*---- Fall 2017 CS1C : 2D\_Modeler Project, User Stories ----*

Team: Team to be Named (Maximum Override, Coding Comrades)

Contributors: Connor, Yuliy, Mike, Roman

Baseline story point estimate is 1:1 (point : hours)

**Total Current Estimate:** 109

ID#4 As a user, I can save all changes from session to session

a) Description:

A program user should have the option to save progress. The user should be informed of

the location of the saved file and prompted to overwrite a pre-existing file. The history of

saved or overwritten files will be stored and accessible by the admins.

b) Tasks:

As an admin I can see the history of save files by regular users and other admins.

c) Tests:

Verify that users can save and overwrite files.

Verify that the log creates an entry with the date and time of a new save.

Verify that admins can save and overwrite files.

Verify that admins can see the save log.

Verify that the save location is visible to both

d) Assignee:

Connor, implement the save, overwrite, and display location methods

Yuliy, implement class to create a log of save files and test its visibility to make sure it

can only be accessed by admins.

e) Estimation: 15

f) Priority: 3

g) Done:

User is able to save and overwrite files

User is able to see the save location

Admin is able to save and overwrite files

Admin is able to see the save location

Admin is able to see the save log

All task, test conditions satisfied

Roman

ID: 5 - Have objects represented on a display plane with the following: minimum 1000 x 500

pixels (Horizontal x Vertical);top left of plane is origin (0,0), bottom right will be (1000,500)

a) Description:

* The user can only see shapes that are on the display plane (1000x500)
* If the shape is half on the display plane and half off, the shape will be clipped. Only half will be rendered.

b) Tasks:

* Only render a shape if it is on the display plane.
* (Possibly: Scale the shape to fit on the display plane)

c) Tests:

* Verify if a shape is on the display plane. If so, check its position relative to the boundaries of the display plane.
* Verify that the shape will not be drawn if it is off the display plane.

d) Assignee:

* Roman, implement the canvas (display plane) for the application. Check if a shape is on or off the canvas.
* Yuliy, verify if the shape draws appropriately in accordance to the position of the shape.

e) Estimation: 5

f) Priority: 3

g) Done:

* User is able to position shapes where if it were to be half way off the canvas, it will only display/render the side that is on the canvas.

Roman

ID: 8 - Generate a list of shapes which have an area, sorted by area

a) Description:

* The user can access a list which displays every shape that has an area (box, rect, circle) and have that list sorted from smallest area to largest area by default.
* The user can flip the list so the order goes from largest to smallest.

b) Tasks:

* When the user wants to see the list of shapes based on area, a search command must take place in order to get all of the shapes on the canvas. Then the list of shapes must be sorted based on area.
* The user must press a button that will flip the list after is has been sorted by area.

c) Tests:

* Verify if the list gets populated after placing a couple of shapes on the canvas.
* Verify if the list gets sorted once it gets populated with shapes.
* Verify that the list’s default sort is area from smallest to largest.
* Verify that the button re-sorts the list from largest area to smallest area and vis versa.

d) Assignee:

* Michael, identify all shapes that are on the canvas for generating the list. Store all shapes that have an area into a list.
* Roman, display the list of shapes with an area and add the sort button and functionality.

e) Estimation: 10

f) Priority: 4

g) Done:

* User is able to view a list of shapes with an area and sort them based on area, either in ascending order or descending order.

Roman

ID: 9 - Generate a list of shapes which have a perimeter, sorted by perimeter

a) Description:

* The user can access a list which displays every shape that has an area (box, rect, circle) and have that list sorted from smallest perimeter to largest perimeter by default.
* The user can flip the list so the order goes from largest to smallest.

b) Tasks:

* When the user wants to see the list of shapes based on perimeter, a search command must take place in order to get all of the shapes on the canvas. Then the list of shapes must be sorted based on perimeter .
* The user must press a button that will flip the list after is has been sorted by perimeter.

c) Tests:

* Verify if the list gets populated after placing a couple of shapes on the canvas.
* Verify if the list gets sorted once it gets populated with shapes.
* Verify that the list’s default sort is perimeter from smallest to largest.
* Verify that the button re-sorts the list from largest perimeter to smallest perimeter and vis versa.

d) Assignee:

* Michael, identify all shapes that are on the canvas for generating the list. Store all shapes that have a perimeter into a list.
* Roman, display the list of shapes with a perimeter and add the sort button and functionality.

e) Estimation: 5

f) Priority: 4

g) Done:

* User is able to view a list of shapes with a perimeter and sort them based on perimeter, either in ascending order or descending order.

ID#10 - Yuliy As a user, I can have an initial database of shapes read in from a file at the start of

the program.

a) Description:

A program user can have an initial database of shapes read in from a file at the start of

the program.

b) Tasks:

A user can see view the initial database of shapes.

c) Tests:

Verify that users have viewing access only to the database of shapes.

d) Assignee:

James, implement front end UI and code behind logic, member class data & operations

reservation cancellation

Yuliy, implement class to email confirmation to user

e) Estimation: 5

f) Priority: 4

g) Done:

User can view the database of shapes, but cannot add, remove, edit or move any shapes.

