# Ads – SPA with AngularJS Lab

The **goal of this lab** is to continue the development of the Online Ads AngularJS application: add **filtering by category and town** when displaying the public ads.

## Filtering Ads by Category and Town

1. Create a service **categoriesService** to retrieve the list of categories from the back-end REST service:

|  |
| --- |
| **adsServices.js** |
| …  ***app***.**factory**(**'categoriesService'**,  **function** ($resource, baseServiceUrl) {  **var** categoriesResource = $resource(  baseServiceUrl + **'/api/categories'** );   **return** {  getCategories: **function**(success, error) {  **return** categoriesResource.**query**(success, error);  }  }  } );  … |

5 score

1. Create a service **townsService** to retrieve the list of towns from the back-end.

5 score

1. Load the categories and towns in the **RightSidebarController**:

|  |
| --- |
| **RightSidebarController.js** |
| ***app***.*controller*(**'RightSidebarController'**,  **function** ($scope, $rootScope, categoriesService, townsService) {  $scope.**categories** = categoriesService.getCategories();  $scope.**towns** = townsService.getTowns();  } ); |

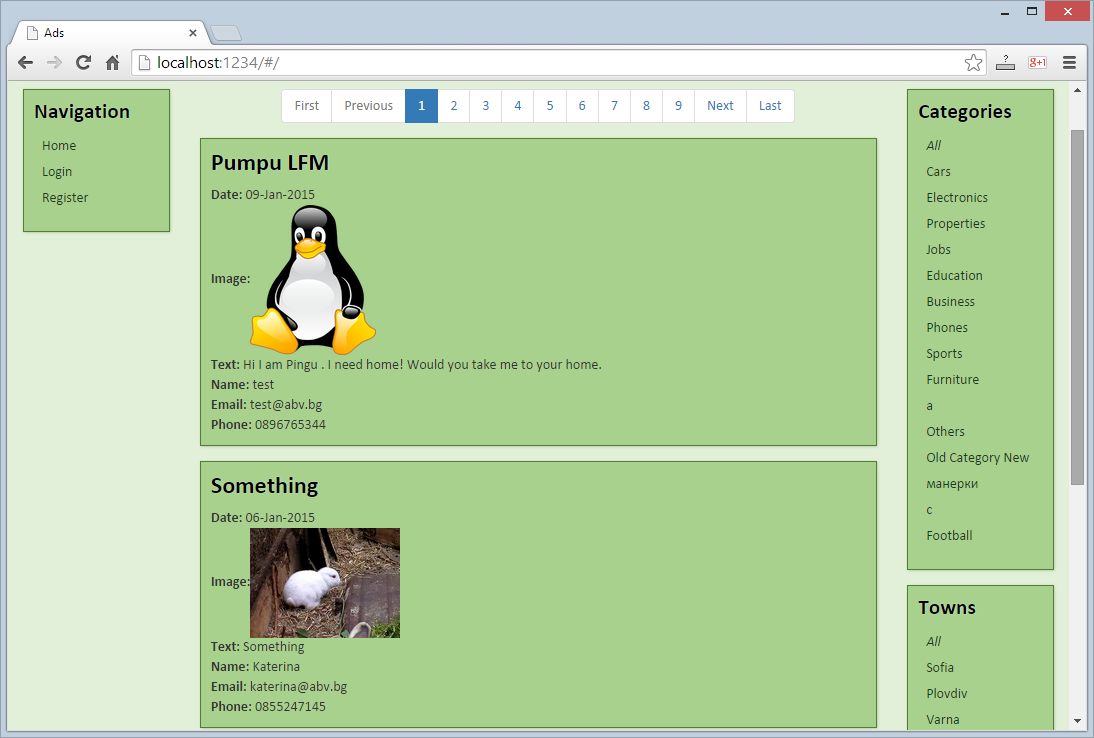
5 score

1. Write a HTML template (view) to **display the categories and towns**:

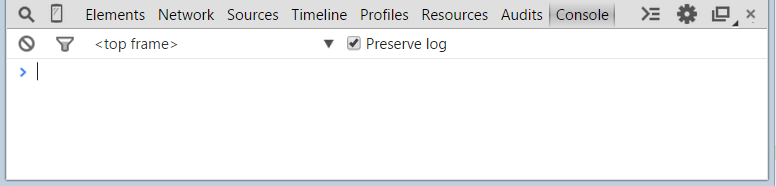
|  |
| --- |
| **right-sidebar.html** |
| <**div class="box"**>    <**h2**>Categories</**h2**>    <**ul class="sidebar-menu"**>        <**li class="special"**>All</**li**>        <**li ng-repeat="category in categories"**>  *<!--* ***TODO: display category name here...*** *-->*        </**li**>    </**ul**>  </**div**>  *<!--* ***TODO: implement the Towns filter box here (just like the categories box above)*** *-->* |

5 score

1. Test your application to ensure the categories and towns are loading correctly:



* + Check for errors at the Web browser's development console:



10 score

1. Make the categories and towns **selectable**. This is achieved in two steps:
   * Handle the "**click**" event to store the ID of the selected item in the **$scope** (or **null** if the filter is clear).
   * Assign a CSS class "**selected**" to the currently selected item in the view.

