

Tipo : Guía de Laboratorio
Capítulo : Angular JS
Duración : 60 minutos

I. OBJETIVO

Refactoriza el código de la vista de product y crea un componente modal para la visualización del formulario de edición, creación y eliminación.

II. REQUISITOS

Los siguientes elementos de software son necesarios para la realización del laboratorio:

- Windows 10 (como mínimo Windows 8.1)
- Visual Studio 2017
- Navegador Web

III. EJECUCIÓN DEL LABORATORIO

Actividades

1. En el proyecto **Cibertec.Angular** crea la carpeta “**app/components**”
2. Procedemos a crear la carpeta “**modal**” y en ella agrega el fichero “**modal.directive.js**” con el siguiente código:

```
(function () {  
    angular.module('app')  
        .directive('modalPanel', modalPanel);  
  
    function modalPanel() {  
        return {  
            templateUrl: 'app/components/modal/modal-directive.html',  
            restrict: 'E',  
            transclude: true,  
            scope: {  
                title: '@',  
                buttonTitle: '@',  
                saveFunction: '=',  
                closeFunction: '=',  
                readOnly: '=',  
                isDelete: '='  
            }  
        };  
    }  
})();
```

3. Agrega el fichero “**modal-directive.html**” con el siguiente código:

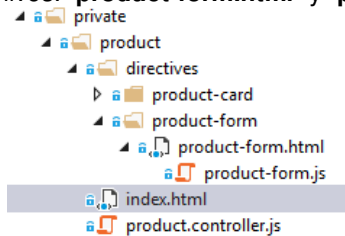
```
<div class="modal fade" id="modal-container" tabindex="-1" role="dialog"  
    aria-labelledby="myModalLabel">  
    <div class="modal-dialog" role="document">  
        <div class="modal-content">  
            <div class="modal-header">  
                <button type="button" class="close"  
                    data-dismiss="modal"
```

```

        aria-label="Close"
        ng-click="closeFunction();">
        <span aria-hidden="true">&times;</span>
    </button>
    <h4 class="modal-title">{{title}}</h4>
</div>
<div class="modal-body">
    <ng-transclude></ng-transclude>
    <button ng-if="!readOnly && !isDelete"
        class="btn btn-success"
        type="submit"
        ng-
click="saveFunction();">{{buttonTitle}}</button>
    <button ng-if="isDelete"
        class="btn btn-danger"
        type="submit"
        ng-click="saveFunction();">
        {{buttonTitle}}
    </button>
    <button type="button"
        class="btn btn-warning"
        data-dismiss="modal"
        ng-click="closeFunction();"
        aria-label="Close">Cerrar</button>
</div>
</div>
</div>
</div>

```

4. En la carpeta “**product-form**” como parte de una nueva directiva para el producto y en ella crea los siguientes archivos: “**product-form.html**” y “**product-form.js**”.



5. En el fichero **product-form.js** agrega el siguiente código:

```

(function () {
    'use strict';
    angular.module('app')
        .directive('productForm', productForm);

    function productForm() {
        return {
            restrict: 'E',
            scope: {
                product: '='
            },
            templateUrl: 'app/private/product/directives/product-
form/product-form.html'
        };
    }

})();

```

6. En el fichero **product-form.html** agrega el siguiente código:

```
<div>
  <input type="hidden" ng-model="product.id" value="{{product.id}}" />
  <div>
    <label for="productName">Producto:</label>
    <input type="text" ng-model="product.productName"
value="{{product.productName}}" />
  </div>
  <div>
    <label for="supplierId">Id de Proveedor:</label>
    <input type="text" ng-model="product.supplierId"
value="{{product.supplierId}}" />
  </div>
  <div>
    <label for="unitPrice">Precio Unitario:</label>
    <input type="text" ng-model="product.unitPrice"
value="{{product.unitPrice}}" />
  </div>
  <div>
    <label for="package">Paquete:</label>
    <input type="text" ng-model="product.package"
value="{{product.package}}" />
  </div>
  <div>
    <label for="isDiscontinued">Descontinuado?:</label>
    <input type="checkbox" ng-model="product.isDiscontinued"
checked="{{product.isDiscontinued? 'checked': ''}}"
value="{{product.isDiscontinued}}" />
  </div>
</div>
```

