
    url: '../api/People',
    data: JSON.stringify(patrick),
    type: "POST",
    contentType: 'application/json; charset=utf-8',
    dataType: 'json',
    success: function (data, status, xhr) {
        $('#div1').html('Insert status: ' +
            data.d.Status + '<br />' +
            data.d.Message);
    },
    error: function (xhr, status, error) {
        alert('Error occurred: ' + status);
    }
});

```

serialize

```

$('form').submit(function() {
    alert($(this).serialize());
    return false;
});

```

Headers setRequestHeader et getRequestHeader

```

$.ajax({
    url: "/api/authentication",
    type: "POST",
    beforeSend: function (request) {
        request.setRequestHeader("AuthToken", authToken);
    }
});

authenticate(authToken)
    .success(function (data, statusText, jqXHR) {
        $("#authToken").html(
            jqXHR.getResponseHeader("AuthToken"));
    })
    .fail(function(jqXHR, statusText, err) {
        alert("Error authenticating: " + err);
    });

```

Créer un data service

Exemple dataService.js

```

var dataService = function () {
  var urlBase = '/api/customers',

  authenticate = function (authToken) {
    return $.ajax({
      url: "/api/authentication",
      type: "POST",
      beforeSend: function (request) {
        request.setRequestHeader("AuthToken", authToken);
      }
    });
  },

  getCustomers = function () {
    return $.getJSON(urlBase);
  },

  getCustomer = function (id) {
    return $.getJSON(urlBase + '/' + id);
  },

  insertCustomer = function (cust) {
    return $.ajax({
      url: urlBase,
      data: cust,
      type: 'POST'
    });
  },

  updateCustomer = function (cust) {
    return $.ajax({
      url: urlBase + '/' + cust.ID,
      data: cust,
      type: 'PUT'
    });
  },

  deleteCustomer = function (id) {
    return $.ajax({
      url: urlBase + '/' + id,
      type: 'DELETE'
    });
  };

  return {
    authenticate: authenticate,
    getCustomers: getCustomers,
    getCustomer: getCustomer,
    updateCustomer: updateCustomer,
    insertCustomer: insertCustomer,
    deleteCustomer: deleteCustomer
  };
}();

```

customersPage.js

```

var customersPage = function () {
  var init = function () {
    $('#GetCustomers').click(function () {
      getCustomers();
    });

    $('#UpdateCustomer').click(function () {
      updateCustomer();
    });

    $('#InsertCustomer').click(function () {
      insertCustomer();
    });

    $('#DeleteCustomer').click(function () {
      deleteCustomer();
    });
  },

  getCustomers = function () {
    dataService.getCustomers()
      .then(function (custs) {
        var custsHtml = ''
        for (var i = 0; i < custs.length; i++) {
          custsHtml += '<li>' + custs[i].FirstName + ' ' + custs[i].LastName
+ '&nbsp;</li>';
        }
        $('#CustomersContainer').html(custsHtml);
      },
      function(jqXHR, textStatus, err) {
        alert('Unable to get the customers: ' + textStatus);
      });
  },

  updateCustomer = function () {
    var cust = {
      ID: 1,
      FirstName: 'Marie',
      LastName: 'Bellin'
    };
    dataService.updateCustomer(cust)
      .then(function () {
        updateStatus('Updated.');
```

```

    }).
    fail(function (jqXHR, textStatus, err) {
        alert('Unable to insert customer: ' + textStatus);
    });
},

deleteCustomer = function () {
    dataService.deleteCustomer(10)
    .done(function () {
        updateStatus('Deleted Customer! Refreshing customer list.');
```

```

        getCustomers();
    })
    .fail(function (jqXHR, textStatus, err) {
        alert('Unable to delete customer: ' + textStatus);
    });
},

updateStatus = function (msg) {
    $('#OutputDiv').html(msg);
};

return {
    init: init
};

}());
```

Dans la page html référencer les scripts **après jQuery**.

```

<script src="Scripts/jquery.min.js"></script>
<script src="Scripts/customersPage.js"></script>
<script src="Scripts/dataService.js"></script>
```

a. Promises et \$.when

```

function getCustomers() {
    return $.getJSON("/api/customers/");
};

getCustomers()
    .done(function (data) { })
    .fail(function (xhr, status, error) { });
```

Multiples appels avec \$.when()

```

$.when(ajaxCall1(), ajaxCall2())
    .done(function (data1, data2) { })
    .fail(function (xhr, status, error) { });
```

Passing Array (meilleures performances)

```

getCustomerAndOrders = function (id) {
    var promises = [getCustomer(custID), getOrders(custID)];
    return $.when(promises);
},
```

```

return $.when.apply($, promises);|
```

6. jQuery Validation

Site de jQuery Validation plugin

```

<head>
  <meta http-equiv="Content-Type" content="text/html; charset=utf-8" />
  <title>jQuery validation plugin demo</title>
  <style>
    .error {
      border-color: red;
      color: red;
    }
  </style>
</head>
<body>
  <form>
    <fieldset>
      <legend>Se connecter</legend>
      <label for="userName">Nom</label>
      <input type="text" id="userName" name="userName" /><br />
      <label for="password">Mot de passe</label>
      <input type="password" id="password" name="password" />
    </fieldset>
    <input type="submit" value="Submit" />
  </form>

  <script src="Scripts/jquery-2.1.3.min.js"></script>
  <script src="Scripts/jquery.validate.min.js"></script>
  <script>
    $('form').validate({
      rules: {
        userName: {
          required: true
        },
        password: {
          required: true
        }
      },
      messages: {
        userName: {
          required: "Entrez votre nom d'utilisateur"
        },
        password: {
          required: "Entrez votre mot de passe"
        }
      },
      submitHandler: function (form) { },
      invalidHandler: function (event, validator) {
        //alert(validator.numberOfInvalids());
      }, highlight: function (element, errorClass) {
        //$(element).addClass('has-error');
      }, unhighlight: function (element, errorClass) {
        //$(element).removeClass('has-error');
      }
    });
  </script>
</body>

```

Classe CSS automatiquement appliquée à l'élément en erreur

Règles de validation pour chaque champ (required, minlength, email, etc.)

liste

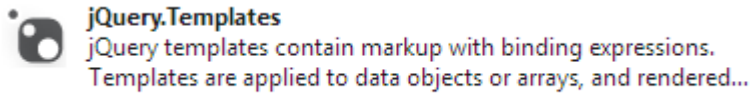
Messages personnalisés.
Pour ne pas avoir un message en anglais par exemple

Méthodes utilisables pour appliquer par exemple un style

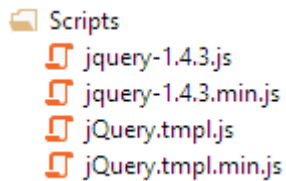
7. jQuery Templates

Installation du plug in:

- Package NuGet



- Ou téléchargement des fichiers <http://plugins.jquery.com/jquery-tmpl/>
Sont ajoutés...



Référencer d'abord jQuery puis le plug in

