

User Interface - Feature #2673

support a kind of virtual desktop in the GUI web client

09/01/2015 02:28 PM - Greg Shah

Status:	WIP	Start date:	
Priority:	Normal	Due date:	
Assignee:	Sergey Ivanovskiy	% Done:	0%
Category:		Estimated time:	0.00 hour
Target version:		vendor_id:	GCD
billable:	No		
Description			
Related issues:			
Related to User Interface - Feature #1811: implement the AJAX client driver		Closed	11/01/2013 03/06/2014

History

#1 - 09/01/2015 02:38 PM - Greg Shah

The idea here is to make sure that users can interact with the GUI web client in the similar use patterns to how they can under the 4GL (on a Windows system).

- ~~The web page already is a kind of desktop or screen. We define that "screen size" in GuiWebDriver, defaulting it to 768 x 1024 BUT allowing each dimension to be overridden by in the directory (see GuiWebDriver.init() and WebClientBuilderOptions.readOptions()). That definition is really only considered on the Java side. We don't fix the size of the web page itself.~~
- ~~More worrisome, is that we don't provide any way for accessing windows that are positioned outside of the currently visible web page space. We should support scrollbars when the defined screen size in Java is larger than the viewable page size in the browser.~~
- We should provide wallpaper support. We can allow a scaled or cropped image configured in the directory and if it is specified, send it down to the JS side for display as the desktop background.
- Provide the ability to set the "desktop background" to a specific color.
- At the login prompt, optionally add a drop-down that allows the user to choose the UI theme. Then make sure to pass that as part of the bootstrap configuration (using client:driver:theme) when launching the web client. If this feature is not enabled in the directory, then the drop-down would not be displayed.
- Provide a way for custom information to be displayed on the desktop. For example, we might want to allow log entries, server status information or other custom data to be displayed in a read-only "window" that can be seen in this "desktop".
- Implement the ability to change the user's OS password. This is needed at the initial login panel, not actually in the virtual desktop itself.

#2 - 09/11/2015 10:21 AM - Greg Shah

More worrisome, is that we don't provide any way for accessing windows that are positioned outside of the currently visible web page space. We should support scrollbars when the defined screen size in Java is larger than the viewable page size in the browser.

Actually, this works now (e.g. branch 1811q revision 11007).

#3 - 03/23/2016 04:21 PM - Greg Shah

- *Target version deleted (Milestone 12)*
- *Parent task deleted (#1811)*

#4 - 09/21/2018 12:04 PM - Greg Shah

The web page already is a kind of desktop or screen. We define that "screen size" in GuiWebDriver, defaulting it to 768 x 1024 BUT allowing each dimension to be overridden by in the directory (see GuiWebDriver.init() and WebClientBuilderOptions.readOptions()). That definition is really only considered on the Java side. We don't fix the size of the web page itself.

Sergey: Is this still an issue?

#5 - 09/21/2018 12:15 PM - Sergey Ivanovskiy

Yes, the web client updates the desktop size using this message MSG_GET_DESKTOP_DIMENSION. I looked at p2j.socket.js and p2j.screen.js. It seems that its implementation is complete.

#6 - 09/21/2018 12:16 PM - Sergey Ivanovskiy

This issue is fixed.

#7 - 09/21/2018 12:27 PM - Greg Shah

- *Assignee set to Sergey Ivanovskiy*
- *Status changed from New to WIP*

Sergey, please work on the following:

- We should provide wallpaper support. We can allow a scaled or cropped image configured in the directory and if it is specified, send it down to the JS side for display as the "desktop background".
- Provide the ability to set the "desktop background" to a specific color.
- At the login prompt, optionally add a drop-down that allows the user to choose the UI theme. Then make sure to pass that as part of the bootstrap configuration (using client:driver:theme) when launching the web client. If this feature is not enabled in the directory, then the drop-down would not be displayed.

I would like to make sure these are implemented by the end of September.

#8 - 09/21/2018 02:14 PM - Sergey Ivanovskiy

We have SET_DESKTOP_BGCOLOR drawing primitive that is used to set background colors in the embedded mode client with help of p2j.js and cross-domain message for the embedded frame page.

```
/**
 * Set the default desktop background color.
```

```

*
* @param    {String} rgb
*           The CSS color string.
*/
p2j.setDefaultDesktopBgcolor = function(rgb)
{
    p2j.defaultDesktopBgcolor = rgb;
};

/**
* Set the desktop background color.
*
* @param    {String} rgb
*           The CSS color string.
*/
p2j.setDesktopBgColor = function (rgb)
{
    if (p2j.embedded)
    {
        if (p2j.defaultDesktopBgcolor)
        {
            rgb = p2j.defaultDesktopBgcolor;
        }

        setStyleForElement(document.body, {background : rgb});
    }
}

```

It seems that the background images for virtual desktop should be implemented similarly to the embedded fonts.

#9 - 10/02/2018 09:12 AM - Sergey Ivanovskiy

Please help to answer if there is a way to enumerate all supported themes. The supported themes can be made hard coded values.

#10 - 10/02/2018 10:51 AM - Ovidiu Maxiniuc

Sergey Ivanovskiy wrote:

Please help to answer if there is a way to enumerate all supported themes. The supported themes can be made hard coded values.

The Themes do not register themselves in any way. For the moment there are just 4 of them: ClassicTheme, Windows10Theme, Windows8Theme

and MaterialTheme, all in the same package com.goldencode.p2j.ui.client.gui.theme. A fifth one exist FwdEnhancedTheme, but I think the issues it addresses were fixed in Windows10Theme so I expect we will drop it. You can hardcode this list and use it to set the Theme in the client using ThemeManager.setTheme(String) method.

Alternatively, we can scan for non-abstract classes that implement Theme interface. They are normally found in the dedicated package: com.goldencode.p2j.ui.client.gui.theme.

#11 - 10/02/2018 03:03 PM - Sergey Ivanovskiy

OK, thank you. I found that there is reflections-0.9.9-RC1.jar in the build/lib but build.gradle doesn't list this library explicitly. We can use this library to list all sub classes of Theme in this package com.goldencode.p2j.ui.client.gui.theme

#12 - 10/02/2018 03:07 PM - Eric Faulhaber

I think the themes available to a user should come from a list in the directory, not from every available subclass of Theme. Administrators will want to control this.

