Product Backlog

KidText

Trent Apple, Chris Bauschka, Tyler Hoffman, Elizabeth Murphy, Aaron Petry, Kyle Ribordy, Brittany Vacchiano

Problem

The product users (children) need to perform various text manipulation functionalities in a simple manner; this must be done in a way which allows them to learn the differences in input and output (result) for each text manipulation.

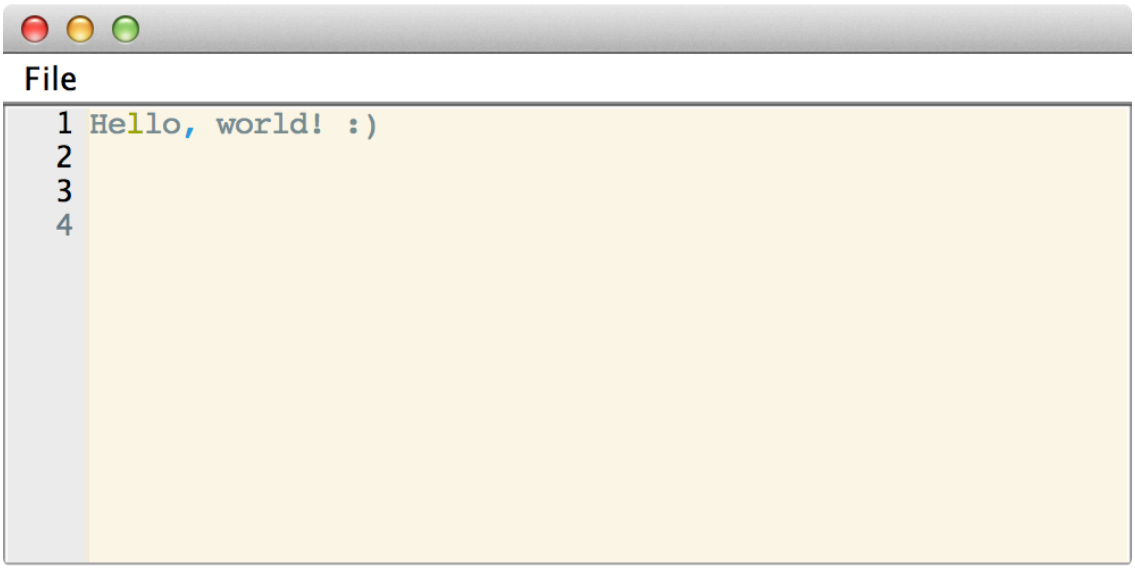
Background Information

As the world becomes more intertwined with technology, there are certain skills that everyone should possess. Text manipulation is a valuable skill for any professional or casual computer user. Whether it be replacing words, capitalizing letters, or tracking text statistics of your song/letter/paper/movie script, there are so many wonderful things you can do through text manipulation! Now that you agree that everyone should have this skill, shouldn't we also expose our children and the youth of our society to it? I'm glad you think so. This is exactly why we built our application! Our program will allow children of all ages to explore the exciting world of text manipulation. Parents will be able to join along in the fun as their children can use all of the different functions that we have to offer like compare, diff, replace, and encryption! With our simple user interface, they will have no problem hitting the ground running and jumping right into the thick of things. With these skills, they will have the necessary basis to follow a career path in basically any direction.

Environment and System Models

The product is a standalone application (.jar) which is platform independent. Since we are using Java to build the application, the only environment requirement is that the system has the Java Runtime Environment

installed.



NOTE: Pre-alpha version. Will become include more features and look-and-feel will change.

Functional Requirements

Replace

- Find and Replace
  - After highlighting a word and clicking “Find” in the GUI’s dropdown menu, users can replace all the words matching the one in which they highlighted with a new word of their choice.
  - This find and replace method can be case-sensitive or case-insensitive.
  - Find and replace will work for whole and partial words.
  - A user can either highlight the word they want to find and replace or they can input the word into a search box.
- Wildcards
  - Users can replace a word matching a wildcard with a new word of their choice.
- Regex
  - Users can replace a word matching a regular expression with a new word of their choice.

Compare / Diff

- After selecting “Compare” in the drop down menu, users can upload two files in which they want to compare. The files will be compared dynamically and their differences will be displayed in a pop-up window.