```
<script src="Scripts/jquery-1.4.3.min.js"></script>
<script src="Scripts/jQuery.templ.min.js"></script>
```

tmpl

`$.tmpl()`

`tmpl(tmpl, data, options, parentItem)`
Return wrapped set of template items, obtained by rendering template against data.

`$("selector").tmpl()`

`tmpl(data, options, parentItem)`
Use first wrapped element as template markup.
Return wrapped set of template items, obtained by rendering template against data.

- Ajout d'html au conteneur sélectionné

```
<div id="personContainer"></div>
```

```
<script type="text/javascript">
  var person = { FirstName: "Marie", LastName: "Bellin" };

  $(function () {
    $.tmpl("<div class='person'>${FirstName} <em>${LastName}</em></div>",
    person).appendTo("#personContainer");
  });
</script>
```

- Utilisation de template défini dans un **fichier** (grâce à Ajax)

La page html servant de template (PersonTemplate.html)

```

<tr>
  <td>${FirstName}</td>
  <td>${LastName}</td>
</tr>

```

```

<script type="text/javascript">
  var people = [
    { FirstName: "Marie", LastName: "Bellin" },
    { FirstName: "Patrick", LastName: "Prenium" }
  ];

  $(function () {
    $.get("../PersonTemplate.html", function (data) {
      $.tmpl(data, people).appendTo("#peopleList");
    });
  });
</script>

```

Dans la page dans laquelle le code sera inséré

```

<table>
  <thead>
    <tr>
      <th>FirstName</th>
      <th>LastName</th>
    </tr>
  </thead>
  <tbody id="peopleList">
    <!-- code insert here -->
  </tbody>
</table>

```

- Utilisation **Template**

```

<script type="text/javascript">
  var people = [
    { FirstName: "Marie", LastName: "Bellin", Birth: new Date(1990, 6, 10),
Mails: [{ Mail: "mb3@laposte.net" }, { Mail: "mb3@hotmail.fr" } ] },
    { FirstName: "Patrick", LastName: "Prenium", Birth: new Date(1980, 1, 1) }
  ];

  $(function () {
    $("#peopleTemplate").tmpl(people).appendTo("#peopleList");
  });

  function formatDate(inputDate) {
    return inputDate.toDateString();
  };
</script>
<!--Template-->
<script id="personTemplate" type="text/x-jquery-tmpl">
  <tr>
    <td>${FirstName}</td>
    <td>${LastName}</td>
    <td>${formatDate(Birth)}</td>
  </tr>

```

```

        {{if Mails }}
        <ul>
            {{each Mails}}
            <li>${Mail}</li>
            {{/each}}
        </ul>
        {{else}}
        <span>No mail</span>
        {{/if}}
    </td>
</tr>
</script>

```

→ On peut faire évoluer le template par exemple

```

<script id="peopleTemplate" type="text/x-jquery-tmpl">
    <tr>
        <td>${FirstName}</td>
        <td>${LastName}</td>
        <td>${formatDate(Birth)}</td>
        <td>
            {{if Mails }}
            {{tpl "#mailsTemplate"}}
            {{else}}
            <span>No mail</span>
            {{/if}}
        </td>
    </tr>
</script>
<script id="mailsTemplate" type="text/x-jquery-tmpl">
    <ul>
        {{each Mails}}
        <li>${Mail}</li>
        {{/each}}
    </ul>
</script>

```

→ Autre possibilité « single template » (template pour un seul élément)

```

    {{if Mails }}
    <ul>
        {{each Mails}}
        {{tpl(Mails) "#mailTemplate"}}
        {{/each}}
    </ul>
    {{else}}
    <span>No mail</span>
    {{/if}}

```

```

<script id="mailTemplate" type="text/x-jquery-tmpl">
    <li>${item.parent.data.FirstName} has ${Mail}</li>
</script>

```

Template tags

- ✓ \${expression}
- ✓ {{if}} {{else}} {{/if}}
- ✓ {{each}} {{/each}}
- ✓ Etc.

\$item

« \$item.data » et « \$item.parent.data » pour accéder aux éléments parents

Caching

Ajout en cache

```
$.template("cachedSelectedPersonTemplate", $("#selectedPersonTemplate"));
```

Retrouver/ utiliser

```
currentPerson.tpl = $.template("cachedSelectedPersonTemplate");
currentPerson.update();
```

Vue Master/details

People list

FristName	LastName	Birth	Mail(s)	Details
Marie	Bellin	Tue Jul 10 1990	<ul style="list-style-type: none"> mb3@laposte.net xx@hotmail.fr 	Bellin, Marie
Patrick	Premium	Fri Feb 01 1980	No mail	

```
<!DOCTYPE html>
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
  <meta http-equiv="Content-Type" content="text/html; charset=utf-8" />
  <title></title>
  <script src="Scripts/jquery-1.4.3.min.js"></script>
  <script src="Scripts/jquery.tpl.min.js"></script>
  <script type="text/javascript">

    var people = [
      { FirstName: "Marie", LastName: "Bellin", Birth: new Date(1990, 6, 10),
Mails: [{ Mail: "mb3@laposte.net" }, { Mail: "xx@hotmail.fr" } ] },
      { FirstName: "Patrick", LastName: "Premium", Birth: new Date(1980, 1, 1) }
    ];

    var currentPerson = null;

    $(function () {

      $("#peopleList").delegate(".person","click", function () {
        // caching
        $.template("personTemplate", $("#peopleTemplate"));
        $.template("cachedSelectedPersonTemplate",
$("#selectedPersonTemplate"));
        // si une personne était sélectionnée
        // on annule "le surlignement gris"
        if (currentPerson)
        {
          currentPerson.tpl = $.template("personTemplate");
          currentPerson.update();
        }

        $("#personContainer").empty();
```

```

        currentPerson = $.tmplItem($(this));
        currentPerson.tmpl = $.template("cachedSelectedPersonTemplate");
        currentPerson.update();
        // affichage du "détail" de la personne sélectionnée dans la div
"personContainer"
        $.tmpl("<h4>Details</h4>${LastName}, ${FirstName}",
currentPerson.data).appendTo("#personContainer");

        $("selector").tmpl()
    });
});
function formatDate(inputDate) {
    return inputDate.toDateString();
};
</script>
<!--Templates -->
<script id="peopleTemplate" type="text/x-jquery-tmpl">
    <tr class="person">
        <td>${FirstName}</td>
        <td>${LastName}</td>
        <td>${formatDate(Birth)}</td>
        <td>
            {{if Mails }}
                {{tmpl "#mailsTemplate"}}
            {{else}}
                <span>No mail</span>
            {{/if}}
        </td>
    </tr>
</script>
<script id="selectedPersonTemplate" type="text/x-jquery-tmpl">
    <tr class="person" style="background-color:gray">
        <td>${FirstName}</td>
        <td>${LastName}</td>
        <td>${formatDate(Birth)}</td>
        <td>
            {{if Mails }}
                {{tmpl "#mailsTemplate"}}
            {{else}}
                <span>No mail</span>
            {{/if}}
        </td>
    </tr>
</script>
<script id="mailsTemplate" type="text/x-jquery-tmpl">
    <ul>
        {{each Mails}}
        <li>${Mail}</li>
        {{/each}}
    </ul>
</script>
</head>
<body>
    <div style="float: left">
        <h2>People list</h2>
        <table>
            <thead>
                <tr>
                    <th>FristName</th>
                    <th>LastName</th>
                    <th>Birth</th>
                    <th>Mail(s)</th>

```

