Building Web Applications with Angular.js

Opening Discussion:

* Who is writing javascript?
* What frameworks are you using?
* Whats the difference between a framework and a library?
* Why would you want to use a front end JavaScript Framework?
* What features do you need in a JavaScript Framework?

Must Have Features of a Front End Framework

* DataBinding
* Routing
* Dependency Injection
* Promises for Async
* Templating
* Http Abstractons
* Testability

Hello World Demo

* A basic MVC application(**helloStart**)
  + jquery
  + underscore
  + bootstrap
  + angular
* Html Only
  + ng-app
  + firstName
  + lastName
  + concatenate
  + ng-show – hide the message
  + ng-cloak
    - flash of unstyled content
    - css
* helloApp Module (**helloModule**)
  + firstName
  + lastName
  + fullName
  + showMessage
* messagesModel (**helloMessages**)
  + factory
    - randomMessage
    - Hello, Hi, Howdy
  + add message to scope
  + add a button to get a new message
* DI syntax
* (**helloFinal**)
* Key Concepts to Review
  + ng-app
  + ng-model
  + ng-show
  + ng-click
  + module
    - define module
    - reference modules
  + controller
  + $scope
  + factory
    - singleton services

Angular

Components

* Modules
  + ApplicationModule
  + OtherModules
    - ngRoute
* Directives
  + builtin
    - ng-app
    - ng-model
    - ng-click
    - ng-if
    - ng-repeat
    - ng-class
  + custom
* Controllers
  + $scope
* Services
  + Reusable components
  + Injected using DI
  + Singletons
  + Many Types
    - Factories
      * Make Injectable Components
      * Singletons
      * revealing module pattern
  + Built IN
    - $location
  + Custom
* Configs

Concepts

* Data Binding
  + {{ }} double curly brace syntax
  + ng-bind – use to avoid flash of unstyled content
* Dependency Injection
* Promises
  + $q
    - promise api is much different that jQuery
* IIFE – Immediately Invoked Function Expressions
  + use to keep the functions and variables out of the global scope

routerDemo

1. Add a reference to angular-route.js

2. Inject the ngRoute module to the flashCards module

3. Add the ng-view directive to index.cshtml

4. Configure the router

(function () {

angular

.module('flashCards')

.config(routeConfig);

routeConfig.$inject = ['$routeProvider'];

function routeConfig($routeProvider) {

$routeProvider

.when('/', {

templateUrl: 'App/views/catalog.html',

controller: 'catalogController'

})

.when('/cards/:name/ids/:id', {

templateUrl: 'App/views/cards.html',

controller: 'cardsController'

})

.otherwise({

redirectTo: '/'

});

}

})();

catalogDemo

ng-repeat="catalogName in catalogNames"

ng-click="navigate(catalogName)"

**{{** catalogName **}}**

cardsDemo

ng-if="selected.found"

ng-if="!selected.found"

previousButton

ng-class="{'hidden': !selected.hasPrevious()}"

ng-click="previous()"

nextButton

ng-class="{'hidden': !selected.hasNext()}"

ng-click="next()"

status

**{{** selected.selectedOf() **}}**

**{{** selected.name **}}**

card

ng-click="flip()"

**{{** selected.card.frontHeading **}}**

**{{** selected.card.front **}}**

**{{** selected.card.backHeading **}}**

**{{** selected.card.back **}}**

References

Angular Documentation

<https://angularjs.org/>

ng-book

Ari Lerner

John Papa’s Style Guide

<https://github.com/johnpapa/angularjs-styleguide>