#13 - 10/02/2018 03:10 PM - Sergey Ivanovskiy

Planning to get all available themes from the directory.

#14 - 10/03/2018 11:40 AM - Sergey Ivanovskiy

I don't understand but this code returns null value for nodePath1

```
DirectoryService ds = DirectoryService.getInstance();

if (!ds.bind())
{
    throw new RuntimeException("directory bind failed");
}

try
{
    String nodePath = Utils.findDirectoryNodePath(ds, "available-themes", "container", DirScope.BOTH, null);
    String nodePath1 = Utils.findDirectoryNodePath(ds, "webClient", "container", DirScope.BOTH, null);
    .....
}
```

I tried to add available themes to the directory but failed?

```
<node class="container" name="available-themes">
  <node class="theme" name="theme1">
    <node-attribute name="value" value="com.goldencode.p2j.ui.client.gui.theme.Windows10Theme"/>
    <node-attribute name="name" value="Windows 10"/>
    <node-attribute name="default" value="TRUE"/>
  </node>
  <node class="theme" name="theme2">
    <node-attribute name="value" value="com.goldencode.p2j.ui.client.gui.theme.Windows8Theme"/>
    <node-attribute name="name" value="Windows 8"/>
  </node>
  <node class="theme" name="theme3">
    <node-attribute name="value" value="com.goldencode.p2j.ui.client.gui.theme.ClassicTheme"/>
    <node-attribute name="name" value="Windows Classic"/>
  </node>
  <node class="theme" name="theme4">
    <node-attribute name="value" value="com.goldencode.p2j.ui.client.gui.theme.MaterialTheme"/>
  </node>
</node>
```

#15 - 10/03/2018 11:45 AM - Ovidiu Maxiniuc

I think you need to define the theme node type in src/dir_schema.xml.

#16 - 10/03/2018 11:53 AM - Sergey Ivanovskiy

I defined this new class but SecurityManager.getAccountIds() throws NullPointerException

```
public String[] getAccountIds()
{
    // find the security cache
    SecurityContextStack ctxs = SecurityContextStack.getContext(); // <-----returns null here
```

with this stack trace

```
NullPointerException.<init>() line: 60 [local variables unavailable]
SecurityManager.getAccountIds() line: 3740
Utils.getDirectoryNodeWorker(DirectoryService, String, String, Utils$DirectoryHelper, Utils$DirScope, String[])
) line: 2786
Utils.findDirectoryNodePath(DirectoryService, String, String, Utils$DirScope, String[]) line: 2667
```

#17 - 10/03/2018 12:04 PM - Eric Faulhaber

Do we really need a separate theme directory class? This essentially is just a key/value mapping between strings. Can't we just use the entry class instead? The only difference is the additional default value, but that can be done with a separate but related directory setting.

Why does the material theme not have a key/name?

#18 - 10/03/2018 12:19 PM - Sergey Ivanovskiy

I realized that it is done intentionally due to security restriction. If this code is invoked from StandardServer during initialization it works properly. OK. I will use entry class with key and value map. The reason to use this new class is to be able to write only the class path name

```
<node class="container" name="available-themes">
.....
    <node class="theme" name="theme4">
        <node-attribute name="value" value="com.goldencode.p2j.ui.client.gui.theme.MaterialTheme"/>
    </node>
.....
</node>
```

#19 - 10/04/2018 12:20 PM - Sergey Ivanovskiy

I was trying to set theme via "client:driver:theme"

```
<node class="container" name="remotelaunchoption">
  <node class="container" name="000100">
    <node class="resource" name="resource-instance">
      <node-attribute name="reference" value="client:driver:theme"/>
      <node-attribute name="reftype" value="TRUE"/>
    </node>
    <node class="remoteLaunchOptionRights" name="rights">
      <node-attribute name="allow" value="true"/>
    </node>
    <node class="strings" name="subjects">
      <node-attribute name="values" value="all_others"/>
    </node>
  </node>
</node>
```

but found that this code in `WebHandler.verifySpawnOptions` doesn't work since the security context for the web client os user is not defined and `SecurityContextStack.scs = SecurityContextStack.getContext();` returns null in this stack trace

```
SecurityManager.getPluginInstance(String) line: 4962
WebHandler.verifySpawnOptions(String, String[]) line: 857
WebHandler.spawnWorker(String, String, boolean, String[], boolean, String, String[]) line: 794
WebHandler.doPost(String, Request, HttpServletRequest, HttpServletResponse) line: 599
WebHandler.handle(String, Request, HttpServletRequest, HttpServletResponse) line: 452
```

#20 - 10/04/2018 03:50 PM - Sergey Ivanovskiy

Greg, please could you advise to do in this case. `WebHandler.verifySpawnOptions` is used for the trusted mode in this case security context is setup properly. Could we bypass `verifySpawnOptions` for this `client:driver:theme` in the virtual desktop mode?

#21 - 10/04/2018 04:06 PM - Constantin Asofiei

Sergey, for reading the available themes from the directory, follow the `FontTable.readFontTable` approach; these should NOT be read on every request of `WebHandler.doGet`, but passed as an argument to the `WebHandler` c'tor, when the server initializes this; you should have no problem reading this from the server's context.

For the `client:driver:theme` case - the idea here is the theme chosen from the drop-down must override any other configured theme; you don't need to set anything in the directory.

WebHandler.doPost just has to pass this theme as client:driver:theme=<themeid> to spawnWorker's options parameter (a single element array). Everything else should fall in place and override the client's theme - have you tried this?

#22 - 10/04/2018 04:43 PM - Sergey Ivanovskiy

Constantin Asofiei wrote:

Sergey, for reading the available themes from the directory, follow the FontTable.readFontTable approach; these should NOT be read on every request of WebHandler.doGet, but passed as an argument to the WebHandler c'tor, when the server initializes this; you should have no problem reading this from the server's context.

Yes. I followed this way.

For the client:driver:theme case - the idea here is the theme chosen from the drop-down must override any other configured theme; you don't need to set anything in the directory.

Yes, I did it and provided theme via option.

WebHandler.doPost just has to pass this theme as client:driver:theme=<themeid> to spawnWorker's options parameter (a single element array). Everything else should fall in place and override the client's theme - have you tried this?

Don't understand. Now the options are verified against security context. Since this context is not set, all options are forbidden. Don't know another ways except these ones: to bypass options verifications for client:driver:theme or to setup the security context as we do for admin applet.