```

        </tr>
      </thead>
      <tbody id="peopleList">
        <!-- -->
      </tbody>
    </table>
  </div>
  <div id="personContainer" style="float: left; margin-left: 50px; margin-
top: 50px"></div>
</body>
</html>

```

D'autres solutions

Il existe d'autres solutions pour faire du binding telles que [Knockout](#) ou [Angular](#)

8. Plugin

a. Créer un plugin

On crée un fichier au nom du plugin exemple « `jquery.polaroid.js` » (utiliser `jquery` devant le nom du plugin étant une bonne pratique)

```

(function ($) {
  'use strict';

  $.fn.polaroid = function (options) {
    var opts = $.extend({
      wrapperClass: 'polaroid',
      bgColor: null,
      color: null,
      onLoad: $.noop()
    }, options);

    return this.each(function (idx, el) {
      var $img = $(el);
      $img
        .wrap('<div class="" + opts.wrapperClass + ""></div>')
        .after('<span class="caption">' + $img.attr('title') + '</span>');

      // Apparence
      if (opts.bgColor) {
        $img.parent().css('background-color', opts.bgColor);
      }
      if (opts.color) {
        $img.siblings('.caption').css('color', opts.color);
      }

      // callback
      opts.onLoaded(idx, el)
    });
  };
}(jQuery));

```

Options avec valeurs par défaut
« `onLoaded` » représentant un callback

Retourner « `this` » permet le chaining de CSS

Callback

On pourrait également valider les entrées utilisateurs

```
var defaultOptions = $.extend({
  a.   wrapperClass: 'polaroid',
      bgColor:null,
      color: null,
      onLoad:$.noop()
});
var opts = $.extend({}, defaultOptions, options);
//if( .....
```

On définit dans un premier temps des valeurs par défaut qui remplaceront les valeurs user si elles sont invalides

Utilisation du plugin

```
<script src="https://ajax.googleapis.com/ajax/libs/jquery/2.1.3/jquery.min.js"></script>
<script>window.jQuery || document.write('<script src="js/jquery.min.js"></script>')</script>
<script src="js/jquery.polaroid.js"></script>
<script>
$(document).ready(function() {
  //$('img').polaroid();
  //$('img').polaroid({ 'wrapperClass':'mypolaroid', 'bgColor':'#000','color': '#fff' });
  //$('img').polaroid({ 'bgColor':'#000','color':'#fff' });

  // call back
  $('img').polaroid({ 'bgColor':'#000','color':'#fff', onLoad:function(idx,el){
    console.log(idx + ' ' + el);
  }
  });
});
</script>
```

On importe jQuery puis le plugin

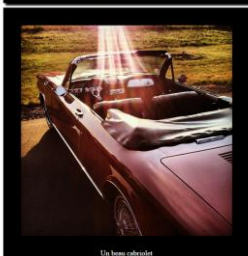
Plusieurs exemples.
« mypolaroid » étant un style CSS à appliquer

HTML pour l'exemple

Des images sont « transformées » en polaroid, le titre étant utilisé comme caption.

```


```



Le code étant modifié par le plugin

```
▼ <body>
  ▼ <div class="polaroid" style="background-color: rgb(0, 0, 0);">
    
    <span class="caption" style="color: rgb(255, 255, 255);">Une belle voiture</span>
  </div>
  ▶ <div class="polaroid" style="background-color: rgb(0, 0, 0);">...</div>
  <script src="https://ajax.googleapis.com/ajax/libs/jquery/2.1.3/jquery.min.js"></script>
  <script src="js/jquery.polaroid.js"></script>
  ▶ <script>...</script>
</body>
```

Publier son plugin

sur son site personnel, sur github, npm et bower

b. Trouver des plugins existants

<http://plugins.jquery.com/>

Quelques plugins intéressants :

Slideshows : [Cycle2](#), [Flexslider](#), [Colorbox](#)

Lecteur vidéo : [Medialementjs](#)

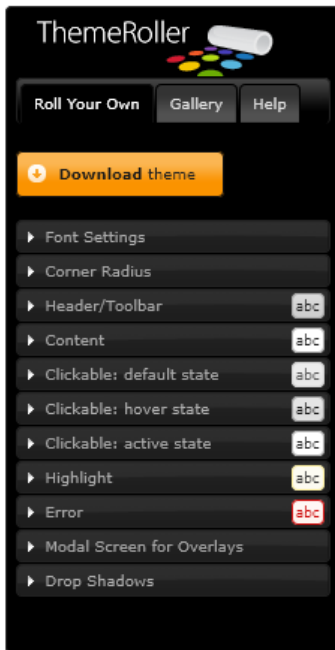
II-jQuery UI

Besoins : jQuery + jQuery UI (feuille de style et script)

```
<link rel="stylesheet"
href="//ajax.googleapis.com/ajax/libs/jqueryui/1.11.0/themes/smoothness/jquery-ui.css" />
<script src="//ajax.googleapis.com/ajax/libs/jquery/2.1.1/jquery.min.js"></script>
<script src="//ajax.googleapis.com/ajax/libs/jqueryui/1.11.0/jquery-ui.min.js"></script>
```

1. Theme roller

Themeroller

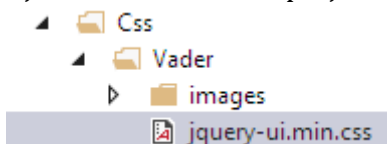


1. Télécharger un thème de la galerie ou un thème personnalisé. Il est possible de choisir la version ainsi que les éléments à inclure (permettant ainsi d'alléger le fichier)

Exemple je télécharge le thème « Vader »



2. Ajouter les fichiers au projet



3. Il suffit de changer la feuille de style pour le thème téléchargé

```
<link href="Css/Vader/jquery-ui.min.css" rel="stylesheet" />
<script src="//ajax.googleapis.com/ajax/libs/jquery/2.1.1/jquery.min.js"></script>
<script src="//ajax.googleapis.com/ajax/libs/jqueryui/1.11.0/jquery-ui.min.js"></script>
```

2. Widgets jQuery UI

Pour chaque Widget :

✓ Options

```
$("#dialog").dialog(
  {
    autoOpen: false,
    modal: true,
    resizable: false
  }
);
```

✓ Méthodes

```
$("#dialog").dialog("close");
```

✓ Events

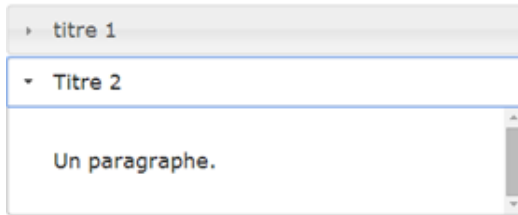
```
$("#dialog").on("dialogopen", function( event, ui ) {});
```

Pas d'attribut data-* spécifique à jQuery UI ni jQuery

<http://jqueryui.com/demos/>

Transformer du **Html** (exemple `<div id="#id">...</div>`) ... **en widget**