(Connor) ID: 6 As a user, I can log in and as a guest or as an administrator to access a

database of shapes via read-in of a .txt file at the start of program compilation if

user has admin privileges.

a) Description:

A user with administrator permissions would enter a series of data which would write the

desired shape, dimensions, position, and orientation of a shape to the internal .txt file of the

program during runtime. New shapes entered would be promptly displayed directly after

entering desired data. The shapes available to the user would be: line, polyline, polygon,

rectangle, ellipse, and ‘text’ shape.

b) Tasks:

As an administrator I can add, remove, move, change color, and change line size of a

shape object.

As a guest i can demo the program and its features, but without functionality.

c) Tests:

Verify that an administrator can create and remove a shape object with the correct

dimensions and display immediately.

Verify that an administrator can move/ rotate or change color/ line size of a shape object

that already exists.

Verify that an administrator can enter the incorrect data for a shape and recieve an error

message.

Verify that all changes are saved to the internal .txt file and read to the main program

compiles through such file.

Verify that a guest cannot edit the .txt file or the program in any kind.

d) Assignee: Michael

e) Estimation: 15

f) Priority: 4

g) Done:

User is able to login as a guest and view the program without full functionality.

Administrator is able to add/ remove, change color/ line size, and move/ rotate a shape

object.

All changes are saved to an internal .txt file and read to the main program during

compilation.

All task, test conditions satisfied

ID#7 - Yuliy. As an admin, I am able to login with the proper credentials and have an initial

database of users read in from a file

a) Description:

An administrator needs to be able login with their correct credentials and have access to

the database of users.

b) Tasks:

An admin need to be able to successfully login with their login name and password.

An admin needs to see the database of users that includes their login names and access

levels (admin, user).

c) Tests:

Verify that the program notifies the admin if the login is entered incorrectly.

Verify that the program notifies the admin if the password is entered incorrectly.

Verify that an admin can have access to the current database of users.

d) Assignee:

Yuliy can implement the password and login verification as well as error output statements.

Roman can implement the view database method.

e) Estimation: 20

f) Priority: 4

g) Done:

Admin is able to login using their username(login) and password.

Admin is able to view the entirety of user(admins included) data base.

(Connor) ID: 15 As a administrator, I can login to an admin account and edit the programs

shapes and accounts.

a) Description:

A user with administrator permissions would be able to login to their account and edit

other shapes and accounts. An admin can add/ remove or edit other accounts and use

the program to its full functionality.

b) Tasks:

As an administrator I can add, remove, move, change color, and change line size of a

shape object.

As an administrator I can add, remove, or change accounts of any kind in the programs

database.

c) Tests:

Verify that an administrator can create and remove a shape object with the correct

dimensions and display immediately.

Verify that an administrator can move/ rotate or change color/ line size of a shape object

that already exists.

Verify that an administrator can enter the incorrect data for a shape and recieve an error

message.

Verify that an administrator can edit an existing accounts information.

Verify that an administrator can add or delete an account to the database.

Verify that an administrator cannot delete or edit an account that does not exist.

Verify that a guest cannot edit the .txt file of the program or any existing account

information.

d) Assignee: Connor

e) Estimation: 6

f) Priority: 4

g) Done:

User is able to login as an admin to add/ remove, change color/ line size, and

move/ rotate a shape object.

User is able to login as an admin to add/ remove or edit account information.

All task, test conditions satisfied

ID#17 - Yuliy As a user, I can access a help feature to see how the program is properly utilized,

and what features are available to me.

a) Description:

Any program user can utilize the help function to see how the program is properly

utilized, including a description and demonstration/images of admin features.

b) Tasks:

As a user I can access

c) Tests:

Verify that the command calling the help function works

Verify that the user can navigate the help window

Verify that text descriptions of program functions are present.

Verify that images are present and open successfully together with the text of the help

feature.

d) Assignee:

Yuliy can document every feature of the program and provide images for the help file.

Michael can code the navigation menu of the help feature.

e) Estimation: 7

f) Priority: 5

g) Done:

Both users and admins can successfully call the help feature of the program.

They can view the text files and descriptions of every feature of the program.

They can navigate different sections of the help manual.

(Connor) ID: 11 As a user, I can log in and as an administrator to change existing shapes via

read-in of a .txt file during program compilation if user has admin privileges.

a) Description:

A user with administrator permissions would edit the attributes of an existing shape by

entering a series of new data for a certain selected shape. This would change the color and

border size of a shape to the internal .txt file of the program during runtime. Changes would

be promptly displayed directly after entering new desired data.

b) Tasks:

As an administrator I can change line size and color of a shape object.

As a guest I can demo the program and these features, but without the necessary

functionalities.

c) Tests:

Verify that an administrator can change color/ line size of a shape object that already

exists and display the changes immediately.

Verify that an administrator can enter the incorrect data for a shape and recieve an error

message.

Verify that all changes are saved to the internal .txt file and read to the main program

compiles through such file.

