Fellowship Project

I. Installation Guide – Everything you need to use this code and program. It would be useful to have a program such as Microsoft Excel to view the csv files in an organized manner.
    a. Python 3.6
    b. Pycharm IDE
    c. Biopython
    d. NCBI BLAST+
    e. Alignment Documentation
    f. Data formatted in sequence output format or split as seen in sorter.py.
        i. If already split, continue to databaseGen.py.

II. User Guide – For all BLASTing and Mutations steps remember to change the input variables in the functions for the script to match that of the antibiotic name and the phenotypes you are testing.
    a. Data Preparation (in order).
        i. Parser - Used to create the fasta files and the gene information files based on the gene annotation files that were given.
        ii. Sorter - Used to sort through each organism and placing them in the correct phenotype file so that they may be compared later.
        iii. databaseGen - Used to compile all the fasta files generated above into 1 fasta file. This file is then made into a Blastn database for each organism.
    b. BLASTing (in order).
        i. GeneMatcher - Blasts all the genes of 1 organism against all the organisms of the same phenotype. This gives us a list of all recip genes.
        ii. uniqueChecker - This blasts all the recip genes from above against all the organisms of the opposite phenotypes. If there is any match the gene is removed as it is not unique to the phenotype.
        iii. commonGenes - This is used to create a csv with the information of the genes that are all found in the same phenotype but are not found in any organisms of the opposite phenotype.
    c. Mutations
        i. GetMutations – Finds matches between recip genes and get the count of mutations.

ii. GetMutationInfo – Find the function information for each of the genes that are mutated.
iii. GetGaps – Find what amino acids are on the other side of gaps for each gene mutated.
iv. FilterMutations – Remove the genes that are not mutated in all strains of the bacterial phenotype.

d. Misc.
i. Blast - Used to return the result of a BLAST against a target organism using a specific gene.
ii. Mutation – Used to find and store all the mutations found between a reference strain and a strain that has matched to it.
iii. Gene – Used to store gene information that can be easily accessed in other parts of the application.

III. Developer's Guide

a. Classes
i. The classes could be more simplified or have added functionality to reduce the code in the other scripts.

b. Code
i. Much of the code is shared between each script. Steps could be made to optimize the code used with pandas specifically and the creation of the csvs.

c. Functions
i. A script can be made to run each of the classes together with the same input variables. Thus, it would work on its own and find mutations without further user input.