#23 - 10/04/2018 05:04 PM - Constantin Asofiei

Sergey Ivanovskiy wrote:

Don't understand. Now the options are verified against security context. Since this context is not set, all options are forbidden.

Ah, you are right, when WebHandler.verifySpawnOptions is called we are on a contextless thread. So, now that I understand correctly your original question: Could we bypass verifySpawnOptions for this client:driver:theme in the virtual desktop mode? - I think this is the way to do it - only client:driver:theme is white-listed, and only if we are in a contextless thread. If there is any other option (in this contextless thread case), we treat it as 'access forbidden' and do not move forward with the spawn command.

Please make sure that the theme ID in doPost is fully validated (is a well known theme ID in FWD, and nothing else) - if this is not valid, return immediately with a 'theme ID not valid' error code.

There is no reason to otherwise validate these options, as we have no user-specific context at the time of this call; we could only rely on the server's context (if we use an AssociatedThread with the server's context and delegate the option validation to that thread), but I think this is overkill, as no other option can be injected into the spawn command by the user, at the time of login.

#24 - 10/04/2018 06:17 PM - Sergey Ivanovskiy

Please review the committed rev 11337 that added UI Theme drop down for the web client. The UI themes are given by available-themes node under /server/default

```
<node class="container" name="available-themes">
  <node class="entry" name="theme1">
    <node-attribute name="key" value="com.goldencode.p2j.ui.client.gui.theme.Windows10Theme"/>
    <node-attribute name="value" value="Windows 10"/>
  </node>
  <node class="entry" name="theme2">
    <node-attribute name="key" value="com.goldencode.p2j.ui.client.gui.theme.Windows8Theme"/>
    <node-attribute name="value" value="Windows 8"/>
  </node>
  <node class="entry" name="theme3">
    <node-attribute name="key" value="com.goldencode.p2j.ui.client.gui.theme.ClassicTheme"/>
    <node-attribute name="value" value="Windows Classic"/>
  </node>
  <node class="entry" name="theme4">
    <node-attribute name="key" value="com.goldencode.p2j.ui.client.gui.theme.MaterialTheme"/>
  </node>
</node>
```

#25 - 10/05/2018 08:55 AM - Greg Shah

Constantin: Please review.

#26 - 10/05/2018 10:10 AM - Constantin Asofiei

Sergey Ivanovskiy wrote:

Please review the committed rev 11337

What branch is this in?

#27 - 10/05/2018 10:25 AM - Sergey Ivanovskiy

This code rev 11337 from 3600e is in the trunk rev 11286 already. The rest independent part of my task is to set the virtual desktop image. It will be committed today at 3706a.

#28 - 10/05/2018 10:40 AM - Constantin Asofiei

Review for 3600e rev 11337 (now in trunk 11286):

- `WebHandler.doPost` - you need to validate `base.getParameter("theme")`; against the configured, available themes, which you have saved in `WebHandler.THEMES_UI`; if this theme parameter is not a valid theme (as configured in the directory's available themse), you need abort authentication. Otherwise, this will be a security hole, where someone can enter whatever it wants, even something like bogus `client:driver:type=gui_swing`.
- `WebHandler.spawnWorker` - you have an extra param in the javadoc
- why is the `OutputManager.instance()` change required for this task? Is this trying to initialize itself during the auth process (when you call `!SecurityManager.getInstance().hasContext();`)?

#29 - 10/05/2018 11:01 AM - Sergey Ivanovskiy

Constantin Asofiei wrote:

Review for 3600e rev 11337 (now in trunk 11286):

- `WebHandler.doPost` - you need to validate `base.getParameter("theme")`; against the configured, available themes, which you have saved in `WebHandler.THEMES_UI`; if this theme parameter is not a valid theme (as configured in the directory's available themse), you need abort authentication. Otherwise, this will be a security hole, where someone can enter whatever it wants, even something like bogus `client:driver:type=gui_swing`.
- `WebHandler.spawnWorker` - you have an extra param in the javadoc

Ok, thank you, planning to fix these two issues in 3706a.

- why is the `OutputManager.instance()` change required for this task? Is this trying to initialize itself during the auth process (when you call `!SecurityManager.getInstance().hasContext();`)?

No, it has been invoked at this step

```
OutputManager<P>.instance() line: 341
OutputManager<P>.getDriver() line: 354
MaterialTheme.<init>() line: 139
NativeConstructorAccessorImpl.newInstance0(Constructor<?>, Object[]) line: not available [native method]
NativeConstructorAccessorImpl.newInstance(Object[]) line: 62
DelegatingConstructorAccessorImpl.newInstance(Object[]) line: 45
Constructor<T>.newInstance(Object...) line: 423
Class<T>.newInstance() line: 442
WebHandler$ThemeUI.instantiate(String) line: 1161
WebHandler.getAvailableThemes() line: 1045
WebHandler.<clinit>() line: 150
StandardServer$14.initialize() line: 1360
StandardServer.hookInitialize() line: 1956
StandardServer.bootstrap(BootstrapConfig) line: 959
```

#30 - 10/05/2018 11:09 AM - Constantin Asofiei

Sergey Ivanovskiy wrote:

- why is the `OutputManager.instance()` change required for this task? Is this trying to initialize itself during the auth process (when you call `!SecurityManager.getInstance().hasContext();`)?

No, it has been invoked at this step
[...]

Hmm... ideally we wouldn't need to instantiate the theme to get its name. An annotation would be helpful here - Ovidiu, any thoughts?

#31 - 10/05/2018 11:13 AM - Sergey Ivanovskiy

Constantin Asofiei wrote:

Sergey Ivanovskiy wrote:

- why is the `OutputManager.instance()` change required for this task? Is this trying to initialize itself during the auth process (when you call `!SecurityManager.getInstance().hasContext();`)?

No, it has been invoked at this step
[...]

Hmm... ideally we wouldn't need to instantiate the theme to get its name. An annotation would be helpful here - Ovidiu, any thoughts?

It has been done in order to check the validity of theme keys.

#32 - 10/05/2018 11:17 AM - Constantin Asofiei

Sergey Ivanovskiy wrote:

Hmm... ideally we wouldn't need to instantiate the theme to get its name. An annotation would be helpful here - Ovidiu, any thoughts?

It has been done in order to check the validity of theme keys.