First, modify the **right-sidebar.html** view template:

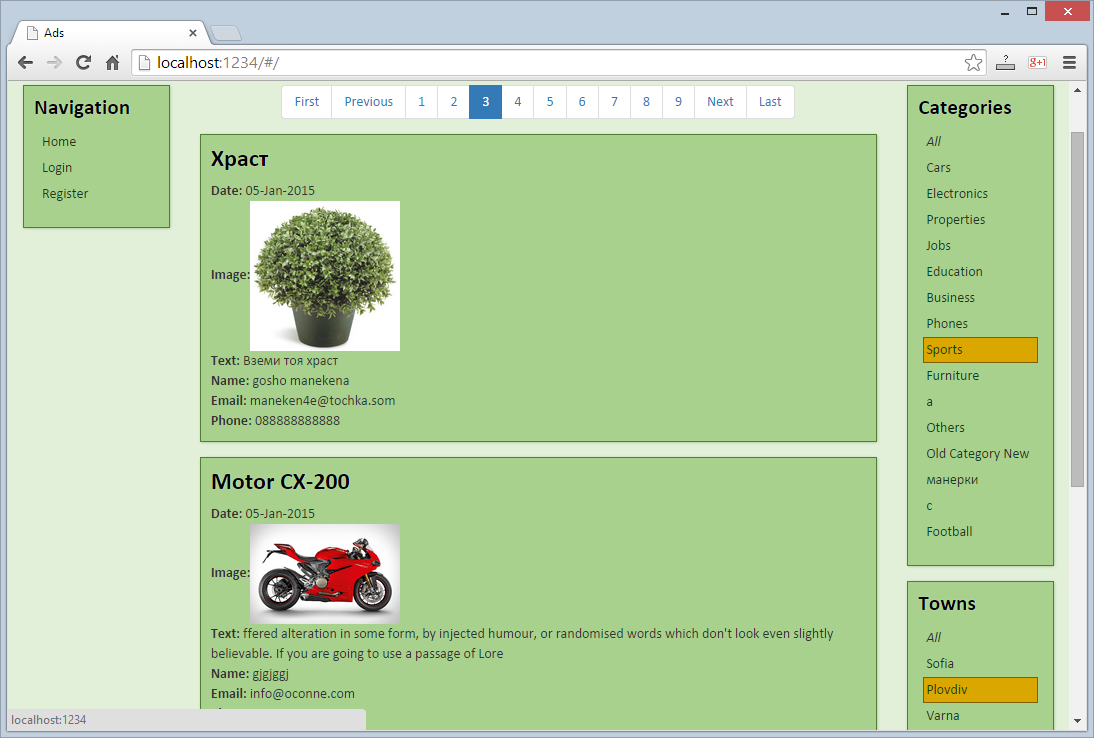
|  |
| --- |
| **right-sidebar.html** |
| **<div class="box">**  **<h2>Categories</h2>**  **<ul class="sidebar-menu">**  **<li class="special"**  **ng-class="{'selected' : selectedCategoryId == null}"**  **ng-click="categoryClicked(null)">**  **All**  **</li>**  **<li ng-repeat="category in categories"**  **ng-click="categoryClicked(category.id)"**  **ng-class="{'selected' : selectedCategoryId == category.id}">**  **{{category.name}}**  **</li>**  **</ul>**  **</div>**  *<!--* ***TODO: implement the Towns filter box here (like the Categories filter) ...*** *-->* |

Second, add click handlers in the **RightSidebarController**:

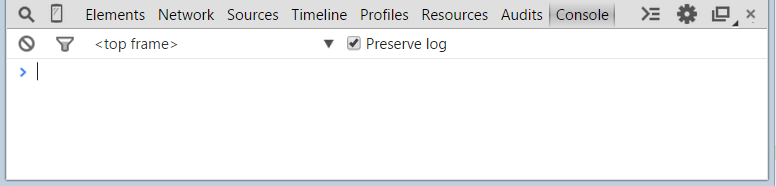
|  |
| --- |
| **RightSidebarController.js** |
| …  $scope.categoryClicked = **function**(clickedCategoryId) {  $scope.**selectedCategoryId** = clickedCategoryId; };  *//* ***TODO: implement the same logic for the "townClicked" event handler*** |

10 score

1. Test your application to ensure **the categories and towns are selectable**:



* + Check for errors at the Web browser's development console:



5 score

1. Filter ads by the selected category and town. This is achieved in two steps:
   * **Broadcast an event "categorySelectionChanged" / "townSelectionChanged"** through the **$rootScope** from the **RightSidebarController** when the category / town selection is changed.
   * **Handle the "categorySelectionChanged" / "townSelectionChanged"** **events** in the **HomeController** and filter the ads by invoking the REST service at the back-end with **categoryId** and **townId** parameters.

