

PROGRAMMATION MULTIMEDIA (S4)

ANGULARJS (partie 1)



IUT Laval

Le Mans
Université

Aous Karoui

aous.karoui@univ-lemans.fr

Organisation



- Définition
- Historique
- Avantages
- Installation
- Les controleurs
- Les directives
- Exercices

ANGULARJS- Définition



- AngularJS est fondé sur l'utilisation de la [programmation déclarative](#) pour les interfaces utilisateurs et la [programmation impérative](#) pour la logique métier.
- Permet de découpler les manipulations du DOM de la logique métier pour une meilleure testabilité du code.
- Permet de découpler les cotés client et serveur pour une meilleure réutilisabilité

ANGULARJS - Historique



- Créé en 2009 par Misko Hervey et Adam Abronsw pour manipuler des données JSON
- Le projet a été basculé ensuite en opensource et repris par Google
- Première version mise sur GitHub le 20 octobre 2010. La version 1.0 en juillet 2012.
- Syntaxe complexe, limites du javascript
- Une V2 sortie en 2016
 - Le typescript remplaçant le javascript
 - Abandon de jQuery
- Aujourd'hui nous en sommes à Angular 5

ANGULARJS - Avantages



- Utilise le [TypeScript](#)
- Un sur-ensemble de javascript
- Permet le typage strict sur les variables
- Intègre les classes et les interfaces pour un code plus modulaire et plus robuste

- Il est utilisé par des Frameworks célèbres pour le développement mobile multi-plateformes comme Ionic

- Géré par Google
- Affiche une stabilité liée à la réputation de l'équipe de développement

ANGULARJS — Installation (méthode 1)



- Repose sur le CLI ou « *Command Line Interface* »
- Un outil permettant d'exécuter des commandes depuis la console

- Pré-requis:
 - Intallation de NodeJS
 - <https://nodejs.org/en/download/>
- Une fois NodeJS installé, installer le package npm via la commande suivante
- `npm install npm@latest -g`
- Si vous avez des erreurs de connexion, configurer le proxy avec
- `npm config set http-proxy http://votreid:mdp@proxy.univ-lemans.fr:3128`
- Installer le CLI avec la commande suivante
- `npm install -g @angular/cli`

ANGULARJS — Installation (méthode 2)

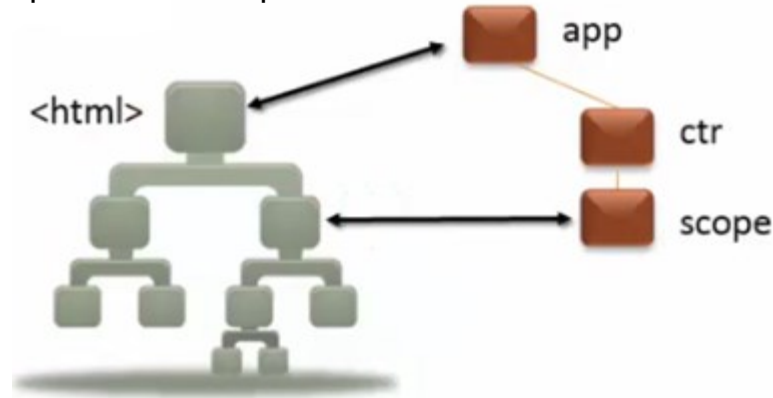


- Inclure directement dans votre page HTML le lien suivant
- `<script src="https://ajax.googleapis.com/ajax/libs/angularjs/1.6.4/angular.min.js"></script>`
- Créer un fichier "monAngular.js" et inclure son chemin également dans le html
- Modifier votre balise html de cette façon
- `<html ng-app="monApp">`

ANGULARJS – Les controleurs



- Élément de base de la couche traitement de AngularJS
- Un controleur est un objet JS qui sera associé à un élément de la View (dans le DOM)
- En fait, c'est le **\$scope** du controleur (objet JS qui contient des données et des traitements) qui est lié à la View
- Les templates de la View pourront exploiter ce scope



ANGULARJS – Les directives (ng-app et ng-controller)



- Ng-app :
- ng-app lie la vue (le DOM) avec l'application Angular (l'App)
- ng-app est un attribut dont la valeur doit correspondre au nom du module principale de l'App

- Ng-controller :
- Ng-controller lie l'élément HTML à un controleur
- Ng-controller est un attribut dont la valeur doit correspondre au nom du controleur

ANGULARJS — Mise en pratique



- Maintenant vous allez créer votre première page Angular
- Dans le fichier "monAngular.js", créer un premier module en vous renseignant sur le lien suivant
- <https://docs.angularjs.org/guide/module>
- Ensuite, coder votre premier contrôleur en précisant sa factory (fonction scope)
- <https://docs.angularjs.org/guide/controller>
- Maintenant, liez les éléments créés à une div de votre page HTML via la directive "ng-controller"
- Ajouter un champ texte (input), lié à votre contrôleur, pour observer les changements

