Text File Analyzer

Software Requirements Specification (SRS)

Mark Buenaflor
Christopher Eich
Austin Peterson
Zoe Vasquez

Table of Contents

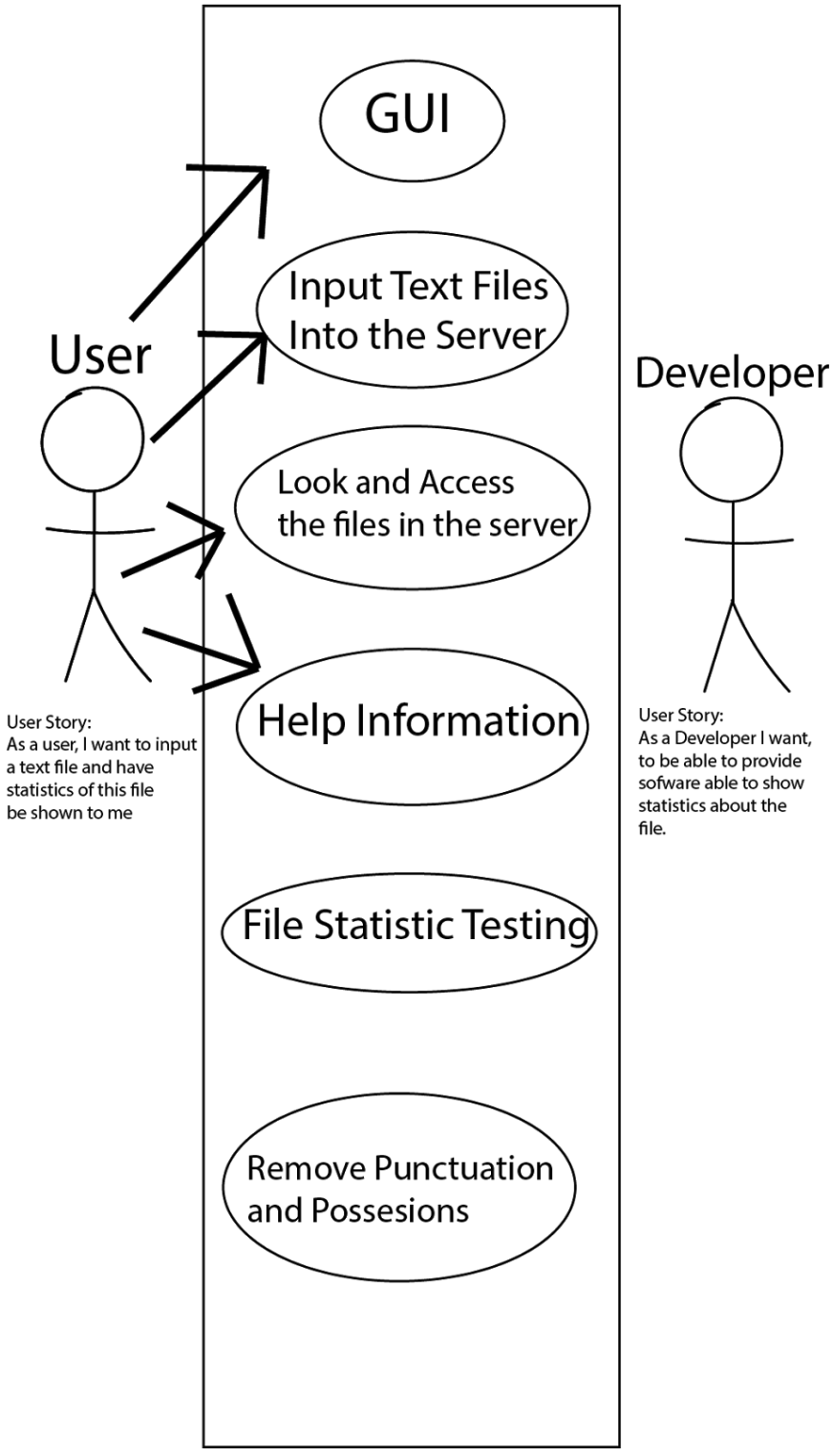
Revision History

Date	Revision	Description	Author
10/19/17	1.0	Initial Version	All
10/19/17	1.01	Added Use Case Scenarios	Mark B./Zoe V.
11/2/17	1.02	Added Plan for Second Deliverable	Zoe V.

Program Overview

The Text File Analyzer is a GUI-based program designed for a single user to enable them to analyze a specific file for useful information regarding its text. All input to the program must be supplied in the form of a .txt file(s), otherwise the program cannot successfully open the file and will instead error. The Text File Analyzer will return information to the user in the form of a processed word count, total number of lines, spaces, and any lines that were blank/whitespace. In addition, the program will calculate the average number of characters per line, average word length, and the most common words contained in the text file. Punctuation and possession will be stripped from this analysis to provide more accurate results. Finally, all information will be displayed to the user through the GUI-based program for use, along with being recorded in a persistent log to track both processed file history and averages across all files. In order to enhance ease of use for the user, documentation is provided in the form of a readme and the GUI will give direction to the user.

