**OVERVIEW**

As shown in our GUI mockup, the user can either solve equations of a single variable or a system of equations. Upon selection, the user will be brought to a new setup menu offering several numerical methods within the respective domain.

For single-variable equations, solutions can be generated through Newton’s Method, Fixed-Point Iteration, and/or Bisection Method.

For systems of equations, solutions can be generated through Gaussian Elimination, LU Decomposition, Iterative Methods-Jacobi, SOR, Multivariable Newton’s Method, and/or Broyden’s Method.

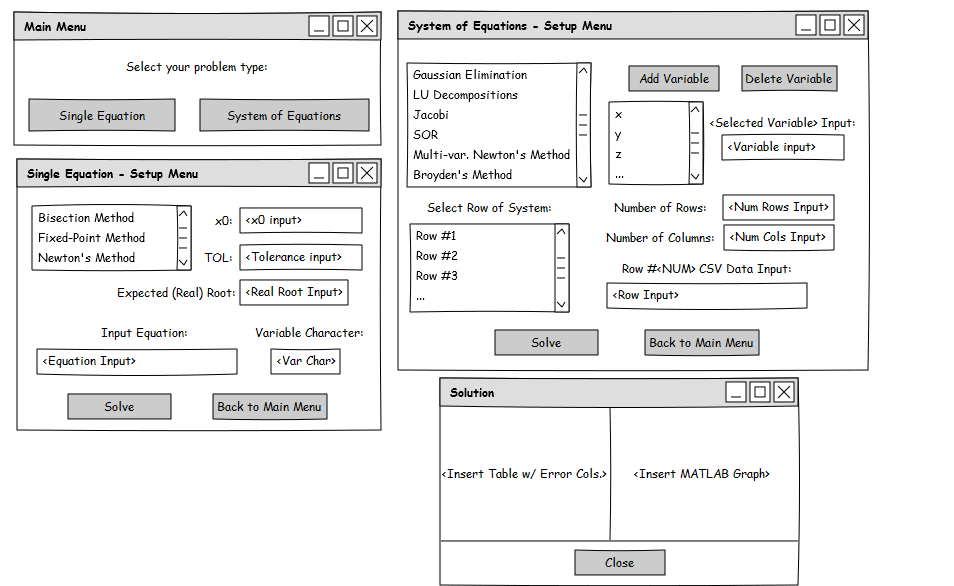
Our program will be easy to use, robust, and easy to visualize. The user can expect tables of iterations, graphs, and measurements relevant to error analysis, algorithmic efficiency, and time-complexity.

**PROCESS**

**Input**

The GUI mockup shows what the user will interface with to enter test problems into the machine. The user will be provided with an initial guess, a tolerance, and an expected root. The software will be robust enough to input only the proper input needed per method. The user will also be able to enter a matrix by row and will be dynamically created in the software and provided to the user in a spreadsheet.

**Output**

****The software will provide the user the output from the function. The output will be displayed in tables and graphs that the user will be able to formulate an analysis. There will be relative error and absolute error provided to the user when relevant as well the iteration counts, flops, flops per digit of accuracy, and execution times.