Cap.1

|  |
| --- |
| On the flip side, being able to work with Plain Old JavaScript Objects (POJOs) makes integrating with other technologies incredibly easy. By consuming and emitting JavaScript without having to wrap and unwrap it in proprietary framework mechanisms, you’re able to consume data from other sources much more efficiently. |
| AngularJS at a glance  Component    Purpose Module  Modules serve as containers to help you organize code within your AngularJS application.  Modules can contain sub-modules, making it easy to compose functionality as needed.  Config The config block of an AngularJS application allows for configuration to be applied before the application actually runs. This is useful for setting up routes, dynamically configuring services, and so on.  Routes Routes allow you to define ways to navigate to specific states within your application. They also allow you to define configuration options for each specific route, such as which template and controller to use.  Views The view in AngularJS is what exists after AngularJS has compiled and rendered the DOM with all of the JavaScript wiring in place.  $scope $scope is essentially the glue between the view and controller within an AngularJS application. With the introduction of the controller-as syntax, the need to explicitly use $scope has been greatly reduced.  Controller The controller is responsible for defining methods and properties that the view can bind to and interact with. As a matter of best practice, controllers should be lightweight and only focus on the view they’re controlling.  Directive A directive is an extension of a view in AngularJS in that it allows you to create custom, reusable elements that encapsulate behavior. You can think of directives as components or decorators for your HTML. Directives are used to extend views and to make these extensions available for use in more than one place.  Service Services provide common functionality to an AngularJS application. For instance, if you have data that more than one controller needs, you would promote that data to a service and then make it available to the controllers via the service. Services extend controllers and make them more globally accessible. |

|  |
| --- |
|  |
| Angello and the big picture |
|  |
| The module |

|  |
| --- |
| // app.js  var myModule = angular.module('Angello', []); |
| // app.js  var myModule = angular.module('Angello', []);  myModule.factory('AngelloHelper', function() { });  myModule.service('AngelloModel', function() { });  myModule.controller('MainCtrl', function() { });  myModule.directive('story', function() { }); |
| <!-- index.html -->  <html ng-app="Angello"> |
|  |
| Model-View-ViewModel |

|  |
| --- |
| <!--index.html-->  <div ng-controller="MainCtrl as main">  </div> |
| // app.js  myModule.controller('MainCtrl', function() {  var main = this;  //...  main.stories = [  {  title: 'First story',  description: 'Our first story.',  criteria: 'Criteria pending.',  status: 'To Do',  type: 'Feature',  reporter: 'Lukas Ruebbelke',  assignee: 'Brian Ford'  },  {  title: 'Second story',  description: 'Do something.',  criteria: 'Criteria pending.',  status: 'Back Log',  type: 'Feature',  reporter: 'Lukas Ruebbelke',  assignee: 'Brian Ford'  },  {  title: 'Another story',  description: 'Just one more.',  criteria: 'Criteria pending.',  status: 'Code Review',  type: 'Enhancement',  reporter: 'Lukas Ruebbelke',  assignee: 'Brian Ford'  }  ];  //...  }); |
| <!-- index.html -->  <div ng-controller="MainCtrl as main">  <div class="col-md-4">  <h2>Stories</h2>  <div class="callout"  ng-repeat="story in main.stories"  ng-click="main.setCurrentStory(story)">  <h4>{{story.title}}</h4>  <p>{{story.description}}</p>  </div>  </div>  </div> |
| // app.js  myModule.controller('MainCtrl', function() {  var main = this;  //...  main.createStory = function() {  main.stories.push({  title:'New Story',  description:'Description pending.',  criteria:'Criteria pending.',  status:'Back Log',  type:'Feature',  reporter:'Pending',  assignee:'Pending'  });  };  //...  }); |
|  |
| Services |

|  |
| --- |
| // app.js  myModule.service('AngelloModel', function() {  var service = this,  stories = [  {  title: 'First story',  description: 'Our first story.',  criteria: 'Criteria pending.',  status: 'To Do',  type: 'Feature',  reporter: 'Lukas Ruebbelke',  assignee: 'Brian Ford'  },  //...  ];  service.getStories = function() {  return stories;  };  }); |
| // app.js  myModule.controller('MainCtrl', function(AngelloModel) {  var main = this;  //...  main.stories = AngelloModel.getStories();  //...  }); |
| // app.js  myModule.factory('AngelloHelper', function() {  var buildIndex = function(source, property) {  var tempArray = [];  for(var i = 0, len = source.length; i < len; ++i) {  tempArray[source[i][property]] = source[i];  }  return tempArray;  };  return {  buildIndex: buildIndex  };  }); |
|  |
| Directives |

|  |
| --- |
| // app.js  myModule.directive('story', function(){  return {  scope: true,  replace: true,  template:'<div><h4>{{story.title}}</h4>  <p>{{story.description}}</p></div>'  }  }); |
| <div ng-controller="MainCtrl as main">  <div class="col-md-4">  <h2>Stories</h2>  <story class="callout"  ng-repeat="story in main.stories"  ng-click="main.setCurrentStory(story)">  </story>  <!-- ... -->  </div>  </div> |

Cap. 2

|  |
| --- |
| Introducing to the official sample application of the book—Angello ….. |
|  |
| The Angello file structure |
| <!-- client/index.html -->  <html ng-app="Angello">  <!-- HTML --> |
| // client/src/angello/Angello.js  var myModule = angular.module('Angello',  [  'ngRoute',  'ngAnimate',  'firebase',  'ngMessages',  'Angello.Common',  'Angello.Dashboard',  'Angello.Login',  'Angello.Storyboard',  'Angello.User',  'auth0',  'angular-jwt',  'angular-storage'  ]); |
| // client/src/angello/storyboard/Storyboard.js  angular.module('Angello.Storyboard', ['Angello.Common']); |
| // client/src/angello/storyboard/controllers/StoryboardController.js  angular.module('Angello.Storyboard')  .controller('StoryboardCtrl',  function () {  //...  }); |
| // client/src/angello/app/models/StoriesModel.js  angular.module('Angello.Common')  .service('StoriesModel',  function () {  //...  }); |
| // client/src/angello/Angello.js  myModule.config(function($routeProvider) {  $routeProvider  .when('/', {  templateUrl: 'src/angello/storyboard/tmpl/storyboard.html',  controller: 'StoryboardCtrl',  controllerAs: 'storyboard'  });  }); |
| <!-- client/index.html -->  <html ng-app="Angello">  <head></head>  <body ng-controller="MainCtrl as main">  <div class="navbar navbar-fixed-top navbar-default"></div>  <div ng-view=""></div>  <div class="modal"></div>  </body>  </html> |
| <!-- client/index.html -->  <div class="navbar navbar-fixed-top navbar-default">  <div class="navbar-header">  <a class="logo navbar-brand" href="#/">  <img src="assets/img/angello.png">  </a>  </div>  </div>  <!-- ... --> |
