# Overview

Galaxy Merchant Trading application basically a combination of conversion and calculation application with dynamic number format and unit (based on user input). The only predefined format in the application is only for conversion mapping between Roman and Arabic number. But in intergalactic trading, mapping between galaxy number and roman number is based on user input.

# Components

Based on the requirements there are several main functionalities

* Galaxy number to Roman number conversion
* Roman number to Arabic number conversion
* Roman number validator
* Four type of input / query processor:
  + Galaxy number to roman number mapping
  + Galaxy unit (Gold, Silver, Iron, etc.) mapping and calculation
  + Galaxy number calculation
  + Galaxy number with unit calculation for credits
* Multiline text processor which handling input validation

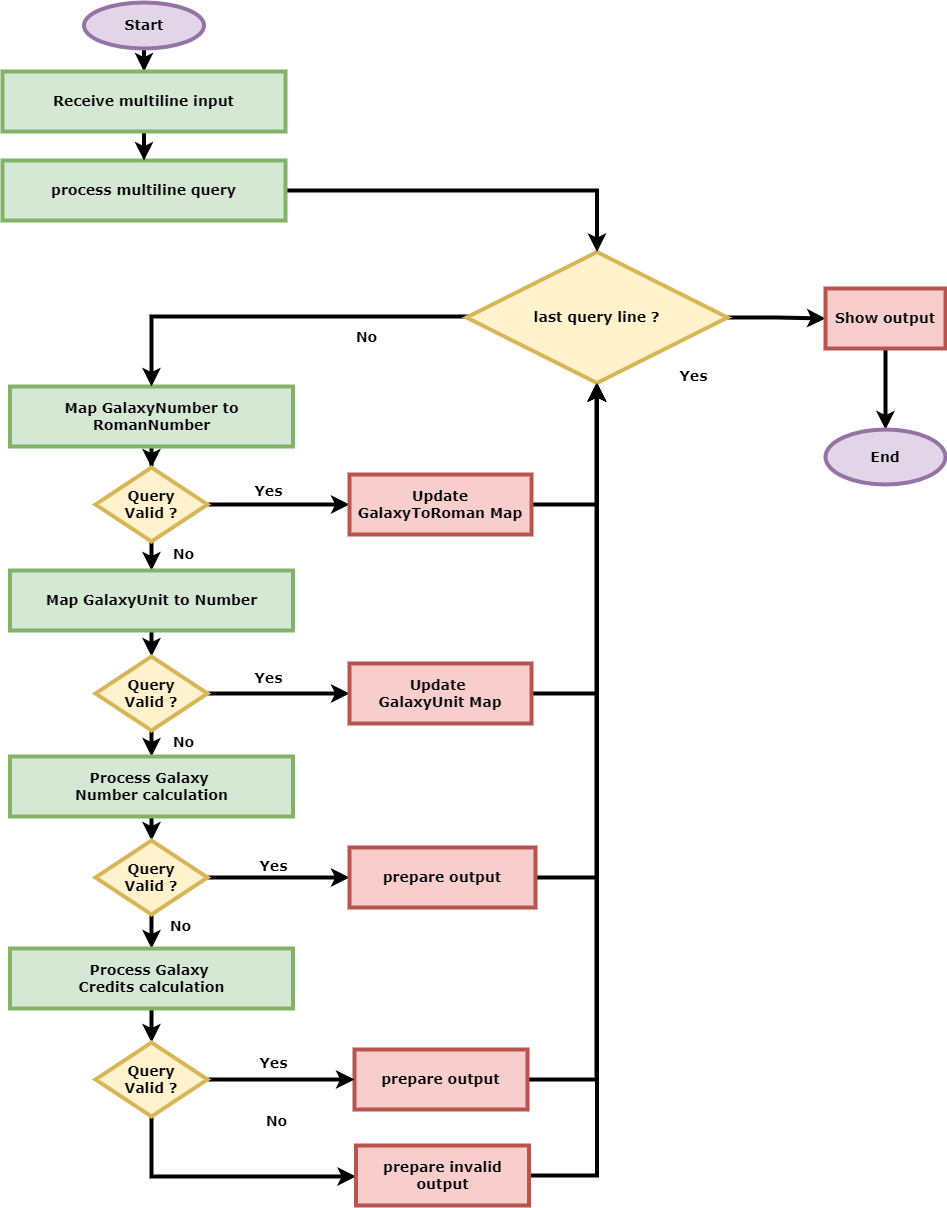
# Implementation

The application is developed using Java 11, Springboot framework and maven dependency management.  
Most of the functionality is created from scratch, some method libraries help for String checking to handle error.

Unit testing also provided to ensure application can handle some scenarios

# Flow Chart

Diagram below describes main processing sequences of the application from receiving input until generate output.



# Run The Application

Below are the steps to running the application. We provide two approaches to running the application

## Run using Java -jar

1. Clone or download the source code from <https://github.com/mbanx/galaxy-trading.git>
2. Ensure maven is installed and configured
3. Change to project root directory
4. Run command: mvn install. It will build the project and download all necessary dependencies
5. Run command java -jar <application.jar>

## Run in IDE

1. Clone or download the source code from <https://github.com/mbanx/galaxy-trading.git>
2. Ensure maven is installed and configured
3. Configure project to download all necessary dependencies
4. You can direction run the main class org.mbanx.challenge.galaxy.Application.java

Once the application started you can access the swagger-ui using browser <http://localhost:8080/swagger-ui.html> and access endpoint POST:/trading/queries and write query on body textbox.  
A picture containing graphical user interface

Description automatically generated

Execute and output will be generatedRectangle

Description automatically generated