```
<script>
  $(function () {
    $("#id").accordion();
  });
</script>
```

a.Accordion<http://api.jqueryui.com/accordion/>

```

<div id="accordion">
  <h3>
    <a href="#">titre 1</a>
  </h3>
  <div>
    <p>
      Un paragraphe.
    </p>
  </div>
  <h3>
    <a href="#">Titre 2</a>
  </h3>
  <div>
    <p>
      Un paragraphe.
    </p>
  </div>
</div>

```

```
$("#accordion").accordion();
```

Options

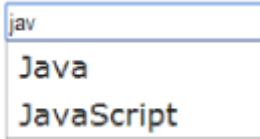
```

active
animate
collapsible
disabled
event
header
heightStyle
icons

```


b.Autocomplete

<http://api.jqueryui.com/autocomplete/>



```
<div>
  <input id="tags">
</div>
```

```
var availableTags = [
    "ActionScript",
    "AppleScript",
    "Asp",
    "BASIC",
    "C",
    "C++",
    "Clojure",
    "COBOL"
];

$("#tags").autocomplete({
  source: availableTags
});
```

Options

```
appendTo
autoFocus
delay
disabled
minLength
position
source
```

c.Button<http://api.jqueryui.com/button/>

A button element

An input submit button

An anchor button

```
<div id="buttons">
  <button>A button element</button>
  <input type="submit" value="An input submit button" />
  <a href="#">An anchor button</a>
</div>
```

```
$("#buttons").children()
    .button({ icons: { primary: "ui-icon-search", secondary: "ui-icon-
wrench" } })
    .click(function () {
        alert($(this).value);
    });
```

On peut également transformer des cases à cocher/boutons radio en boutons

```
<div id="radios">
  <input type="radio" id="radiobutton1" /><label for="radiobutton1">radiobutton
1</label>
  <input type="radio" id="radiobutton2" /><label for="radiobutton2">radiobutton
2</label>
</div>
```

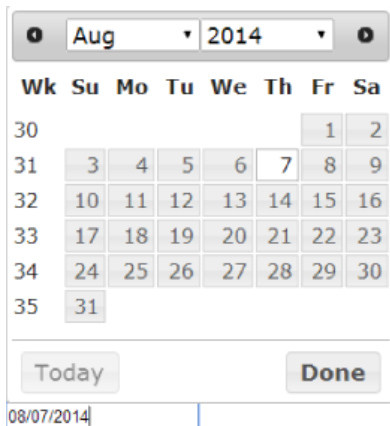
```
$("#radios").buttonset();
```

Options

```
disabled
icons
label
text
```

d. Datepicker

<http://api.jqueryui.com/datepicker/>



```
<div>
  <input type="text" id="dateselection" />
</div>
```

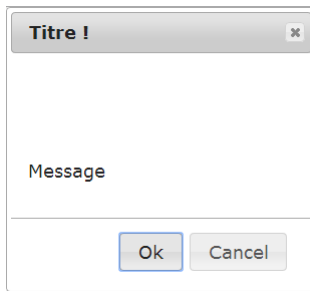
```
$("#dateselection").datepicker({
  showAnim: "bounce",
  showWeek: true,
  changeMonth: true,
  changeYear: true,
  showButtonPanel: true,
  minDate: new Date(2010, 1, 1),
  maxDate: new Date(2014, 12, 31)
});
//
$("#dateselection").datepicker("setDate", new Date(2014, 7, 7));
```

Options

altField	dayNamesMin	onSelect
altFormat	dayNamesShort	prevText
appendText	defaultDate	selectOtherMonths
autoSize	duration	shortYearCutoff
beforeShow	firstDay	showAnim
beforeShowDay	gotoCurrent	showButtonPanel
buttonImage	hideIfNoPrevNext	showCurrentAtPos
buttonImageOnly	isRTL	showMonthAfterYear
buttonText	maxDate	showOn
calculateWeek	minDate	showOptions
changeMonth	monthNames	showOtherMonths
changeYear	monthNamesShort	showWeek
closeText	navigationAsDateFormat	stepMonths
constrainInput	nextText	weekHeader
currentText	numberOfMonths	yearRange
dateFormat	onChangeMonthYear	yearSuffix
dayNames	onClose	

e.Dialog

<http://api.jqueryui.com/dialog/>



```
<div id="dialog" title="Titre ! ">
  
  <p>
    Message
  </p>
</div>
<input type="button" value="Open dialog" id="openDialog"/>
```

```
// dialog + boutons ok cancel
$("#dialog").dialog(
{
  autoOpen: false,
  modal: true,
  resizable: false,
  draggable: false,
  position: "top",
  buttons: {
    "Ok": function () {
      $("#dialog").dialog("close");
      $("#dialogDecision").text("Yes you did!!");
    },
    "Cancel": function () { }
  }
});

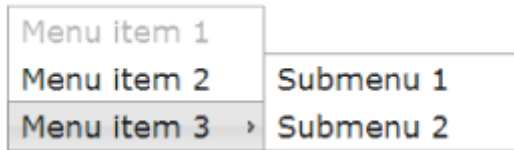
// bouton ouvrant la fenêtre de dialog
$("#openDialog").click(function () {
  $("#dialog").dialog("open");
});
```

Options

appendTo	maxWidth
autoOpen	minHeight
buttons	minWidth
closeOnEscape	modal
closeText	position
dialogClass	resizable
draggable	show
height	title
hide	width
maxHeight	

f.Menu

<http://api.jqueryui.com/menu/>



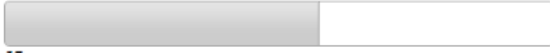
(définir le style de ul , width)

```
<ul>
  <li class="ui-state-disabled">Menu item 1</li>
  <li>Menu item 2</li>
  <li>
    Menu item 3
    <ul>
      <li>Submenu 1</li>
      <li>Submenu 2</li>
    </ul>
  </li>
</ul>
```

```
$("#menu").menu();
```

Options

disabled
icons
menus
position
role

g.Progressbar<http://api.jqueryui.com/progressbar/>

57

```
<div id="progress"></div>
<div id="countdown"></div>
```

```
$("#progress").progressbar({ value: 50 });

// simule un changement
var value = 0;
countdown();

function countdown() {
    value++;
    $("#progress").progressbar("option", "value", value);
    $("#countdown").text(value);

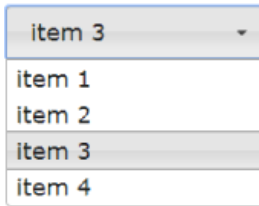
    if (value < 100) {
        setTimeout(countdown, 100);
    }
    else {
        $("#countdown").text("completed");
        $("#progress").progressbar("disable");
    }
}
```

Options

