Software Requirements Specification

for

LA County / UNSPSC Crosswalk

Version 2.0 approved

Prepared by Daniel Geiser, Jesse Rodriguez, Norin Chea, Jordi Alejandre, Adryel Arizaga

Los Angeles County, Internal Services Department:

Trinh Mac, Jon Neill, Mark Martinez, Tiffany Lee

May 2, 2020

Table of Contents

Revision History

Name	Date	Reason For Changes	Version
First Draft	04 / 20 / 2020	First Document for Review	1.0
Second Draft	05 / 01 / 2020	Second Document Review	1.1
Final Draft	05 / 07 / 2020	Final Revision	2.0

1. Introduction

The Introduction section of the Software Requirements Specifications (SRS) provides a general description of the software project named Crosswalk for Los Angeles County Internal Services Department (ISD).

The overall assumption of the software will create and develop a method as a group to translate (i.e., “Crosswalk”) the commodity standard used by the County of Los Angeles with an existing commodity standard used in the procurement industry space.

This is the only authorized source of the “Crosswalk” software requirements. The document’s first version is the one being reviewed and approved. The second version will be the one including all the initial inputs from the reviewers and the one used to design and implement (code) the software. The third and final version of the “Crosswalk” SRS will include all the modifications approved after the SDD, if any.The requirements will be listed using tables with entries for each of the modules initially understood to be part of “Crosswalk”. These tables may be modified accordingly after the SDD has been generated and approved. Lastly, there may be requirements that will be better understood at coding time (implementation) and these will be the last modifications to the requirements table before software full implementation and delivery to its customers.

1.1 Purpose

The purpose of this SRS is to serve as the main source of requirements for the “Crosswalk” by LA County, ISD

1.2 Intended Audience and Reading Suggestions

Other programmers who will be continuing our work. This ‘Crosswalk’ SRS is intended for developers that will write the corresponding SDD. It is also intended for the software implementation programmers that will continue to work on our ‘Crosswalk’ project

Audiences to this document include reviewers who will be accepting and approving this software document.

1.3 Product Scope

The software is identified as “Crosswalk”, it Creates an automated, AI powered “CrossWalk” between different commodity standards for Los Angeles County. The program/application will be able to translate a given code, description, class, etc. from the standard used by Los Angeles County to the UNSPEC standard. The program shall return any and all matches from a code/description in a standard to another and give a confidence level of the accuracy of the matches.

The software will make an excel sheet with the top five Los Angeles County standards which are similar. The software will compare Los Angeles County against the United Nations Standard Products and Service Code (UNSPSC). This will make 250 excel files named after the Commodity Class level in Los Angeles and each file will have every sheet which will have the similar UNSPSC and the percent similar.

There is currently no automated process available for finding potential matches between different standards, only manual algorithms. Los Angeles county has attempted to out-source this project to teams in the past but to no avail. A program with greater than or equal to 80% accuracy in predicting between two standards opens the door translating across multiple standards. The ability to automatically translate between different standards also greatly reduces the time necessary for the equivalent manual process..

1.4 Definitions, Acronyms, and Abbreviations

UNSPSC - United Nations Standard Products and Services Code

LA eCOMM - Los Angeles eCommerce

N/A - Not Applicable

NIGP - National Institute of Governmental Purchasings

NLP - Natural Language Processing

LA County, ISD - Los Angeles County, Internal service department

1.5 References

Python 3 Gensim Library

https://pypi.org/project/gensim/

Word2Vec

https://dzone.com/articles/introduction-to-word-vectors

https://radimrehurek.com/gensim/auto_examples/tutorials/run_word2vec.html

Google's Pre-trained Word2Vec model

https://code.google.com/archive/p/word2vec/

2. Overall Description

2.1 Product Perspective

The LA County ISD "Crosswalk"

The current difficulty is that we need to find the match manually in particular hierarchical levels (Family, Class, Commodity) of UNSPSC among LA eCOMM() , which is nearly impossible. The number of elements in UNSPSC are a lot more compared to the LA eCOMM.

