

EECE 2140: Final Project Proposal

Spell Checker Application (Sample Project Idea 1)

Project Participants

I, Frank Anderson, will be the only participant in the project.

Application Description

The application will take in a text file with some misspellings, and then suggest words which are similar for the user to pick from. It will ignore words which begin with capital letters that are not at the start of a sentence because these are proper nouns. Additionally, it will look for repeated characters the word they are in is not in the dictionary in addition to repeated spaces and other special characters. When the user makes a choice of which word to use as a replacement, it will replace the word in the text file.

The main use of the application will be checking spelling on plain text documents where the writing application does not have its own built-in spell-checker. One potential user is developers who want to check the spelling on their documentation.

Application Design

I suspect I will need three classes: a dictionary class, a file class, and an input manager class. The dictionary class will have a static variable which holds a set of all the words in the dictionary. When initialized, the dictionary class will read a text file which stores the dictionary and file that into the static variable. This way the dictionary will be very easily accessible throughout the program. The file class will read the file when initialized and then split all the text by spaces to identify individual words. Then the words will be split again by special characters in order to maintain those. All the words will be checked for spelling and the special characters for repeated occurrences. If errors are found, they will be added to the input manager class. Once the file is read, the input manager will proceed through each word, asking the user if they would prefer an alternative. When the inputs have all been managed, the input manager will rebuild the text file and save it.

Libraries and Tools

I should not need any libraries or tools beyond the standard Python library in order to read and write files along with compare strings and substrings.

Results

The ideal outcome of the project is completion as I have described it here. The largest risks for not getting the project done on time are getting stuck on one aspect of the required elements (e.g., rebuilding the text file, scanning all the words for misspellings, finding suggestions, etc.). In this case, I will ask the professor and TA for help, and if I continue to be stuck will discuss narrowing the scope of the project with them.