Restaurants’ Info Point

What are the benefits and risks for users and data providers based on the text technology methods you use in your project?

Seemab Hassan

University of Stuttgart

Text Technology

Dr. Kerstin Jung

Contents

1 Introduction

The 'Restaurants' Info Point' is a project that helps user find restaurants of their choice based on their desired filters. A user can look for a restaurant based on its proximity to their address by filtering through the states and postal codes. Similarly, if a user is planning for a fine dine and want to treat someone with the best possible experience, they can look through a list of top rated restaurants. In addition to that, a person can also view the price range and the availability of takeout to further help themselves land with a restaurant of their need. To accomplish the task, the project consists of three phases:

1.1 Collect

Yelp's data-set for business (restaurants) and their reviews, which is available at Kaggle, in JSON format was cleaned using Python and a filtered subset was used to create CSV files using pandas which served as the real data for this project.

1.2 Prepare

The data was then prepared for the user to be able to find the suitable business by storing it in a relational database using SQLite3. Two tables 'business' and 'reviews' were created. Once the database connection was created and the data was stored, SQL queries were conducted on data to extract out the results for above mentioned purposes. These results were stored in an XML format and the XML file was checked against XML grammar to eliminate the possibility of errors.

1.3 Access

XML is portable and hardware independent therefore, an XML file was created out of the SQL results and this file was used to help the user access the data in the form of a web-page. HTML pages were created with AJAX (as an extension to the learned concepts in this course). The users were given a choice to view restaurants according to their choice of address, rating and price range/takeout.

2 Benefits and risks for data providers

2.1 Benefits for data providers

2.1.1 SQLite3

Using SQLite3 and having an SQLite database has several advantages, like portability, reliability, reduced application cost, and better performance. Reading and writing from an SQLite database is often faster than reading and writing individual files from disk. SQLite does not require a separate software to be downloaded. One can get right to creating databases, after downloading SQLite libraries in the computer. Content can be accessed and updated using concise SQL queries instead of lengthy and error-prone procedural routines. Also, SQLite database is accessible through a wide variety of third-party tools. In the event of data being lost, SQLite database's content is more likely to be recoverable. [7]

2.1.2 XML

The motivation for using XML, as mentioned above is that it is portable and independent of hardware. There are automated ways to write XML i.e., using FOR XML keyword with SELECT queries. Four modes (RAW, AUTO, EXPLICIT, PATH) can be used. This can make work much easier and less error prone for a practical use but for learning purposes, we are writing the XML structure manually. [8] XML files can be verified using an XML schema which reduces the chances of any errors.

On top of this, XML is a very organised technology keeping data and presentation separated, that means it does not carry any information about how to be displayed. Hence, the same XML data can be used in various different presentation scenarios. Also, since the project's output display tool is HTML, XML was the go-to option, as it complements HTML. XML takes care of the storage and transport of data, while HTML formats the said data. Also, when the data changes, one should not have to edit the HTML file to display it. XML allows data to be stored in separate XML files, and with a few lines of JavaScript code, one can read the XML file and update the changes to the HTML content accordingly. [6]

2.1.3 HTML

HTML is a very basic and popular technology. Not only is it easy to understand and work on, it is accessible through any device, so output can be tested on-the-go. Also, since it's light-weight, it's fast to load, saving a good amount of time over vast iterations. Additionally, HTML is free and comes on computers by default, so one would not have to invest in an additional software, which might get expensive at times. [2]

2.1.4 AJAX

AJAX is a good extension to access XML via HTML, as it complements a relatively "new" approach and uses a number of existing technologies together including: HTML or XHTML, JavaScript, DOM, XML/JSON, XSLT, XMLHttpRequest. [4] Data transfer is faster, as it can be sent and retrieved without having to reload the current page i.e., asynchronous interactions with invisible data retrieval. [5] Hence, this makes the application more responsive i.e., gets a quicker response time by reducing the server traffic in both side request. [3]

2.2 Risks for data providers

2.2.1 SQLite3

Although the usage of SQLite3 is a good fit for the scope of the project, there are a few limitations to be noted as well. As SQLite is a light-weight query technology, it is used to handle low to medium traffic HTTP requests, so heavy data might not suit it well. Database size is restricted to 2GB in most cases. [7]

2.2.2 HTML

There are also some limitations and risks with HTML. Since the technology is too basic, small tasks might require longer codes, hence making it time-consuming for big changes. It also isn't quite fit if you're looking into making dynamic pages, since it is a static language. Security features with HTML are also moderately limited. [2]

2.2.3 XML

XML does have some limitations as well. Writing XML-file manually lead to problems while dealing with special characters. Checking against grammar in this method is extremely important because the method followed is error prone. This makes it a 'test-driven' approach. Also, XML syntax is redundant as compared to binary representation of similar data, especially tabular data. This may result in affecting application efficiency through higher storage, transmission and processing costs. XML syntax is also verbose for human readers, as compared to other 'text-based' data transmission formats. [6]

2.2.4 AJAX

A few short-comings of AJAX are that it is open-source, and view source is allowed, and anyone can view the code source written for Ajax, which makes it less secure compared to other technologies. Also, search engines would not be able to index an AJAX application. Another limitation is XMLHttpRequest object itself. For security reasons, one can only access information from the web host that serves initial pages. If information is needed to be fetched from another server, it's is not possible within the AJAX. [3]

3 Benefits and risks for users

3.1 Benefits for users

3.1.1 HTML

Since HTML is a fast and light-weight technology, users would save a lot of time on loading the pages. It's also would not require them to get any extra tool to access it, as it comes by default in every window. Also, if the user is looking for a basic and user friendly web page, HTML would serve that purpose. [2]

3.1.2 AJAX

For the user, the data would be organized and view-able on a single click i.e., a better user interface. It also comes with all modern browsers, and the process is fast as the technology entertains reduced server traffic. [1]

3.2 Risks for users

3.2.1 HTML

HTML would not give the user a fancy, eye-catchy or dynamic website for the most part, so if the end-goal is such a website, HTML would not be the most ideal solution. [2]

3.2.2 AJAX

AJAX might present some limitations for the user, such as disabled or no JavaScript/XMLHTTPRequest support may lead to problems with AJAX dependant pages. Furthermore, one failed request may lead to the failure of loading of the whole page. [5]

References

[1] *Advantages and disadvantages of an ajax based client application*. URL: https://www.slideshare.net/salarualin/advantages-and-disadvantages-of-an-ajax-based-client-application.

[2] *Advantages and Disadvantages of HTML*. URL: https://www.geeksforgeeks.org/advantages-and-disadvanatges-of-html/ (visited on 02/01/2022).

[3] *Advantages or Disadvantages of AJAX*. URL: https://way2tutorial.com/ajax/ajax-advantages-and-disadvantages.php.

[4] *AJAX Introduction*. URL: https://www.w3schools.com/js/js_ajax_intro.asp.

[5] Ayush Gupta. *The Pros and Cons of Using Ajax*. URL: https://www.tutorialspoint.com/the-pros-and-cons-of-using-ajax (visited on 11/27/2019).

[6] *How Can XML be Used?* URL: https://www.w3schools.com/xml/xml_usedfor.asp.

[7] *Java T Point*. URL: https://www.javatpoint.com/sqlite-advantages-and-disadvantages.

[8] Jignesh Raiyani. *SQLShack*. URL: https://www.sqlshack.com/generate-xml-forms-or-xml-data-type-documents-in-sql-server/ (visited on 01/27/2020).