**University of Houston Clear Lake**

Spring ‘22

CSCI 4388.02

Senior Project in Computer Science

**Vision & Scope Document**



The Data Extractors

02/23/2022

**1.** **Problem Statement**

Project Background:

Tietronix is a software development and project management company that works with the aerospace industry and numerous government contractors such as NASA. A major part of an aerospace engineer, astronaut, and those working in related fields career is based around following strict procedures when accomplishing tasks, but at the same time relying upon sometimes the most basic, minimalist hardware and software that gets the jobs done. Because of this unique environment of controlled, sometimes borderline outdated tools, they need a simple program that would be able to translate sometimes complex Microsoft Word .docx documents into a simple barebones, labeled and arranged file following a XML schema. This format can then be used by Machine Learning tools, or Natural Language Processors, as well as being readable by the users themselves.

Stakeholders:

* *Victor Tang* – Tietronix President and Co-Founder. His main needs are accuracy and efficiency, in the forms of reliable results in combination with the fast processing/parsing of sometimes lengthy, technical procedures in .docx format Microsoft Word documents.
* *Tietronix* – Tietronix is a project management, software development, and technology consultant company that has worked with the likes of NASA, and in numerous different STEM fields. They need reliable software to be the base for Natural Language Processing and for the creation of basic XML versions of possibly complex .docx procedure documents.
* *The Data Extractors* – The development team consisting of University of Houston – Clear Lake Computer Science students. Needs consist of creating a working software that fulfills the outlined requirements fully and efficiently, as well as to the other stakeholder’s needs.
* *Dr. Alfredo J Davila* – Associate Professor at University of Houston – Clear Lake, overseeing the projects and teams as course leader. His needs consist of proper project management results, such as consistent progress, reports, meetings, and measurable output throughout the entire development, as well as the pre, and post development processes.

Users:

* Natural Language Processors – The program and/or its output could be able to be used in conjunction with Natural Language Processors.
* Researchers – Researchers would be able to learn either from the project and its design or use the project in aid of their research.
* STEM workers – workers in STEM fields, such as but not limited to, aerospace engineers and astronauts, will rely upon this software to process procedures into otherwise useful XML schema.
* Educators
* Students

Risks:

Risks to the project consist mostly of resource issues, such as time wasted or lost, access or possible loss of access to machines or previously worked on code, as well as possible schedule conflicts. There is also the risk that a teammate drops the class, becomes ill or incapacitated, or that the requirements change later on.

Assumptions:

* The input documents will mostly consist of procedures.
* The word document is not protected or limited in any way.
* The Input is a document in modern .docx file format.
* Software will be used for personal, research, or educational purposes.

**2.** **Vision of the solution**

Vision Statement:

Machine Learning techniques with Natural Language Processing can run into difficulties when encountering the raw text of a document without context. This difficulty results from Artificial Intelligence being unable to fully understand the semantics of the text without the context of what that text is within the document. The software we are building will help fix this problem for Machine Learning, specifically for documents that are in the Microsoft Word .docx format, and most importantly, documents with procedures to follow. It will read text from Microsoft Word .docx documents, and output it in a format that is readable to Machine Learning AI in the form of an XML file. It will accomplish this without losing any of the context of the actual text by arranging and including labels for the text in the output. Users of our software will have a means to approach previously inaccessible data in Microsoft Word documents for use in Machine Learning and Natural Language Processing.

List of Features:

* Command Line Interface/Input - The program must have a way to input commands and display messages, alerts, and output.
* A Parser/Scanner - The software should include a parser/scanner to actually go through the documents and parse out the data.
* Form of data sorter/lexical analyzer - The software should have a method to sort and label the different types of data parsed, such as lists/procedures, tables, simple text, objects, etc.
* Inputs .docx format - The software is essentially converting .docx format into XML schema, so it must be able to accept Microsoft Word .docx documents as input.
* Outputs XML file - The program will output a parsed and arranged XML file following a schema of the inputted .docx document.
* Output can be used by Machine Learning/Natural Language Processors - The output should be in a way that can be further used by Machine Learning tools or Natural language Processors.

Features that will not be developed:

* No mobile components
* No web-based components
* No support for outdated formats (.doc)
* No Graphical User Interface

**Project Plan document**

Statement of Work:

This statement of work describes what the team will accomplish over the course of this project’s development cycle, which of the team members is responsible for a specific work product, and how many hours of effort each work product will take to finish.