```
disabled
max
value
```

h.Selectmenu

<http://api.jqueryui.com/selectmenu/>



```
<select id="slmenu" style="width:200px">
  <option>item 1</option>
  <option>item 2</option>
  <option selected="selected">item 3</option>
  <option>item 4</option>
</select>
```

```
$("#slmenu").selectmenu();
```

Options

```
appendTo
disabled
icons
position
width
```

i.Slider

<http://api.jqueryui.com/slider/>



```
<div id="slider"></div>
```

```
$("#slider").slider({
  min: 0, max: 100, value: 100, slide: function (event, ui) {
    $("#slider").prev().css({ opacity: ui.value / 100 });
  }
});
```

Options

```
animate
disabled
max
min
orientation
range
step
value
values
```

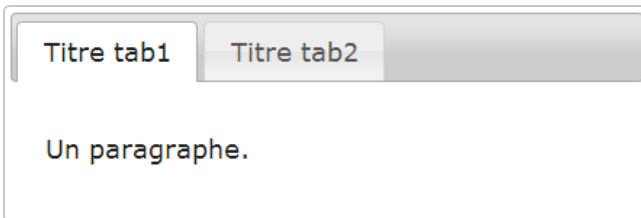
j.Spinner<http://api.jqueryui.com/spinner/>

```
<input id="spinner">
```

```
$("#spinner").spinner();
```

Options

- culture
- disabled
- icons
- incremental
- max
- min
- numberFormat
- page
- step

k.Tabs<http://api.jqueryui.com/tabs/>

```
<div id="tabs">
  <ul>
    <li><a href="#tab-1">Titre tab1</a></li>
    <li><a href="#tab-2">Titre tab2</a></li>
  </ul>
  <div id="tab-1">
    <p>
      Un paragraphe.
    </p>
  </div>
  <div id="tab-2">
    <p>
      Un paragraphe pour tab-2.
    </p>
  </div>
</div>
```

```
$("#tabs").tabs();
```

Options

- active

collapsible
disabled
event
heightStyle
hide
show

l.Tooltip (visuel amélioré par rapport au tooltip par « défaut »)

<http://api.jqueryui.com/tooltip/>

Mon tooltip

Un lien

```
<a href="#" title="Mon tooltip">Un lien</a>
```

```
$(document).tooltip();
```

Options

content
disabled
hide
items
position
show
tooltipClass
track

4. Interactions

Draggable

```
<div id="d1">
  <header>Draggable 1</header>
</div>
```

```
$("#d1").draggable();
// avec retour
$("#d1").draggable({
  revert: "invalid"
});
```

Droppable

```
<div id="droppable">
  
</div>
```

```
$("#droppable").droppable({
  activeClass: "opaque",
  accept: "#d1",
  drop: function (event, ui) {
    ui.draggable.fadeOut(function () {
      ui.draggable.remove();
    });
  }
});
```

Sortable

Permet de glisser/déposer les éléments d'une liste par exemple pour les trier

```
<ul id="sortable">
  <li>London</li>
  <li>Berlin</li>
  <li>Bruxelles</li>
</ul>
```

```
$("#sortable").sortable({ axis: "y" });
```

Resizable

```
<div id="resizable" class="ui-widget-content">
  <h3 class="ui-widget-header">Resizable</h3>
</div>
```

```
$("#resizable").resizable();
```

5. Effects

Colors

```
<div id="effects"></div>
```

```
$("#effects").on("click",function(){
    $(this).animate({
        "width": "+=20",
        "background-color":"red"
    });
});
```

Effect()

Blind	Highlight
Bounce	Puff
Clip	Pulsate
Drop	Scale
Explode	Shake
Fade	Slide
Fold	Transfer

<http://jqueryui.com/effect/>

```
$("#effects ").on("click", function () {
    $(this).effect("bounce");
});
```

Easing

<http://jqueryui.com/easing/>

```
$("#effects").css("position", "relative").on("click", function () {
    $(this).animate({
        left: 300
    },1000, "easeInOutElastic");
});
```

addClass

```
<style type="text/css">
    .newClass {
        background-color:green;
    }
</style>
```

```
$("#addingclass").on("click", function () {
    $(this).addClass("newClass", "slow",callback);
});
function callback() {
    setTimeout(function () {
        $("#addingclass").removeClass("newClass");
    }, 1500);
}
```

toggleClass

```
$("#addingclass").on("click", function () {
    $(this).toggleClass("newClass", "slow");
});
```

III-jQuery Mobile

<http://jquerymobile.com/>

Besoins : jQuery + jQuery Mobile (feuille de style et script)

```
<meta name="viewport" content="width=device-width, initial-scale=1">
<link rel="stylesheet" href="//ajax.googleapis.com/ajax/libs/jquerymobile/1.4.3/jquery.mobile.min.css"
/>
<script src="//ajax.googleapis.com/ajax/libs/jquery/2.1.1/jquery.min.js"></script>
<script src="//ajax.googleapis.com/ajax/libs/jquerymobile/1.4.3/jquery.mobile.min.js"></script>
```

1. ThemeRoller pour jQuery Mobile

<http://themeroller.jquerymobile.com/>

2. Page

ViewPort

- width (exemple width="device-width" ou en px)
- height (exemple height="device-height" ou en px)
- initial-scale zoom initial (exemple initial-scale="1.0")
- minimum-scale zoom minimal (exemple minimum-scale="0.5")
- maximum-scale zoom maximal (exemple maximum-scale="3.0")
- user-scalable possibilité ou non de zoomer (exemple user-scalable="yes")
- target-densitydpi résolution, en dpi

@media

```
@media screen and (max-width: 640px) {
  /**/
}
```

Création de Page jQuery MOBILE

```
<div data-role="page" data-theme="b">
  <div data-role="header" data-position="fixed"></div>
  <div data-role="content"></div>
  <div data-role="footer" data-position="fixed"></div>
</div>
```

Note : il est possible de définir plusieurs « pages jqm » dans une même page html

Mettre en cache

```
<div data-role="page" id="pg1" data-dom-cache="true"></div>
```

Data-* de page

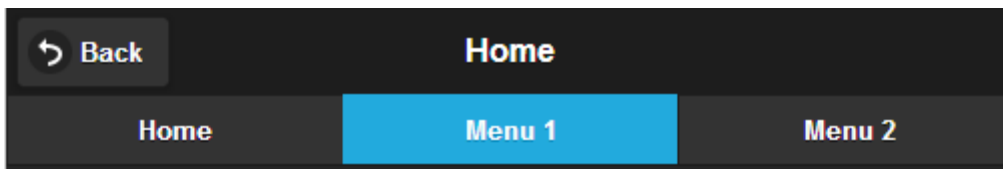
data-dom-cache	true false	Cache
data-overlay-theme	letter	overlay (background) color
data-theme	letter	Thème page
data-title	sometext	Titre de la page
data-url	url	

Header et navbar

