User Interface - Feature #3288

support > 1 browser tabs for a single FWD java client instance

04/27/2017 02:49 PM - Greg Shah

Status:	New	**Start date:**	
Priority:	Normal	**Due date:**	
Assignee:		**% Done:**	0%
Category:		**Estimated time:**	0.00 hour
Target version:			
billable:	No	**vendor_id:**	GCD

Description

Related issues:

Related to User Interface - Feature #3287: local web client installation	**New**	**04/27/2017**

History

#1 - 09/28/2017 01:47 PM - Sergey Ivanovskiy

Greg, please explain what is the target of this task? The web client handles only one thin client that can be called a peer client, correct?

#2 - 09/28/2017 03:11 PM - Greg Shah

- Related to Feature #3287: local web client installation added

#3 - 09/28/2017 03:12 PM - Greg Shah

Please see #3287-2 through #3287-12 for details on this multiple web socket task.

#5 - 02/19/2024 08:03 AM - Greg Shah

- Subject changed from support > 1 web socket for a single FWD java client instance to support > 1 browser tabs for a single FWD java client instance

- Start date deleted (04/27/2017)

We are thinking about this task again in relation to a customer requirement for a multi-screen system where their users often move the application windows to a separate screen so that the same application displays on multiple screens.

In our web client terms, we would think of this as the same FWD client session having canvas instances on multiple browser tabs. We don't have to have multiple websockets. We can multiplex the event processing of multiple tabs in the client session over a single web socket. The communication between tabs could be via cross-document messaging.

Assume that the "main" tab (the first one created) has the websocket and the taskbar. The other tabs would just have one or more application windows. Any event destined for a "remote tab" window would be forwarded to that tab and events from it forwarded up the websocket.

In a perfect world, the end user could easily drag/drop a 4GL window to another tab. The customer is also OK with us exposing this as 4GL syntax, to move a window to a new tab. We'd also need to consider how to move windows back to the main tab, if that is even made possible.

#6 - 02/20/2024 08:43 AM - Hynek Cihlar

Here is a list of high level tasks that I came up with. With comments and an estimation.

- Model changes (2 MD)

    - New data structures to track windows across desktops.
    - Potential changes to the existing data structures (window list, zlist, driver widgets, caches, etc.).
- Secondary desktop bootstrap (1 MD)
    - WebGUI endpoint for serving secondary desktops.
    - Setup secondary desktop, canvases, required model state from the primary desktop.
- Message protocol (1 MD)
    - Between the primary desktop and the secondary.
- Client commands abstraction layer (1 MD)
    - The messages from/to WebGUI driver are sent either to the primary desktop or routed to a secondary desktop.
- UI for moving windows between desktops (0.5 MD)
    - Assuming all or nothing move. A window cannot be displayed on multiple desktops.
- Controller logic for moving windows between desktops (3 MD)
- Desktop UI changes (0.5 MD)
    - For example the task bar should only show on the primary desktop.
- Failover logic (1 MD)
    - End-user closes a secondary desktop
    - End-user closes the primary desktop
- Embedded mode changes (0.5 MD)
    - Disallow multiple desktops there.
- Taskbar popup menu (2 MD)
    - The tools there must work for all the desktops
- Window layout logic (1 MD)
    - For example newly created windows, modal windows, alert boxes should be realized on the same desktop as the parent.
- Modality (3 MD)
    - Modality must work globally on all desktops.
    - When a modal window is displayed, all the desktops must be disabled for input waiting for the modal window to close.
- Driver widgets (?)
    - This is a big unknown
    - Does it mean the driver widget will need to be re-realized when moving across desktops? And so its state is lost?
    - Descope?

My total estimate is around 17 MDs for the core implementation.