

ExcelToData

Team B4 - EXCELSior

formed by:

Alonzo Murrieta

Bryce Stulman

Jaime Perez

Matan Shtepel

Shriniketh Narayana

GitHub:

<https://github.com/LogicCoding/EXCELSior-ExceltoData>

Motivation/Concept:

We have chosen the 6th project (Excel-to-Database Interface) from the proposed list of project ideas provided by our professor, which are relevant to the open-source project openCAESAR. The idea is that Excel, which is used by a lot of people around the world, can be used as a sort of database, albeit a lightweight one. This is because each Excel sheet is essentially a table with rows and columns, which in a way can represent instances and attributes, respectively. The problem that arises (and the reason why an excel-to-database interface is needed) is that users can not really make complex queries on multiple sheets to get information, which necessarily negates Excel as a database.

As such, we will create a web application that will allow a user to take an Excel sheet and essentially be able to send it to a Fuseki graph database as a database itself, in order to then be able to query the Excel sheet (now in RDF format) for whatever information they want. The web app will then be able to export an Excel sheet from the database and allow the user to edit it. In general, the idea would be to let the user specify a mapping that matches the dataset schema used by the desired database, but we will be focusing on the RDF format to use Fuseki (our database server) and query the database using SPARQL.

This is all per the specs given to us by professor Maged Elaasar.

Link to Epics:

<https://github.com/LogicCoding/EXCELSior-ExceltoData/issues/1>
<https://github.com/LogicCoding/EXCELSior-ExceltoData/issues/2>
<https://github.com/LogicCoding/EXCELSior-ExceltoData/issues/3>

Technology Stack:

We intend to use mostly standard software tools to build this web application with the Node.js JavaScript environment using the Express web framework and the React Javascript library for user interface. We chose these tools because they are very commonly used, supporting large amounts of documentation and boasting many libraries that can be easily accessed using the Node Package Manager. Additionally, we have some experience with these technologies and have found them to be relatively simple and intuitive to use.

Node.js is a JavaScript runtime environment, providing a fully asynchronous event loop to handle requests, while Express is a fully asynchronous framework that provides easy ways to build and manage servers and routes for the application backend. React is a frontend focused JavaScript library designed for quickly and efficiently implementing user interfaces by building reusable components. As such, this application will be written primarily in JavaScript.

While Express, React, and Node.js make up most of the popular MERN stack for web development, our application will not use the standard MySQL relational database or MongoDB document-focused database because it is intended specifically to act as interface for databases that use Resource Description Framework (RDF) and the SPARQL specifications for storing

RDF information as a graph and making queries. Instead, we will use the Apache Jena Fuseki database system as it meets those specifications and is accessible with a well documented API.

While we initially intend to host this application locally, we may explore options for hosting online. Heroku has removed its free hosting plans as of late last year, but there may be options to use student credits. Alternatively, other similar hosting sites such as Render or GitHub Pages may be options for hosting, but further research would be required.

Feasibility:

For this project, we'll need to implement the front end of the web app, and implement the back end of the app which performs the conversion from excel to graph database.

Strengths:

Our team is familiar with web app tech stack

Weaknesses:

Our team is not familiar with graph databases or RDF

Since our team has plenty of experience making web apps and with the tech stack for making web apps, we'll have no problem making the backbones of our application. Most of our trouble will likely be in implementing the specific logic behind the conversion from excel to database in the backend. We may need to work with new frameworks or tools in order to convert the excel file to the form of the database. This is unlikely to be a huge issue in terms of time, as we can devote all of our learning toward this in particular since we have experience in the other aspects this project will require.

The costs of the project are unlikely to be prohibitive, as there are cheap web hosting solutions available. There aren't so many features to this project that we will have trouble fulfilling all requirements by the end of 7 weeks. There is a small set of core features that we will be able to complete swiftly once we've completed the sufficient research.

In terms of originality, even excel allows users to export their spreadsheets to a SQL database, but there are limited projects allowing spreadsheets to be converted to graph databases and then queried. Our project provides a novel function that is sure to be appreciated by those working extensively with spreadsheets.

Capability:

Given that the team consists of five members with varying skill sets and knowledge with Jena-Fuseki, all team members are expected to gain individual understanding of the full technology stack in the project. Front-end and Back-end development will be split between group members.

For front-end development, Alonzo and Shriniketh will spearhead most of the app's interface and design. Alonzo has worked on assisting development of the UCLA CEC website with React and JavaScript. Shriniketh has experience constructing various full stack projects

with React, Node, and Express. Since the front-end will not be developed until later on, front-end developers will spend their time assisting the back-end as well as perform auxiliary research on Fuseki and ERN stack.

Back-end development will be led by Bryce, Jaime, and Matan. Bryce has experience in creating applications using the MERN stack, as well as experience with databases through his job where he primarily uses JS/jQuery. Though Jaime does have experience in Python, he plans to further research the ERN stack as well as take over a bulk of the administrative duties. Matan has worked with the ERN stack and has knowledge of graph databases beyond Fuseki.

Fuseki remains to be the main technology component that all group members will have to learn and familiarize themselves with. Still, given our expansive set of skills, the team feels very confident in the abilities to develop this project.