Technical Test - Build an application

Overview

- Create a web application that will extract large amounts of data from the Google search results page. You need to find a way to work around the limitations of mass-searching keywords as Google prevents it.
- Store this data and both display and report back this data in various ways
- Users should be authenticated to use the application
- The application should provide Oauth 2 Restful JSON API (think as if we are going to build a mobile application)

Specifics

1) Authenticated users can upload a CSV file of keywords. This upload file can be in any size from 1 to 1,000 keywords
Create a user interface and an API endpoint to perform this action.

2) The uploaded file contains keywords, each of those keywords will be used to search on http://www.google.com and they will start to run as soon as they're uploaded.

3) Each search result / keyword result page on Google you will store the following information on the first page of results:

   a. Number of AdWords advertisers in the top position.
   b. Number of AdWords advertisers in the right side position.
   c. Total number of AdWords advertisers on the page.
   d. Display URLs (in green) of the AdWords advertisers in the top position.
   e. Display URLs (in green) of the AdWords advertisers in the right side position.
   f. Number of the non-AdWords results on the page.
   g. URLs of the non-AdWords results on the page.
   h. Total number of links (all of them) on the page
   i. Total of search results for this keywords e.g. About *21,600,000* results (0.42 seconds)
   j. HTML code of the page/cache of the page.

4) Create a report from the stored information. This report will give back the details of each of the items stored in the database, sorted by keyword.
This report should also be stored in the database.
Build a user interface and an API endpoint to view past reports.

5) Allow users to query the stored data. Build a user interface and an API endpoint for users to perform queries.

Examples of the SQL queries which users can run:

    a. How many URLs contain the word "technology" in AdWords
    b. How many times a specific URL shows up in SEO.
    c. How many keywords have URLs in SEO with 2+ /'s or 1+ >

Feel free to propose other types of queries which would be useful for users.

Technical requirements

1. Use Ruby on Rails (4.x.x)
2. Use either MySQL or PostgreSQL
3. For the interface, a front-end frameworks such as Twitter Bootstrap, Bourbon.io or Foundation can be used. Use SASS as the css preprocessor.
   Extra points will be provided to the neatness of the frontend.
4. Use Git during the development process (bitbucket or github will make it work)
5. Deploy the application to Heroku (optional but a big plus)

Submit your finished application with the repository and Heroku (optional) links.