

Assignment: Module #2 - SPA Development and Deployment

Objective

By the end of this assignment, you will be able to:

Develop an AngularJS Application

- Create a Single Page Application (SPA) based on AngularJS (1.x)
- Package a SPA with necessary external javascript and CSS libraries
- Define a SPA to be the default page for a Rails server (either internal or thru a re-direct)

Provide Web-based Resource Access to a Resource Hosted on Rails

- Build a page that provides CRUD actions for a back-end resource
- Implement a service in AngularJS to interact with a server-side resource through a JSON API
- Implement an HTML form suitable for CRUD actions for a resource
- Implement a controller in AngularJS to supply a view-model to the page with data and callbacks for page events
- Encapsulate the management of a resource within an isolated scope

Deploy an AngularJS Application

- Make the SPA accessible to the user through a public Internet URL
- Provide web access to a resource served by a Rails server, from module 1, on the Internet
- Deploy a packaging of the SPA optimized for production

Requirements (150 min)

The time estimates below are based on use of the Asset Pipeline as the development option. It does not include the development of any external development environments if that path is chosen.

1. (5 mins) Select a single development option for the submission. It can be based on Rails Asset Pipeline or external development deployed either separately or within the Rails `/public` folder. Please keep this to a single option to reduce any confusion on the peer grader. You are free to implement the alternate approaches in a separate branch and deployed separately from your graded solution.
2. (10 min) Integrate vendor javascript and CSS libraries into your application to include, at a minimum, AngularJS, ngResource, Bootstrap, and JQuery. You may include more.
3. (120 min) Implement the web management of a back-end resource provided by your Rails API server. Choose either your Cities (ActiveModel) or States (MongoDB) resource. Implement access to only one of the resource types implemented in module 1.
 - Provide a list display of all resource instances available on the server.
 - Provide the capability to select any one of those instances through a mouse click.
 - Provide the capability to update and delete a selected instance on the back-end Rails server. One must be able to change the name of the resource (e.g., `city.name` or `state.name`)
 - Provide the capability to create a new instance on the back-end Rails server. One must be able to specify the name of the resource (e.g., `city.name` or `state.name`)
 - Integrate all CRUD (create, read/display, update, and delete) actions into a single form. Button(s) for instance actions should only be visible for existing instances with an ID. Creation button(s) should only be visible for new instances (i.e., without an ID)
 - Encapsulate the HTML required to implement the form using an Angular directive or component (not covered in module #2).
 - Implement all interaction with the back-end server API using ngResource's `$resource` service.
 - Implement a controller that exposes a view-model on the page's scope with data containing a list of instances and the current instance
4. (0 mins) Optimize your assets by your application server and not obtained through a CDN
 - All application and vendor javascript deployed from the application server are to be concatenated into a single file and minimized.

- All application and vendor CSS deployed from the application server are to be concatenated into a single file and minimized.
5. (15 min) Configure the Root URL (/) of your Rails server to either host or re-direct the caller to your SPA application.

Grading

1. Supply the base, public URL of your Rails server that hosts your solution for this assignment.
 - (5 pts) Did the submission provide you a base URL (e.g., ...herokuapp.com/) and not the URL to a specific web page (e.g., .../public/client/index.html)?
 - (5 pts) Does the root URL direct you to a page with the SPA? This can either be the root URL of the Rails server or thru a re-direct (even to a different site). It only matters where you start from and not where it takes you.
 - (5 pts) Does the source of the page provide a single link for all CSS served by that server (i.e., excluding CDN sources)?
 - (5 pts) Is the CSS source minimized (i.e., does it have unnecessary comments and new line formatting removed)?
 - (5 pts) Does the source of the page provide a single script element to download all javascript served by that server (i.e., excluding CDN sources)?
 - (5 pts) Is the javascript source minimized (i.e., does it have unnecessary comments and new line formatting removed)?
 - (5 pts) Can you list existing resource instances – either Cities or States?
 - (5 pts) Can you create a new resource instance? (i.e., after creating the instance – refresh the page and the new instance should still be in the list)
 - (5 pts) Can you update an existing resource instance (i.e., after making a change name – refresh the page and the new instance should still be in the list)?
 - (5 pts) Can you delete an existing resource instance (i.e., after deleting the instance – refresh the page and verify the instance does not come back)?
 - (5 pts) Were you only shown a **Delete** or **Update** button for an existing instance?
 - (5 pts) Were you not shown a **Create** button for an existing instance?
2. Supply the URL of a branch or tag in your public Git repository that represents the version of source code deployed for the SPA portion of your solution. This may be the same URL as the Rails version if you are using a single source tree.
 - (5 pts) Is the application based on AngularJS 1.x?
 - (5 pts) Was ngResource's **\$resource** used to communicate with the back-end API?
 - (10 pts) Was an Angular service (factory, service, or provider) used to configure the **\$resource** for use with the back-end Rails API?
 - (10 pts) Was an Angular controller used to expose a view-model of data and callbacks to the HTML page?
 - (5 pts) Was an Angular directive or component used to specify the HTML template and controller?
3. Supply the URL of a branch or tag in your public Git repository that represents the version of source code deployed for the Rails portion of your solution. This may be the same URL as the SPA version if you are using a single source tree.
 - (5 pts) Is **config/routes.rb** or a controller configured to take the caller of the root URI (/) to the SPA application (either thru implementation or thru re-direct)?

Version: 2016-12-19