**Software Requirements Specification:** The SRS describes the functionality of our Data Extractor by including definitions of the functional and non-functional requirements of the software as well as describing the use-cases which show how users will end up interacting with the software. The SRS and its accompanying PowerPoint presentation was worked on by all 5 members of the team and took about 30 hours of effort to complete.

**Code:** Using Python and more specifically Python-docx library, we estimate we will spend about 100 person-hours coding our program. Our program will read Microsoft Docx documents and output XML files of data from the document. The coding will be worked on by all members of the team using GItHub as source control.

**Test Plan:** Our team estimates it will take about 10 person-hours to create a test plan to test our program and make sure it is reading documents and outputting results in a satisfactory execution time.

**Defect Report:** A document that describes any errors that come up during testing, what caused them, and what the expected outcome should be instead of an error. We estimate that the defect report will take between 8 to 16 hours of effort to complete, and will be handled by the team’s editors/testers Mason and David.

**Reports:** There will be various types of reports done by the team throughout the semester in order to monitor progress. The weekly individual reports will detail how each member of the team spent their time each week, while the weekly team reports, compiled by the team’s scribe David will describe what was discussed or accomplished in each of the team’s meetings for a specific week. Both the individual and team weekly reports take about 30 minutes of effort per week.

The team will also have bi-weekly progress reports, written by project manager Mason, in order to make sure the team members are following the roles and procedures outlined in the team’s charter, as well as two progress reports at the middle and end of the semester, worked on by the whole team, to evaluate how the progression of our software’s development and how our team is working together. Both of the two types of progress reports will take about 1 hour of effort per each report.

**Research:** The research that has been done and will continue to be done by all team members will take roughly 50 person-hours. The team’s research will include, but is not limited to, how to write effective and efficient XML code, as well as how to use Python in order to read text data from a Word document.

Resource List:

* Personal Computers/Laptops/Tablets - Available at most times.
* Publicly accessible computers (such as STEM labs) - Available on campus in rooms:
  + Bayou 3608 - Mon-Thurs @ 8am-10pm, Fridays 8am-5pm
  + Delta 205 - Mon-Thurs @ 8am-10pm, Fridays 8am-5pm
  + SSCB 2201 - Mon-Thurs @ 7:30am-10:30pm, Fridays & Saturdays 8am-5pm
  + Pearland 104 - Mon-Thurs @ 8am-10:30pm, Fridays 8am-5pm
* IDE - Visual Studio Code or similarly basic, yet powerful IDE. Available by download and for free.
* .docx documents provided by Tang for processing.

Work Breakdown Structure:

1.Initiation

1.1 Appoint Project Manager

1.2 Draft Team Charter

1.3 Peer Review Team Charter

1.4 Prepare Team Charter Presentation

1.5 Team Charter Presentation

2.Documentation

2.1 Gather Requirements

2.2 Draft SRS

2.3 Peer Review SRS

2.4 Prepare SRS Presentation

2.5 SRS Presentation

2.6 Draft Vision and Scope Document

2.7 Construct WBS

2.8 Construct Project Schedule

2.9 Draft Risk Plan

2.10 Peer Review Vision and Scope Document, WBS, Risk Plan and Project Schedule

2.11 Prepare Vision and Scope Presentation

2.12 Vision and Scope with Project Plan Presentation

3.Design Document

3.1 Analyze SRS

3.2 Draft Design Document

3.3 Peer Review Design Document

3.4 Prepare Design Presentation

3.5 Design Presentation

4.Coding

4.1 Analyze Design Document

4.2 Code Input Feature

4.3 Unit Test Input Feature

4.4 Code Parsing Feature

4.5 Unit Test Parsing Feature

4.6 Code XML Output Feature

4.7 Unit Test XML Output Feature

4.8 Code Help Feature

4.9 Unit Test Help Feature

5. Testing

5.1 Draft Test Plan

5.2 Create Test Cases

5.3 Peer Review Test Plan and Test Cases

5.4 Prepare Test Plan Presentation

5.5 Test Plan Presentation

5.6 Test Execution

5.7 Defect Tracking

5.8 Debugging

6.Deployment

6.1 Draft Postmortem report

6.2 Peer Review Postmortem Report

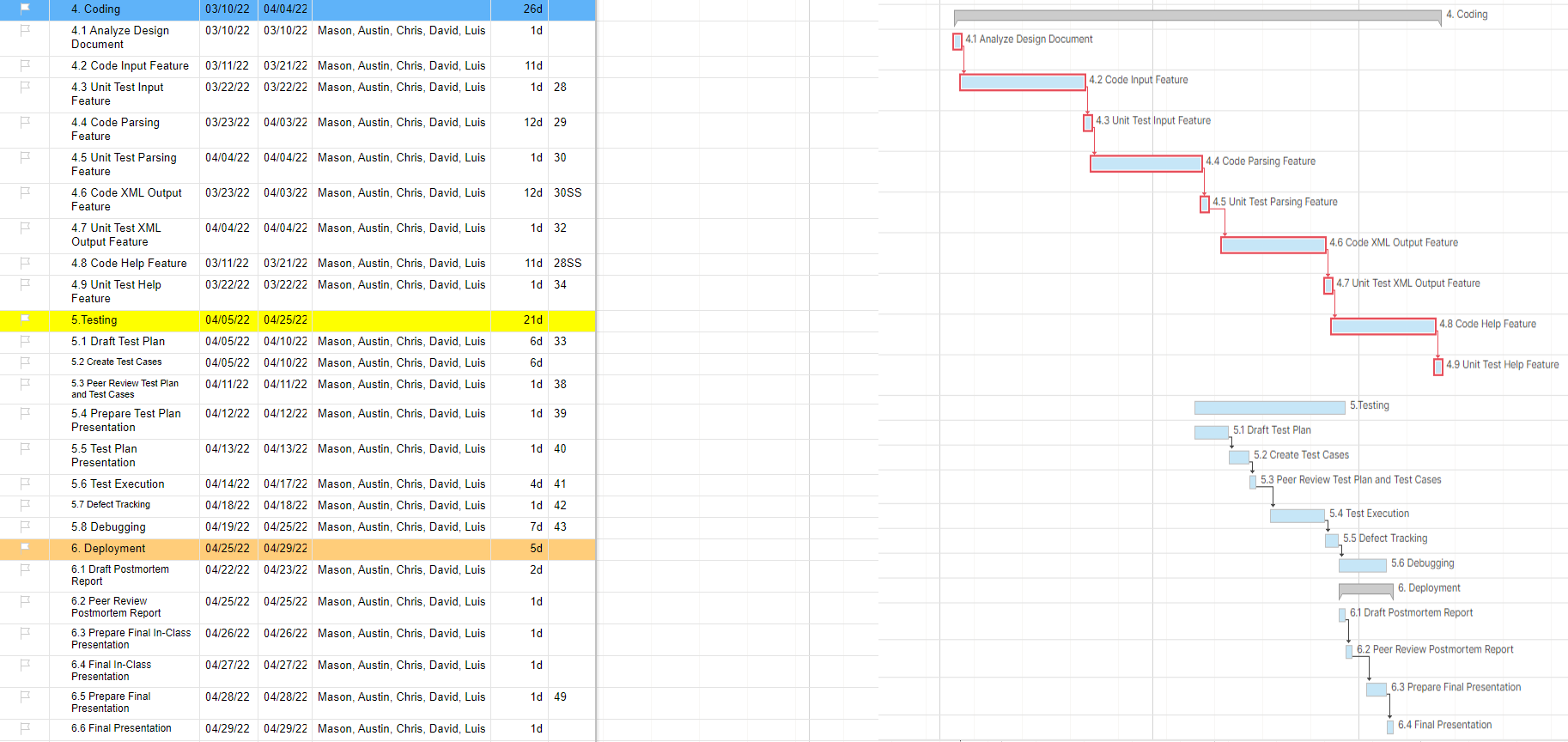
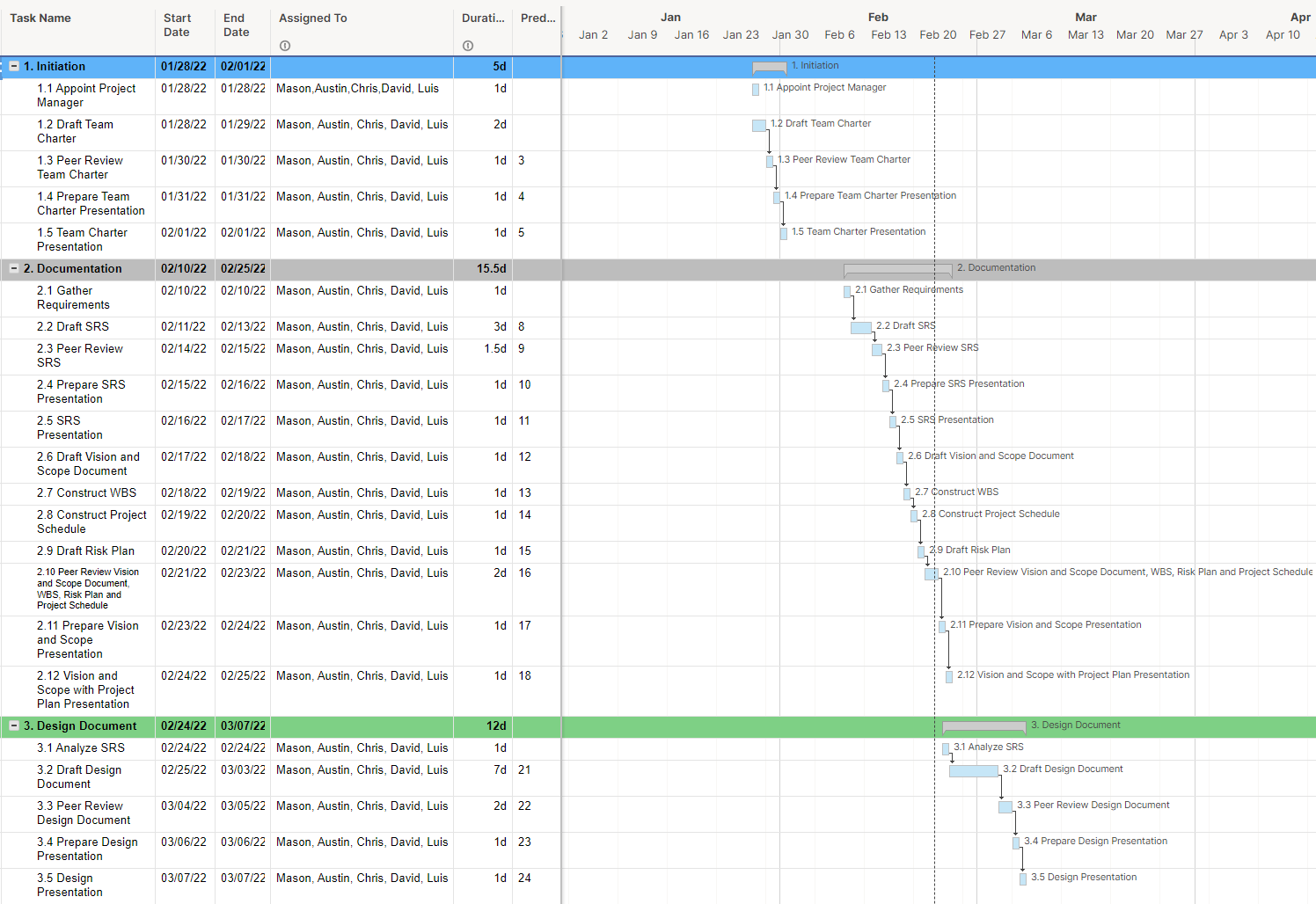
6.3 Prepare Final In-Class Presentation