| // client/src/angello/Angello.js  myModule.config(function($routeProvider) {  $routeProvider  .when('/', {  templateUrl: 'src/angello/storyboard/tmpl/storyboard.html',  controller: 'StoryboardCtrl',  controllerAs: 'storyboard'  })  .otherwise({redirectTo: '/'});  }); |
| // client/src/angello/Angello.js  myModule.config(function($routeProvider) {  $routeProvider  .when('/', {  templateUrl: 'src/angello/storyboard/tmpl/storyboard.html',  controller: 'StoryboardCtrl',  controllerAs: 'storyboard'  })  .when('/dashboard', {  templateUrl: 'src/angello/dashboard/tmpl/dashboard.html',  controller: 'DashboardCtrl',  controllerAs: 'dashboard'  })  .when('/users', {  templateUrl: 'src/angello/user/tmpl/users.html',  controller: 'UsersCtrl',  controllerAs: 'users'  })  .when('/users/:userId', {  templateUrl: 'src/angello/user/tmpl/user.html',  controller: 'UserCtrl',  controllerAs: 'myUser'  })  .when('/login', {  templateUrl: 'src/angello/login/tmpl/login.html',  controller: 'LoginCtrl',  controllerAs: 'login'  })  .otherwise({redirectTo: '/'});  }); |
| <!-- client/index.html -->  <div class="navbar navbar-fixed-top navbar-default">  <div class="navbar-header">  <a class="logo navbar-brand" href="#/">  <img src="assets/img/angello.png">  </a>  </div>  <div class="btn-group pull-right" ng-show="main.currentUser">  <a class="btn btn-danger" href="#/">  <span class="glyphicon glyphicon-home"></span>  </a>  <a class="btn btn-danger" href="#/users">  <span class="glyphicon glyphicon-user"></span>  </a>  <a class="btn btn-danger" href="#/dashboard">  <span class="glyphicon glyphicon-signal"></span>  </a>  <!-- ... -->  </div>  </div> |
| Cap.3 |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
| // client/src/angello/storyboard/controllers/StoryboardController.js  angular.module('Angello.Storyboard')  .controller('StoryboardCtrl', function() {  var storyboard = this;  storyboard.stories = [  {  "assignee": "1",  "criteria": "It tests!",  "description": "This is a test",  "id": "1",  "reporter": "2",  "status": "To Do",  "title": "First Story",  "type": "Spike"  },  {  "assignee": "2",  "criteria": "It works!",  "description": "testing something",  "id": "2",  "reporter": "1",  "status": "In Progress",  "title": "Second Story",  "type": "Enhancement"  }  ];  storyboard.statuses = [  {name: 'To Do'},  {name: 'In Progress'},  {name: 'Code Review'},  {name: 'QA Review'},  {name: 'Verified'}  ];  }); |
|  |
| <!-- client/src/angello/storyboard/tmpl/storyboard.html -->  <div class="list-area">  <div class="list-wrapper">  <ul class="list"  ng-repeat="status in storyboard.statuses">  <h3 class="status">{{status.name}}</h3>  <hr/>  </ul>  </div>  </div> |
|  |
| <!-- client/src/angello/storyboard/tmpl/storyboard.html -->  <div class="list-area">  <div class="list-wrapper">  <ul class="list"  ng-repeat="status in storyboard.statuses">  <h3 class="status">{{status.name}}</h3>  <hr/>  <li class="story"  ng-repeat="story in storyboard.stories">  <article>  <div>  <button type="button" class="close">×</button>  <p class="title">{{story.title}}</p>  </div>  <div class="type-bar {{story.type}}"></div>  <div>  <p>{{story.description}}</p>  </div>  </article>  </li>  </ul>  </div>  </div> |
|  |
| // client/src/angello/storyboard/tmpl/storyboard.html  <ul class="list my-repeat-animation"  ng-repeat="status in ::storyboard.statuses">  <h3 class="status">{{::status.name}}</h3>  <!-- ... -->  </ul> |
|  |
|  |
|  |
| // Hypothetical Scenario  <ul>  <li ng-repeat="item in items"  ng-click="ctrl.currentIndex == $index" >  <h2>{{item.title}}</h2>  <p ng-if="ctrl.currentIndex == $index">  {{item.longDescription}}  </p>  </li>  </ul> |
|  |
| // Hypothetical Scenario  <ul>  <li ng-repeat="(key, item) in items">  <h2>{{key}}</h2>  <h3>{{item.title}}</h3>  </li>  </ul> |
|  |
| <!-- client/src/angello/storyboard/tmpl/storyboard.html -->  <div class="list-area">  <div class="list-wrapper">  <ul class="list"  ng-repeat="status in storyboard.statuses">  <h3 class="status">{{status.name}}</h3>  <hr/>  <li class="story"  ng-repeat="story in storyboard.stories  | filter: {status:status.name}">  <article>  <div>  <button type="button" class="close">×</button>  <p class="title">{{story.title}}</p>  </div>  <div class="type-bar {{story.type}}"></div>  <div>  <p>{{story.description}}</p>  </div>  </article>  </li>  </ul>  </div>  </div> |
|  |
|  |
|  |
|  |
|  |
| <!-- client/src/angello/storyboard/tmpl/storyboard.html -->  <li class="story"  ng-repeat="story in storyboard.stories  | filter: {status:status.name}"  ng-click="storyboard.setCurrentStory(story)">  <article>  <div>  <button type="button" class="close">×</button>  <p class="title">{{story.title}}</p>  </div>  <div class="type-bar {{story.type}}"></div>  <div>  <p>{{story.description}}</p>  </div>  </article>  </li> |
|  |
|  |
|  |
| <!-- client/src/angello/storyboard/tmpl/storyboard.html -->  <div class="details">  <h3>Card Details</h3>  <form name="storyboard.detailsForm">  <div class="form-group">  <div class="controls">  <label class="control-label" for="inputTitle">\*Title</label>  <input type="text" id="inputTitle" name="inputTitle"  placeholder="Title"  ng-model="storyboard.editedStory.title"  ng-required="true" ng-minlength="3" ng-maxlength="30"  class="form-control">  </div>  </div>  <!-- ... -->  </form>  </div> |
|  |
| <!-- client/src/angello/storyboard/tmpl/storyboard.html -->  <select id="inputReporter" name="inputReporter"  ng-model="storyboard.editedStory.reporter" ng-required="true"  ng-options="user.id as user.name for user in storyboard.users"  class="form-control">  <option value="">Please select...</option>  </select> |
|  |
| <!-- client/src/angello/storyboard/tmpl/storyboard.html -->  <div class="details">  <h3>Card Details</h3>  <form name="storyboard.detailsForm">  <!-- ... -->  </form>  <hr/>  <div ng-if="storyboard.currentStory">  <button class="btn btn-default"  ng-click="storyboard.updateCancel()">  Cancel  </button>  <button class="btn pull-right btn-default"  ng-disabled="!storyboard.detailsForm.$valid"  ng-click="storyboard.updateStory()">Update Story</button>  </div>  <!-- ... -->  </div> |
|  |
|  |
|  |
| <!-- client/src/angello/storyboard/tmpl/storyboard.html -->  <div class="details">  <h3>Card Details</h3>  <form name="storyboard.detailsForm">  <!-- ... -->  </form>  <hr/>  <!-- ... -->  <div ng-if="!storyboard.currentStory">  <button class="btn pull-right btn-default"  ng-disabled="!storyboard.detailsForm.$valid"  ng-click="storyboard.createStory()">  Create Story  </button>  </div>  </div> |
|  |
|  |
|  |
|  |
|  |
| <!-- client/src/angello/storyboard/tmpl/storyboard.html -->  <!-- although illustrating a slightly different implementation -->  <li class="story"  ng-repeat="story in storyboard.stories  | filter: {status:status.name}"  ng-click="storyboard.setCurrentStory(story)">  <article>  <div>  <button type="button" class="close"  ng-click="storyboard.deleteStory(story.id)">  ×</button>  <p class="title">{{story.title}}</p>  </div>  <div class="type-bar {{story.type}}"></div>  <div>  <p>{{story.description}}</p>  </div>  </article>  </li> |
