Integration test

# Version history

|  |  |  |  |
| --- | --- | --- | --- |
| Editor | Date | Version | Description |
| N/A | 20 Feb 2021 | 1.0.0 | Initial release |
| N/A | 8 Mar 2021 | 2.0.0 | Add test cases for file input |

Table of Contents

[Version history 2](#_Toc66097247)

[Welcome page 6](#_Toc66097248)

[Case: Should be able to show welcome page when the application is launched 6](#_Toc66097249)

[File input 7](#_Toc66097250)

[Case: Should be able to process commands from file first, before waiting for user manual input on console, if file exists 7](#_Toc66097251)

[Case: Should be able to wait for user manual input on console directly, if file is not found in the designated location 9](#_Toc66097252)

[Case: Should be able to process the commands in file input correctly 10](#_Toc66097253)

[Case: Should be able to process the file even if there is empty line at file end 11](#_Toc66097254)

[Case: Should be able to throw error if any of the commands is invalid 12](#_Toc66097255)

[Case: Should be able to rename file after processing 13](#_Toc66097256)

[Manual console input 14](#_Toc66097257)

[Case: Should be able to throw error if user input is empty 14](#_Toc66097258)

[Case: Should be able to throw error if command is invalid 15](#_Toc66097259)

[Case: Should be able to throw error if the PLACE command has invalid format (More than 2 parts including the command) 16](#_Toc66097260)

[Case: Should be able to throw error if the PLACE command has invalid format (More than 3 elements in the part of the params) 17](#_Toc66097261)

[Case: Should be able to throw error if the MOVE command has more than 1 part (it should contain only the command) 18](#_Toc66097262)

[Case: Should be able to throw error if the LEFT command has more than 1 part (it should contain only the command) 19](#_Toc66097263)

[Case: Should be able to throw error if the RIGHT command has more than 1 part (it should contain only the command) 20](#_Toc66097264)

[Case: Should be able to throw error if the REPORT command has more than 1 part (it should contain only the command) 21](#_Toc66097265)

[PLACE command 22](#_Toc66097266)

[Case: Should be able to place the robot on the table 22](#_Toc66097267)

[Case: Should be able to throw error if the x-coordinate param is not an integer 23](#_Toc66097268)

[Case: Should be able to throw error if attempt to place the robot on the table with the x-coordinate < 0 24](#_Toc66097269)

[Case: Should be able to throw error if attempt to place the robot on the table with the x-coordinate > 5 25](#_Toc66097270)

[Case: Should be able to throw error if the y-coordinate param is not an integer 26](#_Toc66097271)

[Case: Should be able to throw error if attempt to place the robot on the table with the y-coordinate < 0 27](#_Toc66097272)

[Case: Should be able to throw error if attempt to place the robot on the table with the y-coordinate > 5 28](#_Toc66097273)

[Case: Should be able to throw error if the facing is invalid 29](#_Toc66097274)

[MOVE command 30](#_Toc66097275)

[Case: Should be able to move robot east 30](#_Toc66097276)

[Case: Should be able to move robot south 31](#_Toc66097277)

[Case: Should be able to move robot west 32](#_Toc66097278)

[Case: Should be able to move robot north 33](#_Toc66097279)

[Case: Should be able to throw error on MOVE command if robot does not have location 34](#_Toc66097280)

[Case: Should be able to throw error if robot will fall off table after moving east 35](#_Toc66097281)

[Case: Should be able to throw error if robot will fall off table after moving south 36](#_Toc66097282)

[Case: Should be able to throw error if robot will fall of table after moving west 37](#_Toc66097283)

[Case: Should be able to throw error if robot will fall of table after moving north 38](#_Toc66097284)

[LEFT command 39](#_Toc66097285)

[Case: Should be able to turn the robot to face north if it is facing east 39](#_Toc66097286)

[Case: Should be able to turn the robot to face west if it is facing north 40](#_Toc66097287)

[Case: Should be able to turn the robot to face south if it is facing west 41](#_Toc66097288)

[Case: Should be able to turn the robot to face east if it is facing south 42](#_Toc66097289)

[Case: Should be able to throw error on LEFT command if robot does not have location 43](#_Toc66097290)

[RIGHT command 44](#_Toc66097291)

[Case: Should be able to turn the robot to face south if it is facing east 44](#_Toc66097292)

[Case: Should be able to turn the robot to face west if it is facing south 45](#_Toc66097293)

[Case: Should be able to turn the robot to face north if it is facing west 46](#_Toc66097294)

[Case: Should be able to turn the robot to face east if it is facing north 47](#_Toc66097295)

[Case: Should be able to throw error on RIGHT command if robot does not have location 48](#_Toc66097296)

[REPORT command 49](#_Toc66097297)

[Case: Should be able to report the robot's location 49](#_Toc66097298)

[Case: Should be able to throw error on REPORT command if robot does not have location 50](#_Toc66097299)

[Logging 51](#_Toc66097300)

[Case: Should be able to log system logs 51](#_Toc66097301)

[Case: Should be able to log error logs related to program run 52](#_Toc66097302)

[Case: Should not show the logs in the console 53](#_Toc66097303)

# Welcome page

## Case: Should be able to show welcome page when the application is launched

Graphical user interface, text

Description automatically generated

# File input

## Case: Should be able to process commands from file first, before waiting for user manual input on console, if file exists

File location

Graphical user interface, text, application

Description automatically generated

File content

Graphical user interface, application

Description automatically generated

Result

Text

Description automatically generated

## Case: Should be able to wait for user manual input on console directly, if file is not found in the designated location

