TEAM MITTENS USA

User Stories

October 30, 2018

#1 – As an admin, I want to change shape parameters - *Brian*

a) Description:
An administrator needs to be able to change the parameters of the different shapes.
These changes are saved to be accessed by users.
This power is only available when logged in as an administrator.

b) Tasks:
Create different user accounts (user, administrator).
Have a login screen where the user can log in as an administrator. Users should be able to log out and back in as a different user.
Create an administrator-only option that allows the administrator to choose which shape to edit, and then which parameter of that shape to edit.

c) Tests:
Verify that an administrator can log in.
Verify that administrator actions are only available when logged in as an administrator.
Verify that the appropriate shape parameters are edited and saved.

d) Assignee:
Brian & Mariah:
Implement different user accounts - User and Administrator.
Implement a login screen.
Implement the ability for an administrator to edit shape parameters.

e) Estimation: 50

f) Priority: 5

g) Done:
Administrator is able to log in.
Administrator is able to edit the parameters of any existing shape.
The results of the edits are confirmed as changed.
The administrator commands are only available when logged in as an administrator.
The administrator is able to log out of their account.

#2 – As a user, I am able to contact the developers of the software I'm using - *Mariah*

a) Description:
As a user, I want to be able to view team information in the case that I desire to contact the team, as well as the team logo so that I can distinguish the software produced by the team.

b) Tasks:
As a user, I can see the team logo and contact information.
As a user, I can use this information to actually contact the developers.

c) Tests:

Verify that the logo and contact information is visible to every user.
Verify that the contact information is accurate and kept up to date.

d) Assignee:
Mariah:
Implement team logo and contact information on the UI.

e) Estimation: 2

f) Priority: 10

g) Done:
Team logo and contact information will be displayed at the beginning of the program as well as at the end of the program.

#3 – As an admin, I would like to move a shape - *Kori*

a) Description:
An administrator can move a shape, including text, being rendered.
This is accomplished by way of a move shape form.
The changes will be represented in the rendering area.
The center of the shape (and text) will be represented by a coordinate or by dimensions that can be shifted along the x and/or y axis.

b) Tasks:
As an admin, I can move a shape along the x/y axis by choosing new coordinates for the center of the shape or by altering shape dimensions.
As an admin, I can see the change take place in the rendering area.
As an admin, I know that my changes have been saved.

c) Tests:
Verify than an admin can log in.
Verify that only an admin can move a shape, and not a regular user.
Verify that an admin can move a shape, including text, to a different location from its original location in the rendering area.
Verify that an admin can see that their shape has been moved to the correct location.
Verify that an admin cannot move the shape outside of the bounds of the rendering area.
Verify that the changes are saved.

d) Assignee:
Trevor & Kori:
Implement move function for each derived class of Shape ABC.
Ali & Kori:
Implement shape changes in the rendering area.
Brian & Mariah:
Implement this ability only after administrator login.
Michael & Peter:
Implement changes made to shape/text in the database.

e) Estimation: 10

f) Priority: 5

g) Done:
Admin is able to move a shape, including text, being rendered to a new location in the rendering area.
Admin is able to see their changes appear in the rendering area.
Changes are saved.

#4 – As a new user, I would like to create an account - *Michael*

a) Description:
User can create a new account for themselves to that they can create, erase and manipulate objects that they create.
b) Tasks:
As a user I can create my account.
As a user I can delete my own account.
As a user I can edit my own account.
c) Tests:
Verify that my account exists after I create it. I can see my name and know that I am actively logged in.
Verify that changes in my account are reflected to me via text.
Verify that once I delete my account, it doesn't exist and the data connected to my account is gone.
d) Assignee:
Brian & Mariah:
Implement login screen and account capabilities.
e) Estimation: 35
f) Priority: 3
g) Done:
When a user can successfully make an account, login into their account and edit that account.
Users should be able to delete their account as well an be able to verify that there is no data left over about the user's account.
All tasks, test conditions satisfied.

#5 – As a 'forgetful user,' I would like to recover my credentials - *Michael*

a) Description:
A user should be able to have another way of recovering their password. It should be a seamless transaction that uses personal information to bring up their forgotten password.
b) Tasks:
As a forgetful user I can access my account via password recovery.
As a forgetful user I can use personal information to authenticate that it is myself trying to login.
c) Tests:
Verify that a user can recover their password despite not knowing it.
Verify that the user can recover their password using personal answers to recovery questions.
d) Assignee:
Brian & Mariah:
Implement login recovery process.
Implement saving of personal answers to recovery questions per account.
e) Estimation: 10
f) Priority: 10
g) Done:
User is able to recover their password when forgotten using private information that can verify that the correct user is trying to retrieve their password.
All tasks, test conditions satisfied.