For example: UNSPSC has 148032 elements in commodity & LA eCOMM has 6638 elements in Commodity Description

It shall provide a platform for the LA County, USD to input an LA County commodity standard and generate the most similar / equivalent standards from UNSPSC

2.2 Product Functions

Model features:

- Provide an LA County Standard
- Compare against all UNSPSC Standards

- ‘Crosswalk’ model shall return Topmost similar results

2.3 User Classes and Characteristics

Developers: They will be able to use and add or edit any implementations of our model

Users: Shall be able to use our model for topmost results but will not be able to modify our code.

2.4 Operating Environment

LA County “Crosswalk” will be available to anyone with access to any devices with an online browser. The results delivery method should be implemented by next year's Senior Design Team as a React Web Application.

2.5 Design and Implementation Constraints

- Constraints
    - Implementation: Our Model is written using Python 3
- Shall be designed to return top most similar UNSPSC standards to an LA County Standard
- Hardware: N/A Unable to test due to time constraints
- Interface with other application: N/A Unable to complete due to time constraints
- Parallel operation: N/A Unable to test due to time constraints
- Memory Constraints: N/A Unable to test due to time constraints
- Reliability requirements: N/A Unable to test due to time constraints

2.6 User Documentation

Our Project is currently not completed, it still needs refinements and a delivery method. Once completed the user documentation shall be implemented by next year’s Senior Design team.

2.7 Assumptions and Dependencies

1. Python Modules
    - Xlrd (excel file reader)
    - pandas (excel file writer, reader, and manipulator)
    - nltk (Natural Language ToolKit)
    - Gensim (Topic modelling)

- Wordnet (English Lexical database)
- Word2Vec (Word Embedding)