File location

Graphical user interface, text, application

Description automatically generated

Result

Text

Description automatically generated

## Case: Should be able to process the commands in file input correctly

Text

Description automatically generated

## Case: Should be able to process the file even if there is empty line at file end

File content

Graphical user interface, text, application

Description automatically generated

Result

Text

Description automatically generated

## Case: Should be able to throw error if any of the commands is invalid

File content

Graphical user interface, text, application

Description automatically generated

Result

Text

Description automatically generated

## Case: Should be able to rename file after processing

Graphical user interface, text, application

Description automatically generated

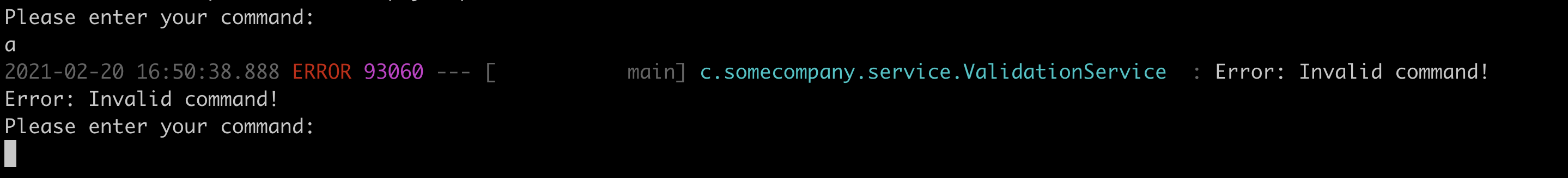
# Manual console input

## Case: Should be able to throw error if user input is empty

Text

Description automatically generated

## Case: Should be able to throw error if command is invalid



## Case: Should be able to throw error if the PLACE command has invalid format (More than 2 parts including the command)

Text

Description automatically generated

## Case: Should be able to throw error if the PLACE command has invalid format (More than 3 elements in the part of the params)

Text

Description automatically generated

## Case: Should be able to throw error if the MOVE command has more than 1 part (it should contain only the command)

Text

Description automatically generated

## Case: Should be able to throw error if the LEFT command has more than 1 part (it should contain only the command)

Text

Description automatically generated

## Case: Should be able to throw error if the RIGHT command has more than 1 part (it should contain only the command)

Graphical user interface, text

Description automatically generated

## Case: Should be able to throw error if the REPORT command has more than 1 part (it should contain only the command)

Graphical user interface, text

Description automatically generated

# PLACE command

## Case: Should be able to place the robot on the table

Graphical user interface, text, application

Description automatically generated

## Case: Should be able to throw error if the x-coordinate param is not an integer

Text

Description automatically generated

## Case: Should be able to throw error if attempt to place the robot on the table with the x-coordinate < 0

Graphical user interface, text

Description automatically generated

## Case: Should be able to throw error if attempt to place the robot on the table with the x-coordinate > 5

Graphical user interface, text

Description automatically generated

## Case: Should be able to throw error if the y-coordinate param is not an integer

Text

Description automatically generated

## Case: Should be able to throw error if attempt to place the robot on the table with the y-coordinate < 0

Graphical user interface, text

Description automatically generated

## Case: Should be able to throw error if attempt to place the robot on the table with the y-coordinate > 5

Graphical user interface, text

Description automatically generated

## Case: Should be able to throw error if the facing is invalid

Text

Description automatically generated

# MOVE command

## Case: Should be able to move robot east

Shape

Description automatically generated with medium confidence

## Case: Should be able to move robot south

Shape

Description automatically generated with medium confidence

## Case: Should be able to move robot west

Shape

Description automatically generated with medium confidence

## Case: Should be able to move robot north

Shape

Description automatically generated with medium confidence

## Case: Should be able to throw error on MOVE command if robot does not have location

Text

Description automatically generated

## Case: Should be able to throw error if robot will fall off table after moving east

Text

Description automatically generated

## Case: Should be able to throw error if robot will fall off table after moving south

Text

Description automatically generated

## Case: Should be able to throw error if robot will fall of table after moving west

Text

Description automatically generated

## Case: Should be able to throw error if robot will fall of table after moving north

Text

Description automatically generated

# LEFT command

## Case: Should be able to turn the robot to face north if it is facing east

Shape

Description automatically generated with medium confidence

## Case: Should be able to turn the robot to face west if it is facing north

Shape

Description automatically generated with medium confidence

## Case: Should be able to turn the robot to face south if it is facing west

Shape

Description automatically generated with medium confidence

## Case: Should be able to turn the robot to face east if it is facing south

Shape

Description automatically generated with medium confidence

## Case: Should be able to throw error on LEFT command if robot does not have location

Text

Description automatically generated

# RIGHT command

## Case: Should be able to turn the robot to face south if it is facing east

Shape

Description automatically generated with medium confidence

## Case: Should be able to turn the robot to face west if it is facing south

Shape

Description automatically generated with medium confidence

## Case: Should be able to turn the robot to face north if it is facing west

Shape

Description automatically generated with medium confidence

## Case: Should be able to turn the robot to face east if it is facing north

Shape

Description automatically generated with medium confidence

## Case: Should be able to throw error on RIGHT command if robot does not have location

Text

Description automatically generated

# REPORT command

## Case: Should be able to report the robot's location

A black background with white text

Description automatically generated with low confidence

## Case: Should be able to throw error on REPORT command if robot does not have location

Text

Description automatically generated

# Logging

## Case: Should be able to log system logs

Graphical user interface, text

Description automatically generated

## Case: Should be able to log error logs related to program run

Graphical user interface, text

Description automatically generated

## Case: Should not show the logs in the console

A picture containing text

Description automatically generated