|  |
|  |
|  |
| // client/tests/specs/controllers/StoryboardController.spec.js  it('should reset the form', function() {  ctrl.editedStory = ctrl.currentStory = {assignee: '1'};  ctrl.resetForm();  expect(ctrl.currentStory).toBeNull();  expect(ctrl.editedStory).toEqual({});  }); |
|  |
| // client/tests/specs/controllers/StoryboardController.spec.js  it('should delete a story', function() {  var story = ctrl.stories[0];  ctrl.deleteStory(story.id);  expect(ctrl.stories).not.toContain(story);  }); |
|  |
| Cap.4 |
|  |
|  |
|  |
| // client/src/angello/Angello.js  myModule.value('STORY\_TYPES', [  {name: 'Feature'},  {name: 'Enhancement'},  {name: 'Bug'},  {name: 'Spike'}  ]); |
|  |
| // client/src/angello/storyboard/controllers/StoryboardController.js  angular.module('Angello.Storyboard')  .controller('StoryboardCtrl',  function (STORY\_TYPES) {  var myStory = this;  //...  myStory.types = STORY\_TYPES;  //...  }); |
|  |
|  |
|  |
| // client/src/angello/Angello.js  myModule.value('STORY\_TYPES', [  {name: 'Feature'},  {name: 'Enhancement'},  {name: 'Bug'},  {name: 'Spike'}  ]); |
|  |
| // client/src/angello/storyboard/controllers/StoryboardController.js  angular.module('Angello.Storyboard')  .controller('StoryboardCtrl',  function (STORY\_TYPES) {  var myStory = this;  //...  myStory.types = STORY\_TYPES;  //...  }); |
|  |
| // client/src/angello/app/services/EndpointConfigService.js  angular.module('Angello.Common')  .constant('CURRENT\_BACKEND', 'firebase'); |
|  |
| // client/src/angello/app/services/LoadingService.js  angular.module('Angello.Common')  .service('LoadingService',  function ($rootScope) {  var service = this;  service.setLoading = function (loading) {  $rootScope.loadingView = loading;  };  }); |
|  |
| angular.module('Angello.Common')  .factory('LoadingService',  function ($rootScope) {  var setLoading = function (loading) {  $rootScope.loadingView = loading;  };  return {  setLoading: setLoading  }  }); |
|  |
| // client/src/angello/app/models/StoriesModel.js  angular.module('Angello.Common')  .service('StoriesModel',  function ($http, EndpointConfigService, UtilsService) {  //...  }); |
|  |
| // client/src/angello/app/models/StoriesModel.js  angular.module('Angello.Common')  .service('StoriesModel',  function ($http, EndpointConfigService, UtilsService) {  var service = this,  MODEL = "/stories/";  //...  }); |
|  |
| // client/src/angello/app/models/StoriesModel.js  angular.module('Angello.Common')  .service('StoriesModel',  function ($http, EndpointConfigService, UtilsService) {  var service = this,  MODEL = "/stories/";  service.all = function () {  return $http.get(EndpointConfigService.getUrl(  MODEL + EndpointConfigService.getCurrentFormat()))  .then(  function(result) {  return UtilsService.objectToArray(result);  }  );  };  //...  }); |
|  |
| // client/src/angello/storyboard/controllers/StoryboardController.js  storyboard.getStories = function () {  StoriesModel.all()  .then(function (result) {  console.log(result.data);  });  }; |
|  |
| .then(function(result) { /\* Handle result here \*/ }); |
|  |
| // client/src/angello/app/services/EndpointConfigService.js  angular.module('Angello.Common')  //.constant('CURRENT\_BACKEND', 'node')  .constant('CURRENT\_BACKEND', 'firebase')  // client/src/angello/app/models/StoriesModel.js  angular.module('Angello.Common')  .service('StoriesModel',  function ($http, EndpointConfigService, UtilsService) {  var service = this,  MODEL = "/stories/";  //...  }); |
|  |
| // client/src/angello/app/models/StoriesModel.js  angular.module('Angello.Common')  .service('StoriesModel',  function ($http, EndpointConfigService, UtilsService) {  var service = this,  MODEL = "/stories/";  service.all = function () {  return $http.get(EndpointConfigService.getUrl(  MODEL + EndpointConfigService.getCurrentFormat()))  .then(  function(result) {  return UtilsService.objectToArray(result);  }  );  };  service.fetch = function (story\_id) {  return $http.get(  EndpointConfigService.getUrlForId(MODEL, story\_id)  );  };  service.create = function (story) {  return $http.post(EndpointConfigService.getUrl(  MODEL  + EndpointConfigService.getCurrentFormat()), story  );  };  service.update = function (story\_id, story) {  return $http.put(  EndpointConfigService.getUrlForId(MODEL, story\_id),  story  );  };  service.destroy = function (story\_id) {  return $http.delete(  EndpointConfigService.getUrlForId(MODEL, story\_id)  );  };  }); |
|  |
| // client/src/angello/app/models/StoriesModel.js  service.all = function () {  return $http.get(EndpointConfigService.getUrl(  MODEL + EndpointConfigService.getCurrentFormat()))  .then(  function(result) {  return UtilsService.objectToArray(result);  }  );  }; |
|  |
| // client/src/angello/storyboard/controllers/StoryboardController.js  storyboard.getStories = function () {  StoriesModel.all()  .then(function (result) {  storyboard.stories = (result !== 'null') ? result : {};  $log.debug('RESULT', result);  }, function (reason) {  $log.debug('REASON', reason);  });  }; |
|  |
| StoriesModel.all()  .then(); |
|  |
| StoriesModel.all()  .then(function (result) {  }, function (reason) {  }); |
|  |
| StoriesModel.all().then(function (result) {  storyboard.stories = (result !== 'null') ? result : {};  $log.debug('RESULT', result);  }, function (reason) {  $log.debug('REASON', reason);  }); |
|  |
| StoriesModel.update(storyboard.currentStoryId, storyboard.editedStory)  .then(function (result) {  storyboard.getStories();  storyboard.resetForm();  $log.debug('RESULT', result);  }, function (reason) {  $log.debug('REASON', reason);  }); |
|  |
| // Hypothetical Scenario  myPromise()  .then(function(result) {  // Success Callback  })  .catch(function(error) {  // Error Callback  })  .finally(function() {  // Gets executed no matter what  })  ; |
|  |
| $http.get(EndpointConfigService.getUrl(  MODEL + EndpointConfigService.getCurrentFormat())  )  .success(function(data, status, headers, config) {  // this callback will be called asynchronously  // when the response is available  })  .error(function(data, status, headers, config) {  // called asynchronously if an error occurs  // or server returns response with an error status.  }); |
|  |
|  |
|  |
| // client/src/angello/Angello.js  myModule.factory('loadingInterceptor', function (LoadingService) {  var loadingInterceptor = {  request: function (config) {  LoadingService.setLoading(true);  return config;  },  response: function (response) {  LoadingService.setLoading(false);  return response;  }  };  return loadingInterceptor;  }); |
|  |
| // client/src/angello/Angello.js  myModule.config(function ($routeProvider, $httpProvider) {  //...  // Interceptor  $httpProvider.interceptors.push('loadingInterceptor');  //...  }); |
|  |
|  |
|  |
|  |
|  |