You can use `Theme.class.isAssignableFrom(Class.forName(<clsName>))` to check if this is a valid theme; the only need to get an instance is to get the theme name; if we have class-level annotation with the theme name, then no instance is required.

#33 - 10/05/2018 11:29 AM - Sergey Ivanovskiy

Agree that class annotations can be used for theme name, but in this case we will add new dependency. Theme interface has method `getName()`. Is it a key only?

Has it meaning of a theme presentation name?

#34 - 10/05/2018 01:27 PM - Ovidiu Maxiniuc

The `Theme.getName()` is practically not used. I added it for similar reasons. Currently the name is only displayed in console and used for identifying the theme using a nick/short name as an alternative to full path.

I understand that instantiating the Theme might be costly because of the eventual resources it need to load. A static method would not require the instantiation but we lose inheritance so this cannot be used. As noted above, a new class annotation would solve the problem. And, to be on safe side, maybe the `getName()` method should return the same value.

There is a 3rd alternative. As workaround we can use the class name, trimming the Theme token. So we would have: Classic for `ClassicTheme`, Windows10 for `Windows10Theme` and so on.

#35 - 10/05/2018 01:38 PM - Constantin Asofiei

Sergey Ivanovskiy wrote:

Agree that class annotations can be used for theme name, but in this case we will add new dependency.

What kind of dependency you mean? The class annotation?

Theme interface has method `getName()`. Is it a key only? Has it meaning of a theme presentation name?

Actually, what Theme.getName() is not really usable in a UX point of view: it returns i.e. windows8 which is not clean/nice to show on a login page. So, the annotation should provide not a name (which really is an additional theme ID), but a theme label (like Windows 8, FWD Enhanced Theme, etc).

Please go with this annotation approach, to provide a theme label. Another point to avoid theme instantiation is that we will need to be careful not to introduce complex code in the theme c'tor, which can be overlooked easily.

#36 - 10/05/2018 01:48 PM - Sergey Ivanovskiy

Thanks for the thoughtful discourse of this task. Planning to follow it.

#37 - 10/05/2018 03:26 PM - Ovidiu Maxiniuc

Constantin Asofiei wrote:

Actually, what Theme.getName() is not really usable in a UX point of view: it returns i.e. windows8 which is not clean/nice to show on a login page.

As noted above, the method does not have a very defined purpose. Sergey, if you consider necessary, you can return any other value, for example Windows 8 for the respective theme. It should be consistent, so it need to match the annotation.

#38 - 10/05/2018 03:28 PM - Sergey Ivanovskiy

Yes, agree.

#39 - 10/05/2018 09:11 PM - Sergey Ivanovskiy

Planning to add new directory settings under webClient/desktop

```
<node class="container" name="desktop">
  <node class="string" name="background">
    <node-attribute name="value" value="0x000000"/>
  </node>
  <node class="string" name="background-image">
    <node-attribute name="value" value="test.png"/>
  </node>
  <node class="string" name="image-mime-type">
    <node-attribute name="value" value="image/png"/>
  </node>
  <node class="string" name="background-origin">
    <node-attribute name="value" value="border-box"/>
  </node>
  <node class="string" name="background-position">
    <node-attribute name="value" value="center"/>
  </node>
  <node class="string" name="background-size">
    <node-attribute name="value" value="100% 100%"/>
  </node>
  <node class="string" name="background-repeat">
    <node-attribute name="value" value="no-repeat"/>
  </node>
</node>
```

and new message type - MSG_SET_DESKTOP_IMAGE. At the moment the code has been tested. Planning to commit these changes after testing.

#40 - 10/06/2018 12:45 PM - Greg Shah

I like it. Please post some details on the meaning of each one.

#41 - 10/07/2018 02:53 PM - Sergey Ivanovskiy

I didn't invent these parameters they define the CSS background image styles:

<https://developer.mozilla.org/en-US/docs/Web/CSS/background-image>

The background-image CSS property sets one or more background images on an element.

<https://developer.mozilla.org/en-US/docs/Web/CSS/background-origin>

The background-origin CSS property sets the background positioning area, i.e., the origin position of an image specified using the background-image property.

<https://developer.mozilla.org/en-US/docs/Web/CSS/background-position>

The background-position CSS property sets the initial position, relative to the background position layer defined by background-origin, for each defined background image.

<https://developer.mozilla.org/en-US/docs/Web/CSS/background-size>

The background-size CSS property specifies the size of the element's background image. The image can be left to its natural size, stretched to a new size, or constrained to fit the available space while preserving its intrinsic proportions.

<https://developer.mozilla.org/en-US/docs/Web/CSS/background-repeat>

The background-repeat CSS property defines how background images are repeated. A background image can be repeated along the horizontal axis, the vertical axis, both axes, or not repeated at all.

The image-mime-type parameter is optional. If the target background-image is given by the file resource, then it, image-mime-type, can be guessed from the file input stream.

#42 - 10/07/2018 03:15 PM - Constantin Asofiei

Sergey, I don't think we want to encode each and every CSS property you want to set for the image, as a distinct directory entry. Maybe just use a 'css' directory entry, where you just set the CSS separated by ;?

#43 - 10/07/2018 03:25 PM - Greg Shah

Constantin Asofiei wrote:

Sergey, I don't think we want to encode each and every CSS property you want to set for the image, as a distinct directory entry. Maybe just use a 'css' directory entry, where you just set the CSS separated by ;?

Sergey's current approach is more secure since it is naturally limited in the values we support. I worry that if we open up the style attribute, there may be some security implications.

#44 - 10/07/2018 03:51 PM - Sergey Ivanovskiy

I didn't add CSS parameters validation now but these parameters are secured by the directory. They can't be inserted from any different source. If the client receives this message MSG_SET_DESKTOP_IMAGE, then it retrieves target parameters and sets the desktop CSS style properties. If somebody can be in the middle of the server and the client and can alter this message, then the parameter validation can help here. Is there another way?

#45 - 10/07/2018 04:44 PM - Greg Shah

I'm saying that your approach has security advantages.

#46 - 10/07/2018 05:48 PM - Sergey Ivanovskiy

Please review committed revision 11292 (3706a) that added desktop images for the virtual desktop mode.

#47 - 10/08/2018 03:24 AM - Sergey Ivanovskiy

Committed revision 11294 fixed NPE if background-image is not set.

