Project Report for D2

SOEN 6481
Software Systems Requirements Specification
(2019 SUMMER)

ETERNITY: NUMBERS

Zan Wang

Student ID: 27779704
Github Account: YipengZan

Contents

Changes from D1 to D2

1. Remove methods from each class

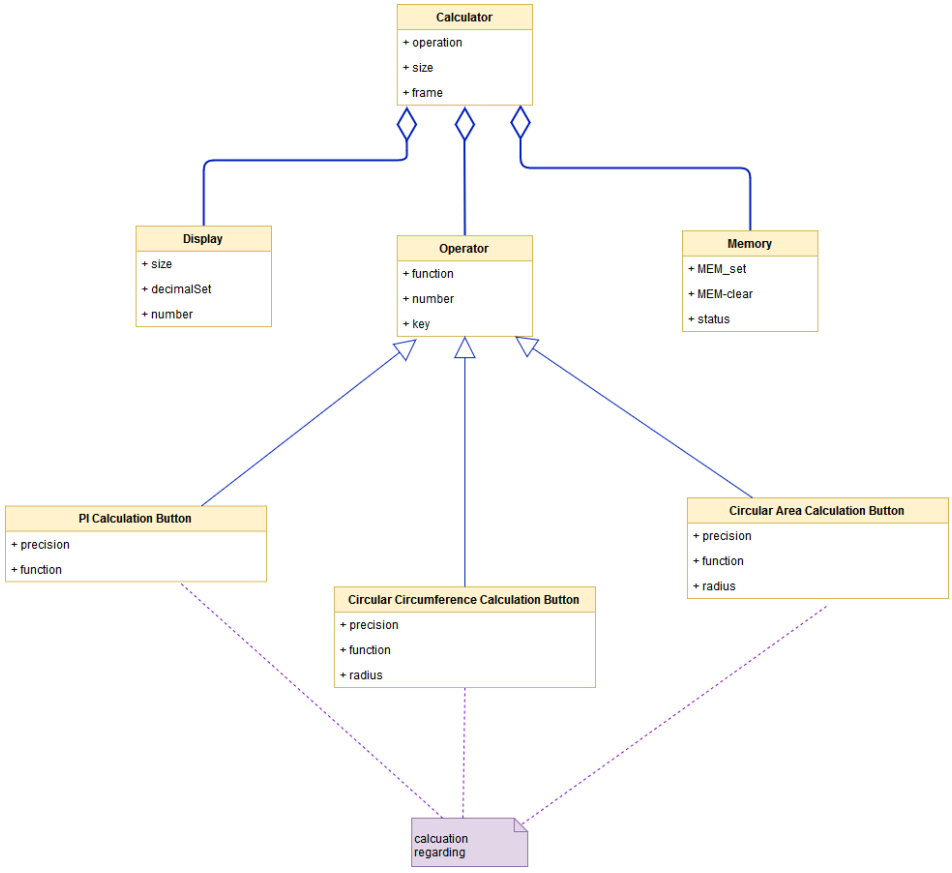

Figure 1: Improved Domain Model

2. Add References to D1:

- in section 4.1: Definition

Personas are actually fictional characters, which we create based upon our research in order to represent the different user types that might use our service, product, site, or brand in a similar way. Creating personas will help us to understand our users' needs, experiences, behaviors and goals [1].

- in section 6.4: Activity Diagram

Activity Diagrams describe how activities are coordinated to provide a calculation results which can be at different levels of abstraction [2].

[1] "Personas – A Simple Introduction", Available: https://www.interaction-design.org/literature/article/personas-why-and-how-you-should-use-them/

[2] "Activity Diagram Tutorial", Available: https://online.visual-paradigm.com/tutorials/activity-diagram-tutorial/

Information on Github Account

Github Account: YipengZan

Folder Path: YipengZan/SOEN-6481

Please also refer to the following screen capture for account and folder information.