```
<div data-role="header" data-position="fixed">
  <a data-rel="back" data-icon="back">Back</a>
  <h1>Home</h1>
  <div data-role="navbar">
    <ul>
      <li><a href="#">Home</a></li>
      <li><a href="#" class="ui-btn-active">Menu 1</a></li>
      <li><a href="#">Menu 2</a></li>
    </ul>
  </div>
</div>
```

Data-* pour navbar

data-icon		icône
data-iconpos	left right top bottom notext	Position icône



Data-role

- collapsible
- collapsible-set
- content**
- dialog
- fieldcontain
- footer
- header
- navbar
- page

Navigation

Lien vers une « **page jqm** de la même page html » ` ...`

Lien vers une autre **page Html** ` page 2 `

Bouton **retour** `<a data-rel="back" data-icon="back">Back`

Liens et \$.mobile.changePage()

```
<a href="#pg2">Page 2</a>
<a href="page3.html">Page 3</a>
<a href="#dial1" data-rel="dialog" data-role=button>Fenêtre de dialogue</a>
<a href="page4.html" data-rel="dialog" data-role=button>Autre fenêtre de dialogue</a>
<a href="https://fr.yahoo.com/">lien externe</a>
<a href="mailto:mb123@live.com">Envoyer un mail</a>
<a href="sms:0638000000">Envoyer un SMS</a>
<a href="tel:0638000000">Téléphoner</a>
```

Note : jQuery fait un appel Ajax pour une page située sur le même serveur (exemple : « page3.htm ») avec l'icône de chargement (data-ajax="false" pour le désactiver l'appel ajax, rel= « external » désactive l'appel ajax)

\$.mobile.changePage

\$.mobile.changePage (toPage, options)

ChangePage

```
<a href="#" id="link1">Lien dynamique</a>
<script>
  $("#link1").on("click", function (event) {
    $.mobile.changePage("page1.html");
  })
</script>
```

```
$.mobile.changePage($("#pg2"));
$.mobile.changePage(@Url.Action("Index", "Home"));
Passage de paramètres + plusieurs options définies
```

```
<script>
  $("#link1").on("click", function (event) {
    $.mobile.changePage("page2.aspx",
      {
        data: { firstname: "Marie", lastname: "Bellin" },
        transition: "flip"
      }
    );
  })
</script>
```

Récupération des paramètres dans page2.aspx. En C# dans le code-behind

```
public partial class page2 : Page
{
  protected void Page_Load(object sender, EventArgs e)
  {
    Label1.Text = string.Format("{0}, {1}",
Request["lastname"], Request["firstname"]);
  }
}
```

Prefetch

```
<a href="Page2.html" data-prefetch="true"> Go to page 2 </a>
```

Transitions

Slide la fenêtre arrive par la droite
Slideup la fenêtre arrive par le bas
Slidedown La fenêtre arrive par en haut
slidefade
Fade la fenêtre disparaît (opacité ...0) , la seconde apparaît
Pop La fenêtre apparaît par le centre de la première jusqu'à la recouvrir
Flip rotation de la fenêtre
turn
flow
none

```
<a href="#" data-transition="flip"> Go to page 2 </a>
```

3. Widgets jQuery Mobile

<http://api.jquerymobile.com/category/widgets/>

Chaque Widget jQuery Mobile a des

- ✓ **options**
- ✓ **méthodes**
- ✓ **events**

Exemple (button)

Options	Methods	Events
corners	destroy	create
disabled	disable	
enhanced	enable	
icon	option	
iconpos	refresh	
iconshadow		
initSelector		
inline		
mini		
shadow		
theme		
wrapperClass		

1. Définir le **data-role** du widget

Exemple `My button`

Ou Transformer un élément HTML en Widget jqm

Exemple transformation d'un lien en bouton `My button`

```
$(document).on("pagecreate", function (event) {
    $("#btn").button();
});
```

2. Définir les **options** à la création

```
$(document).on("pagecreate", function (event) {
    $("#btn").button({ corners: false, icon: "action" });
});
```

... **Après création** (dans « pagecreated » ou dans script jquery)

```
$("#btn").on("mouseenter", function (event) {
    $(this).button("option", "icon", "star");
});
```

3. Utilisation des **méthodes**

```
$("#btn").button("disable");
```

4. Utilisation **événements**

```
$(document).on("pagecreate", function (event) {

    $("#btn").button({
        create: function (event, ui) {
            alert("created");
        }
    });
    // Ou
    $("#btn").on("buttoncreate", function (event, ui) {
        alert("created in jquery");
    });
    $("#btn").button();
});
```

Data-* attributes par widget

<http://api.jquerymobile.com/data-attribute/>

http://www.w3schools.com/jquerymobile/jquerymobile_ref_data.asp

Css Framework (*Classes ui-**)

<http://api.jquerymobile.com/classes/>

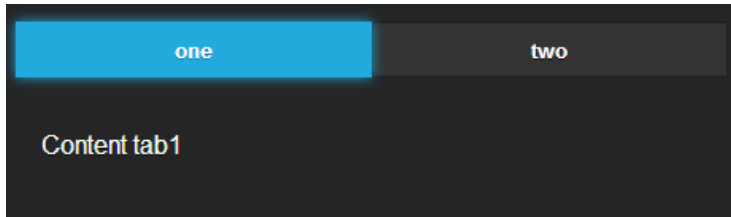
http://www.w3schools.com/jquerymobile/jquerymobile_ref_css.asp

Attributs data-* et jqmData()

jqmData() permet de gérer facilement les attributs jQuery (« data-... ») à la place de attr()

```
alert($("#div#pg1").jqmData("theme")); // lecture
$("#div#pg1").jqmData("theme", "b"); // Affectation
alert($("#div:jqmData(role=page)").html())
alert($("#div:jqmData(role)").length);
```


a. Tabs



```
<div id="tabs" data-role="tabs">
  <div data-role="navbar">
    <ul>
      <li><a href="#tab1" data-ajax="false">one</a></li>
      <li><a href="#tab2" data-ajax="false">two</a></li>
    </ul>
  </div>
  <div class="ui-body-d ui-content" id="tab1">
    <p>Content tab1</p>
  </div>
  <div id="tab2">
    <p>Content tab2</p>
  </div>
</div>
```

<http://api.jqueryui.com/tabs/>

b. Popup

<http://demos.jquerymobile.com/1.4.3/popup/>

```
<a class="ui-btn ui-corner-all ui-shadow ui-btn-inline" href="#popupBasic" data-rel="popup" data-transition="pop">Basic Popup</a>
  <div id="popupBasic" data-role="popup">
    <p>Popup content</p>
  </div>
```

+Popup photo...

Tooltip

```
<p>A paragraph with a tooltip. <a href="#popupInfo" class="ui-btn ui-alt-icon ui-nodisc-icon ui-btn-inline ui-icon-info ui-btn-icon-notext" data-rel="popup" data-transition="pop">here</a></p>
  <div class="ui-content" id="popupInfo" data-role="popup" data-theme="a">
    <p>Tootip</p>
  </div>
```

Data-*

data-corners **true** | false
 data-dismissible **true** | false – la popup se ferme si on clique en dehors
 data-history **true** | false – création ou non d'un historique
 data-overlay-theme letter
 data-position-to **origin** | window | selector position de la popup
 data-shadow **true** | false
 data-theme letter
 data-tolerance 30,15,30,15 - Distance(top, right, bottom, left)

data-rel

data-transitionfade | flip | flow | pop | slide | slidedown | slidefade | slideup | turn | **none** –
 Transition quand la popup est affichée/cachée

Options