#6 – As a returning user, I would like to login - *Ali*

a) Description:
A user should be able to login with the account that they have created in the past.
An administrator should be able to login with the same type of

password but be able to access administrator specific commands.

b) Tasks:
A user can login to their account with the same password that they saved.
As an administrator I can login to my account and have higher permissions.
A user should have the ability to try again if they fail to login.

c) Tests:
Verify that a user can login with a saved password.
Verify that a wrong password notifies the user.
Verify that an incorrect password does not give away the password.

d) Assignee:
Brian & Mariah:
Implement user login interface.
Implement different options available to user vs. admin once they have successfully logged in.

e) Estimation: 15

f) Priority: 8

g) Done:
User is able to navigate to the login page, type in their email and their password properly.
On wrong password: user is notified that the password they tried is wrong.
On correct password: user is navigated to the right page.

#7 – As a standard user, I would like to draw a new shape - *Braden*

a) Description:
A non-admin user can select a shape they would like to draw.

b) Tasks:
A standard user can select the name of the shape they wish to create.
There will be a drop down menu of valid shapes.
A standard user will be able to create multiple shapes.
As a standard user, I can see my shape in the rendering area.

c) Tests:
Verify that the user can see their shape in the rendering area.
Verify that the user's created shapes are stored for future use.

d) Assignee:
Trevor & Kori:
Implement draw function for each derived class of Shape ABC.
Ali & Kori:
Implement shape addition in the rendering area.
Brian & Mariah:
Implement ability to draw a new shape with standard user credentials.
Michael & Peter:
Update database with addition of shape.

e) Estimation: 20

f) Priority: 2

g) Done:
User is able to select any valid shape they wish to create from a drop down menu. After selecting a shape, the user can view their shape in the rendering area. The user can then select additional shapes that they wish to create. The created shapes are stored in memory for future use (i.e. viewing the shape).

#8 – As a standard user, I would like to view my shape - *Braden*

a) Description:
A non-admin user can view the shape they created.
b) Tasks:
A standard user can select the shape they would like to view out of the shapes they have created.
The program will then display the shape in the rendering area.
A standard user can select additional shapes they wish to view.
A standard user can output all of their created shapes at once.
As a standard user, I can see my shape in the rendering area.
c) Tests:
Verify that the shape outputted is the correct shape.
Verify that the user can output additional shapes.
Verify that the user can see their shape in the rendering area.
d) Assignee:
Ali & Kori:
Implement QT rendering area for shape viewing.
e) Estimation: 25
f) Priority: 2
g) Done:
User is able to select from the shapes they have created and view any of them. Furthermore, the user should be able to output all of their created shapes at once if they wanted to. After outputting a shape, the user can see their shape in the rendering area.

#9 – As a standard user, I would like to view all shapes - *Peter*

a) Description:
A user can view all shapes created depending on the user privilege (standard or admin).
b) Tasks:
As a standard user, I can view all shapes created by myself.
As a standard user, I will receive a notification when I have created a shape.
As an administrator, I can view all shapes created by any single user.
c) Tests:
Verify user privilege as standard user or administrator.
Verify that a standard user can view all the shapes they created.
Verify that an administrator can view all the shapes they created, or all the shapes that another standard user created.
d) Assignee:
Trevor & Kori:
Implement shape class and integration with QT.
Ali & Kori:
Implement integration with QT including rendering area for viewing shapes.
Brian & Mariah:
Implement restrictions based on user access level (admin or standard).
e) Estimation: 5
f) Priority: 4
g) Done:
User is able to view all shapes they created.
An admin can view all shapes created by any user.
All task, test conditions satisfied.

#10 – As a standard user, I would like to delete a shape I created - *Trevor*

a) Description:
A user can delete a shape instance that the user created.
The instance in that vector must be deleted.
Following vector entries must be shifted back to fill space left by deleted entry.
Shape count must be decremented, per deleted item.

b) Tasks:
As a standard user, I can delete any shape I have created.
As an admin, I can delete any shape anyone has created.

c) Tests:
Verify a standard member can delete a created shape.
Verify an admin can delete any shape.
Verify the shape vector reflects the deletion.
Verify the shape count reflects the deletion.

d) Assignee:
Brian & Mariah:
Implement accessor from user class.
Trevor & Kori:
Implement Shape class method for deletion, update of shape count.

e) Estimation: 1.5

f) Priority: 5

g) Done:
User is able to login as a standard/admin user and create an object.
Any user can delete a self-created shape, while an admin is able to delete any shape.
Shape vector, count will reflect shape deletion.

*Baseline story point estimate value of 1 = 1 hour of development time for a single developer