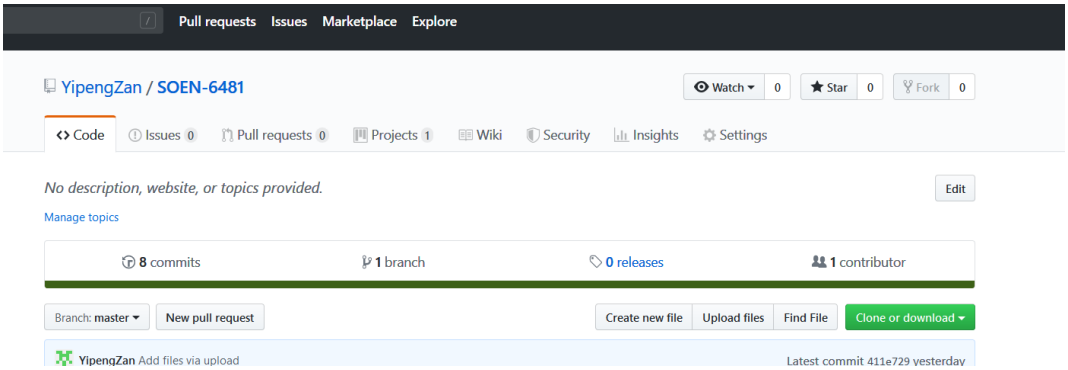

Figure 2: Github Folder Location

Abstract

Along with fast development of human society, it emerges lots of revolution in various territories. The development of traditional regular calculator is just one of them that is under great challenge. People are not only satisfied with the default process to obtain results from using traditional regular calculator, but desire for a more convenient and efficient solution. The widely used solution based on regular calculator with its design of scientific mode, therefore attracts lots of people's attention. Hence, how to validate the scientific number and the scientific mathematical calculation and decent user-device interaction experience become crucial topic for calculator designer. To better serve to this goal here, a simple but efficient calculator design with several specific scientific calculation function with scientific number (PI), is used to model and verify a scientific calculator.

The whole project report for deliver 2 is organized as follows: the section of introduction gives the general topics that the report discusses here; the first chapter, User Stories regarding problem 6, presents the user stories elicited from user's requirements or interview. A matrix is built to map user stories to their acceptance test cases; the second chapter regarding problem 7, shows the traceability matrix for each user story and its related information; the third chapter is Java scratch for problem 8, demonstrating the code implementation work.

Keywords: user stories, traceability, implementation

1. Introduction

In the report of delivery 1, a simple design for a math/scientific calculator similar to most regular calculator in market was introduced. It supports functions for both regular calculation and scientific number calculation regarding PI. Enter a formula by typing on the keyboard. Type in the formula as you would say it, then press "=" button.

In this report for deliver 2, we are going to discuss three topics: user stories, backward traceability matrix and function implementation by coding.

1.1 User Stories

A user story is usually the simplest possible requirement and is about one and only one functionality. It is also is a requirement for any functionality or feature which is written down in one or two lines and max up to 5 lines [1]. An acceptance criterion is a set of accepted conditions or business rules which the functionality or feature should satisfy and meet, in order to be accepted by the Product Owner/Stakeholders.

1.2 Traceability Matrix

Traceability Matrix helps to ensure that all the requirements will be covered in the tests. Traceability Matrix is usually in tabular format as it holds multiple to and fro relationships between requirements and test cases. It is a document which is used to validate that all the requirements are linked to test cases [2].

1.3 Implementation

Functions derived from each use case are implemented through coding by Java.

2. User Stories: Problem 6

2.1 Definition

User stories reflect what a particular class of user needs and the value to be gained. They describe pieces of functionality from a user's point of view, expressed in a solid, compact way [3]. Each user story will have one or more acceptance tests. These tests typically cover a high level test scenario [4].

2.2 User Stories Description

Here, based on the perspective of maintainability or usability, 8 user stories are elicited from different sources which will be demonstrated via traceabilibty matrix in next chapter. Each use story has its own identifier, clear atomic statement, acceptance criteria, usability and maintainability related constrains, both MoSCoW and scale style priority indicators, and estimation in story points.

