Technical Report

This article discusses and analyzes the initial steps in the implementation of the BEM – a beverage manager – which is a web-based application that will allow people to search for non-alcoholic beverages and mocktails they can prepare and serve at events and possible restrictions that are associated with the said beverages. The program itself will also allow users to log their data and preferences in so that they can view prices/availability/comment/reviews/recipes.

Implementation:

Initial idea on the development of this program is to use C# for the development:

* ASP.NET API - It will handle the authentication service
* Using the <http://dotnetrdf.org/> library for managing our RDF.
* OpenLink Virtuoso - We can execute through an ADO.NET connection from C#, through the Virtuoso ADO.NET Provider, by turning it into SPARQL-in-SQL (SPASQL).

Another option since some of us are JAVA developers is to use Spring:

* A main service using OpenLink Virtuoso – though not necessary since
* Spring Boot - authentication service
* Spring Boot with Apache Jena adaptor for SPARQL for inference engine.

As such these elements will be our main connection between front-end and back-end. The program will be hosted on AWS using a virtual machine along with the services and node.js application.

Initial Analysis:

The web-apps purpose is to actually help people make an educated decision to drinks they can buy or make and serve. As such we devised a list of things we want to include as a checklist. In the final website design, we would like to have the elements outlined in the image below in the form of dropdowns so that users can help limit their searches to a certain degree:

* Location – Countries have access to different fruit/beverages so this is important.
* Ingredients – Serves as a link for refining whether someone wants a tea/ a soft drink/ a juice etc…
* Restrictions – This dropdown excludes certain drinks from the search ( ex. Sugary drinks)
* Date – This uses a calendar to select and offers drink based on seasons and/or holidays.

Though the list can be expanded – it is a good starting point for the ontology design. So we decided to create a small number of ontology designs using WebVOWL Editor, afterwards switching over to <https://protege.stanford.edu/products.php> - which is a bit better. Virtuoso handles SPARQL queries well enough and we can load ontologies into it, but we are looking at an open-source alternative at the moment. Information regarding suggested mocktails will use imbedded images to share recipes and these along with user information will be stored in a persistent SQL database, while drink information queries will be directed to Virtuoso.

Midterm Evaluation – Preliminary Consideration about the project:

Scenario 1: User searches for drinks with lime – non-registered

Scenario 2: User saves his preferences - registered – high sugar, milk

Scenario 3: User registers on website

Propose an ontological model – possibly, by reusing (parts of) existing ontologies – regarding the most common non-alcoholic beverages. Also, create the knowledge graph including the (groups of) users' preferences about a certain set of beverages like coffee, tea, milk, juice, soft drink, mocktail, plus their usages based on context – for example, a festive event, age, health restrictions (diabetes, lactose intolerance), location, season. Additionally, specify useful (meta-)data: category, price, ingredients, images, perishability, availability, offers, reviews, recipes, etc. Starting with this conceptual specification, a Web application will be developed in order to manage the collected knowledge and providing "smart" browsing and filtering, plus suggestions and correlations. Explore also [BEVON: Beverage Ontology](http://rdfs.co/bevon/), [FoodOn](https://foodon.org/), [Beverages @ Open Food Facts](https://world.openfoodfacts.org/category/beverages), and [Product concept @ Schema.org](https://schema.org/Product).