| // client/src/angello/storyboard/controllers/StoryboardController.js  storyboard.getStories = function () {  StoriesModel.all()  .then(function (result) {  storyboard.stories = (result !== 'null') ? result : {};  $log.debug('RESULT', result);  }, function (reason) {  $log.debug('REASON', reason);  });  }; |
|  |
| // client/tests/specs/services/LoadingService.spec.js  describe('Loading Service', function () {  var $rootScope, LoadingService;  beforeEach(module('Angello.Common'));  //...  }); |
|  |
| describe('Loading Service', function () {  var $rootScope, LoadingService;  beforeEach(module('Angello.Common'));  beforeEach(inject(function (\_$rootScope\_, \_LoadingService\_) {  $rootScope = \_$rootScope\_;  LoadingService = \_LoadingService\_;  }));  //...  }); |
|  |
| // client/tests/specs/services/LoadingService.spec.js  describe('Loading Service', function () {  var $rootScope, LoadingService;  beforeEach(module('Angello.Common'));  beforeEach(inject(function (\_$rootScope\_, \_LoadingService\_) {  $rootScope = \_$rootScope\_;  LoadingService = \_LoadingService\_;  }));  it('should update $rootScope to false when setLoading is set to false',  function () {  LoadingService.setLoading(false);  expect($rootScope.loadingView).toEqual(false);  });  it('should update $rootScope to true when setLoading is set to true',  function () {  LoadingService.setLoading(true);  expect($rootScope.loadingView).toEqual(true);  });  }); |
|  |
| // client/tests/specs/services/StoriesModel.spec.js  describe('Stories Model', function () {  //...  afterEach(inject(function($httpBackend) {  $httpBackend.verifyNoOutstandingExpectation();  $httpBackend.verifyNoOutstandingRequest();  }));  //...  }); |
|  |
|  |
|  |
| $httpBackend.when(  'GET', 'https://angello-angularjs.firebaseio.com/clients/1/stories/.json'  ).respond(response);  var promise = StoriesModel.all(); |
|  |
| $httpBackend.when(  'POST', 'https://angello-angularjs.firebaseio.com/clients/1/stories/.json'  ).respond(response);  var promise = StoriesModel.create({}); |
|  |
| $httpBackend.when(  'GET', 'https://angello-angularjs.firebaseio.com/clients/1/stories/1.json'  ).respond(response);  var promise = StoriesModel.fetch(1); |
|  |
| $httpBackend.when(  'PUT', 'https://angello-angularjs.firebaseio.com/clients/1/stories/1.json'  ).respond(response);  var promise = StoriesModel.update(1, {}); |
|  |
| $httpBackend.when('DELETE',  'https://angello-angularjs.firebaseio.com/clients/1/stories/1.json'  ).respond(response);  var promise = StoriesModel.destroy(1); |
|  |
| Cap.5 |
|  |
|  |
|  |
|  |
|  |
| myModule.controller('MainCtrl', function ($scope) {  // ...  }); |
|  |
| // client/src/angello/user/directives/UserStoryDirective.js  angular.module('Angello.User')  .directive('userstory',  function () {  var linker = function (scope, element, attrs) {  // Pending  };  var controller = function ($scope) {  // Pending  };  return {  restrict: 'A',  controller: controller,  controllerAs: 'userStory',  link: linker  };  }); |
|  |
| <!-- client/src/angello/storyboard/tmpl/storyboard.html -->  <li ng-repeat="story in storyboard.stories | filter:{status:status.name}"  class="story"  ng-click="storyboard.setCurrentStory(story)">  <article>  <div>  <button type="button" class="close"  ng-click="userStory.deleteStoryBoard(story.id)">  ×  </button>  <p class="title">{{story.title}}</p>  </div>  <div class="type-bar {{story.type}}"></div>  <div>  <p>{{story.description}}</p>  </div>  </article>  </li> |
|  |
| $('#outer').mouseover(function() {  $('#log').append('<div>Handler for .mouseover() called.</div>');  }); |
|  |
| // client/src/angello/user/directives/UserStoryDirective.js  angular.module('Angello.User')  .directive('userstory',  function () {  var linker = function (scope, element, attrs) {  element  .mouseover(function () {  element.css({ 'opacity': 0.9 });  })  .mouseout(function () {  element.css({ 'opacity': 1.0 })  });  };  var controller = function ($scope) {  // Pending  };  return {  restrict: 'A',  controller: controller,  controllerAs: 'userStory',  link: linker  };  }); |
|  |
| element.mouseover(function () {  // ...  }).mouseout(function () {  // ...  }); |
|  |
| // client/src/angello/user/directives/UserStoryDirective.js  angular.module('Angello.User')  .directive('userstory',  function ($rootScope, StoriesModel, $log) {  //...  }); |
|  |
| // client/src/angello/user/directives/UserStoryDirective.js  angular.module('Angello.User')  .directive('userstory',  function ($rootScope, StoriesModel, $log) {  //...  var controller = function () {  var userStory = this;  userStory.deleteStory = function (id) {  StoriesModel.destroy(id)  .then(function (result) {  $rootScope.$broadcast('storyDeleted');  $log.debug('RESULT', result);  }, function (reason) {  $log.debug('ERROR', reason);  });  };  };  return {  //...  controller: controller,  };  }); |
|  |
| <!-- client/src/angello/storyboard/tmpl/storyboard.html -->  <li userstory  ng-repeat="story in storyboard.stories | filter:{status:status.name}"  class="story"  ng-click="storyboard.setCurrentStory(story)">  <article class="{{story.type}}">  <div>  <button type="button" class="close"  ng-click="deleteStory(story.id)">  ×  </button>  <p class="title">{{story.title}}</p>  </div>  <div class="type-bar {{story.type}}"></div>  <div>  <p>{{story.description}}</p>  </div>  </article>  </li> |
|  |
|  |
|  |
| // client/assets/js/boot.js  { file: 'src/angello/storyboard/directives/DragAndDrop.js' }, |
|  |
| // client/src/angello/storyboard/directives/DragAndDrop.js  angular.module('Angello.Storyboard')  .directive('dragContainer', function () {  return {  restrict: 'A',  controller: 'DragContainerController',  controllerAs: 'dragContainer',  link: function ($scope, $element, $attrs, dragContainer) {  dragContainer.init($element);  $element.on('dragstart',  dragContainer.handleDragStart.bind(dragContainer));  $element.on('dragend',  dragContainer.handleDragEnd.bind(dragContainer));  $scope.$watch($attrs.dragContainer,  dragContainer.updateDragData.bind(dragContainer));  $attrs.$observe('mimeType',  dragContainer.updateDragType.bind(dragContainer));  $attrs.$set('draggable', true);  }  };  })  .controller('DragContainerController', function ($dragging) {  }); |
|  |
| <!-- client/src/angello/storyboard/tmpl/storyboard.html -->  <li userstory  ng-repeat="story in storyboard.stories | filter: {status:status.name}"  drag-container="story" mime-type="application/x-angello-status"  class="story my-repeat-animation"  ng-click="storyboard.setCurrentStory(story)"> |
|  |
| // client/src/angello/storyboard/directives/DragAndDrop.js  angular.module('Angello.Storyboard')  .controller('DragContainerController', function ($dragging) {  var dragContainer = this;  dragContainer.init = function (el) {  dragContainer.el = el;  };  }); |
|  |
