FEBRUARY 24, 2019

SRS DOCUMENT
SWE 205 PROJECT
PROJECT NAME: MYPAINTSHOP
PROJECT VERSION: 1.0

HASHIM ALGHAMDI (GROUP LEADER)
HAITHAM ALSAEED
HUSSAIN HAJJI
SALEM BAMUKHIER
SALMAN ALGHAMDI
TEAM 3

Table of Contents

1- Introduction

1.1 Purpose of this document

The purpose of this document is to document the process model, functional and non-functional requirements and showcasing UML diagrams and prototype screens of MyPaintShop.

1.2 Introduction and scope of the product

MyPaintShop is a very basic tool that allows the user to draw or pick shapes on a blank canvas and color them. You can also adjust the colors or shapes by changing their shades or size. Filling shapes with color is another available feature in the software. Shapes will display an indication of selection when chosen to be adjusted. The tool's size is not that big and the specs are minimalistic it should run smoothly on any PC.

1.3 Overview of the remainder of the document

As stated, this document will analyze and document all of the product aspects, those aspects will be briefly discussed here to give an overview of the document:

- Section 2 will discuss and analyze the chosen process model this project will follow.
- Section 3 will give a high level, abstract look at the user requirements, stating both functional and non-functional requirements.
- Section 4 will discuss and analyze the system requirements going through the use case diagrams, stating and describing the actors, giving a brief functionality description for each use case. It will, after that, go through and discuss the non-functional requirements as well.
- Section 5 will include the prototype screen for the project, as sketches and simple drawings of the main windows.
- Section 6 will discuss possible future changes and improvements of the product as system evolution.
- Section 7 will conclude all of the work done on this document by stating all the meetings and roles of the members of the team as a percentage or by stating the sections and subsections every member has done.
- Section 8 will include all the references that have been used to aid in writing this document.

2- Process Model

After discussions with our instructor - who is also the project manager and the customer -, we decided to use the Waterfall Model while using UML as the modeling language. This choice came due to multiple of reasons:

- The requirements are clear with no major changes intended in the future.
- UML will help this project to have a good documentation.
- Using other processes is not a wise choice. Incremental Process is not useful since the project is small. Throw-away Prototype is not needed since the requirements are clear. Lastly, no need for RUP since this is a course project not a middle-sized business project.

3- User Requirements

3.1 Functional requirements

- The software shall enable the user to draw or select from preset shapes .
- The software shall allow the user to select different colors.
- The software should allow the user to fill the shapes with colors.
- The software should enable the user to select drawn shapes on the canvas.
- The software may allow the user to change colors or properties of selected shapes by entering values in an editing menu.
- The software may allow the user to delete existing shapes or drawings.
- The software may enable the user to save drawings as a file for future access.

3.2 Non-functional requirements

- The system shall be easy to use with basic and simple user interface.
- The system should offer the opportunity to "Undo" and "Redo".
- The system should contain a user guide to help the user to learn how to use the software.
- The system shall be efficient in using device resources (ram, processor, etc.).
- The system may offer the opportunity to save the result image as (JPEG) extension.
- The system shall offer the opportunity to save the result file as a special extension as modifiable layers.
- The system shall be Maintainable to enable features to be added later.
- The system shall be dependable.
- The system response shall be quick.

4- System Requirements

4.1 Use case diagram

Select
shape

Edit
shape

Draw
free

USER

Fill
shape

Save
drawing

Delete
shape
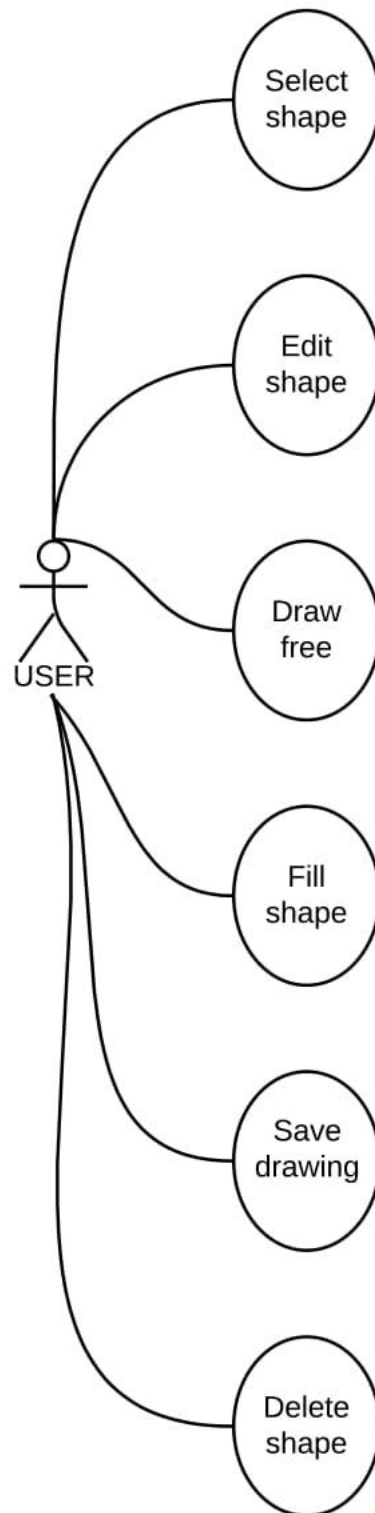
4.2 Actors with their descriptions

