CSE 5320: SPECIAL TOPICS IN SOFTWARE ENGINEERING

ENTERPRISE SOFTWARE DEVELOPMENT WITH INDUSTRY PRACTICES

A diverse data set, consisting of several files in various formats, from different government agencies were provided. Several of the data fields were redundant and did not have a particular single key unique ID. Our objective was to create and automate the data manipulation and gather the data into one single enterprise model from which the needed data can be reported efficiently through a single fetch request. The project was divided into four phases.

Project setup:

Software & framework used.

Interpreter	Python 3.4.2
Version Control	GitHub & Source Tree (a GUI that allows to manage the GitHub account without using inline commands)
Data Base	MongoDB V2.6.4, Robomongo (a MongoDB management tool)
API Framework	Flask v0.10 which supports RESTFUL web services

Phase 0:

Building a good data model is the key for a good application. So in this phase, after we received the data sets from different government agencies, the most important step was to analyze the data according to its various file formats, names and also the contents so as to determine that how could can be related to each other. Once the data was thoroughly analyzed we created the Entity Relationship Diagram (ERD).

Phase 1:

The files that were provided were in a csv format. So the next step was to write python scripts that could efficiently Extract, Transform and Load (ETL) the various CSV files according to the ERD. The CSV files were converted into JavaScript Object Notation (JSON) format objects that could be loaded into the MongoDB Database. All the team members were given common credentials to access the database. The data in the MongoDB was managed by the RoboMongo tool. The main advantage of using MongoDB database was that the data sets provided were huge and the data can be updated at regular intervals. So MongoDB is easier to scale. It does not have any complex joins. The in-built dumps function of JSON a module was used for achieving this task.

Phase 2:

An XML file was also provided to us. This XML file was also parsed to convert it into the JSON object format. ElementTree module of python was done to achieve this. The Element type is a flexible container object, designed to store hierarchical data structures in memory. The type can be described as a cross between a list and a dictionary. Later in the phase, we were required to insert all information in the data sets into the MongoDB database using python scripts. Since we had information from all files in JSON objects which indeed were written inside JSON files, we used the in-built loads function to insert each JSON object inside the data base.

Phase 3:

The Final phase included constructing a web service in which accepts drug name as the end user input and retrieves all related information from a database. We have developed an Application Programming Interface to achieve this object. The advantage of an API is that, direct access of data base by end user can be avoided. An URL of a web service can be provided If a particular stake holder requires access to certain resources, instead of giving direct access to the stake holder. Our API successfully retrieved all the related data of a particular drug which is given as an input from the three different data sets.