**Sanborn Code Test Instructions**

**Overview**

The Sanborn Code test is representative of the type of Rails apps you’ll work with on a regular basis. Knowing how to plot relevant/context sensitive information on a map via latitude and longitude is essential for success at Sanborn.

Our layouts are rendered server side via the Rails asset pipeline, and map data is generally served via AJAX. Efficient database queries are essential to map performance, so you’ll want to be sure to keep an eye out on response times. For maps with heavy data anything below 4000ms is considered good. For this code test target a response time of 1000ms or less.

**Objectives**

**The following characteristics are being reviewed**:

* Ability to follow directions and extrapolate important details from the mockup and general instruction (this document).
* General proficiency of the Rails framework
* Ability to configure the rails framework to meet project requirements (using Rails 6+ without web packer)
* Understanding of Rails erb and helpers for server-side rendering
* Understanding of AJAX and rendering JSON.
* JavaScript & CSS skills
* Ability to find what you need from existing 3rd party documentation (Leaflet)
* Understanding of database relationships between tables (index your relationships!)
* Aesthetic ability (we build client facing apps - aesthetics matter)
* Working with GeoJSON
* Workig with GIT
* Rspec testing – add tests!

We do understand that working with leaflet and maps may be new to you – that’s OK. This test may take you a few hours or a couple days. We are not measuring speed at with this test – we are measuring quality, understanding and the ability to “figure things out”.

Do your best. Google and Stack Overflow are your friends. There are a lot of tutorials and examples online that will help you.

**Project Details: State Facts!**

You’re going to create a simple discovery app called “State Facts”. This app will list fun facts for the selected state. You will create data for **2 states** – I’ve included the Colorado data below. Select any other state and fun facts of your choice.

Useful Resources:

<https://www.uncovercolorado.com/fun-facts-about-colorado/>

<https://www.latlong.net/>

Graphical user interface, application, map

Description automatically generated

**Create a Rails app with the following stack**

Ruby 2.6.6 – server side rendering

Rails 6.1.3.2 - without webpacker

Posgresql – save username and password in rails credentials

Puma 5 – configure to deploy on Heroku (hobby tier)

Bootstrap (any version)

Fontawesome (any version)

Leaflet (latest version)

Jquery (latest version)

JavaScript

Sass or Less (your choice – add to gemfile)

Rspec

In the provided project folder, I’ve included some assets and the Gemfie to help you get started. Feel free to add any additional gems you’ll need. You will need to add gems for your styling framework (sass or less). Refer to the mockup for the map layout. I’ve included a basic JavaScript file for the map nav controls to get you started.

**Database setup**

create your own seed file to populate the database with any 2 states and their fun facts.

You can look up the locations for cities and places here to get the latitude and longitude for the fact location: <https://www.latlong.net/>

Database tables (2)

**State (has\_many :facts, dependent: :destroy)**

name: string

description: text

flag\_image: string

capitol\_name: string

population: integer

**Fact (belongs\_to :state)**

title: string

details: text

lat: float

lng: float

**User Story**

When the map initially loads it will render a GeoJSON of the State(s) that have fun facts (only 2) – shown as light green area in mockup.

We’ve provided a GeoJSON of the US States in your folder:

US\_states.geojson

Here is some useful information about working with GeoJSONs in leaflet:

<https://leafletjs.com>

<https://leafletjs.com/reference-1.7.1.html#geojson>

When the user selects a State from the States dropdown list – it will render the general State details from the database via AJAX (as shown in the mockup):

Name (:name)

Description (:description)

Flag (:flag\_image)

Capitol Name (:capitol)

Population size (:population)

State Facts (:title) – should be clickable

When a State Fact is clicked it will put a **Leaflet Marker** on the map at the latitude and longitude (:lat, :lng) position of the Fact.

<https://leafletjs.com/reference-1.7.1.html#marker>

When the marker is clicked it will show a popup with the fact details

[https://leafletjs.com/reference-1.7.1.html#popup](https://leafletjs.com/reference-1.7.1.html#marker)

If a new state (or the same state) is selected from the drop down the map should clear all popups and markers and load the new information.

**Submitting Project:**

Deploy to Heroku: <your last name>\_state\_facts.herokuapp.com

Send email to: [jcopple@sanborn.com](mailto:jcopple@sanborn.com)

Include link to github and link to Heroku app

**Have questions?** email: coros@sanborn.com