Problem Statement

A program needs to be written to automate a round of “Fizz Buzz,” where numbers increment starting at 0, and each number divisible by 3 is replaced by “Fizz,” each number divisible by 5 is replaced by “Buzz,” and each number divisible by 15 is replaced by “Fizz Buzz”.

Proposed Solution

Our team shall write and test a program that can accurately initiate and carry out a round of Fizz Buzz given a number by the user for the round of Fizz Buzz to count to.

Functional Requirements

- The program shall ask the user for a number to play Fizz Buzz up to.
- The program shall output all numbers and/or words to standard out (stdout) separated by a comma and space.
- The program shall have all instances of “Fizz” and “Buzz” capitalized.
- The program shall output all instances of “Fizz,” “Buzz,” and/or numbers as the current number being iterated over is checked, not at the end of the program.
- The program shall output Fizz for every number divisible by 3, Buzz for any number divisible by 5, and Fizz Buzz for any number divisible by 15.
- Each output shall be separated by a comma and a space (“, “), except for the last number or Fizz/Buzz output from the program.
- The program shall add a line break every 10 outputs.
- The program shall output a message saying that it has completed the game on a new line after the specified number of rounds has been reached.
- The program shall only allow positive numbers and shall output an error message and take another number input if an invalid input is given.

Non-Functional Requirements

- The program shall correctly play a game of Fizz Buzz according to the rules over a range of numbers specified by the user.
- The program shall check if the current number being iterated over is divisible by 15, 5, or 3 in that order.
- The program shall consider an input invalid if it is not a single integer value and/or if it is non-positive.
- The program shall check to see if the current number being iterated over is divisible by 10 to output a new line.

Architecture

The program will follow a pipe-filter architectural pattern, taking in an input and processing it to return an output. There will be two major parts:

1. A `fizzBuzz` function that accepts an integer number of rounds to play and outputs a list of Fizz Buzz outputs.

2. A main function that accepts input from the user (stdin) and outputs the results to the screen (stdio).

UI

The user interface for this application will consist of a console window where a user can input the number used to play Fizz Buzz up to. The output of the program will also be in the console.

Use Cases

System	Fizz Buzz
Name	Fizz Buzz Generation
Primary Actor(s)	Player
Description	The player enters a number they wish to see counted to from 1 with each number divisible by 3 and 5 replaced with Fizz and Buzz. If the number is divisible by both then it shall be replaced by Fizz Buzz
Pre-Conditions	The player must have a number in mind to enter
Post-Conditions	A list where all numbers divisible by 3, 5 and 15 are replaced by Fizz, Buzz, and Fizz Buzz respectively
Trigger	A number is entered

Tests

No.	Requirement Test Case(s)	User Input	Pass Criteria
1	The program shall not allow numbers less than 1 to be inputted to start a round of Fizz Buzz.	-1	The program outputs “’-1’ is not a valid input for Fizz Buzz.”
2	The program shall not allow numbers less than 1 to be inputted to start a round of Fizz Buzz.	0	The program outputs “’0’ is not a valid input for Fizz Buzz.”
3	The program shall not allow any inputs other than a positive integer for starting a round of Fizz Buzz.	a	The program outputs “’a’ is not a valid input for Fizz Buzz.”
4	- The program shall output all instances of “Fizz,” “Buzz,” and/or numbers as the current number being iterated over is checked, not at the end of the program. - The program shall add a line break every 10 outputs.	21	The program asks for an input, and outputs the following: “1, 2, Fizz, 4, Buzz, Fizz, 7, 8, Fizz, Buzz, 11, Fizz, 13, 14, Fizz Buzz, 16, 17, Fizz, 19, Buzz, Fizz The game has been completed.”

	- The program shall output a message saying that it has completed the game on a new line after the specified number of rounds has been reached.		
5	The program shall consider an input invalid if it is not a single value.	1 2	The program outputs “’1 2’ is not a valid input for Fizz Buzz”