7. Procedemos a refactorizar el **index.html** del product de la siguiente manera:

```
<div ng-controller="productController as vm">
  <div ng-show="vm.showCreate" class="alert alert-success alert-
dismissable">
    <a href="" class="close" data-dismiss="alert" aria-
label="close">&times;</a>
    <strong>OK</strong> The product was created properly.
  </div>
  <div>
    <button ng-click="vm.create();" data-toggle="modal" data-
target="#modal-container" class="btn btn-success btn-sm">
      <span class="glyphicon glyphicon-erase"></span> New Product
    </button>
  </div>
  <div class="row">
    <product-card ng-repeat="product in vm.productList "
id="{{::product.id}}"
product-name="{{::product.productName}}"
supplier-id="{{::product.supplierId}}"
unit-price="{{::product.unitPrice}}"
package="{{::product.package}}"
is-discontinued="product.isDiscontinued">
      <button ng-click="vm.getProduct(product.id); vm.edit();"
data-toggle="modal" data-target="#modal-container" class="btn btn-success
btn-sm">
        <span class="glyphicon glyphicon-erase"></span> Edit
      </button>
      <button ng-click="vm.getProduct(product.id); vm.delete();"
data-toggle="modal" data-target="#modal-container" class="btn btn-danger
btn-sm">
        <span class="glyphicon glyphicon-erase"></span> Delete
```

```

        </button>
    </product-card>
    <div ng-repeat-end ng-if="($index+1) % 3 == 0"
class="clearfix"></div>
</div>
<modal-panel title="{{vm.modalTitle}}"
button-title="{{vm.modalButtonText}}"
save-function="vm.modalFunction"
close-function="vm.closeModal"
read-only="vm.readOnly"
is-delete="vm.isDelete">
    <product-form product="vm.product">
    </product-form>
</modal-panel>
</div>

```

8. También refactorizamos el fichero product.controller.js con el siguiente código:

```

(function () {
    'use strict';
    angular.module('app')
        .controller('productController', productController);

    productController.$inject = ['dataService', 'configService',
    '$state'];

    function productController(dataService, configService, $state) {
        var apiUrl = configService.getApiUrl();
        var vm = this;

        //Propiedades
        vm.product = {};
        vm.productList = [];
        vm.modalButtonText = '';
        vm.readOnly = false;
        vm.isDelete = false;
        vm.modalTitle = '';
        vm.showCreate = false;

        //Funciones
        vm.getProduct = getProduct;
        vm.create = create;
        vm.edit = edit;
        vm.delete = productDelete;
        //vm.closeModal = closeModal;

        init();

        function init() {
            if (!configService.getLogin()) return $state.go('login');
            list();
        }

        function list() {
            dataService.getData(apiUrl + '/product')
                .then(function (result) {
                    vm.productList = result.data;
                },
                function (error) {
                    vm.productList = [];
                    console.log(error);
                });
        }
    }
}

```

```

function getProduct(id) {
    vm.product = null;
    dataService.getData(apiUrl + '/product/' + id)
        .then(function (result) {
            vm.product = result.data;
        },
        function (error) {
            vm.product = null;
            console.log(error);
        });
}

function updateProduct() {
    if (!vm.product) return;
    dataService.putData(apiUrl + '/product', vm.product)
        .then(function (result) {
            vm.product = {};
            list();
            closeModal();
        },
        function (error) {
            vm.product = {};
            console.log(error);
        });
}

function createProduct() {
    if (!vm.product) return;
    dataService.postData(apiUrl + '/product', vm.product)
        .then(function (result) {
            getProduct(result.data.id);
            list();
            vm.showCreate = true;
        },
        function (error) {
            console.log(error);
        });
}

function deleteProduct() {
    dataService.deleteData(apiUrl + '/product/' + vm.product.id)
        .then(function (result) {
            list();
            closeModal();
        },
        function (error) {
            console.log(error);
        });
}

function create() {
    vm.product = {};
    vm.modalTitle = 'New Product';
    vm.modalButtonText = 'Create';
    vm.readOnly = false;
    vm.modalFunction = createProduct;
    vm.isDelete = false;
}

function edit() {
    vm.showCreate = false;
}