- User: He is the customer who will use this program to create, paint, and edit shapes.

4.3 Use case scenarios

- Select preset shapes:
  Once the software is launched, a list of the preset shapes appears beside the canvas, the user, then, can select one of these shapes by clicking on it.

- Select preset colors:
  Once the software is launched, a list of the preset colors appears beside the canvas, the user, then, can select one of these colors by clicking on it.

- Open drawings:
  Once the program is launched, the user can open an existing drawing file, by clicking the "Open" button in the toolbar or opening the file menu and choosing the "Open" option.

- Draw shapes:
  The user selects a shape and a color by clicking on the preset options, and then drags the mouse on the canvas while keep pressing the left click.

- Select drawn shapes:
  Provided that there is a drawn shape on the canvas, the user can select the shape by clicking it. The selected shape then would be rendered to indicate that it has been selected.

- Fill-in colors:
  After a shape has been selected, the user can fill the color inside the shape by clicking the "Fill" button in the toolbar or by opening the shapes menu and choosing the "Fill" option.

- Change drawn shapes properties:
  After a shape has been selected, a window in the bottom of the canvas would open, where the user can modify the color and/or dimensions of the shape, by entering values of new dimensions or selecting different color from the preset color panel.

- Delete shapes:
  After a shape has been selected, the user can delete it by clicking the "Delete" icon in the toolbar, or by opening the shapes menu and choosing the "delete" option.

- Save drawings:
  Once the user finishes their drawing, he/she can save their work by clicking the "Save" button in the toolbar or opening the file menu and choosing the "Save" option.

4.4 Non-functional requirements

- The software shall have minimalistic and intuitive interface.
- The training time shall not exceed thirty minutes.
- The software should be maintainable for additional features.
- The software may have help frames.
- The software shall be developed using Java language.
- The development process shall follow the waterfall model.
- UML shall be used as a modeling language.
- The software should be delivered in two calendar months.

5- Prototype Screens
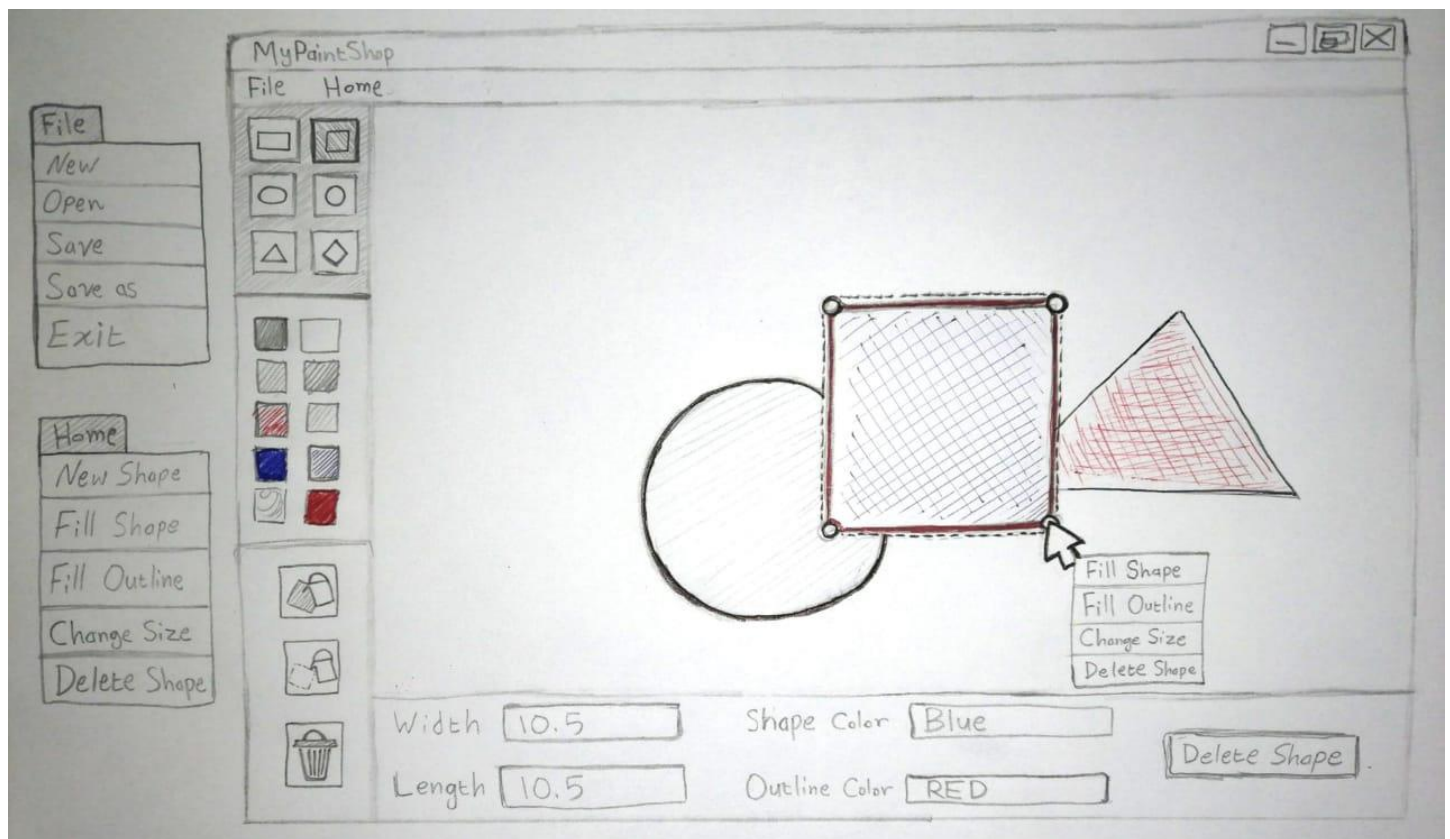
6- System Evolution

