ETERNITY : NUMBERS

(EULER'S NUMBER)

By Surabhi Surabhi

Department of Computer Science and Software Engineering
Concordia University

July 31, 2019

Deliverable-2

Contents

1 User Stories

A user story is a natural language expression of the requirements. The following user stories have been elicited for the Eternity:Numbers project for the Euler's Number. The structure used for expressing a user story is:
As <**user of the system**>, I want to perform <**some kind of task on system**>, so that i can <**reach/achieve my goal**>

User Story 1

ID:	US 1
Statement:	As a **desktop or laptop user** , I want **a scientific calculator with a graphical user interface**, So that **I can perform day to day calculations**
Acceptance Test:	A calculator application on system with a graphical interface that takes numerical input and gives numerical output
Priority:	Medium
Estimate:	1
Constraint:	The application should run on different operating system for the portability of system

User Story 2

ID:	US 2
Statement	As a **normal user**, I want to **easily input the numerical digits in system**, So that **the calculator can produce output by doing some calculations**
Acceptance Test:	Easy to input the digital numbers
Priority:	Medium
Estimate:	13
Constraint:	The system should accept inputs within the range of Euler's number domain

User Story 3

ID:	US 3
Statement	As a **user**, I want to **perform arithmetic operations** , So that **I can view the result of these operations**
Acceptance Test:	The obtained arithmetic operation results are correct
Priority:	Medium
Estimate:	21
Constraint:	The application should be able to calculate the result in 4 seconds for system efficiency

User Story 4

ID:	US 4
Statement	As a **user**, I want to **input number of iterations**, So that **I can calculate value of** e
Acceptance Test:	User gives integer input for iterations
Priority:	Medium
Estimate:	15
Constraint:	The application should accept only integer values for input

User Story 5

ID:	US 5
Statement	As a **normal user**, I want to **calculate Euler's Number**, So that **I can view the result of** e
Acceptance Test:	Gives correct output for $e$should be displayed on screen When clicking on '='
Priority:	Medium
Estimate:	34
Constraint:	The application should be able to calculate the result in 4 seconds for system efficiency

User Story 6

ID:	US 6
Statement	As a **Student or professional**, I want to **use Euler's Number in other arithmetic calculations** , So that **I can view the result of** **basic arithmetic calculations with use of** e
Acceptance Test:	On getting value of e, get correct results for other calculations within acceptable range
Priority:	High
Estimate:	21
Constraint:	The application should be able to calculate the result in 4 seconds

User Story 7

ID:	US 7
Statement	As a **normal user**, I want to **clear the contents on screen**, so that **previous results are no longer visible**
Acceptance Test:	On clicking "Clear", the screen should be cleared.
Priority:	low
Estimate:	13
Constraint:	For conserving the memory of system

User Story 8

ID:	US 8
Statement	As a **normal user**, I want to **store obtained intermediate results**, so that **I can use the values in other calculations**
Acceptance Test:	Intermediate values results should be of the integer or float type
Priority:	high
Estimate:	34
Constraint:	System should have enough memory to store intermediate results for results usability in future.

User Story 9

ID:	US 9
Statement	As a **Mathematician and developer**, I want to **make sure system accepts only valid input**, so that **it gives error messages on wrong input**
Acceptance Test:	Type checking and range checking
Priority:	Medium
Estimate:	8
Constraint:	System should have some support for error handling in case of wrong input for system robustness

2 Traceability Matrix

Requirements Traceability Matrix is used to setup the links between the requirements and design of the system. It helps us to trace from where a particular requirement emerged in the system that is built. There can be two types of requirements traceability matrix: Forward Traceability Matrix and Backward Traceability Matrix.
For Eternity we used the Backward Traceability Matrix to trace our requirements in the built system to the use case diagram, domain model, interviews conducted with people and other online resources. In the table 'Y' has been used as the placeholder for 'Yes', indicating that the following user story has been elicited form the mentioned source.

User Story	Use Case	Interview	Persona	Domain Model	Online resources
US 1	**Y**	**Y**			**Y**
US 2	**Y**			**Y**	
US 3	**Y**	**Y**		**Y**	
US 4				**Y**	
US 5	**Y**	**Y**		**Y**	
US 6	**Y**	**Y**	**Y**	**Y**	
US 7					**Y**
US 8				**Y**	**Y**
US 9			**Y**	**Y**	

Table 1: Backward Traceability Matrix

3 DVCS and Source code

This is the address for the distributed Version Control system that I am using for this project.
The address for the repository is:
DVCS used: Github
Github Repository address: **surabhigosain6/Eternity-Numbers**
Link for Repository: https://github.com/surabhigosain6/Eternity-Numbers
The source code for the Eternity is also uploaded on the Github repository.

References

[Breitman, Leite, 2002] Managing User Stories. By K. K. Breitman, J. C. S. do Prado Leite. The Tenth International Requirements Engineering Conference (RE 2002). Essen, Germany. September 9-13, 2002.

[Cohn, 2004] User Stories Applied: For Agile Software Development. By M. Cohn. Addison-Wesley. 2004.

[Wikipedia1] https: //en.wikipedia.org/wiki/Traceabilitymatrix

[Wikipedia1] https: //en.wikipedia.org/wiki/Userstory

[Cleland-Huang, Gotel, Zisman, 2012] Software and Systems Traceability. By J. ClelandHuang, O. Gotel, A. Zisman (Editors). Springer-Verlag. 2012.