First, modify the **RightSidebarController** to broadcast the category and town selection change events:

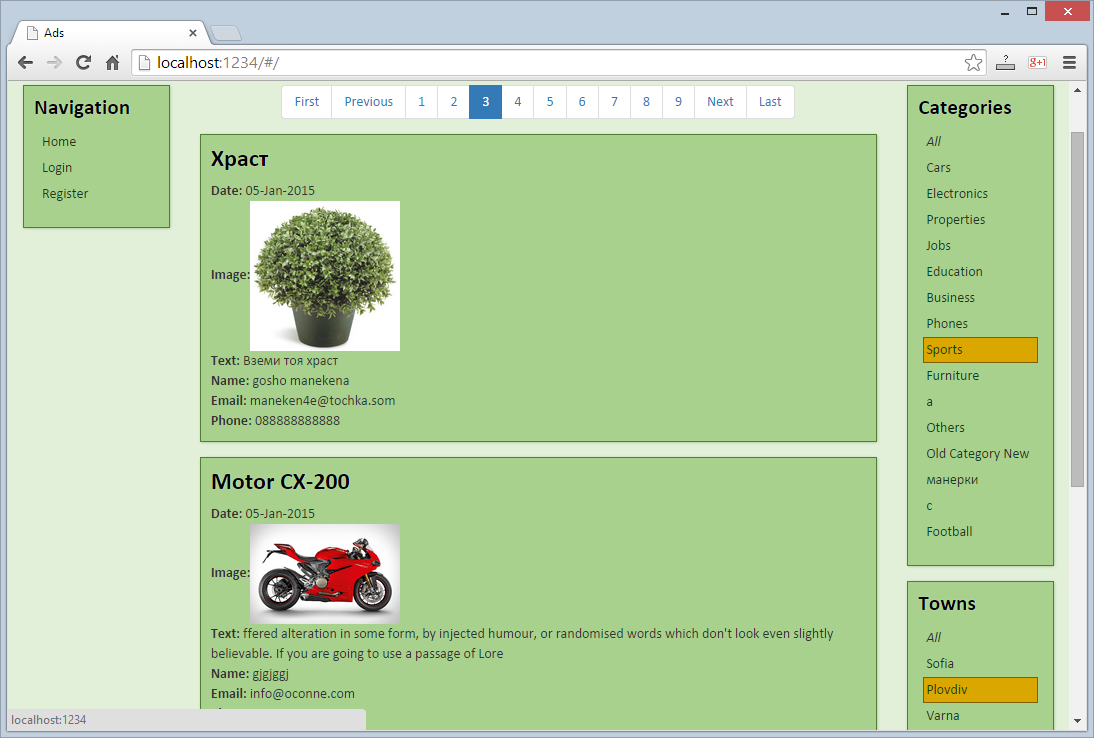
|  |
| --- |
| **RightSidebarController.js** |
| …  $scope.categoryClicked = **function**(clickedCategoryId) {  $scope.**selectedCategoryId** = clickedCategoryId;  $rootScope.$broadcast(**"categorySelectionChanged"**, clickedCategoryId); };  $scope.townClicked = **function**(clickedTownId) {  $scope.**selectedTownId** = clickedTownId;  $rootScope.$broadcast(**"townSelectionChanged"**, clickedTownId); }; |

Second, modify the **HomeController** to implement ads filtering by calling the REST service:

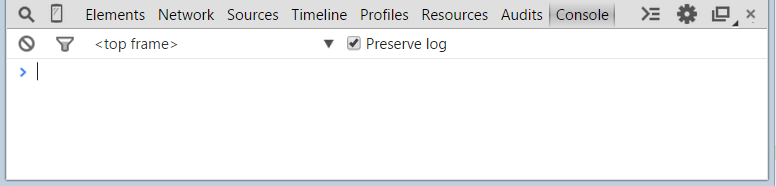
|  |
| --- |
| **HomeController.js** |
| …  *// This event is sent by RightSideBarController when the current category is changed* $scope.$on(**"categorySelectionChanged"**, **function**(event, selectedCategoryId) {  $scope.**adsParams**.**categoryId** = selectedCategoryId;  $scope.**adsParams**.**startPage** = 1;  $scope.reloadAds(); });  *// TODO: implement similar logic for filtering by category* |

15 score

1. Test your application to ensure **the filtering by category and town works correctly**:



* + Check for errors at the Web browser's development console:



5 score