

Recherche – Angular.JS

Rapport de stage

Version

0.3

Client

C2T3

Superviseurs

Simon Boucher

Steven Goulet

Auteurs

Jonathan Lafrenière

Table des matières

[1. Introduction 3](#_Toc476836998)

[1.1. Définitions, acronymes et abréviations 3](#_Toc476836999)

[1.2. Journal d’avancement du projet 3](#_Toc476837000)

[2. Étude de la méthodologie de développement d’une application Web mobile à l’aide de l’IDE spécifié 4](#_Toc476837001)

[2.1. Relation entre Ionic, Cordova et Angular 4](#_Toc476837002)

[2.2. Qu’est-ce qu’Angular.JS? 4](#_Toc476837003)

[2.3. Installation 4](#_Toc476837004)

[2.4. Principes de base 5](#_Toc476837005)

[2.4.1. Introduction à Angular.JS 5](#_Toc476837006)

[2.5. Fonctionnalités avancées d’Angular.JS 8](#_Toc476837007)

[2.5.1. Manipulation et affichage du modèle 8](#_Toc476837008)

[2.5.2. Directive ng-repeat et répétition de structure 9](#_Toc476837009)

[2.5.3. Autres fonctionnalités intéressantes 10](#_Toc476837010)

[2.5.4. Synergie avec AJAX 11](#_Toc476837011)

[2.5.5. Conclusion 11](#_Toc476837012)

Historique du document

| Date | Auteur | Version | Sujet de la modification |
| --- | --- | --- | --- |
| 2017-02-10 | Jonathan Lafrenière | 0.0 | Première remise |
| 2017-02-17 | Jonathan Lafrenière | 0.1 | Deuxième Remise |
| 2017-03-09 | Simon Boucher | 0.2 | Révision |
| 2017-03-15 | Jonathan Lafrenière | 0.3 | Révision |

# Introduction

## Définitions, acronymes et abréviations

| Termes | Définitions |
| --- | --- |
| Angular.JS | Framework Javascript MVVM/MVW |
| AJAX | Asynchronous Javascript And XML, technique de développement permettant d’interroger un serveur à l’aide de requêtes envoyées à l’aide de JavaScript. Les données récupérées peuvent servir, par exemple, à actualiser les données affichées sur une page Web sans avoir à rafraîchir la page au complet. |
| Cordova | Framework permettant de compiler des sites webs en applications natives pouvant accéder aux différents périphériques d’un appareil mobile. |
| Directives | Fonctionnalité d’Angular permettant de créer des balises, classes et attributs à ajouter au DOM d’un document HTML. |
| DOM | Document Object Model, Structure d’un document XML |
| Ionic | Framework Basé sur Angular.JS permettant de définir la structure d’une application mobile de manière à ce qu’elle soit facilement transposable en application Native à l’aide de Cordova. |
| MVC | Modèle Vue Contrôleur – Patron de conception d’application |
| MVVM | Modèle Vue – Vue Modèle, Patron de conception d’application |
| MVW | Modèle Vue Whatever |

Tableau . : Abréviations

## Journal d’avancement du projet

Résumé ce qui a été réalisé durant la semaine. Par exemple : installation de Visual Studio 2015; étude du framework ASP.NET; programmation de l’objet de communication avec la BD; etc.

| Date | Résumé de l’avancement hebdomadaire |
| --- | --- |
| 2017-02-01 | Recherche sur les rôles respectifs d’Ionic, Cordova et Angular.JS  Création de démos |
| 2017-02-08 | Rédaction de la documentation |
| 2017-02-15 | Expérimentation du Framework |
| 2017-02-22 | Début du développement d’un démonstrateur et rédaction de la documentation |
| 2017-03-01 | Développement du démonstrateur |
| 2017-03-08 | Polissage du démonstrateur, commentaires, indentation |

Tableau . : Journal d’avancement hebdomadaire

# Étude de la méthodologie de développement d’une application Web mobile à l’aide de l’IDE spécifié

## Relation entre Ionic, Cordova et Angular

Ionic est un Framework HTML5 composé de plusieurs classes CSS, de librairies et de modules JavaScript s’appuyant sur le Framework Angular.js. Le développement d’applications à l’aide de ces trois technologies se qualifierait plutôt de « développement Ionic » que de développement Angular ou Cordova, car ces deux technologies sont des composants nécessaires au fonctionnement correct d’Ionic.