| // client/src/angello/storyboard/directives/DragAndDrop.js  angular.module('Angello.Storyboard')  .controller('DragContainerController', function ($dragging) {  //...  dragContainer.handleDragStart = function (e) {  if (e.originalEvent) e = e.originalEvent;  e.dataTransfer.dropEffect = 'move';  e.dataTransfer.effectAllowed = 'move';  dragContainer.el.addClass('drag-container-active');  dragContainer.dragging = true;  $dragging.setData(dragContainer.data);  $dragging.setType(dragContainer.type);  };  }); |
|  |
| // client/src/angello/storyboard/directives/DragAndDrop.js  angular.module('Angello.Storyboard')  .controller('DragContainerController', function ($dragging) {  //...  dragContainer.handleDragEnd = function (e) {  if (e.originalEvent) e = e.originalEvent;  angular.element(e.target).removeClass('drag-active');  dragContainer.el.removeClass('drag-container-active');  dragContainer.dragging = false;  $dragging.setData(null);  $dragging.setType(null);  };  }); |
|  |
| // client/src/angello/storyboard/directives/DragAndDrop.js  angular.module('Angello.Storyboard')  .controller('DragContainerController', function ($dragging) {  //...  dragContainer.updateDragData = function (data) {  dragContainer.data = data;  if (dragContainer.dragging)  $dragging.setData(dragContainer.data);  };  }); |
|  |
| // client/src/angello/storyboard/directives/DragAndDrop.js  angular.module('Angello.Storyboard')  .controller('DragContainerController', function ($dragging) {  //...  dragContainer.updateDragType = function (type) {  dragContainer.type = type || 'text/x-drag-and-drop';  if (dragContainer.dragging)  $dragging.setType(dragContainer.type);  };  }); |
|  |
| // client/src/angello/storyboard/directives/DragAndDrop.js  angular.module('Angello.Storyboard')  .directive('dropContainer', function ($document, $parse) {  return {  restrict: 'A',  controller: 'DropContainerController',  controllerAs: 'dropContainer',  link: function ($scope, $element, $attrs, dropContainer) {  var bindTo = function (event) {  return function (e) {  return $scope.$apply(function () {  return dropContainer['handle' + event](e);  });  };  };  var dragEnd =  dropContainer.handleDragEnd.bind(dropContainer);  var handleDragEnter = bindTo('DragEnter');  var handleDragOver = bindTo('DragOver');  var handleDragLeave = bindTo('DragLeave');  var handleDrop = bindTo('Drop');  dropContainer.init($element, $scope, {  onDragEnter: $parse($attrs.onDragEnter),  onDragOver: $parse($attrs.onDragOver),  onDragLeave: $parse($attrs.onDragLeave),  onDrop: $parse($attrs.onDrop),  });  $element.on('dragenter', handleDragEnter);  $element.on('dragover', handleDragOver);  $element.on('dragleave', handleDragLeave);  $element.on('drop', handleDrop);  $scope.$watch($attrs.accepts,  dropContainer.updateMimeTypes.bind(dropContainer));  $document.on('dragend', dragEnd);  $scope.$on('$destroy', function () {  $document.off('dragend', dragEnd);  });  }  };  }); |
|  |
| <!-- client/src/angello/storyboard/tmpl/storyboard.html -->  <li userstory  ng-repeat="story in storyboard.stories | filter: {status:status.name}"  drag-container="story" mime-type="application/x-angello-status"  drop-container="" accepts="['application/x-angello-status']"  class="story my-repeat-animation"  ng-click="storyboard.setCurrentStory(story)">  <!-- ... -->  <div class="emptystatus" drop-container=""  accepts="['application/x-angello- status']"  on-drop="storyboard.finalizeDrop(data)"  on-drag-enter="storyboard.changeStatus(data, status)"  ng-if="storyboard.isEmptyStatus(status)">  </div> |
|  |
| // client/src/angello/storyboard/directives/DragAndDrop.js  angular.module('Angello.Storyboard')  .controller('DropContainerController', function ($dragging) {  var dropContainer = this;  var targets = {};  var validAnchors = 'center top top-right right  bottom-right bottom bottom-left left top-left'.split(' ');  dropContainer.init = function (el, scope, callbacks) {  dropContainer.el = el;  dropContainer.scope = scope;  dropContainer.callbacks = callbacks;  dropContainer.accepts = ['text/x-drag-and-drop'];  dropContainer.el.addClass('drop-container');  };  }); |
|  |
| // client/src/angello/storyboard/directives/DragAndDrop.js  angular.module('Angello.Storyboard')  .controller('DropContainerController', function ($dragging) {  //...  dropContainer.addDropTarget = function (anchor, dropTarget) {  if (validAnchors.indexOf(anchor) < 0)  throw new Error('Invalid anchor point ' + anchor);  if (targets[anchor])  throw new Error('Duplicate drop targets for the anchor ' + anchor);  targets[anchor] = dropTarget;  };  dropContainer.removeDropTarget = function (anchor) {  if (targets[anchor] && targets[anchor] === anchor) {  dropContainer.activeTarget = null;  }  delete targets[anchor];  };  }); |
|  |
| // client/src/angello/storyboard/directives/DragAndDrop.js  angular.module('Angello.Storyboard')  .controller('DropContainerController', function ($dragging) {  //...  dropContainer.updateMimeTypes = function (mimeTypes) {  if (!mimeTypes) mimeTypes = ['text/x-drag-and-drop'];  if (!angular.isArray(mimeTypes)) mimeTypes = [mimeTypes];  dropContainer.accepts = mimeTypes;  };  }); |
|  |
| // client/src/angello/storyboard/directives/DragAndDrop.js  angular.module('Angello.Storyboard')  .controller('DropContainerController', function ($dragging) {  //...  dropContainer.updateDragTarget = function (e, skipUpdateTarget) {  if (e.originalEvent) e = e.originalEvent;  var activeTarget = null;  var activeAnchor = null;  var minDistanceSq = Number.MAX\_VALUE;  var prevAnchor = dropContainer.activeAnchor;  var prevTarget = dropContainer.activeTarget;  if (!skipUpdateTarget) {  angular.forEach(targets, function (dropTarget, anchor) {  var width = dropContainer.el[0].offsetWidth;  var height = dropContainer.el[0].offsetHeight;  var anchorX = width / 2;  var anchorY = height / 2;  if (anchor.indexOf('left') >= 0) anchorX = 0;  if (anchor.indexOf('top') >= 0) anchorY = 0;  if (anchor.indexOf('right') >= 0) anchorX = width;  if (anchor.indexOf('bottom') >= 0) anchorY = height;  var distanceSq = Math.pow(anchorX - e.offsetX, 2)  + Math.pow(anchorY - e.offsetY, 2);  if (distanceSq < minDistanceSq) {  activeAnchor = anchor;  activeTarget = dropTarget;  minDistanceSq = distanceSq;  }  });  }  dropContainer.activeAnchor = activeAnchor;  dropContainer.activeTarget = activeTarget;  var eventData = {  $event: e,  data: $dragging.getData(),  anchor: activeAnchor,  target: activeTarget,  prevAnchor: prevAnchor,  prevTarget: prevTarget  };  if (prevTarget !== activeTarget) {  if (prevTarget) {  dropContainer.el.  removeClass('drop-container-active-' + prevAnchor);  prevTarget.handleDragLeave(eventData);  }  if (activeTarget) {  dropContainer.el.  addClass('drop-container-active-' + activeAnchor);  activeTarget.handleDragEnter(eventData);  }  }  return eventData;  };  }); |
|  |
| // client/src/angello/storyboard/directives/DragAndDrop.js  angular.module('Angello.Storyboard')  .controller('DropContainerController', function ($dragging) {  //...  dropContainer.handleDragEnter = function (e) {  if (e.originalEvent) e = e.originalEvent;  if (!dropContainer.accepts  || dropContainer.accepts.indexOf($dragging.getType()) >= 0) {  e.preventDefault();  } else {  return;  }  var eventData = dropContainer.updateDragTarget(e);  dropContainer.el.children().css({'pointer-events': 'none'});  dropContainer.el.addClass('drop-container-active');  if (dropContainer.callbacks.onDragEnter) {  dropContainer.callbacks  .onDragEnter(dropContainer.scope, eventData);  }  };  }); |