```

```

        vm.modalTitle = 'Edit Product';
        vm.modalButtonTitle = 'Update';
        vm.readOnly = false;
        vm.modalFunction = updateProduct;
        vm.isDelete = false;
    }

    function detail() {
        vm.modalTitle = 'Created Product';
        vm.modalButtonTitle = '';
        vm.readOnly = true;
        vm.modalFunction = null;
        vm.isDelete = false;
    }

    function productDelete() {
        vm.showCreate = false;

        vm.modalTitle = 'Delete Product';
        vm.modalButtonTitle = 'Delete';
        vm.readOnly = false;
        vm.modalFunction = deleteProduct;
        vm.isDelete = true;
    }

    function closeModal() {
        angular.element('#modal-container').modal('hide');
    }
}
})();

```

9. Iniciamos con la refactorización del bundleconfig.json de la siguiente manera:

```

[
    {
        "outputFileName": "wwwroot/css/site.min.css",
        "inputFiles": [
            "wwwroot/css/site.css"
        ]
    },
    {
        "outputFileName": "wwwroot/js/angular.js",
        "inputFiles": [
            "wwwroot/libraries/angular/angular.js",
            "wwwroot/libraries/angular/angular-ui-router.js",
            "wwwroot/libraries/angular/angular-local-storage.js"
        ],
        "minify": {
            "enabled": true,
            "renameLocals": true
        }
    },
    {
        "outputFileName": "wwwroot/js/bootstrap.js",
        "inputFiles": [
            "wwwroot/libraries/jquery/jquery.js",
            "wwwroot/libraries/bootstrap/js/bootstrap.min.js"
        ],
        "minify": {
            "enabled": true,
            "renameLocals": true
        }
    }
]

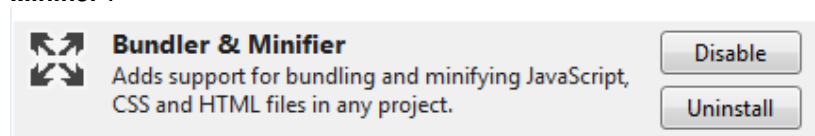
```

```

    },
    {
      "outputFileName": "wwwroot/js/app-site.js",
      "inputFiles": [
        "wwwroot/app/app.js",
        "wwwroot/app/app.routes.js",
        "wwwroot/app/app.config.js",
        "wwwroot/app/app.controller.js"
      ],
      "minify": {
        "enabled": true,
        "renameLocals": true
      }
    },
    {
      "inputFiles": [
        "wwwroot/app/shared/**/*.js",
        "wwwroot/app/components/**/*.js",
        "wwwroot/app/public/**/*.js",
        "wwwroot/app/private/**/*.js"
      ],
      "minify": {
        "enabled": true,
        "renameLocals": true
      }
    },
    "outputFileName": "wwwroot/js/app-cibertec.js"
  }
]

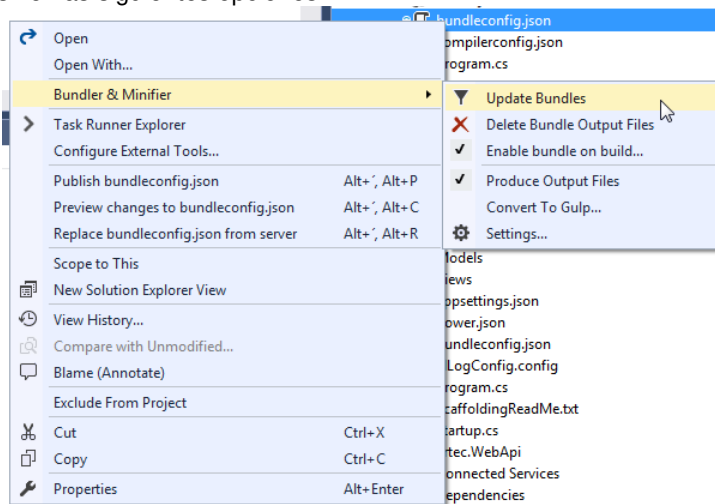
```

