

Tania Edith García Velasco 362184

David García Olivares 362118

Assignment 3

1. Briefly describe and justify any design choices you made in your *WindowSystem* class.

In our WindowSystem Class we generate the desktop window as well as the basic structure of the app windows. In this class we initialise the ArrayList where we keep track of our windows and we override the handlePaint method to call our drawWindow method, this method sets the basic structure of the window as the color of the fill and the outline of the box and with those parameters as well as the coordinates to which we set the object, we call for fillRect and drawRect from GraphicsEventSystem to draw the simple window.

We also have a method call addWindow in which we create and add to our list the new windows, this method is called from our test program MyApp, where we send a SimpleWindow object with all the required parameters (coordinates x,y,z, width, height and title name).

We implemented our class this way to try to separate the functionality according to the layers model from the NeWS book and the lecture. In WindowClass we initialise the main architecture of our window system in a very basic way and this class is the one that imports the GraphicsEventSystem giving us the canvas of the desktop and the visibility of the windows, but no look and feel nor functionality.

2. Briefly describe and justify any design choices you made in your *WindowSystem* class.

On the other hand in our WindowsManager class we enhance the look and feel of the basic architecture of the system, adding the Titlebar as well as the buttons (close only for the regular tasks of the assignment and also minimise button and a dock for the expert task).

In this class we also implement the different handlers for the mouse actions/events, the implementation of this methods allows us to give functionality to the components of the desktop and the windows. With the clickEvent we can select a window and turn it into the active window (it reflect the change with change of color in bar and buttons), also if the click was performed in the closeButton position, the window is eliminated from the desktop as well as from our collection and with the draggedEvent we can drag a window through the desktop. In our expert task we also add the minimise functionality where if the click was performed in the minButton the window setVisible parameter is set to false and an icon in our dock is draw instead of the window.

We implemented our class this way to try to separate the functionality according to the layers model from the NeWS book and the lecture. In WindowManager after the initialisation of the

main components we provide the other elements that enhance the visual experience as well as the controls and response of mouse events that give functionality to the windows and an interactive response to the user enhancing the overall experience.