ANGULARJS — ng-bind, ng-model et ng-click



- Ng-bind :
- Attribut sur une balise HTML qui contient du texte (span, ...)
- Remplace le texte de la balise par la propriété du scope nommée par l'attribut

- Ng-model :
- Attribut sur une balise HTML de type input, select ou textarea
- Lie la valeur de la balise avec la propriété du scope nommée par l'attribut

- Ng-click :
- Attribut sur une balise HTML cliquable
- Exécute l'expression spécifiée par l'attribut lorsqu'on clique sur la balise

ANGULARJS — Exercice



- Maintenant, créer un nouveau contrôleur associé au champ texte que vous avez déjà dans votre page HTML:
- Afin qu'il prenne en compte un nombre
- Afin qu'un nombre puisse de décrémenter ou s'incrémenter en appuyant sur un bouton
- Utilisez les directives que vous venez de voir pour faire le lien entre votre factory et votre div HTML.
- Utilisez deux méthodes différentes pour l'incrémentation et la décrémentation

ANGULARJS — ng-show, ng-hide



- Manipulation du DOM : permettent de masquer ou afficher un élément
- Attributs à mettre sur les balises HTML qui seront masquées ou affichées en fonction de conditions
- La valeur de l'attribut correspond à la condition (soit une expression, soit un boolean du scope)
- La balise HTML (et ses fils) sont masqués ou affichés (exploitation de css)

ANGULARJS — ng-if, ng-include, ng-repeat, ng-switch



- Créer ou supprimer des éléments du DOM
- Attributs à mettre sur les balises HTML
- La valeur de l'attribut correspond à une expression à évaluer pour créer/supprimer un élément
- Lors de la création, un nouveau contrôleur est instancié (lié via la chaîne de prototypage)

ANGULARJS — Exercice



- Maintenant, créer un nouveau contrôleur pour afficher et aussi masquer une suite de chaîne de caractères sur votre page HTML
- Indication:
- En plus des directives déjà utilisées essayer les directives "ng-repeat", "ng-show" et "ng-bind"

ANGULARJS — FORM



- La balise FORM est une directive ANGULAR !
- Elle réalise des controles avant la soumission et exploite des styles CSS en fonction des saisies
- L'attribut **novalidate** du FORM désactive la validation classique du navigateur
- L'attribut **name** donne un nom au FORM et la rend ainsi accessible dans le *\$scope*
- Les inputs d'un FORM sont validés en fonction de leur type (input type="...")

ANGULARJS — FORM et CSS



- ANGULAR ajoute des classes CSS en fonction de la validation
- *ng-valid* : le modèle est valide
- *ng-invalid* : le modèle est non valide
- *ng-pristine* : le contrôleur n'est pas encore été sollicité
- *ng-dirty* : le contrôleur a été sollicité
- *ng-touched* : le contrôleur a été brouillé
- *ng-untouched* : le contrôleur n'a pas été brouillé

ANGULARJS — Exercice



- Créer un petit formulaire qui, change le background du champ texte au moment de la saisie et affiche les éléments saisis au niveau de la console.
- Indications:
- Vous pouvez utiliser les classes CSS vues pour modifier le comportement quand on écrit dans un champ
- Il faudra faire les liaisons entre le \$scope du controleur et les éléments du formulaire via l'attribut *name*

ANGULARJS — Les Factories



- Les Factories sont des fonctions javascript que l'on peut utiliser n'importe où dans l'application angularJS
- La création d'une **factory** se fait généralement dans un fichier javascript externe que l'on ajoute dans le dossier du projet
- (n'oubliez pas d'ajouter l'appel du fichier javascript dans le fichier index.html)

```
1. angular.module('myApp').factory('nomDeLaFactory', function(){  
2. });
```

- Après avoir créé la **factory**, il faut ajouter son nom dans les **controllers** qui l'utiliseront.

```
1. angular.module('myApp').controller('nomDuController', ['nomDeLaFactory', function(nomDeLaFactory){  
2. }]);
```

- La **factory** permet de retourner un objet qui regroupe plusieurs attributs

ANGULARJS — Exercice



- Créer un tableau d'objets « students » qui contient pour chaque étudiant, son nom et le sujet de son stage
- Afficher votre tableau en utilisant le controleur
- Maintenant afficher votre tableau en utilisant la factory
- Ensuite, via votre formulaire, la page actualisera cette liste en fonction de ce que vous insérez