In future builds and releases of the software new features and functionalities will be added to improve the software and what it serves, here is the planned-for features for the software:

- The system shall allow the user to share his/her drawings as images to other platforms.
- The system shall support drawing using Drawing Tablets connected to it.
- The system shall be able to import external images to draw on and edit.
- The system shall include different kinds of brushes to use.
- The system should have an option to cover the canvas with a grid.
- The system may support 3D modeling.
- The system may support online dual-drawing on the same canvas.

7- Conclusion

Meetings:

Date	Time Duration	Members who attended
February 15, 2019	1 hour	Hashim Alghamdi (Group Leader) Hussain Hajji Haitham Alsaeed Salem Bamukhier Salman Alghamdi
February 20, 2019	40 minutes	Hashim Alghamdi (Group Leader) Hussain Hajji Haitham Alsaeed Salem Bamukhier Salman Alghamdi

Tasks:

Member	Tasks Performed	Percentage
Hashim Alghamdi (Group leader)	Cover page, TOC, 1.2, 3.1,7,8	18%
Hussain Hajji	1.3,4.1,6	22%
Haitham Alsaeed	3.2,1.1	17%
Salem Bamukhier	4.3,4.4	21%
Salman Alghamdi	2,4.2,5	22%

8- References

- Tool used for UML diagram: https://www.lucidchart.com
- Sommerville, I. (2016). *Software engineering*. Harlow: Pearson Education.
- http://mathinary.com
- http://mathopenref.com

9- Appendices

We can add three other features when clicking on an object, like: Circumference, Area and Object's Name.
When the object is selected, the program will take the length and width of the object to use it for calculating the Area and the Circumference. Each object has different equations and calculations, which will help determining the name of the object.
If the object selected was a rectangle, the system will take the length as x and width as y, then it will apply $A = xy$ to calculate the Area and $C = 2(x + y)$ to calculate the Circumference. For the square, the process is quite similar, $A = l^2$ and $C = 4l$ since the length and width have equal values.
For the ellipse, the system will take the major axis (which is the longest radius) as a and the minor axis (which is the shortest radius) as b then the system will calculate the Area as $A = \pi ab$ and the Circumference as $C = \pi(a + b)$. Since the circle is a special case of an ellipse, the system will take the radius as r then will calculate the Area as $A = \pi r^2$ and the Circumference as $C = 2\pi r$.
For the equilateral triangle, the system will take the height as h and the base as b then will calculate the Area as $A = \frac{1}{2}hb$ and the Circumference as $C = 3b$. if the triangle wasn't equilateral, then the main base will be a and the height will be b and the hypotenuse as c, the system will take a and b to calculate c as follows: $c = \sqrt{a^2 + b^2}$, then the system will use the three values to calculate the Circumference as $C = a + b + c$.