User Story Identifier	US-1
User Story Statement	As a calculator user I must to find one exclusive button for PI so that I can use PI for various mathematical calculation directly.
Constrains	1. The decimal digits of PI value and iteration of calculation must be able to control and correct. 2. System response time for calculating PI must $<$ 500ms. 3. Any specific button pressed from user end is case sensitive.
Acceptance Criteria	1. User can turn on calculator by pressing "start" button. 2. The cursor will shine on the screen waiting for interaction from user. 3. Then user can press "PI" button. 4. The approximate PI value should be displayed on screen.
Priority	High
Estimate (in Story Points)	3

Table 2.1: User Story 1

User Story Identifier	US-2
User Story Statement	As a calculator user I must have a quick button for circular area calculation so that I can reach an area of a specific circular by giving its radius number only.
Constrains	1. The circular area value must correct to fourteen digit places. 2. System response time for calculating area must $<$ 700ms. 3. Any specific button pressed from user end is case sensitive.
Acceptance Criteria	1. User can turn on calculator by pressing "start" button. 2. The cursor will shine on the screen waiting for interaction from user. 3. User choose to press "shift" button to switch to scientific calculation mode. 4. Then user can input the radius value for the target circular. 5. User can hit "area" button to get the result of circular area calculation. 6. The area of the target circular should be displayed on screen.
Priority	High
Estimate (in Story Points)	5

Table 2.2: User Story 2

User Story Identifier	US-3
User Story Statement	As a calculator user I must have a quick button for circular circumference calculation so that I can reach a circumference of a specific circular by giving its radius number only.
Constrains	1. The circular area value must correct to fourteen digit places. 2. System response time for calculating area must $<$ 700ms. 3. Any specific button pressed from user end is case sensitive.
Acceptance Criteria	1. User can turn on calculator by pressing "start" button. 2. The cursor should shine on the screen waiting for interaction from user. 3. Then user can press "shift" button to switch to scientific calculation mode. 4. Then user can input the radius value for the target circular. 5. User can hit "ccf" button to get the result of circular circumference calculation. 6. The circumference of the target circular should be displayed on screen.
Priority	High
Estimate (in Story Points)	5

Table 2.3: User Story 3

User Story Identifier	US-4
User Story Statement	As a calculator user I should find one exclusive button for historical results calculated previously so that I can use them for the on-going calculation directly.
Constrains	1. The memory should be able to store 20 records automatic update to the latest 10 results up to now. 2. All stored results must correct to fourteen digit places. 3. System response time for calculating area must $<$ 300ms. 4. Any specific button pressed from user end is case sensitive.
Acceptance Criteria	1. User can turn on calculator by pressing "start" button. 2. Cursor should shine on the screen waiting for interaction from user. 3. Then user will inject at least 10 results in calculator memory. 4. User can press "RC" button. 5. Last 10 historical results should be displayed on screen.
Priority	Medium
Estimate (in Story Points)	5

Table 2.4: User Story 4

User Story Identifier	US-5
User Story Statement	As a calculator user I should find one exclusive button for shifting to from regular calculation to scientific calculation so that I can access the area or the circumference of a circular.
Constrains	1. System response time for calculating area must $<$ 300ms.
Acceptance Criteria	1. User can turn on calculator by pressing "start" button. 2. Cursor should shine on the screen waiting for interaction from user. 3. Then user will press "shift" button to switch to scientific calculation mode. 4. Calculator should wait for user to enter necessary parameter such as radius, to begin calculation.
Priority	Medium
Estimate (in Story Points)	3

Table 2.5: User Story 5

User Story Identifier	US-6
User Story Statement	As a calculator user I must be able to make basic arithmetic calculation once I turn on the calculator so that I can process regular number calculation easily.
Constrains	1. System response time for calculating area must $<$ 700ms.
Acceptance Criteria	1. 1. User can turn on calculator by pressing “start” button. 2. User press any button except for “start”. 3. Control panel of regular calculation mode is created by the application
Priority	High
Estimate (in Story Points)	3

