

# Ads – SPA with AngularJS Lab

The **goal of this lab** is to continue the development of the Online Ads AngularJS application: add paging when displaying the public ads.

## Problem 3. Paging for the Public Ads

**Step 1.** Research how to retrieve a **specific page** of ads from the Ads REST service:

- You should perform a **HTTP GET** request from <http://softuni-ads.azurewebsites.net/api/ads>
- You should pass two parameters: **pageSize** and **startPage**
  - E.g. <http://softuni-ads.azurewebsites.net/api/ads?pageSize=2&startPage=3>
- The service will return the ads as JSON object: `{numItems: n, numPages: p, ads: [ad1, ad2, ...]}`.

**Step 2.** Find a UI control for paging in AngularJS – a **pager**.

- You may use the **UI Bootstrap for AngularJS**: <http://angular-ui.github.io/bootstrap/#/pagination>.
- You are free to implement the paging from scratch (this will cost you more time and effort).
- You may use any other pagination library / UI control (at your own risk).

**Step 3.** Include the Angular UI Bootstrap Pagination library to your project:

- The library source code is already added to your project skeleton:
  - The pagination AngularJS module: **lib/ui.bootstrap/ui.bootstrap.pagination-0.12.0.js**
  - The HTML template for rendering the pager: **templates/pagination/pagination.html**
- Include the pagination module JS file in your **index.html**.
- Include the **ui.bootstrap.pagination** module in your Angular app. Modify the **app.js** file:

app.js
... <pre>var app = angular.module('app', ['ngRoute', 'ngResource', 'ui.bootstrap.pagination']);</pre> ...

10 score

**Step 4.** Add a **pagination** UI control in in your home view (**templates/home.html**):

home.html
<pre>&lt;pagination total-items="ads.numItems" ng-model="adsParams.startPage"   items-per-page="adsParams.pageSize" ng-change="reloadAds()"   max-size="9" boundary-links="true"&gt; &lt;/pagination&gt;</pre>

- The pagination control will take the total number of items from the current scope (**ads.numItems**).

- The page size comes also from the scope (**adsParams.pageSize**). We will display 2 ads per page to simplify testing.
- The current page is data-bound to the scope variable **adsParams.startPage**.
- Then the user changes the current page (clicks the pager), the scope function **reloadAds()** will be called.

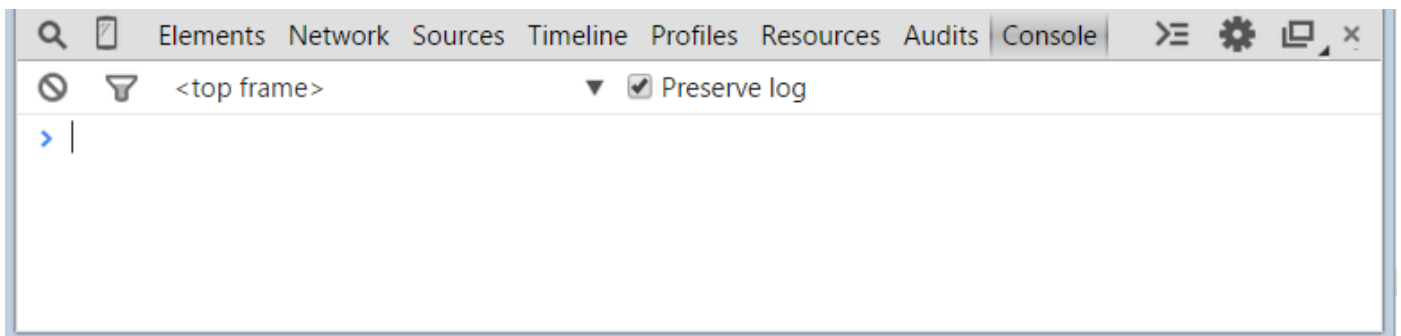
10 score

**Step 5.** Test your application to ensure the pager is loading correctly.

- The pager UI control should be shown at your home screen:



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



**Step 6.** Modify your **HomeController** to initially load the first page in the **\$scope** and reload the ads when the current page is changed:

```

HomeController.js

'use strict';

app.controller('HomeController',
function ($scope, adsService, notifyService, pageSize) {
    $scope.adsParams = {
        'startPage' : 1,
        'pageSize' : pageSize
    };
});