## Qu’est-ce qu’Angular.JS?

Angular est un Framework JavaScript développé par Google semblable au Framework React utilisé par Facebook. Il permet de réduire la quantité de code nécessaire à l’affichage d’éléments répétitifs en permettant au programmeur

* D’organiser son code en modules et contrôleurs
* D’injecter des variables à l’intérieur de fichiers HTML
* De décrire une structure (dans le fichier HTML) qui sera répétée automatiquement pour tous les éléments d’une collection
* De créer des attributs pour balises HTML (appelés « directives »)
* De lier des données de la vue (Fichier HTML) au modèle (Application JavaScript)

## Installation

Angular.js s’installe comme n’importe quelle autre librairie JavaScript, à l’aide de CDN ou en hébergeant le fichier localement. Un lien vers le fichier minimal est disponible au bas de cette page[[1]](#footnote-1). La version étudiée dans ce document sera la version 1.6.1, qui n’est pas la plus récente, mais la plus utilisée actuellement. Cette version sera supportée à l’avenir par Ionic.

## Principes de base

### Introduction à Angular.JS



Figure : Application Angular.JS Minimale

Une application Angular.js est composée de contrôleurs, qui eux-mêmes sont composés de variables et de fonctions dont la portée est limitée. Les contrôleurs permettent, à l’instar du modèle MVC, de limiter les interactions de l’utilisateur avec le modèle. Toutes les variables et fonction du contrôleur sont déclarées en tant qu’attributs de la variable $scope.

Pour accéder à ces variables dans un fichier HTML, il faudra spécifier le nom et le contrôleur utilisé à l’aide de directives.



Figure : Syntaxe correctes des directives permettant d'assigner l'application et le contrôleur au document entier

En déclarant l’application et le contrôleur utilisé dans la balise HTML, il sera possible d’accéder aux différentes variables du « Scope » partout dans le fichier. Il est possible de nicher les contrôleurs les uns aux travers des autres. Ainsi, différentes parties du document auront accès à différentes informations.

Figure : Utilisation de plusieurs contrôleurs

Une fois Angular importé, l’application Angular importée et les applications et contrôleurs affectés au document, il sera possible d’accéder aux variables et fonctions directement depuis le document HTML.

Figure : Contenu du fichier app.js



Figure : Contenu du fichier HTML

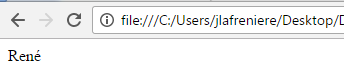


Figure : Résultat

Les doubles accolades à l’intérieur du fichier HTML permettent d’afficher une variable ou d’appeler une fonction Angular à l’intérieur de balises où une application et un contrôleur ont été définis.



Figure 7: Affichage de variables et de fonctions du modèle

Lien JsFiddle disponible au bas[[2]](#footnote-2)

## Fonctionnalités avancées d’Angular.JS

### Manipulation et affichage du modèle

Un des gros avantages d’Angular est la synchronisation instantanée de la vue et du modèle de l’application JavaScript. Une modification d’une variable liée dans un champ texte modifiera donc en temps réel (instantanément au cours de la saisie) la variable associée dans l’application. Inversement, la modification de variables dans l’application sera reflétée instantanément à tous les endroits dans le document où elle est appelée.



Figure 8: Utilisation de la directive ng-model

Lien JsFiddle disponible au bas[[3]](#footnote-3)

Deux nouvelles directives sont introduites dans cet extrait de code : **ng-model**, qui permet d’indiquer à un champ l’attribut du Scope auquel il est lié. Ainsi, modifier le texte modifiera instantanément la valeur de l’attribut « nom » du contrôleur « ctrl ». La directive **ng-click** permet, quant à elle, d’appeler des fonctions du contrôleur.