Table 2.6: User Story 6

User Story Identifier	US-7
User Story Statement	As a calculator user I must be able to input regular number directly from calculator under regular number calculation mode so that I can make regular number calculation conveniently.
Constrains	1. System can read valid digit number exclusively
Acceptance Criteria	1. 1. User can turn on calculator by pressing “start” button. 2. User press any button except for “start”. 3. GUI of regular calculation mode is created by application. 4. Cursor should shine on the screen waiting for user to enter regular numbers and operators to begin calculation.
Priority	High
Estimate (in Story Points)	3

Table 2.7: User Story 7

User Story Identifier	US-8
User Story Statement	As a calculator user I must be able to input circular radius directly from calculator under special number calculation mode so that I can have the calculation results for circular area or circular circumference conveniently.
Constrains	1. System can read valid digit number exclusively. 2. Any specific button pressed from user end is case sensitive.
Acceptance Criteria	1. User can turn on calculator by pressing “start” button. 2. Cursor should shine on the screen waiting for interaction from user. 3. Then user press “shift” button to switch to scientific calculation mode. 4. User can input the radius value for the target circular. 5. Calculator should wait for user to press “ccf” or “area” button to complete calculation.
Priority	High
Estimate (in Story Points)	3

Table 2.8: User Story 8

2.3 Acceptance Test

Acceptance Tests are also called Customer Tests and Customer Acceptance Tests. Acceptance tests are first class requirements artifacts because they describe the criteria by which the stakeholders will determine whether the system meets their needs. These are detailed specification of the software application behavior for all scenarios which are present for the current state of software and to find its correctness. It is a formal test conducted to determine whether the software application satisfies its Acceptance Criteria and also help the customers to decide whether to accept the system or not.The customer implements the acceptance tests to check whether the user story is completed and correctly implemented. An application can have many acceptance tests to be sure that the functionality of software work [5].

The test cases for each user stories are listed as follows:

TC-1	Turn on calculator by pressing “start” button
TC-2	Cursor shining on the screen waiting for interaction from user
TC-3	“PI” button for getting PI number properly functions
TC-4	Screen properly displays results and numbers
TC-5	“shift” button properly functions
TC-6	radius value input by users accepted by calculator correctly
TC-7	“area” button for getting circular area properly functions
TC-8	“ccf” button for getting circular circumference properly functions
TC-9	”RC” button for getting 10 historical results properly functions
TC-10	”=” button for getting calculation results properly functions

Table 2.9: Acceptance Test Cases

A acceptance test cases matrix is built as follows to facilitate observation for the dependency between 8 user stories and their corresponding acceptance test cases.

Test Case ID / User Story ID	1	2	3	4	5	6	7	8	9	10
US-1	X	X	X	X						
US-2	X	X		X	X	X	X			
US-3	X	X		X	X	X		X		
US-4	X	X		X					X	
US-5	X	X		X	X					X
US-6	X	X		X						X
US-7	X	X								
US-8	X	X			X	X				

Table 2.10: Test Cases Matrix(Pass:”X”, Fail:”O”)

3. Traceability Matrix: Problem 7

In figure 3.1 and figure 3.2, a backward traceability matrix is built to trace the source for all user stories. Figure 3.2 is a supplement to figure 3.1 showing the detailed information of each traceable source. Figure 3.3 indicates the table built for the reference to the use cases marked in above tow matrices, showing the identifiers for the 8 use cases appear in this report. In terms of interview question details and professional interviewee persona, please refer to Delivery 1.