Use Cases


GUI
Input Text Files
Into the Server
User
Look and Access
the files in the server
Help Information
File Statistic Testing
Remove Punctuation
and Possesions
Developer
User Story:
As a user, I want to input
a text file and have
statistics of this file
be shown to me
User Story:
As a Developer I want,
to be able to provide
sofware able to show
statistics about the
file.

User Stories

GUI:
As a User, I want a GUI so that I can easily understand how to use the program.

Input text files into server :
As a User, I want to be able to input text files into the server so that I can choose which text file the program analyzes.

Visual history of files processed:
As a User, I want to be able to see what files have been put into the server and which ones have been processed.

Help Information:
As a User, I want to be able to see the help information so that if I do not understand how to use the program, I can view its instructions.

File Statistic Testing:
As a Developer, I want to be able to carry out file statistic testing so that I can see the stats for each document uploaded by the user.

Removing punctuation and possessions:
As a Developer, I want to be able to remove punctuation and possessions from a file so that I can simplify and have more accurate results.

Test Cases

	Feature	Description	Possible Input	Expected Output
1	GUI	The "GUI" encapsulates the graphical interface available to the user within the software. Things like text boxes, buttons and file analysis information are encompassed in the GUI functionality of the program.	This will be tested as a component of all other functionalities. The entire functionality of the program will be tested, which should have a complying GUI.	Correctly sized, scaled GUI with correct information and functionalities present.
2	File Access/Upload File	The software will have a file uploading functionality that can be used to select and upload the file to the file analyzer. The software will read text files.	Files will be chosen from the system to be uploaded to the program. These may be a text file or any other sort of tile, depending on what the user chooses.	Only the text file should be uploaded, with a success message. Any other file type will be ignored/met with an error message.
3	File Reporting/Statistics	The file reporting system inside the software will, when supplied with an uploaded file, will report the following: a. # lines b. # blank lines c. # spaces d. # words e. Average chars per line f. Average word length g. Most common words	The user will use the file report button to request analysis of the uploaded file. The files selected will vary in statistics and size.	The software will output the stats listed in the description correctly and visible to the user.
4	Help information	The software will have a 'help information' option that will allow the user to look for and receive help in regards to operation of the software.	The user can use the 'help center' to read about the functionality of every aspect of the software.	The software should correctly display information regarding help.
5	Analysis updating	The software, if prompted, will be able to return an updated version of the regular file report that removes punctuation and	The user will use the analysis update button to update the statistics displayed.	The software should correctly display the statistics for the current file.

		possession from the file		
6	Multiple File Functionality	The software will have the capability to read and analyze multiple files over time.	The user will first upload a file and then choose another file to upload and analyze.	The software should correctly handle all functionalities on the second file, discarding the reports from the previous files.
7	History statistics	The software will have the capability to report all the files historically uploaded or analyzed by the program..	The user can choose to view a history portion of the software that records all the previous files uploaded	The software should correctly display a list of previously uploaded files
8	Historical file analysis	The software will have the capability to report statistics from previously uploaded files in the software's history.	The user can choose to view an analysis of a previously uploaded file	The software should correctly display statistics from the previously uploaded file.

Class Diagrams

Plan for First Deliverable

	Task	Description	Assignment
1	Class diagrams	Class diagrams provide an overview of the final program, including attributes of each class needed for the program. Designed in Photoshop.	Mark, Chris
2	Test cases	Test cases will be written up using Google Docs to test various inputs. Outlier inputs to test possible error cases will be included.	Austin
3	Program classes (basic)	Basic implementation of the program will begin; Java will be used. The basic program will take an input file name and validate it.	Zoe, Mark, Chris, Austin
4	Input file analysis	Removal of punctuation, calculation of the number of occurrences of lines, spaces, and words in the given file. Will be coded in Java.	Austin
5	Help Information	Rough draft written up using Google Docs; will layout information for the user, including instructions the user will follow to retrieve particular outputs.	Zoe

Plan for Second Deliverable

	Task	Description	Assignment
1	Program classes - file analysis implemented.	Program will be able to read text from the validated text file and report at least half of the necessary statistics.	Austin, Mark, Chris, Zoe
2	Basic implementation of “file history” analysis function	GUI will have an option for the user to maneuver through file history. Ability to report statistics including files from history not yet included, but basic framework is laid out.	Austin, Mark, Chris, Zoe
3	Removal of punctuation and possession from analysis.	Rough draft of code or pseudocode to remove punctuation and possession finished.	Austin, Mark, Chris, Zoe