Java Swing-based GUI

- Text Editor
    - Program will contain a window into which users can type text or code. Multiple lines of text or code can be typed into the editor window.
    - When users type text or code into the editor window, the text will change colors based on the word or words typed. Otherwise known as syntax highlighting.
    - When a user types text or code into the editor window, line numbers will be displayed to the left of the line.
- Dropdown Menu Items
    - The program will contain a dropdown menu in which a user can select various functions to complete on the text in the Text Editor window or on a text file which the user uploads.
    - Some of these dropdown menu items include, but are not limited to, Save, Save As, Quit, Word Count, Number of Brackets, Number of Sentences, Number of Spaces, Search Result Count, Encrypt Text, Decrypt Text, and various Find methods.

Text Statistics

- Word Count
    - The text editor window will display a count of the number of words inside the text editor window at the bottom of the screen.
    - The program will dynamically display, via a pop-up box, the number of words inside a document in which users upload.
- Number of Brackets
    - The text editor will display the number of brackets used inside the text editor window at the bottom of the screen. This count will turn red if there is an odd number of brackets in the text editor window.
- Number of Sentences
    - The text editor will display a count of the number of sentences inside the text editor window at the bottom of the screen.
- Number of Spaces
    - The text editor will display a count of the number of spaces inside the text editor window at the bottom of the screen.
- Search Result Count
    - After users highlight a word and click “Find” in the dropdown menu, a “Find Result Count” will be displayed on the bottom of the screen which displays how many words match the highlighted word.
    - When users enter a word into the “Find” dialog box after clicking “Find” in the dropdown menu, a “Find Result Count” will be displayed on the bottom of the screen which displays how many words match word entered into the dialog.

Basic Cipher

- ROT13 Encryption
  - Users can encrypt, using the ROT13 method, the words they typed into the text editor window. The encrypted file will be saved to users' computers.
  - Users can upload a file to encrypt, using the ROT13 method. The newly encrypted file will be saved to users' computers'.
- ROT13 Decryption
  - Users can decrypt, using the ROT13 method, the words they typed into the text editor window. The decrypted file will be saved to users' computers.
  - Users can upload a file to decrypt, using the ROT13 method. The newly decrypted file will be saved to users' computers'.
- ROT-n Encryption
  - Users can encrypt, using the ROT-n method, the words they typed into the text editor window. The encrypted file will be saved to users' computers.
  - Users can upload a file to encrypt, using the ROT-n method. The newly encrypted file will be saved to users' computers'.
- ROT-n Decryption
  - Users can decrypt, using the ROT-n method, the words they typed into the text editor window. The decrypted file will be saved to users' computers.
  - Users can upload a file to decrypt, using the ROT-n method. The newly decrypted file will be saved to users' computers'.

Non-Functional Requirements

- Performance
  - The application should be able to handle everything from miniscule to relatively large inputs while still completing tasks in a timely manner.
- Usability
  - The application should be easy to use for those unfamiliar with computers. A simple interface should exist without the need for detailed instructions.
- Error Handling / Fault Tolerance
  - The application should be able to handle errored inputs without failing or crashing.
- Cosmetics
  - The application should have a pleasant interface. Users should enjoy using our application.
- Portability
  - The application should function on a wide variety of platforms and in a wide variety of environments.

- Consistency
    - The look, feel, and function of the application should be consistent. A user familiar with one function of our application should be able to operate other facets with relative ease.

Use Cases

- Replace
    - User selects “Replace” command from dropdown menu.
    - An dialog box appears asking the user for two inputs:
        - Search text (e.g. Green)
        - Replacement text (e.g. Orange)
    - The user fills in the desired search terms and replacement text
    - Once completed, the user clicks “Replace” and all search text (Green) is replaced with the desired replacement text (Orange).

- Search Result Count
    - A user may either:
        - Highlight a word
            - Click “Find” in the dropdown menu.
            - A “Find Result Count” will be displayed on the bottom of the screen which displays how many words match the highlighted word.
        - Click “Find” in the dropdown menu.
            - Enter a word into the “Find” dialog box
            - A “Find Result Count” will be displayed on the bottom of the screen which displays how many words match word entered into the dialog.