### Directive ng-repeat et répétition de structures

Cette directive permet de définir une structure qui sera copiée pour chaque élément d’une « array ».



Figure 9: Utilisation de la directive ng-repeat

Lien JsFiddle disponible au bas[[4]](#footnote-4)

Cette fonctionnalité est une des fonctionnalités les plus importantes d’Angular. Il est possible d’utiliser des « arrays » basiques ou JSON (comme dans cet exemple)

### Autres fonctionnalités intéressantes

Il est possible de déclarer un menu déroulant ainsi que les options disponibles de la manière suivante

<select ng-model="selectedName" ng-options="x for x in names">

Il est possible de modifier le modèle à l’aide de boutons radios de la manière suivante

<input type="radio" ng-model="sexe" value="Homme">Homme  
<input type="radio" ng-model="sexe" value="Femme">Femme

Il est possible d’afficher ou de dissimuler facilement un élément à l’aide de la directive **Ng-Show**, la valeur de cette directive doit être un booléen (**True/False**) et peut être le résultat d’une fonction du contrôleur.

<!—Affiche le nom de l’élève si l’élève est inscrit-->

<p ng-show="eleve.estInscrit">{{eleve.nom}}, {{eleve.prenom}}</p>

<p ng-show="false">Invisible</p>

De la même manière, il est possible d’activer ou de désactiver un champ.

<button ng-disabled="false">Enregistrer</button>

Une autre des grandes forces d’Angular est sa capacité à répondre instantanément à un évènement, tel que la modification d’un champ. Il est possible d’ajouter une directive correspondant à un évènement à n’importe quelle balise. Les fonctions et variables appelées à l’intérieur de ces directives doivent être comprises dans le contrôleur. Les évènements pris en charge par Angular sont :

|  |  |  |
| --- | --- | --- |
| Ng-blur | Ng-change | Ng-click |
| Ng-copy | Ng-cut | Ng-dblclick |
| Ng-focus | Ng-keydown | Ng-keyup |
| Ng-Mousedown | Ng-mouseenter | Ng-mousemove |
| Ng-mouseover | Ng-mouseup | Ng-paste |

Exemple : <input type="text" ng-focus="afficherMessage()">

### Synergie avec AJAX

La philosophie derrière le Framework étant de synchroniser le modèle et la vue d’une application, l’utilisation d’AJAX pour tenir les données affichées sur la page à jour est grandement facilitée. En envoyant périodiquement des requêtes au serveur, il est très simple de modifier les variables concernées du modèle pour concevoir une application en temps réel.

### Conclusion

Dans le but de couvrir les autres sujets qui m’ont été assignés pour ce stage, ce guide est intentionnellement court et l’objectif n’est pas d’explorer en profondeur Angular.JS, mais plutôt de comprendre les notions de bases relatives à son fonctionnement.

Si l’utilisation de ce Framework vous intéresse, je vous recommande fortement le tutoriel disponible au bas de cette page[[5]](#footnote-5). D’autres fonctionnalités importantes d’Angular.JS sont

* Les services, qui agissent en tant que fonctionnalités additionnelles aux applications Angular
* Les directives, qui permettent de créer des balises et des attributs utilisables dans un fichier HTML (Le Framework Ionic repose sur cette fonctionnalité)
* Les états des entrées (Input), Angular permet au programmeur de déterminé si le champ a été touché ou modifié, si la valeur entrée est valide selon une fonction de validation et bien plus.

1. https://ajax.googleapis.com/ajax/libs/angularjs/1.6.1/angular.min.js [↑](#footnote-ref-1)
2. https://jsfiddle.net/JLafreniere/2ng13ges/4/ [↑](#footnote-ref-2)
3. https://jsfiddle.net/JLafreniere/yj5d16rg/2/ [↑](#footnote-ref-3)
4. https://jsfiddle.net/JLafreniere/3emapoqt/2/ [↑](#footnote-ref-4)
5. <https://www.w3schools.com/angular/> [↑](#footnote-ref-5)