Given user Story:

The charging current varies during the process of charging. We need to capture the range of current measurements - what range of currents are most often encountered while charging.

Sample**:**

Input: A set of periodic current samples from a charging session, as an array of integers. For example:
`3, 3, 5, 4, 10, 11, 12.`

Functionality: The continuous ranges in there are: 3,4,5 and 10,11,12. The task is to detect the ranges and output the number of readings in each range. In this example, the 3-5 range has 4 readings the 10-12 range has 3 readings.

Output: The expected output would be in comma-separated (csv format):

```
Range, Readings
3-5, 4
10-12, 3
```

Understanding User Story:

Identifying the major Functionalities

Functionality 1 - Validator

Validator checks the input current samples and reject them is they are null or having only one element. The rejected samples is returned with empty array.

Functionality 2 - Sorter

Sorter takes the input array and sorts them in ascending order and gives out the sorted array.

Functionality 3 - Range Detector

Range detector takes the complete array, finds the sub array that continuous range of elements and gives out the list of sub arrays.

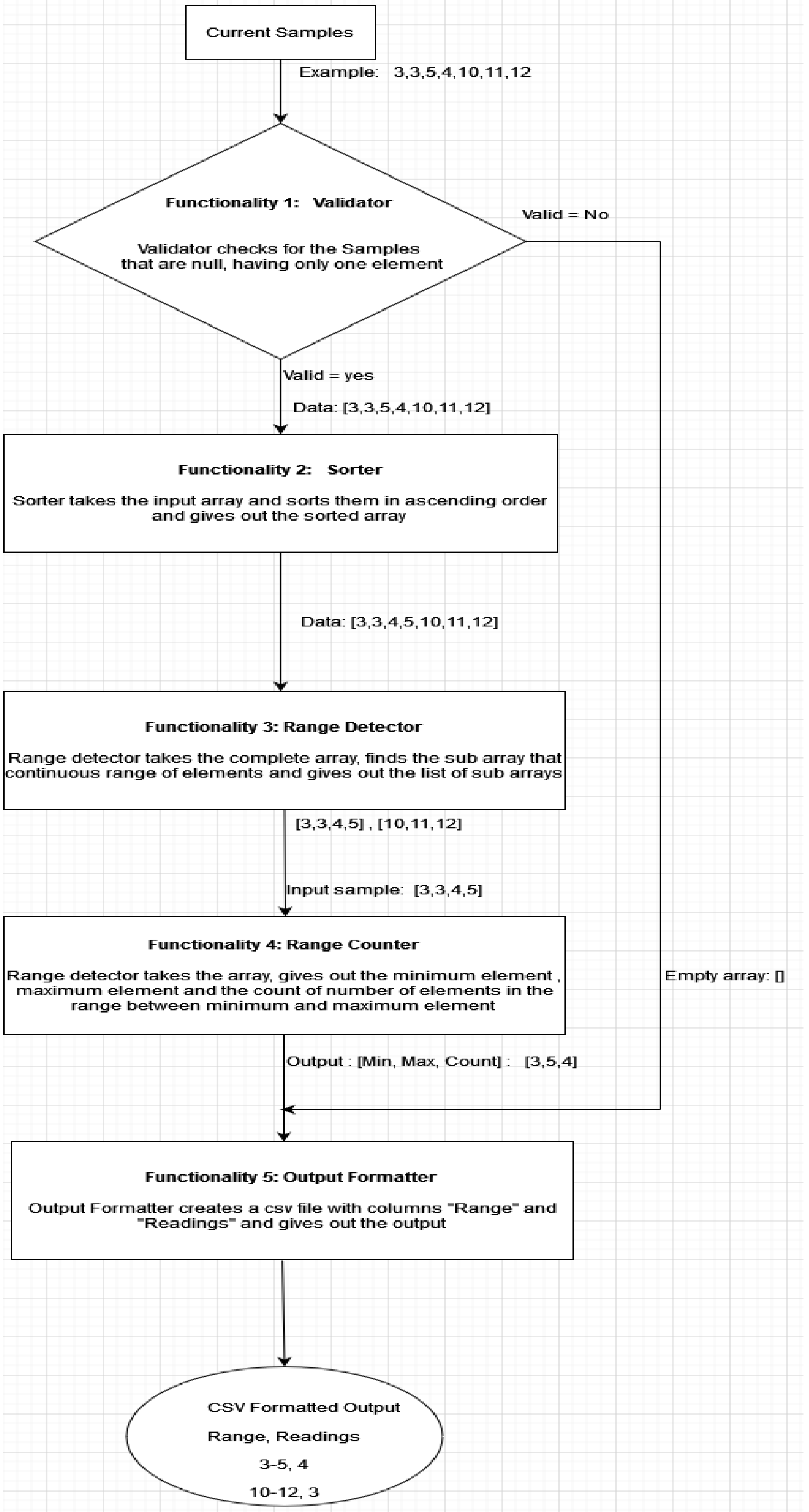
Functionality 4 - Range Counter

Range detector takes the array, gives out the minimum element, maximum element and the count of number of elements in the range between minimum and maximum element

Functionality 5 - Output Formatter

Range detector creates a csv file with columns "Range" and "Readings" and gives out the output.

FUNCTIONALITIES IDENTIFIED WITH DATA FLOW


Current Samples
Example: 3,3,5,4,10,11,12
Functionality 1: Validator
Validator checks for the Samples that are null, having only one element
Valid = No
Valid = yes
Data: [3,3,5,4,10,11,12]
Functionality 2: Sorter
Sorter takes the input array and sorts them in ascending order and gives out the sorted array
Data: [3,3,4,5,10,11,12]
Functionality 3: Range Detector
Range detector takes the complete array, finds the sub array that continuous range of elements and gives out the list of sub arrays
[3,3,4,5] , [10,11,12]
Input sample: [3,3,4,5]
Functionality 4: Range Counter
Range detector takes the array, gives out the minimum element , maximum element and the count of number of elements in the range between minimum and maximum element
Empty array: []
Output : [Min, Max, Count] : [3,5,4]
Functionality 5: Output Formatter
Output Formatter creates a csv file with columns "Range" and "Readings" and gives out the output
CSV Formatted Output
Range, Readings
3-5, 4
10-12, 3

Requirements based on BDD for each functionality

1. **Given** sequence [3,3,5,4,10,11,12] **when** validator is called **then** 1 is expected.
2. **Given** sequence [] **when** validator is called **then** 0 is expected.
3. **Given** sequence [1] **when** validator is called **then** 0 is expected.
4. **Given** sequence [3,3,5,4,10,11,12] **when** sorter is called **then** [3,3,4,5,10,11,12] is expected.
5. **Given** sequence [1,1,6,6,3,3,2,2] **when** sorter is called **then** [1,1,2,2,3,3,6,6] is expected.
6. **Given** sequence [3,3,4,5,10,11,12] **when** range detector is called **then** [3,3,4,5] , [10,11,12]is expected.
7. **Given** sequence [1,1,2,2,3,3,3,7,7,8,8,9] **when** range detector is called **then** [1,1,2,2,3,3,3] , [7,7,8,8,9] is expected.
8. **Given** sequence [1,1,2,2,3,3,3] **when** range counter is called **then** [min,max,count] [1,3,7] is expected.
9. **Given** sequence [3,3,4,5] **when** range counter is called **then** [min,max,count] [3,5,4] is expected.
10. **Given** sequence [3,5,4] **when** Output Formatter is called **then** csv files with column Range as ‘3-5’ and Readings as ‘4’ is expected.