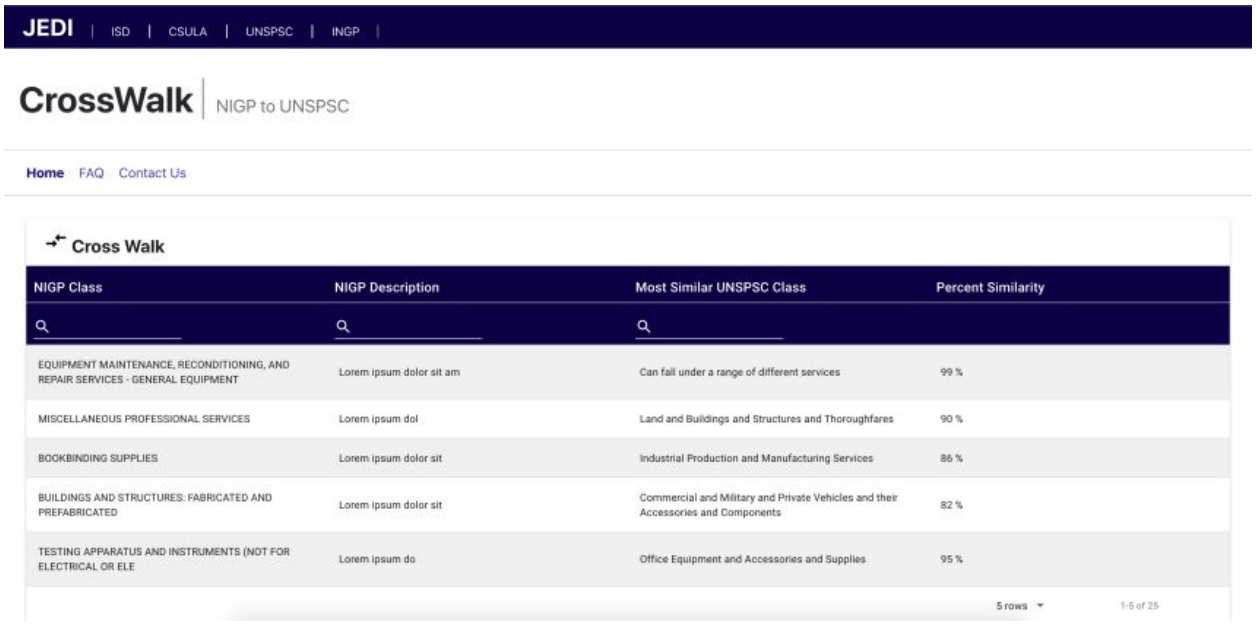
2.8 Apportioning of Requirements

Any remaining requirements will be applied in the next version of our application. Our application gives some accurate results but there are still many improvisations that can be applied to our model. These requirements will be completed by a new team in the following year of Senior Design.

3. External Interface Requirements

3.1 User Interfaces

Once Implemented by next year's Senior design TeamN



- Users can acquire results by crosswalk search within any standards they prefer.
- Our NLP model should return equivalent or top-most similar results sorted by similarity percentage.

3.2 Hardware Interfaces

Once designed by next year's Senior design our React Web Application must be accessed through any compatible devices with an internet connection, all the hardware shall require connecting to the internet. As for e.g. Modem, Router, Ethernet Cross-Cable.

3.3 Software Interfaces

Once implemented, our ‘Cross-walk’ NLP model will be available to anyone with a compatible device with access to a Web Browser.

3.4 Communications Interfaces

Not Applicable

Our React Web application has not been implemented. Therefore, we do not currently know the limitations of the web application.

This should be an open source application which does not require VPN connection from external users to access.

4. Requirements Specification

4.1 Functional Requirements

Currently our NLP model runs as python scripts and jupyter notebooks. Results are printed as Excel sheets.

The results should be implemented into a React Web Application

Inputs: Commodity standards

Outputs: should display equivalent or top-most similar results.

4.2 External Interface Requirements

React Web Application has not yet been implemented

Web Application should include:

- Search query for commodity standards
- Find equivalent or top-most similar
- Display top-most similar results

4.3 Logical Database Requirements

Not Applicable

4.4 Design Constraints

Refer to section 2.5

5. Other Nonfunctional Requirements

5.1 Performance Requirements

Not Applicable

React Web Application has not yet been implemented

5.2 Safety Requirements

Not Applicable

React Web Application has not yet been implemented

5.3 Security Requirements

Not Applicable

React Web Application has not yet been implemented

5.4 Software Quality Attributes

Quality Characteristics for the product: Instead of manually finding a similar or equivalent UNSPSC standard we created an easier method using Natural Language processing allowing the user to view top most similar to an LA County standard

5.5 Business Rules

LA County should be able to modify the parameters and edit code

All users will have access will have access to LA County 'Crosswalk’

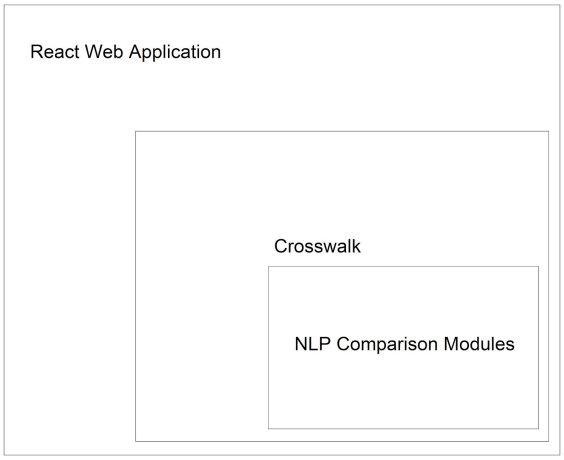
6. Other Requirements

Not Applicable

Appendix A: Glossary

Refer to section 1.4

Appendix B: Analysis Models



Crosswalk results should be implemented into a React Web Application

Appendix C: To Be Determined List

Refer to section 2.8