Source / User Story ID	Use Case	User Story	Interview	Persona	Youtube
US-1					X
US-2			X		
US-3			X		
US-4			X		
US-5				X	
US-6				X	
US-7	X				
US-8		X			

Table 3.1: User Story Source Traceability Matrix

User Story ID	Source Details
US-1	https://blog.csdn.net/xiazdong/article/details/8892105
US-2	Interview Question 3
US-3	Interview Question 3
US-4	Interview Question 3
US-5	Professional Interviewee
US-6	Professional Interviewee
US-7	US-6
US-8	US-5

Table 3.2: Source Details for Traceability

UC-1	Get PI
UC-2	Calculate Circular Area
UC-3	Calculate Circular Circumference
UC-4	Recall Last 10 Results
UC-5	Calculate Special Number
UC-6	Calculate Regular Number
UC-7	Input Regular Number
UC-8	Input Radius Value

Table 3.3: Use Cases ID and Name

4. Implementation: Problem 8

In this chapter, implementation of 6 user stories regarding basic calculator domain (regular calculation, calculation mode shift, last 10 results recall), specific scientific number (PI number calculation), and the practical application of specific scientific Number(circular area calculation, circular circumference calculation) are demonstrated.

Before executing the application, user first uncovers the code package, then find the file named “Main.java”.

Once “Main.java” is correctly run, the program starts to initiate the calculator.

The following description and figures demonstrates the implementation of 6 user stories, namely US-1, US-2, US-3, US-4, US-5 and US-6.

(1) US-6

Figure 4.1 shows the status once calculator enters into regular calculation mode.

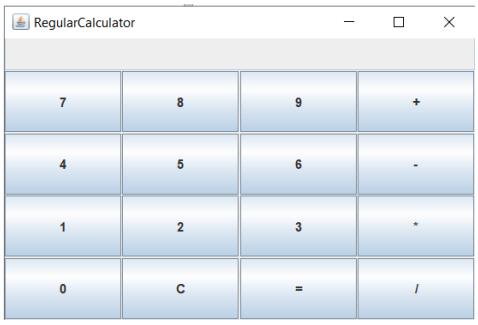

Figure 4.1: Regular Calculation Mode

(2) US-5

User can press ”shift” button to enter scientific mode. Figure 4.2 shows the information provided at the beginning when user steps into scientific mode.

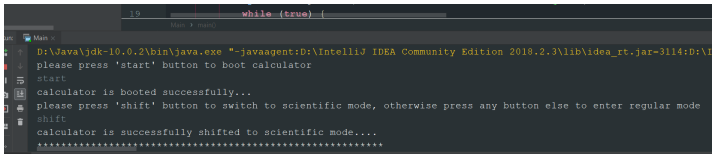

Figure 4.2: Scientific Calculation Mode

(3) US-1

Figure 4.3 indicates that user could get have intended PI value with his/her desired calculation precision and times of iteration in calculation.

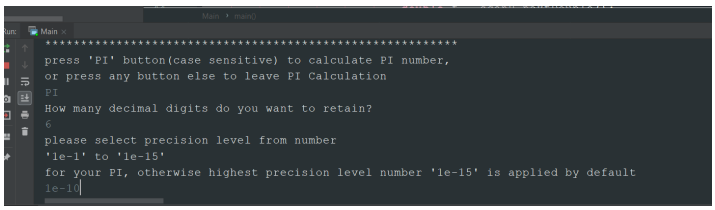

Figure 4.3: PI Calculation

(4) US-3

In scientific mode, user can press “ccf” button to make circumference calculation. Here, the radius of target circular must be input by user, which is indicated by figure 4.4. Alternatively, user could hit any button else to leave circumference calculation

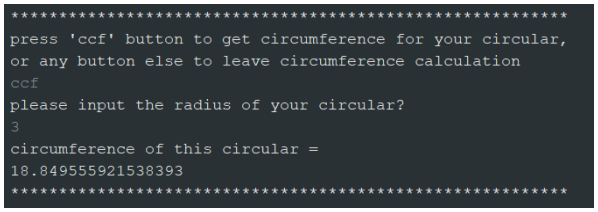

Figure 4.4: Circular Circumference Calculation

(5) US-2

Besides, as you can see in figure 4.5, user can press “area” button to do area calculation under scientific mode. Just as steps aforementioned, user need to provide the radius of target circular for the area calculation, if user wants to skip this part, he/she could press any button else.