#48 - 10/08/2018 05:45 AM - Constantin Asofiei

Sergey Ivanovskiy wrote:

Please review committed revision 11292 (3706a) that added desktop images for the virtual desktop mode.

What happens if one or more of the CSS settings from [#2673-39](#) are not configured? Does the JS and Java code handle this properly?

#49 - 10/08/2018 07:56 AM - Sergey Ivanovskiy

I tested if these settings are not configured. Incorrect settings can lead to CSS style errors. The java web client or server can fail to detect the image mime type. These errors can lead to incorrect CSS styles. Finally incorrect CSS styles are ignored by the browser.

#50 - 10/08/2018 08:16 AM - Greg Shah

You described how incorrect settings were handled. What about missing settings?

#51 - 10/08/2018 08:17 AM - Greg Shah

Do you have settings we can use with Hotel GUI to add the FWD logo and improve the attractiveness of the virtual desktop mode?

#52 - 10/08/2018 08:46 AM - Sergey Ivanovskiy

Greg Shah wrote:

You described how incorrect settings were handled. What about missing settings?

No, missing settings are represented by empty strings in CSS style object. They are incorrect settings. There are mutually exclusive settings. Please look at <https://developer.mozilla.org/en-US/docs/Web/CSS/background-repeat> where if background-repeat is set to space, then the background-position property is ignored unless only one image can be displayed without clipping.

#53 - 10/08/2018 08:50 AM - Sergey Ivanovskiy

Greg Shah wrote:

Do you have settings we can use with Hotel GUI to add the FWD logo and improve the attractiveness of the virtual desktop mode?

I didn't add new decorations for Hotel GUI. But adding a new background image is quite simple, but there is no mechanism to refresh the background image without restarting the server.

#54 - 10/08/2018 09:56 AM - Greg Shah

but there is no mechanism to refresh the background image without restarting the server.

This is OK.

#55 - 10/08/2018 10:01 AM - Sergey Ivanovskiy

Found a copy paste error in the p2j.socket.js and added empty styles validation. Please review committed rev.11296.

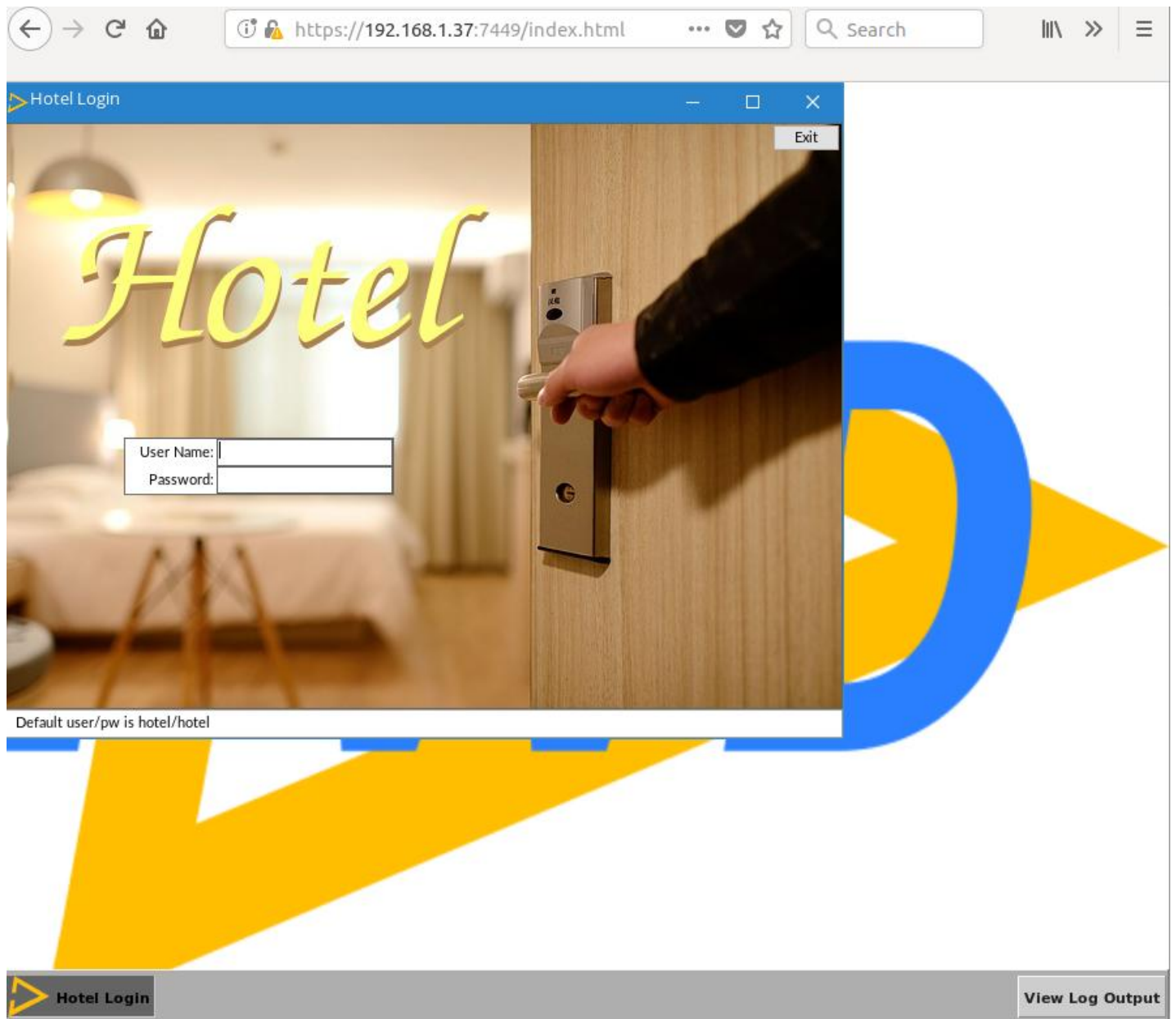
- File HotelGuiDesktopImage.png added

Greg Shah wrote:

Do you have settings we can use with Hotel GUI to add the FWD logo and improve the attractiveness of the virtual desktop mode?