```

```

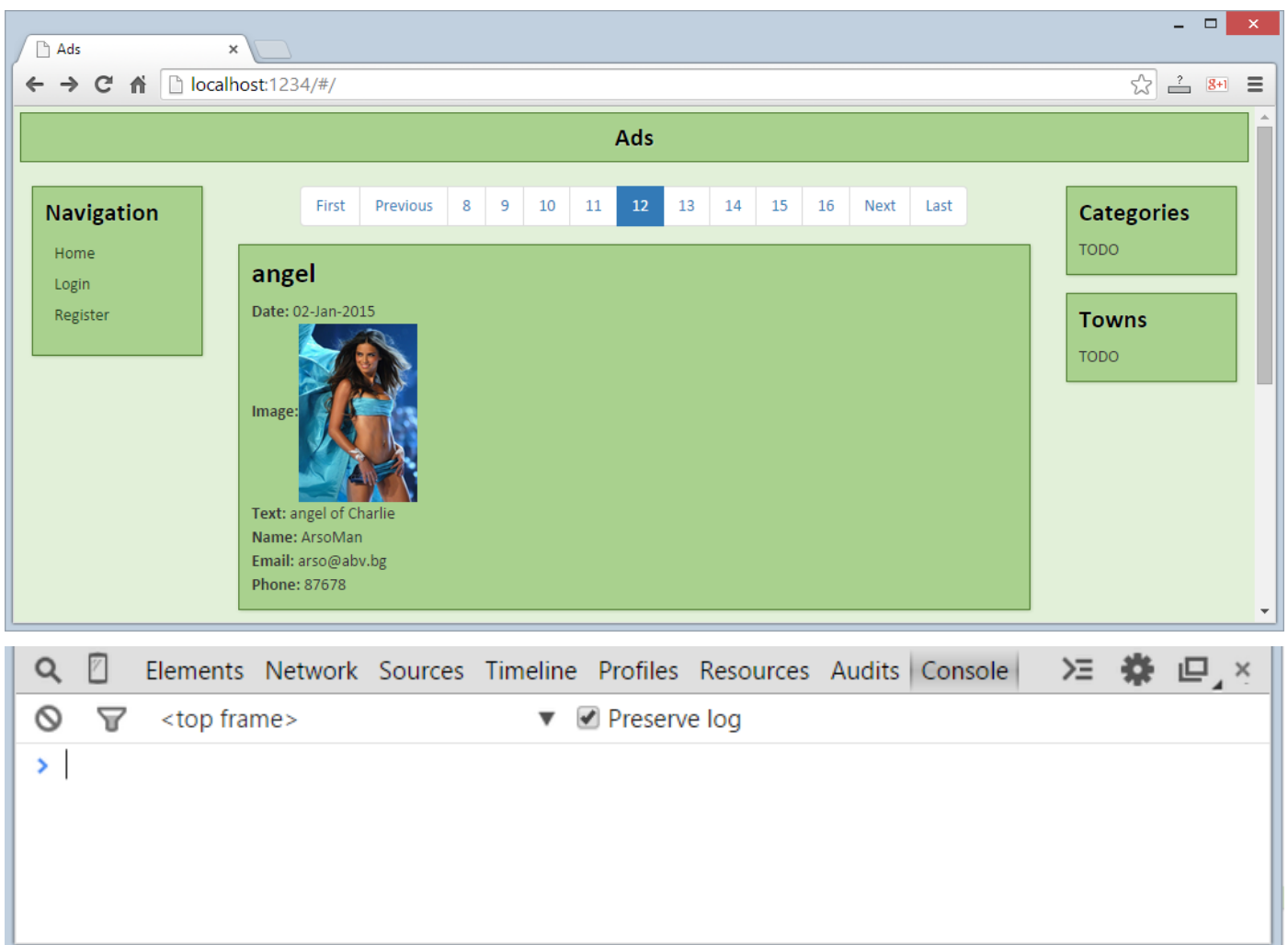
$scope.reloadAds = function() {
    adsService.getAds(
        $scope.adsParams,
        function success(data) {
            // TODO: put the ads in the scope
        },
        function error(err) {
            // TODO: display an error message
        }
    );
};

$scope.reloadAds();
});

```

15 score

**Step 7.** Start and test your application. Fix any bugs. Check the development console. It should be error free:



10 score