10. Para poder validar los cambios necesitamos de una extensión llamada: **“Bundler & Minifier”**.

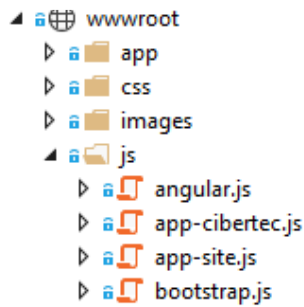


Instalamos y reiniciamos Visual Studio 2017.

11. Una vez reiniciada nuestro Visual Studio, hacemos clic derecho en bundleconfig.json y lograremos ver las siguientes opciones:



12. Hacemos clic en la opción **Update Bundles** y validamos en la carpeta **“wwwroot/js”** la existencia de los siguiente ficheros:



13. Con el bundle trabajando correctamente procedemos a refactorizar el _Layout.cshtml modificando las anteriores referencias a los javascript con esta implementación:

```
<!DOCTYPE html>
<html ng-app="app">
<head>
  <meta charset="utf-8" />
  <meta name="viewport" content="width=device-width, initial-scale=1.0" />
  <title>Cibertec.Angular</title>
  <environment names="Development, Production">
    <link href="~/libraries/bootstrap/css/bootstrap.min.css"
rel="stylesheet" />
    <link rel="stylesheet" href="~/css/site.css" />
  </environment>
</head>
<body ng-controller="applicationController as vm"
  ng-init="init('@ViewBag.WebApiUrl')">
  <nav class="navbar navbar-inverse navbar-fixed-top">
    <div class="container">
      <div class="navbar-header">
        <button type="button" class="navbar-toggle" data-
toggle="collapse" data-target=".navbar-collapse">
          <span class="sr-only">Toggle navigation</span>
          <span class="icon-bar"></span>
          <span class="icon-bar"></span>
          <span class="icon-bar"></span>
        </button>
        <a ui-sref="home" ui-sref-active="active" class="navbar-
brand">Cibertec.Angular</a>
      </div>
      <div class="navbar-collapse collapse">
        <ul class="nav navbar-nav">
          <li><a ng-if="!vm.validate()" ui-sref="login" ui-
sref-active="active">Login</a></li>
          <li><a ng-if="vm.validate()" ng-click="vm.logout()"
href="#" ui-sref-active="active">Logout</a></li>
          <li><a ng-if="vm.validate()" ui-sref="product" ui-
sref-active="active">Product</a></li>
        </ul>
      </div>
    </div>
  </nav>
  <div class="container body-content">
    @RenderBody()
    <hr />
    <footer>
      <p>&copy; 2017 - Cibertec.Angular</p>
    </footer>
  </div>
</environment names="Development, Production">
```



```

        <script src="../../js/bootstrap.js"></script>
        <script src="../../js/angular.js"></script>
        <script src="../../js/app-site.js"></script>
        <script src="../../js/app-cibertec.js"></script>
    </environment>
    <environment names="Production">
        <script src="../../js/bootstrap.min.js"></script>
        <script src="../../js/angular.min.js"></script>
        <script src="../../js/app-site.min.js"></script>
        <script src="../../js/app-cibertec.min.js"></script>
    </environment>
</body>
</html>

```

Con estos cambios estamos buscando es simplificar el modo de usar nuestros ficheros javascript.

14. Para poder realizar la prueba final de este laboratorio debemos de terminar con el laboratorio siguiente que nos permitirá