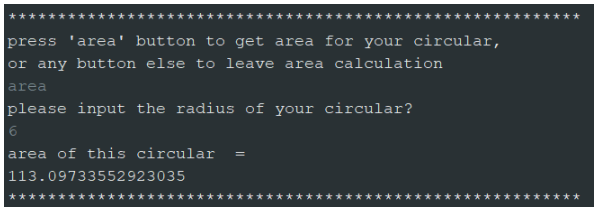

Figure 4.5: Circular Area Calculation

(6) US-4

This calculator could also recollect 10 latest calculation results regarding the assigned scientific number. As the following tow screen snips show, figure 4.6 illustrates the existing 10 results saved in

calculator memory, then, the following figure 4.7 demonstrates the situation when the 11th result comes to memory. We can see that the earliest result removed, the second last result now take the earliest place, and the newly entered result "3.141592653589792" now becomes the latest one.

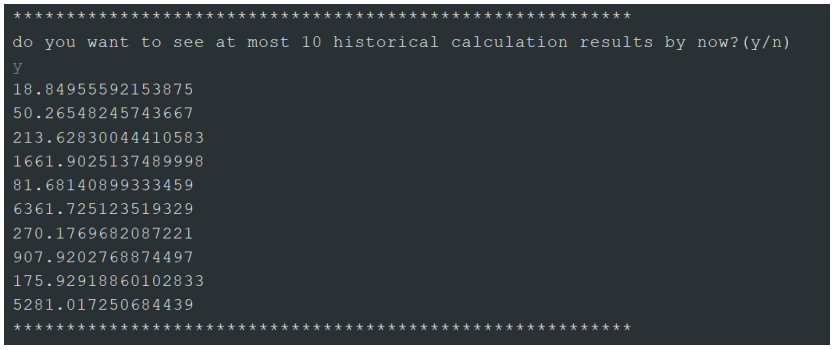

Figure 4.6: 10 Historical Results before Update

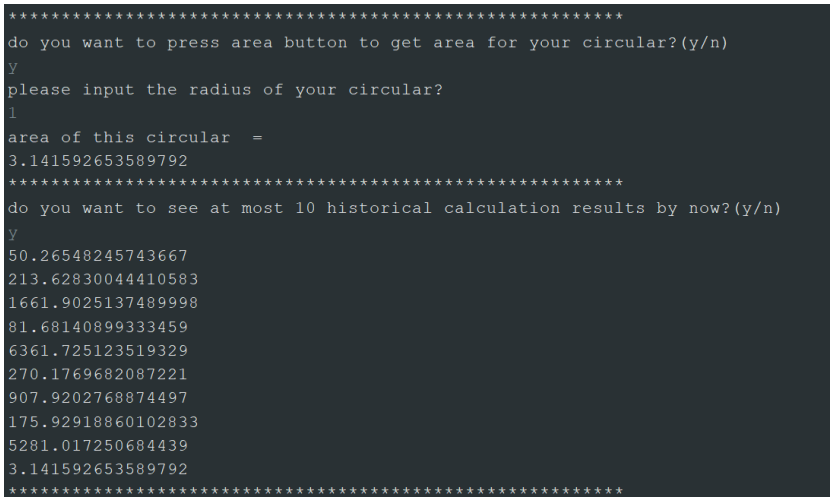

Figure 4.7: 10 Historical Results after Update

The last two figures give you the implementation of unit test Suits for two classes. Test results indicate that all functions passed the designed test cases.