These settings can be used but it doesn't look attractive.

```
<node class="container" name="desktop">
  <node class="string" name="background">
    <node-attribute name="value" value="0x000000"/>
  </node>
  <node class="string" name="background-image">
    <node-attribute name="value" value="/com/goldencode/hotel/embedded/images/fwd.png"/>
  </node>
  <node class="string" name="image-mime-type">
    <node-attribute name="value" value="image/png"/>
  </node>
  <node class="string" name="background-origin">
    <node-attribute name="value" value="border-box"/>
  </node>
  <node class="string" name="background-position">
    <node-attribute name="value" value="center"/>
  </node>
  <node class="string" name="background-size">
    <node-attribute name="value" value="100% 100%"/>
  </node>
  <node class="string" name="background-repeat">
    <node-attribute name="value" value="no-repeat"/>
  </node>
</node>
```

#57 - 10/08/2018 10:45 AM - Sergey Ivanovskiy

It seems now to me that this solution is not good, because we should add the desktop image support to index.html template page without adding new MSG_SET_DESKTOP_IMAGE.

#58 - 10/08/2018 10:57 AM - Constantin Asofiei

Sergey Ivanovskiy wrote:

It seems now to me that this solution is not good, because we should add the desktop image support to index.html template page without adding new MSG_SET_DESKTOP_IMAGE.

I think we need both:

- index.html will work with a full background image for the login purpose.
- MSG_SET_DESKTOP_IMAGE will work with setting not the body's background image, but the virtual desktop's (i.e. the image will not be below our emulated taskbar).

#59 - 10/08/2018 10:58 AM - Constantin Asofiei

Eventually, you can set the body image at login, and after login, move this image as background for our virtual desktop? The settings in directory.xml will remain the same, I don't think you need to change these.

#60 - 10/08/2018 11:03 AM - Sergey Ivanovskiy

I meant that the background images should be managed by the web client web server and should be delivered via https. Now they are delivered via websocket. Do you agree?

#61 - 10/08/2018 11:07 AM - Sergey Ivanovskiy

Constantin, it seems that web_client.html template is for login page but the JS web client uses index.html template.

#62 - 10/08/2018 11:11 AM - Greg Shah

Please make sure that Hotel GUI has a functional solution today. We intend to show this at the PUG Challenge.

#63 - 10/08/2018 11:14 AM - Sergey Ivanovskiy

OK. This task is done. It seems that if the FWD had SET-DESKTOP-IMAGE url. statement, then adding MSG_SET_DESKTOP_IMAGE would make sense.

#64 - 10/08/2018 11:17 AM - Constantin Asofiei

Sergey Ivanovskiy wrote:

Constantin, it seems that web_client.html template is for login page

Did you add support for a background image for the web_client.html login template?

#65 - 10/08/2018 11:21 AM - Sergey Ivanovskiy

Constantin Asofiei wrote:

Sergey Ivanovskiy wrote:

Constantin, it seems that web_client.html template is for login page

Did you add support for a background image for the web_client.html login template?

No, I didn't add a desktop image to this template. I added the desktop image to the virtual desktop according to the [#2673-7](#) task. Where this image should be located?

#66 - 10/08/2018 11:22 AM - Constantin Asofiei

Sergey Ivanovskiy wrote:

Constantin Asofiei wrote:

Sergey Ivanovskiy wrote:

Constantin, it seems that web_client.html template is for login page

Did you add support for a background image for the web_client.html login template?

No, I didn't add a desktop image to this template. I added the desktop image to the virtual desktop according to the note-7 task. Where this image should be located?

I was thinking to use the same image configured in the directory. Greg, do we need this now?

#67 - 10/08/2018 11:34 AM - Constantin Asofiei

Sergey, please update the Hotel GUI project with all directory settings required for the background image (directory.xml and the .template versions).

#68 - 10/08/2018 11:40 AM - Sergey Ivanovskiy

OK. Please wait.

#69 - 10/08/2018 11:59 AM - Greg Shah

Greg, do we need this now?

If it is easily possible, then yes please do it.

#70 - 10/08/2018 12:03 PM - Sergey Ivanovskiy

Greg, would you like to use the same desktop image for web_client.html as for index.html?

#71 - 10/08/2018 12:18 PM - Greg Shah

Yes.

#72 - 10/08/2018 12:30 PM - Sergey Ivanovskiy

Committed revision 177 (Hotel_GUI) added available-themes and background-image style settings.

#73 - 10/08/2018 01:34 PM - Sergey Ivanovskiy

It seems that web_client.html can be decorated for an example using the existing Bootstrap styles. In order to do this the web server should manage these resources. The embedded web client has a custom solution for this task. At this moment I don't plan to develop this way, but planning to add this style login class to web_client.html and assign this class to the body element.

```
<style>
.login {
  "width" : "100%";
  "height" : "100%";
  "background-image" : "${background-image}";
  "background-origin" : "${background-origin}";
  "background-position" : "${position}";
  "background-size" : "${size}";
  "background-repeat" : "${repeat}";
}
</style>
```

Is it enough?

#74 - 10/08/2018 04:15 PM - Greg Shah

If we can't serve the resources, how does this work?

#75 - 10/08/2018 04:25 PM - Sergey Ivanovskiy

Don't understand, please clarify your question. The web_client.html template has been filled by WebHandler on GET / request and then it has been sent to the client. WebHandler itself doesn't handle requests for different resources.

