Requirements Document – Team 17
Average Sentence Count Application

Statement of Purpose

The main purpose of the the Average Sentence Count (ASC) application is to allow a user with the ability of a college student with below average technical skills to count the average number of words in sentences in a selected input file. The goal of the ASC application is to be user friendly to allow users to easily accomplish the given task at hand, flexible to allow the users to customize how he or she wants word counting to occur, and reliable to avoid user frustration from events like runtime errors and glitches.

1 User Requirements

1.1 Software Interfaces

- Java 1.7
- commons.io.2.4.jar

1.2 User Interfaces

The generic format by which a user will enter input into the Java ASC (average Sentence Count) application is as follows:

```
java <MainASC.class> <FileDirectory>space<FileName>space-dspace”
&/@”space-lspace<number>-sspace”#”
```

As an example, If a user wants to read a file from his or her documents folder with delimiters “ “, “|”, and “%”, and a minimum word length of three, the following would be the Java ASC input:

```
example:java MainASC.class test\inputfiles\  <file name> -d “ |%” -l 3
```

1.3 User Characteristics

The ASC application user base is assumed to be college students with below average technical skills as per the gathered requirements.

While there is some ambiguity about how to quantify what technical skills the average college student, with below average technical skills, has, there is a plethora of data collected by the U.S. government indicating the availability of computers in U.S. public schools. Without hard data from a collegiate survey, in order to define the technical ability of the ASC application's user base, an assumption must be made that the majority U.S. public school system students that actually had access to a computer while in school actually learned what we deem as basic computer usage ability. We define this as: the ability to navigate applications within a Windows or MacOS computer, the ability to navigate computer system file directories within a Windows or MacOS computer, the ability to launch a desired application within a Windows or MacOS computer, and the ability to type english on a standard U.S. keyboard layout.

According to the National Science Foundation (NSF), "Nearly all public school teachers (99 percent) reported having computers available somewhere in their schools in 1999; 84 percent had computers available in their classrooms and 95 percent had computers available elsewhere in the school. Thirty-six percent of teachers had one computer in their classrooms, 38 percent reported having two to five computers in their classrooms, and 10 percent reported having more than five computers in their classrooms. Teachers were generally more likely to use computers and the Internet if the computers were located in their classrooms than if they were located elsewhere in the school." Furthermore, the NSF claims, "Internet access existed at 35 percent of public schools in 1994, but this statistic soared to 98 percent by 2000."

Given our assumption paired with hard data from the NSF, we can assume that any college student in the U.S. that attended a public school after the year 2000 has the technical ability to use the ASC application.

2 System Requirements

The following table is an all inclusive compilation of the ASC application functional and non-functional requirements. Any functionality of the ASC application that is not conducive to compliance with the following requirements is assumed to be incorrect functionality.

Requirements	Description
FR.ASC.1	The ASC application shall be platform agnosti
FR.ASC.2	The ASC application shall use JAVA 1.7
FR.ASC.3	The ASC application shall be a JAVA comman line tool
FR.ASC.4	The ASC application shall be usable for colleg students with what is deemed to be below average computer/technical skills.
FR.ASC.5	The ASC application shall accept any file with the .txt file extension.
FR.ASC.6	The ASC application shall have sufficient documentation and user help files to assist students with any ambiguity concerning JAVA ASC usability
FR.ASC.7	The ASC application shall have industry standard error handling capability for any exceptions that may be thrown in the software for the input of a non-existent file path, input o an invalid file path, and any switching options specifically defined in this document.
FR.ASC.8	The ASC application shall count any continuou three character sequence as a word; this is barring, hower, any user specified delimiters.
FR.ASC.9	The ASC application shall have default sentenc delimiters of the characters “.”, “!”, and “?”

FR.ASC.10	The ASC application shall count the number of words in a sentence conducive to any applicabl sentence or word delimiters
FR.ASC.11	The ASC application shall have a default word delimiter of “ “.
FR.ASC.12	The ASC application shall allow the user to inp a custom sentence delimiter list in the format o quotation marked text string.
FR.ASC.13	The ASC application shall allow the user to inp a custom word delimiter list in the format of a quotation marked text string.
FR.ASC.14	The ASC application shall take any input list o custom delimiters as all inclusive. Where applicable, the user must specify the need for any default delimiters.
FR.ASC.15	The ASC application shall allow custom sentence delimiters following the switch statement “-d”
FR.ASC.16	The ASC application shall allow a custom wor length specifiers following the switch statemen in the form of integer input
FR.ASC.17	The ASC application default delimiters shall st apply if ASC receives any delimiter(s) that do not exist in the input file.
FR.ASC.18	The ASC application shall have case sensitive input switches
FR.ASC.19	The ASC application shall not modify any user input delimiters, and any delimiter input shall b case sensitive.
FR.ASC.20	Package ASC application as a console program into ASC.jar file. Allow to distribute it to users with extension .zip