Exercise (Instructions): Handling Errors in Client-Server Communication using $http

Objectives and Outcomes

In this exercise, we will update the code in the controllers to be able to handle errors that might be caused when errors occur while retrieving data from the server. At the end of this exercise, you will be able to:

- Handling errors in communication between the client and the server.
- Ensure that the user is delivered meaningful message on the web page to indicate error in communicating with the server.

Updating the Controllers

- Open *controllers.js* to update the code to handle errors. Update the code in the *MenuController* as follows:

```
            $scope.showMenu = false;
            $scope.message = "Loading ...";
                        $scope.dishes= {};
                        menuFactory.getDishes()
            .then(
                function(response) {
                    $scope.dishes = response.data;
                    $scope.showMenu = true;
                },
                function(response) {
                    $scope.message = "Error: "+response.status + "
" + response.statusText;
                }
            );
```

- Now, open *menu.html* and update the code as follows:

```
        <div class="row row-content" ng-controller="MenuControlle
r">
            <div class="col-xs-12" ng-if="!showMenu">
                <h3>{{message}}</h3>
            </div>
            <div class="col-xs-12" ng-if="showMenu">
```

Note the use of the *ngIf* directive in order to add/delete the div from the DOM.

- Next, update *DishDetailController* in *controllers.js* as follows:

```
            $scope.dish = {};
            $scope.showDish = false;
            $scope.message="Loading ...";
                        menuFactory.getDish(parseInt($stateParams.i
d,10))
            .then(
                function(response){
                    $scope.dish = response.data;
                    $scope.showDish=true;
                },
                function(response) {
                    $scope.message = "Error: "+response.status + "
" + response.statusText;
                }
            );
```

- Also, update *IndexController* as follows:

```
                        $scope.dish = {};
                        $scope.showDish = false;
                        $scope.message="Loading ...";

                        menuFactory.getDish(0)
                        .then(
                            function(response){
                                $scope.dish = response.data;
                                $scope.showDish = true;
                            },
                            function(response) {
                                $scope.message = "Error: "+respons
e.status + " " + response.statusText;
                            }
                        );
```

- Then, update the *dishdetail.html* as follows:

```
        <div class="row row-content" ng-controller="DishDetailContr
oller">
            <div class="col-xs-12" ng-if="!showDish">
                <h3>{{message}}</h3>
            </div>
            <div class="col-xs-12" ng-if="showDish">
```

- Also update *home.html* as follows:

```
            <div class="col-xs-12 col-sm-9 col-sm-pull-3">
                <div ng-if="!showDish">
                    <h3>{{message}}</h3>
                </div>
                <div class="media" ng-if="showDish">
```

- Save all the changes and then have a look at the web page.

Conclusions

In this exercise, we extended the previous exercise to handle errors that result during access to the server.