#76 - 10/08/2018 04:34 PM - Constantin Asofiei

Sergey, when starting a standalone server for Hotel GUI embedded mode (ran from /embedded/ folder), I get this error:

```
java.lang.ExceptionInInitializerError
    at com.goldencode.testcases.embedded.EmbeddedWebAppDriver$EmbeddedPageHandler.handle(EmbeddedWebAppDriver.java:367)
    at org.eclipse.jetty.server.handler.HandlerCollection.handle(HandlerCollection.java:119)
    at org.eclipse.jetty.server.handler.HandlerWrapper.handle(HandlerWrapper.java:134)
    at org.eclipse.jetty.server.Server.handle(Server.java:518)
    at org.eclipse.jetty.server.HttpChannel.handle(HttpChannel.java:308)
    at org.eclipse.jetty.server.HttpConnection.onFillable(HttpConnection.java:244)
    at org.eclipse.jetty.io.AbstractConnection$ReadCallback.succeeded(AbstractConnection.java:273)
    at org.eclipse.jetty.io.FillInterest.fillable(FillInterest.java:95)
    at org.eclipse.jetty.io.ssl.SslConnection.onFillable(SslConnection.java:186)
    at org.eclipse.jetty.io.AbstractConnection$ReadCallback.succeeded(AbstractConnection.java:273)
    at org.eclipse.jetty.io.FillInterest.fillable(FillInterest.java:95)
    at org.eclipse.jetty.io.SelectChannelEndPoint$2.run(SelectChannelEndPoint.java:93)
    at org.eclipse.jetty.util.thread.strategy.ExecuteProduceConsume.produceAndRun(ExecuteProduceConsume.java:246)
    at org.eclipse.jetty.util.thread.strategy.ExecuteProduceConsume.run(ExecuteProduceConsume.java:156)
    at org.eclipse.jetty.util.thread.QueuedThreadPool.runJob(QueuedThreadPool.java:654)
    at org.eclipse.jetty.util.thread.QueuedThreadPool$3.run(QueuedThreadPool.java:572)
    at java.lang.Thread.run(Thread.java:748)
Caused by: java.lang.NullPointerException
    at com.goldencode.p2j.util.Utils.getDirectoryNodeMap(Utils.java:2301)
    at com.goldencode.p2j.main.WebHandler.getAvailableThemes(WebHandler.java:1029)
    at com.goldencode.p2j.main.WebHandler.<clinit>(WebHandler.java:150)
    ... 17 more
```

See the NPe in WebHandler.getAvailableThemes - this should not be called for embedded mode auth.

The built-in embedded URL (<https://localhost:7443/embedded>) works fine, only the <https://localhost:8443> has the problem.

#77 - 10/08/2018 04:40 PM - Greg Shah

Sergey Ivanovskiy wrote:

Don't understand, please clarify your question. The web_client.html template has been filled by WebHandler on GET / request and then it has been sent to the client. WebHandler itself doesn't handle requests for different resources.

How would you send the image down? I thought we have no way to expose the URL. At this point: I just want to make sure everything is working.

I see the bitmap in virtual desktop mode. Can we set this so that it is NOT scaled? It looks pretty weird when it is stretched out.

#78 - 10/08/2018 04:47 PM - Sergey Ivanovskiy

Fixing this NPE now and planning to change background-size to auto can help to preserve the original image.

#79 - 10/08/2018 05:31 PM - Sergey Ivanovskiy

Committed revision 11300 (3706a) should fix NPE by doing conditional static initializations for the virtual desktop mode only and added background image to web_client.html template. I checked that these parameters can preserve the logo image

```
<node class="container" name="desktop">
  <node class="string" name="background">
    <node-attribute name="value" value="0x000000"/>
  </node>
  <node class="string" name="background-color">
    <node-attribute name="value" value="#ffffff"/>
  </node>
  <node class="string" name="background-image">
    <node-attribute name="value" value="/com/goldencode/hotel/embedded/images/fwd.png"/>
  </node>
  <node class="string" name="image-mime-type">
    <node-attribute name="value" value="image/png"/>
  </node>
  <node class="string" name="background-origin">
    <node-attribute name="value" value="border-box"/>
  </node>
  <node class="string" name="background-position">
    <node-attribute name="value" value="left top"/>
  </node>
  <node class="string" name="background-size">
    <node-attribute name="value" value="auto"/>
  </node>
  <node class="string" name="background-repeat">
    <node-attribute name="value" value="no-repeat"/>
  </node>
</node>
```

#80 - 10/08/2018 06:14 PM - Sergey Ivanovskiy

Committed revision 11302 (3706a) has minor fix for background-color and committed rev 178 (Hotel GUI) changed directory.xml and its templates.

#81 - 10/08/2018 06:25 PM - Constantin Asofiei

Sergey Ivanovskiy wrote:

Committed revision 11302 (3706a) has minor fix for background-color and committed rev 178 (Hotel GUI) changed directory.xml and its templates.

Why are you not centering the image? It looks weird in the top-left corner...

#82 - 10/08/2018 06:36 PM - Sergey Ivanovskiy

Constantin, please change position if it looks better. It is my subjective judgement that it looks properly with the web client login page.

#83 - 10/08/2018 07:12 PM - Sergey Ivanovskiy

Committed revision 11303 added background-color setting for the virtual desktop.

#84 - 10/15/2018 05:22 PM - Eric Faulhaber

- File *login_with_logo_bad.png* added

Sergey, I'm trying to get the FWD logo centered horizontally on the login page, with a light grey background, maintaining the original aspect ratio of the image, and with a small margin to the left and right. I'd also like to make the opacity < 1.0. I don't know if opacity is a supported setting, but here are the rest of the settings I'm using:

```
<node class="container" name="desktop">
  <node class="string" name="background">
    <node-attribute name="value" value="0x000000"/>
  </node>
  <node class="string" name="background-color">
    <node-attribute name="value" value="0xbbbbbb"/>
  </node>
  <node class="string" name="background-image">
    <node-attribute name="value" value="/com/goldencode/hotel/embedded/images/fwd.png"/>
  </node>
  <node class="string" name="image-mime-type">
    <node-attribute name="value" value="image/png"/>
  </node>
  <node class="string" name="background-origin">
    <node-attribute name="value" value="border-box"/>
  </node>
  <node class="string" name="background-position">
    <node-attribute name="value" value="center"/>
  </node>
  <node class="string" name="background-size">
    <node-attribute name="value" value="80%"/>
  </node>
  <node class="string" name="background-repeat">
    <node-attribute name="value" value="no-repeat"/>
  </node>
</node>
```

However, the image isn't displaying as I've described above. It looks like this:



How would I accomplish my goals as described above?

#85 - 10/16/2018 06:34 AM - Sergey Ivanovskiy

Now the opacity is not supported, but it can be coded directly in web_client.html

```
<!DOCTYPE html>
<html>
  <head>
    <title>Operating System Login (FWD ${client_ui_type} Web Client)</title>
    <style type="text/css" media="screen">
      .login {
        background-image : ${background-image};
        background-repeat : ${background-repeat};
```



```

background-origin : ${background-origin};
background-position : ${background-position};
background-size : ${background-size};
background-color : ${background-color};
opacity : 0.5;
width : 100%;
height : 100%;
}
</style>
.....

```

or we should add new parameter opacity.

