

## User Interface - Bug #5150

### killing the web page for the web client should raise the equivalent of a window-close event

02/16/2021 04:31 PM - Greg Shah

<b>Status:</b>	New	<b>Start date:</b>	
<b>Priority:</b>	Normal	<b>Due date:</b>	
<b>Assignee:</b>	Constantin Asofiei	<b>% Done:</b>	0%
<b>Category:</b>		<b>Estimated time:</b>	0.00 hour
<b>Target version:</b>		<b>case_num:</b>	
<b>billable:</b>	No		
<b>vendor_id:</b>	GCD		
<b>Description</b>			

#### History

##### #1 - 02/16/2021 04:33 PM - Greg Shah

A customer reported this issue (my paraphrased version):

When the user closes the webbrowser (or only the browser tab) then we don't get a "logout" event or anything, so any resources stored in session-level database records will continue to exist. In Progress we would get the "on window-close of w-win" event in the main menu program but not in FWD.

##### #2 - 02/16/2021 04:37 PM - Greg Shah

I think we need a listener for session termination which will emulate the 'window close' logic.

I think this is correct, though I want to make sure that we actually generate the "window close" event as part of our session cleanup. Doing it that way (if it can be made to work) would have the advantage of naturally working with existing 4GL code.

If we actually have to do something like an init-term hook in the server, this means that someone would have to write extra cleanup code for every project where such a requirement exists. I'd like to avoid that.

##### #4 - 02/16/2021 05:07 PM - Jurjen Dijkstra

Thanks for creating this issue.

FYI the business value is: when the software is licensed for a maximum number of concurrent users, then we must count the user sessions. If abandoned sessions are not detected quickly then we reach the licensed maximum too soon, resulting in people not being allowed to log in, which makes them angry ;-)

**#7 - 12/09/2021 08:57 AM - Greg Shah**

- Assignee set to Constantin Asofiei

**#8 - 01/13/2022 10:53 AM - Greg Shah**

Constantin noted this issue:

There is a problem if we try executing the WINDOW-CLOSE after the FWD client is no longer available.

For example, in <customer\_application\_name> there are scenarios where the application prompts the user (via a DIALOG-BOX) if it wants to save or discard the changes, when the APPLY "WINDOW-CLOSE" to wWin is executed. So this dialog is part of the WINDOW-CLOSE trigger.

If we try to run this trigger with no FWD client available, when this prompt is reached, it will most likely abend. I don't see how we can force 'No' for this prompt.

Executing WINDOW-CLOSE after the FWD client session is gone can work if the business logic performed in the trigger does not rely on any client-side features (files, UI interactive dialogs, etc).

I think we should instead allow FWD to execute a custom listener when it terminates, but this poses the problem of knowing <what\_to\_cleanup>?

**#9 - 01/13/2022 10:57 AM - Greg Shah**

We can make this a limitation of the trigger (in the case that the browser window/tab is forcibly killed). I don't think this is an unreasonable requirement. Perhaps we can give some kind of SESSION attribute that reports if this is a "forced exit". This would allow the 4GL code to bypass any client/UI usage.

Or perhaps we should consider some condition processing here. We should be generating a STOP condition in this case. Many startup programs already handle the STOP condition:

```
do on stop undo, leave :  
    ...  
  
    // run everything here  
  
end.  
  
// insert cleanup processing here  
  
quit.
```

Why not invoke some non-interactive logic to cleanup after that block?

but this poses the problem of knowing <what\_to\_cleanup>?

I think this is a harder/messier problem to solve than doing the cleanup from within the session by handling STOP. For now we will defer work on this task, since the STOP approach is a good general purpose solution. The only down side is that any WINDOW-CLOSE logic must be moved to top-level STOP processing.