Functional

1. The system shall allow users to load a BasicML program into memory, starting at location 00.
2. The system shall provide a 100-word memory space where each word is a signed four-digit decimal number.
3. The system shall interpret and execute BasicML instructions according to their respective operation codes.
4. The system shall include an accumulator register for performing arithmetic and logical operations.
5. The system shall support addition of two numbers
6. The system shall support subtraction of two numbers
7. The system shall support division of two numbers
8. The system shall support multiplication of two numbers
9. The system shall support storing data to specific memory locations.
10. The system shall support loading data from specific memory locations.
11. The system shall allow users to input data into memory locations when required by a program.
12. The system shall display output values as specified by executed BasicML instructions.
13. The system shall correctly decode and differentiate between operation codes memory addresses
14. The system shall support branching flow control
15. The system shall support conditional branching
16. The system shall detect and report errors such as invalid instructions, memory overflows, and division by zero.
17. The system shall halt execution when encountering a termination instruction or an unrecoverable error.
18. The system shall allow users to execute programs step-by-step for debugging purposes.
19. The system shall provide logs of executed instructions and memory changes for debugging purposes.

Non-Functional

1. The system shall provide a GUI for loading, executing, and debugging BasicML programs.
2. The system shall be platform-independent and executable on multiple operating systems (Windows, Linux, Mac).
3. The user interface shall display help buttons with instructional text for the various aspects of the GUI