```
arrow
corners
defaults
disabled
dismissible
history
initSelector
overlayTheme
positionTo
shadow
theme
tolerance
transition
```

c. Boite de dialogue

```
<a href="#dial1" data-rel="dialog" data-transition="pop">Boîte de dialogue</a>
<div data-role="page" id="dial1">
  <div data-role="header">
    <h1>titre de la boite de dialogue</h1>
  </div>
  <div data-role="content">
    <p>Contenu</p>
  </div>
</div>
```

d. Fenêtre superposée

```
<a href="#dial1" data-transition="pop">Boîte de dialogue</a>
```

```
<div data-role="dialog" id="dial1">
  <div data-role="header">
    <h1>Fenêtre 2</h1>
  </div>
  <div data-role="content">
    <p>Contenu de la fenêtre 2</p>
  </div>
</div>
```

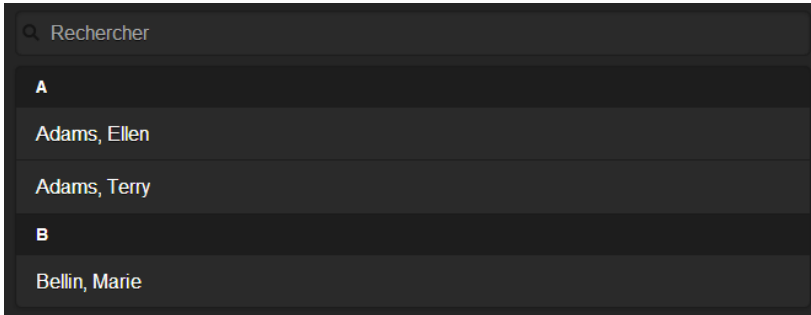
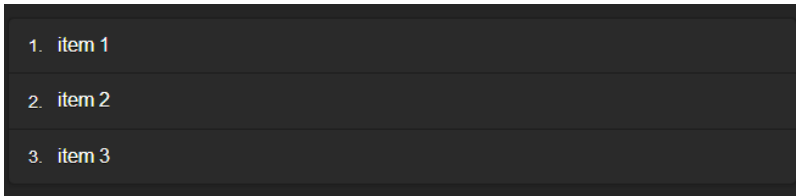
Options

```
closeBtn
closeBtnText
corners
defaults
disabled
initSelector
overlayTheme
```

e. Panel

```
<div data-role="panel" id="pushPanel" data-display="push"/>
```

f. ListView



```

<ol data-role="listview" data-inset="true">
  <li>item 1</li>
  <li>item 2</li>
  <li>item 3</li>
</ol>
<!-- avec séparateurs et liens-->
<ol data-role="listview">
  <li data-role="list-divider">Group A</li>
  <li>item 1</li>
  <li data-role="list-divider">Group B</li>
  <li><a href="#">Go to..</a></li>
</ol>

<!-- champ de recherche et séparation alphabétique -->
<ul data-role="listview" data-filter="true" data-filter-placeholder="Rechercher"
data-autodividers="true" data-inset="true">
  <li> Adams, Ellen</li>
  <li> Adams, Terry</li>
  <li>Bellin, Marie</li>
</ul>

```

Data-*

Pour ListView

data-autodividers	true false	Division automatique
data-count-theme	letter	Thème du compteur
data-divider-theme	letter	Thème division
data-filter	true false	Filtre pour « recherche »
data-filter-placeholder	sometext	Texte affiché dans champ de recherche
data-filter-theme	letter	Thème filtre
data-icon	Icône	
data-inset	true false	Bords arrondis + marges
data-split-icon	Icône du split button	
data-split-theme	letter	Thème split
data-theme	letter	Thème

Pour list items

data-filtertext	sometext	Texte de recherche pour le filtre
data-icon		Icône de l'item
data-role	list-divider	division
data-theme	letter	Thème list item

Options

autodividers	filterReveal
autodividersSelector	filterTheme
countTheme	hideDividers
defaults	icon
disabled	initSelector
dividerTheme	inset
filter	splitIcon
filterCallback	splitTheme
filterPlaceholder	theme

b. Table

header 1	header 2
1	item 1
2	item 2

```

<table class="ui-responsive table-stroke" data-role="table" data-
mode="reflow">
  <thead>
    <tr>
      <th data-priority="1">header 1</th>
      <th data-priority="persist">header 2</th>
    </tr>
  </thead>
  <tbody>
    <tr>
      <th>1</th>
      <td>item 1</td>
    </tr>
    <tr>
      <th>2</th>
      <td>item 2</td>
    </tr>
  </tbody>
</table>

```

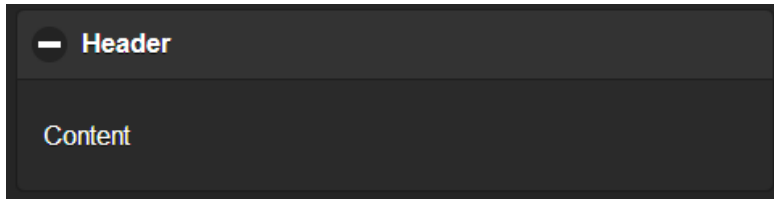
Options

```

classes.table
defaults
disabled
initSelector

```

h. Collapsible



```
<div data-role="collapsible" data-collapsed="false">
  <h2>Header</h2>
  <p>Content</p>
</div>
```

Icons

```
<div data-role="collapsible" data-collapsed="false" data-expanded-icon="carat-u"
data-collapsed-icon="carat-d">
  <h4>Header</h4>
  <p>Content</p>
</div>
```

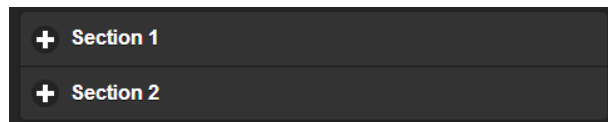
Data-*

data-collapsed	true false	Fermé ou non
data-collapsed-cue-text	<i>sometext</i>	Texte audible (Default "click to collapse contents")
data-collapsed-icon		Icône quand fermé
data-content-theme	<i>letter</i>	Thème content
data-expanded-cue-text	<i>sometext</i>	Texte audible
data-expanded-icon		Icône quand ouvert
data-iconpos	left right top bottom	Position icône
data-inset	true false	Bords arrondis
data-mini	true false	Taille mini
data-theme	<i>letter</i>	thème

Options

collapseCueText	expandCueText
collapsed	expandedIcon
collapsedIcon	heading
corners	iconpos
defaults	initSelector
disabled	inset
enhanced	mini
	theme

Collapsibleset



