

## EXAMPLE

#1 As a user, I can cancel a reservation

a) Description:

A site user can cancel a reservation for a full refund to her credit card

How far ahead a user must cancel reservation depends upon the user type (see tasks)

A cancelation confirmation will be emailed to the site user

b) Tasks:

As a premium site member, I can cancel a reservation up to the last minute

As a non-premium member, I can cancel up to 24 hours in advance

As a site visitor, I am emailed a confirmation of any cancelled reservation

c) Tests: Verify that a premium member can cancel the same day without a fee

Verify that a non-premium member is charged 10% for a same-day cancellation

Verify that an email confirmation is sent

Verify that the hotel is notified of any cancellation

d) Assignee:

e) Estimation: 25

f) Priority: 9

g) Done:

User is able to login as a non-premium, premium, guest user and cancel reservation (refer to tests)

Appropriate fee is charged to user account

User is refunded balance due minus fee

Appropriate cancelation record recorded via hotel accounting system

Email confirmation is sent to user

All task, test conditions satisfied

Baseline story point estimate value of one = 2 hrs of development time for a single developer

-----

## **#1 As a user, I can view customer testimonials and leave a testimonial.**

a) Description:

A user can view previously left customer testimonials

A user can leave a customer testimonial and it will be saved between program executions

A user will be prompted to leave a customer testimonial

b) Tasks:

Create a button which can allow user to view customer testimonials

Provide text box where user can enter their own testimonial

Save customer testimonial to file

c) Tests:

Verify that any user can view customer testimonials

Verify that a user is prompted and can leave a testimonial

Verify that customer testimonials are saved and persist on future program executions

d) Assignee:

e) Estimation: 6

f) Priority: 1

g) Done:

Users are able to view customer testimonials

Users are prompted and able to leave their own testimonial

Customer testimonials save and show up on future program executions

All task, test conditions satisfied

## **#2 As a user, I can view “contact us” info**

a) Description:

A user can view “contact us” info including the team name and logo

b) Tasks:

Provide button/menu where user can view contact info

c) Tests:

Verify that a user can view “contact us” info

Verify that team name and logo are visible

d) Assignee:

e) Estimation: 4

f) Priority: 3

g) Done:

User can view “contact us” info with team name and logo

All task, test conditions satisfied

### **#3 As a user, I can display all shapes in a GUI window**

#### a) Description:

A user I'll see The shape ID displayed above each shape identifying it.

As a user I want the display a specific size

As a user I want the window to open in a specific location.

#### b) Tasks:

Display the object ID above each object

Learn how to use QT and display a GUI window for shapes

#### c) Tests:

Verify the size is up to spec

Verify the location is up to spec

Verify that the ID is displayed above all objects

Verify that the rendering area is 1000x500

#### d) Assignee:

e) Estimation: 25

f) Priority: 10

g) Done:

User can view the GUI window.

User can see shape ID displayed.

User can see shapes drawn

### **#4 As a user I should be able to move shapes and text in the GUI window.**

#### a.) Description

As a user I should be prompted that the shapes and text can be moved.

As admin any changes to the shapes should be visible and are saved somewhere that what changes have been made.

#### c) Tests:

Verify changes can be made

Verify the admin can see the changes

Verify changes are saved.

#### d) Assignee:

e) Estimation: 15

f) Priority: 8

g) Done:

User can can change the shapes and texts

Admin can see the changes made

**#5 As a user, I want to be able to delete shapes that I have drawn.**

a) Description:

The user will be able to select shapes they have created and remove them

b) Tasks:

As a user, I can select different shapes, created and remove them.

c) Tests:

Verify that a user can select a shape object and remove it.

Verify that a user can select multiple shape objects and remove them.

d) Assignee:

e) Estimation: 20

f) Priority: 9

g) Done:

User is able to select a single shape and delete it

User is able to select multiple shapes and delete multiple shapes at once

User is able to delete all objects in one instance

**#6 As a user, I want to be able to specify the output file type for my rendered objects.**

a) Description:

The user will be able to specify the file type for the objects they have created.

b) Tasks:

Ensure there is a default file type for saving rendered objects.

c) Tests:

Verify the user can save without having to manually specify a file type.

d) Assignee:

e) Estimation: 10

f) Priority: 7

g) Done:

User is able to save with or without selecting file type.

**#7 As a user, I should be asked to log in.**

a) Description:

The user is first presented by a login screen that asks for username and password.

Username and password match.

The login screen has an “enter” button maybe or a text that says what to do to pass the login screen.

Usernames have different access level like, administer....

b) Tasks:

Create a vector that stores the usernames and their passwords.

Hard code username and password.

Give administer more access.

c) Tests:

It only allows to login if username and password match.

That admin only have access to specified things.

Gives an error for other users that want access things that are only accessed by admin.

d) Assignee:

e) Estimation: 7

f) Priority: 2

g) Done:

The user is first asked to log in, the user has a place to enter the username and password.

Once username and password are entered, the user is shown the canvas, if username and password is wrong, correct error message is given.

**#8 As a user, I should be able to view a list of objects I have drawn.**

a) Description:

The user will be able to view a list of all drawn objects at any time during execution, the list should be sorted by id, area, or perimeter where applicable.

b) Tasks:

Create a dropdown menu, or some other method for the user to view what objects they have drawn, listed in order based on their id, area, or perimeter.

c) Tests:

Verify that all the objects in the canvas are listed in the list.

Verify that each object has its own ID.

Verify that the sorting default is by ID.

Verify that all users can sort.

d) Assignee:

e) Estimation: 6

f) Priority: 6

g) Done:

The user can view a list of every object currently drawn, sorted by the objects' id's, areas, or perimeters. (Text will not be listed when sorting by area or perimeter)

**#9 As a user, I want to sort the object list by area.**

a) Description:

Users should be able to sort the list by area.

It should only show objects with an area.

b) Tasks:

If sorted by area: shapes' Type, ID, and Area are only reported.

Write a function that does the sorting (selection\_sort) and by default it sorts by ID but it can accept a function pointers to implement different sortings.

Write a function that is passed into our sorting function to sort by area: (compare\_shape\_area)

c) Tests:

Verify that objects without an area, such as text, are not listed when sorting the list by area.

Verify that it allows the user to select between different sortings.

Verify that sorting can occur at anytime.

Verify that sorting by area is allowed to all users.

d) Assignee:

e) Estimation: 6

f) Priority: 5

g) Done:

The user is able to see the object in the list in order of the biggest area value to smallest Perimeter value. And each object has Type, ID, and area reported.

#### **#10 As a user, I want to sort the object list by perimeter.**

a) Description:

Users should be able to sort the list by parameter.

It should only show objects with a parameter.

Text doesn't have parameter and it should not be shown in the list.

b) Tasks:

If sorted by parameter: shapes' Type, ID, and parameter are only reported.

Write a function that does the sorting (selection\_sort) and by default it sorts by ID but it can accept a function pointers to implement different sortings.

Write a function that is passed into our sorting function to sort by perimeter:

(compare\_shape\_perimeter)

c) Tests:

Verify that sorting by perimeter is shown to all users.

Verify that objects without a perimeter such as text, are not listed when sorting the list by perimeter

Verify that it allows the user to select between different sortings.

Verify that sorting can occur at anytime.

d) Assignee:

e) Estimation: 6

f) Priority: 4

g) Done:

The user is able to see the object in the list in order of the biggest perimeter value to smallest Perimeter value. And each object has Type, ID, and perimeter reported.