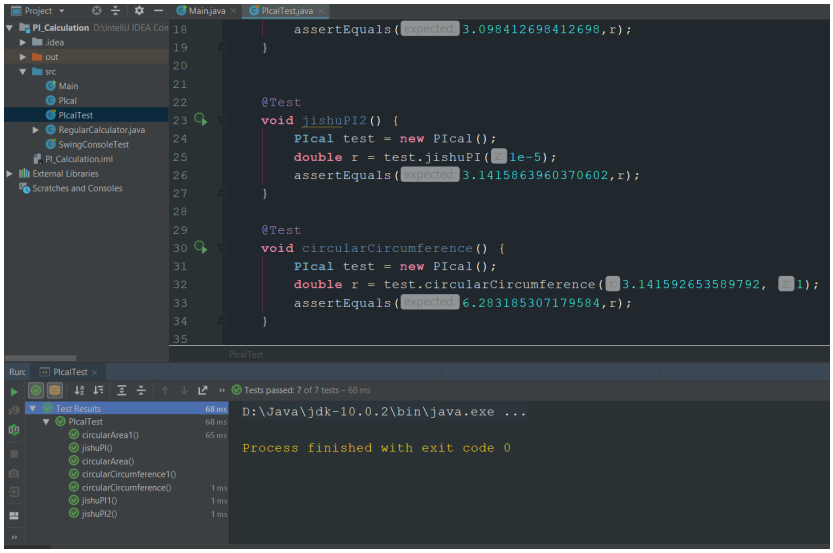

Figure 4.8: Unit Test for Main

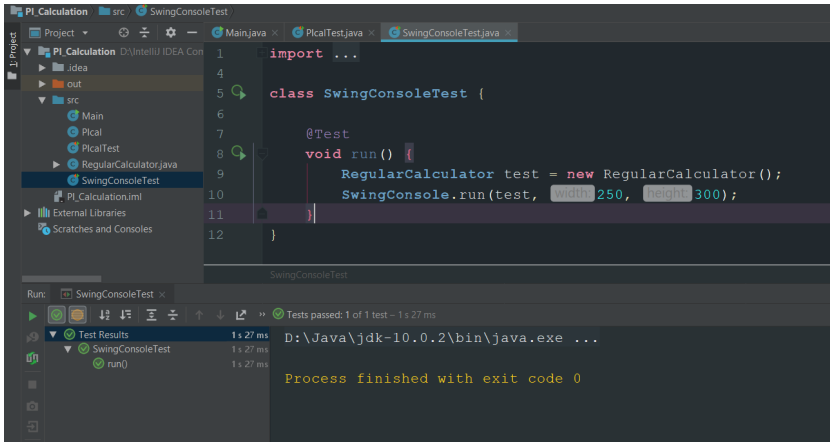

Figure 4.9: Unit Test for UI

5. Conclusion and Future Work

For the designed scientific calculator system demonstrated in this report, its features and functions must be closely comply with each user stories, which puts forwards higher requirements for calculator designer. In this report, the design for assigned scientific number and its relevant practical application in actual life are evidently proved to be gained directly from scratch coding. Apart from this, the design of this calculator is able to aid user in quick switch between regular calculation and scientific calculation, particularly for the assigned scientific number PI in this report.

Notwithstanding, the design of this calculator is still not a flawless tool. User needs to spend time learning specific constrains of it, for example, how to enter scientific mode, how to control the precision of scientific calculation and how to work on the historical results, etc. For this reason, Next stage, advanced improvement on usability and richer applicable functions will be considered to be added. Moreover, further survey and interview will be conducted to modify the interaction experience between users and the device, for example, re-arrange button placement or optimize existing UI.

6. Reference

[1] “What is User Story and Acceptance Criteria”, Available: https://www.softwaretestinghelp.com/user-story-acceptance-criteria/

[2] “What is Requirements Traceability Matrix (RTM)? ”, Available: https://www.guru99.com/traceability-matrix.html

[3] How (and why) to write great User Stories”, Available: https://www.freecodecamp.org/news/how-and-why-to-write-great-user-stories-f5a110668246/

[4] “How to write agile test case requirements”, Available: https://www.getzephyr.com/insights/how-write-agile-test-case-requirements

[5] “Difference between Acceptance Criteria Vs Acceptance Tests”,
Available: http://www.softwaretestingclass.com/difference-between-acceptance-criteria-vs-acceptance-tests/