```
<div data-role="collapsibleset">
  <div data-role="collapsible">
    <h3>Section 1</h3>
    <p>Content section 1</p>
  </div>
  <div data-role="collapsible">
    <h3>Section 2</h3>
    <p>Content section 2</p>
  </div>
</div>
```

Data-*

data-collapsed-icon		Icône quand fermé
data-content-theme	<i>letter</i>	Thème content
data-expanded-icon		Icône quand ouvert
data-iconpos	left right top bottom notext	Position icône
data-inset	true false	Bords arrondis+ marge
data-mini	true false	Taille mini
data-theme	<i>letter</i> (a-z)	thème

Options

```
collapsedIcon
corners
defaults
disabled
enhanced
expandedIcon
iconpos
initSelector
inset
mini
```


i. Grid

item a	item b	item c
item 2a	item 2b	item 2c
Ok	Annuler	

ui-grid-solo	1	100%
ui-grid-a	2	50% / 50%
ui-grid-b	3	33% / 33% / 33%
ui-grid-c	4	25% / 25% / 25% / 25%
ui-grid-d	5	20% / 20% / 20% / 20% / 20%

<http://api.jquerymobile.com/grid-layout/>

<http://api.jquerymobile.com/responsive-grid/>

```
<div class="ui-grid-b ui-bar-a">
  <div class="ui-block-a">item a</div>
  <div class="ui-block-b">item b</div>
  <div class="ui-block-c">item c</div>
</div>
<div class="ui-grid-b ui-bar-b">
  <div class="ui-block-a">item 2a</div>
  <div class="ui-block-b">item 2b</div>
  <div class="ui-block-c">item 2c</div>
</div>
<div class="ui-grid-a" data-role="controlgroup" data-type="horizontal">
  <div class="ui-block-a"><a href="#" data-role="button">Ok</a></div>
  <div class="ui-block-b"><a href="#" data-
role="button">Annuler</a></div>
</div>
```

j. Button

```
<!-- bouton occupant toute la ligne -->
<a href="#" data-role="button">Button</a>

<!-- boutons côte à cote sur une ligne -->
<a href="#" data-role="button" data-inline="true">Button 1</a>
<a href="#" data-role="button" data-inline="true">Button 2</a>
<a href="#" data-role="button" data-inline="true">Button 2</a>

<!-- boutons groupés -->
<div data-role="controlgroup" data-type="horizontal">
  <a href="#" data-role="button">Button 1</a>
  <a href="#" data-role="button">Button 2</a>
</div>

<!-- effets (ombre ,coin carré) -->
<a href="#" data-role="button" data-shadow="true" data-
corners="false">Button 1</a>

<!-- icônes et positionnement -->
<a href="#" data-role="button" data-icon="gear" data-
iconpos="left">Options</a>
```

Data-*

data-corners	true false	Angles arrondis ou non
data-icon		Icône du bouton
data-iconpos	left right top bottom notext	Position de l'icône
data-iconshadow	true false	Ombre pour l'icône
data-inline	true false	Occupe toute la ligne ou non (true)
data-mini	true false	Bouton petite taille ou non

data-shadow **true** | false Ombre pour le bouton
 data-theme lettre Thème du bouton

Data- pour liens*

data-ajax **true** | false Chargement ajax
 data-direction reverse Reverse transition animation

data-dom-cache true | **false**
 data-prefetch true | **false**
 data-rel back | dialog | external | popup
 data-transition **fade** | flip | flow | pop | slide | slidedown | slidefade | slideup | turn | none
 data-position-to **origin** | jQuery selector | window

Css

ui-btn
 ui-btn-inline
 ui-btn-icon-top
 ui-btn-icon-right
 ui-btn-icon-bottom
 ui-btn-icon-left
 ui-btn-icon-notext
 ui-btn-a|b

Options

corners
 disabled
 enhanced
 icon
 iconpos
 iconshadow
 initSelector
 inline
 mini
 shadow
 theme
 wrapperClass

k. Bar

```
<h3 class="ui-bar ui-bar-a ui-corner-all">Bar demo</h3>
```

I.Form

<http://api.jquerymobile.com/category/widgets/>

```

<form>
  <fieldset data-role="controlgroup">
    <label>
      <input name="radio-choice-0" id="radio-choice-0a" type="radio">One
    </label>
    <label for="radio-choice-0b">Two</label>
    <input name="radio-choice-0" class="custom" id="radio-choice-0b"
type="radio">
  </fieldset>

  <fieldset data-role="controlgroup">
    <legend>Vertical:</legend>
    <input name="checkbox-v-2a" id="checkbox-v-2a" type="checkbox">
    <label for="checkbox-v-2a">One</label>
    <input name="checkbox-v-2b" id="checkbox-v-2b" type="checkbox">
    <label for="checkbox-v-2b">Two</label>
  </fieldset>
  <select name="select-native-1" id="select-native-1">
    <option value="1">The 1st Option</option>
    <option value="2">The 2nd Option</option>
    <option value="3">The 3rd Option</option>
    <option value="4">The 4th Option</option>
  </select>
  <input name="flip-checkbox-1" id="flip-checkbox-1" type="checkbox" data-
role="flipswitch">
  <input name="slider-1" id="slider-1" type="range" min="0" max="100"
value="50">
  <input type="submit" value="Ok">
  <div data-role="controlgroup">
    <a class="ui-btn ui-corner-all" href="#">No icon</a>
    <a class="ui-btn ui-corner-all ui-icon-delete ui-btn-icon-left"
href="#">Left</a>
  </div>

</form>

```

m. Icons

<http://demos.jquerymobile.com/1.4.3/icons/#ui-page-top>

attribut > data-icon
classe > ui-icon-* (exemple ui-icon-alert)



4. Page events and orientations

```

pagebeforeload
pageload
pageloadfailed
pagebeforechange
pagechange
pagechangefailed
pagebeforeshow
pagebeforehide
pageshow
pagehide
pagecreate
pageinit

```

Configure defaults

<http://api.jquerymobile.com/global-config/>

```

$(document).on("pagecreate", function (event) {
    $(window).on("orientationchange", function (event) {
        alert("Orientation changed to: " + event.orientation);
    });
});

```

Le bloc de code gérant l'événement mobileinit doit obligatoirement être placé avant l'insertion du code JavaScript de jQuery Mobile

```

<head>
  <meta name=viewport content="user-scalable=no,width=device-width" />
  <link rel=stylesheet href=jquery.mobile/jquery.mobile.css />
  <script src=jquery.js></script>
  <script>
    $(document).on("mobileinit", function () {
      $.mobile.ajaxEnabled = true;
      $.mobile.defaultPageTransition = "flip";
    });
  </script>
  <script src=jquery.mobile/jquery.mobile.js></script>
</head>

```

Tap/Taphold,Swipe/Swipeleft/Swiperight events

http://www.w3schools.com/jquerymobile/jquerymobile_events_touch.asp

```

$(document).bind("swiperight", function () {
    $.mobile.changePage('@Url.Action("Index", "Store")');
});

```

Événements virtuels

vclick,vmouseover,etc.

```

var $content = $("#pg1 div:jqmData(role=content)"); // le contenu de la page

$("#pg1").bind("vclick", function (event) { // event click dans la page
    $content.append(event.type); // ajoute au contenu de la page le text "vclick"
});

```