In this screen [#2673-84](#) the background image is around the horizontal center and the vertical position of the background image is affected by these styles

```

<body class="login" onload="requestHelper.initialize();">
  <div style="margin-top:150px;">
    <div align="center">
      <form name="loginForm" method="POST" onsubmit="return requestHelper.sendRequest();" >
        <table>
          <tr>
            <td>User Name:</td><td><input type="text" size="20" name="usr" autofocus></td>
          </tr>
          <tr>
            <td>Password:</td><td><input type="password" size="20" name="psw"></td>
          </tr>
          <tr id="UI-Theme-Row">
            <td>UI Theme:</td><td><select name="theme"></select></td>
          </tr>
          <tr>
            <td>&nbsp;</td><td><input type="submit" value="Login"></td>
          </tr>
        </table>
      </form>
    </div>
    <div align="center" style="margin-top:100px">
      <h1 id="error" style="color:red">${error}</h1>
    </div>
  </div>
</body>

```

If we set

```

<node class="string" name="background-position">
  <node-attribute name="value" value="center top"/>
</node>

```

, then the top of the background image will be at the top of the screen. I will try to fix css for this template in order that this image will be aligned correctly.

#86 - 10/16/2018 06:48 AM - Greg Shah

we should add new parameter opacity

Please add this.

A customer should not have to edit the html files to get the result Eric is trying to achieve.

#87 - 10/16/2018 07:54 AM - Sergey Ivanovskiy

Should this opacity style be applied to the login form only? Should it be applied to the virtual desktop too? Now we have common settings for the login desktop and for the virtual desktop.

#88 - 10/16/2018 07:56 AM - Greg Shah

Good point. Please create a separate configuration for login and desktop. The same features should be available on both.

#89 - 10/16/2018 09:40 AM - Sergey Ivanovskiy

Please review these changes committed revision 11319. Now the background image position is not affected by the form css styles. This revision added new login form settings under webClient node

```
<node class="container" name="login-form">
  <node class="string" name="background-color">
    <node-attribute name="value" value="0xbbbbbb"/>
  </node>
  <node class="double" name="background-opacity">
    <node-attribute name="value" value="0.5"/>
  </node>
  <node class="string" name="background-image">
    <node-attribute name="value" value="/com/goldencode/p2j/admin/fwd.png"/>
  </node>
  <node class="string" name="image-mime-type">
    <node-attribute name="value" value="image/png"/>
  </node>
  <node class="string" name="background-origin">
    <node-attribute name="value" value="border-box"/>
  </node>
  <node class="string" name="background-position">
    <node-attribute name="value" value="center"/>
  </node>
  <node class="string" name="background-size">
    <node-attribute name="value" value="auto"/>
  </node>
  <node class="string" name="background-repeat">
    <node-attribute name="value" value="no-repeat"/>
  </node>
</node>
```

and added the background opacity settings to make the background be transparent

```
<node class="double" name="background-opacity">
  <node-attribute name="value" value="0.5"/>
</node>
```

#90 - 10/16/2018 11:02 AM - Eric Faulhaber

This is much better, but it only seems to be applying the opacity setting to the background color, not to the background image. I was hoping the opacity would soften the starkness of the image; the contrast is too harsh, IMHO.

Also, I set background-size to 75% instead of auto.

#91 - 10/16/2018 11:24 AM - Sergey Ivanovskiy

May be this task can be done using images composition but I am not sure

```
background-image: linear-gradient(rgba(0, 0, 255, 0.5), rgba(255, 255, 0, 0.5)),
                  url("../media/examples/lizard.png");
```

Now we can't use this composition of several images. Should it be done?

From <https://developer.mozilla.org/en-US/docs/Web/CSS/background-color>

The background-color property is specified as a single <color> value.

<color>

The uniform color of the background. It is rendered behind any background-image that is specified, although the color will still be visible through any transparency in the image.

#92 - 10/16/2018 11:41 AM - Eric Faulhaber

Let's leave it as is for now. We can change the opacity of the image directly in the PNG.

#93 - 10/16/2018 12:53 PM - Eric Faulhaber

- File *login_new.png* added

- File *desktop_new.png* added

Hotel GUI rev 189 changes the configuration for the login page and desktop to present these screens as follows:

Login:



Desktop:



The directory settings are as follows:

```
<node class="container" name="login-form">
  <node class="string" name="background-color">
    <node-attribute name="value" value="0xbbbbbb"/>
  </node>
  <node class="double" name="background-opacity">
    <node-attribute name="value" value="0.8"/>
  </node>
  <node class="string" name="background-image">
    <node-attribute name="value" value="/com/goldencode/hotel/embedded/images/fwd_opacity_60.png"/>
  </node>
</node>
```

```

<node class="string" name="image-mime-type">
  <node-attribute name="value" value="image/png"/>
</node>
<node class="string" name="background-origin">
  <node-attribute name="value" value="border-box"/>
</node>
<node class="string" name="background-position">
  <node-attribute name="value" value="center"/>
</node>
<node class="string" name="background-size">
  <node-attribute name="value" value="65%"/>
</node>
<node class="string" name="background-repeat">
  <node-attribute name="value" value="no-repeat"/>
</node>
</node>
<node class="container" name="desktop">
  <node class="string" name="background">
    <node-attribute name="value" value="0x000000"/>
  </node>
  <node class="string" name="background-color">
    <node-attribute name="value" value="0xbbbbbb"/>
  </node>
  <node class="string" name="background-image">
    <node-attribute name="value" value="/com/goldencode/hotel/embedded/images/fwd.png"/>
  </node>
  <node class="string" name="image-mime-type">
    <node-attribute name="value" value="image/png"/>
  </node>
  <node class="string" name="background-origin">
    <node-attribute name="value" value="border-box"/>
  </node>
  <node class="string" name="background-position">
    <node-attribute name="value" value="center"/>
  </node>
  <node class="string" name="background-size">
    <node-attribute name="value" value="65%"/>
  </node>
  <node class="string" name="background-repeat">
    <node-attribute name="value" value="no-repeat"/>
  </node>
</node>
</node>

```

Eventually, I think we will want to put the background images somewhere more appropriate, instead of "borrowing" them from the embedded resources, but to keep things simple for now, I followed the lead of the previous settings.

Files

HotelGuiDesktopImage.png	313 KB	10/08/2018	Sergey Ivanovskiy
login_with_logo_bad.png	40.9 KB	10/15/2018	Eric Faulhaber
login_new.png	36.5 KB	10/16/2018	Eric Faulhaber
desktop_new.png	126 KB	10/16/2018	Eric Faulhaber