User Interface - Bug #2987

Bug # 2677 (New): fix drawing and functional differences between P2J GUI and 4GL GUI

the combo-box's resources aren't disposed if its dropdown is collapsed

02/09/2016 12:22 AM - Sergey Ivanovskiy

Status:	New		**Start date:**	
Priority:	Normal		**Due date:**	
Assignee:			**% Done:**	80%
Category:			**Estimated time:**	0.00 hour
Target version:				
billable:	No		**case_num:**	
vendor_id:	GCD		**version:**	
Description				

History

#1 - 02/09/2016 12:26 AM - Sergey Ivanovskiy

Found two methods that are supposed to clean up resources, but actually they don't clean up.

```
ComboBoxGuiImpl

    public void exitDropDown()
    {
        // call generic cleanup from the superclass
        super.exitDropDown();

        // clean up registry from drop-down widgets added in GUI code
        OutputManager.instance().getRegistry().removeWidgetsRecursive((Widget) dropDown);
    }
```

This method OutputManager.instance().getRegistry().removeWidgetsRecursive((Widget) dropDown) does nothing for the drop down.
The second code is found in OverlayWindow

```
        addFocusListener(new FocusListener()
        {
            @Override
            public void onFocusLost(FocusEvent event)
            {
                destroy();
            }

            @Override
            public void onFocusGained(FocusEvent event)
            {
                // do nothing
            }
        });
```

This listener doesn't get FocusEvent events.
The first method is used and can be repaired but it needs to cast dropDown to DropDownGuiImpl or develops the fabric to delete widgets that is the opposite to the fabric (GuiWidgetFactory) that creates widgets.

#2 - 02/09/2016 12:28 AM - Sergey Ivanovskiy

Updated by Eugenie Lyzenko 5 days ago:

I think the best place is ComboBoxGuiImpl.postprocessDropDown(). Here we can identify OverlayWindow and call

```
...
            gd.deregisterWindow(ow.getId().asInt());
            WindowManager.remove(ow);
...
```

Where ow is the OverlayWindow instance we need to terminate.

#3 - 02/09/2016 12:29 AM - Sergey Ivanovskiy

- File combobox_2.txt added

Eugenie, please look at this diff, it cleans resources for web and gui. Planning to postpone this fix because the root causes are not difficult, but the design conception is a problem here. According to the DropDown classes hierarchy it can be disposed as a widget and as a window.

#4 - 02/09/2016 12:31 AM - Sergey Ivanovskiy

Updated by Eugenie Lyzenko 5 days ago:

Sergey Ivanovskiy wrote:

> Eugenie, please look at this diff, it cleans resources for web and gui. Planning to postpone this fix because the root causes are not difficult, but the design conception is > a problem here. According to the DropDown classes hierarchy it can be disposed as a widget and as a window.

Looking good. One concern - need to test how it will work with real drop-down overlay window when closing it:
- by pressing comb-box button while drop-down is opened
- by selecting something in the drop-down by mouse or keyboard
- by click mouse outside the drop-down but inside the P2J window while drop-down is opened, including window title
- by click mouse outside the P2J window with some other native OS window while drop-down is opened

All these closing ways should work without NPE or other issues.

#5 - 03/08/2016 10:49 AM - Greg Shah

- % Done changed from 0 to 80

- Subject changed from The combobox'x resources aren't disposed if its dropdown is collapsed to the combo-box's resources aren't disposed if its dropdown is collapsed

- Assignee changed from Sergey Ivanovskiy to Eugenie Lyzenko

- Start date deleted (02/09/2016)

#6 - 11/13/2019 11:50 AM - Greg Shah

- Assignee deleted (Eugenie Lyzenko)

Files

combobox_2.txt	730 Bytes	02/09/2016	Sergey Ivanovskiy