|  |
| // client/src/angello/storyboard/directives/DragAndDrop.js  angular.module('Angello.Storyboard')  .controller('DropContainerController', function ($dragging) {  //...  dropContainer.handleDragLeave = function (e) {  if (e.originalEvent) e = e.originalEvent;  var eventData = dropContainer.updateDragTarget(e, true);  dropContainer.el.children().css({'pointer-events': null});  dropContainer.el.removeClass('drop-container-active');  if (dropContainer.callbacks.onDragLeave) {  dropContainer.callbacks  .onDragLeave(dropContainer.scope, eventData);  }  };  }); |
|  |
| // client/src/angello/storyboard/directives/DragAndDrop.js  angular.module('Angello.Storyboard')  .controller('DropContainerController', function ($dragging) {  //...  dropContainer.handleDragEnd = function (e) {  dropContainer.el.children().css({'pointer-events': null});  dropContainer.el.removeClass('drop-container-active');  };  }); |
|  |
| // client/src/angello/storyboard/directives/DragAndDrop.js  angular.module('Angello.Storyboard')  .controller('DropContainerController', function ($dragging) {  //...  dropContainer.handleDrop = function (e) {  if (e.originalEvent) e = e.originalEvent;  if (!dropContainer.accepts  || dropContainer.accepts.indexOf($dragging.getType()) >= 0) {  e.preventDefault();  } else {  return;  }  var eventData = dropContainer.updateDragTarget(e);  if (eventData.target) {  eventData.target.handleDrop(eventData);  }  if (dropContainer.callbacks.onDrop) {  dropContainer.callbacks  .onDrop(dropContainer.scope, eventData);  }  dropContainer.handleDragEnd(e);  };  }); |
|  |
| // client/src/angello/storyboard/directives/DragAndDrop.js  angular.module('Angello.Storyboard')  .directive('dropTarget', function ($parse) {  return {  restrict: 'A',  require: ['^dropContainer', 'dropTarget'],  controller: 'DropTargetController',  controllerAs: 'dropTarget',  link: function ($scope, $element, $attrs, ctrls) {  var dropContainer = ctrls[0];  var dropTarget = ctrls[1];  var anchor = $attrs.dropTarget || 'center';  var destroy =  dropContainer.removeDropTarget.bind(dropContainer, anchor);  $element.addClass('drop-target drop-target-' + anchor);  dropTarget.init($element, $scope, {  onDragEnter: $parse($attrs.onDragEnter),  onDragOver: $parse($attrs.onDragOver),  onDragLeave: $parse($attrs.onDragLeave),  onDrop: $parse($attrs.onDrop),  });  dropContainer.addDropTarget(anchor, dropTarget);  $scope.$on('$destroy', destroy);  }  };  })  .controller('DropTargetController', function () {  }); |
|  |
| <!-- client/src/angello/storyboard/tmpl/storyboard.html -->  <li userstory  ng-repeat="story in storyboard.stories | filter: {status:status.name}"  drag-container="story" mime-type="application/x-angello-status"  drop-container="" accepts="['application/x-angello-status']"  class="story my-repeat-animation"  ng-click="storyboard.setCurrentStory(story)">  <div drop-target="top"  on-drag-enter="storyboard.insertAdjacent(story, data, true)"  on-drop="storyboard.finalizeDrop(data)"></div>  <div drop-target="bottom"  on-drag-enter="storyboard.insertAdjacent(story, data, false)"  on-drop="storyboard.finalizeDrop(data)"></div>  <!-- ... -->  </li> |
|  |
| // client/src/angello/storyboard/directives/DragAndDrop.js  angular.module('Angello.Storyboard')  .controller('DropTargetController', function () {  var dropTarget = this;  dropTarget.init = function (el, scope, callbacks) {  dropTarget.el = el;  dropTarget.scope = scope;  dropTarget.callbacks = callbacks;  };  }); |
|  |
| // client/src/angello/storyboard/directives/DragAndDrop.js  angular.module('Angello.Storyboard')  .controller('DropTargetController', function () {  //...  dropTarget.handleDragEnter = function (eventData) {  dropTarget.el.addClass('drop-target-active');  if (dropTarget.callbacks.onDragEnter) {  dropTarget.callbacks  .onDragEnter(dropTarget.scope, eventData);  }  };  dropTarget.handleDragOver = function (eventData) {  if (dropTarget.callbacks.onDragOver) {  dropTarget.callbacks  .onDragOver(dropTarget.scope, eventData);  }  };  dropTarget.handleDragLeave = function (eventData) {  dropTarget.el.removeClass('drop-target-active');  if (dropTarget.callbacks.onDragLeave) {  dropTarget.callbacks  .onDragLeave(dropTarget.scope, eventData);  }  };  dropTarget.handleDrop = function (eventData) {  if (dropTarget.callbacks.onDrop) {  dropTarget.callbacks.onDrop(dropTarget.scope, eventData);  }  };  }); |
|  |
| // client/src/angello/storyboard/directives/DragAndDrop.js  angular.module('Angello.Storyboard')  .factory('$dragging', function () {  var data = null;  var type = null;  return {  getData: function () {  return data;  },  getType: function () {  return type;  },  setData: function (newData) {  data = newData;  return data;  },  setType: function (newType) {  type = newType;  return type;  }  };  }); |
|  |
| // client/src/angello/storyboard/controllers/StoryboardController.js  angular.module('Angello.Storyboard')  .controller('StoryboardCtrl',  function ($scope, $log, StoriesModel, UsersModel,  STORY\_STATUSES, STORY\_TYPES) {  //...  storyboard.insertAdjacent = function (target, story, insertBefore) {  if (target === story) return;  var fromIdx = storyboard.stories.indexOf(story);  var toIdx = storyboard.stories.indexOf(target);  if (!insertBefore) toIdx++;  if (fromIdx >= 0 && toIdx >= 0) {  storyboard.stories.splice(fromIdx, 1);  if (toIdx >= fromIdx) toIdx--;  storyboard.stories.splice(toIdx, 0, story);  story.status = target.status;  }  };  storyboard.finalizeDrop = function (story) {  StoriesModel.update(story.id, story)  .then(function (result) {  $log.debug('RESULT', result);  }, function (reason) {  $log.debug('REASON', reason);  });  };  storyboard.changeStatus = function (story, status) {  story.status = status.name;  };  //...  }); |
|  |
|  |
|  |
| // client/assets/js/boot.js  { file: 'vendor/flot/jquery.flot.js' },  { file: 'vendor/flot/jquery.flot.categories.js' }, |
|  |
| // client/src/angello/dashboard/directives/ChartDirective.js  angular.module('Angello.Dashboard')  .directive('chart',  function () {  var linker = function (scope, element, attrs) {  // Link goes here  };  var controller = function($scope) {  // Controller goes here  };  return {  restrict: 'A',  link: linker,  controller: controller  };  }); |
|  |
| <!-- client/src/angello/dashboard/tmpl/dashboard.html -->  <div class="container chart-wrapper">  <h3>User Stories by Status</h3>  <hr/>  <div class="chart-container">  <div chart class="chart-placeholder"></div>  </div>  <h3>User Stories by Type</h3>  <hr/>  <div class="chart-container">  <div chart class="chart-placeholder"></div>  </div>  </div> |
|  |
| var data = [  ["January", 10],  ["February", 8],  ["March", 4],  ["April", 13],  ["May", 17],  ["June", 9]  ]; |