Verify that a guest cannot edit shape attributes in the .txt file for the program in any kind.

d) Assignee: Michael

e) Estimation: 6

f) Priority: 4

g) Done:

User is able to login as a guest and view existing shapes without ability to edit them.

Administrator is able to change color/ line size of a shape object.

All changes are saved to an internal .txt file and read to the main program during

compilation and displayed immediately.

All task, test conditions satisfied

(Connor) ID: 12 As a user, I can log in and as an administrator to add or remove shapes via

read-in of a .txt file during program compilation if user has admin privileges.

a) Description:

A user with administrator permissions would add a new shape by entering the dimensions,

position, orientation, and color/border size for a certain selected shape. This would save to

the internal .txt file of the program during runtime. Changes would be promptly displayed

directly after entering new desired data. An admin could also delete existing shapes during

runtime.

b) Tasks:

As an administrator I can create a new shape of choice by entering the position,

dimensions, orientation, color, and border size.

As an administrator I can remove an existing shape object of choice.

As a guest I can demo the program and these features, but without the necessary

functionalities.

c) Tests:

Verify that an administrator can add a new shape to the program and have it visible right

after entering data.

Verify that an administrator can enter the incorrect data for a shape and recieve an error

message.

Verify that all changes are saved to the internal .txt file and read to the main program

compiles through such file.

Verify that an administrator could remove as many existing shapes as desired.

Verify that a guest cannot edit shape attributes in the .txt file for the program in any kind.

d) Assignee: Connor

e) Estimation: 6

f) Priority: 4

g) Done:

User is able to login as a guest and view existing shapes without ability to edit them.

Administrator is able to add a new shape object and view it successfully.

Administrator can remove existing shapes.

All changes are saved to an internal .txt file and read to the main program during

compilation and displayed immediately.

All task, test conditions satisfied

(Connor) ID: 13 As a user, I can log in and as an administrator to move or rotate existing shapes

via read-in of a .txt file during program compilation if user has admin privileges.

a) Description:

A user with administrator permissions would edit the attributes of an existing shape by

entering a series of new data for a certain selected shape. This would move or rotate the

data of a shape to the internal .txt file of the program during runtime. Changes would be

promptly displayed directly after entering new desired data.

b) Tasks:

As an administrator I can change the position or orientation of a shape object.

As a guest I can demo the program and these features, but without the necessary

functionalities.

c) Tests:

Verify that an administrator can change orientation of a shape object that already exists

and display the changes immediately.

Verify that an administrator can change the position of a shape object that already exists

and display the changes immediately.

Verify that an administrator can enter the incorrect data for a shape and recieve an error

message.

Verify that all changes are saved to the internal .txt file and read to the main program

compiles through such file.

Verify that a guest cannot edit shape attributes in the .txt file for the program in any kind.

d) Assignee: Roman

e) Estimation: 6

f) Priority: 4

g) Done:

User is able to login as a guest and view existing shapes without ability to edit them.

Administrator is able to change orientation and position of a shape object.

All changes are saved to an internal .txt file and read to the main program during

compilation and displayed immediately.

All task, test conditions satisfied

ID#18 - Yuliy. As a user, I can save all changes from session to session

a) Description:

This feature will allow users and admins to submit testimonials and suggestions for the

program.

b) Tasks:

As a user I can write and submit potential changes

As an admin I can write and submit potential changes

c) Tests:

Verify that a user can write and submit text suggestions.

Verify that an admin can write and submit text suggestions.

d) Assignee:

Yuliy can code the input function and the prompt.

e) Estimation: 1

f) Priority: 5

g) Done:

All users can submit feedback.

ID#14 - (Mike) As a user, I want to be able to view contact information for the developer.

a) Description: The program will include an entry containing contact information for the

Developer which will be accessible for any user.

b) Tasks: I want the information to be readily available.

I need to be able to contact the developer by telephone, email, or postage.

c) Tests: Verify that the information is easily found.

Verify that the information provided is correct.

d) Assignee: Michael

e) Estimation: 2

f) Priority: 5

g) Done: The user can access the provided contact information.

ID#16 (Mike) As a user, I should be treated as a guest by the program, providing me with

limited privileges.

a) Description: By default, a user will operate the program with guest credentials, which will

limit the available operations that they can perform; in general, they will be

unable to alter the existing state of the database.

b) Tasks: As a guest I should be able to view the existing shapes stored in the database.

As a guest I should be able to view contact and help information.

As a guest I should be able to view the financial/maintenance information.

As a guest I should be able to view the sales pitch.

As a guest I should not be able to move shapes.

As a guest I should not be able to add/remove shapes.

As a guest I should not be able to modify attributes of shapes.

c) Tests: Verify that the unverified users are treated as guest accounts.

Verify that the guest account is able to access and draw the shapes in the database.

Verify that the guest account is able to access and view the relevant information.

Verify that the guest account is not able to modify existing shapes nor add/remove.

d) Assignee: Michael

e) Estimation: 6

f) Priority: 4

g) Done: The user is automatically treated as a guest account, and has access only to the

information readily available to them; they are not capable of modifying existing

shapes.