6.4 Final In-Class Presentation

6.5 Prepare Final Presentation

6.6 Final Presentation

Project Schedule:



Risk Plan:

| Risk | Probability | Impact | Priority | Actions |
| --- | --- | --- | --- | --- |
| Scheduling Conflict | 4 | 3 | 2 | 1.Rescheduling, 2.Communication through Discord.  3.Notify project members at least 1 week early and/or as soon as possible |
| Requirements Elicitation Failure | 2 | 5 | 10 | 1. Keep constant communication with Mr.Tang as we develop.  2.Upfront and clear communication with Mr.Tang. |
| Coding something unused | 4 | 2 | 8 | 1.Repository Managers continuously monitor what is being uploaded. |
| Illness | 2 | 4 | 8 | 1.Work from home.  2.Online Meetings.  3.Encourage people to wear masks and vaccinate. |
| Power goes out while working | 5 | 1 | 5 | 1.Autosaves on.  2.Constantly save  3. Repository saves work from being corrupted. |
| Dropping/Incapacitation | 1 | 5 | 5 | 1.Have a meeting to re-distribute work/tasks |
| No Access to machine | 1 | 4 | 4 | 1. Can go to campus if unable to work at home.  2. Austin has an extra laptop that he can provide to team members that need a machine to work on |
| Closure of Campus (Covid 19) | 1 | 2 | 2 | 1. Online meetings  2. Meetings at a public place. |