|  |
| var data = [  ["Log", 1],  ["To Do", 2],  ["In Progress", 0],  ["Code Review", 1],  ["QA Review", 0],  ["Verified", 1],  ["Done", 1]  ]; |
|  |
| // client/src/angello/dashboard/directives/ChartDirective.js  angular.module('Angello.Dashboard')  .directive('chart',  function () {  var parseDataForCharts = function(sourceArray, sourceProp,  referenceArray, referenceProp) {  var data = [];  referenceArray.each(function (r) {  var count = sourceArray.count(function (s) {  return s[sourceProp] == r[referenceProp];  });  data.push([r[referenceProp], count]);  });  return data;  };  //...  }); |
|  |
| data.push([r[referenceProp], count]); |
|  |
| // client/src/angello/dashboard/directives/ChartDirective.js  angular.module('Angello.Dashboard')  .directive('chart',  function () {  //...  return {  restrict: 'A',  link: linker,  controller: controller,  scope: {  sourceArray: '=',  referenceArray: '='  }  };  }); |
|  |
| <!-- client/src/angello/dashboard/tmpl/dashboard.html -->  <div class="container chart-wrapper">  <h3>User Stories by Status</h3>  <hr/>  <div class="chart-container">  <div chart class="chart-placeholder"  source-array="dashboard.stories" source-prop="status"  reference-array="dashboard.statuses" reference-prop="name">  </div>  </div>  <h3>User Stories by Type</h3>  <hr/>  <div class="chart-container">  <div chart class="chart-placeholder"  source-array="dashboard.stories" source-prop="type"  reference-array="dashboard.types" reference-prop="name">  </div>  </div>  </div> |
|  |
| // client/src/angello/dashboard/tmpl/dashboard.html  angular.module('Angello.Dashboard')  .directive('chart', function () {  //...  var linker = function (scope, element, attrs) {  scope.$watch('sourceArray', function () {  scope.data = parseDataForCharts(  scope.sourceArray,  attrs['sourceProp'],  scope.referenceArray,  attrs['referenceProp']  );  if(element.is(':visible')){  $.plot(element, [ scope.data ], {  series: {  bars: {  show: true,  barWidth: 0.6,  align: "center"  }  },  xaxis: {  mode: "categories",  tickLength: 0  }  });  }  });  };  //...  }); |
|  |
| client/tests/specs/directives/UserStoryDirective.spec.js  'use strict';  describe('userstory Directive', function () {  var userStory,  element,  StoriesModel,  $rootScope;  beforeEach(module('Angello.User'));  }); |
|  |
| client/tests/specs/directives/UserStoryDirective.spec.js  'use strict';  describe('userstory Directive', function () {  //...  beforeEach(inject(function($q, $compile, \_$rootScope\_, \_StoriesModel\_) {  $rootScope = \_$rootScope\_;  var directiveMarkup = angular.element('<li userstory></li>');  element = $compile(directiveMarkup)($rootScope);  userStory = element.scope().userStory;  StoriesModel = \_StoriesModel\_;  spyOn(StoriesModel, 'destroy').and.callFake(function() {  var deferred = $q.defer();  deferred.resolve('data');  return deferred.promise;  });  spyOn($rootScope,'$broadcast').and.callThrough();  }));  }); |
|  |
| client/tests/specs/directives/UserStoryDirective.spec.js  it('should delete a story', function() {  userStory.deleteStory('0');  expect(StoriesModel.destroy).toHaveBeenCalledWith('0');  $rootScope.$digest();  expect($rootScope.$broadcast).toHaveBeenCalledWith('storyDeleted');  }); |
|  |
| Best practices.  DOM manipulation should be done in the link function and imperative logic in the controller. One of the advantages of using a JavaScript framework like AngularJS is the separation of concerns, especially the separation of the DOM from our imperative logic. We like to keep this theme rolling by putting all of the DOM manipulation logic in our directives’ link function and all of our business logic in our directives’ controller. |
|  |
| Cap.7 |
| Structuring your site with routes. |
|  |
|  |
|  |
| // client/assets/js/boot.js  { file:  '//cdnjs.cloudflare.com/ajax/libs/angular.js/1.3.3/angular-route.min.js'  }, |
|  |
| // client/src/angello/Angello.js  var myModule = angular.module('Angello',  [  'ngRoute',  //...  ]); |
|  |
| <!-- client/index.html -->  <body ng-controller="MainCtrl as main" ng-class="{loading:loadingView}">  <!-- ... -->  <div ng-view=""></div>  <!-- ... -->  </body> |
|  |
|  |
|  |
|  |
|  |
| <!-- client/index.html -->  <div class="navbar navbar-fixed-top navbar-default">  <div class="navbar-header">  <a class="logo navbar-brand" href="#/">  <img src="assets/img/angello.png">  </a>  </div>  <div class="btn-group pull-right" ng-show="main.currentUser">  <a class="btn btn-danger" href="#/">  <span class="glyphicon glyphicon-home"></span>  </a>  <a class="btn btn-danger" href="#/users">  <span class="glyphicon glyphicon-user"></span>  </a>  <a class="btn btn-danger" href="#/dashboard">  <span class="glyphicon glyphicon-signal"></span>  </a>  <button class="btn btn-default" ng-click="main.logout()">  <span class="glyphicon glyphicon-log-out"></span>  </button>  </div>  </div> |
|  |
|  |
|  |
|  |
|  |
| // client/src/angello/Angello.js  myModule.config(function ($routeProvider, $httpProvider, $provide) {  $routeProvider  //...  .when('/users/:userId', {  templateUrl: 'src/angello/user/tmpl/user.html',  controller: 'UserCtrl',  controllerAs: 'myUser'  })  //...  .otherwise({redirectTo: '/'});  }); |
|  |
| // client/src/angello/user/controllers/UserController.js  angular.module('Angello.User')  .controller('UserCtrl',  function ('$routeParams') {  var myUser = this;  myUser.userId = $routeParams['userId'];  }); |
|  |
| <!-- client/src/angello/user/tmpl/users.html -->  <tr ng-repeat="user in users.users">  <!-- ... -->  <td>  <button type="button" class="btn btn-link"  ng-click="users.removeUser(user.id)">Remove</button>  <a class="btn btn-link" href="#/users/{{user.id}}">View</a>  </td>  </tr> |
|  |
|  |
|  |
| // client/src/angello/user/controllers/UserController.js  angular.module('Angello.User')  .controller('UserCtrl',  function ($routeParams, user, stories) {  //...  myUser.user = user.data;  myUser.getAssignedStories = function (userId, stories) {  var assignedStories = {};  Object.keys(stories, function(key, value) {  if (value.assignee == userId) {  assignedStories[key] = stories[key];  }  });  return assignedStories;  };  myUser.stories =  myUser.getAssignedStories(myUser.userId, stories);  }); |
|  |
|  |
|  |
|  |
|  |
| client/tests/specs/routes/UserRoute.spec.js  describe('User Route', function () {  var $route,  $rootScope,  $location,  url = 'login';  // Inject and assign the $route and $rootScope services.  // Put the template in template cache.  beforeEach(module('Angello'));  beforeEach(inject(function  (\_$location\_, \_$route\_, $templateCache, \_$rootScope\_) {  $route = \_$route\_;  $rootScope = \_$rootScope\_;  $location = \_$location\_;  $templateCache.put('src/angello/login/tmpl/login.html', '');  }));  }); |
|  |
| client/tests/specs/routes/UserRoute.spec.js  describe('User Route', function () {  //...  it('should be defined with  correct controller and templateUrl', function() {  $location.path(url);  $rootScope.$digest();  expect($route.current.controller).toEqual('LoginCtrl');  expect($route.current.controllerAs).toEqual('login');  expect($route.current.templateUrl)  .toEqual('src/angello/login/tmpl/login.html');  });  }); |
|  |
| Cap.8 |
| Forms and validations. |
|  |
|  |
|  |
| <!-- client/src/angello/storyboard/tmpl/storyboard.html -->  <form name="storyboard.detailsForm"></form> |
|  |
| <!-- client/src/angello/storyboard/tmpl/storyboard.html -->  <div ng-if="storyboard.currentStory">  <button class="btn btn-default" ng-click="storyboard.updateCancel()">  Cancel  </button>  <button class="btn pull-right btn-default"  ng-disabled="!storyboard.detailsForm.$valid"  ng-click="storyboard.updateStory()">Update</button>  </div>  <div ng-if="!storyboard.currentStory">  <button class="btn pull-right btn-default"  ng-disabled="!storyboard.detailsForm.$valid"  ng-click="storyboard.createStory()">New Story</button>  </div> |
|  |
| // client/assets/js/boot.js  //...  { file:'//cdnjs.cloudflare.com/ajax/libs/angular.js/1.3.3/angular-messages.min.js' },  //... |
|  |
| // client/src/angello/Angello.js  var myModule = angular.module('Angello',  [  //...  'ngMessages',  //...  ]); |
|  |
| <!-- client/src/angello/storyboard/tmpl/storyboard.html -->  <input class="form-control" type="text" id="inputTitle" name="inputTitle"  placeholder="Title" ng-model="storyboard.editedStory.title">  <div class="alert alert-warning"  ng-messages="storyboard.detailsForm.inputTitle.$error"  ng-if="storyboard.showMessages('inputTitle')">  <div ng-message="required">  <small>Required!</small>  </div>  </div> |
|  |
| // client/src/angello/storyboard/controllers/StoryboardController.js  angular.module('Angello.Storyboard')  .controller('StoryboardCtrl',  function($scope, $log, StoriesModel, UsersModel,  STORY\_STATUSES, STORY\_TYPES) {  //...  storyboard.showMessages = function (field) {  return storyboard.detailsForm[field].$touched  && storyboard.detailsForm[field].$invalid;  };  //...  }); |
|  |
|  |
|  |
| <!-- client/src/angello/storyboard/tmpl/storyboard.html -->  <label class="control-label" for="inputTitle">\*Title</label>  <input class="form-control" type="text" id="inputTitle" name="inputTitle"  placeholder="Title" ng-model="storyboard.editedStory.title"  ng-required="true"> |
|  |
|  |
|  |
| <!-- client/src/angello/storyboard/tmpl/storyboard.html -->  <label class="control-label" for="inputTitle">\*Title</label>  <input class="form-control" type="text" id="inputTitle" name="inputTitle"  placeholder="Title" ng-model="storyboard.editedStory.title"  ng-required="true" ng-minlength="3">  <div class="alert alert-warning"  ng-messages="storyboard.detailsForm.inputTitle.$error"  ng-if="storyboard.showMessages('inputTitle')">  <div ng-message="required">  <small>Required!</small>  </div>  <div ng-message="minlength"> <small>Too short!</small> </div>  </div> |
|  |
| // Javascript  var INTEGER\_REGEXP = /^\-?\d+$/;  app.directive('integer', function() {  return {  require: 'ngModel',  link: function(scope, elm, attrs, ctrl) {  ctrl.$validators.integer  = function(modelValue, viewValue) {  if (ctrl.$isEmpty(modelValue)) {  // consider empty models to be valid  return true;  }  if (INTEGER\_REGEXP.test(viewValue)) {  // it is valid  return true;  }  // it is invalid  return false;  };  }  };  });  // HTML  <input type="number" ng-model="size" name="size"  min="0" max="10" integer />{{size}}<br />  <span ng-show="form.size.$error.integer">  The value is not a valid integer!</span>  <span ng-show="form.size.$error.min  || form.size.$error.max">  The value must be in range 0 to 10!</span> |
|  |
|  |
|  |
| <!-- client/src/angello/storyboard/tmpl/storyboard.html -->  <label class="control-label" for="inputTitle">\*Title</label>  <input class="form-control" type="text" id="inputTitle" name="inputTitle"  placeholder="Title" ng-model="storyboard.editedStory.title"  ng-required="true" ng-minlength="3" ng-maxlength="30">  <div class="alert alert-warning"  ng-messages="storyboard.detailsForm.inputTitle.$error"  ng-if="storyboard.showMessages('inputTitle')">  <!-- ... -->  <div ng-message="maxlength"> <small>Too long!</small> </div>  </div> |
|  |
| /\* client/assets/css/angello.css \*/  form.ng-dirty input.ng-invalid {  border: 1px solid #B02B2C;  }  form.ng-dirty input.ng-valid {  border: 1px solid #6BBA70;  } |
|  |
| <!-- angello/storyboard/tmpl/storyboard.html -->  <input class="form-control ng-pristine ng-invalid ng-invalid-required  ng-valid-minlength ng-valid-maxlength ng-touched"  type="text" id="inputTitle" name="inputTitle"  placeholder="Title" ng-model="storyboard.editedStory.title"  ng-required="true" ng-minlength="3" ng-maxlength="30"> |
|  |
| // client/src/angello/storyboard/controllers/StoryboardController.js  storyboard.resetForm = function () {  storyboard.currentStory = null;  storyboard.editedStory = {};  storyboard.detailsForm.$setPristine();  storyboard.detailsForm.$setUntouched();  }; |
|  |
| client/tests/specs/forms/StoryboardForm.spec.js  describe('Storyboard form', function() {  var scope, ctrl;  beforeEach(module('Angello.Storyboard'));  beforeEach(module('Angello.Templates'));  beforeEach(inject(function($q, $rootScope, $controller,  $templateCache, $compile) {  var UsersModel = {  all: function() {  var deferred = $q.defer();  deferred.resolve({});  return deferred.promise;  }  };  var StoriesModel = {  all: function() {  var deferred = $q.defer();  deferred.resolve({});  return deferred.promise;  }  };  }));  }); |
|  |
| client/tests/specs/forms/StoryboardForm.spec.js  describe('Storyboard form', function() {  //...  beforeEach(inject(function($q, $rootScope, $controller,  $templateCache, $compile) {  //...  scope = $rootScope.$new();  ctrl = $controller('StoryboardCtrl', {  $scope: scope,  STORY\_STATUSES: {},  STORY\_TYPES: {},  UsersModel: UsersModel,  StoriesModel: StoriesModel  });  scope.storyboard = ctrl;  var templateHtml =  $templateCache.get('src/angello/storyboard/tmpl/storyboard.html');  var formElem = angular.element(templateHtml);  $compile(formElem)(scope);  scope.$digest()  }));  }); |
|  |
| client/tests/specs/forms/StoryboardForm.spec.js  describe('Storyboard form', function() {  //...  it('should be invalid by default', function() {  expect(ctrl.detailsForm.$invalid).toBeTruthy();  });  it('should be valid with populated fields', function() {  ctrl.editedStory = {  title: 'Title',  status: 'To Do',  type: 'Enhancement',  reporter: 'Lukas Ruebbelke',  assignee: 'Brian Ford'  };  scope.$digest();  expect(ctrl.detailsForm.$valid).toBeTruthy();  });  }); |
|  |
| // Hypothetical Scenario  <form name="myForm">  <div ng-repeat="item in items"  ng-class="{ 'has-error' : item.name.$invalid }">  <ng-form name="itemNameForm">  <label>Name</label>  <input type="text" name="name"  ng-model="item.name" required>  <p ng-show="itemNameForm.name.$invalid">  Name Required</p>  </ng-form>  </div>  </form> |
|  |
| <https://nodejs.org/download/> |